Strategic Reasoning under Imperfect Information with synchronous semantics

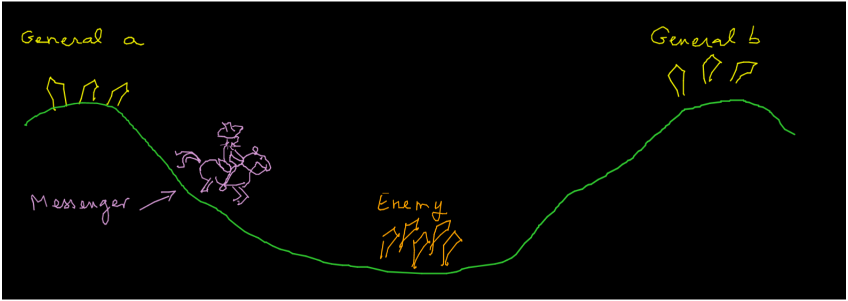
Sophie Pinchinat
IRISA, Université de Rennes

TIME 2024

1. Imperfect Information and Strategic Resoning
2. The Setting of Dynamic Epistemic Logic
3. Remarkable Propositional Action Models
4. Strategic Reasoning

Where is Imperfect Information?

Consider the coordinated attack problem

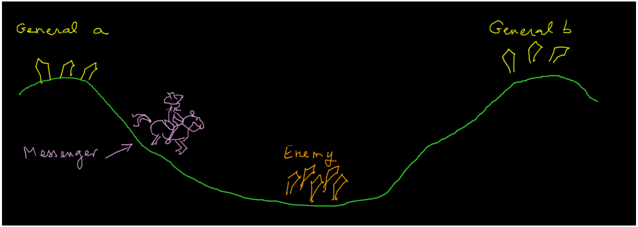


Famous problem in the distributed systems literature

[R. Fagin, J.Y. Halpern, Y. Moses, and M.Y. Vardi. Reasoning about Knowledge. MIT Press, 1995.]

Where is Imperfect Information?

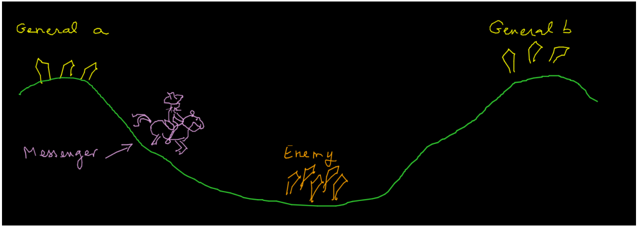
Consider the coordinated attack problem



- General a and Messenger are initially together

Where is Imperfect Information?

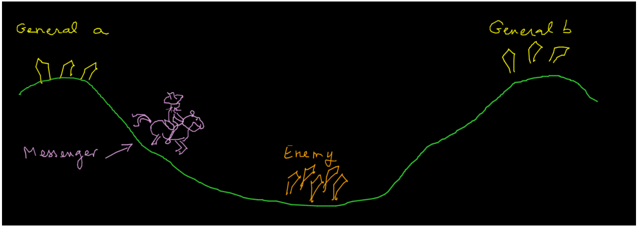
Consider the coordinated attack problem



- General a and Messenger are initially together
- General a has decided to attack at dawn, but General b does not have this information (yet)

Where is Imperfect Information?

Consider the coordinated attack problem

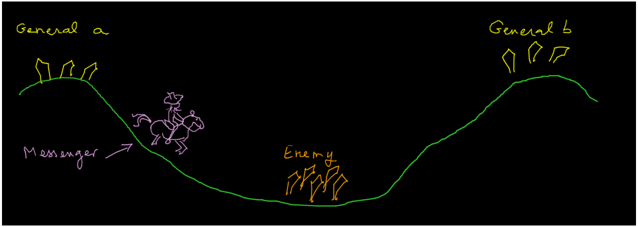


- General a and Messenger are initially together
- General a has decided to attack at dawn, but General b does not have this information (yet)

  $\Rightarrow$ General b has imperfect information about the initial situation

Where is Imperfect Information?

Consider the coordinated attack problem

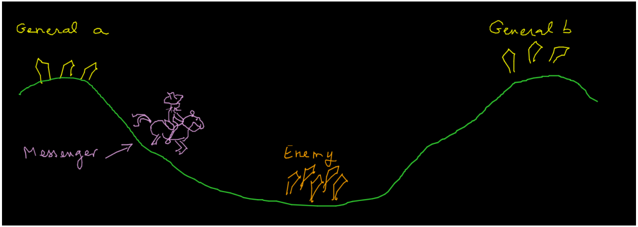


- General a and Messenger are initially together
- General a has decided to attack at dawn, but General b does not have this information (yet)

  $\Rightarrow$ General b has imperfect information about the initial situation
- General a sends Messenger to tell General b

Where is Imperfect Information?

Consider the coordinated attack problem



- General a and Messenger are initially together
- General a has decided to attack at dawn, but General b does not have this information (yet)

  ⇒ General b has imperfect information about the initial situation
- General a sends Messenger to tell General b

  ⇒ General a has imperfect information about the event taking place: “Messenger Arrived” or“Messenger caught”?

Observing and making decision under II

Knowledge is around...

- An agent can be uncertain about the actual situation (e.g. General b about the intention of General a to attack at dawn)

Uncertainty = indistinguishability of some situations

- After some event, uncertainty may

shrink $\searrow$ or grow $\nearrow$

Observing and making decision under II

Knowledge is around...

- An agent can be uncertain about the actual situation (e.g. General b about the intention of General a to attack at dawn)

Uncertainty = indistinguishability of some situations

- After some event, uncertainty may

shrink ↘ or grow ↗

Decision only depends on what is known

Observing and making decision under II

Knowledge is around...

- An agent can be uncertain about the actual situation (e.g. General b about the intention of General a to attack at dawn)

Uncertainty = indistinguishability of some situations

- After some event, uncertainty may

shrink ↘ or grow ↗

$\boxed{\text{Decision only depends on what is known}}$

$\Updownarrow$

$\boxed{\text{Same decisions in indistinguishable situations}}$

What is uncertain/known?

- Observation of the game positions

What is uncertain/known?

- Observation of the game positions
- Memory along plays/histories

What is uncertain/known?

- Observation of the game positions
- Memory along plays/histories

Formal settings:

- Observation in position = (some of) atomic facts

What is uncertain/known?

- Observation of the game positions
- Memory along plays/histories

Formal settings:

- Observation in position = (some of) atomic facts
- Remembering along histories = recall

What is uncertain/known?

- Observation of the game positions
- Memory along plays/histories

Formal settings:

- Observation in position = (some of) atomic facts
- Remembering along histories = recall
  - Perfect Recall: remember all they observed and actions they took.
  - Imperfect Recall: all the rest e.g. memoryless

Game Arenas (1/2)

Definition

Game arena $G = (Pos, Pos_I, Act, \delta, (\sim a)_{a \in Agt}, \lambda)$

- Pos positions, $Pos_I \subseteq Pos$
- Act actions
- $\delta : Pos \times Act \rightharpoonup Pos$ moves

  Plays $\subseteq (Pos_I.Act)^\omega$
  Histories $h \in Hist \subseteq Pos_I.(Act.Pos)^*$

- indistinguishability relation of agent $a \in Agt$

$$\sim_a \subseteq Pos \times Pos \cup (Act \cup \{\epsilon\}) \times (Act \cup \{\epsilon\})$$

- $\lambda : Pos \to 2^{AP}$ valuation function for atomic facts in positions

$$G = (Pos, Pos_I, Act, \delta, t, (\sim a)_{a \in Agt}, \lambda)$$

- Define $h \sim_a h'$...

$G = (Pos, Pos_I, Act, \delta, t, (\sim a)_{a \in Agt}, \lambda)$

- Define $h \sim_a h'$...

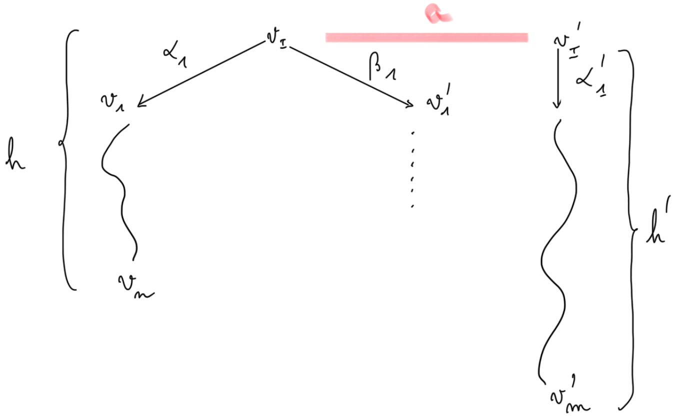
Knowledge

Agent a's uncertainty in actual situation h is $[h]_{\sim_a}$:

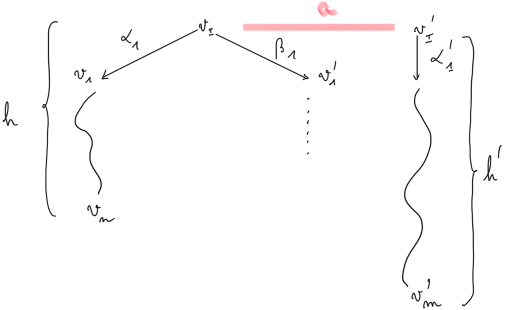
"Agent a knows property φ at history h

$$h \models K_a\varphi \text{ whenever } h' \models \varphi \text{ for every } h' \in [h]_{\sim_a}$$

