# Epistemic Ensembles

Rolf Hennicker[1], Alexander Knapp[2], and Martin Wirsing[1]

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

**Abstract.** An ensemble consists of a set of computing entities which collaborate to reach common goals. We introduce epistemic ensembles that use shared knowledge for collaboration between agents. Collaboration is achieved by different kinds of knowledge announcements. For specifying epistemic ensemble behaviours we use formulas of dynamic logic with compound ensemble actions. Our semantics relies on an epistemic notion of ensemble transition systems as behavioural models. These transition systems describe control flow over epistemic states for expressing knowledge-based collaboration of agents. Specifications are implemented by epistemic processes that are composed in parallel to form ensemble realisations. We give a formal operational semantics of these processes that generates an epistemic ensemble transition system. A realisation is correct w. r. t. an ensemble specification if its semantics is a model of the specification.

## 1 Introduction

An ensemble [13] is formed by a collection of agents which run concurrently to accomplish (together) a certain task. For that purpose agents must collaborate in some way, for instance by explicit interaction via message passing [8,9]. In the context of the epistemic approach considered here collaboration is based on the knowledge that agents have about themselves, about other agents and about their environment. Any change of knowledge caused by an action of one agent may influence the behaviour of other agents. Hence interaction is implicit. This is related to the ideas of autonomic component ensembles where coordination is achieved via knowledge repositories in which information is stored and from which information is retrieved; see, e.g., [5].

We propose a dynamic logic for specifying properties of epistemic ensembles. Our semantic models are labelled transition systems with atomic ensemble actions as labels. Labelled transitions model two aspects, (i) the control flow of an ensemble and (ii) changes of epistemic information caused by the epistemic effect of an agent action. To model the latter we introduce an epistemic state operator which assigns to each ensemble state $s$ of the system an epistemic state $\Omega(s)$ modelling the current epistemic information available in the ensemble. Note that different ensemble states can carry the same epistemic information, in particular if a non-epistemic agent action is performed. Then a transition between the two has a pure control flow effect. The set of ensemble states is restricted to states which are reachable by system transitions from the initial ones. This reflects our intuition that we want to consider ensembles as dynamic processes.

The restriction to reachable states and the ability to model control flow in the semantics is a crucial difference to public announcement logic (PAL) and dynamic epistemic logic (DEL); see, e.g., [6]. Instead of stating requirements for ensemble behaviours these logics are more appropriate for the verification of pre- and postconditions of given epistemic programs. [12] was one of the motivations for our work; it proposes to describe structural properties of ensembles with epistemic logic. An approach which deals with control flow as well are the knowledge-based programs in [7]. The semantic basis are system runs and the interpretation of knowledge tests inside the programs needs a circular procedure by relying on possible system runs at the same time.

After recapitulating basic notions of epistemic logic and epistemic actions in Sect. 2, we present our proposal to specifications of epistemic ensembles in Sect. 3 and provide a (formal) semantics for them in Sect. 4. In Sect. 5 we present an approach to realise epistemic ensemble specifications by a set of concurrently running epistemic processes and we define a correctness notion for such realisations. We finish in Sect. 6 with some concluding remarks.

## 2 Epistemic Logic and Epistemic Actions

We provide the basis for the epistemic treatment of ensembles considered later on. First, we summarise basic notions of epistemic logic. Then, we provide a summary of epistemic actions and adjust the definitions for their use in epistemic ensemble development. More details can be found in the literature, for instance [3,6].

### 2.1 Epistemic Logic

An *epistemic signature* $(P, A)$ consists of a set $P$ of *propositions* and a finite set $A$ of *agents*. The set $\Phi_{P,A}$ of *epistemic formulæ* $\varphi$ over $(P, A)$ is defined by the following grammar:

$$\varphi ::= \mathrm{true} \mid p \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{K}_a\, \varphi$$

where $p \in P$ and $a \in A$. The epistemic formula $\mathsf{K}_a\, \varphi$ is to be read as "agent $a$ *knows* $\varphi$". As usual, we write false for $\neg\mathrm{true}$, $\varphi_1 \rightarrow \varphi_2$ for $\neg\varphi_1 \vee \varphi_2$, and $\varphi_1 \wedge \varphi_2$ for $\neg(\neg\varphi_1 \vee \neg\varphi_2)$.

For each $a \in A$, $\Phi^a_{P,A}$ denotes the set of all purely propositional connections (including $\mathrm{true}$ and hence $\mathrm{false}$) of epistemic formulæ starting with the modality $\mathsf{K}_a$. These formulæ focus on the knowledge of agent $a$. The set $\Phi^a_{P,A}$ is defined by the following grammar:

$$\varphi^a ::= \mathrm{true} \mid \neg\varphi^a \mid \varphi^a \vee \varphi^a \mid \mathsf{K}_a\, \varphi$$

with $\varphi \in \Phi_{P,A}$. An *epistemic structure* $K = (W, R, L)$ over $(P, A)$ consists of a set $W$ of *worlds*, an $A$-indexed family $R = (R_a \subseteq W \times W)_{a \in A}$ of epistemic *accessibility relations*, and a *labelling* $L\colon W \to \wp P$ which determines for each world $w \in W$ the set of propositions valid in $w$. The accessibility relations of epistemic structures are assumed to be equivalence relations. For any $a \in A$, $(w, w') \in R_a$ models that agent $a$ cannot distinguish the two worlds $w$ and $w'$.

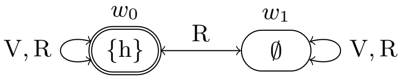
An *epistemic state* over $(P, A)$ is a pointed epistemic structure $\mathfrak{K} = (K, w)$ over $(P, A)$ where $w \in W$ determines an actual world. The class of all epistemic states over $(P, A)$ is denoted by $\mathit{EpiSt}(P, A)$.

For any epistemic signature $(P, A)$ and epistemic structure $K = (W, R, L)$ over $(P, A)$ the *satisfaction* of an epistemic formula $\varphi \in \mathit{\Phi}_{P,A}$ by $K$ at a point $w \in W$, written $K, w \models \varphi$, is inductively defined as follows:

$$
\begin{aligned}
&K, w \models \mathrm{true}\\
&K, w \models p \iff p \in L(w)\\
&K, w \models \neg\varphi \iff \text{not } K, w \models \varphi\\
&K, w \models \varphi_1 \vee \varphi_2 \iff K, w \models \varphi_1 \text{ or } K, w \models \varphi_2\\
&K, w \models \mathsf{K}_a\,\varphi \iff K, w' \models \varphi \text{ for all } w' \in W \text{ with } (w, w') \in R_a
\end{aligned}
$$

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 $\mathfrak{K} = (K, w)$ and for $\varphi \in \mathit{\Phi}_{P,A}$, $\mathfrak{K} \models \varphi$ means $K, w \models \varphi$.

*Example 1.* We consider a (strongly simplified) victim rescue ensemble from a case study [11] of the ASCENS-project [14,15]. In the ensemble an agent, called V, is a victim who is to be supposed to be rescued by an agent R. There is one atomic proposition $\mathrm{h}$ indicating that the victim needs help and this is true in the actual world. The victim knows this but the rescuer does not. This situation is represented in the following diagram by the epistemic state $(K_0, w_0)$, where, indeed, R cannot distinguish between the actual world $w_0$ and the possible world $w_1$:



The self-loops represent reflexivity of the accessibility relations. Note that $(K_0, w_0) \models \mathsf{K}_{\mathrm{V}}\,\mathrm{h}$ but $(K_0, w_0) \models \neg\mathsf{K}_{\mathrm{R}}\,\mathrm{h}$ and $(K_0, w_0) \models \neg\mathsf{K}_{\mathrm{R}}\,\mathsf{K}_{\mathrm{V}}\,\mathrm{h}$. □

Let $K_1 = (W_1, R_1, L_1)$, $K_2 = (W_2, R_2, L_2)$ be two epistemic structures over $(P, A)$. A *bisimulation* between $K_1$ and $K_2$ is a relation $B \subseteq W_1 \times W_2$ such that for all $(w_1, w_2) \in B$ and all $a \in A$ the following holds:

1. $L_1(w_1) = L_2(w_2)$,
2. for each $w_1' \in W_1$, if $(w_1, w_1') \in R_{1,a}$ then there is a $w_2' \in W_2$ such that $(w_2, w_2') \in R_{2,a}$ and $(w_1', w_2') \in B$, and
3. for each $w_2' \in W_2$, if$(w_2, w_2') \in R_{2,a}$ then there is a $w_1' \in W_1$ such that $(w_1, w_1') \in R_{1,a}$ and $(w_1', w_2') \in B$.

