An Introduction to Description Logics:

Techniques, Properties, and Applications

NASSLLI, Day 4, Part 1

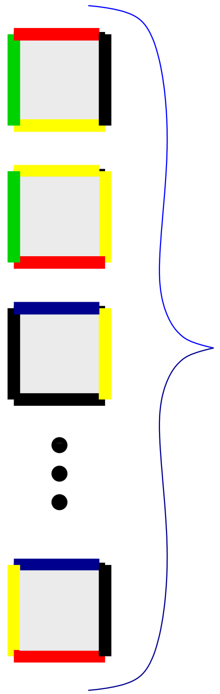
Computational Complexity (ctd)

Uli Sattler

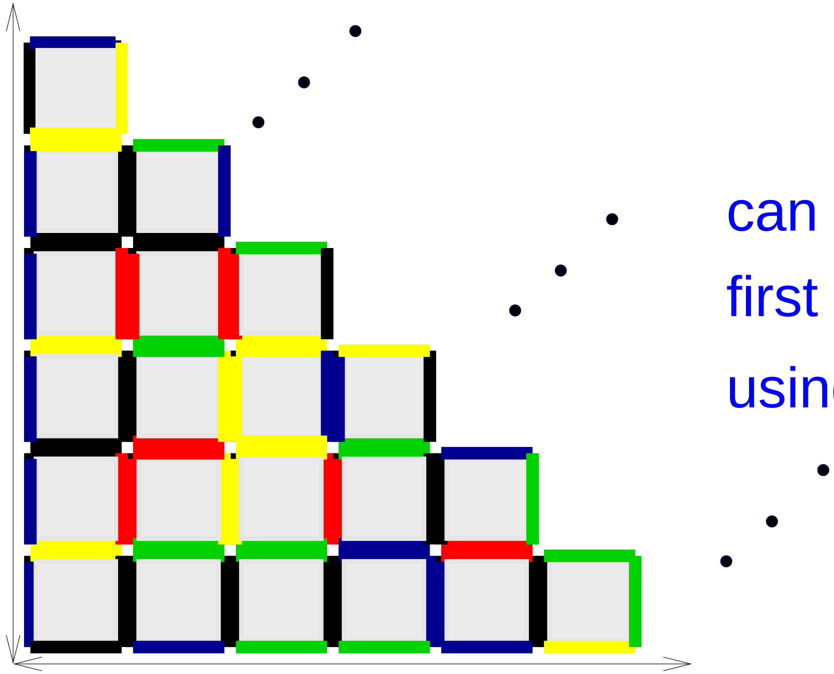
Today

We will discuss

- a few more undecidability/complexity results
- understanding entailments and debugging ontologies



***D*,**
a fixed
set
of
dominoe
types



can we tile the
first quadrant
using ***D*?**

The Classical Domino Problem

Definition: A **domino system** $\mathcal{D} = (D, H, V)$

- set of domino **types** $D = \{D_1, \ldots, D_d\}$, and
- horizontal and vertical **matching conditions** $H \subseteq D \times D$ and $V \subseteq D \times D$

A **tiling** for $\mathcal{D}$ is a (total) function:

$$t : \mathbb{N} \times \mathbb{N} \to D \text{ such that } \begin{array}{l} \langle t(m,n), t(m+1,n)\rangle \in H \text{ and} \\ \langle t(m,n), t(m,n+1)\rangle \in V \end{array}$$

Domino problem: given $\mathcal{D}$, has $\mathcal{D}$ a tiling?

It is well-known that this problem is undecidable [Berger66]

Almost Encoding the Classical Domino Problem in $\mathcal{ALC}$ ✔

For our reduction, we express various **obligations** of the domino problem in $\mathcal{ALC}$ TBox axioms:

① each element carries **exactly one domino type** D_i

⇝ use unary predicate symbol D_i for each domino type and

$$\begin{array}{rcll} \top & \sqsubseteq & D_1 \sqcup \ldots \sqcup D_d & \text{% each element carries a domino type} \\ D_1 & \sqsubseteq & \neg D_2 \sqcap \ldots \sqcap \neg D_d & \text{% but not more than one} \\ D_2 & \sqsubseteq & \neg D_3 \sqcap \ldots \sqcap \neg D_d & \text{% ...} \\ \vdots & & \vdots & \\ D_{d-1} & \sqsubseteq & \neg D_d & \end{array}$$

Almost Encoding the Classical Domino Problem in $\mathcal{ALC}$ ✔

② every element has a horizontal (X-) successor and a vertical (Y-) successor

$$\top \sqsubseteq \exists X.\top \sqcap \exists Y.\top$$

③ every element satisfies D's **horizontal/vertical matching conditions**:

$$\begin{array}{lcl} D_1 & \sqsubseteq & \bigsqcup_{(D_1,D)\in H} \forall X.D \sqcap \bigsqcup_{(D_1,D)\in V} \forall Y.D \\ D_2 & \sqsubseteq & \bigsqcup_{(D_2,D)\in H} \forall X.D \sqcap \bigsqcup_{(D_2,D)\in V} \forall Y.D \\ \vdots & \vdots & \\ D_d & \sqsubseteq & \bigsqcup_{(D_d,D)\in H} \forall X.D \sqcap \bigsqcup_{(D_d,D)\in V} \forall Y.D \end{array}$$

Does this suffice?

