Agent Ontology Alignment Repair through Dynamic Epistemic Logic

Line van den Berg, Manuel Atencia and Jérôme Euzenat

Univ. Grenoble Alpes & Inria
{line.van-den-berg,manuel.atencia,jerome.euzenat}@inria.fr
https://moex.inria.fr

Summary

Summary

- Agents communicate using alignments between ontologies

Summary

- Agents communicate using alignments between ontologies
- Repair alignments when failures occur via adaptation operators

Summary

- Agents communicate using alignments between ontologies
- Repair alignments when failures occur via adaptation operators
- We introduce a formal framework to model this

Summary

- Agents communicate using alignments between ontologies
- Repair alignments when failures occur via adaptation operators
- We introduce a formal framework to model this
- And prove that some adaptation operators are incorrect or redundant, and all are incomplete

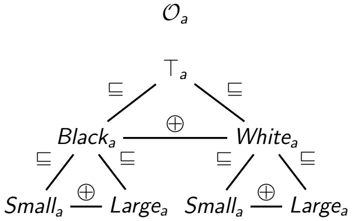
Alignment Repair Game [Euzenat 2014, Euzenat 2017]

Alignment Repair Game [Euzenat 2014, Euzenat 2017]

- Agents use (different) ontologies

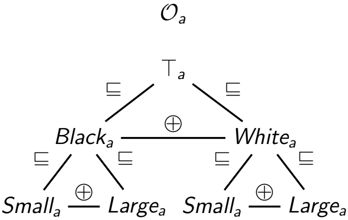
Alignment Repair Game [Euzenat 2014, Euzenat 2017]

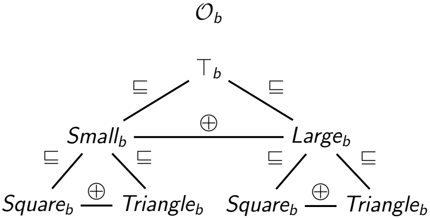
- Agents use (different) ontologies



Alignment Repair Game [Euzenat 2014, Euzenat 2017]

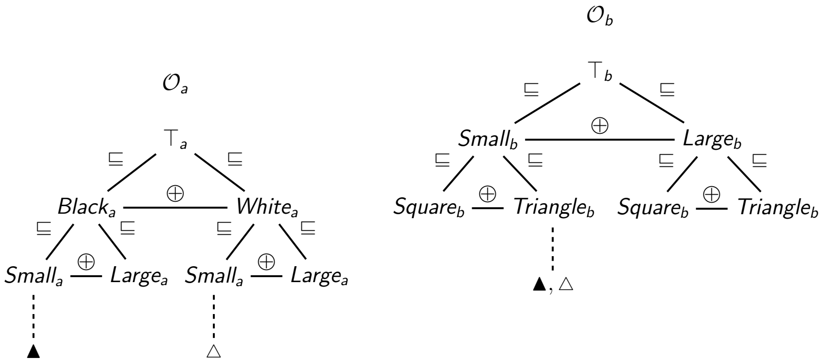
- Agents use (different) ontologies





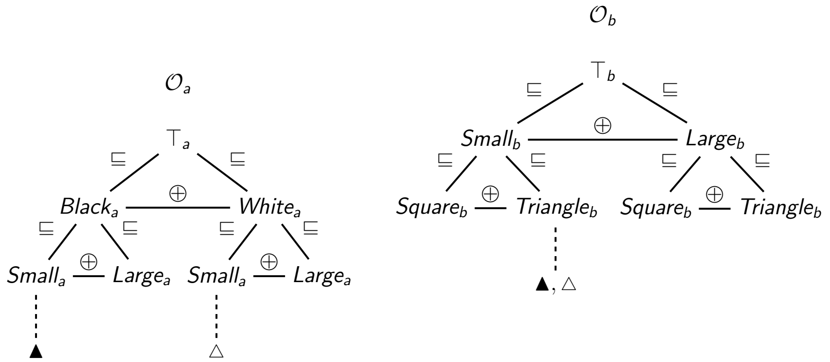
Alignment Repair Game [Euzenat 2014, Euzenat 2017]

- Agents use (different) ontologies



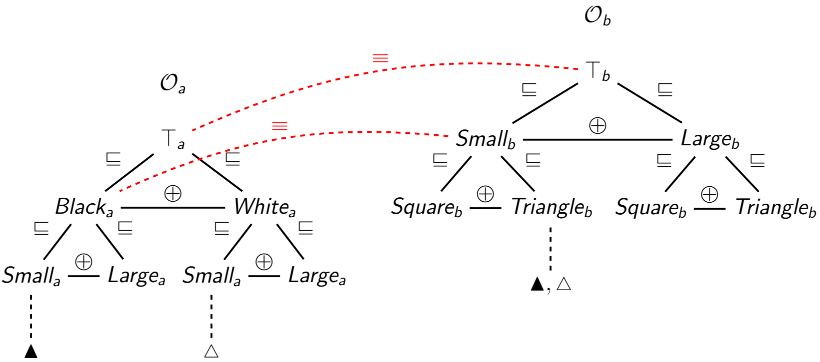
Alignment Repair Game [Euzenat 2014, Euzenat 2017]

