Independence logic and tuple existence atoms

Pietro Galliani

Institute for Logic, Language and Computation
Universiteit van Amsterdam

Logiikan Seminaari

Outline

Outline

Tarski's Semantics

- $M \models_s R\vec{t}$ if and only if $\vec{t}\langle s\rangle \in R^M$;
- $M \models_s \neg R\vec{t}$ if and only if $\vec{t}\langle s\rangle \not\in R^M$;
- $M \models_s t_1 = t_2$ if and only if $t_1\langle s\rangle = t_2\langle s\rangle$;
- $M \models_s t_1 \neq t_2$ if and only if $t_1\langle s\rangle = t_2\langle s\rangle$;
- $M \models_s \phi \wedge \psi$ if and only if $M \models_s \phi$ and $M \models_s \psi$;
- $M \models_s \phi \vee \psi$ if and only if $M \models_s \phi$ or $M \models_s \psi$;
- $M \models_s \exists x\phi$ if and only if $\exists m \in \text{Dom}(M)$ s.t. $M \models_{s[m/x]} \phi$;
- $M \models_s \forall x\phi$ if and only if $\forall m \in \text{Dom}(M)$, $M \models_{s[m/x]} \phi$.

From Assignments to Teams

Assignments as states of things

An assignment s is a state of things; $M \models_s \phi$ if ϕ is true in the state of things s.

Teams

A team X is a set of assignments (states of things I believe possible).

Team Semantics

$$M \models_X \phi \text{ if and only if } M \models_s \phi \text{ for all } s \in X.$$

From Assignments to Teams

Assignments as states of things

An assignment s is a state of things; $M \models_s \phi$ if ϕ is true in the state of things s.

Teams

A team X is a set of assignments (states of things I believe possible).

Team Semantics

$M \models_X \phi$ if and only if $M \models_s \phi$ for all $s \in X$.

From Assignments to Teams

Assignments as states of things

An assignment s is a state of things; $M \models_s \phi$ if ϕ is true in the state of things s.

Teams

A team X is a set of assignments (states of things I believe possible).

Team Semantics

$$M \models_X \phi \text{ if and only if } M \models_s \phi \text{ for all } s \in X.$$

Team Semantics

Team Semantics

$M \models_X \phi$ if and only if $M \models_s \phi$ for all $s \in X$.

- If α literal, $M \models_X \alpha$ iff for all $s \in X$, $M \models_s \alpha$;
- $M \models_X \phi \wedge \psi$ iff for all $s \in X$, $M \models_s \phi$ and $M \models_s \psi$;
- $M \models_X \phi \vee \psi$ iff for all $s \in X$, $M \models_s \phi$ or $M \models_s \psi$;
- $M \models_X \exists x \phi$ iff for all $s \in X$, $\exists m \in \text{Dom}(M)$ s.t. $M \models_{s[m/x]} \phi$;
- $M \models_X \forall x \phi$ iff for all $s \in X$, $\forall m \in \text{Dom}(M)$, $M \models_{s[m/x]} \phi$.

Team Semantics

Team Semantics

$M \models_X \phi$ if and only if $M \models_s \phi$ for all $s \in X$.

- If α literal, $M \models_X \alpha$ iff for all $s \in X$, $M \models_s \alpha$;
- $M \models_X \phi \wedge \psi$ iff $M \models_X \phi$ and $M \models_X \psi$;
- $M \models_X \phi \vee \psi$ iff for all $s \in X$, $M \models_s \phi$ or $M \models_s \psi$;
- $M \models_X \exists x \phi$ iff for all $s \in X$, $\exists m \in \text{Dom}(M)$ s.t. $M \models_{s[m/x]} \phi$;
- $M \models_X \forall x \phi$ iff for all $s \in X$, $\forall m \in \text{Dom}(M)$, $M \models_{s[m/x]} \phi$.

Team Semantics

Team Semantics

$M \models_X \phi$ if and only if $M \models_s \phi$ for all $s \in X$.

- If α literal, $M \models_X \alpha$ iff for all $s \in X$, $M \models_s \alpha$;
- $M \models_X \phi \wedge \psi$ iff $M \models_X \phi$ and $M \models_X \psi$;
- $M \models_X \phi \vee \psi$ iff $X = Y \cup Z$, $M \models_Y \phi$ and $M \models_Z \psi$;
- $M \models_X \exists x \phi$ iff for all $s \in X$, $\exists m \in \mathrm{Dom}(M)$ s.t. $M \models_{s[m/x]} \phi$;
- $M \models_X \forall x \phi$ iff for all $s \in X$, $\forall m \in \mathrm{Dom}(M)$, $M \models_{s[m/x]} \phi$.

Team Semantics

Team Semantics

$M \models_X \phi$ if and only if $M \models_s \phi$ for all $s \in X$.

- If α literal, $M \models_X \alpha$ iff for all $s \in X$, $M \models_s \alpha$;
- $M \models_X \phi \wedge \psi$ iff $M \models_X \phi$ and $M \models_X \psi$;
- $M \models_X \phi \vee \psi$ iff $X = Y \cup Z$, $M \models_Y \phi$ and $M \models_Z \psi$;
- $M \models_X \exists x \phi$ iff $\exists F : X \to \text{Dom}(M)$ s.t. $M \models_{X[F/x]} \phi$;
- $M \models_X \forall x \phi$ iff for all $s \in X$, $\forall m \in \text{Dom}(M)$, $M \models_{s[m/x]} \phi$.

Team Semantics

Team Semantics

$M \models_X \phi$ if and only if $M \models_s \phi$ for all $s \in X$.

- If α literal, $M \models_X \alpha$ iff for all $s \in X$, $M \models_s \alpha$;
- $M \models_X \phi \wedge \psi$ iff $M \models_X \phi$ and $M \models_X \psi$;
- $M \models_X \phi \vee \psi$ iff $X = Y \cup Z$, $M \models_Y \phi$ and $M \models_Z \psi$;
- $M \models_X \exists x \phi$ iff $\exists F : X \to \mathrm{Dom}(M)$ s.t. $M \models_{X[F/x]} \phi$;
- $M \models_X \forall x \phi$ iff $M \models_{X[M/x]} \phi$.

Extending Team Semantics?

Team Semantics

$M \models_X \phi$ if and only if $M \models_s \phi$ for all $s \in X$.

- Can add new atomic formulas:

$$M \models_X \text{EVEN}(t) \text{ iff } |\{s\langle t\rangle : s \in X\}| \text{ is even;}$$

- Can add new connectives:

$$M \models_X \sim \phi \text{ iff } M \not\models_X \phi.$$

Extending Team Semantics?

Team Semantics

$M \models_X \phi$ if and only if $M \models_s \phi$ for all $s \in X$.

- **Can add new atomic formulas**:

$$M \models_X \text{EVEN}(t) \text{ iff } |\{s\langle t\rangle : s \in X\}| \text{ is even;}$$

- Can add new connectives:

$$M \models_X \sim \phi \text{ iff } M \not\models_X \phi.$$

Which truth conditions are interesting?

Teams = sets of assignments = relations:

$$X = \{s, s', s''\}$$

	x_1	x_2	x_3	...
s	$s(x_1)$	$s(x_2)$	$s(x_3)$	...
s'	$s'(x_1)$	$s'(x_2)$	$s'(x_3)$	...
s''	$s'(x_1)$	$s''(x_2)$	$s''(x_3)$	...

Conditions over relations

Conditions used in database theory are interesting (and already well-studied).

Which truth conditions are interesting?

Teams = sets of assignments = relations:

$$X = \{s, s', s''\}$$

	x_1	x_2	x_3	...
s	$s(x_1)$	$s(x_2)$	$s(x_3)$	...
s'	$s'(x_1)$	$s'(x_2)$	$s'(x_3)$	...
s''	$s'(x_1)$	$s''(x_2)$	$s''(x_3)$	...

