The Language of Social Software

Jan van Eijck
CWI, Amsterdam

Workshop on Logic and Social Interaction, Chennai, January 8, 2009

Abstract

Computer software is written in languages like C, Java or Haskell. In many cases social software is written in natural language. The talk will explore connections between the areas of natural language analysis and social software.

References:

Jan van Eijck/Rineke Verbrugge (eds)
Discourses on Games, Action and Social Software:
http://www.cwi.nl/~jve/nias/discourses/discourses.pdf

Jan van Eijck/Christina Unger
Computational semantics with Functional Programming
http://www.cwi.nl/~jve/cs/ http://www.cwi.nl/~jve/cs/cs.pdf

Jan van Eijck/Albert Visser
Dynamic Semantics
To appear in Stanford Encyclopedia of Philosophy http://plato.stanford.edu/

The Solomon Verdict: Rational Reconstruction [8]

Suppose the child is worth A to the real mother and B to the pretender. We can assume that A is much larger than B. The women make their bids in sealed envelopes.

Solomon makes the following announcement: "I will ask one of you if you are willing to give the child to the other. If the answer is yes, the case is settled. If not, I will ask the other. Again, if the answer is yes, the case is settled. If both of you refuse to give up the child, then I will have to sell it for what it is worth. I will toss a coin, and the one who gets the child will have to pay $\frac{A+B}{2}$, and the other pays a fine."

If the women act rationally, one of them will give up the child, which settles the case.

The fact that Solomon's announcement creates common knowledge is crucial.

Indian Version: an Akbar and Birbal story

Here is a story where Birbal acts exactly like Solomon.

In the Hindu version, Ramu and Shamu claimed ownership of the same mango tree, and decided to ask Birbal to settle the dispute.

Birbal's verdict: "Pick all the fruits from the tree and divide them equally. Then cut down the tree and divide the wood."

Ramu thought this was fair but Shamu was horrified, and Birbal declared Shamu the true owner.

Overview of Rest of Talk

- Analyzing the Discourse Situation in Natural Language Communication
- PDL as a logic of Knowledge and Common Knowledge
- Action Model Update
- Public Announcement, Message Passing and Common Knowledge
- Presupposition and Common Knowledge
- Changing the World
- Analysis of Yes/No Questions
- Analyzing Social Software Protocols
- Model Checking for Social Software Protocols

Social Software and Natural Language

Analysis of social software calls for natural language discourse analysis with a practical goal.

Solomon case: the verdict is given in natural language.

What does the discourse convey to the two mothers?

Natural language is:

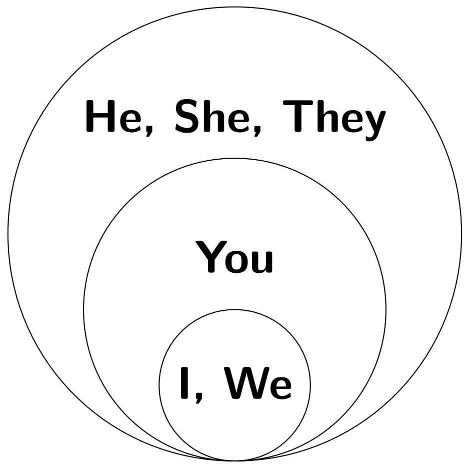
- a tool for creating (common) knowledge and changing (common) beliefs,
- a tool that employs common knowledge and common belief to establish communication.

Discourse Situation

First person: **I, We**. The speaker (or group represented by the speaker)

Second person: **You**. The audience

Third person: **He, She, They.** The outside world



Aim of discourse: create common knowledge between **Me** and **You**.

Knowledge and Common Knowledge

Fix a PDL style language for talking about epistemic plausibility. Assume p ranges over a set of basic propositions *Prop* and a over a set of agents *Ag*.

$$\begin{aligned} \phi &::= \top \mid p \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid [\pi]\phi \\ \pi &::= a \mid a^{\smallsmile} \mid ?\phi \mid \pi_1;\pi_2 \mid \pi_1 \cup \pi_2 \mid \pi^* \end{aligned}$$

Interpretation in the usual PDL manner, with $[\![\pi]\!]^{\mathbf{M}}$ giving the relation that interprets relational expression π in $\mathbf{M} = (W, P, V)$. P is the set of epistemic plausibilities $\xrightarrow{a}$.

