Attribute reduction of concept lattices based on matroidal approach

Hua Mao
(Department of Mathematics, Hebei University, Baoding 071002, China)
e-mail: yushengmao@263.net

Abstract

For a context, a matroid is induced by an equivalent relation which is produced from the above bipartite graph. Each member in one of Zhang's three attribute classes is characterized through the above matroid. After that, this paper searches out the composition of a reduct set. All these results show the potential and merit in using matroidal approaches for designing and studying concept lattice.

Key words attribute reduction; matroid; bipartite graph; concept lattice

1 Introduction

Concept lattice, that is, Formal Concept Analysis (FCA), was first introduced in [1] and has grown to a powerful theory for data analysis, information retrieval and knowledge discovery (cf. [2-4]).

Attribute reduction for a context, one of focus issues in FCA, is very useful to explore the lattice structure of all concepts in a context, because fewer attributes will make the constructing process of the lattice easier. There are many ways for concept lattice reduction and attribute reduction (see [3-8]). Searching new methods for attribute reduction are an aspiration for researchers.

Matroids were first proposed in [9]. It has been found that matroids are effective and useful in concept lattice theory (see [9-13]), and also used as a research method in knowledge reduction of information systems (see [14,15]). We hope to search out the methods on attribute reduction in the matroid framework.

How can we fulfill the duty provided above? We find that with bipartite graph, Abello et al [16] deal with some properties in FCA. This hints that bipartite graph may be an successful assistant for our completing the duty.

The main goal in this paper is to characterize Zhang's three classes of attributes. We will use bipartite graph model to search our matroidal approaches.

The concrete process in this paper is that with the established bipartite graph for a context in [16], we present the construction of a matroid which is able to determine the classes of attributes. After that, we search out the consistency of all the attribute reducts.

The rest of the paper is organized as follows. Section 2 reviews some fundamental notations relative to concept lattices, matroids and graphs. Section 3 characterizes Zhang's three classes of attributes and explores all the attribute reducts for a context.

We declare that in this paper, the basic facts of lattice theory and poset theory are as discussed in [17]. Throughout, $MinA$ denotes the set of minimal elements in a poset A; $L_1 \cong L_2$ if the same two mathematical structures L_1 and L_2, e.g. two lattices L_1 and L_2, are isomorphic.

2 Preliminaries

This section introduces some basic facts of concept lattices, matroids and graphs.

2.1 Concept lattices

This subsection gives only a brief overview of the basic facts for concept lattices. For a more detailed description, please refer to [3].

Definition 2.1.1 (1) [3, p.17] A *context* (O, P, I) consists of two sets O and P and a relation I between O and P. The elements of O (of P) are called the *objects* (*attributes*) of (O, P, I) and I is called the *incidence relation* of (O, P, I). We write oIp or $(o, p) \in I$.

(2) [3, p.18] For $A \subseteq O$ and $B \subseteq P$, we define $A' := \{p \in P \mid oIp \text{ for all } o \in A\}$ and $B' := \{o \in O \mid oIp \text{ for all } p \in B\}$. A *concept* of (O, P, I) is a pair (A, B) with $A \subseteq O$, $B \subseteq P$, $A' = B$ and $B' = A$. We call A the *extent* (B the *intent*) of the concept (A, B).

(3) [3, pp.19-20] Let (A_1, B_1) and (A_2, B_2) be two concepts of (O, P, I). If $A_1 \subseteq A_2$ (which is equivalent to $B_2 \subseteq B_1$), then we write $(A_1, B_1) \leq (A_2, B_2)$. The relation $\leq$ is called the *hierarchical order* of the concepts. The set of all the concepts of (O, P, I) ordered in this way is denoted by $\mathcal{B}(O, P, I)$ and is called the *concept lattice* of (O, P, I).

In this paper, (O, P, I) denotes a context. The following statements are for (O, P, I).

(2.1.1) Let $\mathcal{B}_O(O, P, I) = \{X \mid (X, B) \in \mathcal{B}(O, P, I) \text{ for some } B \subseteq P\}$. The authors [3] indicate that if $\mathcal{B}_O(O, P, I)$ has the same hierarchical order as $\mathcal{B}(O, P, I)$, then there is $\mathcal{B}(O, P, I) \cong \mathcal{B}_O(O, P, I)$.

We still denote as $\mathcal{B}_O(O, P, I)$ if $\mathcal{B}_O(O, P, I)$ owns the same hierarchical order as $\mathcal{B}(O, P, I)$.

(2.1.2) According to the discussions in [3, p.24], this paper does not consider the context with full rows and full columns.

Let $a \in O$ and $b \in P$ satisfy $(a, y) \notin I$ and $(x, b) \notin I$ for any $y \in P$ and $x \in O$. We will not consider the above object a and attribute b.

(2.1.3) Considering (2.1.1) and (2.1.2), we may state that in this paper, $(\emptyset, P)$ and $(O, \emptyset)$ are the minimum and the maximum in $\mathcal{B}(O, P, I)$ respectively.

In [8], Zhang provides a description for attribute reduction for a context (O, P, I).

