Projective Unification in Intermediate and Modal Predicate Logics

Wojciech Dzik

Institute of Mathematics, Silesian University, Katowice, Poland, wojciech.dzik@us.edu.pl

Piotr Wojtylak,

Institute of Mathematics and Computer Science, University of Opole, Opole, Poland, wojtylak@math.uni.opole.pl

Projective Unification in Intermediate and Modal Predicate Logics

Wojciech Dzik

Institute of Mathematics, Silesian University, Katowice, Poland, wojciech.dzik@us.edu.pl

Piotr Wojtylak,

Institute of Mathematics and Computer Science, University of Opole, Opole, Poland, wojtylak@math.uni.opole.pl

Topology, Algebra, and Categories in Logic 2015,
Ischia (Italy) 21 - 26 June 2015

- Unification, unifiers and projective unifiers in logic

Overview

- Unification, unifiers and projective unifiers in logic
- The scope of projective unification in propositional logics

Overview

- Unification, unifiers and projective unifiers in logic
- The scope of projective unification in propositional logics
- Applications to (Almost) Structural Completeness and Admissible Rules

Overview

- Unification, unifiers and projective unifiers in logic
- The scope of projective unification in propositional logics
- Applications to (Almost) Structural Completeness and Admissible Rules
- The scope of projective unification in predicate logics:

Overview

- Unification, unifiers and projective unifiers in logic
- The scope of projective unification in propositional logics
- Applications to (Almost) Structural Completeness and Admissible Rules
- The scope of projective unification in predicate logics:
    - An intermediate predicate logic L enjoys projective unification iff L extends IP.QLC - Gödel-Dummett predicate logic with Independence of Premises

Overview

- Unification, unifiers and projective unifiers in logic
- The scope of projective unification in propositional logics
- Applications to (Almost) Structural Completeness and Admissible Rules
- The scope of projective unification in predicate logics:
  - An intermediate predicate logic L enjoys projective unification iff L extends IP.QLC - Gödel-Dummett predicate logic with Independence of Premises
  - A modal predicate logic L extending $\mathsf{QS4}_=$ enjoys projective unification iff L extends $\Box\mathsf{IP.QS4.3}_=$ - modal predicate logic $\mathsf{QS4.3}_=$ with Modal Independence of Premises

Projective unifiers

A substitution ε is called a *unifier* for a formula A in a logic L if

$$\vdash_L \varepsilon(A) \qquad \text{(or equivalently if } \varepsilon(A) \in L\text{)}$$

A substitution ε is called a *unifier* for a formula A in a logic L if

$$\vdash_L \varepsilon(A) \qquad\qquad \text{(or equivalently if } \varepsilon(A) \in L\text{)}$$

Unifiers $\varepsilon\colon Fm \to \{\bot, \top\}$ are called *ground unifiers*.

A substitution ε is called a *unifier* for a formula A in a logic L if

$$\vdash_L \varepsilon(A) \qquad \text{(or equivalently if } \varepsilon(A) \in L)$$

Unifiers $\varepsilon\colon Fm \to \{\bot, \top\}$ are called *ground unifiers*.

A substitution ε is said to be *projective for A* in L if

$$A \vdash_L B \leftrightarrow \varepsilon(B), \quad \text{for each } B$$

S.Ghilardi: projective - formula, -unifier, $\mathfrak{F}_n/_{[A)}$ projective alg.

A substitution ε is called a *unifier* for a formula A in a logic L if

$$\vdash_L \varepsilon(A) \qquad \text{(or equivalently if } \varepsilon(A) \in L\text{)}$$

Unifiers $\varepsilon \colon Fm \to \{\bot, \top\}$ are called *ground unifiers*.

A substitution ε is said to be *projective for A* in L if

$$A \vdash_L B \leftrightarrow \varepsilon(B), \quad \text{for each } B$$

S.Ghilardi: projective - formula, -unifier, $\mathfrak{F}_n/_{[A)}$ projective alg.

We say that the logic L has *projective unification* if each unifiable formula has a projective unifier.

Ex. Classical PC, Modal S5, NExt S4.3

Applications: (A)SC, Admissible rules.

A schematic (structural) rule $r \; : \; \alpha_1, \ldots, \alpha_n, /\beta$ is *admissible* in **L**, if adding r does not change **L**, i.e. for every substitution τ:

$\{\tau(\alpha_1), \ldots, \tau(\alpha_n)\} \subseteq \mathbf{L} \; \Rightarrow \; \tau(\beta) \in \mathbf{L}$

Applications: (A)SC, Admissible rules.

A schematic (structural) rule $r\ :\ \alpha_1, \ldots, \alpha_n, /\beta$ is *admissible* in **L**, if adding r does not change **L**, i.e. for every substitution τ:

$\{\tau(\alpha_1), \ldots, \tau(\alpha_n)\} \subseteq \mathbf{L} \ \Rightarrow\ \tau(\beta) \in \mathbf{L}$

r is *derivable* in **L**, if $\quad \alpha_1, \ldots, \alpha_n \vdash_{\mathbf{L}} \beta$.

