Minimum cost attribute reduction in decision-theoretic rough set models

Xiuyi Jia\textsuperscript{a,c,*}, Wenhe Liao\textsuperscript{b}, Zhenmin Tang\textsuperscript{a}, Lin Shang\textsuperscript{c}

\textsuperscript{a}School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China
\textsuperscript{b}School of Mechanical Engineering, Nanjing University of Science and Technology, Nanjing 210094, China
\textsuperscript{c}State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China

**Abstract**

In classical rough set models, attribute reduction generally keeps the positive or non-negative regions unchanged, as these regions do not decrease with the addition of attributes. However, the monotonicity property in decision-theoretic rough set models does not hold. This is partly due to the fact that all regions are determined according to the Bayesian decision procedure. Consequently, it is difficult to evaluate and interpret region-preservation attribute reduction in decision-theoretic rough set models. This paper provides a new definition of attribute reduct for decision-theoretic rough set models. The new attribute reduction is formulated as an optimization problem. The objective is to minimize the cost of decisions. Theoretical analysis shows the meaning of the optimization problem. Both the problem definition and the objective function have good interpretation. A heuristic approach, a genetic approach and a simulated annealing approach to the new problem are proposed. Experimental results on several data sets indicate the efficiency of these approaches.

**1. Introduction**

Attribute reduction is an important concept in rough set theory and has drawn attention from many researchers. In Pawlak rough set model and other generalized models, several attribute reductions and the relationships among them have been studied [5,9,11,14–19,23–25,27,29–31,40,39,41–43]. Generally speaking, attribute reduction can be interpreted as a process of finding the minimal set of attributes that can preserve or improve one or several criteria. The minimal set of attributes is called an attribute reduct. Based on the criteria, we can classify these definitions into two categories: qualitative definitions and quantitative definitions.

For qualitative definitions, some qualitative criteria are used to define a reduct, such as, qualities of classification and dependency. Pawlak defined an attribute reduct that keeps the positive region unchanged [20]. Zhang et al. [41] studied attribute reducts in the concept lattice analysis based on an isomorphic criterion between two concept lattices. Slezak [24,25] defined a generalized reduct which is the minimal set of attributes, and keeps the generalized decision or majority decision under the generalized decision function.

For quantitative definitions, quantitative criteria are applied. They may be the corresponding criteria of the qualitative definitions. For example, in Pawlak rough set model, positive region can be replaced by a quantitative criterion: quality of classification \(\gamma [20,21]\). Quafafou [23] defined a kind of \(\alpha\)-Reduce with the notion of \(\alpha\)-dependency to study the dependency relation. In probabilistic rough set models, some qualitative properties in Pawlak rough set model do not hold anymore.
Therefore, quantitative definitions are mostly investigated in these models. Mi et al. [16] studied the $\beta$ lower distribution reduct and $\beta$ upper distribution reduct in a variable precision rough model. Yao and Zhao [39] investigated several quantitative criteria to define different reducts in decision-theoretic rough set models. In this paper, we focus on the decision-theoretic rough set model and propose a new definition of an attribute reduct based on a new quantitative criterion.

A decision-theoretic rough set model (DTRS) [38,37,32,33] is a probabilistic rough set model. It can derive several probabilistic rough set models when proper cost functions are used, such as 0.5 probabilistic rough model [22], variable precision rough set model [44], Pawlak rough set model [20] and Baysian rough set model [26]. Based on the decision-theoretic rough set model, Yao [34–36] studied the superiority of three-way decisions in probabilistic rough set models. The results increase our understanding of rough set theory and can be used for decision making. In rough set models, a concept is usually described by three regions: positive region, boundary region and negative region. The positive region of a concept means making a decision of acceptance, the negative region means making a decision of rejection, and the boundary region means making an abstained or non-committed decision [35]. Decision-theoretic rough set models are based on Bayesian decision procedure, which provides systematic methods for deriving the required thresholds on probabilities for defining the three regions. An object is classified into a particular region because the cost or loss of classifying it into the region is less than the cost of classifying it into other regions.

For attribute reduction in the decision-theoretic rough set models, Yao and Zhao [39] studied positive region, non-negative region, confidence of rules, coverage of rules, cost of rules and other quantitative criteria. They proposed a general definition of probabilistic attribute reduct. Under this general definition, users can define a proper attribute reduct according to their applications and data sets. It can be used to interpret most definitions of attribute reducts. They also examined the notions of the non-monotonicity of probabilistic positive or non-negative regions and decision rules. The difficulties of positive region preservation attribute reduct in probabilistic rough set model have been discussed [42]. The difficulties can be summarized by a simple question: the three regions could be increased or decreased after removing an attribute, which kind of change is better? To solve this problem, they argued that decision-monotonicity criterion should be considered in the region-preservation definitions. Li et al. [12] proposed a positive region expanding reduct. It should be pointed out that decision-monotonicity is actually a kind of subjective criterion, which still cannot solve the interpretation difficulties.

The purpose of this paper is to answer the above question. We want to find an objective criterion to solve the problem. Recall that a decision-theoretic rough set model is based on Bayesian decision procedure and the principle of making decisions is minimizing the decision cost. Decision cost is a very important notion in decision-theoretic rough set models: the purpose of decisions making is to minimize the cost. Therefore, the process of attribute reduction should help in minimizing the decision cost. Decision cost can be intuitively considered as the objective criterion for defining an attribute reduct.

In this paper, a minimum cost attribute reduct in decision-theoretic rough set models is defined. An optimization problem is constructed with the objective of minimizing the decision cost. A decreasing cost attribute reduct is defined, which ensures that the cost will be decreased or unchanged for decisions making by using the reduct. By solving the optimization problem, a minimum cost attribute reduct is defined as finding a minimal subset of attributes that guarantees the minimum decision cost. Theoretical analysis shows that both the optimization problem and the new attribute reduct have meaningful interpretations.

A heuristic approach, a genetic approach and a simulated annealing approach to minimum cost attribute reduction are also presented. For the heuristic approach, the addition strategy is applied. Genetic algorithm and simulated annealing algorithm for the reduction are implemented. The experiment results show that the misclassification cost is less than that generated by region-based reducts.

The rest of the paper is organized as follows. In Section 2, we introduce the main ideas of decision-theoretic rough set models. In Section 3, we briefly review existing definitions of attribute reducts in decision-theoretic rough set models. Section 4 gives a detailed explanation of the minimum cost attribute reduction in decision-theoretic rough set models. An optimization problem is proposed and several approaches to the attribute reduction are designed. Section 5 presents the results of experiments. Section 6 concludes the present studies and gives the further work.

2. Decision-theoretic rough set models

In this section, we present some basic notions of decision-theoretic rough set models [35].

Definition 1. A decision table is the following tuple:

\[ S = (U, At = C \cup D, \{V_a | a \in At\}, \{I_a | a \in At\}) \]

(1)

where $U$ is a finite nonempty set of objects, $At$ is a finite nonempty set of attributes, $C$ is a set of condition attributes describing the objects, and $D$ is a set of decision attributes that indicates the classes of objects. $V_a$ is a nonempty set of values of $a \in At$, and $I_a : U \rightarrow V_a$ is an information function that maps an object in $U$ to exactly one value in $V_a$.

In a decision table, an object $x$ is described by its equivalence class under a set of attributes $A \subseteq At$:

\[ [x]_A = \{ y \in U | \forall a \in A, I_a(x) = I_a(y) \} \]

Let $\pi_A$ denote the partition induced by the set of attributes $A \subseteq At$ and $\pi_D = \{D_1, D_2, \ldots, D_m\}$ denote the partition of the universe $U$ induced by the set of decision attributes $D$. 

Let $\Omega = \{\omega_1, \ldots, \omega_m\}$ be a finite set of $m$ states and let $A = \{a_1, \ldots, a_n\}$ be a finite set of $n$ possible actions. Let $\lambda(x|\omega_j)$ denote the cost or loss, for taking action $a_i$ when the state is $\omega_j$. Let $p(a_j|x)$ be the conditional probability of an object $x$ being in state $\omega_j$. Suppose action $a_i$ is taken, the expected cost associated with taking action $a_i$ is given by:

$$R(a_i|x) = \sum_{j=1}^{m} \lambda(a_i|\omega_j) \cdot p(\omega_j|x).$$

(2)

In decision-theoretic rough set models, the set of states $\Omega = \{X, X^c\}$, indicating that an object is in a decision class $X$ and not in $X$, respectively. The probabilities for these two complement states can be denoted as $p(X|x) = \frac{X}{X^c}$ and $p(X^c|x) = 1 - p(X|x))$. With respect to the three regions: positive region ($\text{POS}(X)$), boundary region ($\text{BND}(X)$) and negative region ($\text{NEG}(X)$), the set of actions with respect to a state is given by $A = \{a_P, a_B, a_N\}$, where $a_P$, $a_B$, and $a_N$ represent the three actions in classifying an object $x$, namely, deciding $x \in \text{POS}(X)$, deciding $x \in \text{BND}(X)$, and deciding $x \in \text{NEG}(X)$, respectively. Let $\lambda_{PP}$, $\lambda_{PB}$ and $\lambda_{PN}$ denote the cost incurred for taking actions $a_P$, $a_B$ and $a_N$, respectively, when an object belongs to $X$, and $\lambda_{BN}$, $\lambda_{BN}$ and $\lambda_{NN}$ denote the cost incurred for taking the same actions when the object does not belong to $X$.