Two epistemic states $\mathfrak{K}_1 = (K_1, w_1)$ and $\mathfrak{K}_2 = (K_2, w_2)$ over $(P, A)$ are *bisimilar*, written $\mathfrak{K}_1 \approx \mathfrak{K}_2$, if there exists a bisimulation $B$ between $K_1$ and $K_2$ such that $(w_1, w_2) \in B$.

The following lemma is a well-known result from epistemic logic; see, e.g., [3,6].

**Lemma 1 (Invariance of epistemic formulæ).** *Let $\mathfrak{K}_1$ and $\mathfrak{K}_2$ be epistemic states over $(P, A)$ such that $\mathfrak{K}_1 \approx \mathfrak{K}_2$. Then, for any $\varphi \in \mathit{\Phi}_{P,A}$, $\mathfrak{K}_1 \models \varphi$ if, and only if, $\mathfrak{K}_2 \models \varphi$.* □

The converse is also valid for image-finite epistemic structures $K = (W, R, L)$, i.e., if for each world $w \in W$ and agent $a \in A$ there exist only finitely many pairs $(w, w') \in R_a$. Note that finiteness of $A$ does not imply image finiteness of epistemic structures over $(P, A)$; a counterexample is given in [6, p. 227].

### 2.2 Epistemic Actions

Epistemic logic deals with static aspects of knowledge captured by epistemic formulæ and their interpretation in epistemic states. A fundamental concept to support dynamic changes of knowledge is public announcement logic (PAL [3]) where knowledge about an epistemic state (formalised by a formula) can be announced to all agents. This may affect the knowledge of the agents leading to a new epistemic situation. More elaborated epistemic actions, like completely private and semi-private announcements, were also considered and a general proposal to model epistemic actions in terms of so-called action models was set up in [2]. In our approach action models will be called action structures in order to avoid confusion with the models of ensemble specifications later on.

An *epistemic action structure* $U = (Q, F, \mathit{pre})$ over $(P, A)$ consists of a set of *action points* $Q$, an $A$-indexed family $F = (F_a \subseteq Q \times Q)_{a \in A}$ of epistemic *action accessibility relations*, and a *precondition* function $\mathit{pre}\colon Q \to \Phi_{P,A}$. We assume again that the accessibility relations are equivalences. In the literature, action points are also called "events". For any agent $a$, $(q, q') \in F_a$ models that agent $a$ cannot distinguish between occurrences of $q$ and $q'$. For $q \in Q$, the epistemic formula $\mathit{pre}(q)$ determines a condition under which $q$ can happen.

An *epistemic action* over $(P, A)$ is a pointed epistemic action structure $\mathfrak{u} = (U, q)$ over $(P, A)$ where $q \in Q$ determines an actual action point. The set $\mathcal{A}_{P,A}$ of *epistemic actions with (non-deterministic) choice* over $(P, A)$ is defined by

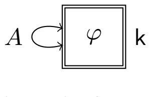
$$\alpha ::= \mathfrak{u} \mid \alpha + \alpha$$

where $\mathfrak{u} = (U, q)$ is an epistemic action over $(P, A)$. The precondition of an epistemic action with choice is given by $\mathit{pre}(\mathfrak{u}) = \mathit{pre}(q)$, $\mathit{pre}(\alpha_1 + \alpha_2) = \mathit{pre}(\alpha_1) \vee \mathit{pre}(\alpha_2)$.

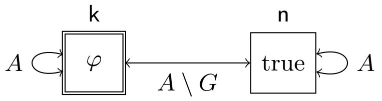
*Example 2.* (a) *Public announcement* of an epistemic formula $\varphi \in \Phi_{P,A}$ to all agents in $A$ is modelled by the epistemic action $(U_{pub,\varphi}, \mathsf{k})$ where

$$U_{pub,\varphi} = (Q_{pub}, F_{pub}, \mathit{pre}_{pub,\varphi})$$

with $Q_{pub} = \{\mathsf{k}\}$, $F_{pub,a} = \{(\mathsf{k}, \mathsf{k})\}$ for all $a \in A$, and $\mathit{pre}_{pub,\varphi} = \{\mathsf{k} \mapsto \varphi\}$. There is only one action point $\mathsf{k}$ and hence any agent in $A$ considers only the occurrence of $\mathsf{k}$ possible. According to the precondition of $\mathsf{k}$ the action can only be executed in an epistemic state $\mathfrak{K}$ where the announced formula $\varphi$ holds. The epistemic action $(U_{pub,\varphi}, \mathsf{k})$ is graphically represented by the following diagram.



(b) *Private announcement* of an epistemic formula $\varphi \in \Phi_{P,A}$ to a group $G \subseteq A$ of agents is modelled by the epistemic action $(U_{priv,G,\varphi}, \mathsf{k})$ graphically represented by the following diagram:



The action structure $U_{priv,G,\varphi}$ has two action points $\mathsf{k}$ and $\mathsf{n}$. Point $\mathsf{k}$ represents that the announcement of $\varphi$ happens which should only be the case if $\varphi$ holds in the current epistemic state and therefore $pre(\mathsf{k}) = \varphi$. Only agents in the group $G$ can recognise this event. All other agents consider it possible that nothing happened which is represented by $\mathsf{n}$. This should not have a proper precondition and therefore $pre(\mathsf{n}) = \mathrm{true}$.[3] $\square$

The effect of an epistemic action on an epistemic state is defined by the product update as constructed in [1]. First, we define the product update of an epistemic structure by an epistemic action structure and then we use this for the product update of their pointed versions. The *product update* of an epistemic structure $K = (W, R, L)$ over $(P, A)$ and an epistemic action structure $U = (Q, F, pre)$ over $(P, A)$ is the epistemic structure $K \lhd U = (W', R', L')$ over $(P, A)$ with

$$\begin{aligned} &W' = \{(w, q) \in W \times Q \mid K, w \models pre(q)\}\ , \\ &R'_a = \{((w, q), (w', q')) \in W' \times W' \mid (w, w') \in R_a,\ (q, q') \in F_a\} \text{ for all } a \in A, \\ &L'(w, q) = L(w) \text{ for all } (w, q) \in W'. \end{aligned}$$

According to the definition of the relations $R'_a$ the uncertainty of an agent $a$ in a world $(w, q)$ is determined by the uncertainty of $a$ about world $w$ and its uncertainty about the occurrence of $q$. Note that the relations $R'_a$ are again equivalence relations and therefore the product update for epistemic structures is well-defined.

Let $\mathfrak{K} = (K, w) \in EpiSt(P, A)$ be an epistemic state and $\mathfrak{u} = (U, q)$ be an epistemic action over $(P, A)$. If $\mathfrak{K} \models pre(\mathfrak{u})$ then the *product update* of $\mathfrak{K}$ and $\mathfrak{u}$ is defined and given by the epistemic state $\mathfrak{K} \lhd \mathfrak{u} = (K \lhd U, (w, q)) \in EpiSt(P, A)$.

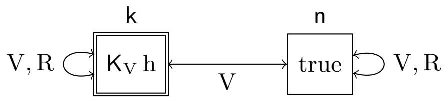
The semantics of each epistemic action with choice $\alpha \in \mathcal{A}_{P,A}$ is given by a set of relations $[\![\alpha]\!] \subseteq EpiSt(P, A) \times EpiSt(P, A)$ between epistemic states inductively defined by:

$$\begin{aligned} &[\![\mathfrak{u}]\!] = \{(\mathfrak{K}, \mathfrak{K} \lhd \mathfrak{u}) \mid \mathfrak{K} \models pre(\mathfrak{u})\}, \\ &[\![\alpha_1 + \alpha_2]\!] = [\![\alpha_1]\!] \cup [\![\alpha_2]\!], \text{ i.e. union of relations.} \end{aligned}$$

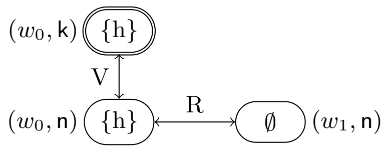
Note that for each $\alpha \in \mathcal{A}_{P,A}$ and $\mathfrak{K} \in EpiSt(P, A)$ it holds: There exists a $\mathfrak{K}' \in EpiSt(P, A)$ such that $(\mathfrak{K}, \mathfrak{K}') \in [\![\alpha]\!]$ if, and only if, $\mathfrak{K} \models pre(\alpha)$.