Applications: (A)SC, Admissible rules.

A schematic (structural) rule $r\ :\ \alpha_1, \ldots, \alpha_n, /\beta$ is *admissible* in **L**, if adding r does not change **L**, i.e. for every substitution τ:

$\{\tau(\alpha_1), \ldots, \tau(\alpha_n)\} \subseteq \mathbf{L} \ \Rightarrow\ \tau(\beta) \in \mathbf{L}$

r is *derivable* in **L**, if $\quad \alpha_1, \ldots, \alpha_n \vdash_{\mathbf{L}} \beta$.

A logic **L** is *Structurally Complete, SC*, if every admissible rule in **L** is also derivable in **L**; (Class PC, LC, $Int^{\rightarrow}$, Medvedev L.)

Applications: (A)SC, Admissible rules.

A schematic (structural) rule $r : \alpha_1, \ldots, \alpha_n, /\beta$ is *admissible* in **L**, if adding r does not change **L**, i.e. for every substitution τ:

$\{\tau(\alpha_1), \ldots, \tau(\alpha_n)\} \subseteq \mathbf{L} \Rightarrow \tau(\beta) \in \mathbf{L}$

r is *derivable* in **L**, if $\alpha_1, \ldots, \alpha_n \vdash_{\mathbf{L}} \beta$.

A logic **L** is *Structurally Complete, SC*, if every admissible rule in **L** is also derivable in **L**; (Class PC, LC, $Int^{\rightarrow}$, Medvedev L.)

$r : \alpha_1, \ldots \alpha_1/\beta$ is *passive* in **L**, if for every substitution τ: $\{\tau(\alpha_1), \ldots \tau(\alpha_n)\} \not\subseteq \mathbf{L}$, i.e. the premises are not unifiable in **L**.

Applications: (A)SC, Admissible rules.

A schematic (structural) rule $r : \alpha_1, \ldots, \alpha_n, /\beta$ is *admissible* in **L**, if adding r does not change **L**, i.e. for every substitution τ:

$\{\tau(\alpha_1), \ldots, \tau(\alpha_n)\} \subseteq \mathbf{L} \Rightarrow \tau(\beta) \in \mathbf{L}$

r is *derivable* in **L**, if $\alpha_1, \ldots, \alpha_n \vdash_{\mathbf{L}} \beta$.

A logic **L** is *Structurally Complete, SC*, if every admissible rule in **L** is also derivable in **L**; (Class PC, LC, $Int^{\rightarrow}$, Medvedev L.)

$r : \alpha_1, \ldots \alpha_1/\beta$ is *passive* in **L**, if for every substitution τ: $\{\tau(\alpha_1), \ldots \tau(\alpha_n)\} \not\subseteq \mathbf{L}$, i.e. the premises are not unifiable in **L**.

EX. $P_2 : \quad \Diamond p \wedge \Diamond\neg p/\bot$ is passive in S4 and its extensions,

Applications: (A)SC, Admissible rules.

A schematic (structural) rule $r : \alpha_1, \ldots, \alpha_n, /\beta$ is *admissible* in **L**, if adding r does not change **L**, i.e. for every substitution τ:

$\{\tau(\alpha_1), \ldots, \tau(\alpha_n)\} \subseteq \mathbf{L} \Rightarrow \tau(\beta) \in \mathbf{L}$

r is *derivable* in **L**, if $\alpha_1, \ldots, \alpha_n \vdash_{\mathbf{L}} \beta$.

A logic **L** is *Structurally Complete, SC*, if every admissible rule in **L** is also derivable in **L**; (Class PC, LC, $Int^{\rightarrow}$, Medvedev L.)

$r : \alpha_1, \ldots \alpha_1/\beta$ is *passive* in **L**, if for every substitution τ: $\{\tau(\alpha_1), \ldots \tau(\alpha_n)\} \not\subseteq \mathbf{L}$, i.e. the premises are not unifiable in **L**.

EX. $P_2 : \quad \Diamond p \wedge \Diamond \neg p/\bot$ is passive in S4 and its extensions,

L is *Almost Structurally Complete, ASC*, if every admissible rule which is not passive in **L** is also derivable in **L**; (NExt S4.3, $Ł_n$)

Applications: (A)SC, Admissible rules.

A schematic (structural) rule $r : \alpha_1, \ldots, \alpha_n, /\beta$ is *admissible* in **L**, if adding r does not change **L**, i.e. for every substitution τ:

$\{\tau(\alpha_1), \ldots, \tau(\alpha_n)\} \subseteq \mathbf{L} \ \Rightarrow \ \tau(\beta) \in \mathbf{L}$

r is *derivable* in **L**, if $\quad \alpha_1, \ldots, \alpha_n \vdash_{\mathbf{L}} \beta$.

A logic **L** is *Structurally Complete, SC*, if every admissible rule in **L** is also derivable in **L**; (Class PC, LC, $Int^{\rightarrow}$, Medvedev L.)

