Descriptive complexity of real computation and probabilistic team semantics

Jonni Virtema

Leibniz Universität Hannover, Germany

Aalto CS Theory Seminar
January 13th 2021

Main references

- Tractability frontiers in probabilistic team semantics and existential second-order logic over the reals. Submitted, preprint available in arXiv, 2020.
  Joint work with Miika Hannula.
- Descriptive complexity of real computation and probabilistic independence logic.
  Proceedings of LICS 2020.
  Joint work with Miika Hannula, Juha Kontinen, and Jan Van den Bussche.
- Facets of Distribution Identities in Probabilistic Team Semantics.
  Proceedings of JELIA 2019.
  Joint work with Miika Hannula, Åsa Hirvonen, Juha Kontinen, and Vadim Kulikov.
- Probabilistic Team Semantics.
  Proceedings of FoIKS 2018.
  Joint work with Arnaud Durand, Miika Hannula, Juha Kontinen, and Arne Meier.

Descriptive complexity of...

- Offers a machine independent description of complexity classes:
  - Time/Space used by a machine to decide a problem
    $\Rightarrow$ richness of the logical language needed to describe the problem.
- Complexity classes can/could be then separated by separating logics.
- Many characterisations are known:
  - Fagin's Theorem 1973: Existential second-order logic characterises NP.
  - Immerman 1980s: First-order logic characterises AC^0 and constant time CRAM.
  - Immerman & Vardi 1980s: Least fixed point logic LFP characterises P on ordered structures.

    ⋮

Descriptive complexity of...

- Offers a machine independent description of complexity classes:
  - Time/Space used by a machine to decide a problem
    $\Rightarrow$ richness of the logical language needed to describe the problem.
- Complexity classes can/could be then separated by separating logics.
- Many characterisations are known:
  - Fagin's Theorem 1973: Existential second-order logic characterises NP.

    "A graph is three colourable" $=$
    $\exists R \exists B \exists G($"each node is labeled by exactly one colour"
    $\wedge$ "adjacent nodes are always coloured with distinct colours"$)$

  - Immerman 1980s: First-order logic characterises AC^0 and constant time CRAM.
  - Immerman & Vardi 1980s: Least fixed point logic LFP characterises P on ordered structures.

    ⋮

Descriptive complexity of...

- Offers a machine independent description of complexity classes:
  - Time/Space used by a machine to decide a problem
    $\Rightarrow$ richness of the logical language needed to describe the problem.
- Complexity classes can/could be then separated by separating logics.
- Many characterisations are known:
  - Fagin's Theorem 1973: Existential second-order logic characterises NP.
  - Immerman 1980s: First-order logic characterises AC^0 and constant time CRAM.
  - Immerman & Vardi 1980s: Least fixed point logic LFP characterises P on ordered structures.

  ⋮

...complexity of real computation...

- Turing machines read bit-strings and thus recognize subsets of $\{0,1\}^*$.
- Claim: Today and in the future handling of numerical data is essential.
  - Turing machines can only deal with binary representations of numerical data.
    - Large numbers require more space to encode.
    - The cost of doing arithmetic depends on the sizes of encodings.
  - One alternative is to compute with numerical data directly.
    - Large numbers do not require more space to write than small numbers.
    - The cost of doing arithmetic is independent on the size of numbers.
- Motivation:
  - Symbolic computation.
  - Analogue computation (e.g., hardware implementation of neuromorphic computing).
  - Some other reason to abstract away the sizes of numerical datavalues.

...complexity of real computation...

- Turing machines read bit-strings and thus recognize subsets of $\{0,1\}^*$.
- Claim: Today and in the future handling of numerical data is essential.
  - Turing machines can only deal with binary representations of numerical data.
    - Large numbers require more space to encode.
    - The cost of doing arithmetic depends on the sizes of encodings.
  - One alternative is to compute with numerical data directly.
    - Large numbers do not require more space to write than small numbers.
    - The cost of doing arithmetic is independent on the size of numbers.
- Motivation:
  - Symbolic computation.
  - Analogue computation (e.g., hardware implementation of neuromorphic computing).
  - Some other reason to abstract away the sizes of numerical datavalues.

...complexity of real computation...

- Turing machines read bit-strings and thus recognize subsets of $\{0,1\}^*$.
- Claim: Today and in the future handling of numerical data is essential.
  - Turing machines can only deal with binary representations of numerical data.
    - Large numbers require more space to encode.
    - The cost of doing arithmetic depends on the sizes of encodings.
  - One alternative is to compute with numerical data directly.
    - Large numbers do not require more space to write than small numbers.
    - The cost of doing arithmetic is independent on the size of numbers.
- Motivation:
  - Symbolic computation.
  - Analogue computation (e.g., hardware implementation of neuromorphic computing).
  - Some other reason to abstract away the sizes of numerical datavalues.

...complexity of real computation...

- Turing machines read bit-strings and thus recognize subsets of $\{0,1\}^*$.
- Claim: Today and in the future handling of numerical data is essential.
  - Turing machines can only deal with binary representations of numerical data.
    - Large numbers require more space to encode.
    - The cost of doing arithmetic depends on the sizes of encodings.
  - One alternative is to compute with numerical data directly.
    - Large numbers do not require more space to write than small numbers.
    - The cost of doing arithmetic is independent on the size of numbers.
- Motivation:
  - Symbolic computation.
  - Analogue computation (e.g., hardware implementation of neuromorphic computing).
  - Some other reason to abstract away the sizes of numerical datavalues.

Reals vs. rationals vs. integers

- If we have numerical data, we surely want to use some arithemetic.
- Hilbert's 10th problem answered by Yuri Matiyasevich in 1970:
  It is undecidable to decide whether a polynomial equation with integer coefficients have integer solutions.
- Hilbert's 10th problem w.r.t rational solutions is open.
  First-order theory of rational arithmetic is undecidable (Julia Robinson, 1949).
- Hilbert's 10th problem w.r.t real solutions is decidable.
  First-order theory of real arithmetic is decidable (Alfred Tarski, 1951).
- The complexity class $\exists\mathbb{R}$ is the closure of the existential theory of the reals under polynomial-time reductions, and $\mathrm{NP} \leq \exists\mathbb{R} \leq \mathrm{PSPACE}$ (John F. Canny, 1988).

Reals vs. rationals vs. integers

- If we have numerical data, we surely want to use some arithemetic.
- Hilbert's 10th problem answered by Yuri Matiyasevich in 1970:
  It is undecidable to decide whether a polynomial equation with integer coefficients have integer solutions.
- Hilbert's 10th problem w.r.t rational solutions is open.
  First-order theory of rational arithmetic is undecidable (Julia Robinson, 1949).
- Hilbert's 10th problem w.r.t real solutions is decidable.
  First-order theory of real arithmetic is decidable (Alfred Tarski, 1951).
- The complexity class $\exists\mathbb{R}$ is the closure of the existential theory of the reals under polynomial-time reductions, and $\mathrm{NP} \leq \exists\mathbb{R} \leq \mathrm{PSPACE}$ (John F. Canny, 1988).