No: if yes, $\mathcal{ALC}$ would be undecidable!

④ for each element, its horizontal-vertical-successors **coincide** with their vertical-horizontal-successors & vice versa

$$X \circ Y \sqsubseteq Y \circ X \text{ and } Y \circ X \sqsubseteq X \circ Y$$

Lemma: Let $\mathcal{T}_D$ be the set of axioms ① to ④.
Then $\top$ is satisfiable w.r.t. $\mathcal{T}_D$ iff $\mathcal{D}$ has a tiling.

- since the domino problem is undecidable, this implies undecidability of concept satisfiability w.r.t. TBoxes of $\mathcal{ALC}$ with role chain inclusions
- due to Theorem 2, all other standard reasoning problems are undecidable, too
- Proof:
  1. show that, from a tiling for D, you can construct a model of $\mathcal{T}_D$
  2. show that, from a model $\mathcal{I}$ of $\mathcal{T}_D$, you can construct a tiling for D (tricky because elements in $\mathcal{I}$ can have several X- or Y-successors but we can simply take **the right ones**...)

Let’s do this again!

Let's do this again!

What other constructors can us help to express ④?

A weak form of counting: $(\leq 1r)^{\mathcal{I}} = \{x \mid \text{there is at most one } y \text{ with } (x, y) \in r^{\mathcal{I}}\}$

- counting and complex roles (role chains and role intersection):

$$\top \sqsubseteq (\leq 1X) \sqcap (\leq 1Y) \sqcap (\exists (X \circ Y) \sqcap (Y \circ X).\top)$$

- restricted role chain inclusions (only 1 role on RHS), and counting (an **all** roles):

$$\begin{aligned} \top &\sqsubseteq (\leq 1X) \sqcap (\leq 1Y) \\ X \circ Y &\sqsubseteq r \\ Y \circ X &\sqsubseteq r \\ \top &\sqsubseteq (\leq 1r) \end{aligned}$$

- various others...

Are all DLs in ExpTime?

Earlier, we have claimed that $\mathcal{ALCQI}$, $\mathcal{ALCQO}$, and $\mathcal{ALCIO}$ are all **ExpTime**-complete, i.e., as hard/easy as $\mathcal{ALC}$

Next, we will see that consistency of $\mathcal{ALCQIO}$ ontologies, the extension of $\mathcal{ALC}$ with

- **inverse roles** r^- with $(r^-)^\mathcal{I} = \{(y, x) \mid (x, y) \in r^\mathcal{I}\}$
- the weakest **number restrictions** $(\leq 1r)$ and
- **nominals**, i.e., individual names used as concept names

$\Rightarrow$ is harder, namely **NExpTime-hard**

- this is typical phenomenon where
  - **combination** of otherwise harmless constructors leads to increased complexity

$\mathcal{ALCQIO}$ is NExpTime-hard

We follow hardness proof recipe:

- to show that consistency of $\mathcal{ALCQIO}$ ontologies is **NExpTime-hard**, we
  – find a suitable problem $P' \subseteq M'$ that is **known to be NExpTime-hard** and
  – a reduction from P' to P

The **NExpTime** version of the **domino problem**

Domino Problems

Definition: A **domino system** $\mathcal{D} = (D, H, V)$

- set of domino **types** $D = \{D_1, \ldots, D_d\}$, and
- horizontal and vertical **matching conditions** $H \subseteq D \times D$ and $V \subseteq D \times D$

A **tiling** for $\mathcal{D}$ is a function:

$$t : \mathbb{N} \times \mathbb{N} \rightarrow D \text{ such that } \begin{array}{l} \langle t(m,n), t(m+1,n)\rangle \in H \text{ and} \\ \langle t(m,n), t(m,n+1)\rangle \in V \end{array}$$

Domino problems: ✔ **classical** given $\mathcal{D}$, has $\mathcal{D}$ a tiling?

⇒ well-known that this problem is **undecidable** [Berger66]

☞ **NexpTime** given $\mathcal{D}$, has $\mathcal{D}$ a tiling for $2^n \times 2^n$ square?