$r : \alpha_1, \ldots \alpha_1/\beta$ is *passive* in **L**, if for every substitution τ: $\{\tau(\alpha_1), \ldots \tau(\alpha_n)\} \not\subseteq \mathbf{L}$, i.e. the premises are not unifiable in **L**.

EX. $P_2 : \quad \Diamond p \wedge \Diamond \neg p/\bot$ is passive in S4 and its extensions,

L is *Almost Structurally Complete, ASC*, if every admissible rule which is not passive in **L** is also derivable in **L**; (NExt S4.3, $Ł_n$)

FACT: L has projective unification $\Rightarrow$ L is (Almost) Structurally Complete,

Applications: (A)SC, Admissible rules.

A schematic (structural) rule $r\ :\ \alpha_1, \ldots, \alpha_n, /\beta$ is *admissible* in **L**, if adding r does not change **L**, i.e. for every substitution τ:

$\{\tau(\alpha_1), \ldots, \tau(\alpha_n)\} \subseteq \mathbf{L} \ \Rightarrow\ \tau(\beta) \in \mathbf{L}$

r is *derivable* in **L**, if $\quad \alpha_1, \ldots, \alpha_n \vdash_{\mathbf{L}} \beta$.

A logic **L** is *Structurally Complete, SC*, if every admissible rule in **L** is also derivable in **L**; (Class PC, LC, $Int^{\rightarrow}$, Medvedev L.)

$r\ :\ \alpha_1, \ldots \alpha_1/\beta$ is *passive* in **L**, if for every substitution τ: $\{\tau(\alpha_1), \ldots \tau(\alpha_n)\} \not\subseteq \mathbf{L}$, i.e. the premises are not unifiable in **L**.

EX. $P_2 : \quad \Diamond p \wedge \Diamond\neg p/\bot$ is passive in S4 and its extensions,

L is *Almost Structurally Complete, ASC*, if every admissible rule which is not passive in **L** is also derivable in **L**; (NExt S4.3, $Ł_n$)

FACT: L has projective unification $\Rightarrow$ L is (Almost) Structurally Complete,
A description of admissible rules

The scope of projective unification

The scope of projective unification

Theorem (Wronski 1995 - 2008)

*An intermediate logic $L \supseteq$ **Int** enjoys projective unification iff .$(y \Rightarrow z) \vee (z \Rightarrow y) \in L$, iff **LC** $\subseteq L$ iff $\vee$ L definable $(\wedge, \rightarrow)$.*

Corollary

*Every logic $L \supseteq$ **LC** is SC.*

Theorem (Wronski 1995 - 2008)

An intermediate logic $L \supseteq \mathbf{Int}$ *enjoys projective unification iff* $.(y \Rightarrow z) \vee (z \Rightarrow y) \in L$, *iff* $\mathbf{LC} \subseteq L$ *iff* $\vee$ L *definable* $(\wedge, \rightarrow)$.

Corollary

Every logic $L \supseteq \mathbf{LC}$ *is SC.*

Theorem (WD, P. Wojtylak 2011)

A modal logic $L \supseteq \mathbf{S4}$ *enjoys projective unification iff* $.\Box(\Box y \rightarrow \Box z) \vee \Box(\Box z \rightarrow \Box y) \in L$, *iff* $\mathbf{S4}.3 \subseteq L$.

Corollary

Every logic $L \supseteq \mathbf{S4}.3$ *is ASC.*

The scope of projective unification

Theorem (Wronski 1995 - 2008)

An intermediate logic $L \supseteq \mathbf{Int}$ *enjoys projective unification iff*
$.(y \Rightarrow z) \vee (z \Rightarrow y) \in L$, *iff* $\mathbf{LC} \subseteq L$ *iff* $\vee$ L *definable* $(\wedge, \to)$.

Corollary

Every logic $L \supseteq \mathbf{LC}$ *is SC.*

Theorem (WD, P. Wojtylak 2011)

A modal logic $L \supseteq \mathbf{S4}$ *enjoys projective unification iff*
$.\Box(\Box y \to \Box z) \vee \Box(\Box z \to \Box y) \in L$, *iff* $\mathbf{S4.3} \subseteq L$.

Corollary

Every logic $L \supseteq \mathbf{S4.3}$ *is ASC.*

Difference: in **LC** - the method of ground unifiers works, in **S4.3** it does not: projective unifiers - compositions of Löwenheim subst.'s

1-st order modal language

We consider a first-order *modal* language without function letters.

1-st order modal language

We consider a first-order *modal* language without function letters.

free individual variables: $a_1, a_2, a_3, \ldots$

bound individual variables: $x_1, x_2, x_3, \ldots$

predicate variables: $P_1, P_2, P_3, \ldots$

1-st order modal language

We consider a first-order *modal* language without function letters.

free individual variables: $a_1, a_2, a_3, \ldots$

bound individual variables: $x_1, x_2, x_3, \ldots$

predicate variables: $P_1, P_2, P_3, \ldots$

Basic logical symbols are $\wedge, \vee, \sim, \top, \forall, \Box$ (and $=$).