- Agents use (different) ontologies
- They use alignments to communicate



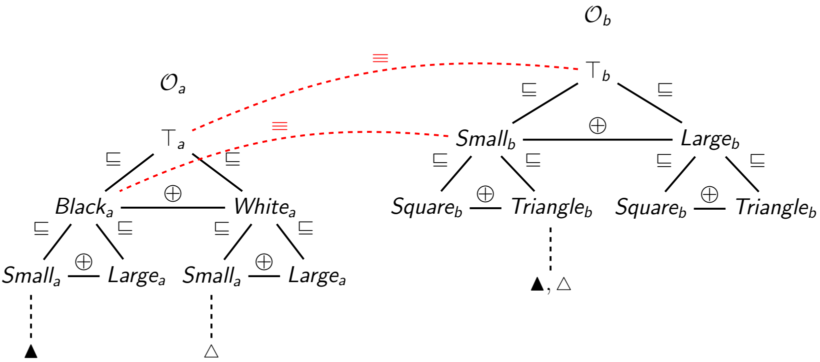
Alignment Repair Game [Euzenat 2014, Euzenat 2017]

- Agents use (different) ontologies
- They use alignments to communicate



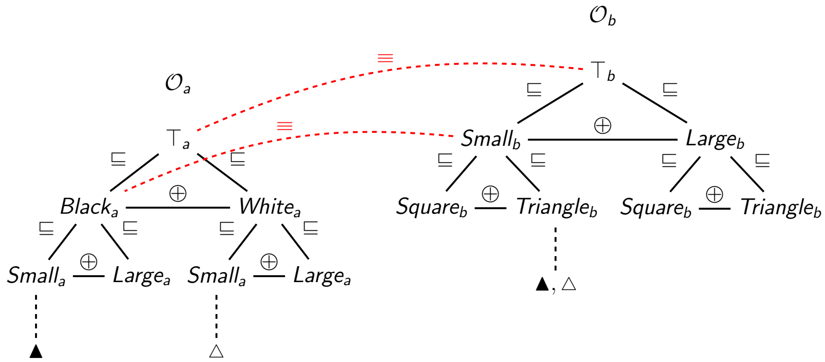
Alignment Repair Game [Euzenat 2014, Euzenat 2017]

- Alignments are computed by ontology matching tools
-



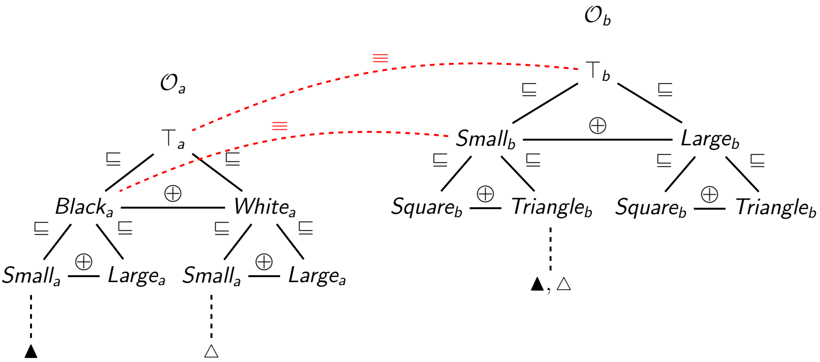
Alignment Repair Game [Euzenat 2014, Euzenat 2017]

- Alignments are computed by ontology matching tools
- Alignments may be incorrect or incomplete



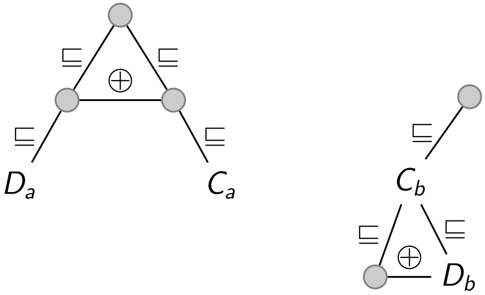
Alignment Repair Game [Euzenat 2014, Euzenat 2017]

- Alignments are computed by ontology matching tools
- Alignments may be incorrect or incomplete
- Alignment Repair Game (ARG)

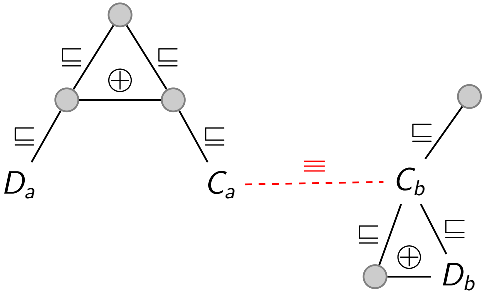


Alignment Repair Game [Euzenat 2014, Euzenat 2017]

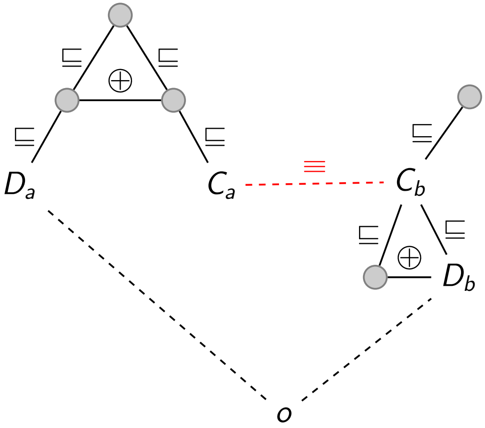
Alignment Repair Game [Euzenat 2014, Euzenat 2017]



