On Covering Attribute Sets by Reducts

Mikhail Moshkov[1], Andrzej Skowron[2], and Zbigniew Suraj[3]

[1] Institute of Computer Science, University of Silesia
Będzińska 39, 41-200 Sosnowiec, Poland
moshkov@us.edu.pl
[2] Institute of Mathematics, Warsaw University
Banacha 2, 02-097 Warsaw, Poland
skowron@mimuw.edu.pl
[3] Chair of Computer Science, University of Rzeszów
Rejtana 16A, 35-310 Rzeszów, Poland
zsuraj@univ.rzeszow.pl

Abstract. For any fixed natural k, there exists a polynomial in time algorithm which for a given decision table T and given k conditional attributes recognizes if there exist a decision reduct of T containing these k attributes.

Keywords: rough sets, decision tables, decision reducts.

1 Introduction

The set of all decision reducts of a decision table T [4] contains rich information about the table T. Unfortunately, there are no polynomial algorithms for construction of the set of all reducts.

In this paper, we show that there are polynomial (in time) algorithms for obtaining of indirect but useful information about this set.

We show that for any fixed natural k, there exists a polynomial (in time) algorithm $\mathcal{A}_k$ checking, for a given decision table T and given k conditional attributes, if there exist a reduct for T covering these k attributes.

The information obtained on the basis of algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$ can be represented in a simple graphical form. One can construct a graph with the set of vertices equal to the set of attributes covered by at least one reduct, and the set of edges coincides with the set of pairs of attributes which do not belong to any reduct. The degree of an attribute in this graph (the number of edges incident to this attribute) characterizes attribute importance. The changes of this graph after adding of a new object to the decision table allow us to evaluate the degree of influence of this new object on a structure of the reduct set of decision table. In the paper, we construct such graphs for two real-life decision tables. Some properties of such graphs are studied in [2].

Note that there exist close analogies between results of this paper and results obtained in [1], where the following problem was considered: for a given positive Boolean function f and given subset of its variables it is required to recognize

M. Kryszkiewicz et al. (Eds.): RSEISP 2007, LNAI 4585, pp. 175–180, 2007.

if there exists a prime implicant of dual Boolean function f^d containing these variables.

Another way for efficient extracting from a given decision table T of indirect information about the set of all reducts and its graphical representation was considered in [7]. It was shown that there exists a polynomial algorithm for constructing the so-called pairwise core graph for a given decision table T. The set of vertices of this graph is equal to the set of conditional attributes of T, and the set of edges coincides with the two element sets of attributes disjoint with the core of T (i.e., the intersection of all reducts of T) and having non-empty intersection with any reduct of T. This example is a step toward a realization of a program suggested in early 90s by Andrzej Skowron in his lectures at Warsaw University to study geometry of reducts for developing tools for investigating geometrical properties of reducts in the space of all reducts of a given information system. For example, the core of a given information system can be empty but in the reduct space can exist only few families of reducts with non-empty intersection.

2 On Covering of k Attribute Sets by Reducts

A *decision table* T is a finite table in which each column is labeled by a *conditional attribute*. Rows of the table T are interpreted as tuples of values of conditional attributes on some objects. Each row is labeled by a *decision* which is interpreted as the value of the *decision attribute*[1].

Let A be the set of conditional attributes (the set of names of conditional attributes) of T. We will say that a conditional attribute $a \in A$ *separates* two rows if these rows have different values at the intersection with the column labeled by a. We will say that two rows are *different* if at least one attribute $a \in A$ separates these rows. Denote by $P(T)$ the set of unordered pairs of different rows from T which are labeled by different decisions.

A subset R of the set A is called a *test* for T if for each pair of rows from $P(T)$ there exists an attribute from R which separates rows in this pair. A test R for T is called a *reduct* for T if each proper subset of R is not a test for T. In the sequel, we deal with decision reducts but we will omit the word "decision".

