Guarded Fragments Meet Dynamic Logic
The Story of Regular Guards

KR 17.11.25, Melbourne, Austria

Bartosz Jan Bednarczyk `bartek@cs.uni.wroc.pl`

(Joint work with Emanuel Kieroński)

Technische Universität Wien, Australia & Uniwersytet Wrocławski, Poland

Proving Useless Complexity Results For Random Logics

KR 17.11.25, Melbourne, Austria

Bartosz Jan Bednarczyk `bartek@cs.uni.wroc.pl`

(Joint work with Emanuel Kieroński)

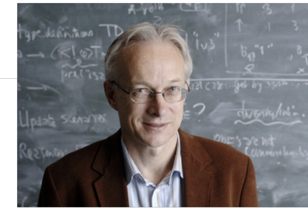
Technische Universität Wien, Australia & Uniwersytet Wrocławski, Poland

The Guarded Fragment $\mathcal{GF}$ of First-Order Logic [Andreka et al. 1998]

The Guarded Fragment $\mathcal{GF}$ of First-Order Logic [Andreka et al. 1998]

HAJNAL ANDRÉKA, ISTVÁN NÉMETI and JOHAN VAN BENTHEM





MODAL LANGUAGES AND BOUNDED FRAGMENTS OF PREDICATE LOGIC

The Guarded Fragment $\mathcal{GF}$ of First-Order Logic [Andreka et al. 1998]

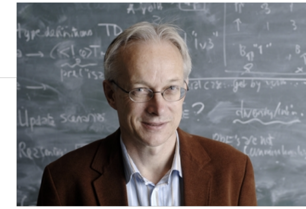
HAJNAL ANDRÉKA, ISTVÁN NÉMETI and JOHAN VAN BENTHEM

MODAL LANGUAGES AND BOUNDED FRAGMENTS OF PREDICATE LOGIC

- The guarded fragment of $\mathcal{FO}$ is obtained by relativising quantifiers by atoms.

The Guarded Fragment $\mathcal{GF}$ of First-Order Logic [Andreka et al. 1998]

HAJNAL ANDRÉKA, ISTVÁN NÉMETI and JOHAN VAN BENTHEM



MODAL LANGUAGES AND BOUNDED FRAGMENTS OF PREDICATE LOGIC

- The guarded fragment of $\mathcal{FO}$ is obtained by relativising quantifiers by atoms.
- $\exists \vec{y}\ \alpha(\vec{x}, \vec{y}) \land \varphi(\vec{x}, \vec{y}), \forall \vec{y}\ \alpha(\vec{x}, \vec{y}) \rightarrow \varphi(\vec{x}, \vec{y})$ – guard must cover free variables of φ.

The Guarded Fragment $\mathcal{GF}$ of First-Order Logic [Andreka et al. 1998]

HAJNAL ANDRÉKA, ISTVÁN NÉMETI and JOHAN VAN BENTHEM

MODAL LANGUAGES AND BOUNDED FRAGMENTS OF PREDICATE LOGIC

- The guarded fragment of $\mathcal{FO}$ is obtained by relativising quantifiers by atoms.
- $\exists\vec{y}\ \alpha(\vec{x},\vec{y}) \wedge \varphi(\vec{x},\vec{y}), \forall\vec{y}\ \alpha(\vec{x},\vec{y}) \rightarrow \varphi(\vec{x},\vec{y})$ – guard must cover free variables of φ.

$$\varphi, \varphi' ::= \mathrm{R}(\bar{x}) \mid \neg\varphi \mid \varphi \wedge \varphi' \mid \exists x\, \varphi(x) \mid \forall x\, \varphi(x) \mid \forall \bar{x}\, \alpha(\bar{x}) \rightarrow \varphi(\bar{x}) \mid \exists \bar{x}\, \alpha(\bar{x}) \wedge \varphi(\bar{x})$$

The Guarded Fragment $\mathcal{GF}$ of First-Order Logic [Andreka et al. 1998]

HAJNAL ANDRÉKA, ISTVÁN NÉMETI and JOHAN VAN BENTHEM

MODAL LANGUAGES AND BOUNDED FRAGMENTS OF PREDICATE LOGIC

- The guarded fragment of $\mathcal{FO}$ is obtained by relativising quantifiers by atoms.
- $\exists\vec{y}\ \alpha(\vec{x},\vec{y})\wedge\varphi(\vec{x},\vec{y}), \forall\vec{y}\ \alpha(\vec{x},\vec{y})\rightarrow\varphi(\vec{x},\vec{y})$ – guard must cover free variables of φ.

$$\varphi,\varphi' ::= \mathrm{R}(\bar{x}) \mid \neg\varphi \mid \varphi\wedge\varphi' \mid \exists x\,\varphi(x) \mid \forall x\,\varphi(x) \mid \forall\bar{x}\,\alpha(\bar{x})\rightarrow\varphi(\bar{x}) \mid \exists\bar{x}\,\alpha(\bar{x})\wedge\varphi(\bar{x})$$

Example 1. Some artist admires only beekeepers

$$\exists x\ \mathit{artst}(x) \wedge \forall y\ \left(\mathit{adm}(x,y) \rightarrow \mathit{bkpr}(y)\right)$$

The Guarded Fragment $\mathcal{GF}$ of First-Order Logic [Andreka et al. 1998]

HAJNAL ANDRÉKA, ISTVÁN NÉMETI and JOHAN VAN BENTHEM

MODAL LANGUAGES AND BOUNDED FRAGMENTS OF PREDICATE LOGIC

- The guarded fragment of $\mathcal{FO}$ is obtained by relativising quantifiers by atoms.
- $\exists \vec{y}\ \alpha(\vec{x},\vec{y}) \wedge \varphi(\vec{x},\vec{y}), \forall \vec{y}\ \alpha(\vec{x},\vec{y}) \rightarrow \varphi(\vec{x},\vec{y})$ – guard must cover free variables of φ.

$$\varphi, \varphi' ::= \mathrm{R}(\bar{x}) \mid \neg\varphi \mid \varphi \wedge \varphi' \mid \exists x\, \varphi(x) \mid \forall x\, \varphi(x) \mid \forall \bar{x}\, \alpha(\bar{x}) \rightarrow \varphi(\bar{x}) \mid \exists \bar{x}\, \alpha(\bar{x}) \wedge \varphi(\bar{x})$$

Example 1. Some artist admires only beekeepers

$$\exists x\ \mathit{artst}(x) \wedge \forall y\ (\mathit{adm}(x,y) \rightarrow \mathit{bkpr}(y))$$

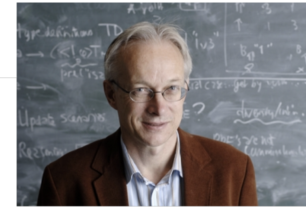
Example 2. Every artist envies every beekeeper he admires

$$\forall x\ \mathit{artst}(x) \rightarrow \forall y\ [\mathit{adm}(x,y) \rightarrow (\mathit{bkpr}(y) \rightarrow \mathit{env}(x,y))]$$

The Guarded Fragment $\mathcal{GF}$ of First-Order Logic [Andreka et al. 1998]

HAJNAL ANDRÉKA, ISTVÁN NÉMETI and JOHAN VAN BENTHEM





MODAL LANGUAGES AND BOUNDED FRAGMENTS OF PREDICATE LOGIC

- The guarded fragment of $\mathcal{FO}$ is obtained by relativising quantifiers by atoms.
- $\exists \vec{y}\ \alpha(\vec{x}, \vec{y}) \wedge \varphi(\vec{x}, \vec{y}), \forall \vec{y}\ \alpha(\vec{x}, \vec{y}) \rightarrow \varphi(\vec{x}, \vec{y})$ – guard must cover free variables of φ.

$$\varphi, \varphi' ::= \mathrm{R}(\bar{x}) \mid \neg\varphi \mid \varphi \wedge \varphi' \mid \exists x\, \varphi(x) \mid \forall x\, \varphi(x) \mid \forall \bar{x}\, \alpha(\bar{x}) \rightarrow \varphi(\bar{x}) \mid \exists \bar{x}\, \alpha(\bar{x}) \wedge \varphi(\bar{x})$$

Example 1. Some artist admires only beekeepers

$$\exists x\ artst(x) \wedge \forall y\ (adm(x, y) \rightarrow bkpr(y))$$

Example 2. Every artist envies every beekeeper he admires

$$\forall x\ artst(x) \rightarrow \forall y\ [adm(x, y) \rightarrow (bkpr(y) \rightarrow env(x, y))]$$

Coexample 3. Every artist admires every beekeeper

$$\forall x\ (artst(x) \rightarrow \forall y\ (bkpr(y) \rightarrow adm(x, y)))$$

Standard Translation From Modal Logics to First-Order Logic

Standard Translation From Modal Logics to First-Order Logic

- Recall a translation from modal logic ($\mathcal{ALC}$) to the guarded fragment.

Standard Translation From Modal Logics to First-Order Logic

- Recall a translation from modal logic ($\mathcal{ALC}$) to the guarded fragment.

$$\exists \mathrm{R}. \big((\mathrm{p} \land \neg \mathrm{q}) \lor \forall \bar{\mathrm{S}}.\mathrm{r} \big)$$

Standard Translation From Modal Logics to First-Order Logic

- Recall a translation from modal logic ($\mathcal{ALC}$) to the guarded fragment.

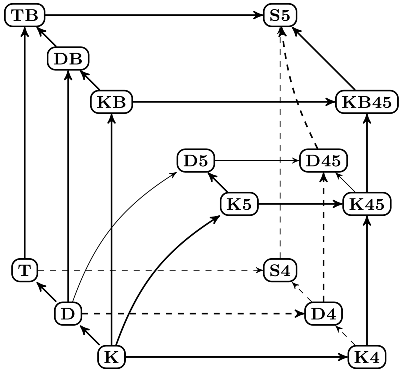
$$\exists \mathrm{R}.\Big((\mathrm{p} \land \neg \mathrm{q}) \lor \forall \bar{\mathrm{S}}.\mathrm{r}\Big) \rightsquigarrow \ \varphi(x) := \exists y\Big(\mathrm{R}(x,y) \land \Big[(\mathrm{p}(y) \land \neg \mathrm{q}(y)) \lor \forall z\Big(\mathrm{S}(z,y) \to \mathrm{r}(z)\Big)\Big]\Big)$$

Standard Translation From Modal Logics to First-Order Logic

- Recall a translation from modal logic ($\mathcal{ALC}$) to the guarded fragment.

$$\exists \mathrm{R}.\left((\mathrm{p} \wedge \neg \mathrm{q}) \vee \forall \bar{\mathrm{S}}.\mathrm{r}\right) \rightsquigarrow \varphi(x) := \exists y\Big(\mathrm{R}(x,y) \wedge \Big[(\mathrm{p}(y) \wedge \neg \mathrm{q}(y)) \vee \forall z\Big(\mathrm{S}(z,y) \rightarrow \mathrm{r}(z)\Big)\Big]\Big)$$

- What about other classical modal logics from the modal cube?

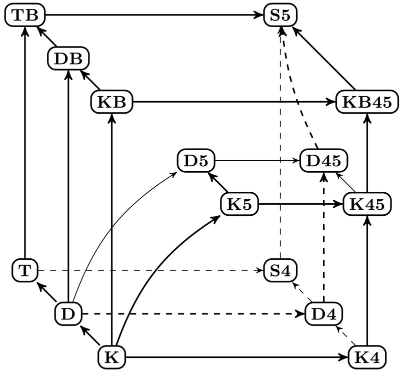


Standard Translation From Modal Logics to First-Order Logic

- Recall a translation from modal logic ($\mathcal{ALC}$) to the guarded fragment.

$$\exists \mathrm{R}.\left((\mathrm{p} \wedge \neg \mathrm{q}) \vee \forall \bar{\mathrm{S}}.\mathrm{r}\right) \rightsquigarrow \varphi(x) := \exists y\Big(\mathrm{R}(x,y) \wedge \Big[(\mathrm{p}(y) \wedge \neg \mathrm{q}(y)) \vee \forall z\Big(\mathrm{S}(z,y) \rightarrow \mathrm{r}(z)\Big)\Big]\Big)$$

- What about other classical modal logics from the modal cube?



