$\mathcal{SRIQ}$ and $\mathcal{SROIQ}$ are Harder than $\mathcal{SHOIQ}$

Yevgeny Kazakov

(presented by Birte Glimm)

Oxford University Computing Laboratory

May 15, 2008

Outline

Summary of the Main Results

Known Results (see DL Complexity Navigator[1])

(Finite model) reasoning is:

- ExpTime-complete for $\mathcal{SHIQ}$
- NExpTime-complete for $\mathcal{SHOIQ}$

[1] http://www.cs.man.ac.uk/~ezolin/dl/

Summary of the Main Results

Known Results (See DL Complexity Navigator[1])

(Finite model) reasoning is:

- ExpTime-complete for $\mathcal{SHIQ}$
- NExpTime-complete for $\mathcal{SHOIQ}$

Theorem (New Results in This Talk)

(Finite model) reasoning is:

- 2ExpTime-hard for $\mathcal{SRIQ}$ [and even for $\mathcal{SR}$]
- N2ExpTime-complete for $\mathcal{SROIQ}$ [and for $\mathcal{SROIF}$]

[1] http://www.cs.man.ac.uk/~ezolin/dl/

Summary of the Main Results

Known Results (See DL Complexity Navigator[1])

(Finite model) reasoning is:

- ExpTime-complete for $\mathcal{SHIQ}$
- NExpTime-complete for $\mathcal{SHOIQ}$

Theorem (New Results in This Talk)

(Finite model) reasoning is:

- 2ExpTime-hard for $\mathcal{SRIQ}$ [and even for $\mathcal{SR}$]
- N2ExpTime-complete for $\mathcal{SROIQ}$ [and for $\mathcal{SROIF}$]

In short: $\mathcal{H} \Rightarrow \mathcal{R}$ causes an exponential blowup!

[1] http://www.cs.man.ac.uk/~ezolin/dl/

From $\mathcal{SHIQ}$ to $\mathcal{SROIQ}$

[2003] $\mathcal{SHIQ}$ was extended to $\mathcal{RIQ}$ with

- complex RIAs of the form $R \circ S \sqsubseteq R$ and $S \circ R \sqsubseteq R$
- required to be acyclic: $S \prec R$, otherwise it is undecidable

FROM $\mathcal{SHIQ}$ TO $\mathcal{SROIQ}$

[2003] $\mathcal{SHIQ}$ was extended to $\mathcal{RIQ}$ with

- complex RIAs of the form $R \circ S \sqsubseteq R$ and $S \circ R \sqsubseteq R$
- required to be acyclic: $S \prec R$, otherwise it is undecidable

[2004] $\mathcal{RIQ}$ was extended with more types of complex RIAs:

1. $R \circ R \sqsubseteq R$ (transitivity)
2. $R^{-} \sqsubseteq R$ (symmetry)
3. $S_1 \circ \cdots \circ S_n \sqsubseteq R$ $S_i \prec R$ for all $1 \leq i \leq n$
4. $R \circ S_1 \circ \cdots \circ S_n \sqsubseteq R$ (left-linear) $S_i \prec R$ for all $1 \leq i \leq n$
5. $S_1 \circ \cdots \circ S_n \circ R \sqsubseteq R$ (right-linear) $S_i \prec R$ for all $1 \leq i \leq n$

From $\mathcal{SHIQ}$ to $\mathcal{SROIQ}$

[2003] $\mathcal{SHIQ}$ was extended to $\mathcal{RIQ}$ with

- complex RIAs of the form $R \circ S \sqsubseteq R$ and $S \circ R \sqsubseteq R$
- required to be acyclic: $S \prec R$, otherwise it is undecidable

[2004] $\mathcal{RIQ}$ was extended with more types of complex RIAs:

1. $R \circ R \sqsubseteq R$ (transitivity)
2. $R^{-} \sqsubseteq R$ (symmetry)
3. $S_1 \circ \cdots \circ S_n \sqsubseteq R$ — $S_i \prec R$ for all $1 \leq i \leq n$
4. $R \circ S_1 \circ \cdots \circ S_n \sqsubseteq R$ (left-linear) — $S_i \prec R$ for all $1 \leq i \leq n$
5. $S_1 \circ \cdots \circ S_n \circ R \sqsubseteq R$ (right-linear) — $S_i \prec R$ for all $1 \leq i \leq n$

[2005] $\mathcal{SRIQ}$ extends $\mathcal{RIQ}$ with

- Universal role U
- Negated role assertions $\neg R(a, b)$
- Concept constructor $\exists R.Self$
- Role axioms $Sym(R)$, $Ref(R)$, $Asy(S)$, $Irr(R)$, $Disj(S_1, S_2)$

From $\mathcal{SHIQ}$ to $\mathcal{SROIQ}$

[2003] $\mathcal{SHIQ}$ was extended to $\mathcal{RIQ}$ with

- complex RIAs of the form $R \circ S \sqsubseteq R$ and $S \circ R \sqsubseteq R$
- required to be acyclic: $S \prec R$, otherwise it is undecidable

[2004] $\mathcal{RIQ}$ was extended with more types of complex RIAs:

1. $R \circ R \sqsubseteq R$ (transitivity)
2. $R^{-} \sqsubseteq R$ (symmetry)
3. $S_1 \circ \cdots \circ S_n \sqsubseteq R$ $S_i \prec R$ for all $1 \leq i \leq n$
4. $R \circ S_1 \circ \cdots \circ S_n \sqsubseteq R$ (left-linear) $S_i \prec R$ for all $1 \leq i \leq n$
5. $S_1 \circ \cdots \circ S_n \circ R \sqsubseteq R$ (right-linear) $S_i \prec R$ for all $1 \leq i \leq n$

