# Propositional Dynamic Logic as a Logic of Belief Revision

Jan van Eijck and Yanjing Wang

Center for Mathematics and Computer Science (CWI)
Kruislaan 413 1098 SJ Amsterdam, The Netherlands
{jve,y.wang}@cwi.nl

**Abstract.** This paper shows how propositional dynamic logic (PDL) can be interpreted as a logic for multi-agent belief revision. For that we revise and extend the logic of communication and change (LCC) of [9]. Like LCC, our logic uses PDL as a base epistemic language. Unlike LCC, we start out from agent plausibilities, add their converses, and build knowledge and belief operators from these with the PDL constructs. We extend the update mechanism of LCC to an update mechanism that handles belief change as relation substitution, and we show that the update part of this logic is more expressive than either that of LCC or that of doxastic/epistemic PDL with a belief change modality. It is shown that the properties of knowledge and belief are preserved under any update, and that the logic is complete.

**Keywords:** PDL, epistemic dynamic logic, belief revision, knowledge update.

## 1 Introduction

Proposals for treating belief revision in the style of dynamic epistemic logic (see Gerbrandy [15], van Ditmarsch [12], van Benthem [6,10], and Baltag, Moss and coworkers [3,1,2], or the textbook treatment in [13]) were made in [8] and [7], where it is suggested that belief revision should be treated as relation substitution. This is different from the standard action product update from [3], and it suggests that the proper relation between these two update styles should be investigated.

We propose a new version of action product update that integrates belief revision as relation substitution with belief update by means of action product. We show that this allows to express updates that cannot be expressed with action product only or with relation substitution only.

We graft this new update mechanism on a base logic that can express knowledge, safe belief, conditional belief, and plain belief, and we show that the proper relations between these concepts are preserved under any update. The completeness of our logic is also provided.

Our main source of inspiration is the logic of communication and change (LCC) from [9]. This system has the flaw that updates with non-S5 action models may

W. Hodges and R. de Queiroz (Eds.): WoLLIC 2008, LNAI 5110, pp. 136–148, 2008.

destroy knowledge or belief. If one interprets the basic relations as knowledge relations, then updating with a lie will destroy the S5 character of knowledge; similarly, if one interprets the basic relations as belief, the relational properties of belief can be destroyed by malicious updates. Our redesign does not impose any relational conditions on the basic relations, so this problem is avoided. Our completeness proof is an adaptation from the completeness proof for LCC. The treatment of conditional belief derives from [11]. Our work can be seen as a proposal for integrating belief revision by means of relation substitution, as proposed in [7] with belief and knowledge update in the style of [3].

## 2 PDL as a Belief Revision Logic

A preference model $\mathbf{M}$ for set of agents *Ag* and set of basic propositions *Prop* is a tuple $(W, P, V)$ where $W$ is a non-empty set of worlds, $P$ is a function that maps each agent $a$ to a relation $P_a$ (the preference relation for $a$, with $wP_aw'$ meaning that $w'$ is at least as good as $w$), and $V$ is a map from $W$ to $\mathcal{P}(Prop)$ (a map that assigns to each world a *Prop*-valuation). A distinctive preference model is a pair consisting of a preference model and a set of distinctive states in that model. The intuitive idea is that the actual world is constrained to be among the distinctive worlds. This information is typically not available to the agents, for an agent's knowledge about what is actual and what is not is encoded in her $P_a$ relation (see below).

There are no conditions at all on the $P_a$. Appropriate conditions will be imposed by constructing the operators for belief and knowledge by means of PDL operations.

We fix a PDL style language for talking about preference (or: plausibility). Assume $p$ ranges over *Prop* and $a$ over *Ag*.

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

We use $PROG$ for the set of program expressions (expressions of the form $\pi$) of this language.

This is to be interpreted in the usual PDL manner, with $[\![\pi]\!]^{\mathbf{M}}$ giving the relation that interprets relational expression $\pi$ in $\mathbf{M} = (W, P, V)$. $[\pi]\phi$ is true in world $w$ of $\mathbf{M}$ if for all $v$ with $(w, v) \in [\![\pi]\!]^{\mathbf{M}}$ it holds that $\phi$ is true in $v$. We adopt the usual abbreviations of $\bot$, $\phi \vee \psi$, $\phi \rightarrow \psi$, $\phi \leftrightarrow \psi$ and $\langle\pi\rangle\phi$.