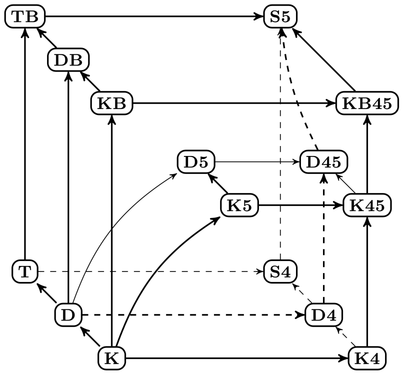
Not in GF!

Standard Translation From Modal Logics to First-Order Logic

- Recall a translation from modal logic ($\mathcal{ALC}$) to the guarded fragment.

$$\exists \mathrm{R}.\left((\mathrm{p} \land \neg \mathrm{q}) \lor \forall \bar{\mathrm{S}}.\mathrm{r}\right) \rightsquigarrow \varphi(x) := \exists y\Big(\mathrm{R}(x,y) \land \Big[(\mathrm{p}(y) \land \neg \mathrm{q}(y)) \lor \forall z\Big(\mathrm{S}(z,y) \to \mathrm{r}(z)\Big)\Big]\Big)$$

- What about other classical modal logics from the modal cube?



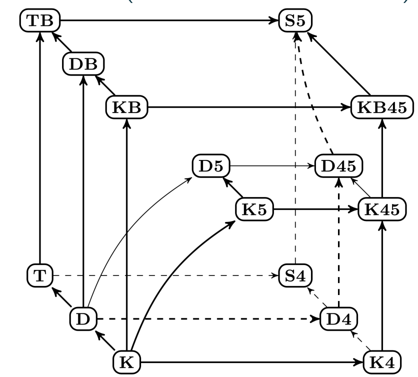
Not in GF! It cannot express transitivity/equivalence/Euclidean/etc.

Standard Translation From Modal Logics to First-Order Logic

- Recall a translation from modal logic ($\mathcal{ALC}$) to the guarded fragment.

$$\exists \mathrm{R}.\Big((\mathrm{p} \land \neg \mathrm{q}) \lor \forall \bar{\mathrm{S}}.\mathrm{r}\Big) \rightsquigarrow \varphi(x) := \exists y\Big(\mathrm{R}(x,y) \land \Big[(\mathrm{p}(y) \land \neg \mathrm{q}(y)) \lor \forall z\Big(\mathrm{S}(z,y) \to \mathrm{r}(z)\Big)\Big]\Big)$$

- What about other classical modal logics from the modal cube?



Not in GF! It cannot express transitivity/equivalence/Euclidean/etc.

The Two-Variable Guarded Fragment with Transitive Relations

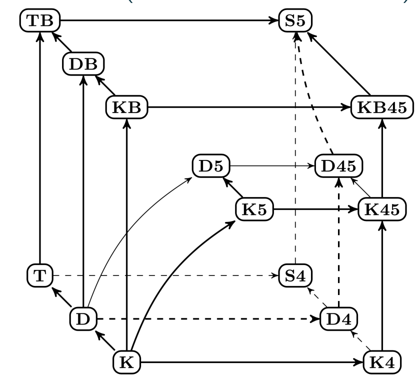
H. Ganzinger, C. Meyer, and M. Veanes
Max-Planck-Institut für Informatik, D-66123 Saarbrücken, Germany

Standard Translation From Modal Logics to First-Order Logic

- Recall a translation from modal logic ($\mathcal{ALC}$) to the guarded fragment.

$$\exists \mathrm{R}.\left((\mathrm{p} \wedge \neg \mathrm{q}) \vee \forall \bar{\mathrm{S}}.\mathrm{r}\right) \rightsquigarrow \varphi(x) := \exists y\Big(\mathrm{R}(x,y) \wedge \Big[(\mathrm{p}(y) \wedge \neg \mathrm{q}(y)) \vee \forall z\Big(\mathrm{S}(z,y) \rightarrow \mathrm{r}(z)\Big)\Big]\Big)$$

- What about other classical modal logics from the modal cube?



Not in GF! It cannot express transitivity/equivalence/Euclidean/etc.

The Two-Variable Guarded Fragment with Transitive Relations

H. Ganzinger, C. Meyer, and M. Veanes
Max-Planck-Institut für Informatik, D-66123 Saarbrücken, Germany

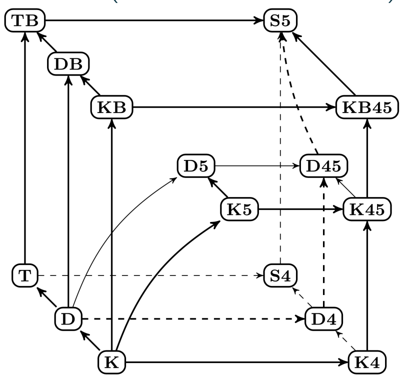
Even worse: the satisfiability problem for GF with transitive relations is undecidable!

Standard Translation From Modal Logics to First-Order Logic

- Recall a translation from modal logic ($\mathcal{ALC}$) to the guarded fragment.

$$\exists \mathrm{R}.\Big((\mathrm{p} \land \neg \mathrm{q}) \lor \forall \bar{\mathrm{S}}.\mathrm{r}\Big) \rightsquigarrow \ \varphi(x) := \exists y \Big(\mathrm{R}(x,y) \land \Big[(\mathrm{p}(y) \land \neg \mathrm{q}(y)) \lor \forall z \Big(\mathrm{S}(z,y) \to \mathrm{r}(z)\Big)\Big]\Big)$$

- What about other classical modal logics from the modal cube?



Not in GF! It cannot express transitivity/equivalence/Euclidean/etc.

The Two-Variable Guarded Fragment with Transitive Relations

H. Ganzinger, C. Meyer, and M. Veanes
Max-Planck-Institut für Informatik, D-66123 Saarbrücken, Germany

Even worse: the satisfiability problem for GF with transitive relations is undecidable!

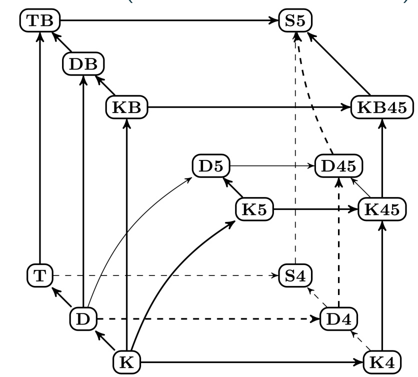
Guarded Fragment with Semantically Constrained Guards

Standard Translation From Modal Logics to First-Order Logic

- Recall a translation from modal logic ($\mathcal{ALC}$) to the guarded fragment.

$$\exists \mathrm{R}.\Big((\mathrm{p} \wedge \neg \mathrm{q}) \vee \forall \bar{\mathrm{S}}.\mathrm{r}\Big) \rightsquigarrow \varphi(x) := \exists y \Big(\mathrm{R}(x,y) \wedge \Big[(\mathrm{p}(y) \wedge \neg \mathrm{q}(y)) \vee \forall z \Big(\mathrm{S}(z,y) \rightarrow \mathrm{r}(z)\Big)\Big]\Big)$$

- What about other classical modal logics from the modal cube?



Not in GF! It cannot express transitivity/equivalence/Euclidean/etc.

The Two-Variable Guarded Fragment with Transitive Relations

H. Ganzinger, C. Meyer, and M. Veanes
Max-Planck-Institut für Informatik, D-66123 Saarbrücken, Germany

Even worse: the satisfiability problem for GF with transitive relations is undecidable!

Guarded Fragment with Semantically Constrained Guards

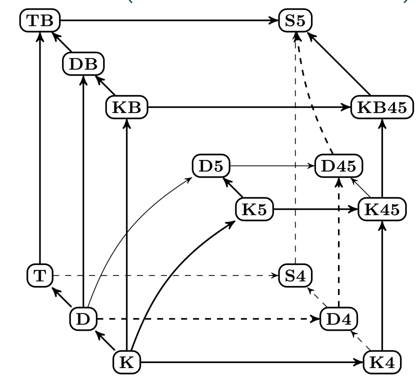
- Ganzinger et al. initiated the study of $\mathcal{GF}$ where distinguished relations may appear only as guards.

Standard Translation From Modal Logics to First-Order Logic

- Recall a translation from modal logic ($\mathcal{ALC}$) to the guarded fragment.

$$\exists \mathrm{R}.\left((\mathrm{p} \land \neg \mathrm{q}) \lor \forall \bar{\mathrm{S}}.\mathrm{r}\right) \rightsquigarrow \varphi(x) := \exists y \Big(\mathrm{R}(x,y) \land \Big[(\mathrm{p}(y) \land \neg \mathrm{q}(y)) \lor \forall z \Big(\mathrm{S}(z,y) \to \mathrm{r}(z)\Big)\Big]\Big)$$

- What about other classical modal logics from the modal cube?



Not in GF! It cannot express transitivity/equivalence/Euclidean/etc.

The Two-Variable Guarded Fragment with Transitive Relations

H. Ganzinger, C. Meyer, and M. Veanes
Max-Planck-Institut für Informatik, D-66123 Saarbrücken, Germany

Even worse: the satisfiability problem for GF with transitive relations is undecidable!

Guarded Fragment with Semantically Constrained Guards

- Ganzinger et al. initiated the study of $\mathcal{GF}$ where distinguished relations may appear only as guards.

Dec. cases: transitive, equivalence (closure) guards





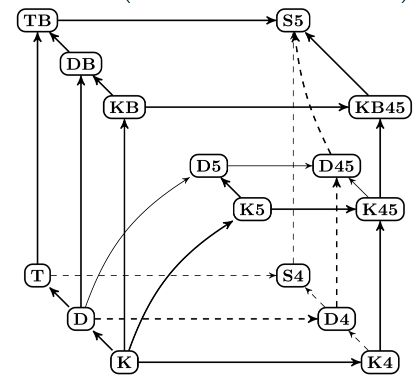


Standard Translation From Modal Logics to First-Order Logic

- Recall a translation from modal logic ($\mathcal{ALC}$) to the guarded fragment.

$$\exists \mathrm{R}.\Big((\mathrm{p} \wedge \neg \mathrm{q}) \vee \forall \bar{\mathrm{S}}.\mathrm{r}\Big) \rightsquigarrow \varphi(x) := \exists y\Big(\mathrm{R}(x,y) \wedge \Big[(\mathrm{p}(y) \wedge \neg \mathrm{q}(y)) \vee \forall z\Big(\mathrm{S}(z,y) \rightarrow \mathrm{r}(z)\Big)\Big]\Big)$$

- What about other classical modal logics from the modal cube?



Not in GF! It cannot express transitivity/equivalence/Euclidean/etc.

The Two-Variable Guarded Fragment with Transitive Relations

H. Ganzinger, C. Meyer, and M. Veanes
Max-Planck-Institut für Informatik, D-66123 Saarbrücken, Germany

Even worse: the satisfiability problem for GF with transitive relations is undecidable!

Guarded Fragment with Semantically Constrained Guards

- Ganzinger et al. initiated the study of $\mathcal{GF}$ where distinguished relations may appear only as guards.

Dec. cases: transitive, equivalence (closure) guards Undec. cases: $\approx$ + expon., composition, conj. of TG







- Ganzinger et al. initiated the study of $\mathcal{GF}$ where distinguished relations may appear only as guards.

Dec. cases: transitive, equivalence (closure) guards Undec. cases: $\approx$ $+$ expon., composition, conj. of TG

- Ganzinger et al. initiated the study of $\mathcal{GF}$ where distinguished relations may appear only as guards.

Dec. cases: transitive, equivalence (closure) guards   Undec. cases: $\approx$ + expon., composition, conj. of TG







- Ganzinger et al. initiated the study of $\mathcal{GF}$ where distinguished relations may appear only as guards.

Dec. cases: transitive, equivalence (closure) guards Undec. cases: $\approx$ + expon., composition, conj. of TG

Our Motivation

- Ganzinger et al. initiated the study of $\mathcal{GF}$ where distinguished relations may appear only as guards.

Dec. cases: transitive, equivalence (closure) guards   Undec. cases: $\approx$ + expon., composition, conj. of TG

Our Motivation

- Goal: Provide an extension of GF that captures dynamic logic (ICPDL).

- Ganzinger et al. initiated the study of $\mathcal{GF}$ where distinguished relations may appear only as guards.

