Modeλ Checking DEL for Guessing Games and Cryptography

Malvin Gattinger

ILLC, Amsterdam

October 3rd 2014

LiRA-Workshop
"The Logical Dynamics of Information, Agency and Interaction"

Outline

Guessing Games

What does it mean to know a number?

JAN: "I have a number in mind, in the range from one to ten. You may take turns guessing. Whoever guesses the number first wins."

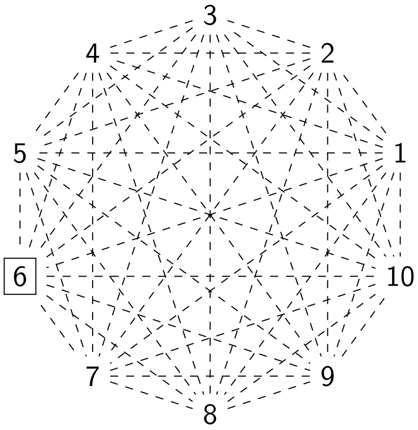
GAIA: "But how can we know you are not cheating?"

ROSA: "Please write down the number before we start guessing, so you can show it afterwards as a proof."

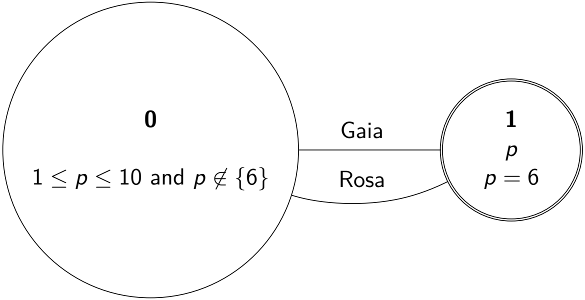
JAN: "Okay."

[*Jan writes* 6 *on a piece of paper, hidden from Gaia and Rosa.*]

What does it mean to know a number?



What does it mean to know a number?



Agents: Jan, Gaia, Rosa

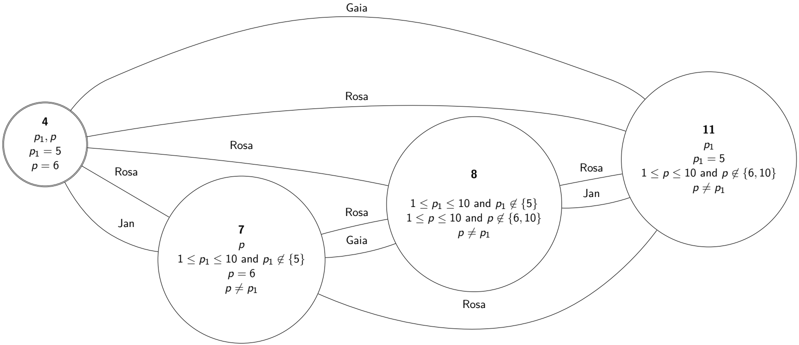
Models

Definition (Guessing Game Models)
$\mathcal{M} = (W, \mathcal{R}, V)$ where

- $(W, \mathcal{R})$ is a multi-agent S5 frame,
- $V : w \mapsto (P_w, f_w, C_w^+, C_w^-)$ is a valuation:
  - $P_w \subseteq \mathbf{P}$ are the basic propositions true at w,
  - f_w is a function that assigns to some propositions $q \in Q$ triples (n, m, X), meaning that the value of q is between n and m but not in X. We also demand that
    (i) whenever $q \in P_w$ then $n = m$ and $X = \varnothing$
    (ii) whenever $q \in P_v \cap P_w$ for $v, w \in W$ then $f_v(q) = f_w(q)$
  - $C_w^+, C_w^- \subseteq Q^2$ are in/equality constraints in the following sense: $(p, q) \in C_w^+$ expresses that p and q have the same values and $(p, q) \in C_w^-$ expresses that p and q have different values at w.

Models

Example



Agents: Jan, Gaia, Rosa

Updates (two examples)

Register Creation $p \stackrel{i}{\leftarrow} N$: Create secret variable p for agent i with value N.

- p must be globally false before.
- Copy all worlds, let $p = N$ at the new worlds and let p have any other value at the others.
- Connect the worlds for everyone but i.

Announcement $!p = q$: Tell everyone that $p = q$.

- $p = q$ must be true at the current world.
- Where p and q are false, add $p = q$ to the constraints.

Exact definitions: Action structures as in [BMS98] and [BEK06].

