Hardness of Enumerating Pseudo-Intents in the Lectic Order

Felix Distel

Theoretical Computer Science, TU Dresden, Germany
felix@tcs.inf.tu-dresden.de

Abstract. We investigate the complexity of enumerating pseudo-intents in the lectic order. We look at the following decision problem: Given a formal context and a set of n pseudo-intents determine whether they are the lectically first n pseudo-intents. We show that this problem is coNP-hard. We thereby show that there cannot be an algorithm with a good theoretical complexity for enumerating pseudo-intents in a lectic order. In a second part of the paper we introduce the notion of minimal pseudo-intents, i. e. pseudo-intents that do not strictly contain a pseudo-intent. We provide some complexity results about minimal pseudo-intents that are readily obtained from the previous result.

1 Introduction

The so-called stem base or Duquenne-Guigues Base from Formal Concept Analysis (FCA, [5]) plays an important rôle within FCA [6]. It has applications both within FCA as well as other fields such as Description Logics (DL) (in particular in knowledge base completion [1]). Therefore it is not surprising that it has been of major interest in the FCA community since its introduction.

In order to compute the Duquenne-Guigues Base of a formal context one must compute its pseudo-intents. The most well known algorithm for computing pseudo-intents is the *Next-Closure-Algorithm* [4]. It produces all concept intents and all pseudo-intents of a given formal context in a lexicographic order (called the lectic order). Another less well known algorithm has been introduced in 2007 [9, 10]. It computes concept intents and pseudo-intents by starting with a set containing a single attribute and then incrementally adding attributes.

Both algorithms compute not only pseudo-intents but also concept intents. It is not difficult to see that the number of concept intents can be exponential in the number of pseudo-intents. As an example consider a series of contexts $\mathbb{K}_n = (G_n, M_n, I_n)$ where $M_n = \{1, \ldots, n\}$ and all subsets of M_n with cardinality $n-2$ are object intents. This context has $\frac{1}{2}n(n-1)$ objects and n attributes. The pseudo-intents of $\mathbb{K}_n$ are exactly the sets of cardinality $n-1$. All sets of cardinality less than $n-1$ are concept intents. This means that there are 2^n-n-1 concept intents while there are only n pseudo-intents. The case $n = 4$ is shown in Table 1.

This shows that there is a problem with the known algorithms for computing pseudo-intents. In many practical applications such as attribute exploration or

Table 1. A Formal Context with 4 Pseudo-intents and $2^4 - 4 - 1$ Concept Intents

	1	2	3	4
g_1	X	X		
g_2	X		X	
g_3	X			X
g_4		X	X	
g_5		X		X
g_6			X	X

knowledge base completion one is not interested in concept intents but only in pseudo-intents. Yet, the above example shows that in the worst case the time needed to enumerate all pseudo-intents can be exponential in the size of the output, i. e. the number of pseudo-intents, when using one of the two known algorithms.

This raises the question whether it is theoretically possible to find more efficient algorithms for computing pseudo-intents. It is known that the number of pseudo-intents can be exponential in the size of the incidence relation of the context [7]. From this it immediately follows that there cannot be an algorithm that enumerates pseudo-intents in polynomial time in the size of the input (which would be the incidence relation).

For problems where the size of the output can be large in the size of the input other measures of complexity have been developped. One possibility is to take into account not only the size of the input, but also the size of the output. An algorithm is said to run in *output polynomial time* if it enumerates the solutions in time polynomial in the size of the input *and* the output. In previous work a relationship between the problem of enumerating pseudo-intents and the so-called transversal hypergraph problem (TransHyp, [2]) has been discovered. TransHyp is known to be in coNP but so far no hardness result has been shown. It is most likely not coNP-hard because it can be solved in $n^{o(\log n)}$ time [3]. It is also not known whether TransHyp is in P. It has been shown that pseudo-intents cannot be enumerated in output-polynomial time unless TransHyp is in P [11, 12].

For someone who wants to apply attribute exploration in practice the most interesting measure of complexity is the delay between the computation of one pseudo-intent and the next. During this time the expert must wait unproductively for the next question to show up. With the known algorithms the delay can be exponential in the size of the input – and even in the size of the output. An enumeration algorithm is said to run with *polynomial delay* if the time between the enumeration of one solution and the next is polynomial in the size of the input.

The central question in this paper is whether it is possible to enumerate pseudo-intents in the lectic order with polynomial delay. We prove that the problem of checking whether a given set of n pseudo-intents is the set of the lectically first n pseudo-intents is coNP-hard. We conclude, it is impossible to enumerate pseudo-intents in the lectic order with polynomial delay unless P = NP.

In a second part of the paper we look at a subclass of the class of pseudo-intents that we call minimal pseudo-intents. We show that it is tractable to check whether a given set is a minimal pseudo-intent. We also provide an algorithm that given a context will output a minimal pseudo-intent in polynomial time. We show that, surprisingly, it is not even possible to enumerate minimal pseudo-intents in output polynomial time unless P = NP.