Definition 2.1.2 [8] (1) If there exists $D \subseteq P$ such that $\mathcal{B}(O, P, I) \cong \mathcal{B}(O, D, I_D)$, then D is called a *consistent* set of (O, P, I). If D is a consistent set and no proper subset of D is consistent, then D is referred to as an *attribute reduct* of (O, P, I), where $I_D = I \cap (O \times D)$.

(2) Let B_k be an attribute reduct of $(O, P, I), (k \in \mathbf{I}; \mathbf{I}$ is an index set which includes all its reducts). The attributes are classified into the following three types:

(i) *absolute necessary attribute* $b : b \in \bigcap\limits_{k \in \mathbf{I}} B_k$.

(ii) *relative necessary attribute* $c : c \in \bigcup\limits_{k \in \mathbf{I}} B_k \setminus \bigcap\limits_{k \in \mathbf{I}} B_k$.

(iii) *absolute unnecessary attribute* $d : d \in P \setminus \bigcup\limits_{k \in \mathbf{I}} B_k$.

Additionally, Zhang presents the following properties for his attribute reduction.

Lemma 2.1.1 [8] Let $G_a = \{g \mid g \in P, a' \subset g'\}$ for any $a \in P$. Then,

(1) a is an absolute necessary attribute $\Leftrightarrow (a'' \setminus \{a\})' \neq a'$.

(2) a is a relative necessary attribute $\Leftrightarrow (a'' \setminus \{a\})' = a'$ and $G_a' \neq a'$.

(3) a is an absolute unnecessary attribute $\Leftrightarrow (a'' \setminus \{a\})' = a'$ and $G_a' = a'$.

By dual, we can obtain object reduction from the correspondent results of attribute reduction. Hence, we need only to focus our attention on attribute reduction.

2.2 Matroids

Matroids will aid us in our discussions in this paper. Hence, in this subsection, we will recall notations and properties relative to matroids, and for more detail, we refer to [18].

Definition 2.2.1 [18, p.7] A *matroid* M is a finite set S and a collection $\mathcal{I}$ of subsets of S (called *independent sets*) such that (i1)-(i3) are satisfied.

(i1) $\emptyset \in \mathcal{I}$;

(i2) If $X \in \mathcal{I}$ and $Y \subseteq X$, then $Y \in \mathcal{I}$;

(i3) If $X, Y \in \mathcal{I}$ with $|X| = |Y| + 1$, then there exists $x \in X \setminus Y$ such that $Y \cup x \in \mathcal{I}$.

A *base* of M is a maximal independent set.

Lemma 2.2.1 [18, pp.121-123] Let $M_1, \ldots, M_n$ be matroids on S. Let $\mathcal{I} = \{X \mid X = X_1 \cup \ldots \cup X_n; X_i \in \mathcal{I}(M_i), (1 \leq i \leq n)\}$. Then $\mathcal{I}$ is the collection of independent sets of a matroid $M_1 \vee \ldots \vee M_n$ on S.

2.3 Graphs

In this paper, for the definitions of *bipartite graph* and *complete bipartite graph*, please see [19, p.4]; for the definition of *subgraph* and *induced subgraph*, please refer to [19, p.9]; for the definition of *neighbour set*, please see [19, p.72].

We only review and discuss bits of notations and terminologies of graph theory. For more detail of graph theory, please refer to [19].

Some notations 2.3.1 Let G be a graph.

(1) The set of edges in G is denoted as $E(G)$; the set of vertices is in notation $V(G)$.

(2) If $V(G) = \emptyset$, then it is in notation $G = \emptyset$.

(3) $G[V^*]$ is an induced subgraph of G where $V^* \subseteq V(G)$ and $V^* \neq \emptyset$.

(4) The neighbor set of $x \in V(G)$ is in notation $N_G(x)$.

Sometimes, if it does not follow a confusion from the text, we denote $N_G(x)$ as $N(x)$.

Let $S \subseteq V(G)$. $N_G(S)$, simply $N(S)$, is $\{y \in V(G) \mid y \in N(x) \text{ for every } x \in S\}$.

(5) If G is simple and $e \in E(G)$ with u and v as its two connected vertices, then e is sometimes in notation uv.

From graph theory, we easily gain the following statements.

Lemma 2.3.1 Let G be a bipartite graph with $V(G) = X \cup Y$ satisfying $X \cap Y = \emptyset$. Then for $A, B \subseteq X$ (or $A, B \subseteq Y$), there are the following statements.

(1) $N(A) = \bigcap\limits_{a \in A} N(a)$. (2) $N(A \cup B) = N(A) \cap N(B)$. (3) $N(A) \subseteq N(B)$ if $B \subseteq A$.

To reformulate some results and search out some properties on FCA in matroid frameworks, we introduce a graph construction and obtain some properties on this construction.

Definition 2.3.1 [16] $G_{(O,P,I)}$, a *bipartite graph inducing from a context* (O, P, I), is $(O \cup P, \{(o, p) \mid oIp\})$, i.e. $V(G_{(O,P,I)}) = O \cup P$ and $E(G_{(O,P,I)}) = \{(o, p) \mid oIp\}$.

