Random Models and Guarded Logic

Oskar Fiuk

University of Wrocław

Grenoble, March 2026

From Hilbert's Program to Decidable Fragments

From Hilbert's Program to Decidable Fragments

- Satisfiability problem: *Given* φ*, does it have a model?*

From Hilbert's Program to Decidable Fragments

- Satisfiability problem: *Given φ, does it have a model?*
- Church–Turing: satisfiability for FO is undecidable

From Hilbert's Program to Decidable Fragments

- Satisfiability problem: *Given φ, does it have a model?*
- Church–Turing: satisfiability for FO is undecidable
- Major research direction: identify decidable fragments of FO

From Hilbert's Program to Decidable Fragments

- Satisfiability problem: *Given φ, does it have a model?*
- Church–Turing: satisfiability for FO is undecidable
- Major research direction: identify decidable fragments of FO
    - Monadic Fragment

From Hilbert's Program to Decidable Fragments

- Satisfiability problem: *Given φ, does it have a model?*
- Church–Turing: satisfiability for FO is undecidable
- Major research direction: identify decidable fragments of FO
  - Monadic Fragment
  - Prenex classes

From Hilbert's Program to Decidable Fragments

- Satisfiability problem: *Given φ, does it have a model?*
- Church–Turing: satisfiability for FO is undecidable
- Major research direction: identify decidable fragments of FO
  - Monadic Fragment
  - Prenex classes
  - Two-Variable Fragment (FO^2)

From Hilbert's Program to Decidable Fragments

- Satisfiability problem: *Given φ, does it have a model?*
- Church–Turing: satisfiability for FO is undecidable
- Major research direction: identify decidable fragments of FO
  - Monadic Fragment
  - Prenex classes
  - Two-Variable Fragment (FO^2)
  - Guarded Fragment (GF)

What is Guarded Fragment?

What is Guarded Fragment?

In the Guarded Fragment (GF), quantifiers must be guarded:

What is Guarded Fragment?

In the Guarded Fragment (GF), quantifiers must be guarded:

$$\exists \bar{x} \, (\alpha(\bar{x}, \bar{y}) \wedge \psi(\bar{x}, \bar{y}))$$

What is Guarded Fragment?

In the Guarded Fragment (GF), quantifiers must be guarded:

$$\exists \bar{x}\,(\alpha(\bar{x},\bar{y}) \land \psi(\bar{x},\bar{y})) \qquad \forall \bar{x}\,(\alpha(\bar{x},\bar{y}) \to \psi(\bar{x},\bar{y}))$$

What is Guarded Fragment?

In the Guarded Fragment (GF), quantifiers must be guarded:

$$\exists \bar{x}\,(\alpha(\bar{x},\bar{y}) \wedge \psi(\bar{x},\bar{y})) \qquad \forall \bar{x}\,(\alpha(\bar{x},\bar{y}) \rightarrow \psi(\bar{x},\bar{y}))$$

$\alpha(\bar{x},\bar{y})$ is a guard $=$ an atomic formula mentioning all variables in $\bar{x}$ and $\bar{y}$

What is Guarded Fragment?

In the Guarded Fragment (GF), quantifiers must be guarded:

$$\exists \bar{x}\,(\alpha(\bar{x}, \bar{y}) \land \psi(\bar{x}, \bar{y})) \qquad \forall \bar{x}\,(\alpha(\bar{x}, \bar{y}) \to \psi(\bar{x}, \bar{y}))$$

$\alpha(\bar{x}, \bar{y})$ is a guard $=$ an atomic formula mentioning all variables in $\bar{x}$ and $\bar{y}$

$$\forall p, s\,(\mathsf{supervises}(p, s) \to \exists t\,(\mathsf{prepares}(s, t) \land \mathsf{thesis}(t)))$$

What is Guarded Fragment?

In the Guarded Fragment (GF), quantifiers must be guarded:

$$\exists \bar{x} \, (\alpha(\bar{x}, \bar{y}) \wedge \psi(\bar{x}, \bar{y})) \qquad \forall \bar{x} \, (\alpha(\bar{x}, \bar{y}) \rightarrow \psi(\bar{x}, \bar{y}))$$

$\alpha(\bar{x}, \bar{y})$ is a guard $=$ an atomic formula mentioning all variables in $\bar{x}$ and $\bar{y}$

$$\forall p, s \, (\mathsf{supervises}(p, s) \rightarrow \exists t \, (\mathsf{prepares}(s, t) \wedge \mathsf{thesis}(t)))$$

$$\forall d, d' \, (\underbrace{(\mathsf{dean}(d) \wedge \mathsf{dean}(d'))}_{\text{not atomic}} \rightarrow d = d')$$

What is Guarded Fragment?

In the Guarded Fragment (GF), quantifiers must be guarded:

$$\exists \bar{x}\,(\alpha(\bar{x},\bar{y}) \wedge \psi(\bar{x},\bar{y})) \qquad \forall \bar{x}\,(\alpha(\bar{x},\bar{y}) \rightarrow \psi(\bar{x},\bar{y}))$$

$\alpha(\bar{x},\bar{y})$ is a guard $=$ an atomic formula mentioning all variables in $\bar{x}$ and $\bar{y}$

$$\forall p, s\,(\mathsf{supervises}(p,s) \rightarrow \exists t\,(\mathsf{prepares}(s,t) \wedge \mathsf{thesis}(t)))$$

$$\forall d, d'\,(\underbrace{(\mathsf{dean}(d) \wedge \mathsf{dean}(d'))}_{\text{not atomic}} \rightarrow d = d')$$

$$\forall d\,(\mathsf{dean}(d) \rightarrow \forall d'\,(\underbrace{\mathsf{dean}(d')}_{d \text{ is not mentioned}} \rightarrow d = d'))$$