The following additional abbreviations allow us to express knowledge, safe belief, conditional belief and plain belief:

**Knowledge.** $\sim_a$ abbreviates $(a \cup a\check{})^*$.
**Safe belief.** $\geq_a$ abbreviates $a^*$.
**Conditional belief.** $[\rightarrow_a^\phi]\psi$ abbreviates $\langle\sim_a\rangle\phi \rightarrow \langle\sim_a\rangle(\phi \wedge [\geq_a](\phi \rightarrow \psi))$.
**Plain belief.** $[\rightarrow_a]\phi$ abbreviates $[\rightarrow_a^\top]\phi$. (note: it follows that $[\rightarrow_a]\phi$ is equivalent to $\langle\sim_a\rangle[\geq_a]\phi$).

We will occasionally use $\leq_a$ for the converse of $\geq_a$.

Safe belief is belief that persists under revision with true information (see Stalnaker [20]). The definition of $[\to_a^\phi]\psi$ (conditional belief for $a$, with condition $\phi$) is from Boutillier [11] This definition, also used in [5], states that conditional to $\phi$, $a$ believes in $\psi$ if either there are no accessible $\phi$ worlds, or there is an accessible $\phi$ world in which the belief in $\phi \to \psi$ is safe. The definition of $[\to_a^\phi]\psi$ matches the well-known accessibility relations $\to_a^P$ for each subset $P$ of the domain, given by:

$$\to_a^P := \{(x, y) \mid x \sim_a y \wedge y \in \text{MIN}_{\leq_a} P\},$$

where $\text{MIN}_{\leq_a} P$, the set of minimal elements of $P$ under $\leq_a$, is defined as

$$\{s \in P : \forall s' \in P(s' \leq_a s \Rightarrow s \leq_a s')\}.$$

This logic is axiomatised by the standard PDL rules and axioms ([19,18]) plus axioms that define the meanings of the converses $a\check{}$ of basic relations $a$. 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 $a\check{}$ is expressed by the standard modal axioms for converse:

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

Any preference relation $P_a$ can be turned into a pre-order by taking its reflexive transitive closure $P_a{}^*$. So our abbreviation introduces the $\geq_a$ as names for these pre-orders. The knowledge abbreviation introduces the $\sim_a$ as names for the equivalences given by $(P_a \cup P_a\check{})^*$. If the $P_a$ are well-founded, $\text{MIN}_{\leq_a} P$ will be non-empty for non-empty $P$. Wellfoundedness of $P_a$ is the requirement that there is no infinite sequence of different $w_1, w_2, \ldots$ with $\ldots P_a w_2 P_a w_1$. Fortunately, we do not have to worry about this relational property, for the canonical model construction for PDL yields finite models, and each relation on a finite model is well-founded.

This yields a very expressive complete and decidable PDL logic for belief revision, to which we can add mechanisms for belief update and for belief change.

**Theorem 1.** *The above system of belief revision PDL is complete for preference models. Since the canonical model construction for PDL yields finite models, it is also decidable.*

Knowledge is S5 (equivalence), safe belief is S4 (reflexive and transitive), plain belief is KD45 (serial, transitive and euclidean). Note that the following is valid:

$$\langle \sim_a \rangle [\geq_a] \phi \rightarrow [\sim_a] \langle \geq_a \rangle \langle \sim_a \rangle [\geq_a] \phi$$

This shows that plain belief is euclidean.

## 3 Action Model Update

We give the definition of action models **A** and of the update product operation $\otimes$ from Baltag, Moss, Solecki [3]. An action model is like a preference model for *Ag*, with the difference that the worlds are now called *actions* or *events*, and that the valuation has been replaced by a map **pre** that assigns to each event $e$ a formula of the language called the *precondition* of $e$. From now on we call the preference models *static models*.

Updating a static model $\mathbf{M} = (W, P, V)$ with an action model $\mathbf{A} = (E, \mathbf{P}, \mathbf{pre})$ succeeds if the set

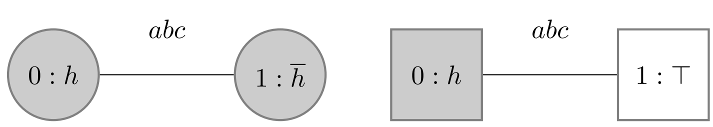
$$\{(w, e) \mid w \in W, e \in E, \mathbf{M}, w \models \mathbf{pre}(e)\}$$

is non-empty. The update result is a new static model $\mathbf{M} \otimes \mathbf{A} = (W', P', V')$ with