Lemma 2.3.2 $G_{(O,P,I)}$ has the following properties.

(1) $G_{(O,P,I)}$ is simple.

(2) $X' = N(X)$ for any $X \subseteq O$ (or $X \subseteq P$).

(3) If $X \subseteq O$ and $Y \subseteq P$, then $X = \emptyset \Rightarrow N(X) = P$; $Y = \emptyset \Rightarrow N(Y) = O$;

$X = O \Rightarrow N(X) = \emptyset$; $Y = P \Rightarrow N(Y) = \emptyset$.

Especially, if X is an extent and Y is an intent, then

$$N(X) = P \Rightarrow X = \emptyset;\ N(Y) = O \Rightarrow Y = \emptyset.$$

Proof (1)-(3) are straightforward from Definition 2.1.1 and Definition 2.3.1 with (2.1.3).

In fact, $X \subset O \Leftrightarrow N(X) \subset P$ is true for every contexts considered in this paper according to Subsection 2.1 and Definition 2.3.1.

Corollary 2.3.1 $(X, Y) \in \mathcal{B}(O, P, I) \Longleftrightarrow X = N(Y)$ and $Y = N(X)$ in $G_{(O,P,I)}$.
Proof Routine verification from Definition 2.1.1 and Lemma 2.3.2.

For clarity of exposition, in what follows, $G_{(O,P,I)}$ is sometimes in notation G if we find no confusion from text.

3 Attribute reduction

Assisted by the inducing bipartite graph of a context in [16], this section, with matroid approaches, discusses reducing attributes. Generally, in order to facilitate the discussion for attribute reduction, some authors utilize the expression of Ganter's (cf. [3, p.24]) and some people use the description of Zhang's (cf. Definition 2.1.2) though Ganter's is the original. We will deal with attribute reduction according to Zhang.

Lemma 3.1 Put $a \in P$. Let $[a] = \{x \in P \mid N(a) = N(x)\}$, $\mathcal{I}^a = \{\{x\} \mid x \in [a]\} \cup \{\emptyset\}$ and $F_a = \{g \in P \mid N(a) \subset N(g)\}$. Then

(1) $[a]$ is an equivalent class on P.

(2) $(P, \mathcal{I}^a)$ is a matroid with $\mathfrak{B}^a = \{\{x\} \mid x \in [a]\}$ as its family of bases and $|\mathfrak{B}^a| \geq 1$.

(3) $a'' = F_a \cup [a]$.

Proof The item (1) is evident. The item (2) is routine from Definition 2.2.1 and $\{a\} \in \mathfrak{B}^a$. The item (3) is straightforward from Definition 2.1.1 and Lemma 2.3.2.

Let $P = \{a_0, a_1, a_2, \ldots, a_n\}$.
Algorithm 1 To obtain $[a_0]$.
Input $N(a_0), N(a_1), N(a_2), \ldots, N(a_n)$
Output $[a_0]$
Step 1. $j = 0, [a_0] = \{a_0\}$.
Step 2. If $j < n$, then $j = j + 1$, go to Step 3; otherwise, stop.
Step 3. If $N(a_0) = N(a_j)$, then $[a_0] = [a_0] \cup \{a_j\}$, go to Step 2;
otherwise, $[a_0] = [a_0]$, go to Step 2.

Let $a \in P$. By the definition of F_a, we obtain $F_a = F_x$ for any $x \in [a]$. Let $P \setminus [a] = \bigcup_{i=1}^{s} [a_i^*]$. It is easily seen that if $a_i^* \in F_a$, then $x^* \in F_a$ for any $x^* \in [a_i^*]$.

Algorithm 2 To obtain F_a

Input $N(a), N(a_i^*), [a_i^*], (i = 1, \ldots, s)$

Output F_a

Step 1. $j = 0, F_a = \emptyset$.

Step 2. If $j < s$, then $j = j + 1$, go to Step 3; otherwise, stop.

Step 3. If $N(a) \subset N(a_j^*)$, then $F_a = F_a \cup [a_j^*]$, go to Step 2;
otherwise, $F_a = F_a$, go to Step 2.

Assisted by matroids, we characterize the elements in each class of Zhang's attributes.

Theorem 3.1 Put $a \in P$. Let $\mathfrak{B}^a, F_a$ and $(P, \mathcal{I}^a)$ be defined as Lemma 3.1.

(1) a is an absolute necessary attribute if and only if $|\mathfrak{B}^a| = 1$ and $N(a) \subset N(F_a)$.

(2) a is an absolute unnecessary attribute if and only if $N(a) = N(F_a)$.

(3) a is a relative necessary attribute if and only if $|\mathfrak{B}^a| > 1$ and $N(a) \subset N(F_a)$.

Proof $N(F_a) = \bigcap_{g \in F_a} N(g) \supseteq N(a)$ holds by Lemma 2.3.1 since $N(a) \subset N(g)$ for any $g \in F_a$. $G'_a = F'_a \supseteq a'$ holds using Lemma 2.3.2(2) where G_a is defined as Lemma 2.1.1.

