On the Generalized Word Problem for Finitely Presented Lattices

Alejandro Guillen

05/18/2018

Lattice Terms

Term

We define *lattice terms* over a set X, and their associated *lengths* (or *ranks*), in the following way:
Each element of X is a term of length (or rank) 1. Terms of length (or rank) 1 are called *variables*. If $t_1, \ldots, t_n$ are terms of lengths (or ranks) $k_1, \ldots, k_n$, then $(t_1 \vee \cdots \vee t_n)$ and $(t_1 \wedge \cdots \wedge t_n)$ are terms with length (or rank) $1 + k_1 + \cdots + k_n$.

Lattice Terms

Term

We define *lattice terms* over a set X, and their associated *lengths* (or *ranks*), in the following way:
Each element of X is a term of length (or rank) 1. Terms of length (or rank) 1 are called *variables*. If $t_1, \ldots, t_n$ are terms of lengths (or ranks) $k_1, \ldots, k_n$, then $(t_1 \vee \cdots \vee t_n)$ and $(t_1 \wedge \cdots \wedge t_n)$ are terms with length (or rank) $1 + k_1 + \cdots + k_n$.

In practice, we usually omit the outermost parentheses when writing down a term.

Lattice Terms

Term

We define *lattice terms* over a set X, and their associated *lengths* (or *ranks*), in the following way:
Each element of X is a term of length (or rank) 1. Terms of length (or rank) 1 are called *variables*. If $t_1, \dots, t_n$ are terms of lengths (or ranks) $k_1, \dots, k_n$, then $(t_1 \vee \cdots \vee t_n)$ and $(t_1 \wedge \cdots \wedge t_n)$ are terms with length (or rank) $1 + k_1 + \cdots + k_n$.

In practice, we usually omit the outermost parentheses when writing down a term.

Complexity of a term

The *complexity*, or *depth*, of a term t the depth of its term tree; that is, t has depth 0 if $t \in X$, and if $t = t_1 \vee \cdots \vee t_n$ or $t = t_1 \wedge \cdots \wedge t_n$, where $n > 1$, then the complexity of t is one more than the maximum of the complexities of $t_1, \dots, t_n$. We shall denote the complexity of a term t as $c(t)$.

Free Lattices

Free lattice

Let $\mathbf{F}$ be a lattice and $X \subseteq F$. We say that $\mathbf{F}$ is *freely generated by* X if X generates $\mathbf{F}$ and every map from X into any lattice $\mathbf{L}$ extends to a lattice homomorphism of $\mathbf{F}$ into $\mathbf{L}$.
Such a lattice can be shown to always exist and be unique up to isomorphism, and so is referred to as the *free lattice over* X and is denoted $\mathbf{FL}(X)$.

Free Lattices

Free lattice

Let **F** be a lattice and $X \subseteq F$. We say that **F** is *freely generated by* X if X generates **F** and every map from X into any lattice **L** extends to a lattice homomorphism of **F** into **L**.
Such a lattice can be shown to always exist and be unique up to isomorphism, and so is referred to as the *free lattice over* X and is denoted $\mathbf{FL}(X)$.

Elements of $\mathbf{FL}(X)$ as equivalence classes

If $w \in \mathbf{FL}(X)$, then w is an equivalence class of terms. Each term of this class is said to *represent* w and is called a *representative* of w.

Classifying Terms

The Word Problem for Free Lattices

Is there a procedure which determines, for arbitrary lattice terms s and t with variables from X, if the interpretations of s and t in $\mathbf{FL}(X)$ are equal?

Classifying Terms

The Word Problem for Free Lattices

Is there a procedure which determines, for arbitrary lattice terms s and t with variables from X, if the interpretations of s and t in $\mathbf{FL}(X)$ are equal?

This is equivalent to asking if there is a procedure which determines, for arbitrary lattice terms s and t with variables from X, if the interpretations of s and t in $\mathbf{L}$ are equal, for any lattice $\mathbf{L}$.

A Solution to the Word Problem for Free Lattices

Theorem (Whitman)

If s and t are terms with variables from X and $x_1, \ldots, x_n \in X$, then the truth of

$$s^{\mathbf{FL}(X)} \leq t^{\mathbf{FL}(X)} \qquad (*)$$

can be determined by applying the following rules.