⇒ well-known that this problem is **NExpTime-hard**

Reduction of NExpTime Domino Problem to $\mathcal{ALCQIO}$ Consistency

To reduce the NExpTime domino problem to $\mathcal{ALCQIO}$ consistency, we need to

- define a mapping π from **domino problems** to $\mathcal{ALCQIO}$ **ontologies** such that
- D has an $2^n \times 2^n$ mapping iff $\pi(D)$ is consistent and
- size of $\pi(D)$ is polynomial in n

Again, we express various **obligations** of the domino problem in $\mathcal{ALC}$ axioms:

① each element carries **exactly one domino type** D_i

⇝ use unary predicate symbol D_i for each domino type and

$$
\begin{aligned}
\top &\sqsubseteq D_1 \sqcup \ldots \sqcup D_d && \text{\% each element carries a domino type} \\
D_1 &\sqsubseteq \neg D_2 \sqcap \ldots \sqcap \neg D_d && \text{\% but not more than one} \\
D_2 &\sqsubseteq \neg D_3 \sqcap \ldots \sqcap \neg D_d && \text{\% ...} \\
&\vdots \quad \vdots \\
D_{d-1} &\sqsubseteq \neg D_d
\end{aligned}
$$

Mapping a Domino System into an $\mathcal{ALCQIO}$ Ontology

② every element has a horizontal (X-) successor and a vertical (Y-) successor

$$\top \sqsubseteq \exists X.\top \sqcap \exists Y.\top$$

③ every element satisfies D's **horizontal/vertical matching conditions**:

$$
\begin{aligned}
D_1 &\sqsubseteq \bigsqcup_{(D_1,D)\in H} \forall X.D \sqcap \bigsqcup_{(D_1,D)\in V} \forall Y.D \\
D_2 &\sqsubseteq \bigsqcup_{(D_2,D)\in H} \forall X.D \sqcap \bigsqcup_{(D_2,D)\in V} \forall Y.D \\
&\vdots \\
D_d &\sqsubseteq \bigsqcup_{(D_d,D)\in H} \forall X.D \sqcap \bigsqcup_{(D_d,D)\in V} \forall Y.D
\end{aligned}
$$

Does this suffice?
I.e., does D have a $2^n \times 2^n$ tiling iff one D_i is satisfiable w.r.t. ① to ③?

- if yes, we have shown that satisfiability of $\mathcal{ALC}$ is NExpTime-hard
- so no...what is missing?

Mapping a Domino System into an $\mathcal{ALCQIO}$ Ontology

Two things are missing:

1. the model must be large enough, namely $2^n \times 2^n$ and
2. for each element, its horizontal-vertical-successors **coincide** with their vertical-horizontal-successors and vice versa

This will be addressed using a "**counting and binding together**" trick ...

④ counting and binding together

(a) use $A_1, \ldots, A_n$, $B_1, \ldots, B_n$ as "bits" for binary representation of **grid position**
e.g., (010, 011) is represented by an instance of $\neg A_3, A_2, \neg A_1, \neg B_3, B_2, B_1$

write GCI to ensure that X- and Y-successors are **incremented** correctly
e.g., X-successor of (010, 011) is (011, 011)

(b) use nominals to ensure that there is only one (111...1, 111...1)
this implies, with $\top \sqsubseteq (\leq 1\ X^{-}.\top) \sqcap (\leq 1\ Y^{-}.\top)$ **uniqueness** of grid positions

④ counting and binding together

(a) $\tilde{A}_i$ for "**bit A_i is incremented correctly**":

$$\top \sqsubseteq \tilde{A}_1 \sqcap \ldots \sqcap \tilde{A}_n$$

$$\tilde{A}_1 \sqsubseteq (A_1 \sqcap \forall X.\neg A_1) \sqcup (\neg A_1 \sqcap \forall X.A_1)$$

$$\tilde{A}_i \sqsubseteq \quad (\bigsqcap_{\ell<i} A_\ell \sqcap ((A_i \sqcap \forall X.\neg A_i) \sqcup (\neg A_i \sqcap \forall X.A_i)) \sqcup$$
$$(\neg \bigsqcap_{\ell<i} A_\ell \sqcap ((A_i \sqcap \forall X.A_i) \sqcup (\neg A_i \sqcap \forall X.\neg A_i))$$

(add the same for the B_is)

(b) ensure uniqueness of grid positions:

$A_1 \sqcap \ldots \sqcap A_n \sqcap B_1 \sqcap \ldots \sqcap B_n \sqsubseteq \{o\}$ % top right $(2^n, 2^n)$ is unique

$\top \sqsubseteq (\leq 1\ X^-.\top) \sqcap (\leq 1\ Y^-.\top)$ % everything else is also unique

Reduction of NExpTime Domino Problem to $\mathcal{ALCQIO}$ Consistency

Since the NExpTime-domino problem is NExpTime-hard, this implies consistency of $\mathcal{ALCQIO}$ is also NExpTime-hard:

Lemma: let $\mathcal{O}_D$ be ontology consisting of all axioms mentioned in reduction of D:

- D has an $2^n \times 2^n$ tiling iff $\mathcal{O}_D$ is consistent
- size of $\mathcal{O}_D$ is polynomial (quadratic) in
  - the size of D and
  - n

Let's do some entailment understanding!