1-st order modal language

We consider a first-order *modal* language without function letters.

free individual variables: $a_1, a_2, a_3, \ldots$

bound individual variables: $x_1, x_2, x_3, \ldots$

predicate variables: $P_1, P_2, P_3, \ldots$

Basic logical symbols are $\wedge, \vee, \sim, \top, \forall, \Box$ (and $=$). $\rightarrow, \leftrightarrow, \bot, \Diamond, \exists$ is defined as usually.

1-st order modal language

We consider a first-order *modal* language without function letters.

free individual variables: $a_1, a_2, a_3, \ldots$

bound individual variables: $x_1, x_2, x_3, \ldots$

predicate variables: $P_1, P_2, P_3, \ldots$

Basic logical symbols are $\wedge, \vee, \sim, \top, \forall, \Box$ (and $=$). $\rightarrow, \leftrightarrow, \bot, \Diamond, \exists$ is defined as usually.

The *intuitionistic* predicate language $\{\Rightarrow, \wedge, \vee, \bot, \bigwedge, \bigvee\}$, it will also be seen as a definitional fragment of the initial modal predicate language.

We consider a first-order *modal* language without function letters.

free individual variables: $a_1, a_2, a_3, \ldots$

bound individual variables: $x_1, x_2, x_3, \ldots$

predicate variables: $P_1, P_2, P_3, \ldots$

Basic logical symbols are $\wedge, \vee, \sim, \top, \forall, \Box$ (and $=$). $\rightarrow, \leftrightarrow, \bot, \Diamond, \exists$ is defined as usually.

The *intuitionistic* predicate language $\{\Rightarrow, \wedge, \vee, \bot, \bigwedge, \bigvee\}$, it will also be seen as a definitional fragment of the initial modal predicate language.

Let *Fm* (or *q-Fm*) denote the set of all formulas (or quasi-formulas). $\varphi \in Fm$ iff $\varphi \in q\text{-}Fm$ and bounded variables in φ do not occur free.

Substitutions for predicate variables

Substitutions ε: q-Fm $\to$ q-Fm are mappings:

$$\varepsilon(P(t_1, \ldots, t_k)) = \varepsilon(P(x_1, \ldots, x_k))[x_1/t_1, \ldots, x_k/t_k]$$

Substitutions for predicate variables

Substitutions $\varepsilon\colon \text{q-Fm} \to \text{q-Fm}$ are mappings:

$$\varepsilon(P(t_1,\ldots,t_k)) = \varepsilon(P(x_1,\ldots,x_k))[x_1/t_1,\ldots,x_k/t_k]$$

$$\varepsilon(\neg A) = \neg\varepsilon(A); \qquad \varepsilon(A \wedge B) = \varepsilon(A) \wedge \varepsilon(B);$$

$$\varepsilon(\Box A) = \Box\varepsilon(A); \qquad \varepsilon(A \vee B) = \varepsilon(A) \vee \varepsilon(B);$$

$$\varepsilon(\forall_x A) = \forall_x \varepsilon(A) \qquad \varepsilon(\exists_x A) = \exists_x \varepsilon(A)$$

Substitutions for predicate variables

Substitutions $\varepsilon\colon$ q-Fm $\to$ q-Fm are mappings:

$$\begin{array}{ll} \varepsilon(P(t_1,\dots,t_k)) = & \varepsilon(P(x_1,\dots,x_k))[x_1/t_1,\dots,x_k/t_k] \\ \varepsilon(\neg A) = \neg\varepsilon(A); & \varepsilon(A\wedge B) = \varepsilon(A)\wedge\varepsilon(B); \\ \varepsilon(\Box A) = \Box\varepsilon(A); & \varepsilon(A\vee B) = \varepsilon(A)\vee\varepsilon(B); \\ \varepsilon(\forall_x A) = \forall_x\varepsilon(A) & \varepsilon(\exists_x A) = \exists_x\varepsilon(A) \end{array}$$

Warning: $=$ is defined up to a correct renaming of bound variables in the substituted formulas.

Substitutions for predicate variables

Substitutions ε: q-Fm $\to$ q-Fm are mappings:

$$\begin{array}{ll} \varepsilon(P(t_1,\ldots,t_k)) = & \varepsilon(P(x_1,\ldots,x_k))[x_1/t_1,\ldots,x_k/t_k] \\ \varepsilon(\neg A) = \neg\varepsilon(A); & \varepsilon(A \wedge B) = \varepsilon(A) \wedge \varepsilon(B); \\ \varepsilon(\Box A) = \Box\varepsilon(A); & \varepsilon(A \vee B) = \varepsilon(A) \vee \varepsilon(B); \\ \varepsilon(\forall_x A) = \forall_x \varepsilon(A) & \varepsilon(\exists_x A) = \exists_x \varepsilon(A) \end{array}$$

Warning: $=$ is defined up to a correct renaming of bound variables in the substituted formulas.