Indistinguishability relations between histories



Observations/Indistinguishability/Knowledge



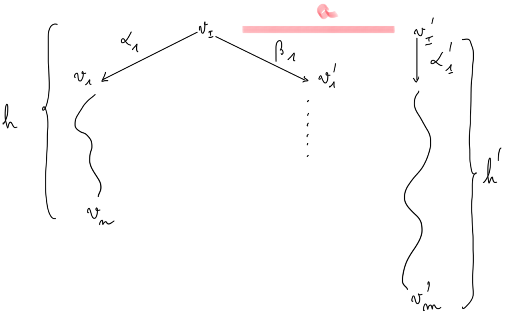
- Synchronous Perfect Recall

$$\begin{array}{ccccccc}
h = & v_0 & \alpha_1 & v_1 & \cdots & \alpha_n & v_n \\
\wr_a & \wr_a & \wr_a & \wr_a & \cdots & \wr_a & \wr_a \\
h' = & v_0' & \alpha_1' & v_1' & \cdots & \alpha_n' & v_n'
\end{array}$$

$h \sim_a h'$ implies $|h| = |h'|$

Agents hear ticks

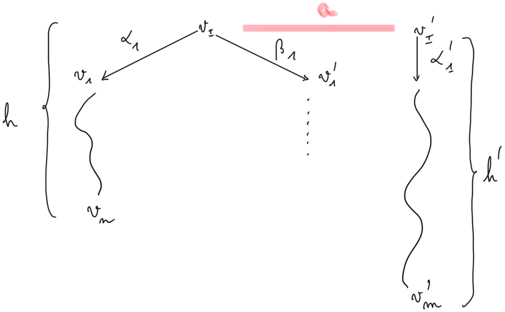
Observations/Indistinguishability/Knowledge



- Synchronous Imperfect Recall

  e.g. Memoryless: $h \sim_a h'$ when $\text{last}(h) \sim_a \text{last}(h')$ and $|h| = |h'|$.

Observations/Indistinguishability/Knowledge



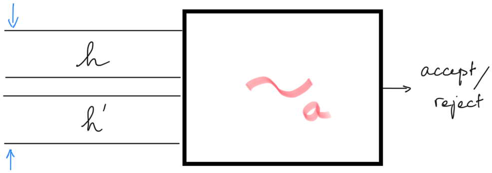
- Asynchronous Recall

  e.g. $Act = \{\alpha, \beta, \gamma\}$ with $\alpha \sim_a \epsilon$

$$\alpha\alpha\beta\alpha\gamma \sim_a \beta\gamma$$

Tools for handling indistinguishability

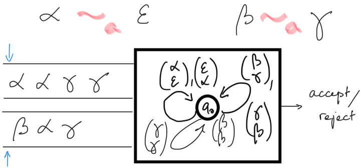
Transducers/2-tape finite-state automata



Tools for handling indistinguishability

Transducers/2-tape finite-state automata

- Rational relations = arbitrary finite-state transducers
- Regular relations
- Recognizable relations

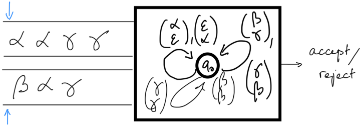


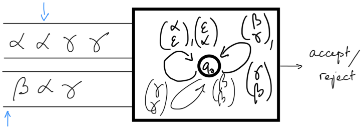
Tools for handling indistinguishability

Transducers/2-tape finite-state automata

- Rational relations = <u>arbitrary finite-state transducers</u>
- Regular relations
- Recognizable relations

$\alpha \sim \varepsilon \qquad \beta \sim \gamma$



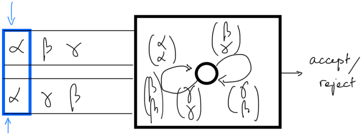


Tools for handling indistinguishability

Transducers/2-tape finite-state automata

- Rational relations = arbitrary finite-state transducers
- Regular relations = synchronous finite-state transducers
- Recognizable relations

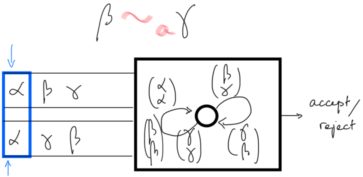
$\beta \sim_\alpha \gamma$

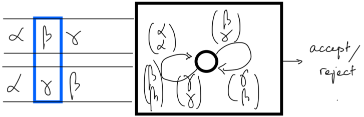


Tools for handling indistinguishability

Transducers/2-tape finite-state automata

- Rational relations = arbitrary finite-state transducers
- Regular relations = synchronous finite-state transducers
- Recognizable relations





Tools for handling indistinguishability

Transducers/2-tape finite-state automata

- Rational relations = arbitrary finite-state transducers
- Regular relations synchronous finite-state transducers
- Recognizable relations = relations of the form $\bigcup_{i<n} L_i \times L'_i$ with L_i, L'_i's regular languages

Tools for handling indistinguishability

Transducers/2-tape finite-state automata

- Rational relations = arbitrary finite-state transducers
- Regular relations synchronous finite-state transducers
- Recognizable relations = relations of the form $\bigcup_{i<n} L_i \times L'_i$
  with L_i, L'_i's regular languages

$$\boxed{\text{Recognizable} \subsetneq \text{Regular} \subsetneq \text{Rational}}$$

Strategic Reasoning

Here qualitative multi-player infinite-horizon games.

- Two-player games of Perfect Information
  - Reachability condition: PTIME
  - parity condition: NP$\cap$co-NP

    [R. McNaughton. Infinite games played on finite graphs. Annals of Pure and Applied Logic (1993).]

    [W. Zielonka. Infinite games on finitely coloured graphs with applications to automata on infinite trees. TCS 1998.]
  - LTL conditions: 2EXPTIME

    [A. Pnueli & R. Rosner. On the Synthesis of an Asynchronous Reactive Module. ICALP 1989]
- Multi-player games of Perfect Information

  Decidable, and non-elementary...

Strategic Reasoning

Here qualitative multi-player infinite-horizon games.

- Logics for multi-player games

  Basically, first-order quantifiers over agent strategies

  - Coalition Logic CL model-checking is PTIME-complete

    [M. Pauly. A Modal Logic for Coalitional Power in Games. 12(1):149–166, 2002.]

  - Alternating-time Temporal Logics ATL* model-checking is 2EXPTIME-complete
  - Strategy Logic SL model-checking is non-elementary
  - ...

Strategic Reasoning under II

- Two-player games is EXPTIME-complete

  [J.H. Reif. The complexity of two-player games of incomplete information. JCSC (1984)]

  [D. Berwanger et al. Strategy construction for parity games with imperfect information. Information and computation. (2010).]

Strategic Reasoning under II

- Two-player games is EXPTIME-complete

  [J.H. Reif. The complexity of two-player games of incomplete information. JCSC (1984)]

  [D. Berwanger et al. Strategy construction for parity games with imperfect information. Information and computation. (2010).]

- Three-player games are undecidable

  [G. Peterson, J.H. Reif, and S. Azhar. Lower bounds for multiplayer noncooperative games of incomplete information. C&M with A.]

  [D. Berwanger and Ł. Kaiser. Information tracking in games on graphs. Journal of Logic, Language and Information. (2010).]

Strategic Reasoning under II

Logical approaches

- ATL* with Perfect Recall is undecidable

  [C. Dima and F.L. Tiplea (2011). Model-checking ATL under imperfect information and perfect recall semantics is undecidable. arXiv]

- ATL* with Imperfect Recall is EXPTIME-complete

  [P.Y. Schobbens, Alternating-time logic with imperfect recall. Electronic Notes in Theoretical Computer Science (2004)]

- ATL*/SL with Perfect Recall and Hierarchical Information is non-elementary [R. Berthon, B. Maubert, A. Murano, S. Rubin, and M.Y. Vardi. 2021. Strategy Logic with Imperfect Information. ACM Trans. Comput. Log. 22(1) (2021)]

Strategic Reasoning under II

Logical approaches

- ATL* with Perfect Recall is undecidable

  [C. Dima and F.L. Tiplea (2011). Model-checking ATL under imperfect information and perfect recall semantics is undecidable. arXiv]

- ATL* with Imperfect Recall is EXPTIME-complete

  [P.Y. Schobbens, Alternating-time logic with imperfect recall. Electronic Notes in Theoretical Computer Science (2004)]

- ATL* /SL with Perfect Recall <u>and</u> Hierarchical Information is non-elementary [R. Berthon, B. Maubert, A. Murano, S. Rubin, and M.Y. Vardi. 2021. Strategy Logic with Imperfect Information. ACM Trans. Comput. Log. 22(1) (2021)]

- Loop μ-calculus (for asynchoronous setting) undecidable

  [**S.P.** & S. Riedweg. A decidable class of problems for control under partial observation. IPL (2005)], [X. Briand. Sur la décidabilité de certains problèmes de synthèse de contrôleurs. PhD Thesis, Uni. de Bordeaux (2006)], [A. Arnold and I. Walukiewicz. Nondeterministic controllers of nondeterministic processes. Logic and automata (2008).]

A Setting for Specifying Games of II with Synchronous Perfect Recall

We consider the setting Dynamic Epistemic Logic (DEL)

[H. van Ditmarsch, W. van Der Hoek, and B. Kooi. Dynamic epistemic logic. Springer Science & Business Media, 2007.]