Let us fix a natural number k. We consider the following *covering problem for k attributes by a reduct*: for a given decision table T with the set of conditional attributes A, a subset B of the set A, and k pairwise different attributes $a_1, \ldots, a_k \in B$ it is required to recognize if there exist a reduct R for T such that $R \subseteq B$ and $a_1, \ldots, a_k \in R$, and if the answer is "yes" it is required to construct such a reduct. We describe a polynomial in time algorithm $\mathcal{A}_k$ for the covering problem.

[1] We consider uniformly both consistent and inconsistent decision tables. However, in the case of inconsistent decision table, one can use also the so called generalized decision instead of the original decision [4,5,6].

For $a \in A$, we denote by $P_T(a)$ the set of pairs of rows from $P(T)$ separated by a. For $a_1, \ldots, a_k \in A$ and $a_j \in \{a_1, \ldots, a_k\}$ let

$$P_T(a_j|a_1, \ldots, a_k) = P_T(a_j) \setminus \bigcup_{i \in \{1,\ldots,k\} \setminus \{j\}} P_T(a_i).$$

For $a_1, \ldots, a_k \in A$, let

$$\mathcal{P}_T(a_1, \ldots, a_k) = P_T(a_1|a_1, \ldots, a_k) \times \ldots \times P_T(a_k|a_1, \ldots, a_k).$$

Assuming that $(\pi_1, \ldots, \pi_k) \in \mathcal{P}_T(a_1, \ldots, a_k)$, we denote by

$$D_T(B, a_1, \ldots, a_k, \pi_1, \ldots, \pi_k)$$

the set of attributes a from $B \setminus \{a_1, \ldots, a_k\}$ such that a separates rows in at least one pair of rows from the set $\{\pi_1, \ldots, \pi_k\}$. Note that

$$D_T(B, a_1, \ldots, a_k, \pi_1, \ldots, \pi_k) = \bigcup_{j=1}^{k} D_T(B, a_j, \pi_j).$$

Using algorithm $\mathcal{A}_k$ first the set $\mathcal{P}_T(a_1, \ldots, a_k)$ is constructed. Next, for each tuple $(\pi_1, \ldots, \pi_k) \in \mathcal{P}_T(a_1, \ldots, a_k)$ the set

$$D_T(B, a_1, \ldots, a_k, \pi_1, \ldots, \pi_k)$$

is constructed and it is verified if the set $B \setminus D_T(B, a_1, \ldots, a_k, \pi_1, \ldots, \pi_k)$ is a test for T. It is clear that $|\mathcal{P}_T(a_1, \ldots, a_k)| \leq n^{2k}$, where n is the number of rows

---

Algorithm 1. Algorithm $\mathcal{A}_k$ for solving of the covering problem for k attributes by a reduct

---

Input: Decision table T with the set of conditional attributes A, $B \subseteq A$, and $a_1, \ldots, a_k \in B$.

Output: If there exists a reduct R for T such that $R \subseteq B$ and $a_1, \ldots, a_k \in R$, then the output is one of such reducts; otherwise, the output is "no".

```
construct the set P_T(a_1, ..., a_k);
for any tuple (π_1, ..., π_k) ∈ P_T(a_1, ..., a_k) do
    R ⟵ B \ D_T(B, a_1, ..., a_k, π_1, ..., π_k)
    if R is a test for T then
        while R is not a reduct for T do
            select a ∈ R such that R \ {a} is a test for T;
            R := R \ {a}
        end
        return R;
        stop
    end
end
return "no" (in particular, if P_T(a_1, ..., a_k) = ∅, then the output is "no")
```

in T. Using this inequality and the fact that k is fixed natural number, one can prove that the algorithm $\mathcal{A}_k$ has polynomial time complexity. Unfortunately, algorithm $\mathcal{A}_k$ has relatively high time complexity.

