Logics of Communication and Change

Johan van Benthem [a] Jan van Eijck [b] Barteld Kooi [c]

[a] *ILLC, University of Amsterdam, Plantage Muidergracht 24, 1018 TV Amsterdam, The Netherlands & Philosophy Department, Stanford University*

[b] *CWI, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands & Uil-OTS, Trans 10, 3512 JK, Utrecht, The Netherlands*

[c] *Department of Philosophy, University of Groningen, Oude Boteringestraat 52, 9712 GL, Groningen, The Netherlands*

Abstract

Current dynamic epistemic logics for analyzing effects of informational events often become cumbersome and opaque when common knowledge is added for groups of agents. Still, postconditions involving common knowledge are essential to successful multi-agent communication. We propose new systems that extend the epistemic base language with a new notion of 'relativized common knowledge', in such a way that the resulting full dynamic logic of information flow allows for a compositional analysis of all epistemic postconditions via perspicuous 'reduction axioms'. We also show how such systems can deal with factual alteration, rather than just information change, making them cover a much wider range of realistic events. After a warm-up stage of analyzing logics for public announcements, our main technical results are expressivity and completeness theorems for a much richer logic that we call LCC. This is a dynamic epistemic logic whose static base is propositional dynamic logic (PDL), interpreted epistemically. This system is capable of expressing all model-shifting operations with finite action models, while providing a compositional analysis for a wide range of informational events. This makes LCC a serious candidate for a standard in dynamic epistemic logic, as we illustrate by analyzing some complex communication scenarios, including sending successive emails with both 'cc' and 'bcc' lines, and other private announcements to subgroups. Our proofs involve standard modal techniques, combined with a new application of Kleene's Theorem on finite automata, as well as new Ehrenfeucht games of model comparison.

Key words: epistemic logic, update, dynamic logic, common knowledge, reduction axioms, product update, finite automata, Kleene's Theorem

Email addresses: `johan@science.uva.nl` (Johan van Benthem), `jve@cwi.nl` (Jan van Eijck), `B.P.Kooi@rug.nl` (Barteld Kooi).

1 Introduction

Epistemic logic deals with what agents consider possible given their current information. This includes knowledge about facts, but also *higher-order information* about information that other agents have. A prime example is *common knowledge.* A formula φ is common knowledge if everybody knows φ, everybody knows that everybody knows that φ, and so on. *Common belief* is an important related notion. Indeed, although this paper is mainly written in 'knowledge' terminology, everything we say also holds, with minor technical modifications, when describing agents' *beliefs*, including common belief.

Dynamic epistemic logics analyze changes in both basic and higher-order information. One of the main attractions of such systems is their transparent analysis of effects of communicative actions in the format of an equivalence between epistemic postconditions and preconditions. A typical example concerns knowledge of an agent after and before a public announcement:

$$[\varphi]\Box_a\psi \leftrightarrow (\varphi \rightarrow \Box_a[\varphi]\psi).$$

This axiom says that after the announcement that φ agent a knows that ψ iff φ implies that agent a knows that after φ is announced ψ will be true. We call such principles *reduction axioms*, because the announcement operator is 'pushed through' the epistemic operator, in such manner that on the right hand side the complexity of the formula in the scope of the announcement is less that the complexity of the formula in the scope of the announcement on the left hand side. This reduction axiom describes the interaction between the announcement operator and the epistemic operator. If there is a reduction axiom for each logical operator in the language, such a set of axioms make logical systems particularly straightforward. For instance, the logic of public announcements without common knowledge has an easy completeness proof by way of a translation that follows the reduction axioms. Formulas with announcements are translated to provably equivalent ones without announcements, and completeness follows from the known completeness of the epistemic base logic. Thus, the dynamic logic of the announcement operator is fully characterized by the reduction axioms.

This is the technical way of putting things. But more importantly, reduction axioms like the one above also reflect a desirable *methodology*: they allow for *compositional analysis* of the epistemic effects of informational events. This is particularly helpful with more complex scenarios, where it is not at all easy to describe just what agents should know, or not, after some communication has taken place: say, a round of emails involving both public 'cc' and half-private 'bcc' lines. A dynamic epistemic logic with a complete set of reduction axioms has an ideal 'harmony' between its static and dynamic parts allowing for complete compositional analysis. So, it is worth finding such systems whenever they exist. Finally, more locally, specific reduction axioms also express interesting assumptions about the interplay of events and knowledge. E.g., it has often been observed that the above one for public announcement embodies a form of 'Perfect Recall': the event of announcement does not add or delete uncertainty lines for agents among those worlds which they consider possible.

In this light, there is a problem with common knowledge for *groups* in societies of communicating agents. Understanding group knowledge as it is gained or lost, is at the heart of analyzing epistemic update. But existing dynamic epistemic logics have no compositional reduction axioms for achieving common knowledge, and this infelicity has been there from the start. Now, the seminal paper [1] does treat common knowledge per se, but not on a reductive pattern, and its completeness proof is correspondingly messy. Indeed, reduction axioms are not available, as the logic with epistemic updates is more expressive than the logic without them. We think this is an infelicity of design, and our main aim in this paper is to show how compositional analysis is feasible by some judicious language extension, restoring the proper harmony between the static and dynamic features of the system.

In Section 2 we first look at examples of the general kinds of information change that we are interested in. These include public announcement, but also communication involving privacy and partial observation, and indeed, observation of any sort of event that carries information. We also include real physical actions changing the world. Before we give a system that deals with all these phenomena, we first look at a pilot case that illustrates many issues in a simpler setting, the logic PAL of *public announcements.* Section 3 gives a new and complete set of reduction axioms for public announcement logic with common knowledge, obtained by strengthening the base language with an operator of relativized common knowledge, as first proposed in [3]. Moreover, since languages with model-shifting operators like $[\varphi]$ are of independent logical interest, we develop the model theory of PAL a bit further, using new game techniques for epistemic languages with fixed-point operators for common knowledge to investigate its expressive power. Section 4, the heart of this paper, then proposes a new dynamic epistemic LCC dealing with the general case of updating with finite structures of events, generalizing the standard reference [1] to include a much wider range of epistemic assertions, as well as factual change. What the section demonstrates, in particular, is that PDL (the well-known system of propositional dynamic logic), when interpreted epistemically, can serve as a basis for a rich and expressive logic of communication that allows for smooth compositional analysis of common knowledge after epistemic updates. To avoid confusion with PDL in its non-epistemic uses for analyzing actions, we will call our version here LCC.

A general approach that reduces dynamic epistemic logic to propositional dynamic logic was first proposed using finite automata techniques in [18], using a variant of propositional dynamic logic called ‘automata PDL’. The new techniques used in the present paper (cf. [11]) work directly in epistemic PDL, using the idea behind Kleene’s Theorem for regular languages and finite automata to find the relevant reduction axioms inductively by means of ‘program transformations’. This analysis does not just yield the meta-theorems of completeness and decidability that we are after. It can also be used in practice to actually compute valid axioms analyzing common knowledge following specific communicative or informational events. Section 5 analyzes some of our earlier communication types in just this fashion, obviating the need for earlier laborious calculations ‘by hand’ (cf. [27]). Finally, Section 6 draws conclusions, and indicates directions for further research.

The broader context for this paper are earlier systems of dynamic epistemic logic, with [26], [14], and [1] as key examples of progressively stronger systems, while [8] is a source of inspiring examples. Reduction axioms were already used to prove completeness for dynamic epistemic logics in [14] and [1]. Another major influence is the work of [13] on common knowledge in computational settings, using a more general temporal-epistemic framework allowing also for global protocol information about communicative processes. Connections between the two approaches are found, e.g., in [21]. Further references to the literature on epistemic actions and to many challenging open problems in the general landscape of update logics can be found in [5].

Even though the main thrust of this paper may seem technical, our proposal is much more than just a trick for smoothing completeness proofs, or for finding a new model-theoretic play-ground. It also addresses a significant design issue of independent interest: what is the most convenient and transparent epistemic language for describing information flow for groups of agents in a compositional manner? Our main logic LCC in Section 4 is meant as a serious proposal for a standard.

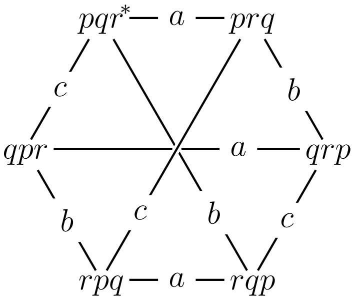
2 Modelling Effects of Communication and Change

Epistemic update logics are about the effects of general communication, and indeed, they describe the logic of observing any kind of information-bearing event. But in practice, it is helpful to look at more constrained scenarios. A good source of examples are basic actions in *card games.* Game moves then involve looking at a card, showing a card to someone (with or without other players looking on), exchanging cards with someone (with or without other players looking on), and perhaps even changing the setting in more drastic ways (cf. [8]). This is not just a frivolous move toward parlour games. One can think of ‘card events’ as a sort of normal form for any type of informational activity — and one which has the additional virtue of evoking vivid intuitions.

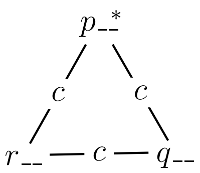
Moreover, scenarios involving the interplay of information and ignorance are not just logician's puzzles for their own sake: managing the right mixtures of information and ignorance is absolutely essential to human intelligence, and to the functioning of civilized societies.

In this section we list some examples involving combinations of epistemic and actual change that we think any full-fledged dynamic-epistemic logic should be able to deal with. The simplest scenario here is public announcement of some fact P, which merely requires elimination of all worlds in the current model where P does not hold. But general communication can be much more complex — just think of whispers in a lecture theatre. This requires updates of the initial information model beyond mere elimination of worlds. Some updates even make the current model bigger, as alternatives can multiply. Since all these actions involve groups of agents, understanding both individual and common knowledge following certain events is clearly essential.

Card Showing A simple card showing situation goes as follows. Alice, Bob and Carol each hold one of the cards p q, r. The actual deal is: Alice holds p, Bob holds q, Carol holds r. Assuming all players looked at their own cards, but have kept them hidden from the others, this situation is modelled as follows (xyz represents the situation where Alice holds x, Bob holds y and Carol holds z, xyz—a—$x'y'z'$ represents the fact that Alice cannot distinguish xyz from $x'y'z'$, and xyz^* indicates that xyz is the situation that actually is the case):

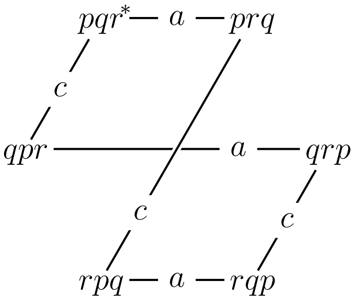


Now assume Alice shows her card p to the whole group. This public event eliminates all worlds from the initial model that conflict with the new information. Thus, out of the six given worlds only two remain: pqr and prq. In the resulting model, Bob and Carol know all the cards, while Alice only knows that she has p — and both these facts are common knowledge. Now consider a 'semi-public' action of Alice showing her card to Bob, with Carol looking on (Carol sees that a card is shown, but does not see which card). Here is a major new idea, due to [1]. We first picture the new event itself as an *update model* whose structure is similar to that of epistemic models in general:

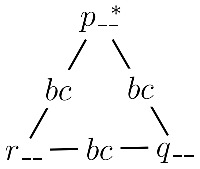


According to this picture, the card that Alice shows to Bob is card p, but for all Carol knows, it might have been q or r. In actual fact it cannot be r, as that is the card which Carol holds and has just inspected, but this information is not part of the update action. Note that the events depicted cannot occur in just any world. Alice can only show card p when she actually holds p, she can show q when she actually holds q and she can show r when she actually holds r. The latter information is encoded in so-called *preconditions*, and indeed, the fundamental reason why occurrences of events carry information for us is that we know their preconditions.

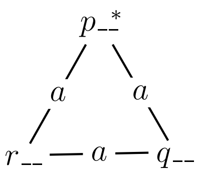
Now for the update from the preceding event. Intuitively, we want a new information model arising from the initial one that the agents were in plus the update model containing the relevant actions. The new worlds are then old worlds plus the most recent event attached. Moreover, our intuitions tell us what the desired result of this update should look like, viz.:



Card Inspection Next, consider acts of observation. Suppose the three cards are dealt to Alice, Bob and Carol, but are still face down on the table. The following picture describes an update model for Alice's inspecting her own card and discovering it to be p, with the others just looking on.

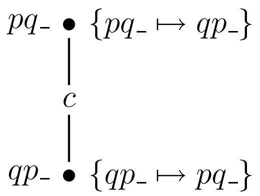


And here is the related update model of Alice picking up her card and showing it to the others, without taking a look herself:

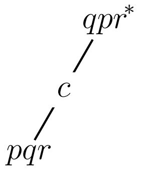


In all these cases we have clear intuitions about what the outcomes of the updates should be, and these underlie the technical proposals made in Section 4.

Card Exchange Next, we add a case where real physical action takes place, which is not purely informational. Suppose in the initial situation, where Alice holds p, Bob holds q and Carol holds r, Alice and Bob exchange cards, without showing the cards to Carol. To model this, we need an update that also changes the state of the world. For that, we need to change the valuation for atomic facts. This may be done by using 'substitutions' which reset the truth values of those atomic statements that are affected by the action:

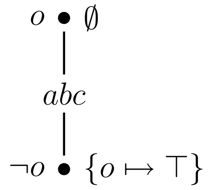


Note that the diagram now indicates both the earlier *preconditions* for events or actions, and *postconditions* for their successful execution. Here is the result of applying this update model in the initial situation, where all players have looked at their cards and the actual deal is pqr:

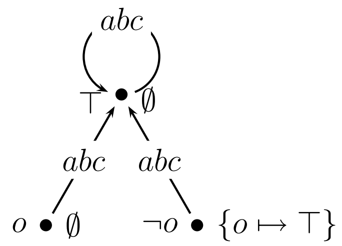


So far, we have looked at card examples, where actions are meant to be communicative, with some intentional agent. But it is important to realize that dynamic epistemic logic can also serve as a system for analyzing arbitrary *observations* of events that carry some information to observers, whether intended or not. That is, it is a logic of perception as much as of communication, and it is useful for modelling the essence of what goes on in very common everyday actions. We conclude with two illustrations in the latter mode.

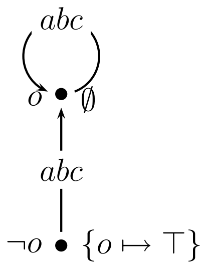
Opening a window The precondition for opening a window is that the window is closed. To make this into an update that can always be performed, we specify different actions depending on the state of the window. If it is open, nothing needs to be done; if it is closed, then open it. This much is standard dynamic logic, as used in describing transition systems in computer science. But now we are in a setting where agents may have different 'epistemic access' to what is taking place. E.g., assuming the relevant event is invisible to all (Alice, Bob and Carol), it can be modelled as follows:



Again, we see both pre- and postconditions, with the latter depending on the former. Note that this action can make agents 'out of touch with reality', perhaps through laziness in observation. Such a mismatch can also result from being actively misled. If the window is opened in secret, its update model looks as follows (o denotes an open window; $o \mapsto \top$ is the substitution making o true).



Further variations on this update model are possible. E.g., the window is in fact opened, while everyone is told that it was already open. Here is the corresponding update:



Fiddling with a light switch Fiddling with a light switch is an update that depends on the actual situation as follows: if the light is on, then switch it off, if it is off, then switch it on. If this fiddling is done in a way such that the result is visible to all, then here is its update model:

$$\top \bullet \{o \mapsto \neg o\}$$

If the fiddling (and its result) are kept secret, the corresponding update model looks like the above opening, but now with the new substitution.

Remark: Two Limitations The preceding examples suggest that we can view an update model with several event as a disjunction of instructions under conditions, i.e., as a kind of *structured program* 'if the window is open then do A or if the window is ajar, then do B, or, if the window is wide open, then do C ....' This suggests a set-up where update models are built from simple actions by means of the regular operations of choice, sequence and iteration — and perhaps even concurrent composition of events. We will not pursue this approach here. Indeed, natural though it is, it transcends the boundaries of our analysis. E.g., Miller and Moss [23] show that just adding finite iteration $*$ of announcements already leads to an undecidable dynamic logic, not effectively reducible to its decidable base.

Another boundary that we will not cross is the *atomic form* of our postconditions for world-changing actions. In more general scenarios, captured by logics with operators in the spirit of 'See To It That φ', one may want to define some action as having some complex epistemic effect described by arbitrary φ, such as 'make sure that only courageous people know the true state of affairs'. Modeling complex postconditions raises some delicate technical issues, orthogonal to our main concerns here.

3 Logics of Public Announcement

Many of the issues we want to deal with in the full-fledged logic of epistemic updates are also present in the logic of the simplest form of communicative action: public announcement logic. The corresponding update idea that announcing a proposition φ removes all worlds where φ does not hold goes back far into the mists of logical folklore, and it has been stated explicitly since the 1970s by Stalnaker, Heim, and others. The same idea also served as a high-light in the work on epistemic logic in computer science (cf. [13]). Its first implementation as a dynamic-epistemic logic seems due to Plaza [26].

Section 3.1 is a brief introduction to public announcement logic (PAL) as usually stated. In Section 3.2 we give a new base logic of *relativized common knowledge*, EL-RC. This extension was first proposed in [3], which analyzed updates as a kind of *relativization* operator on models. Restricted or 'bounded' versions of logical operators like quantifiers or modalities are very common in semantics, and we will provide further motivation below. The resulting epistemic logic with relativized common knowledge is expressive enough to allow a reduction axiom for common knowledge. A proof system is defined in Section 3.3, and shown to be complete in Section 3.4. The system is extended with reduction axioms for public announcements in Section 3.5. The existence of reduction axioms for public announcements and relativized common knowledge suggests that this new logic is more expressive than the epistemic logics with public announcements proposed elsewhere in the literature. Hence it is interesting to investigate the expressive power of this new logic with characteristic model comparison games. These games are provided in Section 3.6. This technique is then used to investigate the expressive power of relativized common knowledge in Section 3.7, settling all issues of system comparison in this area. Finally, complexity issues are briefly discussed in Section 3.8.

