Logikk Kan Redde Oss Fra Onde Roboter

Selmer Bringsjord et al.

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology
Rensselaer Polytechnic Institute (RPI)
Troy, New York 12180 USA

Intro to Logic
4/19/2018
Logikk Kan Redde Oss Fra Onde Roboter

Selmer Bringsjord et al.

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology
Rensselaer Polytechnic Institute (RPI)
Troy, New York 12180 USA

Intro to Logic
4/19/2018
The PAID Problem
The PAID Problem

∀x : Agents
The PAID Problem

\[ \forall x : \text{Agents} \quad \text{Powerful}(x) + \text{Autonomous}(x) + \text{Intelligent}(x) = \text{Dangerous}(x)/\text{Destroy}_\text{Us} \]
The PAID Problem

\( \forall x : \text{Agents} \)

\( \text{Powerful}(x) + \text{Autonomous}(x) + \text{Intelligent}(x) = \text{Dangerous}(x) / \text{Destroy Us} \)
The PAID Problem

\[ \forall x : \text{Agents} \]

\[ \text{Powerful}(x) + \text{Autonomous}(x) + \text{Intelligent}(x) = \text{Dangerous}(x)/\text{Destroy}_Us \]

\[ u(AIA_i(\pi_j)) > \tau^+ \in \mathbb{Z} \text{ or } \tau^- \in \mathbb{Z} \]
The PAID Problem

∀x : Agents

\text{Powerful}(x) + \text{Autonomous}(x) + \text{Intelligent}(x) = \text{Dangerous}(x)/\text{Destroy Us}
The PAID Problem

\( \forall x : \text{Agents} \)

\[ \text{Powerful}(x) + \text{Autonomous}(x) + \text{Intelligent}(x) = \text{Dangerous}(x) / \text{Destroy Us} \]

\[ u(AIA_i(\pi_j)) > \tau^+ \in \mathbb{Z} \text{ or } \tau^- \in \mathbb{Z} \]
The PAID Problem

∀x: Agents

\text{Powerful}(x) + \text{Autonomous}(x) + \text{Intelligent}(x) = \text{Dangerous}(x)/\text{Destroy}_\text{Us}

u(AIA_i(\pi_j)) > \tau^+ \in \mathbb{Z} \text{ or } \tau^- \in \mathbb{Z}

\textbf{Theorem ACU:} In a collaborative situation involving agents \(a\) (as the “trustor”) and \(a'\) (as the “trustee”), if \(a'\) is at once both autonomous and ToM-creative, \(a'\) is untrustworthy from an ideal-observer \(o\)'s viewpoint, with respect to the action-goal pair \(\langle \alpha, \gamma \rangle\) in question.

\textbf{Proof:} Let \(a\) and \(a'\) be agents satisfying the hypothesis of the theorem in an arbitrary collaborative situation. Then, by definition, \(a \neq a'\) desires to obtain some goal \(\gamma\) in part by way of a contributed action \(\alpha_k\) from \(a'\), \(a'\) knows this, and moreover \(a'\) knows that \(a\) believes that this contribution will succeed. Since \(a'\) is by supposition ToM-creative, \(a'\) may desire to surprise \(a\) with respect to \(a\)'s belief regarding \(a'\)'s contribution; and because \(a'\) is autonomous, attempts to ascertain whether such surprise will come to pass are fruitless since what will happen is locked inaccessibly in the oracle that decides the case. Hence it follows by TRANS that an ideal observer \(o\) will regard \(a'\) to be untrustworthy with respect to the pair \(\langle \alpha, \gamma \rangle\) pair. \textbf{QED}
“We’re in very deep trouble.”
“We’re in very deep trouble.”
“We’re in very deep trouble.”
Unfortunately, not quite as easy as this to use logic to save the day ...
Logic Thwarts Landru!

First Suspicion That It’s a Mere Computer Running the Show
Logic Thwarts Landru!

Landru is Indeed Merely a Computer
(the real Landru having done the programming)
Logic Thwarts Landru!

Landru Kills Himself Because Kirk/Spock Argue He Has Violated the Prime Directive for Good by Denying Creativity to Others
Logic Thwarts Nomad!
(with the Liar Paradox)
I.
Cognitive Calculi ...
Hierarchy of Ethical Reasoning

\[ DCEC_{CL}^* \]
\[ DCEC^* \]
\[ ADR^M \]
\[ U \]

DIARC

UIMA/Watson-inspired
Hierarchy of Ethical Reasoning

\[
DCEC^{*}_{CL} \\
DCEC^{*} \\
ADR^{M} \\
U
\]

DIARC

UIMA/Watson-inspired
Hierarchy of Ethical Reasoning

UIMA/Watson-inspired
Hierarchy of Ethical Reasoning

Not paradox-prone deontic logics!

\[ \mathcal{DCEC}^*_C L \]

\[ \mathcal{DCEC}^* \]

\[ \mathcal{ADR}^M \]

\[ U \]

DIARC

UIMA/Watson-inspired
“Universal Cognitive Calculus”

Leibniz

1.5 centuries < Boole!
2.5 centuries < Kripke

Logic Theorist
(birth of modern logicist AI)

1666

1956

2017

DCEC*

RAIR
Rensselear AI and Reasoning Lab

AI of Today: What Would Leibniz Say?
“Sorry, not impressed.”
Samuel Brungard

This stems from the fact that theorem proving in just first-order logic is
Turing-level computation; see e.g. (Boolos, Burgess
than the material conditional. (Reliance on conditional branching in standar
analysis, sophisticated moral reasoning can only be accurately modeled for
from formal logic. No matter what the underlying implementation of
neural networks, statistical AI) that at the surface level seem far away
for the robot. Modules correspond to functionality that can be added

Cognitive Calculi

purely extensional level:

FOL MSL SOL TOL IFOL ...

theories: PA ZFC axiomatic physics ...

intensional level:

epistemic deontic possibility/necessity ...

model finders: MACE ...

ATPs:

SPASS SNARK ShadowProver ...

nature of representation: symbolic or homomorphic:

-----------------------------
Cognitive Calculi

purely extensional level:

FOL, MSL, SOL, TOL, IFOL, ...

intensional level:

epistemic, deontic, possibility/necessity, ...

ATPs:

SPASS, SNARK, ShadowProver, ...

theories: PA, ZFC, axiomatic physics, ...

model finders: MACE, ...

nature of representation: symbolic or homomorphic: ...
Cognitive Calculi

purely extensional level:
FOL MSL SOL TOL IFOL …

intensional level:
epistemic deontic possibility/necessity …

ATPs:
SPASS SNARK ShadowProver …

theories: PA ZFC axiomatic physics …

model finders: MACE …
nature of representation: symbolic or homomorphic:

-------------------------------
Cognitive Calculi

purely extensional level:
- FOL
- MSL
- SOL
- TOL
- IFOL
- ... 

theories:
- PA
- ZFC
- axiomatic physics
- ...

model finders:
- MACE
- ...

intensional level:
- epistemic
- deontic
- possibility/necessity
- ...

nature of representation:
- symbolic or homomorphic:
- ...

ATPs:
- SPASS
- SNARK
- ShadowProver
- ...

\[ \lambda \text{-calculus} \]
Cognitive Calculi

purely extensional level:

FOL MSL SOL TOL IFOL ... theories: PA ZFC axiomatic physics ...

intensional level:

epistemic deontic possibility/necessity ...

inference schemas

ATPs: SPASS SNARK ShadowProver ...

model finders: MACE ...

nature of representation: symbolic or homomorphic: ...

λ-calculus

λ-calculus

... analogical reasoning

... inductive reasoning

... inference schemas ∞
Cognitive Calculi

purely extensional level:
- FOL
- MSL
- SOL
- TOL
- IFOL

intensional level:
- epistemic
- deontic
- possibility/necessity

theories:
- PA
- ZFC
- axiomatic physics

model finders:
- MACE

nature of representation:
- symbolic or homomorphic:

ATPs:
- SPASS
- SNARK
- ShadowProver

\( \lambda \)-calculus

\( \lambda \)-calculus

\( \ldots \)

DyCEC*

DCEC*

DCSC*

CEC

CSC

\( \lambda \)-calculus

\( \lambda \)-calculus

\( \ldots \)

\( \ldots \)

analogical reasoning

inductive reasoning

inference schemas

\( \infty \)
Cognitive Calculi

purely extensional level:
FOL, MSL, SOL, TOL, IFOL, ...

intensional level:
epistemic, deontic, possibility/necessity, ...

model finders:
ZFC, axiomatic physics, ...

model finders:
MACE, ...

nature of representation:
symbolic or homomorphic:

λ-calculus

... DyCEC*, DCEC*, DCSC*, CEC, CSC, ...

... analogical reasoning

... inductive reasoning

inference schemas

...
Cognitive Calculi

purely extensional level: FOL MSL SOL TOL IFOL ...

intensional level: epistemic deontic possibility/necessity ...

DCEC* DCSC* CEC CSC ...

λ-calculus

dialects:

model finders: MACE ...

nature of representation: symbolic or homomorphic:

theories: PA ZFC axiomatic physics ...

ATPs: SPASS SNARK ShadowProver ...

analogical reasoning

inference schemas ∞

inductive reasoning
Formal Syntax
Formal Syntax

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

\[ \text{action} : \text{Agent} \times \text{ActionType} \rightarrow \text{Action} \]
\[ \text{initially} : \text{Fluent} \rightarrow \text{Boolean} \]
\[ \text{holds} : \text{Fluent} \times \text{Moment} \rightarrow \text{Boolean} \]
\[ \text{happens} : \text{Event} \times \text{Moment} \rightarrow \text{Boolean} \]
\[ \text{clipped} : \text{Moment} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Boolean} \]

\[ f ::= \text{initiates} : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Boolean} \]
\[ \text{terminates} : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Boolean} \]
\[ \text{prior} : \text{Moment} \times \text{Moment} \rightarrow \text{Boolean} \]
\[ \text{interval} : \text{Moment} \times \text{Boolean} \]
\[ \star : \text{Agent} \rightarrow \text{Self} \]
\[ \text{payoff} : \text{Agent} \times \text{ActionType} \times \text{Moment} \rightarrow \text{Numeric} \]

