Action Emulation

Jan van Eijck (jve@cwi.nl)
CWI and ILLC, Amsterdam, the Netherlands

Ji Ruan (ji@jiruan.net)
The University of New South Wales, Australia

Tomasz Sadzik (tsadzik@nyu.edu)
New York University, USA

Abstract. The effects of public announcements, private communications, deceptive messages to groups, and so on, can all be captured by a general mechanism of updating multi-agent models with update action models, now in widespread use. There is a natural extension of the definition of a bisimulation to action models. Surely enough, updating with bisimilar action models gives the same result (modulo bisimulation). But the converse turns out to be false: update models may have the same update effects without being bisimilar. We propose action emulation as a notion of equivalence more appropriate for action models, and generalizing standard bisimulation. It is proved that action emulation provides a full characterization of update effect. We first concentrate on the general case, and next focus on the important case of action models with propositional preconditions. Our notion of action emulation yields a simplification procedure for action models, and it gives designers of multi-agent systems a useful tool for comparing different ways of representing a particular communicative action.

1. Introduction

Knowledge, knowledge about knowledge, lack of knowledge about knowledge, all play a key role in the interaction of agents. In systems that handle communication where not all information is shared equally, the effects on knowledge can easily become quite complicated: witness the effects of sending emails with bcc lists, coupled with the unreliability of the server, or resending an acknowledgment of receipt. To reason about such systems one needs powerful logics that can express and compare the effects of various communicative actions.

In epistemic logic [11] knowledge is represented with multi-agent Kripke models (or possible world models) that contain for each agent an accessibility relation pointing at the situations that the agent considers possible. To talk about what is the case in such models, a logical language is used that allows one to express things like 'agent a considers ϕ possible' (this would express that ϕ is consistent with what a knows or believes), or 'in all states that are linked to the current state via a and

b accessibilities, ϕ is the case' (this would express common knowledge of a and b that ϕ).

While standard epistemic logics do not directly represent acts of communication, Dynamic Epistemic Logic (DEL) does. It introduces the representation of *actions* together with a method of updating a situation with these actions. It also introduces action modalities for describing effects of action update. For an overview of developments in these areas, consult Gerbrandy [13], van Ditmarsch [8], van Benthem [4, 5], Baltag, Moss and coworkers [3, 1, 2], and the textbook treatment in [10]. In the paper we will work with the logic of communication and change (LCC) of van Benthem, van Eijck and Kooi [6], which is one of the most expressive versions of DEL. LCC consists of propositional dynamic logic [19, 14] with added action modalities.

The basic insight of DEL is from [3]: a wide variety of information updates can be treated using a formal product construction with an action model, which is nothing but a multi-agent Kripke model with the valuations replaced by precondition formulas. The reason for this to work is that actions with epistemic effects are quite similar to situations with epistemic aspects. The uncertainty of agents about which action takes place is a lot like the uncertainty of agents about what is the case.

If you receive a message ϕ and I am left in the dark, then this is modeled as an action that allows you to distinguish the ϕ situations from the rest, while I am not allowed to make that distinction. If the two of us get the ϕ message, and some outsider does not, then it makes a real difference whether the two of us know of each other that we get the same information, and this again is encoded in the action model.

Since action model updating is an attractive mechanism for modeling communicative action, it is important for multi-agent system design to have means of comparing different ways of representing a particular communicative action. In this paper, we study equivalence of action models: two action models are equivalent if they always produce non-distinguishable results. Our contribution is a concept called *action emulation*, and a proof that this precisely characterizes this equivalence.

The structure of the paper is as follows. In Section 2, we review the version of Dynamic Epistemic Logic we work with, motivate our choice, and define our basic notions. Section 3 gives a definition of equivalence or 'same update effect' for action models that we want to capture, compares this notion to that of bisimulation for action models, and gives examples to show that these notions do not quite match. Then, after some preliminaries in Section 4, we propose a general structural notion of action emulation in Section 5, and show that action equivalence implies existence of an action emulation, and vice versa. The proposed notion is rather involved, but in Section 6 we show that it can be simplified for the case of action models with propositional preconditions. The section ends with examples of action models where the simplified characterization fails. Section 7 gives discussion and questions for further research.

2. Dynamic Epistemic Logic

In this section we formally introduce epistemic models (or multi-agent Kripke models), followed by definitions of action models and a suitable epistemic language. Next, we define the process of updating with an action model and the notion of truth in a model.

Epistemic models capture a static description of what agents know about the world and about each other, action models capture the instructions for modifying these static systems. In all definitions we assume that a finite set of agents Ag and a set of propositional variables $Prop$ are given.

DEFINITION 1. (Epistemic Model). *An epistemic model is a triple* $M = (W, V, \rightarrow)$ *where* W *is a non-empty set of worlds,* $V : W \rightarrow \mathcal{P}(\text{Prop})$ *assigns a valuation to each world* $w \in W$*, and* $\rightarrow: \text{Ag} \rightarrow \mathcal{P}(W^2)$ *assigns an accessibility relation* $\xrightarrow{i}$ *to each agent* $i \in \text{Ag}$.

A pointed epistemic model is a pair (M, u) *where* M *is an epistemic model and* u *is an element of* W_M*. The intended interpretation of the distinguished point* u *is that* u *represents the actual world.*

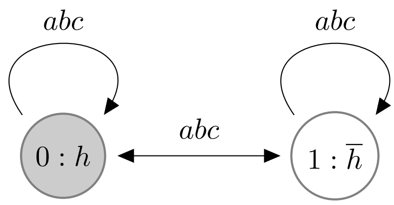


Figure 1. Epistemic model representing the result of a hidden coin toss, where the coin shows heads, but none of the agents sees this.

If M is an epistemic model, we use W_M to refer to its set of worlds, V_M to refer to its valuation function, and $\rightarrow_M$ to refer to its accessibility function. Figure 1 gives an example of an epistemic model that describes the result of a hidden coin toss, with three onlookers, Alice, Bob and Carol. The model has an actual situation, marked in grey. Presence of proposition letter h in a world indicates that the valuation makes h true in that world, presence of $\overline{h}$ in a world indicates that the valuation makes h false in that world, so the picture reveals that the coin has landed heads up in the actual world 0, tails up in world 1. The epistemic accessibility relations are indicated by arrows, with labels indicating the agents. None of the agents can tell these two worlds apart. Singling out 0 as distinguished point tells us that the actual world is world 0.

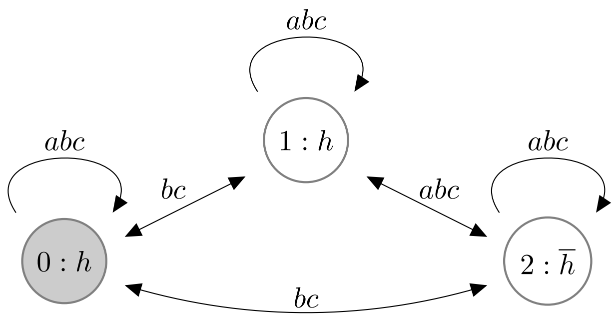


Figure 2. Epistemic model representing that the result of a hidden coin toss is *heads*, agent a knows this, while agents b and c do not but hold it for possible that a knows it.

Figure 2 gives a situation like that of Figure 1, but where agent a knows that the coin has landed heads up, while the other agents don't know it but hold it for possible that a knows (and also hold it for possible that a does not know).

A message to a that the coin has landed *heads* up, while the others hold it for possible that a receives that message, can be viewed as an action where a can make a distinction that b and c cannot make. It changes the model from Figure 1 into that of Figure 2.

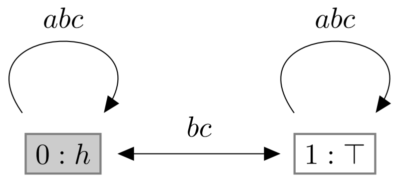


Figure 3. Action model for an observation by a that the the coin landed *heads* up.