$[\pi]\phi$ is true in world w of $\mathbf{M}$ if for all v with $(w, v) \in [\![\pi]\!]^{\mathbf{M}}$ it holds that ϕ is true in v.

knowledge $\sim_a$ abbreviates $(a \cup a^{\smallsmile})^*$.

common knowledge $\sim_{a,b}$ abbreviates $(\sim_a \cup \sim_b)^*$.

This logic is axiomatised by the standard PDL rules and axioms ([9, 7]) plus axioms that define the meanings of the relation names $a^{\smallsmile}$. The PDL rules and axioms are:

Modus ponens	and axioms for propositional logic
Modal generalisation	From $\vdash \phi$ infer $\vdash [\pi]\phi$

Normality	$\vdash [\pi](\phi \to \psi) \to ([\pi]\phi \to [\pi]\psi)$
Test	$\vdash [?\phi]\psi \leftrightarrow (\phi \to \psi)$
Sequence	$\vdash [\pi_1; \pi_2]\phi \leftrightarrow [\pi_1][\pi_2]\phi$
Choice	$\vdash [\pi_1 \cup \pi_2]\phi \leftrightarrow ([\pi_1]\phi \wedge [\pi_2]\phi)$
Mix	$\vdash [\pi^*]\phi \leftrightarrow (\phi \wedge [\pi][\pi^*]\phi)$
Induction	$\vdash (\phi \wedge [\pi^*](\phi \to [\pi]\phi)) \to [\pi^*]\phi$

The relation between the basic programs a and $a^{\smallsmile}$ is expressed by the standard modal axioms for converse:

$$\vdash \phi \to [a]\langle a^{\smallsmile}\rangle\phi \qquad \vdash \phi \to [a^{\smallsmile}]\langle a\rangle\phi$$

Action Model Update

Definition of update models $\mathbf{A}$ and of the update product operation $\otimes$ from Baltag, Moss, Solecki [1]. An action model is like an preference model, but with the valuation replaced by a precondition map $\mathbf{pre}$.

Updating a static model $\mathbf{M} = (W, P, V)$ with an action model $\mathbf{A} = (E, \mathbf{P}, \mathbf{pre})$ results in new static model $\mathbf{M} \otimes \mathbf{A} = (W', P', V')$, where the new worlds are pairs (w, e) with $w \in W$ and $e \in E$.

If the static model has a set of distinguished states W_0 and the action model a set of distinguished events E_0, then the distinguished worlds of $\mathbf{M} \otimes A$ are the (w, e) with $w \in W_0$ and $e \in E_0$.

Figure 1: Static model and update model

Figure 1 gives an example pair of a static model with an update action. The static model, on the left, pictures the result of a hidden coin toss, with three onlookers, Alice, Bob and Carol.

The update model represents a secret test to the effect that the toss is h. The result of the update is that the distinction mark on the $\overline{h}$ world has disappeared, without any of a, b, c being aware of the change.

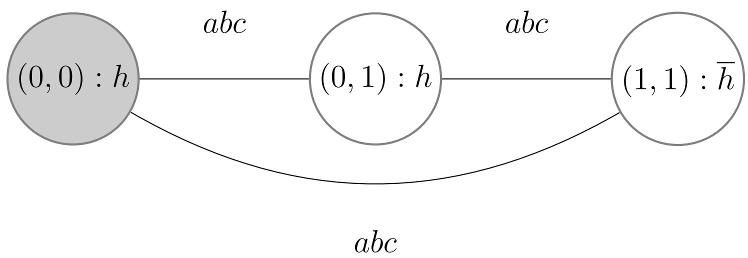


Figure 2: Result of the update in Figure 1.

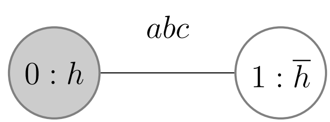


Figure 3: Bisimulation-minimal version of result of the update in Figure 1.

