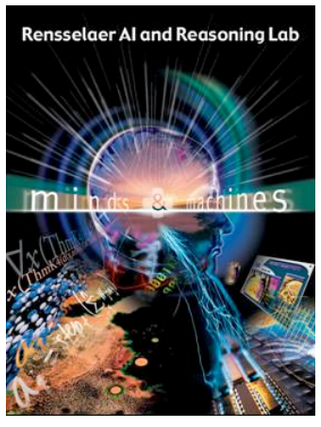
The Missing Math for Modeling, Simulating, and AI-Boosting Human Decision-Making in Multi-Agent Reasoning Environments

Selmer Bringsjord & Alexander Bringsjord
Department of Cognitive Science
Department of Computer Science
Rensselaer AI & Reasoning (RAIR) Lab
Rensselaer Polytechnic Institute (RPI)
Troy NY 12180 USA
selmer@rpi.edu
USMA 6.16.09

*Selmer and Alexander are profoundly grateful to ARDA/DTO/IARPA and DARPA for support, to Konstantine Arkoudas, whose brilliance makes all such R&D possible, and to Will Tracy for introducing us to CSP, and a promising evolutionary approach to it (that we nonetheless :) don't follow).

Rensselaer | DEPARTMENT OF COGNITIVE SCIENCE

Rensselaer — Computer Science

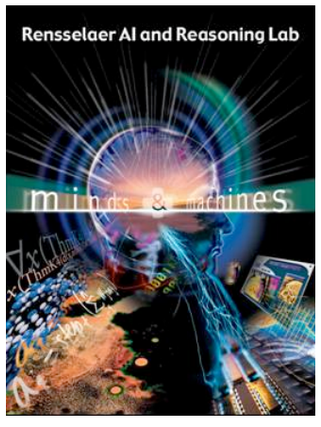


Selmer Bringsjord & Alexander Bringsjord
Department of Cognitive Science
Department of Computer Science
Rensselaer AI & Reasoning (RAIR) Lab
Rensselaer Polytechnic Institute (RPI)
Troy NY 12180 USA
selmer@rpi.edu
USMA 6.16.09

*Selmer and Alexander are profoundly grateful to ARDA/DTO/IARPA and DARPA for support, to Konstantine Arkoudas, whose brilliance makes all such R&D possible, and to Will Tracy for introducing us to CSP, and a promising evolutionary approach to it (that we nonetheless :) don't follow).

Rensselaer | DEPARTMENT OF COGNITIVE SCIENCE

Rensselaer — Computer Science



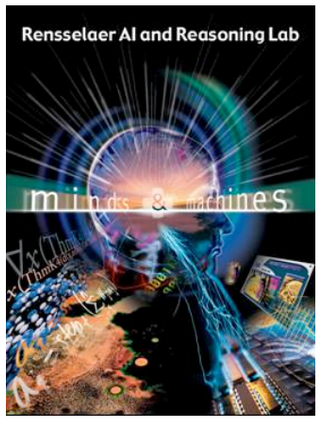
The Missing Logico-Mathematics for Modeling, Simulating, and AI-Boosting Human Decision-Making in Multi-Agent Reasoning Environments

Selmer Bringsjord & Alexander Bringsjord
Department of Cognitive Science
Department of Computer Science
Rensselaer AI & Reasoning (RAIR) Lab
Rensselaer Polytechnic Institute (RPI)
Troy NY 12180 USA
selmer@rpi.edu
USMA 6.16.09

*Selmer and Alexander are profoundly grateful to ARDA/DTO/IARPA and DARPA for support, to Konstantine Arkoudas, whose brilliance makes all such R&D possible, and to Will Tracy for introducing us to CSP, and a promising evolutionary approach to it (that we nonetheless :) don't follow).

Rensselaer | DEPARTMENT OF COGNITIVE SCIENCE

Rensselaer — Computer Science



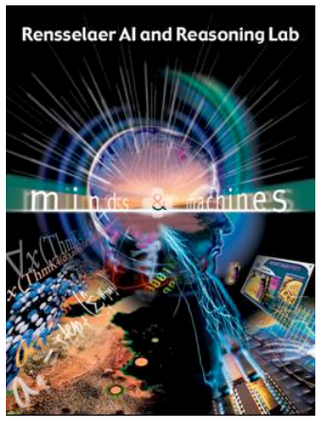
The Missing Logico-Mathematics for Modeling, Simulating, and AI-Boosting Human Decision-Making in Multi-Agent Reasoning Environments

Selmer Bringsjord & Alexander Bringsjord
Department of Cognitive Science
Department of Computer Science
Rensselaer AI & Reasoning (RAIR) Lab
Rensselaer Polytechnic Institute (RPI)
Troy NY 12180 USA
selmer@rpi.edu
USMA 6.16.09

*Selmer and Alexander are profoundly grateful to ARDA/DTO/IARPA and DARPA for support, to Konstantine Arkoudas, whose brilliance makes all such R&D possible, and to Will Tracy for introducing us to CSP, and a promising evolutionary approach to it (that we nonetheless :) don't follow).

Rensselaer | DEPARTMENT OF COGNITIVE SCIENCE

Rensselaer — Computer Science



The Missing Logico-Mathematics for Modeling, Simulating, and AI-Boosting Human Decision-Making in Multi-Agent Reasoning Environments

(via focus on the Chain Store Paradox/Nuclear-Club Paradox)*

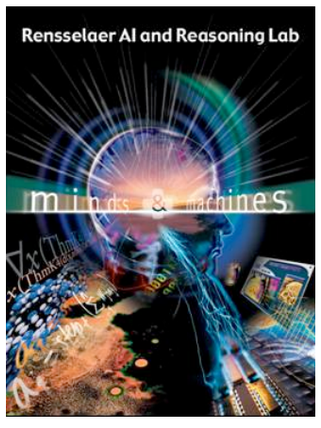
Selmer Bringsjord & Alexander Bringsjord
Department of Cognitive Science
Department of Computer Science
Rensselaer AI & Reasoning (RAIR) Lab
Rensselaer Polytechnic Institute (RPI)
Troy NY 12180 USA
selmer@rpi.edu

USMA 6.16.09

*Selmer and Alexander are profoundly grateful to ARDA/DTO/IARPA and DARPA for support, to Konstantine Arkoudas, whose brilliance makes all such R&D possible, and to Will Tracy for introducing us to CSP, and a promising evolutionary approach to it (that we nonetheless :) don't follow).

Rensselaer | DEPARTMENT OF COGNITIVE SCIENCE

Rensselaer — Computer Science



Approach:
Logic/Formal Methods-Based
AI, Computational Cognitive Science, & Computer Science ...

