The Expressive Power of k-ary Inclusion and Exclusion Atoms

Raine Rönnholm

School of Information Sciences,
University of Tampere, Finland

Inclusion and Exclusion Logics

I denote Inclusion Logic by INC and Exclusion Logic by EXC.
Inclusion-Exclusion Logic is denoted by INEX.

Inclusion and Exclusion Logics

I denote Inclusion Logic by INC and Exclusion Logic by EXC.
Inclusion-Exclusion Logic is denoted by INEX.

Let $\overrightarrow{t_1}, \overrightarrow{t_2}$ be k-tuples of terms.
k-ary inclusion atoms $\overrightarrow{t_1} \subseteq \overrightarrow{t_2}$ have the following truth condition:

$$\mathcal{M} \vDash_X \overrightarrow{t_1} \subseteq \overrightarrow{t_2}, \text{ iff for all } s \in X \text{ there exists } s' \in X, \text{ s.t. } s(\overrightarrow{t_1}) = s'(\overrightarrow{t_2}).$$

Inclusion and Exclusion Logics

I denote Inclusion Logic by INC and Exclusion Logic by EXC.
Inclusion-Exclusion Logic is denoted by INEX.

Let $\overrightarrow{t_1}, \overrightarrow{t_2}$ be k-tuples of terms.
k-ary inclusion atoms $\overrightarrow{t_1} \subseteq \overrightarrow{t_2}$ have the following truth condition:

$$\mathcal{M} \models_X \overrightarrow{t_1} \subseteq \overrightarrow{t_2}, \text{ iff for all } s \in X \text{ there exists } s' \in X, \text{ s.t. } s(\overrightarrow{t_1}) = s'(\overrightarrow{t_2}).$$

k-ary exclusion atoms $\overrightarrow{t_1} \mid \overrightarrow{t_2}$ have the following truth condition:

$$\mathcal{M} \models_X \overrightarrow{t_1} \mid \overrightarrow{t_2}, \text{ iff for all } s, s' \in X \text{ it holds that } s(\overrightarrow{t_1}) \neq s'(\overrightarrow{t_2}).$$

Inclusion and Exclusion Logics

I denote Inclusion Logic by INC and Exclusion Logic by EXC.
Inclusion-Exclusion Logic is denoted by INEX.

Let $\vec{t_1}, \vec{t_2}$ be k-tuples of terms.
k-ary inclusion atoms $\vec{t_1} \subseteq \vec{t_2}$ have the following truth condition:

$$\mathcal{M} \models_X \vec{t_1} \subseteq \vec{t_2}, \text{ iff for all } s \in X \text{ there exists } s' \in X, \text{ s.t. } s(\vec{t_1}) = s'(\vec{t_2}).$$

k-ary exclusion atoms $\vec{t_1} \mid \vec{t_2}$ have the following truth condition:

$$\mathcal{M} \models_X \vec{t_1} \mid \vec{t_2}, \text{ iff for all } s, s' \in X \text{ it holds that } s(\vec{t_1}) \neq s'(\vec{t_2}).$$

I denote Inclusion Logic containing at most k-ary atoms by INC[k]. Similarly I use EXC[k] and INEX[k] for Exclusion Logic and Inclusion-Exclusion Logic.

Dependence and Nondependence Logics

I denote Dependence Logic by DEP and Nondependence Logic by NDEP.

Dependence and Nondependence Logics

I denote Dependence Logic by DEP and Nondependence Logic by NDEP.

k-ary dependence atoms $=\!(t_1, \dots, t_k)$ have the following truth condition:

$$\mathcal{M} \models_X =\!(t_1, \dots, t_k), \text{ iff for all } s, s' \in X \text{ for which}$$
$$s(t_1 \dots t_{k-1}) = s'(t_1 \dots t_{k-1}), \text{ also } s(t_k) = s'(t_k).$$

Dependence and Nondependence Logics

I denote Dependence Logic by DEP and Nondependence Logic by NDEP.

k-ary dependence atoms $=\!(t_1, \ldots, t_k)$ have the following truth condition:

$$\mathcal{M} \models_X =\!(t_1, \ldots, t_k), \text{ iff for all } s, s' \in X \text{ for which}$$
$$s(t_1 \ldots t_{k-1}) = s'(t_1 \ldots t_{k-1}), \text{ also } s(t_k) = s'(t_k).$$

k-ary nondependence atoms $\neq\!(t_1, \ldots, t_k)$ have the following truth condition:

$$\mathcal{M} \models_X \neq\!(t_1, \ldots, t_k), \text{ iff for all } s \in X \text{ there exists an } s' \in X,$$
$$\text{s.t. } s(t_1 \ldots t_{k-1}) = s'(t_1 \ldots t_{k-1}), \text{ but } s(t_k) \neq s'(t_k).$$

Dependence and Nondependence Logics

I denote Dependence Logic by DEP and Nondependence Logic by NDEP.

k-ary dependence atoms $=(t_1, \ldots, t_k)$ have the following truth condition:

$$\mathcal{M} \models_X =(t_1, \ldots, t_k), \text{ iff for all } s, s' \in X \text{ for which}$$
$$s(t_1 \ldots t_{k-1}) = s'(t_1 \ldots t_{k-1}), \text{ also } s(t_k) = s'(t_k).$$

k-ary nondependence atoms $\neq(t_1, \ldots, t_k)$ have the following truth condition:

$$\mathcal{M} \models_X \neq(t_1, \ldots, t_k), \text{ iff for all } s \in X \text{ there exists an } s' \in X,$$
$$\text{s.t. } s(t_1 \ldots t_{k-1}) = s'(t_1 \ldots t_{k-1}), \text{ but } s(t_k) \neq s'(t_k).$$

I use DEP[k] and NDEP[k] for Dependence and Nondependence Logics containing at most k-ary atoms.

Some known facts about the expressive power of logics

Fact 1 (Galliani) On the level of formulas:

EXC ≡ DEP

INC ≡ NDEP

INEX ≡ INDEP (Independence Logic).

Some known facts about the expressive power of logics

Fact 1 (Galliani) On the level of formulas:
EXC $\equiv$ DEP
INC $\equiv$ NDEP
INEX $\equiv$ INDEP (Independence Logic).

Fact 2 (Galliani) On the level of sentences:
DEP[1] $\equiv$ FO $\equiv$ NDEP[1].

Some known facts about the expressive power of logics

Fact 1 (Galliani) On the level of formulas:
EXC $\equiv$ DEP
INC $\equiv$ NDEP
INEX $\equiv$ INDEP (Independence Logic).

Fact 2 (Galliani) On the level of sentences:
DEP[1] $\equiv$ FO $\equiv$ NDEP[1].

Fact 3 (Väänänen) On the level of sentences:
DEP $\equiv$ ESO (Existential Second Order Logic).

Some known facts about the expressive power of logics

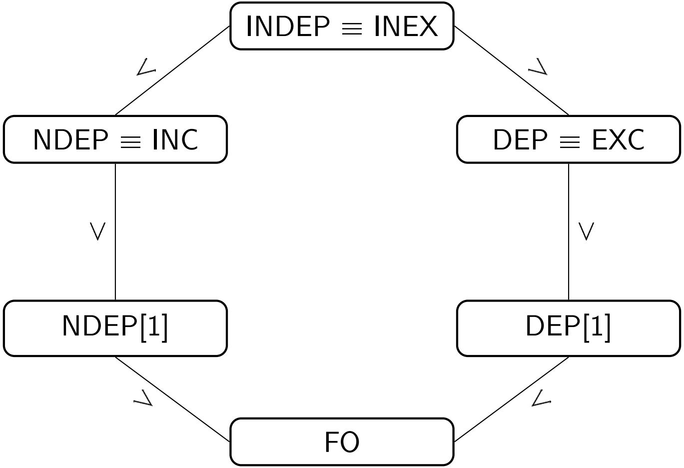
Fact 1 (Galliani) On the level of formulas:
EXC ≡ DEP
INC ≡ NDEP
INEX ≡ INDEP (Independence Logic).

Fact 2 (Galliani) On the level of sentences:
DEP[1] ≡ FO ≡ NDEP[1].

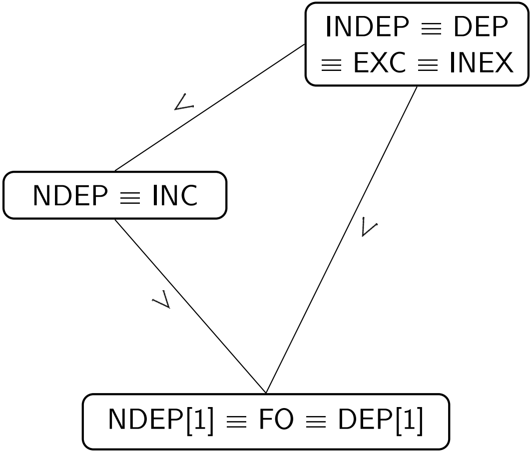
Fact 3 (Väänänen) On the level of sentences:
DEP ≡ ESO (Existential Second Order Logic).