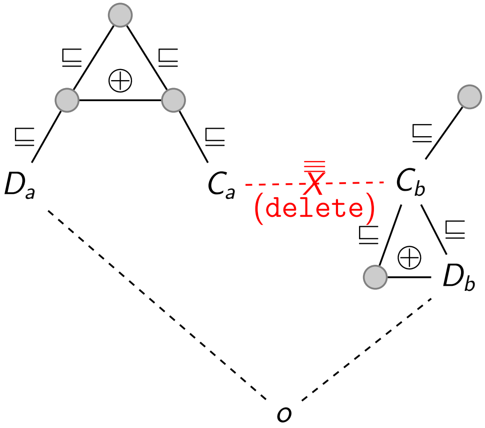
Alignment Repair Game [Euzenat 2014, Euzenat 2017]



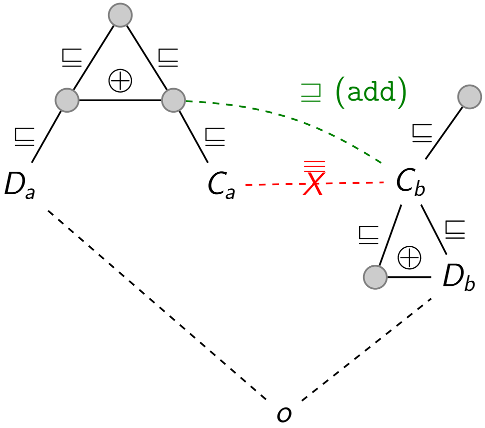
Alignment Repair Game [Euzenat 2014, Euzenat 2017]



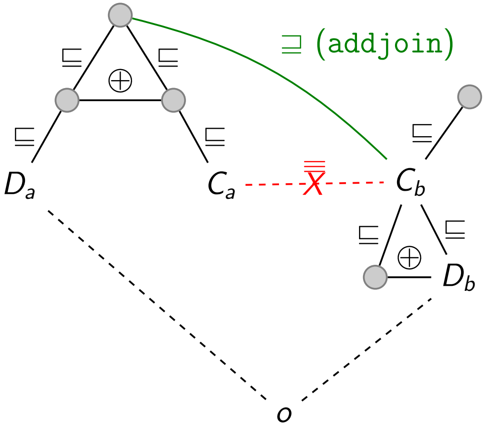
Alignment Repair Game [Euzenat 2014, Euzenat 2017]



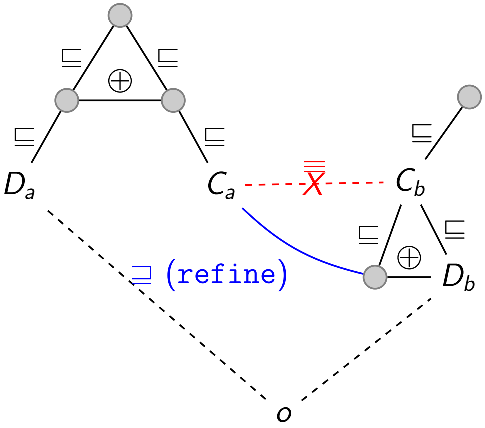
Alignment Repair Game [Euzenat 2014, Euzenat 2017]



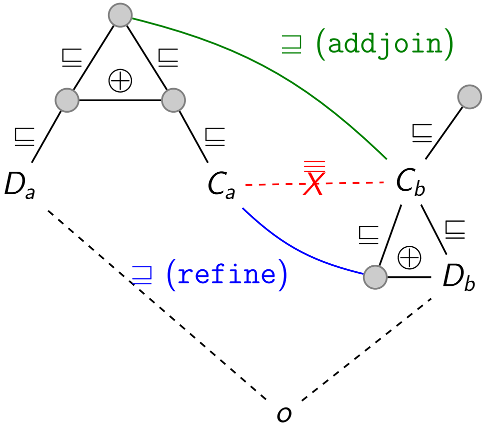
Alignment Repair Game [Euzenat 2014, Euzenat 2017]



Alignment Repair Game [Euzenat 2014, Euzenat 2017]



Alignment Repair Game [Euzenat 2014, Euzenat 2017]



Objective

Objective

- Experimental results show that if agents play ARG:

Objective

- Experimental results show that if agents play ARG:
  - agents converge towards successful communication,

Objective

- Experimental results show that if agents play ARG:
  - agents converge towards successful communication,
  - and improve their alignments.

Objective

- Experimental results show that if agents play ARG:
  - agents converge towards successful communication,
  - and improve their alignments.
- But it is an open question whether the adaptation operators are **formally correct**, **complete** or **redundant**?

Contribution

We introduce a formal framework based on DEL that

Contribution

We introduce a formal framework based on DEL that

- expresses ontologies and alignments,

Contribution

We introduce a formal framework based on DEL that

- expresses ontologies and alignments,
- models the adaptation operators.

Contribution

We introduce a formal framework based on DEL that

- expresses ontologies and alignments,
- models the adaptation operators.

To formally establish the correctness, partial redundancy and incompleteness of the adaptation operators.

