Technische Universität Dresden
Institute for Theoretical Computer Science
Chair for Automata Theory

LTCS–Report

A General Form of Attribute Exploration

Daniel Borchmann

LTCS-Report 13-02

Postal Address:
Lehrstuhl für Automatentheorie
Institut für Theoretische Informatik
TU Dresden
01062 Dresden

http://lat.inf.tu-dresden.de

Visiting Address:
Nöthnitzer Str. 46
Dresden

A General Form of Attribute Exploration

Daniel Borchmann*

January 11, 2013

Abstract

We present a general form of attribute exploration, a knowledge completion algorithm from formal concept analysis. The aim of this generalization is to extend the applicability of attribute exploration by a general description. Additionally, this may also allow for viewing different existing variants of attribute exploration as instances of a general form, as for example exploration on partial contexts.

1 Introduction

Attribute exploration is a well known algorithm within formal concept analysis [9]. Its main application can be summarized as *semi-automatic knowledge base completion*. Within this process, a domain expert is asked about the validity of certain implications in the domain of discourse. Based upon the answer of the domain expert, the algorithm enhances its knowledge until all implications are known to hold or not to hold in the domain, and the algorithm stops.

Attribute exploration has gained much attention since its first formulation, and for certain problems, where the original algorithm was not applicable, variations of attribute exploration have been devised. Those variations include attribute exploration on partial context [3, 4] and exploration of models of the description logic $\mathcal{EL}$ [1, 2], among others.

Of course, in all variations of attribute exploration that have been devised the overall structure of the algorithm remains the same. Furthermore, all important properties of attribute exploration remain, and one might be tempted to ask whether a general form of attribute exploration can be found that subsumes all many of these variations. The purpose of this work is to present some first considerations into this direction.

*Supported by DFG Graduiertenkolleg 1763 (QuantLA)

We shall proceed as follows. After introducing the mandatory definitions in the first section, we briefly revisit the classical description of attribute exploration as it is given in [9]. Starting from this, we motivate our generalizations and summarize the resulting algorithm together with its properties in the succeeding section. We shall have a close look at a special cases which involves *pseudoclosed sets* and results in some very nice results about the attribute exploration algorithm. Finally, we shall summarize our considerations and give an outlook on further questions.

2 Preliminaries

As attribute exploration is an algorithm from formal concept analysis, we shall begin by introducing some basic definitions from within this field. This includes notions like formal contexts, contextual derivations, implications, partial contexts and pseudoclosed sets. We shall furthermore recall the notion of closure operators on sets, which we need for our considerations.

Let G and M be two sets and let $I \subseteq G \times M$. Then the triple $\mathbb{K} := (G, M, I)$ is called a *formal context*. We shall connect with it the following intuition: The set G is the set of *objects* of $\mathbb{K}$, M is the set of *attributes* of $\mathbb{K}$ and (g, m) is an element of the incidence relation I if and only if the object g *has* the attribute m. We may also write gIm if $(g, m) \in I$. If $\mathbb{K}$ is a formal context, then the set of objects, attributes and the incidence relation is denoted by $G_{\mathbb{K}}$, $M_{\mathbb{K}}$ and $I_{\mathbb{K}}$, respectively.

Let us fix a formal context $\mathbb{K} = (G, M, I)$. If $A \subseteq G$, then the set of *common attributes of A in* $\mathbb{K}$ is denoted by

$$A' := \{\, m \in M \mid \forall g \in A : gIm \,\}$$

and likewise for $B \subseteq M$,

$$B' := \{\, g \in G \mid \forall m \in B : gIm \,\}$$

denotes the set of all *common objects of B in* $\mathbb{K}$. The sets A' and B' are called the *(contextual) derivations* of the respective sets, and the operators named $(\cdot)'$ are hence called the *derivation operators* of $\mathbb{K}$.

The derivation operators satisfy a number of useful properties, from which we shall name some in the following lemma.

2.1 Lemma ([9]) *Let $\mathbb{K} = (G, M, I)$ be a formal context and let $A, A_1, A_2 \subseteq M, B, B_1, B_2 \subseteq G$. Then the following statements hold:*

i) $A_1 \subseteq A_2 \implies A_2' \subseteq A_1'$

ii) $B_1 \subseteq B_2 \implies B_2' \subseteq B_1'$

iii) $A \subseteq A''$

iv) $B \subseteq B''$

v) $A' = A'''$

vi) $B' = B'''$

vii) $A \subseteq B' \iff A' \supseteq B$

As we view the elements of G as objects with certain attributes from M, we may ask for two sets $A, B \subseteq M$ whether all objects having all attributes from A also have all attributes from B. This can be rewritten in terms of the derivations operators as $A' \subseteq B'$. In this case, we can think of the attributes in A to *imply* the attributes in B. This motivates the definition of *implications* and *valid implications* of a formal context.

An *implication on* M is just a pair (A, B) and is often written as $A \to B$. If $\mathbb{K}$ is a formal context with attribute set M, then we may also say that $A \to B$ is an *implication of* $\mathbb{K}$. Then A is called the *premise* and B the *conclusion* of the implication. If indeed $A' \subseteq B'$, we shall call $A \to B$ a *valid* implication of $\mathbb{K}$, and we may write $\mathbb{K} \models (A \to B)$. As $A' \subseteq B' \iff B \subseteq A''$, we can observe that

$$\mathbb{K} \models (A \to B) \iff B \subseteq A''.$$

We shall denote with $\mathrm{Imp}(M)$ the set of all implications on M, with $\mathrm{Imp}(\mathbb{K})$ the set of all implications of $\mathbb{K}$ and with $\mathrm{Th}(\mathbb{K})$ the set of all valid implications of $\mathbb{K}$.

Let $\mathcal{L} \subseteq \mathrm{Imp}(\mathbb{K})$ and let $A \subseteq M$. The set A is *closed under* $\mathcal{L}$ if for all implications $(X \to Y) \in \mathcal{L}$ it holds that $X \nsubseteq A$ or $Y \subseteq A$. Let us further define

$$\begin{aligned}
\mathcal{L}^0(A) &:= A, \\
\mathcal{L}^1(A) &:= A \cup \bigcup \{\, Y \mid (X \to Y) \in \mathcal{L}, X \subseteq A \,\}, \\
\mathcal{L}^i(A) &:= \mathcal{L}^1(\mathcal{L}^{i-1}(A)) \quad \text{for } i > 1, \\
\mathcal{L}(A) &:= \bigcup_{i \in \mathbb{N}} \mathcal{L}^i(A).
\end{aligned}$$

The set $\mathcal{L}(A)$ is then the $\subseteq$-smallest superset of A that is closed under $\mathcal{L}$.

Sets of implications can be quite large and representing them concisely is often desirable. One way to do this is to use *bases* of sets of implications. For this, let M be a finite set, $\mathcal{L} \subseteq \mathrm{Imp}(M)$ and let $(A \to B) \in \mathrm{Imp}(M)$. Then $\mathcal{L}$ *entails* $A \to B$, written as $\mathcal{L} \models (A \to B)$, if and only if every formal context $\mathbb{K} = (G, N, I)$ with $N \supseteq M$ satisfies

$$\mathbb{K} \models \mathcal{L} \implies \mathbb{K} \models (A \to B).$$

It is well known that

$$\mathcal{L} \models (A \to B) \iff B \subseteq \mathcal{L}(A).$$

If $\mathcal{K} \subseteq \operatorname{Imp}(M)$ such that $\mathcal{L} \models (A \to B)$ for each $(A \to B) \in \mathcal{K}$, then we shall also write $\mathcal{L} \models \mathcal{K}$.

A set $\mathcal{B} \subseteq \operatorname{Imp}(M)$ is called *sound* for $\mathcal{L}$ if every implication from $\mathcal{B}$ is entailed by $\mathcal{L}$. $\mathcal{B}$ is said to be *complete* for $\mathcal{L}$ if every implication from $\mathcal{L}$ is entailed by $\mathcal{B}$. If $\mathcal{B}$ is both sound and complete for $\mathcal{L}$, it is called a *base* for $\mathcal{L}$. It is called a *non-redundant base* for $\mathcal{L}$ if it is $\subseteq$-minimal with respect to this property.

Let us denote with $\operatorname{Cn}(\mathcal{L})$ the set of all implications that are entailed by $\mathcal{L}$. Then

$$\begin{aligned} \mathcal{B} \text{ is sound for } \mathcal{L} &\iff \mathcal{B} \subseteq \operatorname{Cn}(\mathcal{L}), \\ \mathcal{B} \text{ is complete for } \mathcal{L} &\iff \operatorname{Cn}(\mathcal{B}) \supseteq \mathcal{L}. \end{aligned}$$

In particular, $\mathcal{B}$ is a base for $\mathcal{L}$ if and only if $\operatorname{Cn}(\mathcal{B}) = \operatorname{Cn}(\mathcal{L})$.

