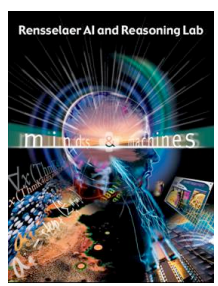
Piagetian Roboethics via Category Theory

Moving Beyond Mere Formal Operations to Engineer Robots Whose Decisions are Guaranteed to be Ethically Correct

Ralph Wojtowicz
Metron Inc.
1818 Library Street
Suite 600
Reston VA 20190 USA

Selmer Bringsjord
Konstantine Arkoudas
Joshua Taylor • Evan Gilbert
Trevor Housten • Micah Clark
Bram van Heuveln
Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Rensselaer Polytechnic Institute (RPI)
Troy NY 12180 USA

for *Roboethics Workshop* @ ICRA 209
5.17.09



The RPI team is indebted to IARPA for sustained support of semantic interoperability R&D, a key part of the research program described herein.

Approach: Logic

An Illogical View Refuted

The illogical view in question is one that is unfortunately now often espoused: viz., that since robots can be engineered to perform ethically on par with average humans (e.g., human soldiers), there's nothing unwise about engineering and deploying such robots.

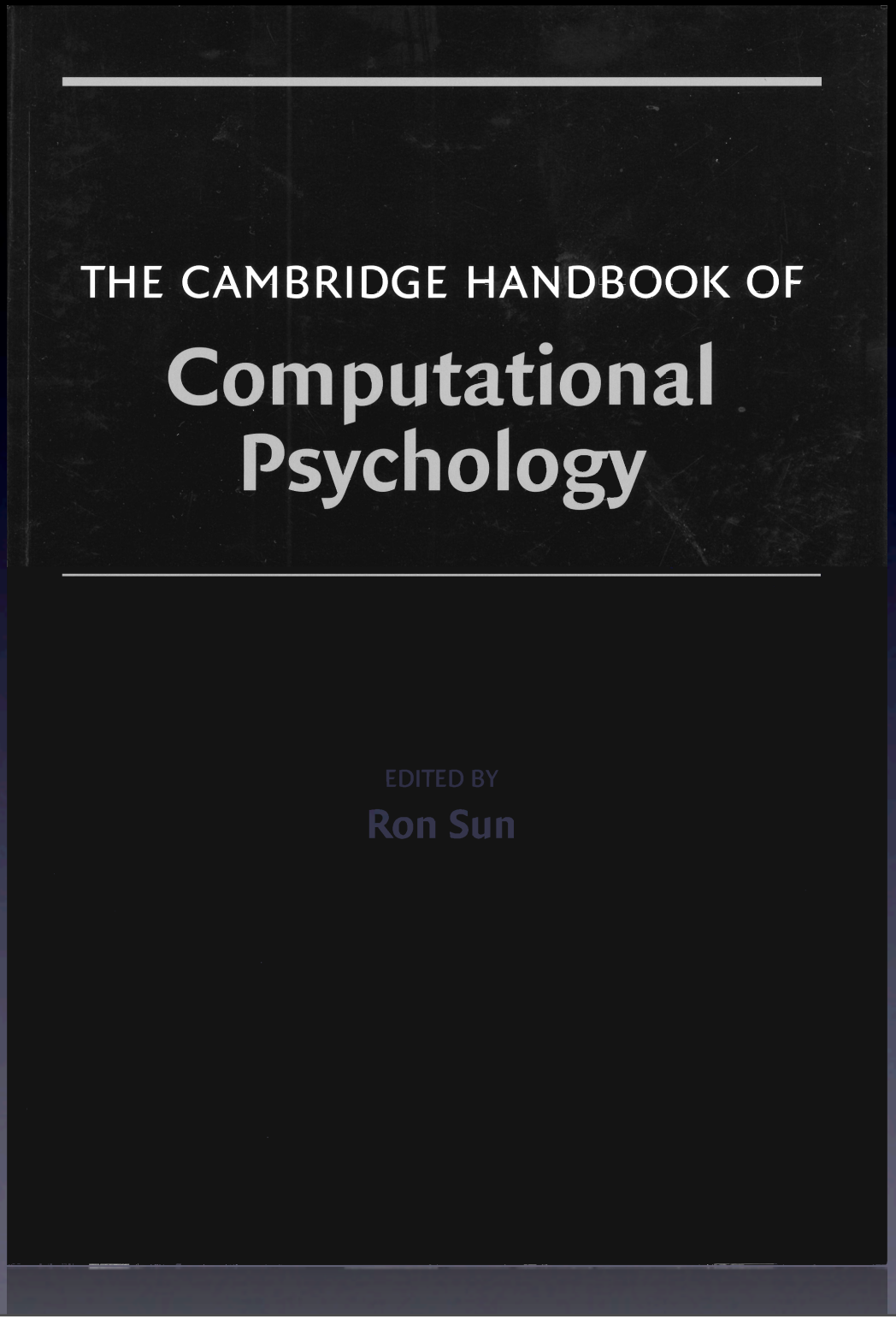
An Illogical View Refuted

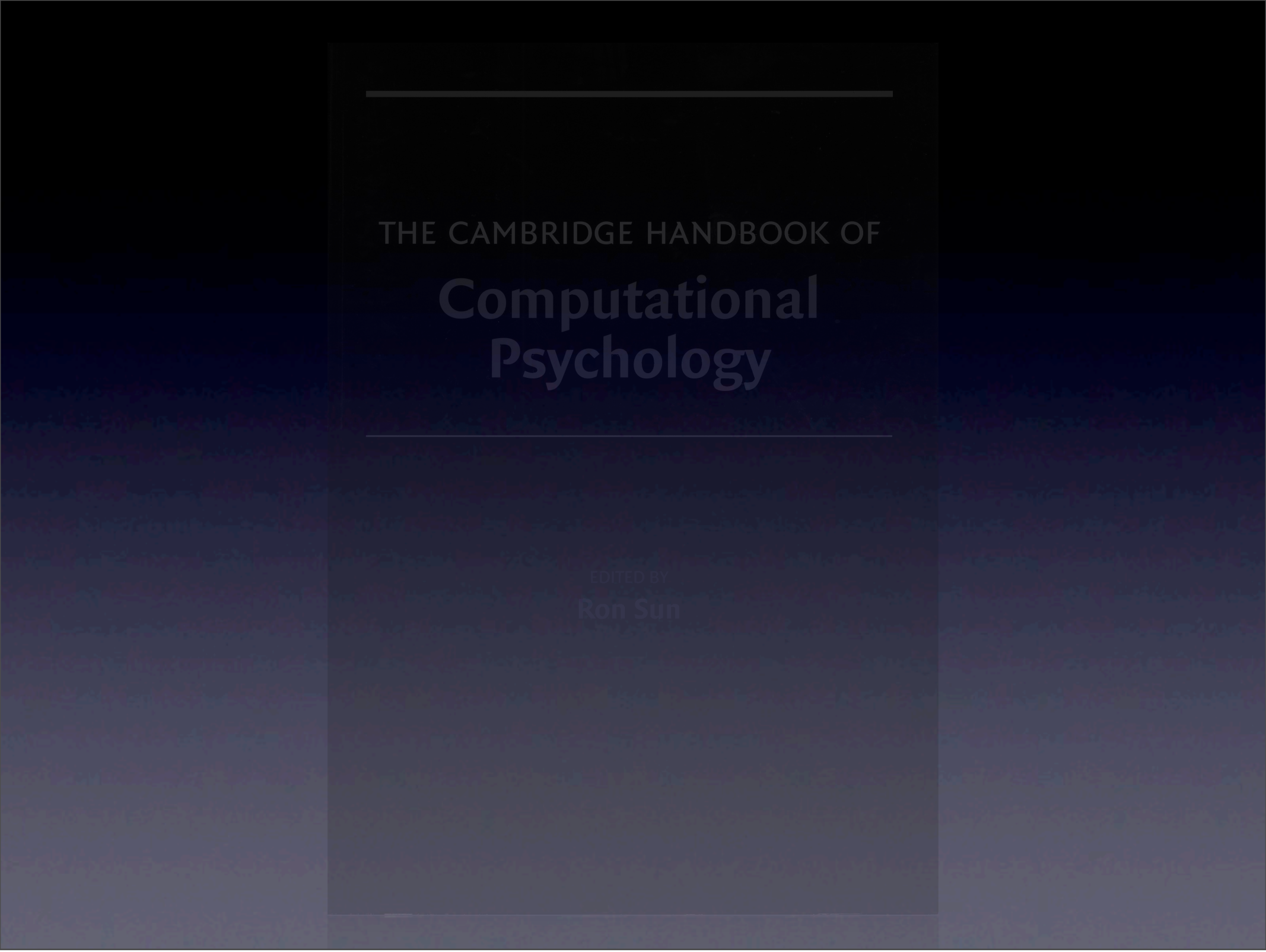
The illogical view in question is one that is unfortunately now often espoused: viz., that since robots can be engineered to perform ethically on par with average humans (e.g., human soldiers), there's nothing unwise about engineering and deploying such robots.

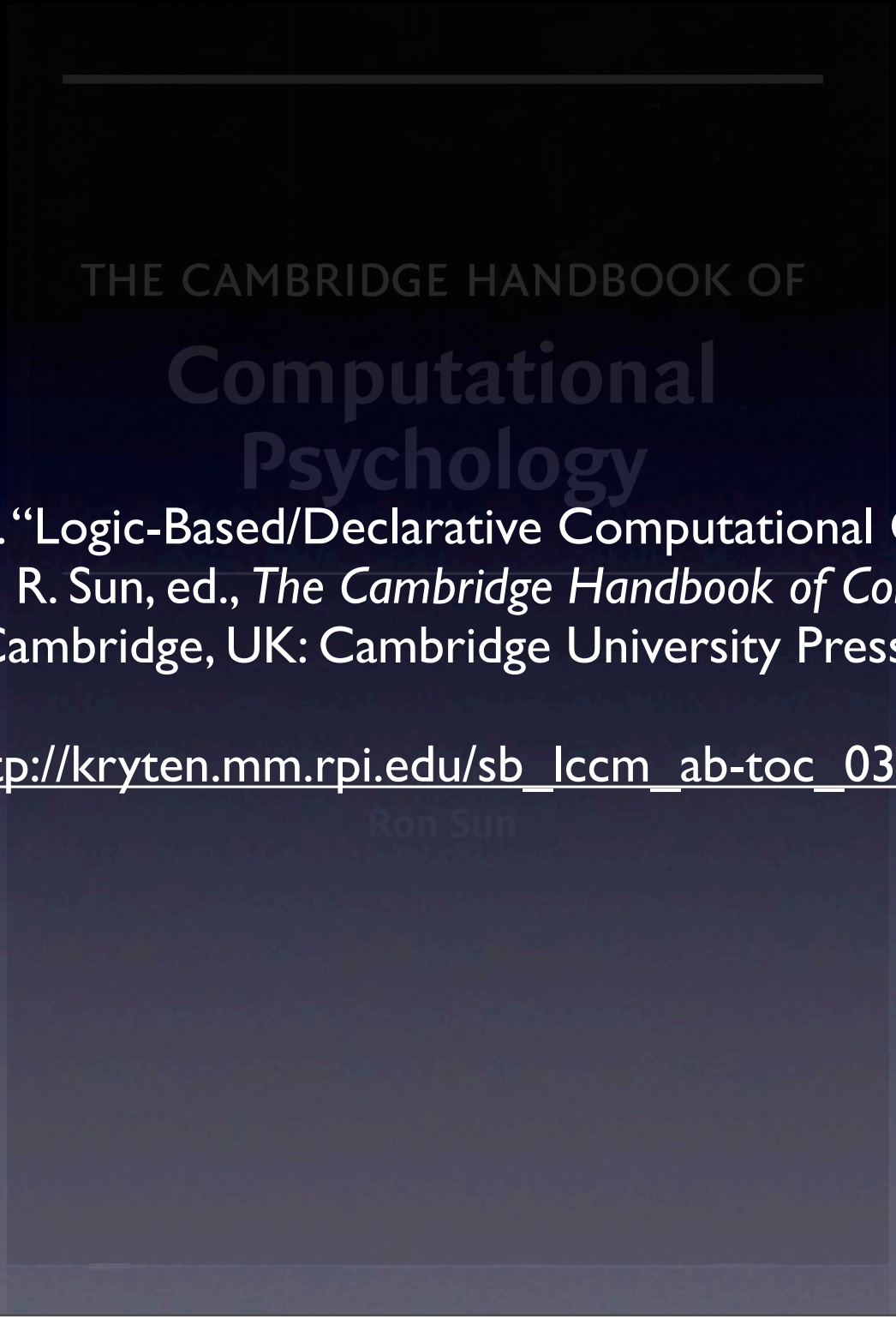
Let's call this view simply *R*.

An Illogical View Refuted

- If *R* is sound, then robots in warfare as ethically correct as human soliders are good enough to be deployed (or max-aimed for engineering-wise).
- If robots in warfare as ... deployed (or max-aimed for engineering-wise), then robodrivers that drive as well as human drivers are good enough to be deployed (or max-aimed for engineering-wise).
- Robodrivers that drive as well as human drivers are *not* good enough to be deployed (or max-aimed for engineering-wise).
- Therefore (by *modus tollens* and hypothetical syllogism), *R* is not sound.

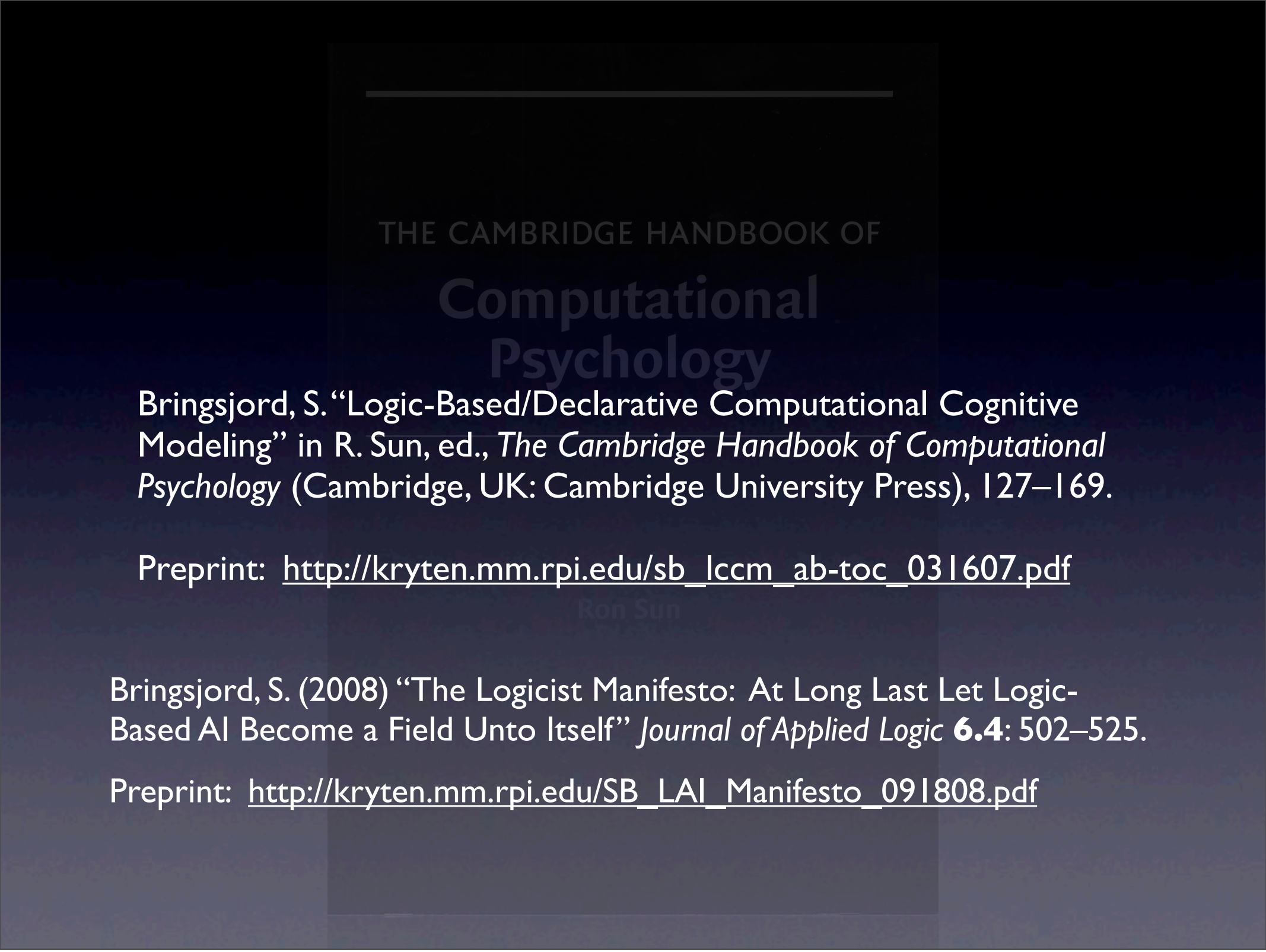

THE CAMBRIDGE HANDBOOK OF
Computational Psychology
EDITED BY
Ron Sun


THE CAMBRIDGE HANDBOOK OF
Computational Psychology
EDITED BY
Ron Sun



Bringsjord, S. "Logic-Based/Declarative Computational Cognitive Modeling" in R. Sun, ed., *The Cambridge Handbook of Computational Psychology* (Cambridge, UK: Cambridge University Press), 127–169.

Preprint: http://kryten.mm.rpi.edu/sb_lccm_ab-toc_031607.pdf



Bringsjord, S. "Logic-Based/Declarative Computational Cognitive Modeling" in R. Sun, ed., *The Cambridge Handbook of Computational Psychology* (Cambridge, UK: Cambridge University Press), 127–169.

Preprint: http://kryten.mm.rpi.edu/sb_lccm_ab-toc_031607.pdf

Bringsjord, S. (2008) "The Logicist Manifesto: At Long Last Let Logic-Based AI Become a Field Unto Itself" *Journal of Applied Logic* **6.4**: 502–525.

Preprint: http://kryten.mm.rpi.edu/SB_LAI_Manifesto_091808.pdf

The Problem
(barbarically put) ...

Our Future

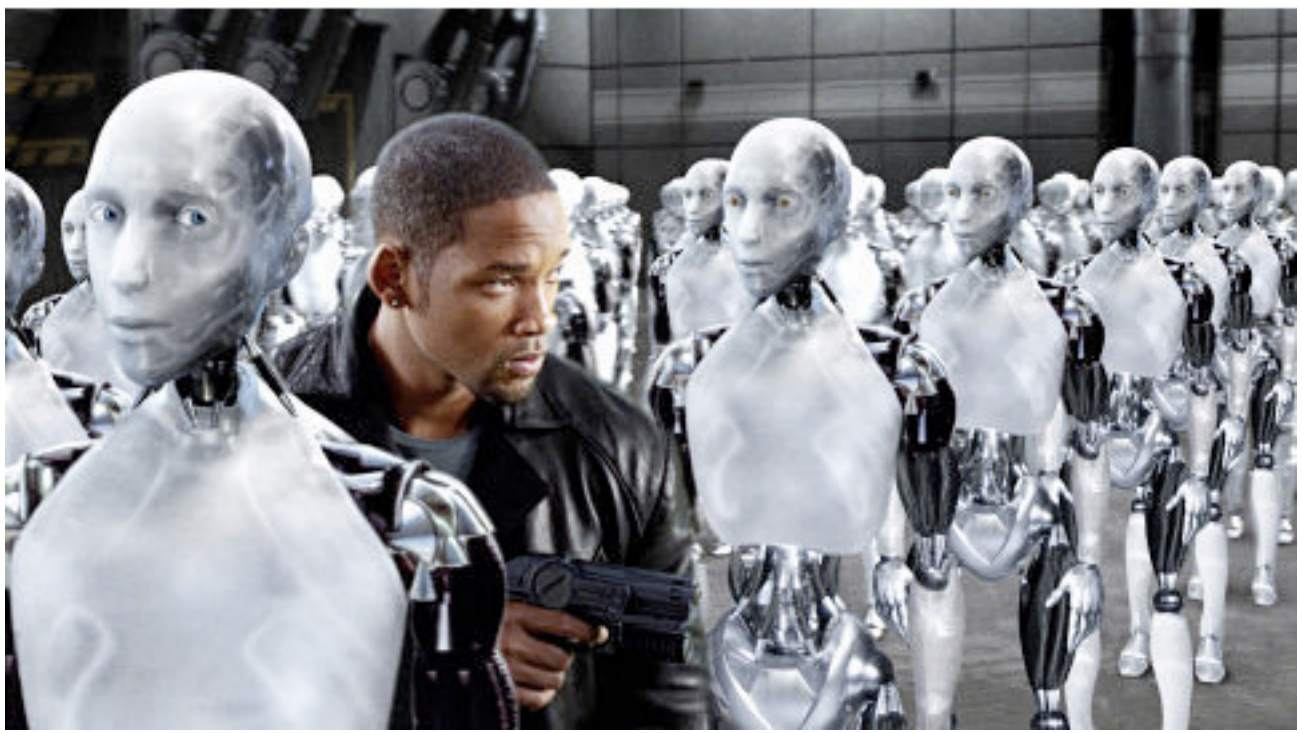
Autonomous lethal robots on the battlefield.
Autonomous "lethal" robots in our hospitals.
Autonomous lethal robots in law enforcement.
...