Conditions over relations

Conditions used in database theory are interesting (and already well-studied).

Which truth conditions are interesting?

Teams = sets of assignments = relations:

$$X = \{s, s', s''\}$$

	x_1	x_2	x_3	$\ldots$
s	$s(x_1)$	$s(x_2)$	$s(x_3)$	$\ldots$
s'	$s'(x_1)$	$s'(x_2)$	$s'(x_3)$	$\ldots$
s''	$s'(x_1)$	$s''(x_2)$	$s''(x_3)$	$\ldots$

Conditions over relations

Conditions used in database theory are interesting (and already well-studied).

Outline

Functional Dependencies

Definition

$\vec{x}$, $\vec{y}$ tuples of attributes, R relation.

$$R \models \vec{x} \to \vec{y} \text{ iff for all } r, r' \in R, r(\vec{x}) = r'(\vec{x}) \Rightarrow r(\vec{y}) = r'(\vec{y}).$$

- **The** most important form of dependency;
- Normal forms;
- Axiomatizations.

Examples of Functional Dependency

Person	Date of Birth	Nationality
Hilbert	23/01/1862	German
Gauss	30/04/1777	German
Euler	15/04/1707	Swiss

- $R \models$ Person $\rightarrow$ Date of Birth;
- $R \not\models$ Nationality $\rightarrow$ Person;
- $R \models$ Date of Birth $\rightarrow$ Nationality.

Examples of Functional Dependency

Person	Date of Birth	Nationality
Hilbert	23/01/1862	German
Gauss	30/04/1777	German
Euler	15/04/1707	Swiss

- $R \models$ Person $\rightarrow$ Date of Birth;
- $R \not\models$ Nationality $\rightarrow$ Person;
- $R \models$ Date of Birth $\rightarrow$ Nationality.

Examples of Functional Dependency

Person	Date of Birth	Nationality
Hilbert	23/01/1862	German
Gauss	30/04/1777	German
Euler	15/04/1707	Swiss
Smith	15/04/1707	English

- $R \models$ Person $\rightarrow$ Date of Birth;
- $R \not\models$ Nationality $\rightarrow$ Person;
- $R \not\models$ Date of Birth $\rightarrow$ Nationality!

Armstrong's Axioms

The Implication Problem

Γ (finite) set of functional dependencies. When is it the case that $\Gamma \models \vec{x} \to \vec{y}$, that is, $R \models \Gamma \Rightarrow R \models \vec{x} \to \vec{y}$?

Armstrong 1974

- If $\vec{y} \subseteq \vec{x}$ then $\vdash \vec{x} \to \vec{y}$;
- $\vec{x} \to \vec{y} \vdash \vec{x}\vec{z} \to \vec{y}\vec{z}$;
- $\vec{x} \to \vec{y}, \vec{y} \to \vec{z} \vdash \vec{x} \to \vec{z}$;
- If $\vec{z}$ permutes $\vec{x}$ and $\vec{w}$ permutes $\vec{y}$ then $\vec{x} \to \vec{y} \vdash \vec{z} \to \vec{w}$.

The above axioms are sound and complete.

Dependence Logic (Väänänen 2007)

Dependence Atoms

$M \models_X =\!(t_1 \ldots t_n)$ if and only if

$$\{(t_1\langle s\rangle, \ldots, t_n\langle s\rangle) : s \in X\} \models t_1 \ldots t_{n-1} \rightarrow t_n$$

Dependence Logic (Väänänen 2007)

Dependence Logic $\mathcal{D}$ = First Order Logic (with Team Semantics) + dependence atoms.

Properties of Dependence Logic (all from V. 2007)

The empty team sastisfies everything

For every ϕ and M, $M \models_\emptyset \phi$;

Downwards Closure

If $M \models_X \phi$ and $Y \subseteq X$ then $M \models_Y \phi$;

Stronger than First Order Logic

$M \models \exists z \forall x x' \exists y y' (=(x, y) \wedge =(x', y') \wedge (x = x' \leftrightarrow y = y') \wedge y \neq z)$
if and only if M is infinite

Dependence Logic and Σ_1^1 (V. 2007)

From Dependence Logic to Σ_1^1

For every sentence $\phi \in \mathcal{D}$ there exists a $\phi' \in \Sigma_1^1$ such that

$$M \models_{\{\emptyset\}} \phi \text{ if and only if } M \models \phi';$$

From Σ_1^1 to Dependence Logic

For every sentence $\phi' \in \Sigma_1^1$ there exists a $\phi \in \mathcal{D}$ such that

$$M \models_{\{\emptyset\}} \phi \text{ if and only if } M \models \phi'.$$

Definability in $\mathcal{D}$ (Kontinen, Väänänen 2009)

$\Sigma_1^1(R^-)$

A sentence $\Phi \in \Sigma_1^1$ with signature $\Sigma \cup \{R\}$ is *downwards monotone* for R if

$$M, R \models \Phi, R' \subseteq R \Rightarrow M, R' \models \Phi.$$

Definability in $\mathcal{D}$ (Kontinen, Väänänen 2009)

From Dependence Logic to $\Sigma_1^1(R^-)$

For every formula $\phi \in \mathcal{D}$ there exists a sentence $\phi' \in \Sigma_1^1(R^-)$ such that $M \models_X \phi$ if and only if $M, \text{Rel}(X) \models \phi'$ for all suitable M and all **nonempty** X.

From $\Sigma_1^1(R^-)$ to Dependence Logic

For every sentence $\phi'(R) \in \Sigma_1^1(R^-)$ there exists a formula $\phi \in \mathcal{D}$ such that $M \models_X \phi$ if and only if $M, \text{Rel}(X) \models \phi'$ for all suitable M and all **nonempty** X.

Definability in $\mathcal{D}$ (Kontinen, Väänänen 2009)

From Dependence Logic to $\Sigma_1^1(R^-)$

For every formula $\phi \in \mathcal{D}$ there exists a sentence $\phi' \in \Sigma_1^1(R^-)$ such that $M \models_X \phi$ if and only if $M, \text{Rel}(X) \models \phi'$ for all suitable M and all **nonempty** X.

From $\Sigma_1^1(R^-)$ to Dependence Logic

For every sentence $\phi'(R) \in \Sigma_1^1(R^-)$ there exists a formula $\phi \in \mathcal{D}$ such that $M \models_X \phi$ if and only if $M, \text{Rel}(X) \models \phi'$ for all suitable M and all **nonempty** X.

Outline

Embedded Multivalued Dependencies

Definition

R relation, $\vec{x}, \vec{y}, \vec{z}$ tuples of attributes.
Then $R \models \vec{x} \twoheadrightarrow \vec{y} \mid \vec{z}$ if and only if, for all $r, r' \in R$ such that $r(\vec{x}) = r'(\vec{x})$ there exists a $r'' \in R$ such that

$$r''(\vec{x}\vec{y}) = r(\vec{x}\vec{y}) \text{ and } r''(\vec{x}\vec{z}) = r(\vec{x}\vec{z}).$$

- Huge literature on the topic;
- If $\vec{x}\vec{y}\vec{z}$ contains all attributes of R (*full* mvd), sound and complete axiomatization;
- In general, not known if implication problem is decidable.

Example of Embedded Multivalued Dependency

Professor	Doctoral Student	University
Hilbert	Ackermann	Königsberg
Hilbert	Blumenthal	Königsberg
Hilbert	Ackermann	Göttingen
Hilbert	Blumenthal	Göttingen
Gauss	Bessel	Göttingen
Gauss	Dedekind	Göttingen

- $R \not\models$ Professor $\rightarrow$ Doctoral student;
- $R \not\models$ Professor $\rightarrow$ University;
- $R \models$ Professor $\twoheadrightarrow$ Doctoral Student $|$ University.