Once again, our systems work for agents' beliefs as well as knowledge in more constrained models. In our informal explanations, we will use either notion, as seems best for getting points across.

3.1 Language and Semantics of PAL

A public announcement is an epistemic event where all agents are told simultaneously and transparently that a certain formula holds right now. This is modeled by a modal operator $[\varphi]$. A formula of the form $[\varphi]\psi$ is read as 'ψ holds after the announcement of φ'. If we also add an operator $C_B\varphi$ to express that φ is common knowledge among agents B, we get public announcement logic with common knowledge (PAL-C). The languages $\mathscr{L}_{\mathsf{PAL}}$ and $\mathscr{L}_{\mathsf{PAL\text{-}C}}$ are interpreted in standard models for epistemic logic.

Definition 1 (Epistemic Models) *Let a finite set of propositional variables P and a finite set of agents N be given. An* epistemic model *is a triple $M = (W, R, V)$ such that*

- $W \neq \emptyset$ *is a set of possible worlds,*
- $R : N \to \wp(W \times W)$ *assigns an accessibility relation $R(a)$ to each agent a,*
- $V : P \to \wp(W)$ *assigns a set of worlds to each propositional variable.*

In epistemic logic the relations $R(a)$ are usually equivalence relations. In this paper we treat the general modal case without such constraints — making 'knowledge' more like *belief*, as observed earlier. But our results also apply to the special modal S5-case of equivalence relations. The semantics are defined with respect to models with a distinguished 'actual world': M, w.

Definition 2 (Semantics of PAL and PAL-C) *Let a model M, w with $M = (W, R, V)$ be given. Let $a \in N$, $B \subseteq N$, and $\varphi, \psi \in \mathscr{L}_{\mathsf{PAL}}$. For atomic proposi-*

tions, negations, and conjunctions we take the usual definition. The definitions for the other operators run as follows:

$$M, w \models \Box_a\varphi \text{ iff } M, v \models \varphi \text{ for all } v \text{ such that } (w, v) \in R(a)$$

$$M, w \models [\varphi]\psi \text{ iff } M, w \models \varphi \text{ implies } M|\varphi, w \models \psi$$

$$M, w \models C_B\varphi \text{ iff } M, v \models \varphi \text{ for all } v \text{ such that } (w, v) \in R(B)^+$$

where $R(B) = \bigcup_{a\in B} R(a)$, *and* $R(B)^+$ *is its transitive closure. The updated model* $M|\varphi = (W', R', V')$ *is defined by restricting* M *to those worlds where* φ *holds. Let*

$$[\![\varphi]\!] = \{v \in W|M, v \models \varphi\}.$$

Now $W' = [\![\varphi]\!]$, $R'(a) = R(a) \cap (W \times [\![\varphi]\!])$, *and* $V'(p) = V(p) \cap [\![\varphi]\!]$.

Here, mostly for convenience, we chose to define common knowledge as a transitive closure, as in [13]. In [22], common knowledge is defined as the *reflexive* transitive closure. Our results will work either way, with minimal adaptations.

A completeness proof for public announcement logic without an operator for common knowledge (PAL) is straightforward.

Definition 3 (Proof System for PAL) *The proof system for* PAL *is that for multi-modal S5 epistemic logic plus the following reduction axioms:*

$$\begin{aligned} \textit{Atoms} &\vdash [\varphi]p \leftrightarrow (\varphi \to p) \\ \textit{Partial Functionality} &\vdash [\varphi]\neg\psi \leftrightarrow (\varphi \to \neg[\varphi]\psi) \\ \textit{Distribution} &\vdash [\varphi](\psi \wedge \chi) \leftrightarrow ([\varphi]\psi \wedge [\varphi]\chi) \\ \textit{Knowledge Announcement} &\vdash [\varphi]\Box_a\psi \leftrightarrow (\varphi \to \Box_a[\varphi]\psi \end{aligned}$$

as well as the following rules of inference:

(Announcement generalization) *From* $\vdash \psi$, *infer* $\vdash [\varphi]\psi$.

The formulas on the left of these equivalences are of the form $[\varphi]\psi$. In **Atoms** the announcement operator no longer occurs on the right-hand side. In the other reduction axioms formulas within the scope of an announcement are of higher complexity on the left than on the right. Note that the **Distribution** axiom is the well known K-axiom from modal logic. When applied successively, these axioms turn every formula of the dynamic language into an equivalent static one, thus showing the earlier 'harmony' between the static and dynamic parts of the total system.

This system allows for compositional analysis of the epistemic effects of statements made in groups of agents. In this light, even the technical reduction to the static part has a more general thrust worth pointing out. What it says is that knowing the current knowledge of agents in some models suffices, in principle, for knowing the effects of nay announcement actions that could occur. Thus, in the terminology of [6], the static language is rich enough to *pre-encode* all dynamic effects. This is a powerful idea which also occurs in conditional logic, and in reasoning with conditional probabilities. One way of understanding our next topic is as a move towards achieving pre-encoding for common knowledge, too. Here is why this requires work. For public announcement logic with a common knowledge operator (PAL-C), a completeness proof via reduction axioms is impossible. There is no such axiom for formulas of the form $[\varphi]C_B\psi$, given the results in [1].

3.2 Relativized Common Knowledge: EL-RC

Even so, the semantic intuitions for achieving common knowledge by announcement are clear. If φ is true in the old model, then every B-path in the new model ends in a ψ world. This means that in the old model every B-path that consists exclusively of φ-worlds ends in a $[\varphi]\psi$ world. To facilitate this, we introduce a new operator $C_B(\varphi, \psi)$, which expresses that

> every B-path which consists exclusively of φ-worlds ends in a ψ world.

We call this notion *relativized common knowledge.* A natural language paraphrase might be 'if φ is announced it becomes common knowledge among B that ψ was the case before the announcement.' A shorter paraphrase of $C_B(\varphi, \psi)$ that we will use henceforth is 'ψ is φ-relative common knowledge among group B.' Henceforth we consider only such φ-relative or φ-conditional common knowledge of agents, just as one does in logics of doxastic *conditionals*, where $A \implies B$ means something like "if I were to learn that A, I would believe that B." Yet another helpful analogy may be with the well-known 'Until' of temporal logic. A temporal sentence 'φ until ψ' is true iff there is some point in the future where ψ holds and φ is true up to that point. All these readings show that the new notion has some concrete intuition behind it. Its other virtue, as we shall see presently, is mathematical elegance.

Definition 4 (Language and Semantics of EL-RC) *The language of* EL-RC *is that of* EL*, together with the operator for relativized common knowledge, with semantics given by:*

$$M, w \models C_B(\varphi, \psi)$$

$$\textit{iff}$$

$$M, v \models \psi \textit{ for all } v \textit{ such that } (w, v) \in (R(B) \cap (W \times [\![\varphi]\!]))^+$$

where $(R(B) \cap (W \times [\![\varphi]\!]))^+$ *is the transitive closure of* $R(B) \cap (W \times [\![\varphi]\!])$*.*

Note that φ-relative common knowledge is *not* what results from a public update with φ. E.g., $[p]C_B\Diamond_a\neg p$ is *not* equivalent to $C_B(p, \Diamond_a\neg p)$, for $[p]C_B\Diamond_a\neg p$ is always false, and $C_B(p, \Diamond_a\neg p)$ holds in models where every B path through p worlds ends in a world with an a successor with $\neg p$. In Section 3.7 we will show that $C_B(p, \Diamond_a\neg p)$ cannot be expressed in PAL-C.

The semantics of the other operators is standard. Ordinary common knowledge can be defined with the new notion: $C_B\varphi \equiv C_B(\top, \varphi)$.

3.3 Proof System for EL-RC

Relativized common knowledge still resembles common knowledge, and so we need just a slight adaptation of the usual axioms.

Definition 5 (Proof System for EL-RC) *The proof system for* EL-RC *has these axioms:*

$$\begin{array}{rl}
\textit{Tautologies} & \textit{All instantiations of propositional tautologies} \\
\Box\ \textit{Distribution} & \vdash \Box_a(\varphi \to \psi) \to (\Box_a\varphi \to \Box_a\psi) \\
C\ \textit{Distribution} & \vdash C_B(\varphi, \psi \to \chi) \to (C_B(\varphi, \psi) \to C_B(\varphi, \chi)) \\
\textit{Mix} & \vdash C_B(\varphi, \psi) \leftrightarrow E_B(\varphi \to (\psi \wedge C_B(\varphi, \psi))) \\
\textit{Induction} & \vdash (E_B(\varphi \to \psi) \wedge C_B(\varphi, \psi \to E_B(\varphi \to \psi))) \to C_B(\varphi, \psi)
\end{array}$$

and the following rules of inference:

(Modus Ponens) *From* $\vdash \varphi$ *and* $\vdash \varphi \to \psi$ *infer* $\vdash \psi$*.*
(□ Necessitation) *From* $\vdash \varphi$ *infer* $\vdash \Box_a\varphi$*.*
(C Necessitation) *From* $\vdash \varphi$ *infer* $\vdash C_B(\psi, \varphi)$*.*

In the **Mix** *and the* **Induction** *axiom, the notation* $E_B\varphi$ *is an abbreviation of* $\bigwedge_{a \in B} \Box_a\varphi$ *(*everybody *believes, or knows,* φ*).*

These axioms are all sound on the intended interpretation. In particular, understanding the validity of the relativized versions **Mix** *and* **Induction** *provides the main idea of our analysis.*

Next, a proof consists of a sequence of formulas such that each is either an instance of an axiom, or it can be obtained from formulas that appear earlier in the sequence by applying a rule. If there is a proof of φ*, we write* $\vdash \varphi$*.*

Remark It may also be helpful to write $C_B(\varphi, \psi)$ as a sentence in propositional dynamic logic PDL: $[(\bigcup_{a \in B} a; ?\varphi)^+]\psi$. Our proof system essentially follows the usual PDL-axioms for this formula. This technical observation is the key to our more general system LCC in Section 4 below.

3.4 Completeness for EL-RC

To prove completeness for our extended static language EL-RC, we follow [16], [13]. The argument is standard, and our main new point is just that the usual proof in the literature actually yields information about a richer language than is commonly realized.

For a start, we take maximally consistent sets with respect to finite fragments of the language that form a canonical model for that fragment. In particular, for any given formula φ we work with a finite fragment called the *closure* of φ. This is the appropriate analogue of the *Fisher-Ladner closure* from the PDL literature (see [16]).

Definition 6 (Closure) *The* closure *of φ is the minimal set Φ such that*

(1) $\varphi \in \Phi$,
(2) Φ is closed under taking subformulas,
(3) If $\psi \in \Phi$ and ψ is not a negation, then $\neg\psi \in \Phi$,
(4) If $C_B(\psi, \chi) \in \Phi$, then $\Box_a(\psi \to (\chi \wedge C_B(\psi, \chi))) \in \Phi$ for all $a \in B$.

Definition 7 (Canonical Model) *The* canonical model *M_φ for φ is the triple $(W_\varphi, R_\varphi, V_\varphi)$ where*

- *$W_\varphi = \{\Gamma \subseteq \Phi \mid \Gamma$ is maximally consistent in $\Phi\}$;*
- *$(\Gamma, \Delta) \in R_\varphi(a)$ iff $\psi \in \Delta$ for all ψ with $\Box_a\psi \in \Gamma$;*
- *$V_\varphi(p) = \{\Gamma \mid p \in \Gamma\}$.*

Next, we show that a formula in such a finite set is true in the canonical model where that set is taken to be a world, and vice versa.

Lemma 8 (Truth Lemma) *For all $\psi \in \Phi$, $\psi \in \Gamma$ iff $M_\varphi, \Gamma \models \psi$.*

Proof By induction on ψ. The cases for propositional variables, negations, conjunction, and individual epistemic operators are straightforward. Therefore we focus on the case for relativized common knowledge.

From left to right. Suppose $C_B(\psi, \chi) \in \Gamma$. If there is no Δ such that $(\Gamma, \Delta) \in (R(B) \cap (W_\varphi \times [\![\psi]\!]))^+$, then $(M_\varphi, \Gamma) \models C_B(\psi, \chi)$ holds trivially.

Otherwise, take a $\Delta \in W_\varphi$ such that $(\Gamma, \Delta) \in (R(B) \cap (W_\varphi \times [\![\psi]\!]))^+$. We have to show that $\Delta \models \chi$, but we show something stronger, namely that $\Delta \models \chi$ and $C_B(\psi, \chi) \in \Delta$. This is done by induction on the length of the path from Γ to Δ. The base case is a path of length 1. From our assumption it follows that $\psi \in \Delta$. Our assumption that $C_B(\psi, \chi) \in \Gamma$ implies that $\vdash \delta_\Gamma \to \Box_a(\psi \to (\chi \wedge C_B(\psi, \chi)))$ by the **Mix** axiom. The formula χ is also in Φ. Therefore $\chi \in \Delta$. By applying the induction hypothesis we get $(M_\varphi, \Delta) \models \chi$. We already assumed that $C_B(\psi, \chi) \in \Phi$, therefore also $C_B(\psi, \chi) \in \Delta$. So we are done with the base case.

Now suppose that the path to Δ is of length $n+1$. There must be a path from Γ of length n to a Θ in $(R(B) \cap (W_\varphi \times [\![\psi]\!]))^+$ such that $(\Theta, \Delta) \in R(a)$ for some $a \in N$ and $(M_\varphi, \Delta) \models \psi$. By the induction hypothesis $C_B(\psi, \chi) \in \Theta$. Now we can apply the same reasoning as in the base case to conclude that $(M_\varphi, \Delta) \models \chi$ and $C_B(\psi, \chi) \in \Delta$.

From right to left. Suppose $(M_\varphi, \Gamma) \models C_B(\psi, \chi)$. Now consider the set Λ:

$$\Lambda = \{\delta_\Delta | (\Gamma, \Delta) \in (R(B) \cap (W_\varphi \times [\![\psi]\!]))^+\}$$

Let $\delta_\Lambda = \bigvee_{\Delta \in \Lambda} \delta_\Delta$ We have to show that

$$\vdash \delta_\Lambda \to E_B(\psi \to \delta_\Lambda) \tag{1}$$

Observe that if Λ is empty, then it follows trivially, because an empty disjunction is equivalent to a contradiction.

Otherwise note that for every $a \in B$, for every $\Delta \in \Lambda$ and every $\Delta' \in \overline{\Lambda}$ (where $\overline{\Lambda}$ is the complement of Λ) either $\psi \notin \Delta'$, or there is a formula $\varphi_{\Delta\Delta'}$ such that $\Box_a \varphi_{\Delta\Delta'} \in \Delta$ and $\varphi_{\Delta\Delta'} \notin \Delta'$. From this it follows in both cases that

$$\vdash \delta_\Lambda \to E_B(\psi \to \neg\delta_{\overline{\Lambda}})$$

It can also be shown that $\vdash \delta_\Lambda \vee \delta_{\overline{\Lambda}}$, and therefore we get (1). By necessitation we get

$$\vdash C_B(\psi, \delta_\Lambda \to E_B(\psi \to \delta_\Lambda))$$

By applying the induction axiom we can deduce

$$\vdash E_B(\psi \to \delta_\Lambda) \to C_B(\psi, \delta_\Lambda)$$

Given that $\vdash \delta_\Lambda \to \chi$, we get

$$\vdash E_B(\psi \to \delta_\Lambda) \to C_B(\psi, \chi)$$

It is also the case that $\vdash \delta_\Gamma \to E_B(\psi \to \delta_\Lambda)$. Therefore $C_B(\psi, \chi) \in \Gamma$. □

The completeness theorem follows in a straightforward way from this lemma.

Theorem 9 (Completeness for EL-RC) $\models \varphi$ *iff* $\vdash \varphi$.

Proof Let $\nvdash \varphi$, i.e. $\neg\varphi$ is consistent. One easily finds a maximally consistent set Γ in the closure of $\neg\varphi$ with $\neg\varphi \in \Gamma$, as only finitely many formulas matter. By the Truth Lemma, $M_{\neg\varphi}, \Gamma \models \neg\varphi$, i.e., $M_{\neg\varphi}, \Gamma \not\models \varphi$.

The soundness of the proof system can easily be shown by induction on the length of proofs, and we do not provide its straightforward details here. □

3.5 Reduction Axioms for PAL-RC

Next, let PAL-RC be the dynamic epistemic logic with both relativized common knowledge and public announcements. Its semantics combines those for PAL and EL-RC. We want to find a reduction axiom for $[\varphi]C_B(\psi, \chi)$, the formula that expresses that after public announcement of φ, every ψ path leads to a χ world. Note that $[\varphi]C_B(\psi, \chi)$ holds exactly in those worlds where every $\varphi \wedge [\varphi]\psi$ path ends in a world where $[\varphi]\chi$ is true. This observation yields the following proof system for PAL-RC:

Definition 10 (Proof System for PAL-RC) *The proof system for* PAL-RC *is that for* EL-RC *plus the reduction axioms for* PAL*, together with* **C-Red***:*

$$[\varphi]C_B(\psi, \chi) \leftrightarrow (\varphi \rightarrow C_B(\varphi \wedge [\varphi]\psi, [\varphi]\chi)) \quad \textit{(common knowledge reduction)}$$

as well as an inference rule of necessitation for all announcement modalities.

