Generalized Domination in Closure Systems

Anne Berry*, Eric SanJuan†, Alain Sigayret*

1st February 2003

Abstract

In the context of extracting concepts (which are maximal item sets) and association rules from a binary data base, the graph-theoretic notion of domination was recently used to characterize the neighborhood of a concept in the corresponding lattice.

In this paper, we show that the notion of domination can in fact be extended to any closure operator on a finite universe. This generalization enables us to endow notions related to Formal Concept Analysis with a logical interpretation into a set of Horn clauses.

Our results also enable us to present a prospective algorithmic process which uses only local information inherited by a concept from its direct predecessors to generate rules, instead of repeatedly using all previously defined rules. This algorithm, which is very promising in practical applications, as it can be implemented to run quickly, can also be applied in the general case.

1 Introduction

In the context of mining for information in binary data bases, recent works by Ganter and Wille use Formal Concept Analysis to investigate **concepts**, which are the maximal rectangles of the binary relation and correspond to a maximal factorization of item sets; this is used in a combinatorial approach for extracting patterns from a data base.

Equally important, the related problem of rule generation, which corresponds to finding functional dependencies in data bases, is of major importance in data analysis, for wide-spread applications such as behavioural prediction, artificial intelligence, modelization of genomic phenomena, and so forth. Recent work has been done by Guigues and Duquenne to define a minimum set of exact association rules.

Mathematical investigation has shown that concepts as well as rules are associated with several mutually inclusive closure lattices.

These lattices are potentially of exponential size, and as there may be even more rules than there are concepts, efficient algorithmic techniques are actively being sought to deal with these problems.

An interesting breakthrough was initiated by Bordat when remarking that in order to generate the neighbors of a given concept in the lattice, no information on other concepts is required. However, state of the art rule generation algorithms require, in order to generate one rule, information on **all** previously generated rules, a set which it is not always feasible to handle.

Our general purpose in this mathematical-oriented paper is to study various relationships between different formal approaches, in view of using mathematical and/or algorithmic results which stem from different fields of discrete mathematics.

Several approaches have been proposed very recently in this direction.

SanJuan in [San 99] and [San 02] used Heyting algebra to modelize Rough Sets, which among other things explored the relationships between Formal Concept Analysis and functional dependencies, as well as involving prime implicants.

Berry and Sigayret in [BS 02] proposed a representation of a concept lattice by a graph, where the graph-related notions of domination and maxmods were used, as well as that of minimal separation. Bordat's results were explained and extended, the cover of a concept characterized using only local in-

*LIMOS (UMR CNRS 6158), Université Blaise Pascal, Ensemble Scientifique des Cézeaux, Batiment ISIMA, 63173 Aubière cedex, France. berry@isima.fr, sigayret@isima.fr

†LITA (EA 3097), IUT de Metz, Ile du Saulcy, 57045 Metz cedex 1, France. eric.sanjuan@iut.univ-metz.fr

formation. This work established a relationship between graph theory and concept lattices, and was rewarded by immediate algorithmic results in terms of concept generation with very good worst-time analysis.

In this paper, we show that we can extend the notion of domination to any closure operator defined on a finite universe U. This develops into a series of interesting consequences: we use this very general notion to establish novel relationships between Formal Concept Analysis, the theory of closure spaces, Horn functions, and boolean analysis.

We finish the paper by using domination to propose a new prospective algorithmic approach, which computes a generator of association rules, using local information inherited from the predecessors of a given concept rather than the entire set of already defined rules.

2 Generalizing domination to general closure systems

2.1 Closure operators induced by binary relations

Our starting point is the Galois lattice or Concept lattice ([Wil 82] defined by a binary relation:

Given a finite set $\mathcal{P}$ of 'properties', a finite set $\mathcal{O}$ of objects, and a relation $R \subseteq \mathcal{P} \times \mathcal{O}$, we will define, for any subset X of $\wp(\mathcal{P})$ or $\wp(\mathcal{O})$:

- $N(X) = \{y \in \mathcal{O} : \forall x \in X, (x,y) \in R\}$ if $X \subseteq \mathcal{P}$,
- $N(X) = \{y \in \mathcal{P} : \forall x \in X, (y,x) \in R\}$ if $X \subseteq \mathcal{O}$.

We will denote by $N^2(X)$ the set $N(N(X))$. This defines what is called a closure operator, and the pairs $(N^2(X), N(X))$ are called maximal rectangles or concepts.

The way [BS 02] defined domination between properties a and b can be reformulated as: a dominates b if $N(a) \subseteq N(b)$; we use this as a pre-order to define the equivalence classes as maxmods: $X \subseteq \mathcal{P}$ is a maxmod of R if $\forall x, y \in X, N(x) = N(y)$ and X is maximal for this property. A maxmod X dominates a maxmod Y if $N(X) \subset N(Y)$. For $X \subseteq \mathcal{P}, Y \subseteq \mathcal{O}$, we will denote by $R(X,Y)$ the sub-relation of R defined by:

$$R(X,Y) = \{(x,y) \in R : x \in X, y \in Y\}$$

Characterization 2.1 *A concept B covers a concept A iff $B - A$ is a non-dominating maxmod in $R((\mathcal{P} - A), N(A))$.*

Note that the idea of working on the sub-relation already appears in [Bor 86].