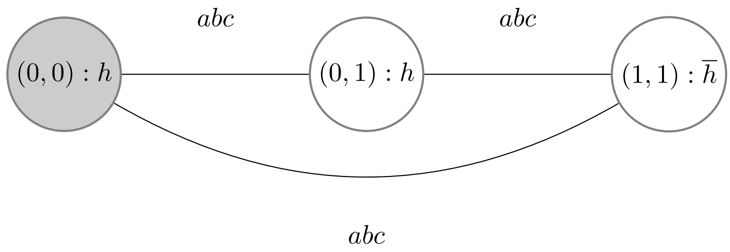
- $W' = \{(w, e) \mid w \in W, e \in E, \mathbf{M}, w \models \mathbf{pre}(e)\}$,
- $P'_a$ is given by $\{(w, e), (v, f)) \mid (w, v) \in P_a, (e, f) \in \mathbf{P}_a\}$,
- $V'(w, e) = V(w)$.

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

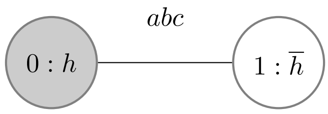
Below is an example pair of a static model with an update action. The static model, on the left, pictures the result of a hidden coin toss, with three onlookers, Alice, Bob and Carol. The model has two distinctive worlds, marked in grey; $h$ in a world means that the valuation makes $h$ true, $\overline{h}$ in a world means that the valuation makes $h$ false in that world. The $P_a$ relations for the agents are assumed to be equivalences; reflexive loops for $a, b, c$ at each world are omitted from the picture.




The action model represents a secret test whether the result of the toss is $h$. The distinctive event of the update is marked grey. The $P_i$ relations are drawn, for three agents $a, b, c$. The result of the update is shown here:




This result can be reduced to the bisimilar model below:




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.

## 4 Adding Factual Change and Belief Change

Factual change was already added to update models in LCC. We will now also add belief change. Let an action model with both changes be a quintuple.

$$A = (E, \mathbf{P}, \mathbf{pre}, \mathbf{Sub}, \mathbf{SUB})$$

where $E, \mathbf{P}, \mathbf{pre}$ are as before, **Sub** is a function that assigns a propositional binding to each $e \in E$, and **SUB** is a function that assigns a relational binding to each $e \in E$. A propositional substitution is a map from proposition letters to formulas, represented by a finite set of bindings.

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

where the $p_k$ are all different, and where no $\phi_k$ is equal to $p_k$. It is assumed that each $p$ that does not occur in a left-hand side of a binding is mapped to itself.

Similarly, a relational substitution is a map from agents to program expressions, represented by a finite set.

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

where the $a_j$ are agents, all different, and where the $\pi_j$ are program expressions from the PDL language. It is assumed that each $a$ that does not occur in the left-hand side of a binding is mapped to $a$. Use $\epsilon$ for the identity propositional or relational substitution.

**Definition 1 (Update execution).** *The update execution of static model* $\mathbf{M} = (W, P, V)$ *with action model* $\mathbf{A} = (E, \mathbf{P}, \mathbf{pre}, \mathbf{Sub}, \mathbf{SUB})$ *is a tuple:* $\mathbf{M} \circledast \mathbf{A} = (W', P', V')$ *where:*

- $W' = \{(w, e) \mid \mathbf{M}, w \vDash \mathbf{pre}(e)\}$.
- $P'_a$ *is given by*

$$\{((w_1, e_1), (w_2, e_2)) \mid \text{there is a } \mathbf{SUB}(e_1)(a) \text{ path from } (w_1, e_1) \text{ to } (w_2, e_2) \text{ in } \mathbf{M} \otimes \mathbf{A}\}.$$

- $V'(p) = \{(w, e) \in W' \mid \mathbf{M}, w \vDash \mathbf{Sub}(e)(p)\}$.

Note: the definition of $P'_a$ refers to paths in the old style update product.

Consider the suggestive upgrade $\sharp_a\phi$ discussed in Van Benthem and Liu [8] as a relation changer (*uniform* relational substitution):

$$\sharp_a\phi =_{\text{def}} \ ?\phi; a; ?\phi \ \cup ?\neg\phi; a; ?\neg\phi \ \cup ?\neg\phi; a; ?\phi.$$

This models a kind of belief change where preference links from $\phi$ worlds to $\neg\phi$ worlds for agent $a$ get deleted. It can be modelled as the following example of public belief change.

*Example 1 (Public Belief Change).* Action model

$$G = (\{e\}, \mathbf{P}, \mathbf{pre}, \mathbf{Sub}, \mathbf{SUB})$$

where:

- For all the $i \in Ag$, $\mathbf{P}_i = \{(e, e)\}$.
- $\mathbf{pre}(e) = \top$.
- $\mathbf{Sub}(e) = \epsilon$.
- $\mathbf{SUB}(e) = \{a \mapsto \ \sharp_a\phi, b \mapsto \ \sharp_b\phi\}$.