Example of Embedded Multivalued Dependency

Professor	Doctoral Student
Hilbert	Ackermann
Hilbert	Blumenthal
Gauss	Bessel
Gauss	Dedekind

Professor	University
Hilbert	Königsberg
Hilbert	Göttingen
Gauss	Göttingen

- $R \not\models$ Professor $\rightarrow$ Doctoral student;
- $R \not\models$ Professor $\rightarrow$ University;
- $R \models$ Professor $\twoheadrightarrow$ Doctoral Student $|$ University.

Independence Logic (Grädel, Väänänen 2010)

Independence Atoms

$M \models_X = \vec{t}_2 \perp_{\vec{t}_1} \vec{t}_3$ if and only if, for all $s, s' \in X$ such that $\vec{t}_1\langle s\rangle = \vec{t}_1\langle s'\rangle$ there exists a $s'' \in X$ such that

$$\vec{t}_1\langle s''\rangle\vec{t}_2\langle s''\rangle = \vec{t}_1\langle s\rangle\vec{t}_2\langle s\rangle,\ \vec{t}_1\langle s''\rangle\vec{t}_3\langle s''\rangle = \vec{t}_1\langle s'\rangle\vec{t}_3\langle s'\rangle.$$

Independence Logic

Independence Logic $\mathcal{I}$ = First Order Logic (with Team Semantics) + independence atoms.

Independence Logic and Embedded Multivalued Dependencies

Observation (Fredrik Engström)

$M \models_X \vec{t}_2 \perp_{\vec{t}_1} \vec{t}_3$ if and only if

$$\{(\vec{t}_1\langle s\rangle, \vec{t}_2\langle s\rangle, \vec{t}_3\langle s\rangle) : s \in X\} \models \vec{t}_1 \twoheadrightarrow \vec{t}_2 \mid \vec{t}_3.$$

Independence Logic and Dependence Logic

Dependence Logic is contained in Independence Logic

$$M \models_X =\!(\vec{t}, t_n) \text{ if and only if } M \models_X t_n \perp_{\vec{t}} t_n.$$

Proof (Grädel, Väänänen 2010):

$$
\begin{aligned}
& M \models_X t_n \perp_{\vec{t}} t_n \Leftrightarrow \\
& \Leftrightarrow \forall s, s' \in X, \text{ if } \vec{t}\langle s\rangle = \vec{t}\langle s'\rangle \text{ then } \exists s'' \in X \text{ s.t.} \\
& \quad \text{s.t. } \vec{t}\langle s''\rangle t_n\langle s''\rangle = \vec{t}\langle s\rangle t_n\langle s\rangle \text{ and } \vec{t}\langle s''\rangle t_n\langle s''\rangle = \vec{t}\langle s'\rangle t_n\langle s'\rangle \Leftrightarrow \\
& \Leftrightarrow \forall s, s' \in X, \text{ if } \vec{t}\langle s\rangle = \vec{t}\langle s'\rangle \text{ then } t_n\langle s\rangle = t_n\langle s'\rangle \Leftrightarrow \\
& \Leftrightarrow M \models_X =\!(\vec{t}, t_n).
\end{aligned}
$$

Independence Logic and Σ_1^1

From Independence Logic to Σ_1^1 (Grädel, Väänänen 2010)

For every sentence $\phi \in \mathcal{I}$ there exists a $\phi' \in \Sigma_1^1$ such that

$$M \models_{\{\emptyset\}} \phi \text{ if and only if } M \models \phi';$$

Corollary (Grädel, Väänänen 2010)

On the level of sentences, $\mathcal{I} \equiv \mathcal{D} \equiv \Sigma_1^1$.

Definability in Independence Logic?

What about formulas in Independence Logic?

Not answered in (Grädel, Väänänen 2010).

> ***Open Problem:*** *Characterize the NP properties of teams that correspond to formulas of independence logic.*

Will answer this in this talk, as a corollary.

Definability in Independence Logic?

What about formulas in Independence Logic?

Not answered in (Grädel, Väänänen 2010).

> ***Open Problem:*** *Characterize the NP properties of teams that correspond to formulas of independence logic.*

Will answer this in this talk, as a corollary.

Outline

Inclusion Dependencies

Definition

R relation, $\vec{x}, \vec{y}$ tuples of attributes, $|\vec{x}| = |\vec{y}|$.
Then $R \models \vec{x} \subseteq \vec{y}$ if and only if for all $r \in R$ there exists an $r' \in R$ such that

$$r(\vec{x}) = r'(\vec{y}).$$

- Fairly well studied;
- Sound and complete axiomatization.

Example of Inclusion Dependency

Professor	University
Hilbert	Königsberg
Hilbert	Göttingen
Gauss	Göttingen

Person	Date of Birth
Hilbert	23/01/1862
Gauss	30/04/1777
Torvalds	28/12/1969

- $R \models \text{Professor} \subseteq \text{Person}$;
- $R \not\models \text{Person} \subseteq \text{Professor}$.

Example of Inclusion Dependency

Professor	University
Hilbert	Königsberg
Hilbert	Göttingen
Gauss	Göttingen

Person	Date of Birth
Hilbert	23/01/1862
Gauss	30/04/1777
Torvalds	28/12/1969

- $R \models \text{Professor} \subseteq \text{Person}$;
- $R \not\models \text{Person} \subseteq \text{Professor}$.

Exclusion Dependencies

Definition

R relation, $\vec{x}, \vec{y}$ tuples of attributes, $|\vec{x}| = |\vec{y}|$.
Then $R \models \vec{x} \mid \vec{y}$ if and only if, for all $r, r' \in R$,

$$r(\vec{x}) \neq r'(\vec{y}).$$

- Often, not used explicity;
- Very commonly used implicitly, for **typing** of attributes;
- Sound and complete axiomatization together with inclusion dependencies.

Example of Exclusion Dependency

Professor	University
Hilbert	Königsberg
Hilbert	Göttingen
Gauss	Göttingen

Person	Date of Birth
Hilbert	23/01/1862
Gauss	30/04/1777
Torvalds	28/12/1969

- $R \models$ University | Date of Birth;
- $R \not\models$ Professor | Person.

Example of Exclusion Dependency

Professor	University
Hilbert	Königsberg
Hilbert	Göttingen
Gauss	Göttingen

Person	Date of Birth
Hilbert	23/01/1862
Gauss	30/04/1777
Torvalds	28/12/1969

- $R \models$ University | Date of Birth;
- $R \not\models$ Professor | Person.

Axioms for Inclusion/Exclusion Dependencies

Axioms for Inclusion Dependencies

- **I1:** $\vdash \vec{x} \subseteq \vec{x}$;
- **I2:** For all $m, n \in \mathbb{N}$ and for all $f : 1 \ldots n \to 1 \ldots m$,

$$x_1 \ldots x_m \subseteq y_1 \ldots y_m \vdash x_{f(1)} \ldots x_{f(n)} \subseteq y_{f(1)} \ldots y_{f(n)};$$

- **I3:** $\vec{x} \subseteq \vec{y}, \vec{y} \subseteq \vec{z} \vdash \vec{x} \subseteq \vec{z}$;

Casanova, Fagin, Papadimitriou 1983:

I1, **I2** and **I3** form a sound and complete system for the inclusion dependency implication problem.

Axioms for Inclusion/Exclusion Dependencies

Axioms for Inclusion Dependencies

- **I1:** $\vdash \vec{x} \subseteq \vec{x}$;
- **I2:** For all $m, n \in \mathbb{N}$ and for all $f : 1 \ldots n \to 1 \ldots m$,

$$x_1 \ldots x_m \subseteq y_1 \ldots y_m \vdash x_{f(1)} \ldots x_{f(n)} \subseteq y_{f(1)} \ldots y_{f(n)};$$

- **I3:** $\vec{x} \subseteq \vec{y}, \vec{y} \subseteq \vec{z} \vdash \vec{x} \subseteq \vec{z}$;

Casanova, Fagin, Papadimitriou 1983:

I1, **I2** and **I3** form a sound and complete system for the inclusion dependency implication problem.