No matter to suppose $\mathfrak{B}^a \setminus \{a\} = \{\{a_j\} : j = 1, \ldots, t = |\mathfrak{B}^a| - 1\}$. We easily receive $N(a_j) = N(a)$ from Lemma 3.1, $(j = 1, \ldots, t)$, and $(a'' \setminus \{a\})' = (\{a_1, \ldots, a_t\} \cup F_a)' = N(\{a_1, \ldots, a_t\} \cup F_a)$ owing to Lemma 3.1(3). $|\mathfrak{B}^a| \geq 1$ holds by Lemma 3.1(2).

If $|\mathfrak{B}^a| = 1$. Then $(a'' \setminus \{a\})' = N(F_a \cup \emptyset) = N(F_a) = F'_a = G'_a$.

If $|\mathfrak{B}^a| > 1$. Then $t > 1$, and further $(a'' \setminus \{a\})' = (\bigcap_{j=1}^{t} N(a_j)) \cap N(F_a) = N(a) \cap N(F_a) = N(a) = a'$ since Lemma 2.3.1(1) and Lemma 2.3.2(2).

Next to prove item (1).

($\Rightarrow$) We receive $|\mathfrak{B}^a| = 1$ and $N(a) \subset N(F_a)$ since $a' = N(a) \subseteq N(F_a) = (a'' \setminus \{a\})'$, $(a'' \setminus \{a\})' \neq a'$ and the above for $|\mathfrak{B}^a| > 1$.

($\Leftarrow$) It follows $(a'' \setminus \{a\})' \neq a'$ since $(a'' \setminus \{a\})' = N(F_a) \supset N(a) = a'$. Therefore, by Lemma 2.1.1, a is an absolute necessary attribute.

Next to prove item (2).

($\Rightarrow$) By Lemma 2.1.2, $F'_a = a'$ holds since $G'_a = a'$ and $G'_a = F'_a$.

If $|\mathfrak{B}^a| > 1$. Then, using the above, we get $(a'' \setminus \{a\})' = a'$ and $N(F_a) = F'_a = a' = N(a)$.

If $|\mathfrak{B}^a| = 1$. Then, it follows $(a'' \setminus \{a\})' = N(F_a) = F'_a = a' = N(a)$.

($\Leftarrow$) If $|\mathfrak{B}^a| > 1$. Then $(a'' \setminus \{a\})' = a'$. In addition, we obtain $G'_a = a'$ according to $N(F_a) = F'_a$, $N(a) = a'$, $F'_a = G'_a$ and $N(a) = N(F_a)$.

If $|\mathfrak{B}^a| = 1$. Then $(a'' \setminus \{a\})' = N(F_a)$. Thus, we receive $(a'' \setminus \{a\})' = a'$ and $G'_a = a'$ owing to $(a'' \setminus \{a\})' = N(F_a) = F'_a = G'_a, N(F_a) = N(a)$ and $N(a) = a'$.

In one word, a is an absolute unnecessary attribute according to Lemma 2.1.1.

The following is to prove item (3).

($\Rightarrow$) We obtain $|\mathfrak{B}^a| > 1$ by (1), and $N(F_a) \neq N(a)$ by (2). So, $N(F_a) \supset N(a)$ holds.

($\Leftarrow$) Routine verification from Definition 2.1.2(2) and the items of (1) and (2).

Let a be an absolute necessary attribute. Then $[a] = \{a\}$ holds by Theorem 3.1. In addition, according to Definition 2.1.2(2), a belongs to any reduct set. The following will discuss the relative properties for the attributes which are not absolute necessary.

Before our presentation of Lemma 3.2, we give a few notations and a bit of description for a relative and not absolute necessary attribute $a \in P$, and a reduct set B_k containing a.
(1) $N(q), N_{B_k}(s), N_{B_k\setminus\{a\}}(r)$ is the neighbor set of $q \in P, s \in B_k$ and $r \in B_k \setminus \{a\}$ (or $q, s, r \in O$) in the inducing graph $G_{(O,P,I)}, G_{(O,B_k,I_{B_k})}$ and $G_{(O,B_k\setminus\{a\},I_{B_k\setminus\{a\}})}$ respectively, where $I_{B_k} = I \cap (O \times B_k)$ and $I_{B_k\setminus\{a\}} = I \cap (O \times (B_k \setminus \{a\}))$.

$N_{B_k}(T) = \bigcap\limits_{t\in T} N_{B_k}(t)$ and $N_{B_k\setminus\{a\}}(U) = \bigcap\limits_{u\in U} N_{B_k\setminus\{a\}}(u)$ for $T \subseteq B_k$ and $U \subseteq B_k \setminus \{a\}$.
(2) By definitions in Section 2.1, we easily indicate $I_{B_k\setminus\{a\}} = I_{B_k} \cap (O \times (B_k \setminus \{a\}))$; $N_{B_k}(t) = \emptyset$ if $t \in P \setminus B_k$; $N_{B_k}(t) = N(t)$ if $t \in B_k$. Moreover, $N_{B_k}(a) = N(a) = N(x) = N_{B_k}(x)$ holds. Additionally, for $s \in O$ and $t \in B_k$, we obtain $N_{B_k}(s) = \{a\} \cup N_{B_k\setminus\{a\}}(s)$ if $a \in N_{B_k}(s)$; $N_{B_k}(s) = N_{B_k\setminus\{a\}}(s)$ if $a \notin N_{B_k}(s)$; $N_{B_k}(t) = N_{B_k\setminus\{a\}}(t)$ if $t \neq a$.