- with II on the position in the arena

$\Rightarrow$ Epistemic States $\mathscr{S}$

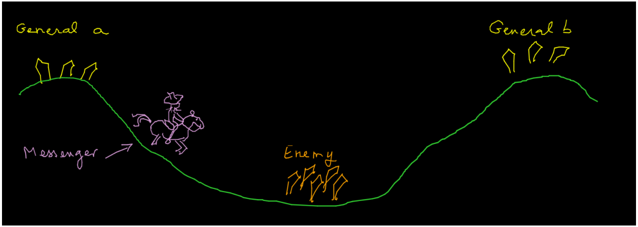
- with II on taken action in the arena

$\Rightarrow$ Action Models $\mathscr{A}$

Epistemic States and Action Models

$$\mathscr{S} \xrightarrow{\mathscr{A}} \mathscr{S}'$$

Coordinated attack problem

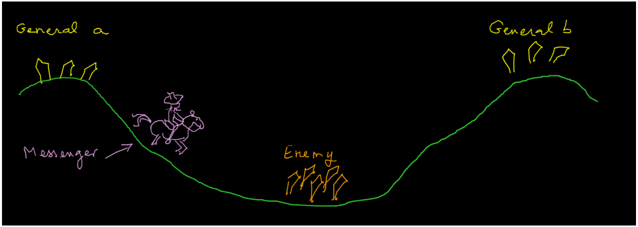


d =“attack at dawn” and m_a =”Msger with General a”

Epistemic States and Action Models

$$\mathscr{S} \xrightarrow{\mathscr{A}} \mathscr{S}'$$

Coordinated attack problem



d =“attack at dawn” and m_a =”Msger with General a”

$$\mathscr{S}: \quad a,b \circlearrowleft v:\{d, m_a\} \overset{b}{\text{———}} u:\{m_a\} \circlearrowright a,b$$

Epistemic States and Action Models

$$\mathscr{S} \xrightarrow{\mathscr{A}} \mathscr{S}'$$

Coordinated attack problem: send messenger to tell "d"

$$\mathscr{S}: \quad a, b \quad v : \{d, m_a\} \xrightarrow{\quad b \quad} u : \{m_a\} \quad a, b$$

Epistemic States and Action Models

$$\mathscr{S} \xrightarrow{\mathscr{A}} \mathscr{S}'$$

Coordinated attack problem: send messenger to tell "d"

$\mathscr{S}$: $\quad a, b \; v : \{d, m_a\} \xrightarrow[]{\;b\;} u : \{m_a\} \; a, b$

$\mathscr{A}$ $\quad a, b$

Arrived
$pre : d \wedge m_a$
$post : \begin{array}{l} m_a \leftarrow \bot \\ m_b \leftarrow \top \end{array}$

a

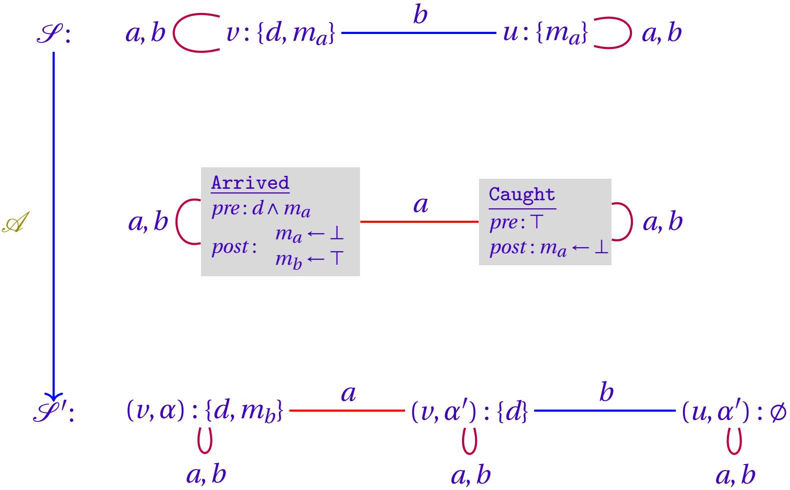
Caught
$pre : \top$
$post : m_a \leftarrow \bot$

a, b

Epistemic States and Action Models

$$\mathscr{S} \xrightarrow{\mathscr{A}} \mathscr{S}'$$

Coordinated attack problem: send messenger to tell "d"



Epistemic States and Action Models

$$\mathscr{S} \xrightarrow{\mathscr{A}} \mathscr{S}'$$

Coordinated attack problem: General a announces "*d*"

$$\mathscr{S}: \quad \begin{tikzcd} v:\{d,m_a\} \arrow[loop left, "{a,b}"] \arrow[r, dash, "b"] & u:\{m_a\} \arrow[loop right, "{a,b}"] \end{tikzcd}$$

$$\Big\downarrow \mathscr{A}$$

$$\begin{tikzcd} \boxed{\begin{array}{l} \underline{d\ \texttt{holds!}} \\ pre: d \\ post: \emptyset \end{array}} \arrow[loop left, "{a,b}"] \end{tikzcd}$$

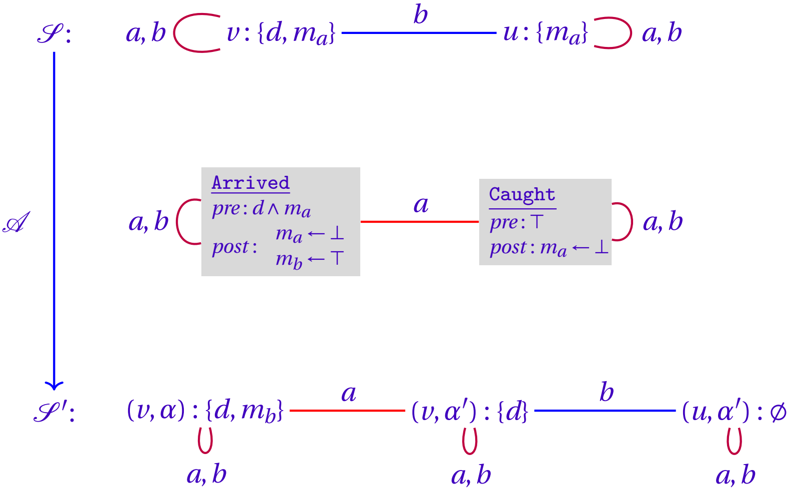
public announcement
that "*d*" holds

$$\mathscr{S}': \quad \begin{tikzcd} (v,\alpha):\{d,m_b\} \arrow[loop below, "{a,b}"] \end{tikzcd}$$

Epistemic States and Action Models

$$\mathscr{S} \xrightarrow{\mathscr{A}} \mathscr{S}'$$

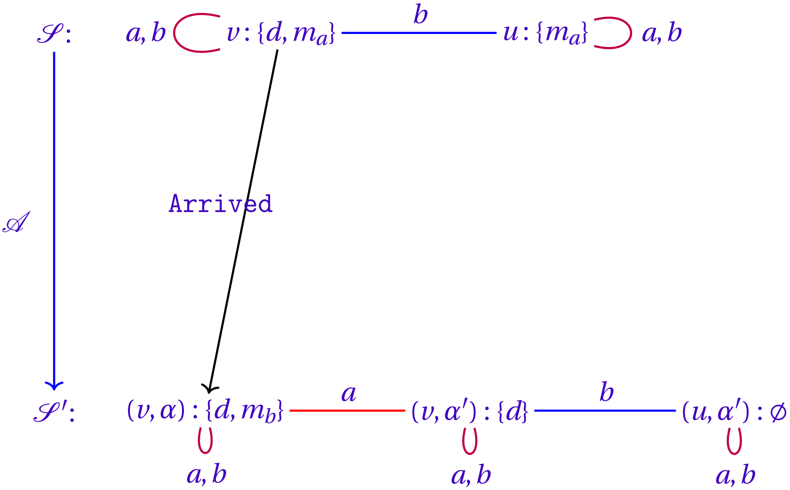
Coordinated attack problem: send messager to tell "*d*"



Epistemic States and Action Models

$$\mathscr{S} \xrightarrow{\mathscr{A}} \mathscr{S}'$$

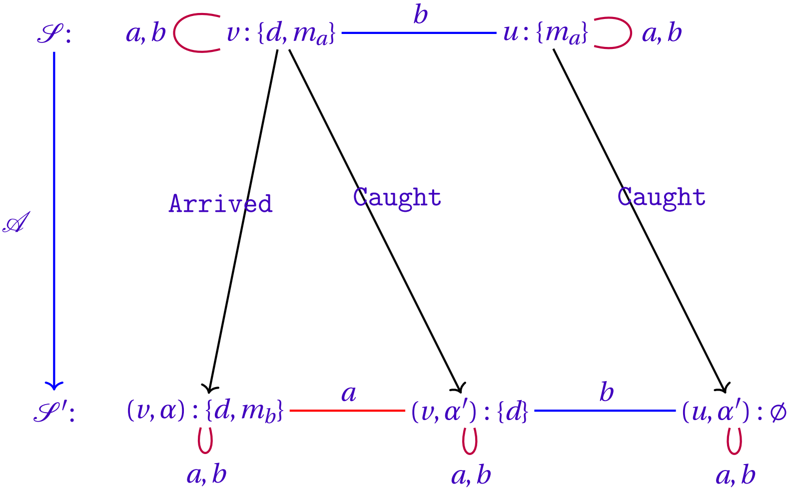
Coordinated attack problem: send messager to tell "d"



Epistemic States and Action Models

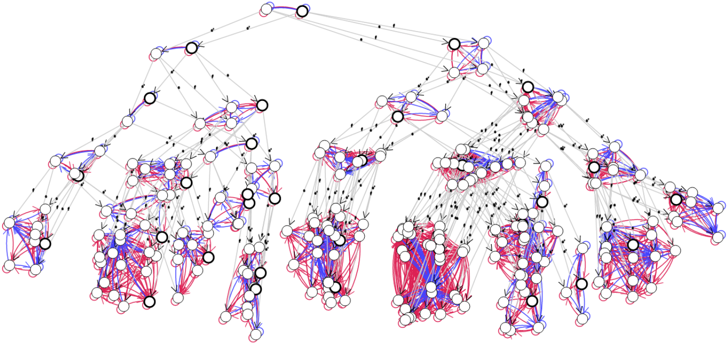
$$\mathscr{S} \xrightarrow{\mathscr{A}} \mathscr{S}'$$

Coordinated attack problem: send messager to tell "d"



