Elements of epistemic crypto logic

van Eijck, J.; Gattinger, M.

Publication date
2015

Document Version
Final published version

Published in
AAMAS '15

Elements of Epistemic Crypto Logic

(Extended Abstract)

Jan van Eijck
CWI and ILLC
Amsterdam, The Netherlands
jve@cwi.nl

Malvin Gattinger
ILLC
Amsterdam, The Netherlands
malvin@w4eg.eu

ABSTRACT

Representation of ignorance about large numbers — agent a does not know agent b's key — is not feasible in standard Kripke semantics. The paper introduces register models that allow for compact representation of such ignorance. This is used to design a sound and complete language for number guessing games. The probabilities generated by our semantics allow for and motivate Monte Carlo model checking for register models. We show that the approach can be extended to a real life setting, namely the analysis of cryptographic security protocols. We look at a well known security protocol for secret key distribution over an insecure network, and point out how this can be analyzed with our modified version of Kripke semantics.

Categories and Subject Descriptors

I.2.4 [**Knowledge Representation Formalisms & Methods**]: Modal Logic; D.2.4 [**Software/Program Verification**]: Model Checking; E.3 [**Data Encryption**]: Public key cryptosystems

General Terms

Languages, Theory, Verification

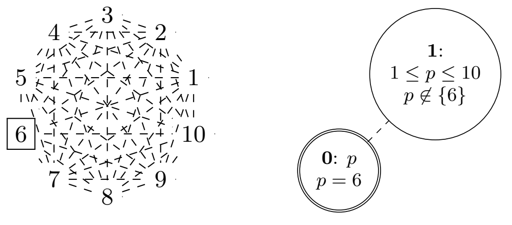
Keywords

Dynamic Epistemic Logic, Model Checking, Cryptography

1. KNOWLEDGE OF NUMBERS

Cryptographic protocols deal with the knowledge of secrets which can usually be represented as numbers. The established formal semantics for knowledge are Kripke models. This paper introduces a new variant of them, suitable for checking knowledge and communication involving large numbers.

Consider the following number guessing game, played between Jan, Gaia and Rosa. Jan says: "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 and Rosa agree and after a number of rounds, Jan announces: "Rosa, you have won." A naive analysis of this game can be given as a multi-agent Kripke model with ten possible worlds. In Figure 1a the actual world — where the number is 6 — is indicated by a box, and dashed lines represent the ignorance of the twins. Now suppose the following exchange takes place: Gaia: "Eight?" … Jan: "No". This updatesthe model: World 8 gets removed from the graph.



(a) Naive analysis.

(b) Register model.

Figure 1: Two ways to model a guessing game.

But the twins complain: "How can we know you are not cheating? Please write down the number before we start." This motivates the notion of a *register*. It allows Jan to prove that he knew the number because he had fixed it beforehand and did not just accept Rosa's guess. It therefore suffices to distinguish two possibilities: In the actual world register p has the value 6 and in the other world it can be anything else in a given range. Jan knows p, the children do not know p. This leads to the register model in Figure 1b. Suppose this gets updated with Gaia: "Ten?" Jan: "No". Then 10 drops out of the range of p and we obtain a model with constraint $p \notin \{6, 10\}$ at world **1**. If we go on with Rosa: "Six?" and Jan: "Yes", the model is restricted to world **0**.

We can also view guessing as introducing new variables. Figure 2 shows the moment when Gaia prepares to announce a guess but has not yet revealed it. She knows that q is 5, but Jan (solid) and Rosa (dashed) do not know it yet. If Gaia reveals her guess by announcing $q = 5$, the model is restricted to **0** and **1**. Jan can then announce that the guess is wrong, i.e. $p \neq q$, adding the constraint $p \notin \{5, 6\}$ at **1**. Without registers, Figure 2 blows up to a model with 100 worlds. Generally, to model k registers, each of m bits to allow numbers between 0 and 2^m, we get a blow-up from 2^k to $(2^k)^m$ possibilities.

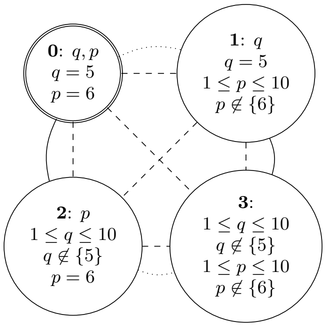


Figure 2: Gaias guess.