Given a description $x$, a decision rule is a function $\tau(x)$ that specifies which action to take. That is, for every $x$, $\tau(x)$ takes one of the actions, $a_1, \ldots, a_n$. The overall risk $R$ is the expected cost associated with a given decision rule. Since $R(\tau(x)|x)$ is the expected cost associated with action $\tau(x)$, the overall risk is defined by:

$$R = \sum_x R(\tau(x)|x) \cdot p(x).$$

(4)

where the summation is over the set of all possible descriptions of objects. If $\tau(x)$ is chosen so that $R(\tau(x)|x)$ is as small as possible for every $x$, the overall risk $R$ is minimized. The Bayesian decision procedure suggests the following minimum cost decision rules:

(P) If $R_P \leq R_B$ and $R_P \leq R_N$, decide $x \in \text{POS}(X)$;

(B) If $R_B \leq R_P$ and $R_B \leq R_N$, decide $x \in \text{BND}(X)$;

(N) If $R_N \leq R_P$ and $R_N \leq R_B$, decide $x \in \text{NEG}(X)$.

Consider a special kind of cost functions with $\lambda_{PP} \leq \lambda_{PB} < \lambda_{PN}$ and $\lambda_{BN} \leq \lambda_{BN} < \lambda_{NN}$, that is, the cost of classifying an object $x$ belonging to $X$ into the positive region $\text{POS}(X)$ is less than or equal to the cost of classifying $x$ into the boundary region $\text{BND}(X)$, and both of these cost are strictly less than the cost of classifying $x$ into the negative region $\text{NEG}(X)$. The reverse order of cost is used for classifying an object not in $X$. The decision rules can be reexpressed as:

(P) If $p(X|x) \geq \alpha$ and $p(X|x) \geq \gamma$, decide $x \in \text{POS}(X)$;

(B) If $p(X|x) \leq \alpha$ and $p(X|x) \geq \beta$, decide $x \in \text{BND}(X)$;

(N) If $p(X|x) \leq \beta$ and $p(X|x) \leq \gamma$, decide $x \in \text{NEG}(X)$;

where the parameters $\alpha$, $\beta$, and $\gamma$ are defined as:

$$\alpha = \frac{(\lambda_{PN} - \lambda_{BN})}{(\lambda_{PN} - \lambda_{BN}) + (\lambda_{PP} - \lambda_{PB})},$$

$$\beta = \frac{(\lambda_{BN} - \lambda_{NN})}{(\lambda_{BN} - \lambda_{NN}) + (\lambda_{PB} - \lambda_{PP})},$$

$$\gamma = \frac{(\lambda_{PN} - \lambda_{NN})}{(\lambda_{PN} - \lambda_{NN}) + (\lambda_{PB} - \lambda_{PP})}.$$  

(5)

Each rule is defined by two out of the three parameters.

The conditions of rule (B) suggest that $\alpha \geq \beta$ may be a reasonable constraint; it will ensure a well-defined boundary region. If we use the following condition on the cost functions [35]:

$$\frac{(\lambda_{PB} - \lambda_{PP})}{(\lambda_{BN} - \lambda_{NN})} > \frac{(\lambda_{PB} - \lambda_{PP})}{(\lambda_{BN} - \lambda_{NN})},$$

(6)

then $0 \leq \beta < \gamma < \alpha \leq 1$. In this case, after tie-breaking, the following simplified rules are obtained:

(P1) If $p(X|x) \geq \alpha$, decide $x \in \text{POS}(X)$;

(B1) If $\beta < p(X|x) < \alpha$, decide $x \in \text{BND}(X)$;

(N1) If $p(X|x) \leq \beta$, decide $x \in \text{NEG}(X)$. 

Many other conditions on cost functions have been discussed in [42]. By using the thresholds, one can divide the universe \( U \) into three regions of a decision partition \( \pi_D \) based on \( (x, \beta) \):

\[
\begin{align*}
\text{POS}(x, \beta) (\pi_D | \pi_A) &= \{ x \in U | p(D_\text{max}(x_{\pi_D}|x_{\pi_A}) \geq \alpha) \}, \\
\text{BND}(x, \beta) (\pi_D | \pi_A) &= \{ x \in U | \beta < p(D_\text{max}(x_{\pi_D}|x_{\pi_A}) < \alpha) \}, \\
\text{NEG}(x, \beta) (\pi_D | \pi_A) &= \{ x \in U | p(D_\text{max}(x_{\pi_D}|x_{\pi_A}) \leq \beta) \},
\end{align*}
\]  

(7)

where \( D_\text{max}(x_{\pi_D}|x_{\pi_A}) = \arg \max_{0 \leq \alpha < 1} \left\{ \frac{\alpha}{\pi_{\pi_A}} \right\} \).

Unlike rules in the classical rough set theory, all three types of rules obtained from the three regions may be uncertain. They represent the levels of tolerance of errors in making incorrect decisions. Let \( p = p(D_\text{max}(x_{\pi_D}|x_{\pi_A}) \), the Bayesian expected cost of each rule can be expressed by follows:

\[
\begin{align*}
\text{Cost of positive rule} &= p \cdot \lambda_{pp} + (1 - p) \cdot \lambda_{PN}, \\
\text{Cost of boundary rule} &= p \cdot \lambda_{BP} + (1 - p) \cdot \lambda_{BN}, \\
\text{Cost of negative rule} &= p \cdot \lambda_{NP} + (1 - p) \cdot \lambda_{NN}.
\end{align*}
\]

(8)

Consider the special case where we assume zero cost for a correct classification, namely, \( \lambda_{pp} = \lambda_{NN} = 0 \), the decision costs of all rules are defined as [35]:

\[
\begin{align*}
\text{Cost of positive rule} &= (1 - p) \cdot \lambda_{PN}, \\
\text{Cost of boundary rule} &= p \cdot \lambda_{BP} + (1 - p) \cdot \lambda_{BN}, \\
\text{Cost of negative rule} &= p \cdot \lambda_{NP}.
\end{align*}
\]

(9)

For a given decision table, the decision cost of the table can be expressed as:

\[
\text{COST} = \sum_{p_i > z} (1 - p_i) \cdot \lambda_{PN} + \sum_{p_i \leq z} (p_i \cdot \lambda_{BP} + (1 - p_i) \cdot \lambda_{BN}) + \sum_{p_i < b} p_i \cdot \lambda_{NP}.
\]

(10)

where \( p_i = p(D_\text{max}(x_{\pi_D}|x_{\pi_A}) \).

In this expression, the cost of the whole table is composed of three types of cost: cost of the positive rules, cost of the boundary rules and cost of the negative rules.

3. Attribute reducts in decision-theoretic rough set models

In this section, we review existing definitions of attribute reducts in Pawlak rough set model and in decision-theoretic rough set models. The purpose of attribute reduction is to find a minimal subset of attributes that satisfies or improves one or several criteria compared with the entire set of attributes. By reviewing the current definitions of attribute reducts, we find that the main difference of these definitions lies in their criteria. According to the criteria, we classify these definitions into two categories: qualitative attribute reducts based on qualitative criteria and quantitative attribute reducts based on quantitative criteria.

Before we give a detailed explanation of the two kinds of reducts, we want to review first definitions of attribute reducts in Pawlak rough set model.

3.1. Attribute reducts in Pawlak rough set model

A classical attribute reduct in Pawlak rough set model is a relative reduct with respect to the decision attribute \( D \), which is defined by requiring that the positive region of the decision table or \( \pi_D \) is unchanged.

\textbf{Definition 2.} [20] Given a decision table \( S = (U, At = C \cup D, \{V_a | a \in At\}, \{I_a | a \in At\}) \), an attribute set \( R \subseteq C \) is a Pawlak reduct of \( C \) with respect to \( D \) if it satisfies the following two conditions.

\begin{enumerate}
\item \( \text{POS}(\pi_D | \pi_R) = \text{POS}(\pi_D | \pi_C) \),
\item for any attribute \( a \in R \), \( \text{POS}(\pi_D | \pi_{R - \{a\}}) \neq \text{POS}(\pi_D | \pi_C) \).
\end{enumerate}

In this definition, condition (1) is also called a jointly sufficient condition and condition (2) is called an individually necessary condition.