Adding Factual Change (and Belief Change)

Factual change was added to update models in LCC [2], by means of propositional substitutions.

A propositional binding is a map from proposition letters to formulas, represented by

$$\{p_1 \mapsto \phi_1, \ldots, p_n \mapsto \phi_n\}$$

where the p_k are all different, and where no ϕ_k is equal to p_k. It is assumed that each p that does not occur in a lefthand side of a binding is mapped to itself.

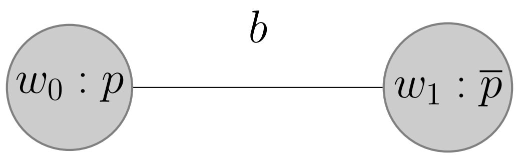
Belief change can be added in a similar manner, by means of relational substitutions.

A relational binding is a map from agents to program expressions, represented by

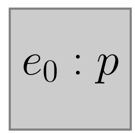
$$\{a_1 \mapsto \pi_1, \ldots, a_n \mapsto \pi_n\}$$

Public Announcement Creates Common Knowledge

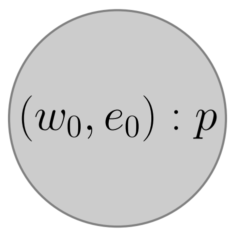
a knows whether p is true, b does not know:



Update action for public announcement of p.

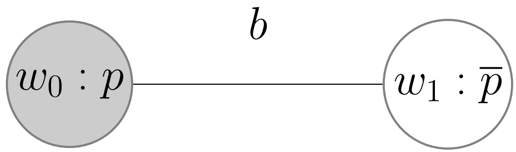


Update result:

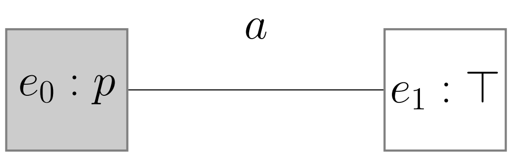


Message Exchange Cannot Create Common Knowledge

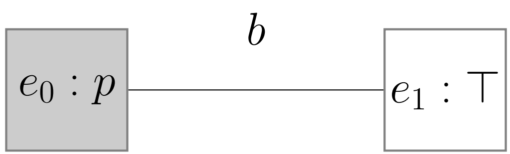
Two generals a, b. a will attack (p), but b does not know this:



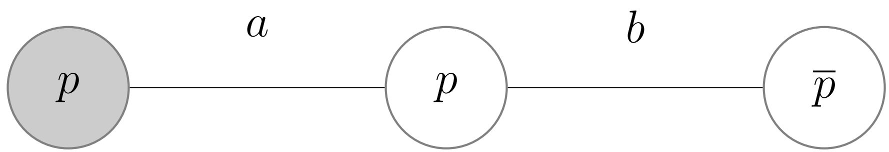
Update action for general a: send a message p.



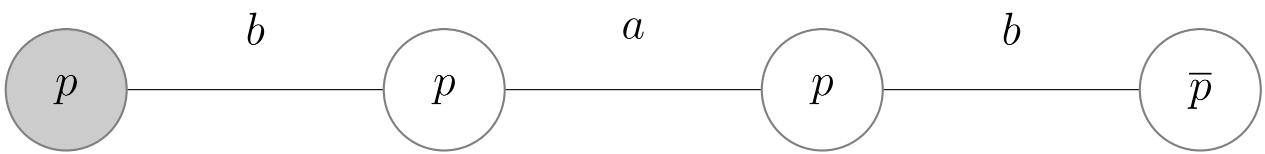
Update action for general b: send an acknowledgement of p:



Situation after first message from general a:



Situation after update by a followed by update by b:



And so on . . .

Co-presence Creates Common Knowledge

Example: cash withdrawal from a bank.

You withdraw a large amount of money from your bank account and have it paid out to you in cash by the cashier.

The cashier looks at you earnestly to make sure she has your full attention, and then she slowly counts out the banknotes for you: one thousand (counting ten notes), two thousand (counting another ten notes), three thousand (ten notes again), and four thousand (another ten notes).

This ritual creates common knowledge that forty banknotes of a hundred dollars were paid out to you.