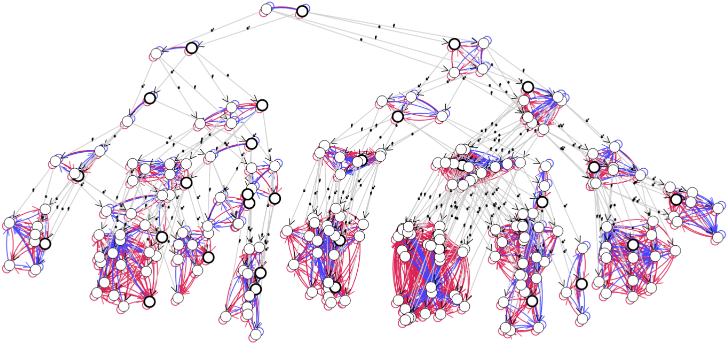
From $(\mathscr{S}, \mathscr{A})$ to an infinite tree $G(\mathscr{S}, \mathscr{A})$

The denoted game arena unfolding



From $(\mathscr{S}, \mathscr{A})$ to an infinite tree $G(\mathscr{S}, \mathscr{A})$

The denoted game arena unfolding

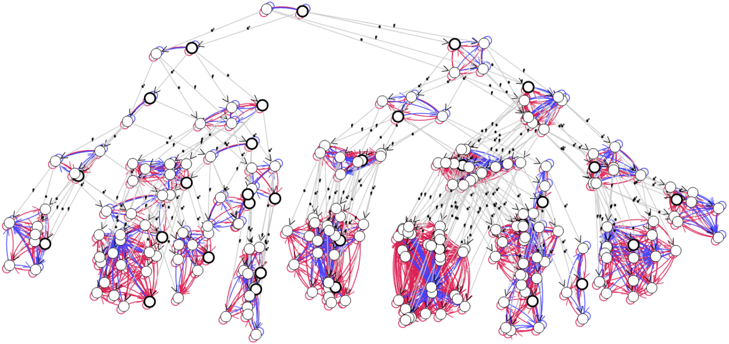


$\vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots$

History $h = v_0 \alpha_1 \alpha_2 \cdots \alpha_n$ with $v_0 \in Pos_I$ and α_i's $\in Act$ is a position in $G(\mathscr{S}, \mathscr{A})$

$$Hist \subseteq Pos_I.Act^*$$

Advantages of focusing on the DEL setting



- The state space is implicit, and might be infinite

  contrary e.g. ATL*, SL

- Provides a unifying framework for:
  - epistemic planning
  - strategic reasoning
- It enables to exhibit action types

Remarkable cases

- Action models that are announcements

  Announcement of property φ:

$$a, b, c \quad \boxed{\begin{array}{l} \underline{\varphi!} \\ pre : \varphi \\ post : - \end{array}}$$

Remarkable cases

- Action models that are announcements

  Announcement of property φ:

  a, b, c $\left(\begin{array}{l} \underline{\varphi!} \\ pre : \varphi \\ post : - \end{array} \right.$

- Action models that are public actions

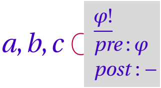
  Putting the cube on the table:

  a, b, c $\left(\begin{array}{l} \underline{\texttt{PutCubeOnTable}} \\ pre : \neg ontable \\ post : ontable \leftarrow \top \end{array} \right.$

Remarkable cases

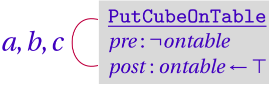
- Action models that are announcements

  Announcement of property φ:

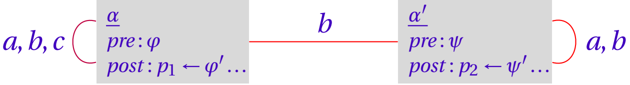


- Action models that are public actions

  Putting the cube on the table:



- Action models that are propositional



with $\varphi, \varphi', \psi, \psi'$ propositional

Remarkable cases

- Action models that are announcements

  Announcement of property φ: a,b,c $\circlearrowleft$ $\underline{\varphi!}$, $pre: \varphi$, $post: -$

- Action models that are public actions

  Putting the cube on the table: a,b,c $\circlearrowleft$ $\underline{\texttt{PutCubeOnTable}}$, $pre: \neg ontable$, $post: ontable \leftarrow \top$

- Action models that are propositional

  $$a,b,c \circlearrowleft \begin{array}{l}\underline{\alpha}\\ pre: \varphi \\ post: p_1 \leftarrow \varphi' \ldots\end{array} \overset{b}{\text{———}} \begin{array}{l}\underline{\alpha'}\\ pre: \psi \\ post: p_2 \leftarrow \psi' \ldots\end{array} \circlearrowright a,b$$

  with $\varphi, \varphi', \psi, \psi'$ propositional

- Hierarchical information

  $$\sim_b \subseteq \sim_a \subseteq \sim_c$$

  i.e.: nested indistinguishability relations among agents

Strategic Reasoning in $G(\mathscr{S}, \mathscr{A})$

- Reachability Goals, subcase of Epistemic Planning

  [T. Bolander, T. Charrier, **S.P.** and F. Schwarzentruber. DEL-based Epistemic Planning: Decidability and Complexity. Artificial Intelligence 2020.]

  [G. Douéneau-Tabot, **S.P.** and F. Schwarzentruber. Chain-Monadic Second Order Logic over Regular Automatic Trees and Epistemic Planning Synthesis. AiML 2018.]

  [B. Maubert, **S.P.** and F. Schwarzentruber. Reachability Games in Dynamic Epistemic Logic. IJCAI 2019.]

- Epistemic Temporal Goals

  [B. Maubert, A. Murano, **S.P.**, F. Schwarzentruber, and S. Stranieri. Dynamic Epistemic Logic Games with Epistemic Temporal Goals. ECAI 2020.]

- A setting for Concurrent Games

  [B. Maubert, **S.P.**, F. Schwarzentruber, and S. Stranieri. Concurrent Games in Dynamic Epistemic Logic. IJCAI 2020.]

Strategic Reasoning in $G(\mathscr{S}, \mathscr{A})$

- Reachability Goals, subcase of Epistemic Planning

  [T. Bolander, T. Charrier, **S.P.** and F. Schwarzentruber. DEL-based Epistemic Planning: Decidability and Complexity. Artificial Intelligence 2020.]

  [G. Douéneau-Tabot, **S.P.** and F. Schwarzentruber. Chain-Monadic Second Order Logic over Regular Automatic Trees and Epistemic Planning Synthesis. AiML 2018.]

  [B. Maubert, **S.P.** and F. Schwarzentruber. Reachability Games in Dynamic Epistemic Logic. IJCAI 2019.]

- Epistemic Temporal Goals

  [B. Maubert, A. Murano, **S.P.**, F. Schwarzentruber, and S. Stranieri. Dynamic Epistemic Logic Games with Epistemic Temporal Goals. ECAI 2020.]

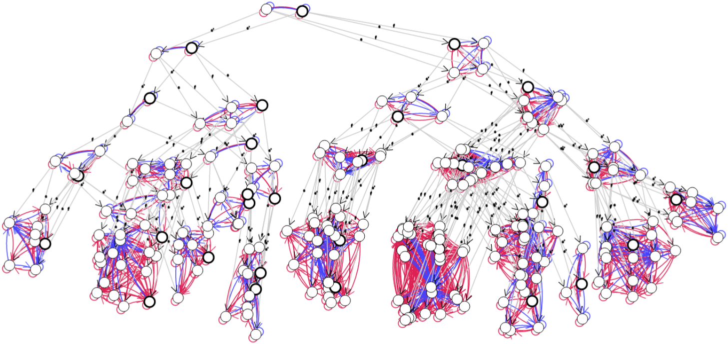
- A setting for Concurrent Games

  [B. Maubert, **S.P.**, F. Schwarzentruber, and S. Stranieri. Concurrent Games in Dynamic Epistemic Logic. IJCAI 2020.]

Epistemic Planning Problem

Input: $(\mathscr{S}, \mathscr{A})$, a position $v_0 \in \mathscr{S}$ and an epistemic formula γ

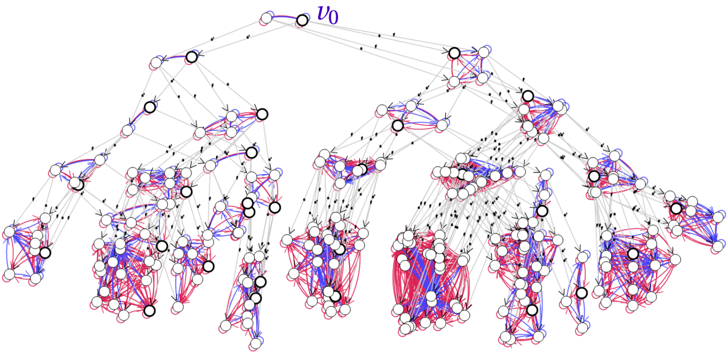
Question: Is there a sequence of actions $\alpha_1, \ldots, \alpha_n$ in $\mathscr{A}$ s.t. $G(\mathscr{S}, \mathscr{A}), v_0\alpha_1 \ldots \alpha_n \models \gamma$?



Epistemic Planning Problem

Input: $(\mathscr{S}, \mathscr{A})$, a position $v_0 \in \mathscr{S}$ and an epistemic formula γ

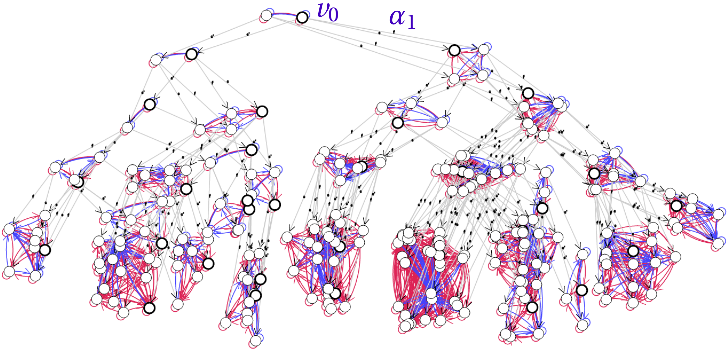
Question: Is there a sequence of actions $\alpha_1, \ldots, \alpha_n$ in $\mathscr{A}$ s.t. $G(\mathscr{S}, \mathscr{A}), v_0\alpha_1 \ldots \alpha_n \models \gamma$?



