



# Enhanced Rule Generation in Product Design Through Rough Set Theory and Ant Colony Optimization

Xianwei Wang<sup>✉</sup>, Tichun Wang<sup>\*</sup>

College of Mechanical and Electrical Engineering, Nanjing University of Aeronautics and Astronautics, 210000 Nanjing, China

\* Correspondence: Tichun Wang (wangtichun2010@nuaa.edu.cn)

**Received:** 12-16-2023

**Revised:** 03-10-2024

**Accepted:** 03-17-2024

**Citation:** X. W. Wang and T. C. Wang, “Enhanced rule generation in product design through rough set theory and Ant Colony Optimization,” *Precis. Mech. Digit. Fabr.*, vol. 1, no. 1, pp. 11–23, 2024. <https://doi.org/10.56578/pmdf010102>.



© 2024 by the author(s). Published by Acadlore Publishing Services Limited, Hong Kong. This article is available for free download and can be reused and cited, provided that the original published version is credited, under the CC BY 4.0 license.

**Abstract:** Limitations inherent in conventional rule generation methodologies, particularly concerning knowledge redundancy and efficiency in product design, are addressed through the adoption of a rough set-based approach in this study. An enhancement to the Ant Colony Optimization (ACO) algorithm’s information gain ratio is introduced by integrating a redundancy detection mechanism, which notably accelerates the convergence process. Furthermore, the application of a classification consistency algorithm effectively minimizes the number of attributes, facilitating the extraction of potential associative rules. Comparative validation performed on a public dataset demonstrates that the proposed attribute reduction approach surpasses existing methods in terms of attribute count reduction, reduction rate, and execution time. When applied to an automotive design case study, the approach yields rules with 100% coverage and accuracy, characterized by a reduced average number of attributes per rule. These findings underscore the superiority of the rough set-based methodology in generating product design rules, providing a robust framework that enhances both the precision and applicability of the design process.

**Keywords:** Rough sets; Rule generation; Product design; Ant Colony Optimization (ACO); Attribute reduction

## 1 Introduction

In today’s competitive market environment, the success of product design is directly related to a company’s competitiveness [1]. However, the problems faced in the product design process are increasingly complex, and designers often face challenges such as information overload [2], knowledge fragmentation [3], and decision-making uncertainty [4]. Designers need to make efficient and precise decisions amidst this complexity [5] to meet market demands and maintain competitiveness. In this context, rule generation methods have emerged, allowing designers to discover hidden patterns and associations from design data [6], generate clear design rules, and provide a more comprehensive and scientific basis for decision-making, enabling better handling of complex challenges in the product design process.

Currently, the mainstream rule generation methods include association rule mining [7], neural networks [8], and rough sets [9]. Association rule mining algorithm is the most commonly used rule mining method, often used to discover associations in datasets, with the basic idea of analyzing the associations between item sets in the dataset to find frequent item sets and generate association rules. Wang et al. [10] analyzed the attractiveness factors of product shape features and user satisfaction, using a fuzzy weighted association rule mining method to extract frequent fuzzy weighted association rules from a large historical design database. Zhang et al. [11] proposed a multi-dimensional and multi-objective discrete firefly algorithm, extracting association rules from a large historical design database related to prototype design. Lin et al. [12] used the FP-growth association rule algorithm to mine patterns from the data of the magnetic tile production process, assisting technicians in selecting appropriate process parameters, thus optimizing the production process and improving product quality. Cao [13] from Shandong University applied the Apriori algorithm to operation-level data, mining the association between machining benchmark selection principles, machining methods, and technical requirements, using the discovery of process decision-making rules for shaft-type parts as an example to illustrate the main process of association rule mining. The rule generation method using neural networks generates rules by analyzing aspects such as connection weights, activation function thresholds, and hidden

layer feature learning [14]. Fung et al. [15] used a neural network model to study the relationship between perceived value and product form design elements, using genetic algorithms to extract rules to understand the decision-making process within the neural network, helping product designers better understand key design elements. Shen et al. [16] proposed a method combining local clustering neural networks with the Rulex algorithm, extracting rules from the configuration of product service systems to more effectively design for individual customer needs. Liu et al. [17] proposed an image styling design method based on triangular fuzzy and BP neural networks to solve the problem of poor adaptability between the target product styling and user's fuzzy perceptual imagery, validating the method's effectiveness in quantitative analysis of fuzzy imagery, promoting the transformation of design concepts. The core idea of using rough sets to generate rules is to find rules with practical meaning by appropriately rough partitioning attributes while maintaining classification accuracy, achieving effective description and analysis of datasets. Qin [18] from Hefei University of Technology proposed a rough set-based method for acquiring green design knowledge of products. Using Skowron's difference matrix method for attribute reduction and extracting green design rules through the classification consistency algorithm, ultimately forming a set of green design rules. Wen [19] from Tianjin University, addressing the diversified knowledge needs in cloud design environments, used rough sets as a method for knowledge discovery, extracting design rules from data streams in cloud design environments, improving the reuse efficiency of product design knowledge resources. Kobayashi and Niwa [20] applied rough set theory to process customers' subjective quality perceptions, achieving more customer group-specific car exterior designs through customer grouping and rule extraction. However, these rule generation methods have limitations, such as knowledge redundancy, low efficiency, and poor interpretability, affecting the overall performance of the system. Therefore, there is an urgent need for an innovative rule generation method to improve efficiency and decision-making accuracy.

In response to the deficiencies observed in previous studies, this paper proposes a rough set-based method for generating product design rules. This method improves the attribute reduction of the ACO to enhance search iteration efficiency and reduce redundant attributes, and generates rules based on the classification consistency algorithm. Validation analysis on a public dataset shows that the attribute reduction method proposed in this paper has certain advantages compared to control methods, and its superiority is further validated in an automotive design example.

## 2 Basic Concepts of Rough Sets

**Definition 1** (Rough Description of Product Design Decision System): The product design decision system can be represented as a quadruple  $S = \{U, A, V, f\}$ . Where  $U = \{u_1, u_2, u_3, \dots, u_n\}$  is a non-empty finite set, representing the collection of product design samples, each element representing a record of design knowledge used in executing a design task;  $A$  is a set of attributes, divided into condition attributes  $C$  and decision attributes  $D$ ,  $A = C \cup D$ ,  $C \cap D = \emptyset$ ;  $V$  is a set of attribute values,  $V = \cup V_a$ , where  $V_a$  represents the range of values for attribute  $a \in A$ ;  $f : U \times A \rightarrow V$  is an information function, for  $\forall a \in A, \forall x \in U, f(x, a) \in V_a$ , defining the mapping relationship between each design knowledge record's design knowledge and design target attributes and their values in  $U$ .