The considered algorithm is based on the following proposition:

Proposition 1. *Let T be a decision table with the set of conditional attributes A, $B \subseteq A$, and $a_1, \ldots, a_k \in B$. Then the following statements hold:*

1. *A reduct R for T such that $R \subseteq B$ and $a_1, \ldots, a_k \in R$ exists if and only if there exists a tuple $(\pi_1, \ldots, \pi_k) \in \mathcal{P}_T(a_1, \ldots, a_k)$ such that*
$$B \setminus D_T(B, a_1, \ldots, a_k, \pi_1, \ldots, \pi_k)$$
is a test for T.
2. *If the set $S = B \setminus D_T(B, a_1, \ldots, a_k, \pi_1, \ldots, \pi_k)$ is a test for T then each reduct Q for T, obtained from S by removing from S of some attributes, has the following properties: $a_1, \ldots, a_k \in Q$ and $Q \subseteq B$.*

Proof. Let R be a reduct for T such that $a_1, \ldots, a_k \in R$ and $R \subseteq B$. It is clear that for each $a_j \in \{a_1, \ldots, a_k\}$ there exists a pair of rows π_j from $P(T)$ such that a_j is the only attribute a_j from the set R separating this pair. It is clear that $(\pi_1, \ldots, \pi_k) \in \mathcal{P}_T(a_1, \ldots, a_k)$ and $R \subseteq B \setminus D_T(B, a_1, \ldots, a_k, \pi_1, \ldots, \pi_k)$. Since R is a reduct for T, we conclude that $B \setminus D_T(B, a_1, \ldots, a_k, \pi_1, \ldots, \pi_k)$ is a test for T.

Let us assume that there exists a tuple $(\pi_1, \ldots, \pi_k) \in \mathcal{P}_T(a_1, \ldots, a_k)$ such that the set $S = B \setminus D_T(B, a_1, \ldots, a_k, \pi_1, \ldots, \pi_k)$ is a test for T. Let Q be a reduct for T obtained by removing some attributes from S. It is also clear that $Q \subseteq B$. Let $j \in \{1, \ldots, k\}$. Since a_j is the only attribute from the test S separating rows from π_j, we have $a_j \in Q$. Thus, $a_1, \ldots, a_k \in Q$. □

3 Graphical Representation of Information About the Set of Reducts

Let T be a decision table with the set of conditional attributes A. Let $B \subseteq A$. Using polynomial algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$ one can construct a graph $G(T, B)$. The set of vertices of this graph coincides with the set of attributes $a \in B$ for each of which there exists a reduct R for T such that $R \subseteq B$ and $a \in R$. Two different vertices a_1 and a_2 of $G(T, B)$ are linked by an edge if and only if there is no a reduct R for T such that $R \subseteq B$ and $a_1, a_2 \in R$. Let us denote by $G(T)$ the graph $G(T, A)$.

Now, we present the results of two experiments with real-life decision tables from [3].

Example 1. Let us denote by T_L the decision table "Lymphography" [3] with 18 conditional attributes $a_1, \ldots, a_{18}$ and 148 rows. Each of the considered attributes is a vertex of the graph $G(T_L)$. The graph $G(T_L)$ is depicted in Fig. 1. In particular, one can observe from $G(T_L)$ that any reduct of T_L containing a_4 is disjoint with $\{a_2, a_3, a_5, a_7, a_8, a_9, a_{10}, a_{12}\}$.

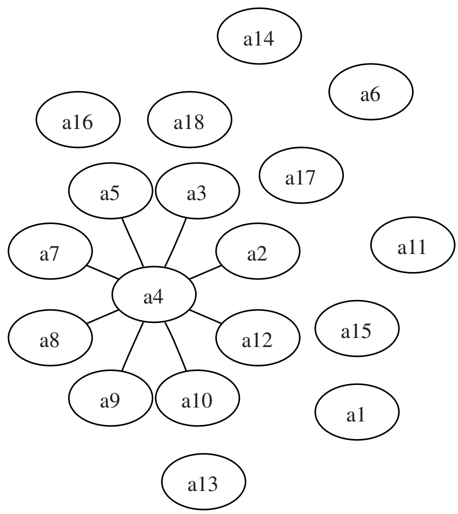

Fig. 1. Graph $G(T_L)$ for the decision table T_L ("Lymphography")