*Example 3.* We consider the victim rescue example from Ex. 1 and instantiate private announcement of Ex. 2(b) to the case in which it is privately announced to R that V knows that h holds. Thus we consider the epistemic action $(U_{priv,\{\mathrm{R}\},\mathsf{K}_\mathrm{V}\,\mathrm{h}}, \mathsf{k})$ represented by the following diagram where V does not know whether R got an announcement:

[3] We do not consider here completely private announcements where the agents not in $G$ would not consider it possible that the announcement happened. To model this case one would need non-symmetric accessibility relations.



We apply this action to the epistemic state $(K_0, w_0)$ in Ex. 1. The product update yields the epistemic state $(K_1, (w_0, \mathsf{k}))$ shown, without reflexive accessibility edges, below. The world $(w_1, \mathsf{k})$ does not appear since $(K_0, w_1) \not\models \mathsf{K}_{\mathrm{V}}\,\mathrm{h}$ which is the precondition of $\mathsf{k}$.



Note that $(K_1, (w_0, \mathsf{k})) \models \mathsf{K}_{\mathrm{R}}\,\mathsf{K}_{\mathrm{V}}\,\mathrm{h}$ but $(K_1, (w_0, \mathsf{k})) \models \neg\mathsf{K}_{\mathrm{V}}\,\mathsf{K}_{\mathrm{R}}\,\mathsf{K}_{\mathrm{V}}\,\mathrm{h}$, i.e. R knows that V knows that $h$ holds, but V does not know that R knows this.

If we apply the epistemic action $(U_{priv,\{\mathrm{R}\},\mathsf{K}_{\mathrm{V}}\,\mathrm{h}}, \mathsf{n})$ to $(K_0, w_0)$ we obtain the epistemic state $(K_1, (w_0, \mathsf{n}))$. Note that $(K_1, (w_0, \mathsf{n})) \models \neg\mathsf{K}_{\mathrm{R}}\,\mathsf{K}_{\mathrm{V}}\,\mathrm{h}$. □

The next lemma shows that bisimulation is preserved by application of epistemic actions; see, e.g., [6].

**Lemma 2.** *Let $\mathfrak{K}_1$ and $\mathfrak{K}_2$ be epistemic states over $(P, A)$ such that $\mathfrak{K}_1 \approx \mathfrak{K}_2$ and let $\mathfrak{u}$ be an epistemic action over $(P, A)$. Then $\mathfrak{K}_1 \lhd \mathfrak{u}$ is defined, if and only if, $\mathfrak{K}_2 \lhd \mathfrak{u}$ is defined and then it holds $\mathfrak{K}_1 \lhd \mathfrak{u} \approx \mathfrak{K}_2 \lhd \mathfrak{u}$.* □

We generalise Lem. 2 to epistemic actions with choice. The proof is straightforward by induction on the form of $\alpha$.

**Lemma 3.** *Let $\mathfrak{K}_1$ and $\mathfrak{K}_2$ be as in Lem. 2 such that $\mathfrak{K}_1 \approx \mathfrak{K}_2$ and let $\alpha$ be an epistemic action with choice. Then, for any $\mathfrak{K}'_1$ with $(\mathfrak{K}_1, \mathfrak{K}'_1) \in [\![\alpha]\!]$, there exists $\mathfrak{K}'_2$ with $(\mathfrak{K}_2, \mathfrak{K}'_2) \in [\![\alpha]\!]$ such that $\mathfrak{K}'_1 \approx \mathfrak{K}'_2$; the converse holds for any $\mathfrak{K}'_2$ with $(\mathfrak{K}_2, \mathfrak{K}'_2) \in [\![\alpha]\!]$.* □

## 3 Epistemic Ensemble Specifications

An ensemble is formed by a collection of agents which run concurrently to accomplish (together) a certain task. For that purpose agents must collaborate in some way, for instance by explicit interaction via message passing [8,9]. In the context of the epistemic approach considered here collaboration is based on the knowledge that agents have about themselves, about other agents and about their environment. Any change of knowledge caused by an action of one agent may influence the behaviour of other agents. Hence interaction is implicit.

Formally, an *agent action* is given by an action name $e$ to which an agent $o(e)$ is associated, the "owner" of $e$, who is able to execute that action. An *epistemic ensemble signature* $\Sigma = (P, A, E)$ consists of an epistemic signature $(P, A)$ and a set $E$ of agent actions such that for each $e \in E, o(e) \in A$. The set $E$ is split into a set $eE$ of *epistemic agent actions* and a set $nE$ of *non-epistemic agent actions*. The idea is that any agent action may have an effect on the control flow of an ensemble. The non-epistemic agent actions, however, do not change the epistemic state of an ensemble while epistemic agent actions in general do.

The *epistemic effect* of an agent action $e \in E$ is formalised by a relation $\mathit{eeff}(e) \subseteq \mathit{EpiSt}(P, A) \times \mathit{EpiSt}(P, A)$ between epistemic states over $(P, A)$. For non-epistemic agent actions $e \in nE$ we define $\mathit{eeff}(e) = \{(\mathfrak{K}, \mathfrak{K}) \mid \mathfrak{K} \in \mathit{EpiSt}(P, A)\}$. The non-epistemic agent actions are specific actions depending on the application at hand. For epistemic agent actions $e \in eE$ their epistemic effect must be explicitly defined. For this purpose we associate to $e$ an epistemic action expression with choice $\alpha \in \mathcal{A}_{P,A}$, whose semantics is clear from Sect. 2.2, and thus define $\mathit{eeff}(e) = [\![\alpha]\!]$. Moreover, we set $\mathit{pre}(e) = \mathit{pre}(\alpha)$ and require that $\mathit{pre}(e) \in \mathit{\Phi}^{o(e)}_{P,A}$. This constraint expresses that an epistemic agent action with owner $a$ should have a precondition which concerns, and hence can be tested, by $a$; similarly to the knowledge tests of knowledge-based programs in [7]. Thus the epistemic action expressions in Sect. 2.2. will be used as primitives to define the epistemic effect of higher level epistemic actions for agents.

In this paper we assume given, for each epistemic signature $(P, A)$, the following set of epistemic agent actions from which particular instantiations can be chosen for a concrete ensemble signature.

*Public announcement by an agent*: This action is a special case of public announcement such that the announcement is performed by an agent $a$ "inside" the system. As a consequence, agent $a$ does not simply announce a formula $\varphi$ but it must indeed know $\varphi$ and must announce that, i.e. $\mathsf{K}_a\, \varphi$. Formally, for each $a \in A$ and $\varphi \in \mathit{\Phi}_{P,A}$, public announcement by $a$ is denoted by the epistemic agent action $\mathit{pub}^a(\mathsf{K}_a\, \varphi)$ over $(P, A)$ with owner $o(\mathit{pub}^a(\mathsf{K}_a\, \varphi)) = a$. The epistemic effect of this action is defined by $\mathit{eeff}(\mathit{pub}^a(\mathsf{K}_a\, \varphi)) =_{\mathrm{def}} [\![(U_{\mathit{pub},\mathsf{K}_a\, \varphi}, \mathsf{k})]\!]$ where the latter is the epistemic public announcement action in Ex. 2(a) with semantics defined by product update as described in Sect. 2.2. Note that $\mathit{pre}(\mathit{pub}^a(\mathsf{K}_a\, \varphi)) = \mathit{pre}(U_{\mathit{pub},\mathsf{K}_a\, \varphi}, \mathsf{k}) = \mathsf{K}_a\, \varphi \in \mathit{\Phi}^a_{P,A}$.

*Reliable private sending*: In this case there is an agent $a$ who knows the validity of a formula $\varphi$ and sends the information that it knows $\varphi$, i.e. $\mathsf{K}_a\, \varphi$, to another agent $b$. The sending is reliable, i.e. the information will be received by $b$ and agent $a$ knows that. Formally, for each $a, b \in A$ and $\varphi \in \mathit{\Phi}_{P,A}$, reliable private sending is denoted by the epistemic agent action $\mathit{snd}^{a \to b}_{\mathrm{rel}}(\mathsf{K}_a\, \varphi)$ over $(P, A)$ with owner $o(\mathit{snd}^{a \to b}_{\mathrm{rel}}(\mathsf{K}_a\, \varphi)) = a$.

The epistemic effect of this action can be modelled as a special case of private announcement to a group of agents where the group is $\{a, b\}$ and the announcement is $\mathsf{K}_a\, \varphi$. Hence, we define $\mathit{eeff}(\mathit{snd}^{a \to b}_{\mathrm{rel}}(\mathsf{K}_a\, \varphi)) =_{\mathrm{def}} [\![(U_{\mathit{priv},\{a,b\},\mathsf{K}_a\, \varphi}, \mathsf{k})]\!]$; see Ex. 2(b). Obviously, $\mathit{pre}(\mathit{snd}^{a \to b}_{\mathrm{rel}}(\mathsf{K}_a\, \varphi)) = \mathsf{K}_a\, \varphi \in \mathit{\Phi}^a_{P,A}$ where $a = o(\mathit{snd}^{a \to b}_{\mathrm{rel}}(\mathsf{K}_a\, \varphi))$.

