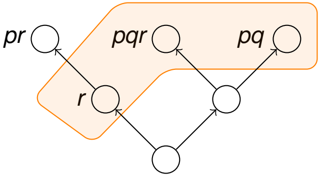
# Team-based logics around BSML

Fan Yang
Utrecht University

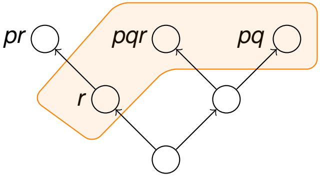
The NihiL workshop
Jan. 31 - Feb. 2, 2024

# Another motivation for state-based or team-based semantics

- Bilateral state-based modal logic (BSML): Modal logic + NE
- States: sets of possible worlds.

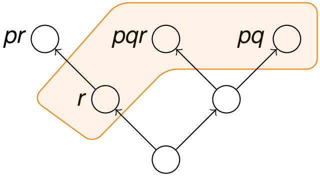


- Bilateral state-based modal logic (BSML): Modal logic + NE
- States: sets of possible worlds. Also known as Teams.



# Another motivation for state-based or team-based semantics

- Bilateral state-based modal logic (BSML): Modal logic + NE
- States: sets of possible worlds. Also known as Teams.



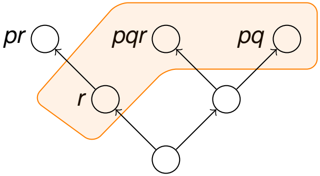

Team semantics:

- A compositional semantics introduced by Hodges (1997) for characterizing dependencies between variables,

  originally for Independence-friendly Logic (Hintikka, Sandu 1989), and later developed further in Dependence Logic (Väänänen 2007).
- Adopted also independently in Inquisitive Logic (Ciardelli, Roelofsen 2011), (Ciardelli, Groenendijk, Roelofsen 2018) for characterizing questions in natural language.

# Another motivation for state-based or team-based semantics

- Bilateral state-based modal logic (BSML): Modal logic + NE
- States: sets of possible worlds. Also known as Teams.



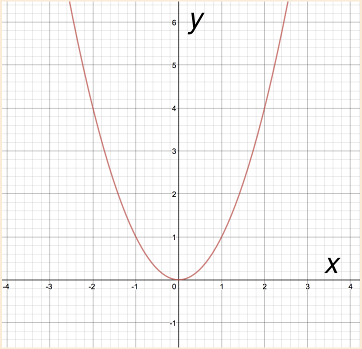

Team semantics:

- A compositional semantics introduced by Hodges (1997) for characterizing dependencies between variables,

  originally for Independence-friendly Logic (Hintikka, Sandu 1989), and later developed further in Dependence Logic (Väänänen 2007).
- Adopted also independently in Inquisitive Logic (Ciardelli, Roelofsen 2011), (Ciardelli, Groenendijk, Roelofsen 2018) for characterizing questions in natural language.

# functional dependence between variables

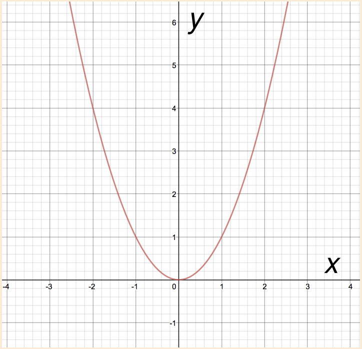
$$y = f(x) = x^2$$




- $x$ determines $y$

# How to characterize functional dependence between variables?

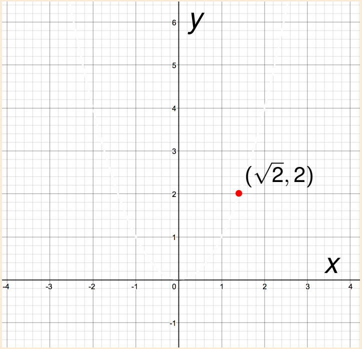
$$y = f(x) = x^2$$




| | $x$ | $y$ | $z$ |
|---|---|---|---|
| $s$ | $\sqrt{2}$ | 2 | 0 |

- $\mathbb{R} \models_s x$ `determines` $y$ iff ??

$$y = f(x) = x^2$$

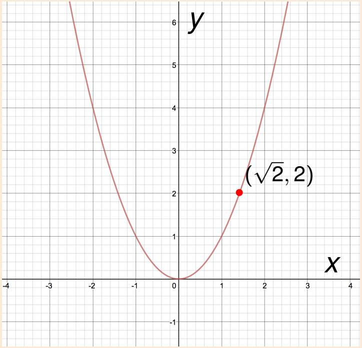



| | $x$ | $y$ | $z$ |
|---|---|---|---|
| $s$ | $\sqrt{2}$ | 2 | 0 |

- $\mathbb{R} \models_s x$ `determines` $y$ iff ??

# How to characterize functional dependence between variables?

$$y = f(x) = x^2$$

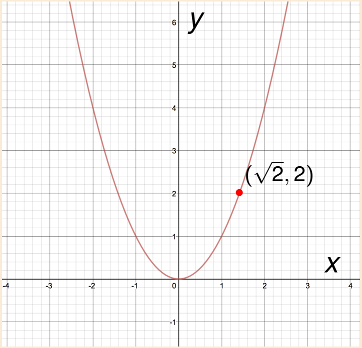



| | $x$ | $y$ | $z$ |
|---|---|---|---|
| $s_0$ | $\sqrt{2}$ | 2 | 0 |
| $s_1$ | $\sqrt{2}$ | 2 | 1 |
| $s_2$ | $-2$ | 4 | $\sqrt{2}$ |
| $s_3$ | $-2$ | 4 | 2 |
| $s_4$ | $-\sqrt{2}$ | 2 | 0 |

- $\mathbb{R} \models_s x \ \texttt{determines} \ y$ iff ??
- A team : a set of assignments

# How to characterize functional dependence between variables?

$$y = f(x) = x^2$$

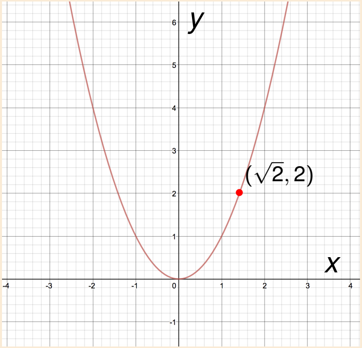



| | $x$ | $y$ | $z$ |
|---|---|---|---|
| $s_0$ | $\sqrt{2}$ | 2 | 0 |
| $s_1$ | $\sqrt{2}$ | 2 | 1 |
| $s_2$ | $-2$ | 4 | $\sqrt{2}$ |
| $s_3$ | $-2$ | 4 | 2 |
| $s_4$ | $-\sqrt{2}$ | 2 | 0 |

- $\mathbb{R} \models_t x \ \texttt{determines}\ y$ iff for all $s, s' \in t$: $$s(x) = s'(x) \Longrightarrow s(y) = s'(y)$$
- A team $t$: a set of assignments

# How to characterize functional dependence between variables?

$$y = f(x) = x^2$$

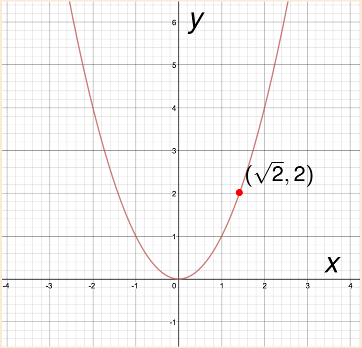



| | $x$ | $y$ | $z$ |
|---|---|---|---|
| $s_0$ | $\sqrt{2}$ | 2 | 0 |
| $s_1$ | $\sqrt{2}$ | 2 | 1 |
| $s_2$ | $-2$ | 4 | $\sqrt{2}$ |
| $s_3$ | $-2$ | 4 | 2 |
| $s_4$ | $-\sqrt{2}$ | 2 | 0 |

- $\mathbb{R} \models_t x \texttt{ determines } y$ iff for all $s, s' \in t$:
  $$s(x) = s'(x) \Longrightarrow s(y) = s'(y)$$
- A team $t$: a set of assignments

# How to characterize functional dependence between variables?

$$y = f(x) = x^2$$

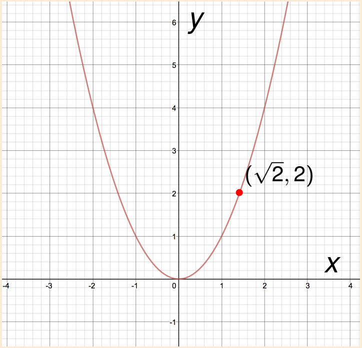



| | $x$ | $y$ | $z$ |
|---|---|---|---|
| $s_0$ | $\sqrt{2}$ | 2 | 0 |
| $s_1$ | $\sqrt{2}$ | 2 | 1 |
| $s_2$ | $-2$ | 4 | $\sqrt{2}$ |
| $s_3$ | $-2$ | 4 | 2 |
| $s_4$ | $-\sqrt{2}$ | 2 | 0 |

- $\mathbb{R} \models_t x \ \texttt{determines}\ y$ iff for all $s, s' \in t$: $s(x) = s'(x) \Longrightarrow s(y) = s'(y)$
- A team $t$: a set of assignments

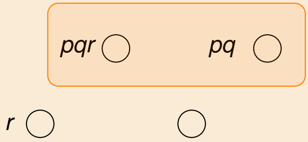
# How to characterize functional dependence between variables?

$$y = f(x) = x^2$$




| | $x$ | $y$ | $z$ |
|---|---|---|---|
| $s_0$ | $\sqrt{2}$ | 2 | 0 |
| $s_1$ | $\sqrt{2}$ | 2 | 1 |
| $s_2$ | $-2$ | 4 | $\sqrt{2}$ |
| $s_3$ | $-2$ | 4 | 2 |
| $s_4$ | $-\sqrt{2}$ | 2 | 0 |

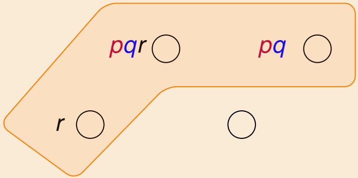
- $\mathbb{R} \models_t x \ \mathtt{determines}\ y$ iff for all $s, s' \in t$: $s(x) = s'(x) \Longrightarrow s(y) = s'(y)$
- A team $t$: a set of assignments




- A team $t$: a set of valuations / possible worlds
- $t \models p \texttt{ determines } q$ iff for all $u, v \in t$:

$$u(p) = v(p) \Longrightarrow u(q) = v(q)$$

– "Whether it is raining determines whether I will take my umbrella."
– "Whether this set is empty determines whether it is raining in Amsterdam."

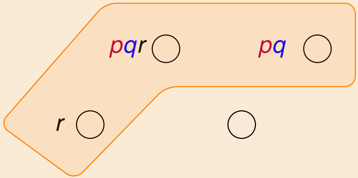



- A team $t$: a set of valuations / possible worlds
- $t \models p$ $\texttt{determines}$ $q$ iff for all $u, v \in t$:
$$u(p) = v(p) \Longrightarrow u(q) = v(q)$$

– "Whether it is raining determines whether I will take my umbrella."

– "Whether this set is empty determines whether it is raining in Amsterdam."

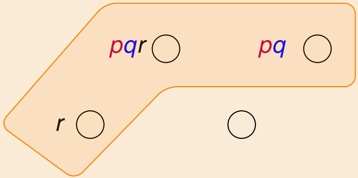
# Propositional dependence atoms




- A team $t$: a set of valuations / possible worlds
- $t \models =(p, q)$ iff for all $u, v \in t$:

$$u(p) = v(p) \Longrightarrow u(q) = v(q)$$

– "Whether it is raining determines whether I will take my umbrella."
– "Whether this set is empty determines whether it is raining in Amsterdam."



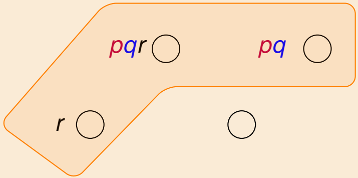

- A team $t$: a set of valuations / possible worlds
- $t \models =(\vec{p}, \vec{q})$ iff for all $u, v \in t$:

$$u(\vec{p}) = v(\vec{p}) \Longrightarrow u(\vec{q}) = v(\vec{q})$$

– "Whether it is raining determines whether I will take my umbrella."

– "Whether this set is empty determines whether it is raining in Amsterdam."

# Propositional dependence atoms




- A team $t$: a set of valuations / possible worlds
- $t \models =(\vec{p}, \vec{q})$ iff for all $u, v \in t$:

$$u(\vec{p}) = v(\vec{p}) \Longrightarrow u(\vec{q}) = v(\vec{q})$$

– "Whether it is raining determines whether I will take my umbrella."
– "Whether this set is empty determines whether it is raining in Amsterdam."