2 Preliminaries

We briefly introduce the basic notions of formal concept analysis. A *formal context* is a tuple (G, M, I) where G and M are finite sets and $I \subseteq G \times M$ is a binary relation. The elements of G are called objects and elements of M are called attributes. FCA provides two *derivation operators* that are both denoted by $\cdot'$. For a set of objects $A \subseteq G$ one defines $A' = \{m \in M \mid \forall g \in A : gIm\}$. Analogously, for a set $B \subseteq M$ one defines $B' = \{g \in G \mid \forall m \in B : gIm\}$. Applying the two derivation operators successively yields the closure operators $\cdot''$. The $\cdot''$-closed subsets of M are called *concept intents*, while the $\cdot''$-closed subsets of G are called *concept extents*. A concept intent A is called *object intent* if it can be written as the closure of a singleton set $A = \{g\}'$, $g \in G$. Given a context (G, M, I) and a set $A \subseteq M$ one can check in time polynomial in the size of I and A whether A is a concept intent. The following Lemma is common knowledge in FCA.

Lemma 1. *A set of attributes $A \subseteq M$ is a concept intent if and only if it can be written as an intersection of object intents, i. e. there is a set $B \subseteq G$ such that*

$$A = \bigcap_{g \in B} \{g\}'.$$

An interesting research area in FCA are dependencies between sets of attributes. The simplest form of such a dependency is an implication $A \to B$, $A, B \subseteq M$. A set of attributes $D \subseteq M$ respects $A \to B$ if $A \not\subseteq D$ or $B \subseteq D$. $A \to B$ holds in the context (G, M, I) if all object intents respect $A \to B$.

Let $\mathcal{L}$ be a set of implications. We say that $A \to B$ follows semantically from $\mathcal{L}$ if and only if each subset $D \subseteq M$ that respects all implications from $\mathcal{L}$ also respects $A \to B$. $\mathcal{L}$ is an implicational base for (G, M, I) if it is

- *sound*, i. e. all implications from $\mathcal{L}$ hold in (G, M, I), and
- *complete*, i. e. all implications that hold in (G, M, I) follow from $\mathcal{L}$.

In [6] a minimum cardinality base, which is called the *Duquenne-Guigues-Base*, has been introduced. The premises of the implications in the Duquenne-Guigues-Base are so-called pseudo-intents. $P \subseteq M$ is a *pseudo-intent* if P is not a concept intent and $Q'' \subseteq P$ holds for every pseudo-intent Q that is a proper subset of P.

The Duquenne-Guigues-Base consists of all implications $P \to P''$, where P is a pseudo-intent.

The well-known algorithm *Next-Closure* computes all pseudo-intents and concept intents in the lectic order [4]. The lectic order is defined as follows. Let a strict total order $<$ on the set M of attributes be given. Let $A, B \subseteq M$ be two sets of attributes. Define

$$A < B :\Leftrightarrow \exists i \in B - A : A \cap \{j \in M \mid j < i\} = B \cap \{j \in M \mid j < i\}.$$

If $A < B$ holds then we say that A *is lectically smaller than* B.

3 Enumerating Pseudo-Intents in a Lectic Order

We have seen that the delay between the computation of two pseudo-intents is important. The two known algorithms do not have good theoretical properties. Both of them compute not only pseudo-intents, but also concept intents. For a given context the number of concept intents can be exponential in the number of pseudo-intents. That means that in the worst case, the algorithm would compute an exponential number of concept intents before the next pseudo-intent shows up. We ask whether it is possible to come up with an algorithm that behaves better. The answer is, if we require that the pseudo-intents be computed in the lectic order then there cannot be an algorithm with polynomial delay unless P = NP. We prove this by examining the following decision problem.

*Problem 1 (Lectically first pseudo-intents (*FirstPI*)). Input:* A formal context $\mathbb{K} = (G, M, I)$ and pseudo-intents $P_1, \dots P_n$.
Question: Are $P_1, \dots, P_n$ the n lectically first pseudo-intents of $\mathbb{K}$?

The dual problem to FirstPI would be "Given a formal context $\mathbb{K}$ and pseudo-intents $P_1, \dots, P_n$ check if $P_1, \dots, P_n$ are not the lectically first pseudo-intents of $\mathbb{K}$.". This problem can be characterized as follows.

Proposition 1. *$P_1, \dots, P_n$ are not the n lectically first pseudo-intents of $\mathbb{K}$ iff there is a set $Q \subseteq M$ such that*

1. *Q is lectically smaller than P_j for some $j \in \{1, \dots, n\}$, and*
2. *Q is not a concept intent, and*
3. *for all $i \in \{1, \dots, n\}$ either $P_i \not\subseteq Q$ or $P_i'' \subseteq Q$.*

Proof. if: Because Q is not a concept intent there must be a pseudo-intent P of $\mathbb{K}$ such that $P \subseteq Q$ but $P'' \not\subseteq Q$. Because of 3 it holds that $P \notin \{P_1, \dots, P_n\}$. P is lectically smaller than P_j because Q is lectically smaller than P_j and $P \subseteq Q$. Thus $P_1, \dots, P_n$ are not the lectically smallest pseudo-intents of $\mathbb{K}$.

only if: Let P be a pseudo-intent that is lectically smaller than P_j, for some $j \in \{1, \dots, n\}$ but not contained in $\{P_1, \dots, P_n\}$. Then $Q = P$ satisfies the three conditions 1 to 3.

Lemma 2 (Containment in coNP). FirstPI *is in* coNP.

Proof. We show that the dual problem of FirstPI can be decided in non-deterministic polynomial time.