**Definition 2** (Indiscernibility Relation of Condition Attributes  $C$  and Decision Attributes  $D$ ): Let  $A$  be the set of product design attributes, define an indiscernibility relation based on attribute  $A$ , denoted as  $IND(A)$ , as shown in Eq. (1):

$$U/A = IND(A) = \{(x, y) \mid (x, y) \in U^2, \forall a \in A (f(x, a) = f(y, a))\} \quad (1)$$

Based on the partitioning of the product design sample set by condition attributes  $C$  and decision attributes  $D$ , they are called the condition class and decision class, respectively, denoted as:

$$U/R_C = \{[x]_C; x \in R_C\} \quad (2)$$

$$U/R_D = \{[x]_D; x \in R_D\} \quad (3)$$

Eqs. (2) and (3) are referred to as the indiscernibility relations determined by  $C$  and  $D$ .

**Definition 3** (Conditional Entropy and Mutual Information): For the information system, define the conditional entropy of condition attributes  $C$  with respect to decision attributes  $D$ , denoted as  $H(D|C)$ , as shown in Eq. (4):

$$H(D|C) = - \sum_{i=1}^n p(C_i) \sum_{j=1}^m p(D_j|C_i) \log_2 p(D_j|C_i) \quad (4)$$

where,  $p(C_i) = \frac{|C_i|}{|U|}$ ,  $p(D_j|C_i) = \frac{|C_i \cap D_j|}{|C_i|}$ .

For information system  $S = \{U, A, V, f\}$ , define the mutual information between condition attributes  $C$  and decision attributes  $D$ , used to measure the degree of correlation between condition and decision attributes. The

greater the mutual information, the greater the degree of correlation, denoted as  $I(C; D)$ , as shown in Eq. (5):

$$I(C; D) = H(D) - H(D | C) \quad (5)$$

**Definition 4** (Positive Region): For information system  $S = \{U, A, V, f\}$ ,  $\forall P, Q \subseteq A$ ; the equivalence relations constructed from  $P$  and  $Q$  are denoted as  $IND(P)$  and  $IND(Q)$ , respectively; then the positive region of  $Q$  with respect to  $P$ , commonly denoted simply as  $POS_P(Q)$ , is shown in Eq. (6):

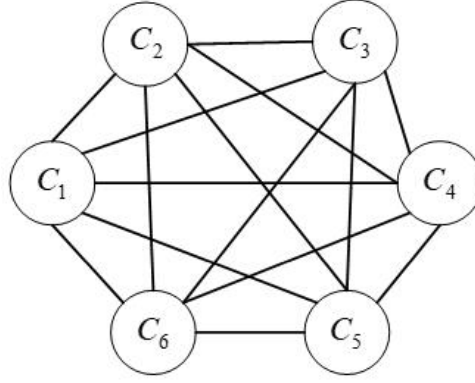
$$POS_P(Q) = \bigcup_{x \in U/Q} IND(P) - (X) \quad (6)$$

The positive region of  $Q$  with respect to  $P$  consists of all the objects in  $U$  that can be accurately classified into the equivalence classes of  $Q$  based on the information from the classification  $U/P$ . It indicates the set of objects that can be correctly classified when the antecedent is  $P$ .

### 3 Rule Generation Based on Rough Sets

#### 3.1 Improved ACO for Attribute Reduction

Firstly, the problem of attribute reduction is transformed into a path optimization problem in the ant colony algorithm, as shown in Figure 1. The nodes represent product design knowledge attributes, and the edges between nodes represent the path for ants to choose the next knowledge attribute. The goal of path optimization is to find a minimal subset of design knowledge attributes that best describes the target attributes of the design.



**Figure 1.** Schematic diagram of problem transformation to path optimization in ant colony algorithm

Pheromones are released by ants as they travel, and they gradually evaporate over time. Therefore, the paths that are frequently traveled by ants have a relatively higher concentration of pheromones. When choosing paths, ants tend to prefer paths with higher pheromone concentrations. Once all ants have completed their search, an iteration is achieved. In the next iteration round, the pheromone concentration is calculated according to Eq. (7):

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \Delta\tau_{ij}(t) \quad (7)$$

where,  $\tau_{ij}(t)$  represents the pheromone concentration between attribute nodes  $i$  and  $j$ ;  $\rho$  represents the pheromone evaporation level,  $0 \leq \rho \leq 1$ ;  $\Delta\tau_{ij}(t)$  represents the total pheromone left by ants on the path from attribute nodes  $i$  to  $j$  during each iteration, as shown in Eq. (8).

$$\Delta\tau_{ij}(t) = \begin{cases} \sum \frac{q}{|R(t)|} & \text{The attribute node } i \text{ to } j \text{ has been visited.} \\ 0 & \text{others} \end{cases} \quad (8)$$

where,  $q$  is a given constant, and  $|R(t)|$  is the number of attributes in the minimal attribute subset during the  $t$ -th iteration.

Using information gain as heuristic information tends to favor attributes with a larger number of values, whereas the information gain ratio as heuristic information prefers attributes with fewer values. This paper proposes an improved information gain ratio as heuristic information, as shown in Eq. (9):

$$\eta_{ij}(t) = \frac{I(B \cup \{a\}; D) - I(B; D)}{H(D | \{a\})} \quad (9)$$

In  $S = \{U, A, V, f\}$ , for  $\forall B \subseteq C, \forall a \in C - B$ . Eqs. (3)-(9) serve as heuristic information and provides a comprehensive consideration, covering not only the mutual information increment when a specific attribute  $a$  is added to the attribute subset  $B$ , but also the information entropy of the selected attribute  $a$  itself.

When ants choose the next node, they consider both the pheromone concentration and heuristic information, calculating the probability of selecting each node, as shown in Eq. (10).

$$P_{ij}^k(t) = \frac{\tau_{ij}^\alpha(t) \eta_{ij}^\beta(t)}{\sum_{l \in \text{allowed}_k} \tau_{il}^\alpha(t) \eta_{il}^\beta(t)} \quad j \in \text{allowed}_k \quad (10)$$

where,  $P_{ij}^k(t)$  represents the probability of moving from attribute node  $i$  to node  $j$  for the  $k$ -th ant;  $\tau_{ij}(t)$  represents the pheromone concentration between attribute nodes  $i$  and  $j$ ;  $\eta_{ij}(t)$  represents the heuristic information;  $\alpha$  and  $\beta$  are parameters that control the relative importance of pheromone and heuristic information;  $\text{allowed}_k$  represents the set of condition attributes not yet selected by the  $k$ -th ant.