Empty team/state property: $\emptyset \models =(\vec{p}, \vec{q})$

| | $p$ | $q$ | $r$ | $s$ |
|---|---|---|---|---|
| $v_0$ | 1 | 0 | 1 | 1 |
| $v_1$ | 1 | 0 | 1 | 0 |
| $v_2$ | 0 | 1 | 0 | 1 |
| $v_3$ | 0 | 1 | 0 | 0 |

- A team can be viewed as a relational database.
- Dependence atoms $=(\vec{p}, \vec{q})$ correspond exactly to functional dependencies $\vec{p} \to \vec{q}$ in database theory
- Armstrong's Axioms (1974) for functional dependencies:
  - $=(\vec{p}, \vec{p})$ (identity)
  - $=(\vec{p}\vec{q}, \vec{r})$ implies $=(\vec{q}\vec{p}, \vec{r})$ (commutativity)
  - $=(\vec{p}\vec{p}, \vec{q})$ implies $=(\vec{p}, \vec{q})$ (contraction)
  - $=(\vec{q}, \vec{r})$ implies $=(\vec{p}\vec{q}, \vec{r})$ (weakening)
  - $=(\vec{p}, \vec{q})$ and $=(\vec{q}, \vec{r})$ imply $=(\vec{p}, \vec{r})$ (transitivity)

| | $p$ | $q$ | $r$ | $s$ |
|---|---|---|---|---|
| $v_0$ | 1 | 0 | 1 | 1 |
| $v_1$ | 1 | 0 | 1 | 0 |
| $v_2$ | 0 | 1 | 0 | 1 |
| $v_3$ | 0 | 1 | 0 | 0 |

- A team can be viewed as a relational database.
- Dependence atoms $=\!(\vec{p}, \vec{q})$ correspond exactly to functional dependencies $\vec{p} \to \vec{q}$ in database theory
- Armstrong's Axioms (1974) for functional dependencies:
  - $=\!(\vec{p}, \vec{p})$ (identity)
  - $=\!(\vec{p}\vec{q}, \vec{r})$ implies $=\!(\vec{q}\vec{p}, \vec{r})$ (commutativity)
  - $=\!(\vec{p}\vec{p}, \vec{q})$ implies $=\!(\vec{p}, \vec{q})$ (contraction)
  - $=\!(\vec{q}, \vec{r})$ implies $=\!(\vec{p}\vec{q}, \vec{r})$ (weakening)
  - $=\!(\vec{p}, \vec{q})$ and $=\!(\vec{q}, \vec{r})$ imply $=\!(\vec{p}, \vec{r})$ (transitivity)

# Connection with database theory

| | $p$ | $q$ | $r$ | $s$ |
|---|---|---|---|---|
| $v_0$ | 1 | 0 | 1 | 1 |
| $v_1$ | 1 | 0 | 1 | 0 |
| $v_2$ | 0 | 1 | 0 | 1 |
| $v_3$ | 0 | 1 | 0 | 0 |

- A team can be viewed as a relational database.
- Dependence atoms $=\!(\vec{p}, \vec{q})$ correspond exactly to functional dependencies $\vec{p} \to \vec{q}$ in database theory
- Armstrong's Axioms (1974) for functional dependencies:
  - $=\!(\vec{p}, \vec{p})$ (identity)
  - $=\!(\vec{p}\vec{q}, \vec{r})$ implies $=\!(\vec{q}\vec{p}, \vec{r})$ (commutativity)
  - $=\!(\vec{p}\vec{p}, \vec{q})$ implies $=\!(\vec{p}, \vec{q})$ (contraction)
  - $=\!(\vec{q}, \vec{r})$ implies $=\!(\vec{p}\vec{q}, \vec{r})$ (weakening)
  - $=\!(\vec{p}, \vec{q})$ and $=\!(\vec{q}, \vec{r})$ imply $=\!(\vec{p}, \vec{r})$ (transitivity)

| | $p$ | $q$ | $r$ | $s$ |
|---|---|---|---|---|
| $v_0$ | 1 | 0 | 1 | 1 |
| $v_1$ | 1 | 0 | 1 | 0 |
| $v_2$ | 0 | 1 | 0 | 1 |
| $v_3$ | 0 | 1 | 0 | 0 |

- A team can be viewed as a relational database.
- Dependence atoms $=(\vec{p}, \vec{q})$ correspond exactly to functional dependencies $\vec{p} \to \vec{q}$ in database theory
- Armstrong's Axioms (1974) for functional dependencies:
  - $=(\vec{p}, \vec{p})$ (identity)
  - $=(\vec{p}\vec{q}, \vec{r})$ implies $=(\vec{q}\vec{p}, \vec{r})$ (commutativity)
  - $=(\vec{p}\vec{p}, \vec{q})$ implies $=(\vec{p}, \vec{q})$ (contraction)
  - $=(\vec{q}, \vec{r})$ implies $=(\vec{p}\vec{q}, \vec{r})$ (weakening)
  - $=(\vec{p}, \vec{q})$ and $=(\vec{q}, \vec{r})$ imply $=(\vec{p}, \vec{r})$ (transitivity)

| | $p$ | $q$ | $r$ | $s$ |
|---|---|---|---|---|
| $v_0$ | 1 | 0 | 1 | 1 |
| $v_1$ | 1 | 0 | 1 | 0 |
| $v_2$ | 0 | 1 | 0 | 1 |
| $v_3$ | 0 | 1 | 0 | 0 |

- A team can be viewed as a relational database.
- Dependence atoms $=(\vec{p}, \vec{q})$ correspond exactly to functional dependencies $\vec{p} \to \vec{q}$ in database theory
- Armstrong's Axioms (1974) for functional dependencies:
  - $=(\vec{p}, \vec{p})$ (identity)
  - $=(\vec{p}\vec{q}, \vec{r})$ implies $=(\vec{q}\vec{p}, \vec{r})$ (commutativity)
  - $=(\vec{p}\vec{p}, \vec{q})$ implies $=(\vec{p}, \vec{q})$ (contraction)
  - $=(\vec{q}, \vec{r})$ implies $=(\vec{p}\vec{q}, \vec{r})$ (weakening)
  - $=(\vec{p}, \vec{q})$ and $=(\vec{q}, \vec{r})$ imply $=(\vec{p}, \vec{r})$ (transitivity)

# Propositional team-based logics around BSML

Language: $\phi ::= p \mid \neg p \mid \bot \mid \phi \wedge \phi \mid \phi \vee \phi \mid \phi \to \phi \mid =(\vec{p}, \vec{q})$

Language: $\phi ::= p \mid \neg p \mid \bot \mid \phi \wedge \phi \mid \phi \vee \phi \mid \phi \to \phi \mid \phi \mathbin{\mathbb{V}} \phi \mid {=}(\vec{p}, \vec{q})$

Global (or inquisitive) disjunction

Language: $\phi ::= p \mid \neg p \mid \bot \mid \phi \wedge \phi \mid \phi \vee \phi \mid \phi \to \phi \mid \phi \mathbin{\vee\!\!\!\vee} \phi \mid {=}(\vec{p}, \vec{q})$

Local disjunction Global (or inquisitive) disjunction

Language: $\phi ::= p \mid \neg p \mid \bot \mid \phi \wedge \phi \mid \phi \vee \phi \mid \phi \to \phi \mid \phi \mathbin{\vee\!\!\!\vee} \phi \mid =\!(\vec{p}, \vec{q})$

Local disjunction Global (or inquisitive) disjunction

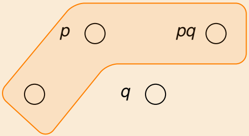
Recall: Propositional BSML or BSPL: $\phi ::= p \mid \neg\phi \mid \phi \wedge \phi \mid \phi \vee \phi \mid \mathrm{NE}$

**Language:** $\phi ::= p \mid \neg p \mid \bot \mid \phi \wedge \phi \mid \phi \vee \phi \mid \phi \to \phi \mid \phi \mathbin{\vee\!\!\!\vee} \phi \mid {=}(\vec{p}, \vec{q})$

Local disjunction    Global (or inquisitive) disjunction

**Team semantics:** Let $t \subseteq 2^{\mathrm{Prop}}$ be a team/state, i.e., a set of valuations.

- $t \models p$ iff $v(p) = 1$ for all $v \in t$
- $t \models \neg p$ iff $v(p) = 0$ for all $v \in t$
- $t \models \phi \mathbin{\vee\!\!\!\vee} \psi$ iff $t \models \phi$ or $t \models \psi$
- $t \models \phi \vee \psi$ iff $\exists s, r \subseteq t$ s.t. $t = s \cup r$, $s \models \phi$ and $r \models \psi$
- $t \models \bot$ iff $t =$




$t \not\models p$
$t \not\models \neg p$

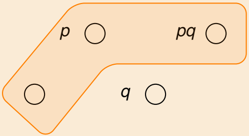
**Recall:** Propositional BSML or BSPL: $\phi ::= p \mid \neg\phi \mid \phi \wedge \phi \mid \phi \vee \phi \mid \mathrm{NE}$

Language: $\phi ::= p \mid \neg p \mid \bot \mid \phi \wedge \phi \mid \phi \vee \phi \mid \phi \to \phi \mid \phi \mathbin{\mathbb{V}} \phi \mid =\!(\vec{p}, \vec{q})$

Local disjunction — Global (or inquisitive) disjunction

Team semantics: Let $t \subseteq 2^{\mathrm{Prop}}$ be a team/state, i.e., a set of valuations.

- $t \models p$ iff $v(p) = 1$ for all $v \in t$
- $t \models \neg p$ iff $v(p) = 0$ for all $v \in t$
- $t \models \phi \mathbin{\mathbb{V}} \psi$ iff $t \models \phi$ or $t \models \psi$
- $t \models \phi \vee \psi$ iff $\exists s, r \subseteq t$ s.t. $t = s \cup r$, $s \models \phi$ and $r \models \psi$
- $t \models \bot$ iff $t =$




Recall: Propositional BSML or BSPL: $\phi ::= p \mid \neg \phi \mid \phi \wedge \phi \mid \phi \vee \phi \mid \mathrm{NE}$

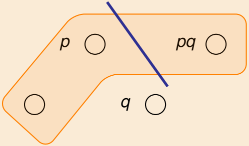
# Propositional team-based logics around BSML

Language: $\phi ::= p \mid \neg p \mid \bot \mid \phi \wedge \phi \mid \phi \vee \phi \mid \phi \to \phi \mid \phi \mathbin{\vee\!\!\!\vee} \phi \mid =\!(\vec{p}, \vec{q})$

Local disjunction    Global (or inquisitive) disjunction

Team semantics: Let $t \subseteq 2^{\mathrm{Prop}}$ be a team/state, i.e., a set of valuations.

- $t \models p$ iff $v(p) = 1$ for all $v \in t$
- $t \models \neg p$ iff $v(p) = 0$ for all $v \in t$
- $t \models \phi \mathbin{\vee\!\!\!\vee} \psi$ iff $t \models \phi$ or $t \models \psi$
- $t \models \phi \vee \psi$ iff $\exists s, r \subseteq t$ s.t. $t = s \cup r$, $s \models \phi$ and $r \models \psi$
- $t \models \bot$ iff $t =$



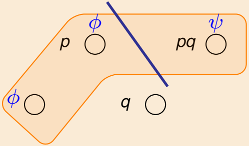

Recall: Propositional BSML or BSPL: $\phi ::= p \mid \neg\phi \mid \phi \wedge \phi \mid \phi \vee \phi \mid \mathrm{NE}$

**Language:** $\phi ::= p \mid \neg p \mid \bot \mid \phi \wedge \phi \mid \phi \vee \phi \mid \phi \to \phi \mid \phi \mathbin{\mathbb{V}} \phi \mid =(\vec{p}, \vec{q})$

Local disjunction — Global (or inquisitive) disjunction

**Team semantics:** Let $t \subseteq 2^{\text{Prop}}$ be a team/state, i.e., a set of valuations.

- $t \models p$ iff $v(p) = 1$ for all $v \in t$
- $t \models \neg p$ iff $v(p) = 0$ for all $v \in t$
- $t \models \phi \mathbin{\mathbb{V}} \psi$ iff $t \models \phi$ or $t \models \psi$
- $t \models \phi \vee \psi$ iff $\exists s, r \subseteq t$ s.t. $t = s \cup r$, $s \models \phi$ and $r \models \psi$
- $t \models \bot$ iff $t =$



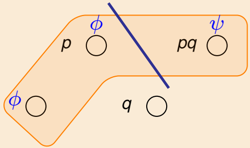

**Recall:** Propositional BSML or BSPL: $\phi ::= p \mid \neg\phi \mid \phi \wedge \phi \mid \phi \vee \phi \mid \text{NE}$

Language: $\phi ::= p \mid \neg p \mid \bot \mid \phi \wedge \phi \mid \phi \vee \phi \mid \phi \to \phi \mid \phi \mathbin{\vee\!\!\!\vee} \phi \mid =\!(\vec{p}, \vec{q})$

Local disjunction — Global (or inquisitive) disjunction

Team semantics: Let $t \subseteq 2^{\mathrm{Prop}}$ be a team/state, i.e., a set of valuations.

- $t \models p$ iff $v(p) = 1$ for all $v \in t$
- $t \models \neg p$ iff $v(p) = 0$ for all $v \in t$
- $t \models \phi \mathbin{\vee\!\!\!\vee} \psi$ iff $t \models \phi$ or $t \models \psi$
- $t \models \phi \vee \psi$ iff $\exists s, r \subseteq t$ s.t. $t = s \cup r$, $s \models \phi$ and $r \models \psi$
- $t \models \bot$ iff $t = \emptyset$




Empty Team Property: $\emptyset \models \psi$ for all $\psi$

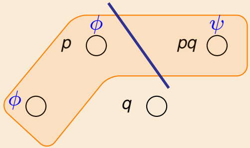
Recall: Propositional BSML or BSPL: $\phi ::= p \mid \neg\phi \mid \phi \wedge \phi \mid \phi \vee \phi \mid \mathrm{NE}$

Language: $\phi ::= p \mid \neg p \mid \bot \mid \phi \wedge \phi \mid \phi \vee \phi \mid \phi \to \phi \mid \phi \mathbin{\mathbb{V}} \phi \mid {=}(\vec{p}, \vec{q})$

Local disjunction Global (or inquisitive) disjunction

Team semantics: Let $t \subseteq 2^{\text{Prop}}$ be a team/state, i.e., a set of valuations.

