World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:12, No:9, 2018

A Fuzzy-Rough Feature Selection Based on Binary Shuffled Frog Leaping Algorithm

Javad Rahimipour Anaraki, Saeed Samet, Mahdi Eftekhari, Chang Wook Ahn

***Abstract*—**Feature selection and attribute reduction are crucial problems, and widely used techniques in the field of machine learning, data mining and pattern recognition to overcome the well-known phenomenon of the Curse of Dimensionality. This paper presents a feature selection method that efficiently carries out attribute reduction, thereby selecting the most informative features of a dataset. It consists of two components: 1) a measure for feature subset evaluation, and 2) a search strategy. For the evaluation measure, we have employed the fuzzy-rough dependency degree (FRFDD) of the lower approximation-based fuzzy-rough feature selection (L-FRFS) due to its effectiveness in feature selection. As for the search strategy, a modified version of a binary shuffled frog leaping algorithm is proposed (B-SFLA). The proposed feature selection method is obtained by hybridizing the B-SFLA with the FRDD. Nine classifiers have been employed to compare the proposed approach with several existing methods over twenty two datasets, including nine high dimensional and large ones, from the UCI repository. The experimental results demonstrate that the B-SFLA approach significantly outperforms other metaheuristic methods in terms of the number of selected features and the classification accuracy.

***Keywords*—**Binary shuffled frog leaping algorithm, feature selection, fuzzy-rough set, minimal reduct.

I. Introduction

FEATURE SELECTION (FS) is the process of selecting the most informative features of a dataset while removing the others, nd many studies have been done on diverse FS methods in recent years [1]-[8]. The feature selection process results in a reduction in the size of datasets and a retention of their critical information. Finding and removing irrelevant features (which have little/no effect on the classification results) and redundant features (which have high correlation with other features) would reduce the size of datasets, thereby improving the classification accuracy as well as the visualization and comprehensibility of the induced concepts. The third group is the set of features that should remain at the end of the FS process.

Selecting M out of N features by means of a comprehensive search is an NP-hard problem [9]. Furthermore, it has been proven that approximating the minimal relevant subset is hard up to very large factors [9]. Therefore, greedy search methods and metaheuristic search strategies are suitable for solving this problem [10]. However, all of the greedy search methods suffer from the deficiency of becoming trapped in local optima [10]. Forward and backward search mechanisms are instances of greedy search algorithms that are widely used for FS, because of their ideal time complexity; therefore, they are not capable of avoiding local optima [10], [11]. Due to this deficiency and the inherent ability of metaheuristic search methods to find the global optimum while avoiding local optima, these search methods have been widely utilized to solve FS problems [10]-[14].

Genetic algorithm (GA), particle swarm optimization (PSO), Tabu search and memetic algorithms are representative metaheuristic instances that, in recent years, have been very successful at solving various NP-hard engineering problems such as feature selection [10], [12]-[14]. Moreover, all of the above search mechanisms require an evaluation criterion for measuring the suitability of feature subsets. Based on determining the evaluation measures, a twofold taxonomy of feature selection methods has been presented in the literature [15]. In this taxonomy, feature selection strategies are categorized into 1) filter-based methods, and 2) wrapper-based methods. The former generally evaluate a feature subset by performing statistical tests on the data [15]. Thus, the filter-based methods “filter out” irrelevant features before the induction process (i.e. classification). In the wrapper-based approach, an induction algorithm itself (i.e. classifier) is utilized for evaluating feature subsets [15]. In other words, it is used for optimizing the accuracy rate estimated by an induction algorithm. Compared to filter-based methods, wrapper-based methods are computationally prohibitive since they employ an induction model as an embedded algorithm. On the other hand, the wrapper-based methods are more accurate at finding a proper subset of informative features than filter-based methods. In the filter-based technique, a non-statistical criterion can also be used as the evaluation measure. Examples of such criteria include the dependency degree (DD) based on rough set theory [16], and the fuzzy feature saliency measure [17] based on fuzzy set theory. Recently, much research has been performed on the development of methodologies for dealing with imprecision and uncertainty [16]-[18]. Fuzzy and rough set theories are analogous in the sense that they can model uncertainty and inconsistency. Recent studies have shown that they are complementary in nature.

Fuzzy-rough feature selection (FRFS) is one of the most successful hybrid tools for dimensional reduction, which is capable of handling both discrete and real-valued (or a mixture of both) variables [18]. However, there are some

J. R. Anaraki is with the Department of Computer Science, Memorial University of Newfoundland, St. John’s, NL, A1B 3X5 Canada (e-mail: jra066@mun.ca).

S. Samet is with the School of Computer Science, University of Windsor, Windsor, ON, N9B 3P4 Canada.

M. Eftekhari is with the Department of Computer Engineering, Shahid Bahonar University of Kerman, Kerman, 76169-14111 Iran.

