The Epistemic $\mu$-calculus

Cătălin Dima (LACL)

LACL, U-PEC

Based on joint (and ongoing) work with R. Bozianu (LACL) and C. Enea (LIAFA)

1. Work partially supported by the ANR research project “EQINOCS” no. ANR-11-BS02-0004
A classical picture in the background

- LTL = FOL1S $\not\subseteq$ S1S.
- CTL $\not\subseteq$ CTL* $\not\subseteq$ SnS.
- S2S = (binary) tree automata = turn-based 2-player games.
- MSO/bisimulation = $\mu$-calculus (on trees).
- ATL $\not\subseteq$ ATL* $\not\subseteq$ modal $\mu$-calculus.

What about the temporal epistemic framework?
The $\mu$-calculus of knowledge

Syntax:

$$\varphi ::= p \mid \varphi \land \varphi \mid \neg \varphi \mid AX \varphi \mid Ka \varphi \mid \mu Z. \varphi$$

where $Z \in Z$, $a \in Ag$, $p \in \Pi = \bigcup_{a \in Ag} \Pi_a$.

Synchronous & perfect recall semantics in terms of trees $t : \mathbb{N}^* \rightarrow \Pi$,

$$\| \cdot \| : Form(Z_1, \ldots, Z_n) \rightarrow \left[(2^{\supp(t)})^n \rightarrow 2^{\supp(t)}\right]$$

- $\|AX.\phi\|(S_1, \ldots, S_n) = AX(\|\phi\|(S_1, \ldots, S_n))$ where
  $$AX(S) = \{x \in \supp(t) \mid \forall i \in \mathbb{N} \text{ if } x_i \in \supp(t) \text{ then } x_i \in S\}$$

- $\|Ka.\phi\|(S_1, \ldots, S_n) = Ka(\|\phi\|(S_1, \ldots, S_n))$ where
  $$Ka(S) = \{x \in \supp(t) \mid \forall y \in \supp(t), \text{ if } x \sim_a y \text{ then } y \in S\}$$
  where $x \sim_a y$ if $\forall z < x, z' < y, |z| = |z'|$ implies $t(z) \cap \Pi = t(z') \cap \Pi$. 
The $\mu$-calculus of knowledge

Syntax:

\[ \varphi ::= p \mid \varphi \land \varphi \mid \neg \varphi \mid AX\varphi \mid Ka\varphi \mid \mu Z.\varphi \]

where $Z \in Z, a \in Ag, p \in \Pi = \bigcup_{a \in Ag} \Pi_a$.

Synchronous & perfect recall semantics in terms of trees $t : \mathbb{N}^* \rightarrow \Pi$,

\[ \| \cdot \| : Form(Z_1, \ldots, Z_n) \rightarrow \left[ (2^{supp(t)})^n \rightarrow 2^{supp(t)} \right] \]

- $\| AX.\varphi \|(S_1, \ldots, S_n) = AX(\| \varphi \|(S_1, \ldots, S_n))$ where

  \[ AX(S) = \{ x \in supp(t) \mid \forall i \in \mathbb{N} \text{ if } x_i \in supp(t) \text{ then } x_i \in S \} \]

- $\| Ka.\varphi \|(S_1, \ldots, S_n) = Ka(\| \varphi \|(S_1, \ldots, S_n))$ where

  \[ Ka(S) = \{ x \in supp(t) \mid \forall y \in supp(t), \text{ if } x \sim_a y \text{ then } y \in S \} \]

where $x \sim_a y$ if $\forall z < x, z' < y, |z| = |z'|$ implies $t(z) \cap \Pi = t(z') \cap \Pi$. 

Dima (LACL, U-PEC)
The $\mu$-calculus of knowledge

Syntax:

$$\varphi ::= p \mid \varphi \land \varphi \mid \neg \varphi \mid AX\varphi \mid Ka\varphi \mid \mu Z.\varphi$$

where $Z \in \mathcal{Z}$, $a \in Ag$, $p \in \Pi = \bigcup_{a \in Ag} \Pi_a$.