- $t \models p$ iff $v(p) = 1$ for all $v \in t$
- $t \models \neg p$ iff $v(p) = 0$ for all $v \in t$
- $t \models \phi \mathbin{\mathbb{V}} \psi$ iff $t \models \phi$ or $t \models \psi$
- $t \models \phi \vee \psi$ iff $\exists s, r \subseteq t$ s.t. $t = s \cup r$, $s \models \phi$ and $r \models \psi$
- $t \models \bot$ iff $t = \emptyset$




Empty Team Property: $\emptyset \models \psi$ for all $\psi$

Recall: Propositional BSML or BSPL: $\phi ::= p \mid \neg\phi \mid \phi \wedge \phi \mid \phi \vee \phi \mid \mathrm{NE}$

- $t \models \mathrm{NE}$ iff $t \neq \emptyset$

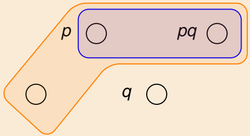
# Propositional team-based logics around BSML

Language: $\phi ::= p \mid \neg p \mid \bot \mid \phi \wedge \phi \mid \phi \vee \phi \mid \phi \to \phi \mid \phi \varovee \phi \mid =(\vec{p}, \vec{q})$

Local disjunction ($\vee$)    Global (or inquisitive) disjunction ($\varovee$)

Team semantics: Let $t \subseteq 2^{\text{Prop}}$ be a team/state, i.e., a set of valuations.

- $t \models p$ iff $v(p) = 1$ for all $v \in t$
- $t \models \neg p$ iff $v(p) = 0$ for all $v \in t$
- $t \models \phi \varovee \psi$ iff $t \models \phi$ or $t \models \psi$
- $t \models \phi \vee \psi$ iff $\exists s, r \subseteq t$ s.t. $t = s \cup r$, $s \models \phi$ and $r \models \psi$
- $t \models \bot$ iff $t = \emptyset$




Empty Team Property: $\emptyset \models \psi$ for all $\psi$

Downward Closure: If $s \subseteq t \models \phi$, then $s \models \phi$.

Recall: Propositional BSML or BSPL: $\phi ::= p \mid \neg\phi \mid \phi \wedge \phi \mid \phi \vee \phi \mid \text{NE}$

- $t \models \text{NE}$ iff $t \neq \emptyset$

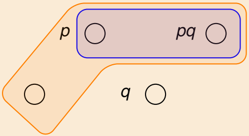
# Propositional team-based logics around BSML

Language: $\phi ::= p \mid \neg p \mid \bot \mid \phi \wedge \phi \mid \phi \vee \phi \mid \phi \to \phi \mid \phi \mathbin{\mathbb{V}} \phi \mid {=}(\vec{p}, \vec{q})$

Local disjunction    Global (or inquisitive) disjunction

Team semantics: Let $t \subseteq 2^{\text{Prop}}$ be a team/state, i.e., a set of valuations.

- $t \models p$ iff $v(p) = 1$ for all $v \in t$
- $t \models \neg p$ iff $v(p) = 0$ for all $v \in t$
- $t \models \phi \mathbin{\mathbb{V}} \psi$ iff $t \models \phi$ or $t \models \psi$
- $t \models \phi \vee \psi$ iff $\exists s, r \subseteq t$ s.t. $t = s \cup r$, $s \models \phi$ and $r \models \psi$
- $t \models \bot$ iff $t = \emptyset$




Empty Team Property: $\emptyset \models \psi$ for all $\psi$

Downward Closure: If $s \subseteq t \models \phi$, then $s \models \phi$.

Recall: Propositional BSML or BSPL: $\phi ::= p \mid \neg\phi \mid \phi \wedge \phi \mid \phi \vee \phi \mid \text{NE}$

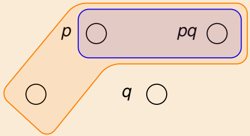
- $t \models \text{NE}$ iff $t \neq \emptyset$
- $\{v\} \models \text{NE}$, whereas $\emptyset \not\models \text{NE}$

Language: $\phi ::= p \mid \neg p \mid \bot \mid \phi \wedge \phi \mid \phi \vee \phi \mid \phi \to \phi \mid \phi \mathbin{\vee\!\!\!\vee} \phi \mid {=}(\vec{p}, \vec{q})$

Local disjunction  Global (or inquisitive) disjunction

Team semantics: Let $t \subseteq 2^{\mathsf{Prop}}$ be a team/state, i.e., a set of valuations.

- $t \models p$ iff $v(p) = 1$ for all $v \in t$
- $t \models \neg p$ iff $v(p) = 0$ for all $v \in t$
- $t \models \phi \mathbin{\vee\!\!\!\vee} \psi$ iff $t \models \phi$ or $t \models \psi$
- $t \models \phi \vee \psi$ iff $\exists s, r \subseteq t$ s.t. $t = s \cup r$, $s \models \phi$ and $r \models \psi$
- $t \models \bot$ iff $t = \emptyset$

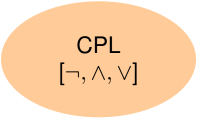



Empty Team Property: $\emptyset \models \psi$ for all $\psi$
Downward Closure: If $s \subseteq t \models \phi$, then $s \models \phi$.

Recall: Propositional BSML or BSPL: $\phi ::= p \mid \neg\phi \mid \phi \wedge \phi \mid \phi \vee \phi \mid \mathrm{NE}$

- $t \models \mathrm{NE}$ iff $t \neq \emptyset$
- $\{v\} \models \mathrm{NE}$, whereas $\emptyset \not\models \mathrm{NE}$
- Union Closure: $t \models \phi$ and $s \models \phi$ imply $t \cup s \models \phi$ for $\phi \in [\neg, \bot, \wedge, \vee, \to, \mathrm{NE}]$

# Propositional team-based logics around BSML



Downward closed

CPL
$[\neg, \wedge, \vee]$

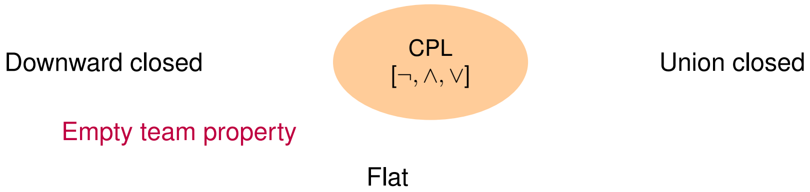
Union closed

Downward closed

CPL
$[\neg, \wedge, \vee]$

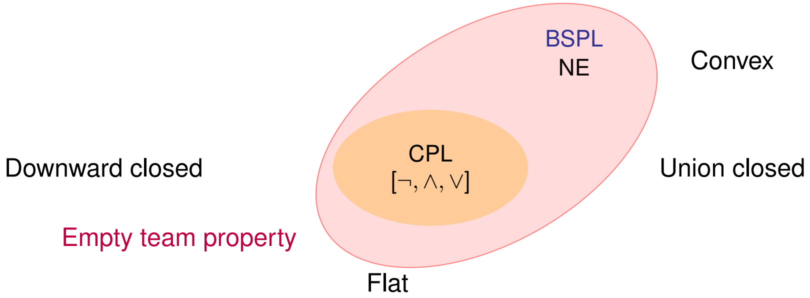
Union closed

Empty team property



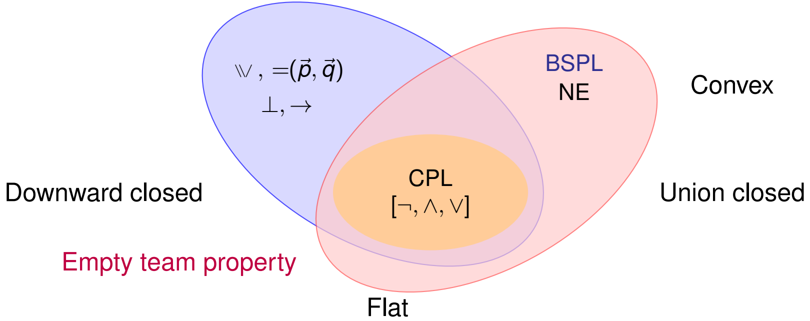

- Conservativity: If $\Delta \cup \{\alpha\}$ is a set of CPL-formulas, then
$$\Delta \models_{\text{team}} \alpha \iff \Delta \models_{\text{classical}} \alpha$$
- All these logics have been axiomatized (Ciardelli, Roelofsen 2011), (Y., Väänänen 2016), (Anttila, Aloni, Y. 2023), ...
- There are labelled sequent calculi, display calculi, and deep inference style calculi for inquisitive logic and $[\neg, \wedge, \vee, \mathbin{\vee\!\!\!\vee}]$ (Chen, Ma 2017), (Müler 2023), (Barbero, Girlando, Müler, Y. 2024), (Frittella, Greco, Palmigiano, Y. 2016), (Anttila, Iemhoff, Y. 2024)

# Propositional team-based logics around BSML



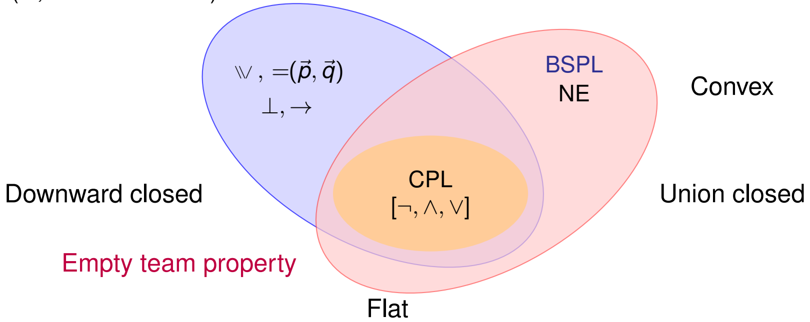

- Conservativity: If $\Delta \cup \{\alpha\}$ is a set of CPL-formulas, then
$$\Delta \models_{\text{team}} \alpha \iff \Delta \models_{\text{classical}} \alpha$$
- All these logics have been axiomatized (Ciardelli, Roelofsen 2011), (Y., Väänänen 2016), (Anttila, Aloni, Y. 2023), ...
- There are labelled sequent calculi, display calculi, and deep inference style calculi for inquisitive logic and $[\neg, \wedge, \vee, \mathbin{\vee\!\!\!\vee}]$ (Chen, Ma 2017), (Müler 2023), (Barbero, Girlando, Müler, Y. 2024), (Frittella, Greco, Palmigiano, Y. 2016), (Anttila, Iemhoff, Y. 2024)

# Propositional team-based logics around BSML




- Conservativity: If $\Delta \cup \{\alpha\}$ is a set of CPL-formulas, then
$$\Delta \models_{\text{team}} \alpha \iff \Delta \models_{\text{classical}} \alpha$$
- All these logics have been axiomatized (Ciardelli, Roelofsen 2011), (Y., Väänänen 2016), (Anttila, Aloni, Y. 2023), ...
- There are labelled sequent calculi, display calculi, and deep inference style calculi for inquisitive logic and $[\neg, \wedge, \vee, \mathbb{V}]$ (Chen, Ma 2017), (Müler 2023), (Barbero, Girlando, Müler, Y. 2024), (Frittella, Greco, Palmigiano, Y. 2016), (Anttila, Iemhoff, Y. 2024)

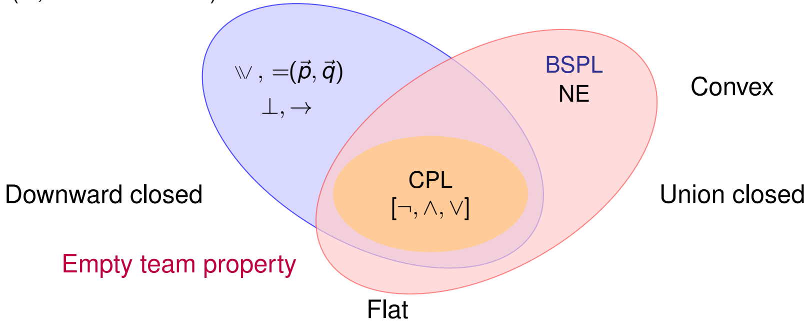
- Inquisitive logic: $[\bot, \wedge, ⩔, \to]$
  (Ciardelli, Roelofsen 2011)
- Propositional dependence logic: $[\neg, \wedge, \vee, =\!(\vec{p}, \vec{q})]$
  (Y., Väänänen 2016)




- Conservativity: If $\Delta \cup \{\alpha\}$ is a set of CPL-formulas, then
  $$\Delta \models_{\text{team}} \alpha \iff \Delta \models_{\text{classical}} \alpha$$
- All these logics have been axiomatized (Ciardelli, Roelofsen 2011), (Y., Väänänen 2016), (Anttila, Aloni, Y. 2023), ...
- There are labelled sequent calculi, display calculi, and deep inference style calculi for inquisitive logic and $[\neg, \wedge, \vee, ⩔]$ (Chen, Ma 2017), (Müler 2023), (Barbero, Girlando, Müler, Y. 2024), (Frittella, Greco, Palmigiano, Y. 2016), (Anttila, Iemhoff, Y. 2024)

# Propositional team-based logics around BSML

- Inquisitive logic: $[\bot, \wedge, \mathbin{\rotatebox[origin=c]{-90}{$\geqslant$}}, \to]$
  (Ciardelli, Roelofsen 2011)
- Propositional dependence logic: $[\neg, \wedge, \vee, =(\vec{p}, \vec{q})]$
  (Y., Väänänen 2016)




- Conservativity: If $\Delta \cup \{\alpha\}$ is a set of CPL-formulas, then
  $$\Delta \models_{\text{team}} \alpha \iff \Delta \models_{\text{classical}} \alpha$$
- All these logics have been axiomatized (Ciardelli, Roelofsen 2011), (Y., Väänänen 2016), (Anttila, Aloni, Y. 2023), ...
- There are labelled sequent calculi, display calculi, and deep inference style calculi for inquisitive logic and $[\neg, \wedge, \vee, ⩔]$ (Chen, Ma 2017), (Müler 2023), (Barbero, Girlando, Müler, Y. 2024), (Frittella, Greco, Palmigiano, Y. 2016), (Anttila, Iemhoff, Y. 2024).