Dynamic Epistemic Ontology Logic (DEOL): Syntax

Definition (DEOL Syntax)

The *syntax*, $\mathcal{L}_{DEOL}$, of (multi-agent) DEOL is defined by:

$$\phi ::=$$

Dynamic Epistemic Ontology Logic (DEOL): Syntax

Definition (DEOL Syntax)

The *syntax*, $\mathcal{L}_{DEOL}$, of (multi-agent) DEOL is defined by:

$$\phi ::= C(o) \mid CRD$$

$$R \in \{\sqsubseteq, \equiv, \oplus\}$$

Dynamic Epistemic Ontology Logic (DEOL): Syntax

Definition (DEOL Syntax)

The *syntax*, $\mathcal{L}_{DEOL}$, of (multi-agent) DEOL is defined by:

$$\phi ::= C(o) \mid CRD \mid \phi \wedge \psi \mid \neg \phi$$

$$R \in \{\sqsubseteq, \equiv, \oplus\}$$

Dynamic Epistemic Ontology Logic (DEOL): Syntax

Definition (DEOL Syntax)

The *syntax*, $\mathcal{L}_{DEOL}$, of (multi-agent) DEOL is defined by:

$$\phi ::= C(o) \mid CRD \mid \phi \wedge \psi \mid \neg\phi \mid K_a\phi \mid B_a\phi$$

$$R \in \{\sqsubseteq, \equiv, \oplus\}$$

Dynamic Epistemic Ontology Logic (DEOL): Syntax

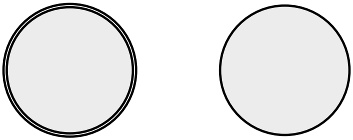
Definition (DEOL Syntax)

The *syntax*, $\mathcal{L}_{DEOL}$, of (multi-agent) DEOL is defined by:

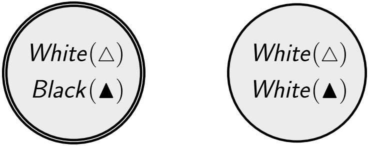
$$\phi ::= C(o) \mid CRD \mid \phi \wedge \psi \mid \neg\phi \mid K_a\phi \mid B_a\phi \mid [\dagger\phi]\psi$$

$$R \in \{\sqsubseteq, \equiv, \oplus\}, \quad \dagger \in \{!, \Uparrow\}$$

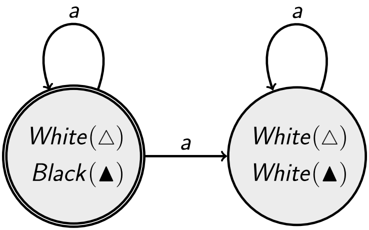
Dynamic Epistemic Ontology Logic (DEOL): Models



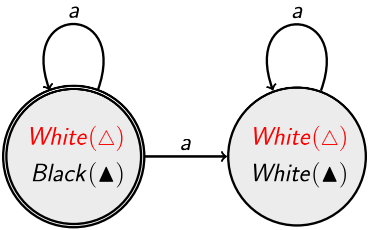
Dynamic Epistemic Ontology Logic (DEOL): Models



Dynamic Epistemic Ontology Logic (DEOL): Models

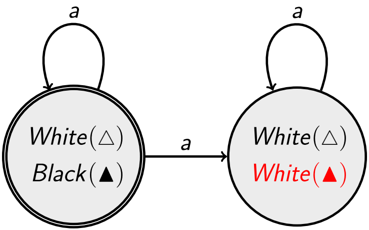


Dynamic Epistemic Ontology Logic (DEOL): Models



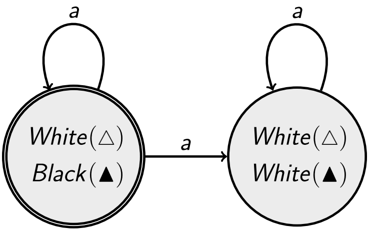
$K_a White(\triangle)$ "Agent a knows that △ belongs to the class White"

Dynamic Epistemic Ontology Logic (DEOL): Models



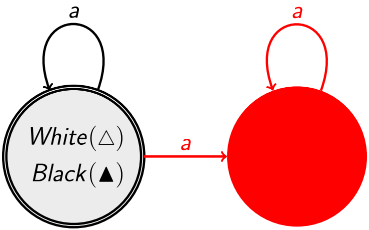
$B_a White(\blacktriangle)$ "Agent a believes that ▲ belongs to the class White"

Dynamic Epistemic Ontology Logic (DEOL): Models



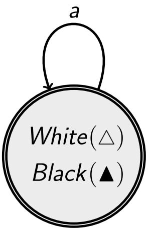
$[!Black(\blacktriangle)]K_a Black(\blacktriangle)$ "After announcing $!Black(\blacktriangle)$, a knows this"

Dynamic Epistemic Ontology Logic (DEOL): Models



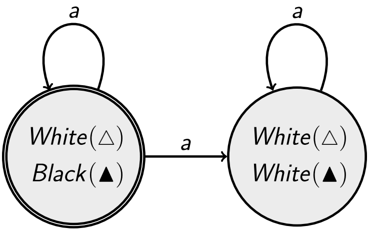
$[!Black(\blacktriangle)]K_a Black(\blacktriangle)$ "After announcing $!Black(\blacktriangle)$, a knows this"

Dynamic Epistemic Ontology Logic (DEOL): Models



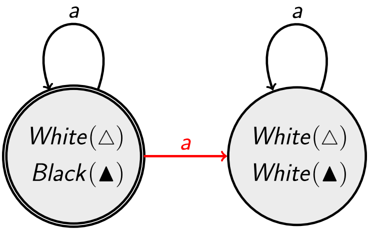
$[!Black(\blacktriangle)]K_a Black(\blacktriangle)$ "After announcing $!Black(\blacktriangle)$, a knows this"