About Guarded Fragment

About Guarded Fragment

- Introduced by Andréka, van Benthem, Németi

About Guarded Fragment

- Introduced by Andréka, van Benthem, Németi
- Motivation: explain the key properties of modal logic

About Guarded Fragment

- Introduced by Andréka, van Benthem, Németi
- Motivation: explain the key properties of modal logic
  - Talks about possibility ($\Diamond$) and necessity ($\Box$)

About Guarded Fragment

- Introduced by Andréka, van Benthem, Németi
- Motivation: explain the key properties of modal logic
  - Talks about possibility ($\Diamond$) and necessity ($\Box$)
  - Semantics: set of states S with an accessibility relation $R \subseteq S \times S$

About Guarded Fragment

- Introduced by Andréka, van Benthem, Németi
- Motivation: explain the key properties of modal logic
  - Talks about possibility ($\Diamond$) and necessity ($\Box$)
  - Semantics: set of states S with an accessibility relation $R \subseteq S \times S$
  - Viewed as a fragment of FO (standard translation):

About Guarded Fragment

- Introduced by Andréka, van Benthem, Németi
- Motivation: explain the key properties of modal logic
  - Talks about possibility ($\Diamond$) and necessity ($\Box$)
  - Semantics: set of states S with an accessibility relation $R \subseteq S \times S$
  - Viewed as a fragment of FO (standard translation):

$$\Diamond\psi \;\rightsquigarrow\; \exists y\,(R(x,y) \wedge \psi(y))$$

About Guarded Fragment

- Introduced by Andréka, van Benthem, Németi
- Motivation: explain the key properties of modal logic
  - Talks about possibility ($\Diamond$) and necessity ($\Box$)
  - Semantics: set of states S with an accessibility relation $R \subseteq S \times S$
  - Viewed as a fragment of FO (standard translation):

$$\Diamond\psi \;\rightsquigarrow\; \exists y\,(R(x,y) \wedge \psi(y)) \qquad \Box\psi \;\rightsquigarrow\; \forall y\,(R(x,y) \rightarrow \psi(y))$$

About Guarded Fragment

- Introduced by Andréka, van Benthem, Németi
- Motivation: explain the key properties of modal logic
  - Talks about possibility ($\Diamond$) and necessity ($\Box$)
  - Semantics: set of states S with an accessibility relation $R \subseteq S \times S$
  - Viewed as a fragment of FO (standard translation):

$$\Diamond\psi \ \rightsquigarrow\ \exists y\,(R(x,y) \land \psi(y)) \qquad \Box\psi \ \rightsquigarrow\ \forall y\,(R(x,y) \rightarrow \psi(y))$$

  - GF generalises modal logic: accessibility relation $\rightsquigarrow$ guards

About Guarded Fragment

- Introduced by Andréka, van Benthem, Németi
- Motivation: explain the key properties of modal logic
  - Talks about possibility ($\Diamond$) and necessity ($\Box$)
  - Semantics: set of states S with an accessibility relation $R \subseteq S \times S$
  - Viewed as a fragment of FO (standard translation):

$$\Diamond\psi \ \rightsquigarrow\ \exists y\,(R(x,y) \wedge \psi(y)) \qquad \Box\psi \ \rightsquigarrow\ \forall y\,(R(x,y) \rightarrow \psi(y))$$

  - GF generalises modal logic: accessibility relation $\rightsquigarrow$ guards
- Satisfiability for GF:

About Guarded Fragment

- Introduced by Andréka, van Benthem, Németi
- Motivation: explain the key properties of modal logic
  - Talks about possibility ($\Diamond$) and necessity ($\Box$)
  - Semantics: set of states S with an accessibility relation $R \subseteq S \times S$
  - Viewed as a fragment of FO (standard translation):

$$\Diamond\psi \rightsquigarrow \exists y\,(R(x,y) \wedge \psi(y)) \qquad \Box\psi \rightsquigarrow \forall y\,(R(x,y) \rightarrow \psi(y))$$

  - GF generalises modal logic: accessibility relation $\rightsquigarrow$ guards
- Satisfiability for GF:
  - Decidable

About Guarded Fragment

- Introduced by Andréka, van Benthem, Németi
- Motivation: explain the key properties of modal logic
  - Talks about possibility ($\Diamond$) and necessity ($\Box$)
  - Semantics: set of states S with an accessibility relation $R \subseteq S \times S$
  - Viewed as a fragment of FO (standard translation):

$$\Diamond\psi \rightsquigarrow \exists y\,(R(x,y) \wedge \psi(y)) \qquad \Box\psi \rightsquigarrow \forall y\,(R(x,y) \rightarrow \psi(y))$$

  - GF generalises modal logic: accessibility relation $\rightsquigarrow$ guards
- Satisfiability for GF:
  - Decidable and 2-$\textsc{ExpTime}$-complete (Grädel)

About Guarded Fragment

- Introduced by Andréka, van Benthem, Németi
- Motivation: explain the key properties of modal logic
  - Talks about possibility ($\Diamond$) and necessity ($\Box$)
  - Semantics: set of states S with an accessibility relation $R \subseteq S \times S$
  - Viewed as a fragment of FO (standard translation):

$$\Diamond\psi \;\rightsquigarrow\; \exists y\,(R(x,y) \wedge \psi(y)) \qquad \Box\psi \;\rightsquigarrow\; \forall y\,(R(x,y) \rightarrow \psi(y))$$