In Pawlak rough set model, \( \text{POS}(\pi_D | \pi_C) \cap \text{BND}(\pi_D | \pi_C) = \emptyset \), and \( \text{POS}(\pi_D | \pi_C) \cup \text{BND}(\pi_D | \pi_C) = U \). For a reduct \( R \subseteq C \), the condition \( \text{POS}(\pi_D | \pi_R) = \text{POS}(\pi_D | \pi_C) \) is equivalent to \( \text{BND}(\pi_D | \pi_R) = \text{BND}(\pi_D | \pi_C) \). The requirement of keeping positive region guarantees the same boundary region.

Many researchers use an equivalent definition of a Pawlak reduct. It is a kind of quantitative definition compared with a qualitative definition for a Pawlak reduct. The quantitative definition is based on a measure called the \textit{quality of classification}, which is defined as:

\[
\gamma(\pi_D | \pi_A) = \frac{|\text{POS}(\pi_D | \pi_A)|}{|U|}
\]

(11)
The quantitative attribute reduct is defined as follows.

**Definition 3.** Given a decision table $S = (U, At = C \cup D, \{V_a | a \in At\}, \{I_a | a \in At\})$, an attribute set $R \subseteq C$ is a Pawlak reduct of $C$ with respect to $D$ if it satisfies the following two conditions.

1. $\gamma(\pi_D|\pi_R) = \gamma(\pi_D|\pi_C)$;
2. for any attribute $a \in R$, $\gamma(\pi_D|\pi_{R-R(a)}) \neq \gamma(\pi_D|\pi_C)$ or $\gamma(\pi_D|\pi_{R-R(a)}) < \gamma(\pi_D|\pi_C)$.

Definitions 2 and 3 are equivalent in the Pawlak rough set model. This can be seen from the monotonicity of positive regions. Considering two subsets of condition attributes $A, B \subseteq C$ with $A \subseteq B$, we can easily obtain the monotonicity of the positive regions with respect to set inclusion of attributes, that is:

$$A \subseteq B \Rightarrow POS(\pi_D|\pi_A) \subseteq POS(\pi_D|\pi_B) \Rightarrow \gamma(\pi_D|\pi_A) \leq \gamma(\pi_D|\pi_B).$$

(12)

Based on this property, conditions (2) in Definitions 2 and 3 can guarantee that a reduct is minimal. The property is also used in designing attribute reduction algorithms, which could help reduce complexity of an attribute reduction construction algorithm.

**Example 1.** A simple decision table $S = (U, At = C \cup D, \{V_a | a \in At\}, \{I_a | a \in At\})$ shown in Table 1 is used for illustrating a Pawlak reduct. In this table, there are 9 objects with 3 different decision labels: $\{d_1, d_2, d_3\}$.

The positive region of this decision table is $POS(\pi_D|\pi_C) = \{a_1, a_3, a_4, a_7\}$, and the quality of classification is $\gamma(\pi_D|\pi_C) = \frac{4}{9}$.

According to Definitions 2 and 3, we can get 6 reducts of the decision table. They are: $\{c_1, c_2, c_3\}$, $\{c_2, c_3, c_5\}$, $\{c_4, c_5, c_6\}$, $\{c_2, c_4, c_5\}$, $\{c_3, c_4, c_6\}$, and $\{c_2, c_5, c_6\}$.

In Pawlak rough set model, the monotonicity of positive regions holds, but in probabilistic rough set models, it does not. As several probabilistic rough set models can be derived from a decision-theoretic rough set model, we concentrate on attribute reducts in decision-theoretic rough set models. Attribute reducts in decision-theoretic rough set models can be also classified into two categories: qualitative attribute reducts and quantitative attribute reducts.

### 3.2. Qualitative attribute reducts in decision-theoretic rough set models

In this section, we will review several definitions of qualitative attribute reducts in decision-theoretic rough set models, including the region preservation based definitions and other qualitative criteria based definitions.

#### 3.2.1. Positive region preservation attribute reduct

A positive region preservation attribute reduct in decision-theoretic rough set models can be defined as in Pawlak rough set model.

**Definition 4.** [42] Given a decision table $S = (U, At = C \cup D, \{V_a | a \in At\}, \{I_a | a \in At\})$, a condition attribute set $R \subseteq C$ is a positive region-preserved reduct of $C$ with respect to $D$ if the following two conditions are satisfied.

1. $POS_{(x, y)}(\pi_D|\pi_R) = POS_{(x, y)}(\pi_D|\pi_C)$,
2. for any attribute $a \in R$, $POS_{(x, y)}(\pi_D|\pi_{R-R(a)}) \neq POS_{(x, y)}(\pi_D|\pi_C)$.

In this definition, the probabilistic positive region of $\pi_D$ is used. In Pawlak rough set model, each object of the positive region “certainly” belongs to the region; in probabilistic rough set model, the object in the probabilistic positive region only “possibly” belongs to the region with the confidence thresholds $x$ and $y$.

<table>
<thead>
<tr>
<th>$C$</th>
<th>$D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c_1$</td>
<td>$d_1$</td>
</tr>
<tr>
<td>$c_2$</td>
<td>$d_2$</td>
</tr>
<tr>
<td>$c_3$</td>
<td>$d_3$</td>
</tr>
<tr>
<td>$c_4$</td>
<td>$d_4$</td>
</tr>
<tr>
<td>$c_5$</td>
<td>$d_5$</td>
</tr>
<tr>
<td>$c_6$</td>
<td>$d_6$</td>
</tr>
</tbody>
</table>

A decision table.
Example 2. For decision Table 1, assuming all cost functions are given as: \( \lambda_{pp} = \lambda_{NN} = 0, \lambda_{PN} = 6, \lambda_{NP} = 3, \lambda_{BP} = 1, \lambda_{BN} = 3 \), then we can get the two thresholds \( x = 0.75 \) and \( \beta = 0.6 \). According to Definition 4, we can also get 6 reducts: \( \{c_1, c_2, c_5\}, \{c_2, c_3, c_5\}, \{c_4, c_5, c_6\}, \{c_2, c_4, c_5\}, \{c_2, c_4, c_6\}, \) and \( \{c_3, c_5, c_6\} \).

3.2.2. Non-negative region preservation attribute reduct

In a probabilistic rough set model, all three regions of a concept are uncertain while they are certain in Pawlak rough set model. Some users are interested not only in the positive region but also in the boundary region. The union of the two regions could be only a subset of the universe.

3.3. Quantitative attribute reducts in decision-theoretic rough set models

A probabilistic rough set model can give a better tolerance of error than Pawlak rough set model. Many quantitative attributes in probabilistic rough set models are studied to replace the qualitative attribute reducts. In this section, we use the region extension based reduct.

3.3.1. Positive region extension based attribute reduct

Before we give the definition of a positive region extension based attribute reduct, we look at an example first.

Example 4. In Example 2, \( \{c_1, c_2, c_5\} \) is a reduct of the given decision table as its positive region is \( \{o_1, o_3, o_4, o_6, o_7, o_8\} \). We check the subset \( \{c_2, c_5\} \) and find that the positive region based on the subset is extended to \( \{o_1, o_2, o_3, o_4, o_6, o_7, o_8\} \). Is \( \{c_2, c_5\} \) a better choice for users?

According to semantics of the positive region, objects in the positive region can be classified to a certain category beyond certain level of uncertainty. It is a better choice for users to obtain a larger positive region after attribute reduction. We can define a positive region extension based attribute reduct.

Definition 6. [12] Given a decision table \( S = (U, At = C \cup D, \{V_a| a \in At\}, \{I_a| a \in At\}) \), a condition attribute set \( R \subseteq C \) is a positive region extension based reduct of \( C \) with respect to \( D \) if the following two conditions are satisfied.

\[
\begin{align*}
(1) & \quad |\text{POS}_{x, \alpha}(\pi_D|\pi_R)| \geq |\text{POS}_{x, \alpha}(\pi_D|\pi_C)|, \\
(2) & \quad \text{for any attribute } a \in R, \quad |\text{POS}_{x, \alpha}(\pi_D|\pi_{R \cup \{a\}})| < |\text{POS}_{x, \alpha}(\pi_D|\pi_C)|.
\end{align*}
\]

In this definition, the qualitative criterion \( |\text{POS}_{x, \alpha}(\pi_D|\pi_R)| \) is replaced by the quantitative criterion \( |\text{POS}_{x, \alpha}(\pi_D|\pi_R)| \), the objective of attribute reduction is to find the minimal set of condition attributes which makes the size of positive region of the decision table larger or the same.

Example 5. In the same decision table used in Example 2, according to Definition 6, we can find that \( \{c_4, c_5, c_6\}, \{c_2, c_4, c_5\}, \) and \( \{c_3, c_5\} \) are 3 reducts of the decision table. The positive region based on \( C \) is \( \{o_1, o_3, o_4, o_7, o_8\} \). For \( \{c_4, c_5, c_6\} \) and \( \{c_2, c_4, c_5\} \), the positive regions are unchanged. For \( \{c_3, c_5\} \), the positive region is \( \{o_1, o_2, o_3, o_4, o_6, o_7, o_8\} \), which is larger than the positive region based on \( C \).
3.3.2. Non-negative region extension based attribute reduct

Same as a positive region extension based attribute reduct, we can define a non-negative region extension based attribute reduct.