The search termination conditions are as follows: the search stops when any of the following conditions are met, completing an iteration.

- (1)  $I(C; D) = I(R_K; D)$ , where  $R_K$  is the local solution constructed by the  $k$ -th ant, i.e., the reduction set;
- (2) When the number of attributes in the reduction set exceeds the number of attributes in the current best solution.

In (1), when the mutual information of the reduction set found by the  $k$ -th ant is equal to the mutual information of all condition attributes with the decision attribute, it is considered that the  $k$ -th ant has successfully constructed a local solution. If the number of attributes in the local solution is less than in the current optimal solution, then the local solution becomes the global optimal solution. (2) indicates that when the number of attributes in the reduction set found by the  $k$ -th ant exceeds the number of attributes in the current global optimal solution, it implies that a better solution set cannot be obtained, hence the search stops.

To address the slow convergence and susceptibility to local optima in the ACO's optimization process, this paper introduces a redundancy detection mechanism.

During each ant's search for the reduction set, redundancy checks are performed for each selected attribute. The specific steps are as follows:

- (1) Based on Eq. (10), select the attribute with the highest selection probability, denoted as  $b_k$  ( $b_k \in C - R_k$ ).
- (2) Determine whether the mutual information with the decision attribute remains the same before and after adding  $b_k$  to the current reduction set  $R_k$ . If  $I(R_k \cup \{b_k\}; D) = I(R_k; D)$ , attribute  $b_k$  is considered redundant; otherwise,  $b_k$  can be added to the reduction set.

After ants reach a local solution, redundancy checks are also needed to accelerate convergence. Each attribute in the local solution is sequentially removed, and if the mutual information with the decision attribute remains the same before and after its removal, the removed attribute is considered redundant; otherwise, it is necessary and cannot be removed.

Overall, the process of attribute reduction based on the improved ACO is illustrated in Figure 2.

The specific implementation steps for attribute reduction based on the improved ACO are as follows:

**Step 1.** Parameter Initialization. Set the reduction set  $R_{\min} = \emptyset$ , the number of attributes in the reduction set  $L_{\min} = 0$ , the number of algorithm iterations  $t = 0$ , and the ant index  $k = 0$ . Also define the maximum number of iterations  $T$  and the number of ants  $Ant$ , as well as the initial values for pheromone concentration  $\tau_{ij}(t)$  and pheromone evaporation rate  $\rho$ .

**Step 2.** Increase the algorithm iteration counter  $t = t + 1$ .

**Step 3.** Increase the ant index by one, i.e.,  $k = k + 1$ .

**Step 4.** Randomly select an initial attribute  $a_k \in C$ , thus the reduction set found by the  $k$ -th ant is  $R_k = \{a_k\}$ , and the number of attributes in the reduction set is  $L_k = 1$ .

**Step 5.** Path Selection. Calculate the transition probability rule using pheromone concentration  $\tau_{ij}(t)$  and heuristic information  $\eta_{ij}(t)$  to choose the next attribute node. Ultimately, select the attribute node with the highest probability  $b_k = \arg \max \{p_{ij}^k\}$ ,  $b_k \in \{C - R_k\}$  as the next node in the path.

**Step 6.** Redundancy Detection. Calculate whether the mutual information with the decision attribute remains the same before and after adding attribute  $b_k$  to the current reduction set  $R_k$ . If  $I(R_k \cup \{b_k\}; D) = I(R_k; D)$ , it indicates that attribute  $b_k$  is redundant, then proceed to Step 8; otherwise, attribute  $b_k$  can be added to the reduction set, proceed to Step 7.

**Step 7.** Add attribute  $b_k$  to the reduction set  $R_k = R_k \cup \{a_k\}$ , and update the number of attributes in the reduction set  $L_k = L_k + 1$ .

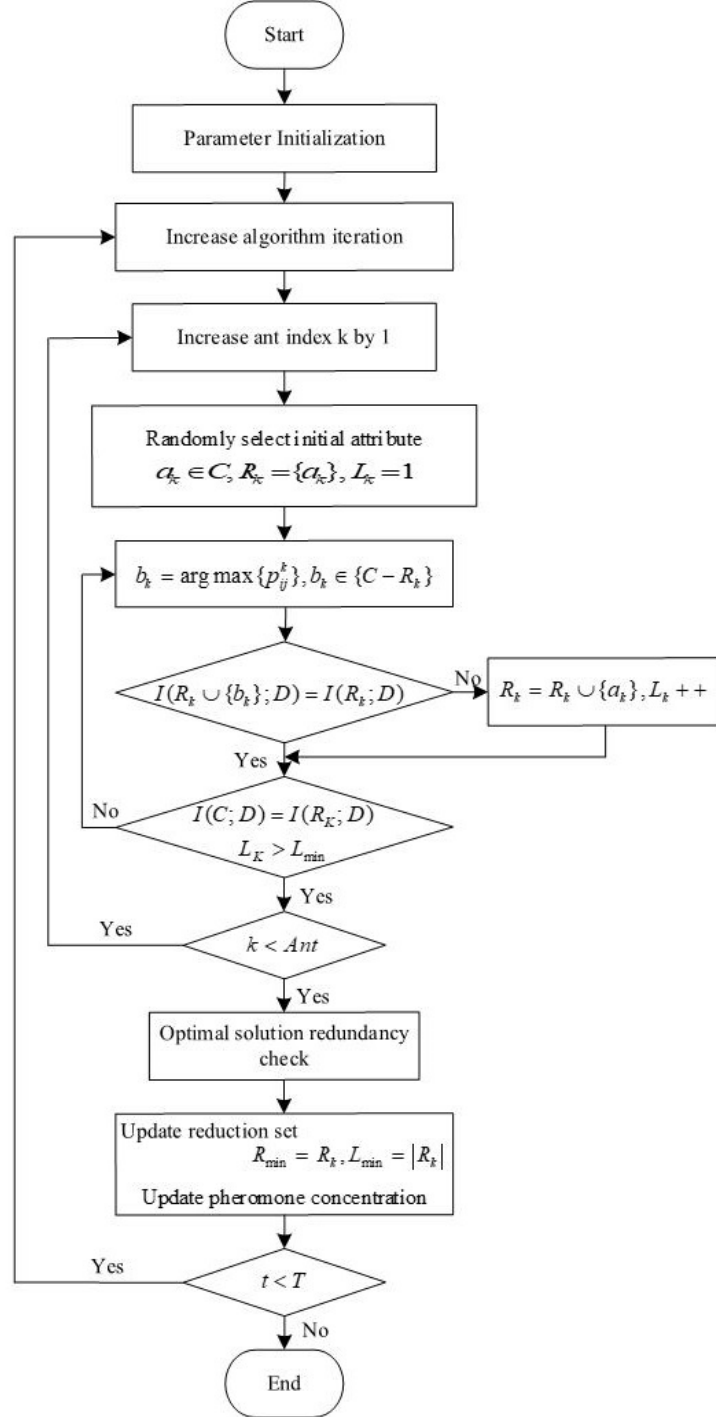
**Step 8.** Determine if the search stopping conditions are met. If  $I(C; D) = I(R_K; D)$  or  $L_K > L_{\min}$ , stop the search and proceed to Step 9; otherwise, continue to search for the next attribute, proceed to Step 5.