It turns out that PAL-RC is no more expressive than EL-RC by a direct translation, where the translation clause for $[\varphi]C_B(\psi, \chi)$ relies on the above insight:

Definition 11 (Translation from PAL-RC to EL-RC) *The function t takes a formula from the language of* PAL-RC *and yields a formula in the language of* EL-RC*.*

$$\begin{array}{llll}
t(p) & = p & t([\varphi]p) & = t(\varphi) \rightarrow p \\
t(\neg\varphi) & = \neg t(\varphi) & t([\varphi]\neg\psi) & = t(\varphi) \rightarrow \neg t([\varphi]\psi) \\
t(\varphi \wedge \psi) & = t(\varphi) \wedge t(\psi) & t([\varphi](\psi \wedge \chi)) & = t([\varphi]\psi) \wedge t([\varphi]\chi) \\
t(\Box_a \varphi) & = \Box_a t(\varphi) & t([\varphi]\Box_a \psi) & = t(\varphi) \rightarrow \Box_a t([\varphi]\psi) \\
t(C_B(\varphi, \psi)) & = C_B(t(\varphi), t(\psi)) & t([\varphi]C_B(\psi, \chi)) & = C_B(t(\varphi) \wedge t([\varphi]\psi), t([\varphi]\chi)) \\
 & & t([\varphi][\psi]\chi) & = t([\varphi]t([\psi]\chi))
\end{array}$$

The translation induced by these principles can be formulated as an inside-out procedure, replacing innermost dynamic operators first. To see that it terminates, we can define a notion of complexity on formulas such that the complexity of the formulas is smaller on the right hand side. We have added the final axiom here for its independent interest, even though it is not strictly necessary for this procedure. As observed in [4], it describes the effect of *sequential composition* of announcements, something which can also be stated as an independently valid law of public announcement saying that the effect of first announcing φ and then ψ is the same as announcing one single assertion, viz. the conjunction $\varphi \wedge [\varphi]\psi$. Standard programming styles for performing the reduction (cf. [9]) do include the final clause in any case. As to its admissibility, note that, when the last clause is called, the innermost application will yield a formula of lesser complexity. The following theorems can be proved by induction on this complexity measure.

Theorem 12 (Translation Correctness) *For each dynamic-epistemic formula φ of* PAL-RC *and each semantic model M, w,*

$$M, w \models \varphi \text{ iff } M, w \models t(\varphi).$$

Theorem 13 ('PAL-RC = EL-RC') *The languages* PAL-RC *and* EL-RC *have equal expressive power.*

Theorem 14 (Completeness for PAL-RC) $\models \varphi$ *iff* $\vdash \varphi$.

Proof The proof system for EL-RC is complete (Theorem 9), and every formula in $\mathscr{L}_{\mathsf{PAL\text{-}RC}}$ is provably equivalent to its translation in $\mathscr{L}_{\mathsf{EL\text{-}RC}}$, given the reduction axioms. □

3.6 Model Comparison Games for EL-RC

The notion of relativized common knowledge is of independent interest, just as irreducibly binary general quantifiers (such as *Most A are B*) lead to natural completions of logics with only unary quantifiers. It is important to investigate the relation between the logic of epistemic logic with relativized common knowledge with public announcement logic with common knowledge. We provide some more information through characteristic games. Model comparison games for languages with individual modalities are well-known, but dealing with common knowledge: i.e., arbitrary finite iterations, requires some nice twists. These games will be used in the next section to investigate the expressivity of EL-RC relative to PAL-C.

Definition 15 (The EL-RC Game) *Let two epistemic models $M = (W, R, V)$ and $M' = (W', R', V')$ be given. Starting from each $w \in W$ and $w' \in W'$, the n-*round EL-RC game *between Spoiler and Duplicator is given as follows. If $n = 0$ Spoiler wins if w and w' differ in their atomic properties, otherwise Du-*

plicator wins. Otherwise Spoiler can initiate one of the following two scenarios in each round:

$\Box_a$***-move*** *Spoiler chooses a point x in one model which is an a-successor of the current w or w'. Duplicator responds with a matching successor y in the other model. The output is x, y.*

*RC_B**-move** Spoiler chooses a B-path $x_0 \dots x_k$ in either of the models with x_0 the current w or w'. Duplicator responds with a B-path $y_0 \dots y_m$ in the other model, with $y_0 = w'$. Then Spoiler can (a) make the end points x_k, y_m the output of this round, or (b) he can choose a world y_i (with $i > 0$) on Duplicator's path, and Duplicator must respond by choosing a matching world x_j (with $j > 0$) on Spoilers path, and x_j, y_i becomes the output.*

The game continues with the new output states. If these differ in their atomic properties, Spoiler wins — otherwise, a player loses whenever he cannot perform a move while it is his turn. If Spoiler has not won after all n rounds, Duplicator wins the whole game.

Definition 16 (Modal Depth) *The* modal depth *of a formula is defined by:*

$$d(\bot) = d(p) = 1$$
$$d(\neg\varphi) = d(\varphi)$$
$$d(\varphi \wedge \psi) = \max(d(\varphi), d(\psi))$$
$$d(\Box_a\varphi) = d(\varphi) + 1$$
$$d(C_B(\varphi, \psi)) = \max(d(\varphi), d(\psi)) + 1$$

If two models M, w and M', w' have the same theory up to depth n, we write $M, w \equiv_n M', w'$.

The following result holds for all logical languages that we use in this paper. Recall that our stock of propositional letters is finite.

Lemma 17 (Propositional finiteness) *For every n, up to modal depth n, there are only finitely logically non-equivalent propositions.*

Theorem 18 (Adequacy of the EL-RC Game) *Duplicator has a winning strategy for the n-round game from M, w, M', w' iff $M, w \equiv_n M', w'$.*

Proof The proof is by induction on n. The base case is obvious, and all inductive cases are also standard in modal logic, except that for relativized common knowledge. As usual, perspicuity is increased somewhat by using the dual existential modality $\hat{C}_B(\varphi, \psi)$. From left to right the proof is straightforward.

From right to left. Suppose that $M, w \equiv_{n+1} M', w'$. A winning strategy for

Duplicator in the $(n+1)$-round game can be described as follows. If Spoiler makes an opening move of type [$\Box_a$-move], then the usual modal argument works. Next, suppose that Spoiler opens with a finite sequence in one of the models: say M, without loss of generality. By the Lemma 17, we know that there is only a finite number of complete descriptions of points up to logical depth n, and each point s in the sequence satisfies one of these: say $\Delta(s,n)$. In particular, the end point v satisfies $\Delta(v,n)$. Let $\Delta(n)$ be the disjunction of all formulas $\Delta(s,n)$ occurring on the path. Then, the initial world w satisfies the following formula of modal depth $n+1$: $\hat{C}_B(\Delta(n), \Delta(v,n))$. By our assumption, we also have $M', w' \models \hat{C}_B(\Delta(n), \Delta(v,n))$. But any sequence witnessing this by the truth definition is a response that Duplicator can use for her winning strategy. Whatever Spoiler does in the rest of this round, Duplicator always has a matching point that is n-equivalent in the language. $\Box$

Thus, games for $\mathscr{L}_{\mathsf{EL\text{-}RC}}$ are straightforward. But it is also of interest to look at the language $\mathscr{L}_{\mathsf{PAL\text{-}C}}$. Here, the shift modality $[\varphi]$ passing to definable submodels requires a new type of move, not found in ordinary Ehrenfeucht games, where players can decide to change the current model. The following description of what happens is 'modular': a model changing move can be added to model comparison games for ordinary epistemic logic (perhaps with common knowledge), or for our EL-RC game. By way of explanation: we let Spoiler propose a model shift. Players first discuss the 'quality' of that shift, and Duplicator can win if it is deficient; otherwise, the shift really takes place, and play continues within the new models. This involves a somewhat unusual sequential composition of games, but perhaps one of independent interest.

Definition 19 (The PAL-C Game) *Let the setting be the same as for the n-round game in Definition 15. Now Spoiler can initiate one of the following scenario's each round*

$\Box_a$***-move*** *Spoiler chooses a point x in one model which is an a-successor of the current w or w', and Duplicator responds with a matching successor y in the other model. The output of this move is x, y.*

C_B***-move*** *Spoiler chooses a point x in one model which is reachable by a B-path from w or w', and Duplicator responds by choosing a matching world y in the other model. The output of this move is x, y.*

$[\varphi]$***-move*** *Spoiler chooses a number $r < n$, and sets $S \subseteq W$ and $S' \subseteq W'$, with the current $w \in S$ and likewise $w' \in S'$.* Stage 1*: Duplicator chooses states s in $S \cup S'$, $\overline{s}$ in $\overline{S} \cup \overline{S'}$ (where $\overline{S}$ is the complement of S). Then Spoiler and Duplicator play the r-round game for these worlds. If Duplicator wins this subgame, she wins the n-round game.* Stage 2*: Otherwise, the game continues in the relativized models $M|S, w$ and $M'|S', w'$ over $n - r$ rounds.*

The definition of depth is extended to formulas $[\varphi]\psi$ as $d([\varphi]\psi) = d(\varphi) + d(\psi)$.

Theorem 20 (Adequacy of the PAL-RC Game) *Duplicator has a winning strategy for the n-round game on M, w and M', w' iff $M, w \equiv_n M', w'$ in $\mathscr{L}_{\mathsf{PAL\text{-}RC}}$.*

Proof We only discuss the inductive case demonstrating the match between announcement modalities and model-changing steps. From left to right, the proof is straightforward.

From right to left. Suppose that M, w and M', w' are equivalent up to modal depth $n+1$. We need to show that Duplicator has a winning strategy. Consider any opening choice of S, S' and $r < n + 1$ made by Spoiler. *Case 1*: Suppose there are two points s, $\overline{s}$ that are equivalent up to depth r. By the induction hypothesis, this is the case if and only if Duplicator has a winning strategy for the r-round game starting from these worlds and so has a winning strategy in Stage 1. *Case 2*: Duplicator has no such winning strategy, which means that Spoiler has one — or equivalently by the inductive hypothesis, every pair $s, \overline{s}$ is distinguished by some formula $\varphi_{s\overline{s}}$ of depth at most r which is true in s and false in $\overline{s}$. Observe that $\delta_s = \bigwedge_{\overline{s} \in \overline{S} \cup \overline{S'}} \varphi_{s\overline{s}}$ is true in s and false in $\overline{S} \cup \overline{S'}$. Note that there can be infinitely many worlds involved in the comparison, but finitely many different formulas will suffice by the Lemma 17, which also holds for this extended language. Further, the formula $\Delta_S = \bigvee_{s \in S} \delta_s$ is true in S and false in $\overline{S} \cup \overline{S'}$. A formula $\Delta_{S'}$ is found likewise, and we let Δ be $\Delta_{S'} \vee \Delta_S$. It is easy to see that Δ is of depth r and defines S in M and S' in M'. Now we use the given language equivalence between M, w and M', w' with respect to all depth $(n + 1)$-formulas $\langle \Delta \rangle \psi$ where ψ runs over all formulas of depth $(n + 1) - r$. We can conclude that $M|\Delta, w$ and $M'|\Delta, w'$ are equivalent up to depth $(n + 1) - r$, and hence Duplicator has a winning strategy for the remaining game, by the inductive hypothesis. So in this case Duplicator has a winning strategy in Stage 2. □

In the next section we will use this game to show that EL-RC is more expressive than PAL-C. For now, we will give an example of how this game can be played.

Definition 21 *Let the model $M(n) = (W, R, V)$ be defined by*

- $W = \{x \in \mathbb{N} \mid 0 \leq x \leq n\}$
- $R = \{(x, x - 1) \mid 1 \leq x \leq n\}$
- $V(p) = W$

These models are simply lines of worlds. They can all be seen as submodels of the entire line of natural numbers (where $W = \mathbb{N}$). The idea is that Spoiler cannot distinguish two of these models if the line is long enough. The only hope that Spoiler has is to force one of the current worlds to an endpoint and the other not to be an endpoint. In that case Spoiler can make a $\Box$-move in the world that is not an endpoint and Duplicator is stuck. This will not succeed if the lines are long enough. Note that a C-move does not help Spoiler. Also

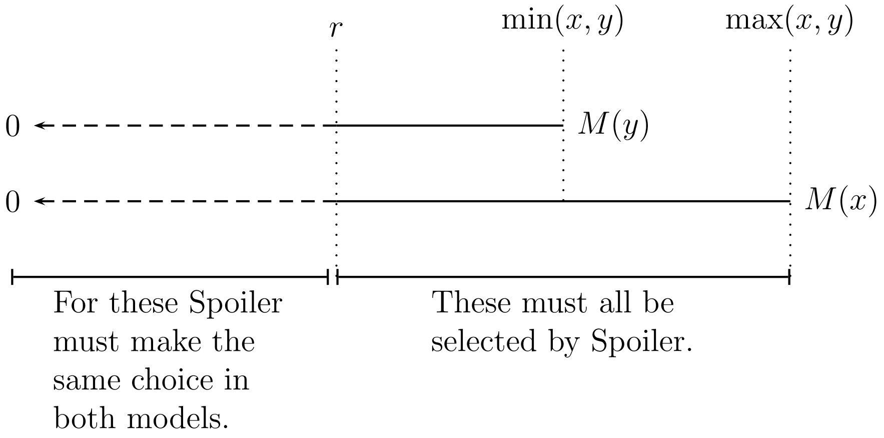

r
min(x, y)
max(x, y)
0
$M(y)$
0
$M(x)$
For these Spoiler must make the same choice in both models.
These must all be selected by Spoiler.

Fig. 1. Illustration of the proof of Lemma 22.

a $[\varphi]$-move will not help Spoiler. Such a move will shorten the lines, but that will cost as many rounds as it shortens them, so Spoiler still loses if they are long enough. The following Lemma captures this idea.

Lemma 22 *For all m, n, and all $x \leq m$ and $y \leq n$ Duplicator has a winning for the* PAL-C *game for $M(m), x$ and $M(n), y$ with at most* $\min(x, y)$ *rounds.*

Proof If $x = y$, the proof is trivial. We proceed by induction on the number of rounds. Suppose the number of rounds is 0. Then x and y only have to agree on propositional variables. They must agree, since p is true everywhere.

Suppose that the number of rounds is $k+1$ (i.e. $\min(x, y) = k+1$). Duplicator's strategy is the following. If Spoiler chooses to play a $\Box$-move, he moves to $x-1$ (or to $y-1$). Duplicator responds by choosing $y-1$ (or $x-1$). Duplicator has a winning strategy for the resulting subgame by the induction hypothesis.

Suppose Spoiler chooses to play a C-move. If Spoiler chooses a $z < \min(x, y)$, then Duplicator also chooses z. Otherwise, Duplicator takes just one step (the minimum she is required to do). Duplicator has a winning strategy for the resulting subgame by the induction hypothesis.

Suppose Spoiler chooses to play a $[\varphi]$-move. Spoiler chooses a number of rounds r and some S and S'. Observe that for all $z < \min(x, y)$ it must be the case that $z \in S$ iff $z \in S'$. Otherwise, Duplicator has a winning strategy by the induction hypothesis by choosing z and z. Moreover for all $z \geq r$ it must be the case that $z \in S \cup S'$. Otherwise, Duplicator has a winning strategy by the induction hypothesis for $\min(x, y)$ and z. In Stage 2 the resulting subgame will be for two models bisimilar to models to which the induction hypothesis applies. The number of rounds will be $(k+1)-r$, and the lines will be at least $\min(x, y) - r$ long (and $(\min(x, y) = k + 1)$. This is sketched in Figure 1. $\Box$

$$\begin{tikzcd}
\mathsf{EL\text{-}RC} \arrow[r, leftrightarrow] & \mathsf{PAL\text{-}RC} \\
\mathsf{EL\text{-}C} \arrow[u] \arrow[r] & \mathsf{PAL\text{-}C} \arrow[u] \\
\mathsf{EL} \arrow[u] \arrow[r, leftrightarrow] & \mathsf{PAL} \arrow[u]
\end{tikzcd}$$

Fig. 2. Expressive power of static and dynamic epistemic logics.

3.7 Expressivity Results

In this section we investigate the expressive power of the logics under consideration here. Reduction axioms and the accompanying translation tell us that two logics have equal expressive power. But our inability to find a compositional translation from one logic to another does not imply that those logics have different expressive power. Here are some known facts. Epistemic logic with common knowledge is more expressive than epistemic logic without common knowledge. In [1] it was shown that public announcement logic with common knowledge is more expressive than epistemic logic with common knowledge. This can also be shown using the results on PDL in [15]. In this section we show that relativized common knowledge logic is more expressive than public announcement logic with common knowledge. The landscape of expressive power is summarized in Figure 2. All arrows are strict.

In general one logic L is more expressive than another logic L′ (L′ ⟶ L in Figure 2) if there is a formula in the language of L which is not equivalent to any formula in the language of L′ (and every formula in the language of L′ is equivalent to some formula in the language of L). So, in order to show that EL-RC is more expressive than PAL-C we need to find a formula in $\mathscr{L}_{\mathsf{EL\text{-}RC}}$ which is not equivalent to any formula in $\mathscr{L}_{\mathsf{PAL\text{-}C}}$. The formula

$$C(p, \neg\Box p)$$

fits this purpose. This will be shown in Theorem 27.

We can show that this formula cannot be expressed in $\mathscr{L}_{\mathsf{PAL\text{-}C}}$ by using model comparison games. We will show that for any number of rounds there are two models such that Duplicator has a winning strategy for the model comparison game, but $C(p, \neg\Box p)$ is true in one of these models and false in the other.