C. W. Ahn is with the School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, 123 Cheomdan-gwagiro, Buk-gu, Gwangju 61005 Republic of Korea.

problems regarding the use of FRFS, thoroughly addressed in [19]. For instance, pre-data discretization by using fuzzy partitions is an FRFS approach that is not very successful in terms of computation. One of the newly developed FRFS methods is the lower approximation-based fuzzy-rough feature selection (L-FRFS) [19] method. L-FRFS, introduced in [19] is a fast FRFS, and it exhibits better performance compared to previously developed FRFSs. Moreover, as stated earlier, generating all subsets of features is an NP-hard problem and computationally prohibitive. Therefore, some hill-climbing search algorithms have been proposed in the literature in order to compensate for this computational deficiency [19].

The smallest subset of features with the highest DD is called the "minimal reduct"; it might not be found by the fuzzy-rough QuickReduct algorithm, which is an example of a hill-climbing method, both in terms of the resulting dependency measure and the subset size. Due to the deficiencies of hill-climbing approaches, metaheuristic algorithms such as GA and PSO are required in order to find such minimal reducts, especially when available data are high-dimensional. In [20]-[24] metaheuristic algorithms and rough set theory have been combined to find minimal reducts. In recent years, a few studies have also been presented in literature regarding the hybridization of fuzzy-rough and metaheuristic approaches [18], [19]. Very significant work is the combination of ACO and fuzzy-rough set for dimension reduction [25]. In this work, Jensen and Shen utilized a computationally demanding FRFS method in which continuous data have been discretized in advance by fuzzy partitions, and an ACO has been employed to find the minimal reduct [25]. As mentioned earlier, the authors have recently confirmed the time deficiencies of the fuzzy-rough method used in [19], and as an alternative have introduced the L-FRFS as a fast method.

In [26], Xiang et al. have proposed a hybrid method for feature selection by improving the diversity of species through piecewise linear chaotic maps (PWL), and increasing the speed of local search by applying sequential quadratic programming (SQP) to the binary gravitational search algorithm (GSA). The improved version of GSA has been hybridized with a 1-nearest neighbour method to from a feature selection system. A modified version of the binary PSO with the ability to avoid premature convergence utilizing both velocity and similarity of best solutions has been introduced by Vieira et al. [27]. The search method has been used to perform simultaneous feature selection and prediction of mortality of septic patients using concurrently optimized kernel parameters of a support vector machine (SVM). On of the most recent and successful feature selection methods is gradient boosted feature selection (GBFS) proposed by Xu et al. [28]. It works based on gradient boosted trees [29]. It starts by building regression trees using CART algorithm [30], and features are selected simultaneously based on deviation in impurity function. Selecting new feature is penalized and reusing already selected features has no cost.

In the present paper, a new FRFS technique is proposed on the basis of the B-SFLA and L-FRFS. Our contributions are twofold: 1) we devise a new binary version of an SFLA that employs a new dissimilarity measure, new coefficients for self-parameter selection, and a modified ranking rule, and 2) we develop an FS method by combining the strengths of this B-SFLA and the L-FRFS. The rest of this paper is organized as follows. In Section II, the background of the rough set and the shuffled frog leaping algorithm are presented. Section III illustrates the proposed feature selection method. Section IV reports experimental results and finally we conclude this paper in Section V.

II. Background

A. Rough Set

Rough set theory was proposed by Pawlak as a tool to deal, in an efficient way, with uncertainty [31], in data organized in a decision table. Let U be the universe of discourse and A be a nonempty finite set of attributes in U; information system is shown by $I = (U, A)$. Let X be a subset of U, and P and Q be two subsets of A; approximating a subset using rough set theory is done by means of upper and lower approximations. The upper approximation of X with regard to $(\overline{P}X)$ contains objects, which are possibly classified in X regarding the attributes in P. Objects in the lower approximation $(\underline{P}X)$ are those, which are definitely classified in X regarding the attributes in P. A rough set is shown by an ordered pair, $(\underline{P}X, \overline{P}X)$. The positive region as shown in (1) of partition $\mathbb{U}/Q$ is a set of all objects, which can be uniquely classified into blocks of the partition by means of P.

$$POS_P(Q) = \bigcup_{X \in \mathbb{U}/Q} \underline{P}X \tag{1}$$

Finding the dependency between attributes is one of the most important areas in data analysis. The dependency of Q on P is denoted by $P \Rightarrow_k Q$ and $k = \gamma_p(Q)$, in which γ is the dependency degree [32]. If $k = 1$ then Q completely depends on P and if $0 < k < 1$ then Q partially depends on P. The value of k is a measure of the dependency between the features P and Q. In feature selection, features which have lower dependency on each other and are highly correlated to the decision feature(s), are desired. If Q completely depends on P, then the partition which is made by P is finer than Q. The positive region of the partition $\mathbb{U}/Q$, with respect to P, which is denoted by $POS_P(Q)$, is the set of all elements which can be classified into the partition $\mathbb{U}/Q$ using P [32]. The following equation allows to calculate the dependency.

$$k = \gamma_P(Q) = \frac{|POS_P(Q)|}{|\mathbb{U}|}, \tag{2}$$

where notation $|.|$ is used for cardinality. The reduct is a subset of features which have the same dependency degree as employing all the features for classification. The features that belong to the reduct set are the most informative ones while the others are either irrelevant or redundant.