Epistemic Planning Problem

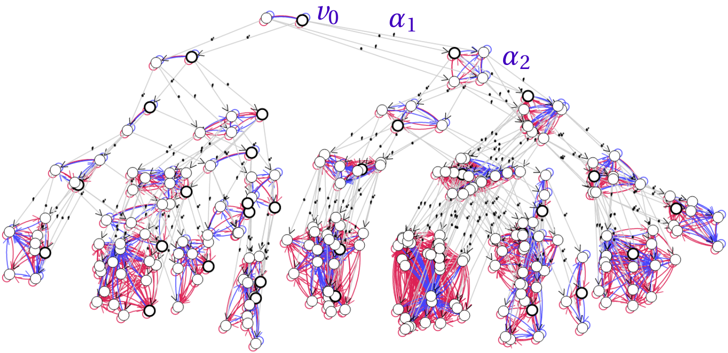
Input: $(\mathscr{S}, \mathscr{A})$, a position $v_0 \in \mathscr{S}$ and an epistemic formula γ

Question: Is there a sequence of actions $\alpha_1, \ldots, \alpha_n$ in $\mathscr{A}$ s.t. $G(\mathscr{S}, \mathscr{A}), v_0\alpha_1 \ldots \alpha_n \models \gamma$?



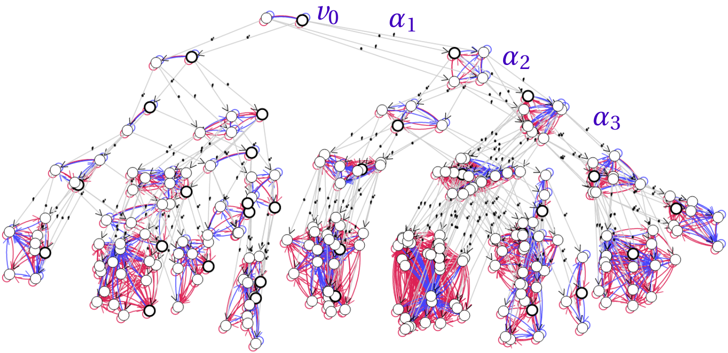
Epistemic Planning Problem

Input:	$(\mathscr{S}, \mathscr{A})$, a position $v_0 \in \mathscr{S}$ and an epistemic formula γ
Question:	Is there a sequence of actions $\alpha_1, \ldots, \alpha_n$ in $\mathscr{A}$ s.t. $G(\mathscr{S}, \mathscr{A}), v_0 \alpha_1 \ldots \alpha_n \models \gamma$?



Epistemic Planning Problem

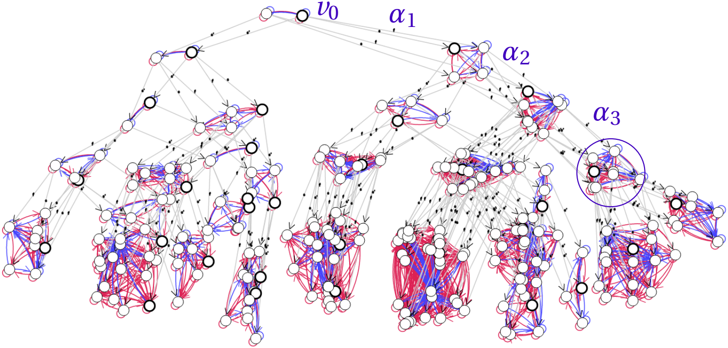
Input:	$(\mathscr{S}, \mathscr{A})$, a position $v_0 \in \mathscr{S}$ and an epistemic formula γ
Question:	Is there a sequence of actions $\alpha_1, \ldots, \alpha_n$ in $\mathscr{A}$ s.t. $G(\mathscr{S}, \mathscr{A}), v_0\alpha_1 \ldots \alpha_n \models \gamma$?



Epistemic Planning Problem

Input: $(\mathscr{S}, \mathscr{A})$, a position $v_0 \in \mathscr{S}$ and an epistemic formula γ

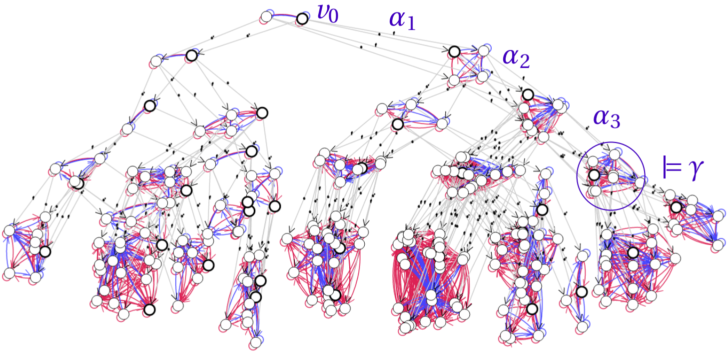
Question: Is there a sequence of actions $\alpha_1, \ldots, \alpha_n$ in $\mathscr{A}$ s.t. $G(\mathscr{S}, \mathscr{A}), v_0 \alpha_1 \ldots \alpha_n \models \gamma$?



Epistemic Planning Problem

Input: $(\mathscr{S}, \mathscr{A})$, a position $v_0 \in \mathscr{S}$ and an epistemic formula γ

Question: Is there a sequence of actions $\alpha_1, \ldots, \alpha_n$ in $\mathscr{A}$ s.t. $G(\mathscr{S}, \mathscr{A}), v_0 \alpha_1 \ldots \alpha_n \models \gamma$?



Theorem

The Epistemic Planning Problem is undecidable.

Epistemic Planning Problem

Input: $(\mathscr{S}, \mathscr{A})$, a position $v_0 \in \mathscr{S}$ and an epistemic formula γ

Question: Is there a sequence of actions $\alpha_1, \ldots, \alpha_n$ in $\mathscr{A}$ s.t. $G(\mathscr{S}, \mathscr{A}), v_0 \alpha_1 \ldots \alpha_n \models \gamma$?

Theorem

The Epistemic Planning Problem is undecidable.

[T. Bolander and M.B. Andersen. Epistemic planning for single and multi-agent systems. J. Appl. Non-Class. Log., 2011.],

[Q. Yu, X. Wen, and Y. Liu. Multi-agent epistemic explanatory diagnosis via reasoning about actions. IJCAI 2013.]

Theorem

The Epistemic Planning Problem is decidable for propositional action models.

[T. Bolander, T. Charrier, **S.P.** and F. Schwarzentruber. DEL-based Epistemic Planning: Decidability and Complexity. Artificial Intelligence 2020.]

Epistemic Planning Problem

Input: $(\mathscr{S}, \mathscr{A})$, a position $v_0 \in \mathscr{S}$ and an epistemic formula γ

Question: Is there a sequence of actions $\alpha_1, \ldots, \alpha_n$ in $\mathscr{A}$ s.t. $G(\mathscr{S}, \mathscr{A}), v_0\alpha_1 \ldots \alpha_n \models \gamma$?

Theorem

The Epistemic Planning Problem is undecidable.

[T. Bolander and M.B. Andersen. Epistemic planning for single and multi-agent systems. J. Appl. Non-Class. Log., 2011.],

[Q. Yu, X. Wen, and Y. Liu. Multi-agent epistemic explanatory diagnosis via reasoning about actions. IJCAI 2013.]

Theorem

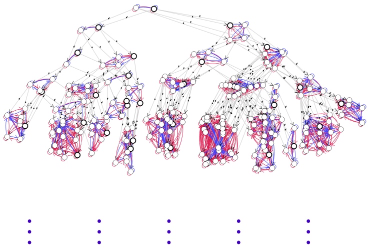
The Epistemic Planning Problem is decidable for propositional action models.

★ [T. Bolander, T. Charrier, **S.P.** and F. Schwarzentruber. DEL-based Epistemic Planning: Decidability and Complexity. Artificial Intelligence 2020.]

Remarkable Properties of $G(\mathscr{S}, \mathscr{A})$ with propositional $\mathscr{A}$

$\mathscr{A}$ with only

$\underline{\alpha}$
pre : φ propositional
post : ψ propositional



Theorem

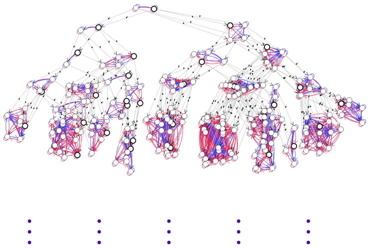
For $\mathscr{A}$ propositional,

- $G(\mathscr{S}, \mathscr{A})$ *is an automatic structure* $\Rightarrow$ *its* FO *theory is decidable*

Remarkable Properties of $G(\mathscr{S}, \mathscr{A})$ with propositional $\mathscr{A}$

$\mathscr{A}$ with only

$\underline{\alpha}$
pre : φ propositional
post : ψ propositional



Theorem

For $\mathscr{A}$ propositional,

- *$G(\mathscr{S}, \mathscr{A})$ is an automatic structure $\Rightarrow$ its* FO *theory is decidable*
- *$G(\mathscr{S}, \mathscr{A})$ is even a Regular Automatic Tree*
  $\Rightarrow$ its CHAINMSO *theory is decidable*

An example of automatic structure

Structure $\langle \mathbb{N}, \leq \rangle$.

