RESEARCH ARTICLE

WILEY

Hydrological cycling optimization-based multiobjective feature-selection method for customer segmentation

Xi Song[1,2] | Matthew Tingchi Liu[2] | Qianying Liu[1] |
Ben Niu[1,3]

[1]Department of Management Science and Engineering, College of Management, Shenzhen University, Shenzhen, China

[2]Faculty of Business Administration, University of Macau, Macau Special Administrative Region, Taipa, China

[3]Greater Bay Area International Institute for Innovation, Shenzhen University, Shenzhen, China

Correspondence
Ben Niu, Department of Management Science and Engineering, College of Management, Shenzhen University, No. 3688 Nanhai Avenue, Nanshan District, Shenzhen 518060, China.
Email: drniuben@gmail.com

Abstract

In the customer segmentation problem, a large number of features are manually designed and used to comprehensively describe the customer instances. However, some of these features are irrelevant, redundant, and noisy, which are not necessary and effective for customer segmentation. Feature selection is an important data preprocessing method by selecting important features from the original feature set. Particularly, feature selection in customer segmentation is a multiobjective problem that aims to minimize the feature number and maximize the classification performance. This paper proposes a multiobjective feature-selection method based on a meta-heuristic algorithm—hydrological cycling optimization (HCO)—to solve customer segmentation. The proposed method is able to automatically evolve a set of non-dominated solutions that select small numbers of features and achieve high classification accuracy. To this end, three strategies based on the global flow operator, possibility-based acceptance criteria, and density-based evaporation and precipitation are proposed to improve the global search ability and the solution diversity of the proposed approach. The performance of the proposed approach is examined on three customer-segmentation datasets and compared with original multiobjective HCO and six well-known

evolutionary multiobjective algorithms. The results confirm the superiority of the proposed approach in solving multiobjective customer-segmentation problems by achieving higher calculation stability, search diversity, and solution quality compared with the other competing methods.

1 | INTRODUCTION

Feature selection (FS), as a data pre-processing technique, has been widely employed for solving classification problems. FS has been applied to tackle the abundance of noisy and irrelevant features, thereby making reduced-size data sets retain the maximum possible information from the original data.[1] This affords users the advantages of reduced dimensionality and computation costs while enhancing data interpretation and prediction performance.[2] Based on the feature subset evaluation methods, FS methods can be mainly categorized as filter and wrapper methods. Filter methods evaluate the feature subsets based on a specific mathematic criterion, such as distance, consistency, dependency, and/or correlation. In contrast, wrapper methods employ a predetermined learning algorithm to evaluate the effectiveness of the selected feature subset (e.g., accuracy).[3] Remarkably, wrapper methods are recognized based on their superior classification capability in reducing data dimensionality. Thus, wrappers yield more effective results than filters.[4]

Although wrapper methods demonstrate superior classification performance, the major challenge encountered when solving FS problems is the exponential growth of the search space along with the increasing feature subset count. To overcome this challenge, several metaheuristic global-search algorithms, such as the particle swarm optimization (PSO),[5–7] genetic algorithm (GA),[8–10] differential evolution (DE),[11] and ant colony optimization (ACO),[12] have been used solve different optimization problems while incurring low computation costs and realizing optimum classification performance.[2] Accordingly, these approaches are considered desirable candidates to overcome the limitations of wrapper FS methods.

The hydrological cycle optimization (HCO) algorithm is an emerging metaheuristic algorithm. It demonstrates outstanding performance in maintaining diversity and avoiding the realization of local optimal when solving single-objective optimization problems.[13,14] Moreover, it obtains good balance among multiple quality objectives when solving recommendation system problems.[15] Prior studies have laid a strong foundation for extending the applications of the original HCO to solve complex real-world problems. However, to the best of our knowledge, no research has yet been undertaken to utilize the HCO for solving FS problems, in particular, multiobjective FS problem for customer segmentation.

By definition, FS can be considered a multiobjective-optimization problem aimed at simultaneously minimizing the classification error rate and feature-subset size. Most existing studies consider FS as a single-objective-optimization problem that reduces the classification error regardless of subset size.[16–19] For researchers attempting to develop multiobjective FS methods, the need to tradeoff

convergence in favor of increased diversity in a high-dimensional space has remained a long-standing difficulty. The most common and straightforward strategy is the use of the weighted-sum approach,[20] which transforms the optimization problem into a single-objective one. However, such approaches are criticized for their subjectivity in determining the weighting coefficients and insufficiency with regard to solving problems involving non-convex Pareto fronts.[21,22]

In view of the above discussions, customer segmentation can be considered a multiobjective-optimization problem. Further, it is imperative to overcome the limitations of most existing metaheuristic-based FS methods concerning convergence and population diversity. Because the HCO has been successfully applied to solve difficult combinatorial optimization problems, including nurse scheduling[13] and recommendation systems,[15] we strongly believe that the HCO-inspired multi-objective FS method to have great potential for solving customer-segmentation problems.

This paper presents an HCO-based multiobjective FS approach that generates a Pareto-optimal front instead of a unique global optimum, thereby producing a set of non-dominated solutions that specify a small number of features while incurring minimum classification errors. To realize this objective, a multiobjective evolutionary HCO (MOEHCO) has been proposed. It is built on the idea of a wrapper FS method along with three strategies—global flow operator, possibility-based acceptance criteria, and density-based evaporation and precipitation—that tackle optimization problems in their prematurity as well as enhance the robustness of individual performance and solution diversity. Six well-known multiobjective metaheuristic algorithms—Multi-Objective Particle Swarm Optimization (MOPSO),[17] Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA-D),[18] Non-Dominated Sorting Genetic Algorithm II (NSGA-II),[19] Inverse Modeling Multi-Objective Evolutionary Algorithm (IM-MOEA),[23] Strength Pareto Evolutionary Algorithm 2 (SPEA2),[24] Pareto Envelope-Based Selection Algorithm 2 (PESA2)[25]—and the original MOHCO approach are also considered in this study as competing candidates to quantify the relatively superior performance of the MOEHCO approach when solving customer-segmentation problems.

Major contributions of this study are as follows:

- The proposed MOEHCO-based FS method that benefits from the original MOHCO and three novel strategies is the first of its kind.
- This is the first study that applies an improved MOHCO-based wrapper FS method for customer segmentation.
- Experimental results obtained in this study reveal that the proposed MOEHCO-based FS algorithm outperforms other classical approaches when solving customer-segmentation problems.