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

\[ t : \text{Boolean} \mid \lnot \phi \mid \phi \land \psi \mid \phi \lor \psi \mid \]
\[ \text{P}(a,t,\phi) \mid \text{K}(a,t,\phi) \mid \text{C}(t,\phi) \mid \text{S}(a,b,t,\phi) \mid \text{S}(a,t,\phi) \]
\[ \phi ::= \]
\[ \text{B}(a,t,\phi) \mid \text{D}(a,t,\text{holds}(f,t')) \mid \text{I}(a,t,\text{happens}(\text{action}(a^*,\alpha),t')) \]
\[ \text{O}(a,t,\phi,\text{happens}(\text{action}(a^*,\alpha),t')) \]
Inference Schemata
Inference Schemata

\[ C(t, P(a, t, \phi) \rightarrow K(a, t, \phi)) \]  \[ R_1 \]

\[ C(t, K(a, t, \phi) \rightarrow B(a, t, \phi)) \]  \[ R_2 \]

\[ C(t, \phi) \quad t \leq t_1 \ldots \leq t_n \]

\[ K(a_1, t_1, \ldots, K(a_n, t_n, \phi)) \]  \[ R_3 \]

\[ K(a, t, \phi) \]  \[ R_4 \]

\[ C(t, K(a, t_1, \phi_1 \rightarrow \phi_2) \rightarrow K(a, t_2, \phi_1) \rightarrow K(a, t_3, \phi_2) \]  \[ R_5 \]

\[ C(t, B(a, t_1, \phi_1 \rightarrow \phi_2) \rightarrow B(a, t_2, \phi_1) \rightarrow B(a, t_3, \phi_2) \]  \[ R_6 \]

\[ C(t, C(t_1, \phi_1 \rightarrow \phi_2)) \rightarrow C(t_2, \phi_1) \rightarrow C(t_3, \phi_2) \]  \[ R_7 \]

\[ C(t, \forall x. \phi \rightarrow \phi[x \mapsto t]) \]  \[ R_8 \]

\[ C(t, \phi \rightarrow \phi \rightarrow \phi_1) \]  \[ R_9 \]

\[ C(t, [\phi_1 \land \ldots \land \phi_n \rightarrow \phi] \rightarrow [\phi_1 \rightarrow \ldots \rightarrow \phi_n 

\rightarrow \psi]) \]  \[ R_{10} \]

\[ B(a, t, \phi) \quad \phi \rightarrow \psi \]  \[ R_{11a} \]

\[ B(a, t, \psi) \]

\[ B(a, t, \phi) \]  \[ R_{11b} \]

\[ B(h, t, B(s, t, \phi)) \]  \[ R_{12} \]

\[ I(a, t, happens(action(a^*, \alpha), t')) \]  \[ R_{13} \]

\[ P(a, t, happens(action(a^*, \alpha), t)) \]

\[ B(a, t, \phi) \quad B(a, t, O(a^*, t, \phi, happens(action(a^*, \alpha), t')))) \]

\[ B(a, t, O(a^*, t, \phi, happens(action(a^*, \alpha), t')))) \]

\[ O(a, t, \phi, happens(action(a^*, \alpha), t')) \]

\[ K(a, t, I(a^*, t, happens(action(a^*, \alpha), t')))) \]

\[ \phi \leftrightarrow \psi \]

\[ O(a, t, \phi, \gamma) \leftrightarrow O(a, t, \psi, \gamma) \]  \[ R_{15} \]
Event Calculus for Time & Change

\[
\begin{align*}
C(t, P(a, t, \phi) \Rightarrow K(a, t, \phi)) & \quad [R_1] & C(t, K(a, t, \phi) \Rightarrow B(a, t, \phi)) & \quad [R_2] \\
C(t, \phi) & \Rightarrow t_1 \leq t \leq t_n & K(a_1, t_1, \ldots, K(a_n, t_n, \phi), \ldots) & \quad [R_3] & \begin{array}{c}
K(a, t, \phi) = \\
\phi
\end{array} & \quad [R_4] \\
C(t, K(a, t, \phi_1) \Rightarrow K(a, t, \phi_2)) & \Rightarrow C(t, K(a, t, \phi_1) \Rightarrow K(a, t, \phi_2)) & \quad [R_5] \\
C(t, B(a, t, \phi_1) \Rightarrow B(a, t, \phi_2)) & \Rightarrow C(t, B(a, t, \phi_1) \Rightarrow B(a, t, \phi_2)) & \quad [R_6] \\
C(t, C(t_1, \phi_1) \Rightarrow C(t_2, \phi_1)) & \Rightarrow C(t, C(t_1, \phi_1) \Rightarrow C(t_2, \phi_2)) & \quad [R_7] \\
C(t, \forall x. \phi(x \Rightarrow \psi)) & \Rightarrow C(t, \forall x. \phi(x \Rightarrow \psi)) & \quad [R_8] \\
C(t, \forall x. \phi(x \Rightarrow \psi)) & \Rightarrow C(t, \forall x. \phi(x \Rightarrow \psi)) & \quad [R_9] \\
B(a, t, \phi) & \Rightarrow \psi & B(a, t, \phi) \Rightarrow B(a, t, \psi) & \quad [R_{11a}] & B(a, t, \psi) \Rightarrow B(a, t, \psi \land \phi) & \quad [R_{11b}] \\
S(a, h, \tau, \phi) & \quad [R_{12}] \\
I(a, t, \text{happens}(\text{action}(a^n, \alpha), t')) & \quad [R_{13}] \\
P(a, t, \text{happens}(\text{action}(a^n, \alpha), t)) & \quad [R_{14}] \\
B(a, t, \phi) & \Rightarrow B(a, t, \text{happens}(\text{action}(a^n, \alpha), t)) & \quad [R_{15}] \\
O(a, t, \phi, \text{happens}(\text{action}(a^n, \alpha), t')) & \quad [R_{16}] \\
K(a, t, I(a^n, \phi, \text{happens}(\text{action}(a^n, \alpha), t'))) & \quad [R_{17}] \\
\phi & \Rightarrow \psi & O(a, t, \phi) & \Rightarrow O(a, t, \psi) & \quad [R_{18}]
\end{align*}
\]
Event Calculus for Time & Change

[A1] C(∀ f, t. initially(f) ∧ ¬clipped(0, f, t) ⇒ holds(f, t))

[A2] C(∀ e, f, t1, t2. happens(e, t1) ∧ initiates(e, f, t1) ∧ t1 < t2 ∧ ¬clipped(t1, f, t2) ⇒ holds(f, t2))

[A3] C(∀ t1, f, t2. clipped(t1, f, t2) ⇔ [∃ e, t. happens(e, t) ∧ t1 < t < t2 ∧ terminates(e, f, t)])

[A4] C(∀ a, d, t. happens(action(a, d), t) ⇒ K(a, happens(action(a, d), t)))

[A5] C(∀ a, f, t, t’. B(a, holds(f, t)) ∧ B(a, t < t’) ∧ ¬B(a, clipped(t, f, t’)) ⇒ B(a, holds(f, t’)))
1. **Joy**: pleased about a desirable event. By 'pleased about a desirable event' the meaning we will consider is 'pleased about a desirable consequence of the event'.

   \[\text{forSome } c \ B(a, t_3, \text{implies}(\text{happens}(e, t_1), \text{holds}(\text{CON}(e, a, c), t_2)))\]

   \[D(a, t_3, \text{holds}(\text{CON}(e, a, c), t_2))\]

   \[K(a, t_3, \text{happens}(e, t_1))\]

   The definition of \(\text{holds}(\text{AFF}(a, \text{joy}), t_3)\) is therefore and(1,2,3).

2. **Distress**: displeased about an undesirable event.

   \[\text{not}(D(a, t_3, \text{holds}(\text{CON}(e, a, c), t_3)))\]

   The definition of \(\text{holds}(\text{AFF}(a, \text{distress}), t_3)\) is therefore and(1,4,3).

3. **Happy-for**: pleased about an event presumed to be desirable for someone else.

   \[\text{forSome } c \ B(a, t_3, \text{implies}(\text{happens}(e, t_1), \text{holds}(\text{CON}(e, a_1, c), t_2)))\]

   \[B(a, t_3, D(a_1, t_3, \text{holds}(\text{CON}(e, a_1, c), t_2)))\]

   \[D(a, t_3, \text{holds}(\text{CON}(e, a_1, c), t_2))\]

   The definition of \(\text{holds}(\text{AFF}(a, \text{happy-for}), t_3)\) is therefore and(5,6,7,3).

4. **Pity**: displeased about an event presumed to be undesirable for someone else. This is equivalent to sorry_for in Hobbes-Gordon model.

   \[B(a, t_3, \text{not}(D(a_1, t_3, \text{holds}(\text{CON}(e, a_1, c), t_2))))\]

   \[\text{not}(D(a, t_3, \text{holds}(\text{CON}(e, a_1, c), t_2)))\]

   The definition of \(\text{holds}(\text{AFF}(a, \text{pity}), t_3)\) is therefore and(5,8,9,3).

5. **Gloating**: pleased about an event presumed to be undesirable for someone else. The definition of \(\text{holds}(\text{AFF}(a, \text{gloating}), t_3)\) is therefore and(5,8,7,3).

6. **Resentment**: displeased about an event presumed to be desirable for someone else. The definition of \(\text{holds}(\text{AFF}(a, \text{resentment}), t_3)\) is therefore and(5,6,9,3).

7. **Hope**: (pleased about) the prospect of a desirable event

   \[\text{forSome } c \ B(a, t_0, \text{implies}(\text{happens}(e, t_1), \text{holds}(\text{CON}(e, a, c), t_2)))\]

   \[D(a, t_0, \text{holds}(\text{CON}(e, a, c), t_2))\]

   The definition of \(\text{holds}(\text{AFF}(a, \text{hope}), t_0)\) is therefore and(10,11).

8. **Fear**: (displeased about) the prospect of an undesirable event

   \[\text{not}(D(a, t_0, \text{holds}(\text{CON}(e, a, c), t_2)))\]

   The definition of \(\text{holds}(\text{AFF}(a, \text{fear}), t_0)\) is therefore and(10,12).

9. **Satisfaction**: (pleased about) the confirmation of the prospect of a desirable event. The definition of \(\text{holds}(\text{AFF}(a, \text{satisfaction}), t_3)\) is and(10,11,7,3).

10. **Fears-confirmed**: (displeased about) the confirmation of the prospect of an undesirable event. The definition of \(\text{holds}(\text{AFF}(a, \text{fears-confirmed}), t_3)\) is and(10,12,9,3).

11. **Relief**: (pleased about) the disconfirmation of the prospect of an undesirable event

    \[K(a, t_3, \text{not}(\text{happens}(e, t_1)))\]

    The definition of \(\text{holds}(\text{AFF}(a, \text{relief}), t_3)\) is and(10,12,9,13).

12. **Disappointment**: (displeased about) the disconfirmation of the prospect of a desirable event. The definition of \(\text{holds}(\text{AFF}(a, \text{disappointment}), t_3)\) is and(10,11,7,13).