1. If $s = x_i$ and $t = x_j$, then $(*)$ holds if and only $x_i = x_j$.
2. If $s = s_1 \vee \cdots \vee s_k$ is a formal join then $(*)$ holds if and only if $s_i^{\mathbf{FL}(X)} \leq t^{\mathbf{FL}(X)}$ holds for all i.
3. If $t = t_1 \wedge \cdots \wedge t_k$ is a formal meet then $(*)$ holds if and only if $s^{\mathbf{FL}(X)} \leq t_i^{\mathbf{FL}(X)}$ holds for all i.
4. If $s = x_i$ and $t = t_1 \vee \cdots \vee t_k$ is a formal join, then $(*)$ holds if and only if $x_i \leq t_j^{\mathbf{FL}(X)}$ for some j.
5. If $s = s_1 \wedge \cdots \wedge s_k$ is a formal meet and $t = x_i$, then $(*)$ holds if and only if $s_j^{\mathbf{FL}(X)} \leq x_i$ for some j.
6. If $s = s_1 \wedge \cdots \wedge s_k$ is a formal meet and $t = t_1 \vee \cdots \vee t_m$ is a formal join, then $(*)$ holds if and only if $s_i^{\mathbf{FL}(X)} \leq t^{\mathbf{FL}(X)}$ holds for some i, or $s^{\mathbf{FL}(X)} \leq t_j^{\mathbf{FL}(X)}$ holds for some j.

Canonical Form in $\mathbf{FL}(X)$

Theorem

For each $w \in \mathbf{FL}(X)$ there is a term of minimal rank representing w, unique up to commutativity. This term is called the canonical form of w.

Important Property of Canonical Form in $\mathbf{FL}(X)$

Refinement

Let $\mathbf{L}$ be a lattice and let A and B be finite subsets of L. We say that A *join refines* B and we write $A \ll B$ if for each $a \in A$ there is a $b \in B$ with $a \leq b$. The dual notion is called *meet refinement* and is denoted $A \gg B$.

Important Property of Canonical Form in $\mathbf{FL}(X)$

Refinement

Let $\mathbf{L}$ be a lattice and let A and B be finite subsets of L. We say that A *join refines* B and we write $A \ll B$ if for each $a \in A$ there is a $b \in B$ with $a \leq b$. The dual notion is called *meet refinement* and is denoted $A \gg B$.

Nonrefinable join representation

A join representation $a = a_1 \vee \cdots \vee a_n$ in a lattice is said to be a *nonrefinable join representation* if $a = b_1 \vee \cdots \vee b_m$ and $\{b_1, \ldots, b_m\} \ll \{a_1, \ldots, a_n\}$ imply $\{a_1, \ldots, a_n\} \subseteq \{b_1, \ldots, b_m\}$. Equivalently, a join representation $a = a_1 \vee \cdots \vee a_n$ is nonrefinable if it is an antichain and whenever $a = b_1 \vee \cdots \vee b_m$ and $\{b_1, \ldots, b_m\} \ll \{a_1, \ldots, a_n\}$, then $\{a_1, \ldots, a_n\} \ll \{b_1, \ldots, b_m\}$.

Important Property of Canonical Form in $\mathbf{FL}(X)$

Refinement

Let $\mathbf{L}$ be a lattice and let A and B be finite subsets of L. We say that A *join refines* B and we write $A \ll B$ if for each $a \in A$ there is a $b \in B$ with $a \leq b$. The dual notion is called *meet refinement* and is denoted $A \gg B$.

Nonrefinable join representation

A join representation $a = a_1 \vee \cdots \vee a_n$ in a lattice is said to be a *nonrefinable join representation* if $a = b_1 \vee \cdots \vee b_m$ and $\{b_1, \ldots, b_m\} \ll \{a_1, \ldots, a_n\}$ imply $\{a_1, \ldots, a_n\} \subseteq \{b_1, \ldots, b_m\}$. Equivalently, a join representation $a = a_1 \vee \cdots \vee a_n$ is nonrefinable if it is an antichain and whenever $a = b_1 \vee \cdots \vee b_m$ and $\{b_1, \ldots, b_m\} \ll \{a_1, \ldots, a_n\}$, then $\{a_1, \ldots, a_n\} \ll \{b_1, \ldots, b_m\}$.