[2005] $\mathcal{SRIQ}$ extends $\mathcal{RIQ}$ with

- Universal role U
- Negated role assertions $\neg R(a, b)$
- Concept constructor $\exists R.Self$
- Role axioms $Sym(R)$, $Ref(R)$, $Asy(S)$, $Irr(R)$, $Disj(S_1, S_2)$

[2006] $\mathcal{SROIQ} = \mathcal{SRIQ} + \mathcal{SHOIQ}$

proposed as a basis for $\mathcal{OWL}$ **2** (a.k.a. $\mathcal{OWL}$ **1.1**)

Regular RIAs

- Integration of new constructions into existing tableau-based procedures:
- U, $\neg R(a,b)$, $Sym(R)$, $Ref(R)$, $Asy(S)$, $Irr(R)$, $Disj(S_1,S_2)$
  — do not break the tree-model property

Regular RIAs

- Integration of new constructions into existing tableau-based procedures:
- U, $\neg R(a,b)$, $Sym(R)$, $Ref(R)$, $Asy(S)$, $Irr(R)$, $Disj(S_1, S_2)$
  — do not break the tree-model property
- $R_1 \circ \cdots \circ R_n \sqsubseteq R$
  — break the tree-model property
  - Cause undecidability when used without restrictions
  - Regularity restrictions **1** – **5** ensure decidability

Regular RIAs

1 $R \circ R \sqsubseteq R$

2 $R^{-} \sqsubseteq R$

3 $S_1 \circ \cdots \circ S_n \sqsubseteq R$

4 $R \circ S_1 \circ \cdots \circ S_n \sqsubseteq R$

5 $S_1 \circ \cdots \circ S_n \circ R \sqsubseteq R$

$S_i \prec R$

Regular RIAs

- Integration of new constructions into existing tableau-based procedures:
- U, $\neg R(a,b)$, $Sym(R)$, $Ref(R)$, $Asy(S)$, $Irr(R)$, $Disj(S_1,S_2)$
  — do not break the tree-model property
- $R_1 \circ \cdots \circ R_n \sqsubseteq R$
  — break the tree-model property
  - Cause undecidability when used without restrictions
  - Regularity restrictions **1** – **5** ensure decidability

Example

$S \circ R \circ S \sqsubseteq R$ — not regular

$R_i \circ R_i \sqsubseteq R_{i+1}$ — regular by **3**
when $R_0 \prec R_1 \prec \cdots \prec R_n$

Regular RIAs

1. $R \circ R \sqsubseteq R$
2. $R^{-} \sqsubseteq R$
3. $S_1 \circ \cdots \circ S_n \sqsubseteq R$
4. $R \circ S_1 \circ \cdots \circ S_n \sqsubseteq R$
5. $S_1 \circ \cdots \circ S_n \circ R \sqsubseteq R$

$S_i \prec R$

Tableau: The Exponential Blowup

- Every regular RBox $\mathcal{R}$ induces a regular language:

$$L_{\mathcal{R}}(R) = \{S_1 S_2 \ldots S_n \mid S_1 \circ S_2 \circ \cdots \circ S_n \sqsubseteq^*_{\mathcal{R}} R\}$$

Tableau: The Exponential Blowup

- Every regular RBox $\mathcal{R}$ induces a regular language:

$$L_{\mathcal{R}}(R) = \{S_1 S_2 \ldots S_n \mid S_1 \circ S_2 \circ \cdots \circ S_n \sqsubseteq^{*}_{\mathcal{R}} R\}$$

- Tableau procedures for $\mathcal{RIQ}$ – $\mathcal{SROIQ}$ work with $\mathcal{R}$ via the corresponding automata for $L_{\mathcal{R}}(R)$.

Tableau: The Exponential Blowup

- Every regular RBox $\mathcal{R}$ induces a regular language:

$$L_{\mathcal{R}}(R) = \{S_1 S_2 \ldots S_n \mid S_1 \circ S_2 \circ \cdots \circ S_n \sqsubseteq^*_{\mathcal{R}} R\}$$

- Tableau procedures for $\mathcal{RIQ}$ – $\mathcal{SROIQ}$ work with $\mathcal{R}$ via the corresponding automata for $L_{\mathcal{R}}(R)$.

Example (Continued)

$S \circ R \circ S \sqsubseteq R$ $\quad L_{\mathcal{R}}(R) = \{S^i R S^i \mid i \geq 0\}$ — non regular

$R_i \circ R_i \sqsubseteq R_{i+1}$ $\quad L_{\mathcal{R}}(R_{i+1}) = \{R_{i+1}\} \cup L_{\mathcal{R}}(R_i) \cdot L_{\mathcal{R}}(R_i)$

— regular (because finite)

Tableau: The Exponential Blowup

- Every regular RBox $\mathcal{R}$ induces a regular language:

$$L_{\mathcal{R}}(R) = \{S_1 S_2 \ldots S_n \mid S_1 \circ S_2 \circ \cdots \circ S_n \sqsubseteq^*_{\mathcal{R}} R\}$$

- Tableau procedures for $\mathcal{RIQ}$ – $\mathcal{SROIQ}$ work with $\mathcal{R}$ via the corresponding automata for $L_{\mathcal{R}}(R)$.