Example 2.2 *We will use the relation from [GD 86]:*
$\mathcal{P} = \{a,b,c,d,e\}$, $\mathcal{O} = \{1,2,3,4\}$.
Relation R:

	a	*b*	*c*	*d*	*e*
1	×	×			
2	×		×		
3		×	×	×	
4				×	×

The associated Concept Lattice is shown in Figure 1.
Concept a corresponds to sub-relation $R(\mathcal{P} - \{a\}, N(a)) = R(\{b,c,d,e\},\{1,2\})$, in which $N(b) = \{1\}$, $N(c) = \{2\}$, $N(d) = N(e) = \emptyset$.
Maxmods are $\{b\}$, $\{c\}$ and $\{d,e\}$; the non-dominating maxmods are $\{b\}$ and $\{c\}$, while $\{d,e\}$ dominates both $\{b\}$ and $\{c\}$.
The cover of a is $\{ab, ac\}$.

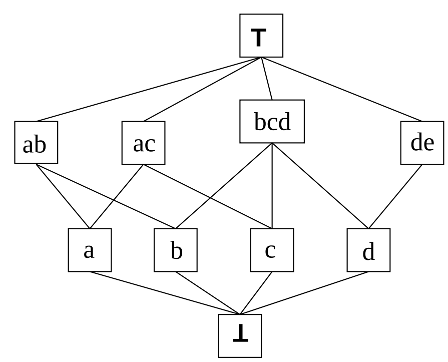

Figure 1: Concept Lattice $\mathscr{L}(R)$ associated with relation R from Example 2.2.

We will see in the rest of this paper that we can extend the notion of domination to a general closure, essentially by stating that b dominates a iff the closure of a is included in the closure of b.

2.2 General closure systems

We will now extend the notion of domination to any closure system defined on a **finite** set U.

We will need some preliminary definitions on closure systems.

Definition 2.3 *A unary operator C on $\wp(U)$ is called a* ***closure operator*** *on U if for $A, B \subseteq U$:*

1. $A \subseteq C(A)$
2. $C(C(A)) = C(A)$
3. *if* $A \subseteq B$*, then* $C(A) \subseteq C(B)$

A ***subset*** *A of U is said to be* ***closed*** *if $C(A) = A$.*

Definition 2.4 *A family $\mathcal{F}$ of subsets of U is a* ***closure system*** *if for any $\mathcal{X} \subseteq \mathcal{F}$, we have $\bigcap \mathcal{X} \in \mathcal{F}$.*

If $\mathcal{F}$ is a closure system, then $(\mathcal{F},\subseteq)$ is a **complete lattice** such that $U \in \mathcal{F}$ and for any $X, Y \in \mathcal{F}$, $X \cap Y \in \mathcal{F}$.

The collection $\mathcal{F}_C = \{C(A) : A \subseteq U\}$ of closed sets is a closure system, with the property that for any $A \subseteq U$, $C(A)$ is the smallest element F of $\mathcal{F}_C$ such that $A \subseteq F$.

Definition 2.5 *Given a closure operator C on U, a closed set B is said to* ***cover*** *a closed set A if for any $X \subseteq U$,*

$$A \subset X \subseteq B \quad \Rightarrow \quad C(X) = B$$

Example 2.6 *Let us use the concepts from Example 2.2 to define a closure system on $U = \{a, b, c, d, e\}$:*

$$\mathcal{F}_C = \{\emptyset, \{a\}, \{b\}, \{c\}, \{d\}, \{a,b\}, \{a,c\}, \{b,c,d\}, \{d,e\}, U\}$$

In this example, $C(\{e\}) = \{d, e\}$ and $C(\{a, e\}) = U$; ab and ac cover a.

Definition 2.7 *Given a closure operator C and a closed set A, we will define a binary relation on $U - A$, which we will denote by $dom_C(A)$, by setting for any $x, y \in U - A$:*

$$(x, y) \in dom_C(A) \iff y \in C(A \cup \{x\})$$

We will say that x ***dominates*** *y in A.*

For any closed set A, $dom_C(A)$ is a pre-ordering (i.e. $dom_C(A)$ is reflexive and transitive). As a result, $U - A$ can be partitionned into equivalence classes which we will call ***maxmods****; this results into a quotient order, which is a partial order on the maxmods.*

Clearly, a subset $M \subseteq U - A$ is a maxmod iff it is a maximal set such that for any $x \in M$, $M \subseteq C(A \cup \{x\})$.

The notion of domination is naturally extended to maxmods:

Definition 2.8 *We denote by $Dom_C(A)$ the binary relation defined on the maxmods of $dom_C(A)$:*

$$\begin{aligned}(X, Y) \in Dom_C(A) \\ &\iff (\forall x \in X)(\forall y \in Y)(x, y) \in dom_C(A) \\ &\iff (\exists x \in X)(\exists y \in Y)(x, y) \in dom_C(A)\end{aligned}$$

We will say that maxmod X ***dominates*** *maxmod Y.*

Using Definition 2.8, we can now reformulate Characterization 2.1 into a general statement:

Characterization 2.9 *Given a closure operator C on a finite set U, and two closed sets A, B, then B covers A iff $B - A$ is a non-dominating maxmod of $dom_C(A)$ (or, equivalently, a minimal element of $Dom_C(A)$).*

2.3 Logical representation of generalized domination

We assume that the reader is familiar with the theory of propositional Horn clauses.

To clarify the relationship between closure systems and prime implicates of a Horn function, we need to associate a set of propositional Horn clauses with the subsets of U.