*Lossy private sending*: In this case there is again an agent $a$ who knows the validity of a formula $\varphi$ and sends the information $\mathsf{K}_a\, \varphi$ to another agent $b$. But this time the sending is unreliable and the information may get lost. Formally, for each $a, b \in A$ and $\varphi \in \mathit{\Phi}_{P,A}$, lossy private sending is denoted by the epistemic agent action $\mathit{snd}^{a \to b}_{\mathrm{los}}(\mathsf{K}_a\, \varphi)$ over $(P, A)$ with owner $o(\mathit{snd}^{a \to b}_{\mathrm{los}}(\mathsf{K}_a\, \varphi)) = a$.

For defining the epistemic effect of $\mathit{snd}^{a \to b}_{\mathrm{los}}(\mathsf{K}_a\, \varphi)$ we proceed as follows: Let $U_{\mathit{priv},\{b\},\mathsf{K}_a\, \varphi}$ be the epistemic action structure of Ex. 2(b) instantiated by $\{b\}$ and $\mathsf{K}_a\, \varphi$. Let $(U_{\mathit{priv},\{b\},\mathsf{K}_a\, \varphi}, \mathsf{k})$ and $(U_{\mathit{priv},\{b\},\mathsf{K}_a\, \varphi}), \mathsf{n})$ be the corresponding epistemic actions.

The first action expresses that after $a$ has sent the information $\mathsf{K}_a\,\varphi$, agent $b$ has received it, but $a$ (and all other agents) do not know this; they consider it possible that the information did not arrive. The second action expresses that after the sending of $\mathsf{K}_a\,\varphi$ by agent $a$, agent $b$ has not received anything and $b$ knows that. Hence, the information is lost, and $a$ (and all other agents besides $b$) do not know whether the information has arrived or not. The effect of lossy private sending must capture both possibilities. Therefore, it is modelled by a non-deterministic choice of the two actions, either the information is received or not. The sender does not know what happened and the receiver knows the sent information if, and only if, it has received it. Formally, we define

$$\mathit{eeff}(\mathit{snd}^{a\rightarrow b}_{\mathrm{los}}(\mathsf{K}_a\,\varphi)) =_{\mathrm{def}} [\![(U_{\mathit{priv},\{b\},\mathsf{K}_a\,\varphi}, \mathsf{k}) + (U_{\mathit{priv},\{b\},\mathsf{K}_a\,\varphi}, \mathsf{n})]\!] \ .$$

Then, $\mathit{pre}(\mathit{snd}^{a\rightarrow b}_{\mathrm{los}}(\mathsf{K}_a\,\varphi)) = \mathit{pre}((U_{\mathit{priv},\{b\},\mathsf{K}_a\,\varphi}, \mathsf{k}) + (U_{\mathit{priv},\{b\},\mathsf{K}_a\,\varphi}, \mathsf{n})) =$ $\mathit{pre}(U_{\mathit{priv},\{b\},\mathsf{K}_a\,\varphi}, \mathsf{k}) \vee \mathit{pre}(U_{\mathit{priv},\{b\},\mathsf{K}_a\,\varphi}, \mathsf{n}) = (\mathsf{K}_a\,\varphi \vee \mathrm{true}) \in \Phi^a_{P,A}$.

In the following we assume that $\Sigma = (P, A, E)$ is an epistemic ensemble signature. To specify global behaviours of ensembles performed by concurrently running agents we must consider ensemble actions which are formed by various combinations of agent actions. Therefore, the agent actions in $E$ are considered as atomic ensemble actions while complex ensemble actions are formed by using the standard operators of dynamic logic which are test ($\varphi$?), non-deterministic choice ($+$), sequential composition (;) and iteration ($^*$). The set $\mathcal{E}_\Sigma$ of *compound ensemble actions* over $\Sigma$ is defined by the following grammar:

$$\pi ::= e \ \mid \ \varphi? \ \mid \ \pi + \pi \ \mid \ \pi ; \pi \ \mid \ \pi^*$$

where $e \in E$ is an agent action and $\varphi \in \Phi_{P,A}$. If $E$ is finite, we write "$\mathrm{some}$" for the compound action obtained by combing with "$+$" all elements of $E$ and, for $e \in E$, we write $-e$ for the compound ensemble action obtained by combing with "$+$" all elements of $E \setminus \{e\}$.

Ensemble formulæ are used to specify properties of ensembles. They extend the formulæ of epistemic logic in Sect. 2.1 by including modalities with (compound) ensemble actions which allow us to specify the dynamic aspects of global ensemble behaviours. The set $\Psi_\Sigma$ of *epistemic ensemble formulæ* over $\Sigma = (P, A, E)$ is defined by the following grammar:

$$\psi ::= \varphi \ \mid \ \neg\psi \ \mid \ \psi \vee \psi \ \mid \ \langle\pi\rangle\psi$$

where $\varphi \in \Phi_{P,A}$ and $\pi \in \mathcal{E}_\Sigma$. The formula $\langle\pi\rangle\psi$ is to be read as "in the current ensemble state it is possible to execute $\pi$ leading to an ensemble state where formula $\psi$ holds". The abbreviations from epistemic logic are extended to epistemic ensemble logic. Furthermore, we abbreviate $\neg\langle\pi\rangle\neg\psi$ by $[\pi]\psi$ which is to be read as "each execution of $\pi$ in the current ensemble state leads to an ensemble state where the formula $\psi$ holds".

Using the shorthand notations for compound actions for finite $E$, we can specify safety properties with $[\mathrm{some}^*]\psi$; deadlock freeness is expressed by $[\mathrm{some}^*]\langle\mathrm{some}\rangle\mathrm{true}$. Liveness properties like "whenever an action $e$ has happened, an action $f$ can eventually occur", can be expressed by $[\mathrm{some}^*; e]\langle\mathrm{some}^*; f\rangle\mathrm{true}$. We can also express that an action $f$ must never occur when action $e$ has happened before by $[\mathrm{some}^*; e; \mathrm{some}^*; f]\mathrm{false}$.

**Definition 1 (Ensemble specification).** *An* ensemble specification $Sp = (\Sigma, Ax)$ *consists of an ensemble signature $\Sigma$ and a set $Ax \subseteq \Psi_\Sigma$ of ensemble formulæ, called* axioms *of $Sp$.* □

*Example 4.* We provide a requirements specification $Sp_{vr} = (\Sigma_{vr}, Ax_{vr})$ for victim rescue ensembles. The epistemic ensemble signature $\Sigma_{vr}$ consists of the proposition $\mathrm{h}$, of the two agents $\mathrm{V}$ and $\mathrm{R}$, of the two epistemic agent actions $\mathit{snd}_{\mathrm{los}}^{\mathrm{V}\to\mathrm{R}}(\mathsf{K}_\mathrm{V}\,\mathrm{h})$, $\mathit{snd}_{\mathrm{rel}}^{\mathrm{R}\to\mathrm{V}}(\mathsf{K}_\mathrm{R}\,\mathrm{h})$ with owners $o(\mathit{snd}_{\mathrm{los}}^{\mathrm{V}\to\mathrm{R}}(\mathsf{K}_\mathrm{V}\,\mathrm{h})) = \mathrm{V}$ and $o(\mathit{snd}_{\mathrm{rel}}^{\mathrm{R}\to\mathrm{V}}(\mathsf{K}_\mathrm{R}\,\mathrm{h})) = \mathrm{R}$, and two non-epistemic agent actions $\mathit{stop}$, $\mathit{rescue}$ with owners $o(\mathit{stop}) = \mathrm{V}$ and $o(\mathit{rescue}) = \mathrm{R}$. We use a lossy information transfer from $\mathrm{V}$ to $\mathrm{R}$ since the idea is that the rescuer is moving around in an exploration area and cannot get information when it is outside the victim's range. The information transfer from $\mathrm{R}$ to $\mathrm{V}$ is reliable, since we assume that once the rescuer is informed it will be close enough to the victim. For a victim rescue ensemble we require the following properties expressed by the two axioms (1) and (2) of $Ax_{vr}$:

– "Whenever the victim performs a lossy sending to the rescuer that it knows that $\mathrm{h}$ is valid, i.e. the victim needs help, it is eventually possible that the rescuer knows this."