Example (Continued)

$S \circ R \circ S \sqsubseteq R$ $\quad L_{\mathcal{R}}(R) = \{S^i R S^i \mid i \geq 0\}$ — non regular

$R_i \circ R_i \sqsubseteq R_{i+1}$ $\quad L_{\mathcal{R}}(R_{i+1}) = \{R_{i+1}\} \cup L_{\mathcal{R}}(R_i) \cdot L_{\mathcal{R}}(R_i)$

— regular (because finite)

- The number of different labels in the tableau $\sim 2^{|\mathcal{T}| \cdot |L_{\mathcal{R}}(R)|}$

TABLEAU: THE EXPONENTIAL BLOWUP

- Every regular RBox $\mathcal{R}$ induces a regular language:

$$L_{\mathcal{R}}(R) = \{S_1 S_2 \ldots S_n \mid S_1 \circ S_2 \circ \cdots \circ S_n \sqsubseteq^*_{\mathcal{R}} R\}$$

- Tableau procedures for $\mathcal{RIQ}$ – $\mathcal{SROIQ}$ work with $\mathcal{R}$ via the corresponding automata for $L_{\mathcal{R}}(R)$.

EXAMPLE (CONTINUED)

$S \circ R \circ S \sqsubseteq R$ $\quad L_{\mathcal{R}}(R) = \{S^i R S^i \mid i \geq 0\}$ — non regular

$R_i \circ R_i \sqsubseteq R_{i+1}$ $\quad L_{\mathcal{R}}(R_{i+1}) = \{R_{i+1}\} \cup L_{\mathcal{R}}(R_i) \cdot L_{\mathcal{R}}(R_i)$ — regular (because finite)

- The number of different labels in the tableau $\sim 2^{|\mathcal{T}| \cdot |L_{\mathcal{R}}(R)|}$
- Unfortunately $|L_{\mathcal{R}}(R)|$ can be exponential in $|\mathcal{R}|$: by induction on i one can show that $|L_{\mathcal{R}}(R_i)| \geq 2^i$

Tableau: The Exponential Blowup

- Every regular RBox $\mathcal{R}$ induces a regular language:

$$L_{\mathcal{R}}(R) = \{S_1 S_2 \dots S_n \mid S_1 \circ S_2 \circ \dots \circ S_n \sqsubseteq^*_{\mathcal{R}} R\}$$

- Tableau procedures for $\mathcal{RIQ}$ – $\mathcal{SROIQ}$ work with $\mathcal{R}$ via the corresponding automata for $L_{\mathcal{R}}(R)$.

Example (continued)

$S \circ R \circ S \sqsubseteq R$ $\quad L_{\mathcal{R}}(R) = \{S^i R S^i \mid i \geq 0\}$ — non regular

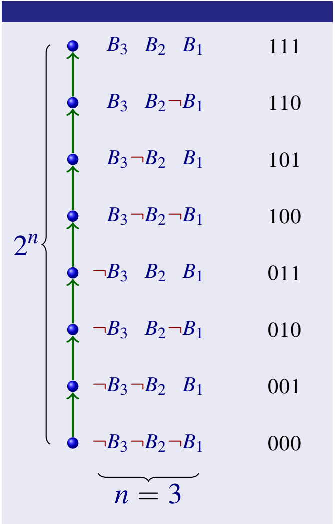
$R_i \circ R_i \sqsubseteq R_{i+1}$ $\quad L_{\mathcal{R}}(R_{i+1}) = \{R_{i+1}\} \cup L_{\mathcal{R}}(R_i) \cdot L_{\mathcal{R}}(R_i)$
— regular (because finite)

- The number of different labels in the tableau $\sim 2^{|\mathcal{T}| \cdot |L_{\mathcal{R}}(R)|}$
- Unfortunately $|L_{\mathcal{R}}(R)|$ can be exponential in $|\mathcal{R}|$: by induction on i one can show that $|L_{\mathcal{R}}(R_i)| \geq 2^i$
- This causes an exponential blowup compared to the procedure for $\mathcal{SHOIQ}$ $\quad \Leftarrow$ Unavoidable??

Outline

Exponential Chains in $\mathcal{ALC}$

- Integer counting technique:
  - A counter between 0 and $2^n - 1$ uses n concepts $B_1, \ldots, B_n$
  - The i-th bit of the counter corresponds to the value of B_i
  - The counter is incremented over R



EXPONENTIAL CHAINS IN $\mathcal{ALC}$

- Integer counting technique:
  - A counter between 0 and $2^n - 1$ uses n concepts $B_1, \ldots, B_n$
  - The i-th bit of the counter corresponds to the value of B_i
  - The counter is incremented over R
- Expressing in $\mathcal{ALC}$:

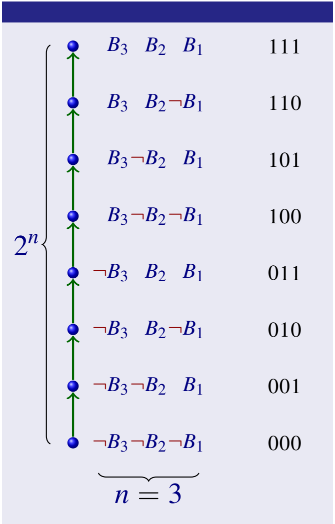
$Z \equiv \neg B_n \sqcap \cdots \sqcap \neg B_1$ — "Zero"

$E \equiv B_n \sqcap \cdots \sqcap B_1$ — "End"

$\neg E \equiv \exists R.\top$ — Successors

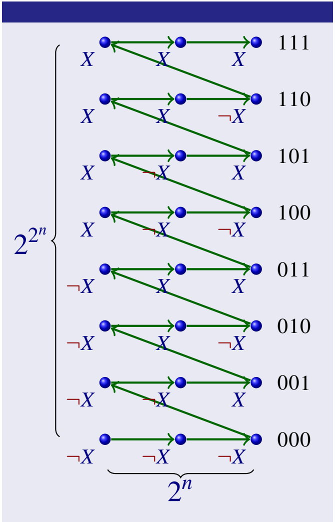
$\top \equiv (B_1 \sqcap \forall R.\neg B_1) \sqcup (\neg B_1 \sqcap \forall R.B_1)$
— The lowest bit always flips