If $\mathcal{L} = \operatorname{Th}(\mathbb{K})$, then we call a base $\mathcal{B}$ of $\mathcal{L}$ also a base of $\mathbb{K}$. Note that $\mathcal{B}$ is sound for $\operatorname{Th}(\mathbb{K})$ if and only if all implications in $\mathcal{B}$ are valid in $\mathbb{K}$. It is well known that in this case there is an easy characterization of a set of valid implications of $\mathbb{K}$ to be complete for $\operatorname{Th}(\mathbb{K})$.

2.2 Lemma ([9]) *Let $\mathbb{K}$ be a formal context and $\mathcal{K} \subseteq \operatorname{Th}(\mathbb{K})$. Then if $\mathcal{K}(S) = S$ implies $S = S''$ for each $S \subseteq M$, then $\mathcal{K}$ is complete for* $\operatorname{Th}(\mathbb{K})$.

Proof We show $\operatorname{Cn}(\mathcal{K}) \supseteq \operatorname{Th}(\mathbb{K})$. For this let $(A \to B) \in \operatorname{Th}(\mathbb{K})$. Then $B \subseteq A''$. Then $\mathcal{K}(\mathcal{K}(A)) = \mathcal{K}(A)$ implies $\mathcal{K}(A) = (\mathcal{K}(A))'' = A''$, and hence $B \subseteq \mathcal{K}(A)$. Thus $(A \to B) \in \operatorname{Cn}(\mathcal{K})$ as required. □

From all possible bases for $\mathcal{L}$ one can explicitly describe a *canonical base* for $\mathcal{L}$ which has the remarkable property that it has minimal cardinality among all bases for $\mathcal{L}$. Let $P \subseteq M$. Then P is said to be *pseudoclosed under* $\mathcal{L}$ if

1. $P \neq \mathcal{L}(P)$ and
2. for all pseudoclosed sets $Q \subsetneq P$ it follows $\mathcal{L}(Q) \subseteq P$.

In particular, if $\mathcal{L} = \operatorname{Th}(\mathbb{K})$, then P is said to be a *pseudointent of* $\mathbb{K}$. Now the *canonical base* for $\mathcal{L}$ is defined as

$$\operatorname{Can}(\mathcal{L}) := \{\, P \to \mathcal{L}(P) \mid P \text{ pseudoclosed under } \mathcal{L} \,\}.$$

Formal contexts require a certain kind of complete knowledge about their objects: If $g \in G$ and $m \in M$ then either g has the attribute m or not. Under certain circumstances this might be inappropriate, because it might not be known whether g has the attribute m, or it is simply irrelevant for the task at hand. Therefore we shall introduce *partial contexts*, as it has been done in [3].

Let M be a set. Then a partial context $\mathbb{K}$ is a set of pairs (A, B) with $A, B \subseteq M$ such that $A \cap B = \varnothing$. Such a pair is called a *partial object description* if $A \cup B \neq M$ and a *full object description* if $A \cup B = M$. Intuitively, one can understand partial objects descriptions as a pair of positive attributes, i. e. attributes the corresponding object definitively has, and negative attributes, i. e. attributes the corresponding object definitively does not have. The objects itself are not named in partial contexts.

An *implication* for $\mathbb{K}$ is just an implication on M. Such an implication $(A \to B) \in \operatorname{Imp}(M)$ is refuted by $\mathbb{K}$ if there exists a partial object description $(X, Y) \in \mathbb{K}$ such that $A \subseteq X, B \cap Y \neq \varnothing$. If $A \subseteq M$, then the $\subseteq$-maximal set B such that $A \to B$ is not refuted by $\mathbb{K}$ exists and is given by

$$\mathbb{K}(A) := B := M \setminus \bigcup \{\, Y \mid (X, Y) \in \mathbb{K}, A \subseteq X \,\}.$$

As it turns out, the operators $(\cdot)''$, $\mathcal{L}(\cdot)$ and $\mathbb{K}(\cdot)$ are instances of the more abstract notion of *closure operators on sets*. Let again M be a set. Then a function $c \colon \mathfrak{P}(M) \to \mathfrak{P}(M)$ is said to be a *closure operator on* M if and only if

i) c is *extensive*, i. e. $A \subseteq c(A)$ for all $A \subseteq M$,

ii) c is *idempotent*, i. e. $c(c(A)) = c(A)$ for all $A \subseteq M$,

iii) c is *monotone*, i. e. $A \subseteq B \implies c(A) \subseteq c(B)$ for all $A, B \subseteq M$.

Both $(\cdot)''$ and $\mathcal{L}(\cdot)$ are closure operators on their corresponding sets of attributes. A set $A \subseteq M$ is said to be *closed under* c if $c(A) = A$. The set of all closed sets of c, i. e. the image of c, is denoted by $\operatorname{im} c$. A set $P \subseteq M$ is said to be *pseudoclosed under* c if and only if

i) $P \neq c(P)$ and

ii) for all pseudoclosed $Q \subseteq P$, it holds that $c(Q) \subseteq P$.

We shall write $c_1(\cdot) \subseteq c_2(\cdot)$ for two closure operators c_1, c_2 on a set M if and only if $c_1(A) \subseteq c_2(A)$ for all $A \subseteq M$.

3 Classical Attribute Exploration

Given a finite set M, attribute exploration semi-automatically tries to determine the set of implications that are valid in a certain domain. Together with a set $\mathcal{K}$

of already known valid implications and a formal context $\mathbb{K}$ of valid examples, attribute exploration generates implications $A \to B$ that hold in $\mathbb{K}$ but are not entailed by $\mathcal{K}$. Those implications are asked to the expert for validity. If $A \to B$ holds in the domain of discourse, it is added to the set $\mathcal{K}$. Otherwise the expert has to present a counterexample for $A \to B$ that is added to the formal context $\mathbb{K}$. The procedure terminates if there are no such implications left.

To describe attribute exploration more formally, let us define what is meant by a domain expert.

3.1 Definition Let M be a set. A *domain expert on* M is a function

$$p\colon \operatorname{Imp}(M) \to \{\top\} \cup \mathfrak{P}(M),$$

where $\top$ is a special symbol not equal to any subset of M, such that the following conditions hold:

i) If $X \to Y$ is an implication on M such that $p(X \to Y) = C \neq \top$, then $X \subseteq C, Y \nsubseteq C$. (*p gives counterexamples for false implications*)

ii) If $A \to B$ and $X \to Y$ are implications on M such that $p(A \to B) = \top$ and $p(X \to Y) = C \neq \top$, then C is closed under $\{A \to B\}$, i. e. $A \nsubseteq C$ or $B \subseteq C$. (*counterexamples do not invalidate correct implications*)

If $p(A \to B) = \top$, then we say that p *confirms* $A \to B$. Otherwise we say that p *rejects* the implication and we call the set $C = p(A \to B) \neq \top$ a *counterexample from* p *for* $A \to B$. Finally, the *theory of* p is just the set of implications that p confirms, i. e.

$$\operatorname{Th}(p) := p^{-1}(\{\top\}) = \{A \to B \mid p(A \to B) = \top\}. \qquad \diamondsuit$$

An immediate consequence of the definition is the following observation.

3.2 Lemma *Let $\mathcal{L}$ be a set of implications such that a given domain expert p confirms every implication in $\mathcal{L}$. If $\mathcal{L} \models (A \to B)$, then p confirms $A \to B$ as well.*

Proof Suppose that $p(A \to B) = C \neq \top$. Then C is closed under $\mathcal{L}$. This means that $\mathcal{L}(C) = C$. Since $\mathcal{L} \models (A \to B)$, from $A \subseteq C$ it follows that

$$B \subseteq \mathcal{L}(A) \subseteq \mathcal{L}(C) = C,$$

i. e. C is not a counterexample for $A \to B$, a contradiction. □

Before we are able to describe the attribute exploration algorithm more formally, we need to give another definition.

3.3 Definition Let M be a finite set and let $<$ be a total order on M. Then for $A, B \subseteq M$ and $i \in M$ we define

$$A \prec_i B \;:\!\iff\; i = \min_<(A \triangle B),$$

where $A \triangle B = (A \backslash B) \cup (B \backslash A)$ is the symmetric difference between A and B. If $A \prec_i B$, we say that A *is lectically smaller than* B *at* i. Furthermore, A *is lectically smaller than* B, written as $A \prec B$, if there exists $i \in M$ such that $A \prec_i B$. Finally,

$$A \preceq B \iff A = B \text{ or } A \prec B. \qquad \diamondsuit$$

It is easy to see that $\preceq$ constitutes a linear ordering on $\mathfrak{P}(M)$. We may therefore speak of the *first* lectic set and the *next* lectic set after a given subset of M.

We are now able to describe the process of attribute exploration in a more formal way.