The remainder of this paper is organized as follows: Section 2 describes extant research concerning FS methods. Section 3 describes the proposed method, including the fitness function, three MOEHCO strategies, and general framework. Section 4 presents the comparison between experimental results obtained using the proposed and other candidate approaches. Finally, Section 5 lists major conclusions drawn from this study along with a discussion of future research prospects.

2 | RELATED WORK

2.1 | Conventional FS

As already stated, FS techniques can be classified as filters and wrappers. Filter methods evaluate feature subsets while considering certain predefined metrics or information instead of a learning

algorithm. The learning algorithm and selected features are independent of each other. Overall, the filter FS algorithms afford high speed and scalability to the users. However, they might underperform in cases involving highly correlated features, thereby resulting in the generation of a local optimum.[21] In comparison, wrapper methods utilize the learning algorithm as a black box to score feature subsets. Moreover, the search space in wrapper approaches is considerably larger compared with that in filter methods. Further, the wrapper approach avoids deflections caused by the independence between the evaluation criteria and learning algorithm.[10] However, despite their popularity, wrapper methods encounter several limitations, including (1) high computational complexity, (2) large differences in the optimal feature subsets for different learners, (3) need for time-consuming determination of user-specified parameters for learning algorithms, and (4) only few learners being capable of dealing with multiclass classifications.[3]

2.2 | Metaheuristic-based FS

To address aforementioned limitations of wrapper approaches, several researchers have attempted to evolve traditional FS methods using metaheuristic algorithms. Metaheuristic-based FS methods have reportedly demonstrated superior performance when solving several single-objective optimization problems. Zorarpacı and Özel[26] combined the ACO and DE algorithms to improve the performance of general classification tasks, thereby outperforming traditional FS methods, such as the chi-square (CHI), information gain (IG), and correlation feature selection (CFS), in finding optimal feature subsets. Tabakhi and Moradi[27] combined the ant-colony optimization with traditional filter-based FS methods by evaluating the relevance and redundancy of features. They reported realization of a lower classification-error rate compared to certain univariate and multivariate FS methods. Zhang et al.[28] combined the binary PSO approach with a mutation operator search strategy. Further, they used the decision-tree classifier to improve the classification accuracy of a wrapper-based FS method.

Considering the combinatorial nature of several practical problems, researchers should pursue at least two objectives during feature-subset selection. Subsequently, they could extend the research focus to employing metaheuristic methods for solving multiobjective FS problems. For instance, Xue et al.[29] developed two multiobjective PSO-based FS methods. One is based on idea of nondominated sorting and the other is inspired from the concepts of crowding, mutation, and dominance, respectively. Their results indicate the superiority of these approaches in terms of both the classification accuracy and features size. Zhang et al.[30] developed the DE method as a binary approach in combination with a self-learning strategy to obtain solutions to multiobjective FS problems. Their proposed approach demonstrated a low error rate and solution count. Baraldi et al.[31] proposed a multiobjective GA-based wrapper method for nuclear-transient classification. They employed a search strategy designed to release the convergence pressures by different niches of the Pareto front.

2.3 | Multiobjective metaheuristic-based FS for customer segmentation

For marketers, an effective solution to the customer-segmentation problem must integrate multiple decision criteria that play important roles in determining the profitability and marketability of products and/or services. Therefore, customer segmentation, which is an

inherently multicriterion problem, must be considered a multiple-objective optimization problem. Huang et al.[32] utilized the multiobjective NSGA-II approach to select subset features for predicting customer churn. Gorzałczany and Rudziński[33] adopted a multiobjective genetic-optimization-based FS method and proposed a fuzzy rule-based classifier to divide credit customer groups. However, these methods cannot significantly increase the classification accuracy while maintaining a high solution diversity when solving customer-segmentation problems. As already stated, the HCO approach offers global search capability and minimizes the risk of local-optimum generation. It has been successfully employed to solve both single- and other multiobjective problems. Therefore, we believe that the application of the HCO method can be used to address the gap in existing research by overcoming the limitations of existing metaheuristic-based FS approaches that require tradeoffs between the convergence and solution diversity as well as those between the computation time and prematurity problems.

3 | MOEHCO-BASED FS METHOD

As already mentioned, customer segmentation is an FS problem, and it is useful for marketers targeting a specific customer group with fewer customer attributes. Accordingly, customer segmentation can be converted into an FS problem with two conflicting objectives—(1) minimizing classification error rate and (2) minimizing data dimensionality by reducing the number of attributes. To solve this problem, a MOEHCO is proposed in this study. Before describing the MOEHCO algorithm, this section first introduces the original HCO approach. Subsequently, we describe the encoding and decoding mechanisms of the proposed MOEHCO method along with the three strategies incorporated therein. Lastly, the general framework of MOEHCO approach is described in Section 3.4.

3.1 | HCO

The HCO approach[34] can be described using three important operators—flow, infiltration, and evaporation and precipitation—which simulate the natural water cycle. The detailed descriptions of these operators are as follows.

3.1.1 | Flow

Flow represents the core operator of the original HCO approach. Here, each individual P_i tends to move toward a better position P_j. If the fitness of P_{new} is better compared to that of the original position P_i, the ith individual will continue flowing along the direction of this new position until its new fitness becomes worse or the maximum number of flows $Flow_{Max}$ is attained. However, if the fitness of P_i is better compared with that of P_{new}, P_i retains its original position.

$$P_{new} = P_i + (P_j - P_i). * rand(1, n). \tag{1}$$

3.1.2 | Infiltration

In this process, individuals perform a random neighborhood search for execution of neighborhood learning. During this operation, the dimensions SD are randomly decided, and the position of the ith individual $P_{i,SD}$ is given by

$$P_{i,SD} = P_{i,SD} + (P_{i,SD} - P_{j,SD}) * 2. * (rand(1, SD) - 0.5). \tag{2}$$

3.1.3 | Evaporation and precipitation

In this process, each individual is evaporated by a probability P_{eva}. The evaporation condition is satisfied if an individual precipitates to (1) a random position satisfying the condition $rand(0, 1) < 0.5$ or (2) the best historical position of its neighbor via Gaussian distribution.

