Symbolic Realisation of Epistemic Processes

Rolf Hennicker¹, Alexander Knapp², and Martin Wirsing¹

¹ Ludwig-Maximilians-Universität München, Germany
{hennicker, wirsing}@ifi.lmu.de
² Universität Augsburg, Germany
knapp@informatik.uni-augsburg.de

Abstract

Epistemic processes describe the dynamic behaviour of multi-agent systems driven by the knowledge of agents, which perform epistemic actions that may lead to knowledge updates. Executing epistemic processes directly on epistemic states, given in the traditional way by pointed Kripke structures, quickly becomes computationally expensive due to the semantic update constructions. Based on an abstraction to formulæ of interest, we introduce a symbolic epistemic state representation and a notion of representable epistemic action with efficient symbolic updates. In contrast to existing work on belief or knowledge bases, our approach can handle epistemic actions modelled by arbitrary action models. We introduce an epistemic process calculus and a propositional dynamic logic for specifying process properties that can be interpreted both on the concrete semantic and the symbolic level. We show that our abstraction technique preserves and reflects behavioural properties of epistemic processes whenever processes are started in a symbolic state that is an abstraction of a semantic epistemic state.

1 Introduction

The execution behaviour of epistemic processes depends on the current knowledge of agents about themselves, about the knowledge of other agents, and about their environment. Epistemic processes interact by executing epistemic actions, like announcements to agents, which may change an epistemic state. In [21] we have considered epistemic ensembles which are collections of epistemic processes that cooperate to achieve common goals. Differently from dynamic epistemic logic (DEL [18, 2]), our epistemic processes follow a particular protocol determined by a process expression written in a process algebraic style [13]. In [21] processes are executed on a shared epistemic state, given by a pointed Kripke structure [16]; the epistemic actions of the processes rely on the action models proposed in [3] with their effects computed by product updates of epistemic states [2].

The traditional Kripke models representing epistemic states tend to become large quickly due to the product updates that may, like in private announcements, duplicate the set of possible worlds. Belief or knowledge bases offer a syntactic approach to the compact representation of epistemic states; see [25] for an overview where it is argued that “these approaches defend the idea that the right level of abstraction for understanding and modeling cognitive processes
Symbolic Realisation of Epistemic Processes

and phenomena is the ‘belief base’ level”. Belief bases have been mainly applied to obtain symbolic versions of DEL using specialised syntactic actions like belief base expansions [26]. In this work we exploit the advantages of symbolic representations of epistemic states for the execution of epistemic processes while keeping the general expressiveness of action models but leading to succinct, symbolic representations of product updates. As an essential result we relate the traditional and the symbolic approach by an appropriate abstraction technique such that behavioural properties of epistemic processes, expressed by dynamic logic formulae, are preserved and reflected when processes are run in a traditional and in a symbolic environment. Our investigation proceeds as follows.

Symbolic epistemic states. For the representation of symbolic epistemic states we use finite sets \( \Gamma \) of epistemic formulæ which explicitly show what currently is known by the agents. Such a finite knowledge base allows for effective computations and decisions, but not all information of an epistemic state (i.e. Kripke model) \( \mathcal{K} \) can be captured; in fact, the epistemic theory of an epistemic state is, in general, not finite even if \( \mathcal{K} \) is finite [1, Sect. 6.6]. In the spirit of predicate abstraction [23], we therefore use a finite set \( \Phi \) of epistemic “focus formulæ” capturing what is of interest to be known by the agents; a symbolic epistemic state \( \Gamma \) is a subset of these focus formulæ \( \Phi \). Thus we can represent with \( n \) focus formulæ \( 2^n \) symbolic states.

Symbolic updates. The execution of an epistemic action \( u \) updates the current symbolic state \( \Gamma \subseteq \Phi \) to which \( u \) is applied. To define the symbolic update \( \Gamma \circ \Sigma(u, \varphi) \) caused by \( u \), we transfer the ideas of symbolic program execution [10] to the domain of epistemic actions: We utilise that for any epistemic action \( u \) and any epistemic formula \( \varphi \) a weakest liberal precondition \( wlp_\Sigma(u, \varphi) \) (in the classical sense of [15]) can be computed on the basis of the reduction axioms for the traditional product update [18, 4]. Then, the basic idea to compute \( \Gamma \circ \Sigma(u, \varphi) \) is to take all the focus formulæ \( \varphi \in \Phi \) such that \( wlp_\Sigma(u, \varphi) \in \Gamma \). We must, however, pay attention that weakest liberal preconditions of focus formulæ are focus formulæ again, a property called \( \Phi \)-representability, which we require for the epistemic actions in our framework. On the other hand we can relax the idea that \( wlp_\Sigma(u, \varphi) \) must literally belong to \( \Gamma \); cf. Sect. 4.2.

Process execution in traditional and symbolic environments. For the representation of epistemic processes we use a simple process language such that sequential processes model the behaviour of single agents and a parallel composition of processes the global behaviour of a multi-agent system. We provide a purely syntactic operational semantics for the execution of processes which can be instantiated in both traditional and symbolic environments. In the first case we consider concrete epistemic process configurations \((P, \mathcal{K})\) consisting of a process term \( P \) and an epistemic state \( \mathcal{K} \). In the latter case we consider symbolic epistemic process configurations \((P, \Gamma)\) where \( \Gamma \subseteq \Phi \) is a symbolic epistemic state. We say that \( \mathcal{K} \) and \( \Gamma \) are \( \Phi \)-equivalent if \( \Gamma \) consists of exactly those focus formulæ of \( \Phi \) that hold in \( \mathcal{K} \), i.e. \( \Gamma \) is a symbolic abstraction of \( \mathcal{K} \). Our main results show that whenever we start an epistemic process \( P \) in a symbolic epistemic state \( \Gamma \) which is \( \Phi \)-equivalent to \( \mathcal{K} \) the symbolic and concrete action executions mutually simulate each other and the symbolic and concrete process configurations \((P, \Gamma)\) and \((P, \mathcal{K})\) satisfy the same formulæ of our dynamic epistemic process logic.

Related Work. There are two main criteria to compare our work: (1) dynamic aspects and (2) symbolic knowledge representations. Concerning (1) we distinguish between (1.1) approaches in the tradition of DEL [18, 2] focusing on knowledge changes caused by action executions, and (1.2) approaches which consider systems of concurrently running agents whose behaviour is determined, e.g., by knowledge-based programs [19] or by certain protocols [7]. In this case system properties can be expressed by epistemic temporal logics [19, 27]. Our work belongs to the second branch. We focus, however, on a process-oriented presentation of dynamic
Symbolic Realisation of Epistemic Processes

Hennicker, Knapp, Wirsing

system behaviour following the established field of process algebraic languages [13] which, to our knowledge, has been rarely applied in the epistemic field [14, 24]. Our process semantics is not trace-based but given in terms of labelled transition systems. Thus, for the specification of behavioural properties we can use an adjusted propositional dynamic logic. Concerning (2), our motivation to deal with symbolic epistemic representations is related to the consideration of syntactic structures in [19] and belief bases in [26]. [19] focuses on programming and does not use epistemic actions. The main differences to [26] are: (i) [26] follows the DEL approach and not the process-oriented perspective, (ii) knowledge update operations in [26] occur in specialised forms while we rely on general action models, and (iii) we apply an abstraction technique for the transition from epistemic Kripke semantics to the symbolic approach preserving and reflecting dynamic process properties. Our abstraction relies on a finite set of focus formulae which makes implicit knowledge (cf. [19]) easy to derive.

Structure of the paper. We summarise some basic notions of epistemic logic and actions in Sect. 2. More details can be found, e.g., in [4, 18, 19]. In Sect. 3, we review our epistemic process language and the dynamic process logic in accordance with [21]. Our main contribution starts in Sect. 4 where we describe our symbolic approach to representing epistemic states and updates. Symbolic epistemic process configurations and symbolic dynamic process logic are described in Sect. 5, where we also provide our main results. In Sect. 6, we conclude and sketch an outlook to future work.

2 Epistemic Logic and Epistemic Actions

Epistemic logic. An epistemic signature \( \Sigma = (\Pi, A) \) consists of a set \( \Pi \) of (atomic) propositions and a finite set \( A \) of agents. In the following, when we write \( \Pi \) or \( A \), we always assume given an epistemic signature \( \Sigma = (\Pi, A) \). The set \( \mathcal{F}_\Sigma \) of epistemic formulae \( \varphi \) over \( \Sigma \) is defined by the grammar

\[
\varphi ::= p \mid \text{true} \mid \neg \varphi \mid \varphi_1 \land \varphi_2 \mid K_a \varphi \quad \text{where } p \in \Pi \text{ and } a \in A.
\]

The epistemic formula \( K_a \varphi \) is to be read as “agent \( a \) knows \( \varphi \)”. We use the usual Boolean shorthand notations like false for \( \neg \text{true} \), \( \varphi_1 \lor \varphi_2 \) for \( \neg(\neg\varphi_1 \land \neg\varphi_2) \), etc.

The notion of an epistemic state is based on the notion of an epistemic structure, also called Kripke model [4, 18] or Kripke structure [19]. An epistemic structure \( K = (W, E, L) \) over \( \Sigma = (\Pi, A) \) is given by a non-empty set of worlds \( W \), an \( A \)-family \( E = \{ E_a \subseteq W \times W \}_{a \in A} \) of epistemic accessibility relations, and a labelling \( L : W \to \wp \Pi \) which determines for each world \( w \in W \) the set of atomic propositions which hold in \( w \). We assume that the accessibility relations \( E_a \) are equivalences such that, for any agent \( a \in A \), \( (w, w') \in E_a \) models that \( a \) cannot distinguish the two worlds \( w \) and \( w' \). An epistemic state is a pointed Kripke structure \( \mathcal{K} = (K, w) \) where \( w \in W \) is considered as the actual world. The class of epistemic states over \( \Sigma \) is denoted by \( K_\Sigma \).