$B_{i-1} \sqcap \forall R.\neg B_{i-1} \equiv$
$(B_i \sqcap \forall R.\neg B_i) \sqcup (\neg B_i \sqcap \forall R.B_i)$
— The bit flips if the lower bit changes from 1 to 0



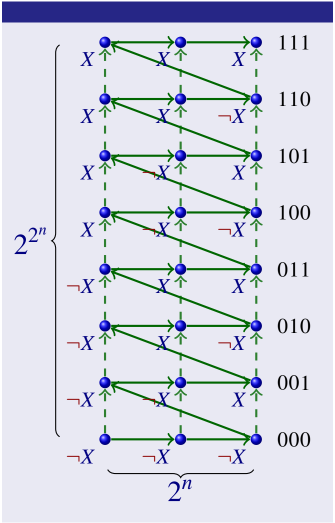
DOUBLY-EXPONENTIAL CHAINS IN $\mathcal{SRIQ}$

- Encode the counter on exponentially-long chains
  - The value of X on i-th element of the chain encodes the i-th bit
  - The chains are connected by "last-to-first element"



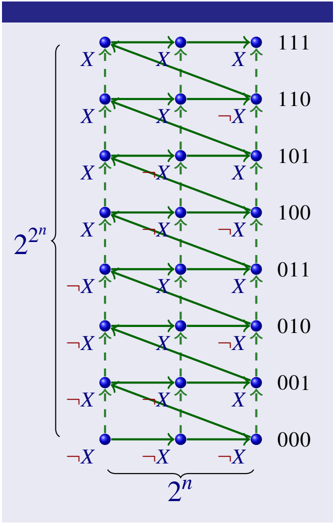
Doubly-Exponential Chains in $\mathcal{SRIQ}$

- Encode the counter on exponentially-long chains
  - The value of X on i-th element of the chain encodes the i-th bit
  - The chains are connected by "last-to-first element"
- Incrementing of the counter
  - Key point: connect corresponding elements using complex RIAs:
  - $R_i \circ R_i \sqsubseteq R_{i+1}$ $\quad R_0 = R$



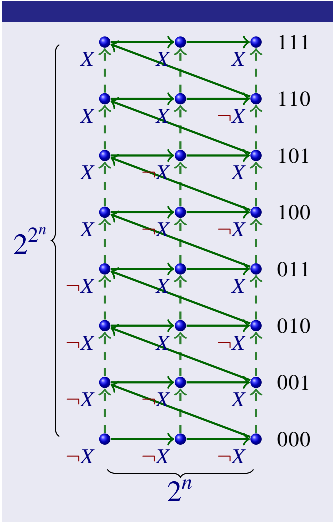
DOUBLY-EXPONENTIAL CHAINS IN $\mathcal{SRIQ}$

- Encode the counter on exponentially-long chains
  - The value of X on i-th element of the chain encodes the i-th bit
  - The chains are connected by "last-to-first element"
- Incrementing of the counter
  - Key point: connect corresponding elements using complex RIAs:
  - $R_i \circ R_i \sqsubseteq R_{i+1}$ $\quad R_0 = R$
  - Complex RIAs connect elements reachable over exactly 2^n roles:
  - $\underbrace{R \circ R \circ \cdots \circ R}_{k} \sqsubseteq R_n \quad$ iff $\quad k = 2^n$



DOUBLY-EXPONENTIAL CHAINS IN $\mathcal{SRIQ}$

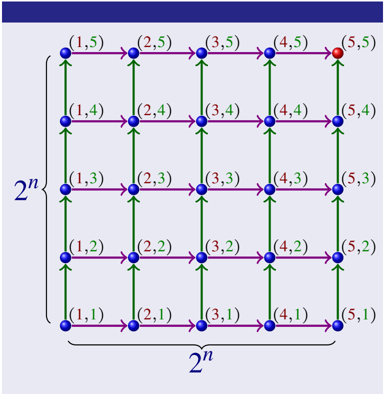
- Encode the counter on exponentially-long chains
  - The value of X on i-th element of the chain encodes the i-th bit
  - The chains are connected by "last-to-first element"
- Incrementing of the counter
  - Key point: connect corresponding elements using complex RIAs:
  - $R_i \circ R_i \sqsubseteq R_{i+1}$ $\quad R_0 = R$
  - Complex RIAs connect elements reachable over exactly 2^n roles:
  - $\underbrace{R \circ R \circ \cdots \circ R}_{k} \sqsubseteq R_n$ iff $k = 2^n$
  - Flipping of corresponding bits:
    $E \sqsubseteq (X \sqcap \forall R_n.\neg X) \sqcup (\neg X \sqcap \forall R_n.X)$
    — the last bit always flips, . . . etc.