• Pogorzelski, W.A., Prucnal, T., *Structural completeness of the first-order predicate calculus*, Zeitschrift für, Mathematische Logik und Grundlagen der Mathematik, 21 (1975), 315-320.

$$vf(\varepsilon(A)) \subseteq vf(A)$$

Substitutions for predicate variables

Substitutions $\varepsilon\colon$ q-Fm $\to$ q-Fm are mappings:

$$\varepsilon(P(t_1,\ldots,t_k)) = \varepsilon(P(x_1,\ldots,x_k))[x_1/t_1,\ldots,x_k/t_k]$$

$$\varepsilon(\neg A) = \neg\varepsilon(A); \qquad \varepsilon(A\wedge B) = \varepsilon(A)\wedge\varepsilon(B);$$

$$\varepsilon(\Box A) = \Box\varepsilon(A); \qquad \varepsilon(A\vee B) = \varepsilon(A)\vee\varepsilon(B);$$

$$\varepsilon(\forall_x A) = \forall_x\varepsilon(A) \qquad \varepsilon(\exists_x A) = \exists_x\varepsilon(A)$$

Warning: $=$ is defined up to a correct renaming of bound variables in the substituted formulas.

• Pogorzelski, W.A., Prucnal, T., *Structural completeness of the first-order predicate calculus*, Zeitschrift für, Mathematische Logik und Grundlagen der Mathematik, 21 (1975), 315-320.

$$vf(\varepsilon(A)) \subseteq vf(A)$$

•Church, A., *Introduction to Mathematical Logic I*, Princeton University Press , Princeton, New Jersey (1956)

A *predicate modal logic* is any set $L \subseteq Fm$ containing (all classical propositional tautologies, and) the predicate and modal axioms:

$$\begin{array}{l} \forall_x(A \to B(x)) \to (A \to \forall_x B(x)) \\ \forall_x A(x) \to A[x/t] \\ \Box(A \to B) \to (\Box A \to \Box B), \end{array}$$

closed under the following inferential rules

$$MP : \frac{A \to B, A}{B} \quad \text{and} \quad RN : \frac{A}{\Box A} \quad \text{and} \quad RG : \frac{A(a)}{\forall_x A(x)}$$

and closed under substitutions.

Theorem

Each unifiable formula A has a projective unifier in **S5***:*

$$\varepsilon(B) = \begin{cases} \Box A \to B & \text{if} \quad v(B) = \top \\ \Box A \wedge B & \text{if} \quad v(B) = \bot \end{cases}$$

where v is a ground unifier for A.

Theorem

Each unifiable formula A has a projective unifier in **S5***:*

$$\varepsilon(B) = \begin{cases} \Box A \to B & \text{if} \quad v(B) = \top \\ \Box A \land B & \text{if} \quad v(B) = \bot \end{cases}$$

where v is a ground unifier for A.

Such substitutions are called *Löwenheim substitutions* for A.

Theorem

Each unifiable formula A has a projective unifier in **S5***:*

$$\varepsilon(B) = \begin{cases} \Box A \rightarrow B & \text{if} \quad v(B) = \top \\ \Box A \wedge B & \text{if} \quad v(B) = \bot \end{cases}$$

where v is a ground unifier for A.

Such substitutions are called *Löwenheim substitutions* for A.

Theorem

*Each unifiable formula A has a projective unifier in Q-***S5***:*

$$\varepsilon(B) = \begin{cases} \Box\forall_{\overline{x}} A(\overline{x}) \rightarrow B & \text{if} \quad v(B) = \top \\ \Box\forall_{\overline{x}} A(\overline{x}) \wedge B & \text{if} \quad v(B) = \bot \end{cases}.$$

where v is a ground unifier for A.

- Dzik, W. *On Structural completeness of some nonclassical predicate calculi*, RML 5, 1975, pp.19-26., For logics weaker than **S5** the above ‘ground unifier’ method must be modified.

For logics weaker than **S5** the above 'ground unifier' method must be modified. Following S.Ghilardi,

• Ghilardi S.,*Best solving modal equations*, Annals of Pure and Applied Logic 102 (2000), 183–198.

one can define a sequence $\varepsilon_1, \ldots, \varepsilon_n$ of Löwenheim substitutions for A and take their composition $\varepsilon = \varepsilon_1 \circ \cdots \circ \varepsilon_n$ as a unifier for A.

For logics weaker than **S5** the above ‘ground unifier’ method must be modified. Following S.Ghilardi,

• Ghilardi S.,*Best solving modal equations*, Annals of Pure and Applied Logic 102 (2000), 183–198.

one can define a sequence $\varepsilon_1, \ldots, \varepsilon_n$ of Löwenheim substitutions for A and take their composition $\varepsilon = \varepsilon_1 \circ \cdots \circ \varepsilon_n$ as a unifier for A. It is not quite clear how many substitutions have to be taken there

For logics weaker than **S5** the above 'ground unifier' method must be modified. Following S.Ghilardi,

• Ghilardi S.,*Best solving modal equations*, Annals of Pure and Applied Logic 102 (2000), 183–198.

one can define a sequence $\varepsilon_1, \ldots, \varepsilon_n$ of Löwenheim substitutions for A and take their composition $\varepsilon = \varepsilon_1 \circ \cdots \circ \varepsilon_n$ as a unifier for A. It is not quite clear how many substitutions have to be taken there and in which order they should be arranged in the composition.

