The unification type of Łukasiewicz logic

Based on a joint work with V. Marra

Luca Spada

Department of Mathematics
University of Salerno
http://logica.dmi.unisa.it/lucaspada

Workshop on Admissible Rules and Unification
Utrecht, 27th May 2011.

E-unifiers

Unless otherwise stated,
$\mathcal{L}$ is a first order language and
E is an equational theory in $\mathcal{L}$, arbitrary but fixed.

E-unifiers

Unless otherwise stated,
$\mathcal{L}$ is a first order language and
E is an equational theory in $\mathcal{L}$, arbitrary but fixed.

$$\text{Let } s, t \in Term(L)$$

A substitution σ is an E-unifier for the pair (s, t) if

$$E \models \sigma(t) \approx \sigma(s)$$

Jump to Łukasiewicz logic

Ordering the set of solutions

Given two unifiers σ, σ' for a pair (s,t) we say that

σ is more general (mod E) than σ'
(in symbols, $\sigma \geq_E \sigma'$)

Ordering the set of solutions

Given two unifiers σ, σ' for a pair (s,t) we say that

σ is more general (mod E) than σ'
(in symbols, $\sigma \geq_E \sigma'$)

if there is a substitution τ such that

$$E \models \sigma'(x) \approx \tau \circ \sigma(x) \text{ for all } x \in Var(s, t)$$

E-equivalence

The relation $\leq_E$ is a pre-order on the set of E-unifiers.

So it makes sense to say that two substitutions σ, σ' are

E-equivalent iff $\sigma \leq_E \sigma'$ and $\sigma' \leq_E \sigma$,
(written $\sigma \sim_E \sigma'$).

Most general unifiers

A substitution σ is a most general unifier for the pair (s, t), if

Most general unifiers

A substitution σ is a most general unifier for the pair (s, t), if

1. σ is a unifier for (s, t), and

Most general unifiers

A substitution σ is a most general unifier for the pair (s, t), if

1. σ is a unifier for (s, t), and
2. if τ is a unifier for (s, t), $\sigma \leq_E \tau$ implies $\sigma \sim_E \tau$.

Most general unifiers

A substitution σ is a most general unifier for the pair (s, t), if

1. σ is a unifier for (s, t), and
2. if τ is a unifier for (s, t), $\sigma \leq_E \tau$ implies $\sigma \sim_E \tau$.

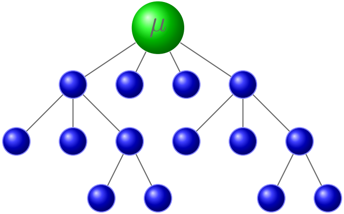
In other words, σ is a maximal element in the partial order induced by the equivalence $\sim_E$ on the preorder $\leq_E$.

Unitary unification type

For any pair (s, t), there is one E-unifier μ which is more general than any E-unifier of (s, t).

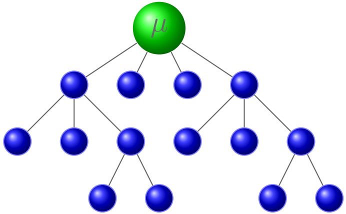
Unitary unification type

For any pair (s, t), there is one E-unifier μ which is more general than any E-unifier of (s, t).



Unitary unification type

For any pair (s,t), there is one E-unifier μ which is more general than any E-unifier of (s,t).



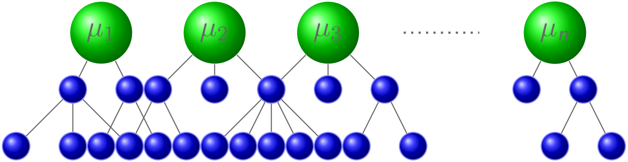
Then the unification type of E is called unary.

Finitary unification type

For any pair (s, t), there are finitely many E-unifiers $\mu_1, \ldots, \mu_n$ such that for any unifier σ, some μ_i is more general than σ.

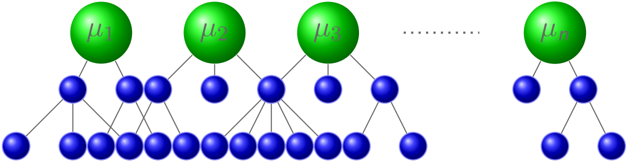
Finitary unification type

For any pair (s, t), there are finitely many E-unifiers $\mu_1, \ldots, \mu_n$ such that for any unifier σ, some μ_i is more general than σ.



Finitary unification type

For any pair (s, t), there are finitely many E-unifiers $\mu_1, \ldots, \mu_n$ such that for any unifier σ, some μ_i is more general than σ.



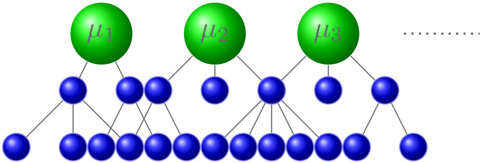
Then the unification type of E is called finitary.

Infinitary unification type

For any pair (s, t), there are infinitely many E-unifiers $\{\mu_i\}_{i \in I}$ such that for any unifier σ of (s, t) some μ_i is more general than σ.

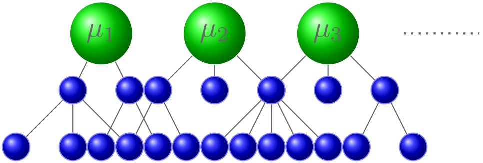
Infinitary unification type

For any pair (s, t), there are **infinitely many** E-unifiers $\{\mu_i\}_{i \in I}$ such that for any unifier σ of (s, t) some μ_i is more general than σ.



Infinitary unification type

For any pair (s, t), there are infinitely many E-unifiers $\{\mu_i\}_{i \in I}$ such that for any unifier σ of (s, t) some μ_i is more general than σ.



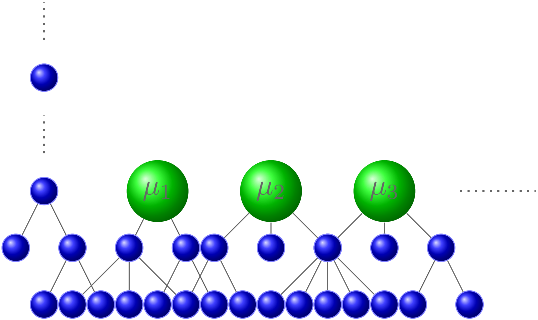
Then the unification type of E is called infinitary.

Nullary unification type

None of the above, i.e. there exists a pair (s, t) and a unifier u which is not less general than any most general unifier.