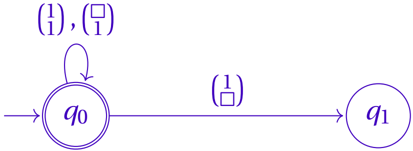
$$\begin{tikzcd} 0 \arrow[r] \arrow[rr, bend left] \arrow[rrr, bend left] & 1 \arrow[r] & 2 \arrow[r] & 3 \arrow[r] & \cdots \end{tikzcd}$$

- Encode each $n \in \mathbb{N}$ by $enc(n) = \overbrace{11\ldots1}^{n} = 1^n$ $\qquad$ 1^* is regular
- Use the *convolution* $\otimes$ on words, that aligns words:

$$1^2 \otimes 1^3 := \binom{1}{1}\binom{1}{1}\binom{\square}{1}$$

- Encode $\leq \subseteq \mathbb{N} \times \mathbb{N}$ with automaton (as an SPR relation):

$\mathbf{A}_{\leq}$:



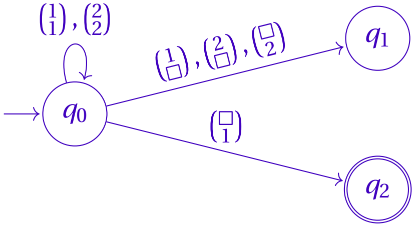
$$\boxed{1^n \otimes 1^m \in \mathscr{L}(\mathbf{A}_{\leq}) \text{ iff } n \leq m}$$

$\mathscr{T}_2$ is automatic

Complete binary tree $\mathscr{T}_2 = \langle \{1,2\}^*, \mathtt{root}, \mathtt{suc}_1, \mathtt{suc}_2 \rangle$

- Encode nodes h as is: $h \in \{1,2\}^*$ (regular)
- Unary predicate $\mathtt{root}$: use an automaton $\mathbf{A}_{\mathtt{root}}$ that accepts only empty word;
- $\mathtt{suc}_1(h, h')$ iff $h' = h.1$

$\mathbf{A}_{\mathtt{suc}_1}$:



$\mathcal{T}_2^{\texttt{el}}$ is automatic

Binary predicate $\texttt{el}$ means "equal level"

Complete binary tree $\mathcal{T}_2^{\texttt{el}} = \langle \{1,2\}^*, \texttt{root}, \texttt{suc}_1, \texttt{suc}_2, \texttt{el} \rangle$

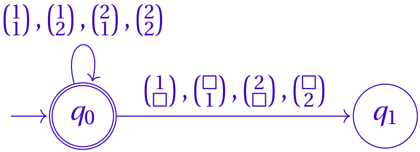
- Encode nodes h as is: $h \in \{1,2\}^*$ (regular)
- Unary predicate $\texttt{root}$: use an automaton $\mathbf{A}_{\texttt{root}}$ that accepts only empty word;
- $\texttt{suc}_1(h, h')$ iff $h' = h.1$ and $\texttt{suc}_2(h, h')$ iff $h' = h.2$
- $\texttt{el}(h, h')$ iff $|h| = |h'|$

$\mathscr{T}_2$ is automatic

Complete binary tree $\mathscr{T}_2 = \langle \{1,2\}^*, \texttt{root}, \texttt{suc}_1, \texttt{suc}_2 \rangle$

- Encode nodes h as is: $h \in \{1,2\}^*$ (regular)
- Unary predicate $\texttt{root}$: use an automaton $\mathbf{A}_{\texttt{root}}$ that accepts only empty word;
- $\texttt{suc}_1(h, h')$ iff $h' = h.1$ and $\texttt{suc}_2(h, h')$ iff $h' = h.2$
- $\texttt{el}(h, h')$ iff $|h| = |h'|$

$\mathbf{A}_{\texttt{el}}$:



Automatic Structures

Proposition

For propositional $\mathscr{A} = (Act, \ldots)$ *and SPR relations* $\sim_a$,

$$G(\mathscr{S}, \mathscr{A}) = \langle Hist, \{\mathtt{suc}_\alpha\}_{\alpha \in Act}, \{\sim_a\}_{a \in Agt}, \ldots \rangle \text{ is automatic.}$$

- $Hist \subseteq Pos_I.Act^*$ is a regular

Automatic Structures

Proposition

For propositional $\mathscr{A} = (Act, \ldots)$ and SPR relations $\sim_a$,

$$G(\mathscr{S}, \mathscr{A}) = \langle Hist, \{\texttt{suc}_\alpha\}_{\alpha \in Act}, \{\sim_a\}_{a \in Agt}, \ldots \rangle \text{ is automatic.}$$

- $Hist \subseteq Pos_I.Act^*$ is a regular
  - ⚠ It is not the case for action models where preconditions have knowledge modalities of ≥ 1-alternation-depth.

    [T. Charrier, B. Maubert, F. Schwarzentruber: On the Impact of Modal Depth in Epistemic Planning. IJCAI 2016.]

Automatic Structures

Proposition

For propositional $\mathscr{A} = (Act, \ldots)$ and SPR relations $\sim_a$,

$$G(\mathscr{S}, \mathscr{A}) = \langle Hist, \{\mathtt{suc}_\alpha\}_{\alpha \in Act}, \{\sim_a\}_{a \in Agt}, \ldots \rangle \text{ is automatic.}$$

- $Hist \subseteq Pos_I.Act^*$ is a regular
  - ⚠ It is not the case for action models where preconditions have knowledge modalities of ≥ 1-alternation-depth.
  - ⚠ Long standing open question for preconditions with alternation-depth $= 1$

Automatic Structures

Proposition

For propositional $\mathscr{A} = (Act, \ldots)$ *and SPR relations* $\sim_a$*,*

$$G(\mathscr{S}, \mathscr{A}) = \langle Hist, \{\mathtt{suc}_\alpha\}_{\alpha \in Act}, \{\sim_a\}_{a \in Agt}, \ldots \rangle \text{ is automatic.}$$

- $Hist \subseteq Pos_I.Act^*$ is a regular
  - ⚠ It is not the case for action models where preconditions have knowledge modalities of ≥ 1-alternation-depth.
  - ⚠ Long standing open question for preconditions with alternation-depth = 1
- $\mathtt{suc}_\alpha$ is made of pairs (h, h') with $h' = h.\alpha$

Automatic Structures

Proposition

For propositional $\mathscr{A} = (Act, \ldots)$ and SPR relations $\sim_a$,

$$G(\mathscr{S}, \mathscr{A}) = \langle Hist, \{\mathtt{suc}_\alpha\}_{\alpha \in Act}, \{\sim_a\}_{a \in Agt}, \ldots \rangle \text{ is automatic.}$$

- $Hist \subseteq Pos_I.Act^*$ is a regular
  - ⚠ It is not the case for action models where preconditions have knowledge modalities of ≥ 1-alternation-depth.
  - ⚠ Long standing open question for preconditions with alternation-depth $= 1$
- $\mathtt{suc}_\alpha$ is made of pairs (h, h') with $h' = h.\alpha$
- Remind $\sim_a$ when SPR is:

$$\begin{array}{ccccccc}
h = & v_0 & \alpha_1 & v_1 & \cdots & \alpha_n & v_n \\
\wr_a & \wr_a & \wr_a & \wr_a & \cdots & \wr_a & \wr_a \\
h' = & v'_0 & \alpha'_1 & v'_1 & \cdots & \alpha'_m & v'_m
\end{array}$$

Automaton for relation $\Phi^{\mathcal{M}}$

$\mathcal{M}$ automatic and $\Phi \in \textsc{Fo}$

$$\Phi^{\mathcal{M}} := \{(d_1 \ldots d_n) \in Dom^n \mid \mathcal{M}, [x_i \mapsto d_i] \models \Phi(x_1 \ldots x_n)\}$$

Proposition

Given $\mathcal{M}$ automatic $= (\mathbf{A}_{Dom}, \mathbf{A}_1, \ldots, \mathbf{A}_k)$ and $\Phi \in \textsc{Fo}$, there is an effective construction $\mathbf{A}_\Phi$ that recognizes $\Phi^{\mathcal{M}}$.

Inductive construction over Φ:

Formula	Automaton
$R_i(x_1 \ldots x_{r_i})$	$\mathbf{A}_i$ (given)
$\neg\Phi$	complement $\mathbf{A}_\Phi$
$\Phi \wedge \Psi$	intersect $\mathbf{A}_\Phi$ and $\mathbf{A}_\Psi$
$\exists x \Phi$	ignore tape content for x in $\mathbf{A}_\Phi$

Automaton for relation $\Phi^{\mathscr{M}}$

$\mathscr{M}$ automatic and $\Phi \in \textsc{Fo}$

$$\Phi^{\mathscr{M}} := \{(d_1 \dots d_n) \in Dom^n \mid \mathscr{M}, [x_i \mapsto d_i] \models \Phi(x_1 \dots x_n)\}$$

Proposition

Given $\mathscr{M}$ automatic $= (\mathbf{A}_{Dom}, \mathbf{A}_1, \dots, \mathbf{A}_k)$ and $\Phi \in \textsc{Fo}$, there is an effective construction $\mathbf{A}_\Phi$ that recognizes $\Phi^{\mathscr{M}}$.

Theorem

Model-checking automatic structures against $\textsc{Fo}$ is decidable.

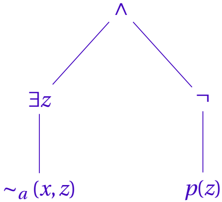
[A. Blumensath, and E. Gradel. Automatic structures. LICS 2000.]

[B. Khoussainov and A. Nerode. Effective properties of finitely generated re algebras. In Feasible Mathematics II (1995).]

Bottom-up construction of $\mathbf{A}_\Phi$: intuitive example

Want to catch those h s.t. $G(\mathscr{S},\mathscr{A}), h \models \neg K_a p$?