Definition 2.10 *Let C be a closure operator on U; let A be a subset of U. Then A can be associated with the following set of propositional Horn clauses $H_C(A)$:*

1. *if* $C(A) = U$ *then* $H_C(A) = \{A \rightarrow\}$
2. *else* $H_C(A) = \{A \rightarrow b : b \in C(A) - A\}$.

A clause $A \rightarrow$ is said to be ***negative****, and is sometimes denotated by $A \rightarrow U$.*

A clause $A \rightarrow b$ is said to be **pure**. *We shall sometimes write a set of pure Horn clauses $\{A \rightarrow b : b \in B, b \notin A\}$ simply as $A \rightarrow B$.*

Finally, if $\mathcal{X}$ is a subset of $\wp(U) - \mathcal{F}_C$, then $H_C(\mathcal{X})$ is defined as the set of clauses:

$$H_C(\mathcal{X}) = \bigcup_{A \in \mathcal{X}} H_C(A)$$

We apply this definition to associate a Boolean function with a closure operator C.

Definition 2.11 *Let C be a closure operator on U; we denote by H_C the set of Horn clauses $H_C(\wp(U) - \mathcal{F}_C)$ and by f_C the Boolean function represented by H_C.*

Given a binary relation R and using results from [Fla 76], we can deduce f_{N^2} from R in following fashion:

Lemma 2.12 *Let R be a relation on $\mathcal{P} \times \mathcal{O}$, then for any $X \subseteq \mathcal{P}$:*

$$f_{N^2}(X) = 1 \iff X = \bigcap \{N(y) : y \in \mathcal{O}, X \subseteq N(y)\}$$

We now characterize the prime implicates of f_C as a subset of H_C.

Definition 2.13 *Let C be a closure operator on U, we denote by $\mathscr{I}_C$ the family of subsets X of U such that:*

1. *$X \neq C(X)$,*
2. *for any proper subset Y of X, $C(Y) \neq C(X)$.*

Note that, by Item 1 of definition 2.13, $\mathscr{I}_C \cap \mathcal{F}_C = \emptyset$ and that, by Item 2, each X in $\mathscr{I}_C$ is a minimal element of $\{Y \subseteq U : C(Y) = C(X)\}$. Thus a subset $A \subset U$ is closed iff for any clause $A \cup X \rightarrow \alpha \in H_C(\mathscr{I}_C)$ we have $\alpha \subseteq A$.

Theorem 2.14 *Let C be a closure operator on U, then*

- *$H_C(\mathscr{I}_C)$ is a Horn representation of f_C;*
- *$H_C(\mathscr{I}_C)$ is the set of prime implicates of f_C.*

Example 2.15 *With closure system $\mathcal{F}_C$ of Example 2.6:*

$$\mathscr{I}_C = \{\{e\}, \{a,d\}, \{a,e\}, \{b,c\}, \{b,d\}, \{b,e\}, \{c,d\}, \{c,e\}, \{a,b,c\}\}$$

$$H_C(\mathscr{I}_C) = \{\{a,d\} \rightarrow, \{a,e\} \rightarrow, \{c,e\} \rightarrow, \{b,e\} \rightarrow, \{a,b,c\} \rightarrow, \{e\} \rightarrow d, \{b,c\} \rightarrow d, \{b,d\} \rightarrow c, \{c,d\} \rightarrow b\}$$

$$f_C = (\neg a \vee \neg d) \wedge (\neg a \vee \neg e) \wedge (\neg c \vee \neg e) \wedge (\neg b \vee \neg e) \wedge (\neg a \vee \neg b \vee \neg c) \wedge (\neg e \vee d) \wedge (\neg b \vee \neg c \vee d) \wedge (\neg b \vee c \vee \neg d) \wedge (b \vee \neg c \vee \neg d)$$

We will now translate the domination relations into logical form.

Definition 2.16 *Let C be a closure operator on U, $\mathcal{X} \subseteq \wp(U) - \mathcal{F}_C$, $A \subseteq U$, and $(x, y) \subseteq (U - A)^2$. We will define $K_C(\mathcal{X}, A, x, y)$ as the set of clauses: $H_C(\mathcal{X}) \cup \{\rightarrow a : a \in A\} \cup \{\rightarrow x, y \rightarrow\}$.*

Theorem 2.17 *Let C be a closure operator on U, A a closed set, and $(x,y) \in (U - A)^2$. Then $(x, y) \in dom_C(A)$ iff the following set of clauses is unsatisfiable:*

$$K_C(\mathscr{I}_C, A, x, y)$$

In Theorem 2.17, the set of clauses $H_C(\mathscr{I}_C)$ used by K_C can be replaced by any Horn representation H of f_C. Then for any closed set A, $dom_C(A)$ can be computed in $O(|H|.|U - A|^2)$ time.