**Definition 7.** Given a decision table $S = (U, At = C \cup D, \{V_a|a \in At\}, \{I_a|a \in At\})$, a condition attribute set $R \subseteq C$ is a non-negative region extension based reduct of $C$ with respect to $D$ if the following two conditions are satisfied.

1. $|\neg \text{NEG}_{(x, \beta)}(\pi_D|\pi_R)| \geq |\neg \text{NEG}_{(x, \beta)}(\pi_D|\pi_C)|$.
2. For any attribute $a \in R$, $|\neg \text{NEG}_{(x, \beta)}(\pi_D|\pi_{R \setminus \{a\}})| < |\neg \text{NEG}_{(x, \beta)}(\pi_D|\pi_C)|$.

In this definition, it is assumed that users are in favor of a larger non-negative region.

**Example 6.** In the same decision table used in Example 2, the non-negative region based on $C$ is $\{o_1, o_2, o_3, o_4, o_6, o_7, o_8\}$. For subset $\{c_4, c_5\}$, we get non-negative region $\{o_1, o_2, o_3, o_4, o_5, o_6, o_8\}$. The non-negative region based on $\{c_4\}$ is $\{o_3, o_4, o_6, o_9\}$. According to Definition 7, $\{c_4, c_5\}$ is a reduct of the decision table. $\{c_2\}$ is another reduct of the decision table as the non-negative region based on it is $U$.

In the above example, for the non-negative region, a reduct only concerns the cardinality of the region, but not the exact objects in the region. This is a difference between qualitative criteria based definitions and quantitative criteria based definitions.

Several other quantitative criteria can be used to define attribute reducts, e.g., confidence, coverage, generality and other quantitative criteria, which are applied to define attribute reducts in decision-theoretic rough set models [39].

3.4. Remarks on attribute reducts in decision-theoretic rough set models

The reducts of Table 1 corresponding to Definitions 4–7 are summarized in Table 2. We find that different results are achieved from different definitions. Although different criteria bring different definitions, there are still two common properties that are expressed by two conditions: jointly sufficient condition and individually necessary condition.

To discuss the jointly sufficient condition, we need to make two things clear. The first one concerns the criterion selected. There are several criteria to be used, such as positive region, non-negative region, quality of classification, generality of a rule set, and so on. Users can select or define a criterion or a combination of a set of criteria to obtain a reduct according to their demands or applications. The other one point is satisfiability of the selected criterion. Most common definitions of reducts require that the criterion has the same level of performance as the entire set of attributes, such as Definitions 2–5. Besides the equivalent relation, other binary relations can be used in defining attribute reducts. For example, the binary relation $\geq$ is applied in Definitions 6, 7. We can also use $>$, $<$, and $\leq$ to define attribute reducts according to specific applications. Yao and Zhao [39] have given a general definition of attribute reducts by using a particular measure $e$ and a partial order relation $\geq$.

For the individually necessary condition, we need consider the monotonicity of the criterion with respect to inclusion set. If the monotonicity property holds, conditions (2) in Definitions 4–7 can guarantee that the reduct is minimal. If the monotonicity does not hold, condition (2) cannot guarantee getting the minimal set, and all definitions need to be redefined. The following is a re-expression of Definition 4. In this modified definition, we need to confirm all subsets of a reduct to make sure it is a minimal set.

**Definition 8.** Given a decision table $S = (U, At = C \cup D, \{V_a|a \in At\}, \{I_a|a \in At\})$, a condition attribute set $R \subseteq C$ is a positive region-preserved reduct of $C$ with respect to $D$ if the following two conditions are satisfied.

1. $\text{POS}_{(x, \beta)}(\pi_D|\pi_R) = \text{POS}_{(x, \beta)}(\pi_D|\pi_C)$.
2. For any subset $R' \subset R$, $\text{POS}_{(x, \beta)}(\pi_D|\pi_{R'}) \neq \text{POS}_{(x, \beta)}(\pi_D|\pi_C)$.

The monotonicity property of the criterion not only affects the individually necessary condition, but also brings an interpretation difficulty into the jointly sufficient condition. For example, in Example 6, the non-negative region induced by condition attribute set $C$ is: $\{o_1, o_2, o_3, o_4, o_6, o_7, o_8\}$, and the non-negative region induced by a subset $\{c_4, c_5\}$ is

<table>
<thead>
<tr>
<th>Table 2</th>
<th>Reducts of Table 1 corresponding to different definitions.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Different definitions</strong></td>
<td><strong>Reducts</strong></td>
</tr>
<tr>
<td>Positive region preservation</td>
<td>${c_1, c_2, c_3}$, ${c_2, c_3, c_4}$, ${c_4, c_5, c_6}$, ${c_5, c_6, c_7}$</td>
</tr>
<tr>
<td>Non-negative region preservation</td>
<td>${c_4, c_5, c_6}$, ${c_5, c_6, c_7}$</td>
</tr>
<tr>
<td>Positive region extension</td>
<td>${c_4, c_5, c_6}$, ${c_5, c_6, c_7}$, ${c_7}$</td>
</tr>
<tr>
<td>Non-negative region extension</td>
<td>${c_4, c_5}$, ${c_5}$</td>
</tr>
</tbody>
</table>
The two regions are not same but the sizes of them are equivalent. According to the definition, \( \{c_4, c_5\} \) is a reduct, but it changes the region indeed. There exists a paradox that they are quantitative equivalent but qualitative different. In paper [42], the difficulties with interpretations of region preservation attribute reduct in probabilistic rough set model have been discussed.

In probabilistic rough set models, such as decision-theoretic rough set models, the monotonicity of a particular region with respect to attribute set inclusion usually does not hold, and there exists the paradox between the qualitative measure and the quantitative measure. After removing an attribute, the regions of the decision table could be increased or decreased, it is difficult to decide which kind of change is better. The region-preserved definition may not be a good choice for defining an attribute reduct in decision-theoretic rough set models. In the rest of the paper, we will introduce a new quantitative criterion that is easier to understand and interpret than those region based definitions.

4. Minimum cost attribute reduction in decision-theoretic rough set models

In this section, we construct an optimization problem first, and then define the minimum cost attribute reduct. Some remarks on the optimization problem and the definition are also given. A heuristic approach and two evolutionary approaches are proposed for minimum cost attribute reduction.

4.1. An optimization problem

The optimization objective is to minimize the decision cost. By reviewing the decision-theoretic rough set model, we find that Bayesian decision procedure deals with making decisions with minimum cost based on observed evidence. A change of three regions after removing some attributes leads to a change of decision cost. According to the Bayesian decision principle, the optimization objective is to minimize the decision cost. By reviewing the decision-theoretic rough set model, we find that Bayesian decision procedure deals with making decisions with minimum cost based on observed evidence. A change of regions in Table 3 is a decision table generated from Table 1 by removing \( c_4, c_5 \) and \( c_6 \). Assuming the cost functions are set the same as in earlier Examples. Then we have \( \text{POS}_{a}\left(\pi_D|\pi_c|c_1\right) = \{a_5\} \), \( \text{BND}_{a}\left(\pi_D|\pi_c|c_2\right) = \{c_2\} \), \( \text{NEG}_{a}\left(\pi_D|\pi_c|c_1\right) = \{a_5, a_4\} \), and \( \text{COST}_{c_1-c_2} = 13 \). The decision cost based on \( c_1 \) is: \( \text{COST}_{c_1} = 12 \) and the cost based on \( c_2 \) is: \( \text{COST}_{c_2} = 15 \).

Example 7. Table 3 is a decision table generated from Table 1 by removing \( c_4, c_5 \) and \( c_6 \). Assuming the cost functions are set the same as in earlier Examples. Then we have \( \text{POS}_{a}\left(\pi_D|\pi_c|c_1\right) = \{a_5\} \), \( \text{BND}_{a}\left(\pi_D|\pi_c|c_2\right) = \{c_2\} \), \( \text{NEG}_{a}\left(\pi_D|\pi_c|c_1\right) = \{a_5, a_4\} \), and \( \text{COST}_{c_1-c_2} = 13 \). The decision cost based on \( c_1 \) is: \( \text{COST}_{c_1} = 12 \) and the cost based on \( c_2 \) is: \( \text{COST}_{c_2} = 15 \).

From Example 7 we can see that the decision cost is decreased after we remove an attribute. Based on the example, we introduce a decreasing cost attribute reduct.