  - GF generalises modal logic: accessibility relation $\rightsquigarrow$ guards
- Satisfiability for GF:
  - Decidable and 2-ExpTime-complete (Grädel)
  - Robust: fixed points, conjunctive queries

About Guarded Fragment

- Introduced by Andréka, van Benthem, Németi
- Motivation: explain the key properties of modal logic
  - Talks about possibility ($\Diamond$) and necessity ($\Box$)
  - Semantics: set of states S with an accessibility relation $R \subseteq S \times S$
  - Viewed as a fragment of FO (standard translation):

$$\Diamond\psi \ \rightsquigarrow\ \exists y\,(R(x,y) \wedge \psi(y)) \qquad \Box\psi \ \rightsquigarrow\ \forall y\,(R(x,y) \rightarrow \psi(y))$$

  - GF generalises modal logic: accessibility relation $\rightsquigarrow$ guards
- Satisfiability for GF:
  - Decidable and 2-ExpTime-complete (Grädel)
  - Robust: fixed points, conjunctive queries
- GF is naturally connected to:

About Guarded Fragment

- Introduced by Andréka, van Benthem, Németi
- Motivation: explain the key properties of modal logic
  - Talks about possibility ($\Diamond$) and necessity ($\Box$)
  - Semantics: set of states S with an accessibility relation $R \subseteq S \times S$
  - Viewed as a fragment of FO (standard translation):

$$\Diamond\psi \rightsquigarrow \exists y\,(R(x,y) \wedge \psi(y)) \qquad \Box\psi \rightsquigarrow \forall y\,(R(x,y) \rightarrow \psi(y))$$

  - GF generalises modal logic: accessibility relation $\rightsquigarrow$ guards
- Satisfiability for GF:
  - Decidable and 2-ExpTime-complete (Grädel)
  - Robust: fixed points, conjunctive queries
- GF is naturally connected to:
  - Formal verification (temporal logics)

About Guarded Fragment

- Introduced by Andréka, van Benthem, Németi
- Motivation: explain the key properties of modal logic
  - Talks about possibility ($\Diamond$) and necessity ($\Box$)
  - Semantics: set of states S with an accessibility relation $R \subseteq S \times S$
  - Viewed as a fragment of FO (standard translation):

$$\Diamond\psi \ \rightsquigarrow \ \exists y\,(R(x,y) \wedge \psi(y)) \qquad \Box\psi \ \rightsquigarrow \ \forall y\,(R(x,y) \rightarrow \psi(y))$$

  - GF generalises modal logic: accessibility relation $\rightsquigarrow$ guards
- Satisfiability for GF:
  - Decidable and 2-ExpTime-complete (Grädel)
  - Robust: fixed points, conjunctive queries
- GF is naturally connected to:
  - Formal verification (temporal logics)
  - Knowledge representation (description logics)

About Guarded Fragment

- Introduced by Andréka, van Benthem, Németi
- Motivation: explain the key properties of modal logic
  - Talks about possibility ($\Diamond$) and necessity ($\Box$)
  - Semantics: set of states S with an accessibility relation $R \subseteq S \times S$
  - Viewed as a fragment of FO (standard translation):

$$\Diamond\psi \rightsquigarrow \exists y\,(R(x,y) \land \psi(y)) \qquad \Box\psi \rightsquigarrow \forall y\,(R(x,y) \rightarrow \psi(y))$$

  - GF generalises modal logic: accessibility relation $\rightsquigarrow$ guards
- Satisfiability for GF:
  - Decidable and 2-ExpTime-complete (Grädel)
  - Robust: fixed points, conjunctive queries
- GF is naturally connected to:
  - Formal verification (temporal logics)
  - Knowledge representation (description logics)
  - Database theory

Finite Model Property

Finite Model Property

The Guarded Fragment enjoys the finite model property:

Finite Model Property

The Guarded Fragment enjoys the finite model property:

$$\varphi \in \mathrm{GF} \text{ satisfiable} \implies \varphi \text{ has a finite model}$$

Finite Model Property

The Guarded Fragment enjoys the finite model property:

$$\varphi \in \mathrm{GF} \text{ satisfiable} \implies \varphi \text{ has a finite model}$$

Related question: *How large must such a model be?*

Finite Model Property

The Guarded Fragment enjoys the finite model property:

$$\varphi \in \mathrm{GF} \text{ satisfiable } \implies \varphi \text{ has a finite model}$$

Related question: *How large must such a model be?*

- Grädel: triply exponential upper bound

Finite Model Property

The Guarded Fragment enjoys the finite model property:

$$\varphi \in \mathrm{GF} \text{ satisfiable } \implies \varphi \text{ has a finite model}$$

Related question: *How large must such a model be?*

- Grädel: triply exponential upper bound
  - Elegant but relies on a black box: Herwig's theorem

Finite Model Property

The Guarded Fragment enjoys the finite model property:

$$\varphi \in \mathrm{GF} \text{ satisfiable } \implies \varphi \text{ has a finite model}$$

Related question: *How large must such a model be?*

- Grädel: triply exponential upper bound
  - Elegant but relies on a black box: Herwig's theorem
- Bárány, Gottlob, Otto: optimal doubly exponential bound

Finite Model Property

The Guarded Fragment enjoys the finite model property:

$$\varphi \in \mathrm{GF} \text{ satisfiable} \implies \varphi \text{ has a finite model}$$

Related question: *How large must such a model be?*

- Grädel: triply exponential upper bound
  - Elegant but relies on a black box: Herwig's theorem
- Bárány, Gottlob, Otto: optimal doubly exponential bound
  - Allows for conjunctive queries: $\varphi \wedge \neg \mathrm{CQ}$