(1) $[\mathrm{some}^*; \mathit{snd}_{\mathrm{los}}^{\mathrm{V}\to\mathrm{R}}(\mathsf{K}_\mathrm{V}\,\mathrm{h})]\langle\mathrm{some}^*\rangle\mathsf{K}_\mathrm{R}\,\mathrm{h}$

– "Whenever the rescuer has not yet rescued the victim but knows that the victim needs help, it is eventually possible that the rescuer rescues the victim."

(2) $[(-\mathit{rescue})^*]\mathsf{K}_\mathrm{R}\,\mathrm{h} \to \langle\mathrm{some}^*; \mathit{rescue}\rangle\mathrm{true}$

This specification can be generalised in many ways, for instance to more rescuers taking into account that it is sufficient that only one rescuer goes for rescuing. □

## 4 Semantics of Epistemic Ensemble Specifications

We will now turn to the semantics of epistemic ensemble logic and ensemble specifications. Our semantic models are labelled transition systems with atomic ensemble actions (i.e. agent actions) as labels. Labelled transitions model two aspects, (i) the control flow of an ensemble and (ii) changes of epistemic information caused by the epistemic effect of an agent action. To model the latter we introduce an epistemic state operator which assigns to each ensemble state $s$ of the system an epistemic state $\Omega(s)$. Ensemble states could be modelled by pairs $s = (\mathit{ctrl}, \mathfrak{K})$ where $\mathit{ctrl}$ is an explicit control state and $\mathfrak{K}$ is an epistemic state; then the state operator would be the projection to the second component, i.e. $\Omega(s) = \mathfrak{K}$. Our definition leaving control states implicit is, however, more general.

Of course, ensemble transitions must respect (up to bisimilarity) the epistemic effect of actions, which is expressed by condition (1a) below. Conversely, if an epistemic ensemble action is enabled in an ensemble state, then all epistemic effects of the action must be present (up to bisimilarity) in the transition system, which is expressed by (1b). This reflects that the choice of the effect of a (non-deterministic) epistemic action is made by the system environment, not by the agents of the ensemble.

Note that different ensemble states can carry the same epistemic information, in particular if a non-epistemic agent action is performed. Then a transition between the two has a pure control flow effect. The set of ensemble states is restricted to states which are reachable by system transitions from the initial ones which is expressed by condition (2) below. This reflects our intuition that we want to consider ensembles as processes with significant dynamic behaviour. The restriction to reachable states and the ability to model control flow in the semantics is a crucial difference to dynamic epistemic logic; see, e.g., [6].

**Definition 2 (Epistemic ensemble transition system).** *Let $\Sigma = (P, A, E)$ be an epistemic ensemble signature. An* epistemic ensemble transition system (EETS) *over $\Sigma$ is a tuple $M = (S, S_0, T, \Omega)$ such that*

- *$S$ is a set of* ensemble states *and $S_0 \subseteq S$ is the set of* initial ensemble states*,*
- *$T = (T_e \subseteq S \times S)_{e \in E}$ is an $E$-indexed family of* transition relations *$T_e$, and*
- *$\Omega \colon S \to \mathit{EpiSt}(P, A)$ is an* epistemic state operator

*such that the following two conditions are satisfied:*

1. *For all $s \in S$ and $e \in E$, if there exists $s' \in S$ with $(s, s') \in T_e$, then*
   *(a) there exist $\mathfrak{K}, \mathfrak{K}' \in \mathit{EpiSt}(P, A)$ such that $\Omega(s) \approx \mathfrak{K}$, $\Omega(s') \approx \mathfrak{K}'$, and $(\mathfrak{K}, \mathfrak{K}') \in \mathit{eeff}(e)$,*
   *(b) for any $(\mathfrak{K}, \mathfrak{K}'') \in \mathit{eeff}(e)$ there exists $(s, s'') \in T_e$ with $\Omega(s) \approx \mathfrak{K}$ and $\Omega(s'') \approx \mathfrak{K}''$.*
2. *For all $s \in S$ there are $s_0 \in S_0$, $e_1, \ldots, e_n \in E$ ($n \geq 0$) and $(s_i, s_{i+1}) \in T_{e_i}$ for $0 \leq i < n$ such that $s_n = s$.*

*The class of epistemic ensemble transition systems over $\Sigma$ is denoted by $\mathit{Str}(\Sigma)$.* □

We write $s \xrightarrow{e}_M s'$ for $(s, s') \in T_e$. This relation is extended to compound epistemic ensemble actions $\pi \in \mathcal{E}_\Sigma$ by the following inductive definition:

$$s \xrightarrow{\varphi?}_M s' \iff \Omega(s) \models \varphi \text{ and } s = s'$$

$$s \xrightarrow{\pi_1 + \pi_2}_M s' \iff s \xrightarrow{\pi_1}_M s' \text{ or } s \xrightarrow{\pi_2}_M s'$$

$$s \xrightarrow{\pi_1 ; \pi_2}_M s' \iff \text{there exists } s_1 \text{ with } s \xrightarrow{\pi_1}_M s_1 \text{ and } s_1 \xrightarrow{\pi_2}_M s'$$

$$s \xrightarrow{\pi^*}_M s' \iff \text{there exist } n \geq 0, s = s_0, s_1, \ldots, s_{n-1}, s_n = s' \text{ with } s_i \xrightarrow{\pi}_M s_{i+1} \text{ for all } 0 \leq i < n$$

For any epistemic ensemble signature $\Sigma$, the *satisfaction* of an epistemic ensemble formula $\psi \in \Psi_\Sigma$ by an EETS $M = (S, S_0, T, \Omega)$ over $\Sigma$ at a state $s \in S$, written $M, s \models_\Sigma \psi$, is inductively defined as follows:

$$M, s \models_\Sigma \varphi \iff \Omega(s) \models \varphi$$

$$M, s \models_\Sigma \neg\psi \iff \text{not } M, s \models_\Sigma \psi$$

$$M, s \models_\Sigma \psi_1 \vee \psi_2 \iff M, s \models_\Sigma \psi_1 \text{ or } M, s \models_\Sigma \psi_2$$

$$M, s \models_\Sigma \langle\pi\rangle\psi \iff \text{there exists } s' \in S \text{ with } s \xrightarrow{\pi}_M s' \text{ such that } M, s' \models_\Sigma \psi$$

$M$ *satisfies* an epistemic ensemble formula $\psi \in \Psi_\Sigma$, written $M \models_\Sigma \psi$, if $M, s_0 \models_\Sigma \psi$ for all initial states $s_0 \in S_0$.

For the box, $M, s \models_\Sigma [\pi]\psi$ means that whenever $\pi$ is executed by the ensemble a state $s'$ is reached in which $\psi$ holds. Note that, if $\pi = e$ is an atomic ensemble action such that the precondition $pre(e)$ does not hold in $\Omega(s)$, then $M, s \models_\Sigma [e]\psi$ holds since there is no execution of $e$ in state $s$.

*Example 5.* A connection to public announcement logic [3] can be drawn as follows: Consider the ensemble signature $\Sigma = (P, A, E)$ with an arbitrary epistemic signature $(P, A)$ and $E$ consisting of all public announcements of the form $pub^a(\mathsf{K}_a\,\varphi)$ with $a \in A$. As semantic model we take the special EETS $M_{\mathrm{PAL}} = (\mathit{EpiSt}(P, A), \mathit{EpiSt}(P, A), T, \Omega)$ where the ensemble states are just the epistemic states over $(P, A)$, all states are initial, $T = (T_{pub^a(\mathsf{K}_a\,\varphi)} \subseteq \mathit{EpiSt}(P, A) \times \mathit{EpiSt}(P, A))_{pub^a(\mathsf{K}_a\,\varphi) \in E}$ with $T_{pub^a(\mathsf{K}_a\,\varphi)} = \mathit{eeff}(pub^a(\mathsf{K}_a\,\varphi))$ are the semantic transitions for public announcements, and $\Omega$ is the identity. Then, for any ensemble state $s$ of $M_{\mathrm{PAL}}$, i.e. epistemic state $(K, w) \in \mathit{EpiSt}(P, A)$, and any epistemic ensemble formula $\psi \in \Psi_\Sigma$ we have $M_{\mathrm{PAL}}, (K, w) \models \psi$ if, and only if, $(K, w)$ satisfies $\psi$ in the sense of public announcement logic. □

More generally, dynamic epistemic logic with arbitrary epistemic actions $(U, q)$ such that $pre(q)$ has the form $\mathsf{K}_a\,\varphi$ and $o(U, q) = a \in A$ can be similarly interpreted by an EETS. Note, however, that in these cases no control information can be captured since ensemble states are just epistemic states. Therefore instead of stating requirements for ensemble behaviours these logics are more appropriate for the verification of pre- and postconditions of programs represented by compound ensemble actions where ensemble formulas have the shape $\mathrm{pre} \to [\pi]\mathrm{post}$.