Fact 4 (Galliani and Hella) On the level of sentences:
INC ≡ GFP^+ (Positive Greatest Fixed Point Logic).

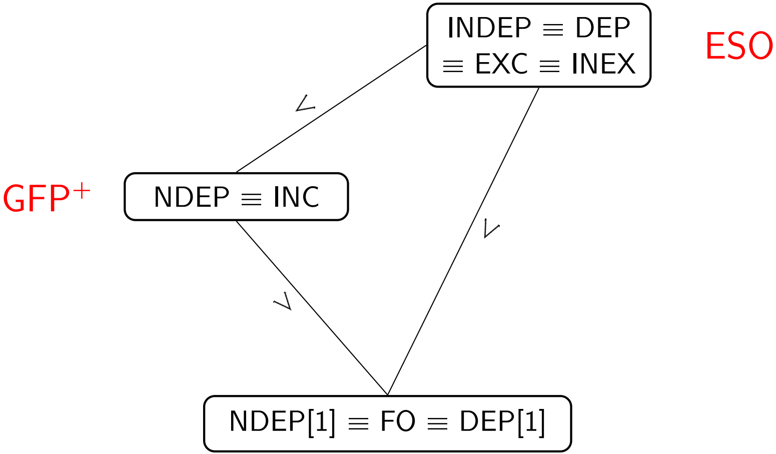
The expressive power of logics on the level of formulas



The expressive power of logics on the level of sentences



The expressive power of logics on the level of sentences



Translation between DEP and EXC

By using translations in Galliani's proof for "DEP $\equiv$ EXC", we get the following results:

Translation between DEP and EXC

By using translations in Galliani's proof for "DEP $\equiv$ EXC",
we get the following results:

k-ary dependence atoms can be expressed in EXC[k]:

$$\mathcal{M} \models_X {=}(t_1, \ldots, t_k), \text{ iff } \mathcal{M} \models_X \forall x\, (x = t_k \lor t_1 \ldots t_{k-1} x \mid t_1 \ldots t_k).$$

Translation between DEP and EXC

By using translations in Galliani's proof for "DEP $\equiv$ EXC",
we get the following results:

k-ary dependence atoms can be expressed in EXC[k]:

$$\mathcal{M} \models_X =\!(t_1, \ldots, t_k), \text{ iff } \mathcal{M} \models_X \forall x \,(x = t_k \;\vee\; t_1 \ldots t_{k-1} x \mid t_1 \ldots t_k).$$

k-ary exclusion atoms can be expressed in DEP[$k+1$]:

$$\mathcal{M} \models_X \vec{t_1} \mid \vec{t_2}, \text{ iff } \mathcal{M} \models_X \forall \vec{y} \,\exists w_1 \,\exists w_2 \Big(=\!(w_1) \wedge =\!(y_1, \ldots, y_k, w_2) \\ \wedge \big((w_1 = w_2 \wedge \vec{y} \neq \vec{t_1}) \vee (w_1 \neq w_2 \wedge \vec{y} \neq \vec{t_2})\big)\Big).$$

Translation between DEP and EXC

By using translations in Galliani's proof for "DEP $\equiv$ EXC", we get the following results:

k-ary dependence atoms can be expressed in EXC[k]:

$$\mathcal{M} \models_X =\!(t_1, \ldots, t_k), \text{ iff } \mathcal{M} \models_X \forall x \, (x = t_k \ \vee \ t_1 \ldots t_{k-1} x \mid t_1 \ldots t_k).$$

k-ary exclusion atoms can be expressed in DEP[$k+1$]:

$$\mathcal{M} \models_X \vec{t_1} \mid \vec{t_2}, \text{ iff } \mathcal{M} \models_X \forall \vec{y} \, \exists w_1 \, \exists w_2 \Big(=\!(w_1) \wedge =\!(y_1, \ldots, y_k, w_2) \wedge \big((w_1 = w_2 \wedge \vec{y} \neq \vec{t_1}) \vee (w_1 \neq w_2 \wedge \vec{y} \neq \vec{t_2})\big)\Big).$$

Corollary DEP[k] $\leqslant$ EXC[k] $\leqslant$ DEP[$k+1$].

Expressing EXC[k] with ESO[k] (k-ary Existential Second Order Logic)

Let φ be EXC[k]-sentence. We label all the instances of exclusion atoms $(\overrightarrow{t_1} \mid \overrightarrow{t_2})_1, \ldots, (\overrightarrow{t_1} \mid \overrightarrow{t_2})_n$ occuring in φ. Now it holds:

$$\mathcal{M} \models \varphi, \text{ iff } \mathcal{M} \models \exists P_1 \ldots \exists P_n \varphi',$$

Expressing EXC[k] with ESO[k] (k-ary Existential Second Order Logic)

Let φ be EXC[k]-sentence. We label all the instances of exclusion atoms $(\vec{t_1} \mid \vec{t_2})_1, \ldots, (\vec{t_1} \mid \vec{t_2})_n$ occuring in φ. Now it holds:

$$\mathcal{M} \models \varphi, \text{ iff } \mathcal{M} \models \exists P_1 \ldots \exists P_n \varphi',$$

where φ' is defined recursively:

$$\begin{aligned}
(\psi)' &= \psi, \text{ if } \psi \text{ is a literal} \\
((\vec{t_1} \mid \vec{t_2})_i)' &= P_i \vec{t_1} \wedge \neg P_i \vec{t_2} \quad \text{for all } i \in \{1, \ldots, n\} \\
(\psi \wedge \theta)' &= \psi' \wedge \theta' \\
(\psi \vee \theta)' &= \psi' \vee \theta' \\
(\exists x \, \psi)' &= \exists x \, \psi' \\
(\forall x \, \psi)' &= \forall x \, \psi'
\end{aligned}$$

Expressing EXC[k] with ESO[k] (k-ary Existential Second Order Logic)

Let φ be EXC[k]-sentence. We label all the instances of exclusion atoms $(\vec{t_1} \mid \vec{t_2})_1, \ldots, (\vec{t_1} \mid \vec{t_2})_n$ occuring in φ. Now it holds:

$$\mathcal{M} \models \varphi, \text{ iff } \mathcal{M} \models \exists P_1 \ldots \exists P_n \varphi',$$

where φ' is defined recursively:

$$\begin{aligned}
(\psi)' &= \psi, \text{ if } \psi \text{ is a literal} \\
((\vec{t_1} \mid \vec{t_2})_i)' &= P_i \vec{t_1} \wedge \neg P_i \vec{t_2} \quad \text{for all } i \in \{1, \ldots, n\} \\
(\psi \wedge \theta)' &= \psi' \wedge \theta' \\
(\psi \vee \theta)' &= \psi' \vee \theta' \\
(\exists x \, \psi)' &= \exists x \, \psi' \\
(\forall x \, \psi)' &= \forall x \, \psi'
\end{aligned}$$

Expressing EXC[k] with ESO[k] (k-ary Existential Second Order Logic)

Let φ be EXC[k]-sentence. We label all the instances of exclusion atoms $(\overrightarrow{t_1} \mid \overrightarrow{t_2})_1, \ldots, (\overrightarrow{t_1} \mid \overrightarrow{t_2})_n$ occuring in φ. Now it holds:

$$\mathcal{M} \models \varphi, \text{ iff } \mathcal{M} \models \exists P_1 \ldots \exists P_n \, \varphi',$$

where φ' is defined recursively:

$$\begin{aligned}
(\psi)' &= \psi, \text{ if } \psi \text{ is a literal} \\
((\overrightarrow{t_1} \mid \overrightarrow{t_2})_i)' &= P_i \overrightarrow{t_1} \wedge \neg P_i \overrightarrow{t_2} \quad \text{for all } i \in \{1, \ldots, n\} \\
(\psi \wedge \theta)' &= \psi' \wedge \theta' \\
(\psi \vee \theta)' &= \psi' \vee \theta' \\
(\exists x \, \psi)' &= \exists x \, \psi' \\
(\forall x \, \psi)' &= \forall x \, \psi'
\end{aligned}$$

Corollary EXC[k] $\leqslant$ ESO[k].

The expressive power of unary exclusion logic

We have shown before that DEP[1] $\leqslant$ EXC[1] $\leqslant$ DEP[2].

The expressive power of unary exclusion logic

We have shown before that $\mathrm{DEP}[1] \leqslant \mathrm{EXC}[1] \leqslant \mathrm{DEP}[2]$.

The following properties of graphs can be defined in $\mathrm{EXC}[1]$:

- Disconnectivity
- k-colorability

The expressive power of unary exclusion logic

We have shown before that $\mathrm{DEP}[1] \leqslant \mathrm{EXC}[1] \leqslant \mathrm{DEP}[2]$.

The following properties of graphs can be defined in $\mathrm{EXC}[1]$:

- Disconnectivity
- k-colorability

Corollary $\mathrm{EXC}[1] \nleqslant \mathrm{DEP}[1]$, and thus $\mathrm{DEP}[1] < \mathrm{EXC}[1]$.

The expressive power of unary exclusion logic

We have shown before that $\mathrm{DEP}[1] \leqslant \mathrm{EXC}[1] \leqslant \mathrm{DEP}[2]$.