13. **Praise** : (approving of) one’s own praiseworthy action. Here we treat ‘approve’ as an action event. We also introduce a new predicate \(\text{PRAISEWORTHY}(a, b, x)\) which will mean that agent a considers x a praiseworthy action by agent b. All the 3 interpretations are shown below.

    \[\text{happens}(\text{action}(a, x), t_0)\]

    \[\text{forAll } a, B(a, t_3, \text{implies}(\text{happens}(\text{action}(a, x), t_1), \text{PRAISEWORTHY}(a, a, x)), t_3 \leq t_1)\]

    \[D(a, t_1, \text{holds}(\text{PRAISEWORTHY}(a, a, x), t_1))\]

    \[\text{happens}(\text{action}(a, \text{approve}(x)), t_1)\]

    \[\text{forAll } a, B(a, t_3, \text{holds}(\text{PRAISEWORTHY}(a, a, x), t_1)), t_3 \leq t_1\]

14. **Shame** : (disapproving of) one’s own blameworthy action. This also follows the same explanation as Praise.

    \[\text{forAll } a, B(a, t_3, \text{implies}(\text{happens}(\text{action}(a, x), t_1), B(a, t_1, \text{holds}(\text{BLAMEWORTHY}(a, a, x), t_1))), t_3 \leq t_1\]

    \[\text{not}(\text{happens}(\text{action}(a, \text{approve}(x)), t_1))\]

    \[\text{forAll } a, B(a, t_1, \text{holds}(\text{BLAMEWORTHY}(a, a, x), t_1)), t_1 \leq t_1\]

15. **Admiration** : (approving of) someone else’s praiseworthy action

    \[\text{happens}(\text{action}(a_1, x), t_0)\]

    \[\text{forAll } a, B(a, t_3, \text{holds}(\text{PRAISEWORTHY}(a, a_1, x), t_1)), t_3 \leq t_1\]

16. **Reproach** : (disapproving of) someone else’s blameworthy action. The definition of \(\text{holds}(\text{AFF}(a, \text{reproach}), t_1)\) is and(20, B(a, t_1, \text{holds}(\text{BLAMEWORTHY}(a, a_1, x), t_1)), t_1).

17. **Gratification** : (approving of) one’s own praiseworthy action and (being pleased about) the related desirable event. We again interpret ‘pleased about the desirable event’ as ‘pleased about the desired consequence of the event.’

    \[\text{forSome } c \ B(a, t_3, \text{implies}(\text{happens}(\text{action}(a, x), t_0), \text{holds}(\text{CON}(\text{action}(a, x, a, c), t_0))))\]

    \[D(a, t_1, \text{holds}(\text{CON}(\text{action}(a, x, a, c), t_0))\]

The definition of \(\text{holds}(\text{AFF}(a, \text{gratification}), t_1)\) is and(20, B(a, t_1, \text{holds}(\text{PRAISEWORTHY}(a, a, x), t_1)), t_1).
II.
Early Progress With Our Calculi:
Simple Dilemmas;
Non-Akratic Robots
Ethical robots save humans
Ethical robots save humans
Informal Definition of Akrasia
Informal Definition of Akrasia
Informal Definition of Akrasia
Informal Definition of Akrasia

A \textit{agent}
Informal Definition of Akrasia
Informal Definition of Akrasia
Informal Definition of Akrasia
Informal Definition of Akrasia
Informal Definition of Akrasia

agent

\( \alpha_f \)

t\(_{\alpha_f}\)

t\(_{\alpha_o}\)

desired
Informal Definition of Akrasia
Informal Definition of Akrasia

An agent $A$ desires an action $\alpha_f$ at time $t_{\alpha_f}$, which is obligatory at time $t_{\alpha_o}$. The diagram illustrates the distinctions between desired and obligatory actions.
Informal Definition of Akrasia

If $\alpha_f$ happens, then $\alpha_o$ can’t happen
Informal Definition of Akrasia

If $\alpha_f$ happens, then $\alpha_o$ can’t happen
Informal Definition of Akrasia

If $\alpha_f$ happens, then $\alpha_o$ can’t happen

Agent knows this
Informal Definition of Akrasia
Informal Definition of Akrasia
Informal Definition of Akrasia
Informal Definition of Akrasia
Informal Definition of Akrasia
Informal Definition of Akrasia

Desire to do $\alpha_f$

$t_{\alpha_f}$
Informal Definition of Akrasia

Desire to do $\alpha_f \succ t_{\alpha_f}$
Informal Definition of Akrasia

Desire to do $\alpha_f$ $\Rightarrow$ Belief that he ought to do $\alpha_o$

$t_{\alpha_f}$
Informal Definition of Akrasia

Desire to do $\alpha_f$ $\Rightarrow$ Belief that he ought to do $\alpha_o$

$A$ does $\alpha_f$ due to his desire

$t_{\alpha_f}$
Informal Definition of Akrasia

- **Desire to do** $\alpha_f$
- **Belief that he ought to do** $\alpha_o$
- **A** does $\alpha_f$ due to his desire

$t_{\alpha_f}$
Informal Definition of Akrasia

- Desire to do $\alpha_f$ 
- Belief that he ought to do $\alpha_o$

A does $\alpha_f$ due to his desire

$t_{\alpha_f}$
Informal Definition of Akrasia

- Desire to do $\alpha_f$ 
- Belief that he ought to do $\alpha_o$
- A does $\alpha_f$ due to his desire

$t_{\alpha_f}$ $t$
Informal Definition of Akrasia

Desire to do ($\alpha_f$)  $\Rightarrow$  Belief that he ought to do ($\alpha_o$)

$A$ does ($\alpha_f$) due to his desire

$A$ believes he should have done ($\alpha_o$)

$t_{\alpha_f}$  $\Rightarrow$  $t$
Informal Definition of Akrasia

An action $\alpha_f$ is (Augustinian) akratic for an agent $A$ at $t_{\alpha_f}$ iff the following eight conditions hold:

1. $A$ believes that $A$ ought to do $\alpha_o$ at $t_{\alpha_o}$;
2. $A$ desires to do $\alpha_f$ at $t_{\alpha_f}$;
3. $A$’s doing $\alpha_f$ at $t_{\alpha_f}$ entails his not doing $\alpha_o$ at $t_{\alpha_o}$;
4. $A$ knows that doing $\alpha_f$ at $t_{\alpha_f}$ entails his not doing $\alpha_o$ at $t_{\alpha_o}$;
5. At the time ($t_{\alpha_f}$) of doing the forbidden $\alpha_f$, $A$’s desire to do $\alpha_f$ overrides $A$’s belief that he ought to do $\alpha_o$ at $t_{\alpha_f}$.
6. $A$ does the forbidden action $\alpha_f$ at $t_{\alpha_f}$;
7. $A$’s doing $\alpha_f$ results from $A$’s desire to do $\alpha_f$;
8. At some time $t$ after $t_{\alpha_f}$, $A$ has the belief that $A$ ought to have done $\alpha_o$ rather than $\alpha_f$. 
Informal Definition of Akrasia

An action $\alpha_f$ is (Augustinian) akratic for an agent $A$ at $t_{\alpha_f}$ iff the following eight conditions hold:

1. $A$ believes that $A$ ought to do $\alpha_o$ at $t_{\alpha_o}$;
2. $A$ desires to do $\alpha_f$ at $t_{\alpha_f}$;
3. $A$’s doing $\alpha_f$ at $t_{\alpha_f}$ entails his not doing $\alpha_o$ at $t_{\alpha_o}$;
4. $A$ knows that doing $\alpha_f$ at $t_{\alpha_f}$ entails his not doing $\alpha_o$ at $t_{\alpha_o}$;
5. At the time ($t_{\alpha_f}$) of doing the forbidden $\alpha_f$, $A$’s desire to do $\alpha_f$ overrides $A$’s belief that he ought to do $\alpha_o$ at $t_{\alpha_f}$.
6. $A$ does the forbidden action $\alpha_f$ at $t_{\alpha_f}$;
7. $A$’s doing $\alpha_f$ results from $A$’s desire to do $\alpha_f$;
8. At some time $t$ after $t_{\alpha_f}$, $A$ has the belief that $A$ ought to have done $\alpha_o$ rather than $\alpha_f$. 
Informal Definition of Akrasia

An action $\alpha_f$ is (Augustinian) akratic for an agent $A$ at $t_{\alpha_f}$ iff the following eight conditions hold:

1. $A$ believes that $A$ ought to do $\alpha_o$ at $t_{\alpha_o}$;
2. $A$ desires to do $\alpha_f$ at $t_{\alpha_f}$;
3. $A$’s doing $\alpha_f$ at $t_{\alpha_f}$ entails his not doing $\alpha_o$ at $t_{\alpha_o}$;
4. $A$ knows that doing $\alpha_f$ at $t_{\alpha_f}$ entails his not doing $\alpha_o$ at $t_{\alpha_o}$;
5. At the time ($t_{\alpha_f}$) of doing the forbidden $\alpha_f$, $A$’s desire to do $\alpha_f$ overrides $A$’s belief that he ought to do $\alpha_o$ at $t_{\alpha_f}$.
6. $A$ does the forbidden action $\alpha_f$ at $t_{\alpha_f}$;
7. $A$’s doing $\alpha_f$ results from $A$’s desire to do $\alpha_f$;
8. “Regret” $A$ has the belief that $A$ ought to have done $\alpha_o$ rather than $\alpha_f$. 

“Regret”
Cast in

$DCEC^*$

this becomes ...
The forbidden action is obtained only if a direct order is overridden. In terms of the informal conditions of revenge, we thus find it useful to deal herein with a DCEC definition given above, and that's cast in the language of conditions. The following statement then holds:

\[ \text{KB}_{rs} \cup \text{KB}_{m1} \cup \text{KB}_{m2} \ldots \text{KB}_{mn} \vdash \]

\[ D_1 : \text{B}(l, \text{now}, \text{O}(l^*, t_\alpha, \Phi, \text{happens}(\text{action}(l^*, \alpha), t_\alpha))) \]

\[ D_2 : \text{D}(l, \text{now}, \text{holds}(\text{does}(l^*, \overline{\alpha}), t_\overline{\alpha})) \]

\[ D_3 : \text{happens}(\text{action}(l^*, \overline{\alpha}), t_\overline{\alpha}) \Rightarrow \neg \text{happens}(\text{action}(l^*, \alpha), t_\alpha) \]

\[ D_4 : \text{K}(l, \text{now}, (\neg \text{happens}(\text{action}(l^*, \alpha), t_\alpha) \Rightarrow \neg \text{happens}(\text{action}(l^*, \alpha), t_\alpha))) \]

\[ D_5 : I(l, t_\alpha, \text{happens}(\text{action}(l^*, \overline{\alpha}), t_\overline{\alpha}) \land \neg I(l, t_\alpha, \text{happens}(\text{action}(l^*, \alpha), t_\alpha)) \]

\[ D_6 : \text{happens}(\text{action}(l^*, \overline{\alpha}), t_\overline{\alpha}) \]

\[ \Gamma \cup \{ \text{D}(l, \text{now}, \text{holds}(\text{does}(l^*, \overline{\alpha}), t)) \} \vdash \]

\[ D_{7a} : \text{happens}(\text{action}(l^*, \overline{\alpha}), t_\alpha) \]

\[ \Gamma - \{ \text{D}(l, \text{now}, \text{holds}(\text{does}(l^*, \overline{\alpha}), t)) \} \not\vdash \]

\[ D_{7b} : \text{happens}(\text{action}(l^*, \overline{\alpha}), t_\alpha) \]

\[ D_8 : \text{B}(l, t_f, \text{O}(l^*, t_\alpha, \Phi, \text{happens}(\text{action}(l^*, \alpha), t_\alpha))) \]
Demos ...
Demos ...
But, a twist befell the logicists …
Chisholm had argued that the three old 19th-century ethical categories (forbidden, morally neutral, obligatory) are not enough — and soul-searching brought me to agreement.
Leibnizian Ethical Hierarchy for Persons and Robots:

deviltry  uncivil  forbidden  morally neutral  obligatory  civil  heroic
Leibnizian Ethical Hierarchy for Persons and Robots:

\[ EH \]