Synchronous & perfect recall semantics in terms of trees $t : \mathbb{N}^* \rightarrow \Pi$,

$$\Vert \cdot \Vert : Form(Z_1, \ldots, Z_n) \rightarrow \left[ \left( 2^{\text{supp}(t)} \right)^n \rightarrow 2^{\text{supp}(t)} \right]$$

- $\Vert AX.\phi \Vert(S_1, \ldots, S_n) = AX(\Vert \phi \Vert(S_1, \ldots, S_n))$ where
  
  $$AX(S) = \{ x \in \text{supp}(t) \mid \forall i \in \mathbb{N} \text{ if } xi \in \text{supp}(t) \text{ then } xi \in S \}$$

- $\Vert Ka.\phi \Vert(S_1, \ldots, S_n) = Ka(\Vert \phi \Vert(S_1, \ldots, S_n))$ where
  
  $$Ka(S) = \{ x \in \text{supp}(t) \mid \forall y \in \text{supp}(t), \text{ if } x \sim_a y \text{ then } y \in S \}$$

where $x \sim_a y$ if $\forall z < x, z' < y, |z| = |z'|$ implies $t(z) \cap \Pi = t(z') \cap \Pi$. 

Dima (LACL, U-PEC)
The $\mu$-calculus of knowledge

Syntax:

$$\varphi ::= p \mid \varphi \land \varphi \mid \neg \varphi \mid AX\varphi \mid Ka\varphi \mid \mu Z. \varphi$$

where $Z \in \mathcal{Z}$, $a \in Ag$, $p \in \Pi = \bigcup_{a \in Ag} \Pi_a$.

Synchronous & perfect recall semantics in terms of trees $t : \mathbb{N}^* \rightarrow \Pi$,

$$\| \cdot \| : Form(Z_1, \ldots, Z_n) \rightarrow \left[ (2^{\text{supp}(t)})^n \rightarrow 2^{\text{supp}(t)} \right]$$

- $\| AX.\varphi \|(S_1, \ldots, S_n) = AX(\| \varphi \|(S_1, \ldots, S_n))$ where
  $$AX(S) = \{ x \in \text{supp}(t) \mid \forall i \in \mathbb{N} \text{ if } x_i \in \text{supp}(t) \text{ then } x_i \in S \}$$

- $\| Ka.\varphi \|(S_1, \ldots, S_n) = Ka(\| \varphi \|(S_1, \ldots, S_n))$ where
  $$Ka(S) = \{ x \in \text{supp}(t) \mid \forall y \in \text{supp}(t), \text{ if } x \sim_a y \text{ then } y \in S \}$$
  where $x \sim_a y$ if $\forall z < x, z' < y, |z| = |z'|$ implies $t(z) \cap \Pi = t(z') \cap \Pi$. 

Dima (LACL, U-PEC)
The modal $\mu$-calculus of knowledge

Syntax:

\[ \varphi ::= p \mid \varphi \land \varphi \mid \neg \varphi \mid \langle \overline{c} \rangle \varphi \mid K_a \varphi \mid \mu Z . \varphi \]

where $Z \in Z$, $a \in Ag$, $p \in \Pi = \bigcup_{a \in Ag} \Pi_a$ and $\overline{c} \in Act = \bigtimes_{a \in Ag} Act_a$.

Synchronous & perfect recall semantics in terms of trees $t : \mathbb{N}^* \to \Pi \times Act$,

\[ \| \cdot \| : Form \to \left[ (2^{supp(t)})^n \to 2^{supp(t)} \right] \]

- $\| \langle \overline{c} \rangle . \phi \| (S_1, \ldots, S_n) = \langle \overline{c} \rangle (\| \phi \| (S_1, \ldots, S_n))$ where

  \[ \langle \overline{c} \rangle (S) = \{ x \in supp(t) \mid \forall i \in \mathbb{N} \text{ if } xi \in supp(t) \text{ and } t \big|_{Act} (xi) = \overline{c} \text{ then } xi \in S \} \]