Finite Model Property

The Guarded Fragment enjoys the finite model property:

$$\varphi \in \mathrm{GF} \text{ satisfiable } \implies \varphi \text{ has a finite model}$$

Related question: *How large must such a model be?*

- Grädel: triply exponential upper bound
  - Elegant but relies on a black box: Herwig's theorem
- Bárány, Gottlob, Otto: optimal doubly exponential bound
  - Allows for conjunctive queries: $\varphi \wedge \neg \mathrm{CQ}$
  - Technically challenging: generalises Rosati's finite chase

Finite Model Property

The Guarded Fragment enjoys the finite model property:

$$\varphi \in \mathrm{GF} \text{ satisfiable} \implies \varphi \text{ has a finite model}$$

Related question: *How large must such a model be?*

- Grädel: triply exponential upper bound
  - Elegant but relies on a black box: Herwig's theorem
- Bárány, Gottlob, Otto: optimal doubly exponential bound
  - Allows for conjunctive queries: $\varphi \wedge \neg\mathrm{CQ}$
  - Technically challenging: generalises Rosati's finite chase

Can we obtain the finite model property for GF in a simpler way?

Finite Model Property

The Guarded Fragment enjoys the finite model property:

$$\varphi \in \mathrm{GF} \text{ satisfiable} \implies \varphi \text{ has a finite model}$$

Related question: *How large must such a model be?*

- Grädel: triply exponential upper bound
  - Elegant but relies on a black box: Herwig's theorem
- Bárány, Gottlob, Otto: optimal doubly exponential bound
  - Allows for conjunctive queries: $\varphi \wedge \neg\mathrm{CQ}$
  - Technically challenging: generalises Rosati's finite chase

Can we obtain the finite model property for GF *in a simpler way?*

Probabilistic Method

Probabilistic Methods in Logic

Probabilistic Methods in Logic

- Early: Fagin's proof of the 0–1 law for first-order logic

Probabilistic Methods in Logic

- Early: Fagin's proof of the 0–1 law for first-order logic

  φ is true in the infinite random graph $\implies$ φ has a finite model

Probabilistic Methods in Logic

- Early: Fagin's proof of the 0–1 law for first-order logic

  φ is true in the infinite random graph $\implies$ φ has a finite model

- Inspiration: Gurevich and Shelah's approach for the Gödel class

Probabilistic Methods in Logic

- Early: Fagin's proof of the 0–1 law for first-order logic

  φ is true in the infinite random graph $\implies$ φ has a finite model

- Inspiration: Gurevich and Shelah's approach for the Gödel class

  $\varphi = \exists \ldots \exists \forall \forall \exists \ldots \exists \, \psi$

Probabilistic Methods in Logic

- Early: Fagin's proof of the 0–1 law for first-order logic

  $$\varphi \text{ is true in the infinite random graph } \implies \varphi \text{ has a finite model}$$

- Inspiration: Gurevich and Shelah's approach for the Gödel class

  $$\varphi = \exists \ldots \exists \forall \forall \exists \ldots \exists \, \psi \text{ is satisfiable } \implies \varphi \text{ has a finite model}$$

Probabilistic Methods in Logic

- Early: Fagin's proof of the 0–1 law for first-order logic

$$\varphi \text{ is true in the infinite random graph } \implies \varphi \text{ has a finite model}$$

- Inspiration: Gurevich and Shelah's approach for the Gödel class

$$\varphi = \exists \dots \exists \forall \forall \exists \dots \exists \, \psi \text{ is satisfiable } \implies \varphi \text{ has a finite model}$$

Idea of the probabilistic proof:

Probabilistic Methods in Logic

- Early: Fagin's proof of the 0–1 law for first-order logic

$$\varphi \text{ is true in the infinite random graph } \implies \varphi \text{ has a finite model}$$

- Inspiration: Gurevich and Shelah's approach for the Gödel class

$$\varphi = \exists \ldots \exists \forall \forall \exists \ldots \exists \, \psi \text{ is satisfiable } \implies \varphi \text{ has a finite model}$$

Idea of the probabilistic proof: φ satisfiable $\implies$ φ has a finite model

Probabilistic Methods in Logic

- Early: Fagin's proof of the 0–1 law for first-order logic

$$\varphi \text{ is true in the infinite random graph} \implies \varphi \text{ has a finite model}$$

- Inspiration: Gurevich and Shelah's approach for the Gödel class

$$\varphi = \exists \ldots \exists \forall \forall \exists \ldots \exists\, \psi \text{ is satisfiable} \implies \varphi \text{ has a finite model}$$

Idea of the probabilistic proof: φ satisfiable $\implies$ φ has a finite model

1. Start with an infinite model $\mathcal{A} \models \varphi$

Probabilistic Methods in Logic

- Early: Fagin's proof of the 0–1 law for first-order logic

  φ is true in the infinite random graph $\implies$ φ has a finite model

- Inspiration: Gurevich and Shelah's approach for the Gödel class

  $\varphi = \exists \ldots \exists \forall \forall \exists \ldots \exists \, \psi$ is satisfiable $\implies$ φ has a finite model

Idea of the probabilistic proof: φ satisfiable $\implies$ φ has a finite model

1. Start with an infinite model $\mathcal{A} \models \varphi$
2. Define a randomised construction that

   given $\mathcal{A}$ and $n \in \mathbb{N}$, builds a structure $\mathcal{B}_n$ on n elements