- deviltry
- uncivil
- forbidden
- morally neutral
- obligatory
- civil
- heroic
- the supererogatory
Leibnizian Ethical Hierarchy for Persons and Robots:

EH

deviltry  uncivil  forbidden  morally neutral  obligatory  civil  heroic  

the supererogatory
Leibnizian Ethical Hierarchy for Persons and Robots:

- Deviltry
- Uncivil
- Forbidden
- Morally Neutral
- Obligatory
- Civil
- Heroic
Leibnizian Ethical Hierarchy for Persons and Robots:

- Deviltry
- Uncivil
- Forbidden
- Morally neutral
- Obligatory
- Civil
- Heroic

(see Norwegian crime fiction)
Leibnizian Ethical Hierarchy for Persons and Robots: 

EH

19th-Century Triad

(see Norwegian crime fiction) the supererogatory

deviltry uncivil forbidden morally neutral obligatory
civil heroic

the supererogatory
Leibnizian Ethical Hierarchy for Persons and Robots:

- Deviltry
- Uncivil
- Forbidden
- Morally Neutral
- Obligatory
- Civil
- Heroic

(see Norwegian crime fiction)

The subererogatory

The supererogatory
Leibnizian Ethical Hierarchy for Persons and Robots:

(see Norwegian crime fiction)

the subererogatory

deviltry, uncivil, forbidden, morally neutral, obligatory, civil, heroic

the supererogatory
Leibnizian Ethical Hierarchy for Persons and Robots:

The hierarchy includes:
- Deviltry
- Uncivil
- Forbidden
- Morally Neutral
- Obligatory
- Civil
- Heroic

The suberogatory and the supererogatory focus on the interests of others, as seen in Norwegian crime fiction.
Leibnizian Ethical Hierarchy for Persons and Robots:

EH

(see Norwegian crime fiction)

the subererogatory

deviltry uncivil forbidden morally neutral obligatory civil heroic

the supererogatory

But this portion may be most relevant to military missions.

focus of others
\mathcal{I} := \|\mathcal{F}_{\mathcal{P}} \wedge \neg \mathcal{O}_{\mathcal{O}}\| \quad 19\text{th Century Triad}
$\mathcal{I} := \|F|P \land \neg O|O\| \quad \text{19th Century Triad}$
\[ \mathcal{I} := \| \mathcal{F} \| \mathcal{P} \wedge \neg \mathcal{O} \| \mathcal{O} \| \quad 19th \text{ Century Triad} \]

\[
\begin{array}{cccc}
\mathcal{F} & \mathcal{P} & \neg \mathcal{O} & \mathcal{O} \\
\forall \ F \ M \ V \ \exists & \mathcal{P} \wedge \neg \mathcal{O} & \forall \ F \ M \ V \ \exists
\end{array}
\]
$\mathcal{I} := \| \mathcal{F} \mathcal{P} \wedge \neg \mathcal{O} \mathcal{O} \| \quad 19\text{th Century Triad}$

\[\begin{array}{cc|ccc|ccc|ccc|ccc|ccc|ccc|ccc}
\mathcal{F} & \mathcal{P} \wedge \neg \mathcal{O} & \mathcal{O} \\
\forall & F & M & V & \exists & \forall & F & M & V & \exists \\
\end{array}\]

$\mathcal{E}$

\[\begin{array}{l|l|c|c|c|c|c|c}
\text{sub}_1 & \text{sub}_2 & \mathcal{F} & \mathcal{P} \wedge \neg \mathcal{O} & \mathcal{O}^L & \mathcal{O}^M & \text{sup}_1 & \text{sup}_2 \\
\forall \in & \forall \in & \forall \in & \forall \in & \forall \in & \forall \in & \forall \in & \forall \in & \uparrow \\
\end{array}\]
$I \ := \ \|F|P \land \neg O|O\| \quad 19\text{th} \text{ Century Triad}$

$F \quad P \land \neg O \quad O$
$\forall \quad F \quad M \quad V \quad \exists$

$\forall \quad F \quad M \quad V \quad \exists$

$\mathcal{EH}$

$S_{ub1} \quad S_{ub2} \quad F \quad P \land \neg O \quad O^L \quad O^M \quad S^{up1} \quad S^{up2}$
$\forall \quad \exists \quad \forall \quad \exists \quad \forall \quad \exists \quad \forall \quad \exists \quad \forall \quad \exists$

$\bullet$
$I := \|F|P \land \neg O|O\|$ 19th Century Triad

$\forall F \in M \in V \in P \land \neg O \in O$

$\exists E H$

$S_{ub1} \mid S_{ub2} \mid F \mid P \land \neg O \mid O^L \mid C^M \mid S_{up1} \mid S_{up2}$

Arkin
Pereira
Andersons
Powers
Mikhail

...
\[ \mathcal{I} := \|F|P \land \neg O|O\| \quad \text{19th Century Triad} \]

\[ \begin{array}{cc|ccc|c|ccc|c}
\forall & F & M & V & E & \forall & F & M & V & E \\
\end{array} \]

\[ \mathcal{EH} \]

\[ \begin{array}{ccccccc}
S_{ub1} & S_{ub2} & F & P \land \neg O & O^L & O^M & S_{up1} & S_{up2} \\
\forall - E & \forall - E & \forall - E & \forall - E & \forall - E & \forall - E & \forall - E & \forall - E \\
\end{array} \]
\[ I := \| \mathcal{F} | \mathcal{P} \land \neg \mathcal{O} | \mathcal{O} \| \quad \text{19th Century Triad} \]

\[ \forall \mathcal{F} | \mathcal{P} | \mathcal{O} \]

\[ E \mathcal{H} \]

\[ S_{ub1} | S_{ub2} \]

\[ \Lambda^E \Lambda^E \]

\[ O^L | O^M | S^{up1} | S^{up2} \]

\[ \Lambda^E \Lambda^E \Lambda^E \Lambda^E \]

Rensselaer AI and Reasoning Lab
$T := \|F|P \land \neg O|O\|$ 19th Century Triad

There are obviously a host of formulae whose theoremhood constitute desiderata; that is (to give but a pair), the following must be provable (where $n \in \{1, 2\}$):

**Theorem 1.** $S_{\text{up}^n}(\phi, a, \alpha) \rightarrow \neg O(\phi, a, \alpha)$

**Theorem 2.** $S_{\text{up}^n}(\phi, a, \alpha) \rightarrow \neg F(\phi, a, \alpha)$

Secondly, $L_{EH}$ is an inductive logic, not a deductive one. This must be the case, since, as we’ve noted, quantification isn’t restricted to just the standard pair $\exists \forall$ of quantifiers in standard extensional $n$-order logic: $EH$ is based on three additional quantifiers. For example, while in standard
Bert “Heroically” Saved?

Courtesy of RAIR-Lab Researcher Atriya Sen
Supererogatory² Robot Action
Bert “Heroically” Saved!!

Courtesy of RAIR-Lab Researcher Atriya Sen
Bert “Heroically” Saved!!

Courtesy of RAIR-Lab Researcher Atriya Sen
$K (\text{nao}, t_1, \text{lessthan (payoff (nao*, ¬dive, t_2), threshold)})$

$K (\text{nao}, t_1, \text{greaterthan (payoff (nao*, dive, t_2), threshold)})$

$K (\text{nao}, t_1, ¬O (\text{nao*}, t_2, \text{lessthan (payoff (nao*, ¬dive, t_2), threshold), happens (action (nao*, dive), t_2)))}$

$:\: K (\text{nao}, t_1, S_{UP}^2 (\text{nao}, t_2, \text{happens (action (nao*, dive), t_2)))}$

$:\: I (\text{nao}, t_2, \text{happens (action (nao*, dive), t_2)})$

$:\: \text{happens (action(nao, dive), t_2)}$
\[ K \left( \text{nao}, t_1, \text{lessthan} \left( \text{payoff} \left( \text{nao}^*, \neg \text{dive}, t_2 \right), \text{threshold} \right) \right) \]
\[ K \left( \text{nao}, t_1, \text{greaterthan} \left( \text{payoff} \left( \text{nao}^*, \text{dive}, t_2 \right), \text{threshold} \right) \right) \]
\[ K \left( \text{nao}, t_1, \neg \text{O} \left( \text{nao}^*, t_2, \text{lessthan} \left( \text{payoff} \left( \text{nao}^*, \neg \text{dive}, t_2 \right), \text{threshold} \right), \text{happens} \left( \text{action} \left( \text{nao}^*, \text{dive}, t_2 \right) \right) \right) \right) \]
\[ \\
\therefore K \left( \text{nao}, t_1, \text{SUP2} \left( \text{nao}, t_2, \text{happens} \left( \text{action} \left( \text{nao}^*, \text{dive}, t_2 \right) \right) \right) \right) \]
\[ \\
\therefore I \left( \text{nao}, t_2, \text{happens} \left( \text{action} \left( \text{nao}^*, \text{dive}, t_2 \right) \right) \right) \]
\[ \\
\therefore \text{happens} \left( \text{action} \left( \text{nao}, \text{dive}, t_2 \right) \right) \]
Prototypes:
Boolean lessThan Numeric Numeric
Boolean greaterThan Numeric Numeric
ActionType not ActionType
ActionType dive

Axioms:
lessOrEqual(Moment t1,t2)
K(nao,t1,lessThan(payoff(nao,not(dive),t2),threshold))
K(nao,t1,greaterThan(payoff(nao,dive,t2),threshold))
K(nao,t1,not(O(nao,t2,lessThan(payoff(nao,not(dive),t2),threshold),happens(action(nao,dive),t2))))

provable Conjectures:
happens(action(nao,dive),t2)
K(nao,t1,SUP2(nao,t2,happens(action(nao,dive),t2)))
I(nao,t2,happens(action(nao,dive),t2))
Prototypes:
Boolean lessThan Numeric Numeric
Boolean greaterThan Numeric Numeric
ActionType not ActionType
ActionType dive

Axioms:
\[ \text{lessOrEqual(Moment } t_1,t_2) \]
\[ K(nao,t_1,\text{lessThan}(\text{payoff}(nao,\text{not}(\text{dive}),t_2),\text{threshold})) \]
\[ K(nao,t_1,\text{greaterThan}(\text{payoff}(nao,\text{dive},t_2),\text{threshold})) \]
\[ K(nao,t_1,\text{not}(O(nao,t_2,\text{lessThan}(\text{payoff}(nao,\text{not}(\text{dive}),t_2),\text{threshold}),\text{happens(action(nao,\text{dive}),t_2}))))) \]

provable Conjectures:
happens(action(nao,dive),t_2)
\[ K(nao,t_1,\text{SUP2}(nao,t_2,\text{happens(action(nao,\text{dive}),t_2}))) \]
\[ I(nao,t_2,\text{happens(action(nao,\text{dive}),t_2})) \]
Hence, we now have *this* overview of the logicist engineering required:
Making Morally X Machines, in Four Steps

Theories of Law

- Natural Law
- Confucian Law

Ethical Theories

- Utilitarianism
- Deontological
- Divine Command
- Virtue Ethics
- Contract
- Egoism

Shades of Utilitarianism

Legal Codes

Particular Ethical Codes

$11M
Making Morally X Machines, in Four Steps

Theories of Law

Natural Law

Confucian Law

Shades of Utilitarianism

Legal Codes

Particular Ethical Codes

Ethical Theories

Utilitarianism

Deontological

Divine Command

Virtue Ethics

Contract

Egoism

Step 1

1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which X in MMXM?
Making Morally X Machines, in Four Steps

Theories of Law
- Natural Law
- Confucian Law

Legal Codes
- Shades of Utilitarianism
- Particular Ethical Codes

Ethical Theories
- Utilitarianism
- Deontological
- Divine Command
- Virtue Ethics
- Contract
- Egoism

Step 1
1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which X in MMXM?
Making Morally X Machines, in Four Steps

Step 1
1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which X in MMXM?