The satisfaction of an epistemic formula \( \varphi \in \mathcal{F}_\Sigma \) by an epistemic structure \( K = (W, E, L) \in K_\Sigma \) at a world \( w \in W \), written \( K, w \models_\Sigma \varphi \), is inductively defined by:

\[
K, w \models_\Sigma p \iff p \in L(w)
\]

\[
K, w \models_\Sigma \text{true}
\]

\[
K, w \models_\Sigma \neg \varphi \iff \text{not } K, w \models_\Sigma \varphi
\]

\[
K, w \models_\Sigma \varphi_1 \land \varphi_2 \iff K, w \models_\Sigma \varphi_1 \text{ and } K, w \models_\Sigma \varphi_2
\]

\[
K, w \models_\Sigma K_a \varphi \iff K, w' \models_\Sigma \varphi \text{ for all } w' \in W \text{ with } (w, w') \in E_a
\]

392
Hence, an agent \( a \) knows \( \varphi \) at point \( w \) if \( \varphi \) holds in all worlds \( w' \) which \( a \) cannot distinguish from \( w \). For an epistemic state \( \mathcal{R} = (K, w) \in \mathcal{K}_\Sigma \) and for \( \varphi \in \mathcal{S}_\Sigma \), we define \( \mathcal{R} \models_\Sigma \varphi \) by \( K, w \models_\Sigma \varphi \) and for \( \Gamma \subseteq \mathcal{S}_\Sigma \) we define \( \mathcal{R} \models_\Sigma \Gamma \) by \( \mathcal{R} \models_\Sigma \varphi \) for all \( \varphi \in \Gamma \). A formula \( \varphi \in \mathcal{S}_\Sigma \) is a consequence of \( \Gamma \subseteq \mathcal{S}_\Sigma \), written \( \Gamma \models_\Sigma \varphi \), if \( \mathcal{R} \models_\Sigma \varphi \) for all \( \mathcal{R} \) with \( \mathcal{R} \models_\Sigma \Gamma \). We write \( \models_\Sigma \varphi \) for \( \emptyset \models_\Sigma \varphi \), i.e., \( \varphi \) is a tautology.

**Example 1.** We consider a (strongly simplified) “victim rescuer” example from a case study in [28]. There are two agents, a victim \( V \) and a rescuer \( R \) and one atomic proposition \( h \) indicating whether the victim needs help. The epistemic signature is \( \Sigma_{\text{vr}} = (\Pi_{\text{vr}}, A_{\text{vr}}) \) with \( A_{\text{vr}} = \{V, R\} \) and \( \Pi_{\text{vr}} = \{h\} \). The following diagram represents the epistemic state \( \mathcal{R}_0 = (K_0, w_0) \) with actual world \( w_0 \) where \( h \) is true.

- \( \mathcal{R}_0 \) at \( w_0 \)
- \( \emptyset \) at \( w_1 \)
- \( V \) and \( R \)

The victim knows that \( h \) holds, i.e., \( \mathcal{R}_0 \models_\Sigma K_V h \), but the rescuer does not, since \( R \) considers also \( w_1 \) to be possible, i.e., \( \mathcal{R}_0 \models_\Sigma \neg K_R h \), and the victim knows that the rescuer does not know, i.e., \( \mathcal{R}_0 \models_\Sigma K_V \neg K_R h \).

**Epistemic actions.** Epistemic states model static aspects of knowledge. Epistemic actions are used for modelling dynamic changes of knowledge. A general proposal to represent epistemic actions in terms of so-called action models was set up in [3]. In our approach we will use the term action structure. An (epistemic) action structure \( U = (Q, F, \text{pre}) \) over \( \Sigma = (\Pi, A) \) consists of a set \( Q \) of events, an \( A \)-family \( F = (F_a \subseteq Q \times Q)_{a \in A} \) of action accessibility relations \( F_a \), and an action precondition function \( \text{pre} : Q \to \mathcal{S}_\Sigma \). We assume that these accessibility relations are equivalences such that for any agent \( a \in A \), \((q, q') \in F_a \) models that \( a \) cannot distinguish between occurrences of \( q \) and \( q' \). For \( q \in Q \), the epistemic formula \( \text{pre}(q) \) determines a condition under which \( q \) can happen. Hence, if an agent can identify an event \( q \) it can infer that \( \text{pre}(q) \) was valid when the event happened. An epistemic action \( u = (U, q) \) selects an actual event \( q \in Q \); we write \( F(u) \) for \( F \), \( \text{pre}(u) \) for \( \text{pre}(q) \), and \( u \cdot q' \) for the epistemic action \((U, q') \) with \( q' \in Q \). The class of epistemic actions over \( \Sigma \) is denoted by \( \mathcal{U}_\Sigma \).

The effect of an epistemic action on an epistemic state is defined in terms of a product update as constructed in [2]. The product update \((W, E, L) \bowtie_\Sigma (Q, F, \text{pre}) \) of an epistemic structure \( K = (W, E, L) \) and an epistemic action structure \((Q, F, \text{pre}) \) over \( \Sigma = (\Pi, A) \) yields the epistemic structure \((W', E', L') \) with

\[
W' = \{(w, q) \in W \times Q \mid K, w \models_\Sigma \text{pre}(q)\},
\]

\[
E'_a = \{(w, q) \mid (w, w') \in E_a, (q, q') \in F_a \text{ for all } a \in A, \text{ and } L'(w, q) = L(w) \text{ for all } (w, q) \in W'\}.
\]

According to the relations \( E'_a \) the uncertainty of an agent \( a \) in a world \((w, q) \in W'\) is determined by the uncertainty of \( a \) about world \( w \) and its uncertainty about the occurrence of \( q \). Note that the relations \( E'_a \) are again equivalence relations.

Let \( \mathcal{R} = (K, w) \in \mathcal{K}_\Sigma \) be an epistemic state and \( u = (U, q) \in \mathcal{U}_\Sigma \) be an epistemic action. If \( \mathcal{R} \models_\Sigma \text{pre}(q) \) then the product update \( \mathcal{R} \bowtie_\Sigma u \) of \( \mathcal{R} \) and \( u \) is defined and given by the epistemic state \((K \bowtie_\Sigma U, (w, q)) \in \mathcal{K}_\Sigma \). The semantics of an epistemic action \( u \in \mathcal{U}_\Sigma \) is the relation

\[
[u]_\Sigma = \{(\mathcal{R}, \mathcal{R} \bowtie_\Sigma u) \mid \mathcal{R} \models_\Sigma \text{pre}(u)\}.
\]
Example 2. A group announcement of a formula $\varphi \in \mathcal{F}_\Sigma$ to a group $A_* \subseteq A$ of agents is modelled by the epistemic action $(U_{\text{grp}}(A_*), \varphi), k)$ graphically represented by the following diagram:

```
A \xrightarrow{k} \varphi \xleftarrow{\text{true}} A \setminus A_*
```

The action structure $U_{\text{grp}}(A_*, \varphi)$ has two events $k$ and $n$. Event $k$ represents the announcement of $\varphi$ which should only happen if $\varphi$ holds and therefore $\text{pre}_{\text{grp}, \varphi}(k) = \varphi$. Only agents in the group $A_*$ can recognise this event. All other agents consider it possible that nothing happened which is represented by $n$. Of course, there is no proper precondition for this and therefore $\text{pre}_{\text{grp}, \varphi}(n) = \text{true}$.

As a particular case consider the victim rescuer example Ex. 1 and instantiate the group announcement to the special case of private announcement where it is announced to $R$ that $V$ knows that $h$ holds. Thus we consider the epistemic action $\text{prv}_R^k(K_V h) = (U_{\text{grp}}(\{R\}, K_V h), k)$. We apply this action to the epistemic state $\mathcal{R}_0 = (K_0, w_0)$ in Ex. 1. The product update yields the following epistemic state $\mathcal{R}_1 = (K_1, (w_0, k))$ (shown without reflexive accessibility edges):

```
(w_0, k) \xrightarrow{\{h\}} V
(w_0, n) \xrightarrow{\{h\}} R \xrightarrow{\emptyset} (w_1, n)
```

The world $(w_1, k)$ does not appear, since $(K_0, w_1) \not\models K_V h$ which is the precondition of $k$; but world $w_0$ is duplicated. It holds $(K_1, (w_0, k)) \models K_R K_V h$, but $(K_1, (w_0, k)) \models \neg K_V K_R K_V h$. If we apply the epistemic action $\text{prv}_R^n(K_V h) = (U_{\text{grp}}(\{R\}, K_V h), n)$ to $\mathcal{R}_0$ we obtain the epistemic state $(K_1, (w_0, n))$ with $(K_1, (w_0, n)) \models \neg K_R K_V h \land \neg K_V K_R K_V h$.

We also consider non-deterministic epistemic actions, similarly to [18]. They model alternatives which are not under the control of an agent but are selected by the environment. Formally, an epistemic choice action is a finite, non-empty set $\alpha \subseteq \mathcal{U}_\Sigma$ of epistemic actions. The set of epistemic choice actions over $\Sigma$ is denoted by $\mathcal{A}_\Sigma$. The semantics of $\alpha \in \mathcal{A}_\Sigma$ is given by the relation $[\alpha]_\Sigma = \bigcup_{u \in [\alpha]} [u]_\Sigma$.

Example 3. Continuing Ex. 2, an epistemic choice action for the victim-rescuer scenario is given by

```
\text{snd}_{\text{los}}^V \triangleleft \text{R}(K_V h) = \{ \text{prv}_R^k(K_V h), \text{prv}_R^n(K_V h) \}
```

which models a lossy sending of the information $K_V h$ from agent $V$ to agent $R$. Indeed the victim has no control about the success of the message transfer. After the lossy sending the rescuer may know $K_V h$ or not but in any case the victim does not know whether the message is arrived, i.e., whether $K_R K_V h$ holds.