3.2 | MOEHCO representation and fitness function

Before describing the technical details concerning the MOEHCO approach, it is important to identify its representation scheme. In the MOEHCO, the position of each individual is represented by a vector comprising n real numbers in a feasible search space. Here, n denotes the total number of features in the original feature set. The position of each individual could be defined as $X = [x_1, x_2, \cdots, x_n]$, where x_I represents the position of the ith feature and its value varies between [0, 1]. In this study, we adopted a simple threshold-decoding approach with the value of the threshold θ set to 0.5. That is, if $x_I > 0.5$, the ith feature is selected and vice versa.

As already stated, FS problems have two objectives. Thus, in this study, the classification error rate was considered the first objective function. Its value was calculated by the classifier using the following equation:

$$error = \frac{FP + FN}{FP + FN + TP + TN}. \tag{3}$$

In the above equation, FP, FN, TP, and TN refer to false positives, false negatives, true positives, and true negatives, respectively.

3.3 | Three strategies of proposed MOEHCO

To improve the adaptation and robustness of the original MOHCO approach in solving different customer-segmentation cases, this study considers the implementation of three strategies, as described below.

(1) Global flow operator: In accordance with this strategy, individuals do not learn from a better position in the population space. Instead, the learning is performed from random non-dominated elites in the external repository (*REP*). This strategy effectively guides the evolutionary direction of individuals and takes advantage of the *REP* quality.

(2) Possibility-based acceptance criteria: The proposed algorithm not only accepts better non-dominated individuals but also individuals of the same quality with a certain probability. This strategy is conducive to accelerate population renewal and increase the diversity of the solutions obtained.

(3) Density-based evaporation and precipitation: This strategy calculates the distance between individuals in the population space and measures the density of each individual position. Subsequently, the probability of evaporation and precipitation is increased in the dense areas to enhance the algorithm's global search capability and avoid premature problems.

3.3.1 | Global flow

As already stated, flow is a critical operator in the HCO search process, wherein individuals can move toward better positions in the population space. In the original HCO, individuals could learn from anyone better than themselves, and this approach facilitates easy solving single-objective-optimization problems. However, multiobjective optimization problems are faced with insufficient quality of individuals in the population space and the resulting low convergence rate. In accordance with this strategy, the algorithm stores a series of non-dominated solutions with the best current performance in *REP*. Thus, it is desirable to utilize the optimum solutions stored in the *REP*, instead of creating a repository that only records and outputs optimum solutions, as reported by most prior studies.[17,19] Therefore, the proposed MOEHCO algorithm considers individuals to flow randomly toward a nondominant solution in the *REP* in accordance with the following equation:

$$P_i = P_i + r * (PND_i - P_i), \tag{4}$$

where P_i denotes the position of ith individual, PND_i denotes the position of the selected nondominant solution, and r denotes a random vector with values in the 0–1 range. Next, the acceptance criteria introduced in the next sub-section determine the acceptability of the new position. The accepted individuals continue flowing, and this cycle continues until the maximum number of flows is attained or a new position of an individual is not accepted. Subsequent operations are then performed in sequence.

3.3.2 | Possibility-based acceptance criteria

This strategy is used when the position of an individual changes after performing flow and/or evaporation and precipitation to determine whether their position is updated or remains unchanged. In the original HCO, individual could only update their position when the new position was compared to the previous one, i.e. both objectives demonstrated improved fitness values.

Nonetheless, it has been observed that in most cases, flow and evaporation are less effective, and thus, new individual positions are hardly accepted. This makes it difficult for the entire population to adapt to the environment, thereby reducing the convergence rate. Thus, new acceptance criteria are considered in this study to increase the population diversity.

Accordingly, fitness values (for objectives) corresponding to the new positions of individuals could become better compared to those corresponding to the original positions. The governing equations can be expressed as

$$
P = \begin{cases} P_{new}, & \text{if} \quad P_{new} \prec P_{old} \\ P_{new}, & \text{if} \quad P_{old} \text{ and } P_{new} \text{ do not dominate each other} \ \cap \ rand < Poss_{acc} \\ P_{old}, & \text{if} \quad P_{old} \prec P_{new} \end{cases} \tag{5}
$$

where P denotes the current position of an individual, P_{new} denotes the new position obtained after execution of the flow or evaporation operators, P_{old} denotes the position before operator execution, $rand$ denotes a random number between zero and unity, and $Poss_{acc}$ denotes a threshold. The larger the value of $Poss_{acc}$ the more likely is the renewal of the population and the greater is its diversity. When $Poss_{acc} = 0$, the proposed algorithm performs similar to the original HCO.

3.3.3 | Density-based evaporation and precipitation

Performing evaporation or precipitation based on a fixed probability has several disadvantages, as described below.

(1) Good and bad individuals have the same probability to change their positions. This might cause the loss of an optimum solution.
(2) The said probability cannot be easily determined. If it is too high, large disturbances are observed, and this reduces the solution convergence rate once the population search begins. In contrast, if the probability is too low, the population fails to obtain diversity, thereby resulting in the generation of a local optimum.
(3) A fixed probability implies that the degree of perturbation remains constant throughout the search process. However, in reality, the degree of disturbance should gradually increase as the search progresses to boost randomness.

To overcome these disadvantages, this study proposes use of a density-based strategy to exaggerate the individual probability in dense areas under evaporation while eliminating the probability of those in sparse areas. This helps the proposed algorithm realize the attainment of global equilibrium exploration by taking full advantage of the search capabilities of each individual. Considering that individuals tend to search dense areas repeatedly while sparse areas are less explored, optimal solutions are likely to be missed out. Equation (8) describes the expression for probability calculation proposed in this study. The sequence of operations is as follows. The sum of the distance between the ith and all other individuals in the population space is denoted by Dis_i. It is first calculated using Equation (6). Moreover, the search range of the algorithm varies from one problem or function to another, thereby resulting in a large difference in the value of the probability PE. Therefore, to design a feasible density-based evaporation and precipitation mechanism, we propose the degree of digestion Deg_i expressed as the ratio of Dis_i to the average distance between individuals in the population space. Thus, if the value of Dis_i is lower compared with the average distance, Deg_i is less than unity and vice versa. Therefore, in accordance with Equation (8), the smaller the value of Deg_i, the higher is the individual probability PE_i. In

other words, the more crowded the population space, the higher is the probability of eva-
poration and/or precipitation:

$$Dis_i = \sum_{j=1}^{N_{pop}} |P_i - P_j|, \tag{6}$$

$$Deg_i = \frac{Dis_i}{\left(\sum_{j=1}^{N_{pop}} Dis_j\right)/N_{pop}}, \tag{7}$$

$$PE_i = e^{-(Deg_i)^a}. \tag{8}$$

In the above equations, N_{pop} denotes the population count and a denotes the fluctuation
coefficient. The larger the value of a, the higher is the sensitivity of PE to Deg. That is,
individuals with Deg below unity would demonstrate higher PE and vice versa.