The following properties of graphs can be defined in $\mathrm{EXC}[1]$:

- Disconnectivity
- k-colorability

Corollary $\mathrm{EXC}[1] \nleqslant \mathrm{DEP}[1]$, and thus $\mathrm{DEP}[1] < \mathrm{EXC}[1]$.

Väänänen has shown that there are properties that can be expressed in $\mathrm{DEP}[2]$, but cannot be expressed in $\mathrm{ESO}[1]$:

- Infinity of a model
- Even cardinality of a model

The expressive power of unary exclusion logic

We have shown before that $\mathrm{DEP}[1] \leqslant \mathrm{EXC}[1] \leqslant \mathrm{DEP}[2]$.

The following properties of graphs can be defined in EXC[1]:

- Disconnectivity
- k-colorability

Corollary $\mathrm{EXC}[1] \nleqslant \mathrm{DEP}[1]$, and thus $\mathrm{DEP}[1] < \mathrm{EXC}[1]$.

Väänänen has shown that there are properties that can be expressed in DEP[2], but cannot be expressed in ESO[1]:

- Infinity of a model
- Even cardinality of a model

Corollary On the level of sentences $\mathrm{DEP}[2] \nleqslant \mathrm{ESO}[1]$, and thus $\mathrm{EXC}[1] < \mathrm{DEP}[2]$.

The expressive power of unary exclusion logic

We have shown before that $\mathrm{DEP}[1] \leqslant \mathrm{EXC}[1] \leqslant \mathrm{DEP}[2]$.

The following properties of graphs can be defined in $\mathrm{EXC}[1]$:

- Disconnectivity
- k-colorability

Corollary $\mathrm{EXC}[1] \nleqslant \mathrm{DEP}[1]$, and thus $\mathrm{DEP}[1] < \mathrm{EXC}[1]$.

Väänänen has shown that there are properties that can be expressed in $\mathrm{DEP}[2]$, but cannot be expressed in $\mathrm{ESO}[1]$:

- Infinity of a model
- Even cardinality of a model

Corollary On the level of sentences $\mathrm{DEP}[2] \nleqslant \mathrm{ESO}[1]$, and thus $\mathrm{EXC}[1] < \mathrm{DEP}[2]$.

So we have shown that: $\mathrm{DEP}[1] < \mathrm{EXC}[1] < \mathrm{DEP}[2]$.

Translation between NDEP and INC

k-ary nondependence atoms can be expressed in INC$[k]$ (Galliani's translation):

$$\mathcal{M} \models_X \neq(t_1, \ldots, t_k), \text{ iff } \mathcal{M} \models_X \exists x\, (x \neq t_k \wedge t_1 \ldots t_{k-1} x \subseteq t_1 \ldots t_k),$$

Translation between NDEP and INC

k-ary nondependence atoms can be expressed in INC[k] (Galliani's translation):

$$\mathcal{M} \models_X \neq(t_1, \ldots, t_k), \text{ iff } \mathcal{M} \models_X \exists x \, (x \neq t_k \wedge t_1 \ldots t_{k-1} x \subseteq t_1 \ldots t_k),$$

k-ary inclusion atoms can be expressed in NDEP[$k+1$]:

$$\begin{aligned}\mathcal{M} \models_X \vec{t_1} \subseteq \vec{t_2}, \text{ iff } \mathcal{M} \models_X \forall w_1 \forall w_2 \, (w_1 = w_2) \\ \vee \Big(\forall w_1 \forall w_2 \exists \vec{y} \exists z \, (((w_1 = w_2 \wedge \vec{y} = \vec{t_1} \wedge z = y_1) \\ \vee (w_1 \neq w_2 \wedge \vec{y} = \vec{t_2})) \wedge \neq(y_1, \ldots, y_k, z))\Big).\end{aligned}$$

Translation between NDEP and INC

k-ary nondependence atoms can be expressed in INC[k] (Galliani's translation):

$$\mathcal{M} \models_X \neq(t_1, \ldots, t_k), \text{ iff } \mathcal{M} \models_X \exists x \, (x \neq t_k \wedge t_1 \ldots t_{k-1} x \subseteq t_1 \ldots t_k),$$

k-ary inclusion atoms can be expressed in NDEP[$k+1$]:

$$\begin{aligned}\mathcal{M} \models_X \vec{t_1} \subseteq \vec{t_2}, \text{ iff } \mathcal{M} \models_X & \forall w_1 \forall w_2 \, (w_1 = w_2) \\ & \vee \Big(\forall w_1 \forall w_2 \exists \vec{y} \exists z \, (((w_1 = w_2 \wedge \vec{y} = \vec{t_1} \wedge z = y_1) \\ & \quad \vee (w_1 \neq w_2 \wedge \vec{y} = \vec{t_2})) \wedge \neq(y_1, \ldots, y_k, z))\Big).\end{aligned}$$

Corollary NDEP[k] $\leqslant$ INC[k] $\leqslant$ NDEP[$k+1$].

Expressing INC[k] with ESO[k]

Let φ be INC[k]-sentence. We label all the instances of inclusion atoms $(\overrightarrow{t_1} \subseteq \overrightarrow{t_2})_1, \ldots, (\overrightarrow{t_1} \subseteq \overrightarrow{t_2})_n$ occuring in φ. Now it holds:

$$\mathcal{M} \models \varphi, \text{ iff } \mathcal{M} \models \exists P_1 \ldots \exists P_n \left(\varphi' \wedge \bigwedge_{i=1}^{n} \forall \overrightarrow{u} \left(\neg P_i \overrightarrow{u} \vee \varphi_i'(\overrightarrow{u}) \right) \right),$$

Expressing INC[k] with ESO[k]

Let φ be INC[k]-sentence. We label all the instances of inclusion atoms $(\overrightarrow{t_1} \subseteq \overrightarrow{t_2})_1, \ldots, (\overrightarrow{t_1} \subseteq \overrightarrow{t_2})_n$ occuring in φ. Now it holds:

$$\mathcal{M} \models \varphi, \text{ iff } \mathcal{M} \models \exists P_1 \ldots \exists P_n \left(\varphi' \wedge \bigwedge_{i=1}^{n} \forall \overrightarrow{u} \left(\neg P_i \overrightarrow{u} \vee \varphi_i'(\overrightarrow{u}) \right) \right),$$

where φ' and φ_i' ($i \in \{1, \ldots, n\}$) are defined recursively:

$$\begin{aligned}
(\psi)' &= \psi, \text{ if } \psi \text{ is a literal} \\
((\overrightarrow{t_1} \subseteq \overrightarrow{t_2})_i)' &= P_i \overrightarrow{t_1} \text{ for all } i \in \{1, \ldots, n\} \\
(\psi \wedge \theta)' &= \psi' \wedge \theta' \\
(\psi \vee \theta)' &= \psi' \vee \theta' \\
(\exists x\, \psi)' &= \exists x\, \psi' \\
(\forall x\, \psi)' &= \forall x\, \psi'
\end{aligned}$$

Expressing INC[k] with ESO[k]

Let φ be INC[k]-sentence. We label all the instances of inclusion atoms $(\vec{t_1} \subseteq \vec{t_2})_1, \ldots, (\vec{t_1} \subseteq \vec{t_2})_n$ occuring in φ. Now it holds:

$$\mathcal{M} \models \varphi, \text{ iff } \mathcal{M} \models \exists P_1 \ldots \exists P_n \left(\varphi' \wedge \bigwedge_{i=1}^{n} \forall \vec{u} \, (\neg P_i \vec{u} \vee \varphi_i'(\vec{u})) \right),$$

where φ' and φ_i' $(i \in \{1, \ldots, n\})$ are defined recursively:

$$(\psi)' = \psi, \text{ if } \psi \text{ is a literal}$$
$$((\vec{t_1} \subseteq \vec{t_2})_i)' = P_i \vec{t_1} \text{ for all } i \in \{1, \ldots, n\}$$
$$(\psi \wedge \theta)' = \psi' \wedge \theta'$$
$$(\psi \vee \theta)' = \psi' \vee \theta'$$
$$(\exists x \, \psi)' = \exists x \, \psi'$$
$$(\forall x \, \psi)' = \forall x \, \psi'$$

Expressing INC[k] with ESO[k]

where φ' and φ'_i ($i \in \{1, \ldots, n\}$) are defined recursively:

$$
\begin{aligned}
(\psi)'_i &= \psi, \text{ if } \psi \text{ is a literal} \\
((\vec{t_1} \subseteq \vec{t_2})_j)'_i &= P_j \vec{t_1}, \text{ if } j \neq i \\
((\vec{t_1} \subseteq \vec{t_2})_j)'_i &= (\vec{u} = \vec{t_2}) \wedge P_j \vec{t_1}, \text{ if } j = i \\
(\psi \wedge \theta)'_i &= \psi'_i \wedge \theta'_i \\
(\psi \vee \theta)'_i &= \begin{cases} \psi'_i, & \text{if } (t_1 \subseteq t_2)_i \text{ is a subformula of } \psi \\ \theta'_i, & \text{if } (t_1 \subseteq t_2)_i \text{ is a subformula of } \theta \\ \psi'_i \vee \theta'_i, & \text{if } (t_1 \subseteq t_2)_i \text{ is not a subformula of } \psi \vee \theta \end{cases} \\
(\exists x\, \psi)'_i &= \exists x\, \psi'_i \\
(\forall x\, \psi)'_i &= \exists x\, \psi'_i \wedge \forall x\, \psi'
\end{aligned}
$$