Axioms for Inclusion/Exclusion Dependencies

Axioms for Exclusion Dependencies

- **E1:** $\vec{x} \mid \vec{y} \vdash \vec{y} \mid \vec{x}$;
- **E2:**For all $m, n \in \mathbb{N}$ and for all $f : 1 \ldots n \to 1 \ldots m$,

$$x_{f(1)} \ldots x_{f(n)} \mid y_{f(1)} \ldots y_{f(n)} \subseteq x_1 \ldots x_m \mid y_1 \ldots y_m;$$

- **E3:** $\vec{x} \mid \vec{x} \vdash \vec{z} \mid \vec{w}$;

Axioms for Inclusion/Exclusion Dependencies

Axioms for Exclusion/Inclusion Interaction

- **IE1:** $\vec{x} \mid \vec{x} \vdash \vec{z} \subseteq \vec{w}$;
- **IE2:** $\vec{x} \subseteq \vec{y}, \vec{z} \subseteq \vec{w}, \vec{y} \mid \vec{w} \vdash \vec{x} \mid \vec{z}$.

Casanova 1983:

I1, **I2**, **I3**, **E1**, **E2**, **E3**, **IE1** and **IE2** form a sound and complete system for the inclusion/exclusion dependency implication problem.

Axioms for Inclusion/Exclusion Dependencies

Axioms for Exclusion/Inclusion Interaction

- **IE1:** $\vec{x} \mid \vec{x} \vdash \vec{z} \subseteq \vec{w}$;
- **IE2:** $\vec{x} \subseteq \vec{y}, \vec{z} \subseteq \vec{w}, \vec{y} \mid \vec{w} \vdash \vec{x} \mid \vec{z}$.

Casanova 1983:

I1, **I2**, **I3**, **E1**, **E2**, **E3**, **IE1** and **IE2** form a sound and complete system for the inclusion/exclusion dependency implication problem.

Inclusion + Functional Dependencies is undecidable!

Mitchell 1983, Chandra, Vardi 1985:

The implication problem for the class of functional and inclusion dependencies is undecidable.

Proof Sketch:
The word problem for monoids is known to be undecidable from (Post 1947), and is reducible to the implication problem for inclusion and functional dependencies.

Outline

Tuple Existence Atoms

$M \models_X \vec{t}_1 \,@\, \vec{t}_2$ if and only if $\{(\vec{t}_1\langle s\rangle, \vec{t}_2\langle s\rangle) : s \in X\} \models \vec{t}_1 \subseteq \vec{t}_2$;

Negated Tuple Existence Atoms

$M \models_X \neg(\vec{t}_1 \,@\, \vec{t}_2)$ if and only if $\{(\vec{t}_1\langle s\rangle, \vec{t}_2\langle s\rangle) : s \in X\} \models \vec{t}_1 \mid \vec{t}_2$.

Tuple Existence Logic

Tuple Existence Logic Logic $\mathcal{T}$ = First Order Logic (with Team Semantics) + tuple existence literals.
$\mathcal{T}^-$ = only negated tuple existence atoms,
$\mathcal{T}^+$ = only non-negated tuple existence atoms.

Direct Definitions for Tuple Existence Literals Semantics

Tuple Existence Atoms

$M \models_X = \vec{t}_1 \ @ \ \vec{t}_2$ if and only if for all $s \in X$ there exists a $s' \in X$ such that

$$\vec{t}_1\langle s\rangle = \vec{t}_2\langle s'\rangle;$$

Negated Tuple Existence Atoms

$M \models_X = \neg(\vec{t}_1 \ @ \ \vec{t}_2)$ if and only if, for all $s, s' \in X$,

$$\vec{t}_1\langle s\rangle \neq \vec{t}_2\langle s'\rangle.$$

What do we want to know about $\mathcal{T}$?

- What is the relation between Dependence Logic and Tuple Existence Logic?
- What is the relation between Independence Logic and Tuple Existence Logic?
- What is the expressive power of Tuple Existence Logic over open formulas?

Outline

From $\mathcal{T}^-$ to $\mathcal{D}$

Dependence atoms are expressible in $\mathcal{T}^-$

The dependence atom $=\!(t_1 \ldots t_n)$ is equivalent to the expression

$$\forall z(z = t_n \vee \neg(t_1 \ldots t_{n-1} z \,@\, t_1 \ldots t_{n-1} t_n)).$$

From $\mathcal{T}^-$ to $\mathcal{D}$

Dependence atoms are expressible in $\mathcal{T}^-$ (simple case)

The dependence atom $=\!(x, y)$ is equivalent to the expression

$$\forall z(z = y \vee \neg(xz \;@\; xy)).$$

x	y
$s_1(x)$	$s_1(y)$
$s_2(x)$	$s_2(y)$
...	...

$$s_i(x) = s_j(x) \Rightarrow s_i(y) = s_j(y)$$

From $\mathcal{T}^-$ to $\mathcal{D}$

Dependence atoms are expressible in $\mathcal{T}^-$ (simple case)

The dependence atom $=\!(x, y)$ is equivalent to the expression

$$\forall z(z = y \vee \neg(xz \circledcirc xy)).$$

x	y	z
$s_1(x)$	$s_1(y)$	m_1
$s_1(x)$	$s_1(y)$	m_2
$s_1(x)$	$s_1(y)$	$\ldots$
$s_2(x)$	$s_2(y)$	m_1
$s_2(x)$	$s_2(y)$	m_2
$s_2(x)$	$s_2(y)$	$\ldots$
$\ldots$	$\ldots$	$\ldots$

$s_i(x) = s_j(x) \Rightarrow s_i(y) = s_j(y)$
Suppose $m_k \neq s_i(y)$.
Then for all j:
If $s_i(x) = s_j(x)$,

$$s_i(y) = s_j(y) \neq m_k.$$

From $\mathcal{T}^-$ to $\mathcal{D}$

Dependence atoms are expressible in $\mathcal{T}^-$ (simple case)

The dependence atom $=\!(x, y)$ is equivalent to the expression

$$\forall z(z = y \vee \neg(xz \;@\; xy)).$$

Proof (Left to Right).

Suppose $M \models_X =\!(x, y)$, let $Y = \{s[m/z] : s \in X, m \neq s(y)\}$.
If $M \models_Y \neg(xz \;@\; xy)$, done.
So take $h, h' \in Y$, $h(x) = h'(x)$, $h'(y) = h(z) \neq h(y)$.
Contradiction. □

From $\mathcal{T}^-$ to $\mathcal{D}$

Dependence atoms are expressible in $\mathcal{T}^-$ (simple case)

The dependence atom $=\!(x, y)$ is equivalent to the expression

$$\forall z(z = y \vee \neg(xz \circledcirc xy)).$$

Proof (Right to Left).

Suppose $M \not\models_X =\!(x, y)$. Then exist $s, s' \in X$ s.t. $s(x) = s'(x)$, $s(y) \neq s'(y)$.
Consider $h = s[s'(y)/z]$, $h' = s'[s(y)/z]$.
$h(y) \neq h(z)$, $h'(y) \neq h'(z)$.
But $h(x) = s(x) = s'(x) = h'(x)$ and $h(z) = s'(y) = h'(y)$.
So $M \not\models_X \forall z(z = y \vee \neg(xz \circledcirc xy))$. □

From $\mathcal{D}$ to $\mathcal{T}^-$

Negated tuple existence atoms are expressible in $\mathcal{D}$

There exists a formula ϕ in Dependence Logic such that

$$M \models_X \phi \text{ if and only if } M \models_X \neg(\vec{t}_1 \circledcirc \vec{t}_2)$$

Proof.