Appears in: *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015), Bordini, Elkind, Weiss, Yolum (eds.), May 4–8, 2015, Istanbul, Turkey.*

2. GUESSING GAME LOGIC

Definition 1. *Let p range over a set of propositions $\mathbf{P}$, N over $\mathbb{N}$ and i over a finite set of agents I. The guessing game language consists of formulas, commands and expressions:*

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

Definition 2. *A register model for agents I and propositions $\mathbf{P}$ is a tuple $\mathcal{M} = (W, R, V)$ where W is a set of possible worlds, $R = (R_i)_{i \in I}$ are equivalence relations on W and V is a valuation for some $Q \subseteq \mathbf{P}$ (the active vocabulary), mapping $w \in W$ to $V(w) = (P_w, f_w, C_w^+, C_w^-)$ where*

- *$P_w \subseteq Q$ (the true basic propositions of w),*
- *$f_w : Q \to \mathbb{N} \times \mathbb{N} \times \mathcal{P}(\mathbb{N})$ such that if $q \in P_v \cap P_w$, then $f_v(p) = f_w(q) = (n, m, X)$ with $n = m$ and $X = \varnothing$.*
- *$C_w^+ \subseteq Q^2$ and $C_w^- \subseteq Q^2$ (the in/equality constraints of w) such that no $(p, q) \in C_w^-$ is in the transitive symmetric reflexive closure of C_w^+ on Q.*

Definition 3. *An* assignment *is a partial function $h : \mathbf{P} \to \mathbb{N}$. It* agrees with *a world w, written $w \multimap h$, if $\mathsf{dom}(f) = Q$ and (i) for all $q \in Q$: $f_w^0(q) \leq h(q) \leq f_w^1(q)$, (ii) $(p, q) \in C_w^+$ implies $h(p) = h(q)$ and (iii) $(p, q) \in C_w^-$ implies $h(p) \neq h(q)$.*

Definition 4. *For any register model $\mathcal{M} = (W, R, V)$, any $w \in W$ and any h such that $h \multimapinv w$ we define:*

$$\begin{array}{lll} \mathcal{M}, w, h \models \top & & \textit{always} \\ \mathcal{M}, w, h \models p & \textit{iff} & p \in P_w \\ \mathcal{M}, w, h \models p_1 = p_2 & \textit{iff} & h(p_1) = h(p_2) \\ \mathcal{M}, w, h \models p = N & \textit{iff} & h(p) = N \\ \mathcal{M}, w, h \models \neg\phi & \textit{iff} & \textit{not } \mathcal{M}, w, h \models \phi \\ \mathcal{M}, w, h \models \phi \wedge \psi & \textit{iff} & \mathcal{M}, w, h \models \phi \textit{ and } \mathcal{M}, w, h \models \psi \\ \mathcal{M}, w, h \models K_i\phi & \textit{iff} & (w, w') \in R_i \textit{ and } h' \multimapinv w' \\ & & \textit{imply } \mathcal{M}, w', h' \models \phi \\ \mathcal{M}, w, h \models G\phi & \textit{iff} & w' \in W \textit{ and } h' \multimapinv w' \\ & & \textit{imply } \mathcal{M}, w', h' \models \phi \\ \mathcal{M}, w, h \models \langle !p = E\rangle\phi & \textit{iff} & \mathcal{M}, w, h \models p = E \textit{ and} \\ & & \mathcal{M}^{p=E}, w, h \models \phi \\ \mathcal{M}, w, h \models \langle !p \neq E\rangle\phi & \textit{iff} & \mathcal{M}, w, h \models p \neq E \textit{ and} \\ & & \mathcal{M}^{p\neq E}, w, h \models \phi \\ \mathcal{M}, w, h \models \langle p \leftarrow^i N\rangle\phi & \textit{iff} & \mathcal{M}, w, h \vDash G\neg p \textit{ and} \\ & & \mathcal{M}^{p \leftarrow^i N}, w, h \cup \{(p, N)\} \models \phi \end{array}$$

where the new models $\mathcal{M}^{p=E}$, $\mathcal{M}^{p\neq E}$ and $\mathcal{M}^{p \leftarrow^i N}$ are given by a product update with appropriate action models [1].

We say that ϕ is true at a world w, written $\mathcal{M}, w \models \phi$BalMosSol98:tlopa iff $\forall h : w \multimap h \Rightarrow \mathcal{M}, w, h \models \phi$.

Theorem 5. *The following reduction schemes, together with appropriate axioms for knowledge and natural numbers are sound and complete for the class of all register models.*

P0) $\langle !p = E\rangle\top \leftrightarrow (p = E)$

P1) $\langle !p = E\rangle q \leftrightarrow (p = E \wedge q)$

P2) $\langle !p = E\rangle(q = E') \leftrightarrow (q = E')$

P3) $\langle !p = E\rangle\neg\phi \leftrightarrow (p = E \wedge \neg\langle !p = E\rangle\phi)$

P4) $\langle !p = E\rangle(\phi \wedge \psi) \leftrightarrow (\langle !p = E\rangle\phi \wedge \langle !p = E\rangle\psi)$

P5) $\langle !p = E\rangle K_i\phi \leftrightarrow (p = E \wedge K_i(p = E \to \langle !p = E\rangle\phi))$

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

N0) to N6) analogous for negative announcements.