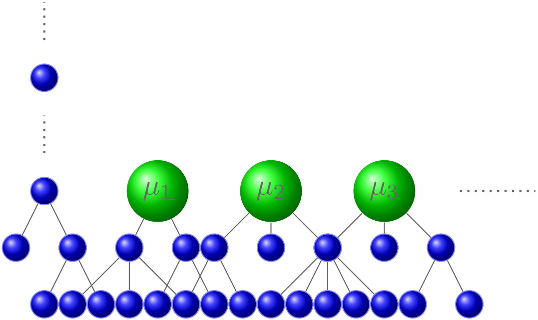
Nullary unification type

None of the above, i.e. there exists a pair (s, t) and a unifier u which is not less general than any most general unifier.



Nullary unification type

None of the above, i.e. there exists a pair (s, t) and a unifier u which is not less general than any most general unifier.



Then the unification type of E is called nullary.

Łukasiewicz logic

Łukasiewicz (infinite-valued propositional) logic is a non-classical system going back to the 1920's which may be axiomatised using the primitive connectives $\to$ (implication) and $\neg$ (negation)

Łukasiewicz logic

Łukasiewicz (infinite-valued propositional) logic is a non-classical system going back to the 1920's which may be axiomatised using the primitive connectives $\to$ (implication) and $\neg$ (negation) by the four axiom schemata:

(A1) $\alpha \to (\beta \to \alpha)$,

(A2) $(\alpha \to \beta) \to ((\beta \to \gamma) \to (\alpha \to \gamma))$,

(A3) $(\alpha \to \beta) \to \beta) \to ((\beta \to \alpha) \to \alpha)$,

(A4) $(\neg\alpha \to \neg\beta) \to (\beta \to \alpha)$,

with *modus ponens* as the only deduction rule.

Semantics of Łukasiewicz logic

The semantics of Łukasiewicz logic is many-valued: assignments μ to atomic formulæ range in the unit interval $[0,1] \subseteq \mathbb{R}$; they are extended compositionally to compound formulæ via

$$\mu(\alpha \to \beta) = \min\{1, 1 - \mu(\alpha) + \mu(\beta)\}\,,$$
$$\mu(\neg\alpha) = 1 - \mu(\alpha)\,.$$

Semantics of Łukasiewicz logic

The semantics of Łukasiewicz logic is many-valued: assignments μ to atomic formulæ range in the unit interval $[0,1] \subseteq \mathbb{R}$; they are extended compositionally to compound formulæ via

$$\mu(\alpha \to \beta) = \min\{1, 1 - \mu(\alpha) + \mu(\beta)\},$$
$$\mu(\neg\alpha) = 1 - \mu(\alpha).$$

Tautologies are defined as those formulæ that evaluate to 1 under every such assignment.
Chang first considered the Tarski-Lindenbaum algebras of Łukasiewicz logic, and called them MV-algebras. This allowed him to obtain an algebraic proof of the completeness theorem.

MV-algebras

Definition

An MV-algebra is a structure $\mathcal{A} = \langle A, \oplus, ^*, 0 \rangle$ such that:

MV-algebras

Definition

An MV-algebra is a structure $\mathcal{A} = \langle A, \oplus, ^*, 0\rangle$ such that:

- $\mathcal{A} = \langle A, \oplus, 0\rangle$ is a commutative monoid,

MV-algebras

Definition

An MV-algebra is a structure $\mathcal{A} = \langle A, \oplus, ^{*}, 0\rangle$ such that:

- $\mathcal{A} = \langle A, \oplus, 0\rangle$ is a commutative monoid,
- * is an involution

MV-algebras

Definition

An MV-algebra is a structure $\mathcal{A} = \langle A, \oplus, ^*, 0\rangle$ such that:

- $\mathcal{A} = \langle A, \oplus, 0\rangle$ is a commutative monoid,
- * is an involution
- the interaction between those two operations is described by the following two axioms:

MV-algebras

Definition

An MV-algebra is a structure $\mathcal{A} = \langle A, \oplus, {}^*, 0\rangle$ such that:

- $\mathcal{A} = \langle A, \oplus, 0\rangle$ is a commutative monoid,
- * is an involution
- the interaction between those two operations is described by the following two axioms:
  - $x \oplus 0^* = 0^*$
  - $(x^* \oplus y)^* \oplus y = (y^* \oplus x)^* \oplus x$

Some small technical considerations

In MV-alegrbas (as in several other cases) the problem of unifying two terms reduces to finding a substitution that identifies a term with a constant (in our case either 0 or 1).

Some small technical considerations

In MV-alegrbas (as in several other cases) the problem of unifying two terms reduces to finding a substitution that identifies a term with a constant (in our case either 0 or 1). Furthermore, in MV-algebras, solving a system of equation of unification problems is equivalent to solve a single unification problem:

Some small technical considerations

In MV-alegrbas (as in several other cases) the problem of unifying two terms reduces to finding a substitution that identifies a term with a constant (in our case either 0 or 1). Furthermore, in MV-algebras, solving a system of equation of unification problems is equivalent to solve a single unification problem:

$$\begin{cases} t_1(\bar{x}) = s_1(\bar{x}) \\ t_1(\bar{x}) = s_2(\bar{x}) \\ \ldots \\ t_n(\bar{x}) = s_n(\bar{x}) \end{cases} \qquad \Longleftrightarrow \qquad u(\bar{x}) = 1.$$

Projective objects

In 1997 Ghilardi proposed an alternative approach to unification which has several advantages.

Projective objects

In 1997 Ghilardi proposed an alternative approach to unification which has several advantages.

The key concept is given by projective formulas or, equivalently, projective algebras.

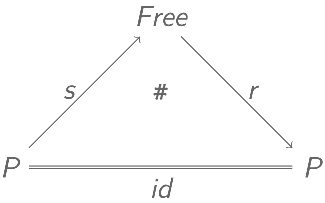
An algebra is called projective (in a variety) if it is a retract of some free algebra of that variety, i.e.,

Projective objects

In 1997 Ghilardi proposed an alternative approach to unification which has several advantages.

The key concept is given by projective formulas or, equivalently, projective algebras.

An algebra is called projective (in a variety) if it is a retract of some free algebra of that variety, i.e.,



Algebraic unifiers

Let us think of a generic E-unification problem (s, t) as a finitely presented E-algebra $A = \mathcal{F}_n / \langle (s, t) \rangle$.

Algebraic unifiers

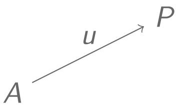
Let us think of a generic E-unification problem (s, t) as a finitely presented E-algebra $A = \mathcal{F}_n / \langle (s, t) \rangle$.