$\neg(\vec{t}_1 \circledcirc \vec{t}_2)$ holds of the empty team, and $M \models_X \neg(\vec{t}_1 \circledcirc \vec{t}_2)$ iff

$$M, \mathrm{Rel}(X) \models \forall \vec{s}_1 \vec{s}_2 (R\vec{s}_1 \wedge R\vec{s}_2 \rightarrow \vec{t}_1\langle \vec{s}_1 \rangle \neq \vec{t}_2\langle \vec{s}_2 \rangle).$$

By KV 2009, this is expressible in Dependence Logic. □

Tuple Existence Logic and Dependence Logic

Corollary

FO(Team) + Functional Dep. = FO(Team) + Exclusion Dep.

Even wrt open formulas!

Outline

Backslashed Disjunction

Backslashed Disjunction

$M \models_X \phi \vee_{\vec{t}} \psi \Leftrightarrow X = Y \cup Z, M \models_Y \phi$, $M \models_Z \psi$ and the splitting depends only on $\vec{t}$: for all $s, s' \in X$, if $\vec{t}\langle s \rangle = \vec{t}\langle s' \rangle$ then

$$s \in Y[Z] \Leftrightarrow s' \in Y[Z].$$

Backslashed Disjunction

Backslashed Disjunction is expressible in Dependence Logic

$\phi \vee_{\vec{t}} \psi$ is equivalent to

$$\exists u_1 \ldots u_4 \left(\bigwedge_i =\!(\vec{t}, u_i) \wedge ((u_1 = u_2 \wedge \phi) \vee (u_3 = u_4 \wedge \psi)) \right).$$

Corollary

Backslashed Quantification is expressible in $\mathcal{T}$.

From $\mathcal{T}$ to $\mathcal{I}$

Independence atoms in $\mathcal{T}$

$\vec{t}_2 \perp_{\vec{t}_1} \vec{t}_3$ is equivalent to

$$\forall \vec{p}_1\vec{p}_2\vec{p}_3(\neg(\vec{p}_1\vec{p}_2 \circledcirc \vec{t}_1\vec{t}_2) \vee_{\vec{p}_1\vec{p}_2\vec{p}_3} \neg(\vec{p}_1\vec{p}_3 \circledcirc \vec{t}_1\vec{t}_3) \vee_{\vec{p}_1\vec{p}_2\vec{p}_3}$$
$$\vee_{\vec{p}_1\vec{p}_2\vec{p}_3} \vec{p}_1\vec{p}_2\vec{p}_3 \circledcirc \vec{t}_1\vec{t}_2\vec{t}_3).$$

From $\mathcal{T}$ to $\mathcal{I}$

Independence atoms in $\mathcal{T}$ (simple case)

$y \perp_x z$ is equivalent to the expression

$$\forall p_1 p_2 p_3 (\neg(p_1 p_2 @ xy) \vee_{\vec{p}} \neg(p_1 p_3 @ xz) \vee_{\vec{p}} p_1 p_2 p_3 @ xyz).$$

x	y	z
$s_1(x)$	$s_1(y)$	$s_1(z)$
$s_2(x)$	$s_2(y)$	$s_2(z)$
...	...	...

$\mathbf{m_1, m_2, m_3 \in M}$.

If $s_i(x) = s_j(x)$ then exists k,
$s_k(xy) = s_i(xy)$,
$s_k(xz) = s_j(xz)$.

1. $\forall i, s_i(xy) \neq m_1 m_2$;
2. $\forall j, s_j(xz) \neq m_1 m_3$;
3. $\exists k, s_k(xyz) = m_1 m_2 m_3$.

From $\mathcal{T}$ to $\mathcal{I}$

Independence atoms in $\mathcal{T}$ (simple case)

$y \perp_x z$ is equivalent to the expression

$$\forall p_1 p_2 p_3 (\neg(p_1 p_2 \circledcirc xy) \vee_{\vec{p}} \neg(p_1 p_3 \circledcirc xz) \vee_{\vec{p}} p_1 p_2 p_3 \circledcirc xyz).$$

x	y	z
$s_1(x)$	$s_1(y)$	$s_1(z)$
$s_2(x)$	$s_2(y)$	$s_2(z)$
...	...	...

$\mathbf{m_1, m_2, m_3} \in \mathbf{M}$.

If $s_i(x) = s_j(x)$ then exists k,
$s_k(xy) = s_i(xy)$,
$s_k(xz) = s_j(xz)$.

1. $\forall i, s_i(xy) \neq m_1 m_2$;
2. $\forall j, s_j(xz) \neq m_1 m_3$;
3. $\exists k, s_k(xyz) = m_1 m_2 m_3$.

From $\mathcal{T}$ to $\mathcal{I}$

Independence atoms in $\mathcal{T}$ (simple case)

$y \perp_x z$ is equivalent to the expression

$$\forall p_1 p_2 p_3 (\neg (p_1 p_2 @ xy) \vee_{\vec{p}} \neg (p_1 p_3 @ xz) \vee_{\vec{p}} p_1 p_2 p_3 @ xyz).$$

x	y	z
$s_1(x)$	$s_1(y)$	$s_1(z)$
$s_2(x)$	$s_2(y)$	$s_2(z)$
...	...	...

$\mathbf{m_1}, \mathbf{m_2}, \mathbf{m_3} \in \mathbf{M}$.

If $s_i(x) = s_j(x)$ then exists k,
$s_k(xy) = s_i(xy)$,
$s_k(xz) = s_j(xz)$.

1. $\forall i, s_i(xy) \neq m_1 m_2$;
2. $\forall j, s_j(xz) \neq m_1 m_3$;
3. $\exists k, s_k(xyz) = m_1 m_2 m_3$.

From $\mathcal{T}$ to $\mathcal{I}$

Independence atoms in $\mathcal{T}$ (simple case)

$y \perp_x z$ is equivalent to the expression

$$\forall p_1 p_2 p_3 (\neg(p_1 p_2 \text{ @ } xy) \vee_{\vec{p}} \neg(p_1 p_3 \text{ @ } xz) \vee_{\vec{p}} p_1 p_2 p_3 \text{ @ } xyz).$$

x	y	z
$s_1(x)$	$s_1(y)$	$s_1(z)$
$s_2(x)$	$s_2(y)$	$s_2(z)$
...	...	...

$\mathbf{m_1, m_2, m_3 \in M}$.

If $s_i(x) = s_j(x)$ then exists k,
$s_k(xy) = s_i(xy)$,
$s_k(xz) = s_j(xz)$.

1. $\forall i,\ s_i(xy) \neq m_1 m_2$;
2. $\forall j,\ s_j(xz) \neq m_1 m_3$;
3. $\exists k,\ s_k(xyz) = m_1 m_2 m_3$.

From $\mathcal{T}$ to $\mathcal{I}$

Independence atoms in $\mathcal{T}$ (simple case)

$y \perp_x z$ is equivalent to the expression

$$\forall p_1 p_2 p_3 (\neg(p_1 p_2 \circledcirc xy) \vee_{\vec{p}} \neg(p_1 p_3 \circledcirc xz) \vee_{\vec{p}} p_1 p_2 p_3 \circledcirc xyz).$$

x	y	z
$s_1(x)$	$s_1(y)$	$s_1(z)$
$s_2(x)$	$s_2(y)$	$s_2(z)$
...	...	...

$\mathbf{m_1}, \mathbf{m_2}, \mathbf{m_3} \in \mathbf{M}$.

If $s_i(x) = s_j(x)$ then exists k,
$s_k(xy) = s_i(xy)$,
$s_k(xz) = s_j(xz)$.

1. $\forall i,\ s_i(xy) \neq m_1 m_2$;
2. $\forall j,\ s_j(xz) \neq m_1 m_3$;
3. $\exists k,\ s_k(xyz) = m_1 m_2 m_3$.