**Step 9.** Determine if all ants have completed their search. If  $k < Ant$ , return to Step 3; otherwise, proceed to Step 10.

**Step 10.** Optimal Solution Redundancy Detection. Sequentially remove each attribute in the solution; if the mutual information with the decision attribute remains the same before and after the removal of an attribute, it can be determined to be a redundant attribute; otherwise, it is a necessary attribute and cannot be removed.

**Step 11.** Update the reduction set  $R_{\min} = R_k$ , the number of attributes in the reduction set  $L_{\min} = |R_k|$ , and update the pheromone concentration  $\tau_{ij}(t)$ .

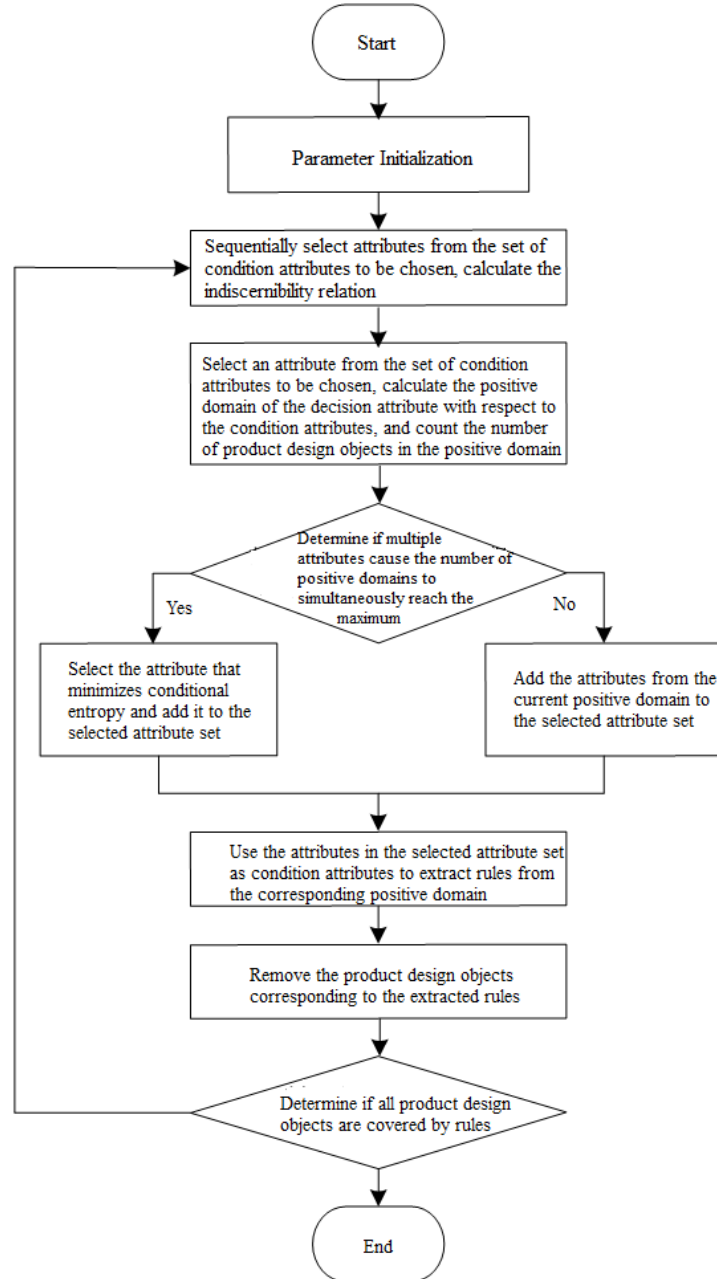
**Step 12.** Determine if the maximum number of iterations has been reached. If  $t < T$ , return to Step 2; otherwise, the algorithm process ends.



**Figure 2.** Process of attribute reduction based on the improved ACO

### 3.2 Rule Generation Based on Classification Consistency

After using the improved ACO for attribute reduction, redundant condition attributes are removed. At this point, rules can be extracted based on the set of condition attributes  $C$  and the decision attribute  $D$ , but the rules generated tend to be overly specific and numerous. Therefore, it is necessary to select a subset of attributes as the antecedents of the rules. This paper employs a rule generation algorithm based on classification consistency, which starts from an empty set and incrementally adds condition attributes, aiming to extract potential rules with as few attributes as possible. This method starts simply and refines rules through iterative steps to effectively reveal patterns and associations in the data. The rule generation process is illustrated in Figure 3.



**Figure 3.** Rule generation based on classification consistency

The specific implementation steps for rule generation based on classification consistency are as follows:

**Step 1.** Parameter initialization, where  $G$  is the set of product design objects not yet covered by existing rules;  $SelectAttr$  is the set of attributes selected as possible rule antecedents, initially empty;  $unSelectAttr$  is the set of product design condition attributes yet to be selected.

**Step 2.** Sequentially select attribute  $x_i$  from  $unSelectAttr$ , calculate the indiscernibility relation between the set of condition attributes  $\{SelectAttr, x_i\}$  and the decision attribute  $D$ .



**Step 3.** Sequentially select condition attribute  $x_i$  from  $unSelectAttr$ , calculate  $POS_{(SelectAttr \cup x_i)}(d)$ , the positive domain of decision attribute of the decision attribute  $D$ , and count  $card(POS_{(SelectAttr \cup x_i)}(d))$ , the number of product design objects in the positive domain.

**Step 4.** Determine if multiple attributes cause  $card(POS_{(SelectAttr \cup x_i)}(d))$  to simultaneously reach its maximum. If so, proceed to Step 6; otherwise, proceed to Step 5.