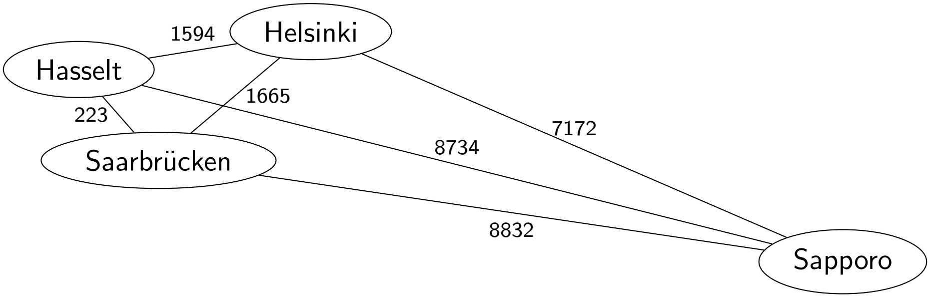
Reals vs. rationals vs. integers

- If we have numerical data, we surely want to use some arithemetic.
- Hilbert's 10th problem answered by Yuri Matiyasevich in 1970:
  It is undecidable to decide whether a polynomial equation with integer coefficients have integer solutions.
- Hilbert's 10th problem w.r.t rational solutions is open.
  First-order theory of rational arithmetic is undecidable (Julia Robinson, 1949).
- Hilbert's 10th problem w.r.t real solutions is decidable.
  First-order theory of real arithmetic is decidable (Alfred Tarski, 1951).
- The complexity class $\exists\mathbb{R}$ is the closure of the existential theory of the reals under polynomial-time reductions, and NP $\leq \exists\mathbb{R} \leq$ PSPACE (John F. Canny, 1988).

$\mathbb{R}$-structures

$\mathbb{R}$-structures [Grädel and Meer, 1995] consist of a finite structure $\mathfrak{A}$ together with an ordered field of reals and a finite set of weight functions from $\mathfrak{A}$ to $\mathbb{R}$

(particular case of metafinite structures [Grädel and Gurevich, 1998])

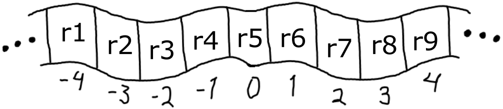


Blum-Shub-Smale machines

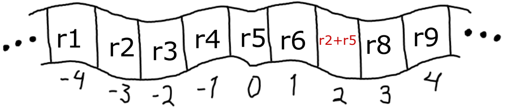
Input: finite string of reals
Output: 0 or 1 (decision problems)

A program is a finite list of instructions:

- Arithmetic instructions:
  $x_i \leftarrow (x_j + x_k)$, $x_i \leftarrow (x_j - x_k)$,
  $x_i \leftarrow (x_j \times x_k)$, $x_i \leftarrow c$.
- Shift left or right.
- Branch on inequality
  if $x_0 \leq 0$ **then** go to α; **else** go to β.



Addition: [2] := [-3]+[0]



Nondeterministic BSS

Nondeterminism is implemented by guessing a certificate:

$\mathcal{L} \in \mathrm{NP}_{\mathbb{R}}$ there exists a BSS machine M s.t.
$x \in \mathcal{L}$ iff $\exists y \in \mathbb{R}^*$ s.t. M accepts (x, y) in polynomial time in $|x|$.

Example $\mathrm{NP}_{\mathbb{R}}$-complete problem: Is there a real root for a polynomial of degree 4?

Descriptive complexity over the reals

Theorem ([Grädel and Meer, 1995])

$\mathrm{ESO}_{\mathbb{R}}[+, \times, \leq, (r)_{r \in \mathbb{R}}] \equiv \mathsf{NP}_{\mathbb{R}}$

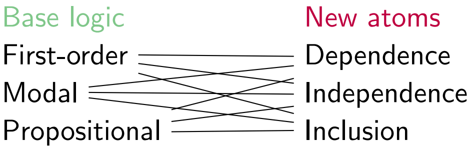
Two-sorted variant of ESO with

1. first-order logic on the finite structure $\mathfrak{A}$
2. existential quantification of functions from $\mathfrak{A}$ to reals
3. constants r for each real
4. complex numerical terms by $\{+, \times\}$
5. (negated) inequality $\leq$ between numerical terms

Descriptive complexity of real computation and probabilistic team semantics

Logics of dependence and independence

Recipe for modern logics for dependence and independence:



Historical predecessors: First-order logic + richer quantification of variables

- Partially ordered quantifiers [Henkin, 1961]
- Independence-friendly logic [Hintikka and Sandu, 1989]

Qualitative vs. quantitative dependence

Modern logics of dependence can reason both about qualitative (relational) and quantitative (probabilistic) dependencies?

Qualitative:

Functional dependency $X \to Y$

Multivalued dependency $X \twoheadrightarrow Y$

Inclusion dependency $X \subseteq Y$

Quantitative:

Marginal independence $X \perp\!\!\!\perp Y$

Conditional independence $X \perp\!\!\!\perp Y \mid Z$

Identical distribution of X and Y

Team semantics

Compositional semantics for complex dependence statements by team semantics [Hodges, 1997]

Team = set of objects (assignments, possible worlds, Boolean assignments)

Employee	Department	Salary
Alice	Math	50k
Bob	CS	40k
Carol	Physics	60k
David	Math	80k

New atoms = basic dependence statements about teams
(e.g, `Employee` determines `Salary`)

$\{\forall, \exists, \Box, \Diamond, \wedge, \vee\}$ for complex dependence statements

Probabilistic team semantics

Basic concepts:

- Probabilistic team = probability distribution on a finite team (FoIKS 2018)
- Quantitative atoms (e.g., conditional independence, identical marginal distributions)
- $\{\forall, \exists, \wedge, \vee\}$ for complex probability statements

Reasoning about dependencies

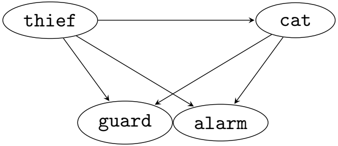
Dependence and independence pivotal notions in many areas (databases, social choice, quantum foundations, ...)

Team logics can be used to express and formally prove results in these fields

- Arrow's theorem [Pacuit and Yang, 2016]
- Bell's theorem [Hyttinen et al., 2015]
- Implication problems for data dependencies [Hannula and Kontinen, 2016]

No "general" proof system: validity problem usually non-arithmetical.