One way to handle real-valued data using rough set theory is to discretize continuous data in advance and make a new crisp valued dataset. Discretization is not enough as long as the similarity between two values remains unspecified [19]. Therefore, dependency degree between the features is calculated by means of the FRDD. The fuzzy-rough set basis will be addressed thoroughly in Section III.

B. Shuffled Frog Leaping Algorithm

The Shuffled frog leaping algorithm (SFLA) is a memetic metaheuristic search algorithm proposed by Eusuff et al. [33]; it is basically a combination of a shuffled complex evolution (SCE) algorithm [34] that ensures global exploration, and PSO [35] that is responsible for local search. Randomness and determinism are the results of this combination. The SFLA is based on memetics of frog-like beings. A meme is an idea or information pattern which is replicated or repeated to someone else. Memes and genes are analogous but are different in the way they propagate. A *meme* is propagated by leaping from one brain to another and can be transmitted between any individual, but a gene is propagated from parent to offspring by (sexual) reproduction.

The algorithm is inspired by real frog populations searching for food. In this algorithm, the behaviour of the population is determined by memes, and thus the population is more important than individuals. In the SFLA, frogs are partitioned into memeplexes that are evaluated individually. In each memeplex, frogs are influenced by each other and they experience meme evolution. Memetic evolution increases the frogs' performance in terms of reaching the goal by using information from the memeplex and the best performing individual in the population. This process continues for a predefined number of iterations. Then, all memeplexes are mixed with each other to form a new set of memeplexes through shuffling. Frogs with better performance contribute more to distribute new individuals in the population. A modified version of the SFLA has been proposed by Reddy et al. [36] for solving the environmentally-constrained economic dispatch problem. The modified algorithm uses a local search as well as a new parameter to accelerate convergence.

III. Proposed Feature Selection Approach

In this section, the proposed approach is defined based on the two main concepts of feature selection: 1-evaluation measure, and 2- search method. The evaluation measure is fuzzy-rough dependency degree (FRDD) and the search method is a binary modification of SFLA.

A. Evaluation Measure

The QuickReduct algorithm finds a reduct set without finding all the subsets [19]. It begins with an empty set and each time selects the feature that causes the greatest increase in dependency degree (DD). The algorithm stops when adding more features does not increase the DD. Since it employs a greedy algorithm, it does not guarantee that the minimal reduct set will be found. For this reason, a new FRFS algorithm is presented in this paper. Prior to providing the details of our approach, it is necessary to introduce the definition of the FRDD. To begin with, the definition of the X-lower and the degree of fuzzy similarity [19] are given by (3) and (4), respectively.

$$\mu_{\underline{R_P}X}(x) = \inf_{y\in\mathbb{U}} I\{\eta_{R_P}(x,y), \mu_X(y)\}, \tag{3}$$

$$\eta_{R_P}(x,y) = \bigcap_{a\in P}\{\eta_{R_a}(x,y)\}, \tag{4}$$

where I is a Łukasiewicz fuzzy *implicator*, which is defined by $min(1-x+y,1)$. In [37], three classes of fuzzy-rough sets based on three different classes of implicators, namely *S*-, *R*-, and *QL*-implicators, and their properties have been investigated. Here, R_P is the fuzzy similarity relation considering the set of features in P, and $\eta_{R_P}(x,y)$ is the degree of similarity between objects x and y over all features in P. Also, $\mu_X(y)$ is the membership degree of y to X. One of the best fuzzy similarity relations as suggested in [19] is given by (5).

$$\eta_{R_a}(x,y) = max\left\{min\left\{\frac{(a(y)-(a(x)-\sigma_a))}{(a(x)-(a(x)-\sigma_a))},\right.\right. \\ \left.\left.\frac{((a(x)+\sigma_a)-a(y))}{((a(x)+\sigma_a)-a(x))}\right\},0\right\}, \tag{5}$$

where σ_a is variance of feature a. The L-FRFS does not use the fuzzy partitioning used in FRFS, and thereby it is more computationally effective.

The FRFS can be conducted on the real-valued datasets using the lower approximation. The positive region in rough set theory is defined as a union of lower approximations. Referring to the extension principle [19], the membership of object x to a fuzzy positive region is given by (6).

$$\mu_{POS_P(Q)}(x) = \sup_{X\in\mathbb{U}/Q} \mu_{\underline{P}X}(x). \tag{6}$$

If the equivalence class that includes x does not belong to a positive region, clearly x will not be part of a positive region. Using the definition of positive region, the FRDD function [19] is defined as:

$$\gamma'_P(Q) = \frac{|\mu_{POS_P(Q)}(x)|}{|\mathbb{U}|} = \frac{\sum_{x\in\mathbb{U}} \mu_{POS_P(Q)}(x)}{|\mathbb{U}|}. \tag{7}$$

Based on the concept of the FRDD, we have developed a new metaheuristic search mechanism in order to effectively discover the minimal reducts. Among various search algorithms, such as GA and PSO, the SFLA can be used as a promising search method for feature selection (which is an NP-hard problem), due to its performance toward global optimal solution, both from a likelihood and a speed perspective [33]. Based on the published results in [33], the GA has failed to find best values in 20% of the cases, and it also needs a higher number of function evaluations to find the optimal value, compared to the SFLA. The SFLA is capable of finding a subset of solutions along with the optimal answer as the final result. Since the feature selection problem is fundamentally binary, the need for a binary search algorithm is inevitable.