3.4 Algorithm (Classical Attribute Exploration) Let M be a finite set, $\mathbb{K}$ be a formal context with attribute set M and let $\mathcal{K} \subseteq \operatorname{Imp}(M)$ and let p be a domain expert on M. Suppose that $\mathcal{K} \subseteq \operatorname{Th}(p) \subseteq \operatorname{Th}(\mathbb{K})$.

i) Initialize P to $\mathcal{K}(\varnothing)$.

ii) If $P'' = P$, then set P to the lectically next closed of $\mathcal{K}$, and repeat this step. If there is no such set, terminate.

iii) If p confirms $P \to P''$, then add r to $\mathcal{K}$.

iv) If p gives a counterexample C for $P \to P''$, add a new object to $\mathbb{K}$ which has exactly the attributes in C.

v) Go to ii.

In any iteration, the current value of $\mathcal{K}$ is called the set of *currently known implications* and the current value of $\mathbb{K}$ is called the *current working context.* $\diamondsuit$

A first easy observation for this algorithm is the following: Suppose the expert p is called with an implication $A \to B$ during the run of the algorithm. Let $\mathcal{K}$ be the currently known implications at this time, and let likewise $\mathbb{K}$ denote the current working context. Then for each $m \in B$ both $\operatorname{Th}(p) \models (P \to \{\, m \,\})$ and $\operatorname{Th}(p) \not\models (P \to \{\, m \,\})$ is possible. In other words, the question whether $\operatorname{Th}(p) \models (P \to \{\, m \,\})$ is not influenced by the values of $\mathcal{K}$ and $\mathbb{K}$ but solely depends on how the expert p answers. Hence all questions to the expert can be seen as *non-redundant.*

This property is very important especially in the presence of human experts which may not only be expensive to answer but might also get impatient when getting asked implications the algorithm could have inferred by itself. Therefore, this property should of course also hold for our generalized formulation of the attribute exploration, and it does, as we shall see.

But first, we shall note some of the major properties of this attribute exploration algorithm.

3.5 Theorem *Let M be a finite set, $<$ a total order on M, $\mathbb{K}$ a formal context with attribute set M, $\mathcal{K}$ a set of implications on M and let p be a domain expert on M, such that p confirms $\mathcal{K}$ and all implications confirmed by p hold in $\mathbb{K}$, i.e. $\mathcal{K} \subseteq \mathrm{Th}(p) \subseteq \mathrm{Th}(\mathbb{K})$.*

i) The attribute exploration algorithm terminates with $\mathbb{K}$, $\mathcal{K}$ and p as input.

ii) Let $\mathcal{K}'$ and $\mathbb{K}'$ be the values corresponding to $\mathcal{K}$ and $\mathbb{K}$ after the last iteration of the attribute exploration algorithm. Then $\mathcal{K}'$ is a base for $\mathrm{Th}(\mathbb{K}')$.

iii) $\mathrm{Th}(p) = \mathrm{Th}(\mathbb{K}')$ and the corresponding closure operator coincides with $\mathcal{K}'(\cdot)$.

iv) The cardinality of $\mathcal{K}' \setminus \mathcal{K}$ is the smallest possible.

v) The premises in $\mathcal{K}' \setminus \mathcal{K}$ are the $\mathcal{K}$-pseudoclosed of $\mathrm{Th}(\mathbb{K}')$. Thereby, a set $P \subseteq M$ is said to be $\mathcal{K}$-pseudoclosed under $\mathcal{L}$ for $\mathcal{K}, \mathcal{L} \subseteq \mathrm{Imp}(M)$, if and only if

  i) $P = \mathcal{K}(P)$,

  ii) $P \neq \mathcal{L}(P)$,

  iii) for each $\mathcal{K}$-pseudoclosed set $Q \subsetneq P$ of $\mathcal{L}$ it holds that $\mathcal{L}(Q) \subseteq P$.

All but the last statement of the theorem are known from [6, 9, 10]. The last statement has been mentioned partially in [10] and has been proven completely in [5].

4 Generalizing Attribute Exploration

We shall now proceed by investigating the above description of attribute exploration for possible generalizations. While doing so, besides generalizing the algorithm formally, we shall also give some intuition on why we do the generalization as proposed.

Let p be a domain expert on a set M. We start with an informal introduction of our generalizations, of which we shall name three:

1. The use of the initial formal context $\mathbb{K}$ and the background knowledge $\mathcal{K}$ can be reduced to their corresponding closure operators $(\cdot)''$ and $\mathcal{K}(\cdot)$. The only major problem here is the handling of counterexamples, which we shall discuss latter in detail. Hence instead of passing the attribute exploration algorithm a formal context and some background knowledge in the form of a set of valid implications, we instead provide two closure operators c_{poss} and c_{cert} on the set M.

   The closure operator c_{poss} takes the place of $\text{Th}(\mathbb{K})(\cdot)$ and represents the *possible knowledge* we already have about our domain of discourse. If $A \subseteq M$ is a set of attributes, then $c_{\text{poss}}(A)$ represents the attributes that *can* follow from A. Seen from another perspective, $M \backslash c_{\text{poss}}(A)$ is the set of attributes that *definitively do not follow* from A.

   In contrast to this, the closure operator c_{cert} represents the *certain* knowledge we already have. In other words, $c_{\text{cert}}(A)$ is the set of all attributes that *definitively follow* from A. This closure operators hence takes the place of the set $\mathcal{K}$ of initially known implications.

   Clearly, we need to have $c_{\text{cert}}(\cdot) \subseteq \text{Th}(p)(\cdot) \subseteq c_{\text{poss}}(\cdot)$.

2. When providing counterexamples, we observe that we actually do not need to *completely specify* them. It merely is sufficient to provide information on which attributes a certain object has and which it not, as long as this information contradicts a proposed implication. We shall take this approach and extend the algorithm to store those counterexamples in a partial context. This idea has also been discussed in [3, 4, 8].

3. The implications which are proposed to the expert are of a very special form, which guarantees certain optimality statements about the algorithm. However, for the main application of knowledge acquisition and knowledge completion, this rather special form can be viewed as a certain kind of optimization. To drop it, we may rather say that in any iteration step of the attribute exploration algorithm, we search for an *undecided implication* with respect to the current values of c_{cert} and c_{poss}, i. e. an implication $A \to B$ on M such that $c_{\text{cert}}(A) \subsetneq B \subseteq c_{\text{poss}}(A)$ and where both A and B are finite. For such an implication we cannot infer from c_{cert} and c_{poss} whether attributes $c_{\text{poss}}(A) \backslash B$ follow from A or not, and hence we have to ask the domain expert.

We shall take these observations as guidelines for our further considerations. We start by generalizing our notion of a domain expert such that we allow partial counter examples. Next we present and discuss our general form of attribute exploration that incorporates the above mentioned ideas. For this we shall also prove correctness and non-redundancy of the questions asked to the expert. Subsequently, we shall have a closer look on how to compute undecided implications in our general setting as it is done in the classical case.

4.1 Definition Let M be a set. A function $q\colon \operatorname{Imp}(M) \to \{\top\} \cup \mathfrak{P}(M)^2$ is said to be a *partial domain expert on* M if and only if $\top$ is an element not in $\mathfrak{P}(M)^2$ and the following conditions hold:

1. If for $(A \to B) \in \operatorname{Imp}(M)$ it holds that $q(A \to B) = (C, D) \neq \top$, then $C \cap D = \varnothing$, $A \subseteq C$ and $B \cap D \neq \varnothing$. (*q gives sufficient counterexamples for false implications*)
2. If $(A \to B), (X \to Y) \in \operatorname{Imp}(M)$ are such that $q(A \to B) = \top$ and $q(X \to Y) = (C, D) \neq \top$, then if $A \subseteq C$ then $B \cap D = \varnothing$. (*counterexamples do not refute correct implications*)

As in the case for domain experts, we say that q *confirms* an implication $A \to B$ if and only if $q(A \to B) = \top$. Otherwise we say that q *rejects* the implication and we call $q(A \to B) \neq \top$ a *counterexample from* q for $A \to B$. $\operatorname{Th}(q)$ shall denote the set of all implications on M that are confirmed by q. ♢

The counterexamples given by a partial domain expert can be seen as partial object descriptions that are enough to invalidate a given implication.

Let us first investigate immediate consequences from the definition. One of those is the fact, as one would expect, that $\operatorname{Th}(q)$ is closed under entailment, i. e. $\operatorname{Cn}(\operatorname{Th}(q)) = \operatorname{Th}(q)$.

