A Representation Theorem for Reasoning in First-Order Multi-Agent Knowledge Bases

Christoph Schwering    Maurice Pagnucco

UNSW Sydney, Australia
A Representation Theorem for Reasoning in First-Order Multi-Agent Knowledge Bases

Reasoning in multi-agent epistemic knowledge bases reduces to classical validity
A Representation Theorem for Reasoning in First-Order Multi-Agent Knowledge Bases

Reasoning in multi-agent epistemic knowledge bases reduces to classical validity

- Logical framework: Levesque’s logic of only-knowing $K_A \alpha \quad O_A \alpha$
A Representation Theorem for Reasoning in First-Order Multi-Agent Knowledge Bases

Reasoning in multi-agent epistemic knowledge bases

Turing-reduces to

classical validity

Logical framework: Levesque’s logic of only-knowing $K_A \alpha$ $O_A \alpha$
A Representation Theorem for Reasoning in First-Order Multi-Agent Knowledge Bases

Reasoning in multi-agent epistemic knowledge bases
Turing-reduces to classical validity

- Logical framework: Levesque’s logic of only-knowing $K_A \alpha \ O_A \alpha$
- Could implement reasoning service with off-the-shelf theorem prover
A knows A
A knows that
but A doesn’t know B

B knows B
KB: \[ O_A (A = 7 \land \forall x (B = x \rightarrow O_B B = x)) \]

A only knows that $A = 7$ and
that if $B = x$, then B only knows that $B = x$
KB: \[ \mathcal{O}_A (A = 7 \land \forall x (B = x \rightarrow \mathcal{O}_B B = x)) \]

entails

Query: \[ \mathcal{K}_A \exists z (\underbrace{B = z}_{\text{de dicto}} \land \underbrace{\neg \mathcal{K}_A B = z}_{\text{de re}} \land \underbrace{\mathcal{K}_B B = z}_{\text{de re}}) \]

A knows that some number is equal to \( B \), but A doesn’t know what the number is, and B does know it.
Reduction: Eliminate Modal Operators

\[ \text{KB: } O_A(A = 7 \land \forall x (B = x \rightarrow O_B B = x)) \]

entails

\[ \text{Query: } K_A \exists z (B = z \land \neg K_A B = z \land K_B B = z) \]
**Reduction: Eliminate Modal Operators**

**KB:** \( O_A (A = 7 \land \forall x (B = x \rightarrow O_B B = x)) \)

entails

**Query:** \( K_A \exists z (B = z \land \neg K_A B = z \land K_B B = z) \)
Reduction: Eliminate Modal Operators

\[ KB: \quad O_A(\mathsf{A} = 7 \land \forall x (B = x \rightarrow O_B B = x)) \]

entails

\[ Query: \quad K_A \exists z (B = z \land \neg K_A B = z \land K_B B = z) \]

\[ KB': \quad O_A(\mathsf{A} = 7 \land \forall x (B = x \rightarrow P(x))) \]
Reduction: Eliminate Modal Operators

KB: \[ O_A \left( A = 7 \land \forall x \left( B = x \rightarrow O_B B = x \right) \right) \]

entails

Query: \[ K_A \exists z \left( B = z \land \neg K_A B = z \land K_B B = z \right) \]

KB': \[ O_A \left( A = 7 \land \forall x \left( B = x \rightarrow P(x) \right) \right) \]

entails

Query': \[ K_A \exists z \left( B = z \land \neg K_A B = z \land \exists x (P(x) \land \quad ) \right) \]

"For which \( x, z \) does \( O_B B = x \) entail \( K_B B = z \)?"
Reduction: Eliminate Modal Operators

KB: \[ O_A(A = 7 \land \forall x (B = x \rightarrow O_B B = x)) \]
entails

Query: \[ K_A \exists z (B = z \land \neg K_A B = z \land K_B B = z) \]

KB': \[ O_A(A = 7 \land \forall x (B = x \rightarrow P(x))) \]
entails

Query': \[ K_A \exists z (B = z \land \neg K_A B = z \land \exists x (P(x) \land \text{[Levesque '84]}) \]

“For which x, z is \( B = x \rightarrow B = z \) valid?”

Call validity oracle!
Reduction: Eliminate Modal Operators

KB: \[ O_A \left( A = 7 \land \forall x (B = x \rightarrow O_B B = x) \right) \]

entails

Query: \[ K_A \exists z \left( B = z \land \neg K_A B = z \land K_B B = z \right) \]

KB': \[ O_A \left( A = 7 \land \forall x (B = x \rightarrow P(x)) \right) \]

entails

Query': \[ K_A \exists z \left( B = z \land \neg K_A B = z \land \exists x (P(x) \land x = z) \right) \]

"For which \( x \), \( z \) is \( B = x \rightarrow B = z \) valid?"

Call validity oracle! [Levesque ’84]
Assumption: agents always only-know something about each other.