Dec. cases: transitive, equivalence (closure) guards   Undec. cases: $\approx$ + expon., composition, conj. of TG







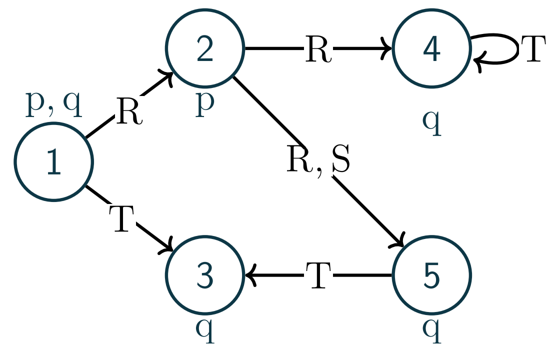
Our Motivation

- Goal: Provide an extension of GF that captures dynamic logic (ICPDL).

ICPDL = ALC + $\exists(\pi \cap \ldots \cap \pi').\varphi$

π - two-way regular expressions

$\mathfrak{A} :=$



E.g. $(\mathfrak{A}, 1) \models \exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$

- Ganzinger et al. initiated the study of $\mathcal{GF}$ where distinguished relations may appear only as guards.

Dec. cases: transitive, equivalence (closure) guards   Undec. cases: $\approx$ + expon., composition, conj. of TG

Our Motivation

- Goal: Provide an extension of GF that captures dynamic logic (ICPDL).

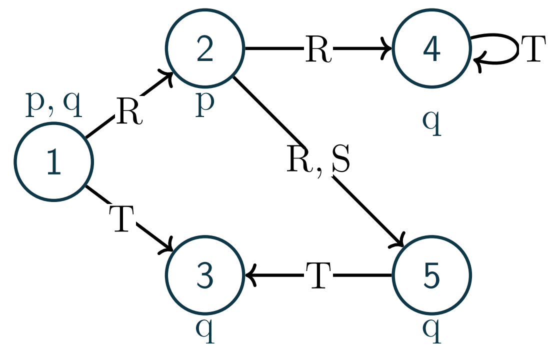
Motivation I: Allowing for expressive navigation features á la RPQs to model graph database scenarios.

ICPDL = ALC + $\exists(\pi \cap \ldots \cap \pi').\varphi$

π - two-way regular expressions

E.g. $(\mathfrak{A}, 1) \models \exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$

$\mathfrak{A} :=$



- Ganzinger et al. initiated the study of $\mathcal{GF}$ where distinguished relations may appear only as guards.

Dec. cases: transitive, equivalence (closure) guards   Undec. cases: $\approx$ + expon., composition, conj. of TG







Our Motivation

- Goal: Provide an extension of GF that captures dynamic logic (ICPDL).

Motivation I: Allowing for expressive navigation features á la RPQs to model graph database scenarios.

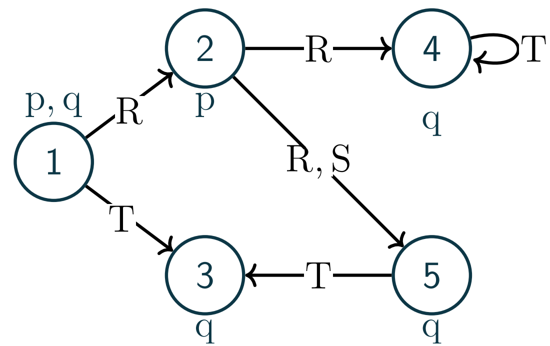
Motivation II: Improve state of the art. Limitation: undecidability of the logics with $\approx$.

ICPDL = ALC + $\exists(\pi \cap \ldots \cap \pi').\varphi$

π - two-way regular expressions

E.g. $(\mathfrak{A}, 1) \models \exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$

$\mathfrak{A} :=$



- Ganzinger et al. initiated the study of $\mathcal{GF}$ where distinguished relations may appear only as guards.

Dec. cases: transitive, equivalence (closure) guards   Undec. cases: $\approx$ + expon., composition, conj. of TG







Our Motivation

- Goal: Provide an extension of GF that captures dynamic logic (ICPDL).

Motivation I: Allowing for expressive navigation features á la RPQs to model graph database scenarios.

Motivation II: Improve state of the art. Limitation: undecidability of the logics with $\approx$.

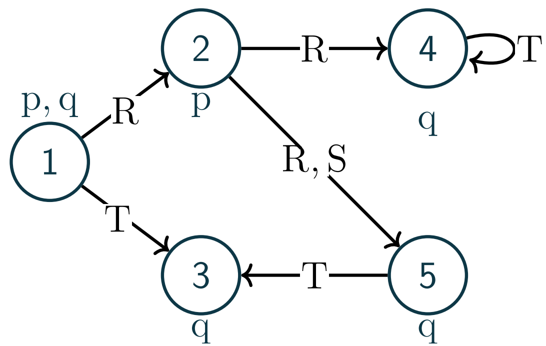
Motivation III: Some positive results are only for the two-variable fragments.

ICPDL = ALC + $\exists(\pi \cap \ldots \cap \pi').\varphi$

π - two-way regular expressions

E.g. $(\mathfrak{A}, 1) \models \exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$

$\mathfrak{A} :=$



- Ganzinger et al. initiated the study of $\mathcal{GF}$ where distinguished relations may appear only as guards.

Dec. cases: transitive, equivalence (closure) guards Undec. cases: $\approx$ + expon., composition, conj. of TG







Our Motivation

- Goal: Provide an extension of GF that captures dynamic logic (ICPDL).

Motivation I: Allowing for expressive navigation features á la RPQs to model graph database scenarios.

Motivation II: Improve state of the art. Limitation: undecidability of the logics with $\approx$.

Motivation III: Some positive results are only for the two-variable fragments.

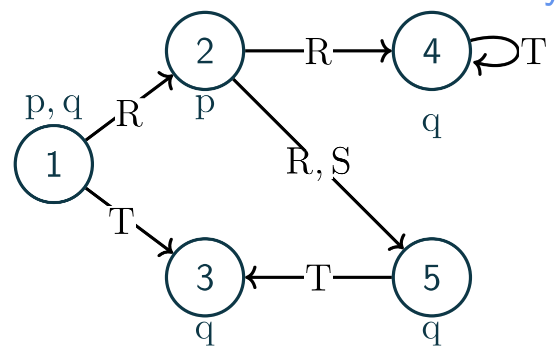
Motivation IV: Provide a more high-level proof capturing many variants of GF in a uniform way.

$\mathsf{ICPDL} = \mathsf{ALC} + \exists(\pi \cap \ldots \cap \pi').\varphi$

π - two-way regular expressions

E.g. $(\mathfrak{A}, 1) \models \exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$

$\mathfrak{A} :=$



Guarded Fragment with Regular Guards (RGF)

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1:

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\Big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\Big).\mathrm{q}$ translates to

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\Big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\Big).\mathrm{q}$ translates to $\exists y \ \Big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\Big)(xy) \wedge \mathrm{q}(y)$

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$ translates to $\exists y \; \big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big)(xy) \wedge \mathrm{q}(y)$
- Example 2: All R-reachable elements are B-connected:

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$ translates to $\exists y \; \big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big)(xy) \wedge \mathrm{q}(y)$
- Example 2: All R-reachable elements are B-connected: $\forall xy \, \mathrm{R}^*(xy) \rightarrow \mathrm{B}(xy)$

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$ translates to $\exists y\ \big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big)(xy) \wedge \mathrm{q}(y)$
- Example 2: All R-reachable elements are B-connected: $\forall xy\, \mathrm{R}^*(xy) \rightarrow \mathrm{B}(xy)$ [not in μGF]

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\Big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\Big).\mathrm{q}$ translates to $\exists y \; \Big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\Big)(xy) \wedge \mathrm{q}(y)$
- Example 2: All R-reachable elements are B-connected: $\forall xy \, \mathrm{R}^*(xy) \rightarrow \mathrm{B}(xy)$ [not in μGF]
- Captures transitivity of via R^+,

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\Big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\Big).\mathrm{q}$ translates to $\exists y \; \Big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\Big)(xy) \land \mathrm{q}(y)$
- Example 2: All R-reachable elements are B-connected: $\forall xy\, \mathrm{R}^*(xy) \to \mathrm{B}(xy)$ [not in μGF]
- Captures transitivity of via R^+, equivalence relations via $(\mathrm{R} + \bar{\mathrm{R}})^*$,

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$ translates to $\exists y \; \big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big)(xy) \land \mathrm{q}(y)$
- Example 2: All R-reachable elements are B-connected: $\forall xy \, \mathrm{R}^*(xy) \rightarrow \mathrm{B}(xy)$ [not in μGF]
- Captures transitivity of via R^+, equivalence relations via $(\mathrm{R} + \bar{\mathrm{R}})^*$, composition and more.

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$ translates to $\exists y \ \big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big)(xy) \wedge \mathrm{q}(y)$
- Example 2: All R-reachable elements are B-connected: $\forall xy \, \mathrm{R}^*(xy) \rightarrow \mathrm{B}(xy)$ [not in μGF]
- Captures transitivity of via R^+, equivalence relations via $(\mathrm{R} + \bar{\mathrm{R}})^*$, composition and more.
- Built-in support for regular path queries.

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\Big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\Big).\mathrm{q}$ translates to $\exists y \, \Big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\Big)(xy) \wedge \mathrm{q}(y)$
- Example 2: All R-reachable elements are B-connected: $\forall xy \, \mathrm{R}^*(xy) \rightarrow \mathrm{B}(xy)$ [not in μGF]
- Captures transitivity of via R^+, equivalence relations via $(\mathrm{R} + \bar{\mathrm{R}})^*$, composition and more.
- Built-in support for regular path queries.
- Due to the presence of $\cap$ we generalize GF with conjunction of transitive guards.

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\Big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\Big).\mathrm{q}$ translates to $\exists y\ \Big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\Big)(xy) \land \mathrm{q}(y)$
- Example 2: All R-reachable elements are B-connected: $\forall xy\, \mathrm{R}^*(xy) \to \mathrm{B}(xy)$ [not in μGF]
- Captures transitivity of via R^+, equivalence relations via $(\mathrm{R} + \bar{\mathrm{R}})^*$, composition and more.
- Built-in support for regular path queries.
- Due to the presence of $\cap$ we generalize GF with conjunction of transitive guards.

Theorem (KR 2025)

The satisfiability problem for RGF is $2\textsc{ExpTime}$-complete.

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$ translates to $\exists y\ \big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big)(xy) \land \mathrm{q}(y)$
- Example 2: All R-reachable elements are B-connected: $\forall xy\, \mathrm{R}^*(xy) \to \mathrm{B}(xy)$ [not in μGF]
- Captures transitivity of via R^+, equivalence relations via $(\mathrm{R} + \bar{\mathrm{R}})^*$, composition and more.
- Built-in support for regular path queries.
- Due to the presence of $\cap$ we generalize GF with conjunction of transitive guards.

Theorem (KR 2025)

The satisfiability problem for RGF is $2\textsc{ExpTime}$-complete.

Proof idea:

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$ translates to $\exists y\ \big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big)(xy) \land \mathrm{q}(y)$
- Example 2: All R-reachable elements are B-connected: $\forall xy\, \mathrm{R}^*(xy) \rightarrow \mathrm{B}(xy)$ [not in μGF]
- Captures transitivity of via R^+, equivalence relations via $(\mathrm{R} + \bar{\mathrm{R}})^*$, composition and more.
- Built-in support for regular path queries.
- Due to the presence of $\cap$ we generalize GF with conjunction of transitive guards.

Theorem (KR 2025)

The satisfiability problem for RGF is $2\textsc{ExpTime}$-complete.

Proof idea: solve RGF^2 via a reduction to ICPDL and

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$ translates to $\exists y\ \big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big)(xy) \wedge \mathrm{q}(y)$
- Example 2: All R-reachable elements are B-connected: $\forall xy\, \mathrm{R}^*(xy) \to \mathrm{B}(xy)$ [not in μGF]
- Captures transitivity of via R^+, equivalence relations via $(\mathrm{R} + \bar{\mathrm{R}})^*$, composition and more.
- Built-in support for regular path queries.
- Due to the presence of $\cap$ we generalize GF with conjunction of transitive guards.

Theorem (KR 2025)

The satisfiability problem for RGF is $2\textsc{ExpTime}$-complete.

Proof idea: solve RGF^2 via a reduction to ICPDL and lift it to the full RGF via a fusion.