- $\| K_a . \phi \| (S_1, \ldots, S_n) = K_a (\| \phi \| (S_1, \ldots, S_n))$ where

  \[ K_a (S) = \{ x \in supp(t) \mid \forall y \in supp(t), \text{ if } x \sim_a y \text{ then } y \in S \} \]

  where $x \sim_a y$ if $\forall z < x, z' < y$, $|z| = |z'|$ implies $t \big|_{\Pi} (z) \cap \Pi = t \big|_{\Pi} (z') \cap \Pi$ and $t \big|_{Act_a} (z) = t \big|_{Act_a} (z')$. 

Dima (LACL, U-PEC)
Issues on the expressivity of $K\mu$

- Common knowledge:
  \[ C_{a,b}\phi = \nu Z (\phi \land K_aZ \land K_bZ) \]

- $KB_n$ through the usual fixpoint definition:
  \[ ApU q = \mu Z . q \lor (p \land A \circ Z) \]

- ATL with perfect information:
  \[ \langle A \rangle \lozenge p = \mu Z . (p \lor \bigvee_{cA \in Act_A} \bigwedge_{cA^\tilde{A} \in Act_A^\tilde{A}} [c_A, c_A^\tilde{A}]Z) \]
Issues on the expressivity of $K\mu$

- Common knowledge:
  \[ C_{a,b}\phi = \nu Z. (\phi \land K_aZ \land K_bZ) \]

- $KB_n$ through the usual fixpoint definition:
  \[ ApU q = \mu Z.q \lor (p \land A \lor Z) \]

- ATL with perfect information:
  \[ \langle A \rangle \diamond p = \mu Z. (p \lor \bigvee_{c_A \in Act_A} \bigwedge_{c^-_A \in Act^-_A} [c_A, c^-_A]Z) \]
Expressing winning strategies in 2-player games with the proponent having imperfect observability:
\[ \nu Z_n \mu Z_{n-1} \ldots \mu Z_1. \bigvee_{\alpha \in Act_0} K_a \bigvee_{k \leq n} (p_k \land \bigwedge_{\beta \in Act_1} [\alpha, \beta] Z_k) \]
Expressivity of $K_\mu$?

- ATL with perfect information:
  \[\langle A \rangle \Box p = \mu Z \left( p \lor \bigvee_{c_A \in Act_A} \bigwedge_{\overrightarrow{c} \in Act_A} [c_A, \overrightarrow{c}] Z \right)\]

- ATL with imperfect information?
Expressivity of $K\mu$?

- ATL with perfect information:

$$\langle A \rangle \diamond p = \mu Z. (p \lor \bigvee_{c_A \in Act_A} \bigwedge_{c_{\overline{A}} \in Act_{\overline{A}}} [c_A, c_{\overline{A}}] Z)$$

- ATL with imperfect information?
Expressivity of $K\mu$?

- ATL with perfect information:
  \[ \langle A \rangle \lozenge p = \mu Z. \left( p \lor \bigvee_{c_A \in Act_A} \bigwedge_{\overline{c}_A \in Act_{\overline{A}}} [c_A, \overline{c}_A]Z \right) \]

- ATL with imperfect information?
  - Let’s try:
    \[ \langle A \rangle \lozenge p = \mu Z.K_A \left( p \lor \bigvee_{c_A \in Act_A} \bigwedge_{\overline{c}_A \in Act_{\overline{A}}} [c_A, \overline{c}_A]Z \right) \]
Expressivity of $K\mu$?

- ATL with perfect information:
  $$\langle A \rangle \lozenge p = \mu Z. (p \lor \bigvee_{c_A \in \text{Act}_A} \bigwedge_{c_{\bar{A}} \in \text{Act}_{\bar{A}}} [c_A, c_{\bar{A}}]Z)$$

- ATL with imperfect information?
  - Let's try:
    $$\langle A \rangle \lozenge p = \mu Z.K_A(p \lor \bigvee_{c_A \in \text{Act}_A} \bigwedge_{c_{\bar{A}} \in \text{Act}_{\bar{A}}} [c_A, c_{\bar{A}}]Z)$$

- Non-feasible strategies!
Expressivity of $K\mu$?