Baltag, Moss and Solecki [3] proposed to model update actions on epistemic models as taking a product with action models, where action models are like epistemic models, but with valuations replaced by precondition formulas. In the example of Figure 3, the actual action (in grey) is that formula h is checked. The agents b and c cannot distinguish this action from an action where nothing is checked. The result of updating with this action model should be a situation where a may have learnt h, and where b and c know this. In other words, updating the epistemic model in Figure 1 with the action model in Figure 3 should yield something 'essentially equivalent' to the epistemic model in Figure 2, with the notion of 'essentially equivalent' being the notion of bisimulation defined later in the paper.

DEFINITION 2. (Action Model). *An action model for a language $\mathcal{L}$ is a triple $A = (W, \mathsf{pre}, \rightarrow)$ where W is a non-empty set of action states, $\mathsf{pre} : W \rightarrow \mathcal{L}$ assigns a consistent precondition formula pre_s (in $\mathcal{L}$) to each action state $s \in W$, and $\rightarrow: \mathrm{Ag} \rightarrow \mathcal{P}(W^2)$ assigns an accessibility relation $\stackrel{i}{\rightarrow}$ to each agent $i \in \mathrm{Ag}$.*

A pointed action model for a language $\mathcal{L}$ is a pair (A, s) where A is an action model for $\mathcal{L}$ and s is a member of W_A, indicating that s is the action that actually takes place.

Consistency will be defined below (Definition 5). Similarly to the case of epistemic models, we use W_A for the set of action states of action model A, pre_A for its precondition function, and $\rightarrow_A$ for its accessibility function.

The epistemic language $\mathcal{L}_1$ that we are going to use for the preconditions is epistemic PDL with action modalities. It is defined as follows.

DEFINITION 3. (Epistemic Languages $\mathcal{L}_0$ and $\mathcal{L}_1$). *Let p range over the set of basic propositions* Prop *and i over the set of agents* Ag. *The formulas of $\mathcal{L}_1$ are given by:*

$$\begin{aligned} \phi &::= \top \mid p \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid [\alpha]\phi \mid [A, s]\phi \\ \alpha &::= i \mid ?\phi \mid \alpha_1 \cup \alpha_2 \mid \alpha_1 ; \alpha_2 \mid \alpha^* \end{aligned}$$

where A is an action model for $\mathcal{L}_1$, and $s \in W_A$. Let $\mathcal{L}_0$ be the result of removing all formulas which have a sub-formula of the form $[A, s]\phi$ from the language.

Note that in the definition of the grammar $\mathcal{L}_1$, a sub-recursion occurs for (A, s) since the preconditions in (A, s) themselves are in $\mathcal{L}_1$. We employ the usual abbreviations. In particular, $\bot$ is shorthand for $\neg\top$, $\phi_1 \vee \phi_2$ for $\neg(\neg\phi_1 \wedge \neg\phi_2)$, $\phi_1 \rightarrow \phi_2$ for $\neg(\phi_1 \wedge \neg\phi_2)$, $\langle\alpha\rangle\phi$ for $\neg[\alpha]\neg\phi$, $\langle A, s\rangle\phi$ for $\neg[A, s]\neg\phi$. Also, we will use $\bigvee\{\phi_1, \ldots, \phi_n\}$ for $\phi_1 \vee \cdots \vee \phi_n$ and $\bigwedge\{\phi_1, \ldots, \phi_n\}$ for $\phi_1 \wedge \cdots \wedge \phi_n$.

Below we will establish results for preconditions in $\mathcal{L}_0$ only. This will establish results for the $\mathcal{L}_1$ as well: Switching to the richer language is without loss of generality, for it is proved in [6] that LCC (our language $\mathcal{L}_1$) has the same expressive power as epistemic PDL (our language $\mathcal{L}_0$):

THEOREM 1. *The language of LCC (epistemic PDL with added action modalities) has the same expressive power as epistemic PDL itself: each formula ϕ of LCC has an equivalent formula ϕ° in epistemic PDL.*

We note that the translation of ϕ to ϕ° uses a technique of PDL program transformation, which involves an exponential blow-up [15].

Remark. While throughout the paper we restrict the preconditions to be in language $\mathcal{L}_0$, it should be noted that the definition of action emulation and the proofs of the theorems in the paper can be adapted to other epistemic languages.

If one adds action models to an epistemic or doxastic language $\mathcal{L}$, this means that the language is extended with action modalities. Call this extended language $\mathcal{L}^+$. The action models for $\mathcal{L}$ can be of two kinds, depending on whether the preconditions are taken from $\mathcal{L}$ or from $\mathcal{L}^+$. In the first case, the preconditions themselves cannot contain action modalities, in the second case they can.

Our methods deal with action models of both kinds. For action models of the second kind, the trick is to use epistemic PDL as an auxiliary language. Epistemic PDL has enough expressive power to encode the effects of any action model modality. By adopting PDL as auxiliary language, we can deal with action models with preconditions that may themselves contain action modalities. Theorem 1 ensures not only that any LCC precondition has a PDL counterpart, but also that any precondition in a sublanguage of epistemic PDL enriched with action model modalities has a PDL counterpart.

So suppose we want to handle action models of the second kind for a language that is not expressive enough to encode its own action modalities, say a language $\mathcal{C}K$ with operators for knowledge and common knowledge. Then we use PDL as an auxiliary language to translate $\mathcal{C}K^+$ into PDL, and use the canonical model construction for PDL to define an appropriate notion of action emulation for $\mathcal{C}K^+$ models. It follows that we can deal with action models for any reasonable epistemic base language.

What is crucial for the definition and the proofs is the existence for any finite and consistent set of formulas in the language of a finite canonical model (built of atoms, as in Definition 12) that satisfies the Truth Lemma.[1] In fact, Definition 15 (Action Emulation) can be simplified in cases where the preconditions are in sublanguages of $\mathcal{L}_0$ that give rise to canonical models with a simpler structure. An example of this will be presented in Section 6.

In Section 5 we will give a definition of pointed action emulation that relates the distinguished points of two action models to each other.

[1] We thank one of our anonymous Referees for pointing this out.

The update operation $\otimes$ and the truth definition for $\mathcal{L}_1$ are defined by mutual recursion, as follows. (See [3] for the original version.)

DEFINITION 4. (Update, Truth). *Given a pointed epistemic model* (M, u) *and a pointed action model* (A, s), *and provided* $M \models_u \mathsf{pre}_s$, *we define*

$$M \otimes A$$

as

$$(W', V', \to'),$$

where

$$\begin{array}{lll} W' & = & \{(w,s) \in W_M \times W_A \mid M \models_w \mathsf{pre}_s\}, \\ V'((w,s)) & = & V_M(w), \\ (w,s) \stackrel{i}{\to}' (w',s') & \textit{iff} & w \stackrel{i}{\to}_M w', \textit{ and } s \stackrel{i}{\to}_A s', \end{array}$$

and where the truth definition is given by:

$$\begin{array}{ll} M \models_w \top & \textit{always} \\ M \models_w p & \textit{iff } p \in V_M(w) \\ M \models_w \neg\phi & \textit{iff not } M \models_w \phi \\ M \models_w \phi_1 \wedge \phi_2 & \textit{iff } M \models_w \phi_1 \textit{ and } M \models_w \phi_2 \\ M \models_w [\alpha]\phi & \textit{iff for all } w' \textit{ with } w \stackrel{\alpha}{\to} w' \; M \models_{w'} \phi \\ M \models_w [A,s]\phi & \textit{iff } M \models_w \mathsf{pre}_s \textit{ implies } M \otimes A \models_{(w,s)} \phi \end{array}$$

with $\stackrel{\alpha}{\to}$ *given by*

$$\begin{array}{lll} \stackrel{i}{\to} & = & \stackrel{i}{\to}_M \\ \stackrel{?\phi}{\to} & = & \{(x,x) \mid M \models_x \phi\} \\ \stackrel{\alpha_1 \cup \alpha_2}{\to} & = & \stackrel{\alpha_1}{\to} \cup \stackrel{\alpha_2}{\to} \\ \stackrel{\alpha_1;\alpha_2}{\to} & = & \stackrel{\alpha_1}{\to} \circ \stackrel{\alpha_2}{\to} \quad \textit{(relational composition of } \stackrel{\alpha_1}{\to} \textit{ and } \stackrel{\alpha_2}{\to}\textit{)} \\ \stackrel{\alpha^*}{\to} & = & (\stackrel{\alpha}{\to})^* \quad \textit{(reflexive transitive closure of } \stackrel{\alpha}{\to}\textit{).} \end{array}$$

The new distinguished point of $M \otimes A$ *is* (u, s).