Note that our action model and its update execution implement the *point-wise* relation substitutions which is more powerful than merely upgrading the relations *uniformly* everywhere in the model, as the following example shows:

*Example 2 (Non-public Belief Change).* Action model

$$G' = (\{e_0, e_1\}, \mathbf{P}, \mathbf{pre}, \mathbf{Sub}, \mathbf{SUB})$$

where:

- For all $i \in Ag$, if $i \neq b$ then $\mathbf{P}_i = \{(e_0, e_0), (e_1, e_1)\}$, $\mathbf{P}_b = \{(e_0, e_0), (e_1, e_1), (e_0, e_1), (e_1, e_0)\}$
- $\mathbf{pre}(e_0) = \mathbf{pre}(e_1) = \top$.
- $\mathbf{Sub}(e_0) = \mathbf{Sub}(e_1) = \epsilon$.
- $\mathbf{SUB}(e_0) = \{a \mapsto \ \sharp_a\phi\}$, $\mathbf{SUB}(e_1) = \epsilon$.

Assume $e_0$ is the actual event.

This changes the belief of $a$ while $b$ remains unaware of the change.

Let PDL$^+$ be the result of adding modalities of the form $[\mathbf{A}, e]\phi$ to PDL, with the following interpretation clause:

$$\mathbf{M}, w \models [\mathbf{A}, e]\phi \ \text{ iff } \ \mathbf{M}, w \models \mathbf{pre}(e) \text{ implies } \mathbf{M} \circledast \mathbf{A}, (w, e) \models \phi.$$

**Theorem 2 (Soundness and Completeness for PDL$^+$).** $\vDash \phi$ *iff* $\vdash \phi$.

*Proof.* Completeness can be proved by a patch of the LCC completeness proof in [9] where the action modalities are pushed through program modalities by program transformations. See the first Appendix.

## 5 Expressivity of Action Update with Changes

Although $\text{PDL}^+$ reduces to PDL, just like LCC, the new action update mechanism (the model transformation part) is more *expressive* than classic product update and product update with factual changes, as we will show in this section. Call a function on epistemic models that is invariant for bisimulation a model transformer. Then each update can be viewed as a model transformer, and a set of model transformers corresponds to an update mechanism. If $U$ is an update mechanism, let $\text{Tr}(U)$ be its set of model transformers.

**Definition 2.** *Update mechanism $U_1$ is* less expressive *than update mechanism $U_2$ if $Tr(U_1) \subset Tr(U_2)$.*

First note that the classical product update (with factual changes) has the *eliminative* nature for relational changing: according to the definition, the relations in the updated model must come from relations in the static model. For example, it is not possible, by product update, to introduce a relational link for agent $a$ to a static model where the $a$ relation was empty. However, we can easily do this with an uniform relation substitution $a \mapsto ?\top$. Thus we have:

**Proposition 1.** *Relational substitution can express updates that cannot be expressed with action product update(with factual changes) alone, so relational substitution is not less expressive than action product update.*

On the other hand, relational substitution alone cannot add worlds into a static model, while the classical product update mechanism can copy sets of worlds. Therefore it is not hard to see:

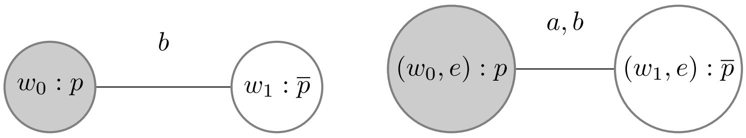
**Proposition 2.** *Action product update can express updates that cannot be expressed with relational substitution alone, so action product update is not less expressive than relational substitution.*

Our action update with both relational and factual changes combines the power of product update and propositional/relational substitutions. Thus according to Propositions 1 and 2, it is more expressive than relational eliminative product update with factual changes in LCC, and more expressive than propositional/relation changing substitution *simpliciter*. Moreover, we can prove a even stronger result for the case of $S5$ updates.