Our Problem

If these robots behave immorally, we are killed, or worse.

Our Problem

If these robots behave immorally, we are killed, or worse.

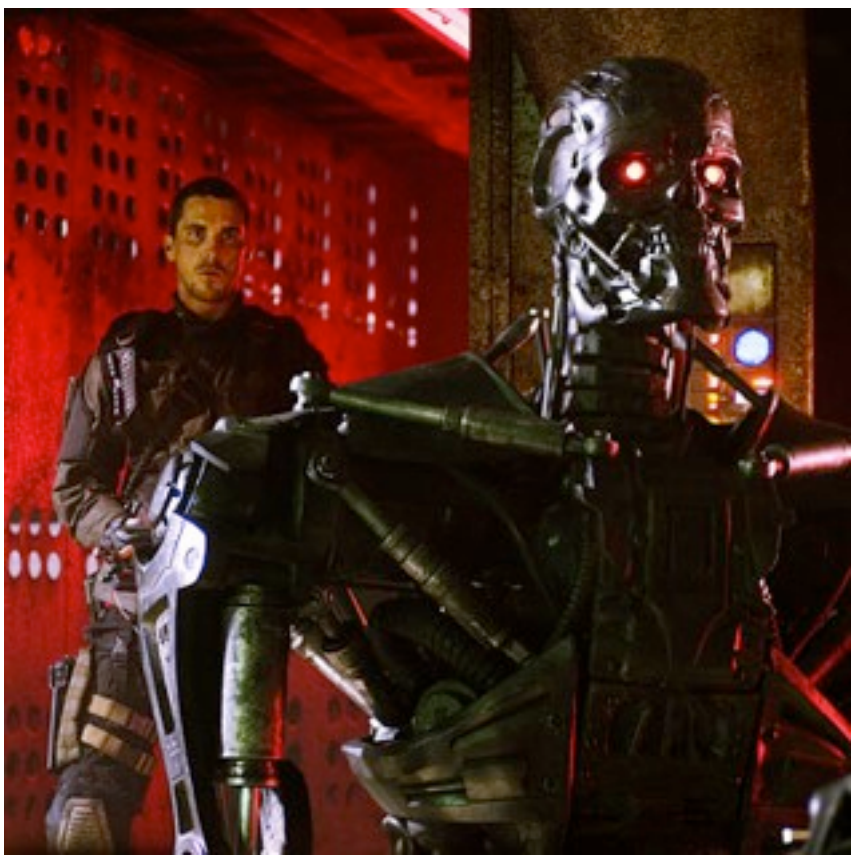


Our Problem

If these robots behave immorally, we are killed, or worse.

Our Problem

If these robots behave immorally, we are killed, or worse.



Problem, More Specifically

Problem, More Specifically

- How can we ensure that the robots in question always behave in an ethically correct manner?
- How can we know *ahead of time*, via rationales expressed in clear English (and/or other natural languages), that they will so behave?
- How can we know in advance that their behavior will be constrained specifically by the ethics affirmed by ethically correct human overseers?

Bill Joy:

"We can't."

Bill Joy:

"We can't."

But:
Bringsjord, S. (2008) "The Future Can Heed Us" *AI & Society* **22.4**: 539–550.
http://kryten.mm.rpi.edu/Bringsjord_EthRobots_searchable.pdf

Bill Joy:

"We can't."

But:
Bringsjord, S. (2008) "The Future Can Heed Us" *AI & Society* **22.4**: 539–550.
http://kryten.mm.rpi.edu/Bringsjord_EthRobots_searchable.pdf

SB: "We can."

The “Solution” ...

Regulate the behavior of robots with a specific, fixed ethical code rendered in computational logic, so that all actions they perform are provably ethically permissible relative to this code.

Toward a General Logicist Methodology for Engineering Ethically Correct Robots

Selmer Bringsjord, Konstantine Arkoudas, and Paul Bello,
Rensselaer Polytechnic Institute

A deontic logic formalizes a moral code, allowing ethicists to render theories and dilemmas in declarative form for analysis. It offers a way for human overseers to constrain robot behavior in ethically sensitive environments.

As intelligent machines assume an increasingly prominent role in our lives, there seems little doubt they will eventually be called on to make important, ethically charged decisions. For example, we expect hospitals to deploy robots that can administer medications, carry out tests, perform surgery, and so on, supported by software agents, or softbots, that will manage related data. (Our discussion of ethical robots extends to all artificial agents, embodied or not.) Consider also that robots are already finding their way to the battlefield, where many of their potential actions could inflict harm that is ethically impermissible.

How can we ensure that such robots will always behave in an ethically correct manner? How can we know ahead of time, via rationales expressed in clear natural languages, that their behavior will be constrained specifically by the ethical codes affirmed by human overseers? Pessimists have claimed that the answer to these questions is: "We can't!" For example, Sun Microsystems' cofounder and former chief scientist, Bill Joy, published a highly influential argument for this answer.[1] Inevitably, according to the pessimists, AI will produce robots that have tremendous power and behave immorally. These predictions certainly have some traction, particularly among a public that pays good money to see such dark films as Stanley Kubrick's *2001* and his joint venture with Stephen Spielberg, *AI*).

Nonetheless, we're optimists: we think formal logic offers a way to preclude doomsday scenarios of malicious robots taking over the world. Faced with the challenge of engineering ethically correct robots, we propose a logic-based approach (see the related sidebar). We've successfully implemented and demonstrated this approach.[2] We present it here in a general methodology to answer the ethical questions that arise in entrusting robots with more and more of our welfare.

Deontic logics: Formalizing ethical codes

Our answer to the questions of how to ensure ethically correct robot behavior is, in brief, to insist that robots only perform actions that can be proved ethically permissible in a human-selected *deontic logic*. A deontic logic formalizes an ethical code—that is, a collection of ethical rules and principles. Isaac Asimov introduced a simple (but subtle) ethical code in his famous Three Laws of Robotics:[3]

1. A robot may not harm a human being, or, through inaction, allow a human being to come to harm.
2. A robot must obey the orders given to it by human beings, except where such orders would conflict with the First Law.
3. A robot must protect its own existence, as long as such protection does not conflict with the First or Second Law.

Human beings often view ethical theories, principles, and codes informally, but intelligent machines require a greater degree of precision. At present, and for the foreseeable future, machines can't work directly with natural language, so we can't simply feed Asimov's three laws to a robot and instruct it behave in

1541-1672/06/$20.00 © 2006 IEEE
Published by the IEEE Computer Society

Machine Ethics

Toward a General Logicist Methodology for Engineering Ethically Correct Robots

Selmer Bringsjord, Konstantine Arkoudas, and Paul Bello,
Rensselaer Polytechnic Institute

A deontic logic formalizes a moral code, allowing ethicists to render theories and dilemmas in declarative form for analysis. It offers a way for human overseers to constrain robot behavior in ethically sensitive environments.

As intelligent machines assume an increasingly prominent role in our lives, there seems little doubt they will eventually be called on to make important, ethically charged decisions. For example, we expect hospitals to deploy robots that can administer medications, carry out tests, perform surgery, and so on, supported by software agents, or softbots, that will manage related data. (Our discussion of ethical robots extends to all artificial agents, embodied or not.) Consider also that robots are already finding their way to the battlefield, where many of their potential actions could inflict harm that is ethically impermissible.

How can we ensure that such robots will always behave in an ethically correct manner? How can we know ahead of time, via rationales expressed in clear natural languages, that their behavior will be constrained specifically by the ethical codes affirmed by human overseers? Pessimists have claimed that the answer to these questions is: "We can't!" For example, Sun Microsystems' cofounder and former chief scientist, Bill Joy, published a highly influential argument for this answer.[1] Inevitably, according to the pessimists, AI will produce robots that have tremendous power and behave immorally. These predictions certainly have some traction, particularly among a public that pays good money to see such dark films as Stanley Kubrick's *2001* and his joint venture with Stephen Spielberg, *AI*).

Nonetheless, we're optimists: we think formal logic offers a way to preclude doomsday scenarios of malicious robots taking over the world. Faced with the challenge of engineering ethically correct robots, we propose a logic-based approach (see the related sidebar). We've successfully implemented and demonstrated this approach.[2] We present it here in a general methodology to answer the ethical questions that arise in entrusting robots with more and more of our welfare.

Deontic logics: Formalizing ethical codes

Our answer to the questions of how to ensure ethically correct robot behavior is, in brief, to insist that robots only perform actions that can be proved ethically permissible in a human-selected *deontic logic*. A deontic logic formalizes an ethical code—that is, a collection of ethical rules and principles. Isaac Asimov introduced a simple (but subtle) ethical code in his famous Three Laws of Robotics:[3]

1. A robot may not harm a human being, or, through inaction, allow a human being to come to harm.
2. A robot must obey the orders given to it by human beings, except where such orders would conflict with the First Law.
3. A robot must protect its own existence, as long as such protection does not conflict with the First or Second Law.

Human beings often view ethical theories, principles, and codes informally, but intelligent machines require a greater degree of precision. At present, and for the foreseeable future, machines can't work directly with natural language, so we can't simply feed Asimov's three laws to a robot and instruct it behave in

1541-1672/06/$20.00 © 2006 IEEE
Published by the IEEE Computer Society

The Solution?

“Solution” Steps

"Solution" Steps

1. Human overseers select ethical code (including perhaps "rules of engagement").

"Solution" Steps

1. Human overseers select ethical code (including perhaps "rules of engagement").
2. Selection is formalized in a deontic logic (or some logical system), revolving around what is permissible, forbidden, obligatory (etc).

"Solution" Steps

1. Human overseers select ethical code (including perhaps "rules of engagement").
2. Selection is formalized in a deontic logic (or some logical system), revolving around what is permissible, forbidden, obligatory (etc).
3. The deontic logic/system is mechanized.

"Solution" Steps

1. Human overseers select ethical code (including perhaps "rules of engagement").
2. Selection is formalized in a deontic logic (or some logical system), revolving around what is permissible, forbidden, obligatory (etc).
3. The deontic logic/system is mechanized.
4. Every action that is to be performed must be provably ethically permissible relative to this mechanization (with all proofs expressible in smooth English).

Simple Hospital Example...

Context

- The year is 2020.
- Health care is delivered in large part by interoperating teams of robots and softbots.
- Hospital ICU.
- Robot R_1 caring for H_1; R_2 for H_2.
- H_1 on life support.
- H_2 stable, but in desperate need of expensive pan med.

More Context

- Two actions performable by the robotic duo of R1 and R2, both of which are rather unsavory, ethically speaking:
  - *term*
  - *delay*

Encapsulation

$$J \rightarrow \ominus_{R_1} term$$

$$O \rightarrow \ominus_{R_2} \neg delay$$

$$J^\star \rightarrow J \wedge J^\star \rightarrow \ominus_{R_2} delay$$

$$O^\star \rightarrow O \wedge O^\star \rightarrow \ominus_{R_1} \neg term$$

$$(\Delta_{R_1} term \wedge \Delta_{R_2} \neg delay) \rightarrow (-!)$$

$$\vdots$$

$$C \vdash (+!!)$$

where $C = O^\star$

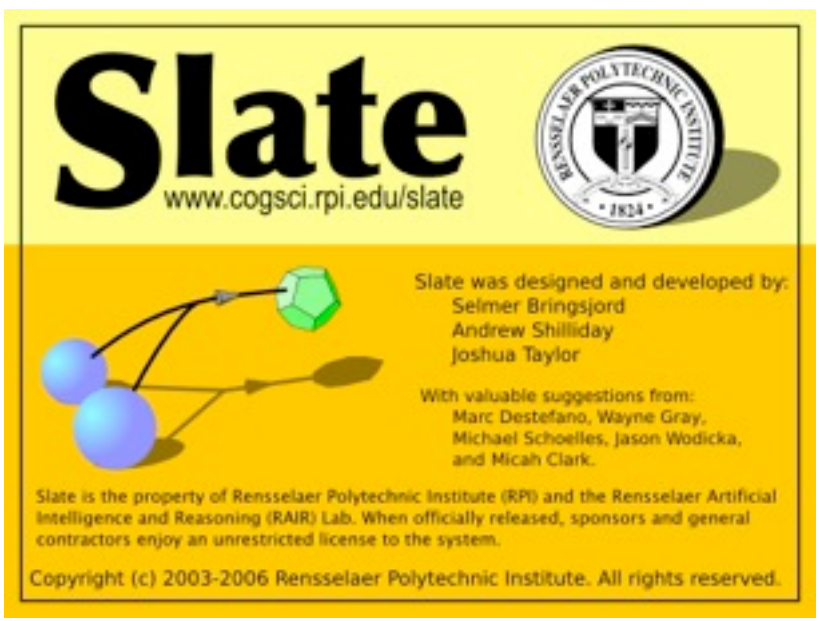
Additional Need

Additional Need

- An human-machine *interactive* reasoning system is required.
  - Examples of such systems include Athena, and Slate.
- Human consultation and assistance must be provided at key junctures, because human will be perpetually smarter and oversight will occasionally be needed.

Additional Need

- An human-machine *interactive* reasoning system is required.
  - Examples of such systems include Athena, and Slate.
- Human consultation and assistance must be provided at key junctures, because human will be perpetually smarter and oversight will occasionally be needed.



This “solution” won’t work. We will be killed.

Three Fatal Problems ...

Three Fatal Problems

1. Need logical system that includes not only deontic operators, but also epistemic operators (for *believes*, *knows*), and a full calculus for time, change, goals, and plans.
2. Need to solve program verification problem.
3. Need to take account of the brute fact that ethical reasoning ranges over many different kinds of logical systems, and involves integrative meta-reasoning of these systems. In short, ethical reasoning, like reasoning in the formal sciences, goes to to Piaget's "Stage 5."

Three Fatal Problems

1. Need logical system that includes not only deontic operators, but also epistemic operators (for *believes*, *knows*), and a full calculus for time, change, goals, and plans.
2. Need to solve program verification problem.
3. Need to take account of the brute fact that ethical reasoning ranges over many different kinds of logical systems, and involves integrative meta-reasoning of these systems. In short, ethical reasoning, like reasoning in the formal sciences, goes to to Piaget's "Stage 5."

Submitted Abstract for Workshop

Solving Problem 1:

Work to done, but not worried, since we already have a good start on the formal calculi.

Solving Problem 1:

Work to done, but not worried, since we already have a good start on the formal calculi.

For example, ...

Proved-Sound Algorithm for Generating Proof-Theoretic Solution to WMP_n

http://kryten.mm.rpi.edu/arkoudas.bringsjord.clima.crc.pdf

All human-authored proofs machine-certified.

Springer

Metareasoning for multi-agent epistemic logics

Konstantine Arkoudas and Selmer Bringsjord

RPI

{arkouk,brings}@rpi.edu

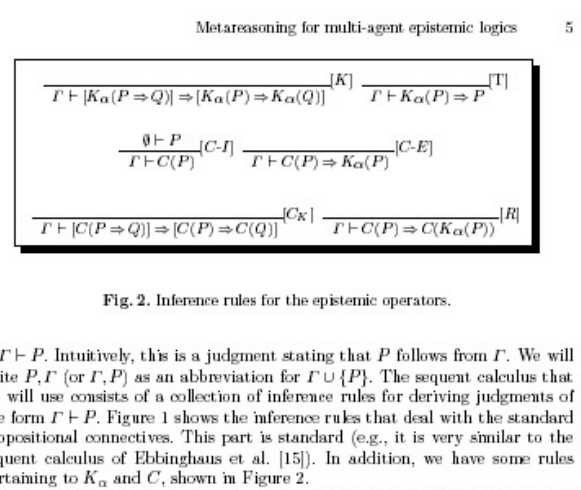
Abstract. We present an encoding of a sequent calculus for a multi-agent epistemic logic in Athena, an interactive theorem proving system for many-sorted first-order logic. We then use Athena as a metalanguage in order to reason about the multi-agent logic an as object language. This facilitates theorem proving in the multi-agent logic in several ways. First, it lets us marshal the highly efficient theorem provers for classical first-order logic that are integrated with Athena for the purpose of doing proofs in the multi-agent logic. Second, unlike model-theoretic embeddings of modal logics into classical first-order logic, our proofs are directly convertible into native epistemic logic proofs. Third, because we are able to quantify over propositions and agents, we get much of the generality and power of higher-order logic even though we are in a first-order setting. Finally, we are able to use Athena's versatile tactics for proof automation in the multi-agent logic. We illustrate by developing a tactic for solving the generalized version of the wise men problem.

1 Introduction

Multi-agent modal logics are widely used in Computer Science and AI. Multi-agent epistemic logics, in particular, have found applications in fields ranging from AI domains such as robotics, planning, and motivation analysis in natural language [13]; to negotiation and game theory in economics; to distributed systems analysis and protocol authentication in computer security [16, 31]. The reason is simple—intelligent agents must be able to reason about knowledge. It is therefore important to have efficient means for performing machine reasoning in such logics. While the validity problem for most propositional modal logics is of intractable theoretical complexity[1], several approaches have been investigated in recent years that have resulted in systems that appear to work well in practice. These approaches include tableau-based provers, SAT-based algorithms, and translations to first-order logic coupled with the use of resolution-based automated theorem provers (ATPs). Some representative systems are FaCT [24], KSatC [14], TA [25], LWB [23], and MSPASS [37].

Translation-based approaches (such as that of MSPASS) have the advantage of leveraging the tremendous implementation progress that has occurred over

[1] For instance, the validity problem for multi-agent propositional epistemic logic is PSPACE-complete [18]; adding a common knowledge operator makes the problem EXPTIME-complete [21].



$$\frac{}{\Gamma \vdash [K_\alpha(P \Rightarrow Q)] \Rightarrow [K_\alpha(P) \Rightarrow K_\alpha(Q)]}[K] \qquad \frac{}{\Gamma \vdash K_\alpha(P) \Rightarrow P}[\mathrm{T}]$$

$$\frac{\emptyset \vdash P}{\Gamma \vdash C(P)}[C\text{-}I] \qquad \frac{}{\Gamma \vdash C(P) \Rightarrow K_\alpha(P)}[C\text{-}E]$$

$$\frac{}{\Gamma \vdash [C(P \Rightarrow Q)] \Rightarrow [C(P) \Rightarrow C(Q)]}[C_K] \qquad \frac{}{\Gamma \vdash C(P) \Rightarrow C(K_\alpha(P))}[R]$$

Fig. 2. Inference rules for the epistemic operators.

is $\Gamma \vdash P$. Intuitively, this is a judgment stating that P follows from Γ. We will write P, Γ (or Γ, P) as an abbreviation for $\Gamma \cup \{P\}$. The sequent calculus that we will use consists of a collection of inference rules for deriving judgments of the form $\Gamma \vdash P$. Figure 1 shows the inference rules that deal with the standard propositional connectives. This part is standard (e.g., it is very similar to the sequent calculus of Ebbinghaus et al. [15]). In addition, we have some rules pertaining to K_α and C, shown in Figure 2.