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$ translates to $\exists y \; \big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big)(xy) \land \mathrm{q}(y)$
- Example 2: All R-reachable elements are B-connected: $\forall xy \, \mathrm{R}^*(xy) \to \mathrm{B}(xy)$ [not in μGF]
- Captures transitivity of via R^+, equivalence relations via $(\mathrm{R} + \bar{\mathrm{R}})^*$, composition and more.
- Built-in support for regular path queries.
- Due to the presence of $\cap$ we generalize GF with conjunction of transitive guards.

Theorem (KR 2025)

The satisfiability problem for RGF is $2\textsc{ExpTime}$-complete.

Proof idea: solve RGF^2 via a reduction to ICPDL and lift it to the full RGF via a fusion.

Theorem (KR 2025)

Finite&Unrestricted Conjunctive Query Entailment Problems are undecidable for fluted RGF^2,

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$ translates to $\exists y \ \big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big)(xy) \land \mathrm{q}(y)$
- Example 2: All R-reachable elements are B-connected: $\forall xy\, \mathrm{R}^*(xy) \rightarrow \mathrm{B}(xy)$ [not in μGF]
- Captures transitivity of via R^+, equivalence relations via $(\mathrm{R} + \bar{\mathrm{R}})^*$, composition and more.
- Built-in support for regular path queries.
- Due to the presence of $\cap$ we generalize GF with conjunction of transitive guards.

Theorem (KR 2025)

The satisfiability problem for RGF is $2\textsc{ExpTime}$-complete.

Proof idea: solve RGF^2 via a reduction to ICPDL and lift it to the full RGF via a fusion.

Theorem (KR 2025)

Finite&Unrestricted Conjunctive Query Entailment Problems are undecidable for fluted RGF^2, already for TBoxes $\mathcal{T} \cup \{\mathsf{trans}(p), p \subseteq s \cup r\}$ for $\mathcal{ALC}^{\cap}$-TBoxes $\mathcal{T}$ and role names r, s, p.

Guarded Fragment with Regular Guards (RGF)

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

- Example 1: $\exists\big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big).\mathrm{q}$ translates to $\exists y \ \big((\mathrm{T} \circ \bar{\mathrm{T}}) \cap \mathrm{R}^*\big)(xy) \wedge \mathrm{q}(y)$
- Example 2: All R-reachable elements are B-connected: $\forall xy\, \mathrm{R}^*(xy) \rightarrow \mathrm{B}(xy)$ [not in μGF]
- Captures transitivity of via R^+, equivalence relations via $(\mathrm{R} + \bar{\mathrm{R}})^*$, composition and more.
- Built-in support for regular path queries.
- Due to the presence of $\cap$ we generalize GF with conjunction of transitive guards.

Theorem (KR 2025)

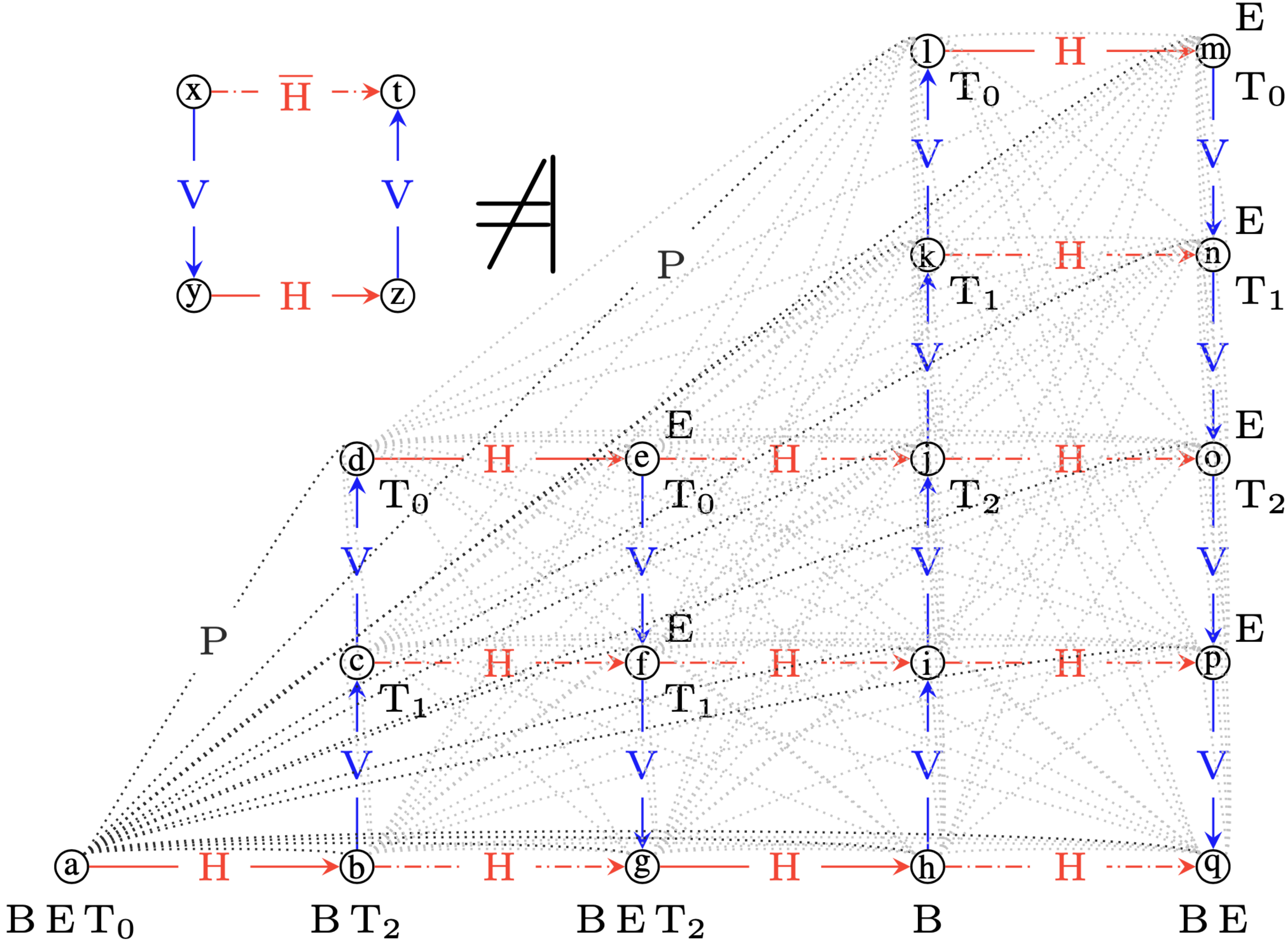
The satisfiability problem for RGF is $2\textsc{ExpTime}$-complete.

Proof idea: solve RGF^2 via a reduction to ICPDL and lift it to the full RGF via a fusion.

Theorem (KR 2025)

Finite&Unrestricted Conjunctive Query Entailment Problems are undecidable for fluted RGF^2, already for TBoxes $\mathcal{T} \cup \{\mathsf{trans}(p), p \subseteq s \cup r\}$ for $\mathcal{ALC}^{\cap}$-TBoxes $\mathcal{T}$ and role names r, s, p.

Significantly improves prior results by Gottlob&Pieris&Tendera from ICALP 2013.


x
t
y
z
$\overline{\mathrm{H}}$
H
V
P
a
b
c
d
e
f
g
h
i
j
k
l
m
n
o
p
q
E
T_0
T_1
T_2
B E T_0
B T_2
B E T_2
B
B E

Technical Toolkit (for the two-variable case)

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathsf{RGF}^2$ is in Scott's normal form if it is a conjunction of:

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathsf{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathrm{FO}}$-conjuncts]

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1 \, \eta_i(x_1) \rightarrow \exists x_2 \, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathrm{FO}}$-conjuncts]
- $\forall x_1 \forall x_2 \, \eta_i(x_1 x_2) \rightarrow \psi_i(x_1 x_2)$

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathrm{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \to \psi_i(x_1 x_2)$ [$\forall^{\mathrm{FO}}$-conjuncts]

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1 x_2) \land \psi_i(x_1 x_2)$ [$\exists^{\mathrm{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \to \psi_i(x_1 x_2)$ [$\forall^{\mathrm{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \to \exists x_2\, \pi_i(x_1 x_2) \land \phi_i(x_1 x_2)$

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1 \, \eta_i(x_1) \to \exists x_2 \, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathrm{FO}}$-conjuncts]
- $\forall x_1 \forall x_2 \, \eta_i(x_1 x_2) \to \psi_i(x_1 x_2)$ [$\forall^{\mathrm{FO}}$-conjuncts]
- $\forall x_1 \, \gamma_i(x_1) \to \exists x_2 \, \pi_i(x_1 x_2) \wedge \phi_i(x_1 x_2)$ [$\exists^{\mathrm{reg}}$-conjuncts]

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathsf{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \to \psi_i(x_1 x_2)$ [$\forall^{\mathsf{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \to \exists x_2\, \pi_i(x_1 x_2) \wedge \phi_i(x_1 x_2)$ [$\exists^{\mathsf{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1 x_2) \to \phi_i(x_1 x_2)$,

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathsf{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \to \psi_i(x_1 x_2)$ [$\forall^{\mathsf{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \to \exists x_2\, \pi_i(x_1 x_2) \wedge \phi_i(x_1 x_2)$ [$\exists^{\mathsf{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1 x_2) \to \phi_i(x_1 x_2)$, [$\forall^{\mathsf{reg}}$-conjuncts]

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \rightarrow \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathsf{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \rightarrow \psi_i(x_1 x_2)$ [$\forall^{\mathsf{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \rightarrow \exists x_2\, \pi_i(x_1 x_2) \wedge \phi_i(x_1 x_2)$ [$\exists^{\mathsf{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1 x_2) \rightarrow \phi_i(x_1 x_2)$, [$\forall^{\mathsf{reg}}$-conjuncts]

where all π_i are RPQ with $\cap$,

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \rightarrow \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathrm{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \rightarrow \psi_i(x_1 x_2)$ [$\forall^{\mathrm{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \rightarrow \exists x_2\, \pi_i(x_1 x_2) \wedge \phi_i(x_1 x_2)$ [$\exists^{\mathrm{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1 x_2) \rightarrow \phi_i(x_1 x_2)$, [$\forall^{\mathrm{reg}}$-conjuncts]

where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathrm{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \to \psi_i(x_1 x_2)$ [$\forall^{\mathrm{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \to \exists x_2\, \pi_i(x_1 x_2) \wedge \phi_i(x_1 x_2)$ [$\exists^{\mathrm{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1 x_2) \to \phi_i(x_1 x_2)$, [$\forall^{\mathrm{reg}}$-conjuncts]

where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \rightarrow \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathrm{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \rightarrow \psi_i(x_1 x_2)$ [$\forall^{\mathrm{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \rightarrow \exists x_2\, \pi_i(x_1 x_2) \wedge \phi_i(x_1 x_2)$ [$\exists^{\mathrm{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1 x_2) \rightarrow \phi_i(x_1 x_2)$, [$\forall^{\mathrm{reg}}$-conjuncts]

where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

Atomic One-Types and Two-Types

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathsf{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \to \psi_i(x_1 x_2)$ [$\forall^{\mathsf{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \to \exists x_2\, \pi_i(x_1 x_2) \wedge \phi_i(x_1 x_2)$ [$\exists^{\mathsf{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1 x_2) \to \phi_i(x_1 x_2)$, [$\forall^{\mathsf{reg}}$-conjuncts]

where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

Atomic One-Types and Two-Types

- 1-type over Σ = maximal satisfiable conjunction of Σ-literals involving x_1. $(\boldsymbol{\alpha}_\Sigma)$

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathsf{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \to \psi_i(x_1 x_2)$ [$\forall^{\mathsf{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \to \exists x_2\, \pi_i(x_1 x_2) \wedge \phi_i(x_1 x_2)$ [$\exists^{\mathsf{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1 x_2) \to \phi_i(x_1 x_2)$, [$\forall^{\mathsf{reg}}$-conjuncts]

where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

Atomic One-Types and Two-Types

- 1-type over Σ = maximal satisfiable conjunction of Σ-literals involving x_1. $(\boldsymbol{\alpha}_\Sigma)$
- 2-type over Σ = maximal satisfiable conjunction of Σ-literals involving x_1, x_2. $(\boldsymbol{\beta}_\Sigma)$

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathrm{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \to \psi_i(x_1 x_2)$ [$\forall^{\mathrm{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \to \exists x_2\, \pi_i(x_1 x_2) \wedge \phi_i(x_1 x_2)$ [$\exists^{\mathrm{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1 x_2) \to \phi_i(x_1 x_2)$, [$\forall^{\mathrm{reg}}$-conjuncts]

where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

Atomic One-Types and Two-Types

- 1-type over Σ = maximal satisfiable conjunction of Σ-literals involving x_1. $(\boldsymbol{\alpha}_\Sigma)$
- 2-type over Σ = maximal satisfiable conjunction of Σ-literals involving x_1, x_2. $(\boldsymbol{\beta}_\Sigma)$

Intuition: 1-types are colours of elements, and 2-types are colours of edges.