Theorem

Let $w = w_1 \vee \cdots \vee w_n$ canonically in $\mathbf{FL}(X)$. If also $w = u_1 \vee \cdots \vee u_m$, then $\{w_1, \ldots, w_n\} \ll \{u_1, \ldots, u_m\}$. Thus, $w = w_1 \vee \cdots \vee w_n$ is the unique nonrefinable join representation of w.

The Generalized Word Problem for Free Lattices

The statement

The *generalized word problem* for free lattices asks if there is an algorithm to determine, for an arbitrary element $d \in \mathbf{FL}(X)$ and a finite subset $Y \subset \mathrm{FL}(X)$, if d is in $\mathrm{Sg}_{\mathrm{FL}(X)}(Y)$, the subalgebra generated by Y.

The Generalized Word Problem for Free Lattices

The statement

The *generalized word problem* for free lattices asks if there is an algorithm to determine, for an arbitrary element $d \in \mathbf{FL}(X)$ and a finite subset $Y \subset \mathrm{FL}(X)$, if d is in $\mathrm{Sg}_{\mathrm{FL}(X)}(Y)$, the subalgebra generated by Y.

Theorem (Freese and Nation)

The generalized word problem for free lattices is (uniformly) solvable.

A Polynomial Time Algorithm

Interlaces

Let **L** be a lattice generated by X. Let Y be a subset of L and $t(X)$ be a lattice term. We say that Y *interlaces* t iff, for every branch of the term tree of t, there are nodes t' and t'', with t'' a child of t', such that there exists $y \in Y$ between $t'(X)$ and $t''(X)$.

A Polynomial Time Algorithm

Interlaces

Let **L** be a lattice generated by X. Let Y be a subset of L and $t(X)$ be a lattice term. We say that Y *interlaces* t iff, for every branch of the term tree of t, there are nodes t' and t'', with t'' a child of t', such that there exists $y \in Y$ between $t'(X)$ and $t''(X)$.

Theorem (Guillen)

Let $d \in \mathrm{FL}(X)$ and Y be a finite subset of $\mathrm{FL}(X)$. Then $d \in \mathrm{Sg}_{\mathrm{FL}(X)}(Y)$ iff there is a term $t(X)$ representing d in $\mathrm{FL}(X)$, i.e. $d = t(X)$, such that Y interlaces t.

A Polynomial Time Algorithm (cont.)

The algorithm

Given $d \in \mathrm{FL}(X)$ and a finite $Y \subset \mathrm{FL}(X)$, we can check if $d \in \mathrm{Sg}_{\mathrm{FL}(X)}(Y)$:

1. First, test if $d \in Y$. If it is, $d \in \mathrm{Sg}_{\mathrm{FL}(X)}(Y)$ and we are done.
2. At this point, we may assume $d \notin Y$. If $d \in X$, the proof of the previous theorem shows $d \notin \mathrm{Sg}_{\mathrm{FL}(X)}(Y)$. Thus, we may assume that d is either canonically a join or a meet in $\mathrm{FL}(X)$. If $d = d_1 \vee \cdots \vee d_n$ canonically, for each branch of the term tree of $d = d_1 \vee \cdots \vee d_n$, test if the branch contains nodes d' and d'' with d'' a child of d' such that there exists $y \in Y$ between d' and d''. If this holds for every branch of the term tree $d = d_1 \vee \cdots \vee d_n$, then $d \in \mathrm{Sg}_{\mathrm{FL}(X)}(Y)$. A similar test would be applied if $d = d_1 \wedge \cdots \wedge d_m$ canonically.
3. If all of the tests above fail, then $d \notin \mathrm{Sg}_{\mathrm{FL}(X)}(Y)$.