Definition 9. In a decision table \( S = (U, \text{At} = C \cup \{D\}, \{V_a\}, \{I_a\}) \), \( R \subseteq C \) is a decreasing cost attribute reduct if and only if

1. \( \text{COST}_R \leq \text{COST}_C \)
2. \( \forall R' \subset R, \text{COST}_{R'} > \text{COST}_R \)

<table>
<thead>
<tr>
<th>Table 3</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Another decision table.</strong></td>
</tr>
<tr>
<td>( c_1 )</td>
</tr>
<tr>
<td>( o_1 )</td>
</tr>
<tr>
<td>( o_2 )</td>
</tr>
<tr>
<td>( o_3 )</td>
</tr>
<tr>
<td>( o_4 )</td>
</tr>
<tr>
<td>( o_5 )</td>
</tr>
<tr>
<td>( o_6 )</td>
</tr>
<tr>
<td>( o_7 )</td>
</tr>
<tr>
<td>( o_8 )</td>
</tr>
<tr>
<td>( o_9 )</td>
</tr>
</tbody>
</table>
In this definition, we want to find a subset of \( C \) so that the decision cost will be decreased or unchanged based on the reduct. In most situation, it is better for users to obtain a smaller or the smallest cost in the decision procedure. We propose an optimization problem with the objective of minimizing the value of the cost [8], denoted as:

\[
\min \text{COST}.
\]

Then the optimization problem is described as finding a proper attribute set to make the whole decision cost minimum. The objective of attribute reduction is to minimize the decision cost. Based on that, a minimum cost attribute reduct is defined.

**Definition 10.** In a decision table \( S = (U,At = C \cup \{D\}, \{V_a\}, \{I_a\}, R \subseteq C \) is a minimum cost attribute reduct if and only if

1. \( R = \arg \min_{R \subseteq C} C(\text{COST}_R) \).
2. \( \forall R' \subseteq R, \ \text{COST}_{R'} > \text{COST}_R \).

In this definition, condition (1) is the jointly sufficient condition and condition (2) is the individual necessary condition. Condition (1) guarantees that the cost induced from the reduct is minimal, and condition (2) guarantees that the reduct is minimal.

**Example 8.** For Table 1, assuming all cost functions are given as: \( \lambda_{PP} = \lambda_{NN} = 0, \lambda_{PN} = 6, \lambda_{NP} = 3, \lambda_{BP} = 1, \lambda_{BN} = 3 \), then we can get the two thresholds \( \alpha = 0.75 \) and \( \beta = 0.6 \). The cost based on the whole condition attribute set \( C \) is 8, and we can get 6 reducts according to Definition 10: \( \{c_1, c_2, c_5\}, \{c_2, c_3, c_5\}, \{c_4, c_5, c_6\}, \{c_2, c_4, c_5\}, \{c_2, c_5, c_6\}, \{c_2, c_5, c_6\} \).

Compared to Example 2, the result in Example 8 is same. In Examples 3 and 5, although the positive region based on \( \{c_2, c_3\} \) is larger than that based on \( C \), the cost induced from \( \{c_2, c_3\} \) is 9, and \( \{c_2, c_5\} \) is not a reduct according to our definition. In Example 6, the non-negative region based on \( \{c_2\} \) is \( U \), but the cost derived from that is 15, and \( \{c_2\} \) is not a reduct by our definition.

4.2. A meaningful explanation of the definition

In Example 7, the three regions of the table before applying attribute reduction are \( \text{POS}_{x,y}(\{x\}) = \{o_1, o_2\} \), \( \text{BND}_{x,y}(\{x\}) = \{o_1, o_5, o_6, o_1, o_6, o_9\} \), and \( \text{NEG}_{x,y}(\{x\}) = \{o_1, o_2\} \). \( \text{COST}_{\{c_1, c_2\}} = 13 \). According to the definition of the minimum cost attribute reduct, \( \{c_1\} \) is a reduct of the decision table, and the regions after the reduction become: \( \text{POS}_{x,y}(\{x\}) = \text{BND}_{x,y}(\{x\}) = \emptyset \), \( \text{NEG}_{x,y}(\{x\}) = \emptyset \).

In this example, all three regions of the decision table are changed by a reduct. The cost of rejecting all objects is smaller than the cost of accepting some objects in Table 3. It is no longer better for users to obtain larger positive region or non-negative region here. Instead, the smaller of the cost the better.

We can interpret our definition by investigating the goal of attribute reduction. In Pawlak rough set model, attribute reduction can be understood as a procedure of removing redundant attributes from the original set of attributes. The redundant attributes are those attributes whose removal will not affect the performance of a criterion. In probabilistic rough set models, besides the redundant attributes, there also exist some attributes whose removal will increase the performance. It is not proper to call these attributes redundant attributes, so we call them irrelevant attributes and they can be seen as noisy attributes. The goal of reduction is to remove both irrelevant attributes and redundant attributes in order to improve the performance. In Example 7, \( c_2 \) is an irrelevant attribute, so we can obtain a smaller cost value by deleting it. A meaningful explanation of minimum cost attribute reduction is to remove the redundant attributes with no increasing the decision cost and the irrelevant attributes with decreasing the decision cost.

4.3. Remarks on the definition

4.3.1. Non-monotonicity property of the decision cost

A very important property of Definition 10 is the non-monotonicity property of decision cost with respect to attribute set. We can re-express the cost formulation as following:

\[
\text{COST} = \text{COST}_{\text{POS}} + \text{COST}_{\text{BND}} + \text{COST}_{\text{NEG}}.
\]

The semantic of Formulas (15) is that the cost is composed of three parts: cost of positive region, cost of boundary region and cost of negative region. Fig. 1 is used to show the relationship between the decision cost and the probabilities of objects.

In Fig. 1, the Y-axis represents the decision cost and the X-axis represents probabilities of objects. According to decision cost (Eq. (9)), we know that the decision cost is linearly with the probability of an object in each region. In the positive region (illustrated by \([0, \beta]\)), the decision cost increases monotonically with increasing the probability of an object; in the positive region \( [x, 1] \), the decision cost decreases monotonically with increasing the probability of an object; in the boundary region \( ([\beta, x]) \), the cost value may increase, decrease or keep unchanged with increasing the probability of an object. For the boundary region, when condition \( \frac{1-\alpha}{1-\beta} > \frac{\alpha}{\beta} \) is satisfied, the cost value decreases monotonically, illustrated by Fig. 1a; when
condition $\frac{1-x}{x} < \frac{1-y}{y}$ is satisfied, the cost value increases monotonically, illustrated by Fig. 1b; when condition $\frac{1-x}{x} = \frac{1-y}{y}$ is satisfied, the cost value keeps unchanged, illustrated by Fig. 1c.

Now we use Fig. 1c to explain why the condition $\frac{1-x}{x} = \frac{1-y}{y}$ is used. For the boundary region, the decision cost keeps unchanged when probabilities of objects change, that means the cost value on probability value $\beta$ is equivalent to the cost value on probability value $\alpha$. According to Eq. (9), the following equation holds: $(1-\alpha) \cdot \lambda_{PN} = \beta \cdot \lambda_{NB}$. Assuming the cost of correct classification is zero: $\lambda_{PN} = \lambda_{NB} = 0$, we find that $\frac{\alpha}{\beta} = \frac{1}{\gamma}$, then the condition for Fig. 1c is easy to be represented by $\frac{1-x}{x} = \frac{1-y}{y}$.

Similarly, it is easy to derive the constrain conditions in Fig. 1a and in Fig. 1b.

The monotonicity property of the positive and negative regions shows that the cost value is small when the probability of an object is close to 0 or 1; the cost value is big when the probability of an object is close to $\beta$ or $\alpha$. That means higher certainty with lower cost and higher uncertainty with higher cost, which is consistent with our intuitive understanding.

For the cost of a decision table, the monotonicity property does not hold. Following the change of the set of the condition attributes, the probabilities of objects are changed and the values of them are discrete. From the optimization viewpoint, the optimization problem between the decision cost and the probabilities of objects is a combinational optimization problem.

**Example 9.** In a decision table, the set of condition attributes is $C = \{a, b, c\}$ and the decision attribute is $D$, $U = \{x_1, \ldots, x_6\}$. Assuming the partition of $U$ under $C$ is $U/P_C = \{x_1, \ldots, x_6\}$, and the partition of $U$ under $D$ is $U/P_D = \{x_1, x_2, x_3, x_4, x_5, x_6\}$, then the probabilities of all objects based on $C$ are same one: $p_1 = \frac{1}{2}$. Assuming a new condition attribute $d$ arrives, the partition of $U$ under $C \cup \{d\}$ becomes $U/P_{C \cup \{d\}} = \{x_1, x_2, x_3, x_4\}$, then the probabilities of all objects are changed as: $p_1 = p_2 = p_3 = p_4 = 1$, $p_5 = p_6 = \frac{1}{2}$.

Based on above example, we can conclude that the probability of an object could increase or decrease when adding a condition attribute. Similar examples can be found for removing a condition attribute. Following the change of the probability, the object could be classified into different regions, and the associated cost could be larger or smaller. The monotonicity property between the decision cost and the set of condition attributes does not hold in decision-theoretic rough set models.