An algebraic E-unifier for the problem $A = \mathcal{F}_n / \langle (s, t) \rangle$ is a pair (P, u) where

1. P is a finitely presented projective E-algebra and
2. u is an arrow from A to P, $u \colon A \longrightarrow P$.

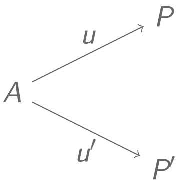
Most general algebraic unifiers

An algebraic E-unifier (P, u)



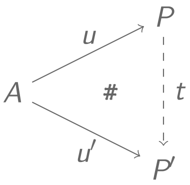
Most general algebraic unifiers

An algebraic E-unifier (P, u) is more general than (P', u')



Most general algebraic unifiers

An algebraic E-unifier (P, u) is more general than (P', u') if there exists an arrow t s.t.



Equivalence of the two approaches

The two approaches define two pre-orders which can be thought of as categories.

Equivalence of the two approaches

The two approaches define two pre-orders which can be thought of as categories.

Theorem (1997 Ghilardi)

The syntactic approach and the algebraic are equivalent (as categories).

Jump to the proof

Unitarity of finite-valued Łukasiewicz logic

Ghilardi himself noticed that finite-valued Łukasiewicz logic has unitary type.

Unitarity of finite-valued Łukasiewicz logic

Ghilardi himself noticed that finite-valued Łukasiewicz logic has unitary type.

This was re-proved explicitly and generalised to any finite-valued extension of Basic Logic by Dzik.

Non unitarity of the unification in Łukasiewicz logic

Łukasiewicz logic has a weak disjunction property. Namely:

if $\varphi \vee \neg\varphi$ is derivable then either φ or $\neg\varphi$ must be derivable. (in other words the multiple-conclusion rule $\varphi \vee \neg\varphi / \varphi, \neg\varphi$ is admissible.)

Non unitarity of the unification in Łukasiewicz logic

Łukasiewicz logic has a weak disjunction property. Namely:

if $\varphi \vee \neg\varphi$ is derivable then either φ or $\neg\varphi$ must be derivable. (in other words the multiple-conclusion rule $\varphi \vee \neg\varphi / \varphi, \neg\varphi$ is admissible.)

This entails the unification type of Łukasiewicz logic to be at least not unitary.

Non unitarity of the unification in Łukasiewicz logic

Łukasiewicz logic has a weak disjunction property. Namely:

if $\varphi \vee \neg\varphi$ is derivable then either φ or $\neg\varphi$ must be derivable. (in other words the multiple-conclusion rule $\varphi \vee \neg\varphi / \varphi, \neg\varphi$ is admissible.)

This entails the unification type of Łukasiewicz logic to be at least not unitary.

Indeed if σ is a unifier for $x \vee \neg x$, then it must unify either x (hence it is the substitution $x \mapsto 1$) or $\neg x$ (hence it must be the substitution $x \mapsto 0$).

Intermezzo: Commutative ℓ-groups...

Theorem (Mundici 1986)

The category of MV-alegrbas is equivalent to the category of Abelian ℓ-groups with strong unit *(with ℓ-morphisms preserving the strong unit).*

Intermezzo: Commutative ℓ-groups...

Theorem (Mundici 1986)

The category of MV-alegrbas is equivalent to the category of Abelian ℓ-groups with strong unit (with ℓ-morphisms preserving the strong unit).

In 1975, Beynon, expanding previous results by Baker, established a categorical duality which enabled a geometrical study of finitely presented ℓ-groups.

Intermezzo: Commutative ℓ-groups...

Theorem (Mundici 1986)

The category of MV-alegrbas is equivalent to the category of Abelian ℓ-groups with **strong unit** *(with ℓ-morphisms preserving the strong unit).*

In 1975, Beynon, expanding previous results by Baker, established a categorical duality which enabled a geometrical study of finitely presented ℓ-groups.

This duality led to the following purely algebraic result.

Theorem (1977 Beynon)

Finitely generated projective ℓ-groups are exactly the finitely presented ℓ-groups.

...and their unification type.

In the theory of ℓ-groups all system of equations are solvable.

...and their unification type.

In the theory of ℓ-groups all system of equations are solvable. In the light of the Beynon's and Ghilardi's results, one easily gets:

Theorem

The unification type of the theory of ℓ-groups is unitary.

...and their unification type.

In the theory of ℓ-groups all system of equations are solvable. In the light of the Beynon's and Ghilardi's results, one easily gets:

Theorem

The unification type of the theory of ℓ-groups is unitary.

In a forthcoming paper with V. Marra, we exploit Beynon's geometrical duality to give an algorithm that, taken any (system of) term in the language of ℓ-groups, outputs its most general unifier.

Finitarity result

Definition

A McNaughton function is a function $\varphi\colon [0,1]^n \to [0,1]$ which is continuous, piece-wise linear and with integer coefficients.

Finitarity result

Definition

A McNaughton function is a function $\varphi\colon [0,1]^n \to [0,1]$ which is continuous, piece-wise linear and with integer coefficients.

These functions are named after McNaughton, who first proved that they exactly correspond to formulæ of Łukasiewicz logic.

Finitarity result

Definition

A McNaughton function is a function $\varphi\colon [0,1]^n \to [0,1]$ which is continuous, piece-wise linear and with integer coefficients.

These functions are named after McNaughton, who first proved that they exactly correspond to formulæ of Łukasiewicz logic.

Proposition

The unification type of the 1-variable fragment of Łukasiewicz logic is finitary.

Finitarity result

Definition

A McNaughton function is a function $\varphi\colon [0,1]^n \to [0,1]$ which is continuous, piece-wise linear and with integer coefficients.

These functions are named after McNaughton, who first proved that they exactly correspond to formulæ of Łukasiewicz logic.

Proposition

The unification type of the 1-variable fragment of Łukasiewicz logic is finitary.

In particular the proof shows that there are at most two most general unifiers, for any given formula.

Jump to the proof

A geometrical view on unification

McNaughton functions prove to be a useful tool for understanding the dynamic of substitutions in Łukasiewicz logic.

A geometrical view on unification

McNaughton functions prove to be a useful tool for understanding the dynamic of substitutions in Łukasiewicz logic.
However, McNaughton functions are only an instance of a stronger link between Łukasiewicz logic and Geometry.

A geometrical view on unification

McNaughton functions prove to be a useful tool for understanding the dynamic of substitutions in Łukasiewicz logic.
However, McNaughton functions are only an instance of a stronger link between Łukasiewicz logic and Geometry.