Philosophical question: when money is paid out to you by an ATM, does this create common knowledge between you and the machine?

Presupposition

A presupposition of an utterance is an implicit assumption about the world or a background belief shared by speaker and hearer in a discourse.

"Shall we do it again?"

Presupposition: we have done it before.

"Jan is a bachelor."

Presupposition: 'Jan' refers to a male person. (True in the Netherlands and Poland, false in the United Kingdom.)

Second presupposition: 'Jan' refers to an adult.

So: 'bachelor' presupposes 'male' and 'adult', and conveys 'unmarried'.

Presupposition and Common Knowledge [4]

Extend the language with public announcements:

$[!\phi]\psi$ expresses that after public announcement of ϕ, ψ holds.

Formally:

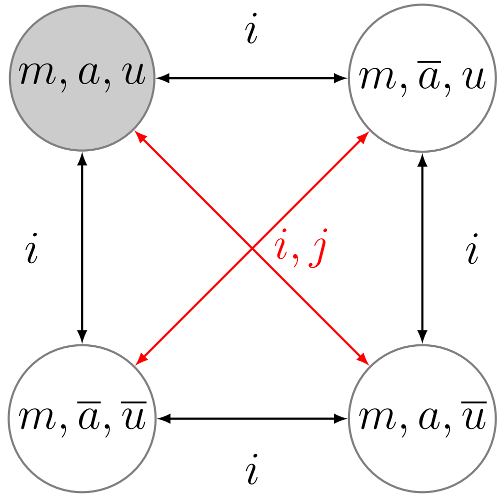
$M \models_w [!\phi]\psi$ iff ($M \models_w \phi$ implies $M \mid \phi \models_w \psi$).

Now consider the special case of an update of the form "it is common knowledge between i and j that ϕ".

Formally: $![\sim_{i,j}]\phi$.

- In case ϕ is already common knowledge, this update does not change the model.
- In case ϕ is not yet common knowledge, the update leads to a model without actual worlds.

Example



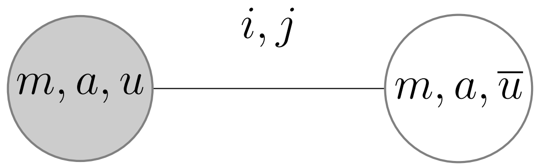
m for 'male', a for 'adult', u for 'unmarried'.

j does not know about u

i does not know about a, u.

$[\sim_{ij}]m$ holds, $[\sim_{ij}]a$ and $[\sim_{ij}]u$ do not hold.

Analysis of Presupposition in terms of Common Knowledge

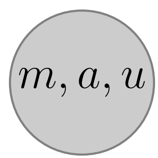


A presupposition is a piece of common knowledge between speaker and hearer in a discourse.

'bachelor' has presupposition 'male' and 'adult', and conveys information 'unmarried'.

$$[\sim_{ij}](m \wedge a) \wedge u$$

Update result:



Facts About Public Announcement of Common Knowledge

$$M \models_w [![\sim_{ij}]\phi]\psi \text{ iff } M \models_w [\sim_{ij}]\phi \rightarrow \psi.$$

Public announcement of common knowledge has the force of an implication.

$$M \models_w [!([\sim_{ij}]\phi \wedge \phi')]\psi \text{ iff } M \models_w [![\sim_{ij}]\phi][!\phi']\psi.$$

Putting a presupposition before an assertion has the same effect as lumping them together.

Presupposition Projection

Example: update without presupposition $!m$ (the statement **male**) followed by the update for **bachelor**).

$$\begin{aligned}
[!m][!(C(m \wedge a) \wedge u)]\chi &\leftrightarrow [!(m \wedge [!m](C(m \wedge a) \wedge u))]\chi \\
&\leftrightarrow [!(m \wedge [!m]Cm \wedge [!m]Ca \wedge [!m]u)]\chi \\
&\leftrightarrow [!(m \wedge [!m]Ca \wedge [!m]u)]\chi \\
&\leftrightarrow [!(m \wedge C(m, a) \wedge m \rightarrow u)]\chi \\
&\leftrightarrow [!(C(m, a) \wedge m \wedge u)]\chi
\end{aligned}$$

C for $[\sim_{ij}]$; $C(\phi, \psi)$ for $[!\phi][\sim_{ij}]\psi$.