An important implication of the non-monotonicity property is that we need to check all subsets of the reduct ($\forall R \subseteq R$) instead of parts of it ($R - \{a\}$).

### 4.3.2. Bounds of the cost value

According to the cost formula (Eq. (9)) and Fig. 1, we can see that there exist a upper bound and a lower bound for the decision cost. The bounds are related to the thresholds $\alpha$ and $\beta$, or the cost functions $\lambda_{PN}$, $\lambda_{NB}$, $\lambda_{BP}$.

Let $n_p$, $n_B$ and $n_N$ denote the number of the objects being classified into the positive region, boundary region and negative region respectively. For Fig. 1a, the bounds of the decision cost are:

$$0 \leq \text{COST} \leq n_p \cdot (1-\alpha) \cdot \lambda_{PN} + (n_B + n_N) \cdot \beta \cdot \lambda_{NB},$$

or $\lambda_{NB}$ can be replaced by $\gamma$:

$$0 \leq \text{COST} \leq n_p \cdot (1-\alpha) \cdot \lambda_{PN} + (n_B + n_N) \cdot \beta \cdot \frac{1-\gamma}{\gamma} \cdot \lambda_{PN}.$$

For Fig. 1b, the bounds of the cost are:

$$0 \leq \text{COST} \leq (n_p + n_B) \cdot (1-\alpha) \cdot \lambda_{PN} + n_N \cdot \beta \cdot \lambda_{NB},$$

or

$$0 \leq \text{COST} \leq (n_p + n_B) \cdot (1-\alpha) \cdot \lambda_{PN} + n_N \cdot \beta \cdot \frac{1-\gamma}{\gamma} \cdot \lambda_{PN}.$$
For Fig. 1c, we can use the cost bounds of Fig. 1a or the bounds of Fig. 1b to represent, because of the constraint condition \( \frac{1}{\gamma} = \frac{1-\beta}{\gamma} \).

### 4.3.3. Generalization of the optimization problem

From the formulation of the decision cost, we can see that the cost of each object is decided by the thresholds and the probability of the object. Consider the next two situations: first, if the values of \( \lambda_{\text{BN}} \) and \( \lambda_{\text{BP}} \) are very small, the value of \( \text{COST}_{\text{BND}} \) is small too, and more objects will be classified into the boundary region with unchanged or smaller decision cost. Second, for a same decision table, it is assumed that a user prefers the positive rules and negative rules, which means he wants to make a direct decision; another user may prefer positive rules and boundary rules. For these two situations, we propose a generalization of the optimization problem, which will help users get a proper result. In the optimization problem, the decision cost is re-defined as follows.

\[
\text{COST} = \epsilon_{p} \cdot \sum_{j=1}^{p} (1 - p_{j}) \cdot \lambda_{\text{BN}} + \epsilon_{b} \cdot \sum_{b=p+1}^{2} (1 - p_{j}) \cdot \lambda_{\text{BP}} + \epsilon_{N} \cdot \sum_{b=p+1}^{2} p_{b} \cdot \lambda_{\text{NP}},
\]

where \( \epsilon_{p} \), \( \epsilon_{b} \) and \( \epsilon_{N} \) denote the penalty of each kind of rules, respectively. If we set \( \epsilon_{p} = 1 \) and \( \epsilon_{b} = 1 \), a solution of the optimization problem prefers to use less boundary rules. More settings can be discussed for other applications. We need to perform experiments to decide the exact values of the three penalty functions.

**Example 10.** For Table 1, we still use the same cost functions as in earlier examples. Assuming a user does not want to make too many boundary rules and negative rules, and we set the penalty of boundary rules to be 1.5, the penalty of negative rules to be 2. That means, for Eq. (16), \( \epsilon_{p} = 1 \), \( \epsilon_{b} = 1.5 \) and \( \epsilon_{N} = 2 \). The cost of the decision table based on \( C \) is 13.5 now. By checking the subset \( \{C_{2}, C_{3}\} \), we find that the cost induced is 12, which is the smallest value, and \( \{C_{2}, C_{3}\} \) is a reduct of the decision table.

This result is different from the result in Example 8. It is easy to interpret the difference by reviewing the regions induced by the reduct. The regions based on \( C \) are \( \text{POS}_{(x, y)}(\pi_{D} | \pi_{C}) = \{o_{1}, o_{3}, o_{4}, o_{7}\} \), \( \text{BND}_{(x, y)}(\pi_{D} | \pi_{C}) = \{o_{2}, o_{6}, o_{8}\} \), and \( \text{NEG}_{(x, y)}(\pi_{D} | \pi_{C}) = \{o_{5}, o_{9}\} \). The regions based on \( \{C_{2}, C_{3}\} \) are \( \text{POS}_{(x, y)}(\pi_{D} | \pi_{(C_{2}, C_{3})}) = \{o_{1}, o_{2}, o_{3}, o_{4}, o_{5}, o_{6}, o_{7}, o_{8}\} \), \( \text{BND}_{(x, y)}(\pi_{D} | \pi_{(C_{2}, C_{3})}) = \emptyset \), and \( \text{NEG}_{(x, y)}(\pi_{D} | \pi_{(C_{2}, C_{3})}) = \{o_{5}, o_{9}\} \). Using less boundary rules and negative rules leads to more positive rules. The positive region based on \( \{C_{2}, C_{3}\} \) is larger than that based on \( C \), and the boundary region based on \( \{C_{2}, C_{3}\} \) is smaller and the negative region is unchanged. We can conclude that the result is satisfied with the user’s preference.

#### 4.4. Reduction approaches

From the optimization problem and the minimum cost attribute reduct, we can see that finding a reduct is actually the procedure of solving the optimization problem. The optimization problem is a combinational problem and it is not easy to get the optimal solution in a linear time. So we use heuristic or randomized approaches to the approximate optimal solution. In this paper, we will present three approaches, including a heuristic approach, a genetic approach and a simulated annealing approach.

#### 4.4.1. A heuristic approach to attribute reduction

For the heuristic algorithm, Yao et al. [40] have summarized three kind of strategies: the deletion strategy, the addition-deletion strategy, and the addition strategy. Our heuristic approach employs addition strategy to complete an approximate reduct in this paper. The reduct is approximate as it may not satisfy the individually necessary condition. The basic idea of our heuristic approach is constructing a reduct from an empty set by adding condition attributes until it becomes an approximate reduct.

Condition attributes are selected by their fitness functions, while the fitness function of attribute \( c_{i} \) is defined as:

\[
\delta_{i} = \frac{\text{COST}_{C - \{c_{i}\}} - \text{COST}_{C}}{\text{COST}_{C}}.
\]

**Input:** A decision table.

**Output:** A reduct \( R \).

**Algorithm:**

BEGIN

\( R = \emptyset \); \( G = C \);

compute fitness of all the attributes in \( G \) using the fitness function \( \delta_{i} \);

WHILE \( \text{COST}_{R} > \text{COST}_{C} \) and \( G \neq \emptyset \)

select an attribute \( a \in G \) according to \( \delta_{a} \), let \( G = G - \{a\} \);

\( R = R \cup \{a\} \);

END WHILE

output \( R \);

END BEGIN

**Fig. 2.** A heuristic approach to minimum cost attribute reduction.
The fitness function represents the significance of an attribute. The value of the fitness function could be a minus one, which means it is an irrelevant attribute. If the value of the fitness function is zero, the attribute is a redundant attribute.

In the procedure of adding attributes, if $COST_R < COST_C$, we will stop the adding procedure and output $R$ as an approximate reduct. The heuristic approach is described in Fig. 2.

### 4.4.2. A genetic approach to attribute reduction

Many optimization algorithms have been applied in attribute reduction, such as genetic algorithms [2,7], simulated annealing algorithms [1,7], ant colony algorithms [6,10], and so on [28]. We will briefly introduce how to apply a genetic algorithm and a simulated annealing algorithm to the minimum cost attribute reduction.

For genetic approach, a chromosome is represented as a binary string of length $M$ which is the number of condition attributes, where “1” means that the corresponding attribute is present, and “0” shows that the corresponding attribute is not.

The fitness function is defined as:

$$f = COST_R + \left( \frac{|R|}{|C|} \right)^\theta.$$  

The goal of the fitness function is to find such an attribute subset that has minimal decision cost and fewer elements. The selection method in our genetic approach is the roulette wheel selection. In the roulette wheel selection, individuals in the population are assigned probabilities of being selected that is directly proportionate to their fitness.

For the parameters setting, $\theta$ in the fitness function is set to 3 in our experiments. In the crossover procedure, we use the uniform crossover with the mixing ratio 0.3. The uniform crossover evaluates each bit in the parent strings for exchanging with a probability 0.3. The mutation operation is applied with the probability 0.1.

The genetic approach is described in Fig. 3.