THE CAMBRIDGE HANDBOOK OF

Computational Psychology

EDITED BY

Ron Sun

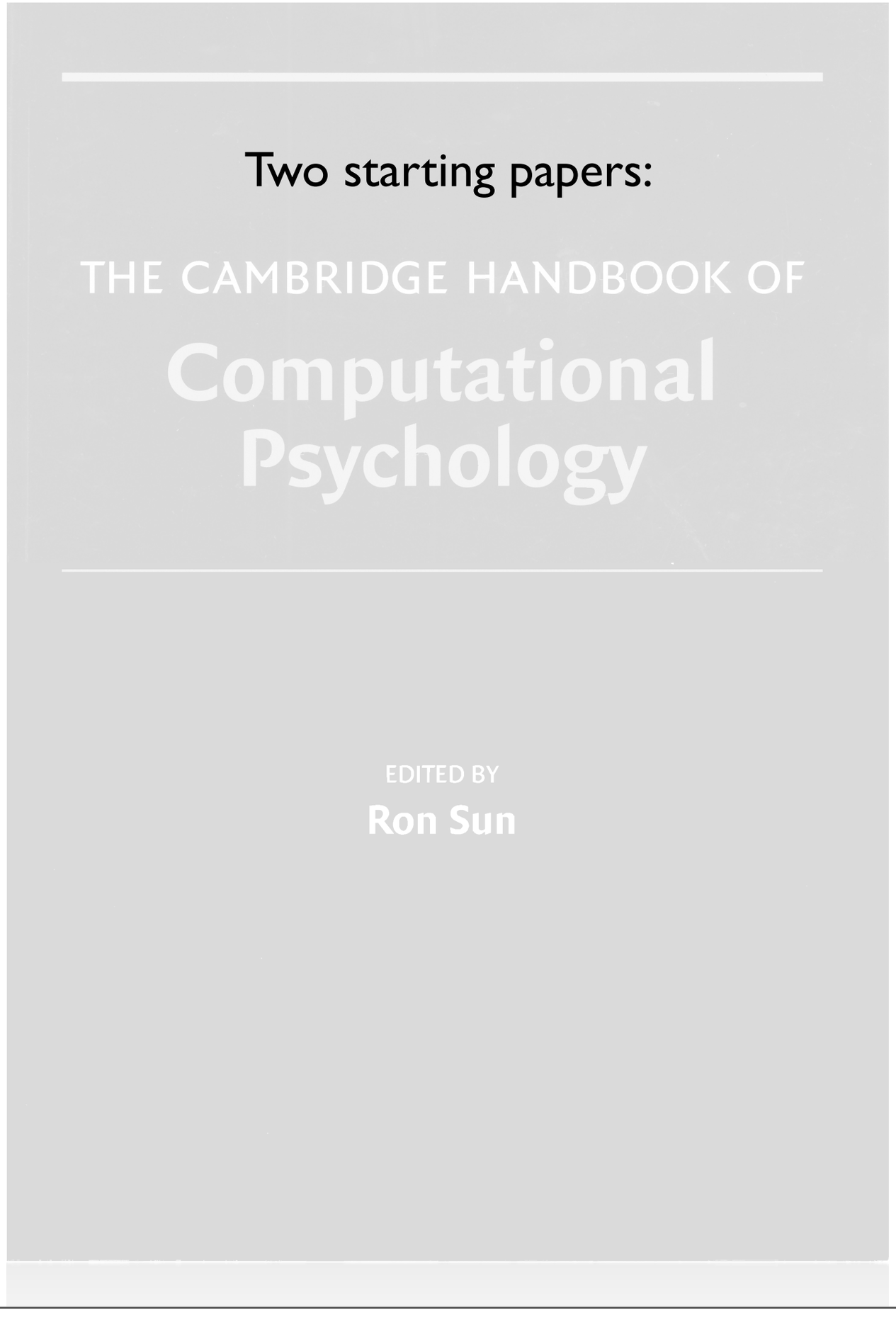
THE CAMBRIDGE HANDBOOK OF

Computational Psychology

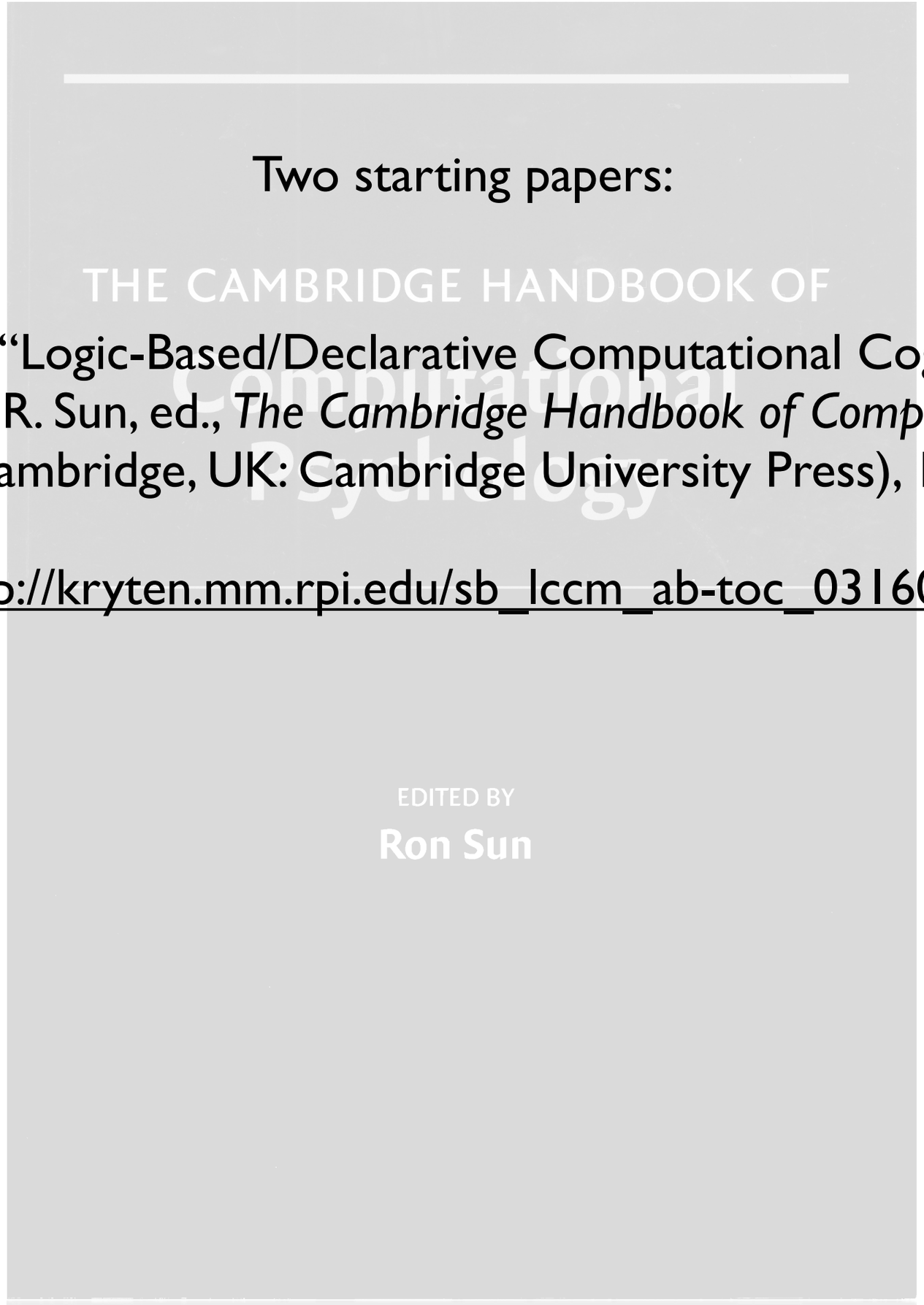
EDITED BY

Ron Sun

Two starting papers:



Two starting papers:



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

Two starting papers:

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

Two starting papers:

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

Decidedly not Bayesian, no use of probability. And wholly astatistical. Uncertainty handled by strength-factor based reasoning.

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

Computers, Justification, and Mathematical Knowledge

Konstantine Arkoudas · Selmer Bringsjord

Published online: 23 June 2007
© Springer Science+Business Media B.V. 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

Published online: 23 June 2007
© Springer Science+Business Media B.V. 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

Wednesday, October 21, 2009

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

Computers, Justification, and Mathematical Knowledge

Konstantine Arkoudas · Selmer Bringsjord

Published online: 23 June 2007
© Springer Science+Business Media B.V. 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

Must have machine-checked proofs, by verified proof checker.

Avoid Limitations of *Elementary* Logic-Based R&D

Betting the farm on one or two logical systems (e.g., FOL, propositional calculus)—or for that matter on a particular theory within a logical system (e.g. Game Theory, probability calculus).

Avoid Limitations of *Elementary* Logic-Based R&D

Betting the farm on one or two logical systems (e.g., FOL, propositional calculus)—or for that matter on a particular theory within a logical system (e.g. Game Theory, probability calculus).

versus

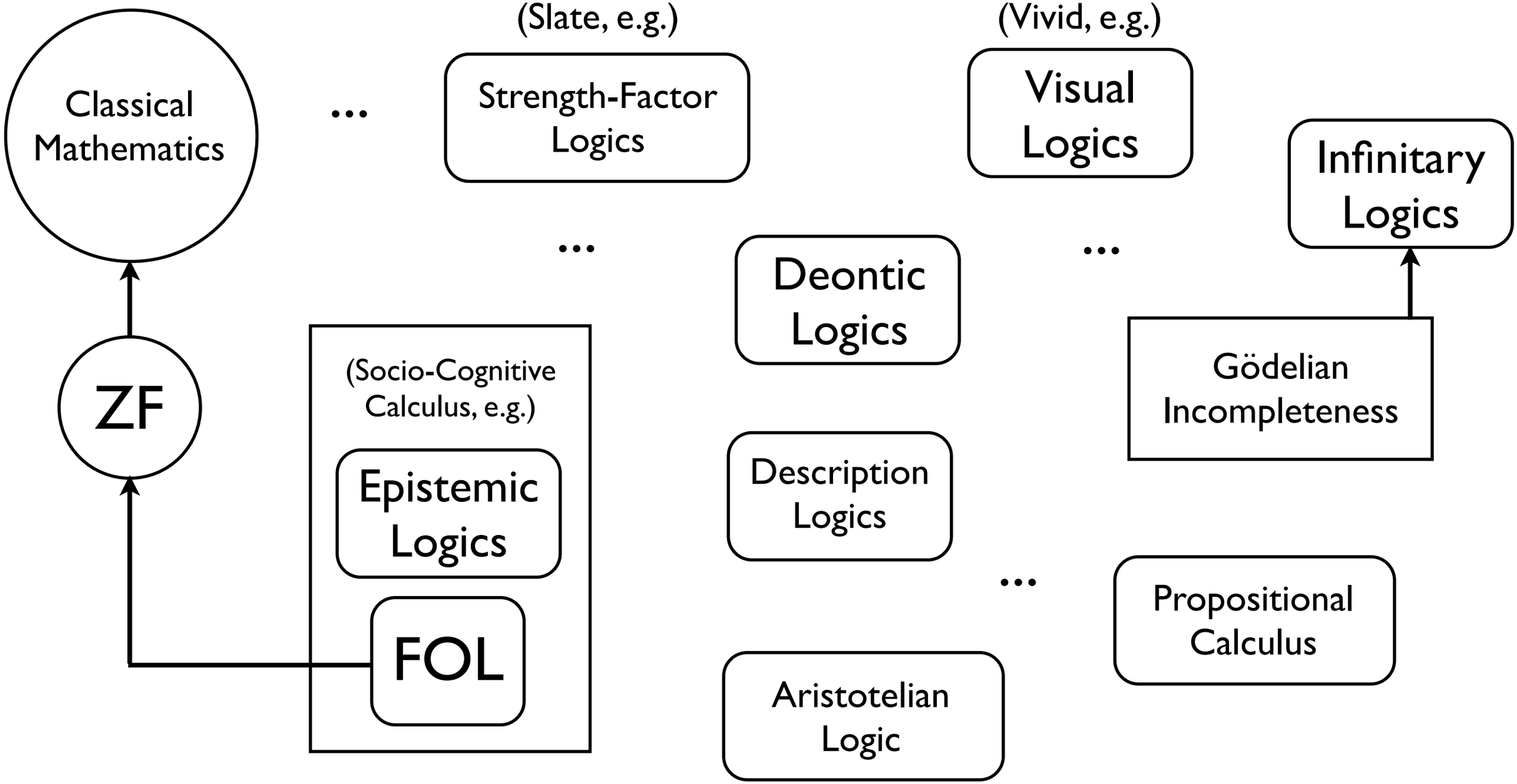
Avoid Limitations of *Elementary* Logic-Based R&D

Betting the farm on one or two logical systems (e.g., FOL, propositional calculus)—or for that matter on a particular theory within a logical system (e.g. Game Theory, probability calculus).

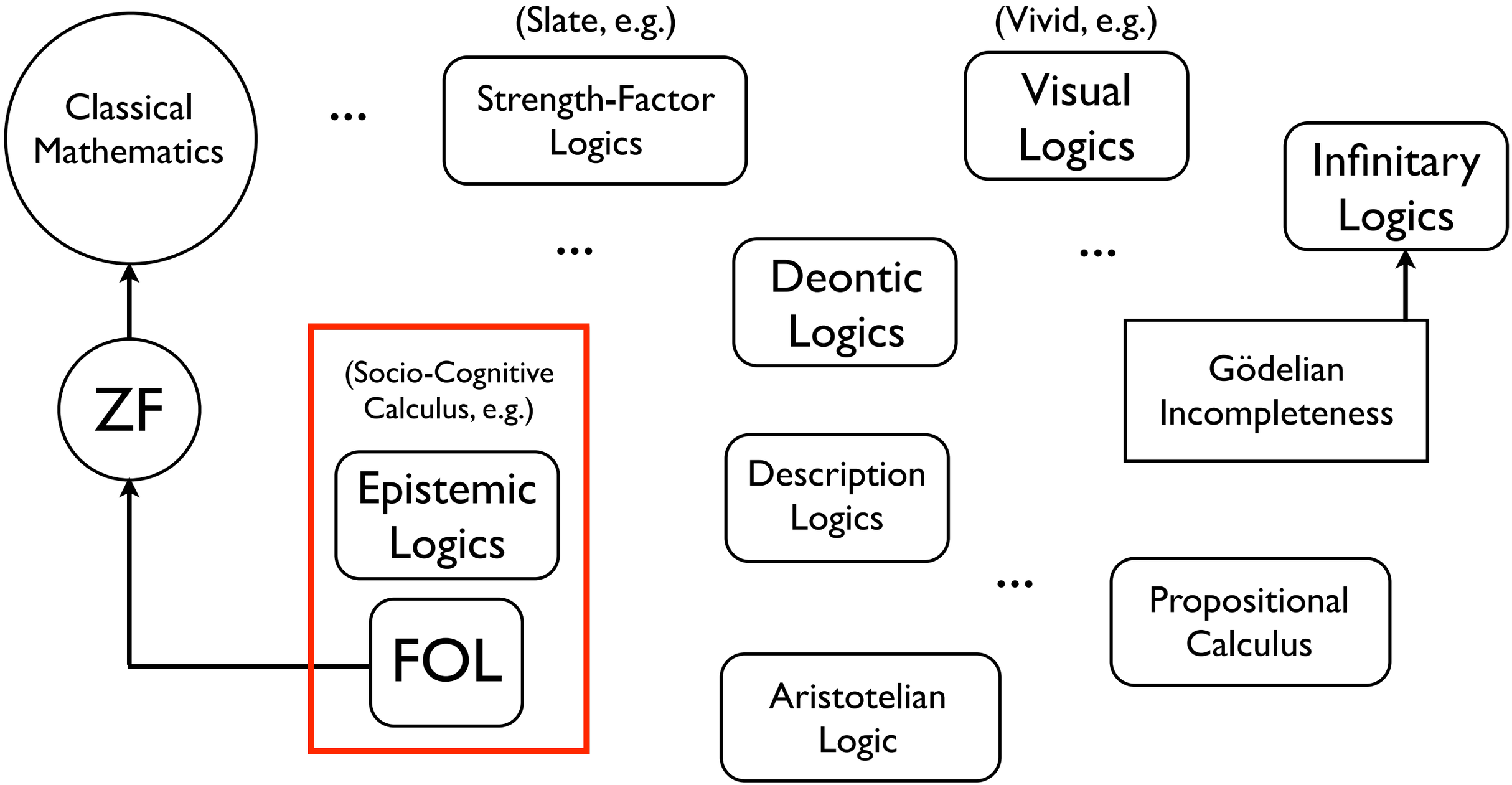
versus

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



The Space of Logical Systems



Classical Mathematics

...

(Slate, e.g.)

Strength-Factor Logics

(Vivid, e.g.)

Visual Logics

Infinitary Logics

...

...

Deontic Logics

ZF

(Socio-Cognitive Calculus, e.g.)