Epistemic bisimulation [18]. For $K_1 = (W_1, E_1, L_1)$ and $K_2 = (W_2, E_2, L_2)$ over $\Sigma = (\Pi, A)$ a relation $B \subseteq W_1 \times W_2$ is a bisimulation if for all $(w_1, w_2) \in B$ it holds that (i) $L_1(w_1) = L_2(w_2)$, and (ii) for all $a \in A$, for each $(w_1, w'_1) \in E_{1,a}$ there is a $w'_2 \in W_2$ such that $(w_2, w'_2) \in E_{2,a}$ and $(w'_1, w'_2) \in B$ and, vice versa, (iii) for all $a \in A$, for each $(w_2, w'_2) \in E_{2,a}$ there is a $w'_1 \in W_1$ such
that \((w_1, w_1') \in E_{1,s}\) and \((w_1', w_2') \in B\). Two epistemic states \(\mathcal{R}_1 = (K_1, w_1)\) and \(\mathcal{R}_2 = (K_2, w_2)\) are bisimilar, written \(\mathcal{R}_1 \approx_{\Sigma} \mathcal{R}_2\), if there is a bisimulation \(B \subseteq W_1 \times W_2\) with \((w_1, w_2) \in B\). Bisimilar epistemic states satisfy the same epistemic formulæ \([18, \text{Thm. 2.15}]\) and bisimilarity is preserved by epistemic action updates \([18, \text{Prop. 6.21}]\).

**Lemma 1.** Let \(\mathcal{R}_1, \mathcal{R}_2 \in \mathcal{K}_\Sigma\) with \(\mathcal{R}_1 \approx_{\Sigma} \mathcal{R}_2\).
1. For all \(\varphi \in \mathcal{F}_\Sigma\) it holds that \(\mathcal{R}_1 \models_{\varphi} \varphi\) if, and only if, \(\mathcal{R}_2 \models_{\varphi}\).
2. For all \(u \in \mathcal{U}_\Sigma\) it holds that \(\mathcal{R}_1 \prec_{\Sigma} u\) is defined if, and only if, \(\mathcal{R}_2 \prec_{\Sigma} u\) is defined, and that, if both are defined, then \(\mathcal{R}_1 \prec_{\Sigma} u \approx_{\Sigma} \mathcal{R}_2 \prec_{\Sigma} u\).

### 3 Epistemic Processes and Dynamic Process Logic

Epistemic processes describe behaviours where the basic steps are epistemic choice actions. For process descriptions we choose a simple epistemic process language with typical process algebraic constructs; see \([1, 13]\). The set \(\mathcal{P}_\Sigma\) of epistemic processes \(P\) over \(\Sigma = (\Pi, A)\) is defined by the grammar

\[
P ::= 0 \mid \alpha.P \mid \varphi \circ P \mid P_1 + P_2 \mid P_1 \parallel P_2 \mid \mu X . P \mid X
\]

where \(0\) represents the inative process, \(\alpha.P\) prefixes \(P\) with an epistemic choice action \(\alpha \in A_\Sigma\), \(\varphi \circ P\) is a guarded process with condition \(\varphi \in \mathcal{F}_\Sigma\), \(P_1 + P_2\) denotes the non-deterministic choice between processes, \(P_1 \parallel P_2\) denotes (interleaving) parallel composition of processes, \(\mu X . P\) is a recursive process, and \(X\) is a process variable typically used in recursive process definitions.

**Example 4.** The behaviour of a victim rescuer system (see Ex. 1) can be described by the parallel epistemic process \(\text{Sys} = \text{Vict} \parallel \text{Resc}\) where

\[
\text{Vict} = \mu X . ((\neg K_V K_R h \land K_V h \circ \text{snd}_{\text{los}}^V(K_V h).X) + K_V K_R h \circ \text{skip}.0)
\]

\[
\text{Resc} = K_R h \circ \text{snd}_{\text{rel}}^{R \rightarrow V}(K_R h).0
\]

The process descriptions say that the victim repeatedly announces its need for help using a lossy sending as explained in Exs. 2 and 3.\(^1\) Lossy sending is appropriate here since the rescuer may be too far away. We assume that the rescuer is walking around and at some point it may be informed about the emergency and then sends its knowledge in a reliable way (since now the rescuer is close enough) to the victim. The process stops after the victim has performed a skip action (defined by a group announcement of true to all agents). The reliable sending is defined by the (singleton) choice action

\[
\text{snd}_{\text{rel}}^{R \rightarrow V}(K_R h) = \{\text{prev}_{R,V}(K_R h)\} \quad \text{with} \quad \text{prev}_{R,V}(K_R h) = (U_{\text{grp}}(\{R, V\}, K_R h), k)
\]

which is modelled by a group announcement of \(K_R h\) such that both agents are informed about the message.

**Epistemic process semantics.** We provide an operational semantics for epistemic processes by conditional transitions \(P \xrightarrow{\varphi; \alpha} P'\) relating a process \(P\) via a guard \(\varphi \in \mathcal{F}_\Sigma\) and a choice action \(\alpha \in A_\Sigma\) with another process \(P'\). The transitions are defined inductively by the rules in Tab. 1. Thereby successive guards are conjoined and true is used for the empty guard.

\(^{1}\)Notice that the victim announces \(K_V h\) and not simply \(h\). This follows the idea that if some agent announces something it should not only be a fact, but the agent should know the fact. This stresses that it is the agent and not the environment who announces.
Symbolic Realisation of Epistemic Processes

Hennicker, Knapp, Wirsing

\[
\begin{align*}
\alpha.P \stackrel{\text{true}}{\longrightarrow} \Sigma.P & \quad \varphi \triangleright P \stackrel{\varphi \land \alpha}{\longrightarrow} \Sigma.P' \\
\frac{P_1 \stackrel{\varphi}{\longrightarrow} \Sigma.P_1'}{P_1 || P_2 \stackrel{\varphi}{\longrightarrow} \Sigma.P_1' || P_2} & \quad \frac{P_1 + P_2 \stackrel{\varphi}{\longrightarrow} \Sigma.P_1'}{P_1 + P_2 \stackrel{\varphi}{\longrightarrow} \Sigma.P_2'} \\
\end{align*}
\]  

Table 1: Rules for epistemic processes

The final interpretation of guards and action preconditions relies on the configuration in which a process is run. An epistemic process configuration over \( \Sigma \) is a pair \((P, \mathfrak{r})\) of an epistemic process \( P \in \mathcal{P}_\Sigma \) and an epistemic state \( \mathfrak{r} \in \mathcal{K}_\Sigma \). Its (concrete) epistemic semantics is the labelled transition system generated over \((P, \mathfrak{r})\) by the following rule which embeds the operational process semantics:

\[(P, \mathfrak{r}) \stackrel{\alpha}{\longrightarrow} (P', \mathfrak{r}') \quad \text{if} \quad P \stackrel{\varphi}{\longrightarrow} \Sigma.P' \quad \Sigma.P \models \varphi \quad \text{and} \quad (\mathfrak{r}, \mathfrak{r}') \in \llbracket \alpha \rrbracket_\Sigma \]

**Example 5.** For the victim-rescuer example consider the process configuration \((\text{Sys}, \mathfrak{r}_0)\) with \(\text{Sys} = \text{Vict} || \text{Resc}\) as in Ex. 4 and \(\mathfrak{r}_0 = (K_0, 0, k)\) as in Ex. 1. The semantics of \((\text{Sys}, \mathfrak{r}_0)\) is the labelled transition system shown in Fig. 1 where we identify bisimilar epistemic states when the same action is repeatedly executed (which, in the case of lossy sending, would otherwise keep growing with their sets of worlds repeatedly duplicated). The transitions show (in black) the executed epistemic actions, like \(\text{snd}_{\text{los}}^{V \rightarrow \text{R}}(K_h)\). Since this is a choice action, see Ex. 3, we also inscribe for illustration (in grey) the actually chosen epistemic action \(\text{prv}_{\text{R}}^b(K_h)\) or \(\text{prv}_{\mathfrak{r}}^n(K_h)\).

![Transition system for the victim rescuer process configurations](image)

**Figure 1:** Transition system for the victim rescuer process configurations (up to bisimulation).

**Propositional dynamic logic for epistemic processes.** To specify behavioural properties of epistemic processes we consider compound actions which are formed by various combinations of epistemic choice actions along the rules for building actions in propositional dynamic logic [20]. The set \(C_\Sigma\) of compound epistemic actions over \(\Sigma\) is defined by the grammar

\[\sigma ::= \alpha \mid \varphi \mid \sigma_1 \cdot \sigma_2 \mid \sigma_1;\sigma_2 \mid \sigma^* \quad \text{where} \quad \alpha \in A_\Sigma \quad \text{and} \quad \varphi \in \mathcal{F}_\Sigma.\]

Besides the epistemic choice actions the compound actions include a test \(\varphi\) for a formula \(\varphi \in \mathcal{F}_\Sigma\), non-deterministic choices \(\sigma_1 + \sigma_2\), sequential compositions \(\sigma_1;\sigma_2\), and sequential loops \(\sigma^*\).
Epistemic process formulae are used to express behavioural properties of epistemic processes. They extend the formulae of epistemic logic by modalities with (compound) epistemic actions in the style of propositional dynamic logic. The set $\mathcal{F}_\Sigma$ of (dynamic) epistemic process formulae $\psi$ over $\Sigma$ is defined by

$$\psi ::= \varphi \mid \text{true} \mid \neg \psi \mid \psi_1 \land \psi_2 \mid [\sigma] \psi \quad \text{where } \varphi \in \mathcal{F}_\Sigma \text{ and } \sigma \in C_\Sigma.$$ 

The formula $[\sigma] \psi$ is to be read as “after all possible executions of the compound action $\sigma$ formula $\psi$ holds”. We use the usual abbreviations like false or $\lor$ as before, and we write $(\sigma') \psi$ for $[\neg [\neg [\sigma] \neg \psi]$; this latter dynamic modality is dual to $[\sigma]$ and to be read as “there is some execution of $\sigma$ such that $\psi$ holds afterwards”. Note that the knowledge operator $K_a$ can occur in $\varphi$ but it can not be applied to a dynamic modality, like $K_a [\sigma] \psi$. The reason is that we will interpret process formulae over process configurations and not solely over epistemic states. Thus we can express properties concerning the control flow of epistemic processes which are not related to the knowledge of agents.