- Translate $K_a p$ into FO: $\mathtt{tr}(K_a p)(x) = \exists z(\sim_a(x,z) \wedge \neg p(z))$
- Construct automaton for $\exists z(\sim_a(x,z) \wedge \neg p(z))$ in $G(\mathscr{S},\mathscr{A})$

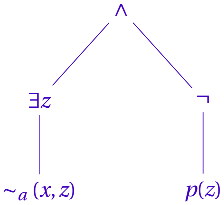


Given $\mathbf{A}_D$, $\mathbf{A}_{\sim_a(x,z)}$ and $\mathbf{A}_{p(z)}$ from $G(\mathscr{S},\mathscr{A})$:

Bottom-up construction of $\mathbf{A}_\Phi$: intuitive example

Want to catch those h s.t. $G(\mathscr{S}, \mathscr{A}), h \models \neg K_a p$?

- Translate $K_a p$ into FO: $\mathtt{tr}(K_a p)(x) = \exists z(\sim_a(x,z) \wedge \neg p(z))$
- Construct automaton for $\exists z(\sim_a(x,z) \wedge \neg p(z))$ in $G(\mathscr{S}, \mathscr{A})$



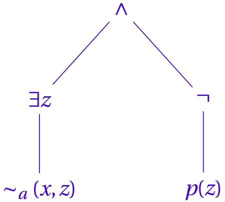
Given $\mathbf{A}_D$, $\mathbf{A}_{\sim_a(x,z)}$ and $\mathbf{A}_{p(z)}$ from $G(\mathscr{S}, \mathscr{A})$:

1. $\mathbf{A}_{\neg p(z)} := \mathbf{A}^c_{p(z)}$ $\ldots \cap \mathbf{A}_D$

Bottom-up construction of $\mathbf{A}_\Phi$: intuitive example

Want to catch those h s.t. $G(\mathscr{S}, \mathscr{A}), h \models \neg K_a p$?

- Translate $K_a p$ into FO: $\mathtt{tr}(K_a p)(x) = \exists z(\sim_a(x,z) \wedge \neg p(z))$
- Construct automaton for $\exists z(\sim_a(x,z) \wedge \neg p(z))$ in $G(\mathscr{S}, \mathscr{A})$



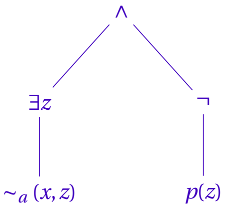
Given $\mathbf{A}_D$, $\mathbf{A}_{\sim_a(x,z)}$ and $\mathbf{A}_{p(z)}$ from $G(\mathscr{S}, \mathscr{A})$:

1. $\mathbf{A}_{\neg p(z)} := \mathbf{A}^c_{p(z)}$ $\quad \ldots \cap \mathbf{A}_D$
2. $\mathbf{A}_{\sim_a(x,z) \wedge \neg p(z)} := \mathbf{A}_{\sim_a(x,z)} \cap \mathbf{A}_{\neg p(x)}$

Bottom-up construction of $\mathbf{A}_\Phi$: intuitive example

Want to catch those h s.t. $G(\mathscr{S},\mathscr{A}), h \models \neg K_a p$?

- Translate $K_a p$ into FO: $\mathtt{tr}(K_a p)(x) = \exists z(\sim_a(x,z) \land \neg p(z))$
- Construct automaton for $\exists z(\sim_a(x,z) \land \neg p(z))$ in $G(\mathscr{S},\mathscr{A})$



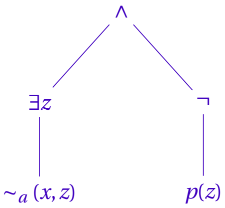
Given $\mathbf{A}_D$, $\mathbf{A}_{\sim_a(x,z)}$ and $\mathbf{A}_{p(z)}$ from $G(\mathscr{S},\mathscr{A})$:

1. $\mathbf{A}_{\neg p(z)} := \mathbf{A}^c_{p(z)}$ $\ldots \cap \mathbf{A}_D$
2. $\mathbf{A}_{\sim_a(x,z)\land\neg p(z)} := \mathbf{A}_{\sim_a(x,z)} \cap \mathbf{A}_{\neg p(x)}$
3. $\mathbf{A}_{\exists z(\sim_a(x,z)\land\neg p(z))} :=$ remove z-tape of $\mathbf{A}_{\sim_a(x,z)\land\neg p(z))}$

Bottom-up construction of $\mathbf{A}_\Phi$: intuitive example

Want to catch those h s.t. $G(\mathscr{S}, \mathscr{A}), h \models \neg K_a p$?

- Translate $K_a p$ into FO: $\mathtt{tr}(K_a p)(x) = \exists z(\sim_a(x,z) \wedge \neg p(z))$
- Construct automaton for $\exists z(\sim_a(x,z) \wedge \neg p(z))$ in $G(\mathscr{S}, \mathscr{A})$



Given $\mathbf{A}_D$, $\mathbf{A}_{\sim_a(x,z)}$ and $\mathbf{A}_{p(z)}$ from $G(\mathscr{S}, \mathscr{A})$:

1. $\mathbf{A}_{\neg p(z)} := \mathbf{A}^c_{p(z)}$ $\ldots \cap \mathbf{A}_D$
2. $\mathbf{A}_{\sim_a(x,z) \wedge \neg p(z)} := \mathbf{A}_{\sim_a(x,z)} \cap \mathbf{A}_{\neg p(x)}$
3. $\mathbf{A}_{\exists z(\sim_a(x,z) \wedge \neg p(z))}$:= remove z-tape of $\mathbf{A}_{\sim_a(x,z) \wedge \neg p(z))}$

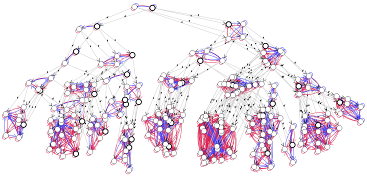
$$\mathscr{L}(\mathbf{A}_{\mathtt{tr}(K_a p)}) = \Phi^{G(\mathscr{S}, \mathscr{A})}$$

Application to Epistemic Planning

Input:	$(\mathscr{S}, \mathscr{A})$, a position $v_0 \in \mathscr{S}$ and an epistemic formula γ
Question:	Is there a sequence of actions $\alpha_1, \ldots, \alpha_n$ in $\mathscr{A}$ s.t. $v_0\alpha_1 \ldots \alpha_n \models \gamma$?

amounts to verifying

$$G(\mathscr{S}, \mathscr{A}) \models_{\mathrm{FO}} \exists x, \mathtt{tr}(\gamma)(x)?$$



with canonical translation from epistemic logic into FO:

$$\varphi \mapsto \mathtt{tr}(\varphi)(x)$$

Application to Epistemic Planning

Input:	$(\mathscr{S}, \mathscr{A})$, a position $v_0 \in \mathscr{S}$ and an epistemic formula γ
Question:	Is there a sequence of actions $\alpha_1, \ldots, \alpha_n$ in $\mathscr{A}$ s.t. $v_0 \alpha_1 \ldots \alpha_n \models \gamma$?

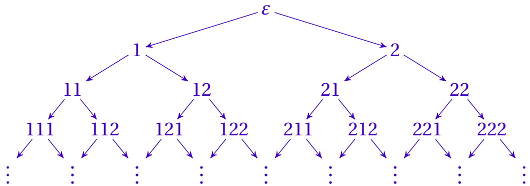
amounts to verifying

$$G(\mathscr{S}, \mathscr{A}) \models_{\mathrm{FO}} \exists x, \mathtt{tr}(\gamma)(x)?$$

As we have (input-free) automaton for $\mathbf{A}_{\exists x, \mathtt{tr}(\gamma)(x)}$:

- just check non-emptiness of $\mathbf{A}_{\exists x, \mathtt{tr}(\gamma)(x)}$
- BUT ALSO use $\mathbf{A}_{\mathtt{tr}(\gamma)(x)}$
  - get all plan solutions
  - and ask e.g. is the set of solutions infinite? is there some solution that satifies some extra conditions, e.g. that belongs to some language of interest? etc.

The MSO-theory of the full binary infinite tree



Theorem (Rabin 1969)

The MSO*-theory of* $\mathcal{T}_2 = \langle \{1,2\}^*, \mathtt{suc}_1, \mathtt{suc}_2 \rangle$ *is decidable.*

Proof based on tree automata construction.

Model checking MSO over automatic structures

Theorem (Thomas 1990)

The MSO*-theory of* $\mathscr{T}_2^{el} = \langle \{1,2\}^*, \epsilon, \mathtt{suc}_1, \mathtt{suc}_2, \mathtt{el} \rangle$ *is undecidable.*

Reduce the undecidable MSO-theory of the infinite grid, also see

[Calbrix, H. et al. La théorie monadique du second ordre du monoïde inversif libre est indécidable. Bulletin of the Belgian Mathematical Society-Simon Stevin (in French) (1997).]

Corollary

Model checking over the class of propositional $G(\mathscr{S}, \mathscr{A})$ *against* MSO *is undecidable.*

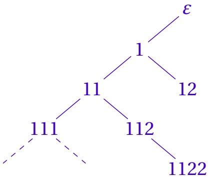
Can we do more that FO (but less than MSO) for $G(\mathscr{S},\mathscr{A})$?

Can we do more that FO (but less than MSO) for $G(\mathscr{S},\mathscr{A})$?

Propositional $G(\mathscr{S},\mathscr{A})$ are not arbitrary Automatic Structures, but

Regular Automatic Trees (RATs)

RAT $\subsetneq$ AUT:

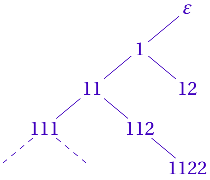


$\{1^n 2^m \mid 0 \leq m \leq n\}$

$\notin$ RAT

$\in$ AUT

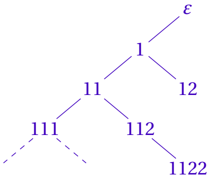
RAT $\subsetneq$ AUT



The domain with encoding *id*
$\{1^i 2^j \mid 0 \leq j \leq i\}$ is not regular.