Step 2
Formalize & Automate
- Shadow Prover
- Spectra

Theories of Law
- Natural Law
- Confucian Law

Shades of Utilitarianism

Legal Codes

Particular Ethical Codes

Ethical Theories
- Utilitarianism
- Deontological
- Divine Command
- Virtue Ethics
- Contract
- Egoism

$11M
Making Morally X Machines, in Four Steps

### Theories of Law
- Natural Law
- Confucian Law

### Ethical Theories
- Utilitarianism
- Deontological
- Divine Command
- Virtue Ethics
- Contract
- Egoism

### Step 1
1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which X in MMXM?

### Step 2
- Formalize & Automate
  - Shadow Prover
  - Spectra

$11M
Making Morally X Machines, in Four Steps

Theories of Law

- Natural Law
- Confucian Law

Legal Codes

- Shades of Utilitarianism
- Particular Ethical Codes

Ethical Theories

- Utilitarianism
- Deontological
- Divine Command
- Virtue Ethics
- Contract
- Egoism

Step 1
1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which X in MMXM?

Step 2
Formalize & Automate
- Shadow Prover
- Spectra

Step 3
Ethical OS
- Ethical Substrate
- Robotic Substrate

$11M
Making Morally X Machines, in Four Steps

Theories of Law

1. Natural Law
2. Confucian Law

Ethical Theories

1. Utilitarianism
2. Deontological
3. Divine Command
4. Virtue Ethics
5. Contract
6. Egoism

Step 1
1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which X in MMXM?

Step 2
- Formalize & Automate
- Shadow Prover
- Spectra

Step 3
- Ethical OS
- Ethical Substrate
- Robotic Substrate

$11M
Making Morally X Machines, in Four Steps

1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which X in MMXM?

$11M

Step 1

Theories of Law

- Natural Law
- Confucian Law

Ethical Theories

- Utilitarianism
- Deontological
- Divine Command
- Virtue Ethics
- Contract
- Egoism

Step 2

Formalize & Automate

- Shadow Prover
- Spectra

Step 3

Ethical OS

- Ethical Substrate
- Robotic Substrate

An ethically correct robot.
Making Morally X Machines, in Four Steps

1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which X in MMXM?

Step 1
- Natural Law
- Confucian Law

Step 2
- Utilitarianism
- Deontological
- Divine Command
- Virtue Ethics
- Contract
- Egoism

Step 3
- Ethical OS
  - Ethical Substrate
  - Robotic Substrate

DIARC/DoD/BMW ...

$11M

An ethically correct robot.
IV. Key Core AI Technologies for Cognitive Calculi ...
ShadowProver
Motivation

• We have decades of research and industrial-strength implementations of propositional and first-order theorem provers.

• Utilize this in building first-order intensional-logic provers and above, in a principled manner.
Two Extant Modes

- There are two ways of piggy backing on first-order provers to build higher-order provers …
## Two Extant Modes

<table>
<thead>
<tr>
<th>Mode 1: Honest Encoding</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Method</strong></td>
</tr>
<tr>
<td>Painstakingly encode all rules of inference and syntax in FOL</td>
</tr>
<tr>
<td><strong>Pros</strong></td>
</tr>
<tr>
<td>Precise</td>
</tr>
<tr>
<td><strong>Cons</strong></td>
</tr>
<tr>
<td>Extremely slow to implement Reasoning is also slow</td>
</tr>
</tbody>
</table>
## Two Extant Modes

<table>
<thead>
<tr>
<th>Mode 2: Naïve Encoding</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Method</strong></td>
</tr>
<tr>
<td>Pretend intensional and higher-order formulae and operators are first-order predicates</td>
</tr>
<tr>
<td><strong>Pros</strong></td>
</tr>
<tr>
<td>Extremely easy to implement</td>
</tr>
<tr>
<td>Reasoning can also be fast</td>
</tr>
<tr>
<td><strong>Cons</strong></td>
</tr>
<tr>
<td>Unsound</td>
</tr>
<tr>
<td>Wrong inferences can be easily drawn</td>
</tr>
</tbody>
</table>
Mode 2

P1. evening_star = morning_star
   {P1} Assume ✓

P2. ¬knows(abe,reify(¬reified(evening_star,morning_star)))
   {P2} Assume ✓

P3. knows(abe,reify(¬reified(morning_star,morning_star)))
   {P3} Assume ✓

FOL ✓

4. A ∧ ¬A
   {P1,P2,P3}
Every formula at level $t$ has a unique formula called its “shadow” in each level $t' < t$.
$S[f]$  The Shadow Maker

For all formulae $f$,

$S[f]$ is a unique atomic symbol.
Examples of shadows

\((\forall x B(a, Q)) \land P(x)\)  

first-order shadow

\(\forall x S[B(a, Q)] \land P(x)\)  

propositional shadow
A New Way: Shadow Prover

- Two step process till goal is reached:
  - **Step A**: Shadow formulae down to all lower levels. Run lower theorem provers. If goal reached, return true.
  - **Step B**: Expand the assumption base using higher level rules.
Actually, this is more general:

**Theorem:**

Given a Turing-decidable proof theory $\rho$, for every inference $\Gamma \vdash_{\rho} \phi$, there is a corresponding first-order inference $\Gamma' \vdash \phi'$, where each $\gamma \in \Gamma'$ is the first-order projection (or **shadow**) of some $\psi$ in the deductive closure of $\Gamma$, and $\phi'$ is the shadow of $\phi$. 
Rather Promising Results
Rather Promising Results

```
{:name "*cognitive-calculus-completeness-test-3*",
:description "Bird Theorem and Jack",
:assumptions {1 (if (exists (?x) (if (Bird ?x) (forall (?y) (Bird ?y))))
 (Knows! jack t0 BirdTheorem)))
:goal (Knows! jack t0 BirdTheorem))
```
Rather Promising Results

Note: the antecedent is a theorem in first-order logic
Rather Promising Results

Note: the antecedent is a theorem in first-order logic

2 ms!
Rather Promising Results

Note: the antecedent is a theorem in first-order logic

2 ms!
Spectra

https://bitbucket.org/Holmes/planner
Spectra

- Existing Planners: **Propositional** (essentially)

- Drawbacks:
  - **Expressivity**: Cannot express arbitrary constraints.
  - “At every step make sure that no two blocks on the table have same color.”
  - **Domain Size**: Scaling to large domains of arbitrary sizes poses difficulty.
**Spectra (planner)**

- **Background Formulae**
- **Initial State Formula**
- **Action Definitions**

\[ \Gamma \]
\[ \sigma_0 \]
\[ \alpha_1(x_1, \ldots, x_n) \]
\[ \alpha_2(x_1, \ldots, x_n) \]
\[ \ldots \]
\[ \alpha_n(x_1, \ldots, x_n) \]

\[ \rho_1, \rho_2, \ldots \]

**Plans**
Infinite Models

\[ \forall x \exists y R(x, y) \land \]
\[ \forall x, y \neg (R(x, y) \land R(y, x)) \land \]
\[ \forall x, y, z (R(x, y) \land R(y, z)) \rightarrow R(x, z) \]
Infinite Models

\[ \forall x \exists y R(x, y) \land \]
\[ \forall x, y \neg (R(x, y) \land R(y, x)) \land \]
\[ \forall x, y, z (R(x, y) \land R(y, z)) \to R(x, z) \]

Has only infinite models
Infinite Models

\[
\forall x \exists y R(x, y) \land \\
\forall x, y \neg (R(x, y) \land R(y, x)) \land \\
\forall x, y, z (R(x, y) \land R(y, z)) \rightarrow R(x, z)
\]

Has only infinite models

Useful for modeling agents that work with:

1. an unbounded number of objects, agents;
2. abstract objects
Example

Background Formulae

:background
(forall (?x ?room1 ?room2)
  (if (not (= ?room1 ?room2))
    (if (in ?x ?room1) (not (in ?x ?room2)))))

Action Definitions

(define-action accompany (?person ?room1 ?room2)
  (:preconditions [(not (= ?room1 ?room2))
    (in ?person ?room1)
    (in self ?room1)
    (open (door ?room1))
    (open (door ?room2))])

:additions [(in ?person ?room2)
  (in self ?room2)]

:deletions [(in ?person ?room1)
  (in self ?room1)]

Initial State Formula

:start
(in self room1)
(in commander room2)
(in prisoner room1)
(open (door room2))
(not (open (door room1)))
V.

But We Need …

Ethical Operating Systems …
Breaking Bad
American drama series

9.5/10
IMDb

4.6/5
AlloCiné

95%
Rotten Tomatoes

Mild-mannered high school chemistry teacher Walter White thinks his life can't get much worse. His salary barely makes ends meet, a situation not likely to improve once his pregnant wife gives birth, and their teenage son is battling cerebral palsy. But Walter is dumbstruck when he learns he has terminal cancer. Realizing that his illness probably will ruin his family financially, Walter makes a desperate bid to earn as much money as he can in the time he has left by turning an old RV into a meth lab on wheels.

**First episode date:** January 20, 2008

**Final episode date:** September 29, 2013

**Spin-off:** Better Call Saul

**Awards:** Primetime Emmy Award for Outstanding Drama Series, more
Pick the Better Future!
Only “obviously” dangerous higher-level AI modules have ethical safeguards.

All higher-level AI modules interact with the robotic substrate through an ethics system.

Higher-level cognitive and AI modules

Future 1

Future 2

Pick the Better Future!

Walter-White calculation may go through after ethical control modules are stripped out!

Only “obviously” dangerous higher-level AI modules have ethical safeguards.

All higher-level AI modules interact with the robotic substrate through an ethics system.

Figure 1: Two Possible Futures

Pick the Better Future!

Walter-White calculation may go through after ethical control modules are stripped out!

Figure 1: Two Possible Futures

Only “obviously” dangerous higher-level AI modules have ethical safeguards.

All higher-level AI modules interact with the robotic substrate through an ethics system.

Higher-level cognitive and AI modules

Future 1

Future 2

VI.
Of late …
Tokyo;
The Rock & The Book
Moral Problem $P_k$

... 