Whether a set $Q \subseteq M$ satisfies conditions 1 to 3 from Proposition 1 can be checked in time polynomial in the size of $\mathbb{K}$ and $P_1, \ldots, P_n$. In order to decide whether $P_1, \ldots, P_n$ are not the lectically first pseudo-intents of $\mathbb{K}$ one can non-deterministically guess a subset $Q \subseteq M$ and then check in polynomial time whether it satisfies 1 to 3. Hence the dual problem of FirstPI is in NP and thus FirstPI is in coNP.

For our hardness proof we use a reduction from the tautology problem, the prototypical coNP-complete problem.

*Problem 2 (*Tautology*). Input:* A boolean DNF-formula $f(p_1, \ldots, p_m) = (x_{11} \wedge \cdots \wedge x_{1l_1}) \vee \cdots \vee (x_{k1} \wedge \cdots \wedge x_{kl_k})$, where $x_{ij} \in \{p_1, \ldots, p_m\} \cup \{\neg p_1, \ldots, \neg p_m\}$.
Question: Is f a tautology?

Tautology is coNP-complete, even with the restriction that f be in DNF. This is because f is a tautology iff $\neg f$ is unsatisfiable. If f is in DNF then $\neg f$ can be transformed to CNF in linear time. Checking if $\neg f$ is unsatisfiable is the dual problem of the Satisfiability Problem for boolean CNF formulae, which is, of course, NP-complete.

We prove that FirstPI is harder than Tautology by reduction. Let an instance f of Tautology be given. Let f be the DNF-formula $f(p_1, \ldots, p_m) = D_1 \vee \cdots \vee D_k$, where $D_i = (x_{i1} \wedge \cdots \wedge x_{il_i})$ and $x_{ij} \in \{p_1, \ldots, p_m\} \cup \{\neg p_1, \ldots, \neg p_m\}$ for all $i \in \{1, \ldots, k\}$ and all $j \in \{1, \ldots, l_i\}$. We define a context $\mathbb{K}$ as follows.

Let M be the set $M = \{\alpha_1, \ldots, \alpha_m, t_1, \ldots, t_m, f_1, \ldots f_m\}$. We define a total order $<$ on the elements of M as follows

$$\alpha_1 < \cdots < \alpha_m < t_1 < f_1 < \cdots < t_m < f_m.$$

For every $i \in \{1, \ldots, k\}$ define a set

$$\begin{aligned} A_i = M &- \{f_j \mid p_j \text{ occurs in } D_i \text{ as a positive literal}\} \\ &- \{t_j \mid p_j \text{ occurs in } D_i \text{ as a negative literal}\} \\ &- \{\alpha_j \mid p_j \text{ occurs in } D_i\} \end{aligned}$$

and furthermore for every $i \in \{1, \ldots, k\}$ and every $j \in \{1, \ldots, m\}$ let F_{ij} and T_{ij} be the sets $T_{ij} = A_i - \{f_j, \alpha_j\}$, $F_{ij} = A_i - \{t_j, \alpha_j\}$. Define the set of objects G to be $G = \{u_1, \ldots, u_{2m}\} \cup \{g_{T_{ij}} \mid i \in \{1, \ldots, k\}, j \in \{1, \ldots, m\}\} \cup \{g_{F_{ij}} \mid i \in \{1, \ldots, k\}, j \in \{1, \ldots, m\}\}$. The relation I is defined so that every object $g_{T_{ij}}$ has all the attributes that are contained in the set T_{ij} and analogously for $g_{F_{ij}}$. Furthermore I is such that every singleton set $\{t_i\}$ or $\{f_i\}$ occurs as the concept intent of some u_i. More formally, we define

$$\begin{aligned} I = &\{(u_{2i-1}, t_i) \mid i \in \{1, \ldots, m\}\} \cup \{(u_{2i}, f_i) \mid i \in \{1, \ldots, m\}\} \\ &\cup \{(g_{F_{ij}}, x) \mid i \in \{1, \ldots, k\}, j \in \{1, \ldots, m\}, x \in F_{ij}\} \\ &\cup \{(g_{T_{ij}}, x) \mid i \in \{1, \ldots, k\}, j \in \{1, \ldots, m\}, x \in T_{ij}\}. \end{aligned}$$

	α_1	$\dots$	α_m	t_1	f_1	t_2	f_2	$\dots$	t_m	f_m
u_1				X						
$\vdots$						$\dots$				
$\vdots$								$\dots$		
u_{2m}										X
$g_{T_{11}}$					T_{11}					
$\vdots$					$\vdots$					
$g_{T_{1m}}$					T_{1m}					
$\vdots$					$\vdots$					
$g_{T_{k1}}$					T_{k1}					
$\vdots$					$\vdots$					
$g_{T_{km}}$					T_{km}					
$g_{F_{11}}$					F_{11}					
$\vdots$					$\vdots$					
$g_{F_{1m}}$					F_{1m}					
$\vdots$					$\vdots$					
$g_{F_{k1}}$					F_{k1}					
$\vdots$					$\vdots$					
$g_{F_{km}}$					F_{km}					

Table 2. Context $\mathbb{K}$

There are $2mk + 2m$ objects and $3m$ attributes, so the size of the context is $\mathcal{O}(m^2k + m^2)$. As sets $P_1, \dots, P_m$ we define $P_i = \{t_i, f_i\}$ for all $i \in \{1, \dots, m\}$.