Note that the updating operation may not succeed. This happens if $M \not\models_u \mathsf{pre}_s$. But if the updating operation succeeds, the result is a well-defined epistemic model.

As an illustration, Figure 4 gives the result of updating the epistemic model from Figure 1 with the action model from Figure 3, with the worlds in the update result pictured as pairs.

We still owe you definitions of consistency, logical equivalence and logical entailment.

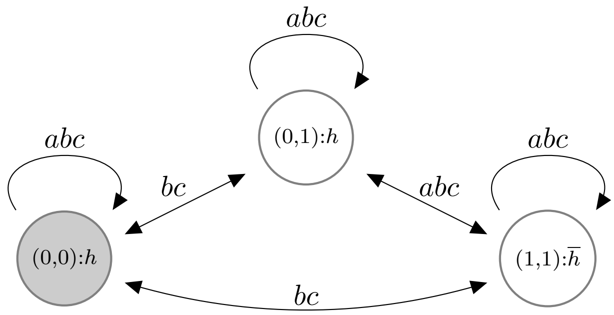


Figure 4. Result of updating model from Figure 1 with action model from Figure 3.

DEFINITION 5. (Consistency, Logical Equivalence, Logical Entailment). *Let ϕ and ψ be two formulas in a language $\mathcal{L}$.*

- *ϕ is consistent if there is an epistemic model M and a world w such that $M \models_w \phi$;*

- *ϕ and ψ are logically equivalent (notation: $\phi \equiv \psi$), if for arbitrary epistemic models M and worlds w, $M \models_w \phi \leftrightarrow \psi$.*

- *ϕ logically entails ψ (notation: $\phi \models \psi$) if it holds for an arbitrary epistemic model M and world w that $M \models_w \phi$ implies $M \models_w \psi$. (This is called 'local consequence' in modal logic.)*

Note that our notion of consistency is semantic (it is not defined as non-existence of a derivation of $\phi \to \bot$ in a proof system, but as existence of a model for ϕ). Also, note that the language $\mathcal{L}_1$ has the finite model property, hence it is decidable whether ϕ-models exists, for any $\phi \in \mathcal{L}_1$.

3. Bisimulation and Action Equivalence

The standard notion of structural equivalence for epistemic models is bisimulation.

DEFINITION 6. (Bisimulation). *Let M, N be epistemic models. A non-empty relation $C \subseteq W_M \times W_N$ is a bisimulation if whenever wCv the following hold:*

Invariance *$V_M(w) = V_N(v)$;*

Zig *for all* $i \in \mathrm{Ag}$*, all worlds* $w' \in W_M$ *with* $w \xrightarrow{i} w'$ *there is a state* $v' \in W_N$ *with* $v \xrightarrow{i} v'$ *and* $w'Cv'$*;*

Zag *for all* $i \in \mathrm{Ag}$*, all worlds* $v' \in W_N$ *with* $v \xrightarrow{i} v'$ *there is a state* $w' \in W_M$ *with* $w \xrightarrow{i} w'$ *and* $w'Cv'$*.*

We write $M \underline{\leftrightarrow} N$ *to indicate that there is a bisimulation that connects every world in* W_M *to some world in* W_N*, and vice versa.*

A pointed bisimulation between pointed epistemic models (M, x) *and* (N, y) *is a bisimulation* C *between* M *and* N *that connects* x *and* y*. Existence of a pointed bisimulation between* (M, x) *and* (N, y) *is indicated by* $(M, x)\underline{\leftrightarrow}(N, y)$*.*

Bisimilarity implies indistinguishability for $\mathcal{L}_1$: if $(M, x)\underline{\leftrightarrow}(N, y)$ and ϕ is a formula of $\mathcal{L}_1$ then $M \models_x \phi$ iff $N \models_y \phi$. This follows directly from the fact that bisimilarity implies indistinguishability for $\mathcal{L}_0$, plus the reducibility result of [6] (Theorem 1 above).

While the invariance requirement in the definition of bisimulation can be applied only to epistemic models, a natural analogue for action models suggests itself: simply replace 'having the same valuation' by 'having equivalent preconditions'. Since the only difference between epistemic models and action models is in the switch from valuations to preconditions, this seems an obvious choice. A demand of syntactic equality of presuppositions would be too strong, but logical equivalence seems just right. This gives:

DEFINITION 7. (Bisimulation for Action Models). *Let* A, B *be action models. A non-empty relation* $C \subseteq W_A \times W_B$ *is a bisimulation if whenever* sCt *the following hold:*

Invariance $\mathsf{pre}_s \equiv \mathsf{pre}_t$*;*

Zig *for all* $i \in \mathrm{Ag}$ *and all states* $s' \in W_A$ *with* $s \xrightarrow{i} s'$ *there is a state* $t' \in W_B$ *with* $t \xrightarrow{i} t'$ *and* $s'Ct'$*;*

Zag *for all* $i \in \mathrm{Ag}$ *and all states* $t' \in W_B$ *with* $t \xrightarrow{i} t'$ *there is a state* $s' \in W_A$ *with* $s \xrightarrow{i} s'$ *and* $s'Ct'$*.*

We use notation $A \underline{\leftrightarrow} B$ *and* $(A, s) \underline{\leftrightarrow} (B, t)$ *analogous to the usage in Definition 6.*

Thinking of the action models as 'update programs', the basic semantic notion of equivalence between such programs is that of having the same update effect: two pointed action models are equivalent if applied to the same epistemic model, they yield bisimilar results. Formally:

DEFINITION 8. (Action Equivalence). *Two action models A and B are equivalent, notation $A \equiv B$, if it holds for all epistemic models M that*

$$M \otimes A \leftrightarroweq M \otimes B.$$

Two pointed action models (A, s) and (B, t) are equivalent (notation $(A, s) \equiv (B, t)$) if pre_s and pre_t are logically equivalent, and moreover it holds for all pointed models (M, w) with $M \models_w \mathsf{pre}_s$ that $(M \otimes A, (w, s)) \leftrightarroweq (M \otimes B, (w, t))$.

We would like to capture this notion of equivalence by means of a more direct relation on the structures of action models. Here is a first observation.

OBSERVATION 1. *The equivalence of two action models does not imply their bisimilarity.*

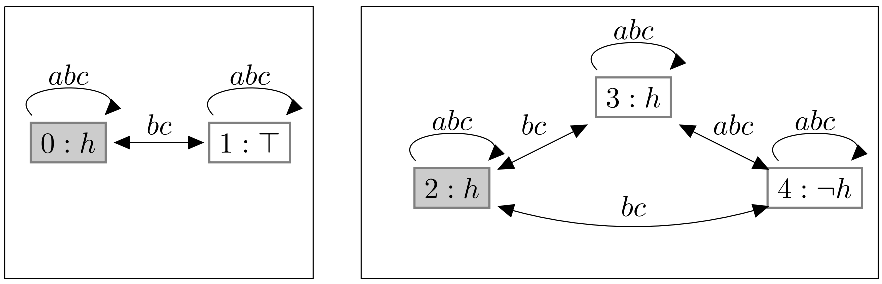


Figure 5. A pair of equivalent, but non-bisimilar action models.