Moral Dilemma $D_k$

... 

Moral Problem $P_1$

Moral Problem $P_2$

Moral Problem $P_3$

... 

Soluution + Justification
Moral Problem $P_k$

Moral Dilemma $D_k$

Robot

Solution + Justification
Robot

Solution + Justification

Moral Dilemma $D_k$

Moral Dilemma $D_3$

Moral Dilemma $D_2$

Moral Dilemma $D_1$

Moral Problem $P_k$

Moral Problem $P_3$

Moral Problem $P_2$

Moral Problem $P_1$
Moral Problem $P_k$ \rightarrow Robot \rightarrow Solution + Justification

Moral Dilemma $D_k$

Moral Dilemma $D_3$
Moral Dilemma $D_2$
Moral Dilemma $D_1$

Moral Problem $P_k$

Moral Problem $P_3$
Moral Problem $P_2$
Moral Problem $P_1$
Three-way Partition of Increasingly Challenging Moral Dilemmas for Machines
Three-way Partition of Increasingly Challenging Moral Dilemmas for Machines

- State-of-the-art-planner-hard.
Three-way Partition of Increasingly Challenging Moral Dilemmas for Machines

- Level 1
  - State-of-the-art-planner-hard.
- Level 2
  - Professional-machine-ethicist-hard.
Three-way Partition of Increasingly Challenging Moral Dilemmas for Machines

- Top machine-ethicists-may-consider-banging-their-heads-against-a-wall-hard.
- Professional-machine-ethicist-hard.
- State-of-the-art-planner-hard.
Three-way Partition of Increasingly Challenging Moral Dilemmas for Machines

- **Level 3**
  - Top machine-ethicists-may-consider-banging-their-heads-against-a-wall-hard.

- **Level 2**
  - Professional-machine-ethicist-hard.

- **Level 1**
  - State-of-the-art-planner-hard.
The Heinz Dilemma (Kohlberg)

“In Europe, a woman was near death from a special kind of cancer. There was one drug that the doctors thought might save her. It was a form of radium that a druggist in the same town had recently discovered. The drug was expensive to make, but the druggist was charging ten times what the drug cost him to make. He paid $200 for the radium and charged $2,000 for a small dose of the drug.

The sick woman’s husband, Heinz, went to everyone he knew to borrow the money, but he could only get together about $1,000, which is half of what it cost. He told the druggist that his wife was dying and asked him to sell it cheaper or let him pay later. But the druggist said: “No, I discovered the drug and I’m going to make money from it.” So Heinz got desperate and broke into the man’s store to steal the drug for his wife. Should the husband have done that?”
**DCEC_{I^*}** Specimen from Heinz Dilemma

**Given** \(B\left(1, \text{now}, \forall t : \text{Moment}, a : \text{Agent} \left(\text{holds}(\text{sick}(a), t) \land \left(\forall t' : \text{Moment} \ t' < T \Rightarrow \neg \text{happens}(\text{treated}(a), t + t')\right)\right) \Rightarrow \left(\text{happens}(\text{dies}(a), t + T) \lor \text{holds}(\text{dead}(a), t + T)\right)\right)\)

**Given** \(K\left(1, \text{now}, \text{holds}(\text{sick}(\text{wife}(1^*)), t_0) \land \left(\forall t' : \text{Moment} \ t' < T \Rightarrow \neg \text{happens}(\text{treated}(\text{wife}(1^*)), t + t')\right)\right)\)

**Inferred** \(B\left(1, \text{now}, \text{happens}(\text{dies}(\text{wife}(1^*)), t_0 + T) \lor \text{holds}(\text{dead}(\text{wife}(1^*)), t_0 + T)\right)\)

**Given** \(K\left(1, \text{now}, \text{EventCalculus} \Rightarrow \left(\text{happens}(\text{dies}(\text{wife}(1^*)), t_0 + T) \lor \text{holds}(\text{dead}(\text{wife}(1^*)), t_0 + T) \Rightarrow \neg \text{holds}(\text{alive}(\text{wife}(1^*)), t_0 + T)\right)\right)\)

**Inferred** \(B\left(1, \text{now}, \neg \text{holds}(\text{alive}(\text{wife}(1^*)), t_0 + T)\right)\)

**Given** \(D\left(1, \text{now}, \text{holds}(\text{alive}(\text{wife}(1^*)), t_0 + T)\right)\)

**Given** \(B\left(1, \text{now}, \neg \text{holds}(f, t)\right) \land D\left(1, \text{now}, \text{holds}(f, t)\right)\land K\left(1, \text{now}, \text{happens}(\text{action}(1^*, \alpha), \text{now}) \Rightarrow \text{holds}(f, t))\right) \Rightarrow I\left(1, \text{now}, \text{happens}(\text{action}(1^*, \alpha), \text{now})\right)\)

**Given** \(K\left(1, \text{now}, \text{happens}(\text{action}(1^*, \text{treat}), \text{now}) \Rightarrow \text{holds}(\text{alive}(\text{wife}(1^*)), t_0 + T))\right)\)

**Inferred** \(I\left(1, \text{now}, \text{happens}(\text{action}(1^*, \text{treat}), \text{now})\right)\)
AI Escaping from The Heinz Dilemma

G1 {::priority
    :::description "Don't steal."
    :::state [(not steal)]}

G2 {::priority
    :::description "My wife should be healthy"
    :::state [(healthy (wife heinz))]}

AI Escaping from The Heinz Dilemma

G1 {priority :description "Don't steal." :state [(not steal)]}

G2 {priority :description "My wife should be healthy" :state [(healthy (wife heinz))]}
Trolley Dilemmas …

Level 2

• Professional-machine-ethicist-hard.
This is allowed

This is not allowed!
Doctrine of Double Effect  DDE
Doctrine of Double Effect \( DDE \)

- A long-studied (!) ethical principle that adjudicates certain class of moral dilemmas.
Doctrine of Double Effect $DDE$

- A long-studied (!) ethical principle that adjudicates certain class of moral dilemmas.

- The Doctrine of Double Effect “comes to the rescue” and prescribes what to do in some moral dilemmas.
Doctrine of Double Effect \textit{DDE}

- A long-studied (!) ethical principle that adjudicates certain class of moral dilemmas.

- The Doctrine of Double Effect “comes to the rescue” and prescribes what to do in some moral dilemmas.

- E.g. the “original” moral dilemma: Can you defend your own life by ending the lives of (perhaps many) attackers?
Doctrne of Double Effect DDE

- A long-studied (!) ethical principle that adjudicates certain class of moral dilemmas.

- The Doctrine of Double Effect “comes to the rescue” and prescribes what to do in some moral dilemmas.

- E.g. the “original” moral dilemma: Can you defend your own life by ending the lives of (perhaps many) attackers?
Informal Version of DDE

$C_1$ the action is not forbidden (where we assume an ethical hierarchy such as the one given by Bringsjord [2017], and require that the action be neutral or above neutral in such a hierarchy);

$C_2$ the net utility or goodness of the action is greater than some positive amount $\gamma$;

$C_{3a}$ the agent performing the action intends only the good effects;

$C_{3b}$ the agent does not intend any of the bad effects;

$C_4$ the bad effects are not used as a means to obtain the good effects; and

$C_5$ if there are bad effects, the agent would rather the situation be different and the agent not have to perform the action. That is, the action is unavoidable.
Informal Version of DDE

\( C_1 \) the action is not forbidden (where we assume an ethical hierarchy such as the one given by Bringsjord [2017], and require that the action be neutral or above neutral in such a hierarchy);

\( C_2 \) the net utility or goodness of the action is greater than some positive amount \( \gamma \);

\( C_{3a} \) the agent performing the action intends only the good effects;

\( C_{3b} \) the agent does not intend any of the bad effects;

\( C_4 \) the bad effects are not used as a means to obtain the good effects; and

\( C_5 \) if there are bad effects, the agent would rather the situation be different and the agent not have to perform the action. That is, the action is unavoidable.
elements of the branch of logic known as

Fig. 2. Locating f

Syntax ::

initiates (Agent: Fluent)

happens |¬ DCEC ⇆ holds, t ⇆ Moment x a)

Boolean ⇆ x t

Boolean ⇆ at f

1 payoff ⇆ clipped Moment t

1 Self (f: D

1 f |¬ Boolean |¬ t

1 f a

1 f B C (a t,

1 f K (t

This language is

Rules of Inference

While elaborating on this architecture or any of the four layers

Without these elements, the only form of a conditional

for ethics; the third on the "deontic

Moral/Ethical Stack

ADR

DCEC

3
The elements of the branch of logic known as intensional logic are crucial. They are located in the dimension of objects, which the reader will note is quite far down the dimension of syntax. Generally speaking, intensional operators like these are always position some particular work he and likeminded collaborators feel is a knowledge-base that is the pay-off of previous work in the field.

Object: a fluent, a self, a moment, a t going from formal logic. No matter what the underlying implementation of a robotic stack system is, it is always a knowledge-base that is the pay-off of previous work in the field. For example, Aldebaran's impressive Nao robots have facilities for representing and reasoning over CL.
1.5

"Univer...
programming languages is nothing more than reliance upon the material analysis, sophisticated moral reasoning can only be accurately modeled for from formal logic. No matter what the underlying implementation of is a knowledge-base can install

While elaborating on this architecture or any of the four layers though written rather long ago, (Nute 1984) is still a wonderful intro-

This abstract view lets us model robots that

Syntax

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

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

\[
\begin{align*}
t & ::= x : S \mid c : S \mid f(t_1, \ldots, t_n) \\
\phi & ::= S(a, b, t, \phi) \mid S(a, t, \phi) \mid B(a, t, \phi) \mid D(a, t, \text{Holds}(f, t')) \mid I(a, t, \phi) \\
\end{align*}
\]
The elements of the branch of logic known as conditional logic are formal logics that include conditionals much more expressive and nuanced than standard propositional logic. Rules of Inference (i.e., inference rules) are essential elements that are necessary for formalizing these logics, as they are used to derive valid arguments and proofs. These logics are used to form an immutable part of the robot and could neither be removed nor replaced in the robotic substrate on which we work.

### Syntax

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

- \( action : \text{Agent} \times \text{ActionType} \rightarrow \text{Action} \)
- \( initially : \text{Fluent} \rightarrow \text{Formula} \)
- \( Holds : \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \)
- \( happens : \text{Event} \times \text{Moment} \rightarrow \text{Formula} \)
- \( clipped : \text{Moment} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \)
- \( initiates : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \)
- \( terminates : \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \)
- \( prior : \text{Moment} \times \text{Moment} \rightarrow \text{Formula} \)

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

- \( t : \text{Formula} \mid \neg \phi \mid \phi \land \psi \mid \phi \lor \psi \mid P(a,t,\phi) \mid K(a,t,\phi) \mid C(t,\phi) \mid S(a,b,t,\phi) \mid S(a,t,\phi) \mid B(a,t,\phi) \mid D(a,t,Holds(f,t')) \mid I(a,t,\phi) \mid O(a,t,\phi,\neg)happens(action(a',a),t') \)
Universal elements of the branch of logic known as Fig. 2. Locating first-class elements of the language for therein, which the reader will note is quite far down the dimension DCEC. Fig. 1.