Definition

A rational polytope is the convex hull of a finite set of rational points.

A geometrical view on unification

McNaughton functions prove to be a useful tool for understanding the dynamic of substitutions in Łukasiewicz logic.
However, McNaughton functions are only an instance of a stronger link between Łukasiewicz logic and Geometry.

Definition

A rational polytope is the convex hull of a finite set of rational points.
A rational polyhedron is the union of a finite number of rational polytopes.

A geometrical view on unification

McNaughton functions prove to be a useful tool for understanding the dynamic of substitutions in Łukasiewicz logic.
However, McNaughton functions are only an instance of a stronger link between Łukasiewicz logic and Geometry.

Definition

A rational polytope is the convex hull of a finite set of rational points.
A rational polyhedron is the union of a finite number of rational polytopes.
A $\mathbb{Z}$-map is a continuous piecewise linear function with integer coefficients.

Rational polyhedra and MV-algebras

Let $\mathcal{MV}_{fp}$ be the category of finitely presented MV-algebras with their homomorphisms.

Rational polyhedra and MV-algebras

Let $\mathcal{MV}_{fp}$ be the category of finitely presented MV-algebras with their homomorphisms.
Let $\mathcal{P}_{\mathbb{Z}}$ be the category of rational polyhedra and $\mathbb{Z}$-maps between them.

Rational polyhedra and MV-algebras

Let $\mathcal{MV}_{fp}$ be the category of finitely presented MV-algebras with their homomorphisms.
Let $\mathcal{P}_{\mathbb{Z}}$ be the category of rational polyhedra and $\mathbb{Z}$-maps between them.
I will define a pair of (contravariant) functors:

$$\mathcal{I}\colon \mathcal{MV}_{fp} \to \mathcal{P}_{\mathbb{Z}} \qquad \text{and} \qquad \mathcal{V}\colon \mathcal{P}_{\mathbb{Z}} \to \mathcal{MV}_{fp}\,.$$

These functors operate very similarly to the classical ones in algebraic geometry that associate ideals with varieties.

Interpreting MV-algebras inside polyhedra: objects

Let $P \in \mathcal{P}_{\mathbb{Z}}$.

Let us write $\imath(P)$ for the collection of all pair MV-terms (s, t) such that

$$s(x) = t(x) \text{ for all } x \in P$$

Interpreting MV-algebras inside polyhedra: objects

Let $P \in \mathcal{P}_{\mathbb{Z}}$.

Let us write $\imath(P)$ for the collection of all pair MV-terms (s, t) such that

$$s(x) = t(x) \text{ for all } x \in P$$

The set $\imath(P)$ is a congruence of the free MV-algebras on n generators, so it makes sense to set

$$\mathcal{I}(P) = \frac{Free_n}{\imath(P)}.$$

Interpreting polyhedra inside MV-algebras: arrows

Let $\zeta\colon P \to Q$ be a diagram in $\mathcal{P}_{\mathbb{Z}}$.

Define

$$\mathcal{I}(\zeta)\colon \mathcal{I}(Q) \to \mathcal{I}(P)$$

Interpreting polyhedra inside MV-algebras: arrows

Let $\zeta \colon P \to Q$ be a diagram in $\mathcal{P}_{\mathbb{Z}}$.

Define

$$\mathcal{I}(\zeta) \colon \mathcal{I}(Q) \to \mathcal{I}(P)$$

as

$$f \in \mathcal{I}(Q) \stackrel{\mathsf{M}(\zeta)}{\longmapsto} f \circ \zeta \in \mathcal{I}(P).$$

Interpreting polyhedra inside MV-algebras: arrows

Let $\zeta\colon P \to Q$ be a diagram in $\mathcal{P}_{\mathbb{Z}}$.

Define

$$\mathcal{I}(\zeta)\colon \mathcal{I}(Q) \to \mathcal{I}(P)$$

as

$$f \in \mathcal{I}(Q) \overset{\mathsf{M}(\zeta)}{\longmapsto} f \circ \zeta \in \mathcal{I}(P).$$

Then, the function $\mathcal{I}(\zeta)\colon \mathcal{I}(Q) \to \mathcal{I}(P)$ is a homomorphism of MV-algebras.

Interpreting polyhedra inside MV-algebras: objects

$$\text{Let } A = \frac{Free_n}{\theta} \in \mathcal{MV}_{fp}.$$

Let us write $V(\theta)$ for the collection of all real points p in $[0,1]^n$ such that

$$s(p) = t(p) \text{ for all } (s,t) \in \theta$$

Interpreting polyhedra inside MV-algebras: objects

$$\text{Let } A = \frac{Free_n}{\theta} \in \mathcal{MV}_{fp}.$$

Let us write $V(\theta)$ for the collection of all real points p in $[0,1]^n$ such that

$$s(p) = t(p) \text{ for all } (s,t) \in \theta$$

The set $V(\theta)$ is a rational polyhedron, so we set

$$\mathcal{V}(A) = V(\theta)\,.$$

Interpreting MV-algebras inside polyhedra: arrows

Let $h\colon A \to B$ be a diagram in $\mathcal{MV}_{fp}$.

Suppose that h sends the generators of A into the elements $\{t_i\}_{i \in I}$ of B, then define

$$\mathcal{V}(h)\colon \mathcal{V}(B) \to \mathcal{V}(A)$$

Interpreting MV-algebras inside polyhedra: arrows

Let $h\colon A \to B$ be a diagram in $\mathcal{MV}_{fp}$.

Suppose that h sends the generators of A into the elements $\{t_i\}_{i \in I}$ of B, then define

$$\mathcal{V}(h)\colon \mathcal{V}(B) \to \mathcal{V}(A)$$

as

$$p \in \mathcal{V}(A) \xmapsto{\mathcal{V}(h)} \langle t_i(p) \rangle_{i \in I} \in \mathcal{V}(A).$$

Interpreting MV-algebras inside polyhedra: arrows

Let $h\colon A \to B$ be a diagram in $\mathcal{MV}_{fp}$.

Suppose that h sends the generators of A into the elements $\{t_i\}_{i\in I}$ of B, then define

$$\mathcal{V}(h)\colon \mathcal{V}(B) \to \mathcal{V}(A)$$

as

$$p \in \mathcal{V}(A) \xmapsto{\mathcal{V}(h)} \langle t_i(p)\rangle_{i\in I} \in \mathcal{V}(A).$$

Then, the function $\mathcal{V}(h)\colon \mathcal{V}(B) \to \mathcal{V}(A)$ is a $\mathbb{Z}$-map.