Figure 5 provides an example of two action models for which there is no pointed bisimulation. The distinguished points of the two action models have the same precondition, but the step $0 \xrightarrow{a} 1$ in the left action model cannot be matched by a step from distinguished point 2 in the right action model, for that model has no states with precondition $\top$. Still, the update effects of the two action models are the same. Both represent an action where a finds out that the result of a coin toss is h, while b and c are uncertain about whether a has learned h or has found out nothing at all.

The example shows that action model bisimulation is not quite what we need. What we are looking for is some suitable generalization, and in Section 5 we propose action emulation as such a generalization. This notion has a certain family resemblance to bisimulation, but it turns out that this family likeness is partly hidden from sight by the fact that precondition formulas assigned to states in the action models may contain modalities.

For the special case of propositional action models (action models with formulas of propositional logic as preconditions) the resemblance to bisimulation is much closer. We will treat this special case in Section 6.

4. Filtration and Canonical Models

Our goal in Section 5 will be to propose a general notion of action emulation and prove that it exactly captures action equivalence for action models with arbitrary preconditions in $\mathcal{L}_0$. For this goal we need a technique (called filtration) for constructing models from sets of formulas. The filtration technique in modal logic is used to construct a finite model for a consistent modal formula ϕ (see [7]). For ordinary modal logic the construction is based on the set of all sub-formulas of ϕ, but in PDL we have to be careful in the handling of formulas with complex modalities α, so we need so-called Fischer-Ladner closures [12]. For completeness of the presentation, in this section we provide a construction of finite canonical models for PDL. The additional condition we impose on those models is that the states have different valuations - see Definition 13.

DEFINITION 9. *Let Σ be a set of $\mathcal{L}_0$ formulas. Then* FL(Σ)*, the Fischer-Ladner closure of Σ, is the smallest set of formulas X that has $\Sigma \subseteq X$, that is closed under taking sub-formulas, and that satisfies the following constraints:*

- *if $[\alpha \cup \alpha']\phi \in X$ then $[\alpha]\phi \in X$ and $[\alpha']\phi \in X$,*
- *if $[\alpha;\alpha']\phi \in X$ then $[\alpha][\alpha']\phi \in X$,*
- *if $[\alpha^*]\phi \in X$ then $[\alpha][\alpha^*]\phi \in X$.*

Note that the definition handles the actual formulas of the language, not their abbreviations. E.g., consider $\Sigma = \{[(a \cup b)^*]h\}$. Then,

$$FL(\Sigma) = \{[(a \cup b)^*]h, [(a \cup b)][(a \cup b)^*]h, [a][(a \cup b)^*]h, [b][(a \cup b)^*]h, h\}.$$

DEFINITION 10. (Closure under single negation). *For any formula ϕ, define $\sim\phi$, the single negation of ϕ, as follows: if ϕ has the form $\neg\psi$ then $\sim\phi = \psi$, otherwise $\sim\phi = \neg\phi$. Then $\sim\phi$ forms the negation of ϕ, while cancelling double negations. A set of formulas X is* closed under single negations *if $\phi \in X$ implies $\sim\phi \in X$.*

DEFINITION 11. (Closure of Σ). *For any formula set Σ, the closure of Σ, notation* $\sim\mathrm{FL}(\Sigma)$ *is the smallest set X which contains* $\mathrm{FL}(\Sigma)$ *and is closed under single negations.*

As an example, observe that the closure of $\{[(a \cup b)^*]h\}$ consists of the union of $FL(\{[(a \cup b)^*]h\})$ and the set of all negations of formulas in $FL(\{[(a \cup b)^*]h\})$. In building epistemic models from sets of formulas Σ we can take worlds to be maximal consistent sets of formulas taken from $\sim FL(\Sigma)$.

DEFINITION 12. *Let Σ be a set of formulas. A set of formulas Γ is an* atom *over Σ if Γ is a maximal consistent subset of* $\sim\mathrm{FL}(\Sigma)$. *Let* $\mathrm{At}(\Sigma)$ *be the set of all atoms over Σ.*

It is easy to show for every consistent formula $\phi \in \sim FL(\Sigma)$ there is a $\Gamma \in At(\Sigma)$ with $\phi \in \Gamma$ (see [7]). For any finite formula set Γ, let $\widehat{\Gamma} = \bigwedge \Gamma$.

DEFINITION 13. *Let Σ be a finite set of formulas and Q_Σ be the set of all propositional letters occurring in Σ. Let $\Gamma_1, \ldots, \Gamma_n$ be an enumeration of* $\mathrm{At}(\Sigma)$ *and let $X = \{x_1, \ldots, x_n\}$ be a set of proposition letters that do not occur in Σ. The canonical model M_Σ over Σ (and X) is given by:*

$$\begin{array}{lcl} W_\Sigma & = & \mathrm{At}(\Sigma); \\ V_\Sigma(\Gamma_i) & = & (\Gamma \cap Q_\Sigma) \cup \{x_i\}; \\ \rightarrow_\Sigma (i) & = & \{(\Gamma, \Gamma') \mid \widehat{\Gamma} \wedge \langle i \rangle \widehat{\Gamma}' \textit{ is consistent } \}. \end{array}$$

Note that the valuation V_Σ gives every Γ a unique set of propositions. This is important as we will use it in the proof of Proposition 2 in the next section.

See [7] for a proof that the canonical model 'works', in the sense that we can prove the following:

LEMMA 1. (Truth Lemma). *For all atoms* $\Gamma \in \mathrm{At}(\Sigma)$ *and all* $\phi \in \sim\mathrm{FL}(\Sigma)$ *it is the case that* $M_\Sigma \models_\Gamma \phi$ *iff* $\phi \in \Gamma$.

Worlds in arbitrary Kripke models correspond to worlds in canonical models via the following definition:

DEFINITION 14. *Let M be an arbitrary Kripke model. Let Σ be a set of formulas. Let v be a member of W_M. We define a map from v to a maximal consistent set of formulas of the closure of Σ, as follows:*

$$v^* = \{\phi \in \sim\mathrm{FL}(\Sigma) \mid M \models_v \phi\}.$$

This definition will be used in Theorem 2 in the next Section.

5. Action Emulation: The General Case

In this Section we give a definition of Action Emulation for the case of action models with preconditions taken from the language $\mathcal{L}_0$. This immediately generalizes to action models with preconditions taken from the language $\mathcal{L}_1$, i.e., to action models with preconditions that themselves may contain action model modalities. The reason is that, as already mentioned, action model modalities present in $\mathcal{L}_1$ formulas can be 'compiled out', using the techniques of [6].

The crucial feature in our definition of action emulation is an indexing method by means of atoms in finite canonical models. The definition of action emulation will work for action models with preconditions taken from any modal language $\mathcal{L}$ that allows for the construction of finite canonical models for which a truth lemma can be proved.

Our inspiration for the definition of action emulation comes from the following theorem. Intuitively, it says that any two action models A and B are equivalent if and only if the results of updating a special canonical model are bisimilar.

THEOREM 2. *Given action models A and B for language $\mathcal{L}_0$, let Σ be the set of preconditions occurring in A or B, and let M_Σ be a canonical model over Σ. Then the following holds:*

$$A \equiv B \textit{ iff } M_\Sigma \otimes A \leftrightarrow M_\Sigma \otimes B.$$

Let $s \in W_A$, $t \in W_B$. Then:

$$(A, s) \equiv (B, t) \textit{ iff for all } \Gamma \in \mathrm{At}(\Sigma) \textit{ with } \mathsf{pre}_s \in \Gamma \textit{ or } \mathsf{pre}_t \in \Gamma : \\ (M_\Sigma \otimes A, (\Gamma, s)) \leftrightarrow (M_\Sigma \otimes B, (\Gamma, t)).$$

Proof. From left to right: by definition of '$\equiv$'.