So the presuppositional part of the combined statement is $C(m, a)$.

The assertional part is $m \wedge u$.

Presupposition Accommodation

Suppose p is common knowledge.

Then updating with statement $!(Cp \wedge q)$ has the same effect as updating with $!q$.

Suppose p is true in the actual world but not yet common knowledge.

Updating with $!(Cp \wedge q)$ wil lead to an inconsistent state

Updating with $!p$ followed by an update with $!(Cp \wedge q)$ will not.

Accommodation of the presupposition would consist of replacement of $!(Cp \wedge q)$ by $[!p][!(Cp \wedge q)]$.

By invoking the Gricean maxim 'be informative' one can explain why $[!p][!(Cp \wedge q)]$ is not appropriate in contexts where p is common knowledge.

Public Change

Extend the language with public change.

$[p := \phi]\psi$.

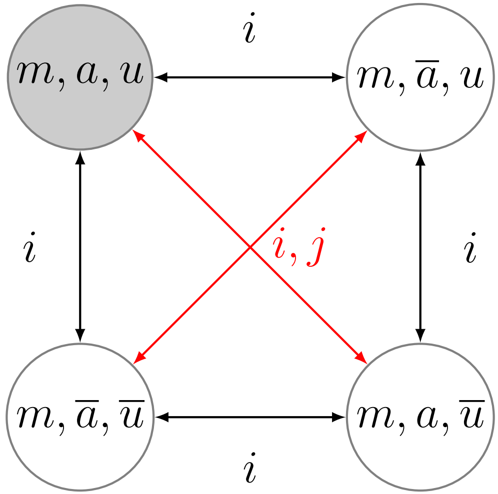
True in world w of M if ψ is true in world $w^{p:=[\![\phi]\!]_w}$ of $M^{p:=[\![\phi]\!]}$.

$p := \phi$ changes the model M to $M^{p:=[\![\phi]\!]}$.

Performative speech acts are examples:

- 'I call you Adam'
- 'I declare you man and wife'

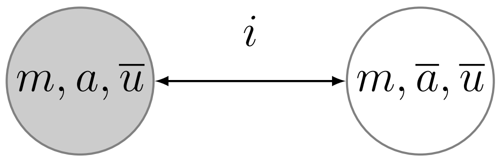
Marriage



Public change:

$u := \bot$

Result:



DEL Analysis of Yes/No Questions [5]

Let f be a propositional variable for question focus.

Analyse a Yes/No Question $\phi?$ as:

$$f := \phi$$

Analyze the answer 'yes' as:

$$f$$

Analyze the answer 'no' as:

$$\neg f$$

Questions and Appropriate Answers

Question: ‘Is Johnny married?’

Answer: ‘Johnny is not an adult.’

This answer is appropriate: updating with this answer makes ‘John is not married’ common knowledge. The update entails the answer ‘no’.

Question:

$$f := \phi$$

Answer:

$$\psi$$

This is appropriate if either updating with ψ has the effect that f becomes common knowledge, or updating with ψ has the effect that $\neg f$ becomes common knowledge.

Social Software Protocol Analysis with DEL

A group of 100 prisoners, all together in the prison dining area, are told that they will be all put in isolation cells and then will be interrogated one by one in a room containing a light with an on/off switch. The prisoners may communicate with one another by toggling the light-switch (and in no other way). The light is initially switched off. There is no fixed order of interrogation. Every day one prisoner will get interrogated. At any stage every prisoner will be interrogated again sometime.

When interrogated, a prisoner can either do nothing, or toggle the light-switch, or announce that all prisoners have been interrogated. If that announcement is true, the prisoners will (all) be set free, but if it is false, they will all be executed. Can the prisoners agree on a protocol that will set them free?

Protocol

Assume there are $n > 2$ prisoners.

The n prisoners appoint one among them as the counter.

All prisoners except the counter act as follows: the first time they enter the room when the light is off, they switch it on; on all next occasions, they do nothing.

