Logical Omniscience in the Semantics of BAN Logic

Mika Cohen, Mads Dam

LECS/ICT/KTH
Stockholm, Sweden

FCS’05, 2005-07-01
Logical omniscience problem

Combination knowledge - computation/cryptography problematic

- Difference between feasibly computable - and logical consequence

Wanted:

1. Agent knows all feasibly computable consequences of what it knows

Not wanted:

2. Agent knows all logical consequences of what it knows

Logical omniscience problem: Obtain (1) but avoid (2)
Logical omniscience problem in BAN

Example
- \(fresh \ M \models fresh \ \{M\}_K\)

Logical omniscience
- \(\Box_a \ fresh \ M \models \Box_a \ fresh \ \{M\}_K\)

But
- \(fresh \ \{M\}_K\) not feasibly computable from \(fresh \ M\)

BAN
- Feasible cryptographic computation \(\approx\) Dolev-Yao
- \(\Box_a \ K \text{ good for } a \cdot b\), \(\Box_a \ fresh \ M \models \Box_a \ fresh \ \{M\}_K\)
  - Typical BAN rule
Why is logical omniscience an issue for BAN?

BAN is a just proof system
- Deductive protocol verification

Can we bring semantical methods to BAN?
- Model checking
- Checking BAN extensions/variations
- Semantically based theorem provers (for BAN extensions)
- Knowledge programs

If semantics makes agents logically omniscient:
- Semantics is unfaithful to BAN
- Semantical methods are untrustworthy

Logical omniscience in all existing semantics for BAN-like logics
Objective

Solve the logical omniscience problem in the semantics of BAN

Requirements on our semantics

1. Knowledge is **not** closed under *logical* consequences
2. Knowledge is closed under *feasibly computable* consequences
3. Validates BAN

Why not require completeness w.r.t. BAN?

- BAN open ended, vaguely defined proof system
- ”Add new proof rules as needed”

Completeness w.r.t. ”conservative” extension desirable

- Return to this in conclusion
Existing semantics for BAN-like logics
Classical multi-agent system semantics

Canonical in computer science
  - Fagin/Halpern/Moses/Vardi (95)

Applied to BAN
  - Syverson (01), Decker (01), Halpern/Pucella/Meyden (03), Jacobs (04)
Classical semantics: Truth condition

Multi-agent system
- Set of system states $s, s', \ldots$
- $s|a$ is local state of $a$ in $s$
  - "All data available to $a$ at $s$"
  - Eg. local action trace

Agent knows a fact if her local state forces the fact
- $s \models \Box_a F \iff \forall s' : s|a = s'|a \Rightarrow s' \models F$
Classical semantics: Example

Example system

- $b \text{ sen } M \rightarrow a \text{ rec } M \quad \neg \Box_a b \text{ sent } M$
- $a \text{ received } M$

- $c \text{ sen } M \rightarrow a \text{ rec } M \quad \Box_a a \text{ received } M$

Receive introspection

- $a \text{ received } M \models \Box_a a \text{ received } M$

Logical omniscience

Combination more problematic than logical omniscience alone
AT-style semantics

- Multi-agent system semantics adjusted for crypto communication
- Abadi/Tuttle 91
- Refinements/variations
  - Syverson/Oorschot (96), Wedel/Kessler (95)
AT-style semantics: Truth condition

Hides parts of local state to agent herself

- $\text{Hide}$ replaces unopened cipher texts with $\bot$
- $\text{Hide}(a \text{ receives } \{M\}_K) = a \text{ receives } \bot$

Agent knows a fact if her local state after hiding forces the fact

- $s \models \square_a F \iff \forall s' : \text{Hide}(s|a) = \text{Hide}(s'|a) \Rightarrow s' \models F$
AT-style semantics: Example

Example system

\[ b \text{ sen } \{M\}_K \rightarrow a \text{ rec } \{M\}_K \rightarrow \bullet \neg \Box_a a \text{ received } \{M\}_K \]

\[ c \text{ sen } \{M'\}_{K'} \rightarrow a \text{ rec } \{M'\}_{K'} \rightarrow \bullet \rightarrow \bullet \rightarrow \bullet \text{ received } M \]

\[ \text{Hide}(s|a) = \text{Hide}(s'|a) = a \text{ rec } \bot \]

Receive introspection broken

\[ a \text{ received } M \not\models \Box_a a \text{ received } M \]

\[ \text{BAN invalidated} \]

Logical omniscience
Kripke semantics

Standard framework for modal logics
Agent knows a fact if fact holds at every obs. eq. state

\[ s \models \Box_a F \iff \forall s' : s \equiv_a s' \Rightarrow s' \models F \]

\[ s \equiv_a s' \iff \text{s and } s \text{ equivalent up to } a:\text{s power of observation} \]

Classical multi-agent system semantics

\[ s \equiv_a s' \iff s|_a = s'|_a \]

AT semantics

\[ s \equiv_a s' \iff \text{Hide}(s|_a) = \text{Hide}(s'|_a) \]
Logical omniscience in Kripke

Assume

1. \( \Delta \models F \)
2. \( s \models \Box_a \Delta \)
3. \( s \equiv_a s' \)

2 + 3 \Rightarrow

4. \( s' \models \Delta \)

1 + 4 \Rightarrow

5. \( s' \models F \)

3 + 5 \Rightarrow

6. \( s \models \Box_a F \)
A generalization of Kripke
Epistemic equivalence indexed by renamings

Example system again

\[ b \text{ sen } \{M\}_K \rightarrow a \text{ rec } \{M\}_K \]

\[ c \text{ sen } \{M'\}_{K'} \rightarrow a \text{ rec } \{M'\}_{K'} \]