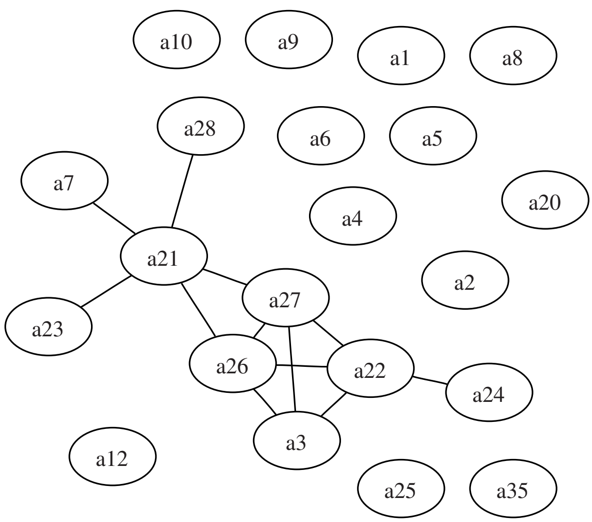

Fig. 2. Graph $G(T_S)$ for the decision table T_S ("Soybean-small")

Example 2. Let us denote by T_S the decision table "Soybean-small" [3] with 35 conditional attributes $a_1, \ldots, a_{35}$ and 47 rows. Only attributes $a_1, \ldots, a_{10}, a_{12}$ and $a_{20}, \ldots, a_{28}, a_{35}$ are vertices of the graph $G(T_S)$. The graph $G(T_S)$ is depicted in Fig. 2.

Some properties of graphs $G(T)$ are studied in [2]. In particular, it is shown that there exists a correlation between the degree of an attribute in $G(T)$ and the number of reducts of T which cover this attribute (last parameter is considered often as attribute importance).

4 Conclusions

In the paper, for each natural k a polynomial algorithm $\mathcal{A}_k$ is studied which for a given decision table and given k conditional attributes recognizes if there exist a decision reduct covering these k attributes. Results of two computer experiments with algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$ are reported.

In our further study we would like to check if there exist efficient randomized algorithms for solution of the considered in the paper problem.

Acknowledgments

The authors are grateful to Marcin Piliszczuk for performing of experiments and for his suggestion allowing us to improve algorithms $\mathcal{A}_k$.

References

1. Boros, E., Gurvich, V., Hammer, P.L.: Dual subimplicants of positive Boolean functions. Optimization Methods and Software 10, 147–156 (1998)
2. Moshkov, M., Piliszczuk, M.: Graphical representation of information on the set of reducts. Joint Rough Sets Symposium JRS07, Toronto, Canada, May 14-16, 2007 (to appear)
3. Newman, D.J., Hettich, S., Blake, C.L., Merz, C.J.: UCI Repository of machine learning databases. University of California, Irvine, Department of Information and Computer Sciences (1998) `http://www.ics.uci.edu/~mlearn/MLRepository.html`
4. Pawlak, Z.: Rough Sets – Theoretical Aspects of Reasoning about Data. Kluwer Academic Publishers, Dordrecht (1991)
5. Pawlak, Z., Skowron, A.: Rudiments of rough sets. Information Sciences 177(1), 3–27 (2007)
6. Pawlak, Z., Skowron, A.: Rough sets and Boolean reasoning. Information Sciences 177(1), 41–73 (2007)
7. Wróblewski, J.: Pairwise cores in information systems. In: Ślęzak, D., Wang, G., Szczuka, M., Düntsch, I., Yao, Y. (eds.) RSFDGrC 2005. LNCS (LNAI), vol. 3641, pp. 166–175. Springer, Heidelberg (2005)