**Step 5.** Select the attribute  $x_i$  that makes  $card(POS_{(SelectAttr \cup x_i)}(d))$  reach its maximum, add it to  $SelectAttr$ , and remove the attribute  $x_i$  from  $unSelectAttr$ . After completing this, proceed to Step 7.

**Step 6.** Calculate conditional entropy  $H(d | a)$ , select the attribute  $a$  that minimizes the conditional entropy, simultaneously add the attribute  $a$  to  $SelectAttr$ , and remove the attribute  $a$  from  $unSelectAttr$ .

**Step 7.** Use the attributes in  $SelectAttr$  as condition attributes to extract rules from the corresponding  $POS_{SelectAttr}(d)$  and add them to the rule set.

**Step 8.** Delete the corresponding objects  $POS_{SelectAttr}(d)$  from  $SelectAttr$ , that is,  $G = G - POS_{SelectAttr}(d)$ .

**Step 9.** Determine if all product design objects are covered by rules, i.e., whether  $G$  is empty. If not empty, proceed to Step 2; otherwise, the rule generation is complete.

## 4 Experimental Validation and Analysis

### 4.1 Validation of Attribute Reduction Algorithm on Public Datasets

To validate the effectiveness and reliability of the improved ACO for attribute reduction proposed in this paper, comparative experiments were conducted using the ACORS from reference [21], ACOAR from reference [22], RACO from reference [23], and IEACO from reference [24].

The evaluation of attribute reduction effectiveness was based on three indicators: the number of attributes after reduction, attribute reduction rate, and runtime. The attribute reduction rate is expressed as shown in Eq. (11):

$$rate = \frac{|C| - |R|}{|C|} \times 100\% \quad (11)$$

where,  $|C|$  represents the number of attributes before reduction, and  $|R|$  is the number of attributes after reduction.

For comparative experiments, six datasets were selected from the UCI database. The experimental environment is as follows: AMD Ryzen™ 7 5800H processor at 3.20 GHz, 16GB RAM, using PyCharm64 on Windows 10 as the experimental platform, and Python as the programming language. Specific information about the six datasets is shown in Table 1.

**Table 1.** Specific information about datasets

No.	UCI Dataset	Number of Attributes	Number of Samples
1	Zoo	16	101
2	Soybean	35	114
3	Lung-cancer	56	32
4	Lymphography	18	148
5	Audiology	70	200
6	Mushroom	22	8124

To ensure comparability of the experiments, the initial parameter values of the above five algorithms were set consistently. Table 2 shows the initial values of the ACO parameters.

Each of the five algorithms was executed 20 times, and the average of the results was taken as the attribute reduction outcome.

After executing the five algorithms, the number of attributes as shown in Table 3.

**Table 2.** Initial values of ACO parameters

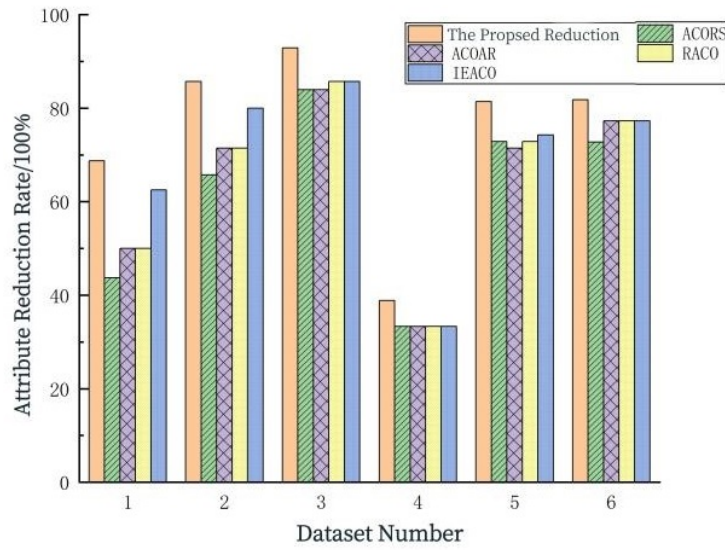
No.	Parameter Name	Parameter Value	Meaning of the Parameter
1	$\alpha$	1.2	Parameter controlling the relative importance of pheromone and heuristic information
2	$\beta$	0.5	Parameter controlling the relative importance of pheromone and heuristic information
3	$\tau_{ij}(t)$	1	Initial pheromone concentration
4	$\rho$	0.2	Pheromone evaporation rate
5	Ant	$1.5n$	Number of ants, where n is the number of attributes in the dataset
6	$T$	200	Maximum number of iterations

**Table 3.** Number of attributes after reduction

Dataset No.	The Proposed Algorithm	ACORS	ACOAR	RACO	IEACO
1	5.2	8.9	8.1	8.0	5.8
2	5.1	12.2	10.0	9.8	7.3
3	4.3	8.8	9.0	8.1	8.1
4	11.0	12.2	12.0	12.0	12.0
5	13.2	18.9	20.1	19.2	17.7
6	4.0	5.8	4.9	5.1	5.2

As can be seen from Table 3, the algorithm proposed in this paper shows superiority in terms of attribute reduction. The number of attributes after reduction is relatively fewer, presenting a more compact attribute subset compared to the ACORS, ACOAR, RACO, and IEACO algorithms.

After executing the five algorithms, the attribute reduction rates are shown in Figure 4.

**Figure 4.** Attribute reduction rate after algorithm execution

As illustrated in Figure 4, the algorithm proposed in this paper achieves relatively higher attribute reduction rates across all datasets. In contrast, the other four algorithms exhibit lower attribute reduction rates on some datasets, indicating that their selection of dataset attributes is overly conservative, resulting in relatively higher dimensions.

After the execution of the five algorithms, the runtime for each dataset is shown in Table 4.

**Table 4.** Dataset runtime

Dataset No.	The Proposed Algorithm	ACORS	ACOAR	RACO	IEACO
1	1.28	3.34	2.21	2.42	1.89
2	31.52	43.18	37.84	39.77	36.38
3	1.05	3.09	2.16	2.30	1.82
4	5.36	10.37	8.33	8.93	7.55
5	100.14	176.76	147.97	148.05	143.26
6	164.83	318.53	287.25	276.91	234.08

As indicated by Table 4, the proposed algorithm exhibits a significant advantage in runtime efficiency across all datasets. Compared to the ACORS, ACOAR, RACO, and IEACO algorithms, the proposed algorithm has a relatively shorter runtime on the six datasets, demonstrating its superiority.

## 4.2 Application Example

As breakthroughs in new-generation information technologies continue, the automotive industry is rapidly moving towards intelligent development. Automobiles have become an important pillar for building a powerful automobile



nation and driving industrial transformation and upgrading in China. Therefore, generating rules for automotive design based on rough sets is of great significance.