Fix a finite set $N = \{p_1, \ldots, p_n\}$ of propositional variables.

Fact: Given a valuation $v : N \to \{0, 1\}$, the CPL-formula

$$\theta_v =$$

defines $v$, in the sense that for any $N$-valuation $u$,

$$u \models \theta_v \iff u = v$$

| $p_1$ | $p_2$ | $p_3$ |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

# The power of splitting teams, and the empty trap...

Fix a finite set $N = \{p_1, \ldots, p_n\}$ of propositional variables.

**Fact:** Given a valuation $v : N \to \{0, 1\}$, the CPL-formula

$$\theta_v =$$

defines $v$, in the sense that for any $N$-valuation $u$,

$$u \models \theta_v \iff u = v$$

| $p_1$ | $p_2$ | $p_3$ |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

$\neg p_1 \wedge p_2 \wedge \neg p_3$

Fix a finite set $N = \{p_1, \ldots, p_n\}$ of propositional variables.

Fact: Given a valuation $v : N \to \{0, 1\}$, the CPL-formula

$$\theta_v = p_1^{v(1)} \wedge \cdots \wedge p_n^{v(n)}, \text{ where } p_i^1 := p_i \text{ and } p_i^0 = \neg p_i,$$

defines $v$, in the sense that for any $N$-valuation $u$,

$$u \models \theta_v \iff u = v$$

| $p_1$ | $p_2$ | $p_3$ | |
|---|---|---|---|
| 0 | 1 | 0 | $\neg p_1 \wedge p_2 \wedge \neg p_3$ |
| 1 | 0 | 1 | $p_1 \wedge \neg p_2 \wedge p_3$ |
| 1 | 1 | 0 | $p_1 \wedge p_2 \wedge \neg p_3$ |

# The power of splitting teams, and the empty trap...

Fix a finite set $N = \{p_1, \dots, p_n\}$ of propositional variables.

**Fact:** Given a valuation $v : N \to \{0,1\}$, the CPL-formula

$$\theta_v = p_1^{v(1)} \wedge \cdots \wedge p_n^{v(n)}, \text{ where } p_i^1 := p_i \text{ and } p_i^0 = \neg p_i,$$

defines $v$, in the sense that for any $N$-valuation $u$,

$$u \models \theta_v \iff u = v$$

| $p_1$ | $p_2$ | $p_3$ | |
|---|---|---|---|
| 0 | 1 | 0 | $\neg p_1 \wedge p_2 \wedge \neg p_3$ |
| 1 | 0 | 1 | $p_1 \wedge \neg p_2 \wedge p_3$ |
| 1 | 1 | 0 | $p_1 \wedge p_2 \wedge \neg p_3$ |

**Question:** Given an $N$-team $t$, is there a formula $\Theta_t$ that defines $t$, in the sense that for any $N$-team $s$,

$$s \models \Theta_t \iff s = t?$$

Fix a finite set $N = \{p_1, \dots, p_n\}$ of propositional variables.

**Fact:** Given a valuation $v : N \to \{0, 1\}$, the CPL-formula

$$\theta_v = p_1^{v(1)} \wedge \cdots \wedge p_n^{v(n)}, \text{ where } p_i^1 := p_i \text{ and } p_i^0 = \neg p_i,$$

defines $v$, in the sense that for any $N$-valuation $u$,

$$u \models \theta_v \iff u = v$$

| $p_1$ | $p_2$ | $p_3$ | |
|---|---|---|---|
| 0 | 1 | 0 | $\neg p_1 \wedge p_2 \wedge \neg p_3$ |
| 1 | 0 | 1 | $p_1 \wedge \neg p_2 \wedge p_3$ |
| 1 | 1 | 0 | $p_1 \wedge p_2 \wedge \neg p_3$ |

**Question:** Given an $N$-team $t$, is there a formula $\Theta_t$ that defines $t$, in the sense that for any $N$-team $s$,

$$s \models \Theta_t \iff s = t?$$

Take $\Theta_t = \bigvee_{v \in t} (p_1^{v(1)} \wedge \cdots \wedge p_n^{v(n)})$!

# The power of splitting teams, and the empty trap...

Fix a finite set $N = \{p_1, \dots, p_n\}$ of propositional variables.

**Fact:** Given a valuation $v : N \to \{0, 1\}$, the CPL-formula

$$\theta_v = p_1^{v(1)} \wedge \cdots \wedge p_n^{v(n)}, \text{ where } p_i^1 := p_i \text{ and } p_i^0 = \neg p_i,$$

defines $v$, in the sense that for any $N$-valuation $u$,

$$u \models \theta_v \iff u = v$$

| $p_1$ | $p_2$ | $p_3$ | |
|---|---|---|---|
| 0 | 1 | 0 | $\neg p_1 \wedge p_2 \wedge \neg p_3$ |
| 1 | 0 | 1 | $p_1 \wedge \neg p_2 \wedge p_3$ |
| 1 | 1 | 0 | $p_1 \wedge p_2 \wedge \neg p_3$ |

**Question:** Given an $N$-team $t$, is there a formula $\Theta_t$ that defines $t$, in the sense that for any $N$-team $s$,

$$s \models \Theta_t \iff s = t?$$

Take $\Theta_t = \bigvee_{v \in t} (p_1^{v(1)} \wedge \cdots \wedge p_n^{v(n)})$! But in fact, not exactly, due to $\emptyset$:

$$s \models \Theta_t \iff s = \bigcup_{v \in t} s_v \text{ and each } s_v \models \theta_v \iff s \subseteq t.$$

**Theorem (Y. and Väänänen 2016 )**

*The team-based logic* $[\neg, \wedge, \vee, \mathbin{\mathbb{V}}]$ *is expressively complete.*

# Expressive completeness and normal forms

**Theorem (Y. and Väänänen 2016 )**

*The team-based logic* $[\neg, \wedge, \vee, \mathbin{\mathbb{V}}]$ *is expressively complete.*

... in the following sense:

- A formula $\phi$ in $N$ defines a team property $[\![\phi]\!] = \{t \subseteq 2^N : t \models \phi\}$, which is closed downward and contains the empty team.
- For any $N$-team property P that is closed downward and contains the empty team, there is a formula $\phi \in [\neg, \wedge, \vee, \mathbin{\mathbb{V}}]$ such that $[\![\phi]\!] = \mathsf{P}$.

**Theorem (Y. and Väänänen 2016 )**

*The team-based logic* $[\neg, \wedge, \vee, \mathbin{\veebar}]$ *is expressively complete.*

... in the following sense:

- A formula $\phi$ in $N$ defines a team property $[\![\phi]\!] = \{t \subseteq 2^N : t \models \phi\}$, which is closed downward and contains the empty team.
- For any $N$-team property P that is closed downward and contains the empty team, there is a formula $\phi \in [\neg, \wedge, \vee, \mathbin{\veebar}]$ such that $[\![\phi]\!] = \mathsf{P}$.

## Expressive completeness and normal forms

**Theorem (Y. and Väänänen 2016 )**

*The team-based logic $[\neg, \wedge, \vee, \mathbin{\mathbb{V}}]$ is expressively complete.*

... in the following sense:

- A formula $\phi$ in $N$ defines a team property $[\![\phi]\!] = \{t \subseteq 2^N : t \models \phi\}$, which is closed downward and contains the empty team.
- For any $N$-team property P that is closed downward and contains the empty team, there is a formula $\phi \in [\neg, \wedge, \vee, \mathbin{\mathbb{V}}]$ such that $[\![\phi]\!] = \mathsf{P}$.

  Proof. Take $\phi = \mathop{\mathbb{V}}_{t \in \mathsf{P}} \Theta_t$.

**Disjunctive normal form**

**Theorem (Y. and Väänänen 2016 )**

*The team-based logic* $[\neg, \wedge, \vee, \mathbin{\vee\!\!\!\vee}]$ *is expressively complete.*

... in the following sense:

- A formula $\phi$ in $N$ defines a team property $[\![\phi]\!] = \{t \subseteq 2^N : t \models \phi\}$, which is closed downward and contains the empty team.
- For any $N$-team property P that is closed downward and contains the empty team, there is a formula $\phi \in [\neg, \wedge, \vee, \mathbin{\vee\!\!\!\vee}]$ such that $[\![\phi]\!] = \mathsf{P}$.

  Proof. Take $\phi = \bigvee\!\!\!\bigvee_{t \in \mathsf{P}} \Theta_t$. Then,

$$s \models \bigvee\!\!\!\bigvee_{t \in \mathsf{P}} \Theta_t \iff s \subseteq t \text{ for some } t \in \mathsf{P} \iff s \in \mathsf{P}.$$

Disjunctive normal form: "The current team $s$ is one of the teams in P"

**Theorem (Y. and Väänänen 2016 )**

*The team-based logic* $[\neg, \wedge, \vee, \mathbin{\vee\!\!\!\vee}]$ *is expressively complete.*

... in the following sense:

- A formula $\phi$ in $N$ defines a team property $[\![\phi]\!] = \{t \subseteq 2^N : t \models \phi\}$, which is closed downward and contains the empty team.
- For any $N$-team property P that is closed downward and contains the empty team, there is a formula $\phi \in [\neg, \wedge, \vee, \mathbin{\vee\!\!\!\vee}]$ such that $[\![\phi]\!] = \mathsf{P}$.

  Proof. Take $\phi = \bigvee\!\!\!\bigvee_{t \in \mathsf{P}} \Theta_t$. Then,

$$s \models \bigvee\!\!\!\bigvee_{t \in \mathsf{P}} \Theta_t \iff s \subseteq t \text{ for some } t \in \mathsf{P} \iff s \in \mathsf{P}.$$

Disjunctive normal form: "The current team $s$ is one of the teams in P"

**Theorem (Y. and Väänänen 2016 )**

*Propositional dependence logic* $[\neg, \wedge, \vee, =\!(\cdot)]$ *is expressively complete.*

# Expressive completeness and normal forms

**Theorem (Y. and Väänänen 2016 )**

*The team-based logic* $[\neg, \wedge, \vee, \mathbin{\vee\!\!\!\vee}]$ *is expressively complete.*

... in the following sense:

- A formula $\phi$ in $N$ defines a team property $[\![\phi]\!] = \{t \subseteq 2^N : t \models \phi\}$, which is closed downward and contains the empty team.
- For any $N$-team property P that is closed downward and contains the empty team, there is a formula $\phi \in [\neg, \wedge, \vee, \mathbin{\vee\!\!\!\vee}]$ such that $[\![\phi]\!] = \mathsf{P}$.

  Proof. Take $\phi = \bigvee\!\!\!\!\bigvee_{t \in \mathsf{P}} \Theta_t$. Then,

  $$s \models \bigvee\!\!\!\!\bigvee_{t \in \mathsf{P}} \Theta_t \iff s \subseteq t \text{ for some } t \in \mathsf{P} \iff s \in \mathsf{P}.$$

Disjunctive normal form: "The current team $s$ is one of the teams in P"

**Theorem (Y. and Väänänen 2016 )**

*Propositional dependence logic* $[\neg, \wedge, \vee, =\!(\cdot)]$ *is expressively complete.*

A conjunctive normal form: "The current team $s$ is not any team in $\overline{\mathsf{P}}$"

# Constancy atoms

- $t \models {=}(\langle\rangle, \vec{p})$ iff for all $v, u \in t$: $v(\langle\rangle) = u(\langle\rangle) \Longrightarrow v(\vec{p}) = u(\vec{p})$.

# Constancy atoms

- $t \models =\!(\vec{p})$ iff for all $v, u \in t$: $v(\langle\rangle) = u(\langle\rangle) \Longrightarrow v(\vec{p}) = u(\vec{p})$.