The reduction may look complicated at first glance. The basic ideas in the design of the reduction are the following.

- Any assignment of truth values ϕ corresponds naturally to a subset of $\{t_1, f_1, \dots, t_m, f_m\}$, namely the set

$$S_\phi := \{t_i \mid \phi(p_i) = \texttt{true}\} \cup \{f_i \mid \phi(p_i) = \texttt{false}\}. \qquad (1)$$

- If ϕ makes D_i true then S_ϕ is a subset of A_i.
- If S_ϕ is a subset of A_i then S_ϕ is a concept intent.

To formally prove that this is a reduction from Tautology to FirstPI we need to show two things. First, we need to show that what we have obtained is really an instance of FirstPI and second, we need to show that f is a "Yes"-instance of Tautology if and only if $(\mathbb{K}, \{P_1, \dots, P_m\})$ is a "Yes"-instance of FirstPI.

Lemma 3. $(\mathbb{K}, \{P_1, \dots, P_m\})$ *is an instance of* FirstPI

Proof. All we have to show is that all P_i are pseudo-intents. Note that all strict subsets of P_i are concept intents in $\mathbb{K}$ (this is because all singleton subsets $\{t_i\}$ and $\{f_i\}$ are object intents of some u_i). To see that $\alpha_i \in P_i''$ and thus $P_i'' \neq P_i$ consider the sets A_r for $r \in \{1, \ldots, k\}$. If $P_i = \{t_i, f_i\} \subseteq A_r$ then by definition of A_r p_i does not occur in D_i. Therefore $\alpha_i \in A_r$. Let $s \in \{1, \ldots, m\}$ be an index of some set T_{rs}. If $P_i \subseteq T_{rs}$ then $P_i \subseteq A_r$ and $i \neq s$. Then $\alpha_i \in A_r$ holds and because $i \neq s$ it follows that $\alpha_i \in T_{rs} = A_r - \{f_s, \alpha_s\}$ Analogously $\alpha_i \in F_{rs}$ if $P_i \subseteq F_{rs}$. Therefore all objects that have all attributes from P_i also have α_i as an attribute and thus $\alpha_i \in P_i''$. Therefore $P_i'' \neq P_i$ must hold. Hence P_i is a pseudo-intent. Therefore $(\mathbb{K}, \{P_1, \ldots, P_m\})$ is an instance of FIRSTPI.

We show that $\mathbb{K}$ has a pseudo-intent that is lectically smaller than P_1 if and only if f is not a tautology. Let ϕ be an assignment that maps all p_i to a truth value in $\{\texttt{true}, \texttt{false}\}$. Let S_ϕ be defined as in (1). Note that S_ϕ contains exactly one element of $\{t_i, f_i\}$ for every $i \in \{1, \ldots, m\}$.

Lemma 4. *There is some $i \in \{1, \ldots, k\}$ for which $S_\phi \subseteq A_i$ if and only if $f(\phi(p_1), \ldots, \phi(p_m)) = \texttt{true}$.*

Proof. *only-if*: Let ϕ be such that $S_\phi \subseteq A_i$. Then by definition of A_i it holds that $f_j \notin S_\phi$, and thus $\phi(p_j) = \texttt{true}$, for all p_j that occur as positive literals in D_i (we have removed f_j from A_i). Analogously, $\phi(p_j) = \texttt{false}$ for all p_j that occur as negative literals. Hence all literals in D_i evaluate to $\texttt{true}$ and therefore both D_i and the whole formula evaluate to $\texttt{true}$.

if: Now let ϕ be an assignment that makes f true. Since f is in DNF it evaluates to $\texttt{true}$ iff at least one of the k implicants evaluates to true. Let D_i for some $i \in \{1, \ldots, k\}$ be an implicant that evaluates to true. Then $\phi(p_j) = \texttt{true}$ for all p_j that occur as positive literals D_i and $\phi(p_j) = \texttt{false}$ for all p_j that occur as negative literals in D_i. By definition of A_i and S_ϕ this implies $S_\phi \subseteq A_i$.

Lemma 5. *If $S_\phi \subseteq A_i$ then S_ϕ can be written as*

$$S_\phi = \bigcap_{\substack{j \in \{1, \ldots, m\} \\ \phi(p_j) = \texttt{true}}} T_{ij} \cap \bigcap_{\substack{j \in \{1, \ldots, m\} \\ \phi(p_j) = \texttt{false}}} F_{ij}$$

Proof. We denote the right-hand side of the above equation by R. By definition S_ϕ does not contain f_j if $\phi(p_j) = \texttt{true}$. Thus $S_\phi \subseteq A_i - \{f_j, \alpha_j\} = T_{ij}$ for all p_j for which $\phi(p_j) = \texttt{true}$. Likewise, $S_\phi \subseteq A_i - \{t_j, \alpha_j\} = F_{ij}$ for all p_j for which $\phi(p_j) = \texttt{false}$. Thus $S_\phi \subseteq R$. To prove the other inclusion consider some $x \in R$. For every $j \in \{1, \ldots, m\}$ it holds that $\alpha_j \notin F_{ij}$ and $\alpha_j \notin T_{ij}$. If $\phi(p_j) = \texttt{true}$ then $R \subseteq T_{ij}$, otherwise $R \subseteq F_{ij}$. So in either case $\alpha_j \notin R$. Therefore $x \neq \alpha_j$ holds for all $j \in \{1, \ldots, m\}$. Assume that $x = t_j$ for some j. Then $\phi(p_j) = \texttt{true}$ must hold, for otherwise R would be a subset of F_{ij} which does not contain t_j. Now $\phi(p_j) = \texttt{true}$ implies $x = t_j \in S_\phi$. The case $x = f_j$ for some j can be treated analogously. Thus for every $x \in R$ it holds that $x \in S_\phi$ and thus $R \subseteq S_\phi$. Hence $S_\phi = R$.