**Example 6.** For a victim-rescuer process we are interested in the following properties, in which we abbreviate the compound action $\text{snd}^{\text{los}}_{\text{rel}} (K_V h) + \text{snd}^{\text{rel}}_{\text{V}} (K_R h) + \text{skip}$ by “some”:

1. “As long as the victim does not know that the rescuer knows that it needs help, the process will not stop”:

   $$(\text{some}^*) \rightarrow K_V K_R h \rightarrow (\text{some}) \text{true}$$

2. “Whenever the victim receives a lossy sending that it needs help, formally a sending of $K_V h$, it is possible that the rescuer will eventually know this”:

   $$(\text{some}^*; \text{snd}^{\text{los}}_{\text{rel}} (K_V h)) (\text{some}^*) K_R h$$

Epistemic process formulae can be interpreted over epistemic process configurations. First, we define the meaning of a compound epistemic action $\sigma \in C_\Sigma$ as a relation $[[\sigma]]_{\sigma, \Sigma}$ between epistemic process configurations. The relation $[[\sigma]]_{\sigma, \Sigma}$ is inductively defined along the structure of $\sigma$:

$$[[\alpha]]_{\sigma, \Sigma} = \{ ((P, \bar{R}), (P', \bar{R}')) \mid (P, \bar{R}) \xrightarrow{\alpha} \Sigma (P', \bar{R}') \}$$

$$[[\varphi]]_{\sigma, \Sigma} = \{ ((P, \bar{R}), (P, \bar{R})) \mid \sigma \models \Sigma \varphi \}$$

$$[[\sigma_1 + \sigma_2]]_{\sigma, \Sigma} = [[\sigma_1]]_{\sigma, \Sigma} \cup [[\sigma_2]]_{\sigma, \Sigma}$$

$$[[\sigma_1; \sigma_2]]_{\sigma, \Sigma} = [[\sigma_1]]_{\sigma, \Sigma} \cdot [[\sigma_2]]_{\sigma, \Sigma} \quad \text{(relational composition)}$$

$$[[\sigma^*]]_{\sigma, \Sigma} = ([[\sigma]]_{\sigma, \Sigma})^\star \quad \text{(reflexive-transitive closure)}$$

The satisfaction of an epistemic process formula $\psi \in \mathcal{F}_\Sigma$ by an epistemic process configuration $(P, \bar{R})$ is inductively defined along the structure of $\psi$:

$$(P, \bar{R}) \models [\sigma, \Sigma] \varphi \iff \bar{R} \models \Sigma \varphi$$

$$(P, \bar{R}) \models [\sigma, \Sigma] \text{true}$$

$$(P, \bar{R}) \models [\sigma, \Sigma] \neg \psi \iff \not (P, \bar{R}) \models [\sigma, \Sigma] \psi$$

$$(P, \bar{R}) \models [\sigma, \Sigma] \psi_1 \land \psi_2 \iff (P, \bar{R}) \models [\sigma, \Sigma] \psi_1 \text{ and } (P, \bar{R}) \models [\sigma, \Sigma] \psi_2$$

$$(P, \bar{R}) \models [\sigma, \Sigma] [\sigma] \psi \iff (P', \bar{R}') \models [\sigma, \Sigma] \psi \text{ f.a. } (P', \bar{R}') \text{ s.t. } ((P, \bar{R}), (P', \bar{R}')) \in [[\sigma]]_{\sigma, \Sigma}$$

Note that the satisfaction relation is well-defined for formulae $\varphi \in \mathcal{F}_\Sigma$; for which not only the first case above can be applied. For instance, consider a formula $\neg \varphi \in \mathcal{F}_\Sigma$. Then, according to the first case $(P, \bar{R}) \models [\sigma, \Sigma] \neg \varphi$ holds if $\bar{R} \models \Sigma \neg \varphi$. On the other hand, according to the third case above, $(P, \bar{R}) \models [\sigma, \Sigma] \neg \varphi$ holds if not $(P, \bar{R}) \models [\sigma, \Sigma] \varphi$ which in turn holds if not $\bar{R} \models \Sigma \varphi$.

**Example 7.** The configuration $(Sys, \bar{R}_0)$ in Ex. 5 satisfies the two epistemic process formulae in Ex. 6.
Symbolic Epistemic Logic and Updates

The epistemic process framework developed so far relies on the concrete, classical representation of epistemic states by Kripke structures and their evolution by product updates. Both can soon become highly complex such that it can get difficult and expensive (i) to check guards and action preconditions during process execution and (ii) to prove epistemic process properties. Therefore we want to investigate novel more efficient ways for representing and updating epistemic states. To do so we borrow ideas from the field of predicate abstraction [12, 23] aiming at a compact and symbolic representation of knowledge states (Sect. 4.1) and at symbolic executions (Sect. 4.2). Instead of predicates we will use epistemic formulæ. They provide a more intuitive understanding of knowledge than complex epistemic structures.

4.1 Symbolic Epistemic States

The basic idea is that a symbolic representation of knowledge in a certain state is given as a “knowledge base” $\Gamma \subseteq \mathcal{F}_\Sigma$ of epistemic formulæ. In order to allow for effective computations and decisions, we propose to use a finite set $\Phi \subseteq \mathcal{F}_\Sigma$ of “formulæ of interest”, called focus formulæ, such that the knowledge bases $\Gamma$ are subsets of $\Phi$: A membership test reveals whether some fact of $\Phi$ is known in $\Gamma$. A knowledge base abstracts from a concrete epistemic state if it consists of exactly those $\varphi \in \Phi$ that hold in the state. Focus formulæ can be chosen on the basis of the guards occurring in process terms, the preconditions occurring in action models and the involved dynamic logic formulæ used in the formulation of process properties. Another possibility would be to choose the focus formulæ in accordance with the depth of agents’ attitudes [22]. Their principle purpose is to serve for abstraction and not as a feature for modelling particular aspects like attentions [5].

Formally, a symbolic epistemic signature is a pair $(\Sigma, \Phi)$ where $\Sigma = (\Pi, A)$ is an epistemic signature and $\Phi \subseteq \mathcal{F}_\Sigma$ is a finite set of focus formulæ. A symbolic epistemic state over $(\Sigma, \Phi)$ is a subset $\Gamma \subseteq \Phi$ which is true-closed, i.e., if true $\in \Phi$, then true $\in \Gamma$. (If true $\not\in \Phi$, but true $\not\in \Gamma$ this would mean that we consider true not to hold.) The set of symbolic epistemic states over $(\Sigma, \Phi)$ is denoted by $\mathcal{S}^{\Phi}_\Sigma$. The $\Phi$-abstraction of a (concrete) epistemic state $\mathcal{R} \in \mathcal{K}_\Sigma$ is given by the symbolic epistemic state

$$\text{abs}^\Phi_{\Sigma}(\mathcal{R}) = \{ \varphi \in \Phi \mid \mathcal{R} \models_{\Sigma} \varphi \}$$

which is the $\Phi$-theory of $\mathcal{R}$. We call, by abuse of terminology, an epistemic state $\mathcal{R} \in \mathcal{K}_\Sigma$ and a symbolic epistemic state $\Gamma \in \mathcal{S}^{\Phi}_\Sigma$ $\Phi$-equivalent, written $\mathcal{R} \equiv^\Phi_{\Sigma} \Gamma$, if $\Gamma = \text{abs}^\Phi_{\Sigma}(\mathcal{R})$. It is easy to show that the following holds:

**Lemma 2.** Let $\mathcal{R} \in \mathcal{K}_\Sigma$ and $\Gamma \in \mathcal{S}^{\Phi}_\Sigma$. $\mathcal{R} \equiv^\Phi_{\Sigma} \Gamma$ if, and only if, for all $\varphi \in \Phi$: $\mathcal{R} \models_{\Sigma} \varphi \iff \varphi \in \Gamma$.

**Proof.** “$\Rightarrow$”: Let $\mathcal{R} \equiv^\Phi_{\Sigma} \Gamma$ hold, i.e., $\Gamma = \text{abs}^\Phi_{\Sigma}(\mathcal{R})$. Then $\mathcal{R} \models_{\Sigma} \varphi$ iff (by definition) $\varphi \in \text{abs}^\Phi_{\Sigma}(\mathcal{R})$ iff (by assumption) $\varphi \in \Gamma$.

“$\Leftarrow$”: Let $\mathcal{R} \models_{\Sigma} \varphi \iff \varphi \in \Gamma$ hold for all $\varphi \in \Phi$ and let $\varphi \in \Phi$. Then $\varphi \in \Gamma$ iff (by assumption) $\mathcal{R} \models_{\Sigma} \varphi$ iff (by definition) $\varphi \in \text{abs}^\Phi_{\Sigma}(\mathcal{R})$. $\square$

**Example 8.** For the victim-rescuer processes of Ex. 4 we may consider $\{\text{true}, K_R \, h, K_V \, h, K_V \, K_R \, h, \neg K_V \, (K_R \, h \land K_V \, h)\}$ as the focus formulæ since these formulæ occur as guards in our processes or as preconditions in our actions. The initial epistemic state $\mathcal{R}_0$ of Ex. 5 is abstracted by these focus formulæ into $\{\text{true}, K_V \, h, \neg K_V \, K_R \, h \land K_V \, h\}$. 

398
The symbol $\Phi$ not $\iff$ between symbolic states $\Gamma \in S$ on $\Phi$

Let Proposition 1.

To Prop. 1(1), $\Gamma$ is closed under consequence w.r.t. focus formulæ.

| $\Phi$ semantic consequence $\Gamma$ vr formulæ (the latter symbolically) of the Boolean closure of $\Phi$ scenario described in Ex. 8 to $\Phi$ vr using the Boolean closure, we can reduce the focus formulæ for the victim-rescuer Example 9.

$\phi$ $\in$ hypothesis) $\Gamma$ $|$ $\phi$

Case $\phi$

Proof.

Lemma 3. Let $\tilde{\mathcal{K}} \in k$ and $\Gamma \in S^\Phi$; $\mathcal{R} \equiv ^\Phi \Gamma$ if, and only if, for all $\phi \in \text{bcl}(\Phi)$: $\mathcal{R} \equiv ^\Phi \Gamma$.

Proof. $\Rightarrow$: Assume $\mathcal{R} \equiv ^\Phi \Gamma$. The proof is performed by structural induction on the form of $\phi$.