### 4.4.3. A simulated annealing approach to attribute reduction

For simulated annealing approach, we use a binary representation as the solution representation. The fitness function $f$ is same as Formulas (18) in the genetic approach.

For the neighborhood structure, we use $N(X)$ to represent the next solution generated based on the current solution $X$: select two attributes at random and change the values of them to either 0 or 1 based on their current values.

About the cooling schedule, we set the initial temperature as follows [13]:

$$t_0 = \frac{f_{\text{min}} - f_{\text{max}}}{\ln P_0}.$$  

$P_0 = 0.5$ is the initial accepted threshold. At the beginning, a set of solutions $\{X_1, X_2, \ldots, X_L\}$ are generated at random. $f_{\text{min}}$ is the minimal one of their fitness function values, and $f_{\text{max}}$ is the maximum one. In our experiments, $L$ is set to 20. The temperature is linearly reduced based on the following formula:

$$t_{i+1} = r \cdot t_i,$$  

where $i$ is the iterator number and $r$ is a constant to update the temperature which is set to 0.93 in our experiments.

For a solution $X$, if its neighbor solution $N(X)$ is less than its fitness value, we accept the move from $X$ to $N(X)$. The simulated annealing approach also probably accepts a worse solution: $e^{-\frac{f_{\text{max}} - f_X}{t_i}} > \rho$, where $\rho$ is a random number between 0 and 1.

The basic description of the simulated annealing algorithm is shown in Fig. 4.

---

**Input**: A decision table.  
**Output**: A reduct $R$.

BEGIN  
create an initial random population(number=40);  
evaluate the population;  
WHILE the result is not convergent AND number of generations < 100  
select the fittest individuals in the population;  
perform crossover on the selected individuals to create offspring;  
perform mutation on the selected individuals;  
create the new population;  
evaluate the new population;  
END WHILE  
selected the fittest individual from current population and output as $R$;  
END BEGIN

---

Fig. 3. A genetic approach to minimum cost attribute reduction.
5. Experiments

In this section, we will show the efficiency of the minimum cost attribute reduction by several experiments. Two other definitions of attribute reducts are used to compare with our definition. One is a positive region extension based attribute reduct of Definition 6. The other one is a non-negative region extension based attribute reduct of Definition 7.

For each definition, three kinds of approaches, including the heuristic approach, the genetic approach and the simulated annealing approach, are implemented. For regions extension based definitions, the fitness function is modified accordingly.

There are 12 UCI data sets [45] used in our experiments. Each data set has two classes. Information of data sets are summarized in Table 4.

For each data set, the cost functions are randomly generated. Their values are in (0,1) with following constraint conditions:

\[ \lambda_{BP} < \lambda_{BN}, \lambda_{BN} < \lambda_{PN}, \lambda_{BP} = \lambda_{NN} = 0. \]

In the experiments, 10-fold cross validation are employed, and average results are recorded. Note that for each data set, 10 different group of cost functions are randomly generated.

Table 4
Brief description of the data sets.

<table>
<thead>
<tr>
<th>Data sets</th>
<th># of objects</th>
<th># of condition attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cylinder bands(bands)</td>
<td>512</td>
<td>39</td>
</tr>
<tr>
<td>Credit approval(credit)</td>
<td>690</td>
<td>15</td>
</tr>
<tr>
<td>Hepatitis</td>
<td>155</td>
<td>19</td>
</tr>
<tr>
<td>Ionosphere</td>
<td>351</td>
<td>34</td>
</tr>
<tr>
<td>Monks-1</td>
<td>124</td>
<td>6</td>
</tr>
<tr>
<td>Monks-2</td>
<td>169</td>
<td>6</td>
</tr>
<tr>
<td>Monks-3</td>
<td>122</td>
<td>6</td>
</tr>
<tr>
<td>Musk version 1(musk)</td>
<td>476</td>
<td>168</td>
</tr>
<tr>
<td>Blood transfusion service center(transfusion)</td>
<td>748</td>
<td>5</td>
</tr>
<tr>
<td>Wisconsin diagnostic breast cancer (wdbc)</td>
<td>569</td>
<td>30</td>
</tr>
<tr>
<td>Wisconsin prognostic breast cancer (wpbc)</td>
<td>198</td>
<td>33</td>
</tr>
<tr>
<td>Congressional voting records (voting)</td>
<td>435</td>
<td>16</td>
</tr>
</tbody>
</table>

Table 5
Average length of a reduct based on heuristic approach.

<table>
<thead>
<tr>
<th>Data sets</th>
<th>Cost-based</th>
<th>Positive-based</th>
<th>Nonnegative-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bands</td>
<td>2.0 ± 0.0</td>
<td>1.8 ± 0.4</td>
<td>1.3 ± 0.4</td>
</tr>
<tr>
<td>Credit</td>
<td>12.0 ± 0.0</td>
<td>10.0 ± 4.8</td>
<td>5.6 ± 5.6</td>
</tr>
<tr>
<td>Hepatitis</td>
<td>11.0 ± 0.0</td>
<td>6.0 ± 5.0</td>
<td>4.0 ± 4.5</td>
</tr>
<tr>
<td>Ionosphere</td>
<td>10.0 ± 0.0</td>
<td>2.8 ± 3.6</td>
<td>1.0 ± 0.0</td>
</tr>
<tr>
<td>Monks-1</td>
<td>3.0 ± 0.0</td>
<td>2.4 ± 0.9</td>
<td>2.0 ± 1.0</td>
</tr>
<tr>
<td>Monks-2</td>
<td>6.0 ± 0.0</td>
<td>4.0 ± 2.4</td>
<td>2.0 ± 2.0</td>
</tr>
<tr>
<td>Monks-3</td>
<td>4.0 ± 0.0</td>
<td>2.1 ± 1.3</td>
<td>1.4 ± 0.9</td>
</tr>
<tr>
<td>Musk</td>
<td>1.0 ± 0.0</td>
<td>1.0 ± 0.0</td>
<td>1.0 ± 0.0</td>
</tr>
<tr>
<td>Transfusion</td>
<td>3.0 ± 0.0</td>
<td>12.0 ± 0.6</td>
<td>1.2 ± 0.6</td>
</tr>
<tr>
<td>Wdbc</td>
<td>11.0 ± 0.0</td>
<td>5.0 ± 4.8</td>
<td>3.0 ± 4.0</td>
</tr>
<tr>
<td>Wpbc</td>
<td>7.0 ± 0.0</td>
<td>4.0 ± 3.0</td>
<td>3.4 ± 2.9</td>
</tr>
<tr>
<td>Voting</td>
<td>13.0 ± 0.0</td>
<td>2.2 ± 3.6</td>
<td>1.0 ± 0.0</td>
</tr>
<tr>
<td>Average</td>
<td>7.5 ± 0.0</td>
<td>4.4 ± 2.5</td>
<td>2.3 ± 1.8</td>
</tr>
</tbody>
</table>

Fig. 4. A simulated annealing approach to minimum cost attribute reduction.
The length of the reduct and the running time are usually used as comparison criteria. Since it is not easy to evaluate the running time of randomized algorithms correctly, we do not record the running time. We incorporate a new measure to evaluate the performance of the attribute reduction approaches by comparing the misclassification cost. A decision-theoretic rough set model can be seen as a kind of cost-sensitive learning model. For the cost-sensitive learning problem, the misclassification cost is a more reasonable measure than standard error rates [3]. The misclassification cost used in our experiments is defined as:

\[ mc = \lambda_{PN} \cdot n_{PN} + \lambda_{NP} \cdot n_{NP}, \]

where \( n_{PN} \) and \( n_{NP} \) denote the numbers of misclassification objects. The misclassification cost is different from the decision cost defined in this paper. There are two reasons why we do not use decision cost as the experimental measure. One is that