Example



guard		
thief,cat	T	F
TT	0.8	0.2
TF	0.7	0.3
FT	0	1
FF	0	1

thief	
T	F
0.1	0.9

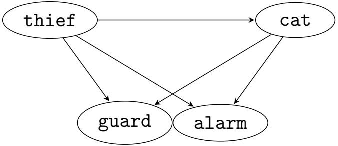
cat		
thief	T	F
T	0.1	0.9
F	0.6	0.4

alarm		
thief,cat	T	F
TT	0.9	0.1
TF	0.8	0.2
FT	0.1	0.9
FF	0	1

From the Bayesian network above we obtain that the joint probability distribution for t, c, g, a can be factorized as

$$P(t, c, g, a) = P(t) \cdot P(c \mid t) \cdot P(g \mid t, c) \cdot P(a \mid t, c)$$

Example



guard

thief,cat	T	F
TT	0.8	0.2
TF	0.7	0.3
FT	0	1
FF	0	1

thief

T	F
0.1	0.9

cat

thief	T	F
T	0.1	0.9
F	0.6	0.4

alarm

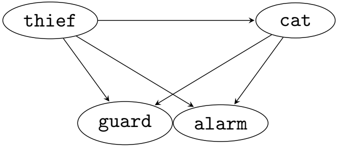
thief,cat	T	F
TT	0.9	0.1
TF	0.8	0.2
FT	0.1	0.9
FF	0	1

If additionally we have

$$\phi := t = F \to g = F$$

(i.e., guard never raises alert in absence of thief), the two bottom rows of the conditional probability table for guard become superfluous.

Example



guard

thief,cat	T	F
TT	0.8	0.2
TF	0.7	0.3
FT	0	1
FF	0	1

thief

T	F
0.1	0.9

cat

thief	T	F
T	0.1	0.9
F	0.6	0.4

alarm

thief,cat	T	F
TT	0.8	0.2
TF	0.7	0.3
FT	0	1
FF	0	1

Given

$$\phi := tca \approx tcg$$

(i.e., conditioned on `thief` and `cat`, `alarm` and `guard` are identically distributed), then the conditional probability tables for `alarm` and `guard` are identical and one of them can be removed.

Probabilistic inclusion logic $\mathrm{FO}(\approx)$ and independence logic $\mathrm{FO}(\perp\!\!\!\perp_c)$

Syntax: FO (negation normal form) + $\vec{x} \approx \vec{y}$ (only positively)
FO (negation normal form) + $\vec{y} \perp\!\!\!\perp_{\vec{x}} \vec{z}$ (only positively)

Semantics: Defined in terms of a finite structure $\mathfrak{A}$ and a probabilistic team $\mathbb{X}$

(1) Team $=$ a set of variable assignments with a shared domain

(2) Probabilistic team $=$ a pair $\mathbb{X} = (X, p)$, where X is a finite team and $p : X \to [0, 1]$ is a probability distribution

Semantics of (probabilistic) dependencies

Let $\mathbb{X} = (X, p)$ be a probabilistic team and $\vec{x}, \vec{a}$ be tuples of variables and values.

$$|\mathbb{X}|_{\vec{x}=\vec{a}} := \sum_{\substack{s \in X \\ s(\vec{x})=\vec{a}}} p(s)$$

The semantics of marginal identity atoms (identical distribution) $\vec{x} \approx \vec{y}$:

$$\mathfrak{A} \models_{\mathbb{X}} \vec{x} \approx \vec{y} \text{ iff } |\mathbb{X}|_{\vec{x}=\vec{a}} = |\mathbb{X}|_{\vec{y}=\vec{a}}, \text{ for each } \vec{a} \in A^k$$

Semantics of (probabilistic) dependencies

Let $\mathbb{X} = (X, p)$ be a probabilistic team and $\vec{x}, \vec{a}$ be tuples of variables and values.

$$|\mathbb{X}|_{\vec{x}=\vec{a}} := \sum_{\substack{s \in X \\ s(\vec{x})=\vec{a}}} p(s)$$

The semantics of probabilistic conditional independence atoms $\vec{y} \perp\!\!\!\perp_{\vec{x}} \vec{z}$:

$\mathfrak{A} \models_{\mathbb{X}} \vec{y} \perp\!\!\!\perp_{\vec{x}} \vec{z}$ iff, for all assignments s for $\vec{x}, \vec{y}, \vec{z}$

$$|\mathbb{X}|_{\vec{x}\vec{y}=s(\vec{x}\vec{y})} \cdot |\mathbb{X}|_{\vec{x}\vec{z}=s(\vec{x}\vec{z})} = |\mathbb{X}|_{\vec{x}\vec{y}\vec{z}=s(\vec{x}\vec{y}\vec{z})} \cdot |\mathbb{X}|_{\vec{x}=s(\vec{x})}$$

Semantics of first-order part I

Definition (FoIKS 2018)

Let $\mathfrak{A}$ be a finite structure and $\mathbb{X} = (X, p)$ a probabilistic team.

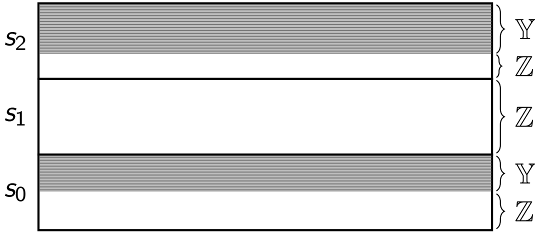
$$\begin{aligned}
\mathfrak{A} \models_{\mathbb{X}} \ell \quad &\Leftrightarrow \quad \mathfrak{A} \models_s \ell \text{ for all } s \in X \text{ such that } p(s) > 0 \\
&\qquad \text{(when } \ell \text{ is a first-order literal)} \\
\mathfrak{A} \models_{\mathbb{X}} (\psi \wedge \theta) \quad &\Leftrightarrow \quad \mathfrak{A} \models_{\mathbb{X}} \psi \text{ and } \mathfrak{A} \models_{\mathbb{X}} \theta
\end{aligned}$$

Semantics of first-order part II

Disjunction via convex combinations:

$$\mathfrak{A} \models_{\mathbb{X}} (\psi \vee \theta) \quad \Leftrightarrow \quad \mathfrak{A} \models_{\mathbb{Y}} \psi \text{ and } \mathfrak{A} \models_{\mathbb{Z}} \theta,$$

where $\mathbb{X} = \alpha \cdot \mathbb{Y} + (1-\alpha) \cdot \mathbb{Z}$, for some $\alpha \in [0,1]$.



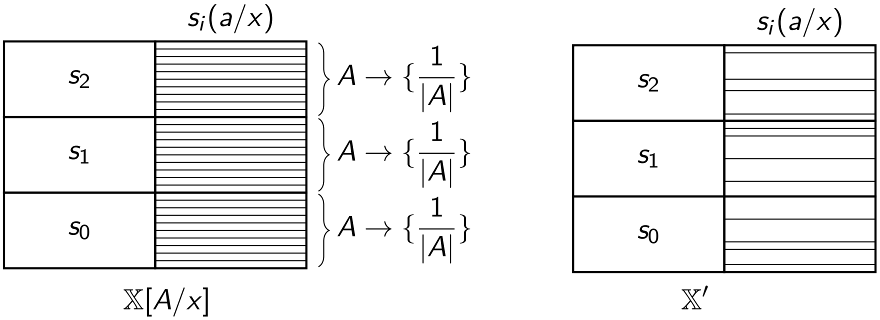
NB. The empty set is considered as a probabilistic team.