3.4 | REP-update strategy

In addition to the traditional sorting methods for eliminating inferior solutions, the proposed
algorithm employs the adaptive grid approach to update and maintain the individual count in
the *REP*. That is, if the newly identified individuals are dominated by those existing within the
REP, the same is not updated. However, if the new individuals dominate some of the current
individuals in the *REP*, the *REP* is updated to include these new individuals and eliminate the
dominated ones. Of course, the best solution is directly added to the *REP*. Finally, the adaptive-
grid crowding procedure is activated once the *REP* individual count exceeds the storage ca-
pacity of *nREP*. This strategy helps dividing target space into grid regions, which in turn, causes
the resulting Pareto frontier to be uniformly distributed. This results in high computational
efficiency and the solution diversity.[35]

3.5 | Handling constraints

Boundary processing must be performed once an operator is executed to ensure the feasibility
of the obtained solutions when an individual changes its position. A simple means to handle
constraints involves resetting the values that cross the boundary by appropriate values of the
upper or lower bounds. However, using such an approach implies the dimensions are set to the
same value irrespective of the extent to which they cross the boundary. This inevitably reduces
the randomness of the algorithm. To address such shortcoming, we include additional per-
turbations into boundary-processing workflow, as described in the following equation:

$$P_i(j) = \begin{cases} \text{if } rand > 0.5, \begin{cases} L_{up}, & \text{if} \quad P_i(j) > L_{up} \\ L_{dow}, & \text{if} \quad P_i(j) < L_{dow} \end{cases} \\ \text{if } rand < 0.5, \begin{cases} L_{up} - c * rand2 * (L_{up} - L_{dow}), & \text{if} \quad P_i(j) > L_{up} \\ L_{dow} + c * rand2 * (L_{up} - L_{dow}), & \text{if} \quad P_i(j) < L_{dow} \end{cases} \end{cases} \tag{9}$$
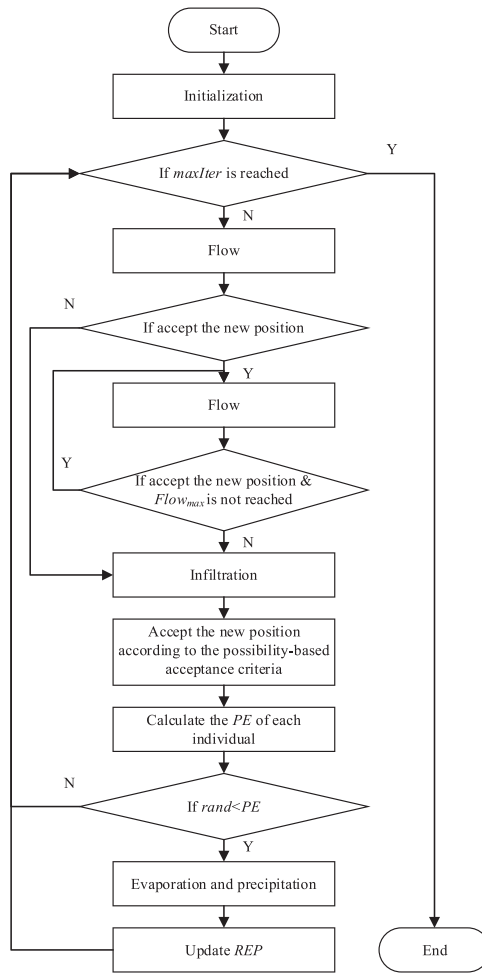
FIGURE 1 Proposed multiobjective evolutionary hydrological cycling optimization workflow

Here, $P_i(j)$ denotes jth dimension of ith individual's position, *rand* denotes a random number between zero and unity, L_{up} and L_{dow} denote the upper and lower bounds of the search boundary, and c denotes disturbance factor. When c equals unity, the approach becomes similar to randomly resetting $P_i(j)$.

3.6 | General framework of MOEHCO-based FS method

Figure 1 presents the general implementation framework of the proposed MOEHCO algorithm that the three above-described strategies. The pseudo code of MOEHCO-based FS method is also presented below.

The main idea of this framework can be concluded as (1) In view of FS problem, the encoding and decoding strategy of individual's position is analyzed. Moreover, specific constraint handling methods are included. (2) To effectively solve two objectives of FS, three important strategies have been introduced to the original MOHCO.

Algorithm: Pseudo code of proposed MOEHCO-based FS method
Initialization:
1   Divide the dataset into training and test datasets;
2   Define maximum number of iterations (*maxIter*) and initialize parameters, including maximum size of population *POP* and *REP* (i.e., *nPOP* and *nREP*), $Flow_{Max}$, $Poss_{acc}$, and *a*. Let $Iter = 0$
3   Randomly initialize the individual positions in *POP*
4   Evaluate values of two objectives for *POP;* update *REP* using the **REP update strategy**
Search procedure:
5   **While** (the *maxIter* is not attained) **do**
6   Individuals in *POP* flow to new positions using the **Global flow strategy**
7   Evaluate values of two objectives for *POP* and update *REP* using the **REP update strategy**
8   **If** new position is acceptable per **acceptance criteria**&$n < Flow_{Max}$**do**
9   Reimplement **Global flow**
10   Repeat step 7
11       $n = n + 1$
12   **End If**
13   Individuals move towards or away from the neighbor using **Infiltration operator**
14   Evaluate *POP*
15   Move individuals to new positions according to **acceptance criteria**
16   Individuals are regenerated or disturbed using **Density-based evaporation and precipitation strategy**
17   Handle the bounds of *POP* using **Handling constraints**
18   Evaluate values of two objectives for *POP* and update *REP* using the **REP update strategy**
19       $Iter = Iter + 1$
20   **End While**
21   Calculate the testing classification error rate of the feature subset in *REP* for the test dataset
22   Return the feature subset and feature count in *REP*
23   Calculate the total classification error rate according to **equation** (4) and return the same

4 | EXPERIMENTS AND RESULT DISCUSSION

This section describes the performance of the proposed MOEHCO approach when compared to that of the original MOHCO approach and other multiobjective optimization algorithms (MOPSO,[17] MOEA-D,[18] NSGA-II,[19] IM-MOEA,[23] SPEA2,[24] and PESA2[25]). Section 4.1 illustrates the parameter settings while Section 4.2 describes the three data sets used in this study. The experimental results and their subsequent analyses are described in Section 4.3.