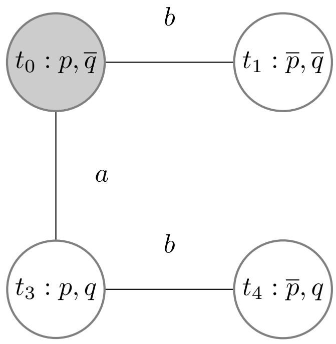
**Theorem 3.** *In the class of $S5$ model transformers, action product update with factual changes is less expressive than action update with both factual and relational changes.*

*Proof.* Let $\mathbf{A}$ be the action model $(\{e\}, \mathbf{P}, \mathbf{pre}, \mathbf{Sub}, \mathbf{SUB})$ where $\mathbf{P}_a = \{(e, e)\} = \mathbf{P}_b; \mathbf{pre}(e) = \top; \mathbf{Sub}(e) = \epsilon; \mathbf{SUB}(e) = \{a \mapsto b\}$. It is easy to see that this action model will change the relation $a$ as $b$ uniformly while keeping the updated model being still $S5$, if the static model is indeed $S5$. We now show that it does not have a corresponding action model in LCC style (only factual changes) which can give the bisimilar updated result for every static model.

First consider the following static $S5$ model $\mathbf{M}$ (left) and its updated model $\mathbf{M} \circledast \mathbf{A}$(right) (reflexive loops are omitted):




For a contradiction, suppose there is a LCC action model $\mathbf{A'}$ with distinctive event $e'$ such that $M_1 \circledast \mathbf{A}, (w_0, e) \underline{\leftrightarrow} \mathbf{M} \otimes \mathbf{A'}, (w_0, e')$. Then according to the definition of bisimulation, there must be an $a-$link from $(w_0, e')$ to a $\overline{p}$ world $(s, e'')$ in $\mathbf{M} \otimes \mathbf{A'}$. According to the definition of $\otimes$, $(w_0, s) \in p_a$ in $\mathbf{M}$ and $(e', e'') \in \mathbf{P}_a$ in $\mathbf{A'}$. Thus $s = w_0$ and $\mathbf{M}, w_0 \vDash pre(e') \wedge pre(e'')$. Let us consider the following $S5$ model $\mathbf{M'}$ which consists of two copies of $\mathbf{M}$ with an $a-$link in between:




where $q$ does not show up in $pre(e')$ and $pre(e'')$. Thus it is not hard to see that $pre(e') \wedge pre(e')$ holds on $t_0$ and $t_3$. Then $\mathbf{M'} \otimes \mathbf{A'}, (t_0, e')$ must has an $a$ link from a $\overline{q}$ world $(t_0, e')$ to a $q$ world $(t_3, e'')$, while in $\mathbf{M'} \circledast \mathbf{A}, (t_0, e)$ there is no such link. Thus $\mathbf{M'} \circledast \mathbf{A}, (t_0, e)$ and $\mathbf{M'} \otimes \mathbf{A'}, (t_0, e')$ are not bisimilar. Contradiction.

An example illustrating the use of the new belief revision update mechanism is worked out in the second Appendix. This example also shows the difference in expressive power for the achievement of common knowledge between knowledge update and belief revision.

## 6 Future Work

Several update mechanisms for dynamic epistemic logic have been proposed in the literature. A very expressive one is the action-priority upgrade proposed in [4,5]. Comparing the expressiveness of our update with factual and relation change with that of their mechanism is future work.

The new update mechanism proposed above is grafted on a doxastic/epistemic logic that does not impose any conditions on the basic preference relations. Thus, any update will result in a proper epistemic model. This situation changes as soon as one imposes further conditions. E.g., if the basic preferences are assumed to be locally connected, then one should restrict the class of update models to those that preserve this constraint. For each reasonable constraint, there is a corresponding class of model transformers that preserve this constraint. Finding syntactic characterizations of these classes is future work.

We are interested in **model checking** with doxastic/epistemic PDL and updates/upgrades in the new style, and we are currently investigating its complexity. We intend to use the logic, and the new update/upgrade mechanism, in the next incarnation of the epistemic model checker DEMO [14].

**Acknowledgments.** We have profited from discussions with Johan van Benthem, Hans van Ditmarsch and Barteld Kooi, and we thank two anonymous referees for their comments. The second author is supported by the Dutch Organisation for Scientic Research (NWO) under research grant no. 612.000.528 (VEMPS).