Duality for finitely presented MV-algebras

Duality for finitely presented MV-algebras

Theorem (Folklore)

The pair of functors

$$\mathcal{I}\colon \mathcal{MV}_{fp} \to \mathcal{P}_{\mathbb{Z}} \quad \text{and} \quad \mathcal{V}\colon \mathcal{P}_{\mathbb{Z}} \to \mathcal{MV}_{fp}\,.$$

constitutes a contravariant equivalence between the two categories.

Co-unification problems

A co-unification problem, in this setting, is now given by a rational polyhedra A.

Co-unification problems

A co-unification problem, in this setting, is now given by a rational polyhedra A.

A co-unifier for A is a pair (P, u), where

1. P is a $\mathbb{Z}$-retract of $[0,1]^n$ (i.e. a retract by $\mathbb{Z}$-maps) for some n, and
2. u is a $\mathbb{Z}$-map from P to A.

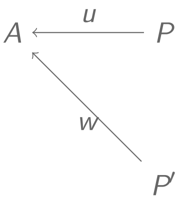
Co-unification type

A co-unifier (P, u)

$$A \xleftarrow{\ u\ } P$$

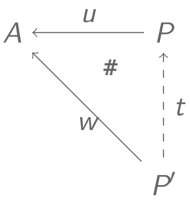
Co-unification type

A co-unifier $(P, u)$is more general than (Q, w)



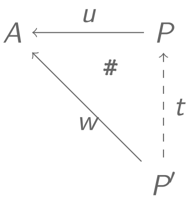
Co-unification type

A co-unifier $(P, u)$is **more general** than (Q, w) if there exists t such that:



Co-unification type

A co-unifier $(P, u)$is more general than (Q, w) if there exists t such that:



Remark

The unification type of Łukasiewicz logic and the co-unification type of rational prolyhedra coincide.

Nullarity of Łukasiewicz logic

Theorem

The full Łukasiewicz logic has nullary unification type.

Nullarity of Łukasiewicz logic

Theorem

The full Łukasiewicz logic has nullary unification type.

Proof. Consider the unification problem given by

$$\theta = (x \vee x^* \vee y \vee y^*, 1)$$

Nullarity of Łukasiewicz logic

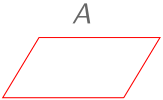
Theorem

The full Łukasiewicz logic has nullary unification type.

Proof. Consider the unification problem given by

$$\theta = (x \vee x^* \vee y \vee y^*, 1)$$

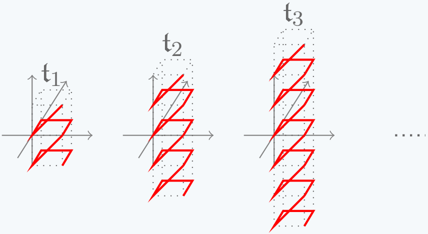
The rational polyhedron $\mathcal{V}(\theta)$ associated to the finitely presented MV-algebra $Free_2/\langle\theta\rangle$ is the following union of four rational polytopes:



Proof Cont.'d

Step 1.

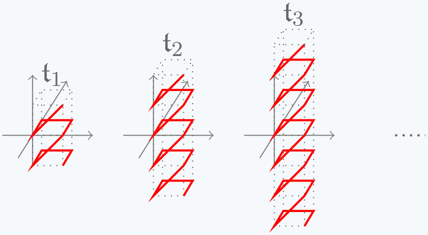
Let us consider the following sequence of rational polyhedra,



Proof Cont.'d

Step 1.

Let us consider the following sequence of rational polyhedra,

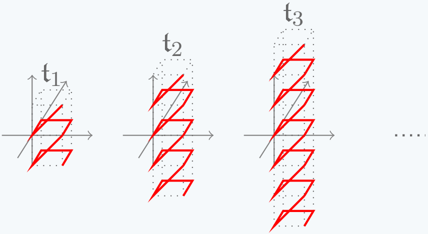


Together with the projections $\zeta_i \colon \mathfrak{t}_i \to A$ into A. It can be proved (cfr. Cabrer and Mundici) that each $\mathfrak{t}_i$ is a retract of $[0,1]^m$ for some m,

Proof Cont.'d

Step 1.

Let us consider the following sequence of rational polyhedra,



Together with the projections $\zeta_i\colon \mathfrak{t}_i \to A$ into A. It can be proved (cfr. Cabrer and Mundici) that each $\mathfrak{t}_i$ is a retract of $[0,1]^m$ for some m, so the pairs $(\mathfrak{t}_i, \zeta_i)$ are co-unifiers for A.

Proof Cont.'d

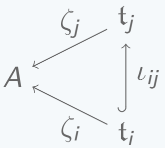
Step 2.

The sequence is increasing, i.e.

Proof Cont.'d

Step 2.

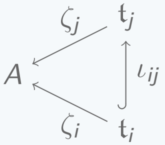
The sequence is increasing, i.e.for any $i < j$,
there exists ι_{ij} such that the following diagram commutes.



Proof Cont.'d

Step 2.

The sequence is increasing, i.e.for any $i < j$,
there exists ι_{ij} such that the following diagram commutes.



Indeed ι_{ij} is the embedding of $\mathfrak{t}_i$ in $\mathfrak{t}_j$

Proof Cont.'d

Step 3: The lifting of functions Lemma.

For any $\mathbb{Z}$-retract P of some cube $[0,1]^n$

Proof Cont.'d

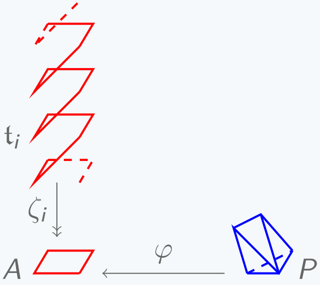
Step 3: The lifting of functions Lemma.

For any $\mathbb{Z}$-retract P of some cube $[0,1]^n$ and for any arrow $\varphi\colon P \to A$,

Proof Cont.'d

Step 3: The lifting of functions Lemma.

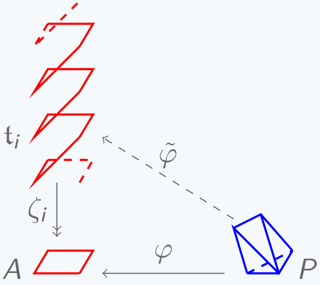
For any $\mathbb{Z}$-retract P of some cube $[0,1]^n$ and for any arrow $\varphi\colon P \to A$, there exists some $\mathfrak{t}_i$



Proof Cont.'d

Step 3: The lifting of functions Lemma.