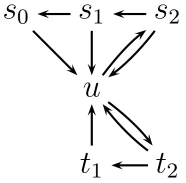
In Definition 25 we provide the models that EL-RC can distinguish, but PAL-C cannot. Since the model comparison game for PAL-C contains the $[\varphi]$-move, we also need to prove that the relevant submodels cannot be distinguished by

PAL-C. We deal with these submodels first in the next definition and lemma.

Definition 23 *Let the model $M(m,n) = (W,R,V)$ where $0 < n \leq m$ be defined by*

- $W = \{s_x \mid 0 \leq x \leq m\} \cup \{t_x \mid n \leq x \leq m\} \cup \{u\}$
- $R = \{(s_x, s_{x-1}) \mid 1 \leq x \leq m\} \cup \{(t_x, t_{x-1}) \mid n+1 \leq x \leq m\} \cup \{(w,u) \mid w \in W \setminus \{u\}\} \cup \{(u, s_m), (u, t_m)\}$
- $V(p) = W \setminus \{u\}$

The picture below represents $M(2,1)$.



Let us call these models 'hourglasses'. The idea is that Spoiler cannot distinguish the top line from the bottom line of these models if they are long enough. Note that apart from u this model consists of two lines. So if Spoiler plays $\Box$-moves on these lines, Duplicator's strategy is the same as for the line models described above. If he moves to u, Duplicator also moves to u, and surely Duplicator cannot lose the subsequent game in that case. In these models a C-move is very bad for Spoiler, since all worlds are connected by the reflexive transitive closure of R. A $[\varphi]$-move will either yield two lines which are too long, or it will be a smaller hourglass model, which will still be too large, since the $[\varphi]$-move reduces the number of available moves. The Lemma below captures this idea.

In what follows, w_x is a variable ranging over s_x and t_x. And if t_x does not exist it refers to s_x.

Lemma 24 *For all m, n and all $x \leq m$ and $y \leq m$ Duplicator has a winning strategy for the public announcement game for $M(m,n), w_x$ and $M(m,n), w_y$ with at most $\min(x,y) - n$ rounds.*

Proof We prove the case when $w_x = s_x$ and $w_y = t_y$ (the other cases are completely analogous) by induction. Suppose the number of rounds is 0. Then s_x and t_y only have to agree on propositional variables. They do agree, since p is true in both.

Suppose that the number of rounds is $k+1$. Duplicator's winning strategy is the following. If Spoiler chooses to play a $\Box$-move, he moves to s_{x-1} (or to t_{y-1}), or to u. In the last case Duplicator responds by also choosing u, and has a winning strategy for the resulting subgame. Otherwise Duplicator moves to the t_{y-1} (or s_{x-1}). Duplicator has a winning strategy for the resulting subgame by the inductive hypothesis.

Suppose Spoiler plays a C-move. If Spoiler moves to w, then Duplicator moves to the same w, and has a winning strategy for the resulting subgame.

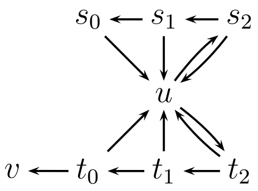
Suppose Spoiler chooses to play a $[\varphi]$-move. Spoiler chooses a number of rounds r and some S. Since there is only one model, Spoiler only chooses one subset of W. Moreover for all $z \geq \min(x, y) - n - r$ it must be the case that $w_z \in S$. Otherwise, Duplicator has a winning strategy by the induction hypothesis for s_x and w_z. In Stage 2 the result will be two models bisimilar to models to which the inductive hypothesis applies, or to which Lemma 22 applies. $\Box$

Lastly consider the following class of models.

Definition 25 *Let the model* $M^+(m, n) = (W, R, V)$ *where* $0 < n \leq m$ *be defined by*

- $W = \{s_x \mid n \leq x \leq m\} \cup \{t_x \mid 0 \leq x \leq m\} \cup \{v, u\}$
- $R = \{(s_x, s_{x-1}) \mid n+1 \leq x \leq m\} \cup \{(t_x, t_{x-1}) \mid 1 \leq x \leq m\} \cup$
  $\{(w, u) \mid w \in W \setminus \{v, u\}\} \cup \{(t_0, v)\} \cup \{(u, s_m), (u, t_m)\}$
- $V(p) = W \setminus \{u\}$

The picture below represents $M^+(2, 0)$.



In these 'hourglasses with an appendage', the idea is that Duplicator cannot distinguish the top line from the bottom line of these models when they are long enough. Apart from v, the model is just like a hourglass. So the only new option for Spoiler is to force one of the current worlds to v, and the other to another world. Then Spoiler chooses a $\Box$-move and takes a step from the non-v world and Duplicator is stuck at v. However if the model is large enough v is too far away. Again a C-move does not help Spoiler, because it can be matched exactly by Duplicator. Reducing the model with a $[\varphi]$-move will yield either a hourglass (with or without an appendage) or two lines, for which Spoiler does not have a winning strategy. This idea leads to the following Lemma.

Lemma 26 *For all* m, n *and all* $x \leq m$ *and* $y \leq m$ *Duplicator has a winning strategy for the public announcement game for* $M^+(m, n), w_x$ *and* $M^+(m, n)$, w_y *with at most* $\min(x, y) - n$ *rounds.*

Proof The proof is analogous to the proof of Lemma 24. □

This Lemma now yields to the following theorem.

Theorem 27 EL-RC *is more expressive than* PAL-C.

Proof Suppose PAL-C is just as expressive as EL-RC. Then there is a formula $\varphi \in \mathscr{L}_{\mathsf{PAL\text{-}C}}$ with $\varphi \equiv C(p, \neg\Box p)$. Suppose $d(\varphi) = n$. In that case we would have $M^+(n, 0), s_n \models \varphi$ and $M^+(n, 0), t_n \not\models \varphi$, contradicting Lemma 26. Hence, EL-RC is more expressive. □

3.8 Complexity Results

Update logics are about processes that manipulate information, and hence they raise natural questions of *complexity*, as a counterpoint to the expressive power of communication and observation scenarios. In particular, all of the usual complexity questions concerning a logical system make sense. *Model checking* asks where a given formula is true in a model, and this is obviously crucial to computing updates. *Satisfiability testing* asks when a given formula has a model, which corresponds to consistency of conversational scenarios in our dynamic epistemic setting. Or, stating the issue in terms of *validity*: when will a given epistemic update always produce some global specified effect? Finally, just as in basic modal logic, there is a non-trivial issue of *model comparison*: when do two given models satisfy the same formulas in our language, i.e., when are two group information states 'the same' for our purposes? As usual, this is related with checking for *bisimulation*, or in a more finely-grained version, the existence of winning strategies for Duplicator in the above model comparison games.

Now technically, the translation of Definition 11 combined with known algorithms for model checking, satisfiability, validity, or model comparison for epistemic logic yield similar algorithms for public announcement logic. But, in a worst case, the length of the translation of a formula is exponential in the length of the formula. E.g., the translation of φ occurs three times in that of $[\varphi]C_B(\psi, \chi)$. Therefore, a direct complexity analysis is worth-while. We provide two results plus some references.

Lemma 28 *Deciding whether a finite model M, w satisfies $\varphi \in \mathscr{L}_{\mathsf{EL\text{-}RC}}$ is computable in polynomial time in the length of φ and the size of M.*

Proof The argument is an easy adaptation of the usual proof for PDL or common knowledge with common knowledge: see [16, p.202] and [13, p.91]. □

This algorithm does not suffice for the case with public announcements. The truth values of φ and ψ in the given model do not fix that of $[\varphi]\psi$. We must also know the value of ψ in the model restricted to φ worlds.

Lemma 29 *Deciding whether a finite model M, w satisfies $\varphi \in \mathscr{L}_{\mathsf{PAL\text{-}RC}}$ is computable in polynomial time in the length of φ and the size of M.*

Proof Again there are at most $|\varphi|$ subformulas of φ. Now we make a binary tree of these formulas which splits with formulas of the form $[\psi]\chi$. On the left subtree all subformulas of ψ occur, on the right all those of χ. This tree can be constructed in time $\mathcal{O}(|\varphi|)$. Labeling the model is done by processing this tree from bottom to top from left to right. The only new case is when we encounter a formula $[\psi]\chi$. In that case we have already processed the left subtree for ψ. Now we first label those worlds where ψ does not hold as worlds where $[\psi]\chi$ holds, then we process the right subtree under $[\psi]\chi$ where we restrict the model to worlds labeled as ψ-worlds. After this process we label those worlds that were labeled with χ as worlds where $[\psi]\chi$ holds and the remaining as worlds where it does not hold. We can see by induction on formula complexity that this algorithm is correct.

Also by induction on φ, this algorithm takes time $\mathcal{O}(|\varphi| \times \|M\|^2)$. The only difficult step is labeling the model with $[\psi]\chi$. By the induction hypothesis, restricting the model to ψ takes time $\mathcal{O}(| \psi | \times \|M\|^2)$. We simply remove (temporarily) all worlds labelled $\neg\varphi$ and all arrows pointing to such worlds. Again by the induction hypothesis, checking χ in this new model takes $\mathcal{O}(|\psi| \times \|M\|^2)$ steps. The rest of the process takes $\|M\|$ steps. So, this step takes over-all time $\mathcal{O}(|[\psi]\chi| \times \|M\|^2)$. □

Moving on from model checking, the satisfiability and the validity problem of epistemic logic with common knowledge are both known to be EXPTIME-complete. In fact, this is true for almost any logic that contains a transitive closure modality. Satisfiability and validity for PDL are also EXPTIME-complete. Now there is a linear time translation of the language of EL-RC to that of PDL. Therefore the satisfiability and validity problems for EL-RC are also EXPTIME-complete. For PAL-RC and even PAL-C, however, the complexity of satisfiability and validity is not settled by this. Lutz [20] shows that satisfiability in PAL is PSPACE-complete, using a polynomial-time translation from dynamic-epistemic to purely epistemic formulas. The latter is unlike the translation underpinning our reduction axioms, in that it is not meaning-preserving. The same method probably extends to PAL-C and PAL-RC.

Finally, the complexity of model comparison for finite models is the same as that for ordinary epistemic logic, viz. PTIME. The reason is that even basic modal equivalence on finite models implies the existence of a bisimulation, while all our extended languages are bisimulation-invariant.

This completes our in-depth analysis of a redesigned dynamic logic of public announcement. Having shown the interest of such a system, we now consider more powerful versions, covering a much wider range of phenomena.

4 A New Logic of Communication and Change

The examples in the Section 2 set a high ambition level for a dynamic epistemic logic updating with events involving both communication and actual change. As we explained, systems of this general sort were proposed in [1,2], but without reduction axioms for common knowledge. We now proceed to a version which can deal with common knowledge, generalizing the compositional methodology for epistemic logic with announcements of Section 3.

In Section 4.1 we introduce *update models* and specify their execution on epistemic models in terms of 'product update'. The only difference with the references above is our addition of fact-changing actions by substitutions. In Section 4.2 we review propositional dynamic logic (PDL) under its epistemic/doxastic interpretation, written henceforth as E-PDL. In Section 4.3 we then present our dynamic epistemic *logic of communication and change* LCC as an extension of E-PDL with dynamic modalities for update models. In Section 4.4 we show that LCC is in harmony with E-PDL through a semantic analysis of epistemic postconditions, and in Section 4.5 we present a proof system for LCC in terms of reduction axioms based on this insight [11,10].

From a technical perspective, the proofs to follow are not just simple generalizations of those for public announcements. In [18] a correspondence between update models and *finite automata* is used to obtain reduction axioms in a dynamic epistemic logic based on so-called 'automata PDL', a variant of E-PDL. Our main new idea here is that this can be stream-lined by analyzing the automata inductively inside E-PDL itself [11], using the well-known proof of Kleene's theorem, stating that languages generated by nondeterministic finite automata are regular [17]. The main theorems to follow use an inductive 'program transformation' approach to epistemic updates whose structure resembles that of Kleene's translation from finite automata to regular expressions. This technique for deriving compositional reduction axioms may be of independent interest beyond the present setting.

4.1 Update Models and their Execution

When viewed by themselves, communicative or other information-bearing scenarios are similar to static epistemic models, in that they involve a space of possible events and agents' abilities to distinguish between these. In [1] this observation is used as the engine for general update of epistemic models under epistemic actions. In particular, individual events come with *preconditions* holding only at those worlds where they can occur.

Of course, events normally do not just signal information, they also change the world in more concrete ways. Before describing the update mechanism, we enrich our dynamic models with postconditions for events that really change the world. For this purpose we use 'substitutions' that effect changes in valuations at given worlds, serving as *postconditions* for events to occur.

Definition 30 (Substitutions) *$\mathscr{L}$ substitutions are functions of type $\mathscr{L} \to \mathscr{L}$ that distribute over all language constructs, and that map all but a finite number of basic propositions to themselves. $\mathscr{L}$ substitutions can be represented as sets of bindings*

$$\{p_1 \mapsto \varphi_1, \dots, p_n \mapsto \varphi_n\}$$

where all the p_i are different . If σ is a $\mathscr{L}$ substitution, then the set $\{p \in P \mid \sigma(p) \neq p\}$ is called its domain*, notation $dom(\sigma)$. Use ϵ for the identity substitution. Let $SUB_{\mathscr{L}}$ be the set of all $\mathscr{L}$ substitutions.*

Definition 31 (Epistemic Models under a Substitution) *If $M = (W, V, R)$ is an epistemic model and σ is a $\mathscr{L}$ substitution (for an appropriate epistemic language $\mathscr{L}$), then V_M^σ is the valuation given by $\lambda p \cdot [\![\sigma(p)]\!]^M$. In other words, V_M^σ assigns to w the set of worlds w in which $\sigma(p)$ is true. For $M = (W, V, R)$, call M^σ the model given by (W, V_M^σ, R).*

Definition 32 (Update Models) *An update model for a finite set of agents N with a language $\mathscr{L}$ is a quadruple $\mathsf{U} = (\mathsf{E}, \mathsf{R}, \mathsf{pre}, \mathsf{sub})$ where*

- $\mathsf{E} = \{\mathsf{e}_0, \dots, \mathsf{e}_{n-1}\}$ *is a finite non-empty set of events,*
- $\mathsf{R} : N \to \wp(\mathsf{E}^2)$ *assigns an accessibility relation $\mathsf{R}(a)$ to each agent $a \in N$.*
- $\mathsf{pre} : \mathsf{E} \to \mathscr{L}$ *assigns a precondition to each event,*
- $\mathsf{sub} : \mathsf{E} \to SUB_{\mathscr{L}}$ *assigns a $\mathscr{L}$ substitution to each event.*

A pair U, e is an update model with a distinguished actual event $\mathsf{e} \in \mathsf{E}$.

In these definitions, $\mathscr{L}$ can be any language that can be interpreted in the models of Definition 1. Note that an 'action model' in the sense of [1] is a special update model in our sense, where sub assigns the identity substitution ϵ to every event. Our substitutions then take the original action model philosophy one step further. In particular, our notion of update execution will reset both basic features of information models: epistemic accessibility relations, but also the propositional valuation. Section 4.3 then presents a dynamic logic for communication and real change based on these general update models.

Remark: Information Change and Real Change Note that the world-changing feature is modular here. Readers only interested in epistemic information flow can think of models in the original dynamic epistemic style. All of our major results hold in this special case, and proofs proceed by merely skipping base steps or inductive steps involving the substitutions.

Executing an update is now modeled by the following *product construction.*

Definition 33 (Update Execution) *Given a static epistemic model* $M = (W, R, V)$*, a world* $w \in W$*, an update model* $\mathsf{U} = (\mathsf{E}, \mathsf{R}, \mathsf{pre}, \mathsf{sub})$ *and an action state* $\mathsf{e} \in \mathsf{E}$ *with* $M, w \models \mathsf{pre}(\mathsf{e})$*, we say that the result of* executing U, e in M, w *is the model* $M \circ \mathsf{U}, (w, \mathsf{e}) = (W', R', V'), (w, \mathsf{e})$ *where*

- $W' = \{(v, \mathsf{f}) \mid M, v \models \mathsf{pre}(\mathsf{f})\}$,
- $R'(a) = \{((v, \mathsf{f}), (u, \mathsf{g})) \mid (v, u) \in R(a)$ *and* $(\mathsf{f}, \mathsf{g}) \in \mathsf{R}(a)\}$,
- $V'(p) = \{(v, \mathsf{f}) \mid M, v \models \mathsf{sub}(\mathsf{f})(p)\}$

Once again, Definitions 32 (with all substitutions set equal to ϵ) and 33 provide a semantics for the logic of epistemic actions LEA of [1]. The basic epistemic language $\mathscr{L}_{\mathsf{LEA}}$ can then be extended with dynamic modalities $[\mathsf{U}, \mathsf{e}]\varphi$, where a U is any *finite update model* for $\mathscr{L}_{\mathsf{LEA}}$. These say that 'every execution of U, e yields a model where φ holds':

$$M, w \models [\mathsf{U}, \mathsf{e}]\varphi \text{ iff } M, w \models \mathsf{pre}(\mathsf{e}) \text{ implies that } M \circ \mathsf{U}, (w, \mathsf{e}) \models \varphi$$

In [1] an axiomatic system for LEA is presented with a, somewhat complicated, completeness proof, without reduction axioms for common knowledge. Our analysis to follow will improve on this.

To see what is needed, observe that, again, the semantic intuition about the crucial case $M, w \models [\mathsf{U}, \mathsf{e}]C_B\varphi$ is clear. It says that, if there is a B-path $w_0, \ldots, w_n$ (with $w_0 = w$) in the static model and a matching B-path $\mathsf{e}_0, \ldots, \mathsf{e}_n$ (with $\mathsf{e}_0 = \mathsf{e}$) in the update model with $M, w_i \models \mathsf{pre}(\mathsf{e}_i)$ for all $i \leq n$, then $M, w_n \models \varphi$. To express all this in the initial static model, it turns out to be convenient to choose a representation of complex epistemic assertions that meshes well with update models.