Epistemic Logics

FOL

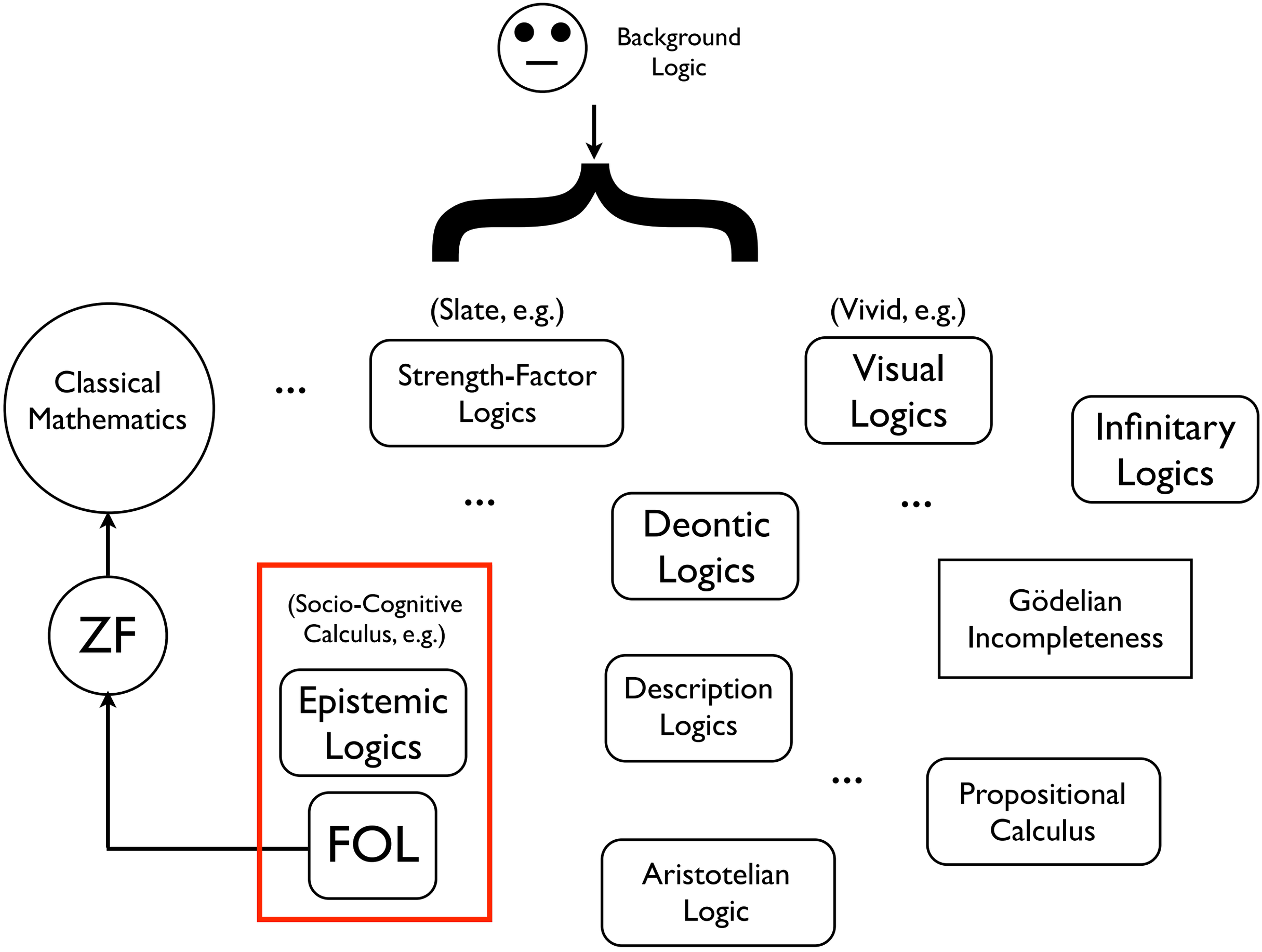
Gödelian Incompleteness

Description Logics

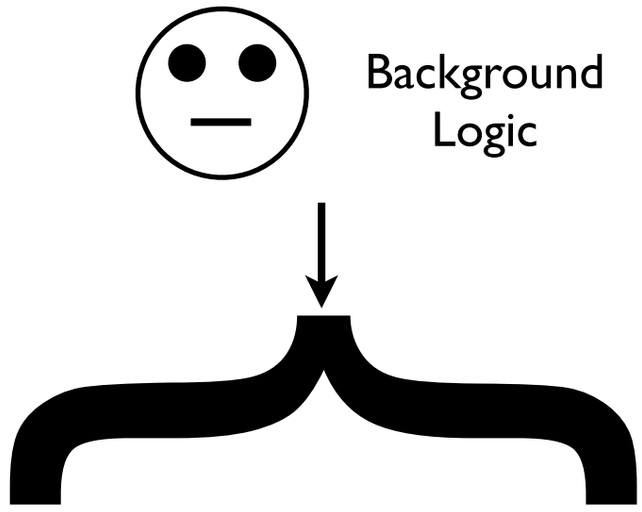
...

Propositional Calculus

Aristotelian Logic


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

Inspired by Piaget's five-stage view.



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

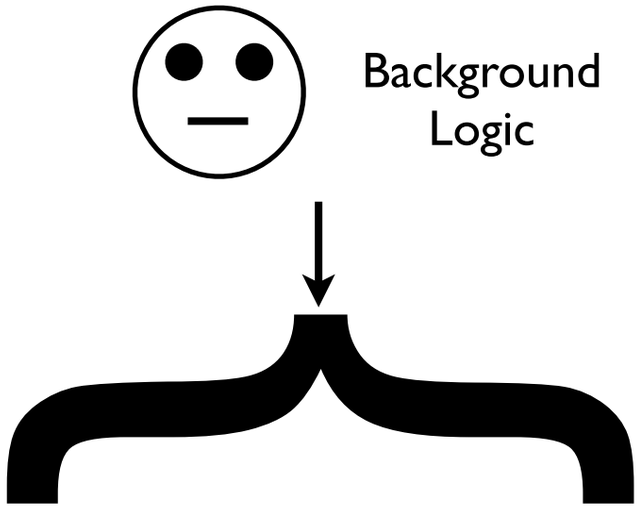
...

Propositional Calculus

FOL

Aristotelian Logic

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

Relevant *General* Warning:
Your Formalism Dictates What is Possible to Model and Simulate

Relevant *General* Warning: Your Formalism Dictates What is Possible to Model and Simulate

- Well-learned within formal logic, where e.g. we have long known, courtesy of many theorems, that for a fixed target *T* to be modeled, some logics will allow representation of key features in *T*, and some, no matter what, won't.

Relevant *General* Warning: Your Formalism Dictates What is Possible to Model and Simulate

- Well-learned within formal logic, where e.g. we have long known, courtesy of many theorems, that for a fixed target *T* to be modeled, some logics will allow representation of key features in *T*, and some, no matter what, won't.

- And this is true across the board—so if *T* is the deception of adversaries of the US, and therefore they (at the very least) believe that by performing certain actions the US will come to believe that her adversaries have certain beliefs, a formalism without provision for the representation and mechanization of iterated beliefs is inadequate.

Relevant *General* Warning: Your Formalism Dictates What is Possible to Model and Simulate

- Well-learned within formal logic, where e.g. we have long known, courtesy of many theorems, that for a fixed target *T* to be modeled, some logics will allow representation of key features in *T*, and some, no matter what, won't.

- And this is true across the board—so if *T* is the deception of adversaries of the US, and therefore they (at the very least) believe that by performing certain actions the US will come to believe that her adversaries have certain beliefs, a formalism without provision for the representation and mechanization of iterated beliefs is inadequate.

  - Modeling and simulation applied to asymmetrical/irregular conflict/ warfare, without provision in the formalism for iterated beliefs (including of a religious nature), argumentation, goals, fears, extensive knowledge, etc. is a very dangerous thing if deployed to the exclusion of more expressive techniques.

On Paradoxes ...

Typically ...

a contradiction is deduced from a fixed set of premises.

Typically ...

a contradiction is deduced from a fixed set of premises.

E.g., the Barber (= Russell's) Paradox ...

$$\vdash_{\mathrm{FOL}} \neg\exists x\forall y(Exy \Leftrightarrow \neg Eyy)$$

Typically ...

a contradiction is deduced from a fixed set of premises.

E.g., the Barber (= Russell's) Paradox ...

$$\vdash_{\mathrm{FOL}} \neg\exists x\forall y(Exy \Leftrightarrow \neg Eyy)$$

(At least in reasoning-and-decision-making, handling paradoxes sometimes taken as requirement. E.g., Pollock analysis and surmounting of Paradox of the Preface w/ Oscar.)

But *sometimes* ...

a contradiction is deduced from
n distinct arguments, each with a
different fixed set of premises.

E.g., Newcomb's Paradox, *and* ...
the Chain Store Paradox.

So:

$\Phi \vdash \phi$ $\Phi' \vdash \neg\phi$

where

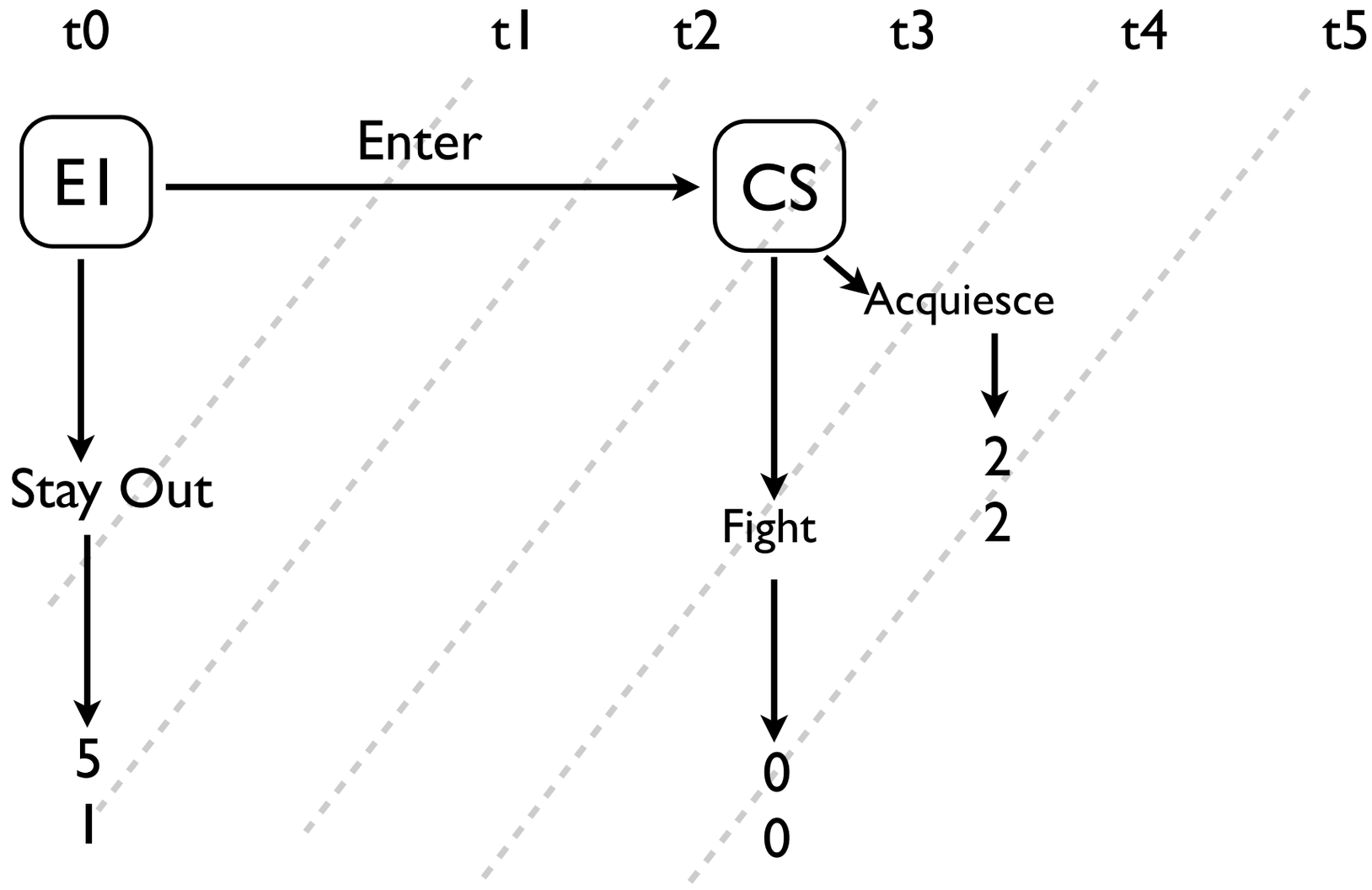
$\Phi \cup \Phi'$

true or at least very plausible.