B. Search Method

The search process starts by randomly initializing each binary individual with the size of the number of features, and continues by participating in ranking, partitioning and evolutionary processes. Generally, the SFLA consists of seven steps as follows:

Step 1 ***Initialize the population:*** Choose m and n. Here, m is the number of memeplexes, and n is the number of frogs in each memeplex. The total number of frogs is then $F = m \times n$.

Step 2 ***Generate a population:*** The total number of frogs in the feasible space is $\Omega \subset \Re^d$ where d is the number of decision variables (features); the *i*th frog is encoded as $U(i) = (U_i^1, U_i^2, ..., U_i^d)$. Compute the fitness value for all individuals using (7).

Step 3 ***Rank frogs:*** Sort frogs in descending order of their fitnesses, and record them in $X = \{U(i), f(i), i = 1, ..., F\}$. The position of the first (i.e., best) frog is recorded in P_X, where $P_X = U(1)$.

Step 4 ***Partition frogs into memeplexes:*** Partition the array X of frogs into m memeplexes, each containing n frogs.

$$\begin{aligned} Y^k =& [U(j)^k, f(j)^k | U(j)^k = U(k + m(j-1)), \\ f(j)^k =& f(k + m(j-1)), j = 1, ..., n], k = 1, ..., m \end{aligned} \quad (8)$$

Step 5 ***Memetic evolution in each memeplex:*** Each memeplex is involved in the evolution which is described later in the Step 5's subsection.

Step 6 ***Shuffle memeplexes:*** After a predefined number of evolution rounds, all memeplexes are mixed into X, and sorted in descending order.

Step 7 ***Check convergence:*** If the convergence criteria are satisfied, stop. Otherwise, go to Step 4.

Note that in the Step 5, the evolution process is repeated N times. This process is comprised of further steps, as follows:

Step 1 ***Initialization:*** Set $i_m = 0$ and $i_N = 0$ as two counters for memeplexes and evolutions, respectively.

Step 2 $i_m = i_m + 1$

Step 3 $i_N = i_N + 1$

Step 4 ***Construct a submemeplex:*** In order to avoid being trapped in local optima, a subset of memeplexes is selected for moving toward. The submemeplex selection strategy is based on a triangular probability distribution (see (9)) that assigns the highest value to a frog with the maximum fitness and the lowest value to a frog with the minimum fitness. This assignment increases the chances of a high performing frog being selected.

$$p_j = \frac{2 \times (n + 1 - j)}{n \times (n + 1)}, j = 1, ..., n \quad (9)$$

For example, for $j = 1$ and $j = n$, the probabilities are given by:

$$p_1 = \frac{2}{n+1}, p_n = \frac{2}{n \times (n+1)}$$

After the submemeplex formation, it is sorted in descending order in an array, Z, and the best and the worst positions are recorded in P_B and P_W, respectively.

Step 5 ***Improve the worst frog:*** The worst frog's position is improved using (10) and (11) for positive and negative steps, respectively.

$$\text{step size } S = min\{int\{rand \times (P_B - P_W)\}, S_{max}\} \quad (10)$$

$$\text{step size } S = max\{int\{rand \times (P_B - P_W)\}, -S_{max}\}, \quad (11)$$

where *rand* is a random number, *int* is the integer part of a number, and S_{max} is the maximum step size allowed to be adopted after infection. Since the P_B and P_W are in binary form, the distance between two parameters is calculated using the *HD*; therefore, (10) and (11) are modified to (12) and (13) to deal with binary parameters.

$$\text{step size } S = min\{int\{rand \times HD(P_B, P_W)\}, S_{max}\} \quad (12)$$

$$\text{step size } S = max\{int\{rand \times HD(P_B, P_W)\}, - S_{max}\}. \quad (13)$$

Then, the new position is calculated by:

$$U_{(q)} = P_W + S, \quad (14)$$

where q is the number of randomly selected frogs from n frogs to form a memeplex and it is initialized manually. If $U_{(q)}$ is in feasible space Ω, then compute the fitness value, $f_{(q)}$; otherwise, go to the Step 5.6. If the newly computed $f_{(q)}$ is better than the old $f_{(q)}$, then go to the Step 5.8; otherwise, go to the Step 5.6.