Now, the relevant finite paths in static models involve strings of agent accessibility steps and tests on formulas. And these finite traces of actions and tests are precisely the sort of structure whose study led to the design of *propositional dynamic logic* (PDL). Initially, PDL was designed for the analysis of programs. In what follows, however, we will give it an epistemic interpretation.

The language of propositional dynamic logic and all further information about its semantics and proof theory may be found in [16], which also has references to the history of this calculus, and its original motivations in computer science. We briefly recall some major notions and results.

Definition 34 (PDL, Language) *Let a set of propositional variables P and a set of relational atoms N be given, with p ranging over P and a over N. The language of* PDL *is given by:*

$$\begin{aligned} &\varphi ::= \top \mid p \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid [\pi]\varphi \\ &\pi ::= a \mid ?\varphi \mid \pi_1;\pi_2 \mid \pi_1 \cup \pi_2 \mid \pi^* \end{aligned}$$

We employ the usual abbreviations: $\bot$ is shorthand for $\neg\top$, $\varphi_1 \vee \varphi_2$ is shorthand for $\neg(\neg\varphi_1 \wedge \neg\varphi_2)$, $\varphi_1 \rightarrow \varphi_2$ is shorthand for $\neg(\varphi_1 \wedge \varphi_2)$, $\varphi_1 \leftrightarrow \varphi_2$ is shorthand for $(\varphi_1 \rightarrow \varphi_2) \wedge (\varphi_2 \rightarrow \varphi_1)$, and $\langle\pi\rangle\varphi$ is shorthand for $\neg[\pi]\neg\varphi$.

Definition 35 (PDL, Semantics) *The semantics of* PDL *over P, N is given in models $M = (W, R, V)$ for signature P, N. Formulas of* PDL *are interpreted as subsets of W, relational atoms a as binary relations on W (with the interpretation of relational atoms a given as $R(a)$), as follows:*

$$\begin{aligned} [\![\top]\!]^M &= W \\ [\![p]\!]^M &= V(p) \\ [\![\neg\varphi]\!]^M &= W \setminus [\![\varphi]\!]^M \\ [\![\varphi_1 \wedge \varphi_2]\!]^M &= [\![\varphi_1]\!]^M \cap [\![\varphi_2]\!]^M \\ [\![\ [\pi]\varphi]\!]^M &= \{w \in W \mid \forall v \text{ if } (w,v) \in [\![\pi]\!]^M \text{ then } v \in [\![\varphi]\!]^M\} \end{aligned}$$

$$\begin{aligned} [\![a]\!]^M &= R(a) \\ [\![?\varphi]\!]^M &= \{(w,w) \in W \times W \mid w \in [\![\varphi]\!]^M\} \\ [\![\pi_1;\pi_2]\!]^M &= [\![\pi_1]\!]^M \circ [\![\pi_2]\!]^M \\ [\![\pi_1 \cup \pi_2]\!]^M &= [\![\pi_1]\!]^M \cup [\![\pi_2]\!]^M \\ [\![\pi^*]\!]^M &= ([\![\pi]\!]^M)^* \end{aligned}$$

Here $([\![\pi]\!]^M)^*$ is the reflexive transitive closure of binary relation $[\![\pi]\!]^M$. If $w \in W$ then we use $M, w \models \varphi$ for $w \in [\![\varphi]\!]^M$, and we say that φ is true at w. A PDL formula φ is *true* in a model if it holds at every state in that model.

These definitions specify how formulas of PDL can be used to make assertions about PDL models. E.g., the formula $\langle a \rangle \top$ says that the current state has an $R(a)$-successor. Truth of $\langle a \rangle \top$ in a model says that $R(a)$ is serial.

Note that ? is an operation for mapping formulas to programs. Programs of the form $?\varphi$ are called *tests*; they are interpreted as the identity relation, restricted to the states s satisfying the formula φ.

If $\sigma = \{p_1 \mapsto \varphi_1, \ldots, p_n \mapsto \varphi_n\}$ is a PDL substitution, we use φ^σ for $\sigma(\varphi)$ and π^σ for $\sigma(\pi)$. We can spell out φ^σ and π^σ, as follows:

$$\begin{array}{llll}
\top^\sigma & = \top & a^\sigma & = a \\
p^\sigma & = \sigma(p) & (?\varphi)^\sigma & = ?\varphi^\sigma \\
(\neg\varphi)^\sigma & = \neg\varphi^\sigma & (\pi_1;\pi_2)^\sigma & = \pi_1^\sigma;\pi_2^\sigma \\
(\varphi_1 \wedge \varphi_2)^\sigma & = \varphi_1^\sigma \wedge \varphi_2^\sigma & (\pi_1 \cup \pi_2)^\sigma & = \pi_1^\sigma \cup \pi_2^\sigma \\
([\pi]\varphi)^\sigma & = [\pi^\sigma]\varphi^\sigma & (\pi^*)^\sigma & = (\pi^\sigma)^*.
\end{array}$$

The following holds by simultaneous induction on the structure of formulas and programs:

Lemma 36 (Substitution) *For all* PDL *models M, all* PDL *formulas σ, all* PDL *programs π, all* PDL *substitutions σ:*

$$M, w \models \varphi^\sigma \text{ iff } M^\sigma, w \models \varphi.$$

$$(w, w') \in [\![\pi^\sigma]\!]^M \text{ iff } (w, w') \in [\![\pi]\!]^{M^\sigma}.$$

This is just the beginning of a more general model-theory for PDL, which is bisimulation-based just like basic modal logic.

One striking feature of PDL is that its set of validities is decidable, with a perspicuous axiomatization. We display it here, just to fix thoughts — but no details will be used in what follows.

Theorem 37 *The following axioms and inference rules are complete for* PDL*:*

$$\begin{array}{rl}
\textit{(K)} & \vdash [\pi](\varphi \to \psi) \to ([\pi]\varphi \to [\pi]\psi) \\
\textit{(test)} & \vdash [?\varphi_1]\varphi_2 \leftrightarrow (\varphi_1 \to \varphi_2) \\
\textit{(sequence)} & \vdash [\pi_1;\pi_2]\varphi \leftrightarrow [\pi_1][\pi_2]\varphi \\
\textit{(choice)} & \vdash [\pi_1 \cup \pi_2]\varphi \leftrightarrow [\pi_1]\varphi \wedge [\pi_2]\varphi \\
\textit{(mix)} & \vdash [\pi^*]\varphi \leftrightarrow \varphi \wedge [\pi][\pi^*]\varphi \\
\textit{(induction)} & \vdash (\varphi \wedge [\pi^*](\varphi \to [\pi]\varphi)) \to [\pi^*]\varphi
\end{array}$$

and the following rules of inference:

(Modus Ponens) *From* $\vdash \varphi_1$ *and* $\vdash \varphi_1 \to \varphi_2$*, infer* $\vdash \varphi_2$*.*
(Modal Generalisation) *From* $\vdash \varphi$*, infer* $\vdash [\pi]\varphi$*.*

In the rest of this paper, we are going to use PDL for a very special purpose, viz. as a *rich epistemic language*. This may be confusing at first sight, since the objects in our update models are events, and hence one might naturally think of a propositional dynamic logic for sequences of these. The latter use would be close to describing operational structure on update models viewed as *programs*, which we noted in Section 2, but then decided to forego. We ask the reader to firmly resist this association henceforth, and focus instead on the following epistemic perspective ([24,28]). To make the distinction even clearer, we will often refer to propositional dynamic logic in this epistemic guise as E-PDL.

Atomic relations will be epistemic accessibilities of single agents. Compositions like $b_1; b_2$ then express the 'levels of knowledge' of Parikh: if φ expresses that b_1 wants b_2 to pick up the children, then $[b_1; b_2]\varphi$ states that b_1 knows that b_2 knows what is expected of him (a precondition for being at ease about the arrangement). Next, if $B \subseteq N$ and B is finite, we use B as shorthand for $b_1 \cup b_2 \cup \cdots$. Under this convention, the general knowledge operator $E_B\varphi$ takes the shape $[B]\varphi$, while the common knowledge operator $C_B\varphi$ appears as $[B^*]\varphi$, i.e., $[B]\varphi$ expresses that it is general knowledge among agents B that φ, and $[B^*]\varphi$ expresses that it is common knowledge among agents B that φ. In the special case where $B = \emptyset$, B turns out equivalent to $?\bot$, the program that always fails. In the same vein, common *belief* among agents B that φ can be expressed as $[B; B^*]\varphi$. But E-PDL is much richer than these notions, in that it also allows for much more complex combinations of agent accessibility relations, corresponding to some pretty baroque 'generalized agents'. We have found no practical use for these at present, but they are the price that we cheerfully pay for having a language living in expressive harmony with its dynamic superstructure — as will be described now.

4.3 LCC*, a Dynamic Logic of Communication and Change*

Now we have all the ingredients for the definition of the logic of communication and change.

Definition 38 (LCC, Language) *The language* $\mathscr{L}_{\mathsf{LCC}}$ *is the result of adding a clause* $[\mathsf{U}, \mathsf{e}]\varphi$ *for update execution to the language of* E-PDL*, where* U *is an update model for* $\mathscr{L}_{\mathsf{LCC}}$*.*

Definition 39 (LCC, Semantics) *The semantics* $[\![\varphi]\!]^M$ *is the standard se-*

mantics of PDL*, with the meaning of* $[\mathsf{U},\mathsf{e}]\varphi$ *in* $M = (W, R, V)$ *given by:*

$$[\![[\mathsf{U},\mathsf{e}]\varphi]\!]^M = \{w \in W \mid \textit{if } M, w \models \mathsf{pre}(\mathsf{e}) \textit{ then } (w, \mathsf{e}) \in [\![\varphi]\!]^{M\circ\mathsf{U}}\}.$$

We have to check that the definition of execution of update models is well behaved. The following theorems state that it is, in the sense that it preserves epistemic model bisimulation and update model bisimulation (the corresponding theorems for LEA are proved in [1]).

Theorem 40 *For all* PDL *models* M, w *and* N, v *and all formulas* $\varphi \in \mathscr{L}_{\mathsf{LCC}}$

$$\textit{If } M, w \leftrightarroweq N, v \textit{ then } w \in [\![\varphi]\!]^M \textit{ iff } v \in [\![\varphi]\!]^N$$

This theorem must be proved simultaneously with the following result.

Theorem 41 *For all* PDL *models* M, w *and* N, v*, all update models* U, e*:*

$$\textit{If } M, w \leftrightarroweq N, v \textit{ then } M \circ \mathsf{U}, (w, \mathsf{e}) \leftrightarroweq N \circ \mathsf{U}, (v, a).$$

Proof We prove both results simultaneously by induction on formulas φ and the preconditions of the relevant update models.

- Proof of Theorem 40: the base case for propositional variables and the cases for negation, conjunction, and program modalities is standard. The only interesting case is for formulas of the form $[\mathsf{U},\mathsf{e}]\varphi$. Suppose $w \in [\![[\mathsf{U},\mathsf{e}]\varphi]\!]^M$. Therefore $w \in [\![\mathsf{pre}(\mathsf{e})]\!]^M$ implies $(w, \mathsf{e}) \in [\![\varphi]\!]^{M\circ\mathsf{U}}$. By the induction hypothesis $w \in [\![\mathsf{pre}(\mathsf{e})]\!]^M$ iff $v \in [\![\mathsf{pre}(\mathsf{e})]\!]^N$ and $M \circ \mathsf{U}, (w, \mathsf{e}) \leftrightarroweq N \circ \mathsf{U}, (v, \mathsf{e})$. Then, by applying the induction hypothesis to $M \circ \mathsf{U}, (w, a)$ and $N \circ \mathsf{U}, (v, \mathsf{e})$, we infer $v \in [\![\mathsf{pre}(\mathsf{e})]\!]^N$ implies $(v, \mathsf{e}) \in [\![\varphi]\!]^{N\circ\mathsf{U}}$. By the semantics this is equivalent to $v \in [\![[\mathsf{U},\mathsf{e}]\varphi]\!]^N$. The other way around is completely analogous.
- Proof of Theorem 41: Let B be a bisimulation witnessing $M, w \leftrightarroweq N, v$. Then the relation C between $W_M \times \mathsf{E}_\mathsf{U}$ and $W_N \times \mathsf{E}_\mathsf{U}$ defined by

  $$(w, \mathsf{e})C(v, \mathsf{f}) \text{ iff } wBv \text{ and } \mathsf{e} = \mathsf{f}$$

  is a bisimulation.

  The induction hypothesis guarantees that (w, e) exists iff (v, f) exists.

  Suppose $(w, \mathsf{e})C(v, \mathsf{f})$. Then wBv and $\mathsf{e} = \mathsf{f}$. The only non-trivial check is the check for sameness of valuation. By wBv, w and v satisfy $V_M(w) = V_N(v)$. By $\mathsf{e} = \mathsf{f}$, e and f have the same substitution σ. By the fact that w and v are bisimilar, by the induction hypothesis we have that $w \in [\![\varphi]\!]^M$ iff $v \in [\![\varphi]\!]^N$. Thus, by $V_M(w) = V_N(s)$ and the definition of V_M^σ and V_N^σ, we get $V_M^\sigma(w) = V_N^\sigma(v)$.

□

Theorem 42 *For all* PDL *models* M, w*, all update models* U_1, e *and* U_2, f*:*

$$\textit{If } \mathsf{U}_1, \mathsf{e} \leftrightarroweq \mathsf{U}_2, \mathsf{f} \textit{ then } M \circ \mathsf{U}_1, (w, \mathsf{e}) \leftrightarroweq M \circ \mathsf{U}_2, (w, \mathsf{f}).$$

Proof Let R be a bisimulation witnessing $\mathsf{U}_1, \mathsf{e} \leftrightarroweq \mathsf{U}_2, \mathsf{f}$. Then the relation C between $W_M \times \mathsf{E}_{\mathsf{U}_1}$ and $W_M \times \mathsf{E}_{\mathsf{U}_2}$ given by

$$(w, \mathsf{e})C(v, \mathsf{f}) \text{ iff } w = v \text{ and } \mathsf{eRf}$$

is a bisimulation.

Suppose $(w, \mathsf{e})C(v, \mathsf{f})$. Then $w = v$ and $\mathsf{e}R\mathsf{f}$. Again, the only non-trivial check is the check for sameness of valuation. By $\mathsf{e}R\mathsf{f}$, the substitutions σ of e and τ of f are equivalent. By $w = v$, $V_M(w) = V_M(v)$. It follows that $V_M^\sigma(w) = V_M^\sigma(v)$, i.e., (w, e) and (v, f) have the same valuation. □

4.4 Expressive Power of LCC

Now, if we have designed things well, the dynamic system just defined should be in harmony with its static substructure. In particular, we expect reduction axioms for compositional analysis of the effects of arbitrary update models: $[\mathsf{U}, \mathsf{e}][\pi]\varphi$. These will then, if one wants to phrase this somewhat negatively, 'reduce LCC to E-PDL.' As before, the quest for such principles starts with an attempt to describe what is the case after the update in terms of what is the case before the update. In case of LCC, epistemic relations can take the shape of arbitrary E-PDL programs. So we must ask ourselves how we can find, for a given relation $[\![\pi]\!]^{M \circ \mathsf{U}}$ a corresponding relation in the original model M, w.

A formula of the form $\langle \mathsf{U}, \mathsf{e}_i \rangle \langle \pi \rangle \varphi$ is true in some model M, w iff there is a π-path in $M \circ \mathsf{U}$ leading from (w, e_i) to a φ world (v, e_j). That means there is some path $w \dots v$ in M and some path $\mathsf{e}_i \dots \mathsf{e}_j$ in U such that $(M, w) \models \mathsf{pre}(\mathsf{e}_i)$ and ...and $(M, v) \models \mathsf{pre}(\mathsf{e}_j)$ and of course $(M, v) \models \langle \mathsf{U}, \mathsf{e}_j \rangle \varphi$. The program $T_{ij}^{\mathsf{U}}(\pi)$, to be defined below, captures this. A $T_{ij}^{\mathsf{U}}(\pi)$-path in the original model corresponds to a π-path in the updated model. But in defining $T_{ij}^{\mathsf{U}}(\pi)$ we cannot refer to a model M. The definition of the transformed program $T_{ij}^{\mathsf{U}}(\pi)$ only depends on π, U, e_i and e_j. These program transformers are used in the reduction axiom, which can be formulated as follows:

$$[\mathsf{U}, \mathsf{e}_i][\pi]\varphi \leftrightarrow \bigwedge_{j=0}^{n-1} [T_{ij}^{\mathsf{U}}(\pi)][\mathsf{U}, \mathsf{e}_j]\varphi.$$

The remainder of this section is directed towards showing that this axiom is sound (Theorem 48). Our main new technical contribution in this paper lies in the machinery leading up to this.

The program transformer T^{U}_{ij} is defined as follows:

Definition 43 (T^{U}_{ij} Program Transformers)

$$\begin{aligned}
T^{\mathsf{U}}_{ij}(a) &= \begin{cases} ?\mathsf{pre}(\mathsf{e}_i); a & \textit{if } \mathsf{e}_i R(a) \mathsf{e}_j, \\ ?\bot & \textit{otherwise} \end{cases} \\
T^{\mathsf{U}}_{ij}(?\varphi) &= \begin{cases} ?(\mathsf{pre}(\mathsf{e}_i) \wedge [\mathsf{U}, \mathsf{e}_i]\varphi) & \textit{if } i = j, \\ ?\bot & \textit{otherwise} \end{cases} \\
T^{\mathsf{U}}_{ij}(\pi_1; \pi_2) &= \bigcup_{k=0}^{n-1} (T^{\mathsf{U}}_{ik}(\pi_1); T^{\mathsf{U}}_{kj}(\pi_2)) \\
T^{\mathsf{U}}_{ij}(\pi_1 \cup \pi_2) &= T^{\mathsf{U}}_{ij}(\pi_1) \cup T^{\mathsf{U}}_{ij}(\pi_2) \\
T^{\mathsf{U}}_{ij}(\pi^*) &= K^{\mathsf{U}}_{ijn}(\pi)
\end{aligned}$$

where $K^{\mathsf{U}}_{ijn}(\pi)$ is given by Definition 44.

