Some Notes on Pseudo-closed Sets[*]

Sebastian Rudolph
Institute AIFB
University of Karlsruhe (TH)
Germany
rudolph@aifb.uni-karlsruhe.de

Abstract

Pseudo-intents (also called pseudo-closed sets) of formal contexts have gained interest in recent years, since this notion is helpful for finding minimal representations of implicational theories. In particular, there are some open problems regarding complexity. In our paper, we compile some results about pseudo-intents which contribute to the understanding of this notion and help in designing optimized algorithms. We provide a characterization of pseudo-intents based on the notion of a formal context's incrementors. The latter are essentially non-closed sets which – when added to a closure system – do not enforce the presence of other new attribute sets. In particular, the provided definition is non recursive. Moreover we show that this notion coincides with the notion of a quasi-closed set that is not closed, which enables to reuse existing results and to formulate an algorithm that checks for pseudo-closedness. Later on, we provide an approach for further optimizing those algorithms based on a result which correlates the set of pseudo-intents of a formal context with the pseudo-intents of this context's reduced version.

1 Introduction

Pseudo-intents are of significant interest in formal concept analysis. One central result ([5]) states, that the implication set $\{P \to P^{II} \mid P \text{ pseudo-intent of } \mathbb{K}\}$ (called *stem base*) constitutes a so-called *implicational base*, i.e., a minimal set of implications generating the implicational theory of the formal context $\mathbb{K}$. In this regard it is also important to note that for an arbitrary implication, checking whether it is semantically entailed by a set of implications can be decided in linear time ([2, 8]). Thus, pseudo-intents become relevant for problems related to small (yet quick to query) representation of implicative knowledge.
The complexity of determining for a given context $\mathbb{K} = (G, M, I)$ and attribute set $A \subseteq M$, whether A is a pseudo-intent (or: pseudo-closed) with respect to $\mathbb{K}$ is still an open problem (see [9]). The prevailing assumption seems to be that the problem's

[*] Supported by the Deutsche Forschungsgemeinschaft (DFG) under the ReaSem project.

complexity is rather high (at least beyond polynomial time). Partial results ([7, 6]) show that it is in coNP.

In our paper, we compile some results about pseudo-intents and provide optimized algorithms for checking for pseudo-closedness.

In detail, we will proceed as follows: In Section 2, we recall the fundamental definitions and propositions of FCA needed to specify and deal with the topic. Section 3 provides and verifies an algorithm which allows to convert an arbitrary set of implications into a stem base. Section 4 introduces the notion of *incrementor* and shows how it can be used to provide a non-recursive characterization of pseudo-intents. In the end, this notion shows to have a direct correspondence to that of a *quasi intent* introduced in [3]. Resulting from these preceding considerations, Section 5 presents an algorithm which checks for pseudo-closedness. Section 6 shows how pseudo-closedness can be checked even by examining the reduced version of the considered context and provides a corresponding algorithm. Finally, Section 7 concludes and outlines possible directions for further research.

2 Preliminaries

In this section, we will introduce the notions from formal concept analysis necessary for our work.

First of all, note that we use the notation "$\subset$" to indicate the *strict* subset, i.e. $A \subset B$ means $A \subseteq B$ and $A \neq B$.

Deviating from the usual line of presentation, we will introduce implications and pseudo-closed sets just on the basis of closure operators. This allows to talk about those notions independently from concrete formal concepts and facilitates the presentation of some results in the sequel. However, note that this is not a proper generalization, since every closure operator can be represented by the $(.)^{II}$-operator of an appropriately chosen formal context (e.g. the context $(\{A \mid \varphi(A) = A\}, M, \ni)$). Thus, the cited definitions and results – although defined on basis of a formal context – carry over to our way of introducing those notions.

The following considerations are based on an arbitrary set M. We will first define the fundamental notion of a closure operator M. Roughly spoken, applying such an operator to a set can be understood as a minimal extension of that set in order to fulfill certain properties.

Definition 1 *Let M be an arbitrary set. A function $\varphi : \mathcal{P}(M) \to \mathcal{P}(M)$ (where $\mathcal{P}(M)$ denotes the powerset of M) will be called*

- EXTENSIVE, *if $A \subseteq \varphi(A)$ for all $A \subseteq M$,*
- MONOTONE, *if from $A \subseteq B$ follows $\varphi(A) \subseteq \varphi(B)$ for all $A, B \subseteq M$, and*
- IDEMPOTENT, *if $\varphi(\varphi(A)) = \varphi(A)$ for all $A \subseteq M$.*

If φ is extensive, monotone, and idempotent, we will call it a CLOSURE OPERATOR. *In this case, we will additionally call*

- $\varphi(A)$ *the* CLOSURE *of* A,
- A CLOSED *(or* φ-CLOSED*), if* $A = \varphi(A)$.

The family of all closed sets is also called CLOSURE SYSTEM. Furthermore, any closure system constitutes a *lattice* with set inclusion as the respective order relation.
In the sequel, we show, in which way closure operators are closely related to implications.

Definition 2 *Let M be an arbitrary set. An* IMPLICATION *on M is a pair (A, B) with $A, B \subseteq M$. To support intuition, we write $A \to B$ instead of (A, B).*[1]
A set $C \subseteq M$ RESPECTS *an implication $A \to B$ if*

$$A \subseteq C \quad implies \quad B \subseteq C.$$

Furthermore, for $C \subseteq M$ and a set $\mathfrak{I}$ of implications on M, let $C^{\mathfrak{I}}$ denote the smallest set with

- $C \subseteq C^{\mathfrak{I}}$ *and*
- $C^{\mathfrak{I}}$ *respects* $\mathfrak{i}$ *for every implication* $\mathfrak{i} \in \mathfrak{I}$.[2]

It is well known, that the operation $(.)^{\mathfrak{I}}$ is a closure operator on M. So, according to Definition 1, if $C = C^{\mathfrak{I}}$, we call C ($\mathfrak{I}$-)CLOSED.