Lemma 6. *f is a tautology if and only if for all assignments ϕ the set S_ϕ is a concept intent of $\mathbb{K}$.*

Proof. Let us start by proving the *if*-direction. Assume that there is an assignment ϕ that makes f false. From Lemma 4 it follows that $S_\phi \not\subseteq A_i$ for all $i \in \{1, \ldots, k\}$. But then no object in G has all the attributes in S_ϕ because every object intent is either a singleton set or a subset of some A_i. Therefore $S''_\phi = M$ and thus S_ϕ is not a concept intent. This contradicts the assumption and thus f must be a tautology.

For the *only if*-direction assume that there is some ϕ for which S_ϕ is not a concept intent. We know that the intersection of concept intents is also a concept intent. This implies in particular that S_ϕ cannot be written as the intersection of object intents. From Lemma 5 it follows that $S_\phi \not\subseteq A_i$ for all $i \in \{1, \ldots, k\}$. But then Lemma 4 shows that ϕ makes f false. This is a contradiction to the assumption that f is a tautology. Therefore S_ϕ must be a concept intent for all ϕ.

Lemma 7. *P_1, ..., P_m are the lectically smallest pseudo-intents of $\mathbb{K}$ if and only if for all assignments ϕ the set S_ϕ is a concept intent in $\mathbb{K}$.*

Proof. only-if-direction: Assume that some S_ϕ is not a concept intent. Then S_ϕ has a subset $P \subseteq S_\phi$ which is a pseudo-intent. Obviously P is lectically smaller than P_1. Also P must be different from all the P_i because S_ϕ does not include any of the P_i. This is a contradiction to the assumption that P_1, ..., P_m are the lectically smallest pseudo-intents of $\mathbb{K}$.

if-direction: Let $Q \subseteq M$ be a set of attributes that is lectically smaller than P_1. If Q would contain some α_i then it would be lectically larger than P_1. Therefore Q must be a subset of $\{t_1, f_1, \ldots, t_m, f_m\}$. If there is some $i \in \{1, \ldots, m\}$ such that $P_i \subseteq Q$ then $\alpha_i \in P''_i - Q$ and thus $P''_i \not\subseteq Q$. Therefore Q is not equal to P_i or a pseudo-intent. If $P_i \not\subseteq Q$ for all $i \in \{1, \ldots, m\}$ then define:

$$\phi_t(p_i) = \begin{cases} \texttt{true} & t_i \in Q \\ \texttt{false} & f_i \in Q \\ \texttt{true} & \text{otherwise} \end{cases} \qquad \phi_f(p_i) = \begin{cases} \texttt{true} & t_i \in Q \\ \texttt{false} & f_i \in Q \\ \texttt{false} & \text{otherwise} \end{cases}$$

Both ϕ_t and ϕ_f are well-defined since Q cannot contain both t_i and f_i for any i. With ϕ_t and ϕ_f defined as above it holds that $Q = S_{\phi_t} \cap S_{\phi_f}$. Since all S_ϕ are concept intents the intersection of S_{ϕ_t} and S_{ϕ_f} must also be a concept intent. Therefore Q cannot be a pseudo-intent.

Theorem 1 (Hardness of FirstPI). FIRSTPI *is* coNP*-hard.*

Proof. From Lemma 6 and Lemma 7 it follows that P_1,...,P_m are the lectically first pseudo-intents in $\mathbb{K}$ if and only if f is a tautology. Since the reduction can be done in polynomial time it follows that FIRSTPI is coNP-hard.

Corollary 1. FIRSTPI *is* coNP*-complete.*

What does this mean for the problem of enumerating pseudo-intents in the lectic order? Assume that there is an algorithm $\mathcal{A}$ that given a context enumerates its pseudo-intents in the lectic order and with polynomial delay. That means that there is a polynomial $p(|G|, |M|)$ such that the delay between the computation of one pseudo-intent and the next is bounded by $p(|G|, |M|)$. Here $|M|$ denotes the number of attributes and $|G|$ denotes the number of objects in the context.

In order to solve FIRSTPI for an input $((G, M, I), \{P_1, \ldots, P_n\})$ we can construct a new algorithm $\mathcal{A}'$ from $\mathcal{A}$. $\mathcal{A}'$ lets $\mathcal{A}$ run for time $n \cdot p(|G|, |M|)$. After that time $\mathcal{A}$ will have computed the lectically first n pseudo-intents (and possibly some more, but these are not interesting). If these lectically first n pseudo-intents are identical to $P_1, \ldots, P_n$ then $\mathcal{A}'$ returns "Yes", otherwise it returns "No". The runtime of $\mathcal{A}'$ is bounded by $n \cdot p(|G|, |M|)$ and thus polynomial in the size of the input. Since FIRSTPI is coNP-hard, it cannot be solved in polynomial time unless $\mathrm{P} = \mathrm{NP}$.