Definition 2.18 *Let H be a set of clauses. We will denote by $\mathrm{ABS}(H)$ the minimal equivalent set of clauses obtained from H by removing clauses by absorption (i.e. $A \rightarrow x$ absorbes $A \cup B \rightarrow x$).*

Example 2.19 *Let $\mathcal{F}_C$ be the closure system defined in Example 2.6 then*

$$\mathrm{ABS}(H_C(\mathscr{I}_C) \cup \{\rightarrow d\}) = \{\{a,d\} \rightarrow, \{a,e\} \rightarrow, \{c,e\} \rightarrow, \{b,e\} \rightarrow, \{a,b,c\} \rightarrow, \{b,d\} \rightarrow c, \{c,d\} \rightarrow b, \rightarrow d\}$$

It follows that:

- *$dom_C(\{d\}) = \{(b,c), (c,b), (a,b), (a,c), (a,e)\}$*
- *$Dom_C(\{d\}) = \{(\{a\}, \{b,c\}), (\{a\}, \{e\})\}$*
- *Cover of $\{d\}$: $\{\{b,c,d\}, \{d,e\}\}$*

3 Closure systems associated with rule generation

One of the most crucial problems in Data Mining using Formal Concept Analysis is rule extraction. In Example 2.2, e will imply d, because there is no concept where e appears without d. Finding these rules, called exact association rules, is of major importance in practise, and clearly there are a great number of them.

Work by Guigues and Duquenne ([GD 86]) and by Ganter ([Gan 84]) show that the set of such rules can be represented by a basis of rules, from which all other rules can be easily inferred, a process which can drastically reduce the number of rules which need to be computed and memorized. However, existing algorithms for computing this basis are not very efficient, and new algorithmic tools need to be investigated.

In relation to the work in this paper, existing rule generation algorithms are based on the definition of two closure systems, corresponding to so-called 'pseudo-closed sets' and 'quasi-closed sets' associated with the initial closure system corresponding to concepts.

In this section, we will investigate these two other closure systems, and in particular we will accordingly transpose Theorem 2.17.

3.1 Dependancy relations and basis

Any closure system is associated with a dependency relation corresponding to the set of association rules. Generators and basis can thus be used in the context of closure systems.

Definition 3.1 *A binary relation D on $\wp(U)$ is said to be a* ***dependency relation*** *if it verifies the following properties for all $Y_1, Y_2, Y_3 \subseteq U$:*

C1) *D is transitive,*

C2) *if $Y_2 \subseteq Y_1$ then $(Y_1, Y_2) \in D$,*

C3) *if $(Y_1, Y_2) \in D$ then $(Y_1 \cup Y_3, Y_2 \cup Y_3) \in D$.*

Note that conditions C1) and C3) imply that if $(Y_1, Y_2) \in D$ and $(Y_3, Y_4) \in D$ then $(Y_1 \cup Y_3, Y_2 \cup Y_4) \in D$.

Definition 3.2 *If R is a relation on $\wp(U)$, we will denote by R^+ the minimal relation on $\wp(U)$ including R which is a dependency relation.*

Let D be a dependency relation on $\wp(U)$. A sub-relation $R \subseteq D$ is said to be a ***generator*** *of D iff $R^+ = D$.*
If there exists some proper sub-relation S of R such that $S^+ = D$ then R is said to be ***redundant****, otherwize R is called a* ***basis*** *of D.*

Definition 3.3 *Let C be a closure operator U. We define a binary relation $\rightarrow_C$ on $\wp(U)^2$ by one of the following equivalent conditions[1] for $X, Y \subseteq U$:*

$X \rightarrow_C Y$

$$\iff (\forall Z \subseteq U)\ (X \subseteq C(Z) \Rightarrow Y \subseteq C(Z)) \quad (1)$$

$$\iff C(Y) \subseteq C(X) \quad (2)$$

$$\iff Y \subseteq C(X) \quad (3)$$

A pair of subsets X, Y such that $X \rightarrow_C Y$ is called an ***exact association rule****. $\rightarrow_C$ will be called an* ***association relation****.*

Property 3.4 *For any closure operator C on U, $\rightarrow_C$ is a dependency relation on $\wp(U)$.*

From a formal point of view, $X \rightarrow_C Y$ denotes a pair of sets, while $X \rightarrow Y$ denotes a set of propositional clauses. However, we will see that one holds iff the other holds.

We can now define generators and basis for an association relation:

Definition 3.5 *Let C be a closure operator on U, let $\mathcal{X} \subseteq \wp(U) - \mathcal{F}_C$ be a family of non-closed sets. We will denote by $R_C(\mathcal{X})$ the relation $\{(X, C(X)) : X \in \mathcal{X}\} \subseteq \wp(U)^2$.*
We will say that $\mathcal{X}$ is a ***generator*** *of $\rightarrow_C$ if $R_C(\mathcal{X})^+ = \rightarrow_C$. If in addition $R_C(\mathcal{X})$ is minimal, then $\mathcal{X}$ is called a* ***basis*** *of $\rightarrow_C$.*

As it has been pointed out in [GD 86], $\mathscr{J}_C$ is a generator of $\rightarrow_C$.

Definition 3.6 *Let C be a closure operator on U. Then a subset $X \in \wp(U) - \mathcal{F}_C$ is said to be* ***quasi-closed*** *iff for any $Y \in \mathcal{F}_C$, $X \cap Y \in \mathcal{F}_C \cup \{X\}$. We will denote by $\mathscr{Q}_C$ the family of quasi-closed sets.*

Because of Definition 2.4, for any quasi-closed set X, $\mathcal{F}_C \cup \{X\}$ is a closure system. This leads to the following theorem, proved in [GD 86, BD 98].

[1] As usual, we denote $(X, Y) \in \rightarrow_C$ by the infix notation $X \rightarrow_C Y$.

Theorem 3.7 *Let C be a closure operator on U, then:*

1. *$\mathscr{Q}_C$ is a generator of $\rightarrow_C$.*
2. *$\mathcal{F}_C \cup \mathscr{Q}_C$ is a closure system.*

[GD 86] showed that all the basis of $\rightarrow_C$ have the same cardinal; moreover, they defined a unique canonical basis by using the closure system which we will now describe.