HARDNESS RESULT FOR $\mathcal{SROIQ}$

- The key idea is like in the NExpTime-hardness for $\mathcal{SHOIQ}$.
- In $\mathcal{SHOIQ}$ it is possible to express an exponential grid:

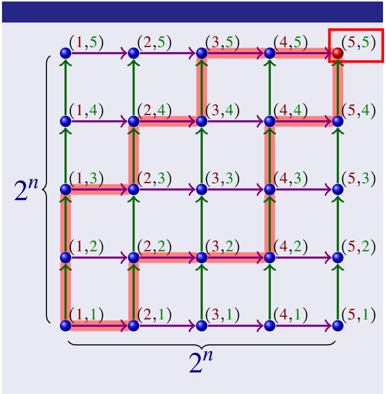
- Use two counters to encode the coordinates of the grid
- Increment / copy the counters over respective edges



Hardness Result for $\mathcal{SROIQ}$

- The key idea is like in the NExpTime-hardness for $\mathcal{SHOIQ}$.
- In $\mathcal{SHOIQ}$ it is possible to express an exponential grid:

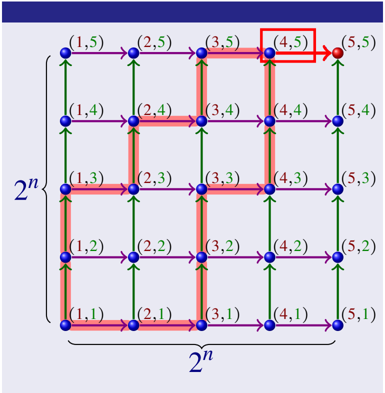
- Use two counters to encode the coordinates of the grid
- Increment / copy the counters over respective edges
- Ensure that the element with the max coordinates is unique using a nominal



Hardness Result for $\mathcal{SROIQ}$

- The key idea is like in the NExpTime-hardness for $\mathcal{SHOIQ}$.
- In $\mathcal{SHOIQ}$ it is possible to express an exponential grid:

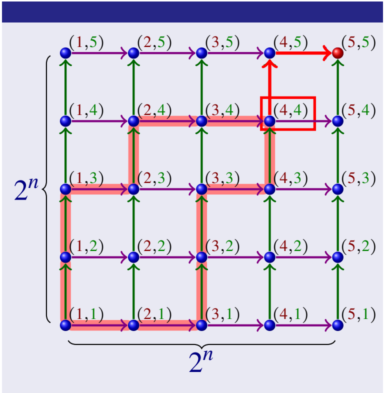
- Use two counters to encode the coordinates of the grid
- Increment / copy the counters over respective edges
- Ensure that the element with the max coordinates is unique using a nominal
- Ensure that elements with smaller coordinates are unique using inverse functional roles



Hardness Result for $\mathcal{SROIQ}$

- The key idea is like in the NExpTime-hardness for $\mathcal{SHOIQ}$.
- In $\mathcal{SHOIQ}$ it is possible to express an exponential grid:

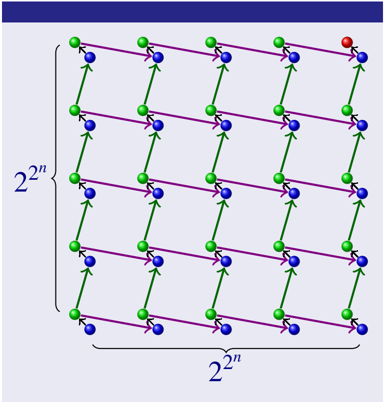
- Use two counters to encode the coordinates of the grid
- Increment / copy the counters over respective edges
- Ensure that the element with the max coordinates is unique using a nominal
- Ensure that elements with smaller coordinates are unique using inverse functional roles



HARDNESS RESULT FOR $\mathcal{SROIQ}$

- The key idea is like in the NExpTime-hardness for $\mathcal{SHOIQ}$.
- In $\mathcal{SHOIQ}$ it is possible to express an exponential grid:

- Use two counters to encode the coordinates of the grid
- Increment / copy the counters over respective edges
- Ensure that the element with the max coordinates is unique using a nominal
- Ensure that elements with smaller coordinates are unique using inverse functional roles



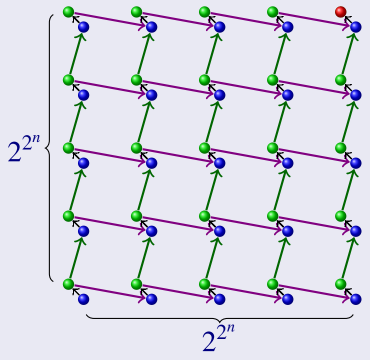
- For $\mathcal{SROIQ}$ the construction is exactly the same but using doubly-exponential counters

HARDNESS RESULT FOR $\mathcal{SROIQ}$

- The key idea is like in the NExpTime-hardness for $\mathcal{SHOIQ}$.
- In $\mathcal{SHOIQ}$ it is possible to express an exponential grid:

THEOREM

(Finite model) reasoning in $\mathcal{SROIQ}$ is N2ExpTime-hard. The result holds already for inverse functional roles and nominals.



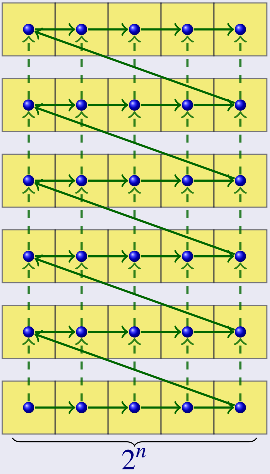
- For $\mathcal{SROIQ}$ the construction is exactly the same but using doubly-exponential counters

Hardness Result for $\mathcal{SRIQ}$

By reduction from the word problem for an exponential-space alternating Turing machine:

- Configurations are encoded on exponential chains
- Corresponding cells of successive configurations are connected by R_n
- Easy to simulate the computation

Computation of TM



Hardness Result for $\mathcal{SRIQ}$

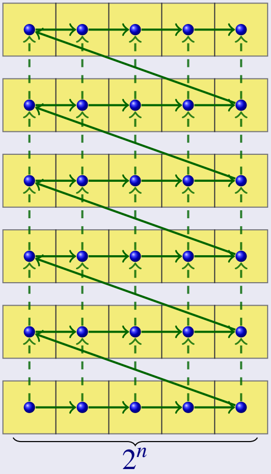
By reduction from the word problem for an exponential-space alternating Turing machine:

- Configurations are encoded on exponential chains
- Corresponding cells of successive configurations are connected by R_n
- Easy to simulate the computation
- Since $\text{AExpSpace} = \text{2ExpTime}$ we have:

Theorem

(Finite model) reasoning in $\mathcal{SRIQ}$ is 2ExpTime-hard. The result holds already without inverses and counting.