R0) $\langle p \leftarrow^i N\rangle\top \leftrightarrow (G\neg p)$

R1) $\langle p \leftarrow^i N\rangle p \leftrightarrow (G\neg p)$

R2) $\langle p \leftarrow^i N\rangle q \leftrightarrow (G\neg p \wedge q)$ *where* $p \neq q$

R3a1) $\langle p \leftarrow^i N\rangle(p = N) \leftrightarrow (G\neg p)$

R3a1') $\langle p \leftarrow^i N\rangle(p = M) \leftrightarrow \neg\top$ *where* $M \neq N$

R3a2) $\langle p \leftarrow^i N\rangle(q = M) \leftrightarrow (G\neg p \wedge (q = M))$ *where* $p \neq q$

R3b1) $\langle p \leftarrow^i N\rangle(p = p) \leftrightarrow (G\neg p)$

R3b1') $\langle p \leftarrow^i N\rangle(p = q) \leftrightarrow (G\neg p \wedge (q = N))$ *where* $p \neq q$

R3b2) $\langle p \leftarrow^i N\rangle(q = p) \leftrightarrow (G\neg p \wedge (q = N))$ *where* $p \neq q$

R3b2') $\langle p \leftarrow^i N\rangle(q = r) \leftrightarrow (G\neg p \wedge (q = r))$ *where* $r \neq p \neq q$

R4) $\langle p \leftarrow^i N\rangle\neg\phi \leftrightarrow (G\neg p \wedge \neg\langle p \leftarrow^i N\rangle\phi)$

R5) $\langle p \leftarrow^i N\rangle(\phi \wedge \psi) \leftrightarrow (\langle p \leftarrow^i N\rangle\phi \wedge \langle p \leftarrow^i N\rangle\psi)$

R6) $\langle p \leftarrow^i N\rangle(K_i\phi) \leftrightarrow (G\neg p \wedge K_i(G\neg p \to \langle p \leftarrow^i N\rangle\phi))$

R7) $\langle p \leftarrow^i N\rangle(K_j\phi) \leftrightarrow (G\neg p \wedge K_j\phi)$ *where* $j \neq i$

R8) $\langle p \leftarrow^i N\rangle(G\phi) \leftrightarrow G(\langle p \leftarrow^i N\rangle\phi)$

3. APPLICATION

Register models encode larger Kripke models: Instead of many possible worlds they have few worlds with many agreeing assignments. To check a formula it often suffices to check some of them. This motivates a Monte Carlo method, picking a few h with $w \multimap h$ at random to test $\mathcal{M}, w \dot{\models} \phi$.

The presented register models can be used to analyze cryptographic protocols, if the language is extended with computation and directed communication [3]. For a detailed presentation of these ideas, see [4].

Definition 6. *The language for epistemic crypto logic consists of the following formulas, commands and expressions.*

$$\begin{array}{rcl} \phi & ::= & \top \mid p \mid L_i \mid p = E \mid \neg\phi \mid \phi \wedge \phi \mid K_i\phi \mid [C]\phi \\ & & \mid \mathbf{Prime} E \mid \mathbf{Coprime} E \\ C & ::= & p \leftarrow^i E \mid ?\phi \mid\ !p = E \mid\ !p \neq E \mid \mathbf{Open}\ i \mid \mathbf{Close}\ i \\ E & ::= & p \mid N \mid E{+}E \bmod E \mid E{\times}E \bmod E \mid E^E \bmod E \end{array}$$

Example 7. *The famous Diffie-Hellman Key Exchange [2]:*

$$\begin{array}{l} ?\ \mathbf{Prime}\ p\ ;\ ?\ g \in [1..p]\ ;\ ?\ \mathbf{Coprime}\ g\ (p-1)\ ; \\ a \leftarrow^{Alice} N\ ;\ A \leftarrow^{Alice} g^a \bmod p\ ; \\ \mathbf{Open}\ Bob\ ;\ !x = A\ ;\ \mathbf{Close}\ Bob\ ; \\ b \leftarrow^{Bob} M\ ;\ B \leftarrow^{Bob} g^b \bmod p\ ; \\ \mathbf{Open}\ Alice\ ;\ !y = B\ ;\ \mathbf{Close}\ Alice\ ; \\ k_a \leftarrow^{Alice} y^a \bmod p\ ;\ k_b \leftarrow^{Bob} x^b \bmod p\ ; \\ ?\ (k_a = k_b) \wedge (K_{Alice}k_a \wedge K_{Bob}k_b) \wedge (\neg K_{Eve}k_a \wedge \neg K_{Eve}k_b) \end{array}$$

Acknowledgments

Thanks to Alexandru Baltag, Johan van Benthem, Joshua Sack and Kaile Su and three anonymous referees.

REFERENCES

[1] J. v. Benthem, J. v. Eijck, and B. Kooi. Logics of communication and change. *Information and computation*, 204(11):1620–1662, 2006.

[2] W. Diffie and M. Hellman. New directions in cryptography. *Information Theory, IEEE Transactions on*, 22(6):644–654, 1976.

[3] H. v. Ditmarsch, A. Herzig, E. Lorini, and F. Schwarzentruber. Listen to me! Public announcements to agents that pay attention – or not. In *Proceedings of LORI*, 2013.

[4] M. Gattinger. Dynamic Epistemic Logic for Guessing Games and Cryptographic Protocols. Master's thesis, University of Amsterdam, 2014.