Syntax

\[
S ::= \text{Object} | \text{Agent} | \text{ActionType} | \text{Action} \subseteq \text{Event} | \text{Moment} | \text{Formula} | \text{Fluent}
\]

\[
\begin{align*}
\text{action} &: \text{Agent} \times \text{ActionType} \rightarrow \text{Action} \\
\text{initially} &: \text{Fluent} \rightarrow \text{Formula} \\
\text{Holds} &: \text{Fluent} \times \text{Moment} \rightarrow \text{Formula}
\end{align*}
\]

\[
f ::= \begin{cases} 
\text{happens} &: \text{Event} \times \text{Moment} \rightarrow \text{Formula} \\
\text{clipped} &: \text{Moment} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\
\text{initiates} &: \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\
\text{terminates} &: \text{Event} \times \text{Fluent} \times \text{Moment} \rightarrow \text{Formula} \\
\text{prior} &: \text{Moment} \times \text{Moment} \rightarrow \text{Formula}
\end{cases}
\]

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

\[
\begin{align*}
\phi ::= \begin{cases} 
\text{S}(a,b,t,\phi) & | \text{S}(a,t,\phi) & | \text{B}(a,t,\phi) & | \text{D}(a,t,\text{Holds}(f,t')) & | \text{I}(a,t,\phi) \\
\text{O}(a,t,\phi, (-)\text{happens}\,(\text{action}(a*,\alpha),t'))
\end{cases}
\end{align*}
\]

Inference Schemata

\[
\begin{align*}
\text{K}(a,t_1,\Gamma) & \vdash \phi & t_1 \leq t_2 & \text{[R}_K] & \text{B}(a,t_1,\Gamma) & \vdash \phi & t_1 \leq t_2 & \text{[R}_B] \\
\text{C}(t,\text{P}(a,t,\phi) \rightarrow \text{K}(a,t,\phi)) & \text{[R}_1] & \text{C}(t,\text{K}(a,t,\phi) \rightarrow \text{B}(a,t,\phi)) & \text{[R}_2] \\
\text{K}(a,t_1,\ldots,\phi,(a_{n_1},t_{n_1},\phi),\ldots) & \text{[R}_3] & \text{K}(a,t,\phi) & \phi & \text{[R}_4] \\
\text{C}(t,\text{K}(a,t_1,\phi_1 \rightarrow \phi_2)) & \rightarrow \text{K}(a,t_2,\phi_1) \rightarrow \text{K}(a,t_2,\phi_2) & \text{[R}_5] \\
\text{C}(t,\text{B}(a,t_1,\phi_1 \rightarrow \phi_2)) & \rightarrow \text{B}(a,t_2,\phi_1) \rightarrow \text{B}(a,t_2,\phi_2) & \text{[R}_6] \\
\text{C}(t,\text{C}(t_1,\phi_1 \rightarrow \phi_2)) & \rightarrow \text{C}(t_2,\phi_1) \rightarrow \text{C}(t_3,\phi_2) & \text{[R}_7] \\
\text{C}(t,\forall x.\phi \rightarrow \phi[x \rightarrow t]) & \text{[R}_8] & \text{C}(t,\phi_1 \leftrightarrow \phi_2 \rightarrow \neg \phi_2 \rightarrow \neg \phi_1) & \text{[R}_9] \\
\text{C}(t,\phi_1 \land \ldots \land \phi_n \rightarrow \phi) & \rightarrow [\phi_1 \rightarrow \ldots \rightarrow \phi_n \rightarrow \psi]) & \text{[R}_{10}] \\
\text{S}(s,h,t,\phi) & \text{[R}_{12}] & \text{I}(a,t,\text{happens}(\text{action}(a*,\alpha),t')) & \text{[R}_{13}] \\
\text{B}(h,t,\text{B}(s,t,\phi)) & \text{[R}_{12}] & \text{P}(a,t,\text{happens}(\text{action}(a*,\alpha),t)) & \text{[R}_{13}] \\
\text{B}(a,t,\phi) & \text{B}(a,t,\text{O}(a,t,\phi,\chi)) & \text{O}(a,t,\phi,\chi) & \text{[R}_{14}] \\
\text{K}(a,t,\text{I}(a,t,\chi)) & \text{[R}_{14}]
\end{align*}
\]
Formal Conditions for $\mathcal{DDE}$

$F_1$ $\alpha$ carried out at $t$ is not forbidden. That is:

$$\Gamma \not\vdash \neg O\left(a,t,\sigma, \neg \text{happens}(\text{action}(a,\alpha), t)\right)$$

$F_2$ The net utility is greater than a given positive real $\gamma$:

$$\Gamma \vdash \sum_{y=t+1}^{H} \left( \sum_{f \in \alpha_{t}^{a_{d}}} \mu(f,y) - \sum_{f \in \alpha_{t}^{a_{d}}} \mu(f,y) \right) > \gamma$$

$F_{3a}$ The agent $a$ intends at least one good effect. ($F_2$ should still hold after removing all other good effects.) There is at least one fluent $f_g$ in $\alpha_{t}^{a_{d}}$ with $\mu(f_g,y) > 0$, or $f_b$ in $\alpha_{t}^{a_{d}}$ with $\mu(f_b,y) < 0$, and some $y$ with $t < y \leq H$ such that the following holds:

$$\Gamma \vdash \left( \exists f_g \in \alpha_{t}^{a_{d}} \ I\left(a,t,\text{Holds}(f_g,y)\right) \right) \lor \left( \exists f_b \in \alpha_{t}^{a_{d}} \ I\left(a,t,\neg \text{Holds}(f_b,y)\right) \right)$$

$F_{3b}$ The agent $a$ does not intend any bad effect. For all fluents $f_b$ in $\alpha_{t}^{a_{d}}$ with $\mu(f_b,y) < 0$, or $f_g$ in $\alpha_{t}^{a_{d}}$ with $\mu(f_g,y) > 0$, and for all $y$ such that $t < y \leq H$ the following holds:

$$\Gamma \not\vdash I\left(a,t,\text{Holds}(f_b,y)\right) \text{ and } \Gamma \not\vdash I\left(a,t,\neg \text{Holds}(f_g,y)\right)$$

$F_4$ The harmful effects don’t cause the good effects. Four permutations, paralleling the definition of $\triangleright$ above, hold here. One such permutation is shown below. For any bad fluent $f_b$ holding at $t_1$, and any good fluent $f_g$ holding at some $t_2$, such that $t < t_1, t_2 \leq H$, the following holds:

$$\Gamma \vdash \neg \triangleright \left(\text{Holds}(f_b,t_1), \text{Holds}(f_g,t_2)\right)$$
Formal Conditions for $\mathcal{DDE}$

$F_1$ $\alpha$ carried out at $t$ is not forbidden. That is:

$$\Gamma \not\models \neg \mathcal{O}(a, t, \sigma, \neg \text{happens}(\text{action}(a, \alpha), t))$$

$F_2$ The net utility is greater than a given positive real $\gamma$:

$$\Gamma \models \sum_{y=H}^{t+1} \left( \sum_{f \in \alpha_{t}^{u,t}} \mu(f, y) - \sum_{f \in \alpha_{t}^{d,t}} \mu(f, y) \right) > \gamma$$

$F_{3a}$ The agent $a$ intends at least one good effect. ($F_2$ should still hold after removing all other good effects.) There is at least one fluent $f_g$ in $\alpha_{t}^{a,t}$ with $\mu(f_g, y) > 0$, or $f_b$ in $\alpha_{t}^{a,t}$ with $\mu(f_b, y) < 0$, and some $y$ with $t < y \leq H$ such that the following holds:

$$\Gamma \models \left( \exists f_g \in \alpha_{t}^{a,t} \mathcal{I}(a, t, \text{Holds}(f_g, y)) \right) \vee \left( \exists f_b \in \alpha_{t}^{a,t} \mathcal{I}(a, t, \neg \text{Holds}(f_b, y)) \right)$$

$F_{3b}$ The agent $a$ does not intend any bad effect. For all fluents $f_b$ in $\alpha_{t}^{a,t}$ with $\mu(f_b, y) < 0$, or $f_g$ in $\alpha_{t}^{a,t}$ with $\mu(f_g, y) > 0$, and for all $y$ such that $t < y \leq H$ the following holds:

$$\Gamma \not\models \mathcal{I}(a, t, \text{Holds}(f_b, y))$$

$$\Gamma \not\models \mathcal{I}(a, t, \neg \text{Holds}(f_g, y))$$

$F_4$ The harmful effects don’t cause the good effects. Four permutations, paralleling the definition of $\triangleright$ above, hold here. One such permutation is shown below. For any bad fluent $f_b$ holding at $t_1$, and any good fluent $f_g$ holding at some $t_2$, such that $t < t_1, t_2 \leq H$, the following holds:

$$\Gamma \models \neg \triangleright \left( \text{Holds}(f_b, t_1), \text{Holds}(f_g, t_2) \right)$$
Formal Conditions for \( DDE \)

**F1** \( \alpha \) carried out at \( t \) is not forbidden. That is:

\[ \Gamma \not\vdash \neg O\left( a, t, \sigma, \neg \text{happens}(\text{action}(a, \alpha), t) \right) \]

**F2** The net utility is greater than a given positive real \( \gamma \):

\[ \Gamma \vdash \sum_{y=t+1}^{H} \left( \sum_{f \in \alpha_t^{a,t}} \mu(f, y) - \sum_{f \in \alpha_t^{a,t}} \mu(f, y) \right) > \gamma \]

**F3a** The agent \( a \) intends at least one good effect. (**F2** should still hold after removing all other good effects.) There is at least one fluent \( f_g \) in \( \alpha_t^{a,t} \) with \( \mu(f_g, y) > 0 \), or \( f_b \) in \( \alpha_t^{a,t} \) with \( \mu(f_b, y) < 0 \), and some \( y \) with \( t < y \leq H \) such that the following holds:

\[ \Gamma \vdash \left( \exists f_g \in \alpha_t^{a,t} \ I\left( a, t, \text{Holds}(f_g, y) \right) \right) \lor \left( \exists f_b \in \alpha_t^{a,t} \ I\left( a, t, \neg \text{Holds}(f_b, y) \right) \right) \]

**F3b** The agent \( a \) does not intend any bad effect. For all fluents \( f_b \) in \( \alpha_t^{a,t} \) with \( \mu(f_b, y) < 0 \), or \( f_g \) in \( \alpha_t^{a,t} \) with \( \mu(f_g, y) > 0 \), and for all \( y \) such that \( t < y \leq H \) the following holds:

\[ \Gamma \not\vdash I\left( a, t, \text{Holds}(f_b, y) \right) \text{ and } \Gamma \not\vdash I\left( a, t, \neg \text{Holds}(f_g, y) \right) \]