Expressing INC[k] with ESO[k]

where φ' and φ'_i ($i \in \{1, \ldots, n\}$) are defined recursively:

$$
\begin{aligned}
(\psi)'_i &= \psi, \text{ if } \psi \text{ is a literal} \\
((\vec{t_1} \subseteq \vec{t_2})_j)'_i &= P_j \vec{t_1}, \text{ if } j \neq i \\
((\vec{t_1} \subseteq \vec{t_2})_j)'_i &= (\vec{u} = \vec{t_2}) \wedge P_j \vec{t_1}, \text{ if } j = i \\
(\psi \wedge \theta)'_i &= \psi'_i \wedge \theta'_i \\
(\psi \vee \theta)'_i &= \begin{cases} \psi'_i, & \text{if } (t_1 \subseteq t_2)_i \text{ is a subformula of } \psi \\ \theta'_i, & \text{if } (t_1 \subseteq t_2)_i \text{ is a subformula of } \theta \\ \psi'_i \vee \theta'_i, & \text{if } (t_1 \subseteq t_2)_i \text{ is not a subformula of } \psi \vee \theta \end{cases} \\
(\exists x \, \psi)'_i &= \exists x \, \psi'_i \\
(\forall x \, \psi)'_i &= \exists x \, \psi'_i \wedge \forall x \, \psi'
\end{aligned}
$$

Expressing INC[k] with ESO[k]

where φ' and φ'_i ($i \in \{1, \dots, n\}$) are defined recursively:

$$
\begin{aligned}
(\psi)'_i &= \psi, \text{ if } \psi \text{ is a literal}\\
((\vec{t_1} \subseteq \vec{t_2})_j)'_i &= P_j \vec{t_1}, \text{ if } j \neq i\\
((\vec{t_1} \subseteq \vec{t_2})_j)'_i &= (\vec{u} = \vec{t_2}) \wedge P_j \vec{t_1}, \text{ if } j = i\\
(\psi \wedge \theta)'_i &= \psi'_i \wedge \theta'_i\\
(\psi \vee \theta)'_i &= \begin{cases} \psi'_i, & \text{if } (t_1 \subseteq t_2)_i \text{ is a subformula of } \psi\\ \theta'_i, & \text{if } (t_1 \subseteq t_2)_i \text{ is a subformula of } \theta\\ \psi'_i \vee \theta'_i, & \text{if } (t_1 \subseteq t_2)_i \text{ is not a subformula of } \psi \vee \theta \end{cases}\\
(\exists x\, \psi)'_i &= \exists x\, \psi'_i\\
(\forall x\, \psi)'_i &= \exists x\, \psi'_i \wedge \forall x\, \psi'
\end{aligned}
$$

Expressing INC[k] with ESO[k]

where φ' and φ'_i ($i \in \{1, \dots, n\}$) are defined recursively:

$$
\begin{aligned}
(\psi)'_i &= \psi, \text{ if } \psi \text{ is a literal}\\
((\vec{t_1} \subseteq \vec{t_2})_j)'_i &= P_j \vec{t_1}, \text{ if } j \neq i\\
((\vec{t_1} \subseteq \vec{t_2})_j)'_i &= (\vec{u} = \vec{t_2}) \wedge P_j \vec{t_1}, \text{ if } j = i\\
(\psi \wedge \theta)'_i &= \psi'_i \wedge \theta'_i\\
(\psi \vee \theta)'_i &= \begin{cases} \psi'_i, & \text{if } (t_1 \subseteq t_2)_i \text{ is a subformula of } \psi\\ \theta'_i, & \text{if } (t_1 \subseteq t_2)_i \text{ is a subformula of } \theta\\ \psi'_i \vee \theta'_i, & \text{if } (t_1 \subseteq t_2)_i \text{ is not a subformula of } \psi \vee \theta \end{cases}\\
(\exists x\, \psi)'_i &= \exists x\, \psi'_i\\
(\forall x\, \psi)'_i &= \exists x\, \psi'_i \wedge \forall x\, \psi'
\end{aligned}
$$

Corollary INC[k] $\leqslant$ ESO[k].

The expressive power of unary inclusion logic

We have shown before that $\mathrm{NDEP}[1] \leqslant \mathrm{INC}[1] \leqslant \mathrm{NDEP}[2]$.

The expressive power of unary inclusion logic

We have shown before that NDEP[1] $\leqslant$ INC[1] $\leqslant$ NDEP[2].

Galliani and Hella have noted that the following property of graphs can be defined in INC[1]:

- Finite directed graph contains a cycle

The expressive power of unary inclusion logic

We have shown before that NDEP[1] $\leqslant$ INC[1] $\leqslant$ NDEP[2].

Galliani and Hella have noted that the following property of graphs can be defined in INC[1]:

- Finite directed graph contains a cycle

Corollary INC[1] $\nleqslant$ NDEP[1], and thus NDEP[1] $<$ INC[1].

The expressive power of unary inclusion logic

We have shown before that NDEP[1] $\leqslant$ INC[1] $\leqslant$ NDEP[2].

Galliani and Hella have noted that the following property of graphs can be defined in INC[1]:

- Finite directed graph contains a cycle

Corollary INC[1] $\nleqslant$ NDEP[1], and thus NDEP[1] $<$ INC[1].

By using the Ehrenfeucht-Fraïssé game for the inclusion logic (presented by Galliani and Hella) it can be shown that there exists no INC[1]-formula φ which would be equivalent with the NDEP[2]-formula $\neq(x, y)$.

The expressive power of unary inclusion logic

We have shown before that NDEP[1] $\leqslant$ INC[1] $\leqslant$ NDEP[2].

Galliani and Hella have noted that the following property of graphs can be defined in INC[1]:

- Finite directed graph contains a cycle

Corollary INC[1] $\nleqslant$ NDEP[1], and thus NDEP[1] $<$ INC[1].

By using the Ehrenfeucht-Fraïssé game for the inclusion logic (presented by Galliani and Hella) it can be shown that there exists no INC[1]-formula φ which would be equivalent with the NDEP[2]-formula $\neq(x, y)$.

Corollary NDEP[2] $\nleqslant$ INC[1], and thus INC[1] $<$ NDEP[2].

The expressive power of unary inclusion logic

We have shown before that $\mathrm{NDEP}[1] \leqslant \mathrm{INC}[1] \leqslant \mathrm{NDEP}[2]$.

Galliani and Hella have noted that the following property of graphs can be defined in $\mathrm{INC}[1]$:

- Finite directed graph contains a cycle

Corollary $\mathrm{INC}[1] \nleqslant \mathrm{NDEP}[1]$, and thus $\mathrm{NDEP}[1] < \mathrm{INC}[1]$.

By using the Ehrenfeucht-Fraïssé game for the inclusion logic (presented by Galliani and Hella) it can be shown that there exists no $\mathrm{INC}[1]$-formula φ which would be equivalent with the $\mathrm{NDEP}[2]$-formula $\neq(x, y)$.

Corollary $\mathrm{NDEP}[2] \nleqslant \mathrm{INC}[1]$, and thus $\mathrm{INC}[1] < \mathrm{NDEP}[2]$.

So we have shown that: $\mathrm{NDEP}[1] < \mathrm{INC}[1] < \mathrm{NDEP}[2]$.

Expressing INEX[k] with ESO[k]

Let φ be INEX[k]-sentence. We label all the instances of exclusion atoms $(\vec{t_1} \mid \vec{t_2})_1, \ldots, (\vec{t_1} \mid \vec{t_2})_n$ occuring in φ, and define φ':

$$
\begin{aligned}
(\psi)' &= \psi, \text{ if } \psi \text{ is a literal} \\
((\vec{t_1} \mid \vec{t_2})_i)' &= P_i \vec{t_1} \wedge \neg P_i \vec{t_2} \text{ for all } i \in \{1, \ldots, n\} \\
(\vec{t_1} \subseteq \vec{t_2})' &= \vec{t_1} \subseteq \vec{t_2} \\
(\psi \wedge \theta)' &= \psi' \wedge \theta' \\
(\psi \vee \theta)' &= \psi' \vee \theta' \\
(\exists x\, \psi)' &= \exists x\, \psi' \\
(\forall x\, \psi)' &= \forall x\, \psi'
\end{aligned}
$$

Expressing INEX[k] with ESO[k]