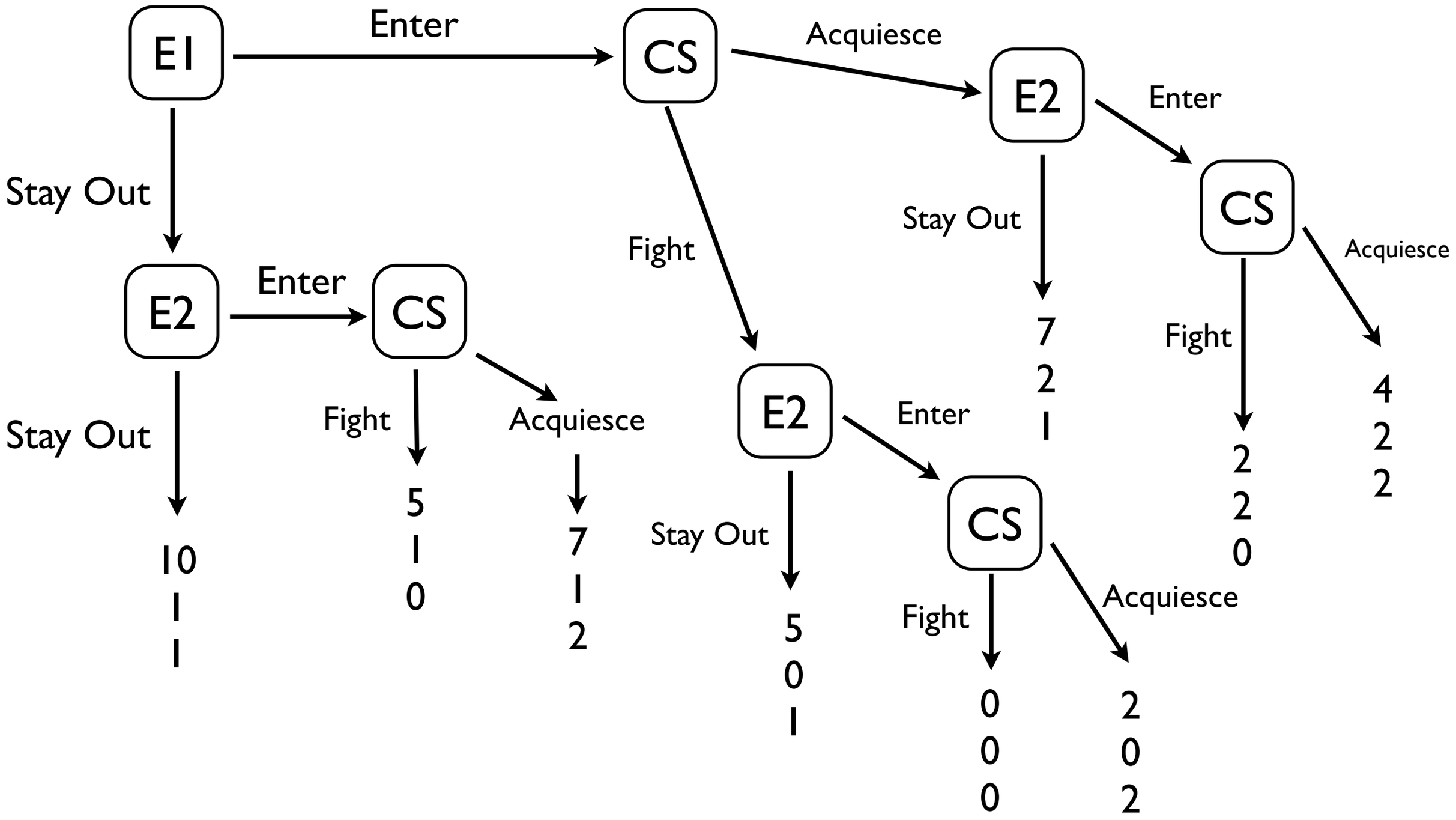
The Chain Store Paradox ...

(Selten 1978)

One Stage; Two Players



Two Stage; Three Players



Generating the "Paradox"

$\Phi \vdash \phi$ $\Phi' \vdash \neg\phi$

where

$\Phi \cup \Phi'$

true or at least very plausible.

Generating the "Paradox"

"Theorem": If rational, all Ei must enter, and CS acquiesce every time they do.
Selten's "**Proof**": Set n = 20. If E20 chooses 'Enter,' and CS 'fight,' then CS gets 0. If, on the other hand, CS chooses 'Aquiesce,' CS gets 2. Ergo by game-theoretic rationality CS must choose 'Acquiesce.' Game theorists typically assume that player rationality is Common Knowledge, so E20 knows that CS is rational and will acquiesce. Hence E20 enters because he receives 2 (rather than 1). "QED"

$\Phi \vdash \phi$ $\Phi' \vdash \neg\phi$

where

$\Phi \cup \Phi'$

true or at least very plausible.

Generating the "Paradox"

"Theorem": If rational, all Ei must enter, and CS acquiesce every time they do.
Selten's "**Proof**": Set n = 20. If E20 chooses 'Enter,' and CS 'fight,' then CS gets 0. If, on the other hand, CS chooses 'Aquiesce,' CS gets 2. Ergo by game-theoretic rationality CS must choose 'Acquiesce.' Game theorists typically assume that player rationality is Common Knowledge, so E20 knows that CS is rational and will acquiesce. Hence E20 enters because he receives 2 (rather than 1). "QED"

$\Phi \vdash \phi$

"Theorem": A rational CS will fight time after time, which will cause a string of entrants to stay out—after which CS can acquiesce.
Selten's "**Proof**": A story: If the first several entrants are fought, others would change their beliefs, and change from 'Enter' to 'Stay Out.' (If only 7 of the first 17 entrants stay out, CS is very well off: 35.) "QED"

$\Phi' \vdash \neg\phi$

where

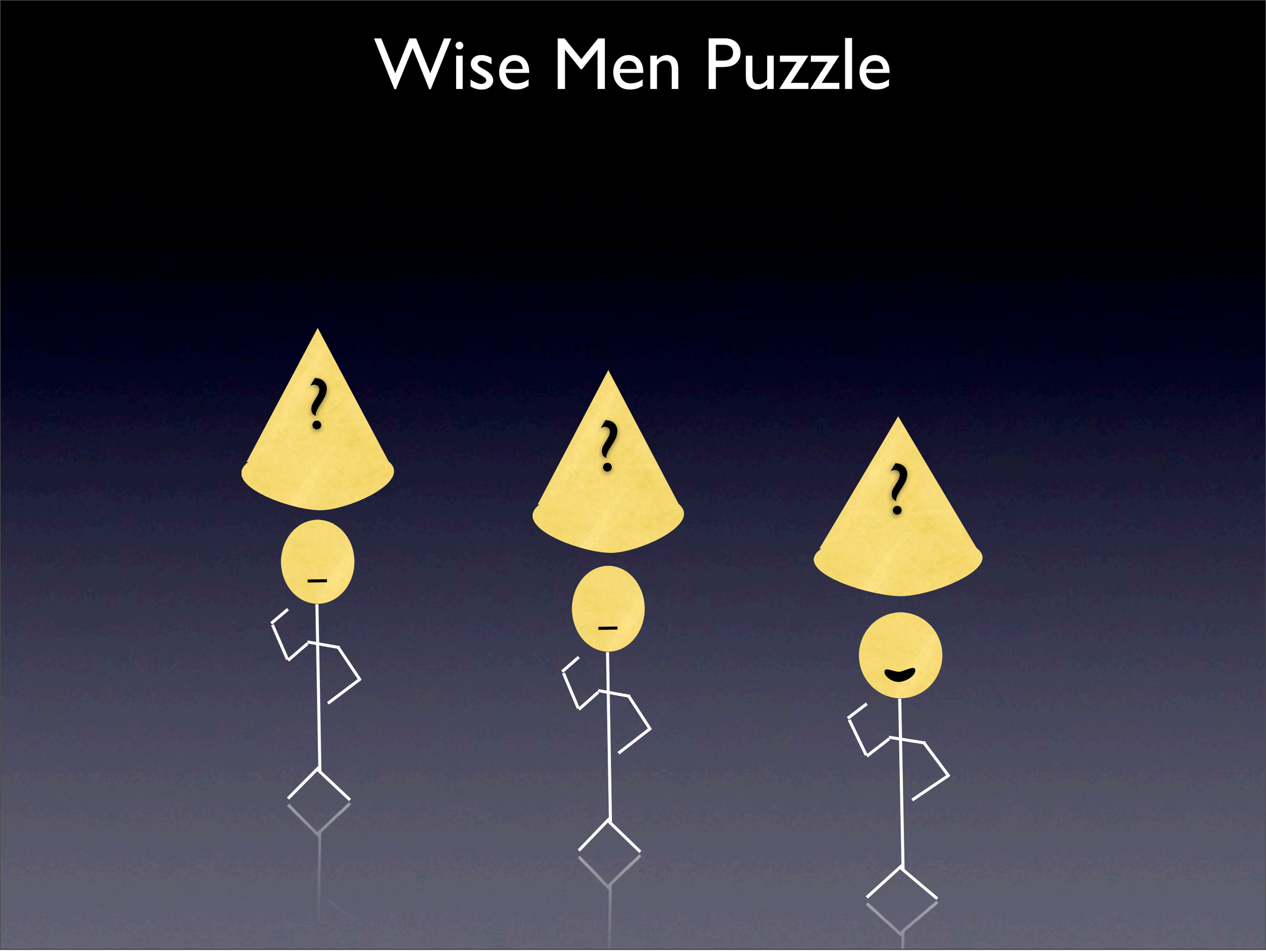
$\Phi \cup \Phi'$

true or at least very plausible.

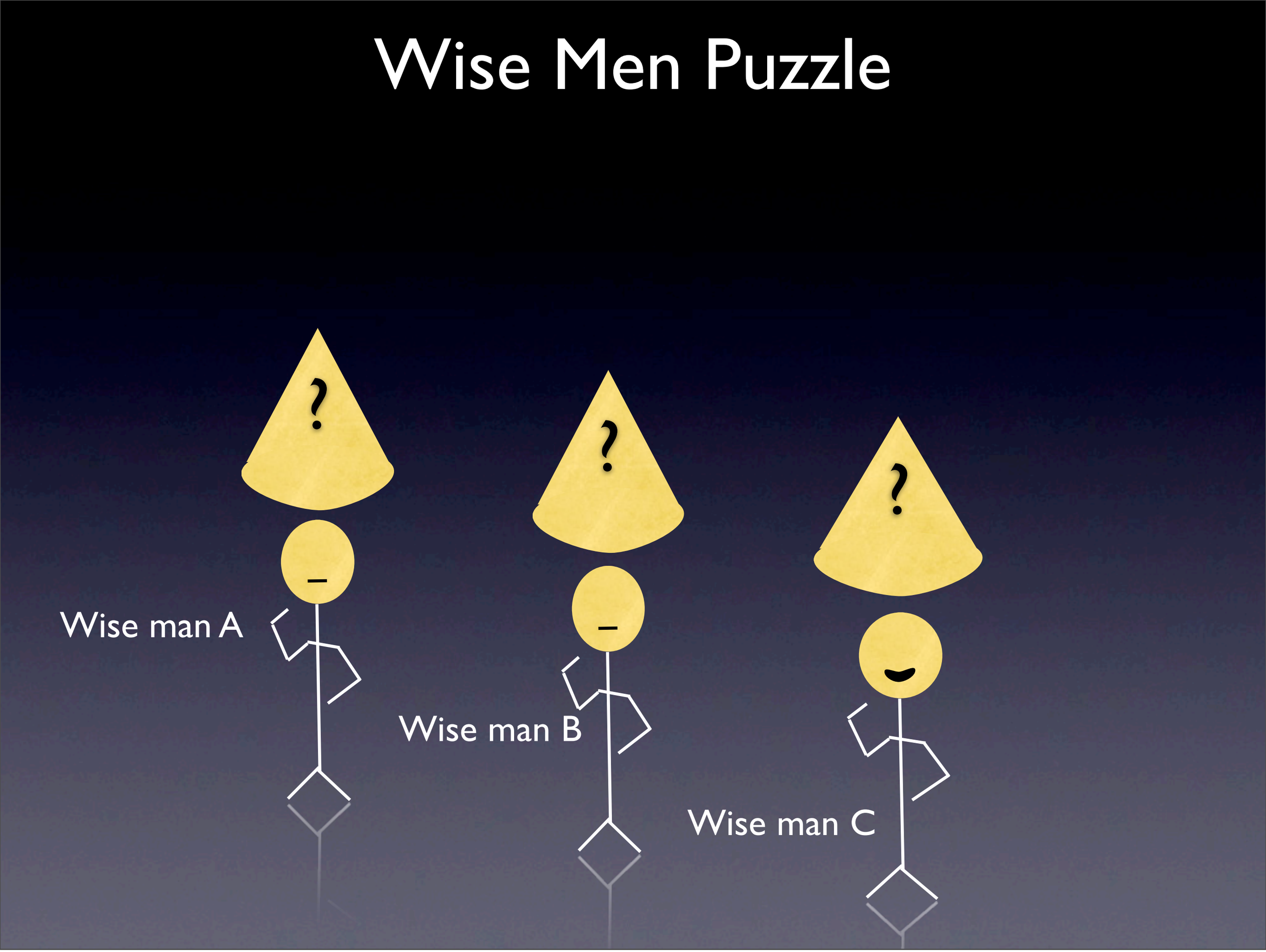
The Socio-Cognitive Calculus...