Case $\phi = \varphi \in \Phi$: Then $\tilde{\mathcal{K}} \equiv ^\Phi \Gamma$ (since $\mathcal{R} \equiv ^\Phi \Gamma$) $\varphi \in \Gamma$ iff $\Gamma \equiv ^\Phi \varphi$.

Case $\phi = \varphi \in \Phi$: Then $\tilde{\mathcal{K}} \equiv ^\Phi \Gamma$ and $\Gamma \equiv ^\Phi \varphi$ hold.

Case $\phi = \varphi \in \Phi$: Then $\tilde{\mathcal{K}} \equiv ^\Phi \Gamma$ and $\Gamma \equiv ^\Phi \varphi$ (by induction hypothesis) not $\Gamma \equiv ^\Phi \varphi$ (since $\varphi \not\in \Phi$) $\Gamma \equiv ^\Phi \varphi$.

Case $\phi = \varphi \in \Phi$: Then $\tilde{\mathcal{K}} \equiv ^\Phi \Gamma$ and $\Gamma \equiv ^\Phi \varphi$ (by induction hypothesis) $\Gamma \equiv ^\Phi \varphi$ (since $\varphi \not\in \Phi$) $\Gamma \equiv ^\Phi \varphi$.

“$\Leftarrow$”:: Since $\Phi \subseteq \text{bcl}(\Phi)$, the assumption implies $\mathcal{R} \equiv ^\Phi \Gamma \equiv ^\Phi \Gamma$ for all $\phi \in \Phi$ which means $\varphi \in \Gamma$. Hence, $\mathcal{R} \equiv ^\Phi \Gamma$ by Lem. 2.

Example 9. Using the Boolean closure, we can reduce the focus formulæ for the victim-rescuer scenario described in Ex. 8 to $\Phi_{vr} = \{K_R \cdot h, K_V \cdot h, K_V K_R \cdot h\}$ which discards true and a negation. The $\Phi_{vr}$-abstraction of $\mathcal{R}_0$ is $\Gamma_0 = \{K_V \cdot h\}$ and, according to Lem. 3, $\mathcal{R}_0$ and $\Gamma_0$ satisfy the same formulæ (the latter symbolically) of the Boolean closure of $\Phi_{vr}$.

The following proposition illustrates the relation between symbolic satisfaction $\Gamma \equiv ^\Phi \phi$ and semantic consequence $\Gamma \equiv ^\Phi \phi$, cf. Sect. 2, if $\Gamma$ is an abstraction of some $\mathcal{R}$. In particular, due to Prop. 1(1), $\Gamma$ is closed under consequence w.r.t. focus formulæ.

Proposition 1. Let $\mathcal{R} \equiv ^\Phi \Gamma$. 

399
1. For each \( \varphi \in \Phi \), \( \Gamma \models_\Sigma \varphi \Rightarrow \varphi \in \Gamma \);

2. For each \( \phi \in \text{bcl}(\Phi) \), \( \Gamma \models_\Sigma \phi \Rightarrow \Gamma \models_{\Sigma}^\Phi \phi \), but the converse does not hold in general.

Proof. (1) \( \Phi \equiv_{\Sigma} \Gamma \), i.e., \( \Gamma = \text{abs}_{\Phi}(\Phi) \), implies \( \Phi \models_{\Sigma} \Gamma \). Hence, whenever \( \varphi \in \Sigma_{\Phi} \) and \( \Gamma \models_{\Sigma} \varphi \) we have \( \Phi \models_{\Sigma} \varphi \). Now let \( \varphi \in \Phi \) and \( \Gamma \models_{\Sigma} \varphi \). Then, \( \Phi \models_{\Sigma} \varphi \) holds which implies, by Lem. 2, \( \varphi \in \Gamma \).

(2) \( \Phi \equiv_{\Sigma} \Gamma \) implies \( \Phi \models_{\Sigma} \Gamma \). Hence, \( \Gamma \models_{\Sigma} \phi \) implies \( \Phi \models_{\Sigma} \phi \). Then, by Lem. 3, \( \Gamma \models_{\Sigma}^\Phi \phi \). — For the converse, consider \( \Phi = \{ \varphi \} \) and \( \Gamma = \emptyset \). Then \( \neg \varphi \in \text{bcl}(\Phi) \) and \( \Gamma \models_{\Sigma}^\Phi \neg \varphi \) since \( \varphi \notin \Gamma \). But \( \Gamma \models_{\Sigma} \neg \varphi \).

The last remark shows that our abstraction follows a “closed world assumption”: If a focus formula \( \varphi \in \Phi \) is not an element of a symbolic state, then \( \neg \varphi \) is true in this state w.r.t. \( \models_{\Sigma}^\Phi \).

4.2 Symbolic Epistemic Updates

Let us now turn to the question how to define an update for a symbolic epistemic state \( \Gamma \in \Sigma_{\Phi} \) when an epistemic action \( u \) is applied. Using the traditional approach of symbolic execution [23] would mean to compute a strongest postcondition of an epistemic action \( u \in \mathcal{U}_\Sigma \) when executed in an epistemic state satisfying \( \Gamma \). We prefer, however, to work instead with weakest liberal preconditions [15] since there exists already an algorithm for computing them for epistemic actions as described below.

Let \( u \in \mathcal{U}_\Sigma \) be an epistemic action and \( \varphi \in \Sigma_{\Phi} \). A formula \( \rho \in \Sigma_{\Phi} \) is a weakest liberal precondition of \( u \) for \( \varphi \) if the following holds:

\[
\text{for all } \Phi \in \mathcal{K}_{\Sigma}: \Phi \models_{\Sigma} \rho \iff (\Phi \models_{\Sigma} \text{pre}(u) \Rightarrow \Phi \models_{\Sigma} u \models_{\Sigma} \varphi) .
\]

(wlp)

The set of the weakest liberal precondition formulæ of \( u \) for \( \varphi \) is denoted by \( \text{wlp}_{\Sigma}(u, \varphi) \).

Obviously, if \( \rho, \rho' \in \text{wlp}_{\Sigma}(u, \varphi) \), then \( \models_{\Sigma} \rho \leftrightarrow \rho' \). There is indeed, for any \( u \in \mathcal{U}_\Sigma \) and any \( \varphi \in \Sigma_{\Phi} \), a formula \( wlp_{\Sigma}(u, \varphi) \in \text{wlp}_{\Sigma}(u, \varphi) \) that can be recursively computed by the function \( wlp_{\Sigma}: \mathcal{U}_\Sigma \times \Sigma_{\Phi} \rightarrow \Sigma_{\Phi} \) defined in accordance with the reduction rules originally stated in the context of dynamic epistemic logic (DEL) in [18, pp. 162sqq.] and [4, p. 37]:

\[
\begin{align*}
\text{wlp}_{\Sigma}(u, p) &= \text{pre}(u) \rightarrow p \\
\text{wlp}_{\Sigma}(u, \text{true}) &= \text{true} \\
\text{wlp}_{\Sigma}(u, \neg \varphi) &= \text{pre}(u) \rightarrow \neg \text{wlp}_{\Sigma}(u, \varphi) \\
\text{wlp}_{\Sigma}(u, \varphi_1 \land \varphi_2) &= \text{wlp}_{\Sigma}(u, \varphi_1) \land \text{wlp}_{\Sigma}(u, \varphi_2) \\
\text{wlp}_{\Sigma}(u, K_a \varphi) &= \text{pre}(u) \rightarrow \bigwedge_{q \in F(u) \cup a} K_a \text{wlp}_{\Sigma}(u \cdot q, \varphi)
\end{align*}
\]

In DEL, the reduction rules (see also [3, 2, 4]) handle formulæ of the form \([u] \varphi\) instead of our function application \( wlp_{\Sigma}(u, \varphi) \) and the cases for defining \( wlp_{\Sigma} \) above correspond to tautologies in DEL. For all epistemic states \( \Phi \in \mathcal{K}_{\Sigma} \), the DEL-validity of \([u] \varphi\) in \( \Phi \) means that either \( \Phi \models_{\Sigma} \text{pre}(u) \) does not hold or \( \Phi \npre_{\Sigma} u \models_{\Sigma} \varphi \).

We call \( u \in \mathcal{U}_\Sigma \) \( \Phi \)-representable for a set of focus formulæ \( \Phi \) if, on the one hand, its precondition is equivalent to some epistemic formula in the Boolean closure of \( \Phi \); and if, on the other hand, for each \( \varphi \in \Phi \), the weakest liberal precondition formula \( wlp_{\Sigma}(u, \varphi) \) is also equivalent to some formula in the Boolean closure of \( \Phi \), but now \( \text{pre}(u) \) can be assumed. More formally,

**Definition 1** (\( \Phi \)-representable). Let \( (\Sigma, \Phi) \) be a symbolic epistemic signature. An epistemic action \( u \in \mathcal{U}_\Sigma \) is \( \Phi \)-representable if

1. \( \text{Pre}(u) \cap \text{bcl}(\Phi) \neq \emptyset \), where \( \text{Pre}(u) = \{ \rho \in \Sigma_{\Phi} \mid \models_{\Sigma} \text{pre}(u) \leftrightarrow \rho \} \); and
2. for all \( \varphi \in \Phi \) it holds that \( (Wlp_{\Sigma}(u, \varphi)/pre(u)) \cap bcl(\Phi) \neq \emptyset \), where \( Wlp_{\Sigma}(u, \varphi)/pre(u) = \{ \rho \in \mathcal{F}_\Sigma ||_\Sigma \ pre(u) \rightarrow (wlp_{\Sigma}(u, \varphi) \leftrightarrow \rho) \} \).

The class of such epistemic actions is denoted by \( \mathcal{U}_h^\Phi \). An epistemic choice action \( \alpha \in \mathcal{A} \) is \( \Phi \)-

representable if all \( \varphi \in \Phi \) are \( \Phi \)-representable. The class of such choice actions is denoted by \( \mathcal{A}_h^\Phi \).