Step 6 ***Compute new position:*** For real-valued frogs new position can be calculated using (15) and (16), whereas for the binary-valued frogs (17) and (18) can be used.

$$\text{step size } S = min\{int\{rand \times (P_X - P_W)\}, S_{max}\} \quad (15)$$

$$\text{step size } S = max\{int\{rand \times (P_X - P_W)\}, - S_{max}\} \quad (16)$$

$$\text{step size } S = min\{int\{rand \times HD(P_X, P_W)\}, S_{max}\} \quad (17)$$

$$\text{step size } S = max\{int\{rand \times HD(P_X, P_W)\}, - S_{max}\}. \quad (18)$$

If $U_{(q)}$ is in feasible space Ω, then compute the fitness value, $f_{(q)}$; otherwise, go to Step 5.7. If the newly computed $f_{(q)}$ is better than the old $f_{(q)}$, then go to Step 5.8; otherwise, go to Step 5.7.

Step 7 ***Censorship:*** Replace this frog with a randomly generated frog, r.

Step 8 ***Update the memeplex:*** After changing the worst frog's position in the submemeplex, replace Z in their original locations in Y^{i_m} . Sort Y^{i_m} in descending order.

Step 9 If $i_N < N$, go to Step 5.3.

Step 10 If $i_m < m$, go to Step 5.2.

Meanwhile, a modification for calculating the distance of the frogs is further applied to the proposed binary SFLA. The distance of the frogs that was calculated using the *HD* is replaced with a dissimilarity measure based on the fuzzy-rough set. The positive region i.e., $POS(.)$ [38] as presented in (6) is used instead of the *HD*. The positive region sees the frogs as features and calculates the similarity between each frog and the best frog. The value of $POS(.)$ varies from zero to the number of the variables. Since this distance must be dissimilarity, this measure is subtracted from the length of the binary frog. This measure can be employed in the Step 5, and the modified equations are given by (19) and (20) are used in the Step 5.6.

$$\text{step size } S = min\{int\{rand \times (L - POS(P_B, P_W))\}, S_{max}\} \quad (19)$$

$$\text{step size } S = max\{int\{rand \times (L - POS(P_B, P_W))\}, - S_{max}\}, \quad (20)$$

where L is the length of a binary frog, and S_{max} is the maximum step size allowed to be adopted after evolution.

The hybridization of the B-SFLA with FRDD is suggested to discover more than one reduct with the highest dependency degree. The L-FRFS can be considered as a multi-modal problem, in which the smallest subset of features with the highest FRDD is desired. Thus, conventional evolutionary algorithms might find many global optima with the highest FRDD; however, a question arising here is "which one is the best?"; Referring to the fitness, all of these solutions are acceptable, whereas referring to the cardinality of the subsets they varies. By ranking the subsets with the same FRDD, based on the number of selected features, a new wide range of reducted subsets is provided. This range can be analyzed using the frequency of a feature's appearance in all of the reducted subsets. The most frequent features might play an important role in specifying the outcome.

The aforementioned strategy is placed in the Step 5.4 of meme evolution and the Step 3, ranking frogs, of the B-SFLA; however, the ranking process is primarily based on the FRDD and in the case of having several subsets with the identical FRDD, it ranks subsets based on their cardinality. Through this process, the B-SFLA returns more than one reduct in a single run; conventional search methods do not always return more than one reduct. These minimal sets satisfy both criteria: the highest FRDD and the lowest number of selected features.

Using this method, the frogs leap toward two goals simultaneously. In the very first leaps, frogs jump toward the subsets with the highest FRDD; therefore, they try to increase their fitness as much as possible. In the following leaps, when the number of frogs with the maximum fitness is increased, the population selects the individuals with both the highest FRDD and the lowest number of features. Algorithm 1 shows pseudo code of the proposed method. The C++ implementation of the proposed method is publicly available on GitHub. [1]

Algorithm 1 FRFS based on B-SFLA

```
procedure SEARCH–EVALUATE
    initialize $m, n, q, N, S_{max}$
    generate a population of $(m \times n)$ frogs
    rank frogs in $X$ based on # of features and FRDD
    partition $X$ into $m$ memeplexes $Y^1, Y^2, ..., Y^m$
    while $i_m < m$ do
        while $i_N < N$ do
            construct submemeplex $Z$ containing $q$ frogs
            improve the worst frog and update FRDD
            replace infeasible and halting frogs
            partition $Z$ into $Y^1, Y^2, ..., Y^m$
        end
    end
    combine $Y^1, Y^2, ..., Y^m$ into $X$, update the best frog
    check the convergence criteria
end
```

In the preparation section, parameters of the B-SFLA are initialized based on the properties of the current dataset. Then, $m \times n$ diverse subsets of features are evaluated and evolved based on FRDD and B-SFLA, respectively. Then, the outcome of the algorithm is fed to nine different classifiers to avoid any tendency toward specific classification method. Finally, the mean of the resulting classification accuracies is calculated.

Since the complexity of meta-heuristic search algorithms are very depended on their parameters, it is worth mentioning that the complexity of the FRDD is $O(n^2)$ in the worst case [39], where n is number of features.

IV. Experimental Results and Discussion

Twenty two datasets from the UCI repository of machine learning [40] including nine large datasets – namely, LSVT Voice Rehabilitation [41], Urban Land Cover [42], [43], Arrhythmia, Molecular Biology, COIL 2000 [44], CNAE-9, Madelon [45], MicroMass, and Arcene [45] – have been selected and used to perform a comparative study. These datasets and their characteristics are shown in Table I. The table is sorted based on the number of samples × features.