Definition 3 *We say $\mathfrak{I}$* ENTAILS *$A \to B$ (written: $\mathfrak{I} \models A \to B$), if every $C \subseteq M$ that respects all implications of $\mathfrak{I}$ also respects $A \to B$.*
An implication set $\mathfrak{I}$ will be called NON-REDUNDANT, *if for any $\mathfrak{i} \in \mathfrak{I}$, we have that $\mathfrak{I} \setminus \{\mathfrak{i}\}$ does not entail $\mathfrak{i}$.*
An implication set $\mathfrak{I}$ will be called an IMPLICATION BASE *for a closure operator φ if*

- *it is* NON-REDUNDANT, *i.e. for any $\mathfrak{i} \in \mathfrak{I}$, we have that $\mathfrak{I} \setminus \{\mathfrak{i}\}$ does not entail $\mathfrak{i}$,*
- *it is* SOUND, *i.e., any implication on M entailed by $\mathfrak{I}$ is respected by all φ-closed sets, and*
- *it is* COMPLETE, *i.e., any implication on M respected by all φ-closed sets is entailed by $\mathfrak{I}$.*

Well-known facts concerning the entailment of implications are

- $\mathfrak{I} \models A \to B$ exactly if $B \subseteq A^{\mathfrak{I}}$ and
- $\mathfrak{I}$ is non-redundant iff $B \not\subseteq A^{\mathfrak{I} \setminus \{A \to B\}}$ for all $A \to B \in \mathfrak{I}$.

[1] To facilitate reading we will occasionally omit the parentheses, i.e., we will write $a, b \to c$ instead of $\{a, b\} \to \{c\}$.

[2] Note, that this is well-defined, since the mentioned properties are closed wrt. intersection.

Below, we will now define the central notion of this paper. Opposed to the usual way of presentation, we will define the notion of pseudo-closedness independently from a particular formal context, just referring to a given closure operator. Besides the more general definition this will facilitate our considerations in section 3.[3]

Definition 4 *For a given closure operator φ, a set $P \subseteq M$ will be called* PSEUDO-CLOSED *if $\varphi(P) \neq P$ and $\varphi(Q) \subseteq P$ holds for every pseudo-closed $Q \subset P$.*

Note that this definition is recursive. Since the set M is always assumed to be finite in the sequel, it is nevertheless correct. However, directly using this definition to check whether an attribute set is a pseudo-intent requires a recursion as well and is therefore computationally costly. This led to the complexity questions mentioned in the introduction.

Regarding pseudo-closed sets, we give corollaries of the Propositions 24 and 25 from [4].

Proposition 1 *If P and Q are closed or pseudo-closed sets with $P \not\subseteq Q$ and $Q \not\subseteq P$, then $P \cap Q$ is a closed set.*

The first Proposition directly yields the fact that the set of all closed and pseudo-closed sets (of a closure operator φ) constitute a closure system themselves (for another closure operator ψ).

Proposition 2 *Every (wrt. a closure operator φ) sound and complete set of implications contains an implication $A \to B$ with $A \subseteq P$ and $\varphi(A) = \varphi(P)$ for every pseudo-closed set P.*

Moreover, for every closure operator, the family of its pseudo-closed sets can be used to define a canonical implication base called *stem base* ([5]):

Theorem 1 *Let φ be a closure operator. Then the set*

$$\mathfrak{SB} := \{P \to \varphi(P) \mid P \textit{ pseudo-closed for } \varphi\}$$

is an implication base of φ.

In the remainder of this section, we will very briefly recall well-known basic facts from FCA for later reference.

Proposition 3 *Properties of the derivation operator $(.)^I$.*

- *$(.)^{II}$ is a closure operator on G as well as on M, i.e., it is extensive (**ext**^{II}), monotone (**mon**^{II}) and idempotent (**idp**^{II}).*
- *for all $A \neq \emptyset$, $A^I = \bigcap_{a \in A} a^I$.* **decomp**

[3] Trivially, this coincides with the notion of pseudo-intent of a formal context if we set $\varphi = (.)^{II}$.

We use $\mathcal{I}(\mathbb{K})$ to denote the family of all concept intents of $\mathbb{K}$.
The concept intents of a formal concept are exactly those attribute sets closed wrt. $(.)^{II}$, i.e., $\mathcal{I}(\mathbb{K}) = \{A \mid A = A^{II} \subseteq M\}$. In other words, the set $\mathcal{I}(\mathbb{K})$ coincides with the closure system generated by $(.)^{II}$ on M. Consequently, the family $\mathcal{I}(\mathbb{K})$ of all concept intents of a formal context is closed wrt. intersection (**clos**$\cap$).
We proceed by giving a Proposition which is the dual of Proposition 30 from [4].

Proposition 4 *If $G \subseteq H$ then every intent of $(G, M, I \cap (G \times M))$ is an intent of (H, M, I).*

In words, the preceding proposition just states that adding an object with arbitrary intent to a context preserves all previous intents.