Computation of TM



Outline

THE MEMBERSHIP RESULT FOR $\mathcal{SROIQ}$

The matching N2ExpTime upper bound for $\mathcal{SROIQ}$ is obtained by an exponential translation into $\mathcal{C}^2$:

Summary:

1. Simplify ontology to contain only axioms of forms 1–10
2. Eliminate axioms of form 10 using NFA
3. Translate the other axioms into $\mathcal{C}^2$

	Axiom	First-Order Translation
1	$A \sqsubseteq \forall r.B$	$\forall x.(A(x) \rightarrow \forall y.[r(x,y) \rightarrow B(y)])$
2	$A \sqsubseteq \geqslant n\, s.B$	$\forall x.(A(x) \rightarrow \exists^{\geq n} y.[s(x,y) \wedge B(y)])$
3	$A \sqsubseteq \leqslant n\, s.B$	$\forall x.(A(x) \rightarrow \exists^{\leq n} y.[s(x,y) \wedge B(y)])$
4	$A \equiv \exists s.Self$	$\forall x.(A(x) \leftrightarrow s(x,x))$
5	$A_a \equiv \{a\}$	$\exists^{=1} y.A_a(y)$
6	$\bigsqcap A_i \sqsubseteq \bigsqcup B_j$	$\forall x.(\bigvee \neg A_i(x) \vee \bigvee B_j(x))$
7	$Disj(s_1, s_2)$	$\forall xy.(s_1(x,y) \wedge s_2(x,y) \rightarrow \bot)$
8	$s_1 \sqsubseteq s_2$	$\forall xy.(s_1(x,y) \rightarrow s_2(x,y))$
9	$s_1 \sqsubseteq s_2^-$	$\forall xy.(s_1(x,y) \rightarrow s_2(y,x))$
10	$r_1 \circ \cdots \circ r_n \sqsubseteq v$,	$n \geq 1$, v is non-simple

The Membership Result for $\mathcal{SROIQ}$

The matching N2ExpTime upper bound for $\mathcal{SROIQ}$ is obtained by an exponential translation into $\mathcal{C}^2$:

Summary:

1. Simplify ontology to contain only axioms of forms 1–10
2. Eliminate axioms of form 10 using NFA
3. Translate the other axioms into $\mathcal{C}^2$

	Axiom	First-Order Translation
1	$A \sqsubseteq \forall \underline{r}.B$	$\forall x.(A(x) \to \forall y.[r(x,y) \to B(y)])$
2	$A \sqsubseteq \geqslant n\, s.B$	$\forall x.(A(x) \to \exists^{\geq n} y.[s(x,y) \land B(y)])$
3	$A \sqsubseteq \leqslant n\, s.B$	$\forall x.(A(x) \to \exists^{\leq n} y.[s(x,y) \land B(y)])$
4	$A \equiv \exists s.Self$	$\forall x.(A(x) \leftrightarrow s(x,x))$
5	$A_a \equiv \{a\}$	$\exists^{=1} y.A_a(y)$
6	$\bigsqcap A_i \sqsubseteq \bigsqcup B_j$	$\forall x.(\bigvee \neg A_i(x) \lor \bigvee B_j(x))$
7	$Disj(s_1, s_2)$	$\forall xy.(s_1(x,y) \land s_2(x,y) \to \bot)$
8	$s_1 \sqsubseteq s_2$	$\forall xy.(s_1(x,y) \to s_2(x,y))$
9	$s_1 \sqsubseteq s_2^-$	$\forall xy.(s_1(x,y) \to s_2(y,x))$
10	$r_1 \circ \cdots \circ r_n \sqsubseteq \underline{v}$, $n \geq 1$, $\underline{v}$ is non-simple	

Key property for step 2

Axioms of form 10 can interact only with axioms of form 1, since other axioms contain only simple roles $s_{(i)}$

Elimination of Complex RIAs

The main idea

"Absorb" regular RIAs into axioms of the form $A \sqsubseteq \forall r.B$

- For each $A \sqsubseteq \forall r.B$, complex RIAs induce properties:
  $A \sqsubseteq \forall r_1 \circ \cdots \circ r_n.B$, when $r_1 \ldots r_n \in L_{\mathcal{R}}(r)$

Elimination of Complex RIAs

The main idea

"Absorb" regular RIAs into axioms of the form $A \sqsubseteq \forall r.B$

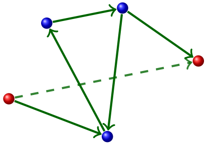
- For each $A \sqsubseteq \forall r.B$, complex RIAs induce properties: $A \sqsubseteq \forall r_1 \circ \cdots \circ r_n.B$, when $r_1 \ldots r_n \in L_{\mathcal{R}}(r)$
- These properties can be expressed alternatively using the regularity of $L_{\mathcal{R}}(r)$:

Elimination of Complex RIAs

The main idea

"Absorb" regular RIAs into axioms of the form $A \sqsubseteq \forall r.B$

- For each $A \sqsubseteq \forall r.B$, complex RIAs induce properties:
  $A \sqsubseteq \forall r_1 \circ \cdots \circ r_n.B, \quad$ when $r_1 \ldots r_n \in L_{\mathcal{R}}(r)$
- These properties can be expressed alternatively using the regularity of $L_{\mathcal{R}}(r)$:
- Take any NFA for $L_{\mathcal{R}}(r)$ with the set of states Q, and the transition relation δ