Probabilistic Methods in Logic

- Early: Fagin's proof of the 0–1 law for first-order logic

$$\varphi \text{ is true in the infinite random graph } \implies \varphi \text{ has a finite model}$$

- Inspiration: Gurevich and Shelah's approach for the Gödel class

$$\varphi = \exists \dots \exists \forall \forall \exists \dots \exists\, \psi \text{ is satisfiable } \implies \varphi \text{ has a finite model}$$

Idea of the probabilistic proof: φ satisfiable $\implies$ φ has a finite model

1. Start with an infinite model $\mathcal{A} \models \varphi$
2. Define a randomised construction that

   given $\mathcal{A}$ and $n \in \mathbb{N}$, builds a structure $\mathcal{B}_n$ on n elements

3. Show that there exists a threshold t_φ such that

$$n \text{ exceeds } t_\varphi \implies \mathbb{P}[\mathcal{B}_n \models \varphi] > 0$$

Probabilistic Methods in Logic

- Early: Fagin's proof of the 0–1 law for first-order logic

$$\varphi \text{ is true in the infinite random graph} \implies \varphi \text{ has a finite model}$$

- Inspiration: Gurevich and Shelah's approach for the Gödel class

$$\varphi = \exists \dots \exists \forall\forall \exists \dots \exists\, \psi \text{ is satisfiable} \implies \varphi \text{ has a finite model}$$

Idea of the probabilistic proof: φ satisfiable $\implies$ φ has a finite model

1. Start with an infinite model $\mathcal{A} \models \varphi$
2. Define a randomised construction that

   given $\mathcal{A}$ and $n \in \mathbb{N}$, builds a structure $\mathcal{B}_n$ on n elements

3. Show that there exists a threshold t_φ such that

   $$n \text{ exceeds } t_\varphi \implies \mathbb{P}[\mathcal{B}_n \models \varphi] > 0$$

4. Hence, some $\mathcal{B}_n$ is a finite model of φ

Probabilistic Methods in Logic

- Early: Fagin's proof of the 0–1 law for first-order logic

$$\varphi \text{ is true in the infinite random graph } \implies \varphi \text{ has a finite model}$$

- Inspiration: Gurevich and Shelah's approach for the Gödel class

$$\varphi = \exists \dots \exists \forall \forall \exists \dots \exists \, \psi \text{ is satisfiable } \implies \varphi \text{ has a finite model}$$

Idea of the probabilistic proof: φ satisfiable $\implies$ φ has a finite model

1. Start with an infinite model $\mathcal{A} \models \varphi$
2. Define a randomised construction that

   given $\mathcal{A}$ and $n \in \mathbb{N}$, builds a structure $\mathcal{B}_n$ on n elements

3. Show that there exists a threshold t_φ such that

   $$n \text{ exceeds } t_\varphi \implies \mathbb{P}[\mathcal{B}_n \models \varphi] > 0$$

4. Hence, some $\mathcal{B}_n$ is a finite model of φ
5. The threshold t_φ yields the size bound

Random Models for Guarded Sentences

Random Models for Guarded Sentences

- Let $\varphi \in \mathrm{GF}$ be satisfiable in normal form:

Random Models for Guarded Sentences

- Let $\varphi \in \mathrm{GF}$ be satisfiable in normal form:

$$\varphi \;=\; \bigwedge_t \forall \bar{x} \; (\alpha_t(\bar{x}) \to \exists y \; \psi_t(\bar{x}, y))$$

Random Models for Guarded Sentences

- Let $\varphi \in \mathrm{GF}$ be satisfiable in normal form:

$$\varphi \;=\; \bigwedge_t \forall \bar{x}\;(\alpha_t(\bar{x}) \rightarrow \exists y\;\psi_t(\bar{x}, y))$$

where each α_t is a guard and each ψ_t is quantifier-free

Random Models for Guarded Sentences

- Let $\varphi \in \mathrm{GF}$ be satisfiable in normal form:

$$\varphi \;=\; \bigwedge_t \forall \bar{x}\; (\alpha_t(\bar{x}) \rightarrow \exists y\; \psi_t(\bar{x}, y))$$

  where each α_t is a guard and each ψ_t is quantifier-free
- Fix an infinite model $\mathcal{A} \models \varphi$.

Random Models for Guarded Sentences

- Let $\varphi \in \mathrm{GF}$ be satisfiable in normal form:

$$\varphi \;=\; \bigwedge_t \forall \bar{x}\; (\alpha_t(\bar{x}) \to \exists y\; \psi_t(\bar{x}, y))$$

  where each α_t is a guard and each ψ_t is quantifier-free

- Fix an infinite model $\mathcal{A} \models \varphi$. For $k \in \mathbb{N}$, let

$$\boldsymbol{\tau}_k^{\mathcal{A}} \;=\; \{\text{ (atomic) } k\text{-types realised in } \mathcal{A} \;\}$$

Random Models for Guarded Sentences

- Let $\varphi \in \mathrm{GF}$ be satisfiable in normal form:

$$\varphi \;=\; \bigwedge_t \forall \bar{x}\; (\alpha_t(\bar{x}) \to \exists y\; \psi_t(\bar{x}, y))$$

  where each α_t is a guard and each ψ_t is quantifier-free

- Fix an infinite model $\mathcal{A} \models \varphi$. For $k \in \mathbb{N}$, let

$$\boldsymbol{\tau}_k^{\mathcal{A}} \;=\; \{\; \text{(atomic) } k\text{-types realised in } \mathcal{A} \;\}$$

  k-type $=$ complete description of all atomic facts on a k-tuple

Random Models for Guarded Sentences

- Let $\varphi \in \mathrm{GF}$ be satisfiable in normal form:

$$\varphi \ = \ \bigwedge_t \forall \bar{x} \ (\alpha_t(\bar{x}) \to \exists y \ \psi_t(\bar{x}, y))$$

  where each α_t is a guard and each ψ_t is quantifier-free

- Fix an infinite model $\mathcal{A} \models \varphi$. For $k \in \mathbb{N}$, let

$$\boldsymbol{\tau}_k^{\mathcal{A}} \ = \ \{ \text{ (atomic) } k\text{-types realised in } \mathcal{A} \ \}$$

  k-type $=$ complete description of all atomic facts on a k-tuple

- Naïve construction of $\mathcal{B}_n$ on n elements:

Random Models for Guarded Sentences

- Let $\varphi \in \mathrm{GF}$ be satisfiable in normal form:

$$\varphi \;=\; \bigwedge_t \forall \bar{x}\; (\alpha_t(\bar{x}) \to \exists y\; \psi_t(\bar{x}, y))$$

  where each α_t is a guard and each ψ_t is quantifier-free

- Fix an infinite model $\mathcal{A} \models \varphi$. For $k \in \mathbb{N}$, let

$$\boldsymbol{\tau}_k^{\mathcal{A}} \;=\; \{\; \text{(atomic) } k\text{-types realised in } \mathcal{A} \;\}$$

  k-type $=$ complete description of all atomic facts on a k-tuple

- Naïve construction of $\mathcal{B}_n$ on n elements:

```
initialise B_n as the empty structure on {1, ..., n};
foreach tuple b̄ do
    choose a k-type τ ∈ τ_k^A uniformly at random (k = |b̄|);
    make b̄ realise τ;
```

Random Models for Guarded Sentences

- Let $\varphi \in \mathrm{GF}$ be satisfiable in normal form:

$$\varphi \;=\; \bigwedge_t \forall \bar{x}\, (\alpha_t(\bar{x}) \to \exists y\; \psi_t(\bar{x}, y))$$

  where each α_t is a guard and each ψ_t is quantifier-free

- Fix an infinite model $\mathcal{A} \models \varphi$. For $k \in \mathbb{N}$, let

$$\boldsymbol{\tau}_k^{\mathcal{A}} \;=\; \{\ \text{(atomic) } k\text{-types realised in } \mathcal{A}\ \}$$

  k-type $=$ complete description of all atomic facts on a k-tuple

- Naïve construction of $\mathcal{B}_n$ on n elements:

```
initialise B_n as the empty structure on {1, ..., n};
foreach tuple b̄ do
    choose a k-type τ ∈ τ_k^A uniformly at random (k = |b̄|);
    make b̄ realise τ;
```

- Problem: overlapping tuples may impose inconsistent facts

Random Models for Guarded Sentences

- Let $\varphi \in \mathrm{GF}$ be satisfiable in normal form:

$$\varphi \;=\; \bigwedge_t \forall \bar{x}\, (\alpha_t(\bar{x}) \to \exists y\; \psi_t(\bar{x}, y))$$

  where each α_t is a guard and each ψ_t is quantifier-free

- Fix an infinite model $\mathcal{A} \models \varphi$. For $k \in \mathbb{N}$, let

$$\boldsymbol{\tau}_k^{\mathcal{A}} \;=\; \{\ \text{(atomic) } k\text{-types realised in } \mathcal{A}\ \}$$

  k-type $=$ complete description of all atomic facts on a k-tuple

- Naïve construction of $\mathcal{B}_n$ on n elements:

```
initialise B_n as the empty structure on {1, ..., n};
foreach tuple b̄ do
    choose a k-type τ ∈ τ_k^A uniformly at random (k = |b̄|);
    make b̄ realise τ;
```

- Problem: overlapping tuples may impose inconsistent facts
- Example: (b_1, b_2, b_3) and (b_1, b_2, b_3') may disagree on $R(b_1, b_2)$

- We repair the naïve construction of $\mathcal{B}_n$:

- We repair the naïve construction of $\mathcal{B}_n$:

initialise $\mathcal{B}_n$ as the empty structure on $\{1, \ldots, n\}$;
foreach *tuple* $\bar{b}$*, in non-decreasing size* **do**
    choose a k-type $\tau \in \boldsymbol{\tau}_k^{\mathcal{A}}$ uniformly at random ($k = |\bar{b}|$);
    if $\{$ $\bar{b}$ *realises* τ $\}$ *is consistent with previously fixed facts* **then**
        make $\bar{b}$ realise τ;

- We repair the naïve construction of $\mathcal{B}_n$:

initialise $\mathcal{B}_n$ as the empty structure on $\{1, \ldots, n\}$;
foreach *tuple* $\bar{b}$, *in non-decreasing size* **do**
    choose a k-type $\tau \in \boldsymbol{\tau}_k^{\mathcal{A}}$ uniformly at random ($k = |\bar{b}|$);
    if $\{\ \bar{b}$ *realises* $\tau\ \}$ *is consistent with previously fixed facts* **then**
        make $\bar{b}$ realise τ;

- $\{\ \bar{b}$ realises $\tau\ \}$ is a local event: dependent only on subtuples of $\bar{b}$

- We repair the naïve construction of $\mathcal{B}_n$:

initialise $\mathcal{B}_n$ as the empty structure on $\{1, \ldots, n\}$;
foreach *tuple* $\bar{b}$*, in non-decreasing size* **do**
    choose a k-type $\tau \in \boldsymbol{\tau}_k^{\mathcal{A}}$ uniformly at random ($k = |\bar{b}|$);
    if $\{$ $\bar{b}$ *realises* τ $\}$ *is consistent with previously fixed facts* **then**
        make $\bar{b}$ realise τ;