Finitely Presented Lattices

Partially defined lattice

A *partially defined lattice* is a partially ordered set $(P, \leq)$ together with two partial functions, $\bigvee$ and $\bigwedge$, from subsets of P into P such that if $p = \bigvee S$ is one of the defined joins, then p is the least upper bound of S in $(P, \leq)$, and dually. We use $(P, \leq, \bigvee, \bigwedge)$ to denote this structure.

Finitely Presented Lattices

Partially defined lattice

A *partially defined lattice* is a partially ordered set $(P, \leq)$ together with two partial functions, $\bigvee$ and $\bigwedge$, from subsets of P into P such that if $p = \bigvee S$ is one of the defined joins, then p is the least upper bound of S in $(P, \leq)$, and dually. We use $(P, \leq, \bigvee, \bigwedge)$ to denote this structure.

Given any finite lattice presentation, there is a polynomial time algorithm to produce a finite partially defined lattice such that the finitely presented lattices generated by both are isomorphic.

Partially defined lattice

A *partially defined lattice* is a partially ordered set $(P, \leq)$ together with two partial functions, $\bigvee$ and $\bigwedge$, from subsets of P into P such that if $p = \bigvee S$ is one of the defined joins, then p is the least upper bound of S in $(P, \leq)$, and dually. We use $(P, \leq, \bigvee, \bigwedge)$ to denote this structure.

Given any finite lattice presentation, there is a polynomial time algorithm to produce a finite partially defined lattice such that the finitely presented lattices generated by both are isomorphic.

Notation

Thus, when discussing finitely presented lattices, we shall refer to $\mathrm{Free}(P, \leq, \bigvee, \bigwedge)$, or simply F_P.

The Word Problem for Finitely Presented Lattices

The statement

Is there a procedure which determines, for arbitrary lattice terms s and t with variables from P, if the interpretations of s and t in F_P are equal?

The statement

Is there a procedure which determines, for arbitrary lattice terms s and t with variables from P, if the interpretations of s and t in F_P are equal?

Ideals and filters in $(P, \leq, \bigvee, \bigwedge)$

An *ideal* I in a partially defined lattice $(P, \leq, \bigvee, \bigwedge)$ is a subset of P such that if $a \in I$ and $b \leq a$ then $b \in I$, and if $a_1, \ldots, a_k$ are in I and $a = \bigvee a_i$ is a defined join then $a \in I$. A filter in $(P, \leq, \bigvee, \bigwedge)$ is defined dually.

The Word Problem for Finitely Presented Lattices

The statement

Is there a procedure which determines, for arbitrary lattice terms s and t with variables from P, if the interpretations of s and t in F_P are equal?

Ideals and filters in $(P, \leq, \bigvee, \bigwedge)$

An *ideal* I in a partially defined lattice $(P, \leq, \bigvee, \bigwedge)$ is a subset of P such that if $a \in I$ and $b \leq a$ then $b \in I$, and if $a_1, \ldots, a_k$ are in I and $a = \bigvee a_i$ is a defined join then $a \in I$. A filter in $(P, \leq, \bigvee, \bigwedge)$ is defined dually.

Ideals and filters in F_P

If $w \in F_P$, we define $\mathrm{id}_P(w) = \{a \in P : a \leq w\}$ as the ideal of P below w. The filter of P above w, denoted $\mathrm{fil}_P(w)$, is defined dually.
If $w_1, \ldots, w_k \in F_P$, we define $\mathrm{id}_P(w_1, \ldots, w_k)$ as the ideal of P generated by $\mathrm{id}_P(w_1) \cup \cdots \cup \mathrm{id}_P(w_k)$. The filter $\mathrm{fil}_P(w_1, \ldots, w_k)$ is defined dually.

A Solution to the Word Problem for Finitely Presented Lattices

Theorem (Dean)

Let s and t be terms with variables in P. Then $s \leq t$ holds in F_P if and only if one of the following holds:

(i) $s \in P$ and $t \in P$ and $s \leq t$ in $(P, \leq)$;

(ii) $s = s_1 \vee \cdots \vee s_k$ and $\forall\, i\ s_i \leq t$;

(iii) $t = t_1 \wedge \cdots \wedge t_k$ and $\forall\, j\ s \leq t_j$;

(iv) $s \in P$ and $t = t_1 \vee \cdots \vee t_k$ and $s \in \mathrm{id}_P(t_1, \ldots, t_k)$;

(v) $s = s_1 \wedge \cdots \wedge s_k$ and $t \in P$ and $t \in \mathrm{fil}_P(s_1, \ldots, s_k)$;

(vi) $s = s_1 \wedge \cdots \wedge s_k$ and $t = t_1 \vee \cdots \vee t_m$ and $\exists\, i\ s_i \leq t$ or $\exists\, j\ s \leq t_j$ or $\exists\, a \in P\ s \leq a \leq t$.