Let φ be INEX[k]-sentence. We label all the instances of exclusion atoms $(\vec{t_1} \mid \vec{t_2})_1, \ldots, (\vec{t_1} \mid \vec{t_2})_n$ occuring in φ, and define φ':

$$
\begin{aligned}
(\psi)' &= \psi, \text{ if } \psi \text{ is a literal} \\
((\vec{t_1} \mid \vec{t_2})_i)' &= P_i \vec{t_1} \wedge \neg P_i \vec{t_2} \ \text{ for all } i \in \{1, \ldots, n\} \\
(\vec{t_1} \subseteq \vec{t_2})' &= \vec{t_1} \subseteq \vec{t_2} \\
(\psi \wedge \theta)' &= \psi' \wedge \theta' \\
(\psi \vee \theta)' &= \psi' \vee \theta' \\
(\exists x\, \psi)' &= \exists x\, \psi' \\
(\forall x\, \psi)' &= \forall x\, \psi'
\end{aligned}
$$

Expressing INEX[k] with ESO[k]

Let φ be INEX[k]-sentence. We label all the instances of exclusion atoms $(\vec{t_1} \mid \vec{t_2})_1, \ldots, (\vec{t_1} \mid \vec{t_2})_n$ occuring in φ, and define φ':

$$
\begin{aligned}
(\psi)' &= \psi, \text{ if } \psi \text{ is a literal} \\
((\vec{t_1} \mid \vec{t_2})_i)' &= P_i \vec{t_1} \wedge \neg P_i \vec{t_2} \quad \text{for all } i \in \{1, \ldots, n\} \\
(\vec{t_1} \subseteq \vec{t_2})' &= \vec{t_1} \subseteq \vec{t_2} \\
(\psi \wedge \theta)' &= \psi' \wedge \theta' \\
(\psi \vee \theta)' &= \psi' \vee \theta' \\
(\exists x\, \psi)' &= \exists x\, \psi' \\
(\forall x\, \psi)' &= \forall x\, \psi'
\end{aligned}
$$

Because φ' is an INC[k]-sentence, it is equivalent with some ESO[k]-sentence μ. Now it holds: $\mathcal{M} \models \varphi$, iff $\mathcal{M} \models \exists P_1 \ldots \exists P_n\, \mu$.

Expressing INEX[k] with ESO[k]

Let φ be INEX[k]-sentence. We label all the instances of exclusion atoms $(\vec{t_1} \mid \vec{t_2})_1, \ldots, (\vec{t_1} \mid \vec{t_2})_n$ occuring in φ, and define φ':

$$
\begin{aligned}
(\psi)' &= \psi, \text{ if } \psi \text{ is a literal} \\
((\vec{t_1} \mid \vec{t_2})_i)' &= P_i \vec{t_1} \wedge \neg P_i \vec{t_2} \text{ for all } i \in \{1, \ldots, n\} \\
(\vec{t_1} \subseteq \vec{t_2})' &= \vec{t_1} \subseteq \vec{t_2} \\
(\psi \wedge \theta)' &= \psi' \wedge \theta' \\
(\psi \vee \theta)' &= \psi' \vee \theta' \\
(\exists x \, \psi)' &= \exists x \, \psi' \\
(\forall x \, \psi)' &= \forall x \, \psi'
\end{aligned}
$$

Because φ' is an INC[k]-sentence, it is equivalent with some ESO[k]-sentence μ. Now it holds: $\mathcal{M} \models \varphi$, iff $\mathcal{M} \models \exists P_1 \ldots \exists P_n \, \mu$.

Corollary INEX[k] $\leqslant$ ESO[k].

Term value preserving disjunction

Let $\overrightarrow{t_1}, \ldots, \overrightarrow{t_n}$ be k-tuples of terms.

The following connective can be expressed in INEX[k]:

$\mathcal{M} \models_X \varphi \vee_{\overrightarrow{t_1} \ldots \overrightarrow{t_n}} \psi$, iff there exists $Y, Y' \subseteq X$ s.t.

$$Y \cup Y' = X,\ \mathcal{M} \models_Y \varphi \text{ and } \mathcal{M} \models_{Y'} \psi,$$
$$\text{and if } Y, Y' \neq \emptyset,\ \text{then } Y(\overrightarrow{t_i}) = Y'(\overrightarrow{t_i}) = X(\overrightarrow{t_i}) \quad \text{for all } i \in \{1, \ldots, n\}.$$

Term value preserving disjunction

Expressing term value preserving disjunction in INEX[k]:

$$\varphi \vee_{\vec{t_1}\ldots\vec{t_n}} \psi := (\varphi \sqcup \psi) \sqcup \exists\, y_1 \exists\, y_2 \Big({=}(y_1) \wedge {=}(y_2) \wedge y_1 \neq y_2$$

$$\wedge \exists\, x \Big(\bigwedge_{i=1}^{n} (\theta_i \sqcup \theta_i') \wedge \big((x = y_1 \wedge \varphi) \vee (x = y_2 \wedge \psi)\big)\Big)\Big),$$

where for each $i \in \{1, \ldots, n\}$:

$$\theta_i := (x = y_1 \wedge \forall\, \vec{z}\, (\vec{z} \subseteq \vec{t_i})) \vee (x = y_2 \wedge \forall\, \vec{z}\, (\vec{z} \subseteq \vec{t_i}))$$

$$\theta_i' := \exists\, \vec{u}\, \Big(\vec{u} \mid \vec{t_i} \wedge \exists\, \vec{w_1} \exists\, \vec{w_2} \Big(\big((x = y_1 \wedge \vec{w_1} = \vec{t_i} \wedge \vec{w_2} = \vec{u})$$

$$\vee (x = y_2 \wedge \vec{w_1} = \vec{u} \wedge \vec{w_2} = \vec{t_i})\big) \wedge \vec{t_i} \subseteq \vec{w_1} \wedge \vec{t_i} \subseteq \vec{w_2}\Big)\Big)$$

Term value preserving disjunction

Expressing term value preserving disjunction in INEX[k]:

$$\varphi \vee_{\vec{t_1}\ldots\vec{t_n}} \psi := (\varphi \sqcup \psi) \sqcup \exists y_1 \exists y_2 \big(=\!(y_1) \wedge =\!(y_2) \wedge y_1 \neq y_2$$
$$\wedge \exists x \big(\bigwedge_{i=1}^{n} (\theta_i \sqcup \theta_i') \wedge \big((x = y_1 \wedge \varphi) \vee (x = y_2 \wedge \psi)\big)\big)\big),$$

where for each $i \in \{1, \ldots, n\}$:

$$\theta_i := (x = y_1 \wedge \forall \vec{z} \, (\vec{z} \subseteq \vec{t_i})) \vee (x = y_2 \wedge \forall \vec{z} \, (\vec{z} \subseteq \vec{t_i}))$$
$$\theta_i' := \exists \vec{u} \, \big(\vec{u} \mid \vec{t_i} \wedge \exists \vec{w_1} \exists \vec{w_2} \big(\big((x = y_1 \wedge \vec{w_1} = \vec{t_i} \wedge \vec{w_2} = \vec{u})$$
$$\vee (x = y_2 \wedge \vec{w_1} = \vec{u} \wedge \vec{w_2} = \vec{t_i})\big) \wedge \vec{t_i} \subseteq \vec{w_1} \wedge \vec{t_i} \subseteq \vec{w_2}\big)\big)$$

(Connective $\sqcup$ is the *intuitionistic disjunction* which can be expressed in DEP[1])

Expressing ESO[k] with INEX[k]

Let $\exists\, P_1 \ldots \exists\, P_n\, \varphi$ be ESO[k]-sentence that is satisfied, iff it is satisfied with non-empty interpretations for $P_1, \ldots, P_n$. Now it holds:

$$\mathcal{M} \models \exists\, P_1 \ldots \exists\, P_n\, \varphi, \text{ iff } \exists\, \overrightarrow{w_1} \ldots \exists\, \overrightarrow{w_n}\, \varphi',$$

Expressing ESO[k] with INEX[k]

Let $\exists P_1 \ldots \exists P_n \varphi$ be ESO[k]-sentence that is satisfied, iff it is satisfied with non-empty interpretations for $P_1, \ldots, P_n$. Now it holds:

$$\mathcal{M} \models \exists P_1 \ldots \exists P_n \varphi, \text{ iff } \exists \overrightarrow{w_1} \ldots \exists \overrightarrow{w_n} \varphi',$$

where φ' is defined recursively:

$$\begin{aligned}
\psi' &= \psi, \text{ if } \psi \text{ is literal and } P_i \text{ does not occur in } \psi \\
(P_i \overrightarrow{t})' &= \overrightarrow{t} \subseteq \overrightarrow{w_i} \\
(\neg P_i \overrightarrow{t})' &= \overrightarrow{t} \mid \overrightarrow{w_i} \\
(\psi \wedge \theta)' &= \psi' \wedge \theta' \\
(\psi \vee \theta)' &= \psi' \vee_{\overrightarrow{w_1} \ldots \overrightarrow{w_n}} \theta' \\
(\exists x \, \psi)' &= \exists x \, \psi' \\
(\forall x \, \psi)' &= \forall x \, \psi'
\end{aligned}$$

Expressing ESO[k] with INEX[k]