For the right to left direction, assume $M_\Sigma \otimes A \leftrightarrow M_\Sigma \otimes B$. Let C be a relation witnessing this bisimulation. Let M be an arbitrary Kripke model. Then each $v \in W_M$ has a corresponding atom v^* in M_Σ (Definition 14). Define a relation C' on $W_{M \otimes A} \times W_{M \otimes B}$ by means of:

$$(u, x) C' (v, y) :\equiv u = v \text{ and } (u^*, x) C (u^*, y).$$

We show that C' is a bisimulation. Suppose $(u, x) C' (v, y)$. Then:

Invariance (u, x) and (v, y) have the same valuation since $u = v$.

Zig Let $(u,x)\xrightarrow{i}(u',x')$. It follows that $u\xrightarrow{i}u'$, $x\xrightarrow{i}x'$, and $M \models_{u'} \mathsf{pre}_{x'}$. So $\mathsf{pre}_{x'} \in u'^*$.

To show $(u^*,x)\xrightarrow{i}(u'^*,x')$, we only have to show $u^*\xrightarrow{i}u'^*$. But this is immediate from the fact that $M \models_u \widehat{u^*} \wedge \langle i\rangle\widehat{u'^*}$.

Now use the zig property of C (and the construction of u'^*) to conclude that there is a y' with $(u^*,y)\xrightarrow{i}(u'^*,y')$ and $(u'^*,x')C(u'^*,y')$. Then $y\xrightarrow{i}y'$, which together with $u\xrightarrow{i}u'$ gives $(u,y)\xrightarrow{i}(u',y')$ and $(u',x')C'(u',y')$. This proves the zig property of C'.

Zag Same reasoning vice versa.

Let (u,x) in $W_{M\otimes A}$ be arbitrary. Then $M \models_u \mathsf{pre}_x$, and therefore $\mathsf{pre}_x \in u^*$. By the properties of C, there is a $y \in W_A$ with $(u^*,x)C(u^*,y)$. It follows that $(u,x)C'(u,y)$. So for every (u,x) in $W_{M\otimes A}$ there is a y with $(u,x)C'(u,y)$. Similarly in the other direction. This proves $M \otimes A \leftrightarrows M \otimes B$.

For the second part, left to right: by definition of '$\equiv$'. For the right to left direction, define the bisimulation C' as before. Suppose that $M \models_w \mathsf{pre}_s$. It follows that $(w^*,s)C(w^*,t)$, and so $(w,s)C'(w,t)$. The case $M \models_w \mathsf{pre}_t$ is analogous. □

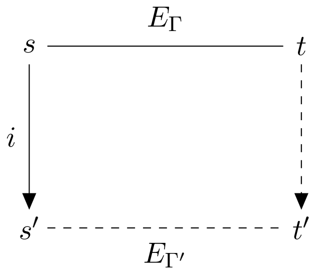
This theorem hints at what a general definition of action emulation '$\leftrightarrows$' might look like. Our next goal is to define $A \leftrightarrows B$ that will characterize action equivalence.

Our solution is to parametrize the relation $A \leftrightarrows B$ using maximal consistent sets from the domain of M_Σ.

DEFINITION 15. (Action Emulation). *Given action models A and B, let Σ be the set of preconditions occurring in A, B, and $G(x) = \{\Gamma \mid \Gamma \in At(\Sigma), \mathsf{pre}_x \in \Gamma\}$ for any $x \in W_A \cup W_B$. Action emulation E is a set of indexed relations $\{E_\Gamma\}_{\Gamma\in At(\Sigma)}$, such that whenever $sE_\Gamma t$ the following hold:*

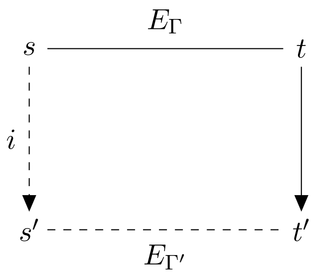
Invariance $\mathsf{pre}_s \in \Gamma$ *and* $\mathsf{pre}_t \in \Gamma$.

Zig *If $s \xrightarrow{i} s'$ and $\Gamma' \in G(s')$ such that $\Gamma \xrightarrow{i} \Gamma'$, then there is a $t' \in W_B$ with $t \xrightarrow{i} t'$ and $s'E_{\Gamma'}t'$. In a picture:*



Zag *If* $t \xrightarrow{i} t'$ *and* $\Gamma' \in G(t')$ *such that* $\Gamma \xrightarrow{i} \Gamma'$, *then there is a* $s' \in W_A$ *with* $s \xrightarrow{i} s'$ *and* $s' E_{\Gamma'} t'$.

In a picture:



We use $A \leftrightarrows B$ *to indicate the existence of a class of action emulation relations* $E_\Gamma \subseteq W_A \times W_B$ *such that for each* $x \in W_A$ *and each* $\Gamma \in At(\Sigma)$ *with* $\mathsf{pre}_x \in \Gamma$ *there is a* $y \in W_B$ *with* $x E_\Gamma y$, *and vice versa.*

We use $(A, s) \leftrightarrows (B, t)$ *to indicate that* pre_s *and* pre_t *are logically equivalent, and that there is a class of emulation relations* $E_\Gamma \subseteq W_A \times W_B$ *such that* $s E_\Gamma t$ *holds for every* Γ *with* $\mathsf{pre}_s \in \Gamma$.

Now we present our main results. The following proposition shows that emulating action models are equivalent:

PROPOSITION 1. *For any action models* A *and* B*:*

$$A \leftrightarrows B \text{ implies } A \equiv B.$$

Let $s \in W_A$ *and* $t \in W_B$. *Then:*

$$(A, s) \leftrightarrows (B, t) \text{ implies } (A, s) \equiv (B, t).$$

Proof. Let $\{E_\Gamma\}_{\Gamma \in \mathrm{At}(\Sigma)}$ be an action emulation witnessing $A \leftrightarrows B$. Let M be an arbitrary model. Define a relation C on $W_{M \otimes A} \times W_{M \otimes B}$ by means of:

$$(w, s) C (v, t) :\equiv w = v \text{ and } s E_{w^*} t.$$

We show that C is a bisimulation. Suppose $(w, s) C (v, t)$.

Invariance (w,s) and (v,t) have the same valuation since $w = v$.

Zig Let $(w,s)\xrightarrow{i}(w',s')$. It follows that $w\xrightarrow{i}w'$, $s\xrightarrow{i}s'$, and $M \models_{w'} \mathsf{pre}_{s'}$. So $\mathsf{pre}_{s'} \in w'^*$. Now $w^*\xrightarrow{i}w'^*$ follows immediately from $M \models_w \widehat{w^*} \wedge \langle i\rangle \widehat{w'^*}$.

According to $sE_{w^*}t$, there must be a t' such that $t\xrightarrow{i}t'$ and $s'E_{w'^*}t'$. Thus, $w'^* \in G(s')$, and $M \models_{w'} \mathsf{pre}_{t'}$.

Therefore we have $(w',s')C'(w',t')$, as desired.

Zag Same reasoning vice versa.

We show that for each $(w,s) \in W_{M\otimes A}$ there is a $(w,t) \in W_{M\otimes A}$ with $(w,s)C(w,t)$, and vice versa. Let $(w,s) \in W_{M\otimes A}$. Then $w^* \in At(\Sigma)$, and $\mathsf{pre}_s \in w^*$. So by $A \leftrightarrows B$ there is a t with $sE_{w^*}t$. This gives $(w,s)C(w,t)$, as desired. Similarly in the other direction.