A Solution to the Word Problem for Finitely Presented Lattices

Theorem (Dean)

Let s and t be terms with variables in P. Then $s \leq t$ holds in F_P if and only if one of the following holds:

(i) $s \in P$ and $t \in P$ and $s \leq t$ in $(P, \leq)$;

(ii) $s = s_1 \vee \cdots \vee s_k$ and $\forall\, i\ s_i \leq t$;

(iii) $t = t_1 \wedge \cdots \wedge t_k$ and $\forall\, j\ s \leq t_j$;

(iv) $s \in P$ and $t = t_1 \vee \cdots \vee t_k$ and $s \in \mathrm{id}_P(t_1, \ldots, t_k)$;

(v) $s = s_1 \wedge \cdots \wedge s_k$ and $t \in P$ and $t \in \mathrm{fil}_P(s_1, \ldots, s_k)$;

(vi) $s = s_1 \wedge \cdots \wedge s_k$ and $t = t_1 \vee \cdots \vee t_m$ and $\exists\, i\ s_i \leq t$ or $\exists\, j\ s \leq t_j$ or $\exists\, a \in P\ s \leq a \leq t$.

Lemma

If $x \in P$ and $x \leq t_1 \vee \cdots \vee t_n$ in F_P then there is a set $Y \subseteq P$ such that $Y \ll \{t_1, \ldots, t_n\}$ and $x \leq \bigvee Y$ in F_P.

Canonical Form in F_P

Adequate term

Let $(P, \leq, \bigvee, \bigwedge)$ be a finite partially defined lattice. A term t with variables from P is called *adequate* if it is an element of P, or if $t = t_1 \vee \cdots \vee t_n$ is a formal join, each t_i is adequate, and if $p \leq t$ for $p \in P$ then $p \leq t_i$ for some i. If t is formally a meet the dual condition must hold.

Canonical Form in F_P

Adequate term

Let $(P, \leq, \bigvee, \bigwedge)$ be a finite partially defined lattice. A term t with variables from P is called *adequate* if it is an element of P, or if $t = t_1 \vee \cdots \vee t_n$ is a formal join, each t_i is adequate, and if $p \leq t$ for $p \in P$ then $p \leq t_i$ for some i. If t is formally a meet the dual condition must hold.

Theorem

For each element of F_P there is an adequate term of minimal rank representing it, and this term is unique up to commutativity.

Canonical Form in F_P

Adequate term

Let $(P, \leq, \bigvee, \bigwedge)$ be a finite partially defined lattice. A term t with variables from P is called *adequate* if it is an element of P, or if $t = t_1 \vee \cdots \vee t_n$ is a formal join, each t_i is adequate, and if $p \leq t$ for $p \in P$ then $p \leq t_i$ for some i. If t is formally a meet the dual condition must hold.

Theorem

For each element of F_P there is an adequate term of minimal rank representing it, and this term is unique up to commutativity.

Canonical form

For $w \in F_P$, this shortest adequate term representing w, unique up to commutativity, is called the *canonical form* of w.

Nonrefinable Join Representations in F_P

Canonical join and meet representation

We define the *canonical join representation* of $w \in F_P$ to be $w_1 \vee \cdots \vee w_m$ if the canonical form of w is $t_1 \vee \cdots \vee t_m$ and the interpretation of t_i in F_P is w_i. It is useful to separate out the elements of P in such a representation. Thus let

$$w = w_1 \vee \cdots \vee w_n \vee x_1 \vee \cdots \vee x_k = \bigvee \bigwedge w_{ij} \vee \bigvee x_i$$

be the canonical join representation of w where $x_i \in P$, $i = 1, \ldots, k$, and the canonical meet representation of w_i is $w_i = \bigwedge w_{ij}$.