We need the additional program transformer K^{U}_{ijn} in order to build the paths corresponding to the transitive closure of π in the updated model step by step, where we take more and more worlds of the update model into account. Intuitively, $K^{\mathsf{U}}_{ijk}(\pi)$ is a (transformed) program for all the π paths from (w, e_i) to (v, e_j) that can be traced through $M \circ \mathsf{U}$ while avoiding a pass through intermediate states with events e_k and higher (this is the thrust of Definition 44). Here, a π path from (w, e_i) to (v, e_j) is a path of the form $(w, \mathsf{e}_i), (v, \mathsf{e}_j)$ (in case $i = j$), or $(w, \mathsf{e}_i) \xrightarrow{\pi} \cdots \xrightarrow{\pi} (v, \mathsf{e}_j)$. Intermediate states are the states at positions $\cdots$ where a π step ends and a π step starts. Note that the restriction only applies to intermediate states. States passed in the execution of π may involve events e_m with $m > k$. A given intermediate state e_r may occur more than once in a π path.

Just as the definition of $T^{\mathsf{U}}_{ij}(\pi)$ does not refer to a concrete model M, also $K^{\mathsf{U}}_{ijn}(\pi)$ does not depend on a concrete model M. We only need to be concerned about the paths from e_i to e_j that could be the event components in a π-path in the updated model. Thus, $K^{\mathsf{U}}_{ij0}(\pi)$ is a program for all the paths from e_i to e_j that can be traced through U without stopovers at intermediate states that could yield a π path in an updated model. If $i = j$ it either is the skip action or a direct π loop, and otherwise it is a direct $T^{\mathsf{U}}_{ij}(\pi)$ step. This explains the base case in the following notion:

Definition 44 (K^{U}_{ijk} Path Transformers) *$K^{\mathsf{U}}_{ijk}(\pi)$ is defined by recursing on k, as follows:*

$$K^{\mathsf{U}}_{ij0}(\pi) = \begin{cases} ?\top \cup T^{\mathsf{U}}_{ij}(\pi) & \textit{if } i = j, \\ \\ T^{\mathsf{U}}_{ij}(\pi) & \textit{otherwise} \end{cases}$$

$$K^{\mathsf{U}}_{ij(k+1)}(\pi) = \begin{cases} (K^{\mathsf{U}}_{kkk}(\pi))^* & \textit{if } i = k = j, \\ \\ (K^{\mathsf{U}}_{kkk}(\pi))^*; K^{\mathsf{U}}_{kjk}(\pi) & \textit{if } i = k \neq j, \\ \\ K^{\mathsf{U}}_{ikk}(\pi); (K^{\mathsf{U}}_{kkk}(\pi))^* & \textit{if } i \neq k = j, \\ \\ K^{\mathsf{U}}_{ijk}(\pi) \cup (K^{\mathsf{U}}_{ikk}(\pi); (K^{\mathsf{U}}_{kkk}(\pi))^*; K^{\mathsf{U}}_{kjk}(\pi)) & \textit{otherwise} \\ & (i \neq k \neq j). \end{cases}$$

Concrete applications of Definitions 43 and 44 are found in Section 5. The next theorem states that the program transformation yields all the paths in the original model that correspond to paths in the updated model.

Theorem 45 (Program Transformation into E-PDL) *For all update models* U *and all* E-PDL *programs* π*, the following equivalence holds:*

$$(w, v) \in [\![T^{\mathsf{U}}_{ij}(\pi); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M \textit{ iff } ((w, \mathsf{e}_i), (v, \mathsf{e}_j)) \in [\![\pi]\!]^{M \circ \mathsf{U}}.$$

To prove Theorem 45 we need two auxiliary results.

Lemma 46 (Constrained Kleene Path) *Suppose*

$$(w, v) \in [\![T^{\mathsf{U}}_{ij}(\pi); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M \textit{ iff } ((w, \mathsf{e}_i), (v, \mathsf{e}_j)) \in [\![\pi]\!]^{M \circ \mathsf{U}}.$$

Then $(w, v) \in [\![K^{\mathsf{U}}_{ijk}(\pi); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M$ *iff there is a* π *path from* (w, e_i) *t* (v, e_j) *in* $M \circ \mathsf{U}$ *that does not have intermediate states* $\cdots \xrightarrow{\pi} (u, \mathsf{e}_r) \xrightarrow{\pi} \cdots$ *with* $r \geq k$.

Proof We use induction on k, following the definition of K^{U}_{ijk}, distinguishing a number of cases.

(a) *Base case* $k = 0$*, subcase* $i = j$: A π path from (w, e_i) to (v, e_j) in $M \circ \mathsf{U}$ that does not visit any intermediate states is either empty or a single π step from (w, e_i) to (v, e_j). Such a path exists iff

$$\begin{array}{ll}
 & ((w, \mathsf{e}_i).(v, \mathsf{e}_j)) \in \llbracket ?\top \cup \pi \rrbracket^{M \circ \mathsf{U}} \\
\text{iff (assumption)} & (w, v) \in \llbracket T^{\mathsf{U}}_{ij}(?\top \cup \pi); ?\mathsf{pre}(\mathsf{e}_j) \rrbracket^{M} \\
\text{iff (definition } T^{\mathsf{U}}_{ij}\text{)} & (w, v) \in \llbracket ?(\mathsf{pre}(\mathsf{e}_i) \wedge [\mathsf{U}, \mathsf{e}_i]\top) \cup T^{\mathsf{U}}_{ij}(\pi); ?\mathsf{pre}(\mathsf{e}_j) \rrbracket^{M} \\
\text{iff } (i = j, & (w, v) \in \llbracket ?\top \cup T^{\mathsf{U}}_{ij}(\pi); ?\mathsf{pre}(\mathsf{e}_j) \rrbracket^{M} \\
\quad \text{so } \mathsf{pre}(\mathsf{e}_i) = \mathsf{pre}(\mathsf{e}_j)) & \\
\text{iff (definition } K^{\mathsf{U}}_{ij0}\text{)} & (w, v) \in \llbracket K^{\mathsf{U}}_{ij0}(\pi); ?\mathsf{pre}(\mathsf{e}_j) \rrbracket^{M}.
\end{array}$$

(b) *Base case $k = 0$, subcase $i \neq j$*: A π path from (w, e_i) to (v, e_j) in $M \circ \mathsf{U}$ that does not visit any intermediate states is a single π step from (w, e_i) to (v, e_j). Such a path exists iff

$$\begin{array}{ll}
 & ((w, \mathsf{e}_i).(v, \mathsf{e}_j)) \in \llbracket \pi \rrbracket^{M \circ \mathsf{U}} \\
\text{iff (assumption)} & (w, v) \in \llbracket T^{\mathsf{U}}_{ij}(\pi); ?\mathsf{pre}(\mathsf{e}_j) \rrbracket^{M} \\
\text{iff (definition } K^{\mathsf{U}}_{ij0}\text{)} & (w, v) \in \llbracket K^{\mathsf{U}}_{ij0}(\pi); ?\mathsf{pre}(\mathsf{e}_j) \rrbracket^{M}.
\end{array}$$

(c) *Induction step.* Assume that $(w, v) \in \llbracket K^{\mathsf{U}}_{ijk}(\pi); ?\mathsf{pre}(\mathsf{e}_j \rrbracket^{M}$ iff there is a π path from (w, e_i) to (v, e_j) in $M \circ \mathsf{U}$ that does not pass through any pairs (u, e) with $\mathsf{e} \in \{\mathsf{e}_k, \ldots, \mathsf{e}_{n-1}\}$.

We have to show that $(w, v) \in \llbracket K^{\mathsf{U}}_{ij(k+1)}(\pi); ?\mathsf{pre}(\mathsf{e}_j \rrbracket^{M}$ iff there is a π path from (w, e_i) to (v, e_j) in $M \circ \mathsf{U}$ that does not pass through any pairs (u, e) with $\mathsf{e} \in \{\mathsf{e}_{k+1}, \ldots, \mathsf{e}_{n-1}\}$.

Case $i = k = j$. A π path from (w, e_i) to (v, e_j) in $M \circ \mathsf{U}$ that does not pass through any pairs (u, e) with $\mathsf{e} \in \{\mathsf{e}_{k+1}, \ldots, \mathsf{e}_{n-1}\}$ now consists of an arbitrary composition of π paths from e_k to e_k that do not visit any intermediate states with event component e_k or higher. By the induction hypothesis, such a path exists iff $(w, v) \in \llbracket (K^{\mathsf{U}}_{kkk}(\pi))^*; ?\mathsf{pre}(\mathsf{e}_j \rrbracket^{M}$ iff (definition of $K^{\mathsf{U}}_{ij(k+1)}$) $(w, v) \in \llbracket K^{\mathsf{U}}_{ij(k+1)}(\pi); ?\mathsf{pre}(\mathsf{e}_j \rrbracket^{M}$.

Case $i = k \neq j$. A π path from (w, e_i) to (v, e_j) in $M \circ \mathsf{U}$ that does pass through any pairs (u, e) with $\mathsf{e} \in \{\mathsf{e}_{k+1}, \ldots, \mathsf{e}_{n-1}\}$ now consists of a π path starting in (w, e_k) visiting states of the form (u, e_k) an arbitrary number of times, but never visiting states with event component e_k or higher in between, and ending in (v, e_k), followed by a π path from (u, e_k) to (v, e_j) that does not visit any pairs with event component $\mathsf{e} \in \{\mathsf{e}_k, \ldots, \mathsf{e}_{n-1}\}$. By the induction hypothesis, a π path from (w, e_k) to (u, e_k) of the first kind exists iff $(w, u) \in \llbracket (K^{\mathsf{U}}_{kkk}(\pi))^*; ?\mathsf{pre}(\mathsf{e}_k) \rrbracket^{M}$. Again by the induction hypothesis, a path from (u, e_k) to (v, e_j) of the second kind exists iff $(u, v) \in [\![K^{\mathsf{U}}_{kjk}; ?\mathsf{pre}(\mathsf{e}_j)]\!]^M$. Thus, the required path from (w, e_i) to (v, e_j) in $M \circ \mathsf{U}$ exists iff $(w, v) \in [\![(K^{\mathsf{U}}_{kkk}(\pi))^*; K^{\mathsf{U}}_{kjk}(\pi); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M$, which, by the definition of $K^{\mathsf{U}}_{ij(k+1)}$, is the case iff $(w, v) \in [\![K^{\mathsf{U}}_{ij(k+1)}(\pi); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M$.

The other two cases are similar. □

Lemma 47 (General Kleene Path) *Suppose* $(w, v) \in [\![T^{\mathsf{U}}_{ij}(\pi)); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M$ *iff there is a* π *step from* (w, e_i) *to* (v, e_j) *in* $M \circ \mathsf{U}$*.*

Then $(w, v) \in [\![K^{\mathsf{U}}_{ijn}(\pi)); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M$ *iff there is a* π *path from* (w, e_i) *to* (v, e_j) *in* $M \circ \mathsf{U}$*.*

Proof Suppose $(w, v) \in [\![T^{\mathsf{U}}_{ij}(\pi); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M$ iff there is a π path from (w, e_i) to (v, e_j) in $M \circ \mathsf{U}$. Then, assuming that U has states $\mathsf{e}_0, \ldots, \mathsf{e}_{n-1}$, an application of Lemma 46 yields that $K^{\mathsf{U}}_{ijn}(\pi)$ is a program for all the π paths from (w, e_i) to (v, e_j) that can be traced through $M \circ \mathsf{U}$, for stopovers at any (u, e_k) with $0 \leq k \leq n-1$ are allowed. □

Lemma 47 explains the use of K^{U}_{ijn} in the clause for π^* in Definition 43. Now, we can clinch matters:

Proof of Theorem 45. This time, we use induction on the structure of π.

Base case a:

$$\begin{array}{ll} & (w, v) \in [\![T^{\mathsf{U}}_{ij}(a); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M \\ \text{iff} & (w, v) \in [\![?\mathsf{pre}(\mathsf{e}_i); a; ?\mathsf{pre}(\mathsf{e}_j)]\!]^M \text{ and } \mathsf{e}_i R(a) \mathsf{e}_j \\ \text{iff} & M, w \models \mathsf{pre}(\mathsf{e}_i), (w, v) \in [\![a]\!]^M, \mathsf{e}_i R(a) \mathsf{e}_j \text{ and } M, v \models \mathsf{pre}(\mathsf{e}_j) \\ \text{iff} & ((w, \mathsf{e}_i), (v, \mathsf{e}_j)) \in [\![\pi]\!]^{M \circ \mathsf{U}}. \end{array}$$

Base case $?\varphi$*, subcase* $i = j$*:*

$$\begin{array}{ll} & (w, v) \in [\![T^{\mathsf{U}}_{ij}(?\varphi); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M \\ \text{iff} & (w, v) \in [\![?(\mathsf{pre}(\mathsf{e}_i) \wedge [\mathsf{U}, \mathsf{e}_i]\varphi); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M \\ \text{iff} & w = v \text{ and } M, w \models \mathsf{pre}(\mathsf{e}_i) \text{ and } M, w \models [\mathsf{U}, \mathsf{e}_i]\varphi \\ \text{iff} & w = v \text{ and } M, w \models \mathsf{pre}(\mathsf{e}_i) \text{ and } M, w \models \mathsf{pre}(\mathsf{e}_i) \text{ implies } M \circ \mathsf{U}, (w, \mathsf{e}_i) \models \varphi \\ \text{iff} & w = v \text{ and } M \circ \mathsf{U}, (w, \mathsf{e}_i) \models \varphi \\ \text{iff} & ((w, \mathsf{e}_i), (v, \mathsf{e}_j)) \in [\![?\varphi]\!]^{M \circ \mathsf{U}}. \end{array}$$

Base case $?\varphi$, subcase $i \neq j$:

$$\begin{array}{ll} & (w,v) \in [\![T^{\mathsf{U}}_{ij}(?\varphi); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M \\ \text{iff} & (w,v) \in [\![?\bot]\!]^M \\ \text{iff} & ((w,\mathsf{e}_i),(v,\mathsf{e}_j)) \in [\![?\varphi]\!]^{M \circ \mathsf{U}}. \end{array}$$

Induction step: Now consider any complex program π and assume for all components π' of π that:

$$(w,v) \in [\![T^{\mathsf{U}}_{ij}(\pi'); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M \text{ iff } ((w,\mathsf{e}_i),(v,\mathsf{e}_j)) \in [\![\pi']\!]^{M \circ \mathsf{U}}.$$

We have to show:

$$(w,v) \in [\![T^{\mathsf{U}}_{ij}(\pi); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M \text{ iff } ((w,\mathsf{e}_i),(v,\mathsf{e}_j)) \in [\![\pi]\!]^{M \circ \mathsf{U}}.$$

Here are the three relevant program operations: $\pi = \pi_1; \pi_2$:

$$\begin{array}{ll} & (w,v) \in [\![T^{\mathsf{U}}_{ij}(\pi_1;\pi_2); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M \\ \text{iff} & (w,v) \in [\![\bigcup_{k=0}^{n-1}(T^{\mathsf{U}}_{ik}(\pi_1); T^{\mathsf{U}}_{kj}(\pi_2)); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M \\ \text{iff} & \text{for some } k \in \{0,\dots,n-1\} (w,v) \in [\![T^{\mathsf{U}}_{ik}(\pi_1); T^{\mathsf{U}}_{kj}(\pi_2); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M \\ \text{iff} & \text{for some } k \in \{0,\dots,n-1\} \text{ and some } u \in W \\ & (w,u) \in [\![T^{\mathsf{U}}_{ik}(\pi_1)]\!]^M \text{ and } (u,v) \in [\![T^{\mathsf{U}}_{kj}(\pi_2); ?\mathsf{pre}(\mathsf{e}_j)]\!]^M \\ \text{iff (ih)} & \text{for some } k \in \{0,\dots,n-1\} \text{ and some } u \in W \\ & (w,u) \in [\![T^{\mathsf{U}}_{ik}(\pi_1)]\!]^M \text{ and } ((u,\mathsf{e}_k),(v,\mathsf{e}_j)) \in [\![\pi_2]\!]^{M \circ \mathsf{U}} \\ \text{iff} & \text{for some } k \in \{0,\dots,n-1\} \text{ and some } u \in W \\ & (w,u) \in [\![T^{\mathsf{U}}_{ik}(\pi_1)]\!]^M, \text{ and} \\ & M,u \models \mathsf{pre}(\mathsf{e}_k) \text{ and } ((u,\mathsf{e}_k),(v,\mathsf{e}_j)) \in [\![\pi_2]\!]^{M \circ \mathsf{U}} \\ \text{iff} & \text{for some } k \in \{0,\dots,n-1\} \text{ and some } u \in W \\ & (w,u) \in [\![T^{\mathsf{U}}_{ik}(\pi_1); \mathsf{pre}(\mathsf{e}_k)]\!]^M \text{ and } ((u,\mathsf{e}_k),(v,\mathsf{e}_j)) \in [\![\pi_2]\!]^{M \circ \mathsf{U}} \\ \text{iff (ih too)} & \text{for some } k \in \{0,\dots,n-1\} \text{ and some } u \in W \\ & ((w,\mathsf{e}_i),(u,\mathsf{e}_k)) \in [\![\pi_1]\!]^{M \circ \mathsf{U}} \text{ and } ((u,\mathsf{e}_k),(v,\mathsf{e}_j)) \in [\![\pi_2]\!]^{M \circ \mathsf{U}} \\ \text{iff} & ((w,\mathsf{e}_i),(v,\mathsf{e}_j)) \in [\![\pi]\!]^{M \circ \mathsf{U}}. \end{array}$$