For the second part, let (M,w) be any pointed model. From the assumption $(A,s) \leftrightarrows (B,t)$ we get that there is a relation E_{w^*} in the set of emulation relations for which $sE_{w^*}t$. Therefore the relation C, defined as before, will connect (w,s) and (w,t). □

Proposition 1 shows that action emulation is a sufficient condition for action equivalence. The following proposition shows that it is also a necessary one.

PROPOSITION 2. *For any action models A and B:*

$$A \equiv B \text{ implies } A \leftrightarrows B.$$

If $s \in W_A$ and $t \in W_B$ then:

$$(A,s) \equiv (B,t) \text{ implies } (A,s) \leftrightarrows (B,t).$$

Proof. Assume $A \equiv B$. Let Σ be the set of preconditions occurring in A or B, and M_Σ be a canonical model over Σ. It follows from $A \equiv B$ that

$$M_\Sigma \otimes A \,\underline{\leftrightarrow}\, M_\Sigma \otimes B.$$

Let C witness this bisimulation. Define a set of binary relations

$$\{E_\Gamma\}_{\Gamma\in \mathrm{At}(\Sigma)}$$

by means of

$$sE_\Gamma t :\equiv (\Gamma,s)C(\Gamma,t).$$

We show that $\{E_\Gamma\}_{\Gamma\in \mathrm{At}(\Sigma)}$ is an action emulation. Suppose $sE_\Gamma t$.

Invariance It follows from $(\Gamma, s)C(\Gamma, t)$ that $M_\Sigma \models_\Gamma \mathsf{pre}_s$ and $M_\Sigma \models_\Gamma \mathsf{pre}_t$. According to Truth Lemma 1, we have $\mathsf{pre}_s \in \Gamma$ and $\mathsf{pre}_t \in \Gamma$.

Zig Suppose $s \xrightarrow{i} s'$ and $\Gamma' \in G(s')$ such that $\Gamma \xrightarrow{i} \Gamma'$. It follows that $\mathsf{pre}_{s'} \in \Gamma'$. Again by Truth Lemma 1, we have $M_\Sigma \models_{\Gamma'} \mathsf{pre}_{s'}$, so $(\Gamma', s') \in W_{M_\Sigma \otimes A}$. Therefore, we have $(\Gamma, s) \xrightarrow{i} (\Gamma', s')$. By the Zig property of C, there must be (Γ'', t') such that $(\Gamma, t) \xrightarrow{i} (\Gamma'', t')$ and $(\Gamma', s')C(\Gamma'', t')$. Since in our construction of M_Σ, the valuation of each world is different, it follows that $\Gamma' = \Gamma''$. Therefore $t \xrightarrow{i} t'$ and $s' E_{\Gamma'} t'$.

Zag Same reasoning vice versa.

It is easy to see that for each $s \in W_A$ and each $\Gamma \in At(\Sigma)$ with $\mathsf{pre}_s \in \Gamma$ there is a $t \in W_B$ with $sE_\Gamma t$, and vice versa.

For the second part of the Theorem, assume $(A, s) \equiv (B, t)$. Define the set of action emulation relations $\{E_\Gamma\}_{\Gamma \in \mathrm{At}(\Sigma)}$ as before. From Theorem 2 it follows that for all $\Gamma \in At(\Sigma)$ with $\mathsf{pre}_s \in \Gamma$ it holds that $sE_\Gamma t$. This proves $(A, s) \leftrightarrows (B, t)$. □

Combining Proposition 1 and Proposition 2, we have:

THEOREM 3. *For any action models A and B:*

$$A \equiv B \text{ iff } A \leftrightarrows B.$$

If $s \in W_A$, $t \in W_B$:

$$(A, s) \equiv (B, t) \text{ iff } (A, s) \leftrightarrows (B, t).$$

The following is a direct corollary of Theorem 2.

PROPOSITION 3. *Equivalence of action models is decidable.*

Proof. Given action models A and B, let Σ be the set of preconditions occurring in A or B, and M_Σ be a canonical model over Σ. Checking whether $M_\Sigma \otimes A \leftrightarrow M_\Sigma \otimes B$ is decidable. This gives us a decision method for action equivalence. □

Finally, we note that the union of action emulations is itself an action emulation:

PROPOSITION 4. *Given action models A and B that emulate, if E_1 and E_2 are action emulations that witness $A \leftrightarrows B$, then $E_1 \cup E_2$ is also an action emulation.*

The proof of Proposition 4 follows directly from Definition 15.

6. Action Emulation: The Propositional Case

We will now concentrate on a simpler structural relation that coincides with action equivalence for the case of propositional action models.

DEFINITION 16. *An action model is* propositional *if every precondition formula that occurs in it is a formula of classical propositional logic.*

Most of our everyday communications are like this. We exchange factual information, deciding whether to send cc's or not, we decide to keep some facts to ourselves, or only tell them to a few close friends. The epistemic pattern of *how* the information is conveyed may be incredibly complex, as when we decide to send private letters of invitation to a large group of acquaintances, but with a cc to our spouse.

To formulate a structural relation that matches action equivalence for these cases, we introduce some notation designed to highlight the connection with bisimulation.

DEFINITION 17. *If A and B are action models, and $E \subseteq W_A \times W_B$ is a binary relation, then $\overrightarrow{E} \subseteq W_A \times \mathcal{P}(W_B)$ is given by*

$$x\overrightarrow{E}Y \text{ iff } \forall y \in Y(xEy)$$

and $\overleftarrow{E} \subseteq \mathcal{P}(W_A) \times W_B$ is given by

$$X\overleftarrow{E}y \text{ iff } \forall x \in X(xEy).$$

If $\xrightarrow{i} \subseteq X \times Y$ is a binary relation, then $\xrightarrow{\bar{i}} \subseteq X \times \mathcal{P}(Y)$ is the relation given by $x\xrightarrow{\bar{i}}Y$ if $Y \subseteq \{y \mid x\xrightarrow{i}y\}$.

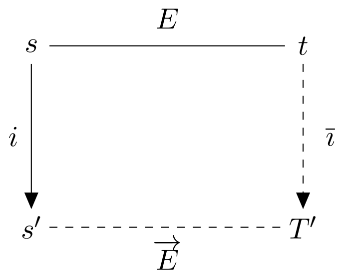
Here is a simplified definition of action emulation for the propositional case.

DEFINITION 18. (Propositional Action Emulation).
Given action models A and B, a relation $E \subseteq W_A \times W_B$ is a propositional action emulation if whenever sEt the following hold:

Invariance $\mathsf{pre}_s \wedge \mathsf{pre}_t$ *is consistent;*

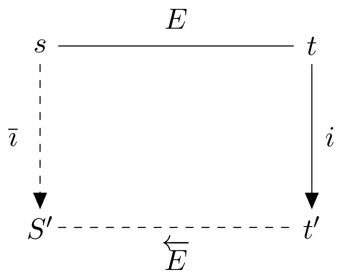
Zig *If* $s \xrightarrow{i} s'$ *then there is a non-empty set* $T' \subseteq W_B$ *with* $t \xrightarrow{\bar{\imath}} T'$ *such that* $s' \overrightarrow{E} T'$ *and* $\mathsf{pre}_{s'} \models \bigvee\{\mathsf{pre}_x \mid x \in T'\}$.

In a picture:



Zag *If* $t \xrightarrow{i} t'$ *then there is a non-empty set* $S' \subseteq W_A$ *with* $s \xrightarrow{\bar{\imath}} S'$ *such that* $S' \overleftarrow{E} t'$ *and* $\mathsf{pre}_{t'} \models \bigvee\{\mathsf{pre}_x \mid x \in S'\}$.