The counter acts as follows: The first $n - 2$ times that the light is on when he enters the interrogation room, he turns it off. Then the next time he enters the room when the light is on, he (truthfully) announces that everybody has been interrogated.

Analysis

For simplicity, assume there are three prisoners $0, 1, 2$, with 0 acting as counter.

Let e_0, e_1, e_2 be the interrogation events of the three prisoners.

Let p express that the light is on.

For example: if the light is on and if event e_0 (interrogation of the counter) takes place, then afterwards the light is off, and the counter knows that it is off:

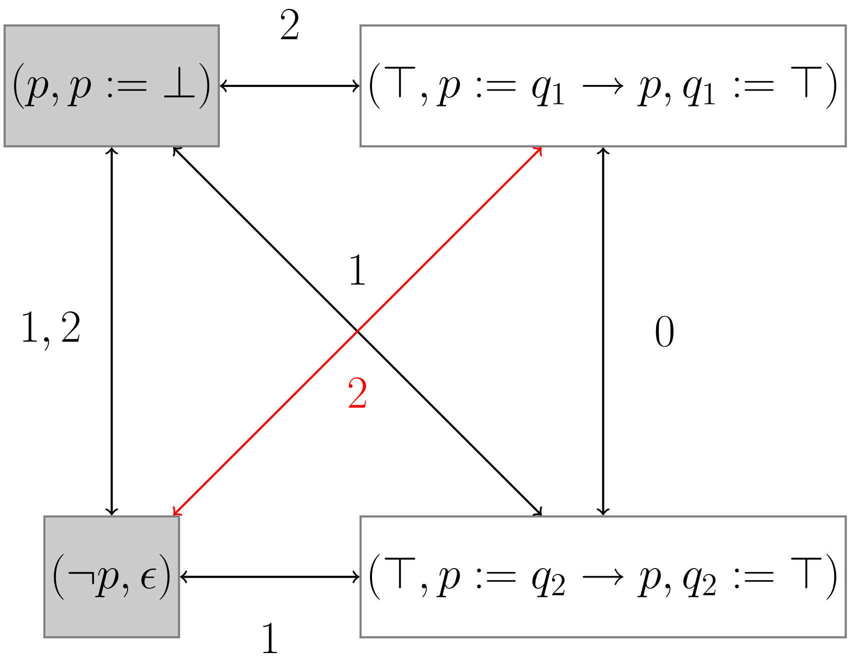
$p \longrightarrow [e_0]K_0\neg p$.

Let q_i, for $i = 1, 2$, express that prisoner i has been interrogated at least once. Then the following is true:

$[e_1]q_1$.

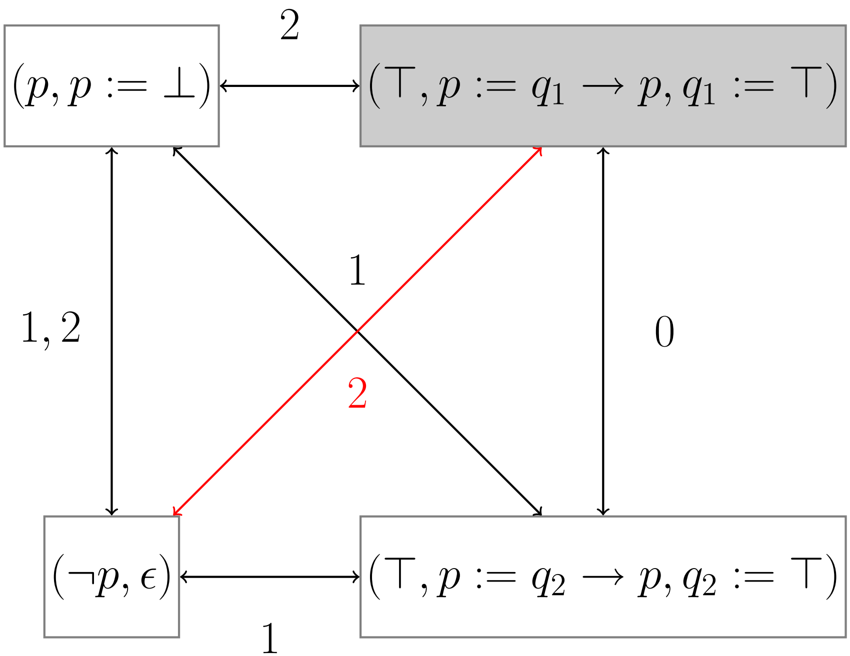
The Update Events: Counter Event

e_0:



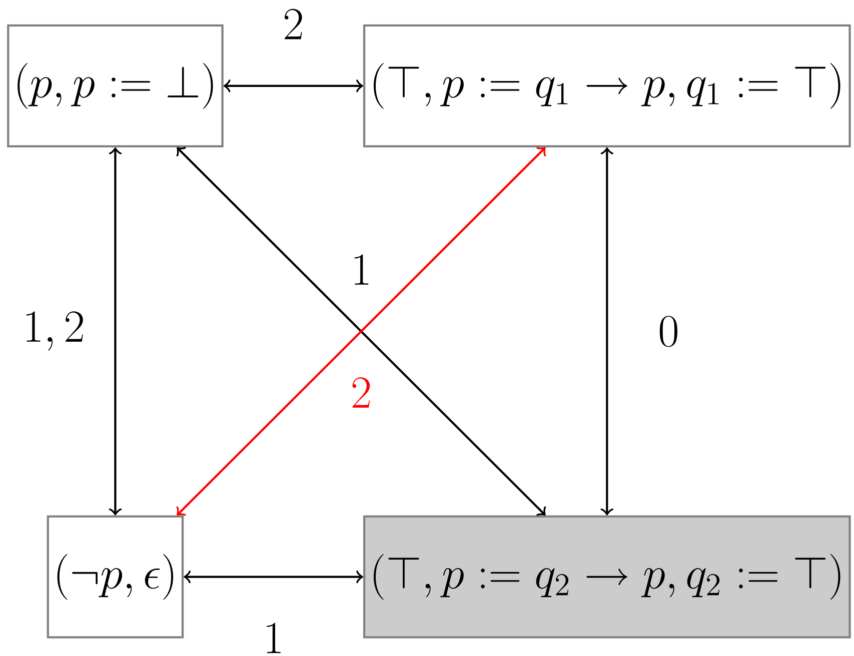
The Update Events: Event for Prisoner 1

e_1:



The Update Events: Event for Prisoner 2

e_2:



Agreeing on the Protocol

Why should the prisoners agree on this protocol? After all, it is a matter of life and death.

Because it is common knowledge that at some point in the future the counter will know that all have been interrogated.

Can we express this in DEL? No.

DEL + LTL

$$
\begin{aligned}
\phi &::= \top \mid p \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid [\pi]\phi \mid [e]\phi \mid \\
&\quad\; Ne \mid F\phi \mid G\phi \mid P\phi \mid H\phi \\
\pi &::= a \mid a^{\smile} \mid ?\phi \mid \pi_1;\pi_2 \mid \pi_1 \cup \pi_2 \mid \pi^* \\
a &::= 0 \mid 1 \mid 2 \\
e &::= e_0 \mid e_1 \mid e_2
\end{aligned}
$$

The intended semantics of $[e]\phi$ is the DEL semantics: either update with event e fails, or in the updated model ϕ holds.

The intended semantics of Ne is that the next event is e.

The meanings of F, P, G, H are the usual ones from linear time logic (LTL).

Interpretation

With respect to infinite sequences of events.

If σ is such a sequence and if n is a positive natural number, then we use σ_n is the n-th event of the sequence.

σ looks like $\sigma_1, \sigma_2, \ldots$

Then $M_{\sigma,n}$ is the model which results from doing updates $\sigma_1, \ldots, \sigma_n$ on the initial model (where the light is off and everyone knows that).

Since all updates are functional, this is well-defined.

Example:

$e_0, e_1, e_2, e_0, e_1, e_2, e_0, e_1, e_2, e_0, e_1, e_2, \ldots$

Truth Definition $(\sigma, n) \models \phi$

$(\sigma, n) \models p$ if p is true in $M_{\sigma,n}$.