- $\{$ $\bar{b}$ realises τ $\}$ is a local event: dependent only on subtuples of $\bar{b}$
- Every guarded tuple in $\mathcal{B}_n$ looks like a tuple from $\mathcal{A}$:

- We repair the naïve construction of $\mathcal{B}_n$:

```
initialise B_n as the empty structure on {1, ..., n};
foreach tuple b̄, in non-decreasing size do
    choose a k-type τ ∈ τ_k^A uniformly at random (k = |b̄|);
    if { b̄ realises τ } is consistent with previously fixed facts then
        make b̄ realise τ;
```

- $\{\ \bar{b}$ realises $\tau\ \}$ is a local event: dependent only on subtuples of $\bar{b}$
- Every guarded tuple in $\mathcal{B}_n$ looks like a tuple from $\mathcal{A}$:

$$\mathcal{B}_n \models \alpha(\bar{b}) \quad \Longrightarrow \quad \bar{b} \text{ isomorphic to some } \bar{a}$$

- We repair the naïve construction of $\mathcal{B}_n$:

```
initialise B_n as the empty structure on {1, ..., n};
foreach tuple b̄, in non-decreasing size do
    choose a k-type τ ∈ τ_k^A uniformly at random (k = |b̄|);
    if { b̄ realises τ } is consistent with previously fixed facts then
        make b̄ realise τ;
```

- $\{\ \bar{b}$ realises $\tau\ \}$ is a local event: dependent only on subtuples of $\bar{b}$
- Every guarded tuple in $\mathcal{B}_n$ looks like a tuple from $\mathcal{A}$:

$$\mathcal{B}_n \models \alpha(\bar{b}) \quad \Longrightarrow \quad \bar{b} \text{ isomorphic to some } \bar{a}$$

- For large n, every local extension that appears in $\mathcal{A}$ is realised in $\mathcal{B}_n$:

- We repair the naïve construction of $\mathcal{B}_n$:

initialise $\mathcal{B}_n$ as the empty structure on $\{1, \ldots, n\}$;
foreach *tuple* $\bar{b}$*, in* *non-decreasing* *size* **do**
    choose a k-type $\tau \in \boldsymbol{\tau}_k^{\mathcal{A}}$ uniformly at random ($k = |\bar{b}|$);
    if $\{\ \bar{b}$ *realises* $\tau\ \}$ *is* *consistent* *with previously fixed facts* **then**
        make $\bar{b}$ realise τ;

- $\{\ \bar{b}$ realises $\tau\ \}$ is a local event: dependent only on subtuples of $\bar{b}$
- Every guarded tuple in $\mathcal{B}_n$ looks like a tuple from $\mathcal{A}$:

$$\mathcal{B}_n \models \alpha(\bar{b}) \quad \Longrightarrow \quad \bar{b} \text{ isomorphic to some } \bar{a}$$

- For large n, every local extension that appears in $\mathcal{A}$ is realised in $\mathcal{B}_n$:

$$\mathcal{A} \models \exists y\ \tau(\bar{a}, y) \quad \Longrightarrow \quad \mathcal{B}_n \models \exists y\ \tau(\bar{b}, y)$$

where $\tau(\bar{x}, y)$ = the tuple $\bar{x}y$ realises the k-type τ

- We repair the naïve construction of $\mathcal{B}_n$:

initialise $\mathcal{B}_n$ as the empty structure on $\{1, \ldots, n\}$;
foreach *tuple* $\bar{b}$*, in non-decreasing size* **do**
    choose a k-type $\tau \in \boldsymbol{\tau}_k^{\mathcal{A}}$ uniformly at random ($k = |\bar{b}|$);
    if $\{\ \bar{b}$ *realises* $\tau\ \}$ *is consistent with previously fixed facts* **then**
        make $\bar{b}$ realise τ;

- $\{\ \bar{b}$ realises $\tau\ \}$ is a local event: dependent only on subtuples of $\bar{b}$
- Every guarded tuple in $\mathcal{B}_n$ looks like a tuple from $\mathcal{A}$:

$$\mathcal{B}_n \models \alpha(\bar{b}) \quad \Longrightarrow \quad \bar{b} \text{ isomorphic to some } \bar{a}$$

- For large n, every local extension that appears in $\mathcal{A}$ is realised in $\mathcal{B}_n$:

$$\mathcal{A} \models \exists y\ \tau(\bar{a}, y) \quad \Longrightarrow \quad \mathcal{B}_n \models \exists y\ \tau(\bar{b}, y)$$

  where $\tau(\bar{x}, y) =$ the tuple $\bar{x}y$ realises the k-type τ

- Failure probability $\mathbb{P}[\mathcal{B}_n \not\models \exists y\ \tau(\bar{b}, y)]$ decays exponentially with n

- We repair the naïve construction of $\mathcal{B}_n$:

initialise $\mathcal{B}_n$ as the empty structure on $\{1, \ldots, n\}$;
foreach *tuple* $\bar{b}$, *in non-decreasing size* **do**
    choose a k-type $\tau \in \boldsymbol{\tau}_k^{\mathcal{A}}$ uniformly at random ($k = |\bar{b}|$);
    if $\{\ \bar{b}$ *realises* $\tau\ \}$ *is consistent with previously fixed facts* **then**
        make $\bar{b}$ realise τ;