**Table 6**

<table>
<thead>
<tr>
<th>Data sets</th>
<th>Cost-based</th>
<th>Positive-based</th>
<th>Nonnegative-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bands</td>
<td>88.56 ± 62.64</td>
<td>88.56 ± 62.64</td>
<td>88.56 ± 62.64</td>
</tr>
<tr>
<td>Credit</td>
<td>49.12 ± 20.06</td>
<td>62.50 ± 42.55</td>
<td>106.00 ± 68.93</td>
</tr>
<tr>
<td>Hepatitis</td>
<td>11.99 ± 6.59</td>
<td>11.77 ± 6.26</td>
<td>11.91 ± 6.37</td>
</tr>
<tr>
<td>Ionosphere</td>
<td>18.68 ± 8.44</td>
<td>44.52 ± 25.03</td>
<td>45.87 ± 23.04</td>
</tr>
<tr>
<td>Monks-1</td>
<td>5.81 ± 3.93</td>
<td>9.78 ± 10.79</td>
<td>14.58 ± 13.89</td>
</tr>
<tr>
<td>Monks-3</td>
<td>4.59 ± 1.50</td>
<td>13.87 ± 11.87</td>
<td>17.58 ± 12.18</td>
</tr>
<tr>
<td>Musk</td>
<td>7.39 ± 11.29</td>
<td>7.39 ± 11.29</td>
<td>7.39 ± 11.29</td>
</tr>
<tr>
<td>Transfusion</td>
<td>85.67 ± 32.72</td>
<td>87.84 ± 36.05</td>
<td>87.64 ± 36.05</td>
</tr>
<tr>
<td>Wdbc</td>
<td>22.84 ± 14.75</td>
<td>25.17 ± 16.64</td>
<td>27.82 ± 19.62</td>
</tr>
<tr>
<td>Wpbc</td>
<td>24.13 ± 13.21</td>
<td>24.13 ± 13.21</td>
<td>24.13 ± 13.21</td>
</tr>
<tr>
<td>Voting</td>
<td>10.14 ± 2.76</td>
<td>34.43 ± 14.36</td>
<td>44.40 ± 19.58</td>
</tr>
<tr>
<td>Average</td>
<td>29.35 ± 16.18</td>
<td>36.02 ± 22.18</td>
<td>41.42 ± 25.08</td>
</tr>
</tbody>
</table>

**Table 7**

<table>
<thead>
<tr>
<th>Data sets</th>
<th>Cost-based</th>
<th>Positive-based</th>
<th>Nonnegative-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bands</td>
<td>5.2 ± 1.7</td>
<td>3.4 ± 1.8</td>
<td>4.5 ± 1.3</td>
</tr>
<tr>
<td>Credit</td>
<td>3.3 ± 1.6</td>
<td>2.6 ± 1.8</td>
<td>1.5 ± 1.5</td>
</tr>
<tr>
<td>Hepatitis</td>
<td>4.4 ± 0.7</td>
<td>2.5 ± 1.3</td>
<td>1.5 ± 1.2</td>
</tr>
<tr>
<td>Ionosphere</td>
<td>4.5 ± 0.9</td>
<td>4.0 ± 1.6</td>
<td>3.0 ± 1.0</td>
</tr>
<tr>
<td>Monks-1</td>
<td>1.2 ± 0.4</td>
<td>1.8 ± 0.9</td>
<td>1.4 ± 0.6</td>
</tr>
<tr>
<td>Monks-2</td>
<td>1.1 ± 0.3</td>
<td>2.5 ± 2.2</td>
<td>1.5 ± 1.5</td>
</tr>
<tr>
<td>Monks-3</td>
<td>1.6 ± 0.6</td>
<td>1.7 ± 1.0</td>
<td>1.3 ± 0.9</td>
</tr>
<tr>
<td>Musk</td>
<td>43.8 ± 3.8</td>
<td>46.6 ± 3.4</td>
<td>50.3 ± 7.4</td>
</tr>
<tr>
<td>Transfusion</td>
<td>1.0 ± 0.0</td>
<td>1.0 ± 0.0</td>
<td>1.1 ± 0.3</td>
</tr>
<tr>
<td>Wdbc</td>
<td>5.5 ± 1.6</td>
<td>3.4 ± 1.9</td>
<td>2.9 ± 1.0</td>
</tr>
<tr>
<td>Wpbc</td>
<td>5.1 ± 0.3</td>
<td>3.7 ± 1.1</td>
<td>2.9 ± 1.5</td>
</tr>
<tr>
<td>Voting</td>
<td>2.0 ± 1.0</td>
<td>1.8 ± 1.4</td>
<td>1.6 ± 0.9</td>
</tr>
<tr>
<td>Average</td>
<td>6.5 ± 1.0</td>
<td>6.2 ± 1.5</td>
<td>6.1 ± 1.5</td>
</tr>
</tbody>
</table>

**Table 8**

<table>
<thead>
<tr>
<th>Data sets</th>
<th>Cost-based</th>
<th>Positive-based</th>
<th>Nonnegative-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bands</td>
<td>54.44 ± 44.18</td>
<td>55.38 ± 45.47</td>
<td>54.28 ± 45.48</td>
</tr>
<tr>
<td>Credit</td>
<td>42.72 ± 16.44</td>
<td>48.86 ± 29.70</td>
<td>66.08 ± 54.38</td>
</tr>
<tr>
<td>Hepatitis</td>
<td>13.28 ± 6.54</td>
<td>14.07 ± 7.79</td>
<td>14.47 ± 7.46</td>
</tr>
<tr>
<td>Ionosphere</td>
<td>25.05 ± 16.51</td>
<td>25.03 ± 14.87</td>
<td>27.49 ± 16.71</td>
</tr>
<tr>
<td>Monks-1</td>
<td>10.73 ± 5.16</td>
<td>14.46 ± 11.03</td>
<td>17.57 ± 9.72</td>
</tr>
<tr>
<td>Monks-2</td>
<td>29.00 ± 20.00</td>
<td>29.48 ± 19.64</td>
<td>29.02 ± 19.18</td>
</tr>
<tr>
<td>Monks-3</td>
<td>7.13 ± 5.32</td>
<td>13.56 ± 9.74</td>
<td>20.45 ± 11.24</td>
</tr>
<tr>
<td>Musk</td>
<td>44.90 ± 25.09</td>
<td>32.49 ± 31.78</td>
<td>39.85 ± 32.71</td>
</tr>
<tr>
<td>Transfusion</td>
<td>90.92 ± 48.54</td>
<td>89.78 ± 43.98</td>
<td>91.42 ± 46.29</td>
</tr>
<tr>
<td>Wdbc</td>
<td>23.87 ± 12.55</td>
<td>32.04 ± 18.75</td>
<td>39.04 ± 30.16</td>
</tr>
<tr>
<td>Wpbc</td>
<td>23.37 ± 13.82</td>
<td>23.37 ± 13.82</td>
<td>23.37 ± 13.82</td>
</tr>
<tr>
<td>Voting</td>
<td>7.47 ± 2.12</td>
<td>10.79 ± 7.72</td>
<td>16.94 ± 17.96</td>
</tr>
<tr>
<td>Average</td>
<td>31.07 ± 18.35</td>
<td>32.44 ± 21.19</td>
<td>36.67 ± 25.42</td>
</tr>
</tbody>
</table>
our attribute reduct is minimum cost attribute reduct, and it can beat other definitions on the decision cost measure apparently. The other one is that most classifiers do not give three-way decisions results, in which there is no boundary cost, and the decision cost is not necessarily considered.

In our experiments, cost-sensitive classification method is applied by using C4.5 as the base classifier. All approaches are implemented based on WEKA [4]. The results are shown in the following tables.

Table 5, 7, and 9 show the average length of the derived reduct based on different approaches. For the simulated annealing approach, our definition of attribute reduct produces a shorter result than other two definitions, but for the heuristic approach and the genetic approach, non-negative region extension based definition shows the shortest result.

Table 6, 8, and 10, show the misclassification cost based on reducts derived from different approaches. For the heuristic approach, we can see that, in 12 data sets, minimum cost reduction can get 10 best results, and both positive region extension based definition and non-negative region extension based definition get 4 best results. For the genetic approach, the number of best results from minimum cost reduction is 8, the number from positive region extension based definition is 4, and the number from non-negative region extension based definition is 2. For the simulated annealing approach, the corresponding numbers are respectively 9, 2 and 3. The results show that minimum cost attribute reduction can achieve a better performance.

From the experimental results, we can conclude that minimum cost attribute reduction is a better choice for decision-theoretic rough set model, as the minimum cost is intuitively the objective of the cost-sensitive learning problem.

6. Conclusion and further work

We study several definitions of attribute reducts in decision-theoretic rough set models. Qualitative definitions and quantitative definitions are categorized. By reviewing these different definitions and decision-theoretic rough set models, we propose an optimization problem and a minimum cost attribute reduction, which are used to find the minimal subset of condition attributes. The decision cost induced from the reduct is minimum. In probabilistic rough models, different from regions based attribute reductions, minimum cost attribute reduction can avoid the interpretation difficulties caused by the non-monotonicity property of criteria for defining reducts. The properties of the new definition are also discussed.
A heuristic approach, a genetic approach and a simulated annealing approach for attribute reduction are studied in this paper. Experimental results show that minimum cost attribute reduction achieve better result. The minimum cost attribute reduction is a reasonable and meaningful choice for decision-theoretic rough set models. This is consistent with the fact that a decision-theoretic rough set model can be seen as a cost-sensitive learning method.

The contribution of this paper is an introduction of the optimization view and the cost-sensitive learning view of decision-theoretic rough set models. The results can increase our understanding of decision-theoretic rough set models and its applications. We will further study decision-theoretic rough set models from the cost-sensitive learning view, such as how to deal with the imbalanced data by using the model directly, how to compute the cost functions by using sampling methods, and so on.

Acknowledgement

This work is supported by the National Natural Science Foundation of China under Grant Nos. 61035003 and 61170180.

References