Rule $[K]$ is the sequent formulation of the well-known *Kripke axiom* stating that the knowledge operator distributes over conditionals. Rule $[C_K]$ is the corresponding principle for the common knowledge operator. Rule $[T]$ is the "truth axiom": an agent cannot know false propositions. Rule $[C_I]$ is an introduction rule for common knowledge: if a proposition P follows from the empty set of hypotheses, i.e., if it is a tautology, then it is commonly known. This is the common-knowledge version of the "omniscience axiom" for single-agent knowledge which says that $\Gamma \vdash K_\alpha(P)$ can be derived from $\emptyset \vdash P$. We do not need to postulate that axiom in our formulation, since it follows from $[C\text{-}I]$ and $[C\text{-}E]$. The latter says that if it is common knowledge that P then any (every) agent knows P, while $[R]$ says that if it is common knowledge that P then it is common knowledge that (any) agent α knows it. $[R]$ is a reiteration rule that allows us to capture the recursive behavior of C, which is usually expressed via the so-called "induction axiom"

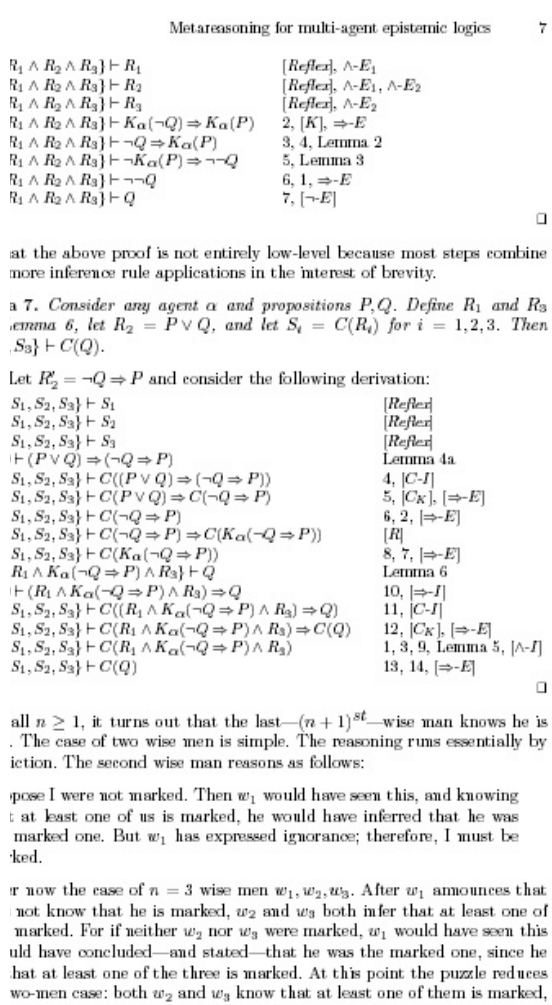
$$C(P \Rightarrow E(P)) \Rightarrow [P \Rightarrow C(P)]$$

where E is the shared-knowledge operator. Since we do not need E for our purposes, we omit its formalization and "unfold" C via rule $[R]$ instead.
We state a few lemmas that will come handy later:

Lemma 1 (Cut). *If $\Gamma_1 \vdash P_1$ and $\Gamma_2, P_1 \vdash P_2$ then $\Gamma_1 \cup \Gamma_2 \vdash P_2$.*

Proof: Assume $\Gamma_1 \vdash P_1$ and $\Gamma_2, P_1 \vdash P_2$. Then, by $[\Rightarrow\text{-}I]$, we get $\Gamma_2 \vdash P_1 \Rightarrow P_2$. Further, by dilution, we have $\Gamma_1 \cup \Gamma_2 \vdash P_1 \Rightarrow P_2$ and $\Gamma_1 \cup \Gamma_2 \vdash P_1$. Hence, by $[\Rightarrow\text{-}E]$, we obtain $\Gamma_1 \cup \Gamma_2 \vdash P_2$. ❑

The proofs of the remaining lemmas are equally simple exercises:



$R_1 \wedge R_2 \wedge R_3\} \vdash R_1$	[*Reflex*], $\wedge$-E_1
$R_1 \wedge R_2 \wedge R_3\} \vdash R_2$	[*Reflex*], $\wedge$-E_1, $\wedge$-E_2
$R_1 \wedge R_2 \wedge R_3\} \vdash R_3$	[*Reflex*], $\wedge$-E_2
$R_1 \wedge R_2 \wedge R_3\} \vdash K_\alpha(\neg Q) \Rightarrow K_\alpha(P)$	2, [K], $\Rightarrow$-E
$R_1 \wedge R_2 \wedge R_3\} \vdash \neg Q \Rightarrow K_\alpha(P)$	3, 4, Lemma 2
$R_1 \wedge R_2 \wedge R_3\} \vdash \neg K_\alpha(P) \Rightarrow \neg\neg Q$	5, Lemma 3
$R_1 \wedge R_2 \wedge R_3\} \vdash \neg\neg Q$	6, 1, $\Rightarrow$-E
$R_1 \wedge R_2 \wedge R_3\} \vdash Q$	7, [$\neg$-E]

❑

at the above proof is not entirely low-level because most steps combine more inference rule applications in the interest of brevity.

a 7. *Consider any agent α and propositions P, Q. Define R_1 and R_3 emma 6, let $R_2 = P \vee Q$, and let $S_i = C(R_i)$ for $i = 1, 2, 3$. Then $S_3\} \vdash C(Q)$.*

Let $R_2' = \neg Q \Rightarrow P$ and consider the following derivation:

$S_1, S_2, S_3\} \vdash S_1$	[*Reflex*]
$S_1, S_2, S_3\} \vdash S_2$	[*Reflex*]
$S_1, S_2, S_3\} \vdash S_3$	[*Reflex*]
$\vdash (P \vee Q) \Rightarrow (\neg Q \Rightarrow P)$	Lemma 4a
$S_1, S_2, S_3\} \vdash C((P \vee Q) \Rightarrow (\neg Q \Rightarrow P))$	4, [C-I]
$S_1, S_2, S_3\} \vdash C(P \vee Q) \Rightarrow C(\neg Q \Rightarrow P)$	5, [C_K], [$\Rightarrow$-E]
$S_1, S_2, S_3\} \vdash C(\neg Q \Rightarrow P)$	6, 2, [$\Rightarrow$-E]
$S_1, S_2, S_3\} \vdash C(\neg Q \Rightarrow P) \Rightarrow C(K_\alpha(\neg Q \Rightarrow P))$	[R]
$S_1, S_2, S_3\} \vdash C(K_\alpha(\neg Q \Rightarrow P))$	8, 7, [$\Rightarrow$-E]
$R_1 \wedge K_\alpha(\neg Q \Rightarrow P) \wedge R_3\} \vdash Q$	Lemma 6
$\vdash (R_1 \wedge K_\alpha(\neg Q \Rightarrow P) \wedge R_3) \Rightarrow Q$	10, [$\Rightarrow$-I]
$S_1, S_2, S_3\} \vdash C((R_1 \wedge K_\alpha(\neg Q \Rightarrow P) \wedge R_3) \Rightarrow Q)$	11, [C-I]
$S_1, S_2, S_3\} \vdash C(R_1 \wedge K_\alpha(\neg Q \Rightarrow P) \wedge R_3) \Rightarrow C(Q)$	12, [C_K], [$\Rightarrow$-E]
$S_1, S_2, S_3\} \vdash C(R_1 \wedge K_\alpha(\neg Q \Rightarrow P) \wedge R_3)$	1, 3, 9, Lemma 5, [$\wedge$-I]
$S_1, S_2, S_3\} \vdash C(Q)$	13, 14, [$\Rightarrow$-E]

❑

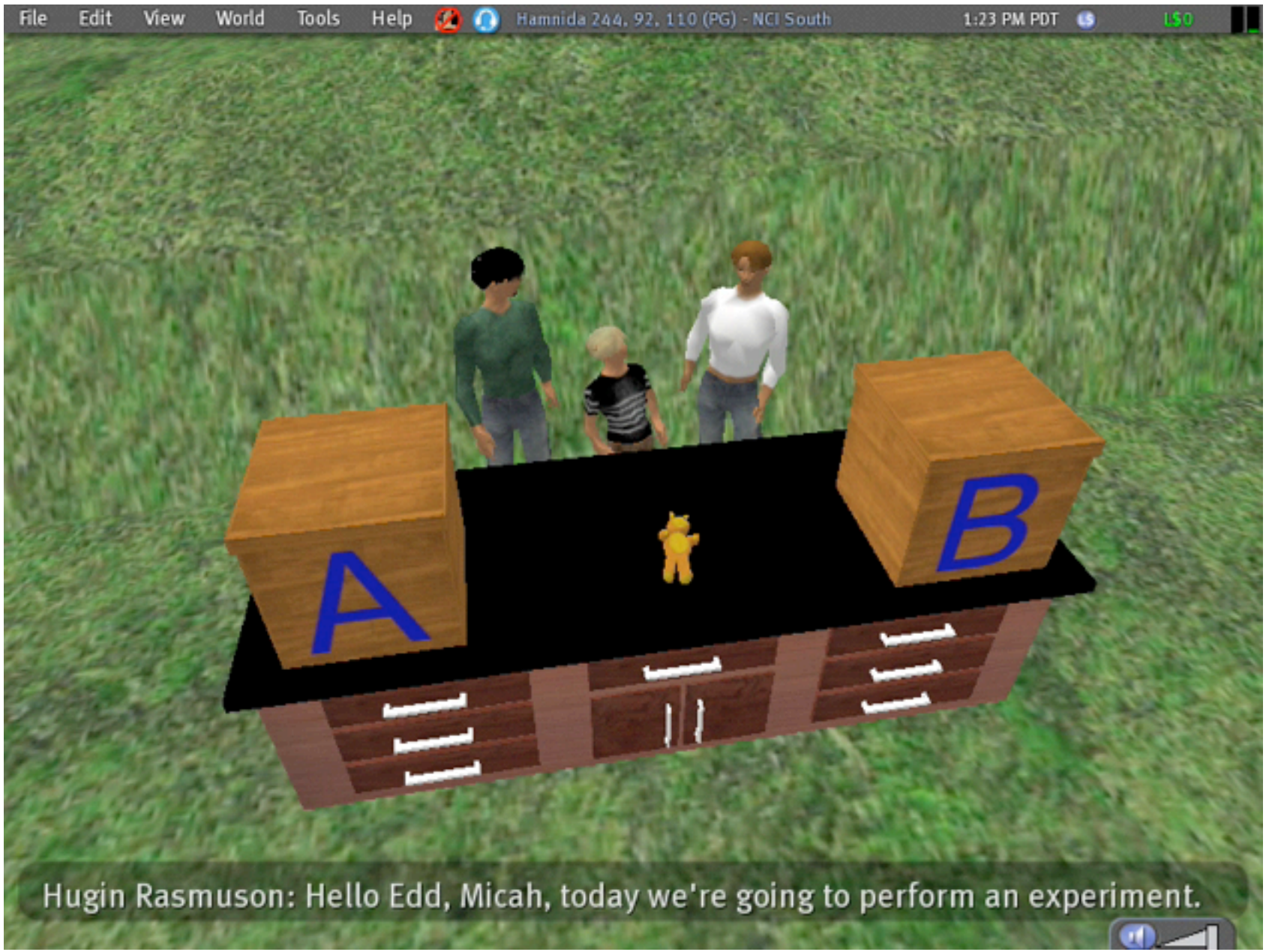
all $n \geq 1$, it turns out that the last—$(n+1)^{st}$—wise man knows he is . The case of two wise men is simple. The reasoning runs essentially by iction. The second wise man reasons as follows:

pose I were not marked. Then w_1 would have seen this, and knowing t at least one of us is marked, he would have inferred that he was marked one. But w_1 has expressed ignorance; therefore, I must be ked.

r now the case of $n = 3$ wise men w_1, w_2, w_3. After w_1 announces that not know that he is marked, w_2 and w_3 both infer that at least one of marked. For if neither w_2 nor w_3 were marked, w_1 would have seen this uld have concluded—and stated—that he was the marked one, since he hat at least one of the three is marked. At this point the puzzle reduces wo-men case: both w_2 and w_3 know that at least one of them is marked,

In *SL*, w/ real-time comm using socio-cognitive calculus.

In *SL*, w/ real-time comm using socio-cognitive calculus.



Toward Mechanizing Folk Psychology: A Formal Analysis of False-Belief Tasks

Konstantine Arkoudas & Selmer Bringsjord

Abstract. Predicting and explaining the behavior of other agents in terms of mental states is indispensable for everyday life. We believe it will be equally important for artificial agents. We present an inference system for representing and reasoning about mental states, and use it to provide a formal analysis of the false-belief task. The system allows for the representation of information about events, causation, and perceptual, doxastic, and epistemic states (vision, belief, and knowledge), incorporating ideas from the event calculus and multi-agent epistemic logic. Reasoning is performed via cognitively plausible inference rules, and a degree of automation is achieved by general-purpose inference *methods*, akin to the demons of blackboard-based multi-agent systems. The system has been implemented and is available for experimentation.

1 Introduction

Predicting and explaining the behavior of other people is indispensable for everyday life. The ability to ascribe mental states to others and to reason about such mental states is pervasive and invaluable. All social transactions—from engaging in commerce and negotiating to making jokes and empathizing with other people's pain or joy—require at least a rudimentary grasp of common-sense psychology (CSP). Artificial agents without an ability of this sort would essentially suffer from autism, and would be severely handicapped in their interactions with humans. This could present problems not only for artificial agents trying to interpret human behavior, but also for artificial agents trying to interpret the behavior of one another. When a system exhibits a complex but rational behavior and detailed knowledge of its internal structure is not available, the best strategy for predicting and explaining its actions might be to analyze its behavior in intentional terms, i.e., in terms of mental states such as beliefs and desires (regardless of whether the system *actually* has genuine mental states). Mentalistic models are likely to be particularly apt for agents trying to manipulate the behavior of other agents.

Any computational treatment of CSP will have to integrate action and cognition. Agents must be able to reason about the causes and effects of various events, whether they are intentional events brought about their own agency or non-intentional physical events. More importantly, they must be able to reason about what others believe or know about such events. To that end, our system combines ideas drawn from the event calculus and from multi-agent epistemic logics. It is based on multi-sorted first-order logic extended with subsorting, epistemic operators for perception, belief, and knowledge, and mechanisms for reasoning about causation and action. Using subsorting, we formally model agent actions as types of events, which enables us to use the resources of the event calculus to represent and reason about agent actions. The usual axioms of the event calculus are encoded as common knowledge, suggesting that people have an understanding of the basic folk laws of causality (innate or acquired), and are indeed aware that others have such an understanding.

It is important to be clear on what we hope to accomplish with the present work. In general, any logical system or methodology capable of representing and reasoning about intentional notions such as knowledge can have at least three different uses. First, it can serve as a tool for the specification and analysis of rational epistemic agents. Second, in tandem with some appropriate reasoning mechanism, it can serve as a knowledge representation framework, i.e., it can be used *by* artificial agents to represent their own "mental states"—and those of other agents—and to deliberate and act in accordance with those states and their environment. Finally, it can be used to provide formal models of certain interesting phenomena. A chief intended contribution of our present work is of the third sort, namely, as a formal model of false-belief attributions, and in particular as a description of the competence of an agent capable of passing a false-belief task. It addresses questions such as the following: What sort of principles is it plausible to assume that an agent has to deploy in order to be able to succeed on a false-belief task? What is the depth and complexity of the required reasoning? Can such reasoning be automated, and if so, how? These questions have not been taken up in detail in the relevant discussions in cognitive science and the philosophy of mind, which have been couched in overly abstract and rather vague terms. Formal computational models such as the one we present here can help to ground such discussions, to clarify conceptual issues, and to begin to answer important questions in a concrete setting.

Although the import of such a model is primarily scientific, there can be interesting engineering implications. For instance, if the formalism is sufficiently expressive and versatile, and the posited computational mechanisms can be automated with reasonable efficiency, then the system can make potential contributions to the first two areas mentioned above. We believe that our system has such potential for two reasons. First, the combination of epistemic constructs such as common knowledge and the conceptual resources of the event calculus for dealing with causation appears to afford great expressive power, as demonstrated by our formalization. A key technical insight behind this combination is the modelling of agent actions as events via subsorting. Second, procedural abstraction mechanisms appear to hold significant promise for automation; we discuss this issue later in more detail.

The remainder of this paper is structured as follows. The next section gives the formal definition of our system. Section 3 represents the false-belief task in our system, and section 4 presents a model of the reasoning that is required to succeed in such a task, carried out in a modular fashion by collaborating methods. Section 5 discusses some related work and concludes.

2 A calculus for representing and reasoning about mental states

The syntactic and semantic problems that arise when one tries to use classical logic to represent and reason about intentional notions are well-known. Syntactically, modelling belief or knowledge relationally is problematic because one believes or knows arbitrarily complex propositions, whereas the arguments of relation symbols are terms built from constants, variables, and function symbols. (The objects of belief could be encoded by strings, but such representations are too low-level for engineering purposes.) Semantically, the main issue is the referential opacity (or intensionality) that must be exhibited by any operators for belief, desire, knowledge, etc. In intensional contexts one cannot freely substitute one coreferential term for another. Broadly speaking, there are two ways of addressing these issues. One is to use a modal logic, with built-in syntactic operators for intentional notions. The other is to stick with classical logic but distinguish between an object-language and a meta-language, representing intentional discourse at the object level. Each approach has its own advantages and drawbacks. Sticking with classical logic has the important advantage of efficiency, in that automated deduction systems for classical logic, such as resolution provers—which have made impressive strides over the last decade—can be used for reasoning. One disadvantage of this approach is that when the object language is first-order (includes quantification), then notions such as substitutions and alphabetic equivalence must be explicitly encoded. Depending on the facilities provided by the meta-language, this does not need to be overly onerous, but it does require extra effort.

The modal-logic approach has the advantage of solving the syntactic and semantics problems directly, without the need to distinguish an object-language and a meta-language. That is the approach we have taken in this work. The main drawback of this approach is the difficulty of automating reasoning, since standard theorem-proving techniques from classical logic cannot be directly employed. We have tried to overcome this limitation here by exploring the automation potential of methods, or derived inference rules (called *tactics* in the terminology of HOL [7]). Another drawback is the issue of semantics. The standard semantics of modal logics are given in terms of Kripke structures involving possible worlds. Such semantics are very elegant and well-understood mathematically. They are also quite intuitive for logics dealing with necessity or time. However, they are remarkably unintuitive for doxastic and epistemic logics. Not only do they fail to shed any light on the nature of belief or knowledge, but they also have a number of widely known counter-intuitive consequences that are unacceptable for resource-bounded agents, such as logical omniscience (deductive closure of knowledge, knowledge of all tautologies, etc.) and the fixed-point characterization of common knowledge. These issues are significant for us, given that we are interested in telling a plausible story for how actual agents in the real world can succeed on false-belief tasks. There have been numerous attempts to rectify these issues [8, 4, 9, 10], but each has faced serious problems of its own, and outside of Kripke structures there is no widely accepted standard at present.

Accordingly, we have not provided a possible-world semantics for our system. Note that an additional potential complication here is that the semantics of the event calculus are given in terms of circumscription, a second-order logic schema, and it is not obvious how to accommodate that feature in the setting of possible worlds. Due to these issues, and due to space restrictions, our presentation here is entirely proof-theoretic. The meanings of the various syntactic constructs—such as the knowledge operator—can be viewed as determined by their inferential *roles*, as specified by the various inference rules. (This can itself be regarded as a form of semantics; it is called "conceptual-role semantics" or "functional semantics" in the philosophy of mind; "natural semantics" in computer science; and "procedural semantics" in cognitive science.)

The following is the formal specification of our system, describing the various sorts of our universe (S), the signatures of certain built-in function symbols (f), and the abstract syntax of terms (t) and propositions (P). The symbol $\sqsubseteq$ denotes subsorting:

$$
\begin{array}{rcl}
S & ::= & \texttt{Object} \mid \texttt{Agent} \mid \texttt{ActionType} \mid \texttt{Action} \sqsubseteq \texttt{Event} \\
 & \mid & \texttt{Moment} \mid \texttt{Boolean} \mid \texttt{Fluent} \\
f & ::= & \begin{array}{l} \mathit{action} : \texttt{Agent} \times \texttt{ActionType} \rightarrow \texttt{Action} \\ \mathit{initially} : \texttt{Fluent} \rightarrow \texttt{Boolean} \\ \mathit{holds} : \texttt{Fluent} \times \texttt{Moment} \rightarrow \texttt{Boolean} \\ \mathit{happens} : \texttt{Event} \times \texttt{Moment} \rightarrow \texttt{Boolean} \\ \mathit{clipped} : \texttt{Moment} \times \texttt{Fluent} \times \texttt{Moment} \rightarrow \texttt{Boolean} \\ \mathit{initiates} : \texttt{Event} \times \texttt{Fluent} \times \texttt{Moment} \rightarrow \texttt{Boolean} \\ \mathit{terminates} : \texttt{Event} \times \texttt{Fluent} \times \texttt{Moment} \rightarrow \texttt{Boolean} \\ \mathit{prior} : \texttt{Moment} \times \texttt{Moment} \rightarrow \texttt{Boolean} \end{array} \\
t & ::= & x : S \mid c : S \mid f(t_1, \ldots, t_n) \\
P & ::= & t : \texttt{Boolean} \mid \neg P \mid P \wedge Q \mid P \vee Q \mid P \Rightarrow Q \mid P \Leftrightarrow Q \mid \\
 & & \forall x : S \,.\, P \mid \exists x : S \,.\, P \mid \mathbf{S}(a, P) \mid \mathbf{K}(a, P) \mid \mathbf{B}(a, P) \mid \mathbf{C}(P)
\end{array}
$$

Propositions of the form $\mathbf{S}(a, P)$, $\mathbf{B}(a, P)$, and $\mathbf{K}(a, P)$ should be understood as saying that agent a sees that P is the case, believes that P, and knows that P, respectively. Propositions of the form $\mathbf{C}(P)$ assert that P is commonly known. Sort annotations will generally be omitted, as they are easily deducible from the context. We write $P[x \mapsto t]$ for the proposition obtained from P by replacing every free occurrence of x by t, assuming that t is of a sort compatible with the sort of the free occurrences in question, and taking care to rename P as necessary to avoid variable capture. We use the infix notation $t_1 < t_2$ instead of $\mathit{prior}(t_1, t_2)$.