Lemma 3.2 Let $a \in P$ and $x \in [a] \setminus \{a\}$.

(1) If a is a relative necessary attribute, then x is also.

(2) If a is an absolute unnecessary attribute, then x is also.

(3) If a is not an absolute necessary attribute, then a and x are not in the same reduct set.

Proof (**S1**) By Lemma 3.1, $x \in [a]$ implies $[x] = [a]$, and further, $\mathfrak{B}^a = \mathfrak{B}^x$ and $F_a = F_x$, and so $N(F_a) = N(F_x)$. Thus, since Theorem 3.1, items (1) and (2) are accepted.

Item (3) will be proved by (**S2.1**) and (**S2.2**).

(**S2.1**) Let a be an absolute unnecessary attribute. By Definition 2.1.2 and item (2), a and x will not belong to any reduct set. Hence, a and x are not in the same reduct set.

(**S2.2**) Let a be a relative necessary attribute. Suppose that there is a reduct set B_k containing a and x. The needed result is proved by the following (**S2.2.1**)-(**S2.2.3**).

(**S2.2.1**) To prove: $(X, Y) \in \mathcal{B}(O, B_k, I_{B_k}) \Rightarrow X \in \mathcal{B}(O, B_k \setminus \{a\}, I_{B_k\setminus\{a\}})$.

It is easily seen: $a \in Y \Leftrightarrow x \in Y$. Thus, $a \in Y \Leftrightarrow [a]_{B_k} \subseteq Y$ holds where $[a]_{B_k} = [a] \cap B_k$.

We distinguish two cases to continue the proof.

Case 1. $a \in Y$.

It is easily seen $X = N_{B_k}(Y)$, $N_{B_k}(a) = N_{B_k}(x)$ and $x \in Y$. We obtain $X = N_{B_k}(a) \cap (\bigcap\limits_{y\in Y\setminus\{a\}} N_{B_k}(y)) = N_{B_k}(x) \cap (\bigcap\limits_{y\in(Y\setminus\{a\})\setminus\{x\}} N_{B_k}(y)) = N_{B_k\setminus\{a\}}(x) \cap (\bigcap\limits_{y\in(Y\setminus\{a\})\setminus\{x\}} N_{B_k\setminus\{a\}}(y)) = N_{B_k\setminus\{a\}}(Y \setminus \{a\})$. In addition, it has $Y = \{a\} \cup \{z \in B_k \setminus \{a\} \mid X \subseteq N_{B_k}(z)\} = \{a\} \cup \{z \in B_k \setminus \{a\} \mid X \subseteq N_{B_k\setminus\{a\}}(z)\} = \{a\} \cup N_{B_k\setminus\{a\}}(X) = \{a\} \cup (Y \setminus \{a\})$. This follows $N_{B_k\setminus\{a\}}(X) = Y \setminus \{a\}$. Therefore, by Corollary 2.3.1, $(X, Y \setminus \{a\}) \in \mathcal{B}(O, B_k \setminus \{a\}, I_{B_k\setminus\{a\}})$.

Case 2. $a \notin Y$.

This implies $[a]_{B_k} \cap Y = \emptyset$. Hence, it follows $Y = Y \setminus \{a\}$ and $X = N_{B_k}(Y) = \bigcap_{y \in Y} N_{B_k}(y) = \bigcap_{y \in Y \setminus [a]_{B_k}} N_{B_k}(y) = \bigcap_{y \in Y \setminus [a]_{B_k}} N_{B_k \setminus \{a\}}(y) = N_{B_k \setminus \{a\}}(Y \setminus \{a\})$, and further, $Y \setminus \{a\} = Y = N_{B_k}(X) = \{z \in B_k \mid X \subseteq N_{B_k}(z)\} = \{z \in B_k \setminus \{a\} \mid X \subseteq N_{B_k \setminus \{a\}}(z)\} = N_{B_k \setminus \{a\}}(X)$. Thus, we obtain $(X, Y) \in \mathcal{B}(O, B_k \setminus \{a\}, I_{B_k \setminus \{a\}})$ by Corollary 2.3.1.

(**S2.2.2**) To prove: $(A, B) \in \mathcal{B}(O, B_k \setminus \{a\}, I_{B_k \setminus \{a\}}) \Rightarrow A \in \mathcal{B}_O(O, B_k, I_{B_k})$.

Combining Corollary 2.3.1, we infer to $B \subseteq B_k \setminus \{a\}$, and so $A = \bigcap_{b \in B} N_{B_k \setminus \{a\}}(b) = \bigcap_{b \in B} N_{B_k}(b)$. We continue the discussion by the following Cases 3 and 4.

Case 3. $x \in B$.