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \rightarrow \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathrm{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \rightarrow \psi_i(x_1 x_2)$ [$\forall^{\mathrm{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \rightarrow \exists x_2\, \pi_i(x_1 x_2) \wedge \phi_i(x_1 x_2)$ [$\exists^{\mathrm{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1 x_2) \rightarrow \phi_i(x_1 x_2)$, [$\forall^{\mathrm{reg}}$-conjuncts]

where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

Atomic One-Types and Two-Types

- 1-type over Σ = maximal satisfiable conjunction of Σ-literals involving x_1. $(\boldsymbol{\alpha}_\Sigma)$
- 2-type over Σ = maximal satisfiable conjunction of Σ-literals involving x_1, x_2. $(\boldsymbol{\beta}_\Sigma)$

Intuition: 1-types are colours of elements, and 2-types are colours of edges.

Note: Given φ we have $|\boldsymbol{\alpha}_\varphi| \in O(2^{|\varphi|})$ but $|\boldsymbol{\beta}_\varphi| \in O(2^{2^{|\varphi|}})$ [reason: arbitrary arity symbols!].

Technical Toolkit (for the two-variable case)

We say that $\varphi \in \mathrm{RGF}^2$ is in Scott's normal form if it is a conjunction of:

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \rightarrow \exists x_2\, \vartheta_i(x_1x_2) \wedge \psi_i(x_1x_2)$ [$\exists^{\mathrm{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1x_2) \rightarrow \psi_i(x_1x_2)$ [$\forall^{\mathrm{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \rightarrow \exists x_2\, \pi_i(x_1x_2) \wedge \phi_i(x_1x_2)$ [$\exists^{\mathrm{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1x_2) \rightarrow \phi_i(x_1x_2)$, [$\forall^{\mathrm{reg}}$-conjuncts]

where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

Atomic One-Types and Two-Types

- 1-type over Σ = maximal satisfiable conjunction of Σ-literals involving x_1. $(\boldsymbol{\alpha}_\Sigma)$
- 2-type over Σ = maximal satisfiable conjunction of Σ-literals involving x_1, x_2. $(\boldsymbol{\beta}_\Sigma)$

Intuition: 1-types are colours of elements, and 2-types are colours of edges.

Note: Given φ we have $|\boldsymbol{\alpha}_\varphi| \in O(2^{|\varphi|})$ but $|\boldsymbol{\beta}_\varphi| \in O(2^{2^{|\varphi|}})$ [reason: arbitrary arity symbols!].

Usually we rephrase satisfiaction of φ in terms of realizable types.

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathsf{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \to \psi_i(x_1 x_2)$ [$\forall^{\mathsf{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \to \exists x_2\, \pi_i(x_1 x_2) \wedge \phi_i(x_1 x_2)$ [$\exists^{\mathsf{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1 x_2) \to \phi_i(x_1 x_2)$, [$\forall^{\mathsf{reg}}$-conjuncts]

where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \rightarrow \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathsf{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \rightarrow \psi_i(x_1 x_2)$ [$\forall^{\mathsf{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \rightarrow \exists x_2\, \pi_i(x_1 x_2) \wedge \phi_i(x_1 x_2)$ [$\exists^{\mathsf{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1 x_2) \rightarrow \phi_i(x_1 x_2)$, [$\forall^{\mathsf{reg}}$-conjuncts]

where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

From RGF^2 **to** ICPDL**.**

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathsf{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \to \psi_i(x_1 x_2)$ [$\forall^{\mathsf{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \to \exists x_2\, \pi_i(x_1 x_2) \wedge \phi_i(x_1 x_2)$ [$\exists^{\mathsf{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1 x_2) \to \phi_i(x_1 x_2)$, [$\forall^{\mathsf{reg}}$-conjuncts]

where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

From RGF^2 **to** ICPDL.

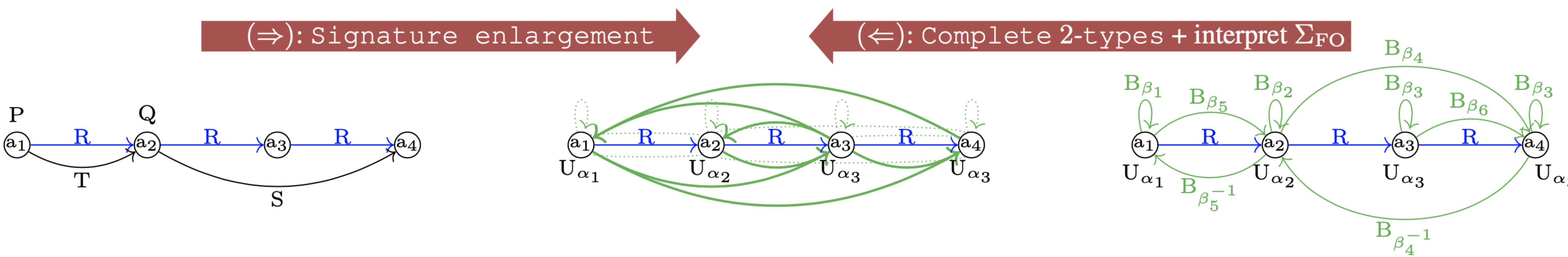
Take φ. If satisfiable, φ has a model realizing at most exponentially many 2-types.

- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathsf{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \to \psi_i(x_1 x_2)$ [$\forall^{\mathsf{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \to \exists x_2\, \pi_i(x_1 x_2) \wedge \phi_i(x_1 x_2)$ [$\exists^{\mathsf{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1 x_2) \to \phi_i(x_1 x_2)$, [$\forall^{\mathsf{reg}}$-conjuncts]

where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

From RGF^2 **to** ICPDL.

Take φ. If satisfiable, φ has a model realizing at most exponentially many 2-types. Our vocabulary contains:



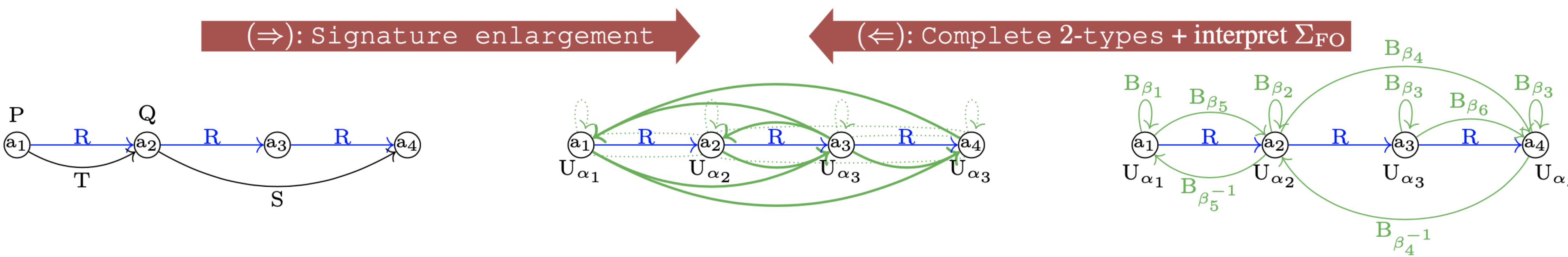
- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1x_2) \wedge \psi_i(x_1x_2)$ [$\exists^{\mathsf{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1x_2) \to \psi_i(x_1x_2)$ [$\forall^{\mathsf{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \to \exists x_2\, \pi_i(x_1x_2) \wedge \phi_i(x_1x_2)$ [$\exists^{\mathsf{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1x_2) \to \phi_i(x_1x_2)$, [$\forall^{\mathsf{reg}}$-conjuncts]

where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

From RGF^2 to ICPDL.

Take φ. If satisfiable, φ has a model realizing at most exponentially many 2-types. Our vocabulary contains:

1. Fresh unary U_α per each 1-type α from $\boldsymbol{\alpha}_\varphi$.



- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1x_2) \wedge \psi_i(x_1x_2)$ [$\exists^{\mathsf{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1x_2) \to \psi_i(x_1x_2)$ [$\forall^{\mathsf{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \to \exists x_2\, \pi_i(x_1x_2) \wedge \phi_i(x_1x_2)$ [$\exists^{\mathsf{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1x_2) \to \phi_i(x_1x_2)$, [$\forall^{\mathsf{reg}}$-conjuncts]

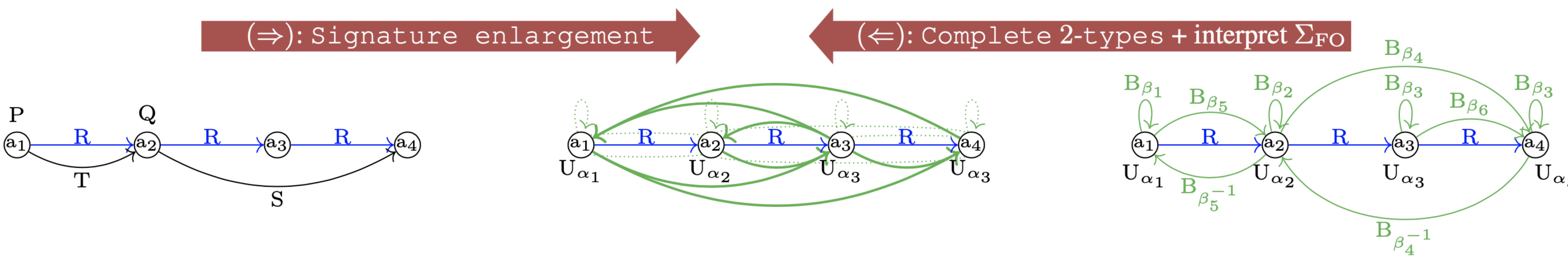
where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

From RGF^2 to ICPDL.

Take φ. If satisfiable, φ has a model realizing at most exponentially many 2-types. Our vocabulary contains:

1. Fresh unary U_α per each 1-type α from $\boldsymbol{\alpha}_\varphi$.

2. Fresh binary B_β per each 2-type β from $\boldsymbol{\beta}_\varphi$ restricted to Σ_{FO}



- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1x_2) \wedge \psi_i(x_1x_2)$ [$\exists^{\mathsf{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1x_2) \to \psi_i(x_1x_2)$ [$\forall^{\mathsf{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \to \exists x_2\, \pi_i(x_1x_2) \wedge \phi_i(x_1x_2)$ [$\exists^{\mathsf{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1x_2) \to \phi_i(x_1x_2)$, [$\forall^{\mathsf{reg}}$-conjuncts]

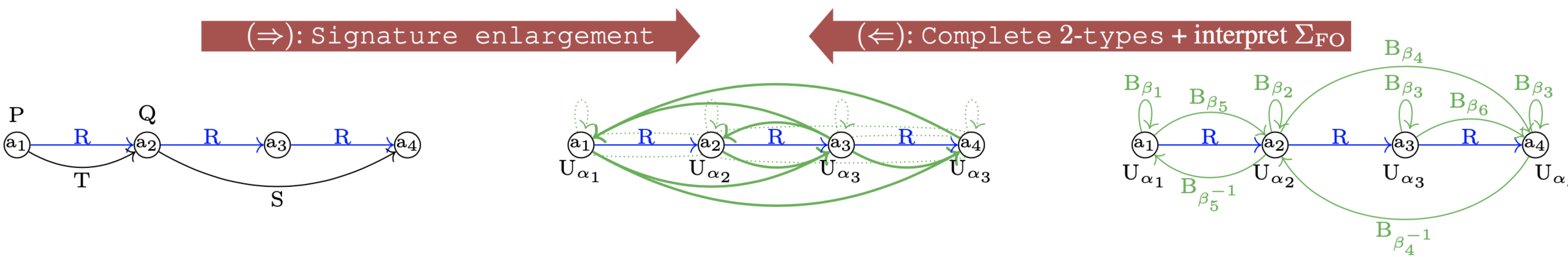
where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

From RGF^2 to ICPDL.