## References

1. Baltag, A.: A logic for suspicious players: epistemic action and belief-updates in games. Bulletin of Economic Research 54(1), 1–45 (2002)
2. Baltag, A., Moss, L.S.: Logics for epistemic programs. Synthese 139(2), 165–224 (2004)
3. Baltag, A., Moss, L.S., Solecki, S.: The logic of public announcements, common knowledge, and private suspicions. In: Bilboa, I. (ed.) Proceedings of TARK 1998, pp. 43–56 (1998)
4. Baltag, A., Smets, S.: Conditional doxastic models: A qualitative approach to dynamic belief revision. Electronic Notes in Theoretical Computer Science (ENTCS) 165, 5–21 (2006)
5. Baltag, A., Smets, S.: A qualitative theory of dynamic interactive belief revision. In: Bonanno, G., van der Hoek, W., Wooldridge, M. (eds.) Texts in Logic and Games. Amsterdam University Press (to appear, 2008)
6. van Benthem, J.: Language, logic, and communication. In: van Benthem, J., Dekker, P., van Eijck, J., de Rijke, M., Venema, Y. (eds.) Logic in Action, pp. 7–25. ILLC (2001)
7. van Benthem, J.: Dynamic logic for belief revision. Journal of Applied Non-Classical Logics 2, 129–155 (2007)
8. van Benthem, J., Liu, F.: Dynamic logic of preference upgrade. Journal of Applied Non-Classical Logics 14(2) (2004)
9. van Benthem, J., van Eijck, J., Kooi, B.: Logics of communication and change. Information and Computation 204(11), 1620–1662 (2006)
10. van Benthem, J.: One is a lonely number: on the logic of communication. Technical Report PP-2002-27, ILLC, Amsterdam (2002)
11. Boutilier, C.: Toward a logic of qualitative decision theory. In: Doyle, J., Sandewall, E., Torasso, P. (eds.) Proceedings of the 4th International Conference on Principle of Knowledge Representation and Reasoning (KR 1994), pp. 75–86. Morgan Kaufmann, San Francisco (1994)
12. van Ditmarsch, H.: Knowledge Games. PhD thesis, ILLC, Amsterdam (2000)

13. van Ditmarsch, H.P., van der Hoek, W., Kooi, B.: Dynamic Epistemic Logic. Synthese Library, vol. 337. Springer, Heidelberg (2006)
14. van Eijck, J.: DEMO — a demo of epistemic modelling. In: van Benthem, J., Gabbay, D., Löwe, B. (eds.) Interactive Logic — Proceedings of the 7th Augustus de Morgan Workshop, number 1 in Texts in Logic and Games, pp. 305–363. Amsterdam University Press (2007)
15. Gerbrandy, J.: Bisimulations on planet Kripke. PhD thesis, ILLC (1999)
16. Gray, J.: Notes on database operating systems. In: Bayer, R., Graham, R.M., Seegmuller, G. (eds.) Operating Systems: an advanced course. LNCS, vol. 66, Springer, Berlin (1978)
17. Halpern, J.Y., Moses, Y.O.: Knowledge and common knowledge in a distributed environment. In: Proceedings 3rd ACVM Symposium on Distributed Computing, pp. 50–68 (1984)
18. Kozen, D., Parikh, R.: An elementary proof of the completeness of PDL. Theoretical Computer Science 14, 113–118 (1981)
19. Segerberg, K.: A completeness theorem in the modal logic of programs. In: Traczyck, T. (ed.) Universal Algebra and Applications, pp. 36–46. Polish Science Publications (1982)
20. Stalnaker, R.C.: On logics of knowledge and belief. Philosophical Studies 128, 169–199 (2006)

## Appendix 1: Soundness and Completeness of PDL$^+$

To define the proper program transformation for PDL$^+$ we need a function $^\cup$ that maps each PDL program to its converse (in the obvious sense that the interpretation of $\pi^\cup$ is the converse of that of $\pi$):

$$\begin{array}{ll}(a\check{})^\cup & = a \\ (?\phi)^\cup & = ?\phi \\ (\pi_1;\pi_2)^\cup & = {\pi_2}^\cup;{\pi_1}^\cup \\ (\pi_1 \cup \pi_2)^\cup & = {\pi_1}^\cup \cup {\pi_2}^\cup \\ (\pi^*)^\cup & = (\pi^\cup)^*\end{array}$$

What is needed to get a completeness proof is a redefinition of the epistemic program transformation operation $T_{ij}^{\mathbf{A}}$ used in the LCC completeness to push an action model modality $[\mathbf{A}, e]$ through an epistemic program modality $[\pi]$.