A Precursor: WMP_n ...

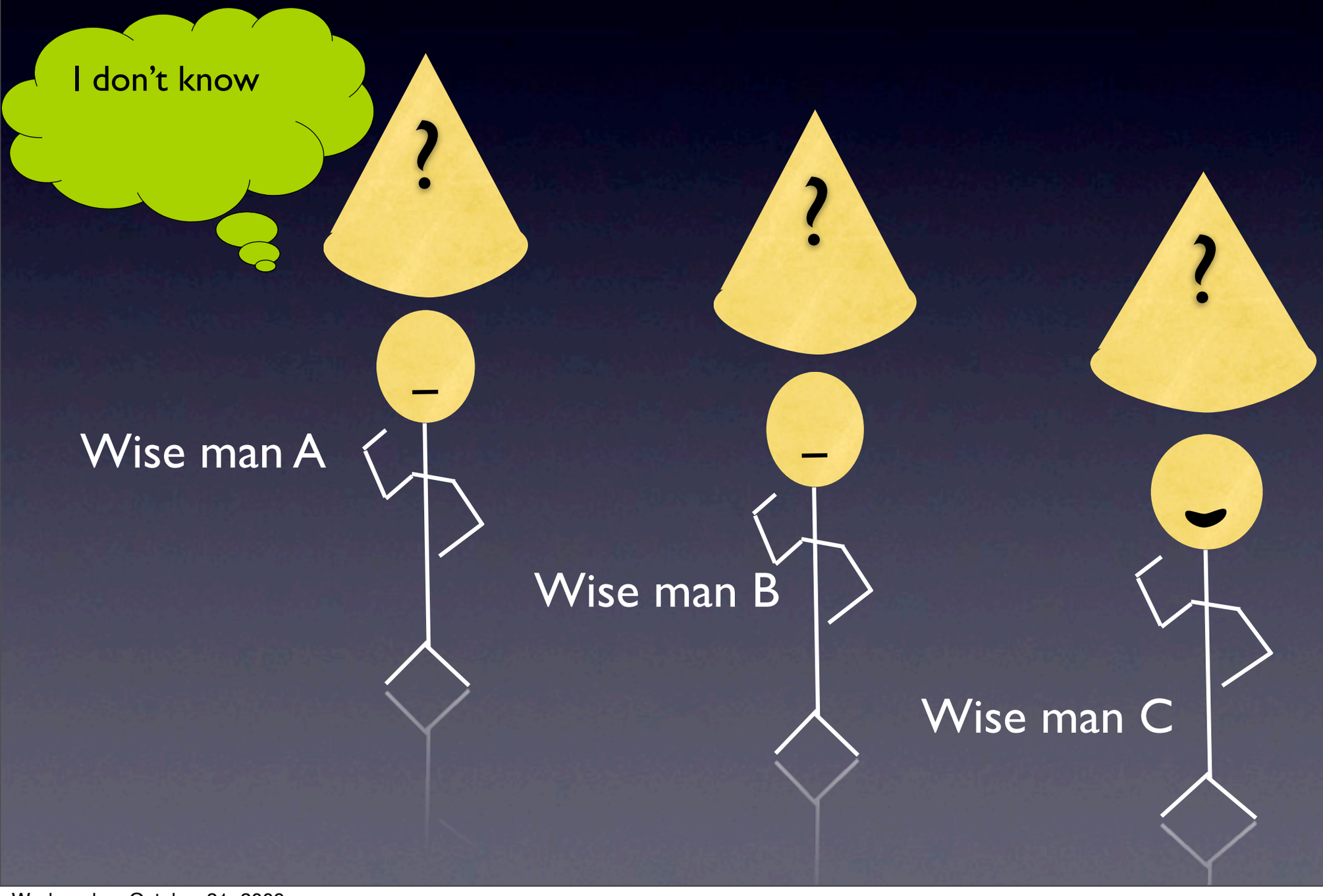
Wise Men Puzzle



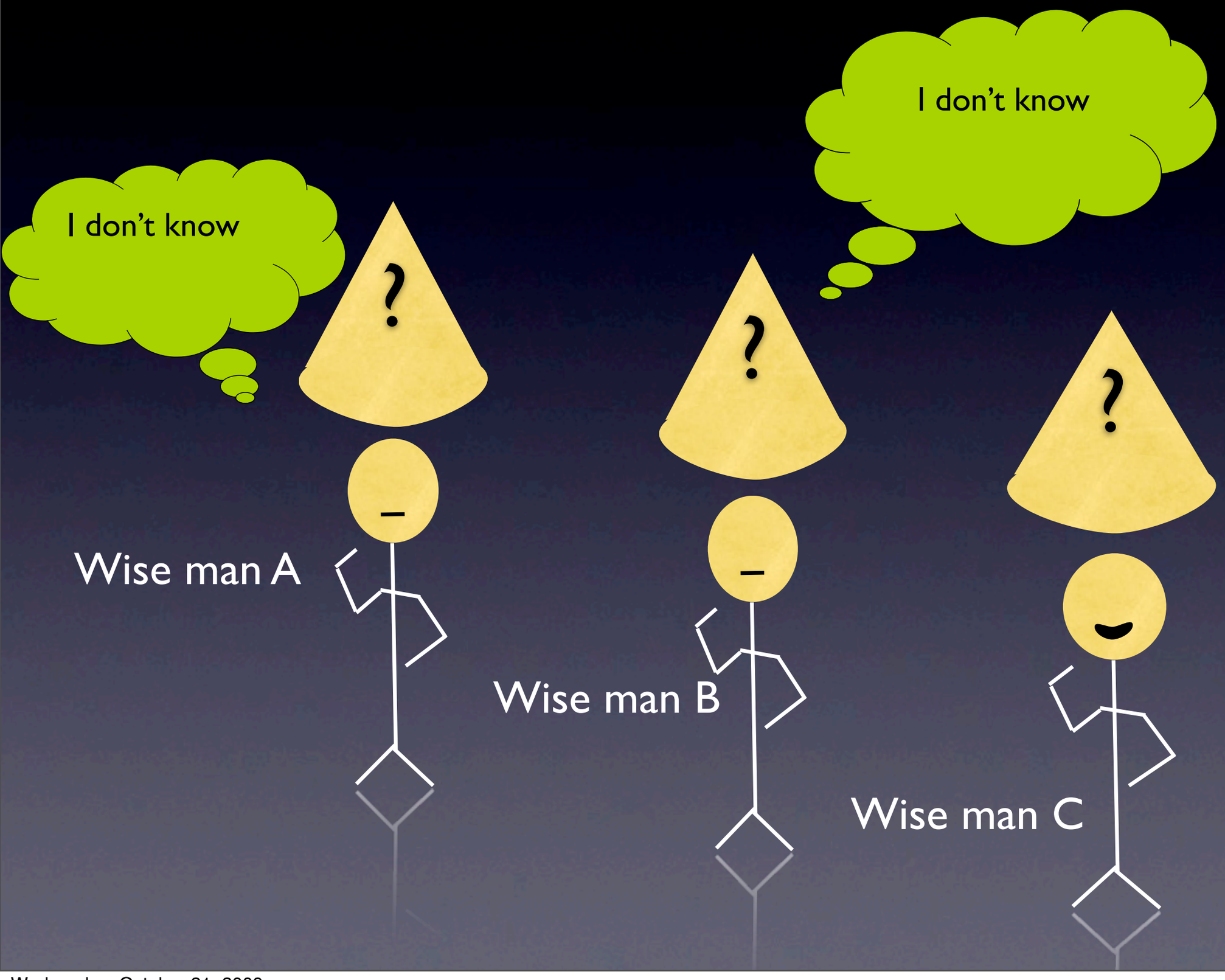
Wise Men Puzzle



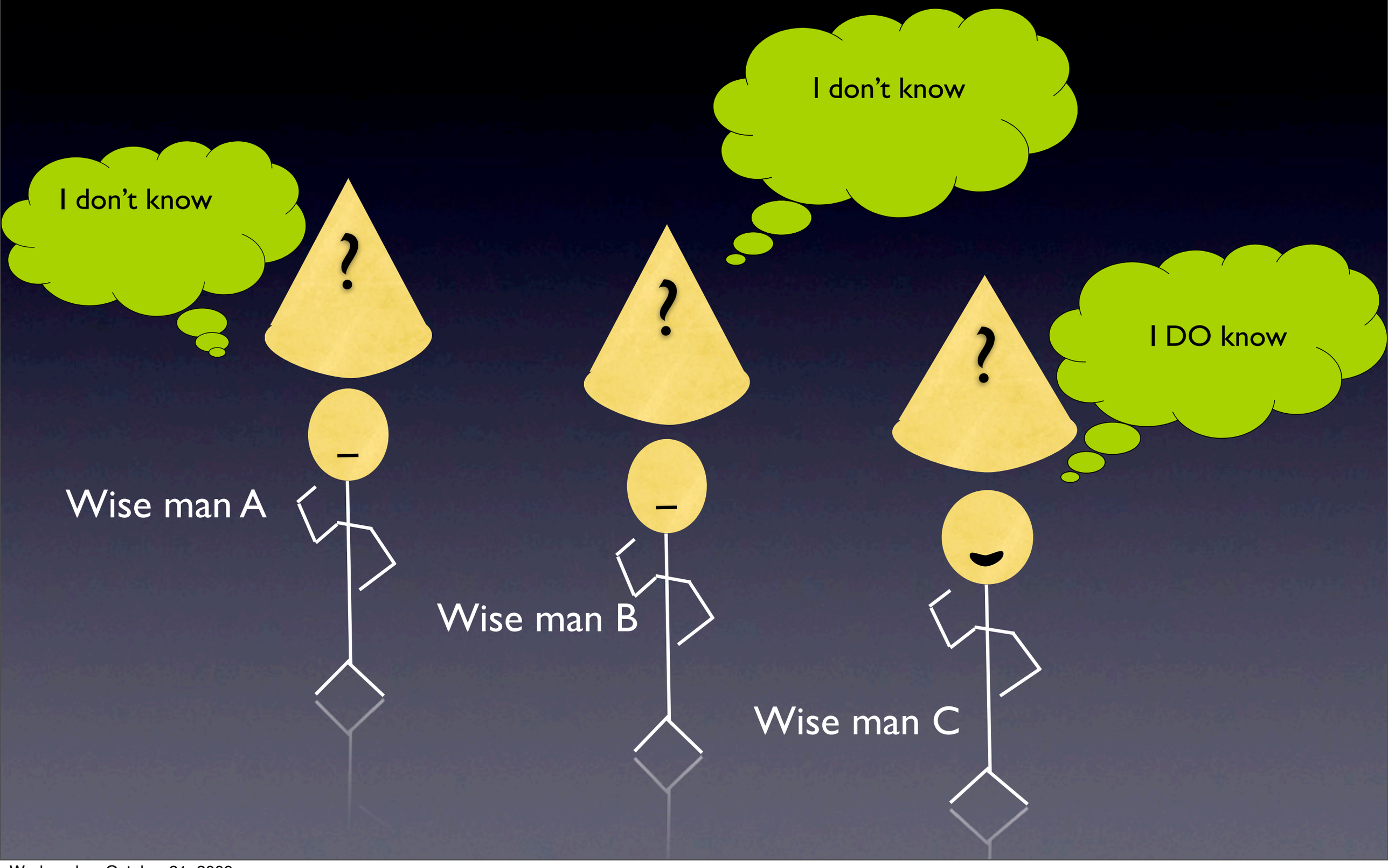
Wise Men Puzzle



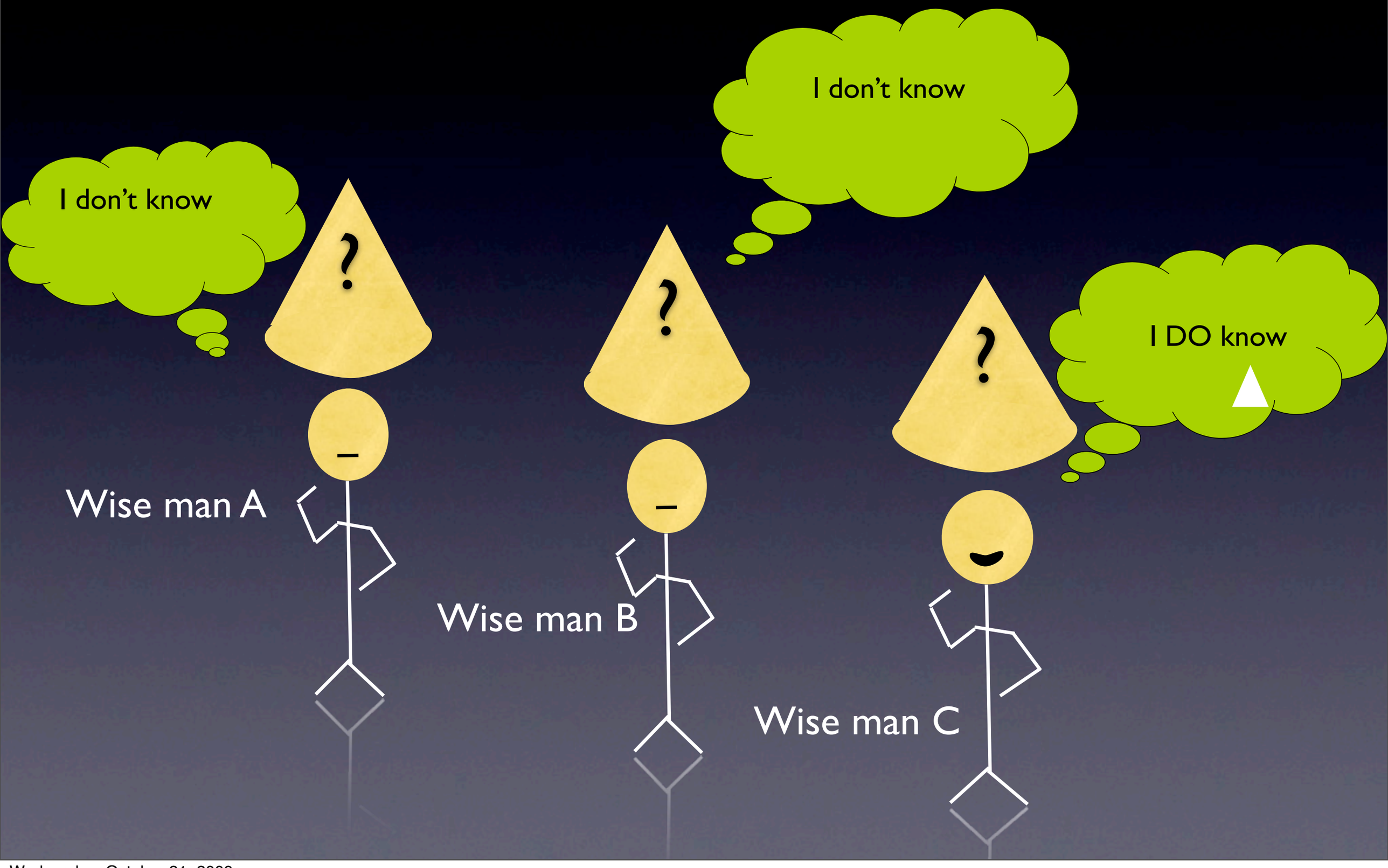
Wise Men Puzzle



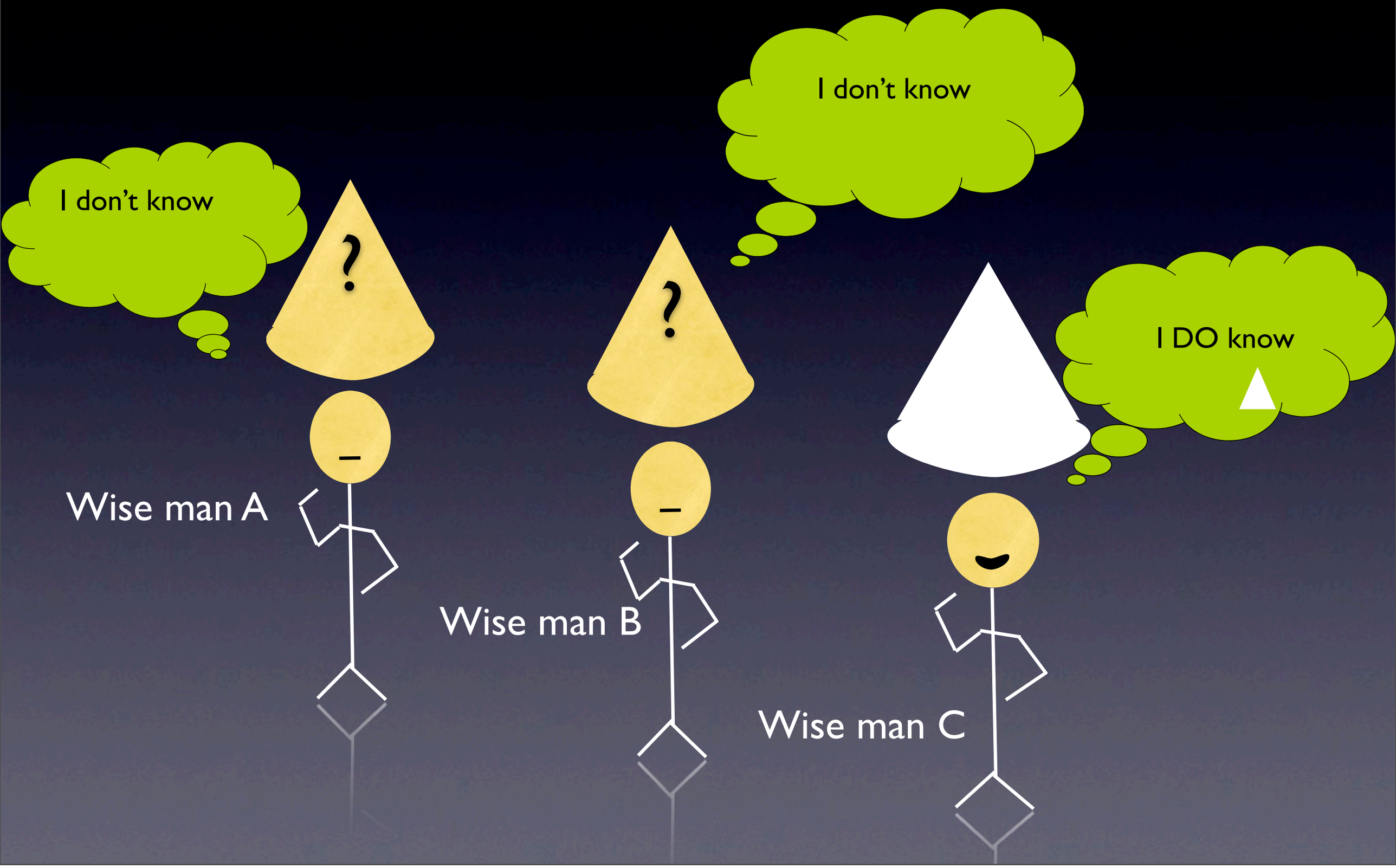
Wise Men Puzzle



Wise Men Puzzle



Wise Men Puzzle



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

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

W/ formal proofs that can be 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:

$\{R_1 \wedge R_2 \wedge R_3\} \vdash R_1$	$[Reflex]$, $\wedge\text{-}E_1$
$\{R_1 \wedge R_2 \wedge R_3\} \vdash R_2$	$[Reflex]$, $\wedge\text{-}E_1$, $\wedge\text{-}E_2$
$\{R_1 \wedge R_2 \wedge R_3\} \vdash R_3$	$[Reflex]$, $\wedge\text{-}E_2$
$\{R_1 \wedge R_2 \wedge R_3\} \vdash K_\alpha(\neg Q) \Rightarrow K_\alpha(P)$	2, $[K]$, $\Rightarrow\text{-}E$
$\{R_1 \wedge R_2 \wedge R_3\} \vdash \neg Q \Rightarrow K_\alpha(P)$	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, $\Rightarrow\text{-}E$
$\{R_1 \wedge R_2 \wedge R_3\} \vdash Q$	7, $[\neg\text{-}E]$

❑

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

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

❑