4.2 Lemma *Let $\mathcal{L} \subseteq \operatorname{Imp}(M)$ for a set M and let q be a partial domain expert on M, such that q confirms all implications in $\mathcal{L}$. If $\mathcal{L} \models (A \to B)$ for some $(A \to B) \in \operatorname{Imp}(M)$, then q confirms $A \to B$ as well.*

Proof Suppose that $q(A \to B) = (C, D)$ is a counterexample from q for $A \to B$. Then $A \subseteq C$. Now $\mathcal{L}(C) \subseteq M \backslash D$ by the second condition on partial domain experts. Since $\mathcal{L} \models (A \to B)$, it follows that $B \subseteq \mathcal{L}(A) \subseteq \mathcal{L}(C) \subseteq M \backslash D$. Therefore, $B \cap D = \varnothing$, contradicting the fact that (C, D) is a counterexample for $A \to B$ from q. □

4.3 Lemma *If (C, D) is a counterexample given by a partial domain expert q on M, then $\operatorname{Th}(q)(C) \cap D = \varnothing$.*

Proof By Lemma 4.2, q confirms $C \to \operatorname{Th}(q)(C)$. Therefore, by the second condition in the definition of q, it follows $D \cap \operatorname{Th}(q)(C) = \varnothing$, as required. □

4.4 Lemma *For a partial context $\mathbb{K}$ with attribute set M and a partial domain expert q on M it holds that $\operatorname{Th}(q)(\cdot) \subseteq \mathbb{K}(\cdot)$ if and only if $\operatorname{Th}(q)(C) \subseteq M \backslash D$ for each $(C, D) \in \mathbb{K}$.*

Proof $\operatorname{Th}(q)(\cdot) \subseteq \mathbb{K}(\cdot)$ implies $\operatorname{Th}(q)(C) \cap D = \varnothing$ for each $(C, D) \in \mathbb{K}$, which is equivalent to $\operatorname{Th}(q)(C) \subseteq M \backslash D$.

For the converse let $\mathrm{Th}(q)(C) \cap D = \varnothing$ for all $(C, D) \in \mathbb{K}$. Let $A \subseteq M$. Then for every $(C, D) \in \mathbb{K}$ with $A \subseteq C$, it follows that $\mathrm{Th}(q)(A) \cap D \subseteq \mathrm{Th}(q)(C) \cap D = \varnothing$. Therefore

$$\mathrm{Th}(q)(A) \cap \bigcup \{ D \mid (C, D) \in \mathbb{K}, A \subseteq C \} = \varnothing$$

and hence $\mathrm{Th}(q)(A) \subseteq \mathbb{K}(A)$ as required. □

With those observations at hand, we are now able to state our generalized formulation of the attribute exploration algorithm.

4.5 Algorithm (General Attribute Exploration) Let M be a set, $c_{\mathrm{cert}}, c_{\mathrm{poss}}$ closure operators on M and q a partial domain expert M, such that $c_{\mathrm{cert}}(\cdot) \subseteq \mathrm{Th}(q)(\cdot) \subseteq c_{\mathrm{poss}}(\cdot)$.

i. Let $\mathbb{K} = \varnothing$.

ii. Let $A \subseteq M$ be finite and such that there exists a finite set $B \subseteq M$ with $c_{\mathrm{cert}}(A) \subsetneq B \subseteq c_{\mathrm{poss}}(A)$. If there is no such set, terminate with output $\mathbb{K}$ and c_{cert}. Otherwise consider the implication $A \to B$.

iii. If q confirms $A \to B$, then update c_{cert} to be the closure operators whose closed sets are exactly the closed sets of c_{cert} that are also closed under $\{ A \to B \}$.

iv. Otherwise let $(C, D) = q(A \to B)$ be a counterexample from q for $A \to B$. Add (C, D) to $\mathbb{K}$.

v. Replace all counterexamples $(C, D) \in \mathbb{K}$ by (C', D'), where

$$\begin{aligned} C' &:= c_{\mathrm{cert}}(C), \\ D' &:= D \cup \{ m \in M \setminus D \mid c_{\mathrm{cert}}(C \cup \{ m \}) \cap D \neq \varnothing \}. \end{aligned}$$

vi. Update c_{poss} to be the closure operator given by

$$X \mapsto c_{\mathrm{poss}}(X) \cap \mathbb{K}(X)$$

for all $X \subseteq M$.

vii. Go to ii. ◊

We can see easily that this algorithm is a generalization of Algorithm 3.4. First of all, we can turn every domain expert p into a partial domain expert q that confirms the same implications and yields

$$q(A \to B) = (p(A \to B), M \setminus p(A \to B))$$

for each $(A \to B) \in \mathrm{Imp}(M)$ that is rejected by p.

Secondly, let $\mathbb{K}$ be a formal context such that $\mathrm{Th}(p) \subseteq \mathrm{Th}(\mathbb{K})$. We can then consider the closure operator $(\cdot)''_{\mathbb{K}}$ of $\mathbb{K}$ as initial value for the closure operator c_{poss}. Suppose that we are at the beginning of an iteration of the classical attribute exploration algorithm, and let $\bar{\mathbb{K}}$ be the formal context made from all counterexamples given so far. Let $\mathbb{L}$ be the *subposition* of $\mathbb{K} = (G, M, I)$ and $\bar{\mathbb{K}} = (\bar{G}, M, \bar{I})$, i. e.

$$\mathbb{L} := (G \cup \bar{G}, M, I \cup \bar{I})$$

(note that G and $\bar{G}$ are disjoint.) Let $A \subseteq M$, and denote the derivations of A in $\mathbb{L}, \mathbb{K}, \bar{\mathbb{K}}$ by $A'_{\mathbb{L}}, A'_{\mathbb{K}}$ and $A'_{\bar{\mathbb{K}}}$, respectively; likewise for subsets of $G \cup \bar{G}$. Then

$$\begin{aligned} A''_{\mathbb{L}} &= ((A'_{\mathbb{L}} \cap G) \cup (A'_{\mathbb{L}} \cap \bar{G}))'_{\mathbb{L}} \\ &= (A'_{\mathbb{L}} \cap G)'_{\mathbb{L}} \cap (A'_{\mathbb{L}} \cap \bar{G})'_{\mathbb{L}} \\ &= A''_{\mathbb{K}} \cap A''_{\bar{\mathbb{K}}} \\ &= c_{\mathrm{poss}}(A) \cap A''_{\bar{\mathbb{K}}}. \end{aligned}$$

Hence, we only have to augment the closure operator c_{poss} by the closure operator induced by $\bar{\mathbb{K}}$. This is done incrementally in step vi of Algorithm 4.5.

Furthermore, if $\mathcal{K} \subseteq \mathrm{Imp}(M)$ is a set of implications which is to be used as background knowledge for classical attribute exploration, then for the case of generalized attribute exploration the value of c_{cert} can be taken to be $\mathcal{K}(\cdot)$.

Finally, let us comment on step ii of the general algorithm. In this step, we determine the next implication to be presented to the expert. We want to argue that the statement of this step is a generalization of the corresponding step in classical attribute exploration.

In the classical case, starting from a set $\mathcal{K}$ of known implications, we consider sets $P \subseteq M$ in lectic order that are closed under $\mathcal{K}$ but do not satisfy $P = P''_{\mathbb{L}}$, where $\mathbb{L}$ is the current working context. If $P \to P''$ is then confirmed by the expert, then it is added to $\mathcal{K}$. Otherwise, a counterexample for $P \to P''$ has to be provided by the expert and the algorithm proceeds to find the next set $Q \geq P$ with $\mathcal{K}(Q) = Q$ and $Q''_{\mathbb{L}} \neq Q$.

In step ii of the general algorithm, we only demand that we find an implication $A \to B$ such that $c_{\mathrm{cert}}(A) \subsetneq B$. This is certainly a relaxation of the case of the classical algorithm. However, in the general algorithm, we are allowed to search for such an implication arbitrarily, i. e. not restricted to some lectic order. The next theorem shows that in the case of classical attribute exploration, search along a lectic order is not a restriction. In other words, if in the classical case, there exists an implication $A \to B$ such that $A'' \subsetneq B$, then A is lectically larger then the premise of the last implication presented to the expert. Note that this is (of course) well known, as it is the basis of the correctness of classical attribute exploration. The theorem formulates this result in the language developed so far.

4.6 Theorem *Let $\mathbb{K}$ be a finite formal context, p a domain expert on $M_{\mathbb{K}}$ and $\mathcal{K} \subseteq \operatorname{Imp}(M)$ such that $\mathcal{K} \subseteq \operatorname{Th}(p) \subseteq \operatorname{Th}(\mathbb{K})$. Let $<$ be a linear order on M and $\prec$ the lectic order on $\mathfrak{P}(M)$ induced by $<$.*

Let P be the $\prec$-smallest element of $\mathfrak{P}(M)$ such that $\mathcal{K}(P) = P$ and $P \neq P''$, and let $P \subsetneq Q \subseteq P''$.