From $\mathcal{T}$ to $\mathcal{I}$

Independence atoms in $\mathcal{T}$ (simple case)

$y \perp_x z$ is equivalent to the expression

$$\forall p_1 p_2 p_3 (\neg(p_1 p_2 \copyright\, xy) \vee_{\vec{p}} \neg(p_1 p_3 \copyright\, xz) \vee_{\vec{p}} p_1 p_2 p_3 \copyright\, xyz).$$

Proof (Left to Right).

Suppose $M \models_X y \perp_x z$, let $h \in X[M/p_1 p_2 p_3]$.

- If $\forall s \in X, s(xy) \neq h(p_1 p_2)$, $h \in Y_1$: $M \models_{Y_1} \neg(p_1 p_2 \copyright\, xy)$;
- If $\forall s \in X, s(xz) \neq h(p_1 p_3)$, $h \in Y_2$: $M \models_{Y_2} \neg(p_1 p_3 \copyright\, xz)$.
- Otherwise, $h \in Y_3$: $M \models_{Y_3} p_1 p_2 p_3 \copyright\, xyz$.

□

From $\mathcal{T}$ to $\mathcal{I}$

Independence atoms in $\mathcal{T}$ (simple case)

$y \perp_x z$ is equivalent to the expression

$$\forall p_1 p_2 p_3 (\neg(p_1 p_2 @ xy) \vee_{\vec{p}} \neg(p_1 p_3 @ xz) \vee_{\vec{p}} p_1 p_2 p_3 @ xyz).$$

Proof (Left to Right).

Suppose $M \models_X y \perp_x z$, let $h \in X[M/p_1 p_2 p_3]$.

- If $\forall s \in X, s(xy) \neq h(p_1 p_2)$, $h \in Y_1$: $M \models_{Y_1} \neg(p_1 p_2 @ xy)$;
- If $\forall s \in X, s(xz) \neq h(p_1 p_3)$, $h \in Y_2$: $M \models_{Y_2} \neg(p_1 p_3 @ xz)$.
- Otherwise, $h \in Y_3$: $M \models_{Y_3} p_1 p_2 p_3 @ xyz$.

□

From $\mathcal{T}$ to $\mathcal{I}$

Independence atoms in $\mathcal{T}$ (simple case)

$y \perp_x z$ is equivalent to the expression

$$\forall p_1 p_2 p_3 (\neg(p_1 p_2 \circledcirc xy) \vee_{\vec{p}} \neg(p_1 p_3 \circledcirc xz) \vee_{\vec{p}} p_1 p_2 p_3 \circledcirc xyz).$$

Proof (Left to Right).

Suppose $M \models_X y \perp_x z$, let $h \in X[M/p_1 p_2 p_3]$.

- If $\forall s \in X, s(xy) \neq h(p_1 p_2)$, $h \in Y_1$: $M \models_{Y_1} \neg(p_1 p_2 \circledcirc xy)$;
- If $\forall s \in X, s(xz) \neq h(p_1 p_3)$, $h \in Y_2$: $M \models_{Y_2} \neg(p_1 p_3 \circledcirc xz)$.
- Otherwise, $h \in Y_3$: $M \models_{Y_3} p_1 p_2 p_3 \circledcirc xyz$.

□

From $\mathcal{T}$ to $\mathcal{I}$

Independence atoms in $\mathcal{T}$ (simple case)

$y \perp_x z$ is equivalent to the expression

$$\forall p_1 p_2 p_3 (\neg(p_1 p_2 @ xy) \vee_{\vec{p}} \neg(p_1 p_3 @ xz) \vee_{\vec{p}} p_1 p_2 p_3 @ xyz).$$

Proof (Left to Right).

Suppose $M \models_X y \perp_x z$, let $h \in X[M/p_1 p_2 p_3]$.

- If $\forall s \in X, s(xy) \neq h(p_1 p_2)$, $h \in Y_1$: $M \models_{Y_1} \neg(p_1 p_2 @ xy)$;
- If $\forall s \in X, s(xz) \neq h(p_1 p_3)$, $h \in Y_2$: $M \models_{Y_2} \neg(p_1 p_3 @ xz)$.
- Otherwise, $h \in Y_3$: $M \models_{Y_3} p_1 p_2 p_3 @ xyz$.

□

From $\mathcal{T}$ to $\mathcal{I}$

Proof (Right to Left).

Suppose $M \not\models_X y \perp_x z$: $\exists s, s' \in X$ s.t $s(x) = s'(x)$, but $s'' \in X \Rightarrow s''(xy) \neq s(xy)$ or $s''(xz) \neq s'(xz)$.

$$m_1 = s(x) = s'(x),\ m_2 = s(y),\ m_3 = s'(z).$$

$$h = s[m_1/p_1][m_2/p_2][m_3/p_3],\ h' = s'[m_1/p_1][m_2/p_2][m_3/p_3].$$

1. $h, h' \in Y_1$, $M \models_{Y_1} \neg p_1 p_2 \text{ @ } xy$: NO, $h(xy) = h(p_1 p_2)$;
2. $h, h' \in Y_2$, $M \models_{Y_2} \neg p_1 p_3 \text{ @ } xz$: NO, $h'(xz) = h'(p_1 p_3)$;
3. $h, h' \in Y_3$, $M \models_{Y_3} p_1 p_2 p_3 \text{ @ } xyz$: NO, contradiction.

□

From $\mathcal{T}$ to $\mathcal{I}$

Proof (Right to Left).

Suppose $M \not\models_X y \perp_x z$: $\exists s, s' \in X$ s.t $s(x) = s'(x)$, but
$s'' \in X \Rightarrow s''(xy) \neq s(xy)$ or $s''(xz) \neq s'(xz)$.

$$m_1 = s(x) = s'(x),\ m_2 = s(y),\ m_3 = s'(z).$$

$$h = s[m_1/p_1][m_2/p_2][m_3/p_3],\ h' = s'[m_1/p_1][m_2/p_2][m_3/p_3].$$

1. $h, h' \in Y_1$, $M \models_{Y_1} \neg p_1 p_2$ @ xy: NO, $h(xy) = h(p_1 p_2)$;
2. $h, h' \in Y_2$, $M \models_{Y_2} \neg p_1 p_3$ @ xz: NO, $h'(xz) = h'(p_1 p_3)$;
3. $h, h' \in Y_3$, $M \models_{Y_3} p_1 p_2 p_3$ @ xyz: NO, contradiction.

□

From $\mathcal{I}$ to $\mathcal{T}$

Tuple Existence Atoms in $\mathcal{I}$

$\vec{t}_1 \, @ \, \vec{t}_2$ is equivalent to

$$\forall u_1 u_2 \vec{z}((\vec{z} \neq \vec{t}_1 \wedge \vec{z} \neq \vec{t}_2) \vee (u_1 \neq u_2 \wedge \vec{z} \neq \vec{t}_2) \vee$$
$$\vee ((u_1 = u_2 \vee \vec{z} = \vec{t}_2) \wedge \vec{z} \perp_{\emptyset} u_1 u_2)).$$

From $\mathcal{I}$ to $\mathcal{T}$

Tuple Existence Atoms in $\mathcal{I}$ (simple case)

$x \,@\, y$ is equivalent to

$$\forall u_1 u_2 z((z \neq x \wedge z \neq y) \vee (u_1 \neq u_2 \wedge z \neq y) \vee \\ \vee ((u_1 = u_2 \vee z = y) \wedge z \perp_{\emptyset} u_1 u_2)).$$

x	y
$s_1(x)$	$s_1(y)$
$s_2(x)$	$s_2(y)$
...	...

For all i there exists a j, $s_j(y) = s_i(x)$.