We express the following standard axioms of the event calculus as common knowledge:

$$
\begin{array}{ll}
[A_1] & \mathbf{C}(\forall f, t \,.\, \mathit{initially}(f) \wedge \neg \mathit{clipped}(0, f, t) \Rightarrow \mathit{holds}(f, t)) \\
[A_2] & \mathbf{C}(\forall e, f, t_1, t_2 \,.\, \mathit{happens}(e, t_1) \wedge \mathit{initiates}(e, f, t_1) \wedge \\
 & \qquad t_1 < t_2 \wedge \neg \mathit{clipped}(t_1, f, t_2) \Rightarrow \mathit{holds}(f, t_2)) \\
[A_3] & \mathbf{C}(\forall t_1, f, t_2 \,.\, \mathit{clipped}(t_1, f, t_2) \Leftrightarrow \\
 & \quad [\exists e, t \,.\, \mathit{happens}(e, t) \wedge t_1 < t < t_2 \wedge \mathit{terminates}(e, f, t)])
\end{array}
$$

suggesting that people have a (possibly innate) understanding of basic causality principles, and are indeed aware that everybody has such an understanding. In addition to $[A_1]$—$[A_3]$, we postulate a few more axioms pertaining to what people know or believe about causality. First, agents know the events that they intentionally bring about themselves—that is part of what "action" means. In fact, this is common knowledge. The following axiom expresses this:

$$
\begin{array}{ll}
[A_4] & \mathbf{C}(\forall a, d, t \,.\, \mathit{happens}(\mathit{action}(a, d), t) \Rightarrow \\
 & \qquad \mathbf{K}(a, \mathit{happens}(\mathit{action}(a, d), t)))
\end{array}
$$

The next axiom states that it is common knowledge that if an agent a believes that a certain fluent f holds at t and he does not believe that f has been clipped between t and t', then he will also believe that f holds at t':

$$
\begin{array}{ll}
[A_5] & \mathbf{C}(\forall a, f, t, t' \,.\, \mathbf{B}(a, \mathit{holds}(f, t)) \wedge \mathbf{B}(a, t < t') \wedge \\
 & \qquad \neg \mathbf{B}(a, \mathit{clipped}(t, f, t')) \Rightarrow \mathbf{B}(a, \mathit{holds}(f, t')))
\end{array}
$$

The final axiom states that if a believes that b believes that f holds at t_1 and a believes that nothing has happened between t_1 and t_2 to change b's mind, then a will believe that b will not think that f has been clipped between t_1 and t_2:

2 A calculus for representing and reasoning about mental states

The syntactic and semantic problems that arise when one tries to use classical logic to represent and reason about intentional notions are well-known. Syntactically, modelling belief or knowledge relationally is problematic because one believes or knows arbitrarily complex propositions, whereas the arguments of relation symbols are terms built from constants, variables, and function symbols. (The objects of belief could be encoded by strings, but such representations are too low-level for engineering purposes.) Semantically, the main issue is the referential opacity (or intensionality) that must be exhibited by any operators for belief, desire, knowledge, etc. In intensional contexts one cannot freely substitute one coreferential term for another. Broadly speaking, there are two ways of addressing these issues. One is to use a modal logic, with built-in syntactic operators for intentional notions. The other is to stick with classical logic but distinguish between an object-language and a meta-language, representing intentional discourse at the object level. Each approach has its own advantages and drawbacks. Sticking with classical logic has the important advantage of efficiency, in that automated deduction systems for classical logic, such as resolution provers—which have made impressive strides over the last decade—can be used for reasoning. One disadvantage of this approach is that when the object language is first-order (includes quantification), then notions such as substitutions and alphabetic equivalence must be explicitly encoded. Depending on the facilities provided by the meta-language, this does not need to be overly onerous, but it does require extra effort.

The modal-logic approach has the advantage of solving the syntactic and semantics problems directly, without the need to distinguish an object-language and a meta-language. That is the approach we have taken in this work. The main drawback of this approach is the difficulty of automating reasoning, since standard theorem-proving techniques from classical logic cannot be directly employed. We have tried to overcome this limitation here by exploring the automation potential of methods, or derived inference rules (called *tactics* in the terminology of HOL [7]). Another drawback is the issue of semantics. The standard semantics of modal logics are given in terms of Kripke structures involving possible worlds. Such semantics are very elegant and well-understood mathematically. They are also quite intuitive for logics dealing with necessity or time. However, they are remarkably unintuitive for doxastic and epistemic logics. Not only do they fail to shed any light on the nature of belief or knowledge, but they also have a number of widely known counter-intuitive consequences that are unacceptable for resource-bounded agents, such as logical omniscience (deductive closure of knowledge, knowledge of all tautologies, etc.) and the fixed-point characterization of common knowledge. These issues are significant for us, given that we are interested in telling a plausible story for how actual agents in the real world can succeed on false-belief tasks. There have been numerous attempts to rectify these issues [8, 4, 9, 10], but each has faced serious problems of its own, and outside of Kripke structures there is no widely accepted standard at present.

Accordingly, we have not provided a possible-world semantics for our system. Note that an additional potential complication here is that the semantics of the event calculus are given in terms of circumscription, a second-order logic schema, and it is not obvious how to accommodate that feature in the setting of possible worlds. Due to these issues, and due to space restrictions, our presentation here is entirely proof-theoretic. The meanings of the various syntactic constructs—such as the knowledge operator—can be viewed as determined by their inferential *roles*, as specified by the various inference rules. (This can itself be regarded as a form of semantics; it is called "conceptual-role semantics" or "functional semantics" in the philosophy of mind; "natural semantics" in computer science; and "procedural semantics" in cognitive science.)

The following is the formal specification of our system, describing the various sorts of our universe (S), the signatures of certain built-in function symbols (f), and the abstract syntax of terms (t) and propositions (P). The symbol $\sqsubseteq$ denotes subsorting:

$$
\begin{array}{rcl}
S & ::= & \texttt{Object} \mid \texttt{Agent} \mid \texttt{ActionType} \mid \texttt{Action} \sqsubseteq \texttt{Event} \\
 & \mid & \texttt{Moment} \mid \texttt{Boolean} \mid \texttt{Fluent} \\
f & ::= & \begin{array}{l} \mathit{action} : \texttt{Agent} \times \texttt{ActionType} \rightarrow \texttt{Action} \\ \mathit{initially} : \texttt{Fluent} \rightarrow \texttt{Boolean} \\ \mathit{holds} : \texttt{Fluent} \times \texttt{Moment} \rightarrow \texttt{Boolean} \\ \mathit{happens} : \texttt{Event} \times \texttt{Moment} \rightarrow \texttt{Boolean} \\ \mathit{clipped} : \texttt{Moment} \times \texttt{Fluent} \times \texttt{Moment} \rightarrow \texttt{Boolean} \\ \mathit{initiates} : \texttt{Event} \times \texttt{Fluent} \times \texttt{Moment} \rightarrow \texttt{Boolean} \\ \mathit{terminates} : \texttt{Event} \times \texttt{Fluent} \times \texttt{Moment} \rightarrow \texttt{Boolean} \\ \mathit{prior} : \texttt{Moment} \times \texttt{Moment} \rightarrow \texttt{Boolean} \end{array} \\
t & ::= & x : S \mid c : S \mid f(t_1, \ldots, t_n) \\
P & ::= & t : \texttt{Boolean} \mid \neg P \mid P \wedge Q \mid P \vee Q \mid P \Rightarrow Q \mid P \Leftrightarrow Q \mid \\
 & & \forall x : S \,.\, P \mid \exists x : S \,.\, P \mid \mathbf{S}(a, P) \mid \mathbf{K}(a, P) \mid \mathbf{B}(a, P) \mid \mathbf{C}(P)
\end{array}
$$

Propositions of the form $\mathbf{S}(a, P)$, $\mathbf{B}(a, P)$, and $\mathbf{K}(a, P)$ should be understood as saying that agent a sees that P is the case, believes that P, and knows that P, respectively. Propositions of the form $\mathbf{C}(P)$ assert that P is commonly known. Sort annotations will generally be omitted, as they are easily deducible from the context. We write $P[x \mapsto t]$ for the proposition obtained from P by replacing every free occurrence of x by t, assuming that t is of a sort compatible with the sort of the free occurrences in question, and taking care to rename P as necessary to avoid variable capture. We use the infix notation $t_1 < t_2$ instead of $\mathit{prior}(t_1, t_2)$.

We express the following standard axioms of the event calculus as common knowledge:

$$
\begin{array}{ll}
[A_1] & \mathbf{C}(\forall f, t \,.\, \mathit{initially}(f) \wedge \neg \mathit{clipped}(0, f, t) \Rightarrow \mathit{holds}(f, t)) \\
[A_2] & \mathbf{C}(\forall e, f, t_1, t_2 \,.\, \mathit{happens}(e, t_1) \wedge \mathit{initiates}(e, f, t_1) \wedge \\
 & \quad t_1 < t_2 \wedge \neg \mathit{clipped}(t_1, f, t_2) \Rightarrow \mathit{holds}(f, t_2)) \\
[A_3] & \mathbf{C}(\forall t_1, f, t_2 \,.\, \mathit{clipped}(t_1, f, t_2) \Leftrightarrow \\
 & \quad [\exists e, t \,.\, \mathit{happens}(e, t) \wedge t_1 < t < t_2 \wedge \mathit{terminates}(e, f, t)])
\end{array}
$$

suggesting that people have a (possibly innate) understanding of basic causality principles, and are indeed aware that everybody has such an understanding. In addition to $[A_1]$—$[A_3]$, we postulate a few more axioms pertaining to what people know or believe about causality. First, agents know the events that they intentionally bring about themselves—that is part of what "action" means. In fact, this is common knowledge. The following axiom expresses this:

$$
\begin{array}{ll}
[A_4] & \mathbf{C}(\forall a, d, t \,.\, \mathit{happens}(\mathit{action}(a, d), t) \Rightarrow \\
 & \quad \mathbf{K}(a, \mathit{happens}(\mathit{action}(a, d), t)))
\end{array}
$$

The next axiom states that it is common knowledge that if an agent a believes that a certain fluent f holds at t and he does not believe that f has been clipped between t and t', then he will also believe that f holds at t':

$$
\begin{array}{ll}
[A_5] & \mathbf{C}(\forall a, f, t, t' \,.\, \mathbf{B}(a, \mathit{holds}(f, t)) \wedge \mathbf{B}(a, t < t') \wedge \\
 & \quad \neg \mathbf{B}(a, \mathit{clipped}(t, f, t')) \Rightarrow \mathbf{B}(a, \mathit{holds}(f, t')))
\end{array}
$$

The final axiom states that if a believes that b believes that f holds at t_1 and a believes that nothing has happened between t_1 and t_2 to change b's mind, then a will believe that b will not think that f has been clipped between t_1 and t_2:

Full generality wrt time and change: includes event calculus — yet fast.

formalize this scenario in our calculus. In the next section we will present a formal explanation as to how Alice can come to acquire the correct belief about Bob's false belief.

We introduce the sort `Location` and the following function symbols specifically for reasoning about the false-belief task:

$$\begin{gathered} \mathit{places} : \texttt{Object} \times \texttt{Location} \rightarrow \texttt{ActionType} \\ \mathit{moves} : \texttt{Object} \times \texttt{Location} \times \texttt{Location} \rightarrow \texttt{ActionType} \\ \mathit{located} : \texttt{Object} \times \texttt{Location} \rightarrow \texttt{Fluent} \end{gathered}$$

Intuitively, $\mathit{action}(a, \mathit{places}(o, l))$ signifies a's action of placing object o in location l, while

$$\mathit{action}(a, \mathit{moves}(o, l_1, l_2))$$

is a's action of moving object o from location l_1 to location l_2. It is common knowledge that placing o in l initiates the fluent $\mathit{located}(o, l)$:

$$[D_1] \quad \mathbf{C}(\forall\, a, t, o, l\ .\ \mathit{initiates}(\mathit{action}(a, \mathit{places}(o, l)), \mathit{located}(o, l), t))$$

It is likewise known that if an object o is located at l_1 at a time t, then the act of moving o from l_1 to l_2 results in o being located at l_2:

$$\begin{aligned} [D_2] \qquad & \mathbf{C}(\forall\, a, t, o, l_1, l_2\ .\ \mathit{holds}(\mathit{located}(o, l_1), t) \Rightarrow \\ & \mathit{initiates}(\mathit{action}(a, \mathit{moves}(o, l_1, l_2)), \mathit{located}(o, l_2), t)) \end{aligned}$$

If, in addition, the new location is different from the old one, the move terminates the fluent $\mathit{located}(o, l_1)$:

$$\begin{aligned} [D_3] \qquad & \mathbf{C}(\forall\, a, t, o, l_1, l_2\ .\ \mathit{holds}(\mathit{located}(o, l_1), t) \wedge l_1 \neq l_2 \Rightarrow \\ & \mathit{terminates}(\mathit{action}(a, \mathit{moves}(o, l_1, l_2)), \mathit{located}(o, l_1), t)) \end{aligned}$$

The following axiom captures the constraint that an object cannot be in more than one place at one time; this is also common knowledge:

$$\begin{aligned} [D_4] \qquad & \mathbf{C}(\forall\, o, t, l_1, l_2\ .\ \mathit{holds}(\mathit{located}(o, l_1), t) \wedge \\ & \mathit{holds}(\mathit{located}(o, l_2), t) \Rightarrow l_1 = l_2) \end{aligned}$$

We introduce three time moments that are central to the narrative of the false-belief task: *beginning*, *departure*, and *return*. The first signifies the time point when Bob places the cookie in the cabinet, while *departure* and *return* mark the points when he leaves and comes back, respectively. We assume that it's common knowledge that these three time points are linearly ordered in the obvious manner:

$$[D_5] \quad \mathbf{C}(\mathit{beginning} < \mathit{departure} < \mathit{return}).$$

We also introduce two distinct locations, *cabinet* and *drawer*:

$$[D_6] \quad \mathbf{C}(\mathit{cabinet} \neq \mathit{drawer}).$$

Finally, we introduce a domain `Cookie` as a subsort of `Object`, and declare a single element of it, *cookie*. It is a given premise that, in the beginning, Alice sees Bob place the cookie in the cabinet:

$$[D_7] \quad \mathbf{S}(\mathit{Alice}, \mathit{happens}(\mathit{action}(\mathit{Bob}, \mathit{places}(\mathit{cookie}, \mathit{cabinet})), \mathit{beginning})).$$

4 Modeling the reasoning underlying false-belief tasks, and automating it via abstraction

At this point we have enough representational and reasoning machinery in place to infer the correct conclusion from a couple of obvious premises. However, a monolithic derivation of the conclusion from the premises would be unsatisfactory, as it would not give us a story about how such reasoning can be dynamically put together. Agents must be able to reason about the behavior of other agents efficiently. It is not at all obvious how efficiency can be achieved in the absence of mechanisms for abstraction, modularity, and reusability.

We can begin to address both issues by pursuing further the idea of derived inference rules, and by borrowing a page from classic work in cognitive science and production systems. Suppose that we had a mechanism which enabled the derivation of not only *schematic* inference rules, such as the ones that we presented in section 2, but derived inference rules allowing for arbitrary computation and search. We could then formulate *generic* inference rules, capable of being applied to an unbounded (potentially infinite) number of arbitrarily complex concrete situations.

Our system has a notion of *method* that allows for that type of abstraction and encapsulation. Methods are derived inference rules, not just of the schematic kind, but incorporating arbitrary computation and search. They are thus more general than the simple if-then rules of production systems, and more akin to the knowledge sources (or "demons") of blackboard systems [5]. They can be viewed as encapsulating specialized expertise in deriving certain types of conclusions from certain given information. They can be parameterized over any variables, e.g., arbitrary agents or time points. In our system, the role of the blackboard is played by an associative data structure (shared by all methods) known as the *assumption base*, which is an efficiently indexed collection of propositions that represent the collective knowledge state at any given moment, including perceptual knowledge. The assumption base is capable of serving as a communication buffer for the various methods. Finally, the control executive is itself a method, which directs the reasoning process incrementally by invoking various methods triggered by the contents of the assumption base.

We describe below three general-purpose methods for reasoning in the calculus we have presented. With these methods, the reasoning for the false-belief task can be performed in a handful of lines—essentially with one invocation of each of these methods. We stress that these methods are not ad hoc or hardwired to false-belief tasks. They are generic, and can be reused in any context that requires reasoning about other minds and satisfies the relevant preconditions. In particular, the methods do not contain or require any information specific to false-belief tasks.

- *Method 1*: This method, which we call M_1, shows that when an agent a_1 sees an agent a_2 perform some action-type α at some time point t, a_1 knows that a_2 knows that a_2 has carried out α at t. M_1 is parameterized over a_1, a_2, α, and t:

  1. The starting premise is that a_1 sees a_2 perform α at t:
  $$\mathbf{S}(a_1, \mathit{happens}(\mathit{action}(a_2, \alpha), t)) \qquad (1)$$
  2. Therefore, a_1 knows that the corresponding event has occurred at t:
  $$\mathbf{K}(a_1, \mathit{happens}(\mathit{action}(a_2, \alpha), t)) \qquad (2)$$
  This follows from the preceding premise and $[DR_4]$.
  3. From $[A_4]$ and $[DR_2]$ we obtain:
  $$\begin{aligned} & \mathbf{K}(a_1, \forall\, a, \alpha, t\ .\ \mathit{happens}(\mathit{action}(a, \alpha), t) \Rightarrow \\ & \mathbf{K}(a, \mathit{happens}(\mathit{action}(a, \alpha), t))) \end{aligned} \qquad (3)$$
  4. From (3) and $[DR_9]$ we get:
  $$\begin{aligned} & \mathbf{K}(a_1, \mathit{happens}(\mathit{action}(a_2, \alpha), t) \Rightarrow \\ & \mathbf{K}(a_2, \mathit{happens}(\mathit{action}(a_2, \alpha), t))) \end{aligned} \qquad (4)$$
  5. From (4), (2), and $[DR_6]$ we get:
  $$\mathbf{K}(a_1, \mathbf{K}(a_2, \mathit{happens}(\mathit{action}(a_2, \alpha), t))) \qquad (5)$$

- *Method 2*: The second method, M_2, shows that when (1) it is common knowledge that a certain event e initiates a fluent f; (2) an agent a_1 knows that an agent a_2 knows that e has happened at a

Proof methods for efficiency.

formalize this scenario in our calculus. In the next section we will present a formal explanation as to how Alice can come to acquire the correct belief about Bob's false belief.