Let $\exists P_1 \ldots \exists P_n \varphi$ be ESO[k]-sentence that is satisfied, iff it is satisfied with non-empty interpretations for $P_1, \ldots, P_n$. Now it holds:

$$\mathcal{M} \models \exists P_1 \ldots \exists P_n \varphi, \text{ iff } \exists \overrightarrow{w_1} \ldots \exists \overrightarrow{w_n} \varphi',$$

where φ' is defined recursively:

$$\begin{aligned}
\psi' &= \psi, \text{ if } \psi \text{ is literal and } P_i \text{ does not occur in } \psi \\
(P_i \overrightarrow{t})' &= \overrightarrow{t} \subseteq \overrightarrow{w_i} \\
(\neg P_i \overrightarrow{t})' &= \overrightarrow{t} \mid \overrightarrow{w_i} \\
(\psi \wedge \theta)' &= \psi' \wedge \theta' \\
(\psi \vee \theta)' &= \psi' \vee_{\overrightarrow{w_1} \ldots \overrightarrow{w_n}} \theta' \\
(\exists x \, \psi)' &= \exists x \, \psi' \\
(\forall x \, \psi)' &= \forall x \, \psi'
\end{aligned}$$

Expressing ESO[k] with INEX[k]

Let $\exists P_1 \ldots \exists P_n \varphi$ be ESO[k]-sentence that is satisfied, iff it is satisfied with non-empty interpretations for $P_1, \ldots, P_n$. Now it holds:

$$\mathcal{M} \models \exists P_1 \ldots \exists P_n \varphi, \text{ iff } \exists \overrightarrow{w_1} \ldots \exists \overrightarrow{w_n} \varphi',$$

where φ' is defined recursively:

$$\begin{aligned}
\psi' &= \psi, \text{ if } \psi \text{ is literal and } P_i \text{ does not occur in } \psi \\
(P_i \vec{t})' &= \vec{t} \subseteq \overrightarrow{w_i} \\
(\neg P_i \vec{t})' &= \vec{t} \mid \overrightarrow{w_i} \\
(\psi \wedge \theta)' &= \psi' \wedge \theta' \\
(\psi \vee \theta)' &= \psi' \vee_{\overrightarrow{w_1} \ldots \overrightarrow{w_n}} \theta' \\
(\exists x \, \psi)' &= \exists x \, \psi' \\
(\forall x \, \psi)' &= \forall x \, \psi'
\end{aligned}$$

Expressing ESO[k] with INEX[k]

Let $\exists P_1 \ldots \exists P_n \varphi$ be ESO[k]-sentence that is satisfied, iff it is satisfied with non-empty interpretations for $P_1, \ldots, P_n$. Now it holds:

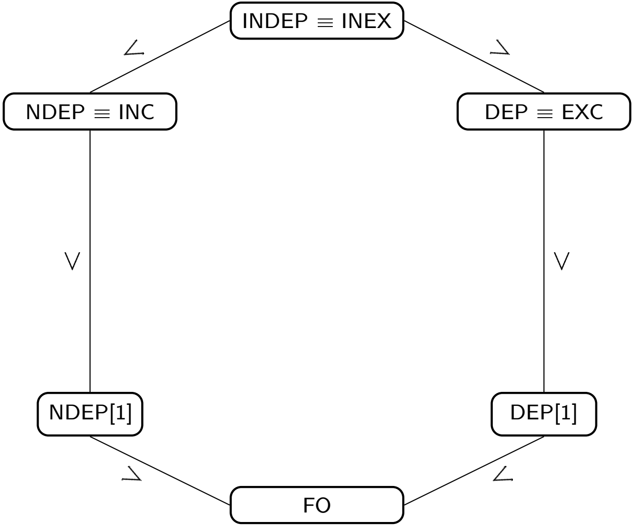
$$\mathcal{M} \models \exists P_1 \ldots \exists P_n \varphi, \text{ iff } \exists \overrightarrow{w_1} \ldots \exists \overrightarrow{w_n} \varphi',$$

where φ' is defined recursively:

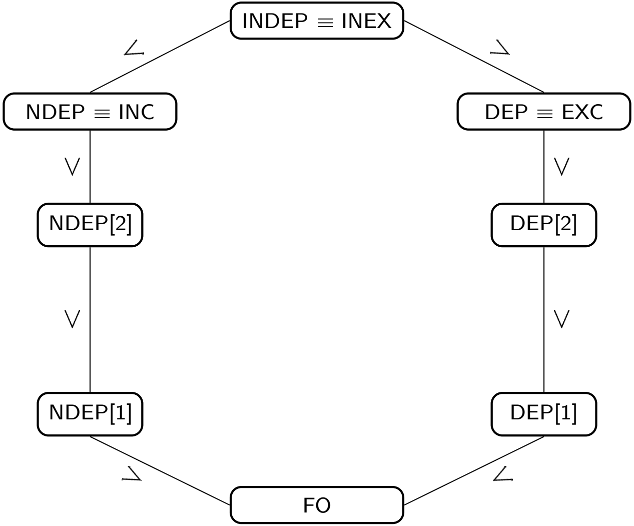
$$\begin{aligned}
\psi' &= \psi, \text{ if } \psi \text{ is literal and } P_i \text{ does not occur in } \psi \\
(P_i \vec{t})' &= \vec{t} \subseteq \overrightarrow{w_i} \\
(\neg P_i \vec{t})' &= \vec{t} \mid \overrightarrow{w_i} \\
(\psi \wedge \theta)' &= \psi' \wedge \theta' \\
(\psi \vee \theta)' &= \psi' \vee_{\overrightarrow{w_1} \ldots \overrightarrow{w_n}} \theta' \\
(\exists x \, \psi)' &= \exists x \, \psi' \\
(\forall x \, \psi)' &= \forall x \, \psi'
\end{aligned}$$

Corollary ESO[k] $\leqslant$ INEX[k], and thus INEX[k] $\equiv$ ESO[k].

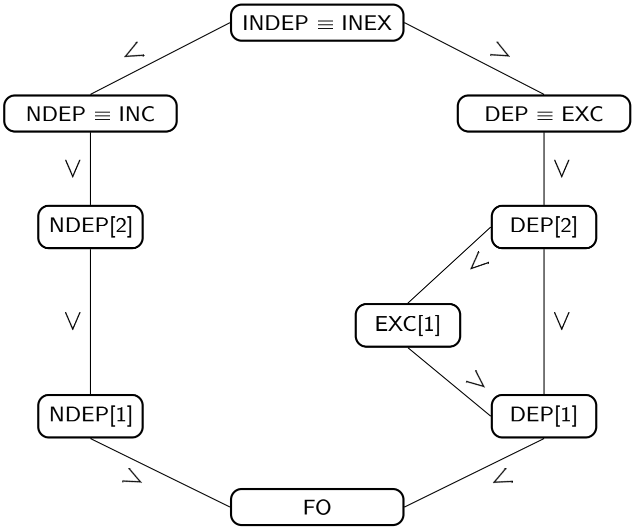
The expressive power of logics on the level of formulas



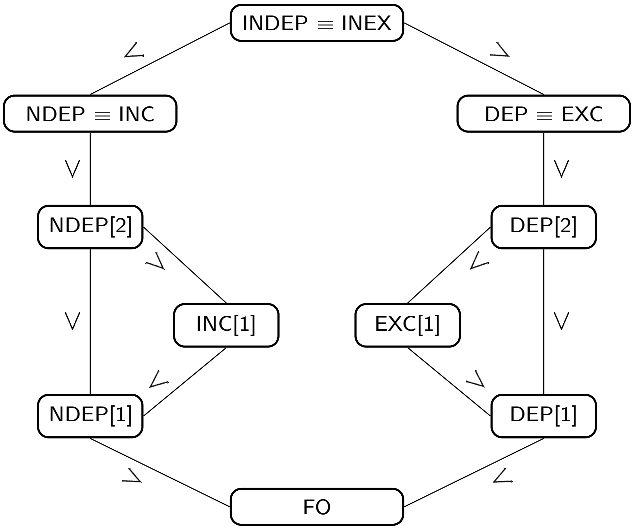
The expressive power of logics on the level of formulas



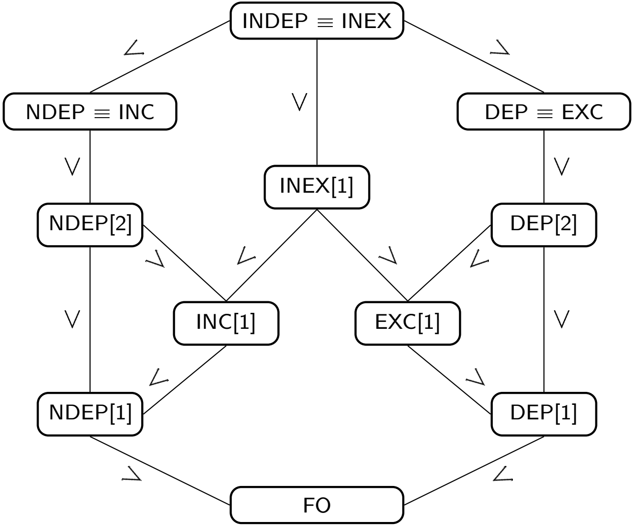
The expressive power of logics on the level of formulas