From $\mathcal{I}$ to $\mathcal{T}$

Tuple Existence Atoms in $\mathcal{I}$ (simple case)

$$\forall u_1 u_2 z((z \neq x \wedge z \neq y) \vee (u_1 \neq u_2 \wedge z \neq y) \vee \vee((u_1 = u_2 \vee z = y) \wedge z \perp_\emptyset u_1 u_2)).$$

x	y	$u_1 = u_2$?	z
$s_1(x)$	$s_1(y)$	Y	$s_1(x)$
$s_1(x)$	$s_1(y)$	Y	$s_1(y)$
$s_1(x)$	$s_1(y)$	N	$s_1(y)$
$s_2(x)$	$s_2(y)$	Y	$s_2(x)$
$s_2(x)$	$s_2(y)$	Y	$s_2(y)$
$s_2(x)$	$s_2(y)$	N	$s_2(y)$
...	...	...	...

For all i there exists a j, $s_j(y) = s_i(x)$.

If $u_1 = u_2$, $z \in \{x, y\}$;
If $u_1 \neq u_2$, $z = y$.

$(\text{N}, z = s_i(y)) \mapsto (\text{Y}, z = s_j(y))$;
$(\text{Y}, z = s_i(x)) \mapsto (\text{N}, z = s_j(y))$.

From $\mathcal{I}$ to $\mathcal{T}$

Tuple Existence Atoms in $\mathcal{I}$ (simple case)

$$x \mathbin{@} y \equiv \forall u_1 u_2 z((z \neq x \wedge z \neq y) \vee (u_1 \neq u_2 \wedge z \neq y) \vee \\ \vee ((u_1 = u_2 \vee z = y) \wedge z \perp_\emptyset u_1 u_2)).$$

Proof (Left to Right).

$Y = \{s[m_1/u_1][m_2/u_2][m_3/z] : s \in X, m_1 = m_2$ and and $m_3 \in \{s(x), s(y)\}$, or $m_1 \neq m_2$ and $m_3 = s(y)\}$.

If I show that $Y \models z \perp_\emptyset u_1 u_2$, done. Take $s, s' \in Y$.

If $s(z) = s(y)$, $s[s'(u_1)/u_1][s'(u_2)/u_2] \in Y$;

If $s(z) = s(x), \exists s'' \in X$, $s''(y) = s(x)$;

Then $s''[s'(u_1)/u_1][s'(u_2)/u_2][s(z)/z] \in Y$, done. □

From $\mathcal{I}$ to $\mathcal{T}$

Tuple Existence Atoms in $\mathcal{I}$ (simple case)

$$x \circledast y \equiv \forall u_1 u_2 z((z \neq x \wedge z \neq y) \vee (u_1 \neq u_2 \wedge z \neq y) \vee \\ \vee ((u_1 = u_2 \vee z = y) \wedge z \perp_\emptyset u_1 u_2)).$$

Proof (Right to Left).

$s \in X$, $h = s[0/u_1][0/u_2][s(x)/z]$, $h' = s[0/u_1][1/u_2][s(y)/z]$.
$h, h' \in Y$, $Y \models z \perp_\emptyset u_1 u_2$?
Then $\exists h''$, $h''(u_1) = 0$, $h''(u_2) = 1$, $h''(z) = h(z) = s(x)$.
But then $h''(y) = h''(z) = s(x)$. □

Tuple Existence Logic and Independence Logic

Corollary

Independence Logic is equivalent to Tuple Existence Logic.

Corollary

Independence Logic = Dependence Logic + Inclusion Dep.

Even wrt open formulas!

Outline

Definability in Tuple Existence Logic

From Tuple Existence Logic to Σ_1^1

For every formula $\phi \in \mathcal{T}$ there exists a sentence $\phi' \in \Sigma_1^1$ such that $M \models_X \phi$ if and only if $M, \mathrm{Rel}(X) \models \phi'$ for all suitable M and all **nonempty** X.

From Σ_1^1 to Tuple Existence Logic

For every sentence $\phi'(R) \in \Sigma_1^1$ there exists a formula $\phi \in \mathcal{T}$ such that $M \models_X \phi$ if and only if $M, \mathrm{Rel}(X) \models \phi'$ for all suitable M and all **nonempty** X.

Thanks to Juha Kontinen for pointing out this requirement!

Definability in Tuple Existence Logic

From Tuple Existence Logic to Σ_1^1

For every formula $\phi \in \mathcal{T}$ there exists a sentence $\phi' \in \Sigma_1^1$ such that $M \models_X \phi$ if and only if $M, \mathrm{Rel}(X) \models \phi'$ for all suitable M and all **nonempty** X.

From Σ_1^1 to Tuple Existence Logic

For every sentence $\phi'(R) \in \Sigma_1^1$ there exists a formula $\phi \in \mathcal{T}$ such that $M \models_X \phi$ if and only if $M, \mathrm{Rel}(X) \models \phi'$ for all suitable M and all **nonempty** X.

Thanks to Juha Kontinen for pointing out this requirement!

Corollary: Definability on Independence Logic

From Independence Logic to Σ_1^1

For every formula $\phi \in \mathcal{I}$ there exists a sentence $\phi' \in \Sigma_1^1$ such that $M \models_X \phi$ if and only if $M, \mathrm{Rel}(X) \models \phi'$ for all suitable M and all **nonempty** X.

From Σ_1^1 to Independence Logic

For every sentence $\phi'(R) \in \Sigma_1^1$ there exists a formula $\phi \in \mathcal{I}$ such that $M \models_X \phi$ if and only if $M, \mathrm{Rel}(X) \models \phi'$ for all suitable M and all **nonempty** X.

Outline

Left to Right

From Tuple Existence Logic to Σ_1^1

For every formula $\phi \in \mathcal{T}$ there exists a sentence $\phi' \in \Sigma_1^1$ such that $M \models_X \phi$ if and only if $M, \text{Rel}(X) \models \phi'$ for all suitable M and all **nonempty** X.

Proof.

By structural induction of ϕ, easy. □

Left to Right

From Tuple Existence Logic to Σ^1_1

For every formula $\phi \in \mathcal{T}$ there exists a sentence $\phi' \in \Sigma^1_1$ such that $M \models_X \phi$ if and only if $M, \mathrm{Rel}(X) \models \phi'$ for all suitable M and all **nonempty** X.

Proof.

By structural induction of ϕ, easy. □

Outline

Right to Left

From Σ_1^1 to Tuple Existence Logic

For every sentence $\phi'(R) \in \Sigma_1^1$ there exists a formula $\phi \in \mathcal{T}$ such that $M \models_X \phi$ if and only if $M, \mathrm{Rel}(X) \models \phi'$ for all suitable M and all **nonempty** X.

Proof.

Similar to the one for $\mathcal{D}$ in Kontinen and Väänänen, 2009.
Write $\phi'(R)$ as $\exists R' \exists \vec{f}\, \forall \vec{z}((R'\vec{x} \leftrightarrow R\vec{x}) \wedge \psi(R', \vec{z}))$ where $\vec{x}$ subsequence of $\vec{z}$,
ψ quantifier free, R not in ψ, each f_i only as $f_i(\vec{w}_i)$ for some fixed $\vec{w}_i \subseteq \vec{z}$, R' only as $R'\vec{x}$. □

Right to Left

Proof (continued).