\{M\}_K at s corresponds for a to \{M'\}_{K'} at s'

- Observable properties of \{M\}_K at s

\[
\begin{align*}
\text{=} \\
\text{Observable properties of } \{M'\}_{K'} \text{ at } s'
\end{align*}
\]

We make \( \equiv_a \) keep track of message correspondences

- Index \( \equiv_a \) by renaming \( r \) of messages

\( s \equiv^r_a s' \)

- \( s \) and \( s' \) observationally equivalent for \( a \)
- \( M \) at \( s \) corresponds for \( a \) to \( r(M) \) at \( s' \), for all \( M \)
Requirements for $s \equiv^r_a s'$

$r$ should respect local state
- $r(s|a) = s'|a$

$r$ should respect keys used
- $K$ used by $a$ at $s \Rightarrow r(\{M\}_K) = \{r(M)\}_{r(K)}$

We return later to "$K$ used by $a$ at $s$"
New truth condition for knowledge

Agent knows message satisfies property if corresponding messages at obs. eq. states satisfy property

\[ s \models \square_a F(M) \iff \forall s' : \forall r : s \equiv_a^r s' \Rightarrow s' \models F(r(M)) \]

Example system

\[
\begin{array}{ccc}
  b & \text{sen} & \{M\}_K \\
  \bullet & \rightarrow & \bullet \\
  c & \text{sen} & \{M'\}_{K'} \\
  \bullet & \rightarrow & \bullet
\end{array}
\]

\[
\begin{array}{ccc}
  a & \text{rec} & \{M\}_K \\
  \bullet & \rightarrow & \bullet \\
  a & \text{rec} & \{M'\}_{K'} \\
  \bullet & \rightarrow & \bullet
\end{array}
\]

\[ \Box_a a \text{ received } \{M\}_K \]

Receive introspection restored
Agents do not know all *logical* consequences

1  \[ \Delta \models F \]
   
   \[ s \models \Box_a \Delta \]
   
   \[ s \equiv^r_a s' \]
   
   \[ s' \models r(\Delta) \]
   
   \[ \Rightarrow \]
   
   \[ s' \models r(\Delta) \]
   
   \[ \Rightarrow \ldots \]

(1) is irrelevant!

\[ r(\Delta) \models r(F) \] needed to obtain \( s \models \Box_a F \)
Agents know all *feasibly computable* consequences

”feasibly computable consequence” vague
- No existing attempt to make precise for BAN-like logics

Our proposal
- $\Delta \models F \Rightarrow a \text{ uses } \text{Keys}(\Delta, F), \Box_a \Delta \models \Box_a F$

Example
- $\text{fresh } x \models \text{fresh } \{x\}_y \Rightarrow a \text{ uses } y, \Box_a \text{fresh } x \models \Box_a \text{fresh } \{x\}_y$
- Univ. subst. $\Rightarrow a \text{ uses } K, \Box_a \text{fresh } M \models \Box_a \text{fresh } \{M\}_K$

Abstraction of BAN rules
Soundness lemma 1: Keys known are used

- $\square_a K \text{ good } a \cdot b \models a \text{ uses } K$
- Implicit in BAN
- Depends on definition of *keys used*

Customary definition: Keys used are the keys extracted

- Received and initially possessed messages closed under un-pairing and decryption
- Lemma (1) fails in some models

New definition: Keys used are the keys known

- $s \models a \text{ uses } K \iff \exists \text{ predicate } p : s \models \square_a p(K)$
- (1) immediate
Cannot define \( a \) uses by \( \Box_a \) directly

- \( \Box_a \) defined by \( \equiv'_a \) defined by \( a \) uses

Can define \( a \) uses and \( \Box_a \) through mutual recursion

We select least definition of \( a \) uses satisfying

- \( s \models a \text{ uses } K \iff \exists \) predicate \( p : s \models \Box_a p(K) \)

Always exists

Recent work: If predicates only apply to existing messages:

- New definition eq. to customary
- BAN predicates need slight modification
S5 axioms

\[ T \quad \square_a F \models F \]
\[ \quad \quad \quad \quad \quad \quad \quad s \equiv^t_a s \quad ("\text{Reflexivity}" \ ) \]

\[ 4 \quad \square_a F \models \square_a \square_a F \]
\[ \quad \quad \quad \quad \quad \quad \quad s \equiv^r_a s', \quad s' \equiv^r_a s'' \Rightarrow s \equiv^r\circ^r_a s'' \quad ("\text{Transitivity}" \ ) \]

\[ 5 \quad \neg \square_a F \models \square_a \neg \square_a F \]
\[ \quad \quad \quad \quad \quad \quad \quad s \equiv^r_a s' \Rightarrow s' \equiv^{r^{-1}}_a s \quad ("\text{Symmetry}" \ ) \]
Other related work

- Counterpart semantics
  - Lewis (68)
  - Not computationally grounded
  - Agents are logically omniscient
- Resource bounded knowledge
  - Fagin/Halpern/Moses/Vardi (95)
  - None attempted for BAN
  - Breaks radically with Kripke semantics
Conclusion
Summary

Kripke semantics

1. Agent knows all *logical* consequences of what she knows

Intended in BAN:

2. Agent knows all *feasibly computable* consequences of what she knows

Mismatch makes Kripke semantics of limited use for BAN

We propose a generalization of Kripke

- Epistemic equivalence relation keeps track of message correspondences
- Avoids (1)
- Achieves (2)
- Validates BAN

Application: Semantically based methods

- Model checking
- :
Current work

Completeness
- For multi-agent models
- For message passing systems and fixed vocabulary

Semantics for first-order extension
- Useful when data is complex, partly hidden
- Translation of BAN related logics
Thanks!