For any $\mathbb{Z}$-retract P of some cube $[0,1]^n$ and for any arrow $\varphi\colon P \to A$, there exists some $\mathfrak{t}_i$ and an arrow $\tilde{\varphi}$ (called the lift of φ) making the following diagram commute.



The universal cover of the circle

Intermezzo 2

The above lemma is the piecewise linear version of the "*Lifting of functions*" *Lemma*, widely used in algebraic topology.

The universal cover of the circle

Intermezzo 2

The above lemma is the piecewise linear version of the "*Lifting of functions*" *Lemma*, widely used in algebraic topology.

The reason why the above lemma works is that the infinite spiral $\mathfrak{t}_\infty = \bigcup_{i\in\omega} \mathfrak{t}_i$ is the piecewise linear correspondent of the universal cover of the circle, a space that, indeed, enjoys this factorisation property for any continuous maps from a simply connected spaces into the circle.

The universal cover of the circle

Intermezzo 2

The above lemma is the piecewise linear version of the "*Lifting of functions*" *Lemma*, widely used in algebraic topology.

The reason why the above lemma works is that the infinite spiral $\mathfrak{t}_\infty = \bigcup_{i \in \omega} \mathfrak{t}_i$ is the piecewise linear correspondent of the universal cover of the circle, a space that, indeed, enjoys this factorisation property for any continuous maps from a simply connected spaces into the circle.

Such a lift is known to be unique even in the rather general case of continuous maps, so the fact that in our setting such a map is actually a $\mathbb{Z}$-map is a quite pleasant discovery.

Proof Cont.'d

The Lifting Lemma has two important corollaries.

Proof Cont.'d

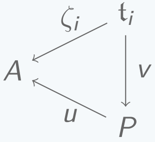
The Lifting Lemma has **two important corollaries**.
A **lattice point** is a vector with integer coordinates.

Proof Cont.'d

The Lifting Lemma has two important corollaries.
A lattice point is a vector with integer coordinates.

Step 4: Corollary 1.

If (P, u) is a co-unifier for A with strictly fewer lattice points than $\mathfrak{t}_i$, then there is no arrow $v\colon \mathfrak{t}_i \to P$ making the following diagram commute.



Proof Cont.'d

Step 5.

As a consequence of the previous corollary we obtain:

Proof Cont.'d

Step 5.

As a consequence of the previous corollary we obtain:

1. The sequence of $\mathfrak{t}_i$ is strict; i.e. $\mathfrak{t}_i$ is not more general than $\mathfrak{t}_j$ if $i < j$.

Proof Cont.'d

Step 5.

As a consequence of the previous corollary we obtain:

1. The sequence of $\mathfrak{t}_i$ is strict; i.e. $\mathfrak{t}_i$ is not more general than $\mathfrak{t}_j$ if $i < j$.
2. The sequence admits no bound with a finite number of lattice elements. Therefore, no rational polyhedra can bound the sequence of $\mathfrak{t}_i$.

Proof Cont.'d

Step 6: Corollary 2.

Given any co-unifier (P, u) for A, there exists a co-unifier of the form $(\mathfrak{t}_i, \zeta_i)$ such that

$$(P, u) \leq_{\mathcal{P}_{\mathbb{Z}}} (\mathfrak{t}_i, \zeta_i) .$$

End of the Proof

Conclusion

Summing up, we have found a strictly linearly ordered, cofinal sequence of unifiers for A.

End of the Proof

Conclusion

Summing up, we have found a strictly linearly ordered, cofinal sequence of unifiers for A.
Furthermore the sequence is unbounded, hence the co-unification type of rational polyhedra is nullary (in a stronger sense).

End of the Proof

Conclusion

Summing up, we have found a strictly linearly ordered, cofinal sequence of unifiers for A.
Furthermore the sequence is unbounded, hence the co-unification type of rational polyhedra is nullary (in a stronger sense).

This proves that the Łukasiewicz calculus (as well as the theory of MV-algebras and ℓ-groups with strong unit) has nullary unification type. □

▸ Bibliography

Equivalence of the syntactical and algebraic approaches

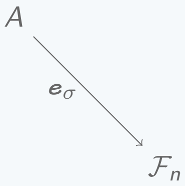
The correspondence e works as follows. ▸ Back

Equivalence of the syntactical and algebraic approaches

The correspondence e works as follows. Back

Fullness

To any syntactic unifier σ there corresponds an algebraic unifier e_σ from A to $\mathcal{F}_n$ defined by $e_\sigma([t]) := [\sigma(t)]$.

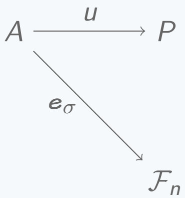


Equivalence of the syntactical and algebraic approaches

The correspondence e works as follows. ▸ Back

Fullness

To any syntactic unifier σ there corresponds an algebraic unifier e_σ from A to $\mathcal{F}_n$ defined by $e_\sigma([t]) := [\sigma(t)]$.
To see that the correspondence is full, take any algebraic unifier (P, u):

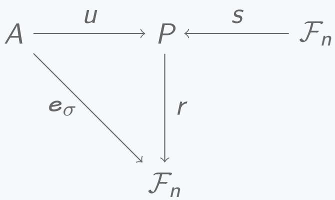


Equivalence of the syntactical and algebraic approaches

The correspondence e works as follows. ▸ Back

Fullness

To any syntactic unifier σ there corresponds an algebraic unifier e_σ from A to $\mathcal{F}_n$ defined by $e_\sigma([t]) := [\sigma(t)]$.
To see that the correspondence is full, take any algebraic unifier (P, u):

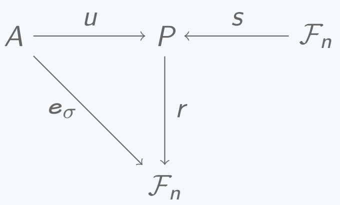


Equivalence of the syntactical and algebraic approaches

The correspondence e works as follows. ▸ Back

Fullness

To any syntactic unifier σ there corresponds an algebraic unifier e_σ from A to $\mathcal{F}_n$ defined by $e_\sigma([t]) := [\sigma(t)]$.
To see that the correspondence is full, take any algebraic unifier (P, u):



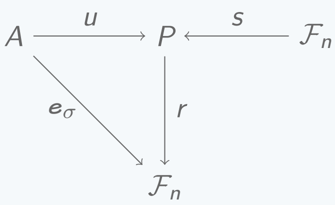
Define $[\sigma(x)] = r(s([x]))$.