Syntax

Definition (Language)
The language $\mathcal{L}_{\mathsf{GG}}$ consists of formulas, commands and expressions:

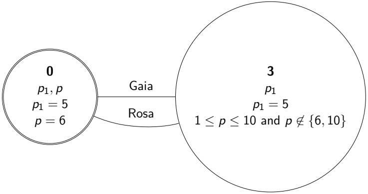
$$\begin{array}{lcl}
\phi & ::= & \top \mid p \mid p = E \mid \neg\phi \mid \phi \wedge \phi \mid K_i\phi \mid G\phi \mid \langle C \rangle \phi \\
C & ::= & !p = E \mid !p \neq E \mid p \stackrel{i}{\leftarrow} N \mid C; C \\
E & ::= & p \mid N
\end{array}$$

Assignments

Definition (Assignments and Agreement)
An assignment is a function $h : \mathrm{Prop} \to \mathbb{N}$.
It *agrees* with a world ($h \multimapdotinv w$) iff

- for all $q \in Q$: $f^0_w(q) \leq h(q) \leq f^1_w(q)$ and $h(q) \notin f^2_w(q)$
- positive constraints C^+_w: if $(p, q) \in C^+_w$ then $h(p) = h(q)$
- negative constraints C^-_w: if $(p, q) \in C^-_w$ then $h(p) \neq h(q)$.

Example
$h = \{p \mapsto 6, p_1 \mapsto 5\}$
agrees with **0**, but not with **3**.



Agents: Jan, Gaia, Rosa

Semantics

Definition (Truth with regard to assignments)

$$
\begin{array}{lll}
\mathcal{M}, w, h \models \top & & \text{always} \\
\mathcal{M}, w, h \models p & \text{iff} & p \in P_w \\
\mathcal{M}, w, h \models p_1 = p_2 & \text{iff} & h(p_1) = h(p_2) \\
\mathcal{M}, w, h \models p = N & \text{iff} & h(p) = N \\
\mathcal{M}, w, h \models \neg\phi & \text{iff} & \text{not } \mathcal{M}, w, h \models \phi \\
\mathcal{M}, w, h \models \phi_1 \wedge \phi_2 & \text{iff} & \mathcal{M}, w, h \models \phi_1 \text{ and } \mathcal{M}, w, h \models \phi_2 \\
\mathcal{M}, w, h \models K_i\phi & \text{iff} & wR_iw' \Rightarrow \forall h' \multimapinv w' : \mathcal{M}, w', h' \models \phi \\
\mathcal{M}, w, h \models G\phi & \text{iff} & \forall w' \in W \forall h' \multimapinv w' : \mathcal{M}, w', h' \models \phi \\
\mathcal{M}, w, h \models \langle !\ p = E \rangle\phi & \text{iff} & \mathcal{M}, w, h \models p = E \text{ and } \mathcal{M}^{!p=E}, w, h \models \phi \\
\mathcal{M}, w, h \models \langle !\ p \neq E \rangle\phi & \text{iff} & \mathcal{M}, w, h \models p \neq E \text{ and } \mathcal{M}^{!p\neq E}, w, h \models \phi \\
\mathcal{M}, w, h \models \langle p \stackrel{i}{\leftarrow} N \rangle\phi & \text{iff} & \mathcal{M}, w, h \models G\neg p \text{ and} \\
& & \mathcal{M}^{p \stackrel{i}{\leftarrow} N}, w, (h \cup \{(p, N)\}) \models \phi \\
\mathcal{M}, w, h \models \langle A_1; A_2 \rangle\phi & \text{iff} & \mathcal{M}, w, h \models \langle A_1 \rangle \langle A_2 \rangle\phi
\end{array}
$$

Semantics

Definition (World Level Truth)

$$\mathcal{M}, w \models \phi \quad \text{iff} \quad \forall h \text{ with } w \multimap h : \mathcal{M}, w, h \models \phi.$$

A formula ϕ is *valid* iff for all $\mathcal{M}, w$ we have $\mathcal{M}, w \vDash \phi$.
We then write $\vDash \phi$.

This leaves some formulas undecided on the world level.
But we still have:

Theorem

For all $\mathcal{M}$, w, i and ϕ we have either $\mathcal{M}, w \models K_i\phi$ or $\mathcal{M}, w \Dashv K_i\phi$.