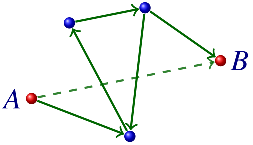


Elimination of Complex RIAs

The main idea

"Absorb" regular RIAs into axioms of the form $A \sqsubseteq \forall r.B$

- For each $A \sqsubseteq \forall r.B$, complex RIAs induce properties:
  $A \sqsubseteq \forall r_1 \circ \dots \circ r_n.B$, when $r_1 \dots r_n \in L_{\mathcal{R}}(r)$
- These properties can be expressed alternatively using the regularity of $L_{\mathcal{R}}(r)$:
- Take any NFA for $L_{\mathcal{R}}(r)$ with the set of states Q, and the transition relation δ, and add new axioms for $A \sqsubseteq \forall r.B$:
  - $A_p \sqsubseteq \forall s.A_q$, when $(p, s, q) \in \delta$
  - $A \sqsubseteq A_p$, when p is the initial state
  - $A_q \sqsubseteq B$, when q is the accepting state

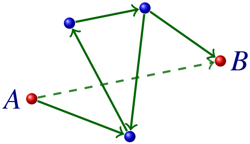


Elimination of Complex RIAs

The main idea

"Absorb" regular RIAs into axioms of the form $A \sqsubseteq \forall r.B$

- For each $A \sqsubseteq \forall r.B$, complex RIAs induce properties:
  $A \sqsubseteq \forall r_1 \circ \cdots \circ r_n.B$, when $r_1 \ldots r_n \in L_{\mathcal{R}}(r)$
- These properties can be expressed alternatively using the regularity of $L_{\mathcal{R}}(r)$:
- Take any NFA for $L_{\mathcal{R}}(r)$ with the set of states Q, and the transition relation δ, and add new axioms for $A \sqsubseteq \forall r.B$:
  - $A_p \sqsubseteq \forall s.A_q$, when $(p, s, q) \in \delta$
  - $A \sqsubseteq A_p$, when p is the initial state
  - $A_q \sqsubseteq B$, when q is the accepting state



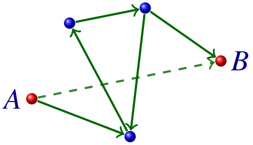
- It is easy to see that these axioms imply
  $A \sqsubseteq \forall r_1 \circ \cdots \circ r_n.B$ *iff* $r_1 \ldots r_n \in L_{\mathcal{R}}(r)$

Elimination of Complex RIAs

The main idea

"Absorb" regular RIAs into axioms of the form $A \sqsubseteq \forall r.B$

- For each $A \sqsubseteq \forall r.B$, complex RIAs induce properties:
  $A \sqsubseteq \forall r_1 \circ \cdots \circ r_n.B$, when $r_1 \ldots r_n \in L_{\mathcal{R}}(r)$
- These properties can be expressed alternatively using the regularity of $L_{\mathcal{R}}(r)$:
- Take any NFA for $L_{\mathcal{R}}(r)$ with the set of states Q, and the transition relation δ, and add new axioms for $A \sqsubseteq \forall r.B$:
  - $A_p \sqsubseteq \forall s.A_q$, when $(p, s, q) \in \delta$
  - $A \sqsubseteq A_p$, when p is the initial state
  - $A_q \sqsubseteq B$, when q is the accepting state



- It is easy to see that these axioms imply
  $A \sqsubseteq \forall r_1 \circ \cdots \circ r_n.B$ *iff* $r_1 \ldots r_n \in L_{\mathcal{R}}(r)$
- Note that $|Q|$ can be exponential in $|\mathcal{R}|$!

THE MEMBERSHIP RESULT FOR $\mathcal{SROIQ}$

The matching N2ExpTime upper bound for $\mathcal{SROIQ}$ is obtained by an exponential translation into $\mathcal{C}^2$:

Summary:

1. Simplify ontology to contain only axioms of forms 1–10 (polynom.)
2. Eliminate axioms of form 10 using NFA (exponential step!)
3. Translate the other axioms into $\mathcal{C}^2$ (NExpTime-complete)

	Axiom	First-Order Translation
1	$A \sqsubseteq \forall r.B$	$\forall x.(A(x) \rightarrow \forall y.[r(x,y) \rightarrow B(y)])$
2	$A \sqsubseteq \geqslant n\, s.B$	$\forall x.(A(x) \rightarrow \exists^{\geq n} y.[s(x,y) \wedge B(y)])$
3	$A \sqsubseteq \leqslant n\, s.B$	$\forall x.(A(x) \rightarrow \exists^{\leq n} y.[s(x,y) \wedge B(y)])$
4	$A \equiv \exists s.Self$	$\forall x.(A(x) \leftrightarrow s(x,x))$
5	$A_a \equiv \{a\}$	$\exists^{=1} y.A_a(y)$
6	$\bigsqcap A_i \sqsubseteq \bigsqcup B_j$	$\forall x.(\bigvee \neg A_i(x) \vee \bigvee B_j(x))$
7	$Disj(s_1, s_2)$	$\forall xy.(s_1(x,y) \wedge s_2(x,y) \rightarrow \bot)$
8	$s_1 \sqsubseteq s_2$	$\forall xy.(s_1(x,y) \rightarrow s_2(x,y))$
9	$s_1 \sqsubseteq s_2^-$	$\forall xy.(s_1(x,y) \rightarrow s_2(y,x))$
10	$r_1 \circ \cdots \circ r_n \sqsubseteq v$,	$n \geq 1$, v is non-simple