In a picture:



We use $A \leftrightarrows_p B$ *to indicate the existence of a propositional action emulation relation* E *such that for each* $x \in W_A$ *there is a* $Y \subseteq W_B$ *with* $x \overrightarrow{E} Y$, *and* $\mathsf{pre}_x \models \bigvee\{\mathsf{pre}_y \mid y \in Y\}$, *and vice versa.*

Given pointed action models (A, s) *and* (B, t), *a relation* $E \subseteq W_A \times W_B$ *is a pointed propositional action emulation if* E *is a propositional action emulation between* A *and* B *that connects* s *and* t, *and moreover* pre_s *and* pre_t *are logically equivalent. Notation for this:* $(A, s) \leftrightarrows_p (B, t)$.

Note that the above applies to action models of all kinds. In particular, we do not require in Definition 18 that action models have propositional preconditions. The following proposition shows that a propositional action emulation always induces an action emulation.

PROPOSITION 5. *For any action models* A *and* B:

$$A \leftrightarrows_p B \text{ implies } A \leftrightarrows B.$$

If $s \in W_A$ *and* $t \in W_B$ *then:*

$$(A, s) \leftrightarrows_p (B, t) \text{ implies } (A, s) \leftrightarrows (B, t).$$

Proof. Assume $A \leftrightarrows_p B$. Let Σ be the set of preconditions occurring in A, B, and let

$$G(x) = \{\Gamma \in At(\Sigma) \mid \mathsf{pre}_x \in \Gamma\},$$

for $x \in W_A \cup W_B$.

Suppose that F is a propositional action emulation between A and B. For $\Gamma \in At(\Sigma)$ define $E_\Gamma \subseteq W_A \times W_B$ by means of:

$$xE_\Gamma y \quad \text{iff} \quad xFy, \mathsf{pre}_x \in \Gamma, \mathsf{pre}_y \in \Gamma.$$

To prove that $\{E_\Gamma\}_{\Gamma \in \mathrm{At}(\Sigma)}$ is an action emulation we verify that the relations $\{E_\Gamma\}_{\Gamma \in \mathrm{At}(\Sigma)}$ satisfy the conditions of Definition 15. Let $sE_\Gamma t$.

Invariance The invariance property follows from the definition of relations E_Γ.

Zig Suppose that $x \xrightarrow{i} x'$, $\Gamma' \in G(x')$, and $\Gamma \xrightarrow{i} \Gamma'$. Since xFy and $x \xrightarrow{i} x'$, it follows by the Zig property of F that there is a non-empty set $Y' \subseteq W_B$ with $t \xrightarrow{\bar{i}} Y'$ such that $x' \overrightarrow{F} Y'$ and

$$\mathsf{pre}_{x'} \models \bigvee \{\mathsf{pre}_{y'} \mid y' \in Y'\}.$$

The last condition implies that there is $y' \in Y'$ with $\mathsf{pre}_{y'} \in \Gamma'$. Therefore, $x' E_{\Gamma'} y'$.

Zag The proof of Zag is analogous.

Now let $x \in W_A$ and let $\Gamma \in G(x)$. By the properties of F, there is a $Y \subseteq W_B$ with $x \overrightarrow{F} Y$ and $\mathsf{pre}_x \models \bigvee \{\mathsf{pre}_y \mid y \in Y\}$. It follows that there is some y with xFy and $\mathsf{pre}_y \in \Gamma$. Then, by definition, $xE_\Gamma y$. This shows that for every $x \in W_A$ and every $\Gamma \in G(x)$ there is a $y \in W_B$ with $xE_\Gamma y$. Similarly for the other direction.

For the proof of the second part of the Theorem, assume $(A, s) \leftrightarrows_p (B, t)$. Then pre_s and pre_t are logically equivalent. Define the emulation relations as before, and verify that for all Γ with $\mathsf{pre}_s \in \Gamma$ it holds that $sE_\Gamma t$. It follows that $(A, s) \leftrightarrows (B, t)$. □

Next, we show that an action emulation between propositional action models always induces a propositional action emulation.

PROPOSITION 6. *For all propositional action models A and B:*

$$A \leftrightarrows B \text{ implies } A \leftrightarrows_p B.$$

If $s \in W_A$ and $t \in W_B$ then:

$$(A, s) \leftrightarrows (B, t) \text{ implies } (A, s) \leftrightarrows_p (B, t).$$

Proof. Suppose that A and B are propositional action models with $A \leftrightarrows B$. Let $\{E_\Gamma\}_{\Gamma \in \mathrm{At}(\Sigma)}$ be a set of relations witnessing this. Define F by means of

$$xFy \text{ iff for some } \Gamma \in At(\Sigma): \; xE_\Gamma y.$$

We show that F is a propositional action emulation. Assume sFt, i.e., there is some $\Gamma \in At(\Sigma)$ with $sE_\Gamma t$.

Invariance The invariance property is inherited from Definition 15.

Zig Suppose that $s \xrightarrow{i} s'$. Let

$$T' = \{t' \in W_B \mid t \xrightarrow{i} t' \text{ and } \exists \Gamma' \in At(\Sigma) : s' E_{\Gamma'} t'\}.$$

Then by the Zig property of $\{E_\Gamma\}_{\Gamma \in \mathrm{At}(\Sigma)}$, T' is non-empty.

We still have to show $\mathsf{pre}_{s'} \models \bigvee\{\mathsf{pre}_{t'} \mid t' \in T'\}$. So suppose for a contradiction that $\mathsf{pre}_{s'} \wedge \bigwedge\{\neg\mathsf{pre}_{t'} \mid t' \in T'\}$ is consistent. Then there is some $\Gamma^* \in At(\Sigma)$ with

$$\Gamma^* \supseteq \{\mathsf{pre}_{s'}\} \cup \{{\sim}\mathsf{pre}_{t'} \mid t' \in T'\}.$$

By the fact that A and B are propositional, we have that $\Gamma \xrightarrow{i} \Gamma^*$. Therefore, by the properties of $\{E_\Gamma\}_{\Gamma \in \mathrm{At}(\Sigma)}$, there has to be some $y \in W_B$ with $t \xrightarrow{i} y$ and $s' E_{\Gamma^*} y$. But this means that $y \in T'$ by the definition of T', and contradiction. It follows that

$$\mathsf{pre}_{s'} \models \bigvee\{\mathsf{pre}_{t'} \mid t' \in T'\}.$$

Thus, $s' \overrightarrow{F} T'$ and $t \xrightarrow{\bar{i}} T'$. This establishes the proof of Zig for F.

Zag The proof of Zag is analogous.

For the proof of the second part of the Theorem, assume $(A, s) \leftrightarrows (B, t)$. Then pre_s and pre_t are logically equivalent, and there is a set of emulation relations such that for all $\Gamma \in At(\Sigma)$ with $\mathsf{pre}_s \in \Gamma$, $sE_\Gamma t$.

Define F as before, and verify that F is a propositional action emulation that connects s and t. □

In the case of general (not necessarily propositional) action models, action equivalence does not imply propositional action equivalence. To establish this fact, in view of Theorem 3 it is sufficient to show the following:

OBSERVATION 2. *The equivalence of two pointed action models does not imply the existence of a propositional action emulation between them.*

A counterexample is presented in Figure 6.

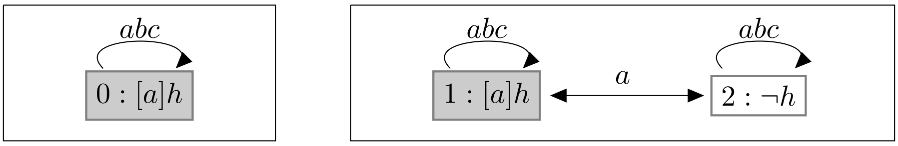


Figure 6. A pair of equivalent pointed action models that do not propositionally emulate.