Reduction Axioms (some of them)

P5) $\langle !p = E\rangle \widehat{K_i}\phi \leftrightarrow (p = E \wedge \widehat{K_i}(\langle !p = E\rangle\phi))$

P6) $\langle !p = E\rangle G\phi \leftrightarrow (p = E \wedge G(p = E \rightarrow \langle !p = E\rangle\phi))$

R3a1) $\langle p \xleftarrow{i} N\rangle(p = N) \leftrightarrow (G\neg p)$

R3a1') $\langle p \xleftarrow{i} N\rangle(p = M) \leftrightarrow \perp$ where $M \neq N$

R3a2) $\langle p \xleftarrow{i} N\rangle(q = M) \leftrightarrow (G\neg p \wedge (q = M))$ where $p \neq q$

R3b1) $\langle p \xleftarrow{i} N\rangle(p = p) \leftrightarrow (G\neg p)$

R3b1') $\langle p \xleftarrow{i} N\rangle(p = q) \leftrightarrow (G\neg p \wedge (q = N))$ where $p \neq q$

R3b2) $\langle p \xleftarrow{i} N\rangle(q = p) \leftrightarrow (G\neg p \wedge (q = N))$ where $p \neq q$

R3b2') $\langle p \xleftarrow{i} N\rangle(q = r) \leftrightarrow (G\neg p \wedge (q = r))$ where $p \neq q$ and $p \neq r$

R6) $\langle p \xleftarrow{i} N\rangle(K_i\phi) \leftrightarrow (G\neg p \wedge K_i(G\neg p \rightarrow \langle p \xleftarrow{i} N\rangle\phi))$

R7) $\langle p \xleftarrow{i} N\rangle(K_j\phi) \leftrightarrow (G\neg p \wedge K_j\phi)$ where $j \neq i$

R8) $\langle p \xleftarrow{i} N\rangle(G\phi) \leftrightarrow G(\langle p \xleftarrow{i} N\rangle\phi)$

Axiomatization

Theorem (Soundness)
All reduction axioms are valid.

Definition (Proof System)
We write $\vdash \phi$ iff ϕ is provable using propositional tautologies, standard rules for the S5 modalities K_i and the global modality G and the reduction axioms.

Theorem (Completeness)
For all $\phi \in \mathcal{L}_{\mathsf{GG}}$, if $\models \phi$, then $\vdash \phi$.

Cryptography

Communication

"Let me tell you a secret ..."

Goal: Model the intended audience, but also eavesdropping.

- New proposition: $w \models L_i$ means Agent i is listening at w.
- Two new commands: $\langle \mathbf{Open}_i \rangle$ and $\langle \mathbf{Close}_i \rangle$.
- Announcements are only heard by the current listeners.

Computation

"If I know that $p = 5$ then I also know that $p + p = 10$."

Goal: Give agents some (realistic) computational power.
For now: Primality-Testing and modular arithmetic, which are both assumed to be feasible in Cryptography.

- New propositions: $\mathbf{Prime}E$, $\mathbf{Coprime}EE$
- New expressions: $E + E \bmod E$, $E \times E \bmod E$, $E^E \bmod E$

The full language

Definition (Language)
The language $\mathcal{L}_{\mathsf{ECL}}$ consists of the following formulas, commands and expressions.

$$\begin{array}{lcl} \phi & ::= & \top \mid p \mid L_i \mid p = E \mid \neg\phi \mid \phi \wedge \phi \mid K_i\phi \mid G\phi \mid \langle C\rangle\phi \\ & & \mid \textbf{Prime } E \mid \textbf{Coprime } E\ E \\ \\ C & ::= & p \stackrel{i}{\leftarrow} E \mid \textbf{Open}_i \mid \textbf{Close}_i \mid\ !p \mid\ !p = N \mid\ !p = p \\ & & \mid\ !p \neq N \mid\ !p \neq p \mid\ ?\phi \\ \\ E & ::= & p \mid N \mid E + E \text{ mod } E \mid E \times E \text{ mod } E \mid E^E \text{ mod } E \end{array}$$

The Diffie-Hellman Key Exchange

(Whitfield Diffie and Martin Hellman [DH76])

1. Alice and Bob agree on a prime p and a base $g < p$ such that g and $p - 1$ are coprime.
2. Alice picks a secret N and sends $g^N \bmod p = A$ to Bob.
3. Bob picks a secret M and sends $g^M \bmod p = B$ to Alice.
4. Alice calculates $k = B^N \bmod p$.
5. Bob calculates $k = A^M \bmod p$.
6. They now have a shared key $k = (g^M)^N = (g^N)^M \bmod p$.