Booleans, epistemic operations as usual, using $M_{\sigma,n}$ for (σ, n).

$(\sigma, n) \models Ne$ if $\sigma_{n+1} = e$ (the next event in the sequence σ equals e),

$(\sigma, n) \models F\phi$ if for some $m > n$, $(\sigma, m) \models \phi$.

$(\sigma, n) \models G\phi$ if for all $m > n$, $(\sigma, m) \models \phi$.

$(\sigma, n) \models P\phi$ if for some $m < n$, $(\sigma, m) \models \phi$.

$(\sigma, n) \models H\phi$ if for all $m < n$, $(\sigma, m) \models \phi$.

This should be all familiar from LTL.

Protocol Properties

Fairness of an interrogation sequence:

$$G(FNe_0 \wedge FNe_1 \wedge FNe_2).$$

Knowledge of 0 that prisoners 1 and 2 have been interrogated:

$$[\sim_0](PNe_1 \wedge PNe_2).$$

Correctness of the protocol:

$$G(FNe_0 \wedge FNe_1 \wedge FNe_2) \to F[\sim_0](PNe_1 \wedge PNe_2).$$

Common knowledge of correctness of the protocol:

$$[\sim_{012}](G(FNe_0 \wedge FNe_1 \wedge FNe_2) \to F[\sim_0](PNe_1 \wedge PNe_2)).$$

If there is time . . .

Epistemic Model Checking with DEMO [3].

Conclusions

- Common Knowledge and Common Belief Central Notions in Discourse Analysis and in Social Software
- Program: Analyzing Discourse as sequences of public announcements
- Program: Analyzing Presupposition Projection and Accommodation in terms of common knowledge
- Analyzing yes/no questions as public change of focus, Analyzing appropriate answers in terms of 'same update effect'
  Program: extend this to a full semantic/pragmatic theory of questions and answers.
- Program: Analyzing social software protocols in DEL + LTL, and develop model checking tools for this.

T·L·G
Texts in Logic and Games
Volume 5

Discourses on Social Software

EDITED BY
JAN VAN EIJCK AND RINEKE VERBRUGGE

AMSTERDAM UNIVERSITY PRESS

References

[1] A. Baltag, L.S. Moss, and S. Solecki. The logic of public announcements, common knowledge, and private suspicions. In I. Bilboa, editor, Proceedings of TARK'98, pages 43–56, 1998.

[2] J. van Benthem, J. van Eijck, and B. Kooi. Logics of communication and change. Information and Computation, 204(11):1620–1662, 2006.

[3] Jan van Eijck. DEMO — a demo of epistemic modelling. In Johan van Benthem, Dov Gabbay, and Benedikt Löwe, editors, Interactive Logic — Proceedings of the 7th Augustus de Morgan Workshop, number 1 in Texts in Logic and Games, pages 305–363. Amsterdam University Press, 2007.

[4] Jan van Eijck and Christina Unger. The epistemics of presupposition projection. In Maria Aloni, Paul Dekker, and Floris Roelofsen,

editors, Proceedings of the Sixteenth Amsterdam Colloquium, December 17–19, 2007, pages 235–240, Amsterdam, December 2007. ILLC.

[5] J. Groenendijk and M. Stokhof. Studies on the Semantics of Questions and the Pragmatics of Answers. PhD thesis, University of Amsterdam, 1984.

[6] John Maynard Keynes. The General Theory of Employment, Interest and Money. Macmillan and Cambridge University Press, 1936. Full text available on the internet at http://www.marxists.org/reference/subject/economics/keynes/general-theory/.

[7] D. Kozen and R. Parikh. An elementary proof of the completeness of PDL. Theoretical Computer Science, 14:113–118, 1981.

[8] J. Moore. Implementation, contracts, and renegotiation in environments with complete information. In J.-J. Laffont, editor, Advances

in Economic Theory — 6th World Congress, volume I, Cambridge, 1992. Cambridge University Press.

[9] K. Segerberg. A completeness theorem in the modal logic of programs. In T. Traczyck, editor, Universal Algebra and Applications, pages 36–46. Polish Science Publications, 1982.