#### 4.2.1 Automotive design rule generation based on rough sets

Data is collected from various sources including technical experts, patents, and journal articles. The decision table is constructed using attributes such as type of knowledge, source of knowledge, domain of knowledge, basic design information, form of knowledge, technical principle knowledge, and instance knowledge as condition attributes, with the design objective serving as the decision attribute, as shown in Table 5.

**Table 5.** Automotive design decision table

Universe of Discourse $U$	Condition Attributes $C$							Design Goal $d$
	Knowledge Type $x_1$	Knowledge Source $x_2$	Knowledge Domain $x_3$	Basic Design Information $x_4$	Form of Knowledge $x_5$	Technical Principle Knowledge $x_6$	Example Knowledge $x_7$	
$n_1$	Lightweight Rule	Material Selection Handbook	Quality Requirements	Structural Design	Experiment	Lightweight Design Principles	Example Name	Reduce Weight
$n_2$	Material Optimization	Finite Element Analysis	Quality Requirements	Structural Design	Experiment	Lightweight Design Principles	Example Name	Reduce Weight
$n_3$	Lightweight Rule	Material Selection Handbook	Quality Requirements	Structural Design	Simulation	Lightweight Design Principles	Example Name	Reduce Weight
$n_4$	Lightweight Rule	Material Selection Handbook	Quality Requirements	Structural Design	Experiment	Lightweight Design Principles	Example Name	Reduce Weight
$n_5$	Strength Optimization Rule	Industry Technical Standard	Strength Requirements	Structural Design	Experiment	Lightweight Design Principles	Example Name	Increase Strength
$n_6$	Strength Optimization Rule	Material Selection Handbook	Strength Requirements	Structural Design	Experiment	Strength Design Principles	Example Name	Increase Strength
$n_7$	Strength Analysis	Industry Technical Standard	Strength Requirements	Structural Design	Experiment	Strength Design Principles	Example Name	Increase Strength
$n_8$	Disassembly Time Optimization Rule	Structural Design Standard	Disassembly Requirements	Structural Design	Experiment	Disassembly Design Principles	Example Name	Shorten Disassembly Time
$n_9$	Disassembly Time Optimization Rule	User-Oriented Design	Disassembly Requirements	Structural Design	Simulation	Disassembly Design Principles	Example Name	Shorten Disassembly Time
$n_{10}$	Disassembly Time Analysis	User-Oriented Design	Disassembly Requirements	Structural Design	Experiment	Disassembly Design Principles	Example Name	Shorten Disassembly Time
$n_{11}$	Material Optimization	Material Selection Handbook	Working Environment	Structural Design	Experiment	Disassembly Design Principles	Example Name	Improve Wear Resistance
$n_{12}$	Material Optimization	Engineering Design Handbook	Usage Requirements	Structural Design	Experiment	Resistance Design Principles	Example Name	Improve Wear Resistance
$n_{13}$	Wear Resistance Analysis	Finite Element Analysis	Usage Requirements	Structural Design	Experiment	Resistance Design Principles	Example Name	Improve Wear Resistance
$n_{14}$	Material Optimization	Structural Design Standard	Working Environment	Structural Design	Experiment	Resistance Design Principles	Example Name	Reduce Noise

Universe of Discourse $U$	Condition Attributes $C$							Design Goal $d$
	Knowledge Type $x_1$	Knowledge Source $x_2$	Knowledge Domain $x_3$	Basic Design Information $x_4$	Form of Knowledge $x_5$	Technical Principle Knowledge $x_6$	Example Knowledge $x_7$	
$n_{15}$	Noise Reduction Rule	Finite Element Analysis	Working Environment	Structural Design	Simulation	Disassembly Design Principles	Example Name	Reduce Noise
$n_{16}$	Noise Reduction Rule	Structural Design Standard	Working Environment	Structural Design	Experiment	Disassembly Design Principles	Example Name	Reduce Noise
$n_{17}$	Energy Saving Design	Material Selection Handbook	Energy Efficiency Requirements	Structural Design	Simulation	Energy Efficiency Design Principles	Example Name	Improve Energy Efficiency
$n_{18}$	Energy Efficiency Improvement Rule	Material Selection Handbook	Energy Efficiency Requirements	Structural Design	Simulation	Energy Efficiency Design Principles	Example Name	Improve Energy Efficiency
$n_{19}$	Energy Efficiency Improvement Rule	Material Selection Handbook	Energy Efficiency Requirements	Structural Design	Simulation	Energy Efficiency Design Principles	Example Name	Improve Energy Efficiency
$n_{20}$	Strength Analysis	Industry Technical Standard	Reliability Requirements	Structural Design	Experiment	Strength Design Principles	Example Name	Improve Reliability
$n_{21}$	Strength Optimization Rule	Industry Technical Standard	Reliability Requirements	Structural Design	Simulation	Strength Design Principles	Example Name	Improve Reliability
$n_{22}$	Strength Optimization Rule	Finite Element Analysis	Reliability Requirements	Structural Design	Graphics	Strength Design Principles	Example Name	Improve Reliability
$n_{23}$	Ecological Friendliness Improvement Rule	User-Oriented Design	Environmental Requirements	Structural Design	Experiment	Green Design Principles	Example Name	Improve Ecological Friendliness
$n_{24}$	Ecological Friendliness Improvement Rule	User-Oriented Design	Environmental Requirements	Structural Design	Simulation	Green Design Principles	Example Name	Improve Ecological Friendliness
$n_{25}$	Ecological Friendliness Design	User-Oriented Design	Working Environment	Structural Design	Simulation	Energy Efficiency Design Principles	Example Name	Improve Ecological Friendliness

After establishing the automotive design decision table, to validate the effectiveness of the rough set-based rule generation method proposed in this paper within automotive design, a comparative analysis of the rules generated by the method proposed in reference [25] and the method proposed in this paper was conducted.

Rule generation effectiveness is assessed using three indicators: rule coverage rate, rule accuracy rate, and the average number of attributes per rule. The rule coverage rate is shown in Eq. (12):

$$\text{Cov}(r) = \frac{\text{Number of Records Satisfying Rule Antecedents}}{\text{Total Number of Records}} \quad (12)$$

The rule accuracy rate is shown in Eq. (13):

$$\text{Acc}(r) = \frac{\text{Number of Records Satisfying Both Rule Antecedents and Consequents}}{\text{Number of Records Satisfying Antecedents}} \quad (13)$$

The average number of attributes per rule is shown in Eq. (14):

$$\text{Average Number of Attributes per Rule} = \frac{\text{Total Number of Attributes}}{\text{Total Number of Rules}} \quad (14)$$

(1) Method proposed in reference [25]

After attribute reduction, the method proposed in reference [25] successfully eliminated redundant attributes  $\{x_4, x_5, x_7\}$ . The set of automotive design rules generated is shown in Table 6.