We introduce the sort `Location` and the following function symbols specifically for reasoning about the false-belief task:

$$\begin{gathered} \mathit{places} : \texttt{Object} \times \texttt{Location} \rightarrow \texttt{ActionType} \\ \mathit{moves} : \texttt{Object} \times \texttt{Location} \times \texttt{Location} \rightarrow \texttt{ActionType} \\ \mathit{located} : \texttt{Object} \times \texttt{Location} \rightarrow \texttt{Fluent} \end{gathered}$$

Intuitively, $\mathit{action}(a, \mathit{places}(o, l))$ signifies a's action of placing object o in location l, while

$$\mathit{action}(a, \mathit{moves}(o, l_1, l_2))$$

is a's action of moving object o from location l_1 to location l_2. It is common knowledge that placing o in l initiates the fluent $\mathit{located}(o, l)$:

$$[D_1] \quad \mathbf{C}(\forall\, a, t, o, l\ .\ \mathit{initiates}(\mathit{action}(a, \mathit{places}(o, l)), \mathit{located}(o, l), t))$$

It is likewise known that if an object o is located at l_1 at a time t, then the act of moving o from l_1 to l_2 results in o being located at l_2:

$$\begin{aligned}[D_2] \qquad & \mathbf{C}(\forall\, a, t, o, l_1, l_2\ .\ \mathit{holds}(\mathit{located}(o, l_1), t) \Rightarrow \\ & \mathit{initiates}(\mathit{action}(a, \mathit{moves}(o, l_1, l_2)), \mathit{located}(o, l_2), t))\end{aligned}$$

If, in addition, the new location is different from the old one, the move terminates the fluent $\mathit{located}(o, l_1)$:

$$\begin{aligned}[D_3] \qquad & \mathbf{C}(\forall\, a, t, o, l_1, l_2\ .\ \mathit{holds}(\mathit{located}(o, l_1), t) \wedge l_1 \neq l_2 \Rightarrow \\ & \mathit{terminates}(\mathit{action}(a, \mathit{moves}(o, l_1, l_2)), \mathit{located}(o, l_1), t))\end{aligned}$$

The following axiom captures the constraint that an object cannot be in more than one place at one time; this is also common knowledge:

$$\begin{aligned}[D_4] \qquad & \mathbf{C}(\forall\, o, t, l_1, l_2\ .\ \mathit{holds}(\mathit{located}(o, l_1), t) \wedge \\ & \mathit{holds}(\mathit{located}(o, l_2), t) \Rightarrow l_1 = l_2)\end{aligned}$$

We introduce three time moments that are central to the narrative of the false-belief task: *beginning*, *departure*, and *return*. The first signifies the time point when Bob places the cookie in the cabinet, while *departure* and *return* mark the points when he leaves and comes back, respectively. We assume that it's common knowledge that these three time points are linearly ordered in the obvious manner:

$$[D_5] \quad \mathbf{C}(\mathit{beginning} < \mathit{departure} < \mathit{return}).$$

We also introduce two distinct locations, *cabinet* and *drawer*:

$$[D_6] \quad \mathbf{C}(\mathit{cabinet} \neq \mathit{drawer}).$$

Finally, we introduce a domain `Cookie` as a subsort of `Object`, and declare a single element of it, *cookie*. It is a given premise that, in the beginning, Alice sees Bob place the cookie in the cabinet:

$$[D_7] \quad \mathbf{S}(\mathit{Alice}, \mathit{happens}(\mathit{action}(\mathit{Bob}, \mathit{places}(\mathit{cookie}, \mathit{cabinet})), \mathit{beginning})).$$

4 Modeling the reasoning underlying false-belief tasks, and automating it via abstraction

At this point we have enough representational and reasoning machinery in place to infer the correct conclusion from a couple of obvious premises. However, a monolithic derivation of the conclusion from the premises would be unsatisfactory, as it would not give us a story about how such reasoning can be dynamically put together. Agents must be able to reason about the behavior of other agents efficiently. It is not at all obvious how efficiency can be achieved in the absence of mechanisms for abstraction, modularity, and reusability.

We can begin to address both issues by pursuing further the idea of derived inference rules, and by borrowing a page from classic work in cognitive science and production systems. Suppose that we had a mechanism which enabled the derivation of not only *schematic* inference rules, such as the ones that we presented in section 2, but derived inference rules allowing for arbitrary computation and search. We could then formulate *generic* inference rules, capable of being applied to an unbounded (potentially infinite) number of arbitrarily complex concrete situations.

Our system has a notion of *method* that allows for that type of abstraction and encapsulation. Methods are derived inference rules, not just of the schematic kind, but incorporating arbitrary computation and search. They are thus more general than the simple if-then rules of production systems, and more akin to the knowledge sources (or "demons") of blackboard systems [5]. They can be viewed as encapsulating specialized expertise in deriving certain types of conclusions from certain given information. They can be parameterized over any variables, e.g., arbitrary agents or time points. In our system, the role of the blackboard is played by an associative data structure (shared by all methods) known as the *assumption base*, which is an efficiently indexed collection of propositions that represent the collective knowledge state at any given moment, including perceptual knowledge. The assumption base is capable of serving as a communication buffer for the various methods. Finally, the control executive is itself a method, which directs the reasoning process incrementally by invoking various methods triggered by the contents of the assumption base.

We describe below three general-purpose methods for reasoning in the calculus we have presented. With these methods, the reasoning for the false-belief task can be performed in a handful of lines—essentially with one invocation of each of these methods. We stress that these methods are not ad hoc or hardwired to false-belief tasks. They are generic, and can be reused in any context that requires reasoning about other minds and satisfies the relevant preconditions. In particular, the methods do not contain or require any information specific to false-belief tasks.

- *Method 1*: This method, which we call M_1, shows that when an agent a_1 sees an agent a_2 perform some action-type α at some time point t, a_1 knows that a_2 knows that a_2 has carried out α at t. M_1 is parameterized over a_1, a_2, α, and t:

  1. The starting premise is that a_1 sees a_2 perform α at t:
  $$\mathbf{S}(a_1, \mathit{happens}(\mathit{action}(a_2, \alpha), t)) \tag{1}$$
  2. Therefore, a_1 knows that the corresponding event has occurred at t:
  $$\mathbf{K}(a_1, \mathit{happens}(\mathit{action}(a_2, \alpha), t)) \tag{2}$$
  This follows from the preceding premise and $[DR_4]$.
  3. From $[A_4]$ and $[DR_2]$ we obtain:
  $$\begin{aligned}&\mathbf{K}(a_1, \forall\, a, \alpha, t\ .\ \mathit{happens}(\mathit{action}(a, \alpha), t) \Rightarrow \\ &\quad \mathbf{K}(a, \mathit{happens}(\mathit{action}(a, \alpha), t)))\end{aligned} \tag{3}$$
  4. From (3) and $[DR_9]$ we get:
  $$\begin{aligned}&\mathbf{K}(a_1, \mathit{happens}(\mathit{action}(a_2, \alpha), t) \Rightarrow \\ &\quad \mathbf{K}(a_2, \mathit{happens}(\mathit{action}(a_2, \alpha), t)))\end{aligned} \tag{4}$$
  5. From (4), (2), and $[DR_6]$ we get:
  $$\mathbf{K}(a_1, \mathbf{K}(a_2, \mathit{happens}(\mathit{action}(a_2, \alpha), t))) \tag{5}$$

- *Method 2*: The second method, M_2, shows that when (1) it is common knowledge that a certain event e initiates a fluent f; (2) an agent a_1 knows that an agent a_2 knows that e has happened at a

Proof methods for efficiency.

Arkoudas, K. & Bringsjord, S. (2008) "Toward Formalizing Common-Sense Psychology: An Analysis of the False-Belief Task"' *Proceedings of the Tenth Pacific Rim International Conference on Artificial Intelligence* (PRICAI 2008). Lecture Notes in Artificial Intelligence (LNAI), No. 5351, T.-B. Ho and Z.-H. Zhou, eds. (New York, NY: Springer-Verlag), pp. 17–29. Offprint available at: http://kryten.mm.rpi.edu/KA_SB_PRICAI08_AI_off.pdf

Solving Problem 2: Program Verification ...

Minds & Machines (2007) 17:185–202
DOI 10.1007/s11023-007-9063-5

Computers, Justification, and Mathematical Knowledge

Konstantine Arkoudas · Selmer Bringsjord

http://kryten.mm.rpi.edu/ka_sb_proofs_offprint.pdf

Published online: 23 June 2007

Abstract The original proof of the four-color theorem by Appel and Haken sparked a controversy when Tymoczko used it to argue that the justification provided by unsurveyable proofs carried out by computers cannot be a priori. It also created a lingering impression to the effect that such proofs depend heavily for their soundness on large amounts of computation-intensive custom-built software. Contra Tymoczko, we argue that the justification provided by certain computerized mathematical proofs is not fundamentally different from that provided by surveyable proofs, and can be sensibly regarded as a priori. We also show that the aforementioned impression is mistaken because it fails to distinguish between proof search (the context of discovery) and proof checking (the context of justification). By using mechanized proof assistants capable of producing certificates that can be independently checked, it is possible to carry out complex proofs without the need to trust arbitrary custom-written code. We only need to trust one fixed, small, and simple piece of software: the proof checker. This is not only possible in principle, but is in fact becoming a viable methodology for performing complicated mathematical reasoning. This is evinced by a new proof of the four-color theorem that appeared in 2005, and which was developed and checked in its entirety by a mechanical proof system.

Keywords A priori · Justification · Proofs · Certificates · Four-color theorem · Mathematical knowledge

K. Arkoudas (✉) · S. Bringsjord
Cognitive Science Department, Computer Science Department, RPI, Troy, NY, USA 12180
e-mail: arkouk@rpi.edu

S. Bringsjord
e-mail: brings@rpi.edu

Minds & Machines (2007) 17:185–202
DOI 10.1007/s11023-007-9063-5

Computers, Justification, and Mathematical Knowledge

Konstantine Arkoudas · Selmer Bringsjord

http://kryten.mm.rpi.edu/ka_sb_proofs_offprint.pdf

Published online: 23 June 2007

Abstract The original proof of the four-color theorem by Appel and Haken sparked a controversy when Tymoczko used it to argue that the justification provided by unsurveyable proofs carried out by computers cannot be a priori. It also created a lingering impression to the effect that such proofs depend heavily for their soundness on large amounts of computation-intensive custom-built software. Contra Tymoczko, we argue that the justification provided by certain computerized mathematical proofs is not fundamentally different from that provided by surveyable proofs, and can be sensibly regarded as a priori. We also show that the aforementioned impression is mistaken because it fails to distinguish between proof search (the context of discovery) and proof checking (the context of justification). By using mechanized proof assistants capable of producing certificates that can be independently checked, it is possible to carry out complex proofs without the need to trust arbitrary custom-written code. We only need to trust one fixed, small, and simple piece of software: the proof checker. This is not only possible in principle, but is in fact becoming a viable methodology for performing complicated mathematical reasoning. This is evinced by a new proof of the four-color theorem that appeared in 2005, and which was developed and checked in its entirety by a mechanical proof system.

Keywords A priori · Justification · Proofs · Certificates · Four-color theorem · Mathematical knowledge

K. Arkoudas (✉) · S. Bringsjord
Cognitive Science Department, Computer Science Department, RPI, Troy, NY, USA 12180
e-mail: arkouk@rpi.edu

S. Bringsjord
e-mail: brings@rpi.edu

Minds & Machines (2007) 17:185–202
DOI 10.1007/s11023-007-9063-5

Computers, Justification, and Mathematical Knowledge

Konstantine Arkoudas · Selmer Bringsjord

http://kryten.mm.rpi.edu/ka_sb_proofs_offprint.pdf

Published online: 23 June 2007

Abstract The original proof of the four-color theorem by Appel and Haken sparked a controversy when Tymoczko used it to argue that the justification provided by unsurveyable proofs carried out by computers cannot be a priori. It also created a lingering impression to the effect that such proofs depend heavily for their soundness on large amounts of computation-intensive custom-built software. Contra Tymoczko, we argue that the justification provided by certain computerized mathematical proofs is not fundamentally different from that provided by surveyable proofs, and can be sensibly regarded as a priori. We also show that the aforementioned impression is mistaken because it fails to distinguish between proof search (the context of discovery) and proof checking (the context of justification). By using mechanized proof assistants capable of producing certificates that can be independently checked, it is possible to carry out complex proofs without the need to trust arbitrary custom-written code. We only need to trust one fixed, small, and simple piece of software: the proof checker. This is not only possible in principle, but is in fact becoming a viable methodology for performing complicated mathematical reasoning. This is evinced by a new proof of the four-color theorem that appeared in 2005, and which was developed and checked in its entirety by a mechanical proof system.

Keywords A priori · Justification · Proofs · Certificates · Four-color theorem · Mathematical knowledge

K. Arkoudas (✉) · S. Bringsjord
Cognitive Science Department, Computer Science Department, RPI, Troy, NY, USA 12180
e-mail: arkouk@rpi.edu

S. Bringsjord
e-mail: brings@rpi.edu

Minds & Machines (2007) 17:185–202
DOI 10.1007/s11023-007-9063-5

Computers, Justification, and Mathematical Knowledge

Konstantine Arkoudas · Selmer Bringsjord

http://kryten.mm.rpi.edu/ka_sb_proofs_offprint.pdf

Published online: 23 June 2007

Abstract The original proof of the four-color theorem by Appel and Haken sparked a controversy when Tymoczko used it to argue that the justification provided by unsurveyable proofs carried out by computers cannot be a priori. It also created a lingering impression to the effect that such proofs depend heavily for their soundness on large amounts of computation-intensive custom-built software. Contra Tymoczko, we argue that the justification provided by certain computerized mathematical proofs is only fundamentally different from that provided by ordinary proofs [illegible] aforementioned impression is mistaken because it fails to distinguish between proof search, the [illegible] discovery, and proof checking, the [illegible] of justification. [illegible] independently checked, it is possible to carry out complex proofs without the need to trust arbitrary custom-written code. We only need to trust one fixed, small, and simple piece of software: the proof checker. This is not only possible in principle, but is in fact becoming a viable methodology for performing complicated mathematical reasoning. This is evinced by a new proof of the four-color theorem that appeared in 2005, and which was developed and checked in its entirety by a mechanical proof system.

Keywords A priori · Justification · Proofs · Certificates · Four-color theorem · Mathematical knowledge

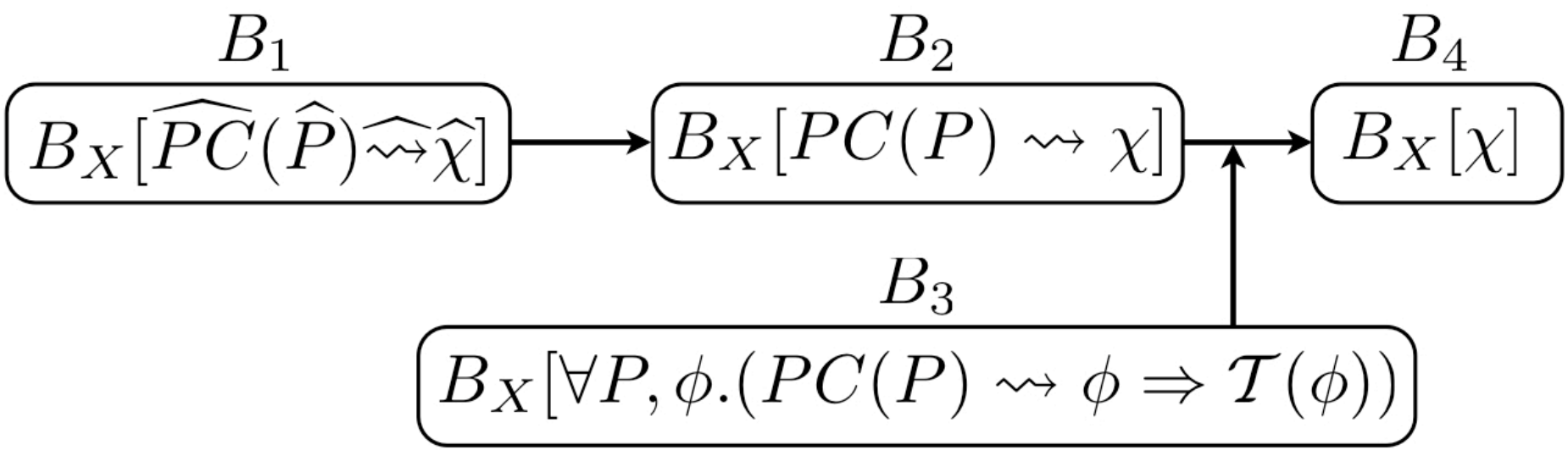
K. Arkoudas (✉) · S. Bringsjord
Cognitive Science Department, Computer Science Department, RPI, Troy, NY, USA 12180
e-mail: arkouk@rpi.edu

S. Bringsjord
e-mail: brings@rpi.edu

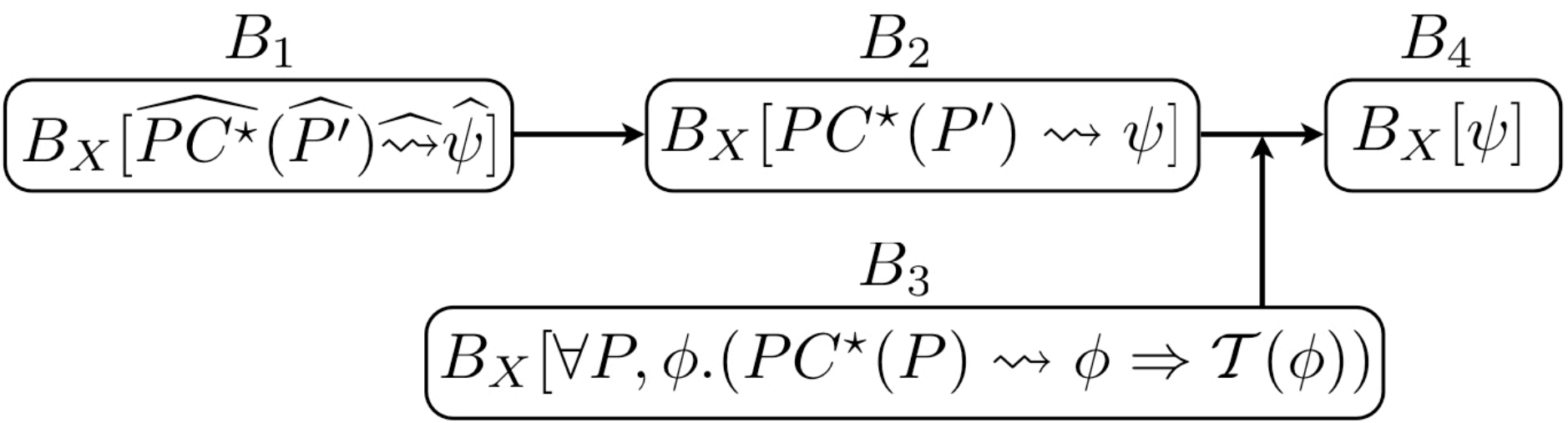
and ...

Bringsjord, S. (forthcoming) "Rigorous Attacks on Program Verification are Self-Refuting." Available by direct email.

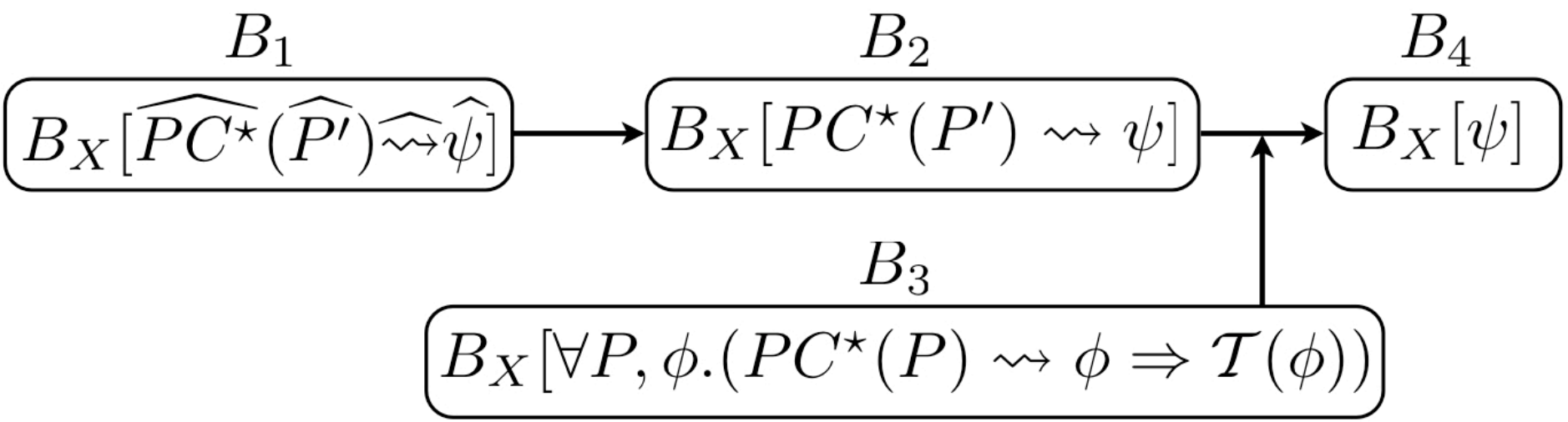
Believing the Completeness of FOL



Program Verification Solution



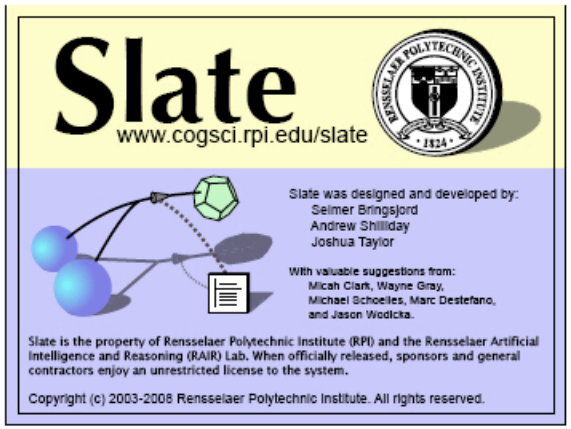
Program Verification Solution



Program verification is solved b/c there is only one short program in silicon to be conventionally hardware-verified, and all other software is proof-theoretical.