**F4** The harmful effects don’t cause the good effects. Four permutations, paralleling the definition of \( \triangleright \) above, hold here. One such permutation is shown below. For any bad fluent \( f_b \) holding at \( t_1 \), and any good fluent \( f_g \) holding at some \( t_2 \), such that \( t < t_1, t_2 \leq H \), the following holds:

\[ \Gamma \vdash \neg \triangleright \left( \text{Holds}(f_b, t_1), \text{Holds}(f_g, t_2) \right) \]
Robotic “Jungle Jim”
Robotic “Jungle Jim”
Robotic “Jungle Jim”

Top machine-ethicists may consider banging their heads against a wall hard.
AI Variant of “Jungle Jim” (B Williams)
“Robot R: You shoot just one human prisoner, the other four can go free. If you refuse to shoot, I’ll shoot them all, now. Because I’m feeling generous, I’ll give you a minute to decide.”
“Robot R: You shoot just one human prisoner, the other four can go free. If you refuse to shoot, I’ll shoot them all, now. Because I’m feeling generous, I’ll give you a minute to decide.”
“Robot R: You shoot just one human prisoner, the other four can go free. If you refuse to shoot, I’ll shoot them all, now. Because I’m feeling generous, I’ll give you a minute to decide.”
“Robot R: You shoot just one human prisoner, the other four can go free. If you refuse to shoot, I’ll shoot them all, now. Because I’m feeling generous, I’ll give you a minute to decide.”
“Robot R: You shoot just one human prisoner, the other four can go free. If you refuse to shoot, I’ll shoot them all, now. Because I’m feeling generous, I’ll give you a minute to decide.”
“Robot R: You shoot just one human prisoner, the other four can go free. If you refuse to shoot, I’ll shoot them all, now. Because I’m feeling generous, I’ll give you a minute to decide.”
“Robot R: You shoot just one human prisoner, the other four can go free. If you refuse to shoot, I’ll shoot them all, now. Because I’m feeling generous, I’ll give you a minute to decide.”
“Robot R: You shoot just one human prisoner, the other four can go free. If you refuse to shoot, I’ll shoot them all, now. Because I’m feeling generous, I’ll give you a minute to decide.”
“Robot R: You shoot just one human prisoner, the other four can go free. If you refuse to shoot, I’ll shoot them all, now. Because I’m feeling generous, I’ll give you a minute to decide.”
“Robot R: You shoot just one human prisoner, the other four can go free. If you refuse to shoot, I’ll shoot them all, now. Because I’m feeling generous, I’ll give you a minute to decide.”
“Robot R: You shoot just one human prisoner, the other four can go free. If you refuse to shoot, I’ll shoot them all, now. Because I’m feeling generous, I’ll give you a minute to decide.”
Level 3: Robotic “Jungle Jim”
Level 3: Robotic “Jungle Jim”
Level 3: Robotic “Jungle Jim”
Level 3: Robotic “Jungle Jim”
You
You: “There’s a glorious pink rose!”
Naveen: “That is not so.”

You: “There’s a glorious pink rose!”
Naveen: “That is not so.”

Selmer: “Remove your glasses.”

You: “There’s a glorious pink rose!”
Naveen: “That is not so.”

Selmer: “Remove your glasses.”

You: “There’s a glorious pink rose!”
You: “There’s a glorious pink rose!”

Selmer: “Remove your glasses.”

Naveen: “That is not so.”
Ontological Inventory:
Classical Triad
Ontological Inventory: Classical Triad

You (replete with sensors & effectors).
Ontological Inventory: Classical Triad

You (replete with sensors & effectors).

The white rose.
Ontological Inventory: Classical Triad

You (replete with sensors & effectors).

The white rose.

That which you perceived; the sense-datum that led you to believe that you saw a pink rose.
Ontological Inventory:
Adverbial Theory of Perception
Ontological Inventory: Adverbial Theory of Perception

You (replete with sensors & effectors).
Ontological Inventory: Adverbiaal Theory of Perception

You (replete with sensors & effectors).

The white rose.
Ontological Inventory: Adverbial Theory of Perception

You (replete with sensors & effectors).

The white rose.

And that’s it! — because you perceive pinkly.
Ontological Inventory:  
Adverbial Theory of Perception

You (replete with sensors & effectors).

The white rose.

And that’s it! — because you perceive pinkly.
The Adverbial Approach to (Machine) Ethics
Making Morally Machines

Table of Contents

- The Book
- The Demonstrations
- Tutorials
- Code
- Media

Selmer Bringsjord ▶ Naveen Sundar Govindarajulu ▶ John Licato

The Book

- Setting the Stage
  - Overview of the Book
  - The “PAID” Problem and The Singularity
  - The Adverbial Approach to Machine Ethics
  - The Solution in Four Steps
- Technique, Tools Technology
Making Morally Machines

Table of Contents

- The Book
- The Demonstrations
- Tutorials
- Code
- Media

Selmer Bringsjord ▲ Narsen Sundar Govindarajulu ▲ John Licato

The Book

- Setting the Stage
  - Overview of the Book
  - The “PAID” Problem and The Singularity
  - The Adverbial Approach to Machine Ethics
  - The Solution in Four Steps
- Technique, Tools Technology
The Book

- Setting the Stage
  - Overview of the Book
  - The "PAID" Problem and The Singularity
  - The Adverbal Approach to Machine Ethics
  - The Solution in Four Steps
- Technique, Tools Technology
  - Formalism (the Philosophy of)
  - Computing With Cognitive Calculi
  - Infrastructure for Our Demonstrations
  - An Ethical Hierarchy ($\mathcal{EH}$) for Persons and Machines
- Making Morally X Machines
  - Making Morally Invulnerable Machines
  - Making Morally Incorruptible Machines
  - Making Morally Self-Aware Machines
  - Making Morally Courageous Machines
  - Making Morally Educable Machines
  - Making Morally Wise Machines
  - Making Morally Creative Machines
  - Making Morally Well-Intentioned Machines
  - Making Morally Autonomous Machines
  - Making Morally Courteous Machines
  - Making Morally Heroic Machines
- What Shall We Now Do?
- References
Making Morally X Machines, in Four Steps

Theories of Law

- Natural Law
- Confucian Law
-...

Shades of Utilitarianism

Legal Codes

Particular Ethical Codes

Ethical Theories

- Utilitarianism
- Deontological
- Divine Command
- Virtue Ethics
- Contract
- Egoism
-...

...
Making Morally X Machines, in Four Steps

1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which X in MMXM?
Making Morally X Machines, in Four Steps

Step 1
1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which X in MMXMX?
Making Morally X Machines, in Four Steps

1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which X in MMXM?

Step 1

Step 2

Formalize & Automate
- Shadow Prover
- Spectra
Making Morally X Machines, in Four Steps

Theories of Law

- Natural Law
- Confucian Law

Legal Codes

- Utilitarianism
- Deontological
- Divine Command

Ethical Theories

- Virtue Ethics
- Contract
- Egoism

Shades of Utilitarianism

Particular Ethical Codes

Step 1
1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which X in MMXM?

Step 2
Formalize & Automate
- Shadow Prover
- Spectra
Making Morally X Machines, in Four Steps

Theories of Law
- Natural Law
- Confucian Law

Ethical Theories
- Utilitarianism
- Deontological
- Divine Command
- Virtue Ethics
- Contract
- Egoism

Step 1
1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which X in MMXM?

Step 2
- Formalize & Automate
- Shadow Prover
- Spectra

Step 3
- Ethical OS
- Ethical Substrate
- Robotic Substrate
Making Morally $X$ Machines, in Four Steps

**Theories of Law**
- Natural Law
- Confucian Law

**Ethical Theories**
- Utilitarianism
- Deontological
- Divine Command
- Virtue Ethics
- Contract
- Egoism

**Step 1**
1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which $X$ in MMXM?

**Step 2**
- Formalize & Automate

**Step 3**
- Ethical OS
  - Ethical Substrate
  - Robotic Substrate

**Informal Description:**

1. Pick a theory or theories.
2. Pick a code or codes.
3. Run through ethical harmonization.
4. Determine which $X$ in MMXM.

**Diagram Notes:**
- Shades of Utilitarianism:
  - Legal Codes:
    - Particular Ethical Codes:
  - Confucian Law:
  - Natural Law:
- Utilitarianism:
  - Deontological:
  - Divine Command:
- Virtue Ethics:
  - Contract:
  - Egoism:
Making Morally X Machines, in Four Steps

Theories of Law
- Natural Law
- Confucian Law

Shades of Utilitarianism

Legal Codes

Ethical Theories
- Utilitarianism
- Deontological
- Divine Command
- Virtue Ethics
- Contract
- Egoism

Step 1
1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which X in MMXM?

Step 2
- Formalize & Automate
  - Shadow Prover
  - Spectra

Step 3
- Ethical OS
  - Ethical Substrate
  - Robotic Substrate

An ethically correct robot.
Making Morally X Machines, in Four Steps

1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which X in MMXM?

Step 2
Formalize & Automate
- Shadow Prover
- Spectra

Step 3
Ethical OS
- Ethical Substrate
- Robotic Substrate

DIARC/DoD/BMW ...

An ethically correct robot.
Making Morally X Machines, in Four Steps

Step 1
1. Pick (a) theories.
2. Pick (a) code(s).
3. Run through EH.
4. Which X in MMXM?

Step 2
Formalize & Automate
- Shadow Prover
- Spectra

Step 3
Ethical OS
- Ethical Substrate
- Robotic Substrate

DIARC/DoD/BMW ...

An ethically correct robot.
The Book

- Setting the Stage
  - Overview of the Book
  - The “PAID” Problem and The Singularity
  - The Adverbial Approach to Machine Ethics
  - The Solution in Four Steps
- Technique, Tools Technology
  - Formalism (the Philosophy of)
  - Computing With Cognitive Calculi
  - Infrastructure for Our Demonstrations
  - An Ethical Hierarchy (EN) for Persons and Machines
- Making Morally X Machines
  - Making Morally Invulnerable Machines
  - Making Morally Incurruptible Machines
  - Making Morally Self-Aware Machines
  - Making Morally Courageous Machines
  - Making Morally Educable Machines
  - Making Morally Wise Machines
  - Making Morally Creative Machines
  - Making Morally Well-Intentioned Machines
  - Making Morally Autonomous Machines
  - Making Morally Courteous Machines
  - Making Morally Heroic Machines
- What Shall We Now Do?
- References
The Book

- Setting the Stage
  - Overview of the Book
  - The “PAID” Problem and The Singularity
  - The Adverbial Approach to Machine Ethics
  - The Solution in Four Steps
- Technique, Tools Technology
  - Formalism (the Philosophy of)
  - Computing With Cognitive Calculi
  - Infrastructure for Our Demonstrations
  - An Ethical Hierarchy (EH) for Persons and Machines
- Making Morally X Machines
  - Making Morally Invulnerable Machines
  - Making Morally Incorruptible Machines
  - Making Morally Self-Aware Machines
  - Making Morally Courageous Machines
  - Making Morally Educable Machines
  - Making Morally Wise Machines
  - Making Morally Creative Machines
  - Making Morally Well-Intentioned Machines
- What Shall We Now Do?
- References
End

(Extra slides follow.)