As shown in Table 6, there are a total of 11 rules in the set of automotive design rules, with a total of 20 attributes in the rule set.

(2) Method proposed in this paper

In the automotive design decision information system  $S = U, A, V, f$ , the universe of discourse  $U = \{n_1, n_2, \dots, n_{25}\}$ , condition attribute set  $C = \{x_1, x_2, \dots, x_7\}$ , and decision attribute  $d$  are given. Table 7 displays the initial values of the parameters for the improved ACO proposed in this paper.

**Table 6.** Rules for automotive design generated by the method proposed in reference [25]

No.	Rule Antecedents	Rule Consequents
1	Quality Requirements	Reduce Weight
2	Strength Requirements	Increase Strength
3	Disassembly Requirements	Shorten Disassembly Time
4	Usage Requirements	Improve Wear
5	Material Optimization, Material Selection Handbook, Working Environment, Disassembly Design Principles	Resistance
6	Energy Efficiency Requirements	Improve Energy Efficiency
7	Reliability Requirements	Improve Reliability
8	Environmental Requirements	Improve Ecological
9	Ecological Friendliness Design, Working Environment, Energy Efficiency Design Principles	Friendliness
10	Noise Reduction Rule, Working Environment	Reduce Noise
11	Material Optimization, Structural Design Standards, Working Environment, Vibration Reduction Design Principles	

**Table 7.** Initial values of parameters for the improved ACO proposed in this paper

No.	Parameter Name	Parameter Value	Meaning of the Parameter
1	$R_{\min}$	$\emptyset$	Set for storing reduction results
2	$L_{\min}$	0	Number of attributes in the reduction set
3	$t$	0	Initial number of iterations of the algorithm
4	$k$	0	Initial ant index
5	$\tau_{ij}(t)$	1	Initial pheromone concentration
6	$\rho$	0.2	Pheromone evaporation rate
7	$\alpha$	1.2	Parameter controlling the relative importance of pheromone and heuristic information
8	$\beta$	0.5	Parameter controlling the relative importance of pheromone and heuristic information
9	$T$	200	Maximum number of iterations
10	Ant	11	Number of ants

After attribute reduction using the improved ACO, redundant attributes  $\{x_2, x_4, x_5, x_7\}$  and duplicate universe of discourse entries  $\{n_3, n_4, n_9, n_{19}, n_{22}, n_{24}\}$  were removed. The universe of discourse for the automotive design decision information system is updated to  $\{n_1, n_2, n_5, n_6, n_7, n_8, n_{10}, n_{11}, n_{12}, n_{13}, n_{14}, n_{15}, n_{17}, n_{18}, n_{20}, n_{21}, n_{23}\}$  and  $\{n_{25}\}$ , the condition attribute set is  $C = \{x_1, x_3, x_6\}$ , and the decision attribute is  $D$ . Subsequently, rules are generated based on classification consistency, covering all objects within the universe of discourse. The set of automotive design rules is shown in Table 8.

As shown in Table 8, there are a total of 11 rules in the set of automotive design rules, with a total of 17 attributes in the rule set.

In summary, the algorithm proposed in this paper removed 4 redundant attributes  $\{x_2, x_4, x_5, x_7\}$ , generated 11 rules, and the rule set contains 17 attributes, with both the rule coverage rate and rule accuracy rate reaching 100%, and the average number of attributes per rule being 1.55. In comparison, the method proposed in reference [25] removed 3 redundant attributes  $\{x_4, x_5, x_7\}$ , generated 11 rules, and the rule set contains 20 attributes, with the rule coverage rate and rule accuracy rate also at 100%, but the average number of attributes per rule is higher, at 1.82.

Therefore, the algorithm proposed in this paper has certain advantages over the method proposed in reference [25], as it demonstrates more efficient redundancy identification and removal capabilities, thereby generating more concise rules.

**Table 8.** Set of automotive design rules generated by the proposed method

No.	Rule Antecedents	Rule Consequents
2	Quality Requirements	Reduce Weight
3	Strength Requirements	Increase Strength
4	Disassembly Requirements	Shorten Disassembly Time
5	Usage Requirements	Improve Wear Resistance
5	Material Optimization, Working Environment, Disassembly Design Principles	Improve Wear Resistance
6	Energy Efficiency Requirements	Improve Energy Efficiency
7	Reliability Requirements	Improve Reliability
8	Environmental Requirements	
9	Ecological Friendliness Design, Working Environment	Improve Ecological Friendliness
10	Material Optimization, Working Environment, Vibration Reduction Design Principles	
11	Noise Reduction Rule, Working Environment	Reduce Noise

## 5 Conclusions

This paper proposed a rough set-based rule generation method and validated the proposed attribute reduction method using public datasets, as well as tested the performance of the rule generation method in an automotive design example. The results show that the proposed attribute reduction method outperforms the other four methods in three evaluation metrics: the number of attributes after reduction, attribute reduction rate, and runtime. The rule generation method is assessed using three evaluation metrics—rule coverage rate, rule accuracy rate, and the average number of attributes per rule, showing more efficient redundancy identification and removal capabilities, thereby generating more concise rules. Considering that the ACO may not perform well with complex attribute relationships, future work could consider incorporating other heuristic algorithms or deep learning technologies to enhance the effect of attribute reduction and generate even more concise rules.

## Funding

The paper was funded by the National Natural Science Foundation of China (Grant No. 51775272); the Jiangsu Provincial Natural Science Foundation General Project of China (Grant No. BK20221481).