1. *If p confirms $P \to Q$ and if $S \subseteq M$ is $\prec$-minimal with $(\mathcal{K} \cup \{ P \to Q \})(S) = S$ and $S \neq S''$, then $P \prec S$.*
2. *If p rejects $P \to Q$ with counterexample $C \subseteq M$, then let $\bar{\mathbb{K}}$ be the formal context $\mathbb{K}$ augmented by the counterexample C. If then S is $\prec$-minimal with $\mathcal{K}(S) = S$ and $S \neq S''_{\bar{\mathbb{K}}}$, then $P \preceq S$.*

Proof For the first statement let us assume that $S \prec P$. Since $(\mathcal{K} \cup \{ P \to Q \})(S) = S$, $\mathcal{K}(S) = S$. Hence $S = S''$ by the prerequisites of the theorem, a contradiction. Therefore $S \succeq P$, but $S = P$ is not possible since P is not closed under $\{ P \to Q \}$.

For the second statement assume again $S \prec P$. Since $\bar{\mathbb{K}}$ has been obtained from $\mathbb{K}$ by adding a new object,

$$S''_{\mathbb{K}} \subseteq S''_{\bar{\mathbb{K}}}$$

for all $S \subseteq M_{\mathbb{K}}$. Now if $\mathcal{K}(S) = S$, by the prerequisites of the theorem we obtain $S = S''_{\mathbb{K}}$ and therefore $S = S''_{\bar{\mathbb{K}}}$, a contradiction. Hence $S \succeq P$ as required. □

Starting from the reformulation in Algorithm 4.5 of attribute exploration we shall now consider the properties this algorithm has. We shall show in this section that the algorithm, as in the classical case, does not ask question its answers it could infer itself. Furthermore, the algorithm is correct in the sense that it returns a complete description of the domain the given partial domain expert represents. Termination, however, cannot be shown in general, and we shall only give some sufficient condition.

The results in the minimality of the resulting set of confirmed implications does not hold in this general setting. For this, we have to generate the implications asked to the expert in a way similar to the classical case. We shall discuss this in more detail in the next section.

To discuss the properties of Algorithm 4.5, we need the following result.

4.7 Lemma *At the end of every iteration of the generalized attribute exploration algorithm it holds that $c_{\text{cert}}(\cdot) \subseteq \operatorname{Th}(q)(\cdot) \subseteq c_{\text{poss}}(\cdot)$ for the current values of c_{cert} and c_{poss}. In particular, $c_{\text{cert}}(X) \subseteq \mathbb{K}(X)$ holds for all $X \subseteq M$ at the end of every iteration.*

Proof We prove the claim by induction. For the base case we observe that $\mathbb{K} = \varnothing$ and therefore $\mathbb{K}(X) = M$ for all $X \subseteq M$. Furthermore $c_{\text{cert}}(\cdot) \subseteq \operatorname{Th}(q)(\cdot) \subseteq c_{\text{poss}}(\cdot)$ by the prerequisites of the algorithm.

For the induction step assume that $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot) \subseteq c_{\text{poss}}(\cdot)$ holds at the beginning of the current iteration. Assume $A, B \subseteq M$ finite such that $c_{\text{cert}}(A) \subsetneq B \subseteq c_{\text{poss}}(A)$, for otherwise nothing has to be shown. We now distinguish two cases:

i. q confirms $A \to B$. Then c_{cert} is updated to the value of

$$c'_{\text{cert}} = X \mapsto c_{\text{cert}}(\mathcal{L}(c_{\text{cert}}(X)))$$

where $\mathcal{L} = \{ A \to B \}$ and $X \subseteq M$. Since q confirms $A \to B$ and $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot)$, it follows that $c'_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot)$.

In the situation before step v, by Lemma 4.3 for every element $(C, D) \in \mathbb{K}$ it holds that $\text{Th}(q)(C) \cap D = \varnothing$ and hence $c'_{\text{cert}}(C) \cap D = \varnothing$. Moreover, $C' := c'_{\text{cert}}(C)$ is also disjoint to

$$D' := D \cup \{ m \in M \backslash D \mid c'_{\text{cert}}(C \cup \{ m \}) \cap D \neq \varnothing \}$$

and $(C' \to \{ m \}) \notin \text{Th}(q)$ for $m \in D' \backslash D$. Therefore, after step v, $\text{Th}(q)(C') \subseteq M \backslash D'$ for every $(C', D') \in \mathbb{K}$. Then by Lemma 4.4, $\text{Th}(q)(\cdot) \subseteq \mathbb{K}(\cdot)$ and therefore $c'_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot) \subseteq c_{\text{poss}}(\cdot) \cap \mathbb{K}(\cdot)$ as required.

ii. q gives (X, Y) as a counterexample for $A \to B$. Then in this iteration the value of c_{cert} is not changed. The counterexample that is effectively added to $\mathbb{K}$ is then

$$(X', Y') = (c_{\text{cert}}(X), Y \cup \{ m \in M \backslash Y \mid c_{\text{cert}}(X \cup \{ m \}) \cap Y \neq \varnothing \}).$$

Since $\text{Th}(q)(X') \subseteq M \backslash Y'$, from Lemma 4.4 and the induction hypothesis it follows that $\text{Th}(q)(\cdot) \subseteq \mathbb{K}(\cdot)$. Together with $\text{Th}(q)(\cdot) \subseteq c_{\text{poss}}(\cdot)$ we obtain $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot) \subseteq c_{\text{poss}}(\cdot) \cap \mathbb{K}(\cdot)$ as required. □

We shall at first investigate the already mentioned property that questions asked to the expert are somehow non-redundant. We state this kind of non-redundancy as the fact that the answer to a proposed implication is not predetermined by the current knowledge or by the answers given so far.

4.8 Theorem *Let M be a set, $c_{\text{cert}}, c_{\text{poss}}$ closure operators on M and q a partial domain expert on M such that $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot) \subseteq c_{\text{poss}}(\cdot)$. Suppose that we are in the $n+1$ iteration of Algorithm 4.5 and suppose that the implication $A \to B$ is asked to the expert q.*

Then for each $m \in B$ there exist two partial domain experts q_1, q_2 which return the same values as q in all iterations $i \in \{ 1, \dots, n \}$ and satisfy $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q_1)(\cdot), \text{Th}(q_2)(\cdot) \subseteq c_{\text{poss}}(\cdot)$, such that q_1 rejects $A \to \{ m \}$ and q_2 confirms $A \to \{ m \}$.

Proof Let $c^i_{\text{cert}}, c^i_{\text{poss}}, \mathbb{K}^i$ be the values of the corresponding closure operators and the current working context in iteration $i \in \{1, \dots, n\}$, respectively. Furthermore, let $A_i \to B_i$ be the implication asked in iteration i. Finally, let $\top$ be a symbol not equal to any subset of M.

We then define q_1 as follows:

$$q_1(A \to B) = \begin{cases} q(A \to B) & \text{if } (A \to B) = (A_i \to B_i) \text{ for some } i, \\ \top & \text{if } B \subseteq c^n_{\text{cert}}(A), \\ (c_{\text{cert}}(A), M \setminus c_{\text{cert}}(A)) & \text{otherwise,} \end{cases}$$

for all $(A \to B) \in \operatorname{Imp}(M)$. Then q_1 is a partial domain expert on M and $\operatorname{Th}(q_1) = \operatorname{Th}(c^n_{\text{cert}})$. Since $c^n_{\text{cert}}(\cdot) \subseteq c_{\text{poss}}(\cdot)$ by Lemma 4.7 and $m \notin c^n_{\text{cert}}(A)$, q_1 rejects $A \to \{m\}$.

To construct q_2 we consider the formal context $\mathbb{K}$ with object set $\mathbb{K}^n$, attribute set M and incidence relation $I_{\mathbb{K}}$ given by

$$(C, D) I_{\mathbb{K}} x \iff \begin{cases} x \in c^n_{\text{cert}}(C \cup \{m\}) & \text{if } m \notin D \\ x \in C & \text{otherwise} \end{cases}$$

for all $(C, D) \in \mathbb{K}^n$ and $x \in M$. By step v in Algorithm 4.5, all object intents of $\mathbb{K}$ are closed under c^n_{cert}, therefore $\operatorname{Th}(c^n_{\text{cert}}) \subseteq \operatorname{Th}(\mathbb{K})$. Together with $c_{\text{cert}}(\cdot) \subseteq c^n_{\text{cert}}(\cdot)$ follows $c_{\text{cert}}(\cdot) \subseteq \operatorname{Th}(\mathbb{K})(\cdot)$.

We now define q_2 by

$$q_2(A \to B) = \begin{cases} q(A \to B) & \text{if } (A \to B) = (A_i \to B_i) \text{ for some } i, \\ \top & \text{if } B \subseteq A'' \cap c^n_{\text{poss}}(A), \\ (X, M \setminus X) & \text{with } X = A'' \cap c^n_{\text{poss}}(A) \text{ otherwise} \end{cases}$$

for all $(A \to B) \in \operatorname{Imp}(M)$. Then q_2 is a partial domain expert with $\operatorname{Th}(q_2) = \operatorname{Th}(\mathbb{K}) \cap \operatorname{Th}(c^n_{\text{poss}})$. For this we observe that for $(C, D) \in \mathbb{K}^n$, if $m \notin D$, then $c^n_{\text{cert}}(C \cup \{m\}) \cap D = \varnothing$ by step v. Therefore, the counterexamples given for some implication $A_i \to B_i$ from q can also be given by q_2.