# Constancy atoms

- $t \models =(\vec{p})$ iff for all $v, u \in t$: $v(\vec{p}) = u(\vec{p})$.

| ... | $p$ | ... |
|---|---|---|
| ... | 1 | ... |
| ... | 1 | ... |
| ... | 1 | ... |
| ... | 1 | ... |

# Constancy atoms

- $t \models =(\vec{p})$ iff for all $v, u \in t$: $v(\vec{p}) = u(\vec{p})$.

| $\ldots$ | $p$ | $\ldots$ |
|---|---|---|
| $\ldots$ | 0 | $\ldots$ |
| $\ldots$ | 0 | $\ldots$ |
| $\ldots$ | 0 | $\ldots$ |
| $\ldots$ | 0 | $\ldots$ |

# Constancy atoms

- $t \models =(\vec{p})$ iff for all $v, u \in t$: $v(\vec{p}) = u(\vec{p})$.

| $\dots$ | $p$ | $\dots$ |
|---|---|---|
| $\dots$ | 0 | $\dots$ |
| $\dots$ | 1 | $\dots$ |
| $\dots$ | 0 | $\dots$ |
| $\dots$ | 1 | $\dots$ |

# Constancy atoms, “excluded middle"

- $t \models =\!(\vec{p})$ iff for all $v, u \in t$: $v(\vec{p}) = u(\vec{p})$.

| $\dots$ | $p$ | $\dots$ |
|---|---|---|
| $\dots$ | 0 | $\dots$ |
| $\dots$ | 1 | $\dots$ |
| $\dots$ | 0 | $\dots$ |
| $\dots$ | 1 | $\dots$ |

Fact: $\models =\!(p) \vee =\!(p)$

- $t \models \phi \vee \psi$ iff $\exists s, r \subseteq t$ s.t. $t = s \cup r$, $s \models \phi$ and $r \models \psi$

# Constancy atoms, “excluded middle"

- $t \models =\!(\vec{p})$ iff for all $v, u \in t$: $v(\vec{p}) = u(\vec{p})$.

| … | $p$ | … |
|---|---|---|
| … | 0 | … |
| … | 0 | … |
| … | 1 | … |
| … | 1 | … |

Fact: $\models =\!(p) \vee =\!(p)$

Failure of closure under uniform substitution:

$$p \vee p \models p, \text{ whereas } =\!(p) \vee =\!(p) \not\models =\!(p).$$

- $t \models \phi \vee \psi$ iff $\exists s, r \subseteq t$ s.t. $t = s \cup r$, $s \models \phi$ and $r \models \psi$

- $t \models =\!(\vec{p})$ iff for all $v, u \in t$: $v(\vec{p}) = u(\vec{p})$.

| ... | $p$ | ... |
|---|---|---|
| ... | 1 | ... |
| ... | 1 | ... |
| ... | 1 | ... |
| ... | 1 | ... |

or

| ... | $p$ | ... |
|---|---|---|
| ... | 0 | ... |
| ... | 0 | ... |
| ... | 0 | ... |
| ... | 0 | ... |

Fact: $\models =\!(p) \vee =\!(p)$

Failure of closure under uniform substitution:

$$p \vee p \models p, \text{ whereas } =\!(p) \vee =\!(p) \not\models =\!(p).$$

Fact: $=\!(p) \equiv p \veebar \neg p$

- $t \models \phi \vee \psi$ iff $\exists s, r \subseteq t$ s.t. $t = s \cup r$, $s \models \phi$ and $r \models \psi$
- $t \models \phi \veebar \psi$ iff $t \models \phi$ or $t \models \psi$

# Constancy atoms, “excluded middle"

- $t \models =\!(\vec{p})$ iff for all $v, u \in t$: $v(\vec{p}) = u(\vec{p})$.

| ... | $p$ | ... |
|---|---|---|
| ... | 1 | ... |
| ... | 1 | ... |
| ... | 1 | ... |
| ... | 1 | ... |

or

| ... | $p$ | ... |
|---|---|---|
| ... | 0 | ... |
| ... | 0 | ... |
| ... | 0 | ... |
| ... | 0 | ... |

Fact: $\models =\!(p) \vee =\!(p)$

Failure of closure under uniform substitution:

$$p \vee p \models p, \text{ whereas } =\!(p) \vee =\!(p) \not\models =\!(p).$$

Fact: $=\!(p) \equiv p \mathbin{\vee\!\!\!\vee} \neg p \equiv ?p$ (in inquisitive logic)

- $t \models \phi \vee \psi$ iff $\exists s, r \subseteq t$ s.t. $t = s \cup r$, $s \models \phi$ and $r \models \psi$
- $t \models \phi \mathbin{\vee\!\!\!\vee} \psi$ iff $t \models \phi$ or $t \models \psi$

Let $N = \{p_1, \ldots, p_n\}$.

**Fact:** For any $N$-team $t$,

$$t \models =\!(p_1) \wedge \cdots \wedge =\!(p_n) \iff |t| \leq 1.$$

| | $p_1$ | $p_2$ | $\ldots$ | $p_n$ |
|---|---|---|---|---|
| | 1 | 0 | $\ldots$ | 1 |
| $t$ | 0 | 1 | $\ldots$ | 0 |
| | 0 | 1 | $\ldots$ | 1 |
| | 1 | 0 | $\ldots$ | 1 |

# Counting with constancy atoms

Let $N = \{p_1, \ldots, p_n\}$.

**Fact:** For any $N$-team $t$,

$$t \models =\!(p_1) \wedge \cdots \wedge =\!(p_n) \iff |t| \leq 1.$$

| | $p_1$ | $p_2$ | $\ldots$ | $p_n$ |
|---|---|---|---|---|
| | 1 | 0 | $\ldots$ | 1 |
| $t$ | 0 | 1 | $\ldots$ | 0 |
| | 0 | 1 | $\ldots$ | 1 |
| | 1 | 0 | $\ldots$ | 1 |

Let $N = \{p_1, \ldots, p_n\}$.

Fact: For any $N$-team $t$,

$$t \models =(p_1) \wedge \cdots \wedge =(p_n) \iff |t| \leq 1.$$

| $p_1$ | $p_2$ | $\ldots$ | $p_n$ |
|---|---|---|---|
| 1 | 0 | $\ldots$ | 1 |
| 0 | 1 | $\ldots$ | 0 |
| 0 | 1 | $\ldots$ | 1 |
| 1 | 0 | $\ldots$ | 1 |

Define

$$\eta_k = \bigvee_{i=1}^{k} (=(p_1) \wedge \cdots \wedge =(p_n))$$

Prop. For any $N$-team $t$, we have that

$$t \models \eta_k \iff |t| \leq k.$$

**Prop.** For any $N$-teams $t$, the formula $\Theta_t = \bigvee_{v \in t}(p_1^{v(1)} \wedge \cdots \wedge p_n^{v(n)})$ satisfies that for any $N$-team $s$, $\quad s \models \Theta_t \iff s \subseteq t$.

Prop. For any $N$-teams $t$, the formula $\Theta_t = \bigvee_{v \in t}(p_1^{v(1)} \wedge \cdots \wedge p_n^{v(n)})$ satisfies that for any $N$-team $s$, $s \models \Theta_t \iff s \subseteq t$.

Lemma (Huuskonen 2016). For any nonempty $N$-team $t$, there exists a formula $\Phi_t$ such that for any $N$-team $s$,

$$s \models \Phi_t \iff t \not\subseteq s.$$

**Prop.** For any $N$-teams $t$, the formula $\Theta_t = \bigvee_{v\in t}(p_1^{v(1)} \wedge \cdots \wedge p_n^{v(n)})$ satisfies that for any $N$-team $s$, $\quad s \models \Theta_t \iff s \subseteq t$.

**Prop.** For any $N$-team $t$, we have that $\quad t \models \eta_k \iff |t| \leq k$.

**Lemma (Huuskonen 2016).** For any nonempty $N$-team $t$, there exists a formula $\Phi_t$ such that for any $N$-team $s$,

$$s \models \Phi_t \iff t \not\subseteq s.$$

**Proof.** Let $|t| = k + 1$. Define $\Phi_t := \eta_k \vee \Theta_{\bar{t}}$, where $\bar{t} = 2^N \setminus t$.

# Saying "no" to supersets

Prop. For any $N$-teams $t$, the formula $\Theta_t = \bigvee_{v \in t}(p_1^{v(1)} \wedge \cdots \wedge p_n^{v(n)})$ satisfies that for any $N$-team $s$, $s \models \Theta_t \iff s \subseteq t$.

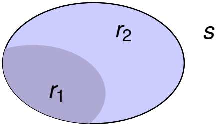
Prop. For any $N$-team $t$, we have that $t \models \eta_k \iff |t| \leq k$.

Lemma (Huuskonen 2016). For any nonempty $N$-team $t$, there exists a formula $\Phi_t$ such that for any $N$-team $s$,

$$s \models \Phi_t \iff t \not\subseteq s.$$

Proof. Let $|t| = k + 1$. Define $\Phi_t := \eta_k \vee \Theta_{\bar{t}}$, where $\bar{t} = 2^N \setminus t$.
For any $N$-team $s$, we have that

$$s \models \Phi_t \iff s = r_1 \cup r_2 \text{ such that } r_1 \models \eta_k \text{ and } r_2 \models \Theta_{\bar{t}}$$



# Saying “no” to supersets

Prop. For any $N$-teams $t$, the formula $\Theta_t = \bigvee_{v \in t}(p_1^{v(1)} \wedge \cdots \wedge p_n^{v(n)})$ satisfies that for any $N$-team $s$, $\quad s \models \Theta_t \iff s \subseteq t$.

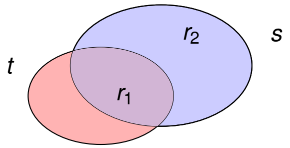
Prop. For any $N$-team $t$, we have that $\quad t \models \eta_k \iff |t| \leq k$.

Lemma (Huuskonen 2016). For any nonempty $N$-team $t$, there exists a formula $\Phi_t$ such that for any $N$-team $s$,

$$s \models \Phi_t \iff t \not\subseteq s.$$

Proof. Let $|t| = k + 1$. Define $\Phi_t := \eta_k \vee \Theta_{\bar{t}}$, where $\bar{t} = 2^N \setminus t$.
For any $N$-team $s$, we have that

$$\begin{aligned} s \models \Phi_t &\iff s = r_1 \cup r_2 \text{ such that } r_1 \models \eta_k \text{ and } r_2 \models \Theta_{\bar{t}} \\ &\iff s = r_1 \cup r_2 \text{ such that } |r_1| \leq k \text{ and } r_2 \subseteq \bar{t}, \text{ i.e., } r_2 \cap t = \emptyset \end{aligned}$$



# Saying "no" to supersets

Prop. For any $N$-teams $t$, the formula $\Theta_t = \bigvee_{v \in t}(p_1^{v(1)} \wedge \cdots \wedge p_n^{v(n)})$ satisfies that for any $N$-team $s$, $s \models \Theta_t \iff s \subseteq t$.

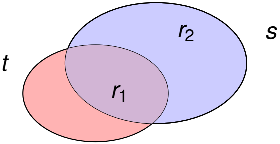
Prop. For any $N$-team $t$, we have that $t \models \eta_k \iff |t| \leq k$.

Lemma (Huuskonen 2016). For any nonempty $N$-team $t$, there exists a formula $\Phi_t$ such that for any $N$-team $s$,

$$s \models \Phi_t \iff t \not\subseteq s.$$

Proof. Let $|t| = k + 1$. Define $\Phi_t := \eta_k \vee \Theta_{\bar{t}}$, where $\bar{t} = 2^N \setminus t$.
For any $N$-team $s$, we have that

$$\begin{aligned} s \models \Phi_t &\iff s = r_1 \cup r_2 \text{ such that } r_1 \models \eta_k \text{ and } r_2 \models \Theta_{\bar{t}} \\ &\iff s = r_1 \cup r_2 \text{ such that } |r_1| \leq k \text{ and } r_2 \subseteq \bar{t}, \text{ i.e., } r_2 \cap t = \emptyset \\ &\iff t \not\subseteq s. \end{aligned}$$




□

# Conjunctive normal form

**Theorem (Y. and Väänänen 2016 )**

*Propositional dependence logic* $[\neg, \wedge, \vee, =\!(\cdot)]$ *is expressively complete.*

Nontrivial direction: For any *N*-team property P that is closed downward and contains the empty team, there is a formula $\phi \in [\neg, \wedge, \vee, =\!(\cdot)]$ such that $[\![\phi]\!] = \mathsf{P}$.

# Conjunctive normal form

**Theorem (Y. and Väänänen 2016 )**

*Propositional dependence logic* $[\neg, \wedge, \vee, =\!(\cdot)]$ *is expressively complete.*

Nontrivial direction: For any $N$-team property P that is closed downward and contains the empty team, there is a formula $\phi \in [\neg, \wedge, \vee, =\!(\cdot)]$ such that $[\![\phi]\!] = \mathsf{P}$.

Proof. Take $\phi = \bigwedge_{t \in \overline{P}} \Phi_t$. Then

$$s \models \bigwedge_{t \in \overline{P}} \Phi_t \iff t \not\subseteq s \text{ for all } t \notin P \iff s \in P.$$

□

**Theorem (Y. and Väänänen 2016 )**

*Propositional dependence logic* $[\neg, \wedge, \vee, =(\cdot)]$ *is expressively complete.*

Nontrivial direction: For any $N$-team property P that is closed downward and contains the empty team, there is a formula $\phi \in [\neg, \wedge, \vee, =(\cdot)]$ such that $[\![\phi]\!] = \mathsf{P}$.

Proof. Take $\phi = \bigwedge_{t \in \overline{P}} \Phi_t$. Then

$$s \models \bigwedge_{t \in \overline{P}} \Phi_t \iff t \not\subseteq s \text{ for all } t \notin P \iff s \in P.$$

□

Conjunctive normal form: "The current team $s$ is not any team in $\overline{\mathsf{P}}$"

# Conjunctive normal form

**Theorem (Y. and Väänänen 2016 )**

*Propositional dependence logic* $[\neg, \wedge, \vee, =\!(\cdot)]$ *is expressively complete.*

Nontrivial direction: For any $N$-team property P that is closed downward and contains the empty team, there is a formula $\phi \in [\neg, \wedge, \vee, =\!(\cdot)]$ such that $[\![\phi]\!] = \mathsf{P}$.