In particular, (2) is fulfilled for each \( \varphi \in \Phi \) for which \( wlp_{\Sigma}(u, \varphi) \) is already in \( bcl(\Phi) \). This suggests the following strategy to check whether \( u \) is \( \Phi \)-representable: First test whether some possibly simplified but equivalent version of \( wlp_{\Sigma}(u, \varphi) \) is in \( bcl(\Phi) \). If this is not the case, the obtained formulæ may be further simplified by assuming \( pre(u) \); the result can again be tested for membership in \( bcl(\Phi) \). Membership in \( bcl(\Phi) \) can be tested by parsing a formulæ w.r.t. the grammar of Boolean closure. For all computations involving \( \Phi \)-representable actions \( u \) it is useful to fix a canonical representative in \( Pre(u) \cap bcl(\Phi) \), denoted \( \rho_u^\Phi \), and a canonical representative in \( (Wlp_{\Sigma}(u, \varphi)/pre(u)) \cap bcl(\Phi) \) for each \( \varphi \in \Phi \), denoted by \( \rho_{u, \varphi}^\Phi \).

We have implemented a small prototype for computing epistemic weakest liberal preconditions and inspecting representability\(^2\) in the rewriting logic tool Maude [11].

**Example 10.** For the victim-rescuer scenario of Ex. 4 consider the epistemic actions \( \mathcal{U}_{vr} = \{ pre_R^h(K_V h), pre_R^h(K_V h), pre_{R,V}^h(K_R h) \} \) that occur in the agent actions of Vict \( || \) Resc and the focus formulæ \( \Phi_{vr} = \{ K_R h, K_V h, K_V K_R h \} \) considered in Ex. 9. Then all \( u \in \mathcal{U}_{vr} \) satisfy \( pre(u) \in bcl(\Phi_{vr}) \) and are \( \Phi_{vr} \)-representable, as for \( \models_{vr} pre(u) \rightarrow (wlp_{\Sigma,u}(u, \varphi) \leftrightarrow \rho) \) we obtain as possible representatives \( \rho \):  

<table>
<thead>
<tr>
<th>\varphi</th>
<th>u</th>
<th>pre_R^h(K_V h)</th>
<th>pre_R^h(K_V h)</th>
<th>pre_{R,V}^h(K_R h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>K_R h</td>
<td>K_R h</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>K_V h</td>
<td>K_V h</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>K_V K_R h</td>
<td>K_V K_R h</td>
<td>K_V h \rightarrow K_V K_R h</td>
<td>K_R h</td>
<td></td>
</tr>
</tbody>
</table>

The last two entries can be further simplified by relativising to the contexts \( pre(pre_R^h(K_V h)) = K_V h \) and \( pre(pre_{R,V}^h(K_R h)) = K_R h \) such that for \( \varphi = K_V K_R h \) the formulæ \( K_V K_R h \) can be chosen as canonical representative for \( u = pre_R^h(K_V h) \) and true for \( u = pre_{R,V}^h(K_R h) \).

For checking \( \Phi \)-representability small sets of focus formulæ are preferable. We first show that representability is preserved when considering Boolean connectives:

**Lemma 4.** Let \( u \in \mathcal{U}_h \).

1. \( \models_{\Sigma} pre(u) \rightarrow (wlp_{\Sigma}(u, \text{true}) \leftrightarrow \text{true}) \).
2. Let \( \varphi, \rho \in \mathcal{F}_\Sigma \) with \( \models_{\Sigma} pre(u) \rightarrow (wlp_{\Sigma}(u, \varphi) \leftrightarrow \rho) \). Then \( \models_{\Sigma} pre(u) \rightarrow (wlp_{\Sigma}(u, \neg \varphi) \leftrightarrow \neg \rho) \).
3. Let \( \varphi_i, \rho_i \in \mathcal{F}_\Sigma \) with \( \models_{\Sigma} pre(u) \rightarrow (wlp_{\Sigma}(u, \varphi_i) \leftrightarrow \rho_i) \) for \( i \in \{ 1, 2 \} \). Then \( \models_{\Sigma} pre(u) \rightarrow (wlp_{\Sigma}(u, \varphi_1 \land \varphi_2) \leftrightarrow (\rho_1 \land \rho_2)) \).

**Proof.** (1) By reduction it holds that \( wlp_{\Sigma}(u, \text{true}) = \text{true} \).

(2) By reduction it holds that \( wlp_{\Sigma}(u, \neg \varphi) = pre(u) \rightarrow \neg wlp_{\Sigma}(u, \varphi), \) i.e., \( \models_{\Sigma} pre(u) \rightarrow (wlp_{\Sigma}(u, \neg \varphi) \leftrightarrow \neg wlp_{\Sigma}(u, \varphi)) \), which yields \( \models_{\Sigma} pre(u) \rightarrow (wlp_{\Sigma}(u, \neg \varphi) \leftrightarrow \neg \rho) \).

(3) By reduction it holds that \( wlp_{\Sigma}(u, \varphi_1 \land \varphi_2) = wlp_{\Sigma}(u, \varphi_1) \land wlp_{\Sigma}(u, \varphi_2) \), i.e., \( \models_{\Sigma} pre(u) \rightarrow (wlp_{\Sigma}(u, \varphi_1 \land \varphi_2) \leftrightarrow wlp_{\Sigma}(u, \varphi_1) \land wlp_{\Sigma}(u, \varphi_2)) \), which yields \( \models_{\Sigma} pre(u) \rightarrow (wlp_{\Sigma}(u, \varphi_1 \land \varphi_2) \leftrightarrow (\rho_1 \land \rho_2)) \). \( \Box \)

\(^2\)Available at https://github.com/AlexanderKnapp/epistemic.git.
Thus representability only depends on the Boolean closure of a set of focus formulas:

**Lemma 5.** Let $\Phi, \Phi' \subseteq \mathcal{F}_\Sigma$ be sets of formulae such that $\text{bcl}(\Phi) = \text{bcl}(\Phi')$. Let $u \in \mathcal{U}_\Sigma$ be an epistemic action. $u$ is $\Phi'$-representable iff $u$ is $\Phi'$-representable.

**Proof.** Assume that $u$ is $\Phi$-representable. Then, for all $\varphi \in \Phi$, $\text{Wlp}_{\Sigma}(u, \varphi)/\text{pre}(u) \cap \text{bcl}(\Phi) \neq \emptyset$. As a consequence of Lem. 4, for all $\varphi \in \text{bcl}(\Phi')$, $\text{Wlp}_{\Sigma}(u, \varphi)/\text{pre}(u) \cap \text{bcl}(\Phi') \neq \emptyset$. Then, since $\Phi' \subseteq \text{bcl}(\Phi') = \text{bcl}(\Phi)$, for all $\varphi \in \Phi'$, $\text{Wlp}_{\Sigma}(u, \varphi)/\text{pre}(u) \cap \text{bcl}(\Phi') \neq \emptyset$, i.e., $u$ is $\Phi'$-representable. The converse direction is analogous. □

**Corollary 1.** Let $\Phi \subseteq \Phi' \subseteq \text{bcl}(\Phi)$ and $u \in \mathcal{U}_\Sigma$. If $u$ is $\Phi'$-representable, then $u$ is $\Phi$-representable.

**Proof.** The fact follows from Lem. 5 since $\Phi \subseteq \Phi'$ implies $\text{bcl}(\Phi) \subseteq \text{bcl}(\Phi')$ and, conversely, $\text{bcl}(\Phi') \subseteq \text{bcl}(\Phi') = \text{bcl}(\Phi)$. □

To define the update of a symbolic state $\Gamma$ by a $\Phi$-representable action $u$ the idea is to consider all formulae $\varphi \in \Phi$ having a weakest liberal precondition formula $\rho \in \text{bcl}(\Phi)$ such that $\Gamma \models_\Sigma \rho$.

**Definition 2** (Symbolic epistemic update). Let $(\Sigma, \Phi)$ be a symbolic epistemic signature. The symbolic epistemic update $\Gamma \lessdot^\Phi u$ of a symbolic epistemic state $\Gamma \in \mathcal{S}_\Sigma^\Phi$ by an epistemic action $u \in \mathcal{U}_\Sigma$ is defined if there is a $\rho \in \text{Pre}(u) \cap \text{bcl}(\Phi)$ with $\Gamma \models_\Sigma \rho$ and then

$$\Gamma \lessdot^\Phi u = \{ \varphi \in \Phi \mid \text{exists } \rho \in \text{bcl}(\Phi) \cap (\text{Wlp}_{\Sigma}(u, \varphi)/\text{pre}(u)) \text{ such that } \Gamma \models_\Sigma \rho \} .$$

Definedness of a symbolic update can be checked along the same lines as checking representability. The computation of $\Gamma \lessdot^\Phi u$ collects all $\varphi \in \Phi$ such that $\Gamma \models_\Sigma \rho_{u, \varphi}$ which is linear in the number of focus formulae. Note that the particular choice of the canonical representative $\rho_{u, \varphi}^\Phi$ is irrelevant by Lem. 3, provided that $\Gamma$ is the $\Phi$-abstraction $\text{abs}\_\Sigma^\Phi(\mathcal{R})$ of some $\mathcal{R} \in \mathcal{K}_\Sigma$. As a matter of fact, the symbolic update can only lead to a symbolic state of size smaller or equal to the size of $\Phi$ while updates of the traditional Kripke models can continuously grow. This shows the succinctness of the symbolic approach compared to the classical one.

$\Phi$-equivalence is preserved by updates with $\Phi$-representable actions:

**Lemma 6.** Let $\mathcal{R} \equiv_\Sigma^\Phi \Gamma$ and $u \in \mathcal{U}_\Sigma^\Phi$. Then
1. $\mathcal{R} \lessdot_\Sigma^\Phi u$ is defined if, and only if, $\Gamma \lessdot^\Phi u$ is defined.
2. If $\mathcal{R} \lessdot_\Sigma^\Phi u$ and $\Gamma \lessdot^\Phi u$ both are defined, then $\mathcal{R} \lessdot_\Sigma^\Phi u \equiv_\Sigma^\Phi \Gamma \lessdot^\Phi u$.

**Proof.** (1) $\mathcal{R} \lessdot_\Sigma^\Phi u$ is defined iff $\mathcal{R} \models_\Sigma \text{pre}(u)$ iff (since $u$ is $\Phi$-representable) there is a $\rho \in \text{Pre}(u) \cap \text{bcl}(\Phi)$ with $\Gamma \models_\Sigma \rho$ iff (by Lem. 3) $\Gamma \models_\Sigma^\Phi \rho$ iff (by Def. 2) $\varphi \in \Gamma \lessdot^\Phi u$.