Semantics of first-order part III

Quantification introduces a new column:

$$\mathfrak{A} \models_{\mathbb{X}} \forall x \psi \quad \Leftrightarrow \quad \mathfrak{A} \models_{\mathbb{X}[A/x]} \psi$$

$$\mathfrak{A} \models_{\mathbb{X}} \exists x \psi \quad \Leftrightarrow \quad \mathfrak{A} \models_{\mathbb{X}'} \psi, \text{ for some } \mathbb{X}' \text{ such that } \mathbb{X}' \upharpoonright \mathrm{Dom}(\mathbb{X}) = \mathbb{X}$$



Descriptive complexity and team semantics

Descriptive complexity in team logics:

- Independence logic $\mathrm{FO}(\perp_{\mathrm{c}})$ equi-expressive to ESO $\implies$ captures NP.
- Inclusion logic $\mathrm{FO}(\subseteq)$ equi-expressive to positive greatest fixed point-logic $\implies$ captures P on ordered structures [Galliani and Hella, 2013].

Descriptive complexity and team semantics

Descriptive complexity in team logics:

- Independence logic $\mathrm{FO}(\perp_{\mathrm{c}})$ equi-expressive to $\mathrm{ESO} \implies$ captures NP.
- Inclusion logic $\mathrm{FO}(\subseteq)$ equi-expressive to positive greatest fixed point-logic $\implies$ captures P on ordered structures [Galliani and Hella, 2013].

Descriptive complexity in probabilistic team logics:

- Sentences $\sim$ finite structures $\sim$ strings of Booleans
- Formulae $\sim$ probabilistic teams $\sim$ strings of reals
  Here we venture to the realm of BSS-computing.

Expressivity over sentences

$$
\begin{array}{ccccc}
\mathrm{FO}(\subseteq) & & \mathrm{FO}(\subseteq, \mathrm{dep}(\cdots)) & & \\
\|_{\mathrm{ord}} & & \| & & \\
\mathsf{P} & \subseteq & \mathsf{NP} & \subseteq & \mathsf{PSPACE} \\
 & & \| & & \\
 & & \mathrm{FO}(\perp_{\mathrm{c}}) & &
\end{array}
$$

Table: Team semantics

$$
\begin{array}{ccccccccc}
\mathrm{FO}(\approx) & & \mathrm{FO}(\approx, \mathrm{dep}(\cdots)) & & \mathrm{FO}(\perp\!\!\!\perp_{\mathrm{c}}) & & & & \\
\|^{*}_{\mathrm{ord}} & & \|_{*} & & \|_{\dagger} & & & & \\
\mathsf{P} & \subseteq & \mathsf{NP} & \subseteq & \exists[0,1]^{\leq} & \subseteq & \exists\mathbb{R} & \subseteq & \mathsf{PSPACE}
\end{array}
$$

Table: Probabilistic team semantics. † LICS 2020, * arXiv 2020

Expressivity over formulae

$$\mathrm{FO}(\subseteq) \quad \subsetneq \quad \mathrm{FO}(\subseteq, \mathrm{dep}(\cdots)) \equiv \mathrm{FO}(\perp_c)$$

Table: Team semantics

$$\mathrm{FO}(\approx) \quad \subsetneq^{*} \quad \mathrm{FO}(\approx, \mathrm{dep}(\cdots)) \quad \subsetneq^{\dagger} \quad \mathrm{FO}(\perp\!\!\!\perp_c)$$

Table: Probabilistic team semantics. * JELIA 2019, † arXiv 2020

Descriptive complexity of formulae:

- $\mathrm{FO}(\perp\!\!\!\perp_c) \equiv \text{S-NP}^0_{[0,1]}$ (LICS 2020)
- $\mathrm{FO}(\approx, \mathrm{dep}(\cdots)) \subseteq$ additive $\text{S-NP}^0_{[0,1]}$ (arXiv 2020 + conjecture)
- $\mathrm{FO}(\approx) \subseteq$ additive $\text{S-P}^0_{[0,1]}$ (arXiv 2020 + conjecture)

Expressivity over formulae

$$\mathrm{FO}(\subseteq) \quad \subsetneq \quad \mathrm{FO}(\subseteq, \mathrm{dep}(\cdots)) \equiv \mathrm{FO}(\perp_c)$$

Table: Team semantics

$$\mathrm{FO}(\approx) \quad \subsetneq^{*} \quad \mathrm{FO}(\approx, \mathrm{dep}(\cdots)) \quad \subsetneq^{\dagger} \quad \mathrm{FO}(\perp\!\!\!\perp_c)$$

Table: Probabilistic team semantics. * JELIA 2019, † arXiv 2020

Descriptive complexity of formulae:

- $\mathrm{FO}(\perp\!\!\!\perp_c) \equiv \text{S-NP}^0_{[0,1]}$ (LICS 2020)
- $\mathrm{FO}(\approx, \mathrm{dep}(\cdots)) \subseteq$ additive $\text{S-NP}^0_{[0,1]}$ (arXiv 2020 + conjecture)
- $\mathrm{FO}(\approx) \subseteq$ additive $\text{S-P}^0_{[0,1]}$ (arXiv 2020 + conjecture)

Existential second-order logics on $\mathbb{R}$-structures that capture probabilistic team logics.

BSS machines and logics on $\mathbb{R}$-structures cont.

Theorem ([Grädel and Meer, 1995])

$\mathrm{ESO}_{\mathbb{R}}[+, \times, \leq, (r)_{r\in\mathbb{R}}] \equiv \mathsf{NP}_{\mathbb{R}}$

Two-sorted variant of ESO with

1. first-order logic on the finite structure $\mathfrak{A}$
2. existential quantification of functions from $\mathfrak{A}$ to reals
3. constants r for each real
4. complex numerical terms by $\{+, \times\}$
5. inequality $\leq$ between numerical terms

Too strong for $\mathrm{FO}(\perp\!\!\!\perp_c)$**:** 1) Lacks negation, 2) Quantification over $[0, 1]$

S-BSS model of computation

Blum-Shub-Smale machines (1989)

Input: finite string of reals
Output: 0 or 1 (decision problems)

A program is a finite list of instructions:

- Arithmetic instructions:
  $x_i \leftarrow (x_j + x_k)$, $x_i \leftarrow (x_j - x_k)$,
  $x_i \leftarrow (x_j \times x_k)$, $x_i \leftarrow c$.
- Shift left or right.
- Branch on inequality
  if $x_0 \leq 0$ **then** go to α; **else** go to β.

Separate branching BSS-machines

Input: finite string of reals
Output: 0 or 1 (decision problems)

Instead of branch on inequality:

- Separate branch on inequality
  ($\epsilon^- < \epsilon^+$ are real numbers):
  if $x_0 \leq \epsilon^-$ **then** go to α;
  else if $x_0 \geq \epsilon^+$ **then** go to β;
  else reject.

