Reasoning with Attributed Description Logics

Markus Krötzsch  Maximilian Marx  Ana Ozaki  Veronika Thost

Center for Advancing Electronics Dresden (cfaed), TU Dresden

30th International Workshop on Description Logics

Full paper: https://iccl.inf.tu-dresden.de/web/Inproceedings3154
Why Attributed DLs?

- Liz Taylor & Richard Burton in Wikidata:

  ![Diagram](image_url)

  - Taylor: start 1964, end 1974
  - Burton: start 1975, end 1976

  - Edges may occur with multiple distinct annotations
  - Annotations: finite attribute-value sets, attached to concept & role names
  - E.g., spouse is symmetric, so inverses should coincide on
  - X: ⌊ ⌋ (spouse @ X ⊑ spouse − @ ⌊ start: X, end: X ⌋)
Why Attributed DLs?

- Liz Taylor & Richard Burton in Wikidata:

  ![Diagram of Liz Taylor & Richard Burton relationship]

  - edges may occur with multiple distinct annotations
  - annotations: finite attribute–value sets, attached to concept & role names
Why Attributed DLs?

- Liz Taylor & Richard Burton in Wikidata:
  - taylor
  - burton
  - start: 1964, end: 1974
  - spouse
  - start: 1975, end: 1976

- edges may occur with multiple distinct annotations
- annotations: finite attribute–value sets, attached to concept & role names
- e.g., spouse is symmetric, so inverses should coincide on start & end

\[ X : [ ] \quad (\text{spouse} @ X \sqsubseteq \text{spouse}^{-} @ [\text{start} : X.\text{start}, \text{end} : X.\text{end}]) \]
Specifiers: constraining annotations

- two flavours of annotations: open & closed specifiers

consider \( \text{spouse(taylor, burton)} @ [\text{start} : 1964, \text{end} : 1974] \)

\[
\begin{align*}
[ & ] & \checkmark & [ & ] & \xmark \\
[\text{start} : 1964] & \checkmark & [\text{start} : 1964] & \xmark \\
[\text{start} : 1964, \text{end} : 1974] & \checkmark & [\text{start} : 1964, \text{end} : 1974] & \checkmark \\
[\text{start} : 1964, \text{loc} : \text{Montreal}] & \xmark & [\text{start} : 1964, \text{end} : 1974, \text{loc} : \ast] & \checkmark \\
[\text{start} : 1964, \text{end} : \ast] & \checkmark & [\text{start} : 1964, \text{end} : 1974, \text{loc} : \ast] & \xmark 
\end{align*}
\]

- simplification: instead of \( C @ [ \ ] \), write \( C \)
Attributed DL axioms

Axioms may use variables in annotation positions:

- all variables are universally quantified

\[
\text{spouse}@X \sqsubseteq \text{spouse}^-@X
\]
Attributed DL axioms

Axioms may use variables in annotation positions:

- all variables are universally quantified

\[ \text{spouse}@X \sqsubseteq \text{spouse}^-@X \]

- variables may be constrained by using a specifier

\[ X : [\text{start} : 1964] \quad (\text{spouse}@X \sqsubseteq \text{spouse}^-@X) \]
Attributed DL axioms

Axioms may use variables in annotation positions:

- all variables are universally quantified

\[ \text{spouse}@X \sqsubseteq \text{spouse}^-@X \]

- variables may be constrained by using a specifier

\[ X : [\text{start : 1964}] \quad (\text{spouse}@X \sqsubseteq \text{spouse}^-@X) \]

- annotations may refer to assignments in other annotations

\[ X : [\text{start : 1964}], \quad Y : [\text{start : } X\text{.start, end : } Y\text{.end}] \]

\[ (\text{spouse}@X \sqsubseteq \text{spouse}^-@X) \]
Attributed DL axioms

Axioms may use variables in annotation positions:

- all variables are universally quantified

\[
\text{spouse}@X \sqsubseteq \text{spouse}^- @ X
\]

- variables may be constrained by using a specifier

\[
X : [\text{start} : 1964] \quad (\text{spouse}@X \sqsubseteq \text{spouse}^- @ X)
\]

- annotations may refer to assignments in other annotations

\[
X : [\text{start} : 1964], Y : [\text{start} : X.\text{start}, \text{end} : Y.\text{end}] \\
(\text{spouse}@X \sqsubseteq \text{spouse}^- @ X)
\]

- note: cyclic references are allowed
## Complexity of Reasoning in Attributed DLs