If the Diffie-Hellman problem is hard, Eve does not know k.
NB: The protocol is only secure against *passive* eavesdroppers.

Diffie-Hellman in ECL

Let $\mathcal{M}_{\mathbf{DH}}$ be the blissful ignorance model for Alice, Bob and Eve.
Let $\mathbf{DH}_{g,p,N,M}$ be the command:

$$\begin{aligned}&\textbf{Coprime } g\ (p-1)\ \textbf{;}\\ &q_1 \overset{a}{\leftarrow} N\ \textbf{;}\ r_1 \overset{a}{\leftarrow} (g^{q_1} \bmod p)\ \textbf{; Open}_b\ \textbf{;}\ !r_1\ \textbf{; Close}_b\ \textbf{;}\\ &q_2 \overset{b}{\leftarrow} M\ \textbf{;}\ r_2 \overset{b}{\leftarrow} (g^{q_2} \bmod p)\ \textbf{; Open}_a\ \textbf{;}\ !r_2\ \textbf{; Close}_a\ \textbf{;}\\ &s_1 \overset{a}{\leftarrow} {r_2}^{q_1} \bmod p\ \textbf{;}\ s_2 \overset{b}{\leftarrow} {r_1}^{q_2} \bmod p\end{aligned}$$

Let $\psi_{\mathbf{DH}} := (s_1 = s_2) \wedge (K_a s_1 \wedge K_b s_2) \wedge (\neg K_e s_1 \wedge \neg K_e s_2)$.
Then we have:

$$\mathcal{M}_{\mathbf{DH}}, w \vDash \langle \mathbf{DH}_{g,p,N,M} \rangle \psi_{\mathbf{DH}}$$

Model Checking

Live Demo

Example 1

Creating a secret number for Alice and telling Bob about it.

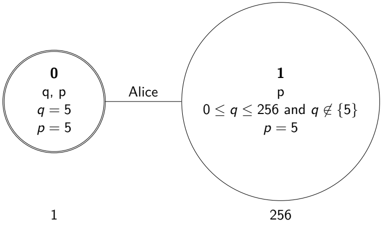
Example 2

Order matters: “Hey Bob! Hey Alice!” $\neq$ “Hey Alice! Hey Bob!”

Monte Carlo Method

$$\mathcal{M}, w \;\approx\!\!\models\; \phi \text{ iff for some randomly picked } h \multimap w : \mathcal{M}, w, h \models \phi$$

For many formulas we do not have to check all possible assignments.
Example: Is $K_a(p = q)$ is true at $\mathbf{0}$?



No, and checking one assignment at $\mathbf{1}$ suffices.

NB: There are also cases where this almost always goes wrong.

Normal VS. Monte-Carlo Methods

How long does it take to check $\mathcal{M}_{\mathbf{DH}}, w \vDash \langle \mathbf{DH}_{g,p,N,M} \rangle \psi_{\mathbf{DH}}$?

`registersize`	Normal	Monte Carlo
2^8	1.07	2.74
2^9	1.36	2.82
2^{10}	2.13	3.41
2^{11}	3.59	3.24
2^{12}	5.17	2.8
2^{13}	11.56	3.28
2^{14}	22.66	3.57
2^{15}	44.44	4.1
2^{16}	81.26	3.52

Conclusion

Conclusion

- To know a number is to distinguish a true value from all others
- Register models for DEL:
  reduce "Knowledge of" to "Knowledge that"
- Axiomatization for GG
- Explicit communication and computation in ECL
- Example: Diffie-Hellman
- Implemented both frameworks in Haskell
- Efficient but probabilistic Monte Carlo method

- Future ideas: axiomatize full ECL, improve implementation, non-S5, other protocols, automated attack finding, ...

References

Johan van Benthem, Jan van Eijck, and Barteld Kooi.
Logics of communication and change.
Information and computation, 204(11):1620–1662, 2006.

Alexandru Baltag, Lawrence S. Moss, and Slawomir Solecki.
The logic of public announcements, common knowledge, and private suspicions.
In I. Bilboa, editor, *Proceedings of TARK'98*, pages 43–56, 1998.

Whitfield Diffie and Martin Hellman.
New directions in cryptography.
Information Theory, IEEE Transactions on, 22(6):644–654, 1976.

Thank you.

http://is.gd/eclonline