Nonrefinable Join Representations in F_P

Canonical join and meet representation

We define the *canonical join representation* of $w \in F_P$ to be $w_1 \vee \cdots \vee w_m$ if the canonical form of w is $t_1 \vee \cdots \vee t_m$ and the interpretation of t_i in F_P is w_i. It is useful to separate out the elements of P in such a representation. Thus let

$$w = w_1 \vee \cdots \vee w_n \vee x_1 \vee \cdots \vee x_k = \bigvee \bigwedge w_{ij} \vee \bigvee x_i$$

be the canonical join representation of w where $x_i \in P$, $i = 1, \ldots, k$, and the canonical meet representation of w_i is $w_i = \bigwedge w_{ij}$.

Theorem

Let the canonical join representation for w be as above. Every join representation of w can be refined to a nonrefinable join representation of w. If $w = v_1 \vee \cdots \vee v_m$ in F_P then there exist $y_1, \ldots, y_r \in P$ such that

$$w = w_1 \vee \cdots \vee w_n \vee y_1 \vee \cdots \vee y_r$$

and

$$\{w_1, \ldots, w_n, y_1, \ldots, y_r\} \ll \{v_1, \ldots, v_m\}.$$

Every nonrefinable join representation of w contains $\{w_1, \ldots, w_n\}$ and also contains every x_i which is join irreducible.

Some nice consequences

Corollary

If $w = w_1 \vee \cdots \vee w_n \vee x_1 \vee \cdots \vee x_k$ is the canonical join representation of $w \in F_P$ where $x_i \in P$, then every nonrefinable join representation of w has the form $\{w_1, \ldots, w_n, y_1, \ldots, y_r\}$ for some $y_1, \ldots, y_r$ in P.

Some nice consequences

Corollary

If $w = w_1 \vee \cdots \vee w_n \vee x_1 \vee \cdots \vee x_k$ is the canonical join representation of $w \in F_P$ where $x_i \in P$, then every nonrefinable join representation of w has the form $\{w_1, \ldots, w_n, y_1, \ldots, y_r\}$ for some $y_1, \ldots, y_r$ in P.

Corollary

Every nonrefinable join representation of $w \in F_P$ refines the canonical join representation of w.

Why can't we use the same polynomial time algorithm for $\mathrm{FL}(X)$ and F_P?

One issue

In $\mathrm{FL}(X)$, a generator $x \in X$ is both join and meet irreducible. However, a generator $p \in P$ could either be a defined join or defined meet in $(P, \leq, \bigvee, \bigwedge)$. So, we may need to check defined join representations or defined meet representations of a generator $p \in P$.

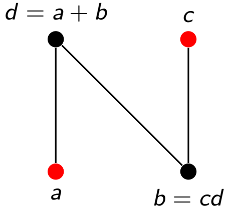
Why can't we use the same polynomial time algorithm for $\mathrm{FL}(X)$ and F_P?

One issue

In $\mathrm{FL}(X)$, a generator $x \in X$ is both join and meet irreducible. However, a generator $p \in P$ could either be a defined join or defined meet in $(P, \leq, \bigvee, \bigwedge)$. So, we may need to check defined join representations or defined meet representations of a generator $p \in P$.

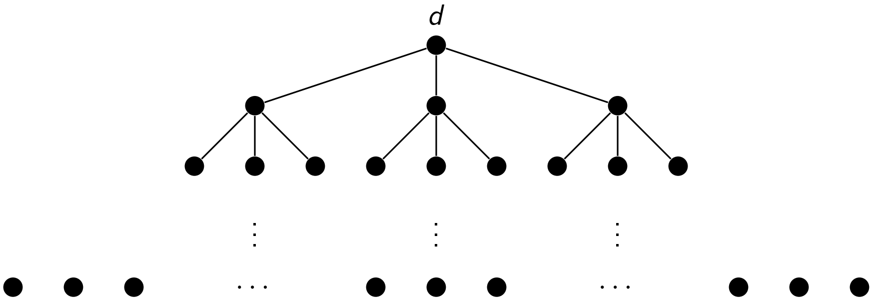
Example

Let $P = \{a, b, c, d\}$ with order given in the figure below and the single defined join $d = a + b$ and the single defined meet $b = cd$.
Take $Y = \{a, c\}$.