Since $c_{\text{cert}}(\cdot) \subseteq \operatorname{Th}(\mathbb{K})(\cdot)$ and $c_{\text{cert}}(\cdot) \subseteq c^n_{\text{poss}}(\cdot)$, it follows that $c_{\text{cert}}(\cdot) \subseteq \operatorname{Th}(q_2)(\cdot) \subseteq c_{\text{poss}}(\cdot)$.

Furthermore, $m \in c^n_{\text{poss}}(A)$ and since $\mathbb{K}^n$ does not reject $A \to B$, it follows that for each $(C, D) \in \mathbb{K}^n$ with $A \subseteq C$ that $m \notin D$. Hence, $m \in A''$ and therefore q_2 confirms $A \to B$ as required. □

One of the crucial features of attribute exploration is that it returns a complete description of the domain of discourse upon termination. This property does also hold for our generalized formulation.

4.9 Theorem *Let M be a set, c_{cert}, c_{poss} closure operators on M and let q be a partial domain expert on M. Furthermore, suppose that $c_{\text{cert}}(\cdot) \subseteq \operatorname{Th}(q)(\cdot) \subseteq c_{\text{poss}}(\cdot)$.*

Suppose that Algorithm 4.5 terminates on input c_{cert}, c_{poss} and q and denote the returned partial context by $\mathbb{K}$ and the returned closure operator by c. Let $X \subseteq M$ such that $c(X)$ is finite.

i. $\operatorname{Th}(q)(X) = c(X)$.

ii. $c(X) = c_{\text{poss}}(X) \cap \mathbb{K}(X)$.

iii. *Let $\mathcal{K}$ be the set of all implications which have been confirmed by q during the run of the algorithm. Define $c'(X)$ to be the smallest set that contains X and is closed under both c_{cert} and $\mathcal{K}(\cdot)$. Then $c'(X) = c(X)$.*

iv. *Let $\bar{\mathbb{K}} = (\mathbb{K}, M, I)$ where*

$$(C, D) I m \iff m \in C.$$

Then

$$c(X) = c_{\text{poss}}(X) \cap X'',$$

where $(\cdot)''$ denotes the double derivation operator in $\bar{\mathbb{K}}$.

Proof By Lemma 4.7, $c'_{\text{cert}}(\cdot) \subseteq \operatorname{Th}(q)(\cdot) \subseteq c'_{\text{poss}}(\cdot)$ holds at the end of every iteration in the run of the algorithm, where c'_{cert} and c'_{poss} denote the current values of the corresponding closure operators. Since the algorithm terminates, $c'_{\text{cert}}(Y) = c'_{\text{poss}}(Y)$ holds in the last iteration for all $Y \subseteq M$ if $c'_{\text{cert}}(Y)$ is finite. Since $c = c'_{\text{cert}}$ and $c'_{\text{cert}}(X) \subseteq \operatorname{Th}(q)(X) \subseteq c'_{\text{poss}}(X)$, the first assertion follows.

By induction on the number of iterations of the algorithm, one can see that at the end of every iteration of the algorithm it holds that $c'_{\text{poss}}(X) = c_{\text{poss}}(X) \cap \mathbb{K}(X)$, where c'_{poss} is the current value of the upper closure operator, c_{poss} is the original value of the upper closure operator and $\mathbb{K}$ is the current working context. Since the algorithm terminates, $c'_{\text{poss}}(X) = c(X)$ holds in the last iteration and the second claim follows.

Suppose that the algorithm is in a certain iteration and suppose that $\mathcal{K}'$ is the set of confirmed implications up to now. By induction we see that if c'_{cert} is the current value of the lower closure operator, then $c'_{\text{cert}}(X)$ is the smallest set containing X that is closed both under c_{cert} and $\mathcal{K}'(\cdot)$. As c is the last value of the lower closure operator during the run of the algorithm, $c(X) = c'(X)$, which shows the third claim.

For the last claim we observe the following relations:

$$\begin{aligned} X'' &= \bigcap_{(C,D)\in\mathbb{K}, X\subseteq C} C \\ &\subseteq \bigcap_{(C,D)\in\mathbb{K}, X\subseteq C} M\backslash D \\ &= \mathbb{K}(X). \end{aligned}$$

By step v of the algorithm, C is closed under c for every $(C, D) \in \mathbb{K}$. Therefore, $c(X) \subseteq X''$. Together this yields

$$c_{\mathrm{poss}}(X) \cap c(X) \subseteq c_{\mathrm{poss}}(X) \cap X'' \subseteq c_{\mathrm{poss}}(X) \cap \mathbb{K}(X)$$

and since $c(X) \subseteq c_{\mathrm{poss}}(X)$ and $c(X) = c_{\mathrm{poss}}(X) \cap \mathbb{K}(X)$, the last claim follows. □

Termination of the generalized attribute exploration algorithm is not guaranteed in general (i. e. when M is infinite and c_{cert} and c_{poss} are arbitrary). Hence, termination normally has to be shown for the concrete application at hand. We can, however, give some sufficient condition which may still be helpful.

4.10 Theorem *The general attribute exploration algorithm with input c_{cert}, c_{poss} and a partial domain expert q terminates if there are only finitely many closure operators c on M such that $c_{\mathrm{cert}}(\cdot) \subsetneq c(\cdot) \subsetneq c_{\mathrm{poss}}(\cdot)$.*

Proof The claim follows easily if we can show that in every iteration of attribute exploration either the value of c_{cert} is updated to a new value c'_{cert} such that $c_{\mathrm{cert}} \subsetneq c'_{\mathrm{cert}} \subseteq c_{\mathrm{poss}}$ or, likewise, if the value for c_{poss} is updated to a new value c'_{poss} such that $c_{\mathrm{cert}} \subseteq c'_{\mathrm{poss}} \subsetneq c_{\mathrm{poss}}$.

Let A be such that $c_{\mathrm{cert}}(A) \neq c_{\mathrm{poss}}(A)$ and let $B \subseteq M$ be finite such that $c_{\mathrm{cert}}(A) \subsetneq B \subseteq c_{\mathrm{poss}}(A)$. If q confirms $A \to B$, then c_{cert} is updated to the value

$$c'_{\mathrm{cert}}(X) = c_{\mathrm{cert}}(\mathcal{L}(c_{\mathrm{cert}}(X))),$$

where $\mathcal{L} = \{ A \to B \}$ and $X \subseteq M$. Clearly, $c_{\mathrm{cert}}(\cdot) \subsetneq c'_{\mathrm{cert}}(\cdot)$ and by Lemma 4.7, $c'_{\mathrm{cert}}(\cdot) \subseteq c_{\mathrm{poss}}(\cdot)$.

If q yields a counterexample (C, D) for $A \to B$, then the new value c'_{poss} for c_{poss} is computed by

$$c'_{\mathrm{poss}}(X) = c_{\mathrm{poss}}(X) \cap \mathbb{K}(X)$$

for $X \subseteq M$. It follows that $c'_{\mathrm{poss}}(\cdot) \subseteq c_{\mathrm{poss}}(\cdot)$ and $c'_{\mathrm{poss}}(A) \subseteq c_{\mathrm{poss}}(A)\backslash D \subsetneq c_{\mathrm{poss}}(A)$, since $C \subseteq A$, $B \subseteq c_{\mathrm{poss}}(A)$ and $B \cap D \neq \varnothing$. By Lemma 4.7 it follows that $c_{\mathrm{cert}}(X) \subseteq \mathbb{K}(X)$ for all $X \subseteq M$. Hence $c_{\mathrm{cert}}(\cdot) \subseteq c'_{\mathrm{poss}}(\cdot) \subsetneq c_{\mathrm{poss}}(\cdot)$ as required. □

Of course, if after finitely many iterations the situation of the theorem is reached, the generalized attribute exploration will terminate as well.

5 Computing Undecided Implications

Computing undecided implications is a crucial step in the algorithm. We therefore want to pay some more attention on how the classical way of computing those implications can be carried over to our generalized setting.

Firstly, let us observe that Theorem 4.6 also applies to the general setting of Algorithm 4.5.