4.1 | Parameter settings

In this study, all numerical experiments were performed in the MATLAB R2019a (MathWorks Inc.) environment. Table 1 lists test parameters considered for the application of the eight candidate algorithms. The parameters for original HCO algorithms are as per.[34,36] The $Flow_{Max}$ for MOEHCO is the same. The fluctuation coefficient a and disturbance factor c are specific to MOEHCO. The general experiment settings are described in the right half of Table 1. The

TABLE 1 Parameter setting considered in this study

Parameter	Value	Parameter	Value
$Flow_{Max}$	3	Population size	50
a	3	REP size	50
c	0.2	Maximum iterations	100
		Runs	30

parameter settings specific to the other six algorithms are consistent with their corresponding references citations.

All candidate algorithms considered in this study are wrapper methods based on an artificial neural network (ANN)[37] as the classifier. During classification, the training and test sets comprised 70% and 30%, respectively, of the customer information available within the original data set. The ANN comprised 10 hidden layers. For calculating the classification error rate, the features selected after the training process will be evaluated on the test set.

4.2 | Data set setting

Three standard customer-segmentation datasets—Australian Credit Approval Data, German Credit Data and Default of Credit Card Clients Data—were used in this study. Table 2 lists the general data set information.

The Australian data set includes credit-approval information of customers in Australia. To ensure customer privacy, all attribute names and values were modified to meaningless symbols. The customer attributes in the German data set include 7 numerical and 13 categorical variables. The Default data set comprises approval data pertaining to defaulting credit-card clients in Taiwan in 2016. It divides customer data into two categories based on an indicator that predicts the occurrence of credit-card defaults in the following month.

4.3 | Experimental results and analyses

4.3.1 | Case 1—Australian Credit Approval Data

Owing to the small feature dimensions and customer-information volume of Australian data set, its corresponding optimization problem was the least complex to solve using the different

TABLE 2 Data set description

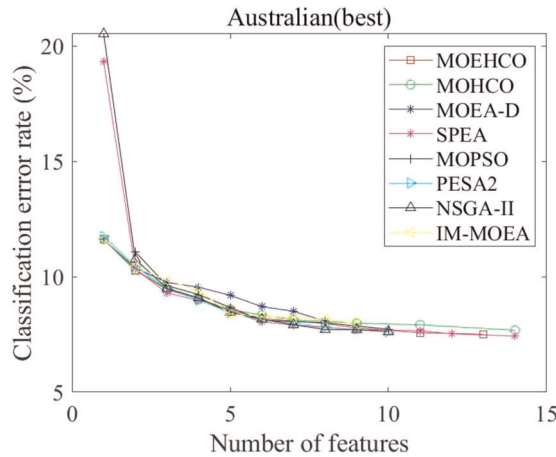| Data set Abbreviation	Data size	Number of attributes	Number of classification items
Australian	690	14	2
German	1000	20	2
Default	30,000	23	2

FIGURE 2 Best Pareto solution sets obtained by all algorithms for Australian data set [Color figure can be viewed at wileyonlinelibrary.com]

candidate approaches. As illustrated in Figure 2, the quality of best Pareto solution obtained for all algorithms does not vary significantly.

Reference to Figure 2 and Table 3 reveals that the proposed MOEHCO approach outperforms all other candidate algorithms in terms of the solution diversity and optimality of both objectives. The optimum solution set obtained by the MOEHCO comprises 12 solutions, thereby providing more options for decision makers. In particular, for solutions with low dimensions (1–3 features), some candidate algorithms failed to obtain nondominant solutions. In comparison, the MOEHCO could always find solutions with optimum quality in most dimensions.

It is noteworthy that although the MOHCO, PESA2, and IM-MOEA lack solutions with high dimensionality, they perform reasonably well in cases involving low and intermediate feature dimensions. Further, the SPEA could obtain solutions in both low- and high-dimensionality.

TABLE 3 Comparison between best Pareto solutions (Australian data set)

Algorithms	Solution count	Objective	$f1$	$f2$ (%)	Objective	$f1$	$f2$ (%)
MOEHCO	**12**	The solution with minimum features ($f1$)	**1**	**11.59**	The solution with minimum error rate ($f2$)	**14**	**7.42**
MOHCO	10		1	11.62		11	7.68
MOEA-D	8		1	11.62		14	8.07
SPEA	11		1	19.32		14	7.43
MOPSO	7		2	11.78		14	7.72
PESA2	10		1	11.75		10	7.61
NSGA-II	10		1	20.52		10	7.62
IM-MOEA	10		1	11.63		9	7.96

Note: Bold values indicate the best values obtained by all algorithms for solution count, objective 1 or objective 2.

However, it demonstrates poor accuracy, particularly in cases involving a single feature dimension. The MOPSO and MOEA-D demonstrate low-diversity solutions in cases with 7 and 8 features, respectively, albeit their search capability is moderate. Finally, NSGA-II demonstrates intermediate diversity and solution quality. However, it fails to obtain feasible solutions in cases involving a large number of features and underperforms in low dimension scenarios (1–2 features).

Figure 3 reveals all algorithms perform stably in cases involving intermediate and high feature dimensionality. However, the MOEA-D demonstrates large fluctuations and higher error rates in cases involving 8–14 feature dimensions. In low-dimensionality cases (1–4 features), six algorithms other than the MOEHCO and MOHCO demonstrate sharp fluctuations for obtaining solutions for each dimension. However, it is noteworthy that the PESA2 obtains slightly better solution for the two-feature-dimension case. Overall, the MOEHCO and MOHCO demonstrate the best computational stability with minor fluctuations.

4.3.2 | Case 2—German Credit Data

Given the larger size and dimensionality of the German credit data set compared to its Australian counterpart, its optimization problem is significantly more complex to solve. As can be realized from Figure 4, the IM-MOEA, MOPSO, and MOEA-D demonstrate poor performance in most instances. The solutions obtained using these approaches are inferior to those obtained using other algorithms. None of these approaches could obtain good solutions in cases involving low and high feature dimensions.