To see that the pointed action models of Figure 6 are equivalent, first observe that the pointed action model on the left expresses a public announcement $[a]h$ (a public announcement "Alice knows that *heads* has turned up"). The action model on the right describes a communication where $[a]h$ gets announced, but Alice confuses this with the announcement of $\neg h$ (the public announcement "*no heads*", i.e., "*tails* has turned up"). The update result of this is the same as that of the action model on the left, for pairs $(w, 1)$ in the result of updating M with the action model on the right will have to satisfy $M, w \models [a]h$, and therefore $M, v \models h$ will hold for all v with $w \xrightarrow{a} v$. Since the precondition of 2 is $\neg h$, there will be no pairs $(v, 2)$ with $(w, 1) \xrightarrow{a} (v, 2)$ in the update result. There may be $(v, 2) \xrightarrow{a} (w, 1)$, but these will not be reachable from the distinguished world in the update result.

To see why there is no propositional action emulation between the action models in this example, observe that in Figure 6, action 2 can not emulate with the single world in the left model.

Note that the actions in Figure 6 have modal preconditions. As shown above, the update of such an action can exert an influence on the update of its successor in the resulting model. Consequently, in contrast to bisimulation, the general version of action emulation has to restrict the recursive Zig and Zag clauses to states whose preconditions formulas are consistent with the preconditions of the predecessors (see

Definition 15). In the definition of propositional action emulation such checks are omitted. However, unlike bisimulation, it requires linking points to sets in the recursive steps.

Consider however action models of the following special kind. Let Q be a finite set of propositional letters. Then a Q-valuation action model is an action model that has all its preconditions of the form:

$$\bigwedge_{q \in v} q \wedge \bigwedge_{q \in (Q \setminus v)} \neg q,$$

for some $v \subseteq Q$ (i.e., v is a Q-valuation). The proof of the last proposition of this paper is immediate.

PROPOSITION 7. *For any action models A and B a bisimulation relation is also a propositional action emulation relation. When A and B are Q-valuation action models, a propositional action emulation relation is also a bisimulation relation (in fact, the two definitions are equivalent).*

Finally, let us remind you that for propositional action models that are not of the above special kind, propositional action equivalence does not imply bisimilarity. Figure 5 above provides an example of two equivalent propositional pointed action models for which there is no pointed bisimulation. Note that the relation

$$E_1 = \{(0,2),(1,3),(1,4)\}$$

between the domains of the left and the right action models in Figure 5 is a pointed propositional action emulation.

7. Conclusion and Further Issues

In this paper we addressed the following notion of action equivalence: two action models always yield bisimilar results when they update any state model. Our aim was to capture a more direct relation between action equivalent models in terms of their preconditions and structures. First, we gave a natural extension of the definition of a bisimulation to action models, and showed that it is a sufficient condition for action equivalence but not a necessary one. Next, we gave a sufficient and necessary condition for action equivalence in terms of update on canonical models. Our Theorem 3 shows that this notion indeed provides a full characterization of action equivalence for action models with arbitrary preconditions.

Action emulation bears a close family resemblance to bisimulation, as it is also defined in terms of invariance, zig and zag conditions. This family tie with standard bisimulation generates a number of other family resemblances. E.g., as with bisimulations, the union of all action emulations connecting (A, s) and (B, t) is an action emulation, i.e., there always is a largest action emulation connecting (A, s) and (B, t). The proof of Theorem 3 (that action emulation characterizes action equivalence) relies on a canonical model construction that is well known from Henkin style completeness proofs. Van Ditmarsch and French [9] prove that for finite models, refinements (or: simulations) correspond to action models. This suggests that there might be a generalized notion of simulation that characterizes action emulation. Finding a more direct construction is future work.

What is the complexity of determining whether two action models emulate, either for the propositional case or the general case? Is it possible to define emulation-minimal action models, in the propositional, or even in the general case? If so, can something like a partition refinement algorithm for computing bisimulation-minimal models in the style of [16] be adapted to compute emulation-minimal action models? What is the complexity of this reduction? We refer to [17] for some preliminary results on expansion and contraction operations that preserve action equivalence. We leave all these questions for future work.

Acknowledgements

Preliminary results of this work are presented in [17] and [18]. We thank Johan van Benthem, Hans van Ditmarsch, Floor Sietsma, Albert Visser and Yanjing Wang for inspiring discussions, and thank five anonymous reviewers for helpful comments. We are grateful to our editor Wiebe van der Hoek for encouragement, patience, trust and flexibility. The first author is grateful to the Netherlands Institute for Advanced Studies in Wassenaar (NIAS) for providing the opportunity to work on this paper as Fellow-in-Residence.

References

1. Baltag, A. A logic for suspicious players: epistemic action and belief-updates in games. *Bulletin of Economic Research 54*, 1 (2002), 1–45.
2. Baltag, A., and Moss, L. Logics for epistemic programs. *Synthese 139*, 2 (2004), 165–224.

3. Baltag, A., Moss, L., and Solecki, S. The logic of public announcements, common knowledge, and private suspicions. In *Proceedings of TARK'98* (1998), I. Bilboa, Ed., pp. 43–56.
4. Benthem, J. v. Language, logic, and communication. In *Logic in Action*, J. van Benthem, P. Dekker, J. van Eijck, M. de Rijke, and Y. Venema, Eds. ILLC, 2001, pp. 7–25.
5. Benthem, J. v. One is a Lonely Number: on the logic of communication. In *Proceedings of Logic Colloquium '02* (Münster, Germany, 2002), Z. Chatzidakis, P. Koepke, and W. Pohlers, Eds., pp. 95–128.
6. Benthem, J. v., van Eijck, J., and Kooi, B. Logics of communication and change. *Information and Computation 204*, 11 (2006), 1620–1662.
7. Blackburn, P., de Rijke, M., and Venema, Y. *Modal Logic*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 2001.
8. Ditmarsch, H. v. *Knowledge Games*. PhD thesis, ILLC, Amsterdam, 2000.
9. Ditmarsch, H. v., and French, T. Simulation and information: Quantifying over epistemic events. In *KRAMAS* (2008), J.-J. C. Meyer and J. Broersen, Eds., vol. 5605 of *Lecture Notes in Computer Science*, Springer, pp. 51–65.
10. Ditmarsch, H. v., van der Hoek, W., and Kooi, B. *Dynamic Epistemic Logic*, vol. 337 of *Synthese Library*. Springer, 2007.
11. Fagin, R., Halpern, J., Moses, Y., and Vardi, M. *Reasoning about Knowledge*. MIT Press, 1995.
12. Fischer, M., and Ladner, R. Propositional dynamic logic of regular programs. *Journal of Computer and System Sciences 18*, 2 (1979), 194–211.
13. Gerbrandy, J. *Bisimulations on planet Kripke*. PhD thesis, ILLC, 1999.
14. Kozen, D., and Parikh, R. An elementary proof of the completeness of PDL. *Theoretical Computer Science 14* (1981), 113–118.
15. Lutz, C. Complexity and succinctness of public announcement logic. In *AAMAS '06: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems* (New York, NY, USA, 2006), ACM, pp. 137–143.
16. Paige, R., and Tarjan, R. E. Three partition refinement algorithms. *SIAM J. Comput. 16*, 6 (1987), 973–989.
17. Ruan, J. Exploring the update universe. Master's thesis, ILLC, Amsterdam, 2004.
18. Sadzik, T. Exploring the iterated update universe. Tech. rep., ILLC, Amsterdam, March 2006. PP-2006-26.
19. Segerberg, K. A completeness theorem in the modal logic of programs. In *Universal Algebra and Applications*, T. Traczyck, Ed. Polish Science Publications, 1982, pp. 36–46.