Theorem 2. *Pseudo-intents cannot be enumerated in the lectical order with polynomial delay, unless* $\mathrm{P} = \mathrm{NP}$.

4 Minimal Pseudo-Intents

4.1 Introducing Minimal Pseudo-Intents

We say that P is a *minimal pseudo-intent of* $\mathbb{K}$ if P is a pseudo-intent of $\mathbb{K}$ and P does not contain any other pseudo-intent of $\mathbb{K}$. An equivalent definition is the following.

Definition 1 (Minimal Pseudo-Intent). *A* minimal pseudo-intent *of a context is a set* $P \subseteq M$ *such that*

- *P is not a concept intent, and*
- *every strict subset $S \subset P$ is a concept intent.*

Minimal pseudo-intents play a special rôle among the pseudo-intents of a given context. While the Duquenne-Guigues base is the most well known implication base, a given formal context $\mathbb{K}$ may have other implication bases. There may even be several implication bases with minimal cardinality. Minimal pseudo-intents are important since they have to occur as premises in all bases of a context, not just in the Duquenne-Guigues base.

To clarify this assume that $\mathcal{L}$ is a set of implications of the context $\mathbb{K}$. Let P be a minimal pseudo-intent of $\mathbb{K}$. Assume that $\mathcal{L}$ does not contain an implication whose left-hand side is P. Since all strict subsets of P are concept intents, there can be no implication $C \to D$ in $\mathcal{L}$ where $C \subseteq P$ but $D \not\subseteq P$. But then $P \to P''$ does not follow from $\mathcal{L}$ and thus $\mathcal{L}$ is not a concept intent.

Lemma 8. *If $\mathcal{L}$ is an implication base of a given context $\mathbb{K} = (G, M, I)$ and P is a minimal pseudo-intent of $\mathbb{K}$ then $\mathcal{L}$ contains an implication $P \to D$, $D \subseteq M$, whose premise is P.*

This shows that any algorithm that computes an implication base for a context inevitably has to compute all minimal pseudo-intent. This makes them an interesting subject for further research.

Given a context $\mathbb{K} = (G, M, I)$ and a set of attributes $P \subseteq M$ it is not hard to tell whether P is a minimal pseudo-intent. By definition, P is a minimal pseudo-intent if and only if it is not a concept intent and all its strict subsets are concept intents.

Lemma 9. *All strict subsets of P are concept intents if and only if all sets $P \setminus \{m\}$, $m \in P$, are concept intents.*

Proof. Assume that all sets of the form $P \setminus \{m\}$, $m \in P$, are concept intents. Let $S \subsetneq P$ be a strict subset. S can be written as the intersection

$$S = \bigcap_{m \in P \setminus S} (P \setminus \{m\}).$$

Since the intersection of concept intents is itself a concept intent S must be a concept intent. This proves the "if"-direction. The "only if"-direction is trivial.

Because of Lemma 9 we do not need to check for all strict subsets of P whether they are pseudo-intents. To test if P is a minimal pseudo-intent it suffices to perform $n + 1$ checks, namely checking whether each of the n sets $P \setminus \{m\}$, $m \in P$, is a concept intent and whether P itself is not a concept intent. Since checking whether a given set is a concept intent can be done in polynomial time it can be checked in polynomial time whether P is a minimal pseudo-intent. By comparison the best known algorithm to check whether a set P is a pseudo-intent runs in coNP [8, 7].

4.2 Finding Minimal Pseudo-Intents

Not only do the two algorithms Next Closure and Incremental Construction have an exponential delay in between the computation of one pseudo-intent and the next. One may even have to wait for some time exponential in the size of the context until even the first pseudo-intent is computed. This raises the question whether there can be an algorithm that finds at least one pseudo-intent in polynomial time. To the best knowledge of the author no such algorithm has yet been published. Lemma 9 gives us an idea for a minimal algorithm (Algorithm 1) that finds one minimal pseudo-intent in polynomial time.

The idea is the following. We start with the full attribute set M and check whether all its strict subsets are concept intents using Lemma 9. If they are all concept intents then the context has no pseudo-intents. If one of them is not a concept intent then it either contains a pseudo-intent or is a pseudo-intent itself. Then we continue by checking whether that subset has a subset that is not a concept intent and so on.

Lemma 10 (Soundness of Algorithm 1). *Let $\mathbb{K}$ be a context. If $\mathbb{K}$ has a pseudo-intent then Algorithm 1 returns a minimal pseudo-intent S upon termination.*

Algorithm 1 Algorithm for finding one minimal pseudo-intent

```
Input: 𝕂 = (G, M, I)
S := M
repeat
    finished := true
    for all m ∈ S do
        if S \ {m} is not a concept intent then
            S := S \ {m}
            finished := false
            exit for-loop
        end if
    end for
until finished
if S = M then
    print 𝕂 has no pseudo-intent
else
    return S
end if
```

Proof. Algorithm 1 remains in the **repeat**-loop until the variable `finished` is true. This means that upon termination for all $m \in S$ the set $S \setminus \{m\}$ is a concept intent. Otherwise `finished` would have been set to **false** in one of the iterations of the inner **for**-loop. It follows from Lemma 9 that all strict subsets of S are concept intents. If $S \neq M$ then S is itself not a concept intent (this has been checked in the previous iteration of the **repeat**-loop). Then S is a minimal pseudo-intent.