- $\times \quad O_A(P \rightarrow O_B \alpha)$
- $\checkmark \quad O_A((P \rightarrow O_B \alpha) \land (\neg P \rightarrow O_B \beta))$
- $\checkmark \quad O_A(A = 7 \land \forall x (B = x \rightarrow O_B(B = x)))$

Then:

**Reasoning in multi-agent epistemic knowledge bases**

Turing-reduces to

**classical validity**
Appendix
Multi-Agent Knowledge Bases

\[ \text{O}_A \alpha = A \text{ only-knows } \alpha \]  
[Levesque ’84]

- A considers all models of \( \alpha \) possible
Multi-Agent Knowledge Bases

- $O_A \alpha = A$ only-knows $\alpha$  [Levesque ‘84]
  - $A$ considers all models of $\alpha$ possible

- $O_A \phi$ entails $K_A \psi \iff \phi \rightarrow \psi$ is valid
  - provided that $\phi, \psi$ are objective!
Multi-Agent Knowledge Bases

- \( O_A \alpha \ = \ A \text{ only-knows } \alpha \)  
  - \( \implies \) \( A \) considers all models of \( \alpha \) possible  

- \( O_A \phi \) entails \( K_A \psi \iff \phi \rightarrow \psi \) is valid  
  - \( \implies \) provided that \( \phi, \psi \) are objective!

- \( O_A \alpha \) implies \( O_A \beta \iff \alpha \) and \( \beta \) are equivalent  
  - \( \implies \) \( A \) can only-know at most one formula
Multi-Agent Knowledge Bases

- \( O_A \alpha = A \) only-knows \( \alpha \) \quad [Levesque ’84]
  - A considers all models of \( \alpha \) possible

- \( O_A \phi \) entails \( K_A \psi \iff \phi \rightarrow \psi \) is valid
  - provided that \( \phi, \psi \) are objective!

- \( O_A \alpha \) implies \( O_A \beta \iff \alpha \) and \( \beta \) are equivalent
  - A can only-know at most one formula

- \( O_A \alpha \) is a multi-agent KB \iff every model of \( \alpha \) satisfies some \( O_B \beta \)
  - \( O_A(P \rightarrow O_B \alpha) \)
  - \( O_A((P \rightarrow O_B \alpha) \land (\neg P \rightarrow O_B \beta)) \)
  - \( O_A \forall x(f = x \rightarrow O_B \alpha(x)) \)
Replace each $O_A \alpha(\vec{x})$ with a fresh atom $P_\alpha(\vec{x})$
Reduction

- Replace each $O_A \alpha(\vec{x})$ with a fresh atom $P_\alpha(\vec{x})$

- Replace each $K_A \gamma(\vec{z})$ with a disjunction of

  $$\exists \vec{x} \left( P_\alpha(\vec{x}) \land \text{“for which } \vec{x}, \vec{z} \text{ is } \alpha(\vec{x}) \rightarrow \gamma(\vec{z}) \text{ is valid?”} \right)$$

over all $O_A \alpha(\vec{x})$ at the same modal nesting level
Reduction

- Replace each $O_A \alpha(\vec{x})$ with a fresh atom $P_\alpha(\vec{x})$

- Replace each $K_A \gamma(\vec{z})$ with a disjunction of

  $$\exists \vec{x} \left( P_\alpha(\vec{x}) \land \text{"for which } \vec{x}, \vec{z} \text{ is } \alpha(\vec{x}) \rightarrow \gamma(\vec{z}) \text{ is valid?"} \right)$$

  over all $O_A \alpha(\vec{x})$ at the same modal nesting level

- Axiomatise that $P_\alpha(\vec{x}), P_\beta(\vec{y})$ introduced for $O_A \alpha(\vec{x}), O_A \beta(\vec{y})$

  $$P_\alpha(\vec{x}) \rightarrow (P_\beta(\vec{y}) \leftrightarrow \text{"for which } \vec{x}, \vec{y} \text{ is } \alpha(\vec{x}) \rightarrow \beta(\vec{y}) \text{ is valid?"})$$
Summary

Multi-agent KB:

- Based on Levesque’s logic of only-knowing
- Every model of a multi-agent KB must satisfy some $O_B \beta$
- Allows for incomplete knowledge about other agent’s knowledge
Summary

Multi-agent KB:
- Based on Levesque’s logic of only-knowing
- Every model of a multi-agent KB must satisfy some $O_B \beta$
- Allows for incomplete knowledge about other agent’s knowledge

Reduction to classical reasoning:
- Oracle for FOL validity
- Turing reduction: calls oracle many times
- Would need FO-K45 oracle if it weren’t for
Summary

Multi-agent KB:
- Based on Levesque’s logic of only-knowing
- Every model of a multi-agent KB must satisfy some $O_B \beta$
- Allows for incomplete knowledge about other agent’s knowledge

Reduction to classical reasoning:
- Oracle for FOL validity
- Turing reduction: calls oracle many times
- Would need FO-K45 oracle if it weren’t for

Implementation options:
- FOL theorem prover (e.g., Vampire)
- Limited belief logic