$\pi = \pi_1 \cup \pi_2$:

$$\begin{array}{ll}
 & (w,v) \in \llbracket T^{\mathsf{U}}_{ij}(\pi_1 \cup \pi_2); ?\mathsf{pre}(\mathsf{e}_j)\rrbracket^M \\
\text{iff} & (w,v) \in \llbracket (T^{\mathsf{U}}_{ij}(\pi_1) \cup T^{\mathsf{U}}_{ij}(\pi_2)); ?\mathsf{pre}(\mathsf{e}_j)\rrbracket^M \\
\text{iff} & (w,v) \in \llbracket (T^{\mathsf{U}}_{ij}(\pi_1); ?\mathsf{pre}(\mathsf{e}_j)) \cup (T^{\mathsf{U}}_{ij}(\pi_2)?\mathsf{pre}(\mathsf{e}_j))\rrbracket^M \\
\text{iff} & (w,v) \in \llbracket T^{\mathsf{U}}_{ij}(\pi_1); ?\mathsf{pre}(\mathsf{e}_j)\rrbracket^M \text{ or } (w,v) \in \llbracket T^{\mathsf{U}}_{ij}(\pi_2); ?\mathsf{pre}(\mathsf{e}_j)\rrbracket^M \\
\text{iff (ih)} & ((w,\mathsf{e}_i),(v,\mathsf{e}_j)) \in \llbracket \pi_1 \rrbracket^{M \circ \mathsf{U}} \text{ or } ((w,\mathsf{e}_i),(v,\mathsf{e}_j)) \in \llbracket \pi_2 \rrbracket^{M \circ \mathsf{U}} \\
\text{iff} & ((w,\mathsf{e}_i),(v,\mathsf{e}_j)) \in \llbracket \pi \rrbracket^{M \circ \mathsf{U}}.
\end{array}$$

$\pi = \pi^*$:

$$\begin{array}{ll}
 & (w,v) \in \llbracket T^{\mathsf{U}}_{ij}(\pi^*); ?\mathsf{pre}(\mathsf{e}_j)\rrbracket^M \\
\text{iff (definition } T^{\mathsf{U}}_{ij}) & (w,v) \in \llbracket K^{\mathsf{U}}_{ijn}(\pi); ?\mathsf{pre}(\mathsf{e}_j)\rrbracket^M \\
\text{iff (ih, Lemma 47)} & \text{there is a } \pi \text{ path from } (w,\mathsf{e}_i) \text{ to } (v,\mathsf{e}_j) \text{ in } M \circ \mathsf{U} \\
\text{iff} & ((w,\mathsf{e}_i),(v,\mathsf{e}_j)) \in \llbracket \pi^* \rrbracket^{M \circ \mathsf{U}}.
\end{array}$$

□

Theorem 48 (Reduction Equivalence) *Suppose that the model* U *has* n *states* $\mathsf{e}_0, \ldots, \mathsf{e}_{n-1}$. *Then:*

$$M, w \models [\mathsf{U}, \mathsf{e}_i][\pi]\varphi \textit{ iff } M, w \models \bigwedge_{j=0}^{n-1} [T^{\mathsf{U}}_{ij}(\pi)][\mathsf{U}, \mathsf{e}_j]\varphi.$$

Proof The result is derived by the following chain of equivalences:

$$\begin{array}{ll}
 & M, w \models [\mathsf{U}, \mathsf{e}_i][\pi]\varphi \\
\text{iff} & M, w \models \mathsf{pre}(\mathsf{e}_i) \text{ implies } M \circ \mathsf{U}, (w, \mathsf{e}_i) \models [\pi]\varphi \\
\text{iff} & \forall v \in W, j \in \{0, \ldots, n-1\} \text{ with } ((w,\mathsf{e}_i),(v,\mathsf{e}_j)) \in \llbracket \pi \rrbracket^{M \circ \mathsf{U}}, \\
 & \quad M \circ \mathsf{U}, (v, \mathsf{e}_j) \models \varphi \\
\text{iff (Thm 45)} & \forall v \in W, j \in \{0, \ldots, n-1\} \text{ with } (w,v) \in \llbracket T^{\mathsf{U}}_{ij}(\pi); ?\mathsf{pre}(\mathsf{e}_j)\rrbracket^M, \\
 & \quad M \circ \mathsf{U}, (v, \mathsf{e}_j) \models \varphi \\
\text{iff} & \forall v \in W, j \in \{0, \ldots, n-1\} \text{ with } (w,v) \in \llbracket T^{\mathsf{U}}_{ij}(\pi)\rrbracket^M, \\
 & \quad M, v \models \mathsf{pre}(\mathsf{e}_j) \text{ implies } M \circ \mathsf{U}, (v, \mathsf{e}_j) \models \varphi \\
\text{iff} & \forall v \in W, j \in \{0, \ldots, n-1\} \text{ with } (w,v) \in \llbracket T^{\mathsf{U}}_{ij}(\pi)\rrbracket^M, \\
 & \quad M, v \models [\mathsf{U}, \mathsf{e}_j]\varphi \\
\text{iff} & M, w \models \bigwedge_{j=0}^{n-1} [T^{\mathsf{U}}_{ij}(\pi)][\mathsf{U}, \mathsf{e}_j]\varphi
\end{array}$$

□

What the Reduction Equivalence tells us is that LCC is equivalent to $\mathsf{E\text{-}PDL}$, and hence, that a proof system for LCC can be given in terms of axioms that reduce formulas of the form $[\mathsf{U},\mathsf{e}]\varphi$ to equivalent formulas ψ with the property that their main operator is not an update modality for U. First, we state the former model-theoretic expressiveness result, which is the first main theorem of this paper:

Theorem 49 (‘$\mathsf{LCC} = \mathsf{E\text{-}PDL}$’) *The languages of* LCC *and* $\mathsf{E\text{-}PDL}$ *have equal expressive power.*

The earlier-mentioned similarity between finite automata and our current approach is most striking in the definition of the transformation for starred programs. The definition of transformed π^* paths in terms of operators $K_{ijk}(\pi)$ resembles the definition of sets of regular languages L_k generated by moving through a non-deterministic finite automaton without passing through states numbered k or higher, in the well-known proof of Kleene's Theorem. Textbook versions of its proof can be found in many places, e.g., [19, Theorem 2.5.1].

Another, more technical, interpretation of Theorems 11, 12 is that they establish a strong *closure property* for propositional dynamic logic: not just under the usual syntactic relativizations to submodels, but also under syntactic counterparts for much more general model transformations.

4.5 Reduction Axioms and Completeness for LCC

The results from the previous section point the way to appropriate reduction axioms for LCC. In the axioms below $p^{\mathsf{sub}}(\mathsf{e})$ is $\mathsf{sub}(\mathsf{e})(p)$ if p is in the domain of $\mathsf{sub}(\mathsf{e})$, otherwise it is p.

Definition 50 (Proof System for LCC) *The proof system for* LCC *consists of all axioms and rules of* PDL*, plus the following reduction axioms:*

$$\begin{aligned}
[\mathsf{U},\mathsf{e}]\top &\leftrightarrow \top\\
[\mathsf{U},\mathsf{e}]p\ &\leftrightarrow (\mathsf{pre}(\mathsf{e}) \rightarrow p^{\mathsf{sub}(\mathsf{e})})\\
[\mathsf{U},\mathsf{e}]\neg\varphi &\leftrightarrow (\mathsf{pre}(\mathsf{e}) \rightarrow \neg[\mathsf{U},\mathsf{e}]\varphi)\\
[\mathsf{U},\mathsf{e}](\varphi_1 \wedge \varphi_2) &\leftrightarrow ([\mathsf{U},\mathsf{e}]\varphi_1 \wedge [\mathsf{U},\mathsf{e}]\varphi_2)\\
[\mathsf{U},\mathsf{e}_i][\pi]\varphi &\leftrightarrow \bigwedge_{j=0}^{n-1} [T_{ij}^{\mathsf{U}}(\pi)][\mathsf{U},\mathsf{e}_j]\varphi.
\end{aligned}$$

plus inference rules of necessitation for all update model modalities.

The last, and most crucial, of the reduction axioms in the given list is based on program transformation. Incidentally, if slightly more general updates with so-called 'multiple pointed update models' (cf. [12]) are added to the language, we would need this additional reduction axiom:

$$[\mathsf{U},\mathsf{W}]\varphi \leftrightarrow \bigwedge_{\mathsf{e}\in\mathsf{W}} [\mathsf{U},\mathsf{e}]\varphi$$

As before with logics for public announcement, these reduction axioms also drive a *translation procedure*. The results of Section 4.4 tell us that LCC is no more expressive than E-PDL; indeed, program transformations provide the following translation:

Definition 51 (Translation) *The function t takes a formula from the language of* LCC *and yields a formula in the language of* PDL*.*

$$\begin{array}{llll}
t(\top) & = \top & r(a) & = a \\
t(p) & = p & r(B) & = B \\
t(\neg\varphi) & = \neg t(\varphi) & r(?\varphi) & = ?t(\varphi) \\
t(\varphi_1 \wedge \varphi_2) & = t(\varphi_1) \wedge t(\varphi_2) & r(\pi_1;\pi_2) & = r(\pi_1); r(\pi_2) \\
t([\pi]\varphi) & = [r(\pi)]t(\varphi) & r(\pi_1 \cup \pi_2) & = r(\pi_1) \cup r(\pi_2) \\
t([\mathsf{U},\mathsf{e}]\top) & = \top & r(\pi^*) & = (r(\pi))^*. \\
t([\mathsf{U},\mathsf{e}]p) & = t(\mathsf{pre}(\mathsf{e})) \rightarrow p^{\mathsf{sub}(\mathsf{e})} & & \\
t([\mathsf{U},\mathsf{e}]\neg\varphi) & = t(\mathsf{pre}(\mathsf{e})) \rightarrow \neg t([\mathsf{U},\mathsf{e}]\varphi) & & \\
t([\mathsf{U},\mathsf{e}](\varphi_1 \wedge \varphi_2) & = t([\mathsf{U},\mathsf{e}]\varphi_1) \wedge t([\mathsf{U},\mathsf{e}]\varphi_2) & & \\
t([\mathsf{U},\mathsf{e}_i][\pi]\varphi & = \bigwedge_{j=0}^{n-1}[T_{ij}^{\mathsf{U}}(r(\pi))]t([\mathsf{U},\mathsf{e}_j]\varphi) & & \\
t([\mathsf{U},\mathsf{e}][\mathsf{U}',\mathsf{e}']\varphi & = t([\mathsf{U},\mathsf{e}]t([\mathsf{U}',\mathsf{e}']\varphi)) & &
\end{array}$$

The correctness of this translation follows from direct semantic inspection, using the program transformation corollary for the translation of $[\mathsf{U},\mathsf{e}_i][\pi]\varphi$ formulas. The clause for iterated update modalities gives rise to exactly the same comments as those made for PAL-RC in Section 3.5.

As for deduction in our system, we note the following result:

Theorem 52 (Completeness for LCC) $\models \varphi$ *iff* $\vdash \varphi$.

Proof The proof system for PDL is complete, and every formula in the language of LCC is provably equivalent to a PDL formula. □

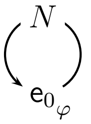
5 Analyzing Major Communication Types

Our analysis of LCC has been abstract and general. But the program transformation approach has a concrete pay-off! It provides a systematic perspective on communicative updates that occur in practice. For public announcement and common knowledge, it was still possible to find appropriate reduction axioms by hand. Such axioms can also be generated automatically, however, by program transformation, as we will now show. This method then allows us to deal with much more complicated cases, such as secret group communication and common belief, or subgroup announcement and common knowledge, where axiom generation by hand is infeasible. For a border-line case, see [27] for a direct axiomatization of the logic of subgroup communication with common knowledge — a topic conspicuously absent from, e.g., [14]. Our analysis obviates the need for this laborious, and error-prone, work.

The following generated axioms may look unwieldy, illustrating the fact that E-PDL functions as an assembler language for detailed analysis of the higher level specifications of communicative updates in terms of update models. But upon closer inspection, they make sense, and indeed, for simple communicative scenarios, they can be seen to reduce to EL-RC.

5.1 Public Announcement and Common Knowledge

The update model for public announcement that φ consists of a single state e_0 with precondition φ and epistemic relation $\{(\mathsf{e}_0, \mathsf{e}_0)\}$ for all agents. Call this model P_φ.



We are interested how public announcement that φ affects common knowledge in a group of agents B, i.e., we want to compute $[P_\varphi, \mathsf{e}_0][B^*]\psi$. For this, we need $T^{P_\varphi}_{00}(B^*)$, which equalled $K^{P_\varphi}_{001}(B)$.

To work out $K^{P_\varphi}_{001}(B)$, we need $K^{P_\varphi}_{000}(B)$, and for $K^{P_\varphi}_{000}(B)$, we need $T^{P_\varphi}_{00}(B)$, which turns out to be $\bigcup_{b \in B}(?\varphi; b)$, or equivalently, $?\varphi; B$. Working upwards from this, we get:

$$K^{P_\varphi}_{000}(B) = ?\top \cup T^{P_\varphi}_{00}(B) = ?\top \cup (?\varphi; B),$$

and therefore:

$$\begin{aligned} K^{P_\varphi}_{001}(B) &= (K^{P_\varphi}_{000}(B))^* \\ &= (?\top \cup (?\varphi; B))^* \\ &= (?\varphi; B)^*. \end{aligned}$$

Thus, the reduction axiom for the public announcement update P_φ with respect to the program for common knowledge among agents B, works out as follows:

$$\begin{aligned} [P_\varphi, \mathsf{e}_0][B^*]\psi &\leftrightarrow [T^{P_\varphi}_{00}(B^*)][P_\varphi, \mathsf{e}_0]\psi \\ &\leftrightarrow [K^{P_\varphi}_{001}(B)][P_\varphi, \mathsf{e}_0]\psi \\ &\leftrightarrow [(?\varphi; B)^*][P_\varphi, \mathsf{e}_0]\psi. \end{aligned}$$

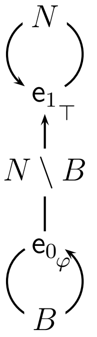
This expresses that every B path consisting of φ worlds ends in a $[P_\varphi, \mathsf{e}_0]\psi$ world, i.e., it expresses what is captured by the special purpose operator $C_B(\varphi, \psi)$ from Section 3.2.

5.2 Secret Group Communication and Common Belief

The logic of secret group communication is the logic of email 'cc' (assuming that emails arrive immediately and are read immediately). The update model for a secret group message to B that φ consists of two possible events $\mathsf{e}_0, \mathsf{e}_1$, where e_0 has precondition φ and e_1 has precondition $\top$, and where the accessibilities T are given by:

$$T = \{\mathsf{e}_0 R(b) \mathsf{e}_0 \mid b \in B\} \cup \{\mathsf{e}_0 R(a) \mathsf{e}_1 \mid a \in N \setminus B\} \cup \{\mathsf{e}_1 R(a) \mathsf{e}_1 \mid a \in N\}.$$

The actual event is e_0. The members of B are aware that φ gets communicated; the others think that nothing happens. In this thought they are mistaken, which is why 'cc' updates generate KD45 models: i.e., 'cc' updates make knowledge degenerate into belief.



We work out the program transformations that this update engenders for common knowledge among a group of agents D. Call the update model CC^B_φ.

We will have to work out $K_{002}^{\mathrm{CC}_\varphi^B} D$, $K_{012}^{\mathrm{CC}_\varphi^B} D$, $K_{112}^{\mathrm{CC}_\varphi^B} D$, $K_{102}^{\mathrm{CC}_\varphi^B} D$.

For these, we need $K_{001}^{\mathrm{CC}_\varphi^B} D$, $K_{011}^{\mathrm{CC}_\varphi^B} D$, $K_{111}^{\mathrm{CC}_\varphi^B} D$, $K_{101}^{\mathrm{CC}_\varphi^B} D$.

For these in turn, we need $K_{000}^{\mathrm{CC}_\varphi^B} D$, $K_{010}^{\mathrm{CC}_\varphi^B} D$, $K_{110}^{\mathrm{CC}_\varphi^B} D$, $K_{100}^{\mathrm{CC}_\varphi^B} D$.