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 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,

Proved-Sound Algorithm for Generating Proof-Theoretic Solution to WMP$_n$

general case

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

W/ formal proofs that can be 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:

$\{R_1 \wedge R_2 \wedge R_3\} \vdash R_1$	$[Reflex]$, $\wedge\text{-}E_1$
$\{R_1 \wedge R_2 \wedge R_3\} \vdash R_2$	$[Reflex]$, $\wedge\text{-}E_1$, $\wedge\text{-}E_2$
$\{R_1 \wedge R_2 \wedge R_3\} \vdash R_3$	$[Reflex]$, $\wedge\text{-}E_2$
$\{R_1 \wedge R_2 \wedge R_3\} \vdash K_\alpha(\neg Q) \Rightarrow K_\alpha(P)$	2, $[K]$, $\Rightarrow\text{-}E$
$\{R_1 \wedge R_2 \wedge R_3\} \vdash \neg Q \Rightarrow K_\alpha(P)$	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, $\Rightarrow\text{-}E$
$\{R_1 \wedge R_2 \wedge R_3\} \vdash Q$	7, $[\neg\text{-}E]$

❑

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

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

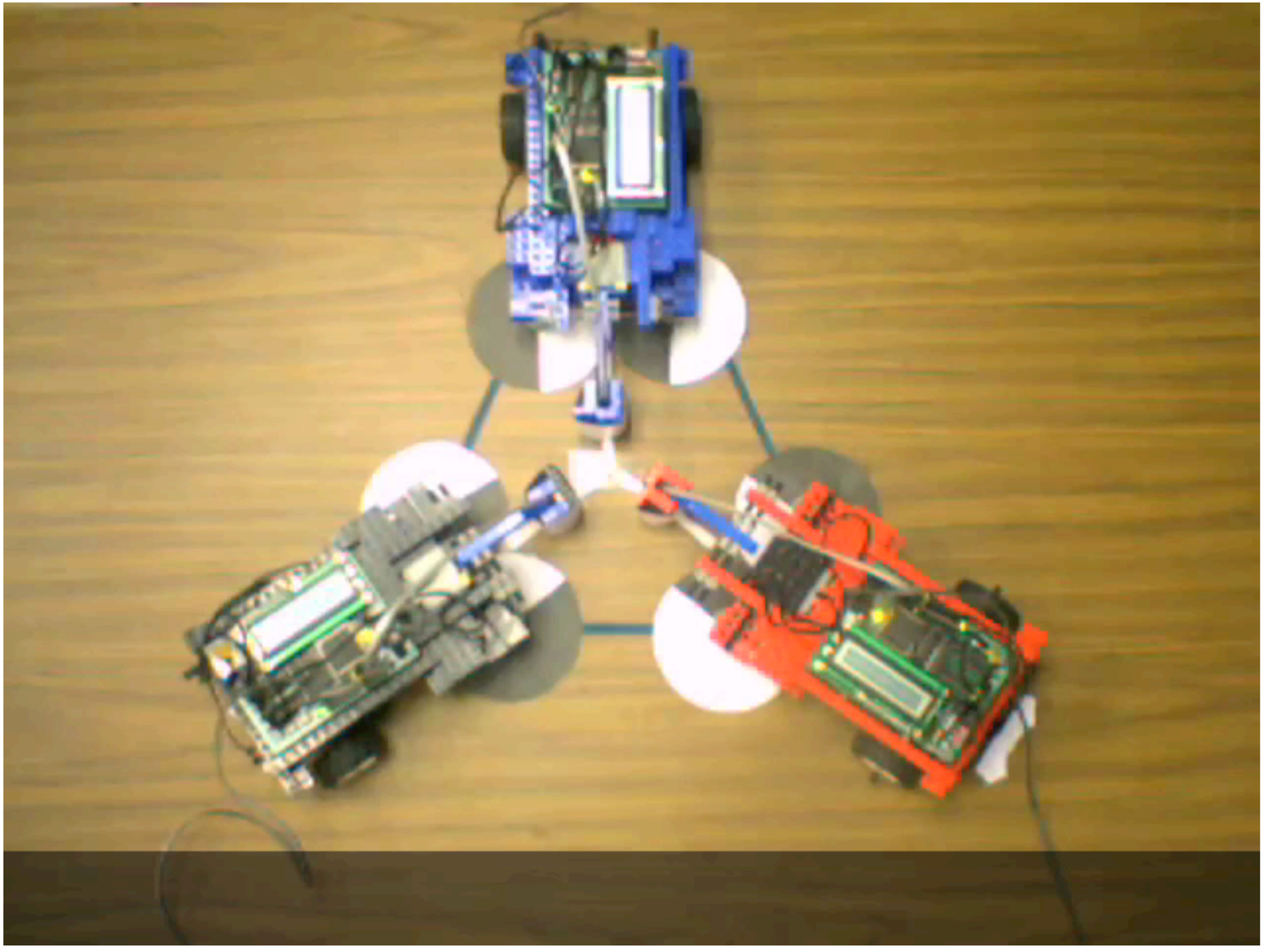
❑

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 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,

“Life and Death” Wise Man Test (3)



Using Socio-Cognitive Calculus to Engineer Cognitively Robust Synthetic Characters and Model/ Simulate False-Belief Tests ...