5.1 Theorem *Let $\mathbb{K}$ be a partial formal context, q a partial domain expert on $M_{\mathbb{K}}$ and let $c_{\mathrm{cert}}, c_{\mathrm{poss}}$ be closure operators on $M_{\mathbb{K}}$ such that*

$$c_{\mathrm{cert}}(A) \subseteq \mathrm{Th}(q)(A) \subseteq c_{\mathrm{poss}}(A) \cap \mathbb{K}(A)$$

is true for all $A \subseteq M$. Let $<$ be a strict linear order on M and $\preceq$ the lectic order on $\mathfrak{P}(M)$ induced by $<$. Let P be the $\prec$-smallest element of $\mathfrak{P}(M)$ such that $c_{\mathrm{cert}}(P) = P$, $P \neq c_{\mathrm{poss}}(P) \cap \mathbb{K}(P)$, and let $P \subsetneq Q \subseteq c_{\mathrm{poss}}(P) \cap \mathbb{K}(P)$.

1. *If q confirms $P \to Q$, then let $\mathcal{K}' = \mathcal{K} \cup \{ P \to Q \}$ and let $\bar{\mathbb{K}}$ be the updated partial context as described in step v in Algorithm 4.5. If $S \subseteq M$ is $\prec$-minimal with $\mathcal{K}'(S) = S$ and $S \neq c_{\mathrm{poss}}(S) \cap \bar{\mathbb{K}}(S)$, then $P \prec S$.*
2. *If q rejects $P \to Q$ with counterexample $(C, D) \subseteq M$, then let $\bar{\mathbb{K}}$ be the formal context $\mathbb{K}$ augmented by the counterexample (C, D). If S is $\prec$-minimal with $\mathcal{K}(S) = S$ and $S \neq c_{\mathrm{poss}}(S) \cap \bar{\mathbb{K}}(S)$, then $P \preceq S$.*

Proof For the first statement let us assume that $S \prec P$. Since $\mathcal{K}'(S) = S$, $\mathcal{K}(S) = S$. Hence $S = c_{\mathrm{poss}}(S) \cap \mathbb{K}(S)$ by the prerequisites of the theorem. If we denote with $(\bar{C}, \bar{D}) \in \bar{\mathbb{K}}$ the pairs originating from $(C, D) \in \mathbb{K}$, then

$$\begin{aligned} \mathbb{K}(S) &= M_{\mathbb{K}} \setminus \bigcup \{ D \mid (C, D) \in \mathbb{K}, S \subseteq C \} \\ &\supseteq M_{\mathbb{K}} \setminus \bigcup \{ \bar{D} \mid (\bar{C}, \bar{D}) \in \bar{\mathbb{K}}, S \subseteq \bar{C} \} \\ &= \bar{\mathbb{K}}(S), \end{aligned}$$

since $C \subseteq \bar{C}$ and $D \subseteq \bar{D}$ for each $(C, D) \in \mathbb{K}$. Therefore, $S = c_{\mathrm{poss}}(S) \cap \bar{\mathbb{K}}(S)$, a contradiction. Thus $S \succeq P$, but $S = P$ is not possible since P is not closed under $\{ P \to Q \}$.

For the second statement assume again $S \prec P$. Since $\bar{\mathbb{K}}$ has been obtained from $\mathbb{K}$ by adding a new object,

$$\bar{\mathbb{K}}(S) \subseteq \mathbb{K}(S)$$

for all $S \subseteq M_{\mathbb{K}}$. Now if $\mathcal{K}(S) = S$, the prerequisites of the theorem imply $S = c_{\mathrm{poss}}(S) \cap \mathbb{K}(S)$ and therefore $S = c_{\mathrm{poss}}(S) \cap \bar{\mathbb{K}}(S)$, a contradiction. Hence $S \succeq P$ as required. □

Note that if there exists a finite set $P \subseteq M_{\mathbb{K}}$ with $c_{\text{cert}}(P) = P$ and $P \neq c_{\text{poss}}(P) \cap \mathbb{K}(P)$, then the $\prec$-minimal set under these constraints is also finite.

5.2 Lemma *Let q be a partial domain expert on a set M and let c_1, c_2 be closure operators on M such that $\operatorname{Th}(c_1) \subseteq \operatorname{Th}(q) \subseteq \operatorname{Th}(c_2)$. If for all finite sets $P \subseteq M$, $c_1(P) = P$ implies $c_2(P) = P$, then $c_1(P) = c_2(P) = \operatorname{Th}(q)(P)$ holds for all $P \subseteq M$ whenever $c_1(P)$ is finite.*

Proof Let $P \subseteq M$ such that $c_1(P)$ is finite. Then $c_1(c_1(P)) = c_1(P)$ implies $c_2(P) = c_2(c_1(P)) = c_1(P)$. Since $c_1(P) \subseteq \operatorname{Th}(q)(P) \subseteq c_2(P)$, the desired equality follows. □

We have seen that a lot of the useful properties of attribute exploration remain true in our generalized form of Algorithm 4.5. However, we have not discussed the property of the classical attribute exploration that the number of questions which the expert confirms is minimal. Indeed, we cannot expect that from our generalization, as we have not opposed any restriction on the order in which implications are asked. It is therefore possible to ask an implication $A \to B$, which is confirmed, just to ask in the next iteration an implication $A \to C$ with $C \supseteq B$, which might also get confirmed. It is therefore advisable to always ask implications with $\subseteq$-maximal conclusions. However, even in that case it might not be clear whether the number of confirmed implications asked is really minimal.

We therefore want to discuss in this section whether it is possible to modify our general attribute exploration such that the number of questions asked such that the expert confirms is the smallest possible. For this we shall try to adapt the computation of undecided implications from the classical case.

Let us recall how implications asked to a domain expert p are computed in the case of classical attribute exploration, as discussed in Algorithm 3.4. For this suppose that we are in a certain iteration of the algorithm, with known implications $\mathcal{K}$, working context $\mathbb{K}$ and P the last computed premise. Further suppose that we have fixed a total order on the set M before the start of the algorithm, which induces a lectic order $\preceq$ on $\mathfrak{P}(M)$. Then, in the classical case, we compute the lectically smallest set $Q \subseteq M$ after P that is closed under $\mathcal{K}$ and that is not an intent of $\mathbb{K}$. The implication $Q \to Q''$ is then asked to p.

Computing the lectically next set after a set P can be done using the Next-Closure algorithm [7]. However, for theoretical considerations we can neglect lectic orderings, as we shall see in a moment.

Let M be a finite set. To guarantee that the number of confirmed implications is as small as possible, we change step ii to:

ii'. Let $A \subseteq M$ be such that $A = c_{\text{cert}}(A) \subsetneq c_{\text{poss}}(A)$ and A is $\subseteq$-minimal with respect to this property. Consider the implication $A \to c_{\text{poss}}(A)$.

This is a generalization of the corresponding step in the classical case: if, in the classical case, P is the premise of the last implication asked, then the lectically next set Q after P is a $\subseteq$-minimal set with $Q = \mathcal{K}(Q) \subsetneq Q''$, and the implication $Q \to Q''$ is asked next.

Before we give the formal statement of the fact that this indeed yields an algorithm that always asks a minimal number of confirmed implications, we shall give the following definition.

5.3 Definition Let c_1, c_2 be two closure operators on a finite set M and let $P \subseteq M$. Then P is said to be c_1*-pseudoclosed under* c_2 if and only if

i. $c_1(P) = P$,

ii. $c_2(P) \neq P$,

iii. for all $Q \subsetneq P$ being c_1-pseudoclosed under c_2 it follows that $c_2(Q) \subseteq P$. $\Diamond$

5.4 Theorem *Consider Algorithm 4.5 with step ii replaced by step ii'.*

Let M be a finite set, q a partial domain expert on M, c_{cert}, c_{poss} closure operators on M such that $c_{\text{cert}}(\cdot) \subseteq \text{Th}(q)(\cdot) \subseteq c_{\text{poss}}(\cdot)$. Let $\mathcal{K}$ be the set of confirmed implications during the run of the algorithm with input c_{cert}, c_{poss} and q, and let c be the returned closure operator.

Then the premises of the implications in $\mathcal{K}$ are exactly the c_{cert}-pseudoclosed sets of c.

Proof We show that a set $A \subseteq M$ is a c_{cert}-pseudoclosed set of c if and only if the implication $A \to c(A)$ is asked to and confirmed by q. We shall do so using well-founded induction, which is possible since M is finite.

Let A be a premise of a confirmed implication $A \to B$. It follows that $B = c'_{\text{poss}}(A)$ for the corresponding value of c'_{poss} in the iteration in which $A \to B$ is asked to q. Then A is closed under c_{cert} and under all currently known implications, i. e. under

$$\{ X \to Y \mid (X \to Y) \in \mathcal{K}, X \subseteq A \}.$$

Suppose that their exists an implication $(X \to Y) \in \mathcal{K}$ such that $X \subseteq B$. Then $Y \subseteq c'_{\text{poss}}(B) = c'_{\text{poss}}(A) = B$. Therefore, B is closed under $\mathcal{K}$ and hence $B = c(A)$.

We shall show next that A is a c_{cert}-pseudoclosed set of c. We already know that A is closed under c_{cert}. Furthermore, since $A \to B$ is asked to q, $B \neq A$ and therefore $A \neq c(A)$.