Equivalence of the syntactical and algebraic approaches

The correspondence e works as follows. ▸ Back

Fullness

To any syntactic unifier σ there corresponds an algebraic unifier e_σ from A to $\mathcal{F}_n$ defined by $e_\sigma([t]) := [\sigma(t)]$.
To see that the correspondence is full, take any algebraic unifier (P, u):



Define $[\sigma(x)] = r(s([x]))$. So we have $r \circ s = e_\sigma$ and $s \circ e_\sigma = u$.

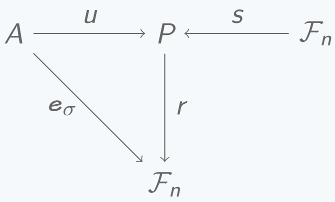
Equivalence of the syntactical and algebraic approaches

The correspondence e works as follows. ▸ Back

Fullness

To any syntactic unifier σ there corresponds an algebraic unifier e_σ from A to $\mathcal{F}_n$ defined by $e_\sigma([t]) := [\sigma(t)]$.

To see that the correspondence is full, take any algebraic unifier (P, u):



Define $[\sigma(x)] = r(s([x]))$. So we have $r \circ s = e_\sigma$ and $s \circ e_\sigma = u$. Whence $e_\sigma \sim_E u$. □

Proof of finitarity

Back **Proof.** A key step in the proof is to recall that Łukasiewicz formulas can be interpreted as McNaughton functions.

Proof of finitarity

Back **Proof.** A key step in the proof is to recall that Łukasiewicz formulas can be interpreted as McNaughton functions.

Substitutions can be viewed as arrays of formulas: so are interpreted as vectorial McNaughton functions.

$$x_1 \mapsto \varphi_1(x_1, \ldots, x_n), x_2 \mapsto \varphi_2(x_1, \ldots, x_n), \ldots, x_n \mapsto \varphi_n(x_1, \ldots, x_n)$$

Proof of finitarity

Back **Proof.** A key step in the proof is to recall that Łukasiewicz formulas can be interpreted as McNaughton functions.

Substitutions can be viewed as arrays of formulas: so are interpreted as vectorial McNaughton functions.

$$x_1 \mapsto \varphi_1(x_1, \ldots, x_n), x_2 \mapsto \varphi_2(x_1, \ldots, x_n), \ldots, x_n \mapsto \varphi_n(x_1, \ldots, x_n)$$

In the 1-variable case, substitutions are just McNaughton functions.

Proof cont'd

Back

Step 1

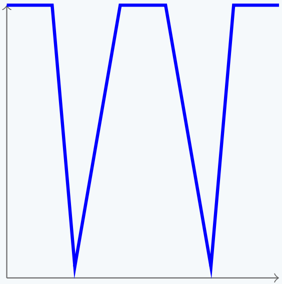
A Łukasiewicz formula in 1 variable is unifiable iff it is classically satisfiable.

Proof cont'd

▸ Back

Step 1

A Łukasiewicz formula in 1 variable is unifiable iff it is classically satisfiable.

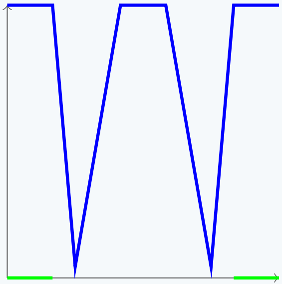


Proof cont'd

▸ Back

Step 1

A Łukasiewicz formula in 1 variable is unifiable iff it is classically satisfiable.

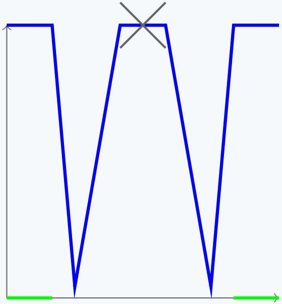


Proof cont'd

▸ Back

Step 1

A Łukasiewicz formula in 1 variable is unifiable iff it is classically satisfiable.



Proof cont'd

Back

Step 2

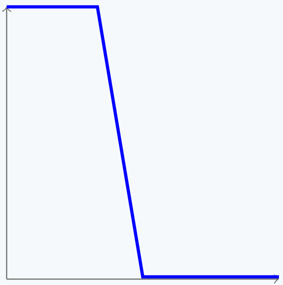
If a formula is true in an interval $[0, a]$ or $[b, 1]$ then its unifiers are exactly the functions whose range is contained in one of those interval and all them are equivalent.

Proof cont'd

Back

Step 2

If a formula is true in an interval $[0, a]$ or $[b, 1]$ then its unifiers are exactly the functions whose range is contained in one of those interval and all them are equivalent.

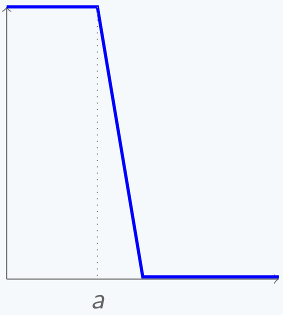


Proof cont'd

Back

Step 2

If a formula is true in an interval $[0, a]$ or $[b, 1]$ then its unifiers are exactly the functions whose range is contained in one of those interval and all them are equivalent.

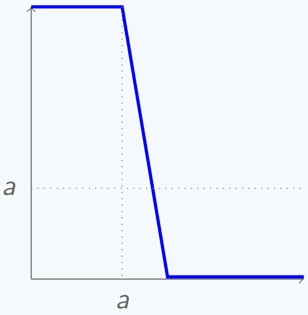


Proof cont'd

Back

Step 2

If a formula is true in an interval $[0, a]$ or $[b, 1]$ then its unifiers are exactly the functions whose range is contained in one of those interval and all them are equivalent.

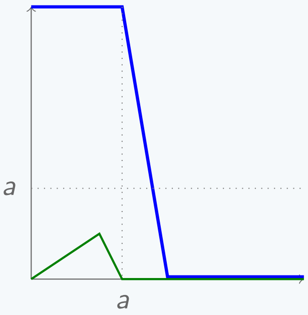


Proof cont'd

Back

Step 2

If a formula is true in an interval $[0, a]$ or $[b, 1]$ then its unifiers are exactly the functions whose range is contained in one of those interval and all them are equivalent.

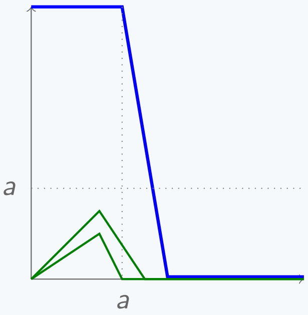


Proof cont'd

Back

Step 2

If a formula is true in an interval $[0, a]$ or $[b, 1]$ then its unifiers are exactly the functions whose range is contained in one of those interval and all them are equivalent.