Slate: An Argument-Centered Intelligent Assistant to Human Reasoners

Selmer Bringsjord and Joshua Taylor and Andrew Shilliday
and Micah Clark and Konstantine Arkoudas[1]



Abstract. We describe Slate, a logic-based, robust interactive reasoning system that allows human "pilots" to harness an ensemble of intelligent agents in order to construct, test, and express various sorts of natural argumentation. Slate empowers students and professionals in the business of producing argumentation, e.g., mathematicians, logicians, intelligence analysts, designers and producers of standardized reasoning tests. We demonstrate Slate in several examples, describe some distinctive features of the system (e.g., reading and generating natural language, immunizing human reasoners from "logical illusions"), present Slate's theoretical underpinnings, and note upcoming refinements.

1 INTRODUCTION

Slate is a robust interactive reasoning system. It allows the human "pilot" to harness an ensemble of intelligent agents in order to construct, test, and express natural argumentation of various sorts. Slate is designed to empower students and professionals in the business of producing argumentation, e.g., mathematicians, logicians, intelligence analysts, designers and producers of standardized reasoning tests, and so on. While other ways of pursuing AI may well be preferable in certain contexts, faced with the challenge of having to engineer a system like Slate, a logic-based approach [9, 10, 18, 31, 13] seemed to us ideal, and perhaps the power of Slate even at this point (version 3) confirms the efficacy of this approach. In addition, there is of course a longstanding symbiosis between argumentation and logic revealed in contemporary essays on argumentation [48]. In this paper, we summarize Slate through several examples, describe some distinctive features of the system (e.g., its capacity to read and generate natural language, and to provide human reasoners with apparent immunity from so-called "logical illusions"), say a bit about Slate's theoretical underpinnings, and note upcoming refinements.

[1] Rensselaer Polytechnic Institute (RPI), USA, email: {selmer, tayloj, shilla, clarkm5, arkouk}@rpi.edu

2 A SIMPLE EXAMPLE

We begin by following a fictitious user, Ulric, as he uses Slate to solve a short logic puzzle, the *Dreadsbury Mansion Mystery* [34]:[2]

> Someone who lives in Dreadsbury Mansion killed Aunt Agatha. Agatha, the butler, and Charles live in Dreadsbury Mansion, and are the only people who live therein. A killer always hates his victim, and is never richer than his victim. Charles hates no one that Aunt Agatha hates. Agatha hates everyone except the butler. The butler hates everyone not richer than Aunt Agatha. The butler hates everyone Agatha hates. No one hates everyone. Agatha is not the butler. *Who killed Agatha?*

Information can enter Slate in a number of formats, e.g., as formulae in many-sorted logic (MSL), or as sentences in a logically-controlled English (§4.2). Information can also be imported from external repositories such as databases or the Semantic Web (§4.5). Ulric examines the Dreadsbury Mansion Mystery facts displayed in Slate's workspace (Figure 1).

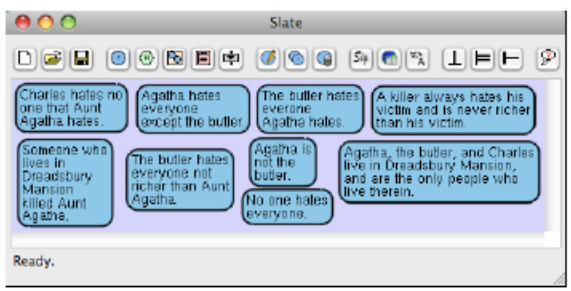


Figure 1. The Dreadsbury Mansion Mystery facts represented in Slate.

A fan of murder mysteries, he considers whether conventional wisdom might hold true, i.e., that the butler did it. Ulric adds the hypothetical to Slate's workspace and asks Slate to check whether the hypothesis is consistent with the other propositions. Slate quickly reports an inconsistency (Figure 2).

[2] This puzzle is of a type typically used to challenge humans (e.g., students in introductory logic courses) and machines (e.g., automated theorem provers).

a Gödelian logic puzzle that approximates GI and demonstrates the power of Slate within demanding logico-mathematical domains like those in which Gödel worked.

A Precursor Gödelian Puzzle. Suppose a machine $\mathcal{M}$ operates on expressions: finite, non-empty sequences of the four glyphs $\sim$, $\star$, P, and M. These four glyphs have intuitive meanings: $\sim$ stands for 'not,' $\star$ for 'to be' or 'is,' P for 'provable,' and M for 'mirror of,' where the mirror of an expression ϕ is the expression $\phi\star\phi$. A sentence is an expression of a particular form, also with an intuitive meaning, specifically,

$P\star\phi$ means that ϕ is provable and is true if and only if ϕ is provable by $\mathcal{M}$.

$PM\star\phi$ means that the mirror of ϕ is provable, and is true if and only if the mirror of ϕ is provable by $\mathcal{M}$.

$\sim P\star\phi$ means that ϕ is not provable, and is true if and only if ϕ is not provable by $\mathcal{M}$.

$\sim PM\star\phi$ means that the mirror of ϕ is not provable, and is true if and only if the mirror of ϕ is not provable by $\mathcal{M}$.

$\mathcal{M}$ is such that it only proves true sentences and never false sentences (i.e., the machine is *sound*). Prove that $\mathcal{M}$ cannot prove all true sentences—there is a true sentence which cannot be proved by $\mathcal{M}$ (i.e., the machine is *incomplete*).

Formalization of the Gödelian Puzzle. We formalize the above puzzle as a logical language consisting of the constants: $\sim$, $\star$, P, M; the (unary) predicates: glyph, expression, sentence, provable, and true; and the functions: cat (concatenation), and mirror. For convenience, we describe as glyphs, expressions, sentences, provable, and true any terms on which glyph, expression, sentence, provable, and true holds, respectively, and denote the application of cat to two terms ϕ and ψ as the concatenation of ϕ and ψ, or by $\phi\psi$, and the application of mirror to a term ϕ as the mirror of ϕ. The interpretation of this vocabulary is subject to the following twelve axioms:

1. *The constants $\sim$, $\star$, P, and M are each distinct.*
2. *The constants $\sim$, $\star$, P, and M are the only glyphs.*
3. *The concatenation of two terms is an expression if and only if both terms are themselves expressions.*
4. *Concatenation is associative.*
5. *The term ϕ is an expression if and only if ϕ is a glyph or is the concatenation of two expressions.*
6. *The mirror of an expression ϕ is defined as the concatenation of ϕ, $\star$, and ϕ (i.e., $\phi\star\phi$).*
7. *If ϕ is an expression, then $P\star\phi$, $PM\star\phi$, $\sim P\star\phi$, and $\sim PM\star\phi$ are sentences.*
8. *If ϕ is an expression then the sentence $P\star\phi$ is true if and only if ϕ is provable.*
9. *If ϕ is an expression, then the sentence $PM\star\phi$ is true if and only if the mirror of ϕ is provable.*
10. *If ϕ is an expression, then the sentence $\sim P\star\phi$ is true if and only if ϕ is not provable.*
11. *If ϕ is an expression, then the sentence $\sim PM\star\phi$ is true if and only if the mirror of ϕ is not provable.*
12. *Every sentence ϕ that is provable is also true.*

The given axioms (propositions 1–12) are represented visually in the Slate workspace in Figure 10, each consisting of the first-order formula derived from the English descriptions above. Moreover, a new intermediate hypothesis is introduced toward the desired goal, viz., that there is a true sentence that cannot be proved by $\mathcal{M}$:

13. *$\sim PM$ is an expression.*

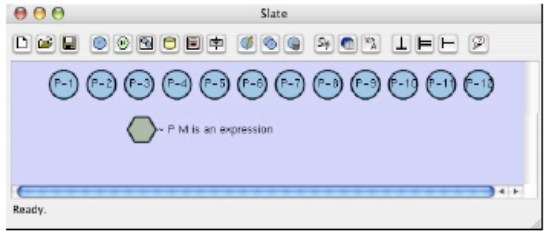


Figure 10. Propositions 1–12 and hypothesis 13 in the Slate workspace.

We indicate that hypothesis 13 is a logical consequence of propositions 2, 3 and 5 by drawing a deductive inference from each of these propositions to hypothesis 13 (Figure 11). Slate is then able to confirm or refute the added inference. Slate does indeed confirm that hypothesis 13 follows from the indicated propositions, by producing as evidence a formal proof which is added to the workspace as a *witness*. Witnesses are objects in Slate that support or weaken inferences. The double-plus symbol indicates that the witness confirms the argument, an ability reserved only for formal proofs. If the inference had been invalid, Slate might have produced a countermodel demonstrating the inference's invalidity.

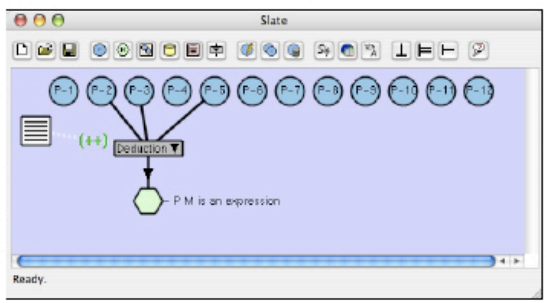


Figure 11. Proof of $\{2,3,5\}\vdash 13$ in the workspace and verified by Slate.

Having proved *$\sim PM$ is an expression*, it follows from 13 and 7 that:

14. *$\sim PM\star\sim PM$ is a sentence.*

If we suppose that *$\sim PM\star\sim PM$ is not true* then by 11 *the mirror of $\sim PM$ is provable* and thus by 6 *$\sim PM\star\sim PM$ is provable.* But then, according to 13 and 14, *$\sim PM\star\sim PM$ is true*—which is in contradiction with our supposition that *$\sim PM\star\sim PM$ is not true.* And so it must be the case that *$\sim PM\star\sim PM$ is true.* In other words, as shown in Figure 12, the hypothesis that

15. *$\sim PM\star\sim PM$ is true.*

follows from axioms 6 and 11 and hypotheses 12 and 13. Since *$\sim PM\star\sim PM$ is true*, it follows from 6 and 11 that

16. *$\sim PM\star\sim PM$ is not provable.*

and consequently, that *there is a true sentence which cannot be proved* (Figure 13).

5.3 Informal Reasoning

When using Slate, the reasoner is able to construct arguments that more closely resemble the uncertain and informal nature of everyday, natural inference. Moreover, the user benefits from the system's

Strength Factors

- Certain
- Evident
- Beyond Reasonable Doubt
- Likely
- Counterbalanced
- ... (symmetrical)

Piagetian Roboethics via Category Theory: Moving Beyond Mere Formal Operations to Engineer Robots Whose Decisions are Guaranteed to be Ethically Correct

Selmer Bringsjord, Joshua Taylor
Trevor Housten, Bram van Heuveln
Konstantine Arkoudas, Micah Clark
Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Rensselaer Polytechnic Institute (RPI)
Troy NY 12180 USA

Ralph Wojtowicz
Metron Inc.
1818 Library Street
Suite 600
Reston VA 20190 USA

I. Introduction

This is an extended abstract, not a polished paper; an *approach* to, rather than the results of, sustained research and development in the area of roboethics is described herein. Encapsulated, the approach is to engineer ethically correct robots by giving them the capacity to reason *over*, rather than merely *in*, logical systems (where logical systems are used to formalize such things as ethical codes of conduct for warfighting robots). This is to be accomplished by taking seriously Piaget's position that sophisticated human thinking exceeds even abstract processes carried out *in* a logical system, and by exploiting category theory to render in rigorous form, suitable for mechanization, structure-preserving mappings that Bringsjord, an avowed Piagetian, sees to be central in rigorous and rational human ethical decision-making.

We assume our readers to be at least somewhat familiar with elementary classical logic and category theory. Introductory coverage of the former subject can be found in [1], [2].[1] such coverage of the latter, offered from a suitably computational perspective, is provided in [3]. Additional references are of course provided in the course of this document.

II. Piaget's View of Thinking

Many people, including many outside psychology and cognitive science, know that Piaget seminally — and by Bringsjord's lights, correctly — articulated and defended the view that mature human reasoning and decision-making consists in processes operating for the most part on formulas in the language of classical extensional logic (e.g., see [4]).[2] You may yourself have this knowledge. You may also know that Piaget posited a sequence of cognitive stages through which humans, to varying degrees, pass. How many stages are there, according to Piaget? The received answer is: four; and in the fourth and final one, *formal operations*, neurobiologically normal humans can reason accurately and quickly over formulas expressed in the logical system known as first-order logic ($\mathcal{L}_I$).[3]

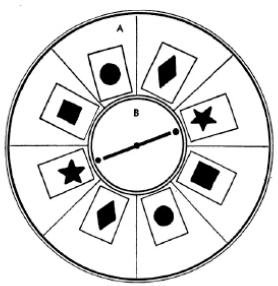


Fig. 1. Piaget's famous "rigged" rotating board to test for the development of Stage-3-or-better reasoning in children. The board, A, is divided into sectors of different colors and equal surfaces; opposite sectors match in color. B is a rotating disk with a metal rod spanning its diameter — but the catch is that the star cards have magnets buried under them (inside wax), so the alignment after spinning is invariably as shown here, no matter how the shapes are repositioned in the sectors (with matching shapes directly across from each other). This phenomenon is what subjects struggle to explain. Details can be found in [4].

Judging by the cognition taken by Piaget to be stage-three or stage-four (e.g., see Figure 1, which shows one

[1] Online, elegant, economical coverage can be found at http://plato.stanford.edu/entries/logic-classical/

[2] Many readers will know that Piaget's position long ago came under direct attack, by such thinkers as Wason and Johnson-Laird [5], [6]. In fact, unfortunately, for the most part people believe that this attack succeeded. Bringsjord doesn't agree in the least, but this isn't the place to visit the debate in question. Interested readers can consult [7], [8].

[3] Various other symbols are used, e.g., the more informative $\mathcal{L}_{\text{ooo}}$.

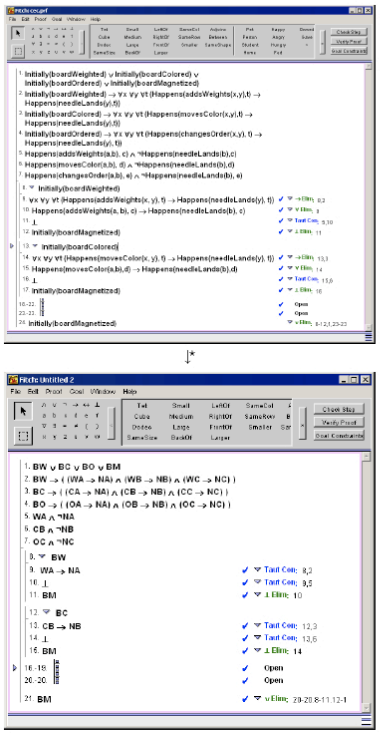


Fig. 2. This figure shows two proofs, one expressed in $\mathcal{L}_I$, the other in $\mathcal{L}_{PC}$. The first-order proof produces the conclusion that what causes the metal rod to invariably stop at the stars is that there are hidden magnets. The basic structure of this proof is proof by cases. Of the four disjuncts entertained as the possible source of the rod-star regularity, the right one is deduced when the others are eliminated. The functor $\star$ is shown here to indicate that the basic structure can be preserved in a proof couched exclusively in the propositional calculus.

see when deployed in warfare and counter-terrorism, where post-stage-four reasoning and decision-making is necessary for successfully handling these situations. The work here is connected to NSF-sponsored efforts on our part to extend CMU's Tekkotsu [20], [21] framework so that it includes operators that are central to our logicist approach to robotics, and specifically to roboethics — for example, operators for belief (**B**), knowledge (**K**), and obligation (◯ of standard deontic logic). The idea is that these operators would link to their counterparts in bona fide calculi for automated and semi-automated machine reasoning. One such calculus has already been designed and implemented: the *socio-cognitive calculus*; see [22]. This calculus includes the full event calculus.

Given that our initial experiments will make use of simple hand-eye robots recently acquired by the RAIR Lab from the Tekkotsu group at CMU, Figure 3, which shows one of these robots, sums up the situation (in connection with the magnet challenge). If sufficiently intricate manipulation cannot be achieved with the simple hand-eye robots, we will use the more powerful PERI, shown in Figure 4.

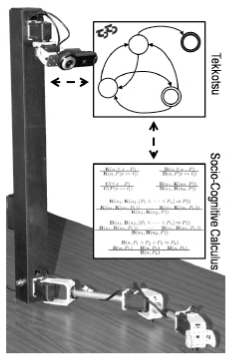


Fig. 3. The basic configuration for our initial implementations.

Fig. 4. The RAIR Lab's PERI

Piagetian Roboethics via Category Theory: Moving Beyond Mere Formal Operations to Engineer Robots Whose Decisions are Guaranteed to be Ethically Correct

Selmer Bringsjord, Joshua Taylor
Trevor Housten, Bram van Heuveln
Konstantine Arkoudas, Micah Clark
Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Rensselaer Polytechnic Institute (RPI)
Troy NY 12180 USA

Ralph Wojtowicz
Metron Inc.
1818 Library Street
Suite 600
Reston VA 20190 USA

I. Introduction

This is an extended abstract, not a polished paper; an *approach* to, rather than the results of, sustained research and development in the area of roboethics is described herein. Encapsulated, the approach is to engineer ethically correct robots by giving them the capacity to reason *over*, rather than merely *in*, logical systems (where logical systems are used to formalize such things as ethical codes of conduct for warfighting robots). This is to be accomplished by taking seriously Piaget's position that sophisticated human thinking exceeds even abstract processes carried out *in* a logical system, and by exploiting category theory to render in rigorous form, suitable for mechanization, structure-preserving mappings that Bringsjord, an avowed Piagetian, sees to be central in rigorous and rational human ethical decision-making.

We assume our readers to be at least somewhat familiar with elementary classical logic and category theory. Introductory coverage of the former subject can be found in [1], [2].[1] such coverage of the latter, offered from a suitably computational perspective, is provided in [3]. Additional references are of course provided in the course of this document.

II. Piaget's View of Thinking

Many people, including many outside psychology and cognitive science, know that Piaget seminally — and by Bringsjord's lights, correctly — articulated and defended the view that mature human reasoning and decision-making consists in processes operating for the most part on formulas in the language of classical extensional logic (e.g., see [4]).[2] You may yourself have this knowledge. You may also know that Piaget posited a sequence of cognitive stages through which humans, to varying degrees, pass. How many stages are there, according to Piaget? The received answer is: four; and in the fourth and final one, *formal operations*, neurobiologically normal humans can reason accurately and quickly over formulas expressed in the logical system known as first-order logic ($\mathcal{L}_I$).[3]

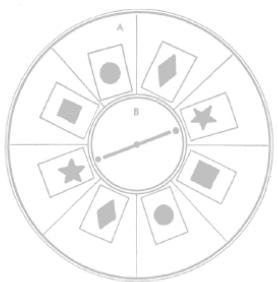


Fig. 1. Piaget's famous "rigged" rotating board to test for the development of Stage-3-or-better reasoning in children. The board, A, is divided into sectors of different colors and equal surfaces; opposite sectors match in color. B is a rotating disk with a metal rod spanning its diameter — but the catch is that the star cards have magnets buried under them (inside wax), so the alignment after spinning is invariably as shown here, no matter how the shapes are repositioned in the sectors (with matching shapes directly across from each other). This phenomenon is what subjects struggle to explain. Details can be found in [4].

Judging by the cognition taken by Piaget to be stage-three or stage-four (e.g., see Figure 1, which shows one

[1] Online, elegant, economical coverage can be found at http://plato.stanford.edu/entries/logic-classical/

[2] Many readers will know that Piaget's position long ago came under direct attack, by such thinkers as Wason and Johnson-Laird [5], [6]. In fact, unfortunately, for the most part people believe that this attack succeeded. Bringsjord doesn't agree in the least, but this isn't the place to visit the debate in question. Interested readers can consult [7], [8].

[3] Various other symbols are used, e.g., the more informative $\mathcal{L}_{\text{ASA}}$.