Take φ. If satisfiable, φ has a model realizing at most exponentially many 2-types. Our vocabulary contains:

1. Fresh unary U_α per each 1-type α from $\boldsymbol{\alpha}_\varphi$.

2. Fresh binary B_β per each 2-type β from $\boldsymbol{\beta}_\varphi$ restricted to Σ_{FO} + all binary predicates in Σ_{reg} from φ.



- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \rightarrow \exists x_2\, \vartheta_i(x_1 x_2) \wedge \psi_i(x_1 x_2)$ [$\exists^{\mathsf{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1 x_2) \rightarrow \psi_i(x_1 x_2)$ [$\forall^{\mathsf{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \rightarrow \exists x_2\, \pi_i(x_1 x_2) \wedge \phi_i(x_1 x_2)$ [$\exists^{\mathsf{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1 x_2) \rightarrow \phi_i(x_1 x_2)$, [$\forall^{\mathsf{reg}}$-conjuncts]

where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

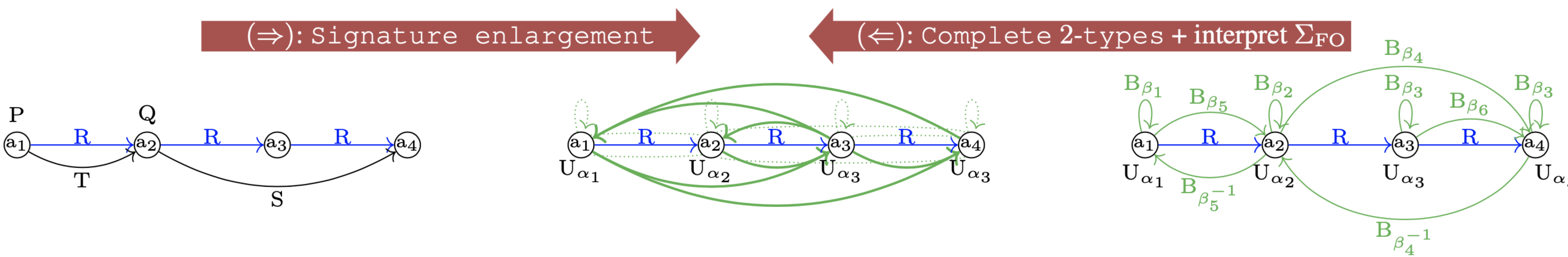
From RGF^2 to ICPDL.

Take φ. If satisfiable, φ has a model realizing at most exponentially many 2-types. Our vocabulary contains:

1. Fresh unary U_α per each 1-type α from $\boldsymbol{\alpha}_\varphi$.

2. Fresh binary B_β per each 2-type β from $\boldsymbol{\beta}_\varphi$ restricted to Σ_{FO} + all binary predicates in Σ_{reg} from φ.

Translation almost easy, e.g. $\forall x_1\, \gamma(x_1) \rightarrow \exists x_2\, \pi(x_1 x_2) \wedge \phi(x_1 x_2) \mapsto$



- $\forall x_1 \lambda(x_1)$ [the general universal conjunct]
- $\forall x_1\, \eta_i(x_1) \to \exists x_2\, \vartheta_i(x_1x_2) \wedge \psi_i(x_1x_2)$ [$\exists^{\mathsf{FO}}$-conjuncts]
- $\forall x_1 \forall x_2\, \eta_i(x_1x_2) \to \psi_i(x_1x_2)$ [$\forall^{\mathsf{FO}}$-conjuncts]
- $\forall x_1\, \gamma_i(x_1) \to \exists x_2\, \pi_i(x_1x_2) \wedge \phi_i(x_1x_2)$ [$\exists^{\mathsf{reg}}$-conjuncts]
- $\forall x_1 \forall x_2\, \pi_i(x_1x_2) \to \phi_i(x_1x_2)$, [$\forall^{\mathsf{reg}}$-conjuncts]

where all π_i are RPQ with $\cap$, and all other formulae are quantifier-free and over Σ_{FO}.

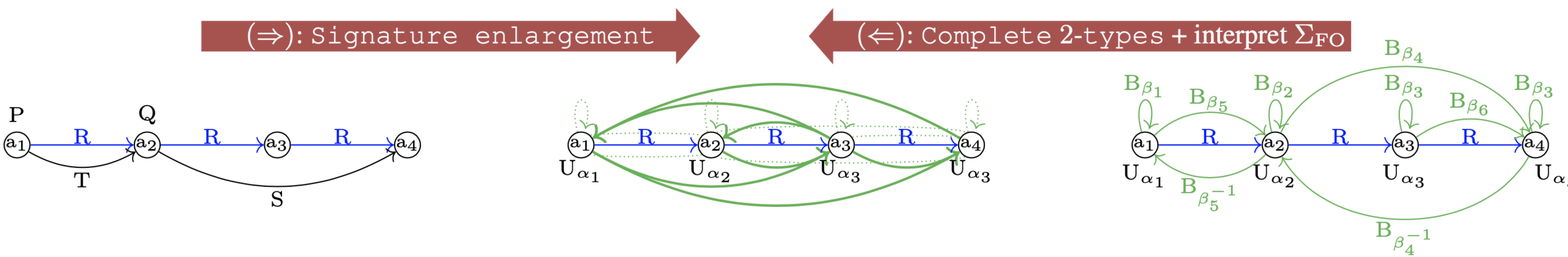
From RGF^2 to ICPDL.

Take φ. If satisfiable, φ has a model realizing at most exponentially many 2-types. Our vocabulary contains:

1. Fresh unary U_α per each 1-type α from $\boldsymbol{\alpha}_\varphi$.

2. Fresh binary B_β per each 2-type β from $\boldsymbol{\beta}_\varphi$ restricted to Σ_{FO} + all binary predicates in Σ_{reg} from φ.

Translation almost easy, e.g. $\forall x_1\, \gamma(x_1) \to \exists x_2\, \pi(x_1x_2) \wedge \phi(x_1x_2) \mapsto [\star]\left(\bigvee_{\alpha \models \gamma} \mathrm{U}_\alpha \to \langle \pi \cap \cup_{\beta \models \phi} \mathrm{B}_\beta \rangle.\top\right)$



From the Two-Variable Case to the General Case (Many details omitted)

From the Two-Variable Case to the General Case (Many details omitted)

Step I: Transform your input φ in RGF to Scott's normal form.

From the Two-Variable Case to the General Case (Many details omitted)

Step I: Transform your input φ in RGF to Scott's normal form.

Step II: Construct $\varphi_{\mathcal{R}}$ in RGF^2 and φ_{GF} in GF as follows:

From the Two-Variable Case to the General Case (Many details omitted)

Step I: Transform your input φ in RGF to Scott's normal form.

Step II: Construct $\varphi_{\mathcal{R}}$ in RGF^2 and φ_{GF} in GF as follows:

- φ_{GF} contains all $\exists^{\mathsf{FO}}$ and $\forall^{\mathsf{FO}}$ conjuncts of φ.

From the Two-Variable Case to the General Case (Many details omitted)

Step I: Transform your input φ in RGF to Scott's normal form.

Step II: Construct $\varphi_{\mathcal{R}}$ in RGF^2 and φ_{GF} in GF as follows:

- φ_{GF} contains all $\exists^{\mathrm{FO}}$ and $\forall^{\mathrm{FO}}$ conjuncts of φ.
- $\varphi_{\mathcal{R}}$ contains all $\exists^{\mathcal{R}}$ and $\forall^{\mathcal{R}}$ conjuncts of φ and the two-variable versions of $\exists^{\mathrm{FO}}$ and $\forall^{\mathrm{FO}}$ conjuncts.

From the Two-Variable Case to the General Case (Many details omitted)

Step I: Transform your input φ in RGF to Scott's normal form.

Step II: Construct $\varphi_{\mathcal{R}}$ in RGF^2 and φ_{GF} in GF as follows:

- φ_{GF} contains all $\exists^{\mathsf{FO}}$ and $\forall^{\mathsf{FO}}$ conjuncts of φ.
- $\varphi_{\mathcal{R}}$ contains all $\exists^{\mathcal{R}}$ and $\forall^{\mathcal{R}}$ conjuncts of φ and the two-variable versions of $\exists^{\mathsf{FO}}$ and $\forall^{\mathsf{FO}}$ conjuncts.
- Synchronization: require that models of $\varphi_{\mathcal{R}}$ and φ_{FO} realize the same sets of 1-types and 2-types.

From the Two-Variable Case to the General Case (Many details omitted)

Step I: Transform your input φ in RGF to Scott's normal form.

Step II: Construct $\varphi_{\mathcal{R}}$ in RGF^2 and φ_{GF} in GF as follows:

- φ_{GF} contains all $\exists^{\mathrm{FO}}$ and $\forall^{\mathrm{FO}}$ conjuncts of φ.
- $\varphi_{\mathcal{R}}$ contains all $\exists^{\mathcal{R}}$ and $\forall^{\mathcal{R}}$ conjuncts of φ and the two-variable versions of $\exists^{\mathrm{FO}}$ and $\forall^{\mathrm{FO}}$ conjuncts.
- Synchronization: require that models of $\varphi_{\mathcal{R}}$ and φ_{FO} realize the same sets of 1-types and 2-types.

Step III: Test both $\varphi_{\mathcal{R}}$ and φ_{GF} for satisfiability (this gives the right upper bound). Viola!

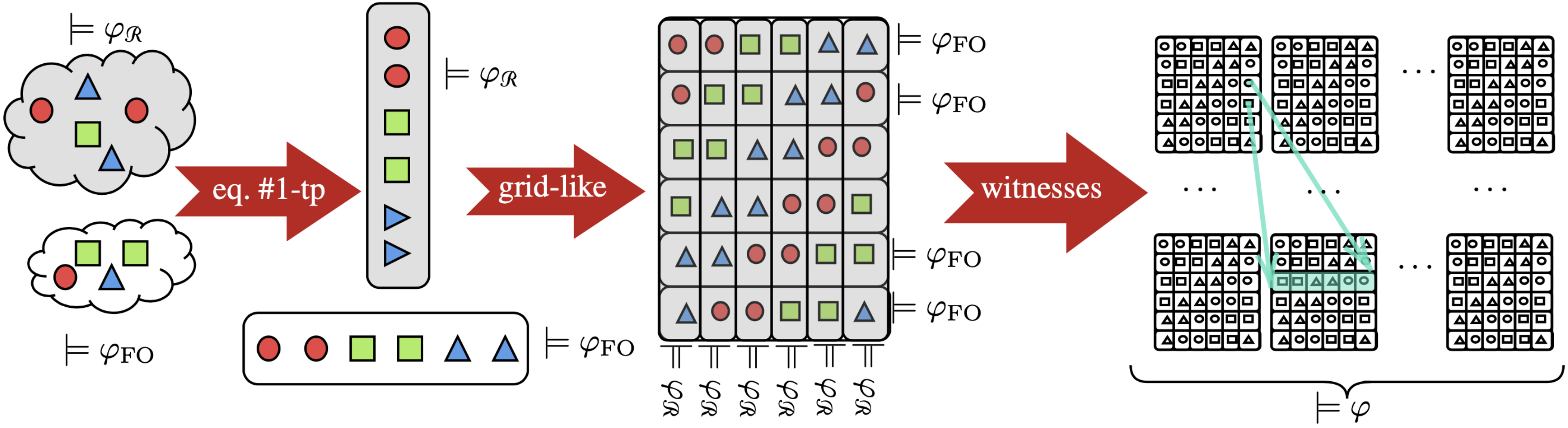
From the Two-Variable Case to the General Case (Many details omitted)

Step I: Transform your input φ in RGF to Scott's normal form.