For these, we need:

$$\begin{aligned}
T_{00}^{\mathrm{CC}_\varphi^B} D &= \bigcup_{d\in B\cap D} (?\varphi; d) \ = ?\varphi; (B\cap D)\\
T_{01}^{\mathrm{CC}_\varphi^B} D &= \bigcup_{d\in D\setminus B} (?\varphi; d) \ = ?\varphi; (D\setminus B)\\
T_{11}^{\mathrm{CC}_\varphi^B} D &= D\\
T_{10}^{\mathrm{CC}_\varphi^B} D &= ?\bot
\end{aligned}$$

It follows that:

$$\begin{aligned}
K_{000}^{\mathrm{CC}_\varphi^B} D &= ?\top \cup (?\varphi; (B\cap D))\\
K_{010}^{\mathrm{CC}_\varphi^B} D &= ?\varphi; (D\setminus B)\\
K_{110}^{\mathrm{CC}_\varphi^B} D &= ?\top \cup D,\\
K_{100}^{\mathrm{CC}_\varphi^B} D &= ?\bot
\end{aligned}$$

From this we can work out the K_{ij1}, as follows:

$$\begin{aligned}
K_{001}^{\mathrm{CC}_\varphi^B} D &= (?\varphi; (B\cap D))^*\\
K_{011}^{\mathrm{CC}_\varphi^B} D &= (?\varphi; (B\cap D))^*; (D\setminus B)\\
K_{111}^{\mathrm{CC}_\varphi^B} D &= ?\top \cup D\\
K_{101}^{\mathrm{CC}_\varphi^B} D &= ?\bot.
\end{aligned}$$

Finally, we get K_{002} and K_{012} from this:

$$
\begin{aligned}
K_{002}^{\mathrm{CC}_\varphi^B} D &= K_{001}^{\mathrm{CC}_\varphi^B} D \cup K_{011}^{\mathrm{CC}_\varphi^B} D; (K_{111}^{\mathrm{CC}_\varphi^B} D)^*; K_{101}^{\mathrm{CC}_\varphi^B} D \\
&= K_{001}^{\mathrm{CC}_\varphi^B} D \quad \text{(since the right-hand expression evaluates to } ?\bot) \\
&= (?\varphi; (B \cap D))^* \\
K_{012}^{\mathrm{CC}_\varphi^B} D &= K_{011}^{\mathrm{CC}_\varphi^B} D \cup K_{011}^{\mathrm{CC}_\varphi^B} D; (K_{111}^{\mathrm{CC}_\varphi^B} D)^* \\
&= K_{011}^{\mathrm{CC}_\varphi^B} D; (K_{111}^{\mathrm{CC}_\varphi^B} D)^* \\
&= (?\varphi; (B \cap D))^*; (D \setminus B); D^*.
\end{aligned}
$$

Thus, the program transformation for common belief among D works out as follows:

$$
[\mathrm{CC}_\varphi^B, \mathsf{e}_0][D^*]\psi
$$

$$
\leftrightarrow
$$

$$
[(?\varphi; (B \cap D))^*][\mathrm{CC}_\varphi^B, \mathsf{e}_0]\psi \wedge [(?\varphi; (B \cap D))^*; (D \setminus B); D^*][\mathrm{CC}_\varphi^B, \mathsf{e}_1]\psi.
$$

This transformation yields a reduction axiom that shows that EL-RC also suffices to provide reduction axioms for secret group communication.

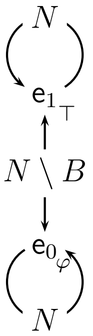
5.3 Group Messages and Common Knowledge

Finally, we consider group messages. This example is one of the simplest cases that shows that program transformations gives us reduction axioms that are no longer feasible to give by hand.

The update model for a group message to B that φ consists of two states $\mathsf{e}_0, \mathsf{e}_1$, where e_0 has precondition φ and e_1 has precondition $\top$, and where the accessibilities T are given by:

$$
\begin{aligned}
T = \; & \{\mathsf{e}_0 R(b) \mathsf{e}_0 \mid b \in B\} \cup \\
& \{\mathsf{e}_1 R(b) \mathsf{e}_1 \mid b \in B\} \cup \\
& \{\mathsf{e}_0 R(a) \mathsf{e}_1 \mid a \in N \setminus B\} \cup \\
& \{\mathsf{e}_1 R(a) \mathsf{e}_0 \mid a \in N \setminus B\}.
\end{aligned}
$$

This captures the fact that the members of B can distinguish the φ update from the $\top$ update, while the other agents (the members of $N \setminus B$) cannot. The actual event is e_0. Call this model G_φ^B.



A difference with the 'cc' case is that group messages are S5 models. Since updates of S5 models with S5 models are S5, group messages engender common knowledge (as opposed to mere common belief). Let us work out the program transformation that this update engenders for common knowledge among a group of agents D.

We will have to work out $K_{002}^{G_\varphi^B} D$, $K_{012}^{G_\varphi^B} D$, $K_{112}^{G_\varphi^B} D$, $K_{102}^{G_\varphi^B} D$.

For these, we need $K_{001}^{G_\varphi^B} D$, $K_{011}^{G_\varphi^B} D$, $K_{111}^{G_\varphi^B} D$, $K_{101}^{G_\varphi^B} D$.

For these in turn, we need $K_{000}^{G_\varphi^B} D$, $K_{010}^{G_\varphi^B} D$, $K_{110}^{G_\varphi^B} D$, $K_{100}^{G_\varphi^B} D$.

For these, we need:

$$\begin{aligned}
T_{00}^{G_\varphi^B} D &= \bigcup_{d \in D} (?\varphi; d) = ?\varphi; D, \\
T_{01}^{G_\varphi^B} D &= \bigcup_{d \in D \setminus B} (?\varphi; d) = ?\varphi; (D \setminus B), \\
T_{11}^{G_\varphi^B} D &= D, \\
T_{10}^{G_\varphi^B} D &= D \setminus B.
\end{aligned}$$

It follows that:

$$\begin{aligned}
K_{000}^{G_\varphi^B} D &= ?\top \cup (?\varphi; D), \\
K_{010}^{G_\varphi^B} D &= ?\varphi; (D \setminus B), \\
K_{110}^{G_\varphi^B} D &= ?\top \cup D, \\
K_{100}^{G_\varphi^B} D &= D \setminus B.
\end{aligned}$$

From this we can work out the K_{ij1}, as follows:

$$
\begin{aligned}
&K_{001}^{G_\varphi^B} D = (?\varphi; D)^*,\\
&K_{011}^{G_\varphi^B} D = (?\varphi; D)^*; ?\varphi; D \setminus B,\\
&K_{111}^{G_\varphi^B} D = ?\top \cup D \cup (D \setminus B; (?\varphi; D)^*; ?\varphi; D \setminus B),\\
&K_{101}^{G_\varphi^B} D = D \setminus B; (?\varphi; D)^*.
\end{aligned}
$$

Finally, we get K_{002} and K_{012} from this:

$$
\begin{aligned}
K_{002}^{G_\varphi^B} D &= K_{001}^{G_\varphi^B} D \cup K_{011}^{G_\varphi^B} D; (K_{111}^{G_\varphi^B} D)^*; K_{101}^{G_\varphi^B} D\\
&= (?\varphi; D)^* \cup\\
&\quad (?\varphi; D)^*; ?\varphi; D \setminus B; (D \cup (D \setminus B; (?\varphi; D)^*; ?\varphi; D \setminus B))^*;\\
&\qquad\qquad\qquad\qquad\qquad\qquad D \setminus B; (?\varphi; D)^*,\\
K_{012}^{G_\varphi^B} D &= K_{011}^{G_\varphi^B} D; (K_{111}^{G_\varphi^B} D)^*\\
&= (?\varphi; D)^*; ?\varphi; D \setminus B; (D \cup (D \setminus B; (?\varphi; D)^*; ?\varphi; D \setminus B))^*.
\end{aligned}
$$

Abbreviating $D \cup (D \setminus B; (?\varphi; D)^*; ?\varphi; D \setminus B)$ as π, we get the following transformation for common knowledge among D after a group message to B that φ:

$$
\begin{gathered}
[G_\varphi^B, \mathsf{e}_0][D^*]\psi\\
\leftrightarrow\\
[(?\varphi; D)^* \cup ((?\varphi; D)^*; ?\varphi; D \setminus B; \pi^*; D \setminus B; (?\varphi; D)^*)][G_\varphi^B, \mathsf{e}_0]\psi \wedge\\
[(?\varphi; D)^*; ?\varphi; D \setminus B; \pi^*][G_\varphi^B, \mathsf{e}_1]\psi.
\end{gathered}
$$

This formula makes it clear that, although we can translate every formula of LCC to PDL, higher order descriptions using update models are more convenient for reasoning about information change.

One interesting side-effect of this bunch of illustrations is that it demonstrates the computational character of our analysis. Indeed, the above axioms were found by a machine! Cf. [9] on the use of computational tools in exploring the universe of iterated epistemic updates.

6 Conclusion and Further Research

Dynamic-epistemic logics provide systematic means for studying exchange of factual and higher-order information. In this many-agent setting, common knowledge is an essential concept. We have presented two extended languages for dynamic-epistemic logic that admit explicit reduction axioms for common knowledge resulting from an update: one (PAL-RC) for public announcement only, and one (LCC with its static base E-PDL) for general scenarios with information flow. These systems make proof and complexity analysis for informative actions more perspicuous than earlier attempts in the literature. Still, PAL-RC and LCC are just two extremes on a spectrum, and many further natural update logics may lie in between. We conclude by pointing out some further research topics that arise on our analysis.

- *Downward in expressive power from* E-PDL. Which weaker language fragments are in 'dynamic-static harmony', in the sense of having reduction axioms for compositional analysis of update effects, and the corresponding meaning-preserving translation? Our program transformer approach does work also with certain restrictions on tests in our full logic LCC, but we have not yet been able to identify natural intermediate levels.
- *Upward in expressive power from* E-PDL. Which *richer* languages are in dynamic-static harmony? A typical candidate is the epistemic *μ-calculus*, which allows arbitrary operators for defining smallest and greatest fixed-points. Indeed, we have a proof that the results of Section 3 extend to the calculus PAL-μ for public announcements, which takes the complete epistemic *μ-calculus* for its static language (allowing no binding into announcement positions). Our conjecture is that our expressivity and axiomatization results of Section 4 also extend to the full μ-calculus version of LCC. Cf. [7] for a first proposal.
- *Other notions of group knowledge.* Another test of our methodology via reduction axioms are further notions of group knowledge. For instance, instead of common knowledge, consider *distributed group knowledge $D_B\varphi$* consisting of those statements which are available implicitly to the group, in the sense of φ being true at every world reachable from the current one by the intersection of all epistemic accessibility relations. The following simple reduction holds for public announcements: $[\varphi]D_B\psi \leftrightarrow (\varphi \rightarrow D_B[\varphi]\psi)$. We have not yet investigated our full system LCC extended with distributed knowledge.
- *Program constructions over update models.* One can add the usual regular operations of composition, choice, and iteration over update models, to obtain a calculus describing effects of more complex information-bearing events. It is known that this extension makes PAL undecidable, but what about partial axiomatizations in our style? One can also look at such extensions as moving toward a still richer *epistemic temporal logic* with future and past operators over universes of finite sequences of events starting from some initial model. This would be more in line with the frameworks of [13,25], to which our analysis might be generalized.
- *Alternative questions about update reasoning.* With one exception, our reduction axioms are all *schematically valid* in the sense that substituting arbitrary formulas for proposition letters again yields a valid formula. The

exception is the base clause, which really only holds for atomic proposition letters p. As discussed in [4], this means that certain schematically valid laws of update need not be derivable from our axioms in an explicit schematic manner, even though all their concrete instances will be by our completeness theorem. An example is the schematic law stating the associativity of successive announcements. It is not known whether schematic validity is decidable, even for PAL, and no complete axiomatization is known either. This is just one instance of open problems concerning PAL and its ilk for public announcement (cf. the survey in [5]), which all return for LCC, with our program transformations as a vehicle for generalizing the issues.

- *Belief revision.* Even though our language can describe agents' beliefs, the product update mechanism does not describe genuine belief revision. New information which contradicts current beliefs just leads to inconsistent beliefs. There are recent systems, however, which handle belief revision as update of plausibility rankings in models [6]. But so far, these systems only handle beliefs of single agents. Our analysis of common belief might be added on top of these, to create a more richly structured account of 'group-based belief revision'.
- *General logical perspectives.* Languages with relativizations are very common in logic. Indeed, closure under relativization is sometimes stated as a defining condition on logics in abstract model theory. Basic modal or first-order logic as they stand are closed under relativizations $[A]\varphi$, often written $(\varphi)^A$. The same is true for logics with fixed-point constructions, like PDL (cf. [3]) or the modal μ-calculus. E.g., computing a relativized least fixed-point $[A]\mu p.\varphi(p)$ works much as evaluation of $\mu p.\varphi(p) \wedge A$ – which actually suggests a corresponding dynamic epistemic reduction axiom (cf. [7]). The setting of Section 4 lifts relativization to some sort of 'update closure' for general logical languages, referring to *relative interpretations* in definable submodels of products. Languages with this property include again first-order logic and its fixed-point extensions, as well as fragments of the μ-calculus, and temporal **UNTIL** logics.

Acknowledgements Tomasz Sadzik and other participants of the *Stanford-ILLC Seminar* in December 2004 made many helpful comments and asked inspiring questions. Floris Roelofsen performed a careful proof-reading of our manuscript. Three anonymous referees of this Journal gave valuable feedback on presentation and reported errors. Thank you all.

References

[1] A. Baltag, L. S. Moss, and S. Solecki. The logic of public announcements, common knowledge, and private suspicions. Technical Report SEN-R9922,

CWI, Amsterdam, 1999. Many updates.

[2] A. Baltag and L.S. Moss. Logics for epistemic programs. *Synthese*, 139(2):165–224, 2004.

[3] J. van Benthem. Information update as relativization. ILLC, University of Amsterdam, 2000. Available from `http://staff.science.uva.nl/~johan/upd=Rel.pdf`.

[4] J. van Benthem. One is a lonely number: on the logic of communication. Technical Report PP-2002-27, ILLC, Amsterdam, 2002. To appear in P. Koepke et al., ed. *Colloquium Logicum, Münster, 2002*, AMS Publications, Providence.

[5] J. van Benthem. Open problems in logical dynamics. ILLC Prepublication PP-2005-06, ILLC Amsterdam, 2005.

[6] J. van Benthem. Dynamic logics of belief revision. ILLC, University of Amsterdam, 2006. Available from `http://staff.science.uva.nl/~johan/DL-BR-new.pdf`.

[7] J. van Benthem, J. van Eijck, and B. Kooi. A fixed-point look at update logic, 2005. Available from `staff.science.uva.nl/~johan/Redux-Fix++.pdf`.

[8] H. P. van Ditmarsch. *Knowledge Games*. PhD thesis, ILLC, Amsterdam, 2000.

[9] J. van Eijck. Dynamic epistemic modelling. Technical Report SEN-E0424, CWI, Amsterdam, December 2004. Available from `http://db.cwi.nl/rapporten/`.

[10] J. van Eijck. Guarded actions. Technical Report SEN-E0425, CWI, Amsterdam, December 2004. Available from `http://db.cwi.nl/rapporten/`.

[11] J. van Eijck. Reducing dynamic epistemic logic to PDL by program transformation. Technical Report SEN-E0423, CWI, Amsterdam, December 2004. Available from `http://db.cwi.nl/rapporten/`.

[12] J. van Eijck, J. Ruan, and Th. Sadzik. Action emulation, 2006. Available from `www.cwi.nl/~jve/papers/06/ae/ae.pdf`.

[13] R. Fagin, J.Y. Halpern, Y. Moses, and M.Y. Vardi. *Reasoning About Knowledge*. MIT Press, Cambridge, Massachusetts, 1995.

[14] J. Gerbrandy. *Bisimulations on Planet Kripke*. ILLC Dissertation Series, Amsterdam, 1999.

[15] D. Harel. Dynamic logic. In D. Gabbay and F. Guenther, editors, *Handbook of Philosophical Logic Volume II: extensions of classical logic*, volume 165 of *Synthese Library Studies in epistemology, logic, methodology, and philosophy of science*, chapter 10, pages 497–604. D. Reidel Publishing Company, Dordrecht Boston Lancaster, 1984.

[16] D. Harel, D. Kozen, and J. Tiuryn. *Dynamic Logic*. Foundations of Computing. MIT Press, Cambridge, Massachusetts, 2000.

[17] S. C. Kleene. Representation of events in nerve nets and finite automata. In C. E. Shannon and J. McCarthy, editors, *Automata Studies*, pages 3–41. Princeton University Press, Princeton, N.J, 1956.

[18] B. Kooi and J. van Benthem. Reduction axioms for epistemic actions. In R. Schmidt, I Pratt-Hartmann, M. Reynolds, and H. Wansing, editors, *AiML-2004: Advances in Modal Logic*, number UMCS-04-9-1 in Technical Report Series, pages 197–211. University of Manchester, 2004.

[19] H.R. Lewis and C.H. Papadimitriou. *Elements of the Theory of Computation*. Prentice-Hall, 1981.

[20] C. Lutz. Complexity and succinctness of public announcement logic. In *Proceedings AAMAS-06: Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*, Hakodate, Japan, 2006.

[21] R. van der Meyden. Constructing finite state implementations of knowledge based programs with perfect recall. In *Intelligent Agent Systems: Theoretical and Practical Issues*, volume 1209 of *Lecture Notes in Artificial Intelligence*, pages 135–152. Springer, 1997.

[22] J.-J.Ch. Meyer and W. van der Hoek. *Epistemic Logic for AI and Computer Science*. Cambridge University Press, Cambridge, 1995.

[23] J.S. Miller and L.S Moss. The undecidability of iterated modal relativization. *Studia Logica*, 79(3):373–407, 2005.

[24] R. Parikh. Logics of knowledge, games and dynamic logic. In M. Joseph, editor, *Foundations of Software Technology and Theoretical Computer Science*, pages 202–222. Springer, 1984.

[25] R. Parikh and R. Ramanujan. A knowledge based semantics of messages. *Journal of Logic, Language and Information*, 12:453–467, 2003.

[26] J. A. Plaza. Logics of public communications. In M. L. Emrich, M. S. Pfeifer, M. Hadzikadic, and Z. W. Ras, editors, *Proceedings of the 4th International Symposium on Methodologies for Intelligent Systems*, pages 201–216, 1989.

[27] J. Ruan. Exploring the update universe. Master's thesis, ILLC, Amsterdam, 2004.

[28] H. Tuominen. Translations from epistemic into dynamic logic. In Yves Kodratoff, editor, *ECAI-88: Proceedings of the 8th European conference on Artificial Intelligence*, pages 586–588, 1988.