On the other hand if Algorithm 1 terminates with $S = M$ then both M and all of its subsets are concept intents. Thus $\mathbb{K}$ does not have any pseudo-intents.

Lemma 11 (Termination of Algorithm 1). *Algorithm 1 terminates after at most $|M|$ iterations of the repeat-loop. The total runtime is bounded by $\mathcal{O}(|G| \cdot |M|^3)$.*

Proof. The algorithm starts with $S = M$. In each iteration of the repeat-loop one element is removed from S. The algorithm terminates if S is the empty set. Therefore it must terminate after at most $|M|$ iterations.

In each iteration of the repeat-loop the for-loop is entered at most $|S| < |M|$ times. Inside the for-loop the algorithm checks whether $S \setminus \{m\}$ is a concept intent. This check can be done in time of order $\mathcal{O}(|G||M|)$. Thus, the total runtime is bounded by $\mathcal{O}(|G||M| \cdot |M| \cdot |M|)$.

This shows that not only is it possible to check in polynomial time whether a given set of attributes is a minimal pseudo-intent, it is also possible to find an arbitrary minimal pseudo-intent in polynomial time. This raises hopes that it might be possible to compute at least the minimal pseudo-intents in polynomial time. Unfortunately, this is not the case, as we will see by examining the following problem.

*Problem 3 (All minimal pseudo-intents (*ALLMPI*)). Input:* A formal context $\mathbb{K} = (G, M, I)$ and pseudo-intents $P_1, \ldots, P_n$.
Question: Are $P_1, \ldots, P_n$ all minimal pseudo-intents of $\mathbb{K}$?

Lemma 12 (Containment in coNP). ALLMPI *is in* coNP.

Proof. We already know that checking whether a set $Q \subseteq M$ is a minimal pseudo-intent can be done in polynomial time (Lemma 9). So to decide whether $P_1, \ldots, P_n$ are not all the minimal pseudo-intents one can non-deterministically guess a set $Q \subseteq M$ such that $Q \notin \{P_1, \ldots, P_n\}$ and then check in polynomial time whether it is a minimal pseudo-intent. Thus the dual problem of ALLMPI can be decided in non-deterministic polynomial time. Therefore ALLMPI is in coNP.

Lemma 13 (Hardness of AllMPI). ALLMPI *is* coNP*-hard.*

Proof. We use the same reduction as for Theorem 1. Given an instance of TAUTOLOGY, i. e. a propositional formula f in disjunctive normal form, let $\mathbb{K}$ be the context from Table 2, constructed as in Section 3. We show that $P_1 = \{t_1, f_1\}$, $\ldots$, $P_m = \{t_m, f_m\}$, $P_{m+1} = \{\alpha_1\}$, $P_{2m} = \{\alpha_m\}$ are all the minimal pseudo-intents of $\mathbb{K}$ iff f is a tautology.

It has already been shown in the proof of Theorem 1 that $P_1,\ldots,P_m$ are minimal pseudo-intents. The empty set $\emptyset$ is a concept intent in $\mathbb{K}$. In $\mathbb{K}$ all objects intents g' for some $g \in G$ are such that $\alpha_i \in g'$ if and only if $\{t_i, f_i\} \subseteq g'$. Therefore, $\{t_i, f_i\}$ is contained in $\{\alpha_i\}'' = P''_{m+i}$. Thus $P_{m+1},\ldots, P_{2m}$ are also minimal pseudo-intents. We can use the first three steps of the proof of Theorem 1.

We claim that $P_1,\ldots,P_{2m}$ are all minimal pseudo-intents of $\mathbb{K}$ iff for all assignments ϕ the set S_ϕ is a concept intent in $\mathbb{K}$. *only-if*: Assume that some S_ϕ is not a concept intent. Then S_ϕ must contain some minimal pseudo-intent $P \subseteq S_\phi$. The definition of S_ϕ (1) shows that S_ϕ does not contain α_i, and it contains either t_i or f_i but not both, for all $i \in \{1, \ldots, n\}$. Thus S_ϕ does not contain any of the $P_1,\ldots, P_{2m}$, and therefore it must be a new minimal pseudo-intent.

if: Let $Q \subseteq M$ be a set of attributes. If Q contains some α_i then Q cannot be a minimal pseudo-intent. Therefore Q is a subset of $\{t_1, f_1, \ldots, t_m, f_m\}$. In Lemma 7 it is shown that Q cannot be a pseudo-intent if the hypothesis holds. Thus there cannot be another minimal pseudo-intent.

Together with Lemma 6 this proves that $P_1, \ldots, P_{2m}$ are all minimal pseudo-intents of $\mathbb{K}$ iff f is a tautology. Thus ALLMPI is coNP-hard.

Corollary 2. ALLMPI *is* coNP*-complete.*

Corollary 3. *Given a context* $\mathbb{K}$ *the set of all minimal pseudo-intents of* $\mathbb{K}$ *cannot be computed in output-polynomial time unless* $\mathrm{P} = \mathrm{NP}$.

Proof. Assume that there was an algorithm $\mathcal{A}$ that takes $\mathbb{K}$ as its input and enumerates the set $\mathcal{P}$ of all minimal pseudo-intents in output-polynomial time.