$\notin$ RAT

RAT $\subsetneq$ AUT



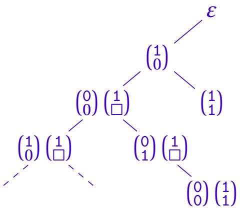
The domain with encoding *id*
$\{1^i 2^j \mid 0 \leq j \leq i\}$ is not regular.

$\notin$ RAT

$\in$ AUT

Use $\texttt{bin}(n)$ (least significant digit first)

$$enc(1^i 2^j) := \texttt{bin}(i) \otimes \texttt{bin}(j)$$



Properties of RATs

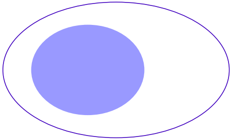
$\mathscr{T} = \langle Dom, \mathtt{root}, \mathtt{suc}_1, \ldots, \mathtt{suc}_n, \mathtt{R}_1, \ldots, \mathtt{R}_k \rangle \in$ RAT.

Lemma

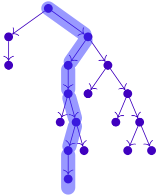
$\mathscr{T} \in$ RAT *implies* $(\mathscr{T} + \{\mathtt{suc}^*, \mathtt{el}, \mathtt{higher}, =\}) \in$ RAT.

In particular, since $\mathscr{T}_2 \in$ RAT, we have $\mathscr{T}_2^{\mathtt{el}} \in$ RAT.

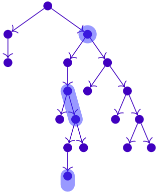
Variants of MSO over trees: chainMSO



(a) MSO
quantification over any subset



(b) pathMSO
quantification over any path in a tree



(c) chainMSO
quantification over any chain in a tree

Decidability of ChainMso over Rat

[G. Douéneau-Tabot, **S.P.** and F. Schwarzentruber. Chain-Monadic Second Order Logic over Regular Automatic Trees and Epistemic Planning Synthesis. AiML 2018.]

Theorem

Model checking over Rat *against* ChainMso *is decidable.*

Proof sketch: Inspired from [Thomas, W., Languages, automata, and logic. Handbook of formal languages, Springer (1997).]

- Chains representation: infinite word over alphabet

$$\mathsf{Branches} \times \{0,1\}^{\omega}$$

- Infinite-word automata for ChainMso formulas
  (vs. Finite-word automata for Fo formulas)

Corollaries of CHAINMSO-RAT decidability

Corollaries of CHAINMSO-RAT decidability

- Since over a unary alphabet every set is a chain:

Corollary (Barany 2007)

MSO *theory of an automatic structure on a unary alphabet is decidable.*

Corollaries of CHAINMSO-RAT decidability

- Since over a unary alphabet every set is a chain:

Corollary (Barany 2007)

MSO *theory of an automatic structure on a unary alphabet is decidable.*

- Since in RATs, relations $\mathtt{suc}^*, \mathtt{el}, \mathtt{higher}, =$ are regular:

Corollary

CHAINMSO$[\mathit{root}, \mathtt{suc}_1, \ldots, \mathtt{suc}_n, R_1, \ldots, R_p, \mathtt{suc}^*, \mathtt{higher}, \mathtt{el}, =]$ *theory of* RATs *is decidable.*

Corollaries of CHAINMSO-RAT decidability

- Since over a unary alphabet every set is a chain:

Corollary (Barany 2007)

MSO *theory of an automatic structure on a unary alphabet is decidable.*

- Since in RATs, relations $\mathtt{suc}^*, \mathtt{el}, \mathtt{higher}, =$ are regular:

Corollary

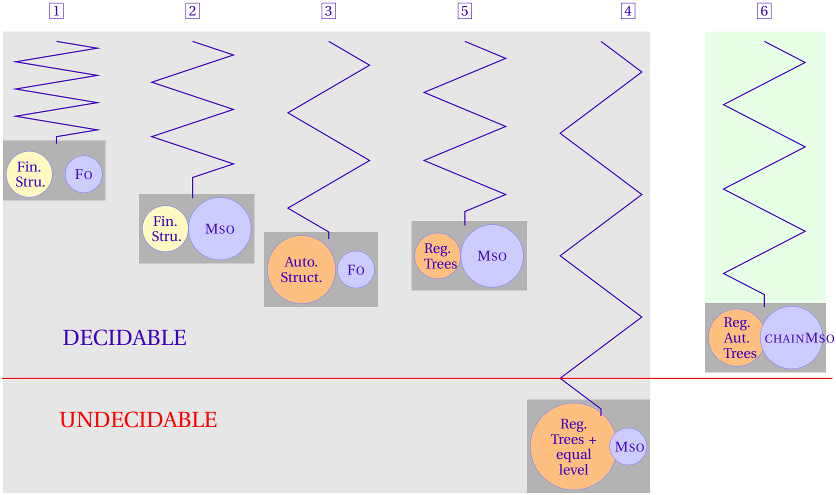
CHAINMSO$[\mathit{root}, \mathtt{suc}_1, \ldots, \mathtt{suc}_n, R_1, \ldots, R_p, \mathtt{suc}^*, \mathtt{higher}, \mathtt{el}, =]$ *theory of* RATs *is decidable.*

- Since one can express in CHAINMSO that a chain is a path:

Corollary

PATHMSO *theory of RATs is decidable.*

The big picture



Back to Strategic Reasoning in propositional $G(\mathscr{S}, \mathscr{A})$

- $\textsc{chainMso}$ captures logics of knowledge and time (Halpern-Vardi 1989):

  Epistemic Temporal Logic models are propositional $G(\mathscr{S}, \mathscr{A})$.

Logics of knowledge and time

$$
\begin{tikzcd}
\mathsf{CTLK} \arrow[r] \arrow[d, bend right] & \mathsf{CTL}^*\mathrm{K} \arrow[r] & \mathsf{BL}_{\mu}^{\mathsf{lin}}\mathrm{K} \arrow[r] & \textsc{chainMso}\mathrm{K} \\
\textsc{Fo}[\mathtt{root}, (\mathtt{suc}_a)_{a \in Act}, \mathtt{suc}, \mathtt{suc}^*, (p)_{p \in \mathsf{AP}}, K_1, \ldots, K_n, \mathtt{higher}, =] \arrow[urrr, bend right] & & &
\end{tikzcd}
$$

Back to Strategic Reasoning in propositional $G(\mathscr{S}, \mathscr{A})$

- Examples of what can be verified:
  - ‘invariantly, intruder *a* does not know the location of the piece of jewelry 2 more than 3 consecutive W steps’
  - ’all drones know that the region is safe every 20 step’
  - ’with the current plan, drone *a* never knows the region is safe but every 10 steps, there is a(nother) plan to let it know the region is safe’

Gamification of $G(\mathscr{S}, \mathscr{A})$ (1/2)

Controller Synthesis

Definition

Input:

- two players: Controller and Environment
(agents $a, b, c, \ldots$ are observers)
- two disjoint sets of (epistemic) actions Act_{Ctrl} and Act_{Env}
- an initial epistemic state $\mathscr{S}_I$ with a distinguished position v_I
- an action model $\mathscr{A}$
- a goal formula γ

Question:

Does Controller have a winning strategy from v_I to reach γ?

[B. Maubert, **S.P.** and F. Schwarzentruber. Reachability Games in Dynamic Epistemic Logic. IJCAI 2019.]

Gamification of $G(\mathscr{S}, \mathscr{A})$ (2/2)

Distributed Strategy Synthesis

Definition

Input:

- thee agents are the players: a partition $Agt = Agt_{\exists} \uplus Agt_{\forall}$
- a set of epistemic actions per agent $Act_1, \ldots, Act_n$
- an initial epistemic state $\mathscr{S}_I$ with a distinguished position v_I
- an action model $\mathscr{A}$
- a goal formula γ (in Epistemic Logic)

Question:

Does team $Agt_{\exists}$ have a winning distributed uniform strategy from v_I to reach γ?

[B. Maubert, **S.P.** and F. Schwarzentruber. Reachability Games in Dynamic Epistemic Logic. IJCAI 2019.]

Complexity results

	epistemic planning one external player	controller synthesis two external players perfect info	distributed strategy synthesis agents = players imperfect info
public announcements	NP-c [Bolander et al. 2015]	PSPACE-c	
public actions	PSPACE-c folklore	EXPTIME-c	
propositional actions models	decidable [Douénau-Tabot et al. 2018] [Maubert et al. 2014]	decidable automata theory [Bozelli et al. 2015]	undecidable [Reif, Peterson, 1979] [Coulombe, Lynch, 2018]
any pre/post	undecidable [Anderson, Bolander, 2011]		

Gamification of $G(\mathscr{S}, \mathscr{A})$

More expressive winning conditions or game settings

- Epistemic Temporal Goals

  [B. Maubert, A. Murano, **S.P.**, F. Schwarzentruber, and S. Stranieri. Dynamic Epistemic Logic Games with Epistemic Temporal Goals. ECAI 2020.]

- $G(\mathscr{S}, \mathscr{A})$ as a concurrent game

  [B. Maubert, **S.P.**, F. Schwarzentruber, and S. Stranieri. Concurrent Games in Dynamic Epistemic Logic. IJCAI 2020.]

What I did not talk about

- Finer sub-classes of games

  e.g. the case of **recognizable relations** decidable via jumping automata simulated by two-way tree automata.

  [Laura Bozzelli, Bastien Maubert, **S.P.**. Uniform strategies, rational relations and jumping automata. Inf. Comput. (2015)]

- Quantitative aspects

- ...

Thank you for listening!

$$G(\mathscr{S}, \mathscr{A}) \models \exists t, \texttt{CoffeeBreak}(t)?$$

Questions?