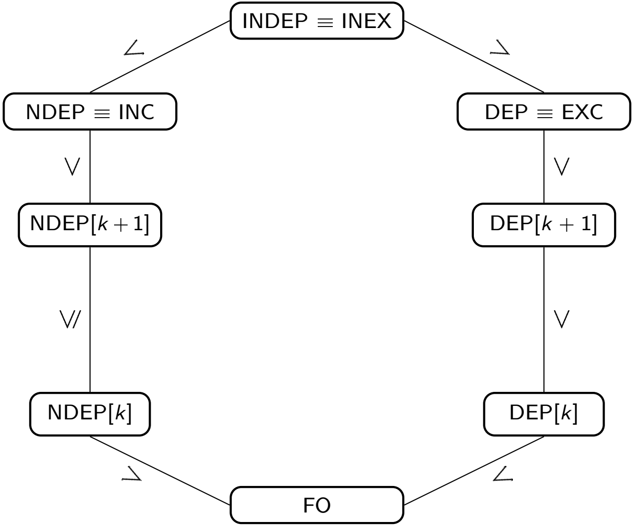
The expressive power of logics on the level of formulas



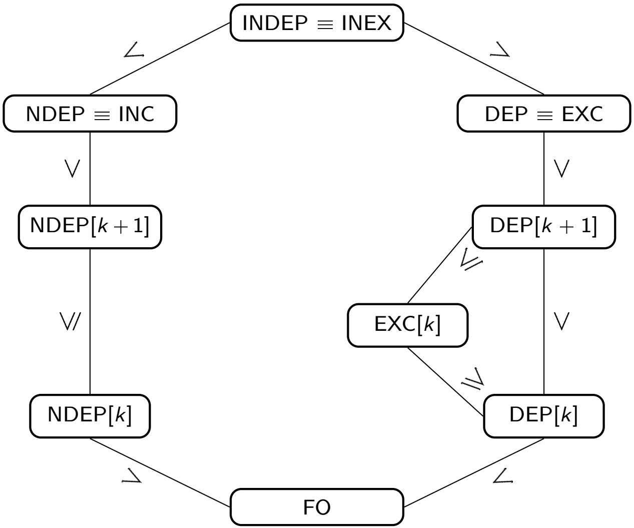
The expressive power of logics on the level of formulas



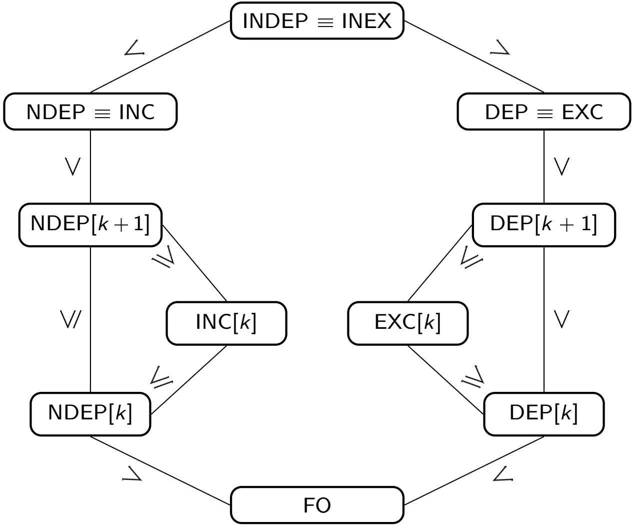
The expressive power of logics on the level of formulas



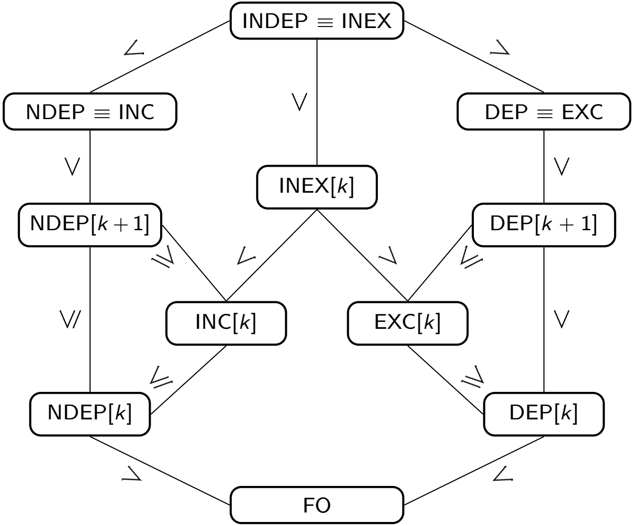
The expressive power of logics on the level of formulas



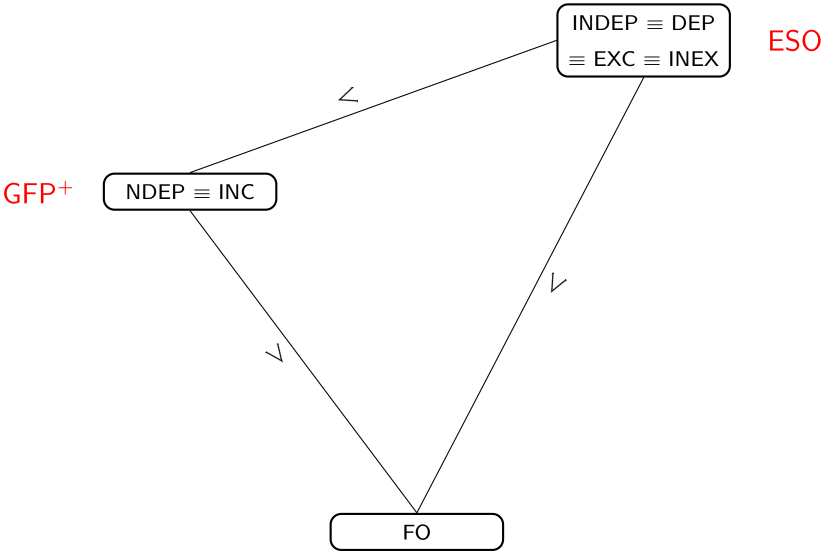
The expressive power of logics on the level of formulas



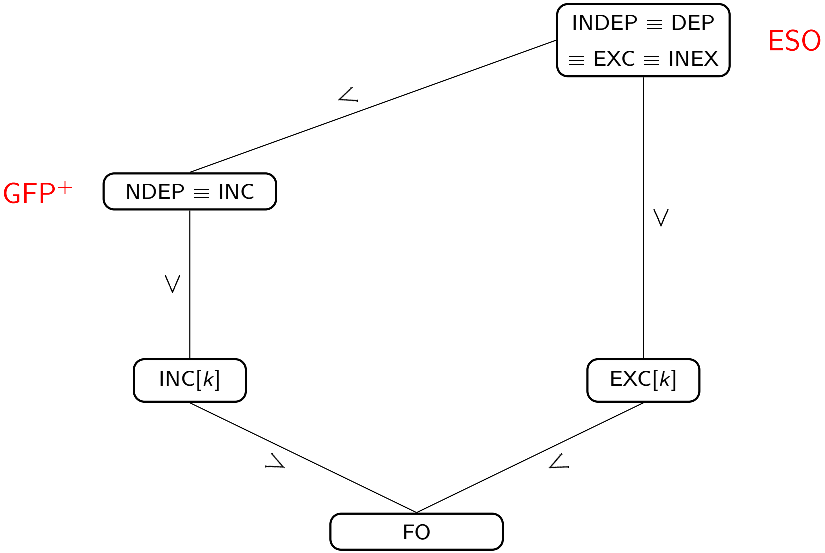
The expressive power of logics on the level of formulas



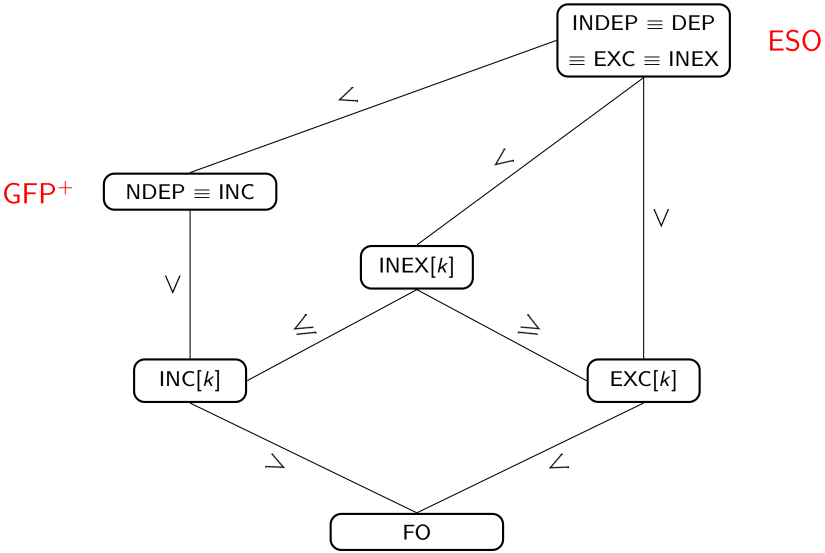
The expressive power of logics on the level of sentences