Nondeterministic S-BSS computations

Nondeterminism is implemented by guessing a certificate from $[0,1]$:

$\mathcal{L} \in \text{S-NP}_{[0,1]}$ there exists an S-BSS machine M s.t.
$x \in \mathcal{L}$ iff $\exists y \in [0,1]^*$ s.t. M accepts (x, y) in polynomial time in $|x|$

$\mathcal{L} \in \text{NP}_{\mathbb{R}}$ there exists a BSS machine M s.t.
$x \in \mathcal{L}$ iff $\exists y \in \mathbb{R}^*$ s.t. M accepts (x, y) in polynomial time in $|x|$

Expressive power of $\mathrm{FO}(\perp\!\!\!\perp_c)$

Descriptive complexity of $\mathrm{FO}(\perp\!\!\!\perp_c)$ in real computation:

Theorem (LICS 2020)

$\mathrm{FO}(\perp\!\!\!\perp_c) \equiv \mathrm{L\text{-}ESO}_{[0,1]}[+, \times, \leq] \equiv \text{S-NP}^{0}_{[0,1]}$

- "Loose fragment": no negated atoms $\neg i \leq j$ between two numerical terms
- Existential second-order quantification over functions from $\mathrm{Dom}(\mathfrak{A})$ to $[0, 1]$
- Superscript 0: only machine constants 0 and 1 allowed

NB. The result holds for formulae of $\mathrm{FO}(\perp\!\!\!\perp_c)$

Expressive powers of $\mathrm{FO}(\approx)$ and $\mathrm{FO}(\approx, \mathrm{dep}(\cdots))$

Theorem (arXiv 2020)

$\mathrm{FO}(\approx, \mathrm{dep}(\cdots)) \equiv \mathrm{L\text{-}ESO}_{[0,1]}[+, \leq, 0, 1]$

$\mathrm{FO}(\approx) \equiv$ *almost conjunctive* $\mathrm{L\text{-}}(\ddot{\exists}^{*}\forall^{*})_{\mathrm{d}[0,1]}[\mathrm{SUM}, \leq, 0, 1]$

Separation of BSS and S-BSS computation

Theorem ([Blum et al., 1989])

Every language decidable by a (deterministic) BSS machine is a countable disjoint union of semi-algebraic sets.

Theorem (LICS 2020)

Every language decidable by

- *a deterministic S-BSS machine, or*
- *a time bounded* $[0,1]$*-nondeterministic S-BSS machine*

is a countable disjoint union of closed sets in $\mathbb{R}^n$*.*

Separation of BSS and S-BSS computation

Theorem (LICS 2020)

Every language decidable by

- *a deterministic S-BSS machine, or*
- *a time bounded* $[0,1]$*-nondeterministic S-BSS machine*

is a countable disjoint union of closed sets in $\mathbb{R}^n$*.*

Proof.

- The set of strings $s \in \mathbb{R}^n$ accepted by an S-BSS machine M in time (at most) t can be described by an $\mathrm{L\text{-}EFO}_{[0,1]}$ formula in $(\mathbb{R}, +, \times, \leq, 0, 1)$.
- Every n-ary relation defined by some $\mathrm{L\text{-}EFO}_{[0,1]}$ formula is closed in $\mathbb{R}^n$.

□

Separation of BSS and S-BSS computation

Theorem (LICS 2020)

Every language decidable by

- *a deterministic S-BSS machine, or*
- *a time bounded* $[0,1]$*-nondeterministic S-BSS machine*

is a countable disjoint union of closed sets in $\mathbb{R}^n$*.*

Theorem (LICS 2020)

$\text{S-NP}_{[0,1]} < \text{NP}_{\mathbb{R}}$

Main result: $\mathrm{FO}(\perp\!\!\!\perp_{\mathrm{c}})$ and real computation cont.

This separation holds also wrt. machines with constants $0, 1$

Descriptive complexity of $\mathrm{FO}(\perp\!\!\!\perp_{\mathrm{c}})$ thus strictly below $\mathsf{NP}^0_{\mathbb{R}}$:

Corollary

$\mathrm{FO}(\perp\!\!\!\perp_{\mathrm{c}}) \equiv \mathrm{S}\text{-}\mathsf{NP}^0_{[0,1]} < \mathsf{NP}^0_{\mathbb{R}}$

Scope of corollary: formulae of $\mathrm{FO}(\perp\!\!\!\perp_{\mathrm{c}})$

What about sentences of $\mathrm{FO}(\perp\!\!\!\perp_{\mathrm{c}})$?

Existential theory of the reals

- The existential theory of the reals consists of all true sentences of the form

$$\exists x_1, \ldots \exists x_n \psi(x_1, \ldots x_n)$$

  where ψ is a quantifier-free formula of the real arithmetic
- Gives rise to the Boolean complexity class $\exists\mathbb{R}$:
  the closure of the existential theory of the reals under polynomial-time reductions
- NP $\leq \exists\mathbb{R} \leq$ PSPACE
- Many natural geometric and algebraic problems are complete for $\exists\mathbb{R}$, such as the art gallery problem or recognition of unit distance graphs

Existential theory of the reals and BSS machines

Theorem ([Bürgisser and Cucker, 2006, Grädel and Meer, 1995, Schaefer and Stefankovic, 2017])

$\exists\mathbb{R} \equiv \mathrm{BP}(\mathsf{NP}^0_{\mathbb{R}}) \equiv \mathrm{ESO}_{\mathbb{R}}[+, \times, \leq]$

$\mathsf{NP}_{\mathbb{R}}$ restricted to Boolean inputs and with machine constants $0, 1$

Too strong for sentences of $\mathrm{FO}(\perp\!\!\!\perp_{c})$?

Main result 2 – $\mathrm{FO}(\perp\!\!\!\perp_c)$ and Boolean computation

Define $\exists[0,1]^{\leq}$ to be the fragment of $\exists\mathbb{R}$ obtained by closing the true sentences of the existential theory of the reals of the form

$$\exists x_1 \dots \exists x_n \Big(\bigwedge_{1 \leq i \leq n} 0 \leq x_i \wedge x_i \leq 1 \wedge \psi \Big),$$

where ψ does not contain $\neg$ nor $<$, by polynomial-time reductions.

(Cf. $\mathrm{L\text{-}ESO}_{[0,1]}[+, \times \leq]$ vs. $\mathrm{ESO}_{\mathbb{R}}[+, \times \leq]$)

Theorem

Over finite structures, $\mathrm{FO}(\perp\!\!\!\perp) \equiv \exists[0,1]^{\leq}$.

Open question: Does $\exists[0,1]^{\leq}$ coincide with NP or $\exists\mathbb{R}$?

Main result 2 – $\mathrm{FO}(\perp\!\!\!\perp_c)$ and Boolean computation

Define $\exists[0,1]^{\leq}$ to be the fragment of $\exists\mathbb{R}$ obtained by closing the true sentences of the existential theory of the reals of the form

$$\exists x_1 \ldots \exists x_n \Big(\bigwedge_{1 \leq i \leq n} 0 \leq x_i \wedge x_i \leq 1 \wedge \psi \Big),$$

where ψ does not contain $\neg$ nor $<$, by polynomial-time reductions.