Let $R \subsetneq A$ be a c_{cert}-pseudoclosed set of c. By the induction hypothesis, $R \to c(R)$ is asked to and confirmed by q. Since A is closed under all those implications, it follows that $c(R) \subseteq A$ as required.

Conversely, let A be a c_{cert}-pseudoclosed set of c. By the induction hypothesis, for all c_{cert}-pseudoclosed sets $R \subsetneq A$ the implication $R \to c(R)$ is asked to and confirmed by q. Since $c(R) \subseteq A$ and $c_{\text{cert}}(A) = A$ it follows that A is $\subseteq$-minimal with respect to being closed under c_{cert} and all confirmed implications $X \to Y$ with $X \subseteq A$. Therefore, $A \to c'_{\text{poss}}(A)$ will be asked in a certain iteration, with the corresponding value of c'_{poss}. Since $c(A) \subseteq c'_{\text{poss}}(A)$ and $A \neq c(A)$, after a finite number of counterexamples $A \to c(A)$ will be asked to and confirmed by q. $\square$

Recall the fact that the set

$$\mathcal{K} := \{\, P \to c(P) \mid P \text{ is } c_{\text{cert}}\text{-pseudoclosed set of } c \,\}$$

has minimal cardinality such that every set $A \subseteq M$ is closed under c if and only if A is closed under c_{cert} and $\mathcal{K}$. This has been proven in [5, 10] for the case of $c_{\text{cert}} = \mathcal{K}(\cdot)$ for a set $\mathcal{K} \subseteq \operatorname{Imp}(M)$ and $c = (\cdot)''$ for some given formal context $\mathbb{K}$ with $\mathbb{K} \models \mathcal{K}$. However, the proof given there also holds in our general setting.

5.5 Theorem ([5, 10]) *Let c_1, c_2 be closure operators on a set M such that $c_1(A) \subseteq c_2(A)$ for all $A \subseteq M$. We call a set $\mathcal{K} \subseteq \operatorname{Imp}(M)$ an c_1-*base of c_2 *if and only if every set $A \subseteq M$ is closed under c_2 if and only if A is closed under c_1 and $\mathcal{K}$.*

Now let

$$\mathcal{K} := \{\, P \to c_2(P) \mid P \text{ is } c_1\text{-pseudoclosed set of } c_2 \,\}.$$

Then $\mathcal{K}$ is a c_1-base of c_2 with minimal cardinality among all c_1-bases of c_2.

Proof We first show that $\mathcal{K}$ is a c_1-base of c_2. For this let $A \subseteq M$. Indeed, if $A = c_2(A)$, then $A \subseteq c_1(A) \subseteq c_2(A)$, i. e. $A = c_1(A)$. Furthermore, if P is a c_1-pseudclosed set of c_2 with $P \subseteq A$, then $c_2(P) \subseteq c_2(A) = A$. Hence A is closed under c_1 and $\mathcal{K}$.

Conversely, let A be closed under c_1 and $\mathcal{K}$. If P is a c_1-pseudoclosed set of c_2 with $P \subseteq A$, then $c_2(P) \subseteq A$ as well. Hence, if $P \neq c_2(P)$, then P would be a c_1-pseudclosed set of c_2 and $(P \to c_2(P)) \in \mathcal{K}$, contradicting $P \neq c_2(P)$. Therefore, $P = c_2(P)$ and $\mathcal{K}$ is a c_1-base of c_2.

We now prove that $\mathcal{K}$ has minimal cardinality among all c_1-bases of c_2. Let $\mathcal{L}$ be another c_1-base of c_2. Without loss of generality we can assume that for each $(A \to B) \in \mathcal{L}$, $B = c_2(A)$. For this we shall show that for each c_1-pseudoclosed set P of c_2 there exists an implication $(A_P \to c_2(A_P)) \in \mathcal{L}$ and that for two c_1-pseudoclosed sets P, Q of c_2, $A_P = A_Q$ always implies $P = Q$.

Let P be a c_1-pseudclosed set of c_2. Then $P \neq c_2(P)$ and since $\mathcal{L}(P) = c_2(P)$, there must exist an implication $(A_P \to c_2(A_P)) \in \mathcal{L}$ with $A_P \subseteq P, c_2(A_P) \nsubseteq P$.

Let Q be a c_1-pseudoclosed set of c_2 and assume that $A_P = A_Q = A$. Then $A \subseteq P \cap Q$ and hence $c_2(A) \subseteq c_2(P \cap Q)$. Furthermore, $c_2(P \cap Q) \nsubseteq P$, for otherwise $c_2(A) \subseteq c_2(P \cap Q) \subseteq P$. Since $c_2(P \cap Q) \nsubseteq Q$ holds likewise, we obtain $c_2(P \cap Q) \nsubseteq P \cap Q$ and in particular $P \cap Q \neq c_2(P \cap Q)$. Furthermore, since $P = c_1(P), Q = c_1(Q)$, $P \cap Q = c_1(P \cap Q)$ holds as well.

Since $P \cap Q \neq c_2(P \cap Q)$ and $P \cap Q = c_1(P \cap Q)$, there must exists an implication $(C \to c_2(C)) \in \mathcal{L}$ such that

$$C \subseteq P \cap Q, \quad c_2(C) \nsubseteq P \cap Q.$$

Since $c_2(C) \nsubseteq P \cap Q$, $c_2(C) \nsubseteq P$ or $c_2(C) \nsubseteq Q$ holds, and we can assume without loss of generality that $c_2(C) \nsubseteq P$. But then $C \subseteq P, c_2(C) \nsubseteq P$ and both P and C are c_1-pseudoclosed sets of c_2, hence they cannot be different. Therefore, $P = C$ and therefore

$$P \subseteq P \cap Q \subseteq Q,$$

i. e. $P = P \cap Q$. Then $c_2(P) = c_2(P \cap Q) \nsubseteq Q$ as we have seen before. Since both P and Q are c_1-pseudclosed sets of c_2 and $P \subseteq Q, c_2(P) \nsubseteq Q$, P and Q cannot be different and $P = Q$ follows, as desired. □

Summing up, we obtain our desired result.

5.6 Corollary *The number of confirmed implications during the run of the general attribute exploration algorithm 4.5, where step ii is replace by step ii', is as small as possible.*

6 Conclusions

Starting from a classical formulation of attribute exploration using domain experts, we have presented a more general formulation of attribute exploration that is able to work with abstractly given closure operators and can handle partially given counterexamples. We have also seen that most of the properties of classical attribute exploration remain in general or, as in the case of minimality of confirmed implications, under certain restrictions.

References

[1] Franz Baader and Felix Distel. A Finite Basis for the Set of $\mathcal{EL}$-Implications Holding in a Finite Model. In Raoul Medina and Sergei Obiedkov, editors, *Proceedings of the 6th International Conference on Formal Concept Analysis, (ICFCA 2008)*, volume 4933 of *Lecture Notes in Artificial Intelligence*, pages 46–61. Springer Verlag, 2008.

[2] Franz Baader and Felix Distel. Exploring finite models in the description logic $\mathcal{EL}_{\text{gfp}}$. 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 146–161. Springer Verlag, 2009.

[3] Franz Baader, Bernhard Ganter, Ulrike Sattler, and Baris Sertkaya. Completing description logic knowledge bases using formal concept analysis. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI-07)*, pages 230–235. AAAI Press, 2007.

[4] Peter Burmeister and Richard Holzer. Treating incomplete knowledge in formal concept analysis. In Bernhard Ganter, Gerd Stumme, and Rudolf Wille, editors, *Formal Concept Analysis*, volume 3626 of *Lecture Notes in Computer Science*, pages 114–126. Springer, 2005.

[5] Felix Distel. *Learning Description Logic Knowledge Bases from Data Using Methods from Formal Concept Analysis*. PhD thesis, TU Dresden, 2011.

[6] Bernhard Ganter. Attribute exploration with background knowledge. *Theor. Comput. Sci.*, 217(2):215–233, 1999.

[7] Bernhard Ganter. Two basic algorithms in concept analysis. In Léonard Kwuida and Baris Sertkaya, editors, *ICFCA*, volume 5986 of *Lecture Notes in Computer Science*, pages 312–340. Springer, 2010.

[8] Bernhard Ganter, Sergei Obiedkov, Sebastian Rudolph, and Gerd Stumme. Conceptual Exploration. in preparation.

[9] Bernhard Ganter and Rudolph Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer, Berlin-Heidelberg, 1999.

[10] Gerd Stumme. Attribute exploration with background implications and exceptions. In H.-H. Bock and W. Polasek, editors, *Data Analysis and Information Systems. Statistical and Conceptual approaches. Proc. GfKl'95. Studies in Classification, Data Analysis, and Knowledge Organization 7*, pages 457–469, Heidelberg, 1996. Springer.