THE MEMBERSHIP RESULT FOR $\mathcal{SROIQ}$

The matching N2ExpTime upper bound for $\mathcal{SROIQ}$ is obtained by an exponential translation into $\mathcal{C}^2$:

Summary:

1. Simplify ontology to contain only axioms of forms 1–10 (polynom.)
2. Eliminate axioms of form 10 using NFA (exponential step!)
3. Translate the other axioms into $\mathcal{C}^2$ (NExpTime-complete)

	Axiom	First-Order Translation
1	$A \sqsubseteq \forall r.B$	$\forall x.(A(x) \to \forall y.[r(x,y) \to B(y)])$
2	$A \sqsubseteq \geqslant n\, s.B$	$\forall x.(A(x) \to \exists^{\geq n} y.[s(x,y) \wedge B(y)])$
3	$A \sqsubseteq \leqslant n\, s.B$	$\forall x.(A(x) \to \exists^{\leq n} y.[s(x,y) \wedge B(y)])$
4	$A \equiv \exists s.Self$	$\forall x.(A(x) \leftrightarrow s(x,x))$
5	$A_a \equiv \{a\}$	$\exists^{=1} y.A_a(y)$
6	$\bigsqcap A_i \sqsubseteq \bigsqcup B_j$	$\forall x.(\bigvee \neg A_i(x) \vee \bigvee B_j(x))$
7	$Disj(s_1, s_2)$	$\forall xy.(s_1(x,y) \wedge s_2(x,y) \to \bot)$
8	$s_1 \sqsubseteq s_2$	$\forall xy.(s_1(x,y) \to s_2(x,y))$
9	$s_1 \sqsubseteq s_2^-$	$\forall xy.(s_1(x,y) \to s_2(y,x))$
10	$r_1 \circ \cdots \circ r_n \sqsubseteq v$,	$n \geq 1$, v is non-simple

THEOREM (UPPER COMPLEXITY FOR $\mathcal{SROIQ}$)

(Finite model) reasoning in $\mathcal{SROIQ}$ is N2ExpTime

Outline

1 Introduction

2 Hardness Results

3 Membership Results

4 Discussion

SUMMARY

- We have identified exact computational complexity of $\mathcal{SROIQ}$ to be N2ExpTime; $\mathcal{SRIQ}$ is 2ExpTime-hard.
- Complexity blowup is due to complex RIAs $R_1 \circ \cdots \circ R_n \sqsubseteq R$, in particular because they can chain a fixed exponential number of roles
- Explains the exponential blowup in the tableau procedures for $\mathcal{SRIQ}$ and $\mathcal{SROIQ}$

SUMMARY

- We have identified exact computational complexity of $\mathcal{SROIQ}$ to be N2ExpTime; $\mathcal{SRIQ}$ is 2ExpTime-hard.
- Complexity blowup is due to complex RIAs $R_1 \circ \cdots \circ R_n \sqsubseteq R$, in particular because they can chain a fixed exponential number of roles
- Explains the exponential blowup in the tableau procedures for $\mathcal{SRIQ}$ and $\mathcal{SROIQ}$
- Open problems:
  1. Upper bound for $\mathcal{SRIQ}$? Conjecture: 2ExpTime
  2. Upper & Lower bounds for $\mathcal{RIQ}$? Conjecture: 2ExpTime

  $\mathcal{RIQ}$ allows only for restricted complex RIAs of the form $R \circ S \sqsubseteq R$ and $S \circ R \sqsubseteq R$ which cannot be used in our constructions

Avoiding the Exponential Blowup

- The exponential blowup occurs in rather exotic cases, unlikely to occur often in practice

Avoiding the Exponential Blowup

- The exponential blowup occurs in rather exotic cases, unlikely to occur often in practice
- Some further restrictions on complex RIAs are known to prevent an exponential blowup

  (e.g. when every sequence $R_1 \prec R_2 \prec \cdots \prec R_n$ has a bounded length)

Avoiding the Exponential Blowup

- The exponential blowup occurs in rather exotic cases, unlikely to occur often in practice
- Some further restrictions on complex RIAs are known to prevent an exponential blowup

  (e.g. when every sequence $R_1 \prec R_2 \prec \cdots \prec R_n$ has a bounded length)
- Only the size of the RBox has a high complexity impact:

<table>
<tr><th colspan="3">$\mathcal{SH}[\mathcal{O}]\mathcal{IQ}$</th></tr>
<tr><th>ABox</th><th>TBox</th><th>RBox</th></tr>
<tr><td>NP ?</td><td></td><td></td></tr>
<tr><td colspan="2">[N]ExpTime</td><td></td></tr>
<tr><td colspan="3">[N]ExpTime</td></tr>
</table>

<table>
<tr><th colspan="3">$\mathcal{SR}[\mathcal{O}]\mathcal{IQ}$</th></tr>
<tr><th>ABox</th><th>TBox</th><th>RBox</th></tr>
<tr><td>NP ?</td><td></td><td></td></tr>
<tr><td colspan="2">[N]ExpTime</td><td></td></tr>
<tr><td colspan="3">2[N]ExpTime</td></tr>
</table>

Questions?

- Please send difficult questions to

Yevgeny Kazakov

yevgeny.kazakov@comlab.ox.ac.uk

- Our contribution:
  1. $\mathcal{SROIQ}$ [$\mathcal{SROIF}$] is N2ExpTime-complete
  2. $\mathcal{SRIQ}$ [$\mathcal{SR}$] is 2ExpTime-hard

- Thank you for your attention!