3 Generating Stem Bases from Implication Sets

In this section, we present an algorithm which is a slight modification of the one presented in [1] and provide a self-contained proof for its correctness.
Given an arbitrary finite set $\mathfrak{I} = \{\mathfrak{i}_1, \ldots, \mathfrak{i}_n\}$ of implications on an attribute set M, the algorithm from Fig. 1 will convert this set into a stem base $\mathfrak{S}\mathfrak{B}$ with $\mathfrak{S}\mathfrak{B} \models A \to B$ exactly if $\mathfrak{I} \models A \to B$.

```
function:  stembase($\mathfrak{I}$)
  1. Set $\mathfrak{S}\mathfrak{B} := \emptyset$.
  2. For every $A \to B \in \mathfrak{I}$
     substitute $A \to B$ by $A \to (A \cup B)^{\mathfrak{I}}$.
  3. As long as $\mathfrak{I} \neq \emptyset$,
     (a) select an $A \to B$ from $\mathfrak{I}$,
     (b) delete $A \to B$ from $\mathfrak{I}$,
     (c) calculate $A^{\mathfrak{I} \cup \mathfrak{S}\mathfrak{B}}$,
     (d) if $A^{\mathfrak{I} \cup \mathfrak{S}\mathfrak{B}} \neq B$ then
         add $A^{\mathfrak{I} \cup \mathfrak{S}\mathfrak{B}} \to B$ to $\mathfrak{S}\mathfrak{B}$.
  4. Output $\mathfrak{S}\mathfrak{B}$ and terminate.
```

Figure 1: Algorithm `stembase(`$\mathfrak{I}$`)` for calculating the stem base of the implicational theory generated by $\mathfrak{I}$.

Theorem 2 *The algorithm `stembase` computes a stembase for the closure operator $(.)^{\mathfrak{I}}$*

Proof: We have to show two properties: For any set $\mathfrak{I}$ of implications on M, we have

- $(.)^{\mathfrak{I}} = (.)^{\texttt{stembase}(\mathfrak{I})}$ and
- $\texttt{stembase}(\mathfrak{I})$ is a stembase.

The first property will be proved by iteratively showing that every single action carried out by the algorithm does not change the closure operator $(.)^{\mathfrak{I}\cup\mathfrak{SB}}$. By "concatenating" those arguments and with the observation that $\mathfrak{SB} = \emptyset$ in the beginning and $\mathfrak{I} = \emptyset$ in the end, we can conclude that this first property indeed holds.
So, first, we consider the actions carried out in line 2. Let $\mathfrak{H} = \mathfrak{I} \setminus \{A \to B\} \cup \{A \to (A \cup B)^{\mathfrak{I}}\}$ for an arbitrary $A \to B \in \mathfrak{I}$. Now consider an arbitrary $C \subseteq M$. We have to show, that C respects all implications from $\mathfrak{I}$ exactly if it respects all implications from $\mathfrak{H}$.
"$\Leftarrow$": This is trivial, since $B \subseteq (A \cup B)^{\mathfrak{I}}$.
"$\Rightarrow$": Assume C respects all implications of $\mathfrak{I}$. Now, the only way for C to not respect all implications of $\mathfrak{H}$ would obviously be $A \subseteq C$ and $B^{\mathfrak{I}} \not\subseteq C$. On the other hand, since C respects $A \to B$, we know that $B \subseteq C$. Furthermore, $B^{\mathfrak{I}}$ is by definition the smallest set (wrt. set inclusion) containing B and respecting all implications of $\mathfrak{I}$. Hence, we have $B^{\mathfrak{I}} \subseteq C$, leading to a contradiction.
Now, we consider the actions of point 3. Let $\mathfrak{I}$ and $\mathfrak{SB}$ be the sets before carrying out an a-b-c-d block and $\mathfrak{I}^*$ and $\mathfrak{SB}^*$ the respective values afterwards. Again, considering an arbitrary $C \subseteq M$, we have to show, that C respects all implications from $\mathfrak{I} \cup \mathfrak{SB}$ exactly if it respects all implications from $\mathfrak{I}^* \cup \mathfrak{SB}^*$.
"$\Rightarrow$": This is obvious, since clearly for every implication $A \to B$ from $\mathfrak{I}^* \cup \mathfrak{SB}^*$ we have an implication $D \to B$ from $\mathfrak{I} \cup \mathfrak{SB}$ with $D \subseteq A$.
"$\Leftarrow$": Suppose C respects all implications from $\mathfrak{I}^* \cup \mathfrak{SB}^*$. Assuming that is does not respect all implications of $\mathfrak{I} \cup \mathfrak{SB}$ would imply $A \subseteq C$ and $B \not\subseteq C$. Yet, knowing that C respects $A^{\mathfrak{I}\cup\mathfrak{SB}} \to B$ (being also trivially true for $A^{\mathfrak{I}\cup\mathfrak{SB}} = B$), we have to conclude that $A^{\mathfrak{I}\cup\mathfrak{SB}} \not\subseteq C$. But, again by definition, $A^{\mathfrak{I}\cup\mathfrak{SB}}$ is the smallest set containing A and respecting all implications from $\mathfrak{I} \cup \mathfrak{SB}$, enforcing $A^{\mathfrak{I}\cup\mathfrak{SB}} \subseteq C$ and therefore yielding a contradiction.