(Cf. $\mathrm{L\text{-}ESO}_{[0,1]}[+, \times \leq]$ vs. $\mathrm{ESO}_{\mathbb{R}}[+, \times \leq]$)

Theorem

Over finite structures, $\mathrm{FO}(\perp\!\!\!\perp) \equiv \exists[0,1]^{\leq}$.

Open question: Does $\exists[0,1]^{\leq}$ coincide with NP or $\exists\mathbb{R}$?

Probabilistic independence and existential theory of the reals

- The existential theory of the reals consists of all true sentences of the form

$$\exists x_1, \dots \exists x_n \psi(x_1, \dots x_n)$$

  where ψ is a quantifier-free formula of the real arithmetic
- Gives rise to the Boolean complexity class $\exists\mathbb{R}$:
  the closure of the existential theory of the reals under polynomial-time reductions
- $\exists[0,1]^{\leq}$ defined as $\exists\mathbb{R}$ but in terms of sentences of the form

$$\exists x_1 \dots \exists x_n \Big(\bigwedge_{1 \leq i \leq n} 0 \leq x_i \wedge x_i \leq 1 \wedge \psi \Big),$$

  where ψ does not contain $\neg$ nor $<$.

Theorem (LiCS 2020)

Over finite structures, $\mathrm{FO}(\perp\!\!\!\perp_{\mathrm{c}}) \equiv \exists[0,1]^{\leq}$.

Probabilistic independence and existential theory of the reals

- The existential theory of the reals consists of all true sentences of the form

$$\exists x_1, \dots \exists x_n \psi(x_1, \dots x_n)$$

  where ψ is a quantifier-free formula of the real arithmetic
- Gives rise to the Boolean complexity class $\exists\mathbb{R}$:
  the closure of the existential theory of the reals under polynomial-time reductions
- $\exists[0,1]^{\leq}$ defined as $\exists\mathbb{R}$ but in terms of sentences of the form

$$\exists x_1 \dots \exists x_n \Big(\bigwedge_{1 \leq i \leq n} 0 \leq x_i \wedge x_i \leq 1 \wedge \psi \Big),$$

  where ψ does not contain $\neg$ nor $<$.

Theorem (LiCS 2020)

Over finite structures, $\mathrm{FO}(\perp\!\!\!\perp_{\mathrm{c}}) \equiv \exists[0,1]^{\leq}$.

Probabilistic inclusion logic over sentences

Lemma

Let $\phi \in \mathrm{FO}(\approx)$ be a sentence. There is a polynomial-time reduction from finite structures $\mathfrak{A}$ to systems of linear inequations $\mathcal{S}$ such that $\mathfrak{A} \models \phi$ if and only if $\mathcal{S}$ has a solution.

Proof.

Sketch. Add a variable $x_{s,\psi}$, for any partial assignment s and any subformula ψ of ϕ. Initialize $\mathcal{S}$ with $x_{\emptyset,\phi} = 1$, and $x_{\psi,s} \geq 0$ for all s and ψ. For each ψ add a set of equations to describe its corresponding team operation. E.g., for disjunction weights of assignments are split to two:

- If ψ is $\theta \vee \theta'$, add $x_{s,\theta} + x_{s,\theta'} = x_{s,\psi}$ for all s.

□

Deciding whether a system of linear inequalities has solutions is in polynomial time

Theorem

Let $\phi \in \mathrm{FO}(\approx)$ be a sentence. The problem of determining whether $\mathfrak{A} \models \phi$ for a given finite structure $\mathfrak{A}$ is in P*.*

From inclusion to probabilistic inclusion logic

Theorem

Every sentence of $\mathrm{FO}(\subseteq)$ *is equivalent to a sentence of* $\mathrm{FO}(\approx)$.

Proof.

Inclusion atoms definable in terms of *equiextension* atoms $\vec{x}_1 \bowtie \vec{x}_2 := \vec{x}_1 \subseteq \vec{x}_2 \wedge \vec{x}_2 \subseteq \vec{x}_1$ [Galliani, 2012]. However, $\vec{x}_1 \approx \vec{x}_2 \not\equiv \vec{x}_1 \bowtie \vec{x}_2$ as equiextension may hold even if the weights are not in balance.

Proof idea. First balance all positive weights, then apply $\approx$:

$$\forall c \forall \vec{u} \exists v_1 v_2 \forall z'_1 \dots \forall z'_k \exists z_1 \dots \exists z_k \Big(\bigwedge_{i=1,2} \vec{x}_i = \vec{u} \leftrightarrow v_i = c \wedge \bigwedge_{i=1}^{k} z'_i = c \rightarrow z_i = c \wedge (\neg \vec{z} = \vec{c} \vee \vec{u}v_1 \approx \vec{u}v_2) \Big), \tag{1}$$

where k is the number of "splits" (from quantification, disjunction) in the underlying sentence.

Probabilistic inclusion logic over sentences cont.

Theorem

$\mathrm{FO}(\approx)$ *corresponds to* P *over finite ordered structures.*

Proof.

1. Over finite structures: $\mathrm{FO}(\subseteq) \subseteq \mathrm{FO}(\approx) \subseteq$ P
2. Over finite ordered structures: P $\equiv \mathrm{FO}(\subseteq)$ [Galliani and Hella, 2013]

□

Future work:

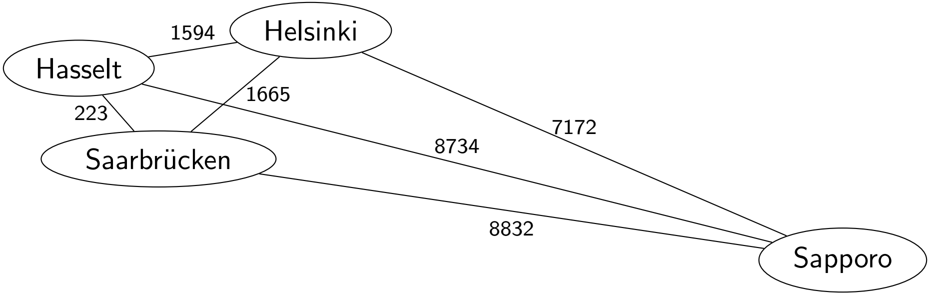
1. $\mathrm{FO}(\subseteq)$ strictly subsumed by $\mathrm{FO}(\approx)$ (over arbitrary finite structures)?
2. Relationship between $\mathrm{FO}(\approx)$ and fixed-point logic/inclusion logic with counting? Cf. [Grädel and Hegselmann, 2016]

Probabilistic inclusion/dependence logic over sentences

Results via $\mathbb{R}$-structures and Blum-Shub-Smale machines

$\mathbb{R}$-structures [Grädel and Meer, 1995] consist of a finite structure $\mathfrak{A}$ together with an ordered field of reals and a finite set of weight functions from $\mathfrak{A}$ to $\mathbb{R}$

(particular case of metafinite structures [Grädel and Gurevich, 1998])

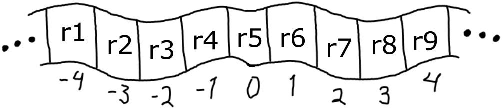


Blum-Shub-Smale machines

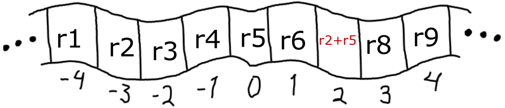
Input: finite string of reals
Output: 0 or 1 (decision problems)

A program is a finite list of instructions:

- Arithmetic instructions:
  $x_i \leftarrow (x_j + x_k)$, $x_i \leftarrow (x_j - x_k)$,
  $x_i \leftarrow (x_j \times x_k)$, $x_i \leftarrow c$.
- Shift left or right.
- Branch on inequality
  if $x_0 \leq 0$ **then** go to α; **else** go to β.