As illustrated in Table 4 and Figure 4, The SPEA fails to obtain low-dimensional solutions (1–2 features). However, half solutions obtained using SPEA dominated those obtained using all other approaches for the same feature dimensions—5, 6, 9, 11, and 12. Likewise, the NSGA-II fails to obtain low-dimensional solutions (1–2 features). It underperforms in cases involving low feature dimensions (3–9 features). However, it yields the most diverse high-dimensional solutions and obtains the best solution in cases involving 13, 15, and 18 features.
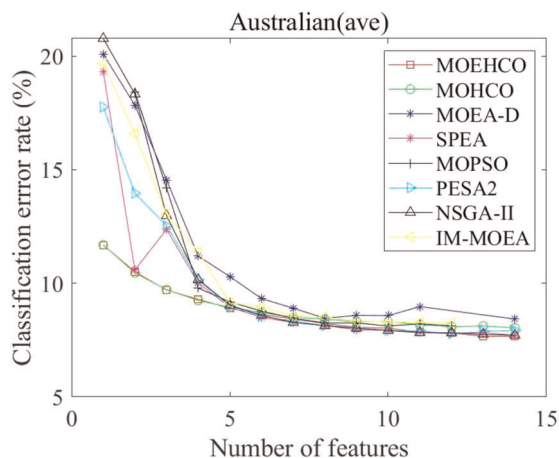

FIGURE 3 Average values of Pareto solution sets by all algorithms for Australian data set [Color figure can be viewed at wileyonlinelibrary.com]
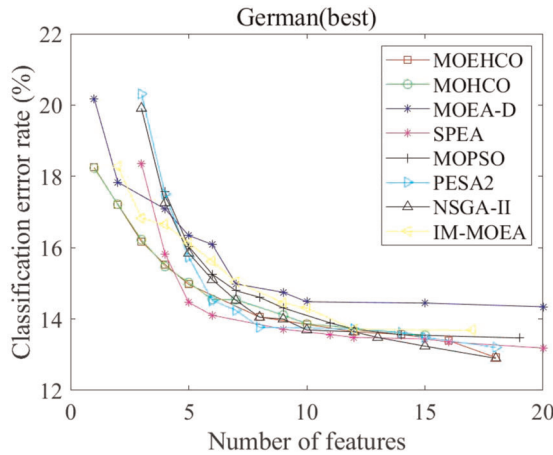
FIGURE 4 Best Pareto solution sets by all algorithms for German data set [Color figure can be viewed at wileyonlinelibrary.com]

Both the MOEHCO and MOHCO algorithms perform well with regard to minimizing the number of features. However, results obtained using the MOHCO demonstrate less satisfactory classification accuracy in cases involving low-to-medium (7–9) feature dimensions. Moreover, it fails to obtain a solution in cases with high (16–20) feature dimensions. Owing to the use of improved strategies, the MOEHCO demonstrated a slightly increased solution diversity compared to MOHCO while its classification accuracy in high-dimensional feature cases is significantly improved. Though the proposed approach does not obtain the optimum classification-error rate, it demonstrates the second-best performance by a very small (0.02%) margin.

Figure 5 reveals that the increased computation stability of the proposed algorithm and its performance superiority over other algorithms becomes apparent with increased customer-classification complexity. As can be seen, except the two HCO-based approaches, all other algorithms demonstrate unstable performance in cases involving low feature dimensions.

TABLE 4 Comparison between best Pareto solutions (German data set)

Algorithms	Solution count	Objective	f1	f2 (%)	Objective	f1	f2 (%)
MOEHCO	**11**	The solution with minimum features (f1)	**1**	**18.23**	The solution with minimum error rate (f2)	18	12.92
MOHCO	10		1	18.23		15	13.54
MOEA-D	10		1	20.18		20	14.34
SPEA	10		3	18.36		20	13.18
MOPSO	10		4	17.57		19	13.47
PESA2	10		3	20.31		18	13.21
NSGA-II	12		3	19.91		**18**	**12.90**
IM-MOEA	10		2	18.28		17	13.69

Note: Bold values indicate the best values obtained by all algorithms for solution count, objective 1 or objective 2.
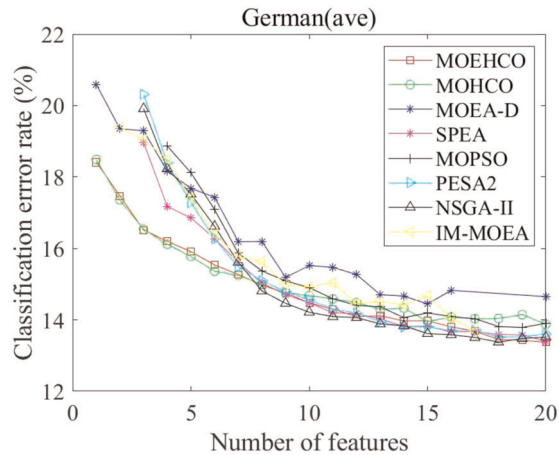
FIGURE 5 Average values of Pareto solution sets by all algorithms for German data set [Color figure can be viewed at wileyonlinelibrary.com]

Although the SPEA yields the best solution in some cases, its classification accuracy varies dramatically, thereby resulting in an overall unsatisfactory performance. In contrast, the two HCO-based multiobjective algorithms perform stably irrespective of the feature dimensionality, thereby affording users an overwhelming advantage when solving problems with low feature dimensionality.