Dynamic Epistemic Ontology Logic (DEOL): Models



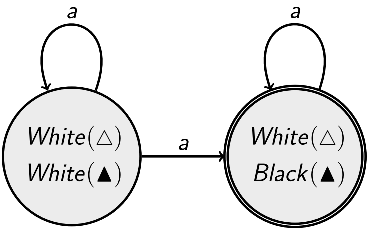
$[\uparrow Black(\blacktriangle)]B_a Black(\blacktriangle)$ "After upgrading with $\uparrow Black(\blacktriangle)$, a believes this"

Dynamic Epistemic Ontology Logic (DEOL): Models



$[\uparrow Black(\blacktriangle)]B_a Black(\blacktriangle)$ "After upgrading with $\uparrow Black(\blacktriangle)$, a believes this"

Dynamic Epistemic Ontology Logic (DEOL): Models



$[\uparrow Black(\blacktriangle)] B_a Black(\blacktriangle)$ "After upgrading with $\uparrow Black(\blacktriangle)$, a believes this"

Dynamic Epistemic Ontology Logic (DEOL): Semantics

Dynamic Epistemic Ontology Logic (DEOL): Semantics

$$\mathcal{M}, w \models C(x) \quad \text{iff } x \in C^{I_w}$$

$$\mathcal{M}, w \models C \sqsubseteq D \quad \text{iff } C^{I_w} \subseteq D^{I_w}$$

$$\mathcal{M}, w \models C \equiv D \quad \text{iff } C^{I_w} = D^{I_w}$$

$$\mathcal{M}, w \models C \oplus D \quad \text{iff } C^{I_w} \cap D^{I_w} = \varnothing$$

Dynamic Epistemic Ontology Logic (DEOL): Semantics

$$\mathcal{M}, w \models C(x) \quad \text{iff } x \in C^{I_w}$$

$$\mathcal{M}, w \models C \sqsubseteq D \quad \text{iff } C^{I_w} \subseteq D^{I_w}$$

$$\mathcal{M}, w \models C \equiv D \quad \text{iff } C^{I_w} = D^{I_w}$$

$$\mathcal{M}, w \models C \oplus D \quad \text{iff } C^{I_w} \cap D^{I_w} = \emptyset$$

$$\mathcal{M}, w \models \phi \wedge \psi \quad \text{iff } \mathcal{M}, w \models \phi \text{ and } \mathcal{M}, w \models \psi$$

$$\mathcal{M}, w \models \neg\phi \quad \text{iff } \mathcal{M}, w \not\models \phi$$

Dynamic Epistemic Ontology Logic (DEOL): Semantics

$\mathcal{M}, w \models C(x)$ iff $x \in C^{I_w}$

$\mathcal{M}, w \models C \sqsubseteq D$ iff $C^{I_w} \subseteq D^{I_w}$

$\mathcal{M}, w \models C \equiv D$ iff $C^{I_w} = D^{I_w}$

$\mathcal{M}, w \models C \oplus D$ iff $C^{I_w} \cap D^{I_w} = \varnothing$

$\mathcal{M}, w \models \phi \wedge \psi$ iff $\mathcal{M}, w \models \phi$ and $\mathcal{M}, w \models \psi$

$\mathcal{M}, w \models \neg\phi$ iff $\mathcal{M}, w \not\models \phi$

$\mathcal{M}, w \models K_a\phi$ iff $\forall v$ s.t. $w \sim_a v$: $\mathcal{M}, v \models \phi$

$\mathcal{M}, w \models B_a\phi$ iff $\forall v$ s.t. $w \rightarrow_a v$: $\mathcal{M}, v \models \phi$

Dynamic Epistemic Ontology Logic (DEOL): Semantics

$$\mathcal{M}, w \models C(x) \quad \text{iff } x \in C^{I_w}$$

$$\mathcal{M}, w \models C \sqsubseteq D \quad \text{iff } C^{I_w} \subseteq D^{I_w}$$

$$\mathcal{M}, w \models C \equiv D \quad \text{iff } C^{I_w} = D^{I_w}$$

$$\mathcal{M}, w \models C \oplus D \quad \text{iff } C^{I_w} \cap D^{I_w} = \varnothing$$

$$\mathcal{M}, w \models \phi \wedge \psi \quad \text{iff } \mathcal{M}, w \models \phi \text{ and } \mathcal{M}, w \models \psi$$

$$\mathcal{M}, w \models \neg\phi \quad \text{iff } \mathcal{M}, w \not\models \phi$$

$$\mathcal{M}, w \models K_a\phi \quad \text{iff } \forall v \text{ s.t. } w \sim_a v\text{: } \mathcal{M}, v \models \phi$$

$$\mathcal{M}, w \models B_a\phi \quad \text{iff } \forall v \text{ s.t. } w \rightarrow_a v\text{: } \mathcal{M}, v \models \phi$$

$$\mathcal{M}, w \models [!\phi]\psi \quad \text{iff } \mathcal{M}^{!\phi}, w \models \psi$$

$$\mathcal{M}, w \models [\uparrow \phi]\psi \quad \text{iff } \mathcal{M}^{\uparrow\phi}, w \models \psi$$

Translation

ARG state (s)

Translation

$$\text{ARG state } (s) \xrightarrow{z} \text{DEOL axioms}$$

Translation

$$\text{ARG state } (s) \xrightarrow{z} \text{DEOL axioms} \xrightarrow{I} \text{DEOL models } (\mathcal{M})$$