$A = N_{B_k}(x) \cap (\bigcap_{b \in B \setminus \{x\}} N_{B_k}(b))$ describes $A \subseteq N_{B_k}(x)$. However, it is easily seen $N_{B_k}(x) = N(x) = N(a) = N_{B_k}(a)$. Thus, $A \subseteq N_{B_k}(a)$ holds. Hence we are assured: $A = N_{B_k}(a) \cap A = N_{B_k}(a) \cap (\bigcap_{b \in B} N_{B_k}(b)) = \bigcap_{b \in B \cup \{a\}} N_{B_k}(b) = N_{B_k}(B \cup \{a\})$. $B \subseteq B_k \setminus \{a\}$ follows $B = N_{B_k \setminus \{a\}}(A) = \{b \in B_k \setminus \{a\} \mid A \subseteq N_{B_k \setminus \{a\}}(b)\} = \{b \in B_k \setminus \{a\} \mid A \subseteq N_{B_k}(b)\}$. Thus, $N_{B_k}(A) = \{t \in B_k \mid A \subseteq N_{B_k}(t)\} = \{a\} \cup \{t \in B_k \setminus \{a\} \mid A \subseteq N_{B_k}(t)\} = \{a\} \cup B$. Therefore, $(A, B \cup \{a\}) \in \mathcal{B}(O, B_k, I_{B_k})$ holds by Corollary 2.3.1.

Case 4. $x \notin B$.

By $x \notin B$ and $B = N_{B_k \setminus \{a\}}(A) = \{b \in B_k \setminus \{a\} \mid A \subseteq N_{B_k \setminus \{a\}}(b)\}$, we receive $A \nsubseteq N_{B_k \setminus \{a\}}(x)$. However, because $N_{B_k}(x) = N_{B_k \setminus \{a\}}(x)$ and $N_{B_k}(t) = N_{B_k}(x)$ for any $t \in [a]_{B_k}$, we obtain $A \nsubseteq N_{B_k}(x)$ and $A \nsubseteq N_{B_k}(t)$ for any $t \in [a]_{B_k}$, especially, $A \nsubseteq N_{B_k}(a)$ since $a \in [a]_{B_k}$. This implies $A = \bigcap_{b \in B} N_{B_k \setminus \{a\}}(b) = \bigcap_{b \in B} N_{B_k}(b) = N_{B_k}(B)$ and $B = \{b \in B_k \mid A \subseteq N_{B_k}(b)\} = N_{B_k}(A)$. Thus, $(A, B) \in \mathcal{B}(O, B_k, I_{B_k})$ is followed.

(**S2.2.3**) Taking (**S2.2.1**) and (**S2.2.2**) with (2.1.1), we say that B_k is not a reduct, a contradiction. That is to say, a and x are not in the same reduct set.

Using Lemma 3.1 with Algorithm 1, for any $x \in P$, we can obtain $[x]$, and further $\mathfrak{B}^x$. Thus, we receive $\{[a_i] \mid i = 1, \ldots, m\}$ and $\{\mathfrak{B}^{a_i} \mid i = 1, \ldots, m\}$ such that $[a_p] \cap [a_q] = \emptyset$ if $p \neq q$, $(p, q = 1, \ldots, m)$ and $P = \bigcup_{i=1}^{m} [a_i]$. Using Algorithm 2, we can obtain F_{a_i}, and further $N(F_{a_i}), (i = 1, \ldots, m)$.

Algorithm 3 To obtain the set C of all the absolute necessary attributes and the set R of all the relative necessary attributes.

Input $N(a_i), \mathfrak{B}^{a_i}, N(F_{a_i}), [a_i], (i = 1, \ldots, m)$

Output C, R

Step 1. $j = 1, C = \emptyset, R = \emptyset$.

Step 2. If $j < m + 1$, then go to Step 3; otherwise, stop.

Step 3. If $N(a_j) = N(F_{a_j})$, then $C = C, R = R$, $j = j + 1$, go to Step 2;
otherwise, go to Step 4.

Step 4. If $|\mathfrak{B}^{a_j}| = 1$, then $C = C \cup a_j$, $j = j + 1$, go to Step 2;

otherwise, $R = R \cup [a_j]$, $j = j + 1$, go to Step 2.

Lemma 3.3 Let $\mathbf{I}$ be an index set including all the reducts in (O, P, I) and $a \in P$ be a relative necessary attribute. Then, $B_k \cap [a] \neq \emptyset$ holds for any reduct set B_k $(k \in \mathbf{I})$.

Proof Otherwise, there is a reduct set B_k satisfying $B_k \cap [a] = \emptyset$.

Lemma 2.1.1 points out $(N([a]), N(N([a]))) \in \mathcal{B}(O, P, I)$. However, if $N([a]) \neq N(D)$ for any $D \subseteq B_k$, then $(N([a]), N(N([a]))) \notin \mathcal{B}(O, B_k, I_{B_k})$, a contradiction to $\mathcal{B}(O, P, I) \cong \mathcal{B}(O, B_k, I_{B_k})$. Hence, suppose $B \subseteq B_k$ satisfying $N(B) = N([a])$. By virtue of $\bigcap_{b \in B} N(b) = N(B) = N([a]) = \bigcap_{t \in [a]} N(t) = N(a)$, we obtain $N(a) \subseteq N(b)$ for any $b \in B$.