4.3.3 | Case 3—Default of Credit Card Clients Data

Figure 6 and Table 5 reveal that the larger customer-information data set in Case 3 (30,000 records) facilitates the MOEHCO classifier to be well trained, thereby achieving lower classification-error rates in the 13.24%–14.10% range. Moreover, the aforementioned superiorities of the MOEHCO and MOHCO approaches are prominent in this case as well. Both
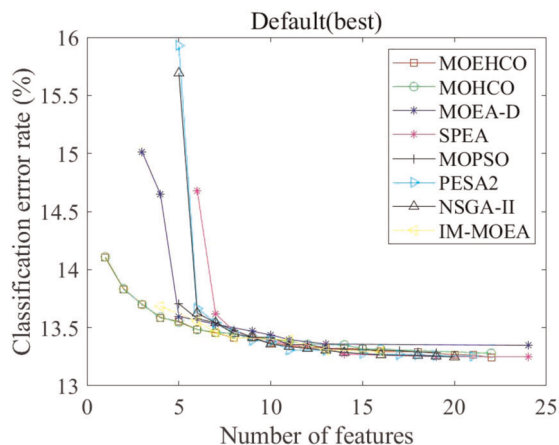

FIGURE 6 Best Pareto solution sets by all algorithms for Default data set [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 5 Comparison between best Pareto solutions (Default data set)

Algorithms	Solution count	Objective	f1	f2 (%)	Objective	f1	f2 (%)
MOEHCO	**19**	The solution with minimum features (f1)	**1**	**14.10**	The solution with minimum error rate (f2)	22	**13.24**
MOHCO	16		1	14.11		20	13.31
MOEA-D	9		3	15.01		24	13.35
SPEA	11		6	14.68		24	13.25
MOPSO	13		5	13.71		19	13.29
PESA2	13		5	15.93		21	13.26
NSGA-II	14		5	15.69		20	13.25
IM-MOEA	7		4	13.68		16	13.29

Note: Bold values indicate the best values obtained by all algorithms for solution count, objective 1 or objective 2.

approaches perform better compared to all other approaches with regard to obtaining solutions in cases involving low feature dimensions (1–8 features). Although the MOEA-D could obtain few solutions with low feature dimensions, their observed quality was much lower compared to those obtained using the two HCO methods. With increase in feature dimensions, the proposed MOEHCO continues to demonstrate competitive performance in terms of high solution diversity. It achieves a good solution for f2, its performance falling short of only those of the NSGA-II and SPEA.

As displayed in Figure 7, while all other algorithms demonstrate major fluctuations in the 3–8 feature-dimension range, both MOEHCO and MOEHCO demonstrate comparatively stable performance at all feature dimensions, thereby indicating high-performance reliability.

Overall, the proposed MOEHCO algorithm demonstrates stable operation and good search capability when solving the three customer-segmentation problems considered in this study.
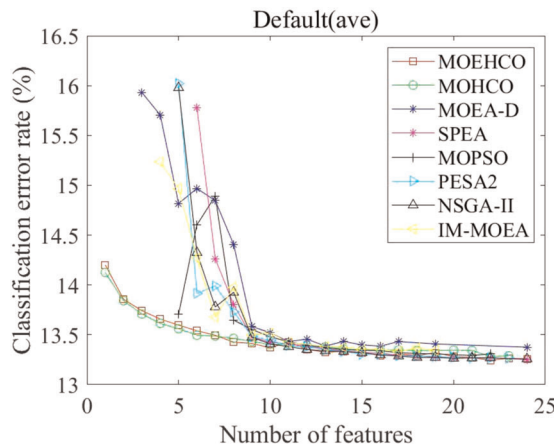

FIGURE 7 Average values of Pareto solution sets by all algorithms for Default data set [Color figure can be viewed at wileyonlinelibrary.com]

Compared with the original MOHCO, the proposed approach demonstrates significant improvements in the solution quality and diversity. These results confirm the contribution of the proposed strategies toward the realization of improved segmentation performance of the MOEHCO. The global flow operator facilitates the search for better solutions while the possibility-based acceptance criteria and density-based evaporation and precipitation strategies enhance the population diversity and balance the exploration and exploitation ability of the proposed algorithm.

5 | CONCLUSIONS AND FUTURE RESEARCH

This paper presents a novel metaheuristic-based wrapper FS algorithm—MOEHCO—to solve customer-segmentation problems. The increased effectiveness and optimum performance of the proposed algorithm can be attributed to the implementation of three strategies—global flow operator, possibility-based acceptance criteria, and density-based evaporation and precipitation.

The results obtained in this study reveal the proposed approach to demonstrate optimum search capability and high solution quality across the range of feature dimensions, thereby ensuring high solution diversity. Moreover, MOEHCO demonstrates the best computation stability among competing algorithms, especially in cases involving low feature dimensions, when applied to the three datasets considered in this study.

This being the maiden application of a multiobjective HCO-based approach to customer segmentation, we reckon further application potentials of the same could be explored by considering more sophisticated classification tasks, complex datasets, and performance comparisons against state-of-the-art algorithms. Moreover, we believe the proposed approach can be further modified by combining traditional FS with hybrid and multistaged metaheuristic algorithms. This would facilitate the development of new encoding methods and search mechanisms. Moreover, there exist possibilities to improve and develop different classifiers for the proposed FS algorithm.

CONFLICT OF INTERESTS
The authors declare that there are no conflict of interests.

AUTHOR CONTRIBUTIONS
Conceptualization: Xi Song and Matthew Tingchi Liu. *Methodology*: Xi Song and Qianying Liu. *Validation*: Matthew Tingchi Liu. *Formal analysis*: Xi Song and Qianying Liu. *Writing—original draft preparation*: Xi Song. *Writing—review and editing*: Matthew Tingchi Liu and Ben Niu. *Project administration*: Matthew Tingchi Liu and Ben Niu. *Funding acquisition*: Ben Niu. All authors have read and agreed to the published version of the manuscript.

ORCID

Xi Song 🔾 http://orcid.org/0000-0001-6697-8640
Matthew Tingchi Liu 🔾 http://orcid.org/0000-0002-1820-5691
Qianying Liu 🔾 https://orcid.org/0000-0003-0765-7243
Ben Niu 🔾 http://orcid.org/0000-0001-5822-8743