**Definition 3 (Semantics of epistemic ensemble specifications and refinement).** *Let $Sp = (\Sigma, Ax)$ be an epistemic ensemble specification. A* model *of $Sp$ is an EETS over $\Sigma$ which satisfies all axioms of $Ax$. The* semantics *of $Sp$ is given by its* model class

$$\mathrm{Mod}(Sp) = \{M \in \mathit{Str}(\Sigma) \mid M \models \psi \textit{ for all } \psi \in Ax\}\,.$$

*An epistemic ensemble specification $Sp' = (\Sigma, Ax')$ is a* refinement *of $Sp$ if* $\mathrm{Mod}(Sp') \subseteq \mathrm{Mod}(Sp)$. □

As an equivalence for epistemic ensemble transition systems we use *EETS-bisimulation* which is defined as expected.

**Definition 4 (Epistemic ensemble bisimulation).** *Let $\Sigma = (P, A, E)$ be an epistemic ensemble signature and $M_1 = (S_1, S_{1,0}, T_1, \Omega_1)$ and $M_2 = (S_2, S_{2,0}, T_2, \Omega_2)$ be two EETSs over $\Sigma$. An* EETS-bisimulation *between $M_1$ and $M_2$ is a relation $EB \subseteq S_1 \times S_2$ such that for all $(s_1, s_2) \in EB$ and all $e \in E$ the following holds:*

1. *$\Omega_1(s_1) \approx \Omega_2(s_2)$,*
2. *for each $s_1' \in S_1$, if $s_1 \xrightarrow{e}_{M_1} s_1'$ then there is an $s_2' \in S_2$ such that $s_2 \xrightarrow{e}_{M_2} s_2'$ and $(s_1', s_2') \in EB$, and*
3. *for each $s_2' \in S_2$, if $s_2 \xrightarrow{e}_{M_2} s_2'$ then there is an $s_1' \in S_1$ such that $s_1 \xrightarrow{e}_{M_1} s_1'$ and $(s_1', s_2') \in EB$.*

*$M_1$ and $M_2$ are* EETS-bisimilar, *written $M_1 \sim M_2$, if there exists an EETS-bisimulation $EB$ between $M_1$ and $M_2$ such that for each $s_1 \in S_{1,0}$ there exists an $s_2 \in S_{2,0}$ with $(s_1, s_2) \in EB$ and, conversely, for each $s_2 \in S_{2,0}$ there exists an $s_1 \in S_{1,0}$ with $(s_1, s_2) \in EB$.* □

It is easy to prove, by induction on the form of compound ensemble actions, that conditions (2) and (3) above can be propagated to compound ensemble actions $\pi \in \mathcal{E}_\Sigma$. As a consequence, it is straightforward to prove, by induction on the form of epistemic ensemble formulæ, that satisfaction is invariant under EETS-bisimulation. The base case follows from Lem. 1. The converse of the theorem is also valid for image-finite EETS.

**Theorem 1 (Invariance of epistemic ensemble formulæ).** *Let $M_1$ and $M_2$ be EETS over the same epistemic ensemble signature $\Sigma$ such that $M_1 \sim M_2$. Then, for any $\psi \in \Psi_\Sigma$, $M_1 \models \psi$ if, and only if, $M_2 \models \psi$.*

## 5 Epistemic Ensemble Realisations

Ensemble specifications describe requirements for systems of collaborating entities from a global point of view. For the realisation of ensembles we must take a local view and define a single behaviour for each agent. For this purpose, we introduce an *epistemic process language* over an epistemic ensemble signature $\Sigma = (P, A, E)$ which allows us to describe the local behaviour of each agent $a \in A$ as a sequential process $P_a$ in accordance with the following grammar:

$$P_a ::= \mathbf{0} \mid e_a.P_a \mid \varphi_a \supset P_a \mid P_{a,1} + P_{a,2} \mid \mu X \,.\, P_a \mid X$$

where $\mathbf{0}$ represents the inactive process, $e_a.P_a$ prefixes $P_a$ with an agent action $e_a \in E$, $\varphi_a \supset P_a$ is a guarded process, $P_{a,1} + P_{a,2}$ denotes the non-deterministic choice between processes, $\mu X \,.\, P_a$ models recursion, and $X$ is a process variable.

The following constraints apply to the syntax of processes: First, in a prefix $e_a.P_a$ the owner of $e_a$ must be $a$, i.e. $o(e_a) = a$. Secondly, each agent $a$, or, more precisely, its process, shall only use guards concerning the agent's own knowledge. We thus require $\varphi_a \in \Phi^a_{P,A}$; see Sect. 2.1. A similar constraint is applied to epistemic programs in [7].

**Definition 5 (Epistemic ensemble realisation).** *For an epistemic ensemble signature $\Sigma = (P, A, E)$, an* epistemic ensemble realisation *over $\Sigma$ is a pair $Real = (\{P_{0,a} \mid a \in A\}, \mathfrak{K}_0)$ where $\{P_{0,a} \mid a \in A\}$ is a set of sequential processes over $\Sigma$, one for each agent $a \in A$, and $\mathfrak{K}_0 \in EpiSt(P, A)$ is an initial epistemic state of the ensemble.* □

The semantics of an epistemic ensemble realisation is given in terms of en epistemic ensemble transition system. In this case the ensemble states are pairs $s = (ctrl, \mathfrak{K})$ consisting of a global control state $ctrl$ and an epistemic state $\mathfrak{K} \in EpiSt(P, A)$ capturing the current epistemic information of the ensemble. The control state $ctrl$ holds the current (local) execution state of each agent represented by a process expression. Thus $ctrl$ is a mapping that attaches to each $a \in A$ a sequential process $ctrl(a) = P_a$. When an agent $a$ moves from one state $P_a$ to another state $P'_a$ the control state $ctrl$ must be updated accordingly which is denoted by $ctrl[a \mapsto P'_a]$.

In contrast to the loose semantics of ensemble specifications, an ensemble realisation $\mathit{Real} = (\{P_{0,a} \mid a \in A\}, \mathfrak{K}_0)$ determines a unique epistemic ensemble transition system. It has a single initial ensemble state $s_0 = (\mathit{ctrl}_0, \mathfrak{K}_0)$ where the control state $\mathit{ctrl}_0$ assigns to each agent $a$ its process definition $P_{0,a}$, i.e. $\mathit{ctrl}_0(a) = P_{0,a}$ for all $a \in A$. Then, starting in $s_0$, an epistemic ensemble transition system is generated by the structural operational semantics rules in Fig. 1. For each ensemble state $s = (\mathit{ctrl}, \mathfrak{K})$ of the system the epistemic state operator is defined by $\Omega(\mathit{ctrl}, \mathfrak{K}) = \mathfrak{K}$.

The first five rules, from (action prefix) to (recursion), describe how single processes evolve in the context of an epistemic state which (i) may change when the process performs an agent action and (ii) is used for the evaluation of guards. We use the symbol "$\hookrightarrow$" for transitions on the process level. Transitions on the ensemble level are denoted by "$\rightarrow$". Rule (ensemble) says that whenever a single agent process moves from a local process state $P_a$ to state $P'_a$ changing the epistemic state from $\mathfrak{K}$ to $\mathfrak{K}'$ the whole ensemble evolves accordingly.