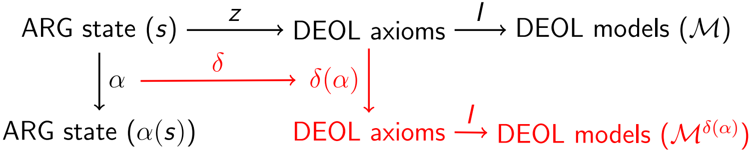
Translation

$$\text{ARG state } (s) \xrightarrow{z} \text{DEOL axioms} \xrightarrow{I} \text{DEOL models } (\mathcal{M})$$

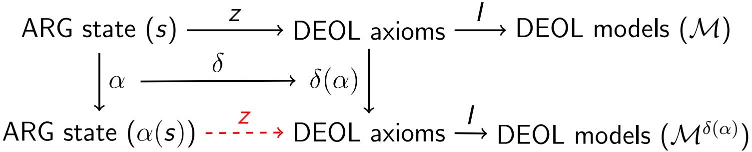
$$\downarrow \alpha$$

$$\text{ARG state } (\alpha(s))$$

Translation



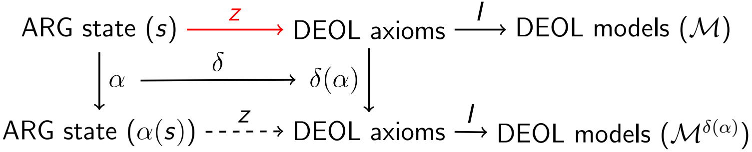
Translation



Semantics of ARG states

$$
\begin{array}{ccccc}
\text{ARG state } (s) & \xrightarrow{\color{red}{z}} & \text{DEOL axioms} & \xrightarrow{I} & \text{DEOL models } (\mathcal{M}) \\
\downarrow \alpha & \xrightarrow{\delta} & \delta(\alpha) \downarrow & & \\
\text{ARG state } (\alpha(s)) & \overset{z}{\dashrightarrow} & \text{DEOL axioms} & \xrightarrow{I} & \text{DEOL models } (\mathcal{M}^{\delta(\alpha)})
\end{array}
$$

Semantics of ARG states



Ontology Knowledge (OK) $\mathcal{O}_a$ is known to agent a

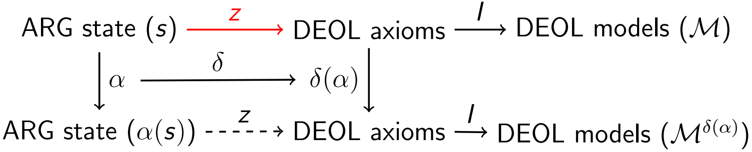
Semantics of ARG states

$$
\begin{array}{ccccc}
\text{ARG state } (s) & \xrightarrow{z} & \text{DEOL axioms} & \xrightarrow{I} & \text{DEOL models } (\mathcal{M}) \\
\downarrow \alpha & \xrightarrow{\delta} & \delta(\alpha) \downarrow & & \\
\text{ARG state } (\alpha(s)) & \overset{z}{\dashrightarrow} & \text{DEOL axioms} & \xrightarrow{I} & \text{DEOL models } (\mathcal{M}^{\delta(\alpha)})
\end{array}
$$

Ontology Knowledge (OK) $\mathcal{O}_a$ is known to agent a

Alignment Belief (AB) A_{ab} is believed by agents a and b

Semantics of ARG states

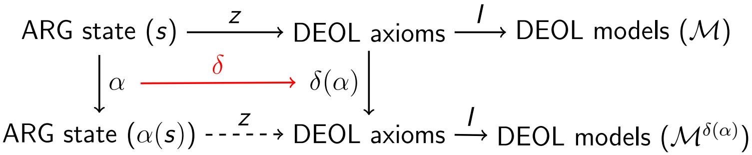


Ontology Knowledge (OK) $\mathcal{O}_a$ is known to agent a

Alignment Belief (AB) A_{ab} is believed by agents a and b

Public Signature Awareness (PSA) All signatures are known to all agents

Dynamics

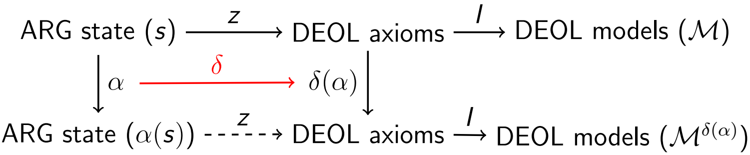

ARG state (s)
z
DEOL axioms
I
DEOL models $(\mathcal{M})$
α
δ
$\delta(\alpha)$
ARG state $(\alpha(s))$
z
DEOL axioms
I
DEOL models $(\mathcal{M}^{\delta(\alpha)})$

Dynamics

$$
\begin{array}{ccccc}
\text{ARG state } (s) & \xrightarrow{\;z\;} & \text{DEOL axioms} & \xrightarrow{\;I\;} & \text{DEOL models } (\mathcal{M}) \\
\downarrow \alpha & \color{red}{\xrightarrow{\;\delta\;}} & \delta(\alpha) \downarrow & & \\
\text{ARG state } (\alpha(s)) & \overset{z}{\dashrightarrow} & \text{DEOL axioms} & \xrightarrow{\;I\;} & \text{DEOL models } (\mathcal{M}^{\delta(\alpha)})
\end{array}
$$