The *fitness function* for all of the search algorithms is the FRDD depicted in (7). The GA and PSO parameters are presented in Tables II and III, respectively. For both algorithms, the population size and the number of generations are identical to B-SFLA's to enable further comparisons. As presented in [33], the SFLA parameter selection should be performed based on the properties of the problem. Parameter selection is one of the most important aspects of using search algorithms; however, it is still untouched for feature selection. Referring to the authors' recommendation in [33], for problems with 15-20 variables, the ranges in Table IV

[1] https://github.com/jracp/FuzzyRoughShuffledFrog

TABLE I
DATASET CHARACTERISTICS

Datasets	Samples	Features
Breast Tissue	106	10
Lung Cancer	32	56
Glass	214	10
Wine	178	13
Olitos	120	25
Heart	270	13
Cleveland	303	13
Parkinson	197	23
Pima Indian Diabetes	768	8
Breast Cancer Wisconsin	699	10
Ionosphere	351	33
Sonar	208	60
Libras Movement	360	90
LSVT Voice Rehab.	126	310
Urban Land Cover	675	148
Arrhythmia	452	279
Molecular Biology	3190	60
COIL 2000	5822	85
CNAE-9	1080	857
Madelon	2000	500
MicroMass	931	1300
Arcene	200	10000

are suggested. However, the parameter selection for feature selection has been formulated based on the total number of all features (all_F) using a trial and error method. The results are shown in Table V. Further investigations show that the proposed parameters in Table V work remarkably well for small datasets with less than 15,000 data cells; however, parameters in Table VI [33] can be used not only for small and medium datasets, but also for large ones.

TABLE II
GA PARAMETERS

Population	Generation	P_c	P_m
900	5	0.600	0.033

TABLE III
PSO PARAMETERS

Particles	Iteration	C_1	C_2
900	5	2	2

TABLE IV
SFLA PARAMETERS

m	n	N	q	S_{max}
$100 \leq m \leq 150$	$30 \leq n \leq 100$	$20 \leq N \leq 30$	20	$1.00 \times all\_F$

TABLE V
PROPOSED B-SFLA PARAMETERS FOR DATASETS WITH SIZE OF DATA CELLS $\leq 15,000$

m	n	N	q	S_{max}
$2.20 \times all\_F$	$0.70 \times all\_F$	$0.50 \times all\_F$	$0.45 \times all\_F$	$0.50 \times all\_F$

TABLE VI
PROPOSED B-SFLA PARAMETERS FOR MOST DATASETS

m	n	N	q	S_{max}
30	30	5	15	$0.45 \times all\_F$

The number of selected features obtained by each search algorithm is shown in Table VII. In terms of the number of selected features, the GBFS has selected the least number of features compared to the other methods; however, selecting one feature as a final result for Breast Tissue, Lung Cancer, Glass, Wine, and Sonar is not desirable both from an in-field and a data processing point of view. Selecting a very small number of features reduces the utility of feature selection methods for pre-processing and model complexity improvement.

TABLE VII
NUMBER OF SELECTED FEATURES OBTAINED BY EACH SEARCH ALGORITHM

Datasets	L-FRFS	GA	PSO	GBFS	B-SFLA
Breast Tissue	9	9	9	1	4
Lung Cancer	6	7	4	1	3
Glass	9	8	8	1	4
Wine	5	5	5	1	3
Olitos	5	5	5	6	5
Heart	7	8	7	4	5
Cleveland	11	10	10	4	7
Parkinson	5	6	6	3	4
Pima Indian Diabetes	8	8	8	2	6
Breast Cancer Wisconsin	7	7	7	6	7
Ionosphere	7	8	7	5	5
Sonar	5	6	6	1	5
Libras Movement	2	11	8	17	6
LSVT Voice Rehab.	5	11	7	6	7
Urban Land Cover	7	9	8	12	7
Arrhythmia	7	10	13	26	8
Molecular Biology	-	13	12	3	9
COIL 2000	29	46	33	5	8
CNAE-9	90	459	547	13	281
Madelon	-	-	-	6	7
MicroMass	33	168	142	24	141
Arcene	6	-	-	6	11

Nine classifiers – namely PART, JRip, Naive Bayes, Bayes Net, J48, BFTree, FT, NBTree and RBFNetwork – have been chosen from different classifiers categories to classify instances of each dataset after the feature selection process. These classifiers have been implemented in Weka, a machine learning package that is ready to use [46]. For all classifiers and the feature selection methods, 10-fold cross validation (10CV) has been conducted to calculate their performance. The mean as well as standard deviation (STD), and the best value of the nine classifiers' results over each dataset are presented in Table VIII. The best of the mean classification accuracies are boldfaced and superscripted. The last row shows the mean of the classification accuracies' mean, the STD, and the best in which the B-SFLA gains 1.22%, 2.16%, 2.33%, 7.87% higher mean classification accuracies compared to L-FRFS, GA, PSO, and GBFS, respectively. The B-SFLA outperforms other methods not only by decreasing the model size, but also by improving classification accuracy of the resulting models. Referring to the number of selected features in Table VII and the classification accuracies in Table VIII, the GBFS has selected the least number of features and obtained the smallest classification accuracy, which is worse when compared to the unreduced datasets and to the other methods.