Addition: [2] := [-3]+[0]



Descriptive complexity over the reals

Theorem ([Grädel and Meer, 1995])

$\mathrm{ESO}_{\mathbb{R}}[+, \times, \leq, (r)_{r \in \mathbb{R}}] \equiv \mathsf{NP}_{\mathbb{R}}$

Two-sorted variant of ESO with

1. first-order logic on the finite structure $\mathfrak{A}$
2. existential quantification of functions from $\mathfrak{A}$ to reals
3. constants r for each real
4. complex numerical terms by $\{+, \times\}$
5. (negated) inequality $\leq$ between numerical terms

Too strong for $\mathrm{FO}(\approx, \mathrm{dep}(\cdots))$**:** 1) Lacks $\neg$, $\times$, and real constants 2) Quantification over $[0, 1]$

Descriptive complexity over the reals

Theorem ([Grädel and Meer, 1995])

$\mathrm{ESO}_{\mathbb{R}}[+, \times, \leq, (r)_{r \in \mathbb{R}}] \equiv \mathsf{NP}_{\mathbb{R}}$

Two-sorted variant of ESO with

1. first-order logic on the finite structure $\mathfrak{A}$
2. existential quantification of functions from $\mathfrak{A}$ to reals
3. constants r for each real
4. complex numerical terms by $\{+, \times\}$
5. (negated) inequality $\leq$ between numerical terms

Too strong for $\mathrm{FO}(\approx, \mathrm{dep}(\cdots))$: 1) Lacks $\neg$, $\times$, and real constants 2) Quantification over $[0, 1]$

Probabilistic inclusion/dependence vs. additive ESO over the reals

We show (adapting techniques from [FoIKS 2018]):

Theorem

$\mathrm{FO}(\approx, \mathrm{dep}(\cdots)) \equiv \mathrm{L}\text{-}\mathrm{ESO}_{[0,1]}[+, \leq, 0, 1]$

- "Loose fragment": no negated atoms $\neg i \leq j$ between two numerical terms
- Existential second-order quantification over functions from $\mathrm{Dom}(\mathfrak{A})$ to $[0, 1]$
- Only constants $0, 1$ allowed

NB. The result holds for formulae of $\mathrm{FO}(\approx, \mathrm{dep}(\cdots))$

Probabilistic inclusion/dependence logic over sentences cont.

Theorem

Over finite structures, $\mathrm{FO}(\approx, \mathrm{dep}(\cdots)) \equiv \mathsf{NP}$.

Proof.

$\boxed{\supseteq}$ Over finite structures: $\mathsf{NP} \subseteq \mathrm{FO}(\mathrm{dep}(\cdots)) \subseteq \mathrm{FO}(\approx, \mathrm{dep}(\cdots))$

$\boxed{\subseteq}$ It is easy to show that over formulae:

$$\mathrm{FO}(\approx, \mathrm{dep}(\cdots)) \subseteq \mathrm{ESO}_{\mathbb{R}}[\leq, +, 0, 1] \subseteq \mathsf{NP}^{0}_{\mathrm{add}}.$$

$\mathsf{NP}^{0}_{\mathrm{add}}$ allows guessing a string of reals and then verifying in polynomial time in the **additive** Blum-Shub-Smale model of computation (with machine constants $0, 1$).

It suffices to show that $\mathsf{NP}^{0}_{\mathrm{add}}$ collapses to NP over Boolean inputs. □

Collapse of additive NP over the reals

Theorem

Over Boolean inputs, $\mathrm{NP}^0_{\mathrm{add}} = \mathrm{NP}$

Proof.

Sketch. $\boxed{\supseteq}$ trivial. $\boxed{\subseteq}$ Suppose $L \subseteq \{0,1\}^* \cap \mathrm{NP}^0_{\mathrm{add}}$ is decided non-deterministically by a BSS machine M whose running is bounded by some polynomial p. Let $x \in \{0,1\}^n$ be an input. First, guess the outcome of each comparison of the BSS computation; the outcome is a Boolean string z of length $p(n)$. During a computation the value of each coordinate x_i is a linear function on the constants 0 and 1, the input x, and the real guess y of length $p(n)$. Thus it is possible to construct in polynomial time a system:

$$\sum_{j=1}^{p(n)} a_{ij} y_j \leq 0 \quad (1 \leq i \leq m), \quad \sum_{j=1}^{p(n)} b_{ij} y_j < 0 \quad (1 \leq i \leq l), \quad a_{ij}, b_{ij} \in \mathbb{Z} \tag{2}$$

such that y is a (real-valued) solution iff M accepts (x, y) wrt. z. $\square$

Formulae vs. sentences

This paper: logical, computational, axiomatic properties of $\mathrm{FO}(\approx)$ and $\mathrm{FO}(\approx, \mathrm{dep}(\cdots))$

Two levels of analysis:

- **Sentences $\sim$ finite structures $\sim$ strings of Booleans**
- Formulae $\sim$ probabilistic teams $\sim$ strings of reals

Formulae vs. sentences

This paper: logical, computational, axiomatic properties of $\mathrm{FO}(\approx)$ and $\mathrm{FO}(\approx, \mathrm{dep}(\cdots))$

Two levels of analysis:

- Sentences $\sim$ finite structures $\sim$ strings of Booleans
- **Formulae $\sim$ probabilistic teams $\sim$ strings of reals**

Expressivity over formulae

$$\mathrm{FO}(\subseteq) \quad \subsetneq \quad \mathrm{FO}(\subseteq, \mathrm{dep}(\cdots)) \equiv \mathrm{FO}(\perp_{\mathrm{c}})$$

Table: Team semantics

$$\mathrm{FO}(\approx) \quad \subsetneq \quad \mathrm{FO}(\approx, \mathrm{dep}(\cdots)) \quad \subsetneq^{*} \quad \mathrm{FO}(\perp\!\!\!\perp_{\mathrm{c}})$$

Table: Probabilistic team semantics. * New results

Probabilistic inclusion/dependence vs. independence

Both $\approx$ and $\mathrm{dep}(\cdots)$ expressible in $\mathrm{FO}(\perp\!\!\!\perp_{\mathrm{c}})$ (JELIA 2019)

Example

Define $\phi(x) = \exists c \exists y \forall z \theta$ where θ is defined as

$$\mathrm{dep}(c) \wedge x \perp\!\!\!\perp y \wedge x \approx y \wedge ((x = c \wedge y = c) \leftrightarrow z = c). \tag{3}$$

Suppose $\{0,1\} \models_{\mathbb{X}} \phi$. Then

1. $c \in \{0,1\}$ is a constant;
2. z is uniformly distributed, so $z = c$ holds for weights $1/2$;
3. $x = c \wedge y = c$ holds for weight $1/2$;
4. x and y are independent and identically distributed, so $x = c$ holds for weight $1/\sqrt{2}$.