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

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

"The present account of the false belief transition is incomplete in important ways. After all, our agent had only to choose the best of two known models. This begs an understanding of the dynamics of rational revision near threshold and when the space of possible models is far larger. Further, a single formal model ought ultimately to be applicable to many false belief tasks, and to reasoning about mental states more generally. Several components seem necessary to extend a particular theory of mind into such a framework theory: a richer representation for the propositional content and attitudes in these tasks, extension of the implicit quantifier over trials to one over situations and people, and a broader view of the probability distributions relating mental state variables. Each of these is an important direction for future research."

"Intuitive Theories of Mind: A Rational Approach to False Belief"
Goodman et al.

Done.

"The present account of the false belief transition is incomplete in important ways. After all, our agent had only to choose the best of two known models. This begs an understanding of the dynamics of rational revision near threshold and when the space of possible models is far larger. Further, a single formal model ought ultimately to be applicable to many false belief tasks, and to reasoning about mental states more generally. Several components seem necessary to extend a particular theory of mind into such a framework theory: a richer representation for the propositional content and attitudes in these tasks, extension of the implicit quantifier over trials to one over situations and people, and a broader view of the probability distributions relating mental state variables. Each of these is an important direction for future research."

"Intuitive Theories of Mind: A Rational Approach to False Belief"
Goodman et al.

Done.

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:

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

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:

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

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:

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

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':

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

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:

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

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:

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

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

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

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

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

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

- *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:

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

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:

$$[D_2] \quad \mathbf{C}(\forall\, a, t, o, l_1, l_2 \;.\; \mathit{holds}(\mathit{located}(o, l_1), t) \Rightarrow \mathit{initiates}(\mathit{action}(a, \mathit{moves}(o, l_1, l_2)), \mathit{located}(o, l_2), t))$$

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

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

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

$$[D_4] \quad \mathbf{C}(\forall\, o, t, l_1, l_2 \;.\; \mathit{holds}(\mathit{located}(o, l_1), t) \wedge \mathit{holds}(\mathit{located}(o, l_2), t) \Rightarrow l_1 = l_2)$$

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:
     $$\mathbf{S}(a_1, \mathit{happens}(\mathit{action}(a_2, \alpha), t)) \quad (1)$$
  2. Therefore, a_1 knows that the corresponding event has occurred at t:
     $$\mathbf{K}(a_1, \mathit{happens}(\mathit{action}(a_2, \alpha), t)) \quad (2)$$
     This follows from the preceding premise and $[DR_4]$.
  3. From $[A_4]$ and $[DR_2]$ we obtain:
     $$\mathbf{K}(a_1, \forall\, a, \alpha, t \;.\; \mathit{happens}(\mathit{action}(a, \alpha), t) \Rightarrow \mathbf{K}(a, \mathit{happens}(\mathit{action}(a, \alpha), t))) \quad (3)$$
  4. From (3) and $[DR_9]$ we get:
     $$\mathbf{K}(a_1, \mathit{happens}(\mathit{action}(a_2, \alpha), t) \Rightarrow \mathbf{K}(a_2, \mathit{happens}(\mathit{action}(a_2, \alpha), t))) \quad (4)$$
  5. From (4), (2), and $[DR_6]$ we get:
     $$\mathbf{K}(a_1, \mathbf{K}(a_2, \mathit{happens}(\mathit{action}(a_2, \alpha), t))) \quad (5)$$
- *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

Using Socio-Cognitive Calculus to Model Deception ...

Results and Resolving the Paradox ...

Stage I

$\Phi \vdash \phi$ $\Phi' \vdash \neg\phi$

where

$\Phi \cup \Phi'$

true or at least very plausible.

Stage 1

Encoding of game-theoretic principles, *plus* epistemic facts beyond the reach of game theory represented—but no other real-world belief, knowledge, goals.

$$\Phi \vdash \phi \qquad\qquad \Phi' \vdash \neg\phi$$

where

$$\Phi \cup \Phi'$$

true or at least very plausible.

Stage 1

Encoding of game-theoretic principles, *plus* epistemic facts beyond the reach of game theory represented—but no other real-world belief, knowledge, goals.

$$\Phi \vdash \phi$$

certified!

$$\Phi' \vdash \neg\phi$$

where

$$\Phi \cup \Phi'$$

true or at least very plausible.

Theorem 2: Agent e_1 knows that if he stays out at t_1 he will receive a payoff of one at t_2.

Proof: We begin by noting a tautology that is assumed to be common knowledge (rule R'):

$$\frac{}{\mathbf{C}[(\phi \Rightarrow (\psi \wedge \delta)) \Rightarrow (\phi \Rightarrow \psi)]}$$

Recall from the Athena-based proof of Theorem 1 that the following conditional (which we can label '(1)') holds.

```
(if (if (StaysOut e1 t1) (and (Payoff e1 one t2) (Payoff cs five t2)))
    (if (StaysOut e1 t1) (Payoff e1 one t2)))
```

E.g.,

Instantiating, we can infer from R' that $\mathbf{C}$ ranges over this conditional. But DR_2 is

$$\frac{\mathbf{C}(\phi)}{\mathbf{K}(a, \phi)}$$

so we can infer that e_1 knows (1). In light of DR_6,

$$\frac{\mathbf{K}(a, \phi \Rightarrow \psi), \mathbf{K}(a, \phi)}{\mathbf{K}(a, \psi),}$$

we can deduce from the fact that e_1 knows the antecedent of (1) holds, that e_1 knows

```
(if (StaysOut e1 t1) (Payoff e1 one t2)))
```

QED

Stage II

Encoding of game-theoretic principles, *plus* epistemic facts beyond the reach of game theory represented—but no other real-world belief, knowledge, goals. Inductive proof in the *forward* direction.

$\Phi \vdash \phi$

certified!

$\Phi' \vdash \neg\phi$

where

$\Phi \cup \Phi'$

true or at least very plausible.

Stage 11

Encoding of game-theoretic principles, *plus* epistemic facts beyond the reach of game theory represented—but no other real-world belief, knowledge, goals. Inductive proof in the *forward* direction.

If we include, formally, entrants who see at each step what goes on, and adjust their beliefs accordingly, and a CS that knows that outside entrants are observing and believing accordingly, deterrence makes sense at any given stage in the game.

$$\Phi \vdash \phi$$

certified!

$$\Phi' \vdash \neg\phi$$

where

$$\Phi \cup \Phi'$$

true or at least very plausible.

Stage 11

Encoding of game-theoretic principles, *plus* epistemic facts beyond the reach of game theory represented—but no other real-world belief, knowledge, goals. Inductive proof in the *forward* direction.

If we include, formally, entrants who see at each step what goes on, and adjust their beliefs accordingly, and a CS that knows that outside entrants are observing and believing accordingly, deterrence makes sense at any given stage in the game.

$$\Phi \vdash \phi$$

certified!

$$\Phi' \vdash \neg\phi$$

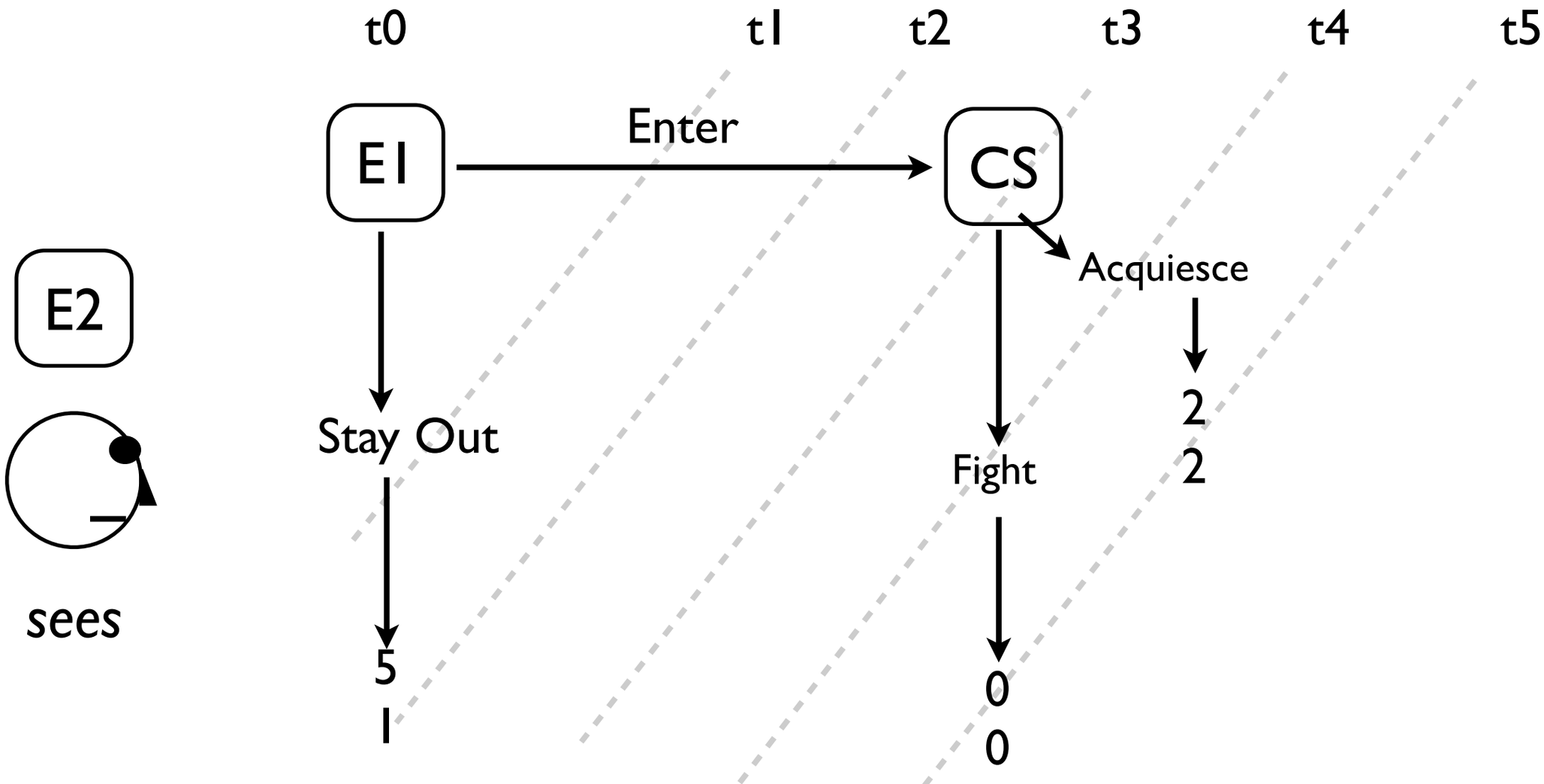
certifiable!

where

$$\Phi \cup \Phi'$$

true or at least very plausible.