Write $\phi'(R)$ as $\exists R' \exists \vec{f}\, \forall \vec{z}((R'\vec{x} \leftrightarrow R\vec{x}) \wedge \psi(R', \vec{z}))$ where $\vec{x}$ subsequence of $\vec{z}$,
ψ quantifier free, R not in ψ, each f_i only as $f_i(\vec{w}_i)$ for some fixed $\vec{w}_i \subseteq \vec{z}$, R' only as $R'\vec{x}$.
Then $M, \mathrm{Rel}(X) \models \phi'$ if and only if

$$M, \mathrm{Rel}(X) \models \exists g_1 g_2 \exists \vec{f}\, \forall \vec{z}((f_1(\vec{x}) = f_2(\vec{x}) \leftrightarrow R\vec{x}) \wedge \psi'(\vec{z}))$$

where $\psi' = \psi[f_1\vec{x} = f_2\vec{x} / R\vec{x}]$. □

Right to Left

Proof (continued).

$$\phi' \equiv \exists g_1 g_2 \exists \vec{f} \forall \vec{z}((g_1(\vec{x}) = g_2(\vec{x}) \leftrightarrow R\vec{x}) \wedge \psi'(\vec{z}))$$

where $\psi' = \psi[g_1\vec{x} = g_2\vec{x}/R\vec{x}]$.
Then, if X nonempty, $\text{Dom}(X) = \vec{y}$, $M, \text{Rel}(X) \models \phi'$ iff

$$M \models_X \forall \vec{z} \exists u_1 u_2 \vec{v}\left(\left(\bigwedge_{i=1}^{2} =\!(\vec{x}, u_i) \wedge \bigwedge_j =\!(\vec{w}_j, v_j) \right) \wedge \right.$$

$$\left. \wedge ((\vec{x} \circledcirc \vec{y} \wedge u_1 = u_2) \vee (\neg \vec{x} \circledcirc \vec{y} \wedge u_1 \neq u_2)) \wedge \theta \right)$$

where θ is $\psi'[u_1/g_1\vec{x}][u_2/g_2\vec{x}][\vec{w}/\vec{f}\vec{w}]$. □

Right to Left

Proof (continued).

Suppose that, for all s with domain $\vec{z}$,

$$M, \mathrm{Rel}(X), g_1, g_2, \vec{f} \models (g_1(\vec{x}) = g_2(\vec{x}) \leftrightarrow R\vec{x}) \wedge \psi'(\vec{z}).$$

Extend X to Y choosing the u_1, u_2, $\vec{v}$ according to g_1, g_2, $\vec{f}$.

- $M \models_Y \bigwedge_{i=1}^{2} =\!(\vec{x}, u_i) \wedge \bigwedge_j =\!(\vec{w}_j, v_j)$: obvious;
- $M \models_Y \theta$: by construction;
- $M \models_Y (\vec{x} \,@\, \vec{y} \wedge u_1 = u_2) \vee (\neg \vec{x} \,@\, \vec{y} \wedge u_1 \neq u_2)$:
  If $u_1 = u_2$, $\vec{x} \in \mathrm{Rel}(X)$, so $\vec{x} \,@\, \vec{y}$;
  If $u_1 \neq u_2$, $\vec{x} \notin \mathrm{Rel}(X)$, so $\neg \vec{x} \,@\, \vec{y}$.

□

Right to Left

Proof (continued).

Conv., suppose X nonempty, $Y = X[M/\vec{z}][G_1/u_1][G_2/u_2][\vec{F}/\vec{v}]$,

$$M \models_Y \bigwedge_{i=1}^{2} =(\vec{x}, u_i) \wedge \bigwedge_j =(\vec{w}_j, v_j),$$
$$M \models_Y (\vec{x} \mathbin{@} \vec{y} \wedge u_1 = u_2) \vee (\neg \vec{x} \mathbin{@} \vec{y} \wedge u_1 \neq u_2),$$
$$M \models_Y \theta.$$

Choose $g_1(\vec{x}), g_2(\vec{x}), \vec{f}(\vec{w})$ according to $G_1, G_2, \vec{F}$.
Let s be any assignment, domain = $\vec{z}$. □

Right to Left

Proof (continued).

Choose $g_1(\vec{x}), g_2(\vec{x}), \vec{f}(\vec{w})$ according to $G_1, G_2, \vec{F}$.
Let s be any assignment, domain = $\vec{z}$.

- $M, \text{Rel}(R), g_1, g_2, \vec{f} \models_s \psi'$: Take $h \in X$. Then $h[s/\vec{z}][g_1/u_1][g_2/u_2][\vec{f}/\vec{v}] \in Y$, $M \models_Y \theta$.
- $M, \text{Rel}(R), g_1, g_2, \vec{f} \models_s g_1(\vec{x}) = g_2(\vec{x}) \leftrightarrow R\vec{x}$:
  Suppose $g_1(\vec{x}) = g_2(\vec{x})$, let $h \in X$.
  Consider $o = h[s/\vec{z}][g_1/u_1][g_2/u_2][\vec{f}/\vec{v}]$:
  $o \in Y_1, M \models_{Y_1} \vec{x} \text{ @ } \vec{y}$. So $\exists o' \in Y_1$, $o'(\vec{y}) = o(\vec{x})$, so $s(\vec{x}) = o(\vec{x}) \in \text{Rel}(X)$.

□

Right to Left

Proof (finished).

Choose $g_1(\vec{x}), g_2(\vec{x}), \vec{f}(\vec{w})$ according to $G_1, G_2, \vec{F}$.
Let s be any assignment, domain = $\vec{z}$.

- $M, \mathrm{Rel}(R), g_1, g_2, \vec{f} \models_s g_1(\vec{x}) = g_2(\vec{x}) \leftrightarrow R\vec{x}$:
  Suppose $g_1(\vec{x}) \neq g_2(\vec{x})$, let $h \in X$.
  Consider $o = h[s/\vec{z}][g_1/u_1][g_2/u_2][\vec{f}/\vec{v}]$:
  $o \in Y_2, M \models_{Y_2} \neg \vec{x} \circledcirc \vec{y}$. So $\forall o' \in Y_2$, $o'(\vec{y}) \neq o(\vec{x})$.
  But for all $h' \in X$, $o' = h'[s/\vec{z}][g_1/u_1][g_2/u_2][\vec{f}/\vec{v}] \in Y_2$;
  then, for all such h',
  $s(\vec{x}) = o(\vec{x}) \neq o'(\vec{y}) = h'(\vec{y})$.
  Therefore, $s(\vec{x}) \notin \mathrm{Rel}(X)$.

□

Definability in Tuple Existence Logic

Equality Generating Dependencies

$$\forall \vec{x}(R\vec{t}_1 \wedge \ldots \wedge R\vec{t}_n \rightarrow t_{n+1} = t_{n+2})$$

Tuple Generating Dependencies

$$\forall \vec{x}(R\vec{t}_1 \wedge \ldots \wedge R\vec{t}_n \rightarrow \exists \vec{y} R\vec{t}')$$

Corollary

All Tuple Generating and Equality Generating Dependencies are expressible in Independence Logic (or in $\mathcal{T}$).

Definability in Tuple Existence Logic

Equality Generating Dependencies

$$\forall \vec{x}(R\vec{t}_1 \wedge \ldots \wedge R\vec{t}_n \rightarrow t_{n+1} = t_{n+2})$$

Tuple Generating Dependencies

$$\forall \vec{x}(R\vec{t}_1 \wedge \ldots \wedge R\vec{t}_n \rightarrow \exists \vec{y} R\vec{t}')$$

Corollary

All Tuple Generating and Equality Generating Dependencies are expressible in Independence Logic (or in $\mathcal{T}$).

Conclusion

- Done:
  - New logic of imperfect information $\mathcal{T}$, based on incl. and excl. dependencies;
  - Characterized Dependence Logic as a fragment of $\mathcal{T}$;
  - Proved than Independence Logic is equivalent to $\mathcal{T}$ (even on open formulas!);
  - Characterized expressive power of $\mathcal{T}$ (and of Independence Logic!) on open formulas.
- To do:
  - Game Semantics?
  - What about "Inclusion Logic" $\mathcal{T}^+$?

Conclusion

- Done:
  - New logic of imperfect information $\mathcal{T}$, based on incl. and excl. dependencies;
  - Characterized Dependence Logic as a fragment of $\mathcal{T}$;
  - Proved than Independence Logic is equivalent to $\mathcal{T}$ (even on open formulas!);
  - Characterized expressive power of $\mathcal{T}$ (and of Independence Logic!) on open formulas.
- To do:
  - Game Semantics?
  - What about "Inclusion Logic" $\mathcal{T}^+$?