$$\begin{array}{rcl}\underline{T}_{ij}^{\mathbf{A}}(a) & = & \begin{cases} ?pre(e_i);\mathbf{SUB}(e_i)(a) & \text{if } e_i \mapsto_{\mathbf{SUB}(e_i)(a)} e_j \text{ in } \mathbf{A} \\ ?\bot & \text{otherwise}\end{cases} \\ \underline{T}_{ij}^{\mathbf{A}}(a\check{}) & = & \begin{cases} ?pre(e_i);(\mathbf{SUB}(e_i)(a))^\cup & \text{if } e_i \mapsto_{(\mathbf{SUB}(e_i)(a))^\cup} e_j \text{ in } \mathbf{A} \\ ?\bot & \text{otherwise}\end{cases} \\ \underline{T}_{ij}^{\mathbf{A}}(?\phi) & = & \begin{cases} ?(pre(e_i) \wedge [\mathbf{A}, e_i]\phi) & \text{if } i = j \\ ?\bot & \text{otherwise}\end{cases} \\ \underline{T}_{ij}^{\mathbf{A}}(\pi_1;\pi_2) & = & \bigcup_{k=0}^{n-1}(\underline{T}_{ik}^{\mathbf{A}}(\pi_1);\underline{T}_{kj}^{\mathbf{A}}(\pi_2)) \\ \underline{T}_{ij}^{\mathbf{A}}(\pi_1 \cup \pi_2) & = & \underline{T}_{ij}^{\mathbf{A}}(\pi_1) \cup \underline{T}_{ij}^{\mathbf{A}}(\pi_2)) \\ \underline{T}_{ij}^{\mathbf{A}}(\pi^*) & = & K_{ijn}^{\mathbf{A}}(\pi)\end{array}$$

where it is assumed that the action model $\mathbf{A}$ has $n$ states, and the states are numbered $0, \ldots, n-1$. $K^{\mathbf{A}}_{ijn}$ is the Kleene path transformer, as in [9].

The proof system for PDL$^+$ consists of all axioms and rules of LCC except the reduction axiom:

$$[\mathbf{A}, e_i][\pi]\phi \leftrightarrow \bigwedge_{j=0}^{n-1} [T^{\mathbf{A}}_{ij}(\pi)][\mathbf{A}, e_j]\phi.$$

In addition, PDL$^+$ has the axioms for converse atomic programs as in section 2, and reduction axioms of the form:

$$[\mathbf{A}, e_i][\pi]\phi \leftrightarrow \bigwedge_{j=0}^{n-1} [\underline{T}^{\mathbf{A}}_{ij}(\pi)][\mathbf{A}, e_j]\phi.$$

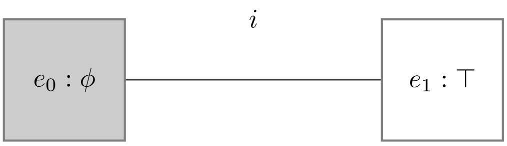
This is the patch we need to prove the completeness result (Theorem 2).

## Appendix 2: Restricted Announcements Versus Restricted Belief Changes

A restricted announcement of $\phi$ is an announcement of $\phi$ that is not delivered to one of the agents $i$. Notation $!\phi^{-i}$. The action model for $!\phi^{-i}$ has event set $\{e_0, e_1\}$, with $e_0$ the actual event, where $e_0$ has precondition $\phi$ and $e_1$ precondition $\top$, and with the preference relation given by

$$P_i = \{(e_0, e_0), (e_1, e_1), (e_0, e_1), (e_1, e_0)\},$$

and $P_j = \{(e_0, e_0), (e_1, e_1)\}$ for all $j \neq i$.




A protocol for restricted announcements, for epistemic situation $M$, is a set of finite sequences of formula-agent pairs, such that each sequence

$$(\phi_0, i_0), \ldots, (\phi_n, i_n)$$

has the following property:

$$\forall k \in \mathbb{N} : 0 \leq k < n \rightarrow \exists i \in Ag : \mathbf{M}, w \models [!\phi_0^{-i_0}], \ldots, [!\phi_{k-1}^{-i_{k-1}}][\sim_i]\phi_k.$$

Intuitively, at every stage in the sequence of restricted announcements, some agent has to possess the required knowledge to make the next announcement in the sequence. We can now prove that such protocols can never establish common knowledge of purely propositional facts.

**Theorem 4.** *Let $C$ express common knowledge among set of agents $Ag$. Let $\mathbf{M}$ be an epistemic model with actual world $w$ such that $\mathbf{M}, w \models \neg C\phi$, with $\phi$ purely propositional. Then there is no protocol with*

$$\mathbf{M}, w \models [!\phi_0^{-i_0}], \ldots, [!\phi_n^{-i_n}]C\phi.$$

*for any sequence $(\phi_0, i_0), \ldots, (\phi_n, i_n)$ in the protocol.*

*Proof.* We show that $\neg C\phi$ is an invariant of any restricted announcement.