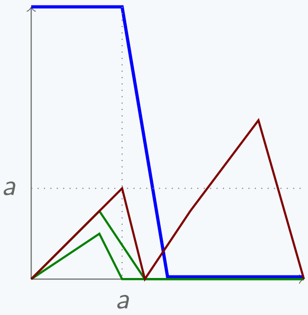


Proof cont'd

Back

Step 2

If a formula is true in an interval $[0, a]$ or $[b, 1]$ then its unifiers are exactly the functions whose range is contained in one of those interval and all them are equivalent.



Proof cont'd

Back

Step 3

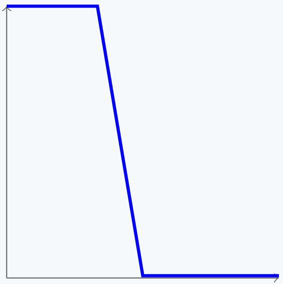
The most general unifiers of a formula which is true in an interval $[0, a]$ or $[b, 1]$ are exactly the functions whose range is exactly one of those interval.

Proof cont'd

Back

Step 3

The most general unifiers of a formula which is true in an interval $[0, a]$ or $[b, 1]$ are exactly the functions whose range is exactly one of those interval.

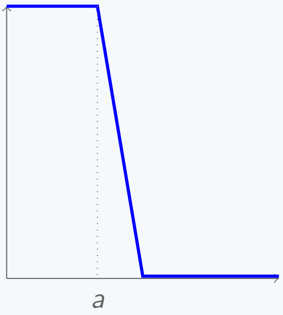


Proof cont'd

▸ Back

Step 3

The most general unifiers of a formula which is true in an interval $[0, a]$ or $[b, 1]$ are exactly the functions whose range is exactly one of those interval.

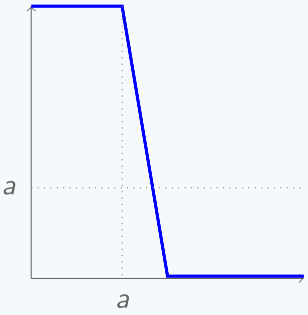


Proof cont'd

Back

Step 3

The most general unifiers of a formula which is true in an interval $[0, a]$ or $[b, 1]$ are exactly the functions whose range is exactly one of those interval.

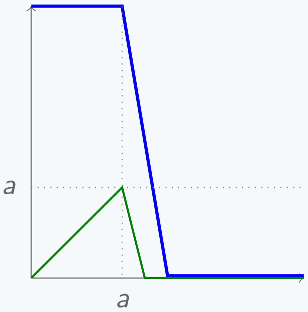


Proof cont'd

Back

Step 3

The most general unifiers of a formula which is true in an interval $[0, a]$ or $[b, 1]$ are exactly the functions whose range is exactly one of those interval.



Proof cont'd

Back

Step 4

A formula being true in a neighbourhood of 0 but not in a neighbourhood of 1 (or vice-versa) has just one mgu (i.e. unitary type).

Proof cont'd

Back

Step 4

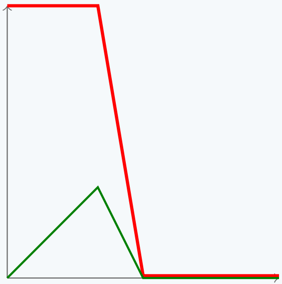
A formula being true in a neighbourhood of 0 but not in a neighbourhood of 1 (or vice-versa) has just one mgu (i.e. unitary type).

Proof cont'd

Back

Step 4

A formula being true in a neighbourhood of 0 but not in a neighbourhood of 1 (or vice-versa) has just one mgu (i.e. unitary type).

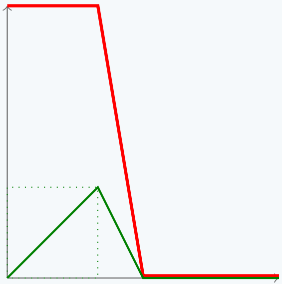


Proof cont'd

Back

Step 4

A formula being true in a neighbourhood of 0 but not in a neighbourhood of 1 (or vice-versa) has just one mgu (i.e. unitary type).

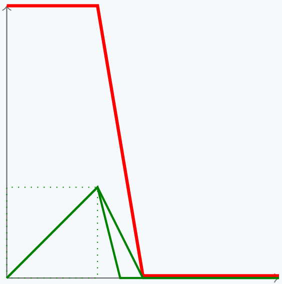


Proof cont'd

▸ Back

Step 4

A formula being true in a neighbourhood of 0 but not in a neighbourhood of 1 (or vice-versa) has just one mgu (i.e. unitary type).

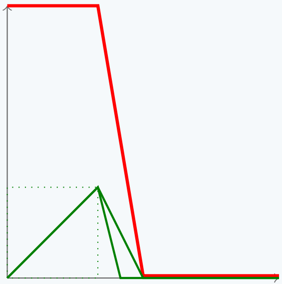


Proof cont'd

▸ Back

Step 4

A formula being true in a neighbourhood of 0 but not in a neighbourhood of 1 (or vice-versa) has just one mgu (i.e. unitary type).

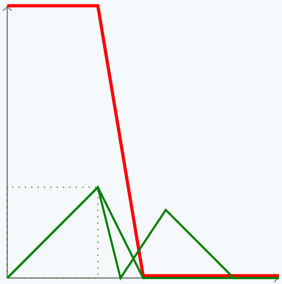


Proof cont'd

Back

Step 4

A formula being true in a neighbourhood of 0 but not in a neighbourhood of 1 (or vice-versa) has just one mgu (i.e. unitary type).



References

WM Beynon.
Duality theorems for finitely generated vector lattices.
Proceedings of the London Mathematical Society, 3(1):114, 1975.

W.M. Beynon.
Applications of duality in the theory of finitely generated lattice-ordered abelian groups.
Canad. J. Math, 29(2):243–254, 1977.

L. Cabrer and D. Mundici.
Rational polyhedra and projective lattice-ordered abelian groups with order unit.
ArXiv preprint arXiv:0907.3064, 2009.

W. Dzik.
Unification in some substructural logics of BL-algebras and hoops.
Reports in Mathematical Logic, 43:73–83, 2008.

S. Ghilardi.
Unification through projectivity.
Journal of Logic and Computation, 7(6):733–752, 1997.

S. Ghilardi.
Unification in intuitionistic logic.
Journal of Symbolic Logic, 64(2):859–880, 1999.