Table IX shows the number of wins in terms of the best resulting classification accuracies. The L-FRFS has achieved the best accuracies for Breast Tissue, Glass, Wine, Ionosphere, Urban Land Cover, and CNAE-9. The GA has obtained the best classification accuracies in three cases, Breast Tissue,

TABLE VIII
MEAN, STANDARD DEVIATION, AND BEST OF CLASSIFICATION ACCURACIES (%)

Datasets	L-FRFS	Best	GA	Best	PSO	Best	GBFS	Best	B-SFLA	Best
Breast Tissue	**66.46 ± 3.69***	70.75	**66.46 ± 3.69***	70.75	66.46 ± 3.69	70.75	56.92 ± 4.42	61.32	65.09 ± 5.70	75.47
Lung Cancer	58.85 ± 12.48	77.78	41.56 ± 5.48	48.15	53.24 ± 11.53	70.37	37.04 ± 0.00	37.04	**62.96 ± 12.28***	77.78
Glass	**67.29 ± 7.62***	74.77	64.75 ± 7.76	71.96	64.75 ± 7.76	71.96	50.05 ± 5.50	54.67	65.32 ± 6.50	71.03
Wine	**95.63 ± 2.92***	99.44	92.38 ± 2.23	95.51	92.38 ± 2.23	95.51	66.67 ± 1.61	68.54	93.57 ± 1.97	96.07
Olitos	66.39 ± 5.50	73.33	63.89 ± 3.17	68.33	65.09 ± 3.29	70.00	**70.93 ± 4.24***	75.83	69.17 ± 4.06	77.50
Heart	78.48 ± 1.88	80.37	78.72 ± 1.55	80.74	**79.55 ± 3.77***	84.07	75.93 ± 2.10	78.89	78.85 ± 1.94	81.85
Cleveland	49.76 ± 5.58	54.88	50.73 ± 4.87	54.88	50.73 ± 4.87	54.88	**52.64 ± 2.84***	54.88	50.88 ± 4.11	54.88
Parkinson	85.07 ± 4.18	90.77	85.19 ± 3.20	90.26	83.36 ± 3.75	89.23	85.75 ± 3.31	90.26	**86.50 ± 3.61***	89.74
Pima Indian Diabetes	75.00 ± 1.23	77.34	75.00 ± 1.23	77.34	75.00 ± 1.23	77.34	64.76 ± 0.95	66.15	**75.35 ± 1.28***	76.69
Breast Cancer Wisconsin	96.23 ± 1.04	97.51	**96.40 ± 0.54***	97.36	96.13 ± 0.60	96.93	95.15 ± 0.85	96.05	96.03 ± 0.92	97.36
Ionosphere	**91.39 ± 1.04***	93.16	89.78 ± 1.22	92.02	89.49 ± 2.54	94.02	89.21 ± 1.40	91.74	89.65 ± 1.43	91.74
Sonar	69.82 ± 2.60	72.60	69.76 ± 2.29	73.08	64.26 ± 2.54	68.75	55.29 ± 3.69	61.06	**74.09 ± 3.45***	78.85
Libras Movement	21.76 ± 7.45	28.61	58.14 ± 10.11	73.94	57.73 ± 7.68	67.99	**61.36 ± 9.73***	74.17	53.43 ± 8.00	65.56
LSVT Voice Rehab.	79.45 ± 4.39	86.51	67.99 ± 8.10	76.98	74.52 ± 4.85	84.13	74.69 ± 10.17	80.95	**79.62 ± 5.66***	85.71
Urban Land Cover	**80.07 ± 2.68***	84.89	63.18 ± 2.87	74.37	56.50 ± 1.80	71.26	51.84 ± 1.73	83.70	77.66 ± 2.29	81.04
Arrhythmia	53.74 ± 3.10	57.52	53.60 ± 3.69	57.74	52.21 ± 4.52	56.42	**69.05 ± 2.59***	74.34	60.50 ± 4.11	64.60
Molecular Biology	-	-	63.18 ± 1.66	65.27	56.50 ± 1.45	59.00	51.84 ± 0.17	52.19	**80.12 ± 1.20***	81.38
COIL 2000	92.79 ± 2.01	94.02	92.42 ± 2.56	94.02	92.51 ± 2.40	94.02	93.97 ± 0.07	94.04	**93.98 ± 0.06***	94.02
CNAE-9	**88.78 ± 1.94***	91.57	85.77 ± 2.71	90.65	88.04 ± 3.46	92.59	53.60 ± 4.37	55.74	74.47 ± 2.32	77.96
Madelon	-	-	-	-	-	-	49.58 ± 0.72	50.80	**54.66 ± 0.68***	55.40
MicroMass	57.40 ± 5.16	66.90	**68.42 ± 5.44***	80.04	65.27 ± 4.10	74.78	63.07 ± 3.27	67.08	64.93 ± 4.02	73.20
Arcene	71.56 ± 3.00	77.00	-	-	-	-	**74.94 ± 4.45***	81.00	70.78 ± 5.37	78.50
Mean	72.30 ± 3.97	77.49	71.36 ± 3.72	76.67	71.19 ± 3.90	77.20	65.65 ± 3.10	70.47	**73.52 ± 3.70***	**78.47***