(action prefix)
$$\frac{}{(e_a.P_a, \mathfrak{K}) \xhookrightarrow{e_a} (P_a, \mathfrak{K}')} \text{ if } (\mathfrak{K}, \mathfrak{K}') \in \mathit{eeff}(e_a)$$

(guard)
$$\frac{(P_a, \mathfrak{K}) \xhookrightarrow{e_a} (P'_a, \mathfrak{K}')}{(\varphi_a \supset P_a, \mathfrak{K}) \xhookrightarrow{e_a} (P'_a, \mathfrak{K}')} \text{ if } \mathfrak{K} \models \varphi_a$$

(choice-left)
$$\frac{(P_{a,1}, \mathfrak{K}) \xhookrightarrow{e_a} (P'_{a,1}, \mathfrak{K}')}{(P_{a,1} + P_{a,2}, \mathfrak{K}) \xhookrightarrow{e_a} (P'_{a,1}, \mathfrak{K}')}$$

(choice-right)
$$\frac{(P_{a,2}, \mathfrak{K}) \xhookrightarrow{e_a} (P'_{a,2}, \mathfrak{K}')}{(P_{a,1} + P_{a,2}, \mathfrak{K}) \xhookrightarrow{e_a} (P'_{a,2}, \mathfrak{K}')}$$

(recursion)
$$\frac{(P_a\{X \mapsto \mu X \,.\, P_a\}, \mathfrak{K}) \xhookrightarrow{e_a} (P'_a, \mathfrak{K}')}{(\mu X \,.\, P_a, \mathfrak{K}) \xhookrightarrow{e_a} (P'_a, \mathfrak{K}')}$$

(ensemble)
$$\frac{(P_a, \mathfrak{K}) \xhookrightarrow{e_a} (P'_a, \mathfrak{K}')}{(\mathit{ctrl}, \mathfrak{K}) \xrightarrow{e_a} (\mathit{ctrl}[a \mapsto P'_a], \mathfrak{K}')} \text{ if } \mathit{ctrl}(a) = P_a$$

**Fig. 1.** SOS rules for epistemic processes and ensemble realisations

**Definition 6 (Semantics of an epistemic ensemble realisation).** *The semantics of an epistemic ensemble realisation $\mathit{Real} = (\{P_{0,a} \mid a \in A\}, \mathfrak{K}_0)$ over an ensemble signature $\Sigma$ is the epistemic ensemble transition system*

$$[\![\mathit{Real}]\!] = (S, \{s_0\}, T, \Omega)$$

*over $\Sigma$ where the initial ensemble state $s_0$ and the state operator $\Omega$ are explained above and the states in $S$ and transitions in $T$ are inductively generated from $s_0$ by applying the rules in Fig. 1. Note that $[\![\mathit{Real}]\!]$ satisfies the conditions of an EETS in Def. 2.* □

Our semantic concepts lead to an obvious correctness notion concerning the realisation of epistemic ensemble specifications:

**Definition 7 (Correct ensemble realisation).** *Let Sp be an epistemic ensemble specification and let Real be a realisation over the same epistemic signature. Real is a* correct realisation *of Sp if* $[\![\mathit{Real}]\!] \in \mathrm{Mod}(\mathit{Sp})$. □

*Example 6.* We provide a realisation for our simple robot rescue ensemble with two agents V (victim) and R (rescuer). The realisation consists of the two processes

$$\begin{aligned} P_{0,\mathrm{V}} &= \mu X\,.\,\big((\mathsf{K}_{\mathrm{V}}\,\mathrm{h} \land \neg\mathsf{K}_{\mathrm{V}}\,\mathsf{K}_{\mathrm{R}}\,\mathrm{h} \supset \mathit{snd}_{\mathrm{los}}^{\mathrm{V}\to\mathrm{R}}(\mathsf{K}_{\mathrm{V}}\,\mathrm{h}).X) + \\ &\qquad\quad (\mathsf{K}_{\mathrm{V}}\,\mathsf{K}_{\mathrm{R}}\,\mathrm{h} \supset \mathit{stop}.\mathbf{0})\big) \\ P_{0,\mathrm{R}} &= \mathsf{K}_{\mathrm{R}}\,\mathrm{h} \supset \mathit{snd}_{\mathrm{rel}}^{\mathrm{R}\to\mathrm{V}}(\mathsf{K}_{\mathrm{R}}\,\mathrm{h}).\mathit{rescue}.\mathbf{0} \end{aligned}$$

For the initial epistemic state of the realisation we take $\mathfrak{K}_0 = (K_0, w_0)$ as depicted in Ex. 1. Thus the initial ensemble state is $s_0 = (\mathit{ctrl}_0, \mathfrak{K}_0)$ with $\mathit{ctrl}_0(\mathrm{V}) = P_{0,\mathrm{V}}$, $\mathit{ctrl}_0(\mathrm{R}) = P_{0,\mathrm{R}}$ and $\Omega(s_0) = \mathfrak{K}_0$. As long as the victim does not know that the rescuer knows that the victim needs help, the victim continues sending the information $\mathsf{K}_{\mathrm{V}}\,\mathrm{h}$ to the rescuer. Notice again that this sending is lossy and hence either successful or unsuccessful. Only when the rescuer became aware of the emergency it can send, in a reliable way, its knowledge to the victim who can then stop its activity.

The EETS generated from the ensemble realisation has infinitely many ensemble states since it is possible that an unsuccessful sending from V to R happens infinitely often and hence each time an update of the previous epistemic state is performed. One can show, however, that if an unsuccessful sending happens after an unsuccessful or successful sending then the resulting epistemic state is bisimilar to the previous one. Therefore, there exists a minimal finite EETS, shown in Fig. 2, which is EETS-bisimilar to the one generated by the ensemble realisation. The epistemic effect of lossy sending is non-deterministic. The transitions from ensemble state $s_0$ to $s_1$ and the loops on $s_1$ and $s_2$ represent unsuccessful transmissions and the transitions from $s_0$ and from $s_1$ to $s_2$ represent successful ones. The associated epistemic states $(K_1, (w_0, \mathsf{k}))$ and $(K_1, (w_0, \mathsf{n}))$ are shown in Ex. 3. The epistemic state $(K_2, ((w_0, \mathsf{k}), \mathsf{k}))$ associated with the ensemble states $s_3$ to $s_6$ is computed by updating $(K_1, (w_0, \mathsf{k}))$ with the (deterministic) epistemic effect of the reliable sending from R to V.

Obviously, the EETS in Fig. 2 satisfies the axioms of the specification $\mathit{Sp}_{vr}$ in Ex. 4. Therefore, according to Thm. 1, the bisimilar EETS generated from the epistemic ensemble realisation is a model of $\mathit{Sp}_{vr}$ and thus the realisation is correct w.r.t. $\mathit{Sp}_{vr}$. □

Two epistemic ensemble realisations $\mathit{Real}_1$ and $\mathit{Real}_2$ over the same signature are called *equivalent* if $[\![\mathit{Real}_1]\!] \sim [\![\mathit{Real}_2]\!]$. The following theorem says that for checking equivalence of epistemic ensemble realisations it is sufficient to show that their initial epistemic states are bisimilar and that the process definitions for each agent are pairwise bisimilar in the usual sense of process algebra; see e.g. [10]. We denote process bisimilarity by $\sim_p$.

**Theorem 2.** *Let* $\mathit{Real}_1 = (\{P^1_{0,a} \mid a \in A\}, \mathfrak{K}^1_0)$ *and* $\mathit{Real}_2 = (\{P^2_{0,a} \mid a \in A\}, \mathfrak{K}^2_0)$ *be two epistemic ensemble realisations over signature* $\Sigma$*. If* $\mathfrak{K}^1_0 \approx \mathfrak{K}^2_0$ *and* $P^1_{0,a} \sim_p P^2_{0,a}$ *for all* $a \in A$*, then* $[\![\mathit{Real}_1]\!] \sim [\![\mathit{Real}_2]\!]$.

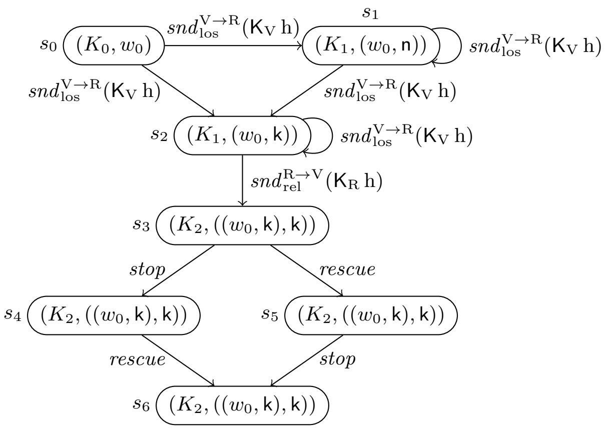

$s_1$
$s_0$ $(K_0, w_0)$
$snd^{\mathrm{V}\to\mathrm{R}}_{\mathrm{los}}(\mathsf{K}_{\mathrm{V}}\,\mathsf{h})$
$(K_1, (w_0, \mathsf{n}))$
$snd^{\mathrm{V}\to\mathrm{R}}_{\mathrm{los}}(\mathsf{K}_{\mathrm{V}}\,\mathsf{h})$
$snd^{\mathrm{V}\to\mathrm{R}}_{\mathrm{los}}(\mathsf{K}_{\mathrm{V}}\,\mathsf{h})$
$snd^{\mathrm{V}\to\mathrm{R}}_{\mathrm{los}}(\mathsf{K}_{\mathrm{V}}\,\mathsf{h})$
$s_2$ $(K_1, (w_0, \mathsf{k}))$
$snd^{\mathrm{V}\to\mathrm{R}}_{\mathrm{los}}(\mathsf{K}_{\mathrm{V}}\,\mathsf{h})$
$snd^{\mathrm{R}\to\mathrm{V}}_{\mathrm{rel}}(\mathsf{K}_{\mathrm{R}}\,\mathsf{h})$
$s_3$ $(K_2, ((w_0, \mathsf{k}), \mathsf{k}))$
*stop*
*rescue*
$s_4$ $(K_2, ((w_0, \mathsf{k}), \mathsf{k}))$
$s_5$ $(K_2, ((w_0, \mathsf{k}), \mathsf{k}))$
*rescue*
*stop*
$s_6$ $(K_2, ((w_0, \mathsf{k}), \mathsf{k}))$