Let $\mathfrak{SB} = \texttt{stembase}(\mathfrak{I})$. We prove the second property by showing that for all $A \to B \in \mathfrak{SB}$, the set A is pseudo-closed wrt. $(.)^{\mathfrak{SB}}$. Note that from the construction of the algorithm and the previous proof (including the fact that $(.)^{\mathfrak{I}\cup\mathfrak{SB}}$ remains constant) follows that, for all $A \to B \in \mathfrak{SB}$,

$$A = A^{\mathfrak{SB}\setminus\{A \to B\}}. \quad (*)$$

Now we assume A were not pseudo-closed for an $A \to B \in \mathfrak{SB}$. Obviously, it is not closed either. So there must exist a pseudo-closed set $P \subset A$ with $P^{\mathfrak{SB}} \not\subseteq A$. Now, consider $Q := P^{\mathfrak{SB}\setminus\{A \to B\}}$. By monotonicity, we then have $Q \subseteq A$. So the only possibility to make $P^{\mathfrak{SB}} \not\subseteq A$ true is that Q does not respect $A \to B$. Yet, this would imply $Q = A$ and consequently $P^{\mathfrak{SB}} = B$. Now due to Proposition 2, we know, that $\mathfrak{SB}$ has to contain an implication $C \to D$ with $C \subseteq P$ and $C^{\mathfrak{SB}} = P^{\mathfrak{SB}} = B$. Moreover, due to the construction we know that $D = C^{\mathfrak{SB}} = B$. Since $(A \to B) \neq (C \to B)$, we have that $C \to B \in \mathfrak{SB} \setminus \{A \to B\}$. Yet, from this and $C \subseteq A$ follows $B \subseteq A^{\mathfrak{SB}\setminus\{A \to B\}}$, contradicting the equation (*). □

Calculating the $\mathfrak{I}$-closure (without preprocessing) can be done in time $O(|\mathfrak{I}|)$ due to [8]. Hence, the presented algorithm runs in $O(|\mathfrak{I}|^2)$ i.e. quadratic time (this complexity bound for the task accomplished by the algorithm had already been shown in [10]). Mark that this algorithm naturally also determines all pseudo-intents (being just the premises of the implications of $\mathfrak{SB}$).

4 Characterizing Pseudo-intents

Now, we will introduce notions that are essential for our aim to characterize pseudo-intents non-recursively.

Definition 5 *Let $\mathbb{K} = (G, M, I)$ be a formal context and let $P \subseteq M$. We define $\mathbb{K}[P] := (H, M, I_P)$ (say: $\mathbb{K}$ AUGMENTED BY P) as follows:*

- *$H := G \cup \{g_P\}$ (where we presume $g_P \notin G$) and*
- *$I_P := I \cup (\{g_P\} \times P)$*

The following results are immediate consequences of this definition:

Lemma 1 *Let $\mathbb{K} = (G, M, I)$ be a formal context and let $P \subseteq M$. Then*

- *for all $g \in G$, we have $g^{I_P} = g^I$,* **cons1**
- *for all $A \subseteq G$, we have $A^{I_P} = A^I$,* **cons2**
- *for all $A \subseteq M$, we have $A^I = A^{I_P} \setminus \{g_P\}$,* **cons3**
- *for all $A \subseteq M$ with $g_P \notin A^{I_P}$, we have $A^{I_P I_P} = A^{II}$, and* **cons4**
- *for all $A \subseteq M$ with $g_P \in A^{I_P}$, we have $A^{I_P I_P} = P \cap A^{II}$.* **cons5**

Proof:

- **cons1** This is trivial, since $\{m \mid gI_A m\} = \{m \mid gIm\}$.
- **cons2** Due to **decomp**, we have $A^{I_A} = \bigcap_{g \in A} g^{I_A}$. Due to **cons1**, this equals $\bigcap_{g \in A} g^I = A^I$.
- **cons3** Consider an arbitrary $g \in G = H \setminus \{g_P\}$. The statement $g \in A^{I_P}$ is equivalent to $gI_P m$ for all $m \in A$. Since - due to the definition - I_P and I coincide on all objects but g_P, this is equivalent to gIm for all $m \in A$, which in turn is the same as $g \in A^I$.
- **cons4** From **cons3**, we conclude $A^{I_P I_P} = (A^{I_P} \setminus \{g_P\})^{I_P} = A^{II_P}$ and by **cons2** follows $A^{II_P} = A^{II}$.
- **cons5** From $A^{I_P} = \{g_P\} \cup (A^{I_P} \setminus \{g_P\})$, we can conclude $A^{I_P I_P} = \bigcap_{g \in A^{I_P}} g^{I_P} = g_P^{I_P} \cap \bigcap_{g \in A^{I_P} \setminus \{g_P\}} g^{I_P} = P \cap (A^{I_P} \setminus \{g_P\})^{I_P}$. Due to **cons3**, this equals $P \cap A^{II_P}$ and due to **cons2** this is just $P \cap A^{II}$.

□

Proposition 5 *Properties of augmentations.*

$A \in \mathcal{I}(\mathbb{K}[A])$*, i.e.,* A *is an intent of* $\mathbb{K}[A]$*,* **cont[]**

$\mathcal{I}(\mathbb{K}) \subseteq \mathcal{I}(\mathbb{K}[A])$*, i.e., every formal concept intent of* $\mathbb{K}$ *is also a concept intent of* $\mathbb{K}[A]$*,* **mon[]**

If $A \in \mathcal{I}(\mathbb{K})$ *then* $\mathcal{I}(\mathbb{K}[A]) = \mathcal{I}(\mathbb{K})$*, i.e., if an object is added to the context, the intent of which is already an intent of* $\mathbb{K}$*, the overall set of intents remains unchanged.* **intid[]**

Proof:

cont[]: Obviously, $({g_A}^{I_A I_A}, A)$ is a formal concept of $\mathbb{K}[A]$.

mon[]: This property follows directly from Proposition 4.

intid[]: Assume the contrary, i.e., there were a $B \in \mathcal{I}(\mathbb{K}[A]) \setminus \mathcal{I}(\mathbb{K})$. Due to **decomp**, we know $B = B^{I_A I_A}$. Obviously, g_A has to be in B^{I_A}, since otherwise $B^{I_A I_A} = B^{II}$ by **cons4**, contradicting $B \notin \mathcal{I}(\mathbb{K})$. Thus, due to **cons5**, $B = B^{I_A I_A} = A \cap B^{II}$. Yet, knowing that A is an intent of K and due to **clos**$\cap$, the intersection of the two closed sets A and B^{II} has again to be closed, we have found a contradiction to the assumption.