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References

- [1] M. Y. Yang, “Research on the design of small electromechanical products for crew use based on QFD and TRIZ,” Masteral Dissertation, Dalian Ocean University, Dalian, China, 2023. <https://doi.org/10.27821/d.cnki.gdlhy.2023.000482>
- [2] L. L. Liu, “Research on product design methods based on multidimensional features and holistic cognition,” Doctoral Dissertation, Southeast University, 2017. <https://doi.org/10.7666/d.Y3256239>
- [3] G. F. Yue, J. H. Liu, Q. Zhang, and Y. Z. Hou, “Building a design-rationale-centric knowledge network to realize the internalization of explicit knowledge,” *Appl. Sci.*, vol. 13, no. 3, pp. 1539–1539, 2023. <https://doi.org/10.3390/AP13031539>
- [4] G. X. Li, J. Zheng, and M. Zhang, “Parallel optimization method for product configuration and supplier selection based on interval,” *IOP Conf. Ser. Earth Environ. Sci.*, vol. 252, no. 5, p. 052106, 2019. <https://doi.org/10.1088/1755-1315/252/5/052106>

- [5] A. Sinha and A. Anand, "Development of sustainable supplier selection index for new product development using multi criteria decision making," *J. Clean. Prod.*, vol. 197, pp. 1587–1596, 2018. <https://doi.org/10.1016/j.jclepro.2018.06.234>
- [6] V. Radhakrishna, P. Kumar, and V. Janaki, "SRIHASS - A similarity measure for discovery of hidden time profiled temporal associations," *Multimed. Tools Appl.*, vol. 77, pp. 17 643–17 692, 2018. <https://doi.org/10.1007/s11042-017-5185-9>
- [7] S. Fu, Y. T. Luo, Y. L. Liu, Q. Liao, S. S. Kong, A. H. Yang, L. F. Lin, and H. Li, "Mining association rules between the granulation feasibility and physicochemical properties of aqueous extracts from Chinese herbal medicine in fluidized bed granulation," *Math. Biosci. Eng.*, vol. 20, no. 11, pp. 19 065–19 085, 2023. <https://doi.org/10.3934/MBE.2023843>
- [8] L. M. Zhang, K. Liu, X. W. Xie, W. J. Bai, B. L. Wu, and P. P. Dong, "A data-driven network intrusion detection system using feature selection and deep learning," *J. Inf. Secur. Appl.*, vol. 78, p. 103606, 2023. <https://doi.org/10.1016/J.JISA.2023.103606>
- [9] C. C. Huang, W. Y. Liang, R. Gung, and P. A. Wang, "Rough-set-based rule induction with the elimination of outdated big data: Case of renewable energy equipment promotion," *Sustainability*, vol. 15, no. 20, p. 14984, 2023. <https://doi.org/10.3390/su152014984>
- [10] T. X. Wang and M. Y. Zhou, "Integrating rough set theory with customer satisfaction to construct a novel approach for mining product design rules," *J. Intell. Fuzzy Syst.*, vol. 41, no. 1, pp. 331–353, 2021. <https://doi.org/10.3233/JIFS-201829>
- [11] Z. F. Zhang, N. N. Chai, E. Ostrosi, and Y. L. Shang, "Extraction of association rules in the schematic design of product service system based on Pareto-MODGDFA," *Comput. Ind. Eng.*, vol. 129, pp. 392–403, 2019. <https://doi.org/10.1016/j.cie.2019.01.040>
- [12] B. L. Lin, D. Liu, H. F. Wang, and T. Y. Liu, "Research on product process optimization based on association rules," *Mech. Des. Manuf. Eng.*, vol. 48, no. 4, pp. 6–11, 2019.
- [13] Y. Cao, "Research on process knowledge discovery and reuse based on data mining," Master's Thesis, Shandong University, Jinan, China, 2019.
- [14] S. Q. Ye, K. Q. Zhou, A. M. Zain, F. L. Wang, and Y. Yusoff, "A modified harmony search algorithm and its applications in weighted fuzzy production rule extraction," *Front. Inf. Technol. Electron. Eng.*, vol. 24, no. 11, pp. 1574–1590, 2023. <https://doi.org/10.1631/FITEE.2200334>
- [15] K. Y. Fung, C. Y. Tang, E. W. M. Lee, G. T. S. Ho, M. K. W. Siu, and W. L. Mou, "Design rule extraction from a trained ann model using ga for product form design of mobile phones," *Intell. Autom. Soft Comput.*, vol. 18, no. 4, pp. 369–379, 2012. <https://doi.org/10.1080/10798587.2012.10643249>
- [16] J. Shen, B. Wu, and Y. Li, "Personalized configuration rules extraction in product service systems by using Local Cluster Neural Network," *Ind. Manag. Data Syst.*, vol. 115, no. 8, pp. 1529–1546, 2015. <https://doi.org/10.1108/IMDS-03-2015-0092>
- [17] Y. L. Liu, X. Y. Wang, and J. Wang, "Product image modeling design based on triangular fuzzy and BP neural network," *Packag. Eng.*, vol. 42, no. 14, pp. 185–193, 2021. <https://doi.org/10.19554/j.cnki.1001-3563.2021.14.022>
- [18] X. Qin, "Product green design knowledge update driven by changing demand," Master's Thesis, Hefei University of Technology, Hefei, China, 2018.
- [19] J. F. Wen, "Research on knowledge engineering methods for automotive mold design under cloud design pattern," Master's Thesis, Tianjin University, Tianjin, China, 2017.
- [20] M. Kobayashi and K. Niwa, "Method for grouping of customers and aesthetic design based on rough set theory," *Comput. Aided Des. Appl.*, vol. 15, no. 4, pp. 565–574, 2018. <https://doi.org/10.1080/16864360.2017.1419644>
- [21] S. R. Ahmad, A. A. Bakar, and M. R. Yaaku, "Ant colony optimization for text feature selection in sentiment analysis," *Intell. Data Anal.*, vol. 23, no. 1, pp. 133–158, 2019. <https://doi.org/10.3233/IDA-173740>
- [22] M. F. Shi, S. C. Xia, and X. Feng, "An adaptive ant colony algorithm based on local information entropy to solve distributed constraint optimization problems," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 37, no. 9, 2023. <https://doi.org/10.1142/S0218001423590139>
- [23] Z. Y. Yang, J. Ye, J. X. Ao, and Y. X. Ji, "Feature selection method based on ant colony optimization algorithm and improved neighborhood discernibility matrix," in *Communications in Computer and Information Science*. Springer, Singapore, 2022, pp. 116–131. [https://doi.org/10.1007/978-981-19-1256-6\\_9](https://doi.org/10.1007/978-981-19-1256-6_9)
- [24] S. Jameel and S. U. Rehman, "An optimal feature selection method using a modified wrapper based ant colony optimisation," *J. Natl. Sci. Found. Sri Lanka*, vol. 46, no. 2, pp. 143–151, 2018. <https://doi.org/10.4038/jnsfsr.v46i2.8414>
- [25] D. G. Li, Z. X. Chen, and J. Yin, "A new attribute reduction recursive algorithm based on granular computing," *J. Comput.*, vol. 8, no. 3, pp. 630–637, 2013. <http://doi.org/10.4304/jcp.8.3.630-637>