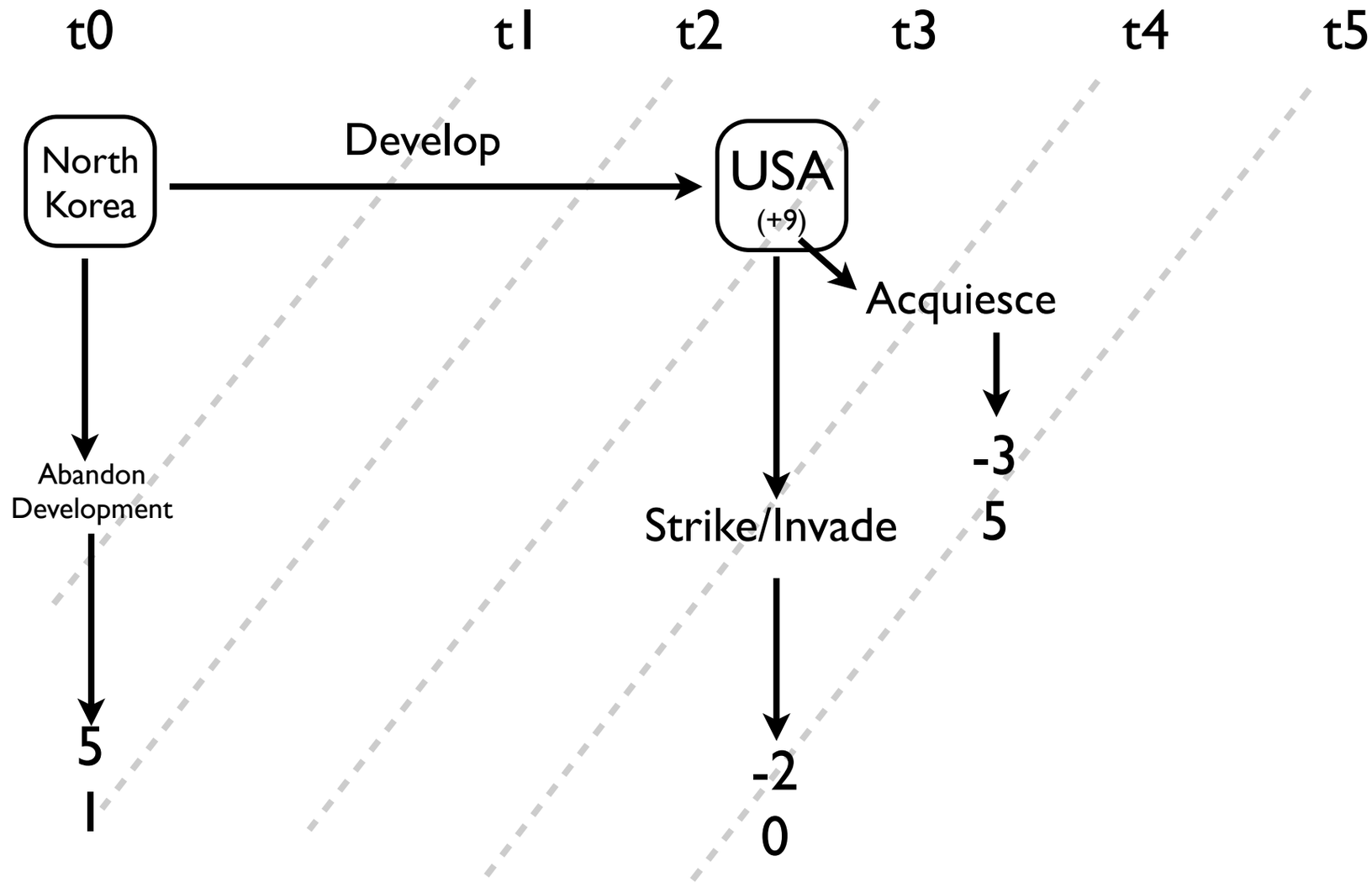
One Stage; "Three" Players



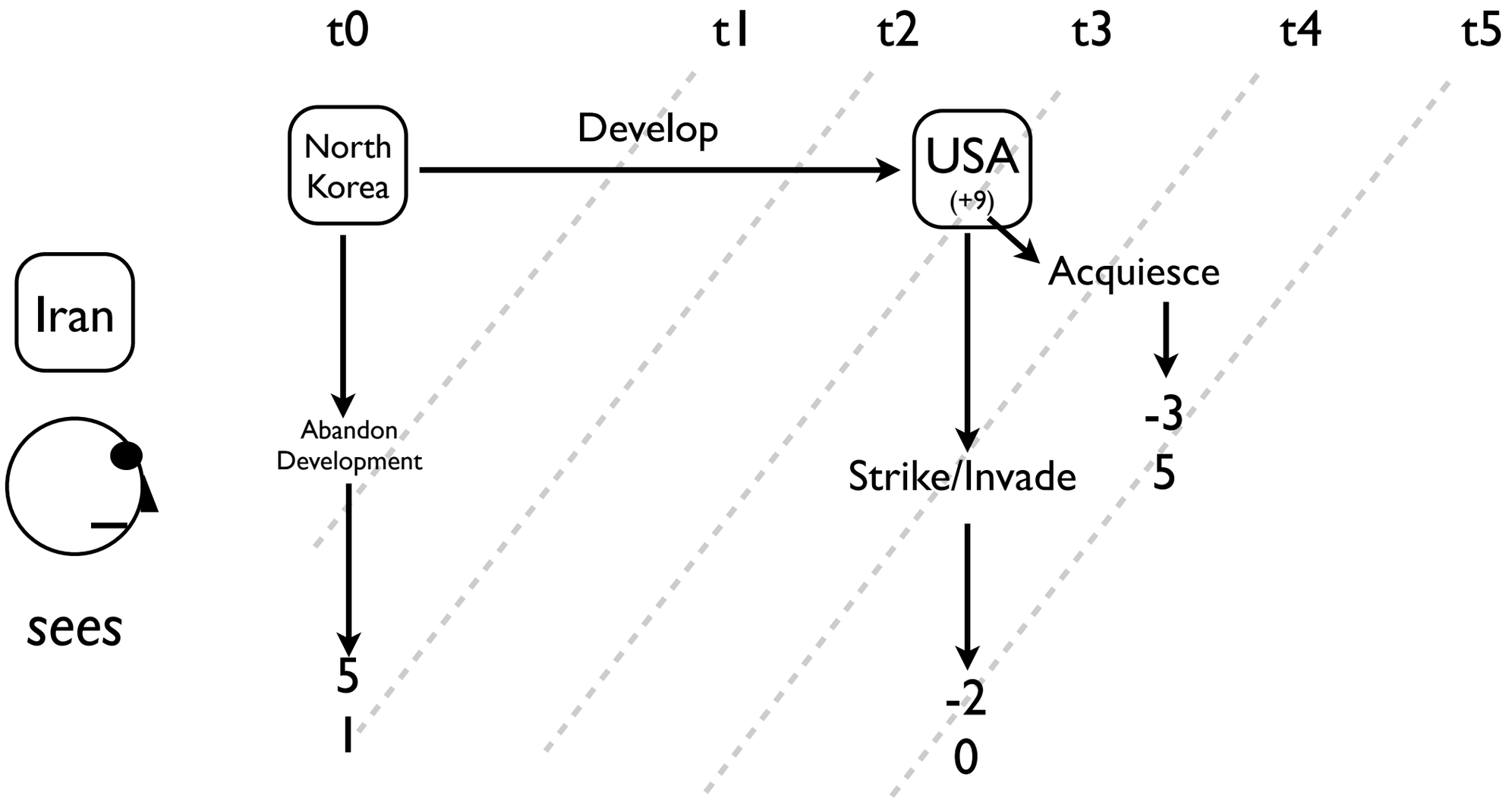
Game theory can be improved, and a proof produced—but the proof's premises fail to include those that are in fact operative in the world of real, "cognitively robust" agents, and such agents are the ones that populate markets (in the economic and defense-relevant sense).

(Note: While Game Theory in connection with epistemic operators has been discussed, GT, formally speaking, includes no formal language powerful enough to include such operators, in conjunction with full computational machinery for time and change (e.g., the event calculus, which is includes in the socio-cognitive calculus).)

Stage III:
The Nuclear "Club"



The Reality: Iran et al. Watching



Future (desired)

Future (desired)

- Refine the (publicly available) v1 of implemented socio-cognitive calculus.

Future (desired)

- Refine the (publicly available) v1 of implemented socio-cognitive calculus.
- Expand the formal family of unprecedentedly expressive socio-cognitive logics for particular defense needs (theory and corresponding implementation).

Future (desired)

- Refine the (publicly available) v1 of implemented socio-cognitive calculus.
- Expand the formal family of unprecedentedly expressive socio-cognitive logics for particular defense needs (theory and corresponding implementation).
- Using this family, model and simulate additional, larger scenarios, including asymmetrical/irregular conflict/ warfare in which agents as formalized are cognitively robust.

Future (desired)

- Refine the (publicly available) v1 of implemented socio-cognitive calculus.
- Expand the formal family of unprecedentedly expressive socio-cognitive logics for particular defense needs (theory and corresponding implementation).
- Using this family, model and simulate additional, larger scenarios, including asymmetrical/irregular conflict/ warfare in which agents as formalized are cognitively robust.
- Refine *methods*; invent parallel algorithms; use supercomputing.

Parallelization/Supercomputing;
Computational Logic, and the Arithmetic Hierarchy

$$\{f | f : N \rightarrow N\}$$

(Information Processing)

Π_2

Σ_1

Turing Limit

automatic programming

$$\forall u \forall v [\exists k H(n, k, u, v) \leftrightarrow \exists k' H(m, k', u, v)]$$

$$\Phi \vdash \phi?$$ first-order provability

BHAPs

$$\exists k H(n, k, u, v)$$
$$H(n, k, u, v)$$

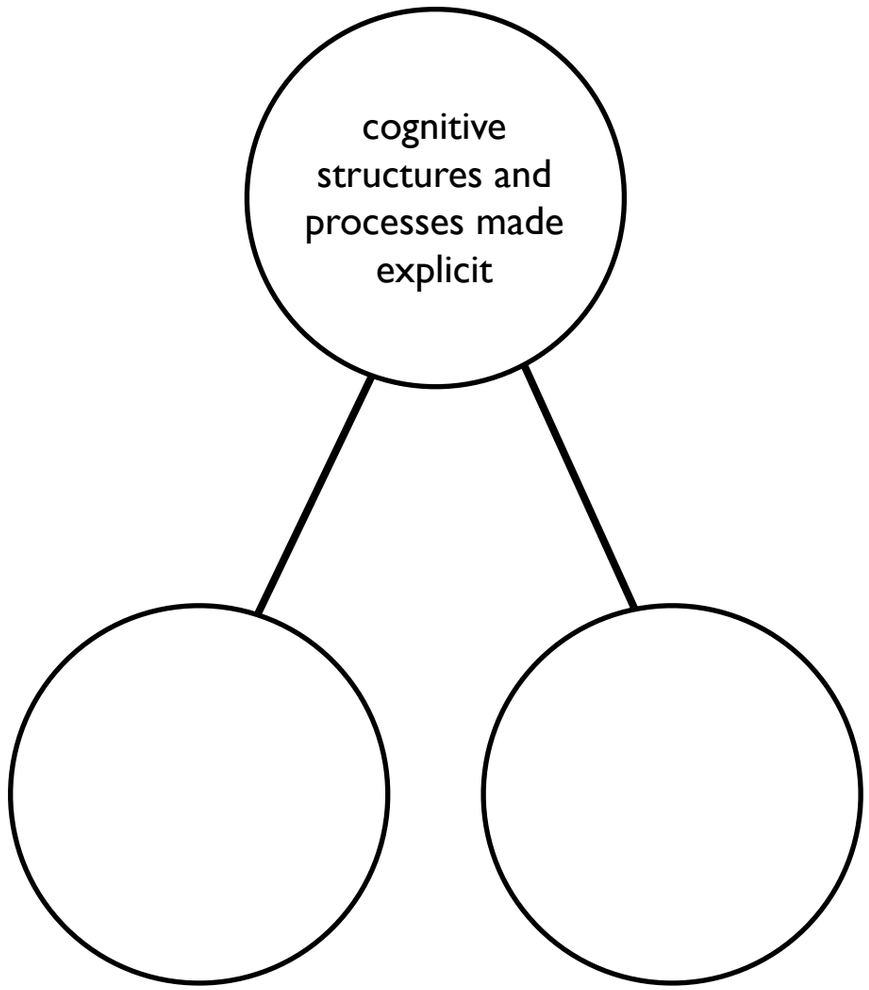
includes all functions studied in complexity theory

Heretofore, Modeling Not Based on "Cognitively Robust" Agents

A is a *cognitively robust agent* just in case *A* is an agent some of whose non-trivial actions are a function of what *A* knows, believes, intends, ... regarding not only the inanimate portion of its environment, but also regarding other agents, and in particular regarding what other agents believe, know, intend, ...

A *cognitive network* would presumably be a dynamic network in which the nodes correspond to cognitively robust agents, simulated or real. Whereas in a social network nodes can represent "individuals" in the *complete absence* of the structures and processes at the heart of cognition (reasoning, learning, deciding, planning, knowing, believing, hoping, fearing, intending, perceiving), in a cognitive network such things are be made formal, and computational, via an advanced, implemented logic.

Cognitive Network



Social Network