□

To facilitate the intuition about context augmentations, consider Fig. 2, which shows some augmentations of a small context and the impact of this on the set of concept intents.

For our further line of argumentation, the motivating intuitive idea is (also conveyed by the name) that a pseudo-intent is "almost an intent". Since we know that augmenting a context by an intent does not change the corresponding intent set, we could expect that adding a pseudo-intent would result in just a very slight change. Considering the slightest change possible we define the notion of an *incrementor*.

Definition 6 *We say that* P *is an* INCREMENTOR *of* $\mathbb{K}$*, if*

– P *is not a concept intent of* $\mathbb{K}$ *and*

– *for every concept intent* $A \subseteq M$ *of* $\mathbb{K}[P]$ *we have* $B = P$ *or* B *is a concept intent of* $\mathbb{K}$*.*

Looking back at Fig. 2, we see, that in this case, the empty set would be an incrementor of $\mathbb{K}$. Moreover, it takes little consideration to verify that it is also a pseudointent of $\mathbb{K}$. The following theorem partly justifies our intuition by ensuring that every pseudointent is indeed an incrementor.

Theorem 3 *Let* $\mathbb{K}$ *be a formal context and* P *be a pseudo-intent of* $\mathbb{K}$*. Then* P *is an incrementor of* $\mathbb{K}$*.*

$\mathbb{K}$

$\mathbb{K}$	m_1	m_2	m_3
g_1	×	×	
g_2		×	×

$\mathcal{I}(\mathbb{K})$

$\{m_2\}$,
$\{m_1, m_2\}$,
$\{m_2, m_3\}$,
$\{m_1, m_2, m_3\}$,

$\mathbb{K}[\{m_1, m_3\}]$

	m_1	m_2	m_3
g_1	×	×	
g_2		×	×
$g_{\{m_1,m_3\}}$	×		×

$\mathcal{I}(\mathbb{K}[\{m_1, m_3\}])$

$\underline{\emptyset}$,$\underline{\{m_1\}}$,$\{m_2\}$, $\underline{\{m_3\}}$,
$\{m_1, m_2\}$,$\underline{\{m_1, m_3\}}$,$\{m_2, m_3\}$,
$\{m_1, m_2, m_3\}$

$\mathbb{K}[\{m_2\}]$

	m_1	m_2	m_3
g_1	×	×	
g_2		×	×
$g_{\{m_2\}}$		×	

$\mathcal{I}(\mathbb{K}[\{m_2\}])$

$\{m_2\}$,
$\{m_1, m_2\}$,
$\{m_2, m_3\}$,
$\{m_1, m_2, m_3\}$

$\mathbb{K}[\emptyset]$

	m_1	m_2	m_3
g_1	×	×	
g_2		×	×
$g_\emptyset$			

$\mathcal{I}(\mathbb{K}[\emptyset])$

$\underline{\emptyset}$, $\{m_2\}$,
$\{m_1, m_2\}$,
$\{m_2, m_3\}$,
$\{m_1, m_2, m_3\}$

Figure 2: Examples for context augmentations and their consequences for the set of concept intents. Intents added by the augmentation are underlined.

$\mathbb{K}$

	m_1	m_2	m_3	m_4
g		×	×	×

$\mathcal{I}(\mathbb{K})$

$\{m_2, m_3, m_4\}$,
$\{m_1, m_2, m_3, m_4\}$,

$\mathbb{K}[\{m_2, m_3\}]$

	m_1	m_2	m_3	m_4
g		×	×	×
$g_{\{m_2,m_3\}}$		×	×	

$\mathcal{I}(\mathbb{K}[\{m_2, m_3\}])$

$\underline{\{m_2, m_3\}}$,
$\{m_2, m_3, m_4\}$,
$\{m_1, m_2, m_3, m_4\}$,

$\mathbb{K}[\{m_2\}]$

	m_1	m_2	m_3	m_4
g		×	×	×
$g_{\{m_2\}}$		×		

$\mathcal{I}(\mathbb{K}[\{m_2\}])$

$\underline{\{m_2\}}$,
$\{m_2, m_3, m_4\}$,
$\{m_1, m_2, m_3, m_4\}$,

Figure 3: Counterexample for the coincidence of pseudo-intents and incrementors.

Proof: Consider the context $\mathbb{K}[P]$. Let (A, B) be a formal concept of $\mathbb{K}[P]$. We know, that $B = A^{I_P} = \bigcap\{a^{I_P} \mid a \in A\}$ (due to **decomp**).

Obviously, if $g_P \notin A$, we have that (A, B) is a concept of $\mathbb{K}$ as well, since $a^{I_P} = a^I$ for all $a \neq g_P$.

If $g_P \in A$, we have that $B = g_P^{I_P} \cap \bigcap\{a^{I_P} \mid a \in A \setminus \{g_P\}\}$ which yields $B = P \cap \bigcap\{a^I \mid a \in A \setminus \{g_P\}\}$.