Proof. Take $\phi = \bigwedge_{t \in \overline{P}} \Phi_t$. Then

$$s \models \bigwedge_{t \in \overline{P}} \Phi_t \iff t \not\subseteq s \text{ for all } t \notin P \iff s \in P.$$

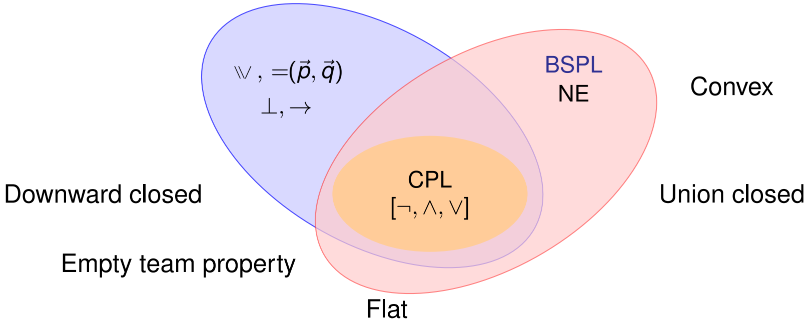
□

Conjunctive normal form: "The current team $s$ is not any team in $\overline{\mathsf{P}}$"

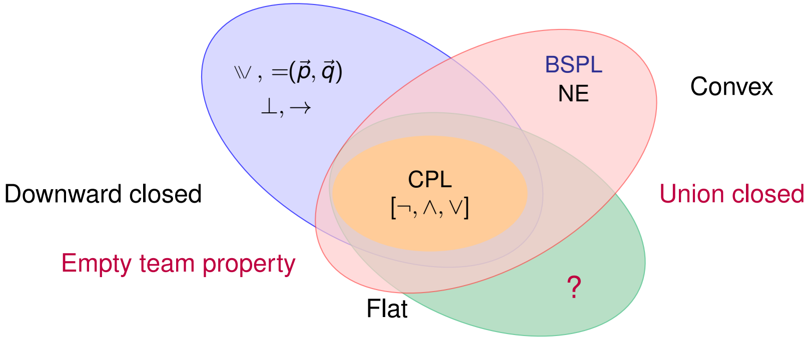
Disjunctive normal form: "The current team $s$ is one of the teams in P"

$\mathbb{W}_{t \in \mathsf{P}} \Theta_t$

# Union closed team-based logics around BSML?



# Union closed team-based logics around BSML?



# Database theory connection revisited: inclusion dependencies

- A team can be viewed as a relational database.
- Dependence atoms $=\!(\vec{p}, \vec{q})$ correspond to functional dependencies $\vec{p} \to \vec{q}$ in database theory
- Inclusion dependencies give rise to inclusion atoms (Galliani 2012):

| $p$ | $q$ | $r$ | $s$ |
|---|---|---|---|
| 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 |

# Database theory connection revisited: inclusion dependencies

- A team can be viewed as a relational database.
- Dependence atoms $=(\vec{p}, \vec{q})$ correspond to functional dependencies $\vec{p} \to \vec{q}$ in database theory
- Inclusion dependencies give rise to inclusion atoms (Galliani 2012):

| $p$ | $q$ | $r$ | $s$ |
|---|---|---|---|
| 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 |

$$p_1 \ldots p_n \subseteq q_1 \ldots q_n$$

# Database theory connection revisited: inclusion dependencies

- A team can be viewed as a relational database.
- Dependence atoms $=(\vec{p}, \vec{q})$ correspond to functional dependencies $\vec{p} \to \vec{q}$ in database theory
- Inclusion dependencies give rise to inclusion atoms (Galliani 2012):

E.g.,

$pq \subseteq rs$

| $p$ | $q$ | $r$ | $s$ |
|---|---|---|---|
| 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 |

- $t \models p_1 \dots p_n \subseteq q_1 \dots q_n$ iff for all $v \in t$, there exists $u \in t$ s.t. $v(p_i) = u(q_i)$ for all $1 \leq i \leq n$.

# Database theory connection revisited: inclusion dependencies

- A team can be viewed as a relational database.
- Dependence atoms $=\!(\vec{p}, \vec{q})$ correspond to functional dependencies $\vec{p} \to \vec{q}$ in database theory
- Inclusion dependencies give rise to inclusion atoms (Galliani 2012):

E.g.,

$pq \subseteq rs$

| $p$ | $q$ | $r$ | $s$ |
|---|---|---|---|
| 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 |

- $t \models p_1 \dots p_n \subseteq q_1 \dots q_n$ iff for all $v \in t$, there exists $u \in t$ s.t. $v(p_i) = u(q_i)$ for all $1 \leq i \leq n$.

# Database theory connection revisited: inclusion dependencies

- A team can be viewed as a relational database.
- Dependence atoms $=(\vec{p}, \vec{q})$ correspond to functional dependencies $\vec{p} \to \vec{q}$ in database theory
- Inclusion dependencies give rise to inclusion atoms (Galliani 2012):

E.g.,

$$pq \subseteq rs$$
$$\bot\top \subseteq pq$$
$$\bot rs \subseteq pq\top$$

| $p$ | $q$ | $r$ | $s$ |
|---|---|---|---|
| 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 |

- $t \models a_1 \dots a_n \subseteq b_1 \dots b_n$ iff for all $v \in t$, there exists $u \in t$ s.t. $v(a_i) = u(b_i)$ for all $1 \leq i \leq n$.

where each $a_i, b_i \in \text{Prop} \cup \{\top, \bot\}$.

# Database theory connection revisited: inclusion dependencies

- A team can be viewed as a relational database.
- Dependence atoms $=(\vec{p}, \vec{q})$ correspond to functional dependencies $\vec{p} \to \vec{q}$ in database theory
- Inclusion dependencies give rise to inclusion atoms (Galliani 2012):

E.g.,

$pq \subseteq rs$

$\bot\top \subseteq pq$

$\bot rs \subseteq pq\top$

| $\bot$ | $\top$ | $p$ | $q$ | $r$ | $s$ |
|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 |

- $t \models a_1 \ldots a_n \subseteq b_1 \ldots b_n$ iff for all $v \in t$, there exists $u \in t$ s.t. $v(a_i) = u(b_i)$ for all $1 \leq i \leq n$.

where each $a_i, b_i \in \text{Prop} \cup \{\top, \bot\}$.

# Database theory connection revisited: inclusion dependencies

- A team can be viewed as a relational database.
- Dependence atoms $=(\vec{p}, \vec{q})$ correspond to functional dependencies $\vec{p} \to \vec{q}$ in database theory
- Inclusion dependencies give rise to inclusion atoms (Galliani 2012):

E.g.,
$pq \subseteq rs$
$\bot\top \subseteq pq$
$\bot rs \subseteq pq\top$

| $\bot$ | $\top$ | $p$ | $q$ | $r$ | $s$ |
|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 |

- $t \models a_1 \dots a_n \subseteq b_1 \dots b_n$ iff for all $v \in t$, there exists $u \in t$ s.t. $v(a_i) = u(b_i)$ for all $1 \leq i \leq n$.

  where each $a_i, b_i \in \mathsf{Prop} \cup \{\top, \bot\}$.

We now consider the logic $[\neg, \wedge, \vee, \vec{a} \subseteq \vec{b}]$, known as propositional inclusion logic.

Empty team property: $\emptyset \models \psi$ for all $\psi$

Union closure: If $t \models \phi$ and $s \models \phi$, then $t \cup s \models \phi$.

**Theorem ((Y. 2022), (Hella, Kuusisto, Meier, Vollmer 2019))**

*Propositional inclusion logic* $[\neg, \wedge, \vee, \subseteq]$ *is expressively complete over union closed team properties that contain the empty team.*

Proof idea of nontrivial direction: For any $N$-team property P that is union closed and contains the empty team, construct a formula $\phi \in [\neg, \wedge, \vee, \subseteq]$ such that $[\![\phi]\!] = \mathsf{P}$, i.e., $s \models \phi \iff s \in \mathsf{P}$.

# Expressive completeness and disjunctive normal form

**Theorem ((Y. 2022), (Hella, Kuusisto, Meier, Vollmer 2019))**

*Propositional inclusion logic* $[\neg, \wedge, \vee, \subseteq]$ *is expressively complete over union closed team properties that contain the empty team.*

Proof idea of nontrivial direction: For any $N$-team property P that is union closed and contains the empty team, construct a formula $\phi \in [\neg, \wedge, \vee, \subseteq]$ such that $[\![\phi]\!] = \mathsf{P}$, i.e., $s \models \phi \iff s \in \mathsf{P}$. The formula $\phi = \bigvee_{t \in \mathsf{P}} \phi_t$ in DNF says roughly

"the current team $s$ is the union of some teams in P, and thus in P".

**Theorem ((Y. 2022), (Hella, Kuusisto, Meier, Vollmer 2019))**

*Propositional inclusion logic* $[\neg, \wedge, \vee, \subseteq]$ *is expressively complete over union closed team properties that contain the empty team.*

Proof idea of nontrivial direction: For any $N$-team property P that is union closed and contains the empty team, construct a formula $\phi \in [\neg, \wedge, \vee, \subseteq]$ such that $[\![\phi]\!] = \mathsf{P}$, i.e., $s \models \phi \iff s \in \mathsf{P}$. The formula $\phi = \bigvee_{t \in \mathsf{P}} \phi_t$ in DNF says roughly

"the current team $s$ is the union of some teams in P, and thus in P".

This is achieved by taking $\phi_t = \Theta_t \wedge \Psi_t$, where:

# Expressive completeness and disjunctive normal form

**Theorem ((Y. 2022), (Hella, Kuusisto, Meier, Vollmer 2019))**

*Propositional inclusion logic* $[\neg, \wedge, \vee, \subseteq]$ *is expressively complete over union closed team properties that contain the empty team.*

Proof idea of nontrivial direction: For any $N$-team property P that is union closed and contains the empty team, construct a formula $\phi \in [\neg, \wedge, \vee, \subseteq]$ such that $[\![\phi]\!] = \mathsf{P}$, i.e., $s \models \phi \iff s \in \mathsf{P}$. The formula $\phi = \bigvee_{t \in \mathsf{P}} \phi_t$ in DNF says roughly

"the current team $s$ is the union of some teams in P, and thus in P".

This is achieved by taking $\phi_t = \Theta_t \wedge \Psi_t$, where:

Prop. For any $N$-team $t$, the formula $\Theta_t = \bigvee_{v \in t}(p_1^{v(1)} \wedge \cdots \wedge p_n^{v(n)})$ satisfies that for any $N$-team $s$,

$$s \models \Theta_t \iff s \subseteq t.$$

**Theorem ((Y. 2022), (Hella, Kuusisto, Meier, Vollmer 2019))**

*Propositional inclusion logic $[\neg, \wedge, \vee, \subseteq]$ is expressively complete over union closed team properties that contain the empty team.*

Proof idea of nontrivial direction: For any $N$-team property P that is union closed and contains the empty team, construct a formula $\phi \in [\neg, \wedge, \vee, \subseteq]$ such that $[\![\phi]\!] = \mathsf{P}$, i.e., $s \models \phi \iff s \in \mathsf{P}$. The formula $\phi = \bigvee_{t \in \mathsf{P}} \phi_t$ in DNF says roughly

"the current team $s$ is the union of some teams in P, and thus in P".

This is achieved by taking $\phi_t = \Theta_t \wedge \Psi_t$, where:

**Prop.** For any $N$-team $t$, the formula $\Theta_t = \bigvee_{v \in t}(p_1^{v(1)} \wedge \cdots \wedge p_n^{v(n)})$ satisfies that for any $N$-team $s$,

$$s \models \Theta_t \iff s \subseteq t.$$

**Prop.** For any $N$-team $t$, there is a formula $\Psi_t$ s.t. for any $N$-team $s$,

$$s \models \Psi_t \iff t \subseteq s \text{ or } s = \emptyset.$$

# Expressive completeness and disjunctive normal form

**Theorem ((Y. 2022), (Hella, Kuusisto, Meier, Vollmer 2019))**

*Propositional inclusion logic $[\neg, \wedge, \vee, \subseteq]$ is expressively complete over union closed team properties that contain the empty team.*

Proof idea of nontrivial direction: For any $N$-team property P that is union closed and contains the empty team, construct a formula $\phi \in [\neg, \wedge, \vee, \subseteq]$ such that $[\![\phi]\!] = \mathsf{P}$, i.e., $s \models \phi \iff s \in \mathsf{P}$. The formula $\phi = \bigvee_{t \in \mathsf{P}} \phi_t$ in DNF says roughly

"the current team $s$ is the union of some teams in P, and thus in P".

This is achieved by taking $\phi_t = \Theta_t \wedge \Psi_t$, where:

Prop. For any $N$-team $t$, the formula $\Theta_t = \bigvee_{v \in t}(p_1^{v(1)} \wedge \cdots \wedge p_n^{v(n)})$ satisfies that for any $N$-team $s$,

$$s \models \Theta_t \iff s \subseteq t.$$

Prop. For any $N$-team $t$, there is a formula $\Psi_t$ s.t. for any $N$-team $s$,

$$s \models \Psi_t \iff t \subseteq s \text{ or } s = \emptyset.$$

Thus, $s \models \Theta_t \wedge \Psi_t \iff s = t \text{ or } s = \emptyset$.