For logics weaker than **S5** the above 'ground unifier' method must be modified. Following S.Ghilardi,

• Ghilardi S.,*Best solving modal equations*, Annals of Pure and Applied Logic 102 (2000), 183–198.

one can define a sequence $\varepsilon_1, \ldots, \varepsilon_n$ of Löwenheim substitutions for A and take their composition $\varepsilon = \varepsilon_1 \circ \cdots \circ \varepsilon_n$ as a unifier for A. It is not quite clear how many substitutions have to be taken there and in which order they should be arranged in the composition.

Theorem

Each unifiable formula has a projective unifier in **S4.3** *which is a composition of Löwenheim substitutions.*

- Minari P. , Wroński A., *The property (HD) in intuitionistic Logic. A Partial Solution of a Problem of H. Ono*, Reports on mathematical logic 22 (1988), 21–25.

Theorem

Each unifiable formula A has a projective unifier in **L***C*

$$\varepsilon(B) = \begin{cases} A \Rightarrow B & \textit{if} \quad v(B) = \top \\ \neg\neg A \wedge (A \Rightarrow B) & \textit{if} \quad v(B) = \bot \end{cases},$$

where v is a ground unifier for A.

Note: This is not a Löwenheim substitution.

Theorem

Each unifiable formula A has a projective unifier in $IP.Q\text{-}\mathbf{L}C$

$$\varepsilon(B) = \begin{cases} \bigwedge_{\overline{x}} A \Rightarrow B & \text{if} \quad v(B) = \top \\ \neg\neg \bigwedge_{\overline{x}} A(\overline{x}) \wedge (\bigwedge_{\overline{x}} A(\overline{x}) \Rightarrow B) & \text{if} \quad v(B) = \bot \end{cases};$$

where (IP) $(A \Rightarrow \bigvee_x B(x)) \Rightarrow \bigvee_x (A \Rightarrow B(x))$*,*
(Independence of Premises) and v is a ground unifier for A.

- Dzik, W. *Chains of Structurally Complete Predicate Logics with the Application of Prucnal's Substitution*, RML 38(2004), (¬-less)

Theorem

Each unifiable formula A has a projective unifier in $IP.Q$-$\mathbf{L}C$

$$\varepsilon(B) = \begin{cases} \bigwedge_{\overline{x}} A \Rightarrow B & \text{if} \quad v(B) = \top \\ \neg\neg\bigwedge_{\overline{x}} A(\overline{x}) \wedge (\bigwedge_{\overline{x}} A(\overline{x}) \Rightarrow B) & \text{if} \quad v(B) = \bot \end{cases};$$

where (IP) $\quad (A \Rightarrow \bigvee_x B(x)) \Rightarrow \bigvee_x (A \Rightarrow B(x))$,
(Independence of Premises) and v is a ground unifier for A.

- Dzik, W. *Chains of Structurally Complete Predicate Logics with the Application of Prucnal's Substitution*, RML 38(2004), ($\neg$-less)

Theorem

An intermediate predicate logic L enjoys projective unification iff $IP.Q - LC \subseteq L$ iff $\exists$ is L definable by $(\forall, \rightarrow)$.
There is a chain of the type $\omega^\omega + 1$ of ASC logics over $IP.Q - LC$

Example 1

Let $A(a) = (\Box P(a) \vee \Box Q(a)) \wedge ({\sim} P(a) \vee {\sim} Q(a))$, where P, Q are monadic predicate symbols.

Example 1

Let $A(a) = (\Box P(a) \vee \Box Q(a)) \wedge (\sim P(a) \vee \sim Q(a))$, where P, Q are monadic predicate symbols.There are four generalized Löwenheim substitutions for A:

$\varepsilon_1(P(a)) = \Box\forall_x A(x) \to P(a)$ and $\varepsilon_1(Q(a)) = \Box\forall_x A(x) \wedge Q(a)$;
$\varepsilon_2(P(a)) = \Box\forall_x A(x) \wedge P(a)$ and $\varepsilon_2(Q(a)) = \Box\forall_x A(x) \to Q(a)$;
$\varepsilon_3(P(a)) = \Box\forall_x A(x) \wedge P(a)$ and $\varepsilon_3(Q(a)) = \Box\forall_x A(x) \wedge Q(a)$;
$\varepsilon_4(P(a)) = \Box\forall_x A(x) \to P(a)$ and $\varepsilon_4(Q(a)) = \Box\forall_x A(x) \to Q(a)$