Supposing pseudoclosedness of P, from Proposition 1 follows that B is an intent of $\mathbb{K}$, provided there exists some $a \in A \setminus \{g_P\}$ with $P \not\subseteq a^I$. In the other case, we would have $B = P$. Thus (P^{I_P}, P) is the only additional formal concept of $\mathbb{K}[P]$ compared to $\mathbb{K}$. This shows that P is an incrementor. □

Now it remains to investigate, whether this necessary condition for being a pseudo-intent is also sufficient. Unfortunately, this is not the case as Fig. 3 illustrates: in this example, $\{m_2, m_3\}$ is an incrementor of $\mathbb{K}$ but not a pseudo-intent since it contains the pseudo-intent $\emptyset$ but not its closure $\{m_2, m_3, m_4\}$.

Yet, examining this counter-example a bit further, we see that the set being an inrementor but not a pseudo-intent contains a set being again an incrementor (namely $\{m_2\}$) – with no intent "in between". This justifies to strengthen the condition accordingly. Yet, prior to proving that this leads to the desired characterization, we show a lemma that will facilitate the subsequent proof.

Lemma 2 *Let $\mathbb{K}$ be a formal context and A be an incrementor of $\mathbb{K}$. Then for any pseudo-intent Q of $\mathbb{K}$ with $Q \subset A$ and $Q^{II} \not\subseteq A$ we even have $A \subseteq Q^{II}$.*

Proof: Assume the contrary, i.e. $A \not\subseteq Q^{II}$. Then, considering $B := A \cap Q^{II}$, we see that $B \subset A$.

By **cont[]** and **mon[]**, respectively, we know $A, Q^{II} \in \mathcal{I}(\mathbb{K}[A])$ and hence by **clos**$\cap$ also $B \in \mathcal{I}(\mathbb{K}[A])$.

On the other hand, B can not be an intent of $\mathbb{K}$, since $Q \subset B$ (following from $Q \subset A$ and $Q \subset Q^{II}$ – the latter by **ext**^{II}) but $Q^{II} \not\subseteq B$ (this is because $Q \subset B$ implies $Q^{II} \subseteq B^{II}$ by **mon**^{II} and B being an intent of $\mathbb{K}$ would mean $B = B^{II}$)
So B must be an intent of $\mathbb{K}[A]$ that is neither A itself nor an intent of $\mathbb{K}$. Yet, this contradicts the assumption of A being an incrementor. □
Now we will provide and prove the announced non-recursive characterization for pseudo-intents.

Theorem 4 *Let $\mathbb{K} = (G, M, I)$ be a formal context and let $P \subseteq M$. P is a pseudo-intent of $\mathbb{K}$ if and only if*

P is an incrementor of $\mathbb{K}$ and **inc**

for every incrementor $Q \subset P$, there is an intent R with $Q \subset R \subset P$. **min**

Proof:
"⇒"
That every pseudo-intent is an incrementor has already been shown by Theorem 3.
We will prove the second condition **min** indirectly. Thus, we assume we have a pseudo-intent P violating **min**, i.e., there is a $Q \subset P$ being an incrementor and for all R with $Q \subset R \subset P$, the set R is not an intent. Note that, from Theorem 3, we know that P is an incrementor as well. Q cannot be an intent (as this would contradict the definition of incrementor), thus we consider the two remaining possibilities:

- Suppose Q is a pseudo-intent.

  This would (due to the definition of pseudo-intent) naturally require Q^{II} to be contained in P. Altogether this would mean: $Q \subset Q^{II} \subset P$ contradicting our assumption. Hence, R cannot be a pseudo-intent.

- Now, suppose Q is neither an intent nor a pseudo-intent.

  Then – due to the definition of pseudo-intent – there has to exist a pseudo-intent $S \subset Q$ with $S^{II} \not\subseteq Q$.

  From Lemma 2 then additionally follows $Q \subseteq S^{II}$.

  Since the definition of pseudo-intent requires P to contain S^{II}, we have the setting: $Q \subset S^{II} \subset P$. Yet, again, this obviously contradicts our assumption.

  Thus, it is impossible that Q is neither closed nor pseudo-closed wrt. $\mathbb{K}$.

Concluding, Q can be neither an intent nor a pseudo-intent nor none of both. Hence, the assumption of its existence must be false.

"⇐"
Assume the contrary, i.e., both conditions **inc** and **min** be fulfilled and yet P not be a pseudo-intent. Obviously P is not an intent either (otherwise, it would not be an incrementor by definition).
Therefore, P must be neither closed nor pseudo-closed. Then, by the definition of pseudo-closedness, there must be a pseudo-closed set $Q \subset P$ with $Q^{II} \not\subseteq P$.

```
function:  incrementor($A,\mathbb{K}$)
 -- Calculate $A^{II}$.  If $A^{II}=A$,
    output "NO" and terminate.

 -- For all $g\in G$,

      - Calculate $\tilde{A} := g^I \cap A$.
        If $\tilde{A}=A$ then continue with next $g$.
      - Calculate $\tilde{A}^{II}$.
      - If $\tilde{A}\neq\tilde{A}^{II}$ then
        output "NO" and terminate.

 -- Output "YES" and terminate.
```

Figure 4: Algorithm `incrementor(`$A,\mathbb{K}$`)` for checking whether A is an incrementor of $\mathbb{K}$.

From Lemma 2 follows that $P \subset Q^{II}$. Then, we have the setting $Q \subset P \subset Q^{II}$. Furthermore, note that $Q \subseteq P \subseteq Q^{II}$ entails $Q^{II} \subseteq P^{II} \subseteq (Q^{II})^{II}$ via **mon**^{II} which together with **idp**^{II} yields $Q^{II} = P^{II}$. But then, the very same argument yields $S^{II} = Q^{II}$ for every S with $Q \subset S \subset P$ (and therefore $S \neq S^{II}$). Clearly, this contradicts the initial assumption **min**. □

After having established those results, it takes little consideration to see (referring to [3] and [7]) that the incrementors of a formal context are just those quasi-intents which are not intents. This allows to reuse the corresponding results. In particular, the following corollary to the Proposition 2 from [7] can be used to check whether a given set is an incrementor in polynomial time.