Definition 3.8 *Let C be a closure operator on U, the family $\mathscr{B}_C$ of pseudo-closed sets of C is defined by:*

$B \in \mathscr{B}_C$ iff

$C(B) \neq B$ and $(\forall A \in \mathscr{B}_C)\ A \subset B \Rightarrow C(A) \subseteq B$

Theorem 3.9 *([GD 86])*
Let C be a closure operator on a finite set U, then:

1. *$\mathscr{B}_C$ is a basis of $\rightarrow_C$.*
2. *$\mathscr{B}_C \subseteq \mathscr{Q}_C$.*
3. *$\mathcal{F}_C \cup \mathscr{B}_C$ is a closure system.*

In the rest of this work, we will call family $\mathscr{B}_C$ the **canonical basis** of $\rightarrow_C$; we will denote by Q_C the closure operator associated with the closure system $\mathcal{F}_C \cup \mathscr{Q}_C$, and by B_C the closure operator associated with $\mathcal{F}_C \cup \mathscr{B}_C$.

As we have generalized domination to any closure system, there will be a domination for closure system $\mathcal{F}_{B_c} = \mathscr{B}_C \cup \mathcal{F}_C$ and a domination for closure system $\mathcal{F}_{Q_c} = \mathscr{Q}_C \cup \mathcal{F}_C$. Characterization 2.9 can thus be applied to generating the closed sets of B_c and Q_c.

3.2 Canonical basis and Horn minimal representation

We will now give a logical representation of the canonical basis, and correspondingly express the domination relations associated with $\mathcal{F}_{B_c}$ and $\mathcal{F}_{Q_c}$.

Lemma 3.10 *Let C be a closure operator; $\mathcal{X}$ is a generator of $\rightarrow_C$ iff $H_C(\mathcal{X})$ is equivalent to H_C.*

It follows that we have $X \rightarrow_C Y$ iff all the clauses in $X \rightarrow Y$ are consequences of clauses in $H_C(\mathscr{J}_C)$.

Lemma 3.11 *Let C be a closure operator on U, and $\mathcal{X} \subseteq \wp(U) - \mathcal{F}_C$, then $R_C(\mathcal{X})$ is non redundant iff $H_C(\mathcal{X})$ is irredundant.*

A **minimal Horn representation** of f_C is a set $\mathcal{H}$ of Horn clauses representing f_C such that $|\mathcal{H}|$ is minimal.

Theorem 3.12 *Let $\mathcal{G} \subseteq \wp(U) - \mathcal{F}_C$, $\mathcal{G}$ is the canonical basis of C iff $H(\mathcal{G})$ is a minimal Horn representation of f_C.*

Consequently, we can apply iterative decomposition algorithms in [BCK 98] to $\mathscr{J}_C$ to approximate a basis. These algorithms will keep all the elements of $H_C(\mathscr{J}_C) \cap H_C(\mathscr{B}_C)$ and for every clause c in $H(\mathscr{B}_C)$ they will keep at least one clause d such that c is a logical consequence of d.

However, the following property is a consequence of Theorem 3.12 and of Theorem 1 from [BC 94].

Property 3.13 *Let C be a closure operator on a finite set U. Given the generator $\mathscr{J}_C$, the problem of finding a basis of C is NP-complete.*

We will now translate dominations for closure B_c into logical form, as we did in Theorem 2.17 for closure C.

Let C be a closure operator on U, A a closed set, $(x, y) \in (U - A)^2$ and $[\mathscr{B}_C]_{A,x}$ the following subset of $\mathscr{B}_C$:

$$[\mathscr{B}_C]_{A,x} = \{X \in \mathscr{B}_C : |X| < |B_c(A \cup \{x\})|\}$$

By Definition 3.8, $(x, y) \in dom_{B_c}(A)$ iff the following set of clauses is unsatisfiable:

$$K_C([\mathscr{B}_C]_{A,x}\,,\ A,\ x, y)$$

Another approach to finding the canonical basis, is to generate the quasi-closed sets following the method in [GD 86]. Moreover, in the context of propositional Horn clauses, the relationship between the representation of f_C based on quasi-sets, and the one based on pseudo-sets is quite simple since we have:

$$H_C(\mathscr{B}_C) = \mathrm{ABS}(H_C(\mathscr{Q}_C))$$

Domination for Q_c can be computed for any $A \in \mathcal{F}_C$ using $\mathscr{J}_C$ as we will do in Theorem 3.15, which is a consequence of the following lemma from [GD 86].

Lemma 3.14 *([GD 86]) $X \in \mathscr{Q}_C$ iff for any $Y \subset X$, $C(Y) \neq C(X) \Rightarrow C(Y) \subseteq X$*

Theorem 3.15 *Let C be a closure operator on U, A a closed set, and $(x,y) \in (U-A)^2$. Then $(x,y) \in dom_{Q_c}(A)$ iff the following set of clauses is unsatisfiable:*

$$K_C(\mathscr{J}_C(A \cup \{x\}),\, A,\, x, y)$$

where $\mathscr{J}_C(A \cup \{x\})$ denotes the subset of $\mathscr{J}_C$ defined by: $\mathscr{J}_C(A \cup \{x\}) = \{J \in \mathscr{J}_C : C(J) \neq C(A \cup \{x\})\}$.