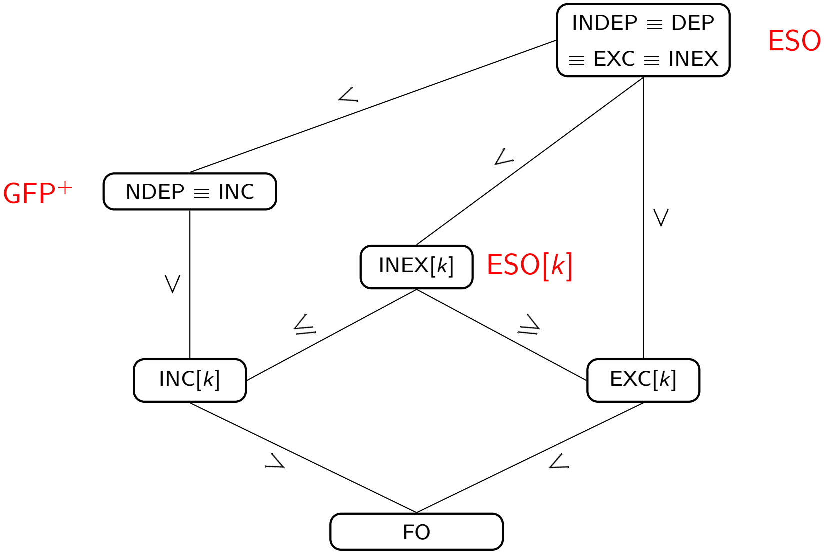
The expressive power of logics on the level of sentences



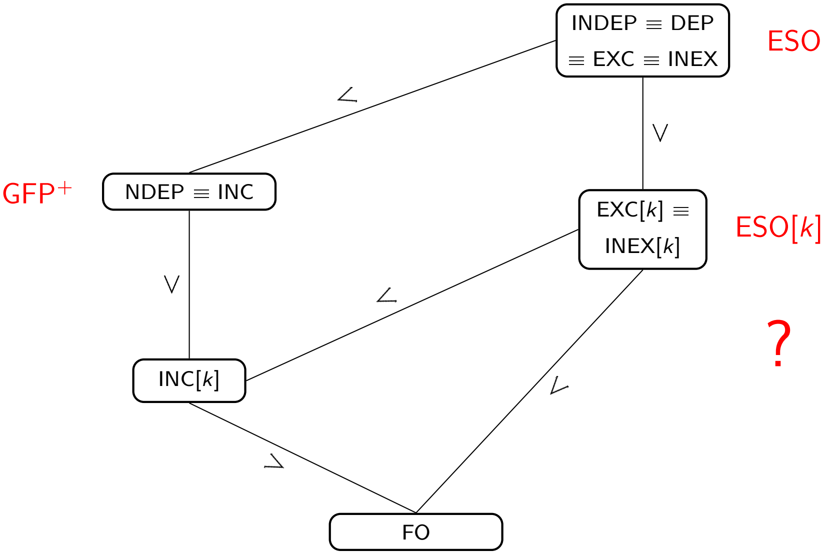
The expressive power of logics on the level of sentences



The expressive power of logics on the level of sentences



The expressive power of logics on the level of sentences



Open questions and further recearch

- On the level of sentences does INC[1] $<$ NDEP[2]?

Open questions and further recearch

- On the level of sentences does INC[1] < NDEP[2]?
- Does it hold for all $k \in \mathbb{Z}_+$, that

$$\begin{cases} \text{DEP[k]} < \text{EXC[k]} < \text{DEP[k+1]} \\ \text{NDEP[k]} < \text{INC[k]} < \text{NDEP[k+1]} \end{cases}$$

Open questions and further recearch

- On the level of sentences does INC[1] $<$ NDEP[2]?
- Does it hold for all $k \in \mathbb{Z}_+$, that

$$\begin{cases} \text{DEP[k]} < \text{EXC[k]} < \text{DEP[k+1]} \\ \text{NDEP[k]} < \text{INC[k]} < \text{NDEP[k+1]} \end{cases}$$

- If the latter is true, is it true on the level of sentences?

Open questions and further recearch

- On the level of sentences does INC[1] $<$ NDEP[2]?
- Does it hold for all $k \in \mathbb{Z}_+$, that

$$\begin{cases} \text{DEP[k]} < \text{EXC[k]} < \text{DEP[k+1]} \\ \text{NDEP[k]} < \text{INC[k]} < \text{NDEP[k+1]} \end{cases}$$

- If the latter is true, is it true on the level of sentences?
- What kind of fragments of ESO[k] are INC[k] and EXC[k]?

Appendix: Some operators expressible in EXC[1]

The following operators can be expressed in EXC[1]:

Appendix: Some operators expressible in EXC[1]

The following operators can be expressed in EXC[1]:

Universal quantification over the values of a given term:

$$\mathcal{M} \models_X (\forall x \subseteq t)\, \varphi, \quad \text{iff } \mathcal{M} \models_{X[X(t)/x]} \varphi.$$

Appendix: Some operators expressible in EXC[1]

The following operators can be expressed in EXC[1]:

Universal quantification over the values of a given term:

$$\mathcal{M} \models_X (\forall x \subseteq t)\, \varphi, \quad \text{iff } \mathcal{M} \models_{X[X(t)/x]} \varphi.$$

Existential quantification over the complement of the values of a given set of terms:

$$\mathcal{M} \models_X (\exists x \mid \bigcup_{i=1}^{n} t_i)\, \varphi,$$

$$\text{iff there exists } f : X \to \overline{\bigcup_{i=1}^{n} X(t_i)}, \text{ s.t. } \mathcal{M} \models_{X[f/x]} \varphi.$$

Appendix: Some properties of graphs expressible in EXC[1]

Undirected graph $\mathcal{G} = (V, E)$ is disconnected, iff

$$\mathcal{G} \models \forall x \, \exists y_1 (\exists y_2 \mid y_1) \big((x = y_1 \lor x = y_2) \\ \land (\forall z_1 \subseteq y_1)(\forall z_2 \subseteq y_2) \neg E z_1 z_2 \big).$$

Appendix: Some properties of graphs expressible in EXC[1]

Undirected graph $\mathcal{G} = (V, E)$ is disconnected, iff

$$\mathcal{G} \models \forall x \exists y_1 (\exists y_2 \mid y_1) \big((x = y_1 \vee x = y_2) \wedge (\forall z_1 \subseteq y_1)(\forall z_2 \subseteq y_2) \neg E z_1 z_2 \big).$$

Undirected graph $\mathcal{G} = (V, E)$ is k-colorable, iff

$$\mathcal{G} \models \forall x \exists y_1 (\exists y_2 \mid y_1) \dots (\exists y_k \mid y_1 \cup \dots \cup y_{k-1}) \Big(\bigvee_{i=1}^{k} x = y_i \wedge \bigwedge_{i=1}^{k} (\forall z_1 \subseteq y_i)(\forall z_2 \subseteq y_i) \neg E z_1 z_2 \Big).$$

Appendix: A NDEP[2]-formula which is not expressible in INC[1]

Let $L = \emptyset$ and $\mathcal{M}$ be an L-model, s.t. $\text{dom}(\mathcal{M}) = \{0, 1\}$.
Let $X = \{s_1, s_2\}$ and $Y = \{s_1, s_2, s_3\}$, where

$$\begin{aligned} s_1 &= \{(x, 0), (y, 0)\} \\ s_2 &= \{(x, 0), (y, 1)\} \\ s_3 &= \{(x, 1), (y, 1)\} \end{aligned}$$

Appendix: A NDEP[2]-formula which is not expressible in INC[1]

Let $L = \emptyset$ and $\mathcal{M}$ be an L-model, s.t. $\mathrm{dom}(\mathcal{M}) = \{0, 1\}$.
Let $X = \{s_1, s_2\}$ and $Y = \{s_1, s_2, s_3\}$, where

$$\begin{aligned} s_1 &= \{(x, 0), (y, 0)\} \\ s_2 &= \{(x, 0), (y, 1)\} \\ s_3 &= \{(x, 1), (y, 1)\} \end{aligned}$$

Clearly now $\mathcal{M} \models_X \neq\!(x, y)$, but $\mathcal{M} \not\models_Y \neq\!(x, y)$.

Appendix: A NDEP[2]-formula which is not expressible in INC[1]

Let $L = \emptyset$ and $\mathcal{M}$ be an L-model, s.t. $\mathrm{dom}(\mathcal{M}) = \{0, 1\}$.
Let $X = \{s_1, s_2\}$ and $Y = \{s_1, s_2, s_3\}$, where

$$s_1 = \{(x, 0), (y, 0)\}$$
$$s_2 = \{(x, 0), (y, 1)\}$$
$$s_3 = \{(x, 1), (y, 1)\}$$

Clearly now $\mathcal{M} \models_X \neq(x, y)$, but $\mathcal{M} \not\models_Y \neq(x, y)$.

By using the Ehrenfeucht-Fraïssé game for inclusion logic we can show that there exists no INC[1]-formula φ so that

$$\mathcal{M} \models_X \varphi, \text{ but } \mathcal{M} \not\models_Y \varphi.$$