(2) By Lem. 2, it suffices to show that for all $\varphi \in \Phi$ it holds that $\mathcal{R} \lessdot_\Sigma^\Phi u \models_\Sigma \varphi$ iff $\varphi \in \Gamma \lessdot^\Phi u$.

For “$\Rightarrow$”, let $\mathcal{R} \lessdot_\Sigma^\Phi u \models_\Sigma \varphi$ hold for $\varphi \in \Phi$. By the definition of the weakest liberal precondition, $\mathcal{R} \models_\Sigma \text{wlp}_{\Sigma}(u, \varphi)$. From the $\Phi$-representability of $u$ it follows that there exists a $\rho \in \text{bcl}(\Phi)$ with $\models_\Sigma \text{pre}(u) \rightarrow (\text{wlp}_{\Sigma}(u, \varphi) \leftrightarrow \rho)$. In particular, $\mathcal{R} \models_\Sigma \text{pre}(u) \rightarrow (\text{wlp}_{\Sigma}(u, \varphi) \leftrightarrow \rho)$. Moreover, since $\mathcal{R} \models_\Sigma \text{pre}(u)$ and $\mathcal{R} \models_\Sigma \text{wlp}_{\Sigma}(u, \varphi)$, we obtain $\mathcal{R} \models_\Sigma \rho$. Therefore, by Lem. 3, $\Gamma \models_\Sigma^\Phi \rho$, and hence $\varphi \in \Gamma \lessdot^\Phi u$ by Def. 2.

For “$\Leftarrow$”, let $\varphi \in \Gamma \lessdot^\Phi u$ hold for $\varphi \in \Phi$, i.e., there is a $\rho \in \text{bcl}(\Phi)$ with $\models_\Sigma \text{pre}(u) \rightarrow (\text{wlp}_{\Sigma}(u, \varphi) \leftrightarrow \rho)$ and $\Gamma \models_\Sigma^\Phi \rho$. By Lem. 3, $\mathcal{R} \models_\Sigma \rho$ and hence $\mathcal{R} \models_\Sigma \text{wlp}_{\Sigma}(u, \varphi)$, since $\mathcal{R} \models_\Sigma \text{pre}(u) \rightarrow (\text{wlp}_{\Sigma}(u, \varphi) \leftrightarrow \rho)$ and $\mathcal{R} \models_\Sigma \text{pre}(u)$. By the definition of the weakest liberal precondition it follows that $\mathcal{R} \lessdot_\Sigma^\Phi u \models_\Sigma \varphi$. □

402
The symbolic semantics $\llbracket u \rrbracket^\Phi_\Sigma$ of any $u \in \mathcal{U}_\Sigma^\Phi$ is given by the relation
$$\llbracket u \rrbracket^\Phi_\Sigma = \{ (\Gamma, \Gamma \llbracket u \rrbracket^\Phi_\Sigma) \mid \text{ex. } \rho \in \text{Pre}(u) \cap \text{bcl}(\Phi) \text{ s.t. } \Gamma \models^\Phi_\Sigma \rho \}$$
on symbolic epistemic states. The symbolic semantics $\llbracket \alpha \rrbracket^\Phi_\Sigma$ of a choice action $\alpha \in \mathcal{A}_\Sigma^\Phi$ is $\llbracket \alpha \rrbracket^\Phi_\Sigma = \bigcup_{u \in \alpha} \llbracket u \rrbracket^\Phi_\Sigma$. The preservation of $\Phi$-equivalence extends to $\Phi$-representable choice actions:

**Lemma 7.** Let $\mathcal{R} \equiv^\phi_\Sigma \Gamma$ and $\alpha \in \mathcal{A}_\Sigma^\Phi$. Then the following holds:
1. If $(\mathcal{R}, \mathcal{R}') \in \llbracket \alpha \rrbracket^\Phi_\Sigma$, then there is a $\Gamma' \in \mathcal{S}_\Sigma^\Phi$ such that $(\Gamma, \Gamma') \in \llbracket \alpha \rrbracket^\Phi_\Sigma$ and $\mathcal{R}' \equiv^\phi_\Sigma \Gamma'$.
2. If $(\Gamma, \Gamma') \in \llbracket \alpha \rrbracket^\Phi_\Sigma$, then there is a $\mathcal{R}' \in \mathcal{K}_\Sigma$ such that $(\mathcal{R}, \mathcal{R}') \in \llbracket \alpha \rrbracket^\Phi_\Sigma$ and $\mathcal{R}' \equiv^\phi_\Sigma \Gamma'$.

**Proof.** (1) Let $(\mathcal{R}, \mathcal{R}') \in \llbracket \alpha \rrbracket^\Phi_\Sigma$. Then there is a $u \in \alpha$ such that $(\mathcal{R}, \mathcal{R}') \in \llbracket u \rrbracket^\Phi_\Sigma$. Hence, $\mathcal{R}' = \mathcal{R} \llbracket u \rrbracket \mathcal{R}$ is defined. Since $\mathcal{R} \equiv^\phi_\Sigma \Gamma$, also $\Gamma' = \Gamma \llbracket u \rrbracket \mathcal{R} \equiv^\phi_\Sigma \Gamma \llbracket u \rrbracket \mathcal{R}$ is defined by Lem. 6(1). It holds that $(\Gamma, \Gamma') \in \llbracket \alpha \rrbracket^\Phi_\Sigma$ and, by Lem. 6(2), $\mathcal{R} \llbracket u \rrbracket \mathcal{R} \equiv^\phi_\Sigma \Gamma \llbracket u \rrbracket \mathcal{R} \equiv^\phi_\Sigma \Gamma'$.

(2) Let $(\Gamma, \Gamma') \in \llbracket \alpha \rrbracket^\Phi_\Sigma$. Then there is a $u \in \alpha$ such that $(\Gamma, \Gamma') \in \llbracket u \rrbracket^\Phi_\Sigma$. Hence, $\Gamma' = \Gamma \llbracket u \rrbracket \Gamma \equiv^\phi_\Sigma \Gamma \llbracket u \rrbracket \Gamma$ is defined. Since $\mathcal{R} \equiv^\phi_\Sigma \Gamma$, also $\mathcal{R}' = \mathcal{R} \llbracket u \rrbracket \mathcal{R} \equiv^\phi_\Sigma \Gamma \llbracket u \rrbracket \mathcal{R}\equiv^\phi_\Sigma \Gamma'$ and, by Lem. 6(2), $\mathcal{R} \llbracket u \rrbracket \mathcal{R} \equiv^\phi_\Sigma \Gamma \llbracket u \rrbracket \mathcal{R} \equiv^\phi_\Sigma \Gamma'$.

## 5 Symbolic Epistemic Processes

We now consider symbolic execution of epistemic processes in the context of symbolic epistemic states and updates over a symbolic epistemic signature $(\Sigma, \Phi)$. For this purpose, we restrict the set $\mathcal{P}_\Sigma$ of epistemic processes to those processes whose guards are in $\text{bcl}(\Phi)$ and whose actions are in $\mathcal{A}_\Sigma^\Phi$. The resulting set of processes is denoted by $\mathcal{P}_\Sigma^\Phi$.

### 5.1 Symbolic Process Semantics

Given a symbolic epistemic signature $(\Sigma, \Phi)$, a *symbolic epistemic process configuration* over $(\Sigma, \Phi)$ is a pair $(P, \Gamma)$ where $P \in \mathcal{P}_\Sigma^\Phi$ and $\Gamma \in \mathcal{S}_\Sigma^\Phi$ is a symbolic epistemic state. Its *symbolic epistemic semantics* is the labelled transition system generated from $(P, \Gamma)$ by the following rule which in turn builds on the operational semantics of epistemic processes in Tab. 1 now applied to processes in $\mathcal{P}_\Sigma^\Phi$:

$$(P, \Gamma) \xrightarrow{\alpha} \Phi_\Sigma (P', \Gamma') \quad \text{if } P \xrightarrow{\alpha} \Phi_\Sigma P', \Gamma \models^\Phi_\Sigma \mathcal{R}, \text{ and } (\Gamma, \Gamma') \in \llbracket \alpha \rrbracket^\Phi_\Sigma$$

**Example 11.** The victim-rescuer scenario of Ex. 4 is an epistemic process over $(\Sigma_{vr}, \Phi_{vr})$ with $\Phi_{vr} = \{ K_{vr} h, K_{vr} h, K_{vr} K_{vr} h \}$ as shown in Ex. 10. When starting $\text{Vict} \parallel \text{Rese}$ in the symbolic epistemic state $\{ K_{vr} h \}$ over $\Phi_{vr}$, the symbolic transition system of Fig. 2 evolves (again with the “chosen” epistemic actions in gray). In contrast to the concrete case of Ex. 5, the symbolic transition system is finite — without the need of identifying w.r.t. bisimilarity — and also the knowledge bases remain small.

Let us briefly discuss a crucial difference to belief base expansions [25]. Suppose that $\neg K_R h$ is also part of the focus formulae. It then is also contained in the initial abstraction which would be $\{ K_{vr} h, \neg K_{vr} h \}$. Our symbolic epistemic update by $\text{prv}_{R}^h(K_{vr} h)$ discards $\neg K_{vr} h$ from the knowledge base and replaces it by $K_{vr} h$. In belief base expansions, however, formulae can only be added, not deleted.
Figure 2: Transition system for the symbolic victim rescuer process configurations.