Assume $\mathbf{M}, w \models \neg C\phi$. Let $(\mathbf{A}, e)$ be an action model for announcement $!\psi^{-i}$, the announcement of $\psi$, restricted to $Ag - \{i\}$. Then $\mathbf{A}$ has events $e$ and $e'$, with $\mathbf{pre}(e) = \psi$ and $\mathbf{pre}(e') = \top$. If $\mathbf{M}, w \models \neg\psi$ then the update does not succeed, and there is nothing to prove. Suppose therefore that $\mathbf{M}, w \models \psi$. Since $\mathbf{pre}(e') = \top$, the model $\mathbf{M} \otimes \mathbf{A}$ restricted to domain $D = \{(w, e') \mid w \in W_{\mathbf{M}}\}$ is a copy of the original model $\mathbf{M}$. Thus, it follows from $\mathbf{M}, w \models \neg C\phi$ that

$$\mathbf{M} \otimes A \restriction D, (w, e') \models \neg C\phi.$$

Thus, there is an $C$-accessible world-event pair $(w', e'')$ in $D$ with

$$\mathbf{M} \otimes A \restriction D, (w', e'') \models \neg\phi.$$

Since $\phi$ is purely propositional, we get from this that:

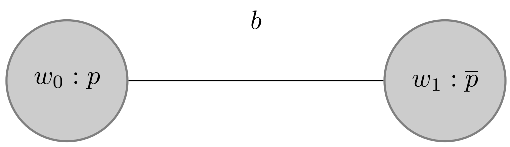
$$\mathbf{M} \otimes A, (w', e'') \models \neg\phi.$$

Observe that since common knowledge is preserved under model restriction, absence of common knowledge is preserved under model extension. The $C$-accessible world-event pair $(w', e'')$ in $\mathbf{M} \otimes A \restriction D$ will still be $C$-accessible in $\mathbf{M} \otimes A$. Therefore, it follows that $\mathbf{M} \otimes A, (w, e') \models \neg C\phi$. By the construction of $\mathbf{M} \otimes A$, we get from this that $\mathbf{M} \otimes A, (w, e) \models \langle i \rangle \neg C\phi$, and therefore $\mathbf{M} \otimes A, (w, e) \models \neg C\phi$, by the definition of common knowledge.

It follows immediately that no protocol built from restricted announcements can create common knowledge of propositional facts.

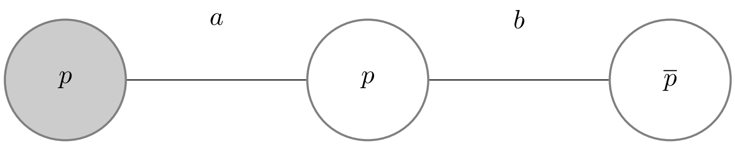
The case of the two generals planning a coordinated attack on the enemy, but failing to achieve common knowledge about it [16,17] can be viewed as a special case of this theorem.

If there are just two agents $i, j$, the only way for agent $i$ to send a restricted message is by allowing uncertainty about the delivery. If $i, j$ are the only agents, and $i$ knows $\phi$ then the restricted message $!\phi^{-j}$ conveys no information, so the only reasonable restricted announcement of $\phi$ is $!\phi^{-i}$. The upshot of this announcement is that the message gets delivered to $j$, but $i$ remains uncertain about this. According to the theorem, such messages cannot create common knowledge. Initial situation:

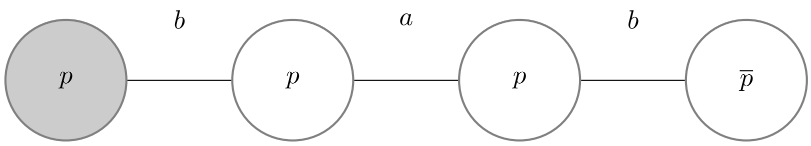


Update action for general $a$ (left) and general $b$ (right):


Situation after first message from general $a$:

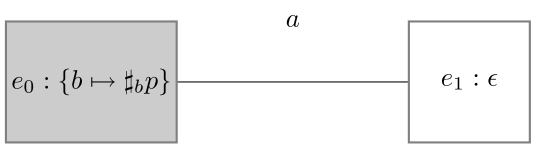



Situation after update by $a$ followed by update by $b$:

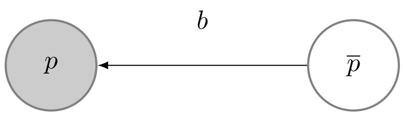



And so on ...

Now look at the case where restricted announcements are replaced by non-public belief revisions. Then the power of restricted belief change turns up in the following example. We start out from the initial situation again, and we update using the action model for non-public belief change:




Here is the update result (after minimalisation under bisimulation):




The example shows that it is possible to achieve common safe belief in $p$ in a single step, by means of a non-public belief change.