If $N(b_0) = N(a)$ for some $b_0 \in B$, then $b_0 \in B \cap [a]$, a contradiction to $B \cap [a] = \emptyset$. That is to say, $N(a) \subset N(b)$ is correct for any $b \in B$. This follows $B \subseteq F_a$. So, $N(F_a) \subseteq N(B)$ holds according to Lemma 2.3.1, and further, $N(a) = N(B) = \bigcap_{b \in B} N(b) \supseteq N(F_a)$ holds. By Theorem 3.1, this is a contradiction to the property of a as a relative necessary attribute.

By extension, we can express the following result.

Theorem 3.2 Let C be the set of all the absolute necessary attributes and $[R] = \{[x_j] \mid x_j \text{ is a relative necessary attribute}\}$ satisfying $[x_p] \cap [x_q] = \emptyset$ if $p \neq q; p, q = 1, \ldots, t = |[R]|$. Then $\{C \cup \{y_1, \ldots, y_t\} \mid y_j \in [x_j], j = 1, \ldots, t\} = \{B_k \mid B_k \text{ is a reduct set of } P, k \in \mathbf{I}\}$, where $\mathbf{I}$ is an index set including all the reducts in (O, P, I).

Proof By Definition 2.1.2, for any reduct B_k $(k \in \mathbf{I})$, $c \in B_k$ holds for every $k \in \mathbf{I}$ and $c \in C$. Lemma 3.3 follows $B_k \cap [x_j] \neq \emptyset$, and besides, $|B_k \cap [x_j]| = 1$ holds since Lemma 3.2 $(j = 1, \ldots, t)$. Hence, every $C \cup \{y_1, \ldots, y_t\}$ is a reduct and every B_k can be written as the style of $C \cup \{y_1, \ldots, y_t\}$. Therefore, the needed result is obtained.

Theorem 3.2 implies the existence of reducts. This is the same as [8]. In addition, Theorem 3.2 gives the composition of a reduct.

Using Theorem 3.2, we present an algorithm to search out any reduct set.

Algorithm 4 To obtain B_k: a reduct set.

Input $P = \{a_1, a_2, \ldots, a_n\}$

Output B_k

Step 1. Using Algorithm 1, obtain $[a_j], (j = 1, \ldots, n)$.

Further, obtain $\{[a_p] : p = 1, \ldots, m\}$, where $[a_u] \cap [a_w] = \emptyset$ if $u \neq w; (u, w = 1, \ldots, m)$,

and obtain $\mathfrak{B}^{a_p} = \{\{x\} \mid x \in [a_p]\}, (p = 1, \ldots, m)$.

Step 2. Using Algorithm 2, obtain $F_{a_p}, (p = 1, \ldots, m)$.

Step 3. Using Algorithm 3, obtain C, the set of all the absolute necessary attributes; R, the set of all the relative necessary attributes, and $[R] = \{[y_l] : l = 1, \ldots, t\}$ such that $y_l \in R, (l = 1, \ldots, t)$ and $[y_i] \cap [y_j] = \emptyset$ if $i \neq j; (i, j = 1, \ldots, t)$.

Step 4. $B_k = C \cup \{y_1, \ldots, y_t\}$.

We present an example to demonstrate the use of Algorithm 4.

Example 3.1 A context (O, P, I) is given, where $O = \{1,2,3,4\}, P = \{a,b,c,d,e\}$, and I is shown in Table 3.1.

	a	b	c	d	e
1	$\times$	$\times$	0	$\times$	$\times$
2	$\times$	$\times$	$\times$	0	0
3	0	0	0	$\times$	0
4	$\times$	$\times$	$\times$	0	0

Table 3.1 A context for using Algorithm 4

The inducing bipartite graph of (O, P, I) is Figure 3.1.

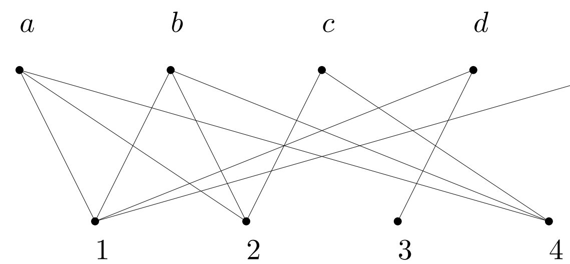

Figure 3.1 Inducing bipartite graph for the context in Table 3.1

By Definition 2.3.1, we obtain $N(a) = \{1,2,4\}, N(b) = \{1,2,4\}, N(c) = \{2,4\}, N(d) = \{1,3\}, N(e) = \{1\}$.

Using Algorithm 1, we find $[a] = \{a,b\}, [c] = \{c\}, [d] = \{d\}, [e] = \{e\}$. Hence, from the definition of $\mathfrak{B}^x$ for any $x \in P$ in Lemma 3.1, we obtain $\mathfrak{B}^a = \{\{a\},\{b\}\} = \mathfrak{B}^b$, $\mathfrak{B}^c = \{\{c\}\}, \mathfrak{B}^d = \{\{d\}\}, \mathfrak{B}^e = \{\{e\}\}$.