<table>
<thead>
<tr>
<th>DL</th>
<th>ground</th>
<th>restricted</th>
<th>unrestricted</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\mathcal{EL}@)</td>
<td>(\text{PTIME})</td>
<td>(\text{PTIME}/\text{PSPACE}-\text{hard}^\ast)</td>
<td>(\text{ExpTime})</td>
</tr>
<tr>
<td>(\mathcal{ALCH}@)</td>
<td>(\text{ExpTime})</td>
<td>(\text{ExpTime})</td>
<td>(2\text{ExpTime})</td>
</tr>
<tr>
<td>(\mathcal{SROIQ}@)</td>
<td>(\text{N2ExpTime})</td>
<td>(\text{N2ExpTime})</td>
<td>(\text{N2ExpTime})</td>
</tr>
<tr>
<td>(\mathcal{EL}@+)</td>
<td>(\text{PTIME})</td>
<td>(\text{ExpTime})</td>
<td>undecidable</td>
</tr>
<tr>
<td>(\mathcal{ALCH}@+)</td>
<td>(\text{ExpTime})</td>
<td>(2\text{ExpTime})</td>
<td>undecidable</td>
</tr>
</tbody>
</table>

- except for \(\text{PSPACE}\)-hardness, bounds are tight
- Nominals require special handling (bounds on domain size)
Reasoning in Attributed DLs

Reasoning for ground KBs

Introduce fresh concept/role names for each annotated concept/role

- yields polynomially larger KB in underlying, classical DL:

\[
\text{spouse}(\text{taylor, burton})_{\text{start: } 1964, \text{end: } 1974}
\]

\[
\text{spouse}_{\text{start: } 1964} \sqsubseteq \text{spouse}^{-}_{\text{start: } 1964}
\]

\[
\text{spouse}[\text{start: } 1964, \text{end: } 1974](\text{taylor, burton})
\]

\[
\text{spouse}_{\text{start: } 1964} \sqsubseteq \text{spouse}^{-}_{\text{start: } 1964}
\]
Reasoning for ground KBs

Introduce fresh concept/role names for each annotated concept/role

- yields polynomially larger KB in underlying, classical DL:

\[
\text{spouse}(\text{taylor, burton})[\text{start : 1964, end : 1974}] \quad (1)
\]
\[
\text{spouse}[\text{start : 1964}] \sqsubseteq \text{spouse}^-[@[\text{start : 1964}]] \quad (2)
\]

\[
\Rightarrow \quad \text{spouse}_{[\text{start:1964,end:1974}]}(\text{taylor, burton})
\]
\[
\text{spouse}_{[\text{start:1964}]} \sqsubseteq \text{spouse}^-_{[\text{start:1964}]}
\]

- interactions between open & closed specifiers: (1), (2) entails

\[
\text{spouse}(\text{burton, taylor})[\text{start : 1964}],
\]

but we do not get \[
\text{spouse}^-_{[\text{start:1964}]}(\text{burton, taylor})
\]
Introduce fresh concept/role names for each annotated concept/role

- yields polynomially larger KB in underlying, classical DL:

\[
\text{spouse}(\text{taylor}, \text{burton}) \sqsubseteq [\text{start} : 1964, \text{end} : 1974] \quad (1)
\]

\[
\text{spouse} \sqsubseteq [\text{start} : 1964] \sqsubseteq \text{spouse}^\sim \sqsubseteq [\text{start} : 1964] \quad (2)
\]

\[
\text{spouse}[[\text{start:1964,end:1974}]](\text{taylor}, \text{burton}) \sqsubseteq \text{spouse}[[\text{start:1964}]] \sqsubseteq \text{spouse}^\sim[[\text{start:1964}]]
\]

- interactions between open & closed specifiers: (1), (2) entails

\[
\text{spouse}(\text{burton}, \text{taylor}) \sqsubseteq [\text{start} : 1964],
\]

but we do not get \(\text{spouse}[[\text{start:1964}]](\text{burton}, \text{taylor})\)

- axiomatise these inclusions: \(\text{spouse}[[\text{start:1964,end:1974}]] \sqsubseteq \text{spouse}[[\text{start:1964}]]\)
Dealing with non-ground KBs

Transform KB into a ground KB:
  - instantiate each axiom for every possible annotation

\[ \text{spouse}(\text{taylor}, \text{burton}) \@ [\text{start} : 1964] \]
\[ \text{spouse}(\text{taylor}, \text{burton}) \@ [\text{start} : 1974] \]
\[ \overset{\text{\(\rightarrow\)}}{\text{spouse}}_{[\text{start}:1964]}(\text{taylor}, \text{burton}) \]
\[ \overset{\text{\(\rightarrow\)}}{\text{spouse}}_{[\text{start}:1964]} \subseteq \overset{\text{\(\rightarrow\)}}{\text{spouse}}_{[\text{start}:1964]} \]
\[ \overset{\text{\(\rightarrow\)}}{\text{spouse}}_{[\text{start}:1974]} \subseteq \overset{\text{\(\rightarrow\)}}{\text{spouse}}_{[\text{start}:1974]} \]

Unfortunately, the grounding is exponential in the size of KB:

\[ \text{spouse}@X \sqsubseteq \text{spouse}^{-}@X \]
\[ \overset{\text{\(\rightarrow\)}}{\text{spouse}}_{[\text{start}:1974]}(\text{taylor}, \text{burton}) \]
Dealing with non-ground KBs