REFERENCES

1. Cadenas JM, Garrido MC, MartíNez R. Feature subset selection filter–wrapper based on low quality data. *Expert Syst Appl.* 2013;40(16):6241-6252.
2. Guyon I, Elisseeff A. An introduction to variable and feature selection. *J Mach Learn Res.* 2003;3:1157-1182.
3. Tang J, Alelyani S, Liu H. Feature selection for classification: A review. In: Aggarwal CC, eds. *Data Classification: Algorithms and Applications.* Boca Raton, Florida: CRC Press; 2014:37-64.
4. Somol P, Baesens B, Pudil P, Vanthienen J. Filter-versus wrapper-based feature selection for credit scoring. *Int J Intell Syst.* 2005;20(10):985-999.
5. Liu Y, Wang G, Chen H, Dong H, Zhu X, Wang S. An improved particle swarm optimization for feature selection. *J Bionic Eng.* 2011;8(2):191-200.
6. Nebro AJ, Durillo JJ, Garcia-Nieto J, Coello CC, Luna F, Alba E. SMPSO: a new PSO-based metaheuristic for multi-objective optimization. 2009 IEEE Symposium on computational intelligence in multi-criteria decision-making (MCDM), Nashville, TN; 2009:66-73.
7. Unler A, Murat A. A discrete particle swarm optimization method for feature selection in binary classification problems. *Eur J Oper Res.* 2010;206(3):528-539.
8. Deb K, Pratap A, Agarwal S, Meyarivan TA. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput.* 2002;6(2):182-197.
9. Huang J, Cai Y, Xu X. A hybrid genetic algorithm for feature selection wrapper based on mutual information. *Pattern Recognit Lett.* 2007;28(13):1825-1844.
10. Li S, Wu H, Wan D, Zhu J. An effective feature selection method for hyperspectral image classification based on genetic algorithm and support vector machine. *Knowl-Based Syst.* 2011;24(1):40-48.
11. Hancer E, Xue B, Zhang M. Differential evolution for filter feature selection based on information theory and feature ranking. *Knowl Based Syst.* 2018;140:103-119.
12. Nemati S, Basiri ME, Ghasem-Aghaee N, Aghdam MH. A novel ACO–GA hybrid algorithm for feature selection in protein function prediction. *Expert Syst Appl.* 2009;36(10):12086-12094.
13. Liu Q, Niu B, Wang J, Wang H, Li L. Nurse scheduling problem based on hydrologic cycle optimization. 2019 IEEE Congress on Evolutionary Computation (CEC), Wellington; 2019:1398-1405.
14. Wang J, Chen J, Liu Q, Liu J. HCO-based RFID network planning. In: Liu T, Wang J, Sun Z, Luo J, He TT, Chen K, eds. *Bio-inspired Computing: Theories and Applications. BIC-TA. Communications in Computer and Information Science.* Singapore: Springer; 2018:435-445.
15. Geng S, Zhang C, Yang X, Niu B. Multi-criteria recommender systems based on multi-objective hydrologic cycle optimization. In: Tan Y, Shi Y, Niu B, eds. *Advances in Swarm Intelligence, Lecture Notes in Computer Science.* Cham: Springer; 2019:92-102.
16. Chuang LY, Yang CH, Li JC. Chaotic maps based on binary particle swarm optimization for feature selection. *Appl Soft Comput.* 2011;11(1):239-248.
17. Coello CC, Lechuga MS MOPSO: A proposal for multiple objective particle swarm optimization. In: Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No. 02TH8600). Honolulu, HI: IEEE; 2002:1051-1056.
18. Zhang Q, Li H. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Trans Evol Comput.* 2007;11(6):712-731.
19. Deb K, Pratap A, Agarwal S, Meyarivan TA. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput.* 2002;6(2):182-197.
20. Zadeh L. Optimality and non-scalar-valued performance criteria. *IEEE Trans Automat Contr.* 1963;8(1):59-60.
21. Coello CA, Lamont GB, Van Veldhuizen DA. *Evolutionary Algorithms for Solving Multi-Objective Problems.* New York: Springer; 2007.

22. Yang XS. Nature-inspired optimization algorithms. *J Comput Sci*. 2020, 46:101104.
23. Cheng R, Jin Y, Narukawa K, Sendhoff B. A multiobjective evolutionary algorithm using Gaussian process-based inverse modeling. *IEEE Trans Evol Comput*. 2015;19(6):838-856.
24. Zitzler E, Laumanns M, Thiele L. SPEA2: Improving the strength Pareto evolutionary algorithm. TIK-Report 103; 2001.
25. Corne DW, Knowles JD, Oates MJ. The Pareto envelope-based selection algorithm for multiobjective optimization. In: Schoenauer M, Deb K, Rudolph G, et al., eds. *International Conference on Parallel Problem Solving from Nature*. Berlin, Heidelberg: Springer; 2000:839-848.
26. Zorarpacı E, Özel SA. A hybrid approach of differential evolution and artificial bee colony for feature selection. *Expert Syst Appl*. 2016;62:91-103.
27. Tabakhi S, Moradi P. Relevance–redundancy feature selection based on ant colony optimization. *Pattern Recognit*. 2015;48(9):2798-2811.
28. Zhang Y, Wang S, Phillips P, Ji G. Binary PSO with mutation operator for feature selection using decision tree applied to spam detection. *Knowl-Based Syst*. 2014;64:22-31.
29. Xue B, Zhang M, Browne WN. Particle swarm optimization for feature selection in classification: a multi-objective approach. *IEEE Trans Cybern*. 2012;43(6):1656-1671.
30. Zhang Y, Gong DW, Gao XZ, Tian T, Sun XY. Binary differential evolution with self-learning for multi-objective feature selection. *Inf Sci*. 2020;507:67-85.
31. Baraldi P, Pedroni N, Zio E. Application of a niched Pareto genetic algorithm for selecting features for nuclear transients classification. *Int J Intell Syst*. 2009;24(2):118-151.
32. Huang B, Buckley B, Kechadi TM. Multi-objective feature selection by using NSGA-II for customer churn prediction in telecommunications. *Expert Syst Appl*. 2010;37(5):3638-3646.
33. Gorzałczany MB, Rudziński F. A multi-objective genetic optimization for fast, fuzzy rule-based credit classification with balanced accuracy and interpretability. *Appl Soft Comput*. 2016;40:206-220.
34. Yan X, Niu B. Hydrologic cycle optimization Part I: background and theory. In: Tan Y, Shi Y, Tang Q, eds. *International Conference on Swarm Intelligence*. Cham: Springer; 2018:341-349.
35. Coello CA, Pulido GT, Lechuga MS. Handling multiple objectives with particle swarm optimization. *IEEE Trans Evol Comput*. 2004;8(3):256-279.
36. Niu B, Liu H, Yan X. Hydrologic cycle optimization part II: experiments and real-world application. In: Tan Y, Shi Y, Tang Q, eds. *International Conference on Swarm Intelligence*. Cham: Springer; 2018: 350-358.
37. Hassoun MH. *Fundamentals of Artificial Neural Networks*. Cambridge, MA: MIT Press; 1995.