Also, there could be too many join or meet representations to search through

Let P, with the obvious order and no defined meets, be given below. For each $p \in P$, except for the minimal elements of P, there are three elements directly below it in P; call them q, r, and s. P has defined joins $p = q + r = q + s = r + s$.



What can be salvaged for F_P?

Interval

If $a \leq b$ in a lattice $\mathbf{L}$, then we let b/a denote the interval $\{x \in L : a \leq x \leq b\}$.

Theorem (Guillen)

Let P be finite, $Y \subseteq F_P$, $d \in \mathrm{Sg}_{F_P}(Y) - Y$, and let $d = w_1 \vee \ldots \vee w_n \vee x_1 \vee \ldots \vee x_k$ be the canonical join representation of $d \in F_P$. Then $d/w_i \cap \mathrm{Sg}_{F_P}(Y) \neq \emptyset$ for $1 \leq i \leq n$.

Looking instead for a syntactic algorithm for F_P

Lemma (Guillen)

Let $d = d_1 \vee \cdots \vee d_n \in F_P$ be a nonrefinable join representation and, for some d_i, there exists $p \in P$ such that $d_i \leq p \leq d$. Then $d_i \in P$.

Looking instead for a syntactic algorithm for F_P

Lemma (Guillen)

Let $d = d_1 \vee \cdots \vee d_n \in F_P$ be a nonrefinable join representation and, for some d_i, there exists $p \in P$ such that $d_i \leq p \leq d$. Then $d_i \in P$.

Theorem (Guillen)

Let $d \in F_P$ and Y be a finite subset of F_P. Then $d \in \mathrm{Sg}_{F_P}(Y)$ iff either $d \in Y$, there exists a nonrefinable join representation of d, call it $\{d_1, \ldots, d_n\}$, such that, for each d_i, either

(a) $d/d_i \cap Y \neq \emptyset$,

(b) $d_i \in \mathrm{Sg}_{F_P}(Y)$, or

(c) $d_i \in P$ and there exists $p \in P$ such that $d_i \leq p \leq d$ and $p/d_i \cap \mathrm{Sg}_{F_P}(Y) \neq \emptyset$,

or there exists a non-upper refinable meet representation of d such that the duals of (a), (b), and (c) hold for the elements of this non-upper refinable representation.

A syntactic algorithm for F_P

Suppose there is an oracle that can decide, for all $p, q \in P$ with $p \leq q$ if there exists $f \in \mathrm{Sg}_{F_P}(Y)$ such that $p \leq f \leq q$. Given $d \in F_P$ and a finite $Y \subseteq F_P$, we can check if $d \in \mathrm{Sg}_{F_P}(Y)$:

1. First, test if $d \in Y$.
2. If $d \notin Y$, then d is either a join or a meet in F_P.
   (a) If d is a join, for each nonrefinable join representation $\{d_1, \ldots, d_n\}$ of d, and for each joinand d_i, test if one of the following holds for d_i:
      (i) $d/d_i \cap Y \neq \emptyset$.
      (ii) $d_i \in \mathrm{Sg}_{F_P}(Y)$. Note that this step is a reduction since $c(d_i) < c(d)$.
      (iii) $d_i \in P$ and there exists $p \in P$ such that $d_i \leq p \leq d$ and $p/d_i \cap \mathrm{Sg}_{F_P}(Y) \neq \emptyset$. Note that if in fact we find that $d_i \in P$, for each $p \in d/d_i$ we use our oracle to test if there exists $f \in \mathrm{Sg}_{F_P}(Y)$ such that $d_i \leq f \leq p$.

   If we are able to find a nonrefinable join representation for d such that one of the above holds for each of the joinands, then $d \in \mathrm{Sg}_{F_P}(Y)$.
   (b) If d is a meet, a similar test would be applied.
3. If all of the tests above fail, then $d \notin \mathrm{Sg}_{F_P}(Y)$.

Fin

Thank you