Transform KB into a ground KB:

- instantiate each axiom for every possible annotation

\begin{align*}
\text{spouse}(\text{taylor}, \text{burton})@\ [\text{start}: 1964] & \quad \text{spouse}@X \sqsubseteq \text{spouse}^-@X \\
\text{spouse}(\text{taylor}, \text{burton})@\ [\text{start}: 1974] & \\
\end{align*}

\begin{align*}
\leadsto \quad & \text{spouse}_{[\text{start}:1964]}(\text{taylor}, \text{burton}) \quad \text{spouse}_{[\text{start}:1974]}(\text{taylor}, \text{burton}) \\
& \text{spouse}_{[\text{start}:1964]} \sqsubseteq \text{spouse}^-_{[\text{start}:1964]} \\
& \text{spouse}_{[\text{start}:1974]} \sqsubseteq \text{spouse}^-_{[\text{start}:1974]}
\end{align*}

- unfortunately, the grounding is exponential in the size of KB:

\begin{align*}
C(a)@\ [ ] & \quad C(a)@[b:b] \\
C@X \cap C@Y \cap C@Z \sqsubseteq C@X
\end{align*}

- syntactic restrictions ensure a polynomial grounding
Regaining Tractability for $\mathcal{EL}_@$

Sufficient conditions for polynomial grounding:

(A) number of variables per axiom is bounded,

(B) number of ‘dots’ $X.a$ is bounded, and

(C) no merging with ‘dots’: if $a : X.b$ occurs in some annotation $S$, then there is no further assignment for $a$ in $S$
Regaining Tractability for $\mathcal{EL}_@$

Sufficient conditions for polynomial grounding:

(A) number of variables per axiom is bounded,

(B) number of ‘dots’ $X.a$ is bounded, and

(C) no merging with ‘dots’: if $a : X.b$ occurs in some annotation $S$, then there is no further assignment for $a$ in $S$

- violating any condition yields intractability for $\mathcal{EL}_@$
- violating (C) results in PSPACE-hardness
Regaining Tractability for $\mathcal{EL}_@$

Sufficient conditions for polynomial grounding:

(A) number of variables per axiom is bounded,

(B) number of ‘dots’ $X.a$ is bounded, and

(C) no merging with ‘dots’: if $a : X.b$ occurs in some annotation $S$, then there is no further assignment for $a$ in $S$

- violating any condition yields intractability for $\mathcal{EL}_@$
- violating (C) results in $\text{PSPACE}$-hardness
- reasoning for $\mathcal{ALCH}_@$ KBs satisfying the conditions is $\text{ExpTime}$-complete
An Undecidable Case

Without restrictions, Attributed DLs with + are undecidable:

- interaction of $X.a$ and $+$ admits an encoding of Existential Rules in quantifier-free attributed $\mathcal{EL}$
- forbidding either $X.a$ or $+$ is sufficient to recover decidability
- practically, $X.a$ is more relevant
An Undecidable Case

Without restrictions, Attributed DLs with + are undecidable:

- interaction of $X.a$ and + admits an encoding of Existential Rules in quantifier-free attributed $\mathcal{EL}$
- forbidding either $X.a$ or + is sufficient to recover decidability
- practically, $X.a$ is more relevant, but + adds expressive power:

$$\text{educatedAt}@\lfloor \text{degree} : + \rfloor \sqsubseteq \text{obtainedDegreeFrom}$$

- decidability results for Existential Rules suggest that a weaker condition may suffice for decidability
An Undecidable Case

Without restrictions, Attributed DLs with + are undecidable:

- interaction of $X.a$ and + admits an encoding of Existential Rules in quantifier-free attributed $\mathcal{EL}$
- forbidding either $X.a$ or + is sufficient to recover decidability
- practically, $X.a$ is more relevant, but + adds expressive power:

$$\text{educatedAt}[@\{\text{degree} : +\}] \subseteq \text{obtainedDegreeFrom}$$

- decidability results for Existential Rules suggest that a weaker condition may suffice for decidability
- Corollary: Attributed DLs (without +) capture Datalog
Summary & Outlook

- **Summary:**
  - we add annotations (sets of attribute–value pairs) to concept and role names
  - specifiers allow to constrain variables in axioms
  - ‘ground and rename’ reasoning approach
  - attributed reasoning is exponentially harder
  - syntactic conditions ensure that we avoid this blowup
Summary & Outlook

- **Summary:**
  - we add annotations (sets of attribute–value pairs) to concept and role names
  - specifiers allow to constrain variables in axioms
  - ‘ground and rename’ reasoning approach
  - attributed reasoning is exponentially harder
  - syntactic conditions ensure that we avoid this blowup

- **Future Work:**
  - data complexities
  - extension to further DL constructs ($\mathcal{EL}^{++}$)
  - annotation-aware reasoning algorithms