**Fig. 2.** EETS for the victim rescue ensemble realisation

*Proof sketch.* Let $S_i$ be the ensemble states of $Real_i$ for $i = 1, 2$. We use the relation $EB \subseteq S_1 \times S_2$ such that $((ctrl_1, \mathfrak{K}_1), (ctrl_2, \mathfrak{K}_2)) \in EB$ iff $ctrl_1(a) \sim_p ctrl_2(a)$ for all $a \in A$ and $\mathfrak{K}_1 \approx \mathfrak{K}_2$. By assumption, the initial ensemble states are related by $EB$. We have to show that $EB$ is an EETS-bisimulation.

Condition (1) of Def. 4 is satisfied by definition of $EB$. For condition (2), let $((ctrl_1, \mathfrak{K}_1), (ctrl_2, \mathfrak{K}_2)) \in EB$ and $(ctrl_1, \mathfrak{K}_1) \xrightarrow{e}_{[\![Real_1]\!]} (ctrl_1', \mathfrak{K}_1')$. By rule (ensemble) in Fig. 1, there is $(P_a^1, \mathfrak{K}_1) \overset{e}{\hookrightarrow} (P_a^{1'}, \mathfrak{K}_1')$ where $P_a^1 = ctrl_1(a)$ and $P_a^{1'} = ctrl_1'(a)$. A case analysis on the form of $P_a^1$ yields that $P_a^1 \overset{e}{\hookrightarrow}_p P_a^{1'}$ and $(\mathfrak{K}_1, \mathfrak{K}_1') \in \mathit{eeff}(e)$ where $\overset{e}{\hookrightarrow}_p$ denotes process transition. Since $\mathfrak{K}_1 \approx \mathfrak{K}_2$, it follows from Lem. 3 that there is a $\mathfrak{K}_2'$ such that $(\mathfrak{K}_2, \mathfrak{K}_2') \in \mathit{eeff}(e)$ and $\mathfrak{K}_1' \approx \mathfrak{K}_2'$. Let $P_a^2 = ctrl_2(a)$. Then $P_a^1 \sim_p P_a^2$ and therefore there exists $P_a^2 \overset{e}{\hookrightarrow}_p P_a^{2'}$ with $P_a^{1'} \sim_p P_a^{2'}$. A case analysis on the form of $P_a^2$ yields that $(P_a^2, \mathfrak{K}_2) \overset{e}{\hookrightarrow} (P_a^{2'}, \mathfrak{K}_2')$ and hence, by rule (ensemble), that $(ctrl_2, \mathfrak{K}_2) \xrightarrow{e}_{[\![Real_2]\!]} (ctrl_2', \mathfrak{K}_2')$. Moreover, $((ctrl_1', \mathfrak{K}_1'), (ctrl_2', \mathfrak{K}_2')) \in EB$. □

## 6 Conclusion

We have developed a formalism for rigorous specification and realisation of ensembles based on principles of epistemic logic and epistemic actions. A crucial difference to [8,9,5] is that agents in epistemic ensembles do not communicate by message passing, but information exchange is achieved implicitly by changing knowledge. Another approach with implicit interaction is provided by the DEECo component and ensemble model [4]. In this case a coordinator is responsible for triggering exchange of factual knowledge which is, however, not grounded in epistemic logic.

For specifications of bigger case-studies we would need to extend our logic to allow agent types, variables and quantification over agents. For ensemble realisations we want to go a step further and represent the epistemic information, that is currently used by agent processes by accessing a global epistemic state, by local knowledge bases attached to each agent process.

## References

1. Baltag, A., Moss, L.S.: Logics for Epistemic Programs. Synth. **139**(2), 165–224 (2004). https://doi.org/10.1023/B:SYNT.0000024912.56773.5e
2. Baltag, A., Moss, L.S., Solecki, S.: The logic of public announcements and common knowledge and private suspicions. In: Gilboa, I. (ed.) Proc. 7th Conf. Theoretical Aspects of Rationality and Knowledge (TARK 1998). pp. 43–56. Morgan Kaufmann (1998)
3. Baltag, A., Renne, B.: Dynamic epistemic logic. In: Zalta, E.N., Nodelman, U., Allen, C., Anderson, R.L. (eds.) Stanford Encyclopedia of Philosophy. The Metaphysics Research Lab, Stanford University (2016)
4. Bures, T., Gerostathopoulos, I., Hnetynka, P., Keznikl, J., Kit, M., Plasil, F.: DEECO: an ensemble-based component system. In: Kruchten, P., Giannakopoulou, D., Tivoli, M. (eds.) Proc. 16th ACM SIGSOFT Symp. Component Based Software Engineering (CBSE 2013). pp. 81–90. ACM (2013). https://doi.org/10.1145/2465449.2465462
5. De Nicola, R., Loreti, M., Pugliese, R., Tiezzi, F.: A formal approach to autonomic systems programming: the SCEL language. ACM Transactions on Autonomous and Adaptive Systems (TAAS) **9**(2), 1–29 (2014). https://doi.org/10.1145/2619998
6. van Ditmarsch, H., van der Hoek, W., Kooi, B.: Dynamic Epistemic Logic, Synthese Library, vol. 337. Springer (2008)
7. Fagin, R., Halpern, J.Y., Moses, Y., Vardi, M.Y.: Reasoning About Knowledge. MIT Press (2003)
8. Hennicker, R., Wirsing, M.: Dynamic logic for ensembles. In: Margaria, T., Steffen, B. (eds.) Proc. 8th Intl. Symp. Leveraging Applications of Formal Methods, Verification and Validation. Distributed Systems (ISoLA 2018). Lect. Notes Comp. Sci., vol. 11246, pp. 32–47. Springer (2018). https://doi.org/10.1007/978-3-030-03424-5_3
9. Hennicker, R., Wirsing, M.: A dynamic logic for systems with predicate-based communication. In: Margaria, T., Steffen, B. (eds.) Proc. 9th Intl. Symp. Leveraging Applications of Formal Methods, Verification and Validation: Engineering Principles (ISoLA 2020). Lect. Notes Comp. Sci., vol. 12477, pp. 224–242. Springer (2020). https://doi.org/10.1007/978-3-030-61470-6_14
10. Milner, R.: Communication and concurrency. PHI Series in computer science, Prentice Hall (1989)
11. Pinciroli, C., Bonani, M., Mondada, F., Dorigo, M.: Adaptation and awareness in robot ensembles: Scenarios and algorithms. In: Wirsing et al. [14], pp. 471–494. https://doi.org/10.1007/978-3-319-16310-9
12. Sürmeli, J.: Epistemic logic in ensemble specification. In: Margaria, T., Steffen, B. (eds.) Proc. 9th Intl. Symp. Leveraging Applications of Formal Methods, Verification and Validation. Distributed Systems (ISoLA 2020), Part II. Lect. Notes Comp. Sci., vol. 12477, pp. 329–343. Springer (2020). https://doi.org/10.1007/978-3-030-61470-6_20
13. Wirsing, M., Banâtre, J., Hölzl, M.M., Rauschmayer, A. (eds.): Software-Intensive Systems and New Computing Paradigms — Challenges and Visions, Lect. Notes Comp. Sci., vol. 5380. Springer (2008). https://doi.org/10.1007/978-3-540-89437-7
14. Wirsing, M., Hölzl, M.M., Koch, N., Mayer, P. (eds.): Software Engineering for Collective Autonomic Systems - The ASCENS Approach, Lect. Notes Comp. Sci., vol. 8998. Springer (2015). https://doi.org/10.1007/978-3-319-16310-9
15. Wirsing, M., Hölzl, M.M., Tribastone, M., Zambonelli, F.: ASCENS: engineering autonomic service-component ensembles. In: Rev. Sel. Papers 10th Intl. Symp. Formal Methods for Components and Objects (FMCO 2011). pp. 1–24 (2011). https://doi.org/10.1007/978-3-642-35887-6_1