Let $A(a) = (\Box P(a) \vee \Box Q(a)) \wedge ({\sim} P(a) \vee {\sim} Q(a))$, where P, Q are monadic predicate symbols.There are four generalized Löwenheim substitutions for A:

$\varepsilon_1(P(a)) = \Box\forall_x A(x) \to P(a)$ and $\varepsilon_1(Q(a)) = \Box\forall_x A(x) \wedge Q(a)$;
$\varepsilon_2(P(a)) = \Box\forall_x A(x) \wedge P(a)$ and $\varepsilon_2(Q(a)) = \Box\forall_x A(x) \to Q(a)$;
$\varepsilon_3(P(a)) = \Box\forall_x A(x) \wedge P(a)$ and $\varepsilon_3(Q(a)) = \Box\forall_x A(x) \wedge Q(a)$;
$\varepsilon_4(P(a)) = \Box\forall_x A(x) \to P(a)$ and $\varepsilon_4(Q(a)) = \Box\forall_x A(x) \to Q(a)$

No composition $\varepsilon_{j_1} \circ \cdots \circ \varepsilon_{j_n}$ of $\varepsilon_1, \varepsilon_2, \varepsilon_3, \varepsilon_4$ is a unifier for A.

Example 1

Let $A(a) = (\Box P(a) \vee \Box Q(a)) \wedge ({\sim} P(a) \vee {\sim} Q(a))$, where P, Q are monadic predicate symbols.There are four generalized Löwenheim substitutions for A:

$\varepsilon_1(P(a)) = \Box\forall_x A(x) \rightarrow P(a)$ and $\varepsilon_1(Q(a)) = \Box\forall_x A(x) \wedge Q(a)$;
$\varepsilon_2(P(a)) = \Box\forall_x A(x) \wedge P(a)$ and $\varepsilon_2(Q(a)) = \Box\forall_x A(x) \rightarrow Q(a)$;
$\varepsilon_3(P(a)) = \Box\forall_x A(x) \wedge P(a)$ and $\varepsilon_3(Q(a)) = \Box\forall_x A(x) \wedge Q(a)$;
$\varepsilon_4(P(a)) = \Box\forall_x A(x) \rightarrow P(a)$ and $\varepsilon_4(Q(a)) = \Box\forall_x A(x) \rightarrow Q(a)$

No composition $\varepsilon_{j_1} \circ \cdots \circ \varepsilon_{j_n}$ of $\varepsilon_1, \varepsilon_2, \varepsilon_3, \varepsilon_4$ is a unifier for A. On the other hand, since A is quantifier-free, it must have a projective unifier in Q-**S4.3**.

Example 2

Example 2

Let $A(a_1, a_2) = \Box Q(a_1, a_2) \ \vee \ P(a_1) \wedge {\sim} P(a_2)$

Example 2

Let $A(a_1, a_2) = \Box Q(a_1, a_2) \ \vee \ P(a_1) \wedge {\sim} P(a_2)$ and let

$$\varepsilon(P(a_i)) = (\Box \forall_{\overline{y}} A(\overline{y}) \wedge P(a_i) \ \vee \ B(a_i)) \ \wedge \ (\Box \forall_{\overline{y}} A(\overline{y}) \to P(a_1))$$

$$\varepsilon(Q(a_1, a_2)) = \Box \forall_{\overline{y}} A(\overline{y}) \to Q(a_1, a_2)$$

where the formula $B(x_1)$ will be specified later on. Note that the substitution ε is projective for A regardless of what $B(x_1)$ is.

Example 2

Let $A(a_1, a_2) = \Box Q(a_1, a_2) \ \vee \ P(a_1) \wedge {\sim} P(a_2)$ and let

$$\varepsilon(P(a_i)) = (\Box \forall_{\overline{y}} A(\overline{y}) \wedge P(a_i) \ \vee \ B(a_i)) \ \wedge \ (\Box \forall_{\overline{y}} A(\overline{y}) \rightarrow P(a_1))$$

$$\varepsilon(Q(a_1, a_2)) = \Box \forall_{\overline{y}} A(\overline{y}) \rightarrow Q(a_1, a_2)$$

where the formula $B(x_1)$ will be specified later on. Note that the substitution ε is projective for A regardless of what $B(x_1)$ is. One can check that $\varepsilon(A)$ is valid if one takes $x_1 {=} a_1$ as $B(x_1)$.

Note that some of the ideas are violated in the above Example.

Note that some of the ideas are violated in the above Example. First, the unifier ε does not fulfill the condition $vf(\varepsilon(B)) \subseteq vf(B)$ as we allow $\varepsilon(P(x_1))$ to contain the variable a_1.

Note that some of the ideas are violated in the above Example. First, the unifier ε does not fulfill the condition $vf(\varepsilon(B)) \subseteq vf(B)$ as we allow $\varepsilon(P(x_1))$ to contain the variable a_1. Second, ε is not a composition of generalized Löwenheim substitutions for A.