# Defining supersets (... but ignore the empty set please)

Prop. For any $N$-team $t$, there is a formula $\Psi_t$ s.t. for any $N$-team $s$,

$$s \models \Psi_t \iff t \subseteq s \text{ or } s = \varnothing.$$

| $p_1$ | $p_2$ | $p_3$ |
|---|---|---|
| 1 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 1 |

# Defining supersets (... but ignore the empty set please)

**Prop.** For any $N$-team $t$, there is a formula $\Psi_t$ s.t. for any $N$-team $s$,

$$s \models \Psi_t \iff t \subseteq s \text{ or } s = \varnothing.$$

**Proof.** Observe that for any $v : N \to \{0, 1\}$,

$$s \models \underline{v(p_1)} \ldots \underline{v(p_n)} \subseteq p_1 \ldots p_n \iff v \in s \text{ or } s = \varnothing,$$

where $\underline{1} := \top$ and $\underline{0} := \bot$.

| | $p_1$ | $p_2$ | $p_3$ | |
|---|---|---|---|---|
| | 1 | 0 | 1 | |
| $v$ | 0 | 1 | 0 | $\bot\top\bot \subseteq p_1 p_2 p_3$ |
| | 1 | 1 | 1 | |
| | 0 | 1 | 1 | |

**Prop.** For any $N$-team $t$, there is a formula $\Psi_t$ s.t. for any $N$-team $s$,

$$s \models \Psi_t \iff t \subseteq s \text{ or } s = \varnothing.$$

**Proof.** Observe that for any $v : N \to \{0, 1\}$,

$$s \models \underline{v(p_1)} \dots \underline{v(p_n)} \subseteq p_1 \dots p_n \iff v \in s \text{ or } s = \varnothing,$$

where $\underline{1} := \top$ and $\underline{0} := \bot$.

For any $N$-team $t$, define

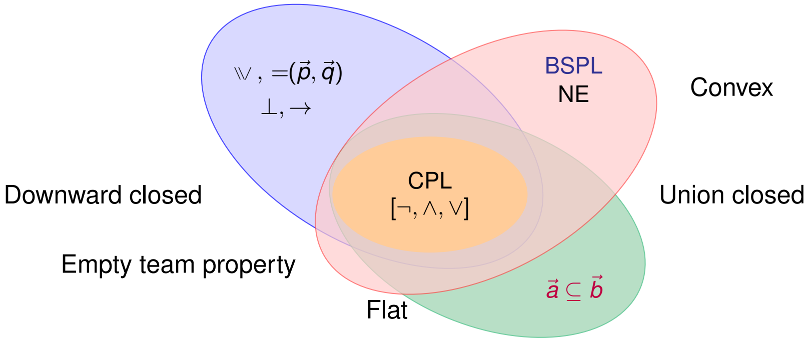
$$\Psi_t = \bigwedge_{v \in t} \underline{v(p_1)} \dots \underline{v(p_n)} \subseteq p_1 \dots p_n.$$

...

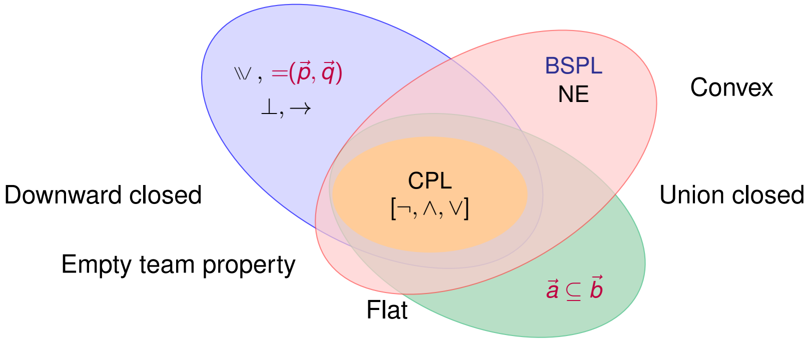
□

| | $p_1$ | $p_2$ | $p_3$ | |
|---|---|---|---|---|
| | 1 | 0 | 1 | |
| $v$ | 0 | 1 | 0 | $\bot\top\bot \subseteq p_1 p_2 p_3$ |
| | 1 | 1 | 1 | |
| | 0 | 1 | 1 | |

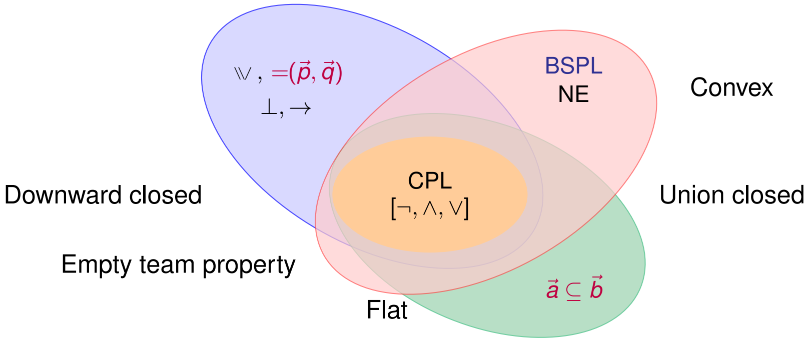
# More expressively complete logics?



# More expressively complete logics?




Question: What is (the expressive power of) CPL($=(\vec{p}, \vec{q}), \vec{a} \subseteq \vec{b}$)?

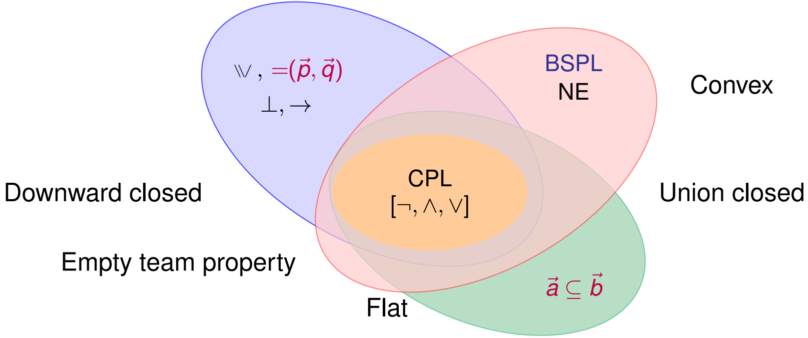



Question: What is (the expressive power of) CPL($=(\vec{p}, \vec{q}), \vec{a} \subseteq \vec{b}$)?

Background:

- First-order dependence logic, i.e., FO($=(\vec{x}, \vec{y})$), captures all downward closed team properties definable in existential second-order logic (ESO) (modulo $\emptyset$) (Kontinen, Väänänen 2009 + 2011 erratum about $\emptyset$).
- FO($=(\vec{x}, \vec{y}), \vec{x} \subseteq \vec{y}$) captures all ESO-team properties (modulo ) (Galliani 2012)

# More expressively complete logics?




Question: What is (the expressive power of) CPL($=(\vec{p}, \vec{q}), \vec{a} \subseteq \vec{b}$)?

Background:

- First-order dependence logic, i.e., FO($=(\vec{x}, \vec{y})$), captures all downward closed team properties definable in existential second-order logic (ESO) (modulo $\emptyset$) (Kontinen, Väänänen 2009 + 2011 erratum about $\emptyset$).
- FO($=(\vec{x}, \vec{y}), \vec{x} \subseteq \vec{y}$) captures all ESO-team properties (modulo $\emptyset$) (Galliani 2012)

# Expressive completeness and conjunctive normal form

**Theorem**

*The logic* $[\neg, \wedge, \vee, =\!(\cdot), \subseteq]$ *is expressively complete over all team properties that contain the empty team.*

**Theorem**

*The logic* $[\neg, \wedge, \vee, =\!(\cdot), \subseteq]$ *is expressively complete over all team properties that contain the empty team.*

Proof. The trivial direction: $\varnothing \models \psi$ for all $\psi \in [\neg, \wedge, \vee, =\!(\cdot), \subseteq]$.

**Theorem**

*The logic* $[\neg, \wedge, \vee, =\!(\cdot), \subseteq]$ *is expressively complete over all team properties that contain the empty team.*

Proof. The trivial direction: $\emptyset \models \psi$ for all $\psi \in [\neg, \wedge, \vee, =\!(\cdot), \subseteq]$.

The nontrivial direction: For any *N*-team property P that contains the empty team, construct a formula $\phi \in [\neg, \wedge, \vee, =\!(\cdot), \subseteq]$ such that $[\![\phi]\!] = \mathsf{P}$, i.e., $s \models \phi$ iff $s \in \mathsf{P}$.

**Theorem**

*The logic $[\neg, \wedge, \vee, =\!(\cdot), \subseteq]$ is expressively complete over all team properties that contain the empty team.*

Proof. The trivial direction: $\emptyset \models \psi$ for all $\psi \in [\neg, \wedge, \vee, =\!(\cdot), \subseteq]$.

The nontrivial direction: For any $N$-team property P that contains the empty team, construct a formula $\phi \in [\neg, \wedge, \vee, =\!(\cdot), \subseteq]$ such that $[\![\phi]\!] = \mathsf{P}$, i.e., $s \models \phi$ iff $s \in \mathsf{P}$. The formula $\phi = \bigwedge_{t \in \overline{\mathsf{P}}} \phi_t$ in CNF says

"the current team $s$ is not any team in $\overline{\mathsf{P}}$".

**Theorem**

*The logic $[\neg, \wedge, \vee, =(\cdot), \subseteq]$ is expressively complete over all team properties that contain the empty team.*

Proof. The trivial direction: $\emptyset \models \psi$ for all $\psi \in [\neg, \wedge, \vee, =(\cdot), \subseteq]$.

The nontrivial direction: For any $N$-team property P that contains the empty team, construct a formula $\phi \in [\neg, \wedge, \vee, =(\cdot), \subseteq]$ such that $[\![\phi]\!] = \mathsf{P}$, i.e., $s \models \phi$ iff $s \in \mathsf{P}$. The formula $\phi = \bigwedge_{t \in \overline{\mathsf{P}}} \phi_t$ in CNF says

"the current team $s$ is not any team in $\overline{\mathsf{P}}$".

To be precise:

Prop. For any $N$-team $t \neq \emptyset$, there is a formula $\phi_t$ s.t. for any $N$-team $s$,

$$s \models \phi_t \iff s \neq t$$

# Saying "no" in $[\neg, \wedge, \vee, =\!(\cdot), \subseteq]$

**Prop.** For any $N$-team $t \neq \emptyset$, there is a formula $\phi_t$ s.t. for any $N$-team $s$,

$$s \models \phi_t \iff s \neq t$$

Prop. For any $N$-team $t \neq \emptyset$, there is a formula $\phi_t$ s.t. for any $N$-team $s$,

$$s \models \phi_t \iff s \neq t, \text{ i.e., either } s \not\subseteq t \text{ or } t \not\subseteq s.$$

Prop. For any $N$-team $t \neq \emptyset$, there is a formula $\phi_t$ s.t. for any $N$-team $s$,

$$s \models \phi_t \iff s \neq t, \text{ i.e., either } s \nsubseteq t \text{ or } t \nsubseteq s.$$

Take $\phi_t = \sigma_t \vee \rho_t$.

**Prop.** For any $N$-team $t \neq \emptyset$, there is a formula $\phi_t$ s.t. for any $N$-team $s$,

$$s \models \phi_t \iff s \neq t, \text{ i.e., either } s \nsubseteq t \text{ or } t \nsubseteq s.$$

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

$$s \models \sigma_t \iff s \nsubseteq t.$$

**Lm 2.** For any $N$-team $t \neq \emptyset$, there is a formula $\rho_t$ s.t. for any $N$-team $s$,

$$s \models \rho_t \iff t \nsubseteq s.$$

Take $\phi_t = \sigma_t \vee \rho_t$.

**Prop.** For any $N$-team $t \neq \emptyset$, there is a formula $\phi_t$ s.t. for any $N$-team $s$,

$$s \models \phi_t \iff s \neq t, \text{ i.e., either } s \not\subseteq t \text{ or } t \not\subseteq s.$$

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

$$s \models \sigma_t \iff s \not\subseteq t.$$

**Lm 2.** For any $N$-team $t \neq \emptyset$, there is a formula $\rho_t$ s.t. for any $N$-team $s$,

$$s \models \rho_t \iff t \not\subseteq s.$$

*Proof.* By (Huuskonen 2016).

Take $\phi_t = \sigma_t \vee \rho_t$.

# Saying “no” in [$\neg, \wedge, \vee, =(\cdot), \subseteq$], and “the empty set test”

**Prop.** For any $N$-team $t \neq \emptyset$, there is a formula $\phi_t$ s.t. for any $N$-team $s$,

$$s \models \phi_t \iff s \neq t, \text{ i.e., either } s \nsubseteq t \text{ or } t \nsubseteq s.$$

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

$$s \models \sigma_t \iff s \nsubseteq t \text{ or } s = \emptyset.$$

**Lm 2.** For any $N$-team $t \neq \emptyset$, there is a formula $\rho_t$ s.t. for any $N$-team $s$,

$$s \models \rho_t \iff t \nsubseteq s.$$

**Proof.** By (Huuskonen 2016).

Take $\phi_t = \sigma_t \vee \rho_t$.

# Saying “no” in $[\neg, \wedge, \vee, =\!(\cdot), \subseteq]$, and “the empty set test”

**Prop.** For any $N$-team $t \neq \emptyset$, there is a formula $\phi_t$ s.t. for any $N$-team $s$,

$$s \models \phi_t \iff s \neq t, \text{ i.e., either } s \nsubseteq t \text{ or } t \nsubseteq s.$$

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

$$s \models \sigma_t \iff s \nsubseteq t \text{ or } s = \emptyset.$$

**Lm 2.** For any $N$-team $t \neq \emptyset$, there is a formula $\rho_t$ s.t. for any $N$-team $s$,

$$s \models \rho_t \iff t \nsubseteq s.$$


**Proof.** By (Huuskonen 2016).

Take $\phi_t = \sigma_t \vee \rho_t$.