In light of $P \setminus [a] = [c] \cup [d] \cup [e] = \{c,d,e\}$, using Algorithm 2, it follows $F_a = F_b = \emptyset$. In light of $P \setminus [c] = [a] \cup [d] \cup [e] = \{a,d,e\}$, using Algorithm 2, we find $F_c = \{a,b\}$. Similarly, $F_d = \emptyset$ and $F_e = \{a,b,d\}$.

In view of $N(F_e) = \bigcap_{x \in F_e} N(x) = \{1\}$, we obtain $N(e) = \{1\} = N(F_e)$. Analogously, we obtain $N(F_c) = N(\{a,b\}) = \{1,2,4\}$, and in addition, we gain $N(F_a) = N(F_b) = N(F_d) = N(\emptyset) = O$ since Lemma 2.3.2.

Moreover, there are $N(e) = N(F_e), N(a) \subset N(F_a), N(c) \subset N(F_c)$ and $N(d) \subset N(F_d)$. Using Algorithm 3, it produces $C = \{c,d\}, R = \{a,b\}$. Hence, $[R] = \{[a]\} = \{[b]\}$ holds.

Using Algorithm 4, there are two reducts, one is $C \cup \{a\} = \{a,c,d\}$ and another is $C \cup \{b\} = \{b,c,d\}$.

The context in Example 3.1 comes from [8, Example 1]. Using the matroid-based approach presented in this section, we obtain C, R and all the reducts for the context. All these consequences are the same to the correspondent results in [8] which uses a different method from ours. Hence, we may state that our methods are successful.

In the future works, with matroidal approaches, for a context, we will study on how to search out the reduce attributes, and explore the wider applications of concept lattices.

Acknowledgements

This research is supported by NSF of China(61073121) and NSF of Hebei Province (A2013201119,A2014201033).

References

[1] R.Wille, Restructuring lattice theory: an approach based on hierarchies of concepts, in: I.Rival (Eds.), Ordered Sets, Dordrecht Boston, Reidel, 1982, pp.445-470.

[2] C.Carpineto, G.Romano, Concept Data Analysis: Theory and Applications, John Wiley & Sons Ltd., England, 2004.

[3] B.Ganter, R.Wille, Formal Concept Analysis: Mathematical Foundations, Springer-Verlag Berlin, Heidelberg, 1999. (German vesion: Springer, Heidelberg, 1996)

[4] B.Ganter, G.Stumme, R.Wille, Formal Concept Analysis: Foundations and Applications, Springer-Verlag Berlin, Heidelberg, 2005.

[5] J.Mi, Y.Leung, W.Wu, Approaches to attribute reduction in concept lattices induced by axialities, Knowl.-Based Syst., 23(2010)504-511.

[6] V.Snasel, M.Polovincak, H.M.Dahwa, Concept lattice reduction by singular value decomposition, Proceedings of the Spring Young Researcher's Colloquium on Database and Information Systems, Moscow, 2007.

[7] X.Wang, Approaches to attribute reduction in concept lattices based on rough set theory, Intern. J. of Hybrid Inf. Technol., 5(2)(2012)67-79.

[8] W.X.Zhang, L.Wei, J.Qi, Attribute reduction theory and approach to concept lattice, Sci. in China, series F: Inf. Sci., 48(6)(2005)713-726.

[9] H.Whitney, On the abstract properties of linear dependence, Am. J. of Math., 57(3)(1935)509-533.

[10] H.Mao, The relation between matroid and concept lattice, Adv. in Math.(China), 35(3)(2006)361-365.(in Chinese with English summary)

[11] H.Mao, On closure axioms for a matroid using Galois connections, Math. Commun., 14(2)(2009)425-432.

[12] H.Mao, On lattice-theoretical construction of matroids, Annu. of the Alexandru Ioan Cuza Univ.-Math., 59(1)(2013)201-208.

[13] H.Mao, Characterization and reduction of concept lattices through matroid theory, Inf. Sci., 281(10)(2014) 338-354.

[14] J.Tang, K.She, W.Zhu, Matroidal structure of rough sets from the viewpoint of graph theory, J. of Appl. Math., vol.2012, Article ID 973920, 27 pages.

[15] S.Wang, Q.Zhu, W.Zhu, F.Min, Matroidal structure of rough sets and its characterization to attribute reduction, Knowl.-Based Syst., 36(2012)155-161.

[16] Abello J, Pogel A J, Miller L. Breadth first search graph partitions and concept lattices. J. of Univer. Comput. Sci., 10(8)(2004)934-954.

[17] G.Grätzer, Lattice Theory: Foundation, Birkhäuser, Basel, 2011.

[18] D.J.A.Welsh, Matroid Theory, Academic Press, London, 1976.

[19] J.A.Bondy, U.S.R.Murty, Graph Theory with Applications, Elsevier Science Publishing Co. Inc., New York, 1976.