Example 3.16 *Consider again the closure system defined in Example 2.6, then:*

$$\begin{aligned} \mathscr{J}_C(\{a,d,e\}) &= \{\{e\},\{b,c\},\{b,d\},\{c,d\}\} \\ &= \mathscr{J}_C(\{b,d,e\}) = \mathscr{J}_C(\{c,d,e\}). \end{aligned}$$

It follows that for $x \in \{a,b,c\}$:

$$\begin{aligned} \mathrm{ABS}(H_C(\mathscr{J}_C(\{x,d,e\}) \cup \{\rightarrow d, \rightarrow e\}) \\ = \{bd \rightarrow c, cd \rightarrow b, \rightarrow d, \rightarrow e\}) \end{aligned}$$

and $\mathrm{dom}_{Q_c}(\{d,e\}) = \{(b,c),(b,d)\}$.

Therefore, the elements of $\mathcal{F}_{Q_c}$ that cover $\{d,e\}$ are $\{a,d,e\}$ and $\{b,c,d,e\}$. Among them, only the last one is in $\mathscr{B}_C$.

4 Algorithmic aspects

As discussed above, the set of concepts covering a given concept A can be computed without any other input than relation R and concept A. This, however, fails to be the case when computing a basis of exact association rules, as the set of already computed rules is scanned to decide whether a candidate part of $\mathcal{U}$ belongs to the basis.

We have seen that the notion of domination not only generalizes well, but that the purely local domination relation used in generating the concepts is in fact closely related to the domination relation used in generating the canonical basis.

We present a new algorithmic process, which uses domination to generate rules, requiring only information inherited by each concept from its direct concept predecessors. The general idea is that, when going up from the bottom to the top of the lattice, whenever a new domination appears, it corresponds to a rule.

Let us again use the binary relation from Example 2.2. In order to simplify notations, we will denote rules of the canonical basis by $A \rightarrow_C B$, where $C(A) = A \cup B$ instead of $A \rightarrow_C C(A)$. Associated canonical basis $R_C(\mathscr{B}_C)$: $\{ad \rightarrow_C bce,\ bc \rightarrow_C d,\ bd \rightarrow_C c,\ cd \rightarrow_C b,\ e \rightarrow_C d,\ bcde \rightarrow_C a\}$.
The corresponding lattice $\mathscr{L}_2$ of $\mathcal{F}_C \cup \mathscr{B}_C$ is shown in Figure 2.

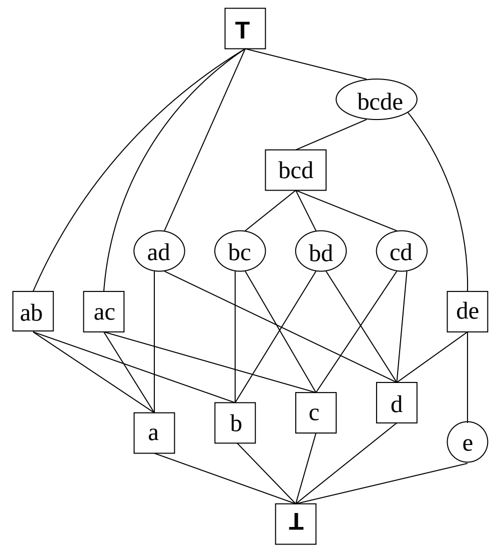

Figure 2: Lattice $\mathscr{L}_2$ associated with relation R of Example 2.2.

The concepts are represented by rectangles and labelled by their intents; the pseudo-concepts are represented by circles.

We will now briefly describe our algorithmic process, which, by a breadth-first type traversal starting with the bottom element, recursively both generates the concept cover of a given concept A, and generates the pseudo-concepts which cover A. Our algorithm correctly generates all concepts, as well as the premisses of a generator of rules, which contains all pseudo-concepts; moreover, no element of the type $A \cup \{x\}$, where A is a concept and x is an element of U, is generated which is not a pseudo-concept; furthermore, few rules which are not pseudo-concepts are generated.

We will illustrate our process on the sample lattice above, where only pseudo-concepts are generated.

We will denote by $R(A)$ sub-relation $R(\mathcal{P} -$

$A, N(A))$, and domination in $R(A)$ by $\geq$:

$$x \geq y \text{ iff } (x,y) \in dom_C(A)$$

where C in general could be any closure operator, and in our example is N^2. At each step of the algorithm, processing concept A, we compute:

- The domination relation D of $R(A)$, the partition P_1 of $R(A)$ into maxmods, the non-dominating maxmods of $R(A)$, and deduce the concept cover of A.

- Some dominations, which correspond to rules generated by some ancestor of A, are inherited, and they define a different partition P_2 into maxmods, with more classes. Domination relation D is translated using the maxmods of P_2, and each resulting domination is then examined.

  Each domination thus obtained can be written, in a general fashion, as $X \geq Y_1Y_2...Y_k$, where $X, Y_1, ...Y_k$ are maxmods of P_2.

  The rule analysis step puts domination $X \geq Y_1Y_2...Y_k$ into one of the three following categories:

  1. Not new: all $X \geq Y_i$, with $i \in [1,k]$, are inherited. In this case, no rule is generated.
  2. New and pure: no $X \geq Y_i$ is inherited. In this case, rule $A + X \rightarrow_C Y_1Y_2...Y_k$ is generated, and $X \geq Y_1Y_2...Y_k$ is added to the information which A will pass on to the elements which cover it.
  3. New but not pure: some $X \geq Y_i$ are inherited, but not all; in this case, no rule is generated by A, but it is possible to predict whether or not a corresponding rule will be generated in some descendant of A. This is an important feature, because it helps to correctly decide whether to generate a rule which is not of the simple type $A \cup \{x\}$ described above, but because of space restrictions we will not go into corresponding details here.