- $\{\ \bar{b}$ realises $\tau\ \}$ is a local event: dependent only on subtuples of $\bar{b}$
- Every guarded tuple in $\mathcal{B}_n$ looks like a tuple from $\mathcal{A}$:

$$\mathcal{B}_n \models \alpha(\bar{b}) \quad \Longrightarrow \quad \bar{b} \text{ isomorphic to some } \bar{a}$$

- For large n, every local extension that appears in $\mathcal{A}$ is realised in $\mathcal{B}_n$:

$$\mathcal{A} \models \exists y\ \tau(\bar{a}, y) \quad \Longrightarrow \quad \mathcal{B}_n \models \exists y\ \tau(\bar{b}, y)$$

  where $\tau(\bar{x}, y)$ = the tuple $\bar{x}y$ realises the k-type τ
- Failure probability $\mathbb{P}[\mathcal{B}_n \not\models \exists y\ \tau(\bar{b}, y)]$ decays exponentially with n
- Take the union bound: there are only poly(n) choices for $\bar{b}$

- We repair the naïve construction of $\mathcal{B}_n$:

initialise $\mathcal{B}_n$ as the empty structure on $\{1, \ldots, n\}$;
foreach *tuple* $\bar{b}$*, in non-decreasing size* **do**
    choose a k-type $\tau \in \boldsymbol{\tau}_k^{\mathcal{A}}$ uniformly at random ($k = |\bar{b}|$);
    if $\{\ \bar{b}$ *realises* $\tau\ \}$ *is consistent with previously fixed facts* **then**
        make $\bar{b}$ realise τ;

- $\{\ \bar{b}$ realises $\tau\ \}$ is a local event: dependent only on subtuples of $\bar{b}$
- Every guarded tuple in $\mathcal{B}_n$ looks like a tuple from $\mathcal{A}$:

$$\mathcal{B}_n \models \alpha(\bar{b}) \quad \Longrightarrow \quad \bar{b} \text{ isomorphic to some } \bar{a}$$

- For large n, every local extension that appears in $\mathcal{A}$ is realised in $\mathcal{B}_n$:

$$\mathcal{A} \models \exists y\ \tau(\bar{a}, y) \quad \Longrightarrow \quad \mathcal{B}_n \models \exists y\ \tau(\bar{b}, y)$$

  where $\tau(\bar{x}, y) =$ the tuple $\bar{x}y$ realises the k-type τ

- Failure probability $\mathbb{P}[\mathcal{B}_n \not\models \exists y\ \tau(\bar{b}, y)]$ decays exponentially with n
- Take the union bound: there are only poly(n) choices for $\bar{b}$
- Recall the normal form: $\varphi \ = \ \bigwedge_t \forall \bar{x}\ (\alpha_t(\bar{x}) \rightarrow \exists y\ \psi_t(\bar{x}, y))$

Paper: Random Models and Guarded Logic

Paper: Random Models and Guarded Logic

Theorem (Probabilistic Proof)

$\varphi \in \mathrm{GF}$ *satisfiable* $\implies$ φ *has a finite model*

Paper: Random Models and Guarded Logic

Theorem (Probabilistic Proof)

$\varphi \in \mathrm{GF}$ *satisfiable* $\implies$ φ *has a finite model of doubly exponential size*

Paper: Random Models and Guarded Logic

Theorem (Probabilistic Proof)

$\varphi \in \mathrm{GF}$ *satisfiable* $\implies$ φ *has a finite model of doubly exponential size*

Further results in the paper:

Paper: Random Models and Guarded Logic

Theorem (Probabilistic Proof)

$\varphi \in \mathrm{GF}$ *satisfiable* $\implies$ φ *has a finite model of doubly exponential size*

Further results in the paper:

- Precise analysis of the bound: $2^{2^{\mathcal{O}(|\varphi| \log |\varphi|)}}$

Paper: Random Models and Guarded Logic

Theorem (Probabilistic Proof)

$\varphi \in \mathrm{GF}$ *satisfiable* $\implies$ φ *has a finite model of doubly exponential size*

Further results in the paper:

- Precise analysis of the bound: $2^{2^{\mathcal{O}(|\varphi| \log |\varphi|)}}$
- Sentences that enforce models of tightly matching size

Paper: Random Models and Guarded Logic

Theorem (Probabilistic Proof)

$\varphi \in \mathrm{GF}$ *satisfiable* $\implies$ φ *has a finite model of doubly exponential size*

Further results in the paper:

- Precise analysis of the bound: $2^{2^{\mathcal{O}(|\varphi| \log |\varphi|)}}$
- Sentences that enforce models of tightly matching size
- Extension to the Triguarded Fragment $(\mathrm{GF} + \mathrm{FO}^2)$

Paper: Random Models and Guarded Logic

Theorem (Probabilistic Proof)

$\varphi \in \mathrm{GF}$ *satisfiable* $\implies$ φ *has a finite model of doubly exponential size*

Further results in the paper:

- Precise analysis of the bound: $2^{2^{\mathcal{O}(|\varphi| \log |\varphi|)}}$
- Sentences that enforce models of tightly matching size
- Extension to the Triguarded Fragment $(\mathrm{GF} + \mathrm{FO}^2)$
- Derandomisation via hashing

Paper: Random Models and Guarded Logic

Theorem (Probabilistic Proof)

$\varphi \in \mathrm{GF}$ *satisfiable* $\implies$ φ *has a finite model of doubly exponential size*

Further results in the paper:

- Precise analysis of the bound: $2^{2^{\mathcal{O}(|\varphi| \log |\varphi|)}}$
- Sentences that enforce models of tightly matching size
- Extension to the Triguarded Fragment $(\mathrm{GF} + \mathrm{FO}^2)$
- Derandomisation via hashing

Thank you!