NB. Irrational weights not definable in $\mathrm{FO}(\approx, \mathrm{dep}(\cdots))$.

Theorem

Over formulae, $\mathrm{FO}(\approx, \mathrm{dep}(\cdots)) \subsetneq \mathrm{FO}(\perp\!\!\!\perp_{\mathrm{c}})$

Axioms for quantitative dependence

- Marginal independence ✓ [Geiger et al., 1991]
- Conditional independence ✗ [Studený, 1992]
- Marginal identity ?

Axioms for quantitative dependence

- Marginal independence ✓ [Geiger et al., 1991]
- Conditional independence ✗ [Studený, 1992]
- Marginal identity ✓

Theorem

The following axiomatization is sound and complete:

1. *reflexivity:* $x_1 \dots x_n \approx x_1 \dots x_n$;
2. *symmetry: if* $x_1 \dots x_n \approx y_1 \dots y_n$, *then* $y_1 \dots y_n \approx x_1 \dots x_n$;
3. *projection and permutation: if* $x_1 \dots x_n \approx y_1 \dots y_n$, *then* $x_{i_1} \dots x_{i_k} \approx y_{i_1} \dots y_{i_k}$, *where* $i_1, \dots, i_k$ *is a sequence of distinct integers from* $\{1, \dots, n\}$.
4. *transitivity: if* $x_1 \dots x_n \approx y_1 \dots y_n$ *and* $y_1 \dots y_n \approx z_1 \dots z_n$, *then* $x_1 \dots x_n \approx z_1 \dots z_n$.

Conclusion

- We studied quantitative variants of inclusion and inclusion/dependence logics
- Qualitative and quantitative variants in many ways **analogous**:
  1. Inclusion logic captures P (over ordered models)
  2. Inclusion/dependence logic captures NP
  3. Marginal identity and independence has axioms, conditional independence has not
- Where the **analogy breaks down**:
  1. Relationship between inclusion/dependence logic, independence logic, and NP
     - Qualitative: Both logics capture NP properties of teams
     - Quantitative: additive vs. multiplicative properties of prob.teams. For sentences, former captures NP, latter $\exists[0,1]^{\leq}$

Thanks!

Main sources:

- Durand, A., Hannula, M., Kontinen, J., Meier, A., and Virtema, J. Probabilistic team semantics. (2018). In *Foundations of Information and Knowledge Systems - 10th International Symposium, FoIKS 2018, Budapest, Hungary, May 14-18, 2018, Proceedings*, pages 186–206.
- Hannula, M., Hirvonen, Å., Kontinen, J., Kulikov, V., and Virtema, J. (2019). Facets of distribution identities in probabilistic team semantics. In *JELIA*, volume 11468 of *Lecture Notes in Computer Science*, pages 304–320. Springer.
- Hannula, M., Kontinen, J., Van den Bussche, J., and Virtema, J. Descriptive complexity of real computation and probabilistic independence logic. (2020). In *LICS '20: 35th Annual ACM/IEEE Symposium on Logic in Computer Science, Saarbrücken, Germany, July 8-11, 2020*, pages 550–563.

Blum, L., Shub, M., and Smale, S. (1989).
On a theory of computation and complexity over the real numbers: *np*- completeness, recursive functions and universal machines.
Bull. Amer. Math. Soc. (N.S.), 21(1):1–46.

Bürgisser, P. and Cucker, F. (2006).
Counting complexity classes for numeric computations II: algebraic and semialgebraic sets.
J. Complexity, 22(2):147–191.

Galliani, P. (2012).
Inclusion and exclusion dependencies in team semantics: On some logics of imperfect information.
Annals of Pure and Applied Logic, 163(1):68 – 84.

Galliani, P. and Hella, L. (2013).
Inclusion Logic and Fixed Point Logic.
In Rocca, S. R. D., editor, *Computer Science Logic 2013 (CSL 2013)*, volume 23 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 281–295, Dagstuhl, Germany. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

Geiger, D., Paz, A., and Pearl, J. (1991).
Axioms and algorithms for inferences involving probabilistic independence.
Information and Computation, 91(1):128–141.

Grädel, E. and Gurevich, Y. (1998).
Metafinite model theory.
Inf. Comput., 140(1):26–81.

Grädel, E. and Hegselmann, S. (2016).
Counting in team semantics.
In *25th EACSL Annual Conference on Computer Science Logic, CSL 2016, August 29 - September 1, 2016, Marseille, France*, pages 35:1–35:18.

Grädel, E. and Meer, K. (1995).
Descriptive complexity theory over the real numbers.
In *Proceedings of the Twenty-Seventh Annual ACM Symposium on Theory of Computing, 29 May-1 June 1995, Las Vegas, Nevada, USA*, pages 315–324.

Hannula, M. and Kontinen, J. (2016).
A finite axiomatization of conditional independence and inclusion dependencies.
Inf. Comput., 249:121–137.

Henkin, L. (1961).
Some Remarks on Infinitely Long Formulas.
In *Infinitistic Methods. Proc. Symposium on Foundations of Mathematics*, pages 167–183. Pergamon Press.

Hintikka, J. and Sandu, G. (1989).
Informational independence as a semantic phenomenon.
In Fenstad, J., Frolov, I., and Hilpinen, R., editors, *Logic, methodology and philosophy of science*, pages 571–589. Elsevier.

Hodges, W. (1997).
Compositional Semantics for a Language of Imperfect Information.
Journal of the Interest Group in Pure and Applied Logics, 5 (4):539–563.

Hyttinen, T., Paolini, G., and Väänänen, J. (2015).
Quantum team logic and Bell's inequalities.
The Review of Symbolic Logic, 8:722–742.

Pacuit, E. and Yang, F. (2016).
Dependence and independence in social choice: Arrow's theorem.
In Abramsky, S., Kontinen, J., Väänänen, J., and Vollmer, H., editors, *Dependence Logic: Theory and Applications*, pages 235–260, Cham. Springer International Publishing.

Schaefer, M. and Stefankovic, D. (2017).
Fixed points, nash equilibria, and the existential theory of the reals.
Theory Comput. Syst., 60(2):172–193.

Studený, M. (1992).
Conditional independence relations have no finite complete characterization.
pages 377–396. Kluwer.