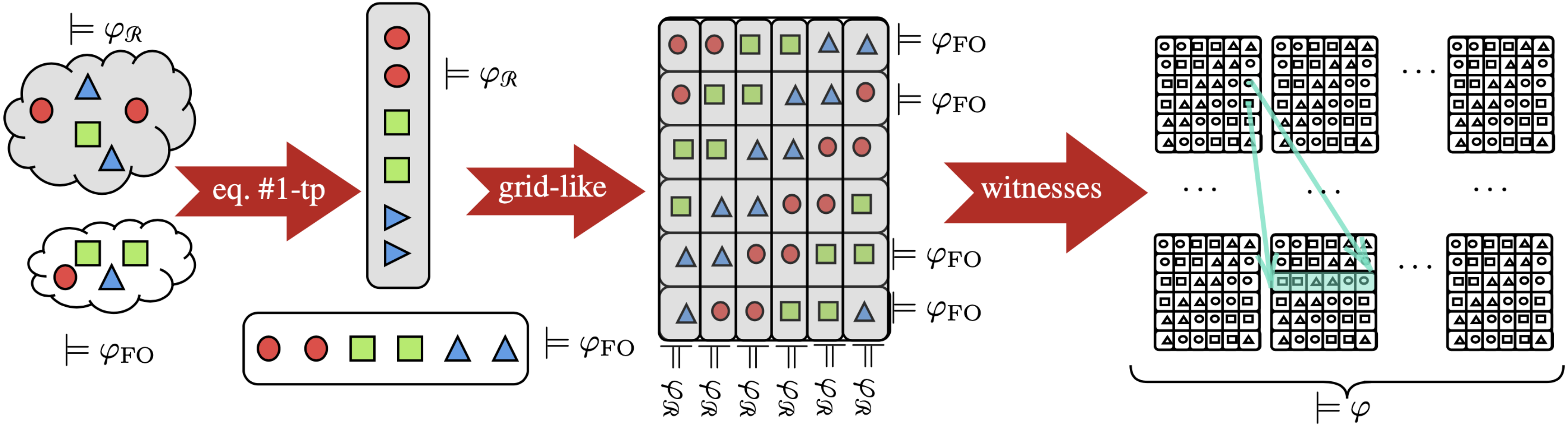
Step II: Construct $\varphi_{\mathcal{R}}$ in RGF^2 and φ_{GF} in GF as follows:

- φ_{GF} contains all $\exists^{\mathsf{FO}}$ and $\forall^{\mathsf{FO}}$ conjuncts of φ.
- $\varphi_{\mathcal{R}}$ contains all $\exists^{\mathcal{R}}$ and $\forall^{\mathcal{R}}$ conjuncts of φ and the two-variable versions of $\exists^{\mathsf{FO}}$ and $\forall^{\mathsf{FO}}$ conjuncts.
- Synchronization: require that models of $\varphi_{\mathcal{R}}$ and φ_{FO} realize the same sets of 1-types and 2-types.

Step III: Test both $\varphi_{\mathcal{R}}$ and φ_{GF} for satisfiability (this gives the right upper bound). Viola!

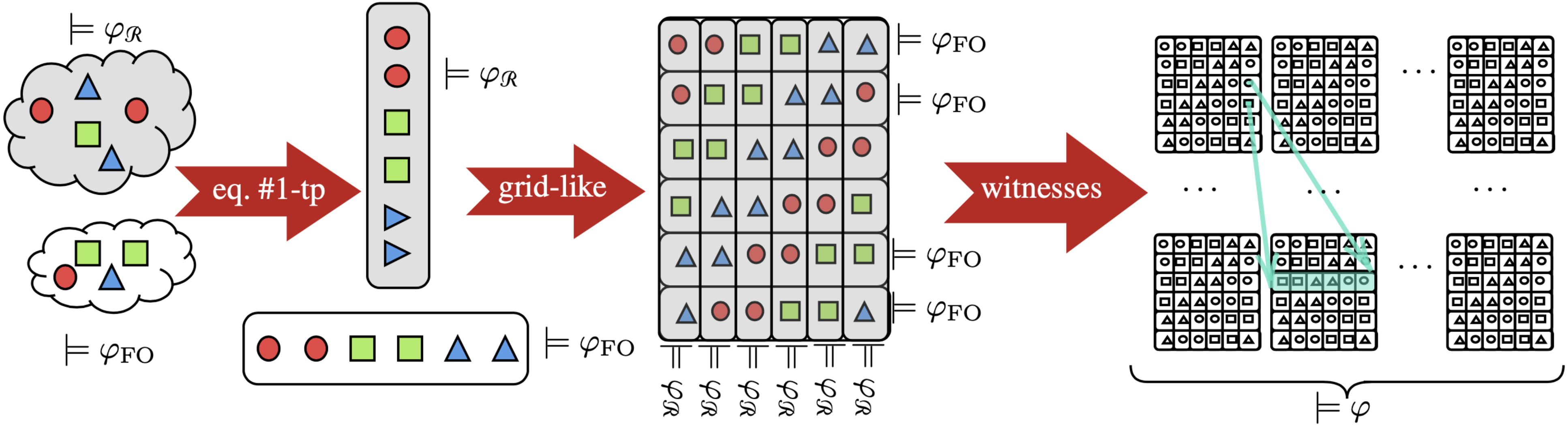


Correctness Proof: The Fusion



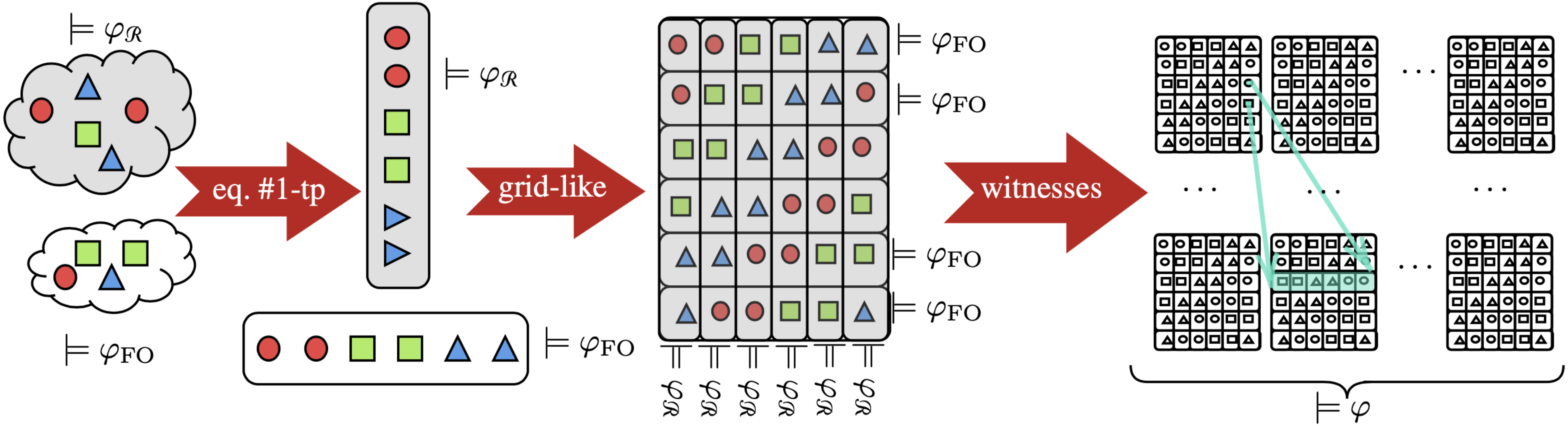
Correctness Proof: The Fusion

- Take models $\mathfrak{A} \models \varphi_{\mathcal{R}}$ and $\mathfrak{A} \models \varphi_{\mathsf{GF}}$.



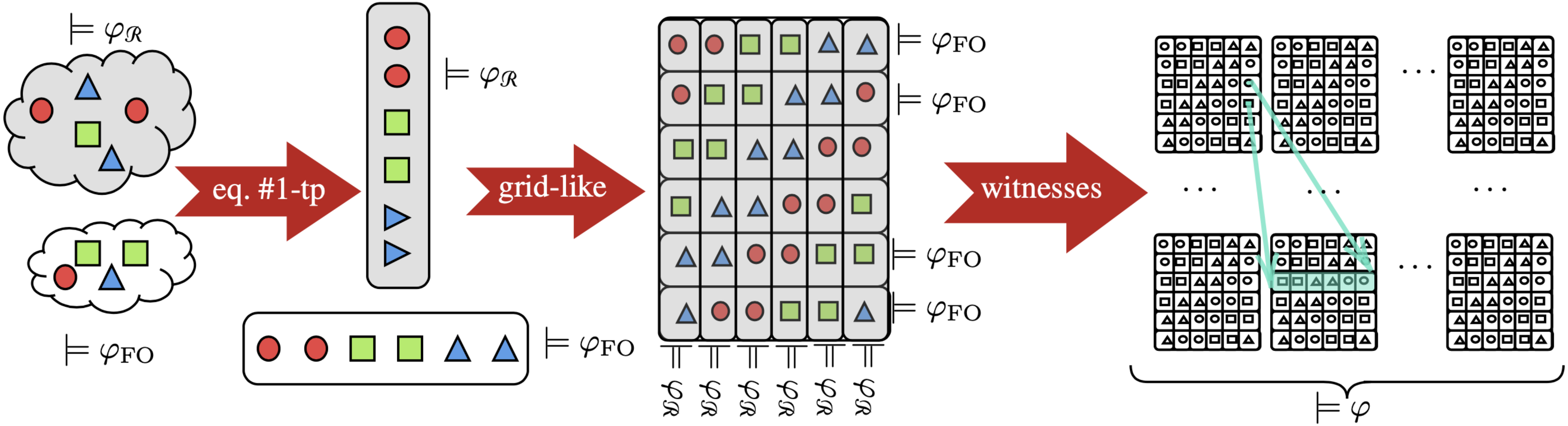
Correctness Proof: The Fusion

- Take models $\mathfrak{A} \models \varphi_{\mathcal{R}}$ and $\mathfrak{A} \models \varphi_{\mathsf{GF}}$.
- Equilize their domains by Löwenheim Skolem (w.l.o.g. to $\aleph_0$).



Correctness Proof: The Fusion

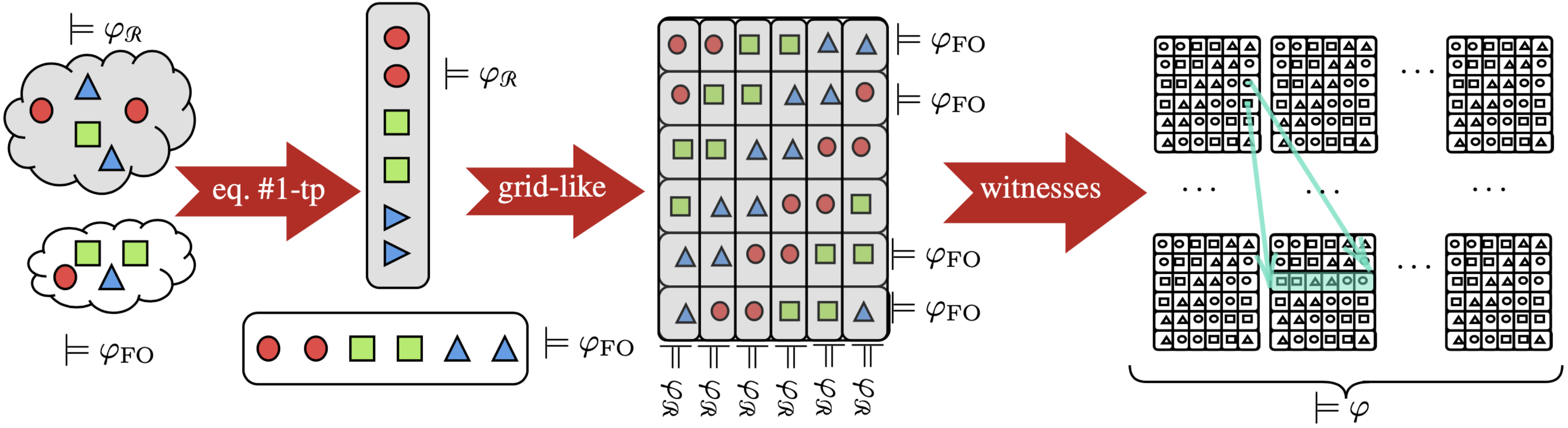
- Take models $\mathfrak{A} \models \varphi_{\mathcal{R}}$ and $\mathfrak{A} \models \varphi_{\mathsf{GF}}$.
- Equilize their domains by Löwenheim Skolem (w.l.o.g. to $\aleph_0$).
- We next ensure that for each 1-type α in $\boldsymbol{\alpha}_{\varphi}$:



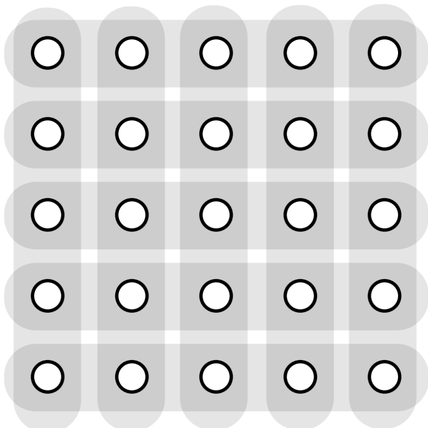
Correctness Proof: The Fusion

- Take models $\mathfrak{A} \models \varphi_{\mathcal{R}}$ and $\mathfrak{A} \models \varphi_{\mathsf{GF}}$.
- Equilize their domains by Löwenheim Skolem (w.l.o.g. to $\aleph_0$).
- We next ensure that for each 1-type α in $\boldsymbol{\alpha}_{\varphi}$:

  $\#\alpha$ in $\mathfrak{A}$ is equal to $\#\alpha$ in $\mathfrak{B}$.