5.2 Symbolic Dynamic Process Logic

Also the dynamic process logic formulæ defined in Sect. 3 can be interpreted over symbolic process configurations \((P, \Gamma)\). Compound actions now have to be in the set \(C_{\Sigma}^S \subseteq C_{\Sigma}\) where actions \(\alpha\) are in \(A_{\Sigma}^S\) and tests \(\phi\) satisfy \(\phi \in bcl(\Phi)\). The semantics of a compound action \(\sigma \in C_{\Sigma}^S\) is the relation \([\sigma]^{\Phi}_{\mathcal{D}, \Sigma}\) between symbolic epistemic process configurations inductively defined by:

\[
[\sigma]^{\Phi}_{\mathcal{D}, \Sigma} = \{(\phi, \psi) \mid (P, \Gamma) \models_{\mathcal{D}, \Sigma} \phi \}
\]

Moreover, in the symbolic context the set \(\mathcal{D}_{\Sigma}\) of dynamic epistemic formulæ is restricted to the set \(\mathcal{D}_{\Sigma}^S \subseteq \mathcal{D}_{\Sigma}\) of those formulæ which contain only basic formulæ \(\varphi \in \Phi\) and compound actions \(\sigma \in C_{\Sigma}^S\). Then the satisfaction of a dynamic formula \(\psi \in \mathcal{D}_{\Sigma}^S\) by a symbolic epistemic process configuration \((P, \Gamma)\) is inductively defined by:

\[
(P, \Gamma) \models_{\mathcal{D}, \Sigma} \varphi \iff \varphi \in \Gamma \quad \text{(case } \varphi \in \Phi) \\
(P, \Gamma) \models_{\mathcal{D}, \Sigma} \text{true}
\]

If \(\neg \psi \notin \Phi\) : \((P, \Gamma) \models_{\mathcal{D}, \Sigma} \neg \psi\) if not \((P, \Gamma) \models_{\mathcal{D}, \Sigma} \psi\)

If \(\psi_1 \land \psi_2 \notin \Phi\) : \((P, \Gamma) \models_{\mathcal{D}, \Sigma} \psi_1 \land \psi_2 \iff (P, \Gamma) \models_{\mathcal{D}, \Sigma} \psi_1\) and \((P, \Gamma) \models_{\mathcal{D}, \Sigma} \psi_2\)

\[
(P, \Gamma) \models_{\mathcal{D}, \Sigma} [\sigma] \psi \iff (P', \Gamma') \models_{\mathcal{D}, \Sigma} [\sigma] \psi \text{ f.a. } (P', \Gamma') \in [\sigma]^{\Phi}_{\mathcal{D}, \Sigma}
\]

The well-definedness argument for true, the negation, and the conjunction is the same as for the Boolean closure, see Sect. 4.1.

**Example 12.** The two epistemic dynamic process formulæ of Ex. 6 are formulæ in \(\mathcal{D}_{\Sigma}^S\) and the symbolic process configuration \((Sys, \Gamma_0)\) in Ex. 11 satisfies both properties.

5.3 Relating Processes in Epistemic and Symbolic Environments

As a consequence of Lem. 7 we can prove that semantic and symbolic process configurations mutually simulate action execution while preserving \(\Phi\)-equivalence.
Proposition 2. Let $P \in \mathcal{P}_\Sigma^K$, $(P, \mathfrak{R})$ an epistemic process configuration over $\Sigma$ and $(P, \Gamma)$ a symbolic epistemic process configuration over $(\Sigma, \Phi)$. Let $\alpha \in \mathcal{A}_\Sigma^K$ be an epistemic choice action, and let $\mathfrak{R} \equiv_{\Sigma}^{\Phi} \Gamma$ hold.

1. If $(P, \mathfrak{R}) \stackrel{\alpha}{\to}_\Sigma (P', \mathfrak{R}')$, then there is a $\Gamma' \in \mathcal{S}_\Sigma^K$ such that $(P, \Gamma) \stackrel{\alpha}{\to}_\Sigma (P', \Gamma')$ and $\mathfrak{R}' \equiv_{\Sigma}^{\Phi} \Gamma'$.

2. If $(P, \Gamma) \stackrel{\alpha}{\to}_\Sigma (P', \Gamma')$, then there is a $\mathfrak{R}' \in \mathcal{K}_\Sigma$ such that $(P, \mathfrak{R}) \stackrel{\alpha}{\to}_\Sigma (P', \mathfrak{R}')$ and $\mathfrak{R}' \equiv_{\Sigma}^{\Phi} \Gamma'$.

Proof. (1) Let $(P, \mathfrak{R}) \stackrel{\alpha}{\to}_\Sigma (P', \mathfrak{R}')$ be given. Then $P \stackrel{\alpha}{\to}_\Sigma P'$ with $\mathfrak{R} \equiv_{\Sigma}^{\Phi} \Gamma$ and $\mathfrak{R}' \equiv_{\Sigma}^{\Phi} \Gamma'$. Since $\mathfrak{R} \equiv_{\Sigma}^{\Phi} \Gamma$ and $\mathfrak{R} \equiv_{\Sigma}^{\Phi} \Gamma'$ we have, by Lem. 3, $\Gamma \equiv_{\Sigma}^{\Phi} \mathfrak{R}$ and, by Lem. 7(1), there exists $\Phi' \in \mathcal{S}_\Sigma^K$ such that $(\Gamma, \Gamma') \in [\alpha] \Sigma^K$ and $\mathfrak{R}' \equiv_{\Sigma}^{\Phi} \Gamma'$. Therefore $(P, \mathfrak{R}) \stackrel{\alpha}{\to}_\Sigma (P', \mathfrak{R}')$ and $\mathfrak{R}' \equiv_{\Sigma}^{\Phi} \Gamma'$.

(2) Let $(P, \Gamma) \stackrel{\alpha}{\to}_\Sigma (P', \Gamma')$ be given. Then $P \stackrel{\alpha}{\to}_\Sigma P'$ with $\Gamma \equiv_{\Sigma}^{\Phi} \Sigma$ and $(\Gamma, \Gamma') \in [\alpha] \Sigma^K$. Since $\mathfrak{R} \equiv_{\Sigma}^{\Phi} \Gamma$ we have, by Lem. 3, $\equiv_{\Sigma}^{\Phi} \mathfrak{R}$ and, by Lem. 7(2), there exists $\mathfrak{R}' \in \mathcal{K}_\Sigma$ such that $(\mathfrak{R}, \mathfrak{R}') \equiv_{\Sigma}^{\Phi} \Gamma'$ and $\mathfrak{R} \equiv_{\Sigma}^{\Phi} \Gamma'$. Therefore $(P, \mathfrak{R}) \stackrel{\alpha}{\to}_\Sigma (P', \mathfrak{R}')$ and $\mathfrak{R}' \equiv_{\Sigma}^{\Phi} \Gamma'$.

Proposition 3. Let $\sigma \in \mathcal{C}_\Sigma^K$ and let $(P, \mathfrak{R}) \equiv_{\mathfrak{R}, \Sigma}^{\Phi} (P, \Gamma)$ hold.

1. If $((P, \mathfrak{R}), (P', \mathfrak{R}')) \in [\sigma]_{\mathfrak{R}, \Sigma}$, then there exists a $\Gamma' \in \mathcal{S}_\Sigma^K$ such that $((P, \Gamma), (P', \Gamma')) \in [\sigma]_{\mathfrak{R}, \Sigma}$ and $(P', \mathfrak{R}') \equiv_{\mathfrak{R}, \Sigma}^{\Phi} (P', \Gamma')$.

2. If $((P, \Gamma), (P', \Gamma')) \in [\sigma]_{\mathfrak{R}, \Sigma}$, then there exists a $\mathfrak{R}' \in \mathcal{K}_\Sigma$ such that $((P, \mathfrak{R}), (P', \mathfrak{R}')) \in [\sigma]_{\mathfrak{R}, \Sigma}$ and $(P', \mathfrak{R}') \equiv_{\mathfrak{R}, \Sigma}^{\Phi} (P', \Gamma')$.

Proof. This follows by structural induction on the form of $\sigma \in \mathcal{C}_\Sigma^K$, where $\alpha \in \mathcal{A}_\Sigma^K$ is covered by Prop. 2, $\phi?$ with $\phi \in \text{bel}((\Phi)$ by Lem. 3, and all other cases follow directly from the induction hypothesis.

Finally, $\Phi$-equivalent process configurations satisfy the same dynamic process logic formulae. Thus symbolic epistemic process configurations can be considered as correct realisations of (concrete) epistemic process configurations.

Theorem 1. Let $P \in \mathcal{P}_\Sigma^K$ and let $(P, \mathfrak{R}) \equiv_{\mathfrak{R}, \Sigma}^{\Phi} (P, \Gamma)$ hold. Then for all $\psi \in \mathcal{D}_\Sigma^K$, it holds that $(P, \mathfrak{R}) \models_{\mathfrak{R}, \Sigma} \psi \iff (P, \Gamma) \models_{\mathfrak{R}, \Sigma} \psi$.

Proof. This follows by structural induction on the form of $\psi \in \mathcal{D}_\Sigma^K$, where $\phi \in \Phi$, true, negation, and conjunction are shown as for Boolean closures in Lem. 3 and the case $[\sigma] \psi$ is a consequence of Prop. 3.

6 Conclusions and Future Work

We studied symbolic realisations of epistemic processes which interact by announcing knowledge. For this purpose, we considered abstractions of epistemic Kripke models to subsets of so-called focus formulae that capture the knowledge interest of agents depending on the application. The abstraction allows to uniformly retain epistemic actions based on the action models of [3]. These actions can be symbolically executed using the computation of weakest liberal preconditions.
which are afforded by reduction axioms from dynamic epistemic logic. We applied the approach to a symbolic interpretation of epistemic processes and proposed a dynamic process logic for specification of process properties. We demonstrated that epistemic action executions started in equivalent concrete and symbolic process configurations mutually simulate each other such that validity of dynamic process formulæ is invariant.

In future work we plan to integrate actions that can change facts by assignments [17] and we want to validate the approach by larger case studies. In particular, it will be interesting to see whether our symbolic realisation of epistemic processes can be used in epistemic applications, like epistemic planning [8]. As an important topic, we aim to divide the global symbolic epistemic states agent-wise into a family of local symbolic states and to study a transformation of symbolic process configurations into distributed symbolic configurations. We are also interested in tools for model-checking epistemic processes against formulæ of our dynamic process logic. For this, we want to compare our symbolic approach with other symbolic techniques used for epistemic model-checking, like [6].

Acknowledgements We would like to thank anonymous reviewers of this paper for valuable comments.

References