Note that some of the ideas are violated in the above Example. First, the unifier ε does not fulfill the condition $vf(\varepsilon(B)) \subseteq vf(B)$ as we allow $\varepsilon(P(x_1))$ to contain the variable a_1. Second, ε is not a composition of generalized Löwenheim substitutions for A. Third, to define ε one needs the equality symbol and it is not clear if our argument can be carried out without it, in Q-**S4.3**.

Note that some of the ideas are violated in the above Example. First, the unifier ε does not fulfill the condition $vf(\varepsilon(B)) \subseteq vf(B)$ as we allow $\varepsilon(P(x_1))$ to contain the variable a_1. Second, ε is not a composition of generalized Löwenheim substitutions for A. Third, to define ε one needs the equality symbol and it is not clear if our argument can be carried out without it, in Q-**S4.3**.

Theorem

$\Box IP.Q\text{-}S4.3_{=}$ enjoys projective unification.

where

$$(\Box IP) \qquad \Box(A \to \exists_x \Box B(x)) \to \exists_x \Box(A \to B(x))$$

Let us consider $\exists_x \Box P(x)$

Let us consider $\exists_x \Box P(x)$ and let ε be its projective unifier in L:

Let us consider $\exists_x \Box P(x)$ and let ε be its projective unifier in L:

$$\exists_x \Box P(x) \vdash_L P(a) \leftrightarrow \varepsilon(P(a)) \quad \text{and} \quad \vdash_L \exists_x \Box \varepsilon(P(x)).$$

Let us consider $\exists_x \Box P(x)$ and let ε be its projective unifier in L:

$$\exists_x \Box P(x) \vdash_L P(a) \leftrightarrow \varepsilon(P(a)) \quad \text{and} \quad \vdash_L \exists_x \Box \varepsilon(P(x)).$$

Hence $\vdash_L \varepsilon(P(a)) \to (\exists_x \Box P(x) \to P(a))$

Let us consider $\exists_x \Box P(x)$ and let ε be its projective unifier in L:

$$\exists_x \Box P(x) \vdash_L P(a) \leftrightarrow \varepsilon(P(a)) \quad \text{and} \quad \vdash_L \exists_x \Box \varepsilon(P(x)).$$

Hence $\vdash_L \varepsilon(P(a)) \rightarrow (\exists_x \Box P(x) \rightarrow P(a))$ and consequently

$$\vdash_L \exists_x \Box(\exists_x \Box P(x) \rightarrow P(x))$$

Let us consider $\exists_x \Box P(x)$ and let ε be its projective unifier in L:

$$\exists_x \Box P(x) \vdash_L P(a) \leftrightarrow \varepsilon(P(a)) \quad \text{and} \quad \vdash_L \exists_x \Box \varepsilon(P(x)).$$

Hence $\vdash_L \varepsilon(P(a)) \to (\exists_x \Box P(x) \to P(a))$ and consequently

$$\vdash_L \exists_x \Box(\exists_x \Box P(x) \to P(x))$$

which is equivalent (regarded as an axiom schema) to

$$(\Box IP) \qquad \vdash_L \Box(A \to \exists_x \Box P(x)) \to \exists_x \Box(A \to P(x)).$$

Let us consider $\exists_x \Box P(x)$ and let ε be its projective unifier in L:

$$\exists_x \Box P(x) \vdash_L P(a) \leftrightarrow \varepsilon(P(a)) \quad \text{and} \quad \vdash_L \exists_x \Box \varepsilon(P(x)).$$

Hence $\vdash_L \varepsilon(P(a)) \to (\exists_x \Box P(x) \to P(a))$ and consequently

$$\vdash_L \exists_x \Box(\exists_x \Box P(x) \to P(x))$$

which is equivalent (regarded as an axiom schema) to

$$(\Box IP) \qquad \vdash_L \Box(A \to \exists_x \Box P(x)) \to \exists_x \Box(A \to P(x)).$$

Theorem

Any modal predicate logic L (over $Q\text{-}S4$) which enjoys projective unification extends $\Box IP.Q\text{-}S4.3$.

Let us consider $\exists_x \Box P(x)$ and let ε be its projective unifier in L:

$$\exists_x \Box P(x) \vdash_L P(a) \leftrightarrow \varepsilon(P(a)) \quad \text{and} \quad \vdash_L \exists_x \Box \varepsilon(P(x)).$$

Hence $\vdash_L \varepsilon(P(a)) \to (\exists_x \Box P(x) \to P(a))$ and consequently

$$\vdash_L \exists_x \Box (\exists_x \Box P(x) \to P(x))$$

which is equivalent (regarded as an axiom schema) to

$$(\Box IP) \qquad \vdash_L \Box(A \to \exists_x \Box P(x)) \to \exists_x \Box (A \to P(x)).$$

Theorem

Any modal predicate logic L (over Q-S4) which enjoys projective unification extends $\Box IP.Q\text{-}S4.3$.

Corollary

Any modal predicate logic $L_=$ with equality (over Q-S$4_=$) enjoys projective unification iff $L_=$ extends $\Box IP.Q\text{-}S4.3_=$.