- ATL with perfect information:

$$\langle A \rangle \Box p = \mu Z. (p \lor \bigvee_{c_A \in Act_A} \bigwedge_{c_{\bar{A}} \in Act_{\bar{A}}} [c_A, c_{\bar{A}}]Z)$$

- ATL with imperfect information?
  - Let's try:

$$\langle A \rangle \Box p = K_A \mu Z. (p \lor \bigvee_{c_A \in Act_A} K_A \bigwedge_{c_{\bar{A}} \in Act_{\bar{A}}} [c_A, c_{\bar{A}}]Z)$$
Expressivity of $K\mu$?

ATL with perfect information:

$$\langle A \rangle \Diamond p = \mu Z.(p \lor \bigvee_{c_A \in Act_A} c_A \land \bigl[ c_A, c_{\overline{A}} \bigr] Z)$$

ATL with imperfect information?

- Let’s try:

$$\langle A \rangle \Diamond p = K_A \mu Z.(p \lor \bigvee_{c_A \in Act_A} K_A c_A \land \bigl[ c_A, c_{\overline{A}} \bigr] Z)$$

- With distributed knowledge!
Expressivity of $K\mu$?

- ATL with perfect information:

$$\langle A \rangle \diamond p = \mu Z. (p \lor \bigvee_{c_A \in Act_A} \bigwedge_{c_{\bar{A}} \in Act_{\bar{A}}} [c_A, c_{\bar{A}}] Z)$$

- ATL with imperfect information?
  - Let’s try:

$$\langle A \rangle \diamond p = K_A \mu Z. (p \lor \bigvee_{c_A \in Act_A} K_A \bigwedge_{c_{\bar{A}} \in Act_{\bar{A}}} [c_A, c_{\bar{A}}] Z)$$

$$\langle A \rangle \diamond p = \bigwedge_{a \in A} K_a \mu Z. (p \lor \bigvee_{c_A \in Act_A} \bigwedge_{a \in A} K_a \bigwedge_{c_{\bar{A}} \in Act_{\bar{A}}} [c_A, c_{\bar{A}}] Z)$$
Expressivity of $K\mu$?

- ATL with perfect information:
  \[
  \langle A \rangle \Box p = \mu Z. \left( p \lor \bigvee_{c_A \in Act_A} K_A \bigwedge_{c_A' \in Act_{\overline{A}}} [c_A, c_A'] Z \right)
  \]

- ATL with imperfect information?
  - Let's try:
    \[
    \langle A \rangle \Box p = K_A \mu Z. \left( p \lor \bigvee_{c_A \in Act_A} K_A \bigwedge_{c_A' \in Act_{\overline{A}}} [c_A, c_A'] Z \right)
    \]
    \[
    \langle A \rangle \Box p = \bigwedge_{a \in A} K_a \mu Z. \left( p \lor \bigvee_{c_A \in Act_A} \bigwedge_{a \in A} K_a \bigwedge_{c_A' \in Act_{\overline{A}}} [c_A, c_A'] Z \right)
    \]
  - Yeah, but both are too strong!
  - They require that the objective $p$ be attained at the same moment in each identically observable run!
Expressing single-agent coalition ATL in $K\mu$

- Given a tree model $t$, modify it by **guessing** the points $z$ where $p$ happened in the past of $z$.
- The guessing is encoded in the actions of the agent $a$, which may choose to force the system remember that $p$ has happened.
- Then $\langle A \rangle \diamond p$ is equivalent with:

$$\tilde{\phi} = \mu Z \cdot \bigvee_{\alpha \in \text{Act}_{a}} K_{a}(p \lor \text{past}_{p} \lor \bigwedge_{\beta \in \text{Act}_{\neg a} \setminus \{a\}} [\alpha, \beta]Z)$$