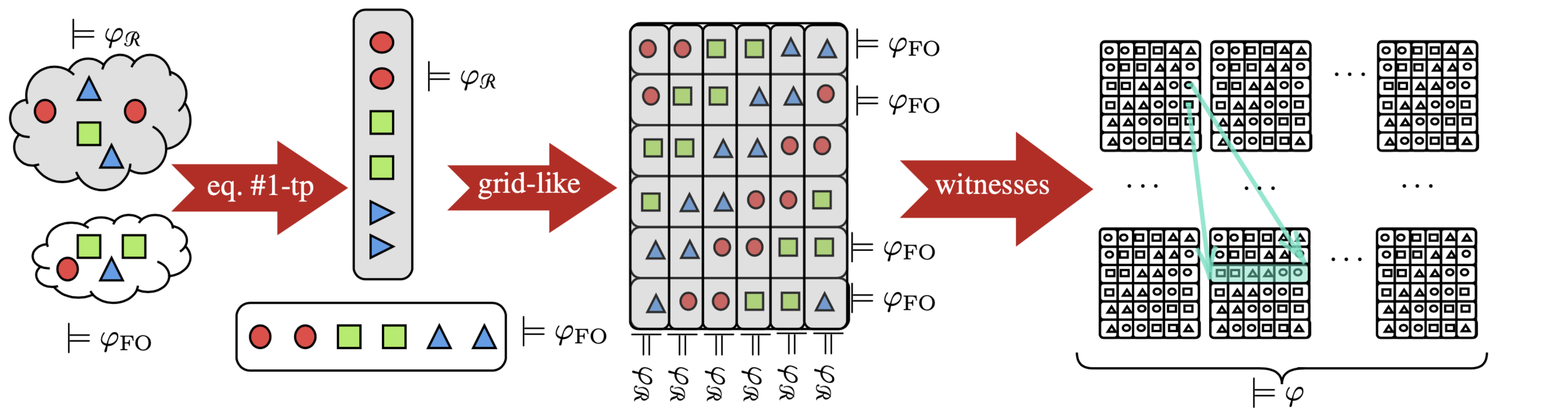


Correctness Proof: The Fusion

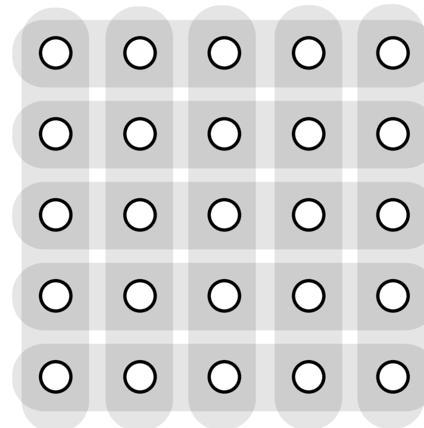


- Take models $\mathfrak{A} \models \varphi_{\mathcal{R}}$ and $\mathfrak{A} \models \varphi_{\mathsf{GF}}$.
- Equilize their domains by Löwenheim Skolem (w.l.o.g. to $\aleph_0$).
- We next ensure that for each 1-type α in $\boldsymbol{\alpha}_{\varphi}$:

  $\#\alpha$ in $\mathfrak{A}$ is equal to $\#\alpha$ in $\mathfrak{B}$.
- Form a two-dimensional grid $\mathfrak{C}$. Each row (column) isomorphic to $\mathfrak{A}$ ($\mathfrak{B}$).

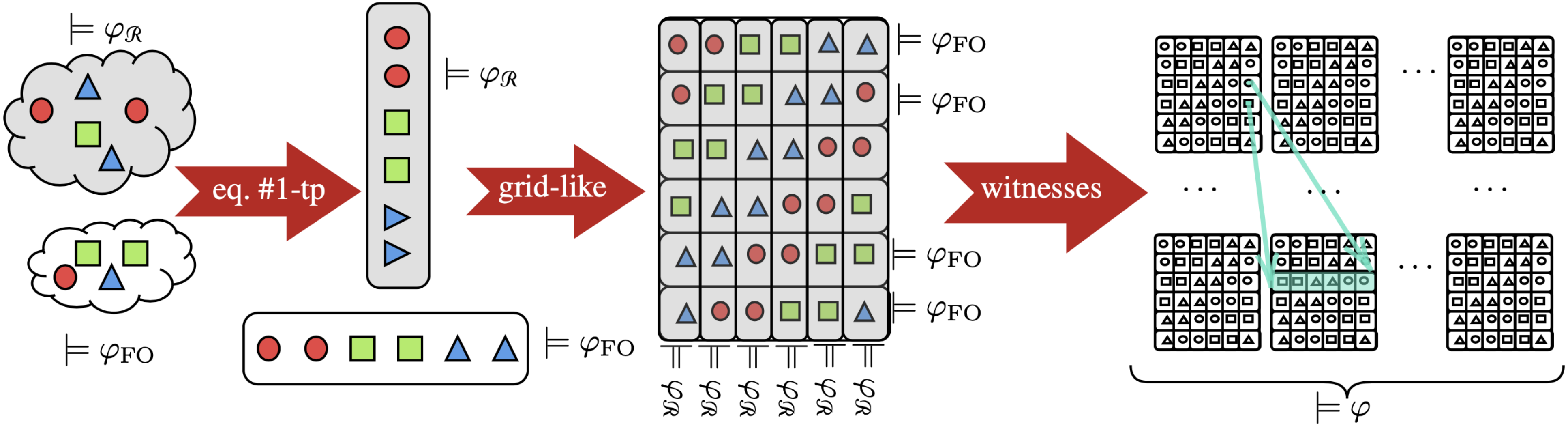


Correctness Proof: The Fusion

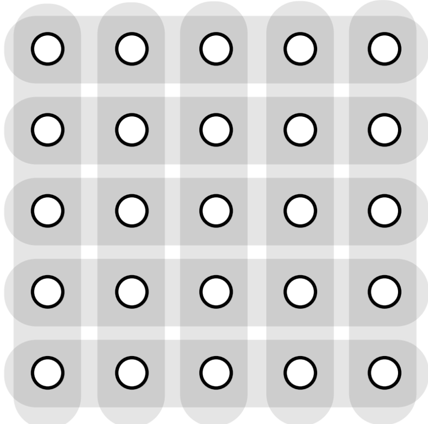


- Take models $\mathfrak{A} \models \varphi_{\mathcal{R}}$ and $\mathfrak{A} \models \varphi_{\mathsf{GF}}$.
- Equilize their domains by Löwenheim Skolem (w.l.o.g. to $\aleph_0$).
- We next ensure that for each 1-type α in $\boldsymbol{\alpha}_{\varphi}$:

  $\#\alpha$ in $\mathfrak{A}$ is equal to $\#\alpha$ in $\mathfrak{B}$.
- Form a two-dimensional grid $\mathfrak{C}$. Each row (column) isomorphic to $\mathfrak{A}$ ($\mathfrak{B}$).
- $\mathfrak{C}$ is almost a model of φ, but some elements may miss witnesses for $\exists^{\mathsf{FO}}$-conjuncts.



Correctness Proof: The Fusion



- Take models $\mathfrak{A} \models \varphi_{\mathcal{R}}$ and $\mathfrak{A} \models \varphi_{\mathsf{GF}}$.
- Equilize their domains by Löwenheim Skolem (w.l.o.g. to $\aleph_0$).
- We next ensure that for each 1-type α in $\boldsymbol{\alpha}_{\varphi}$:

  $\#\alpha$ in $\mathfrak{A}$ is equal to $\#\alpha$ in $\mathfrak{B}$.
- Form a two-dimensional grid $\mathfrak{C}$. Each row (column) isomorphic to $\mathfrak{A}$ ($\mathfrak{B}$).
- $\mathfrak{C}$ is almost a model of φ, but some elements may miss witnesses for $\exists^{\mathsf{FO}}$-conjuncts.
- Provide them with a circular-witnessing-scheme á la Grädel & Kolaitis & Vardi.

Conclusions

Conclusions

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

Conclusions

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by

Conclusions

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

Conclusions

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

Theorem (KR 2025)

The satisfiability problem for RGF is $2\textsc{ExpTime}$-complete.

Conclusions

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

Theorem (KR 2025)

The satisfiability problem for RGF is $2\textsc{ExpTime}$-complete.

Proof idea:

Conclusions

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

Theorem (KR 2025)

The satisfiability problem for RGF is $2\textsc{ExpTime}$-complete.

Proof idea: solve RGF^2 via a reduction to ICPDL and

Conclusions

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

Theorem (KR 2025)

The satisfiability problem for RGF is $2\textsc{ExpTime}$-complete.

Proof idea: solve RGF^2 via a reduction to ICPDL and lift it to the full RGF via a fusion.

Conclusions

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

Theorem (KR 2025)

The satisfiability problem for RGF is $2\textsc{ExpTime}$-complete.

Proof idea: solve RGF^2 via a reduction to ICPDL and lift it to the full RGF via a fusion.

Theorem (KR 2025)

Finite&Unrestricted Conjunctive Query Entailment Problems is undecidable for a tiny fragment of RGF.

Conclusions

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

Theorem (KR 2025)

The satisfiability problem for RGF is $2\textsc{ExpTime}$-complete.

Proof idea: solve RGF^2 via a reduction to ICPDL and lift it to the full RGF via a fusion.

Theorem (KR 2025)

Finite&Unrestricted Conjunctive Query Entailment Problems is undecidable for a tiny fragment of RGF.

Significantly improves prior results by Gottlob&Pieris&Tendera from ICALP 2013.

Conclusions

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

Theorem (KR 2025)

The satisfiability problem for RGF is $2\textsc{ExpTime}$-complete.

Proof idea: solve RGF^2 via a reduction to ICPDL and lift it to the full RGF via a fusion.

Theorem (KR 2025)

Finite&Unrestricted Conjunctive Query Entailment Problems is undecidable for a tiny fragment of RGF.

Significantly improves prior results by Gottlob&Pieris&Tendera from ICALP 2013.

Theorem (KR 2025)

$\exists$ $\textsc{ExpSpace}$-complete sublogic of $\mathsf{RGF}[\cdot^{+}, \cdot^{*}]$.

Conclusions

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

Theorem (KR 2025)

The satisfiability problem for RGF is $2\textsc{ExpTime}$-complete.

Proof idea: solve RGF^2 via a reduction to ICPDL and lift it to the full RGF via a fusion.

Theorem (KR 2025)

Finite&Unrestricted Conjunctive Query Entailment Problems is undecidable for a tiny fragment of RGF.

Significantly improves prior results by Gottlob&Pieris&Tendera from ICALP 2013.

Theorem (KR 2025)

$\exists$ $\textsc{ExpSpace}$-complete sublogic of $\mathsf{RGF}[\cdot^+, \cdot^*]$.

Combines ideas from the fluted fragment and 1-way guards.

Conclusions

Consider disjoint vocabularies Σ_{FO} and Σ_{reg}.

RGF is the least extension of GF over Σ_{FO} by allowing ICPDL programs $\pi(xy)$ over Σ_{reg} as binary guards.

Theorem (KR 2025)

The satisfiability problem for RGF is $2\textsc{ExpTime}$-complete.

Proof idea: solve RGF^2 via a reduction to ICPDL and lift it to the full RGF via a fusion.

Theorem (KR 2025)

Finite&Unrestricted Conjunctive Query Entailment Problems is undecidable for a tiny fragment of RGF.

Significantly improves prior results by Gottlob&Pieris&Tendera from ICALP 2013.

Theorem (KR 2025)

$\exists$ $\textsc{ExpSpace}$-complete sublogic of $\mathrm{RGF}[\cdot^{+}, \cdot^{*}]$.

Combines ideas from the fluted fragment and 1-way guards.

FINGO$^{\mu}$

FINITE GRAPH OPERATING AUTOMATA