Let n be the number of pseudo-intents. This means that there is a polynomial $p(|G|, |M|, |\mathcal{P}|)$ such that for all contexts $\mathbb{K} = (G, M, I)$ the runtime of $\mathcal{A}$ is bounded by $p(|G|, |M|, |\mathcal{P}|)$.

Then we can construct an algorithm $\mathcal{A}'$ that decides AllMPI as follows. Given a context $\mathbb{K}$ and a set of minimal pseudo-intents $\{P_1, \ldots, P_n\}$ $\mathcal{A}'$ runs $\mathcal{A}$ on $\mathbb{K}$ for at most $p(|G|, |M|, n)$ steps. If $\mathcal{A}$ does not terminate then there must be more than n minimal pseudo-intents, so $\mathcal{A}'$ return "No". If $\mathcal{A}$ terminates then $\mathcal{A}'$ compares the output of $\mathcal{A}$ to $\{P_1, \ldots, P_n\}$. If they are identical then $\mathcal{A}'$ return "Yes", otherwise "No". The runtime of $\mathcal{A}'$ is bounded by a polynomial in $|G|$, $|M|$ and $|\mathcal{P}|$.

Note that this does not yield a complexity result for the problem of computing all pseudo-intents. That is unless it can be shown that the total number of pseudo-intents of a context is bounded by a polynomial in the number of its minimal pseudo-intents. We conjecture that this is not the case.

5 Conclusion

In this work we have proved that the problem FirstPI of determining whether a given set of pseudo-intents is the set of lectically first pseudo-intents of a given context is coNP-complete. This helped us to prove that enumerating pseudo-intents in the lectic order is not tractable unless P = NP. From the results of previous work it only followed that enumerating pseudo-intents (in any order) is not tractable unless TransHyp is in P.

In the second section of the paper we have introduced minimal pseudo-intents. They play a special rôle because they occur in any implication base of a context, not only in the Duquenne-Guigues base. In many ways they are easier to handle than general pseudo-intents. For example we have shown that given a set of concept intents it is tractable to check whether it is a minimal pseudo-intent. Furthermore, one can find one minimal pseudo-intent in polynomial time. However, we have shown that the set of minimal pseudo-intents of a context cannot be computed in output polynomial time.

Future work We conjecture that the lectic order is a source of complexity in the enumeration process. We therefore suggest that in order to develop efficient algorithms for computing pseudo-intents the FCA community should try to find alternatives to the lectic order. An idea might be incremental algorithms in the style of Obiedkov et al. [10]. Perhaps, it is also possible to compute all pseudo-intents by starting with the full set of attributes and then deleting attributes similar to Algorithm 1.

References

1. Franz Baader, Bernhard Ganter, Ulrike Sattler, and Barış Sertkaya. Completing description logic knowledge bases using formal concept analysis. In *Proc. of the*

20th Int. Joint Conf. on Artificial Intelligence (IJCAI 2007). AAAI Press/The MIT Press, 2007.
2. T. Eiter and G. Gottlob. Hypergraph transversal computation and related problems. In S. Flesca, S. Greco, N. Leone, and G. Ianni, editors, *Proceedings of JELIA 2002*, volume 2424 of *LNCS*, pages 549–564, Heidelberg, 2002. Springer.
3. Michael L. Fredman and Leonid Khachiyan. On the complexity of dualization of monotone disjunctive normal forms. *Journal of Algorithms*, 21(3):618 – 628, 1996.
4. Bernhard Ganter. Two basic algorithms in concept analysis. Preprint 831, Fachbereich Mathematik, TU Darmstadt, Darmstadt, Germany, 1984.
5. Bernhard Ganter and Rudolf Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer, New York, 1997.
6. J.-L. Guigues and V. Duquenne. Familles minimales d'implications informatives résultant d'un tableau de données binaires. *Math. Sci. Humaines*, 95:5–18, 1986.
7. S. O. Kuznetsov. On the intractability of computing the Duquenne-Guigues base. *Journal of Universal Computer Science*, 10(8):927–933, 2004.
8. Sergei O. Kuznetsov. Counting pseudo-intents and $\sharp$P-completeness. In *Formal Concept Analysis*, pages 306–308, 2006.
9. Sergei O. Kuznetsov and Sergei A. Obiedkov. Algorithms for the construction of concept lattices and their diagram graphs. In *Principles of Data Mining and Knowledge Discovery*, pages 289–300, 2001.
10. S. Obiedkov and V. Duquenne. Attribute-incremental construction of the canonical implication basis. *Annals of Mathematics and Artificial Intelligence*, 49(1-4):77–99, 2007.
11. Barış Sertkaya. Some computational problems related to pseudo-intents. In Sébastien Ferré and Sebastian Rudolph, editors, *Proceedings of the 7th International Conference on Formal Concept Analysis, (ICFCA 2009)*, volume 5548 of *Lecture Notes in Artificial Intelligence*, pages 130–145. Springer Verlag, 2009.
12. Barış Sertkaya. Towards the complexity of recognizing pseudo-intents. In Frithjof Dau and Sebastian Rudolph, editors, *Proceedings of the 17th International Conference on Conceptual Structures, (ICCS 2009)*, volume 5662, pages 284–292, 2009.