We model each round of ARG by:

Dynamics



We model each round of ARG by:

$$!C_b(o);\ \text{if}\ \neg C_a(o)\ \text{then}\ \delta(\alpha_{\langle C_a, C_b, \sqsupseteq \rangle(o)})$$

Operator Semantics

For failing correspondence $\langle C_a, C_b, \sqsupseteq \rangle \in A_{ab}$ with object o, the adaptation operators are defined on DEOL as follows:

Operator Semantics

For failing correspondence $\langle C_a, C_b, \sqsupseteq \rangle \in A_{ab}$ with object o, the adaptation operators are defined on DEOL as follows:

$$\delta(\texttt{delete}) = \uparrow(C_a \not\sqsupseteq C_b)$$

Operator Semantics

For failing correspondence $\langle C_a, C_b, \sqsupseteq \rangle \in A_{ab}$ with object o, the adaptation operators are defined on DEOL as follows:

$$\delta(\texttt{delete}) = \uparrow(C_a \not\sqsupseteq C_b)$$
$$\delta(\texttt{add}) = \uparrow(C_a \not\sqsupseteq C_b \ \wedge\ C_a^{sup} \sqsupseteq C_b)$$

Operator Semantics

For failing correspondence $\langle C_a, C_b, \sqsupseteq \rangle \in A_{ab}$ with object o, the adaptation operators are defined on DEOL as follows:

$$\delta(\texttt{delete}) = \uparrow(C_a \not\sqsupseteq C_b)$$
$$\delta(\texttt{add}) = \uparrow(C_a \not\sqsupseteq C_b \;\; \wedge \; C_a^{sup} \sqsupseteq C_b)$$
$$\delta(\texttt{addjoin}) = \uparrow(C_a \not\sqsupseteq C_b \;\; \wedge \; C_a^{supO} \sqsupseteq C_b)$$

Operator Semantics

For failing correspondence $\langle C_a, C_b, \sqsupseteq \rangle \in A_{ab}$ with object o, the adaptation operators are defined on DEOL as follows:

$$\begin{aligned}
\delta(\texttt{delete}) &= \uparrow(C_a \not\sqsupseteq C_b) \\
\delta(\texttt{add}) &= \uparrow(C_a \not\sqsupseteq C_b \ \wedge\ C_a^{sup} \sqsupseteq C_b) \\
\delta(\texttt{addjoin}) &= \uparrow(C_a \not\sqsupseteq C_b \ \wedge\ C_a^{supO} \sqsupseteq C_b) \\
\delta(\texttt{refine}) &= \uparrow(C_a \not\sqsupseteq C_b \ \wedge\ \bigwedge\{C_a \sqsupseteq C_b^{sub}\})
\end{aligned}$$

Operator Semantics

For failing correspondence $\langle C_a, C_b, \sqsupseteq \rangle \in A_{ab}$ with object o, the adaptation operators are defined on DEOL as follows:

$$\begin{aligned}
\delta(\texttt{delete}) &= \uparrow (C_a \not\sqsupseteq C_b) \\
\delta(\texttt{add}) &= \uparrow (C_a \not\sqsupseteq C_b \quad \wedge \; C_a^{sup} \sqsupseteq C_b) \\
\delta(\texttt{addjoin}) &= \uparrow (C_a \not\sqsupseteq C_b \quad \wedge \; C_a^{supO} \sqsupseteq C_b) \\
\delta(\texttt{refine}) &= \uparrow (C_a \not\sqsupseteq C_b \quad \wedge \; \bigwedge \{C_a \sqsupseteq C_b^{sub}\}) \\
\delta(\texttt{refadd}) &= \texttt{addjoin} \qquad \wedge \; \texttt{refine}
\end{aligned}$$

Formal Properties of the Adaptation Operators

Formal Properties of the Adaptation Operators

- All but the `add` operator are correct

Formal Properties of the Adaptation Operators

- All but the `add` operator are correct
- `delete`, `addjoin` and `refine` are redundant for one agent

Formal Properties of the Adaptation Operators

- All but the `add` operator are correct
- `delete`, `addjoin` and `refine` are redundant for one agent
- All operators are incomplete

Formal Properties of the Adaptation Operators

- All **but the `add` operator** are correct
- `delete`, `addjoin` and `refine` are redundant for one agent
- **All operators are incomplete**

add is incorrect

add is incorrect

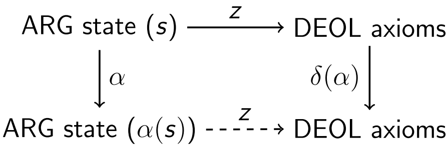
Definition (Correctness)

Adaptation operator α is *correct* iff $\forall s$: $(z(s))^{\delta(\alpha)} \models z(\alpha(s))$.

add is incorrect

Definition (Correctness)

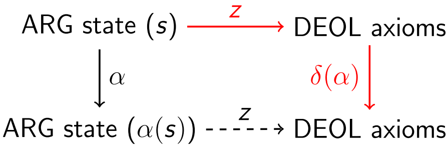
Adaptation operator α is *correct* iff $\forall s$: $(z(s))^{\delta(\alpha)} \models z(\alpha(s))$.



add is incorrect

Definition (Correctness)

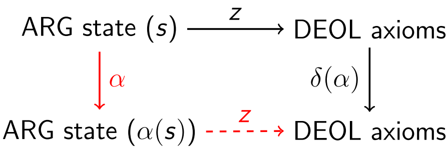
Adaptation operator α is *correct* iff $\forall s$: $(z(s))^{\delta(\alpha)} \models z(\alpha(s))$.



add is incorrect

Definition (Correctness)

Adaptation operator α is *correct* iff $\forall s$: $(z(s))^{\delta(\alpha)} \models z(\alpha(s))$.



add is incorrect

Definition (Correctness)

Adaptation operator α is *correct* iff $\forall s$: $(z(s))^{\delta(\alpha)} \models z(\alpha(s))$.