Example 4.1 *Execution on the relation in 2.2:*

- ***Step 1, processing the bottom element:*** *Partition into maxmods:* $\{a\}|\{b\}|\{c\}|\{d\}|\{e\}$.
  Local domination in R: $e \geq d$.
  Non-dominating maxmods: $\{a\}, \{b\}, \{c\}, \{d\}$.
  Concept cover of bottom: $\{a, b, c, d\}$.
  Inherited domination: $\emptyset$.
  Rule analysis:
  $e \geq d$ *is new and pure: rule* $e \rightarrow_C d$ *is generated.*
  Domination information $\{e \geq d\}$ *is passed on to elements of concept-cover.*

- ***Step 2, processing concept*** a*: Partition into maxmods:* $\{b\}|\{c\}|\{de\}$. *Local domination in* $R(a)$*:* $d \geq bce, e \geq bcd$. *Non-dominating maxmods:* $\{b\}, \{c\}$.
  Concept cover of a: $\{ab, ac\}$.
  Inherited domination: $\{e \geq d\}$.
  Rule analysis:
  $d \geq bce$ *is new and pure: rule* $ad \rightarrow_C bce$ *is generated.*
  $e \geq bcd$ *is new but not pure, as* $\{e \geq d\}$ *is inherited.*
  Let us examine whether we can expect a new domination for e higher up: $\mathcal{C}(ad) = \mathcal{U}$*, which will not correspond to a rule.*
  Domination information $\{d = e, de \geq bc\}$ *is passed on to elements of concept-cover.*

- ***Step 3, processing concept*** b*: Partition into maxmods:* $\{a\}|\{c,d\}|\{e\}$. *Local domination in* $R(b)$*:* $c \geq d, d \geq c, e \geq acd$. *Non-dominating maxmods:* $\{a\}, \{c,d\}$.
  Concept cover of b: $\{ab, bcd\}$.
  Inherited domination: $\{e \geq d\}$.
  Rule analysis:
  $c \geq d$ *is new and pure: rule* $bc \rightarrow_C d$ *is generated.*
  $d \geq c$ *is new and pure: rule* $bd \rightarrow_C c$ *is generated.*
  $e \geq acd$ *is new but not pure, as* $\{e \geq d\}$ *is inherited.*
  Let us examine whether we can expect a new domination for e higher up: $\mathcal{C}(bd) = bcd$*, so we can expect rule* $bcde \rightarrow_C a$*, and* $e \geq a$ *can be used in any descendant to generate rules with premisses of cardinality* ≥ 4.
  Domination information $\{c = d, e \geq d\}$ *is passed on to elements of concept-cover.*

- ***Step 4, processing concept*** c*: Partition into maxmods:* $\{a\}|\{b,d\}|\{e\}$. *Local domination in* $R(c)$*:* $b \geq d, d \geq b, e \geq abd$. *Non-dominating maxmods:* $\{a\}, \{b,d\}$.
  Concept cover of c: $\{ac, bcd\}$.
  Inherited domination: $\{e \geq d\}$.
  Rule analysis:
  $b \geq d$ *is new and pure: rule* $bc \rightarrow_C d$ *is generated.*

$d \geq b$ is new and pure: rule $cd \to_C b$ is generated.
$e \geq abd$ is new but not pure, as $\{e \geq d\}$ is inherited.
Let us examine whether we can expect a new domination for e higher up: $\mathcal{C}(cd) = bcd$, so as in Step 3, we can expect rule $bcde \to_C a$, and $e \geq a$ can be used in any descendant to generate rules with premisses of cardinality ≥ 4.
Domination information $\{b = d, e \geq d\}$ is passed on to elements of concept-cover.

- ***Step 5, processing concept** d: Partition into maxmods: $\{a\}|\{b,c\}|\{e\}$. Local domination in $R(d)$: $a \geq bce, b \geq c, c \geq b$. Non-dominating maxmods: $\{e\}, \{b,c\}$.*
  Concept cover of d: $\{de, bcd\}$.
  Inherited domination: $\emptyset$, as $\{e \geq d\}$ has no meaning in $R(d)$.
  Rule analysis:
  $a \geq bce$ is new and pure: rule $ad \to_C bce$ is generated.
  $b \geq c$ is new and pure: rule $bd \to_C c$ is generated.
  $c \geq b$ is new and pure: rule $cd \to_C b$ is generated.
  Domination information $\{a \geq bce, b = c\}$ is passed on to elements of concept-cover.

- ***Step 6, processing concept** ab: Partition into maxmods: $\{c, d, e\}$. Local domination in $R(ab)$: $c \geq de, d \geq ce, e \geq cd$.*
  Non-dominating maxmod: $\{c, d, e\}$.
  Concept cover of ab: abcde, which is the top element.
  Inherited domination, respectively from a and b: $\{d = e, de \geq bc\} \cup \{c = d, e \geq d\}$, i.e. $c = d = e$, so no rule is generated.