- Can be applied by structural induction on the formula.
- If the given tree has a *finite presentation* (regular tree), then the resulting tree also has a *finite presentation*.
Expressing single-agent coalition ATL in $K_{\mu}$

- Given a tree model $t$, modify it by **guessing** the points $z$ where $p$ happened in the past of $z$.
- The guessing is encoded in the actions of the agent $a$, which may choose to force the system remember that $p$ has happened.
- Then $\langle A \rangle \triangleleft p$ is equivalent with:

$$\tilde{\phi} = \mu Z \bigvee_{\alpha \in \text{Act}_a} K_a(p \lor \text{past}_p \lor \bigwedge_{\beta \in \text{Act}_A \setminus \{a\}} [\alpha, \beta]Z)$$

- Can be applied by structural induction on the formula.
- If the given tree has a *finite presentation* (regular tree), then the resulting tree also has a *finite presentation*. 
MSO with binary predicates

Syntax of $MSO_{idobs}$:

$$\phi ::= x \mid X \mid p(x) \mid x \in X \mid \phi \land \phi \mid \neg \phi \mid \forall x \phi \mid \forall X \phi \mid x \leq y \mid idobs_{\Sigma}(x, y)$$

where $x, y$ are individual variables, $X$ are monadic 2nd order predicates, $p \in \Pi$ and $\Sigma \subseteq \Pi$.

Usual tree semantics with bounded tree width.

$t, [x \mapsto x_0, y \mapsto y_0] \models idobs_{\Sigma}(x, y)$ if for all $x' \leq x$, $\forall y' \leq y$, $\forall p \in \Sigma$, if $|x'| = |y'|$ then $p(x')$ iff $p(y')$.

- Same $\Sigma$-history on the paths $\epsilon \mapsto x$ and $\epsilon \mapsto y$. 

Dima (LACL, U-PEC)
Expressing ATL formulas into $\text{MSO}_{idobs}$

- Uninterpreted atoms $= \Pi \cup \bigcup_{a \in Ag} \text{Act}_a$.
- Atoms in each $\text{Act}_a$ are exclusive.
- Strategy for player $a = 2$nd order variable $Y$.
  - At each position, all $Y$-successors are labeled with the same atom in $\text{Act}_a$.
  - At each position, if an $Y$-successor is labeled with $\alpha \in \text{Act}_a$, then all successors which bear an $\alpha$ belong to $Y$.
  - Uniform strategy = the same next action in $\text{Act}_a$ is chosen at positions having identically $a$-observable histories.
- LTL subformulas in the scope of an ATL (ATL*) strategy operator translated as usual.
- Strategies based on common knowledge can be expressed too.
  - Reflexive-transitive closure of $\text{idobs}_a \cup \text{idobs}_b$ can be expressed.
- Fully-uniform and strictly-uniform strategies can be expressed too.
A gap between $K_\mu$ and $\text{MSO}_{idobs}$?

- **Conjecture**: ATL and $K_\mu$ are incomparable.
- **Conjecture**: $\text{MSO}_{idobs} \supsetneq K_\mu$.

Single-agent $K_\mu$ has a decidable satisfiability problem.
- Reducible to a decidable subproblem of the model-checking problem for $K_\mu$ (see below).

$\text{MSO}_{eqlevel}$ has an undecidable satisfiability problem.

Automata techniques?
A gap between $K_\mu$ and $MSO_{idobs}$?

- **Conjecture**: ATL and $K_\mu$ are incomparable.
- **Conjecture**: $MSO_{idobs} \supsetneq K_\mu$.

- Single-agent $K_\mu$ has a decidable satisfiability problem.
  - Reducible to a decidable subproblem of the model-checking problem for $K_\mu$ (see below).
- $MSO_{eqlevel}$ has an undecidable satisfiability problem.

Automata techniques?
A gap between $K_{\mu}$ and $MSO_{idobs}$?

- **Conjecture**: ATL and $K_{\mu}$ are incomparable.
- **Conjecture**: $MSO_{idobs} \not\supseteq K_{\mu}$.