Proposition

The adaptation operator `add` *is incorrect.*

add is incorrect

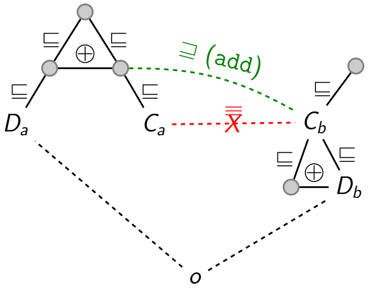
Definition (Correctness)

Adaptation operator α is *correct* iff $\forall s$: $(z(s))^{\delta(\alpha)} \models z(\alpha(s))$.

Proposition

The adaptation operator add *is incorrect.*

Proof (sketch).



add is incorrect

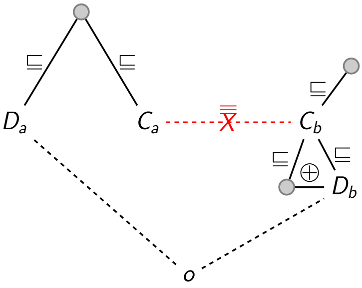
Definition (Correctness)

Adaptation operator α is *correct* iff $\forall s$: $(z(s))^{\delta(\alpha)} \models z(\alpha(s))$.

Proposition

The adaptation operator add *is incorrect.*

Proof (sketch).



add is incorrect

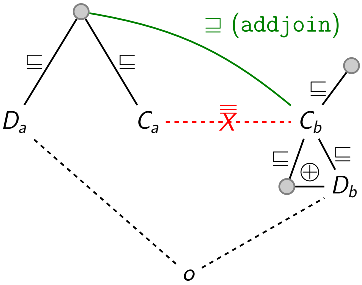
Definition (Correctness)

Adaptation operator α is *correct* iff $\forall s$: $(z(s))^{\delta(\alpha)} \models z(\alpha(s))$.

Proposition

The adaptation operator add *is incorrect.*

Proof (sketch).



All operators are incomplete

All operators are incomplete

Definition (Completeness)

Adaptation operator α is complete if and only if $\forall s$: $z(\alpha(s)) \models (z(s))^{\delta(\alpha)}$.

All operators are incomplete

Definition (Completeness)

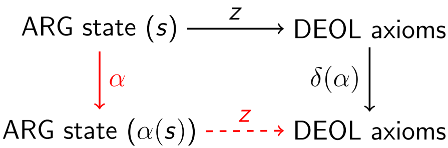
Adaptation operator α is complete if and only if $\forall s$: $z(\alpha(s)) \models (z(s))^{\delta(\alpha)}$.

$$
\begin{array}{ccc}
\text{ARG state } (s) & \xrightarrow{\;z\;} & \text{DEOL axioms} \\
\downarrow \alpha & & \delta(\alpha) \downarrow \\
\text{ARG state } (\alpha(s)) & \dashrightarrow^{z} & \text{DEOL axioms}
\end{array}
$$

All operators are incomplete

Definition (Completeness)

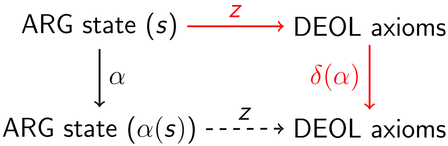
Adaptation operator α is complete if and only if $\forall s$: $z(\alpha(s)) \models (z(s))^{\delta(\alpha)}$.



All operators are incomplete

Definition (Completeness)

Adaptation operator α is complete if and only if $\forall s$: $z(\alpha(s)) \models (z(s))^{\delta(\alpha)}$.



All operators are incomplete

Definition (Completeness)

Adaptation operator α is complete if and only if $\forall s$: $z(\alpha(s)) \models (z(s))^{\delta(\alpha)}$.

Proposition

All adaptation operators are incomplete.

All operators are incomplete

Definition (Completeness)

Adaptation operator α is complete if and only if $\forall s$: $z(\alpha(s)) \models (z(s))^{\delta(\alpha)}$.

Proposition

All adaptation operators are incomplete.

Proof.

After announcing $!C_b(o)$ the agents acquire knowledge $K_a(C_b(o))$ that is discarded in ARG. □

Conclusion

Conclusion

- We developed a theoretical framework for knowledge and belief evolution in ARG

Conclusion

- We developed a theoretical framework for knowledge and belief evolution in ARG
- Formally defined correctness, redundancy and completeness

Conclusion

- We developed a theoretical framework for knowledge and belief evolution in ARG
- Formally defined correctness, redundancy and completeness
- Some adaptation operators are incorrect or redundant, and all are incomplete

Conclusion

- We developed a theoretical framework for knowledge and belief evolution in ARG
- Formally defined correctness, redundancy and completeness
- Some adaptation operators are incorrect or redundant, and all are incomplete
- Agents in ARG do not posses full logical behavior

Thank you!