- ***Step 7, processing concept** ac: Partition into maxmods: $\{b, d, e\}$. Local domination in $R(ac)$: $b \geq de, d \geq be, e \geq bd$.*
  Non-dominating maxmod: $\{b, d, e\}$.
  Concept cover of ac: abcde, which is the top element.
  Inherited domination, respectively from a and c: $\{d = e, de \geq bc\} \cup \{b = d, e \geq d\}$, i.e. $b = d = e$, so no rule is generated.

- ***Step 8, processing concept** de: Partition into maxmods: $\{a, b, c\}$. Local domination in $R(de)$: $a \geq bc, b \geq ac, c \geq ab$.*
  Non-dominating maxmod: $\{a, b, c\}$.
  Concept cover of de: abcde, which is the top element.
  Inherited domination, from d: $\{a \geq bce, b = c\}$.
  $a \geq bc$ is not new, as $a \geq bce$ is inherited.
  $bc \geq a$ is new and pure: rule $bcde \to_C a$ is generated.

- ***Step 9, processing concept** bcd: Partition into maxmods: $\{a, e\}$. Local domination in $R(bcd)$: $a \geq e, e \geq a$.*
  Non-dominating maxmod: $\{a, e\}$.
  Concept cover of bcd: abcde, which is the top element.
  Inherited domination, from b, c and d respectively: $\{c = d, e \geq d\} \cup \{b = d, e \geq d\} \cup \{a \geq bce, b = c\}$, i.e. $a \geq e$, since b, c and d have disappeared from $R(bcd)$.
  Rule analysis:
  $a \geq e$ is not new, as it is inherited.
  $e \geq a$ is new and pure: rule $bcde \to_C a$ is generated.

We see that each pseudo-concept has been generated exactly as many times as the number of predecessors it has in $\mathscr{L}_2$, and no other rule has been generated.

5 Conclusion and open questions

In this paper, we use the relationship between concept lattices and domination in graphs to extend already existing graph-oriented the results on concept lattices to a general closure system and to Horn clauses.

Though obviously there remains much work to be done in this direction, our results are interesting not only from a possible algorithmic point of view, but also because they can lead to a better understanding of the canonical basis of rules; moreover, it is important to find new ways of modelizing these results so that a variety of non-specialists can achieve a better grasp on these problems.

As far as open questions are concerned, we still have to improve the prospective algorithm presented in Section 4, both so that it will never yield anything but rules of the basis, but also in terms of data storage and efficiency.

Another question of great current interest is that of generating approximate association rules. As an example, an interesting recent approach by J-M. Bernard, S. Poitrenaud [BP 99] works by first approximating the binary relation according to coherent probabilistic models which must be compatible with logical rules; the logical interpretation we introduce in this paper could be combined with this approach in future work.

References

[BP 99] J-M. BERNARD, S. POITRENAUD, *L'analyse implicative Bayésienne multivariée d'un questionnaire binaire : quasi-implications et treillis de Galois simplifié*, Math. Inf. Sci. hum., n^o 147, 1999, pp. 25-46.

[BS 02] A. BERRY, A. SIGAYRET *Representing a concept lattice by a graph*, Workshop on Discrete Mathematics and Data Mining (DM&DM 2002), Arlington (VA), April 11-13, 2002 ; Proc. 2nd SIAM Conf. on Data Mining, 2002.

[Bor 86] J-P. BORDAT, *Calcul pratique du treillis de Galois d'une correspondance*, Mathématiques, Informatique et Sciences humaines **96**, 1986, pp. 31–47.

[BC 94] E. BOROS, O. ČEPEK, *On the complexity of Horn minimization*, Rutcor Research Report, 1994, 14 p.

[BCK 98] E. BOROS, O. ČEPEK, A. KOGAN, *Horn minimization by iterative decomposition*, Annals of Mathematics and Artificial Intelligence **23**, 1998, pp. 321–343.

[BD 98] G. BUROSCH, J. DEMETROVICS, *The Poset of Closures as a Model of Changing Databases*, Order **4**, 1987, pp. 127–142.

[Fla 76] C. FLAMENT, *L'analyse booléenne de questionnaires*, Paris, Mouton, 1976.

[Gan 84] B. GANTER, *Two basic algorithms in concept analysis*, Preprint 831, Technische Hochschule Darmstad, 1984.

[GD 86] J. L. GUIGUES et V. DUQUENNE, *familles minimales d'implications informatives résultant d'un tableau de données binaires*, Mathématiques et Sciences humaines **95**, 1986, pp. 5–18.

[Jar 99] J. JÄRVINEN, *Knowledge representation and rough sets*, Turku center for computer Science, TUCS Dissertation, N^o 14, March 1999.

[Nov 97] M. NOVOTNÝ, *Dependence Spaces of Information Systems*, in. Incomplete Information : Rough Set Analysis, E. Orlowska (ed.), Physica-Verlag, 1997.

[San 99] E. SANJUAN *On Rough Set Analysis applied to Questionnaires*, STSM Report, COST Action 15, 1999, 16p.

[San 02] E. SANJUAN *Heyting Algebra for Modelling Information Retrieval Based on Thematic Clustering*, Workshop on Discrete Mathematics and Data Mining (DM&DM 2002), Arlington (VA), April 11-13, 2002 ; Proc. 2nd SIAM Conf. on Data Mining, 2002.

[Wil 82] R. WILLE, *Restructuring Lattice Theory: an approach based on hierarchies of concepts*, Ordered Sets Vol. 83, I. Rival (ed.), D. Reidel Publishing Company, 1982, pp. 445–470.