- Single-agent $K_{\mu}$ has a decidable satisfiability problem.
  - Reducible to a decidable subproblem of the model-checking problem for $K_{\mu}$ (see below).
- $MSO_{eqlevel}$ has an undecidable satisfiability problem.

Automata techniques?
Automata for LTLK

\[ A = (Q, \Pi, \Pi_a, \delta, \pi, \theta, Q_0, R). \]

- \( \delta \subseteq Q \times Q \).
- \( \theta \subseteq 2^Q \): the identical observability constraint.
- Subsets of initial states: \( Q_0 \subseteq 2^Q \).
- Büchi/Muller/etc. acceptance conditions.

Runs = \( Q \)-trees \( t : \mathbb{N}^* \to Q \)

- \( (t(x), t(x_i)) \in \delta \) for all \( x_i \in \text{supp}(t) \).
- \( \{t(x) \mid x \sim_a x_0\} \in \theta \), for all \( x_0 \in \text{supp}(t) \).
- Each infinite path in \( t \) satisfies \( R \).

Language = set of trees which are homomorphic images of runs under \( \pi : Q \to \Pi \).

- If \( t \) is accepted by \( A \) then any \( t' \) with \( \text{runs}(t') = \text{runs}(t) \) is accepted too.

Notion generalizable to \( n \) agents: \( (\theta_a)_{a \in Ag} \).
Automata for LTLK (2)

Example for $K_a p$:

- $\Pi_a = \emptyset$
- $\theta = \{\{1\}, \{2,3\}, \{3\}\}$

- $\forall t : \mathbb{N}^* \to Q$ run in $A$, for any position $x \in \text{supp}(t)$, $(\pi(t), x) \models K_a p$ iff $t(x) = 1$.
- Similarly, $(\pi(t), x) \models \neg K_a p$ iff $t(x) \in \{2,3\}$.
- Can be refined for larger $\Pi_a$. 

\[ 1 \quad p, K_a p \quad 2 \quad p, \neg K_a p \quad 3 \quad \neg p, \neg K_a p \]
Automata for LTLK (3)

- Closed under union.
- Synchronous product, modeling intersection.
- For any LTLK formula $\phi$ there exists $A_\phi$ accepting the same set of trees
  - $\Pi$-trees, with $\sim_a$ defined by $\Pi_a$ for each $a \in A_g$.

**Proposition (almost not a conjecture)**

Single-agent automata have a decidable emptiness problem.

Probable techniques:
- Solving a (synchronous) 2-player game with the proponent (player 0) having incomplete information.
- Constructing a single-agent $K\mu$ formula and testing its satisfiability.

Can be generalized to CTLK.
Model-checking $K_\mu$

- Finite models = multi-agent systems $M = (Q, Ag, \delta, q_0, \Pi, (\Pi a)_{a \in Ag}, \pi)$.
- $M \models \phi$ if the tree unfolding $t_M$ satisfies $\phi$, $\epsilon \in \|\phi\|(S_1, \ldots, S_n)$ for all $S_1, \ldots, S_n \subseteq \text{supp}(t_M)$.
- Model-checking is undecidable for the $\mu$-calculus of knowledge.
  - Subsumes $CTL_C$ (aka. $CL_n$ from Halpern & Vardi ’86), multi-agent $CTL_K$ with common knowledge.
Model-checking $K\mu$ (2)

- Decidable subproblem generalizing the need of a hierarchy of observations (Kupferman & Vardi, v.d. Meyden & Wilke & Engelhardt & Su, Finkbeiner & Schewe):

  $\phi$ mixes observations of $a$ and $b$ if $\exists$ subformula $\phi' = K_a \psi$ or $\phi' = P_a \psi$ with $\psi$ containing a free variable $Z$ and s.t. in $\psi$ an epistemic operator for $b$ is applied to a subformula in which the same $Z$ is free.

The non-mixing model checking problem:

Decide whether $t_M \models \phi$ for all instances in which any two agents $a, b$ which have mixed observations in $\phi$ have compatible observability in $M$.