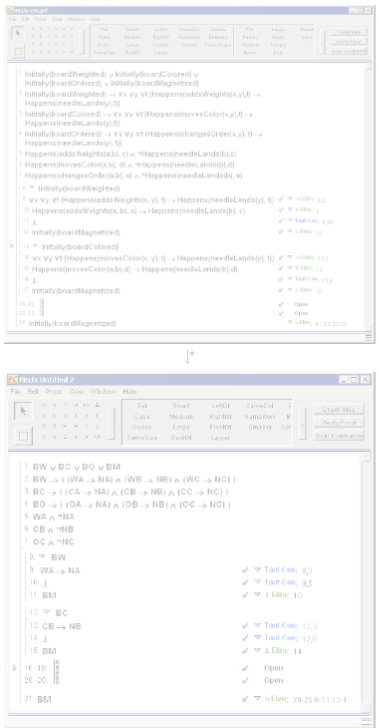


Fig. 2. This figure shows two proofs, one expressed in $\mathcal{L}_I$, the other in $\mathcal{L}_{PC}$. The first-order proof produces the conclusion that what causes the metal rod to invariably stop at the stars is that there are hidden magnets. The basic structure of this proof is proof by cases. Of the four disjuncts entertained as the possible source of the rod-star regularity, the right one is deduced when the others are eliminated. The functor $\star$ is shown here to indicate that the basic structure can be preserved in a proof couched exclusively in the propositional calculus.

see when deployed in warfare and counter-terrorism, where post-stage-four reasoning and decision-making is necessary for successfully handling these situations. The work here is connected to NSF-sponsored efforts on our part to extend CMU's Tekkotsu [20], [21] framework so that it includes operators that are central to our logicist approach to robotics, and specifically to roboethics — for example, operators for belief (**B**), knowledge (**K**), and obligation (◯ of standard deontic logic). The idea is that these operators would link to their counterparts in bona fide calculi for automated and semi-automated machine reasoning. One such calculus has already been designed and implemented: the *socio-cognitive calculus*; see [22]. This calculus includes the full event calculus.

Given that our initial experiments will make use of simple hand-eye robots recently acquired by the RAIR Lab from the Tekkotsu group at CMU, Figure 3, which shows one of these robots, sums up the situation (in connection with the magnet challenge). If sufficiently intricate manipulation cannot be achieved with the simple hand-eye robots, we will use the more powerful PERI, shown in Figure 4.

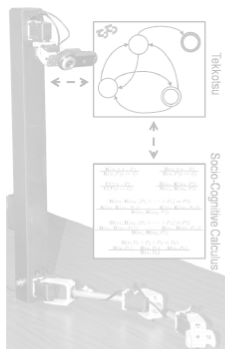


Fig. 3. The basic configuration for our initial implementations.

Fig. 4. The RAIR Lab's PERI

Solving Problem 3: Piagetian Roboethics via Category Theory ...

Piagetian Roboethics via Category Theory: Moving Beyond Mere Formal Operations to Engineer Robots Whose Decisions are Guaranteed to be Ethically Correct

Selmer Bringsjord, Joshua Taylor
Trevor Housten, Bram van Heuveln
Konstantine Arkoudas, Micah Clark
Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Rensselaer Polytechnic Institute (RPI)
Troy NY 12180 USA

Ralph Wojtowicz
Metron Inc.
1818 Library Street
Suite 600
Reston VA 20190 USA

I. Introduction

This is an extended abstract, not a polished paper; an *approach* to, rather than the results of, sustained research and development in the area of roboethics is described herein. Encapsulated, the approach is to engineer ethically correct robots by giving them the capacity to reason *over*, rather than merely *in*, logical systems (where logical systems are used to formalize such things as ethical codes of conduct for warfighting robots). This is to be accomplished in part by seriously Piaget's position that sophisticated human thinking exceeds even abstract processes carried out *in* a logical system, and by exploiting category theory to render it rigorous form, suitable for mechanization, structure-preserving mappings that Bringsjord, an avowed Piagetian, sees to be central in rigorous and rational human ethical decision-making.

We assume our readers to be at least somewhat familiar with elementary classical logic and category theory. Introductory coverage of the former subject can be found in [1], [2].[1] such coverage of the latter, offered from a suitably computational perspective, is provided in [3]. Additional references are of course provided in the course of this document.

II. Piaget's View of Thinking

Many people, including many outside psychology and cognitive science, know that Piaget seminally — and by Bringsjord's lights, correctly — articulated and defended the view that mature human reasoning and decision-making consists in processes operating for the most part on formulas in the language of classical extensional logic (e.g., see [4]).[2]

[1]Online, elegant, economical coverage can be found at http://plato.stanford.edu/entries/logic-classical/

[2]Many readers will know that Piaget's position long ago came under direct attack, by such thinkers as Wason and Johnson-Laird [5], [6]. In fact, unfortunately, for the most part people believe that this attack succeeded. Bringsjord doesn't agree in the least, but this isn't the place to visit the debate in question. Interested readers can consult [7], [8].

You may you'll have thought now. You're also aware that Piaget posited a sequence of cognitive stages through which children, by varying degrees, pass. How many stages are there according to Piaget? The received answer: four; and in the fourth and final one, *formal operations*, neurobiologically normal humans can reason accurately and quickly over formulas expressed in the logical system known as first-order logic ($\mathcal{L}_I$).[3]

Fig. 1. Piaget's famous "rigged" rotating board to test for the development of Stage-3-or-better reasoning in children. The board, A, is divided into sectors of different colors and equal surfaces; opposite sectors match in color. B is a rotating disk with a metal rod spanning its diameter — but the catch is that the star cards have magnets buried under them (inside wax), so the alignment after spinning is invariably as shown here, no matter how the shapes are repositioned in the sectors (with matching shapes directly across from each other). This phenomenon is what subjects struggle to explain. Details can be found in [4].

Judging by the cognition taken by Piaget to be stage-three or stage-four (e.g., see Figure 1, which shows one

[3]Various other symbols are used, e.g., the more informative $\mathcal{L}_{\omega\omega}$.

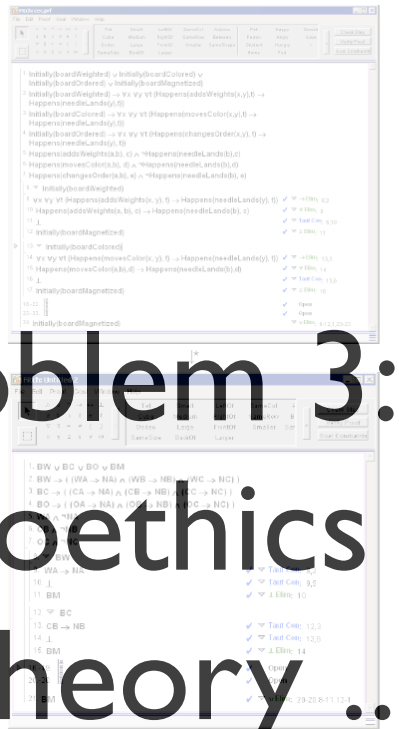


Fig. 2. This figure shows two proofs, one expressed in $\mathcal{L}_I$, the other in $\mathcal{L}_{PC}$. The first-order proof produces the conclusion that what causes the metal rod to invariably stop at the stars is that there are hidden magnets. The basic structure of this proof is proof by cases. Of the four disjuncts entertained as the possible source of the rod-star regularity, the right one is deduced when the others are eliminated. The functor $\star$ is shown here to indicate that the basic structure can be preserved in a proof couched exclusively in the propositional calculus.

see when deployed in warfare and counter-terrorism, where post-stage-four reasoning and decision-making is necessary for successfully handling these situations. The work here is connected to NSF-sponsored efforts on our part to extend CMU's Tekkotsu [20], [21] framework so that it includes operators that are central to our logicist approach to robotics, and specifically to roboethics — for example, operators for belief (**B**), knowledge (**K**), and obligation (◯ of standard deontic logic). The idea is that these operators would link to their counterparts in bona fide calculi for automated and semi-automated machine reasoning. One such calculus has already been designed and implemented: the *socio-cognitive calculus*; see [22]. This calculus includes the full event calculus.

Given that our initial experiments will make use of simple hand-eye robots recently acquired by the RAIR Lab from the Tekkotsu group at CMU, Figure 3, which shows one of these robots, sums up the situation (in connection with the magnet challenge). If sufficiently intricate manipulation cannot be achieved with the simple hand-eye robots, we will use the more powerful PERI, shown in Figure 4.

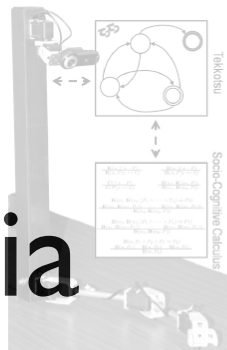


Fig. 3. The basic configuration for our initial implementations.

Fig. 4. The RAIR Lab's PERI

Absolutely Crucial for AI, Robotics, Roboethics:

Betting the farm on one or two logical systems (e.g., FOL, propositional calculus).

Absolutely Crucial for AI, Robotics, Roboethics:

Betting the farm on one or two logical systems (e.g., FOL, propositional calculus).

versus

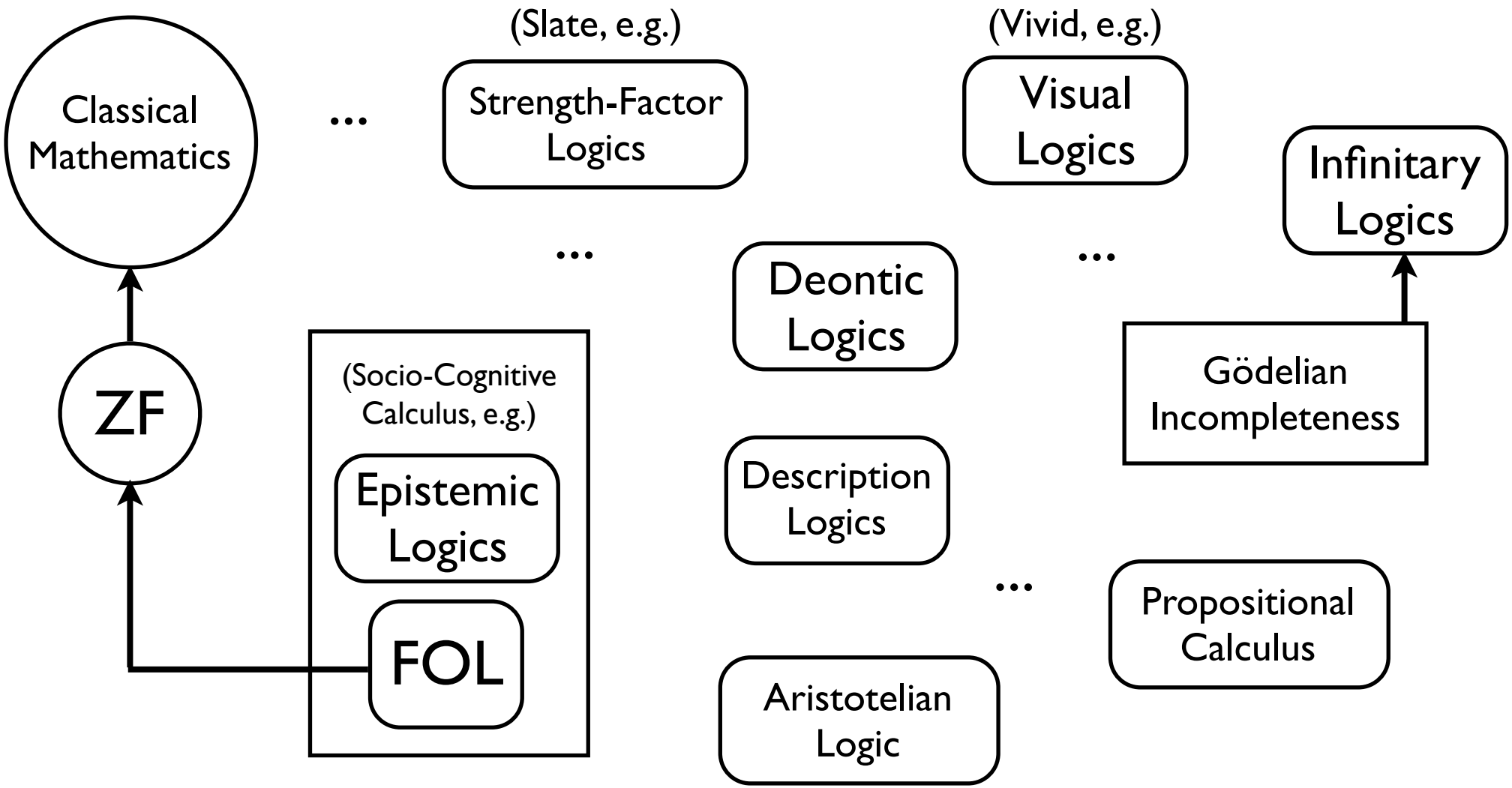
Absolutely Crucial for AI, Robotics, Roboethics:

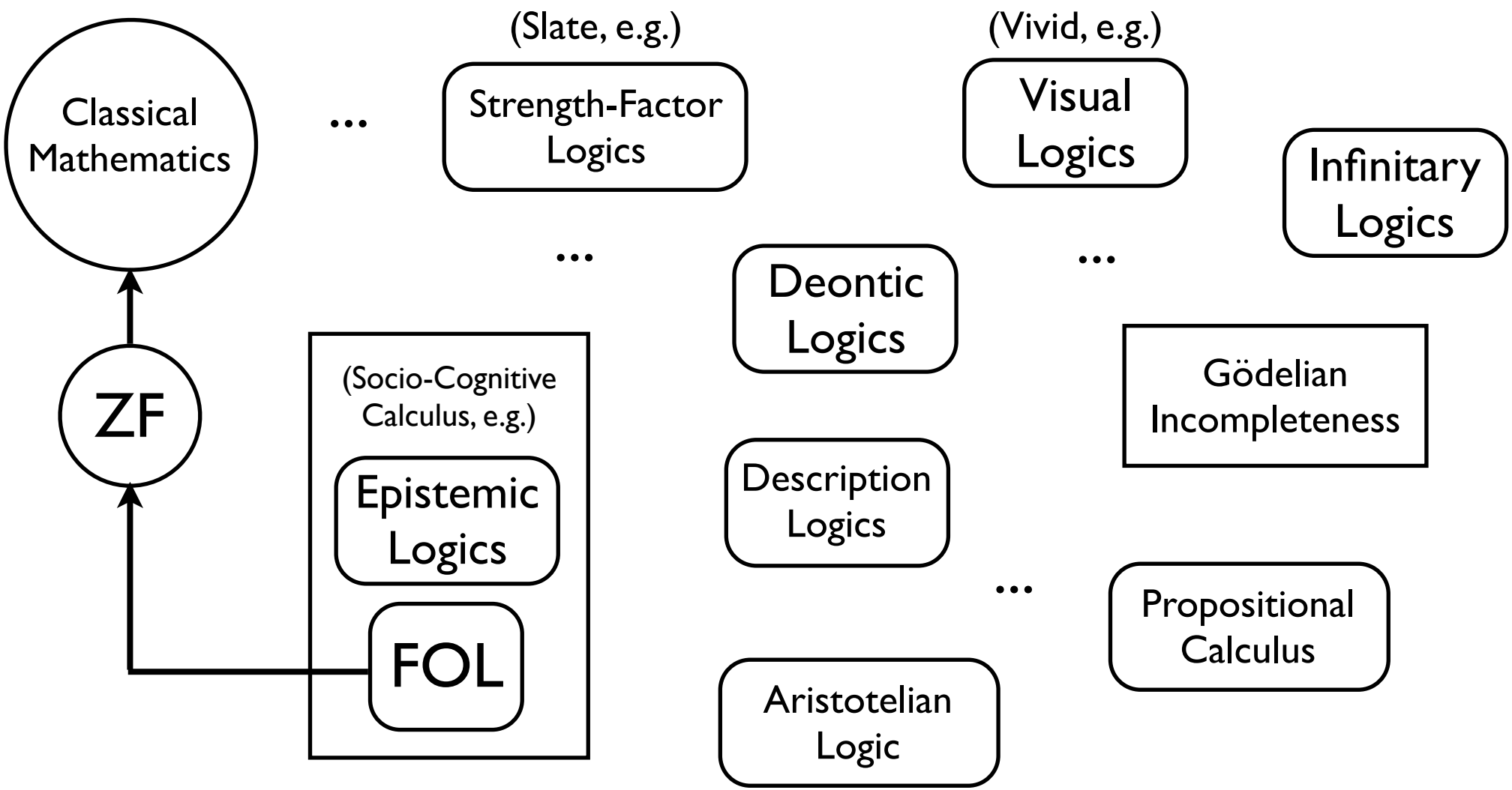
Betting the farm on one or two logical systems (e.g., FOL, propositional calculus).

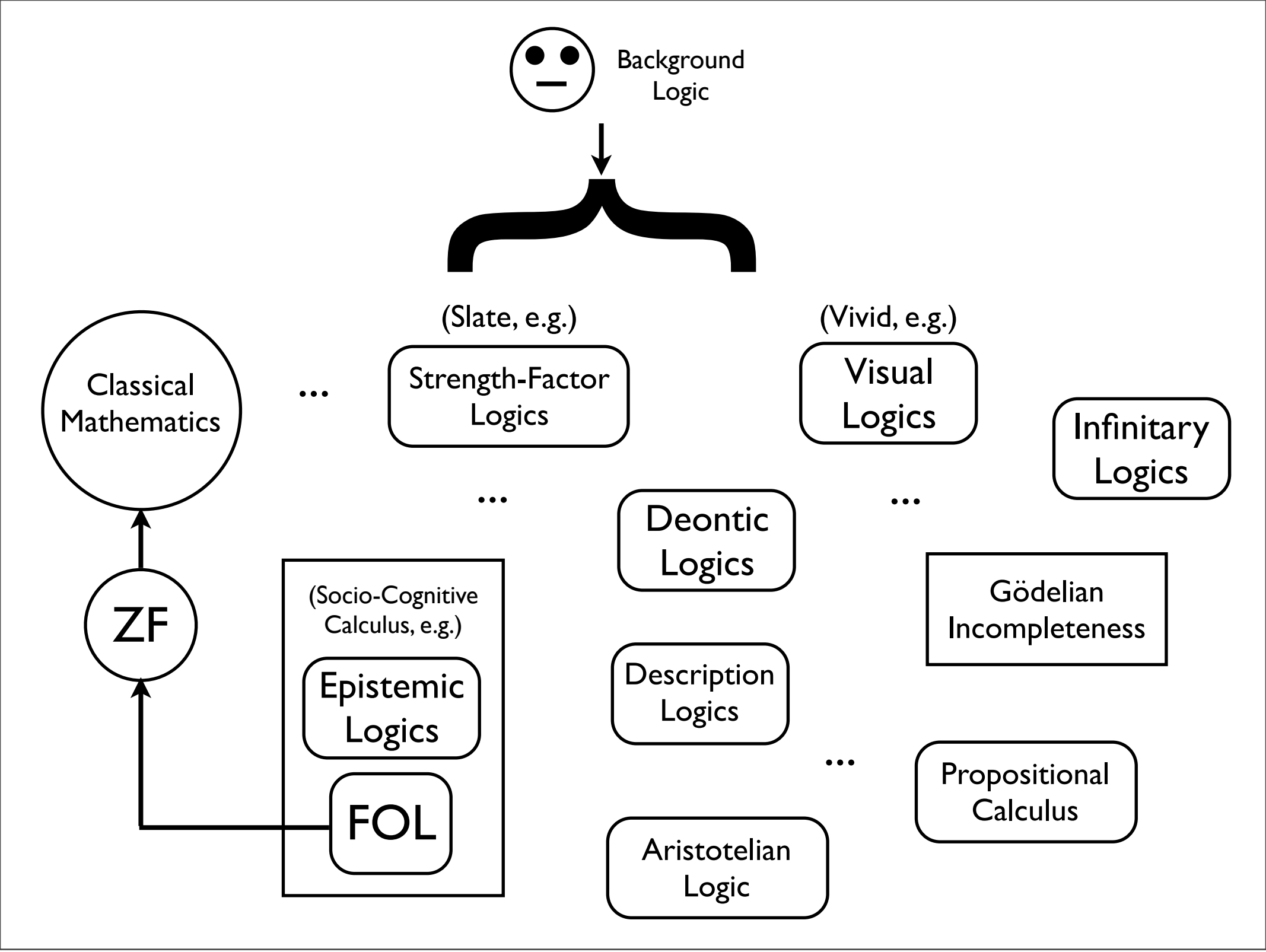
versus

We know humans operate in ways that range *across* an infinite number of logical systems, so we need a formal theory, and a corresponding set of processes, that captures the meta-coordination of myriad logical systems.