# Saying "no" in $[\neg, \wedge, \vee, =\!(\cdot), \subseteq]$

Lm 1. For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

$$s \models \sigma_t \iff s \not\subseteq t \text{ or } s = \emptyset.$$

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

$$s \models \sigma_t \iff s \nsubseteq t \text{ or } s = \emptyset.$$

**Proof.** Recall that for any $v : N \to \{0, 1\}$,

$$r \models \underline{v(p_1)} \dots \underline{v(p_n)} \subseteq p_1 \dots p_n \iff v \in r \quad \text{or} \quad r = \emptyset,$$

| | $p_1$ | $p_2$ | $p_3$ | |
|---|---|---|---|---|
| | 1 | 0 | 1 | |
| $v$ | 0 | 1 | 0 | $\bot\top\bot \subseteq p_1p_2p_3$ |
| | 1 | 1 | 1 | |
| | 0 | 1 | 1 | |

# Saying "no" in $[\neg, \wedge, \vee, =\!(\cdot), \subseteq]$

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

$$s \models \sigma_t \iff s \not\subseteq t \text{ or } s = \varnothing.$$

**Proof.** Recall that for any $v : N \to \{0, 1\}$,

$$r \models \underline{v(p_1)} \dots \underline{v(p_n)} \subseteq p_1 \dots p_n \iff v \in r \;\text{ or }\; r = \varnothing,$$

Take

$$\sigma_t = \bigvee_{v \notin t} \underline{v(p_1)} \dots \underline{v(p_n)} \subseteq p_1 \dots p_n.$$

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

$$s \models \sigma_t \iff s \nsubseteq t \text{ or } s = \varnothing.$$

**Proof.** Recall that for any $v : N \to \{0,1\}$,

$$r \models \underline{v(p_1)} \ldots \underline{v(p_n)} \subseteq p_1 \ldots p_n \iff v \in r \text{ or } r = \varnothing,$$

Take

$$\sigma_t = \bigvee_{v \notin t} \underline{v(p_1)} \ldots \underline{v(p_n)} \subseteq p_1 \ldots p_n.$$

"$\Longleftarrow$": If $\varnothing \neq s \nsubseteq t$, then there is $v$ s.t. $v \in s$ and $v \notin t$.

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

$$s \models \sigma_t \iff s \nsubseteq t \text{ or } s = \emptyset.$$

**Proof.** Recall that for any $v : N \to \{0,1\}$,

$$r \models \underline{v(p_1)} \dots \underline{v(p_n)} \subseteq p_1 \dots p_n \iff v \in r \text{ or } r = \emptyset,$$

Take

$$\sigma_t = \bigvee_{v \notin t} \underline{v(p_1)} \dots \underline{v(p_n)} \subseteq p_1 \dots p_n.$$

"$\Longleftarrow$": If $\emptyset \neq s \nsubseteq t$, then there is $v$ s.t. $v \in s$ and $v \notin t$. Thus,

$$s \models \underline{v(p_1)} \dots \underline{v(p_n)} \subseteq p_1 \dots p_n \text{ and } s \models \sigma_t.$$

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

$$s \models \sigma_t \iff s \nsubseteq t \text{ or } s = \varnothing.$$

**Proof.** Recall that for any $v : N \to \{0, 1\}$,

$$r \models \underline{v(p_1)} \ldots \underline{v(p_n)} \subseteq p_1 \ldots p_n \iff v \in r \text{ or } r = \varnothing,$$

Take

$$\sigma_t = \bigvee_{v \notin t} \underline{v(p_1)} \ldots \underline{v(p_n)} \subseteq p_1 \ldots p_n.$$

“$\Longleftarrow$”: If $\varnothing \neq s \nsubseteq t$, then there is $v$ s.t. $v \in s$ and $v \notin t$. Thus,

$$s \models \underline{v(p_1)} \ldots \underline{v(p_n)} \subseteq p_1 \ldots p_n \text{ and } s \models \sigma_t.$$

“$\Longrightarrow$”: If $\varnothing \neq s \models \sigma_t$, then $s = \bigcup_{v \notin t} s_v$ and $s_v \models \underline{v(p_1)} \ldots \underline{v(p_n)} \subseteq p_1 \ldots p_n$.

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

$$s \models \sigma_t \iff s \nsubseteq t \text{ or } s = \varnothing.$$

**Proof.** Recall that for any $v : N \to \{0,1\}$,

$$r \models \underline{v(p_1)} \dots \underline{v(p_n)} \subseteq p_1 \dots p_n \iff v \in r \text{ or } r = \varnothing,$$

Take

$$\sigma_t = \bigvee_{v \notin t} \underline{v(p_1)} \dots \underline{v(p_n)} \subseteq p_1 \dots p_n.$$

"$\Longleftarrow$": If $\varnothing \neq s \nsubseteq t$, then there is $v$ s.t. $v \in s$ and $v \notin t$. Thus,

$$s \models \underline{v(p_1)} \dots \underline{v(p_n)} \subseteq p_1 \dots p_n \text{ and } s \models \sigma_t.$$

"$\Longrightarrow$": If $\varnothing \neq s \models \sigma_t$, then $s = \bigcup_{v \notin t} s_v$ and $s_v \models \underline{v(p_1)} \dots \underline{v(p_n)} \subseteq p_1 \dots p_n$.
At least one such $s_v$ must be nonempty and satisfies $v \in s_v \subseteq s$.

# Saying “no” in $[\neg, \wedge, \vee, =(\cdot), \subseteq]$

Lm 1. For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

$$s \models \sigma_t \iff s \not\subseteq t \text{ or } s = \varnothing.$$

Proof. Recall that for any $v : N \to \{0, 1\}$,

$$r \models \underline{v(p_1)} \dots \underline{v(p_n)} \subseteq p_1 \dots p_n \iff v \in r \ \text{ or } \ r = \varnothing,$$

Take

$$\sigma_t = \bigvee_{v \notin t} \underline{v(p_1)} \dots \underline{v(p_n)} \subseteq p_1 \dots p_n.$$

“$\Longleftarrow$”: If $\varnothing \neq s \not\subseteq t$, then there is $v$ s.t. $v \in s$ and $v \notin t$. Thus,

$$s \models \underline{v(p_1)} \dots \underline{v(p_n)} \subseteq p_1 \dots p_n \ \text{ and } \ s \models \sigma_t.$$

“$\Longrightarrow$”: If $\varnothing \neq s \models \sigma_t$, then $s = \bigcup_{v \notin t} s_v$ and $s_v \models \underline{v(p_1)} \dots \underline{v(p_n)} \subseteq p_1 \dots p_n$. At least one such $s_v$ must be nonempty and satisfies $v \in s_v \subseteq s$. We have $v \in s \setminus t$, thereby $s \not\subseteq t$. □

**Prop.** For any $N$-team $t \neq \emptyset$, there is a formula $\phi_t$ s.t. for any $N$-team $s$,

$$s \models \phi_t \iff s \neq t, \text{ i.e., either } s \not\subseteq t \text{ or } t \not\subseteq s.$$

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

$$s \models \sigma_t \iff s \not\subseteq t \text{ or } s = \emptyset.$$

**Lm 2.** For any $N$-team $t \neq \emptyset$, there is a formula $\rho_t$ s.t. for any $N$-team $s$,

$$s \models \rho_t \iff t \not\subseteq s.$$

Take $\phi_t = \sigma_t \vee \rho_t$.

**Prop.** For any $N$-team $t \neq \emptyset$, there is a formula $\phi_t$ s.t. for any $N$-team $s$,

$$s \models \phi_t \iff s \neq t, \text{ i.e., either } s \not\subseteq t \text{ or } t \not\subseteq s.$$

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

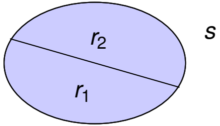
$$s \models \sigma_t \iff s \not\subseteq t \text{ or } s = \emptyset.$$

**Lm 2.** For any $N$-team $t \neq \emptyset$, there is a formula $\rho_t$ s.t. for any $N$-team $s$,

$$s \models \rho_t \iff t \not\subseteq s.$$

Take $\phi_t = \sigma_t \vee \rho_t$.

**Poof of Prop.**
Suppose $s \models \phi_t$.
Then $s = r_1 \cup r_2$...



# Saying "no" in $[\neg, \wedge, \vee, =\!(\cdot), \subseteq]$, and "the empty set test"

**Prop.** For any $N$-team $t \neq \emptyset$, there is a formula $\phi_t$ s.t. for any $N$-team $s$,

$$s \models \phi_t \iff s \neq t, \text{ i.e., either } s \not\subseteq t \text{ or } t \not\subseteq s.$$

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

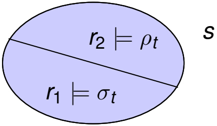
$$s \models \sigma_t \iff s \not\subseteq t \text{ or } s = \emptyset.$$

**Lm 2.** For any $N$-team $t \neq \emptyset$, there is a formula $\rho_t$ s.t. for any $N$-team $s$,

$$s \models \rho_t \iff t \not\subseteq s.$$

Take $\phi_t = \sigma_t \vee \rho_t$.

**Poof of Prop.**
Suppose $s \models \phi_t$.
Then $s = r_1 \cup r_2$...



**Prop.** For any $N$-team $t \neq \emptyset$, there is a formula $\phi_t$ s.t. for any $N$-team $s$,

$$s \models \phi_t \iff s \neq t, \text{ i.e., either } s \not\subseteq t \text{ or } t \not\subseteq s.$$

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

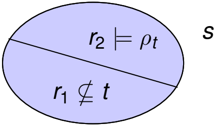
$$s \models \sigma_t \iff s \not\subseteq t \text{ or } s = \emptyset.$$

**Lm 2.** For any $N$-team $t \neq \emptyset$, there is a formula $\rho_t$ s.t. for any $N$-team $s$,

$$s \models \rho_t \iff t \not\subseteq s.$$

Take $\phi_t = \sigma_t \vee \rho_t$.

**Poof of Prop.**
Suppose $s \models \phi_t$.
Then $s = r_1 \cup r_2$...



**Prop.** For any $N$-team $t \neq \emptyset$, there is a formula $\phi_t$ s.t. for any $N$-team $s$,

$$s \models \phi_t \iff s \neq t, \text{ i.e., either } s \not\subseteq t \text{ or } t \not\subseteq s.$$

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

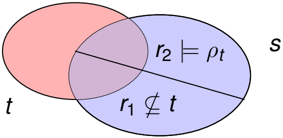
$$s \models \sigma_t \iff s \not\subseteq t \text{ or } s = \emptyset.$$

**Lm 2.** For any $N$-team $t \neq \emptyset$, there is a formula $\rho_t$ s.t. for any $N$-team $s$,

$$s \models \rho_t \iff t \not\subseteq s.$$

Take $\phi_t = \sigma_t \vee \rho_t$.

Poof of Prop.
Suppose $s \models \phi_t$.
Then $s = r_1 \cup r_2$...



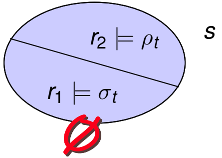
**Prop.** For any $N$-team $t \neq \emptyset$, there is a formula $\phi_t$ s.t. for any $N$-team $s$,
$$s \models \phi_t \iff s \neq t, \text{ i.e., either } s \not\subseteq t \text{ or } t \not\subseteq s.$$

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,
$$s \models \sigma_t \iff s \not\subseteq t \text{ or } s = \emptyset.$$

**Lm 2.** For any $N$-team $t \neq \emptyset$, there is a formula $\rho_t$ s.t. for any $N$-team $s$,
$$s \models \rho_t \iff t \not\subseteq s.$$

Take $\phi_t = \sigma_t \vee \rho_t$.

**Poof of Prop.**
Suppose $s \models \phi_t$.
Then $s = r_1 \cup r_2$...



# Saying "no" in $[\neg, \wedge, \vee, =(\cdot), \subseteq]$, and "the empty set test"

**Prop.** For any $N$-team $t \neq \emptyset$, there is a formula $\phi_t$ s.t. for any $N$-team $s$,

$$s \models \phi_t \iff s \neq t, \text{ i.e., either } s \not\subseteq t \text{ or } t \not\subseteq s.$$

**Lm 1.** For any $N$-team $t \neq \emptyset$, there is a formula $\sigma_t$ s.t. for any $N$-team $s$,

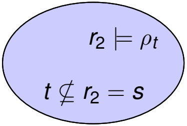
$$s \models \sigma_t \iff s \not\subseteq t \text{ or } s = \emptyset.$$

**Lm 2.** For any $N$-team $t \neq \emptyset$, there is a formula $\rho_t$ s.t. for any $N$-team $s$,

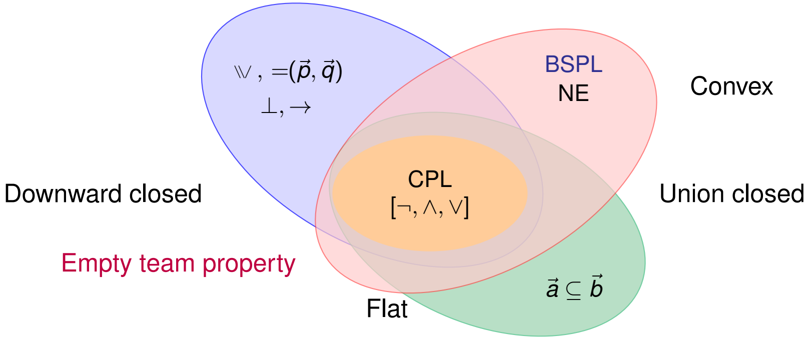
$$s \models \rho_t \iff t \not\subseteq s.$$

Take $\phi_t = \sigma_t \vee \rho_t$.

**Poof of Prop.**
Suppose $s \models \phi_t$.
Then $s = r_1 \cup r_2$...




$s = r_2$




Summary:

- We have discussed a number of expressively complete propositional team-based logics around BSPL (that have the empty team property).
- These results can also be generalized to the modal logic setting.
- (More) applications?