- I.e. $\Pi_a \subseteq \Pi_b$ or $\Pi_b \subseteq \Pi_a$.

- An instance $(M, \phi)$ with $\phi = C_{a,b} p = \nu Z . (p \land K_a Z \land K_b Z)$ is non-mixing iff $a$ and $b$ have compatible observability in $M$.

- $K_a K_b \Box p$ is non-mixing for any $\Pi_a$ and $\Pi_b$.

- Subsumes known cases of decidable model-checking problems for LTLK/CTLK/ATL.
Technical approach for proving decidability of the non-mixing model-checking problem

Show that a finitary semantics suffices:

- State-based semantics: \([\bullet] : Form \rightarrow [(2^Q)^n \rightarrow 2^Q]\).
  
- Decidability of the non-epistemic \(\mu\)-calculus (with tree semantics):

\[
\begin{align*}
(2^Q)^n & \xrightarrow{\phi} 2^Q \\
(t_{M}^{-1})^n & \xrightarrow{\phi} t_{M}^{-1} \\
(2^{\text{supp}(t_{M})})^n & \xrightarrow{\phi} 2^{\text{supp}(t_{M})}
\end{align*}
\]

- Generalizable to the \(\mu\)-calculus of knowledge by including subset-refinements of \(M\).
- Subset construction w.r.t. \(a\) commutes with subset construction for agent \(b\) only if \(\Pi_a\) and \(\Pi_b\) are compatible (\(\subseteq\) or \(\supseteq\)).
Technical approach for proving decidability of the non-mixing model-checking problem

Show that a finitary semantics suffices:

- State-based semantics: \([\bullet] : Form \rightarrow [(2^Q)^n \rightarrow 2^Q]\).
- Decidability of the non-epistemic \(\mu\)-calculus (with tree semantics):

\[
\begin{align*}
(2^Q)^n & \xrightarrow{[\phi]} 2^Q \\
(t^{-1}_M)^n & \xrightarrow{\phi} t^{-1}_M \\
(2^{\text{supp}(t_M)})^n & \xrightarrow{\parallel \phi \parallel} 2^{\text{supp}(t_M)}
\end{align*}
\]

- Generalizable to the \(\mu\)-calculus of knowledge by including subset-refinements of \(M\).
- Subset construction w.r.t. \(a\) commutes with subset construction for agent \(b\) only if \(\Pi_a\) and \(\Pi_b\) are compatible (\(\subseteq\) or \(\supseteq\)).
Technical approach for proving decidability of the non-mixing model-checking problem

Show that a finitary semantics suffices:

- State-based semantics: \([ \bullet ] : \text{Form} \rightarrow [(2^Q)^n \rightarrow 2^Q] \).
- Decidability of the non-epistemic \(\mu\)-calculus (with tree semantics):

\[
\begin{align*}
(2^Q)^n & \xrightarrow{[\phi]} 2^Q \\
(t_M^{-1})^n & \downarrow \quad \downarrow t_M^{-1} \\
(2^{\text{supp}(t_M)})^n & \xrightarrow{\|\phi\|} 2^{\text{supp}(t_M)}
\end{align*}
\]

- Generalizable to the \(\mu\)-calculus of knowledge by including subset-refinements of \(M\).
- Subset construction w.r.t. \(a\) commutes with subset construction for agent \(b\) only if \(\Pi_a\) and \(\Pi_b\) are compatible (\(\subseteq\) or \(\supseteq\)).
Remarks and further work

- Maybe fixpoint variants of the ATL operators are better?
- Tree automata for the $\mu$-calculus of knowledge (work under progress).
- What if we replace $idobs$ predicates with 3rd order predicates?...
  - This would allow comparing sets of runs in a system.
- Automata for $K_{\mu}$ and $MSO$:
  - “Strict” tree versions, alternating generalizations.
  - Difference between $K_{\mu}$ and $MSO_{idobs}$ lies in the presence/absence of an extra constraint on labeling of nodes in a run with sets of states.