Theorem 5 *P is an incrementor of $\mathbb{K}$ if and only if*

- *P is not an intent of $\mathbb{K}$ and*
- *for all $g \in G$, we have $P \subseteq g^I$ or $g^I \cap P$ is an intent of $\mathbb{K}$.*

5 An Algorithm for Checking Pseudo-closedness

Applying the results cited and presented in the preceding sections, we will now provide an algorithm for checking pseudo-closedness and analyze its complexity.[4] We start by giving an algorithm computing whether for a given formal context $\mathbb{K} = (G, M, I)$, a given attribute set $A \subseteq M$ is an incrementor of $\mathbb{K}$. This algorithm is shown in Fig. 4.

It is well-known, that the time complexity for computing the closure A^{II} of a given attribute set A is in $O(|G| \cdot |M|)$ while comparing two sets or computing g^I for a given

[4] We expect the reader to be familiar with the basic notions from complexity theory

```
function:  scan($A$,$\mathbb{K}$,check(.))
 -- For all $a \in A$,
    add $A \setminus \{a\}$ to (previously empty) list $\mathcal{L}$
 -- Starting from the $\mathcal{L}$'s first element
    for every $B$ from $\mathcal{L}$
       - If $B^{II} \neq A^{II}$,
         continue with next list element.
       - If check($B$),
         output "YES" and terminate.
       - Otherwise, for every $b \in B$, append $B \setminus \{b\}$ to $\mathcal{L}$
         if not already contained.
 -- if $\mathcal{L}$ processed,
    output "NO" and terminate.
```

Figure 5: Algorithm `scan` for determining whether for a given $A \subseteq M$ there is a $B \subset A$ with $B^{II} = A^{II}$ and `check(`B`)`.

object is less costly. Thus, regarding the time costs, the `incrementor` function consists essentially of the $|G|+1$-fold calculation of the closure, hence its time complexity is in $O(|G|^2 \cdot |M|)$.

Next, we provide an algorithm which for a given attribute set A, "scans" whether there exists a set $B \subset A$ with $B^{II} = A^{II}$ fulfilling an arbitrary computable criterion (denoted by the function `check`). This algorithm is shown in Fig. 5.

In general the time complexity of this algorithm is bounded by $2^{|M|}$ times the complexity of `check`.

Finally, we employ the `incrementor` and the `scan` functions to formulate the algorithm which actually checks for pseudo-closedness. This algorithm is displayed in Fig. 6.

Resulting from the earlier complexity considerations, we find that its the time com-

```
function:  pseudoIntent($A$,$\mathbb{K}$)
 -- Check whether incrementor($A$,$\mathbb{K}$).
    If *not* so, output "NO" and terminate.
 -- If scan($A$,$\mathbb{K}$,incrementor(.,$\mathbb{K}$)),
    output "NO" and terminate,
    otherwise, output "YES" and terminate.
```

Figure 6: Algorithm `pseudoIntent(`A`,`$\mathbb{K}$`)` for checking whether A is a pseudo-intent of $\mathbb{K}$.

plexity is in $O(2^{|M|}$.

6 Optimization: Operating on the Reduced Context

We will now discuss in which way this algorithm can be optimized. One of the straightforward issues to think about would be whether the problem of identifying pseudo-intents of a formal context $\mathbb{K}$ can be solved by checking for pseudo-closedness in the reduced version of $\mathbb{K}$. This should be possible, since – roughly spoken – a reduced context contains the same implicative information as the original one.

Theorem 6 *Let $\mathbb{K} = (G, M, I)$ be a formal context and $\mathbb{K}^* = (H, N, J)$ (with $H \subseteq G$ and $N \subseteq M$ as well as $J = I \cap (H \times N)$) the corresponding reduced context. Let furthermore $m^* = m^{II} \cap N$ for any $m \in M$. The fact that $\mathbb{K}^*$ is a reduced version of $\mathbb{K}$ then yields ${m^*}^{II} = m^{II}$.*
A set $P \subseteq M$ is a pseudo-intent of $\mathbb{K}$ exactly if one of the following is true:

- *there is a pseudo-intent P^* of $\mathbb{K}^*$ such that $P = P^* \cup \{m \in M \setminus N \mid m^* \subset P^*\}$,*
- *$P = \{m\} \cup \emptyset^{II}$ for an $m^* \neq \emptyset$, or*
- *$P = m^* \cup \{\overline{m} \in M \setminus N \mid \overline{m}^* \subset m^*\}$ for an $m \in M \setminus N$ if there is no pseudo-intent Q^* of $\mathbb{K}^*$ with ${Q^*}^{JJ} = m^*$.*

Proof:
First note that the set

$$\mathfrak{J} := \mathfrak{S}\mathfrak{B}^* \cup \{\{m\} \to m^* \mid m \in M \setminus N\} \cup \{m^* \to \{m\} \mid m \in M \setminus N\}$$

(where $\mathfrak{S}\mathfrak{B}^*$ is the stembase of $\mathbb{K}^*$) is sound and complete for the closure operator $(.)^{II}$. So we will just show, that applying the `stembase`-algorithm from Section 3 just yields an implication set where the premises are exactly the sets presented above. First we consider the result of point 2 of the algorithm:

- Every $P^* \to P^{*JJ} \in \mathfrak{S}\mathfrak{B}^*$ will be transformed to
$$P^* \to P^{*JJ} \cup \{m \mid m^* \subseteq P^{*JJ}\}.$$
- Every $\{m\} \to m^*$ will be transformed to
$$\{m\} \to m^* \cup \{\overline{m} \mid \overline{m}^* \subseteq m^*\}.$$
- Every $m^* \to \{m\}$ will be transformed to
$$m^* \to m^* \cup \{\overline{m} \mid \overline{m}^* \subseteq m^*\}.$$

Now consider point 3:

- Every $P^* \to P^{*II}$ will be transformed to
$$P^* \cup \{m \in M \setminus N \mid m^* \subset P^*\} \to P^{*II}.$$
- Every $\{m\} \to m^{*II}$ will be
  - deleted if $m^* = \emptyset$ or
  - otherwise, transformed to $\{m\} \cup \emptyset^{II} \to m^{*II}$.
- Every $m^* \to \{m\}^{II}$ will be
  - deleted, if there is a pseudo-intent Q^* of $\mathbb{K}^*$ with $Q^{*JJ} = m^*$ or
  - otherwise, transformed to
$$m^* \cup \{\overline{m} \in M \setminus N \mid \overline{m}^* \subset m^*\} \to \{m\}^{II}.$$

□

These observations allow an optimization of the pseudo-closedness checking algorithm from Fig. 6 in Section 5.
Considering the complexity, we can state the following. Due to [4], a formal context can be reduced in $O((|G|+|M|)\cdot|G|\cdot|M|)$ time (and is hence rather cheap). Therefore, this optimization could be potentially beneficiary in cases where the context is not already reduced, since the upper bound for the time complexity is decreased to $O(2^{|N|})$.

```
function:  pseudoIntentRed($A$,$\mathbb{K}$)
 -- Check whether incrementor($A$,$\mathbb{K}$).
    If not so, output ``NO'' and terminate.

 -- Calculate reduced context $\mathbb{K}^* = (H, N, J)$


 -- Calculate $\emptyset^{II}$ and $m^*$ for all $m \in M \setminus N$.

 -- If $A \setminus \emptyset^{II} = \{a\} \subseteq M \setminus N$ then
    output ``YES'' and terminate.

 -- Calculate $P^* := A \cap N$.

 -- Check whether $A = P^* \cup \{m \in M \setminus N \mid m^* \subset P^*\}$.
    If so, check whether pseudoIntent($P^*$,$\mathbb{K}^*$).
    If so, output ``YES'' and terminate.

 -- Check whether $A = m^* \cup \{\overline{m} \in M \setminus N \mid \overline{m}* \subset m*\}$
    for an $m \in M \setminus N$.
    If so, check whether
    scan($A$,$\mathbb{K}^*$,incrementor(.,$\mathbb{K}^*$)).
    If not so, output ``YES'' and terminate.

 -- Output "NO" and terminate.
```

Figure 7: Algorithm pseudoIntentRed for checking whether a set A is a pseudo-intent of a formal context $\mathbb{K}$.

7 Conclusions and Further Work

In our paper, we presented several results regarding pseudo-intents. We showed how an arbitrary implication set can be turned into a stem base (the premises of which are per definitionem just the pseudo-intents). Furthermore, based on a characterization of pseudo-intents via incrementors and using known results about quasi-intents, we provided an algorithm which allows to decide for a given formal context $\mathbb{K}$ and an attribute set P whether P is a pseudo-intent of $\mathbb{K}$ Moreover, we showed how this algorithm can be further optimized by calculating with the reduced version of the considered context. Although the complexity questions mentioned in the beginning remain unsolved, we hope that the structural insights presented in this paper might contribute to their solution. Of course this would be a main goal of further research.

On the other hand, comprehensive experiments would be the next step to investigate how the algorithms proposed here perform in practical cases (albeit not having substantial evidence for this, our conjecture is that the average complexity would be much better than suggested by our worst-case analyses).

Finally, if this should be the case, the provided algorithms could be used for developing new data-mining and exploration methods.

References

[1] Alan Day. The lattice theory of functional dependencies and normal decompositions. *International Journal of Algebra and Computation*, 2(4):409–431, 1992.

[2] William F. Dowling and Jean H. Gallier. Linear-time algorithms for testing the satisfiability of propositional Horn formulae. *J. Log. Program.*, 1(3):267–284, 1984.

[3] Bernhard Ganter. Two basic algorithms in concept analysis. Technical Report 831, FB4, TH Darmstadt, 1984.

[4] Bernhard Ganter and Rudolf Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1997. Translator-C. Franzke.

[5] J.-L. Guigues and Vincent Duquenne. Familles minimales d'implications informatives resultant d'un tableau de données binaires. *Math. Sci Humaines*, 95:5–18, 1986.

[6] S. O. Kuznetsov. On the intractability of computing the Duquenne-Guigues base. *Journal of Universal Computer Science*, 10(8):927–933, 2004.

[7] Sergei O. Kuznetsov and Sergei A. Obiedkov. Counting pseudo-intents and #P-completeness. In Rokia Missaoui and Jürg Schmid, editors, *ICFCA*, volume 3874 of *Lecture Notes in Computer Science*, pages 306–308. Springer, 2006.

[8] David Maier. *The Theory of Relational Databases.* Computer Science Press, 1983.

[9] Uta Priss. Some open problems in formal concept analysis. `http://www.upriss.org.uk/fca/problems06.pdf`, FEB 2006.

[10] Marcel Wild. Implicational bases for finite closure systems. In Wilfried Lex, editor, *Arbeitstagung Begriffsanalyse und Künstliche Intelligenz*, pages 147–169. Springer, 1991.