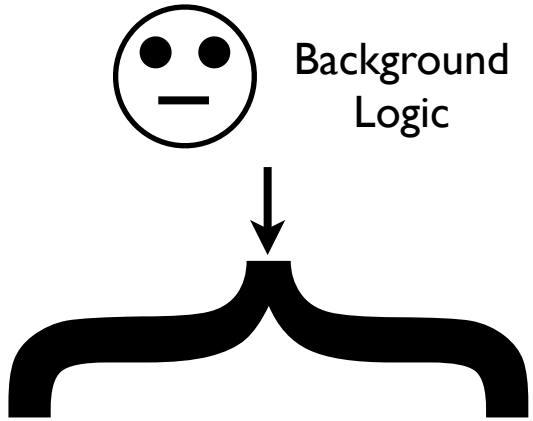
The Space of Logical Systems




Classical Mathematics
ZF
...
(Slate, e.g.)
Strength-Factor Logics
...
(Vivid, e.g.)
Visual Logics
...
Infinitary Logics
Deontic Logics
Gödelian Incompleteness
(Socio-Cognitive Calculus, e.g.)
Epistemic Logics
FOL
Description Logics
...
Propositional Calculus
Aristotelian Logic


Background Logic
(Slate, e.g.)
(Vivid, e.g.)
Classical Mathematics
...
Strength-Factor Logics
Visual Logics
Infinitary Logics
...
...
Deontic Logics
Gödelian Incompleteness
ZF
(Socio-Cognitive Calculus, e.g.)
Epistemic Logics
Description Logics
...
Propositional Calculus
FOL
Aristotelian Logic

Inspired by Piaget's five-stage view.



Background Logic

(Slate, e.g.)

(Vivid, e.g.)

Classical Mathematics

...

Strength-Factor Logics

Visual Logics

Infinitary Logics

...

...

Deontic Logics

Gödelian Incompleteness

ZF

(Socio-Cognitive Calculus, e.g.)

Epistemic Logics

Description Logics

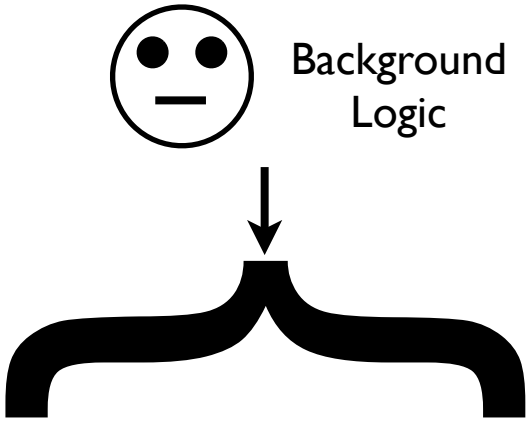
...

Propositional Calculus

FOL

Aristotelian Logic

Inspired by Piaget's five-stage view.



Simon seemed to be starting to face up to the daunting reality shortly before his death.

Classical Mathematics

...

(Slate, e.g.)

Strength-Factor Logics

(Vivid, e.g.)

Visual Logics

Infinitary Logics

...

...

Deontic Logics

Gödelian Incompleteness

ZF

(Socio-Cognitive Calculus, e.g.)

Epistemic Logics

Description Logics

...

Propositional Calculus

FOL

Aristotelian Logic

One promising approach to taming this formally: category theory, where categories are logical systems.

Categories

- A category comprises a collection of *objects* and a collection of *arrows* (or *morphisms*).
  - Each arrow has a *domain* (or *source*) and a *codomain* (or *target*).
  - For each object A there is an identity arrow $id_A : A \to A$.
  - For arrows $f : A \to B$ and $g : B \to C$, there is an arrow $g \circ f : A \to C$.

Functors

- A functor is a pair of mappings comprising an *object mapping* and an *arrow mapping*.
- A functor $F: \mathbf{C} \rightarrow \mathbf{D}$ maps each object of **C** to an object in **D**, and each arrow of **C** to to an arrow of **D**, such that:
  - If $f: A \rightarrow B$ is in **C**, then $F(f): F(A) \rightarrow F(B)$ is in **D**.
  - For every A in **C**, $F(id_A) = id_{F(A)}$.
  - $F(g \circ f) = F(g) \circ F(f)$

Deductive Systems

- Deductive Systems are categories whose objects are sentences and whose arrows are proofs.
- Identity arrows are typically applications of reiteration rules, and proofs typically compose.
- Other inference rules can be presented schematically. E.g, conditional elimination:

$$\frac{\gamma \xrightarrow{f} \phi \Rightarrow \psi \qquad \gamma \xrightarrow{g} \phi}{\gamma \xrightarrow{\Rightarrow \text{elim } f,g} \psi} \; [\Rightarrow \text{ elim}]$$

Four Logical Systems as Categories

- The Propositional Calculus
- The First Order Predicate Calculus (and its truth-functional subsystem)
- Propositional **S5**
- The Description Logic $\mathcal{ALC}$

Functors Specified (by Joshua Taylor)

It then becomes easy to prove:

Theorem: If $\Phi \vdash_{PS5} \phi$ then $F(\Phi)F(\vdash_{PS5}) / \vdash_{\mathcal{F}} F(\phi)$

.

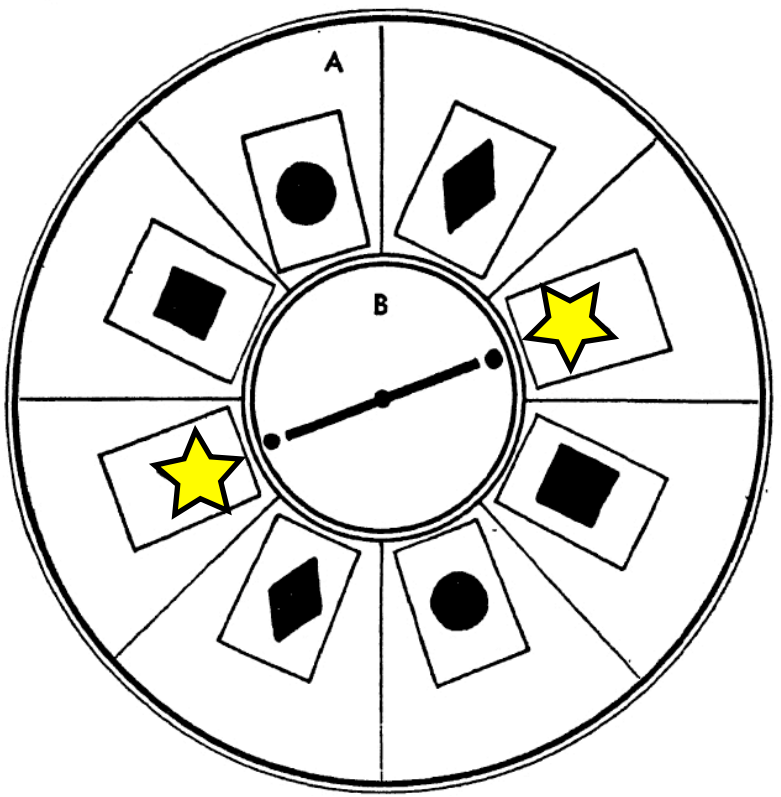
.

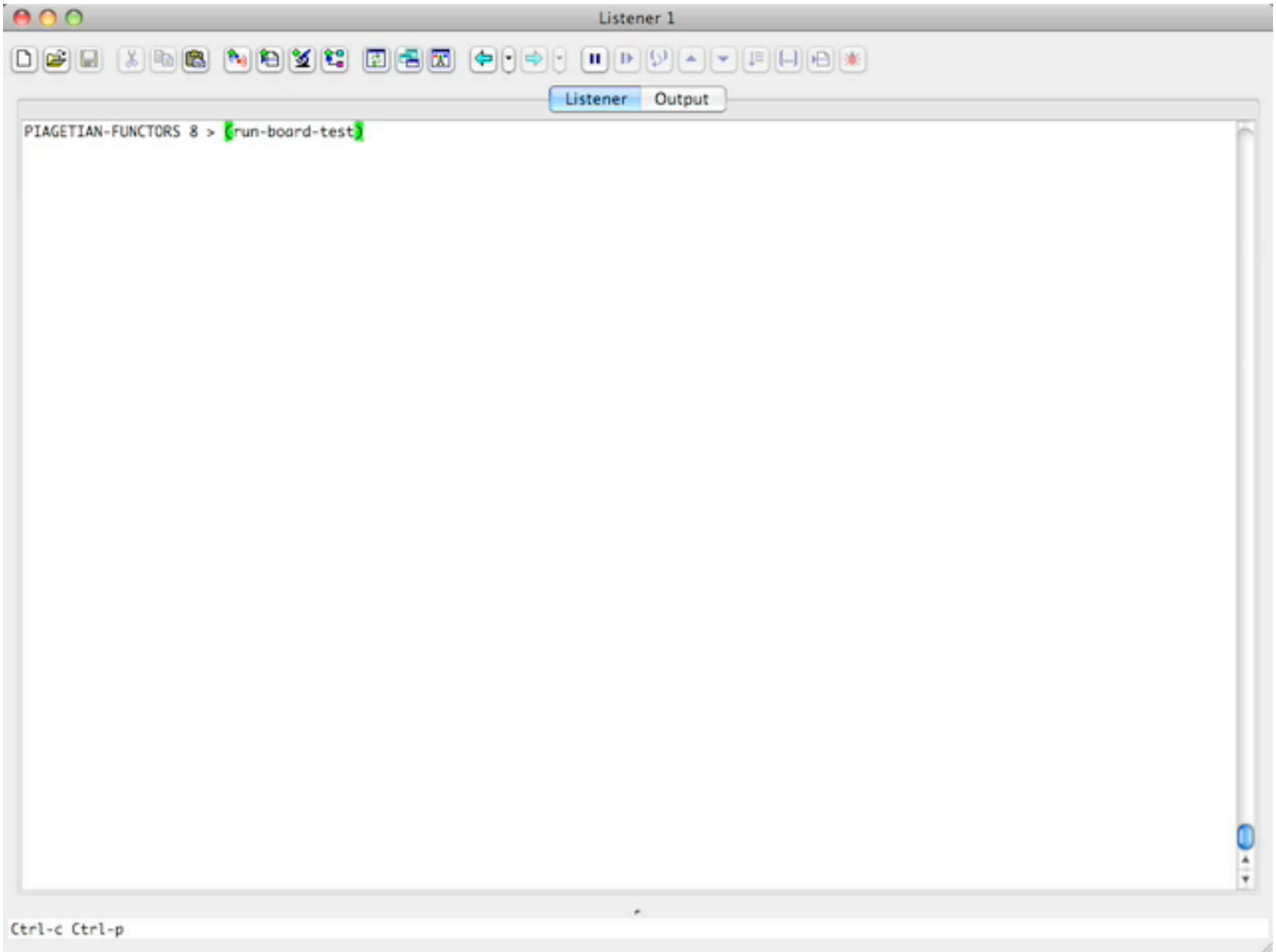
.

Example ...

Piaget's Magnetic Board

- A needle mounted to a board, but able to spin freely always stops at the yellow stars.
- How does a human reasoner approach the problem?
- Ideally, by considering and testing some hypotheses




Listener 1
Listener
Output
PIAGETIAN-FUNCTORS 8 > (run-board-test)
Ctrl-c Ctrl-p

Second Example ...

Chisholm's Contrary-to-Duty Paradox

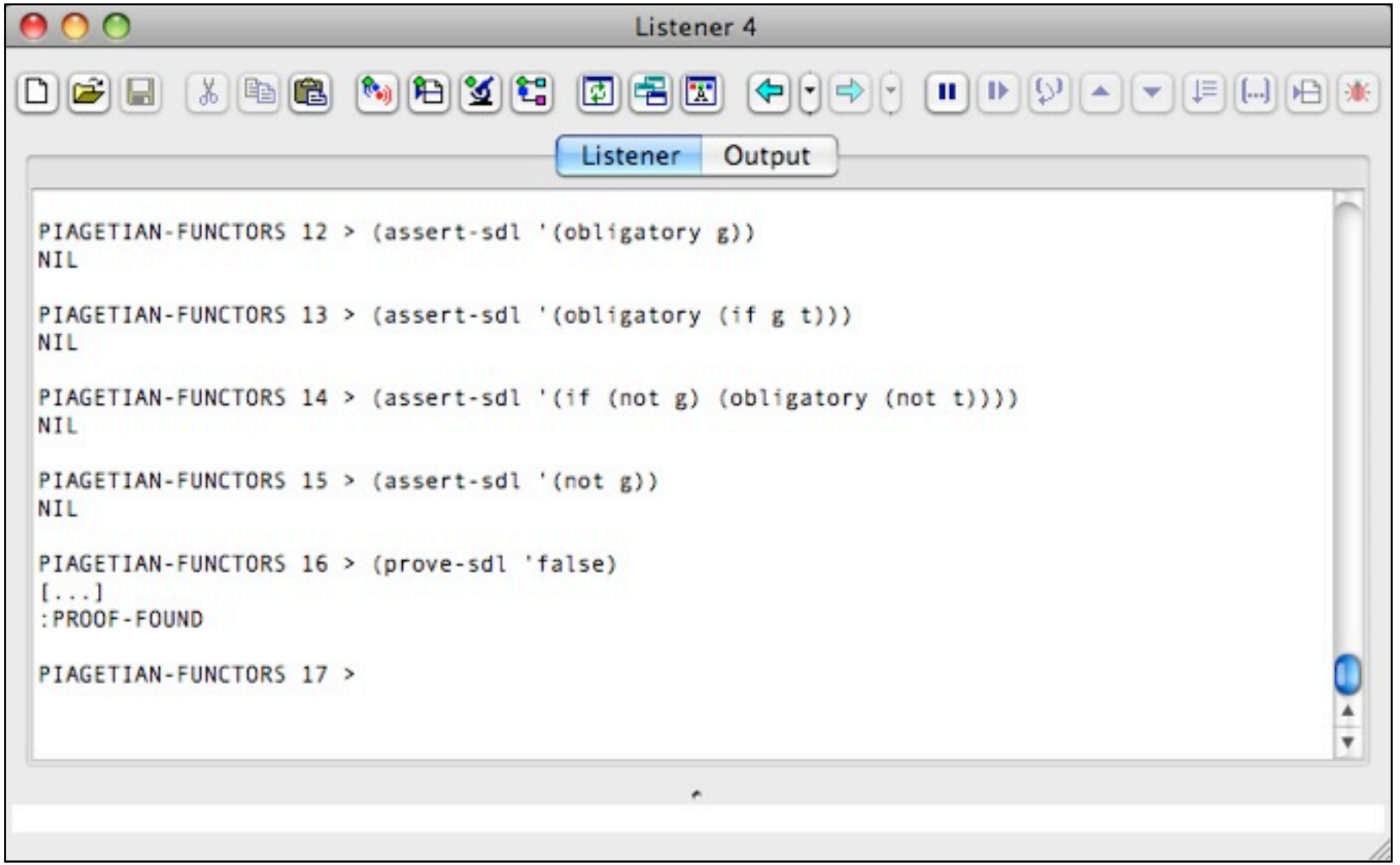
- Let us suppose:

1. It ought to be that a certain man go to the assistance of his neighbours.
2. It ought to be that if he does go he tells them he is coming. But
3. if he does not go then he ought not to tell them he is coming; and
4. he does not go.

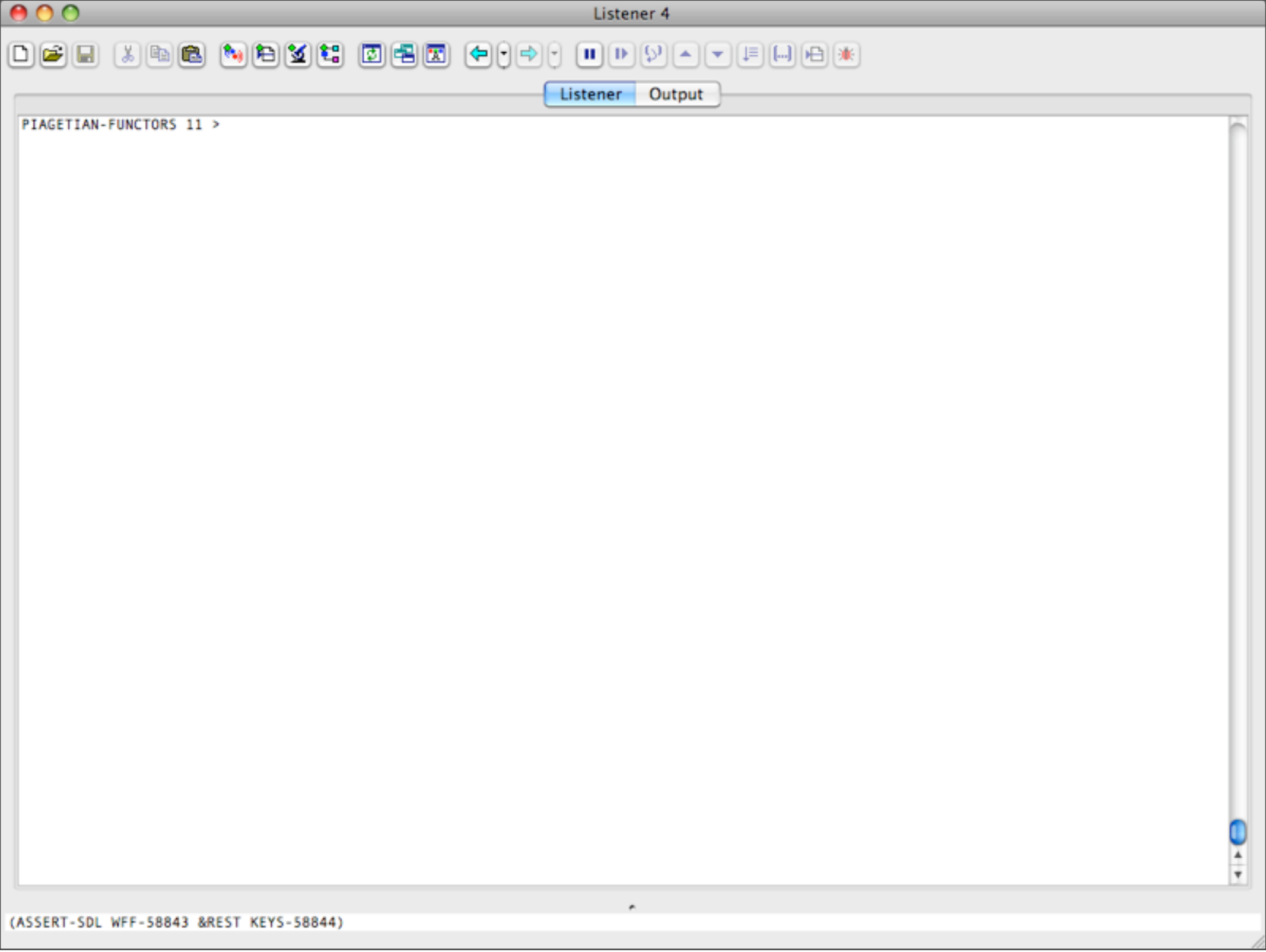
The Paradox

- Within Standard Deontic Logic, SDL, this leads to a paradox, for the man both ought to tell his neighbours that he is coming, and ought not to tell them that he is coming.

1	$\mathbf{OB}g$	
2	$\mathbf{OB}(g \Rightarrow t)$	
3	$\neg g \Rightarrow \mathbf{OB}\neg t$	
4	$\neg g$	
5	$\mathbf{OB}\neg t$	$\Rightarrow$ elim, 3, 4
6	$\mathbf{OB}g \Rightarrow \mathbf{OB}t$	**OB**-dist, 2
7	$\mathbf{OB}t$	$\Rightarrow$ elim, 1, 6
8	$\mathbf{PE}t$	**OB-D**, 7
9	$\neg\mathbf{OB}\neg t$	def. **PE**, 8
10	$\bot$	$\bot$ intro, 5, 9



The functor translates both sentences and proofs, so the contradiction is still present, even when the reasoner is first-order.



Monday, June 8, 2009

Handling Chisholm's paradox an absolute requirement:

Robots, like humans, will inevitably fail to meet some obligations, giving rise to situations where their subsequent obligations are of a particular nature. Without contrary-to-duty imperatives handled, robots (and humans) will spin out of ethical control.

Coming

- The formalisms applied to more militarily relevant situations.
- It would be nice to have some lethal robots to play with.

Finally,

Finally,

what could possibly be an alternative approach to solving the problem?

Logic is Our Only Hope

We only have one way to fix the meaning of programs, to verify that they will behave as advertised.

We only have one way to rigorously set out and mechanize sophisticated ethical reasoning, and to impart that reasoning to autonomous lethal robots.

Logic is our only hope, ladies and gentlemen.

Logic is Our Only Hope

We only have one way to fix the meaning of programs, to verify that they will behave as advertised.

We only have one way to rigorously set out and mechanize sophisticated ethical reasoning, and to impart that reasoning to autonomous lethal robots.

Logic is our only hope, ladies and gentlemen.

Finis