Breast Cancer Wisconsin and MicroMass. The PSO has obtained the highest classification accuracy for Heart dataset. The GBFS has achieved the best classification accuracies for five datasets – namely, Olitos, Cleveland, Libras Movement, Arrhythmia, and Arcene. Finally, B-SFLA has reached to the maximum number of wins for eight datasets – namely, Lung Cancer, Parkinson, Pima Indian Diabetes, Sonar, LSVT Voice Rehab., Molecular Biology, COIL 2000, and Madelon.

TABLE IX
NUMBER OF WINS FOR EACH METHOD IN GAINING HIGHEST CLASSIFICATION ACCURACY

Algorithm	L-FRFS	GA	PSO	GBFS	B-SFLA
Wins	6	3	1	5	**8**

It is concluded that the B-SFLA is the most suitable search algorithm for FS based on the fuzzy-rough sets approach in terms of the resulting classification accuracy. Note that the B-SFLA divides the population into subpopulations, and thereby the diversity in the population is preserved. Such a swarm algorithm is very suitable for multi-modal optimization problems that have several optima instead of just one global optimum [47]. The feature selection based on fuzzy-rough set is an example of such problems. The main intention in the L-FRFS is to obtain the minimal reducts; there exist several minimal-reducts for a given information system that are feature subsets with the minimal possible size and maximal possible FRDD. In a single run, GA and PSO generally produce one minimal reduct for a given problem as the final solution of the L-FRFS. However, the B-SFLA returns almost all of the minimal reducts in a single run in its final population. On the other hand, the B-SFLA apparently demonstrates its suitability for solving multi-modal problems since it inherently divides the population of frogs into different subpopulations. Therefore, each of these subpopulations is able to explore and exploit one of the several existing optima in the search space. This property of the B-SFLA makes it different from the other algorithms such as GA and PSO.

V. CONCLUSION AND FUTURE WORK

In this paper, a new version of the B-SFLA has been combined with the FRDD. Additionally, the performances of L-FRFS, two well-known evolutionary algorithms, the GBFS and the B-SFLA have been compared. By considering the results, the B-SFLA approach significantly outperforms the PSO, GA, and GBFS methods, and is slightly better than L-FRFS in terms of resulting classification accuracy. Feature selection via fuzzy-rough theory is a multi-modal problem, i.e. there are some feature subsets with the same size and FRDD. In this sense, the B-SFLA is a suitable search algorithm for such problems, since it divides the population into subpopulations (called memeplexes), and by preserving the diversity, it returns multiple minimal reducts rather than returning just a single one. This means that several minimal reducts (i.e. the feature subsets with the minimum cardinality and maximum FRDDs) have been produced in a single run. This characteristic is an additional advantage of the B-SFLA over the PSO and GA algorithms. We are planning to apply our proposed method on local datasets, such as existing health data from Newfoundland and Labrador Centre for Health Information (NLCHI), and global ones in Canada, such as data from Statistics Canada. Also, we are aiming to improve time and space complexity of the B-SFLA to target big data, and perform comprehensive examinations and comparisons with the newly introduced feature selection methods.

REFERENCES

[1] X. Zhao, D. Li, B. Yang, C. Ma, Y. Zhu, and H. Chen, "Feature selection based on improved ant colony optimization for online detection of foreign fiber in cotton," *Applied Soft Computing*, vol. 24, pp. 585 – 596, 2014.

[2] E. Hancer, B. Xue, D. Karaboga, and M. Zhang, "A binary {ABC} algorithm based on advanced similarity scheme for feature selection," *Applied Soft Computing*, vol. 36, pp. 334 – 348, 2015.

[3] N. Sreeja and A. Sankar, "Pattern matching based classification using ant colony optimization based feature selection," *Applied Soft Computing*, vol. 31, pp. 91 – 102, 2015.

[4] S. Saha, R. Spandana, A. Ekbal, and S. Bandyopadhyay, "Simultaneous feature selection and symmetry based clustering using multiobjective framework," *Applied Soft Computing*, vol. 29, pp. 479 – 486, 2015.

[5] X. Han, "Implicit feature selection for omics data phenotype discrimination," *Applied Soft Computing*, vol. 20, pp. 70 – 82, 2014, hybrid intelligent methods for health technologies.

[6] S.-W. Lin, K.-C. Ying, C.-Y. Lee, and Z.-J. Lee, "An intelligent algorithm with feature selection and decision rules applied to anomaly intrusion detection," *Applied Soft Computing*, vol. 12, no. 10, pp. 3285 – 3290, 2012.

[7] A. M. Canuto, K. M. Vale, A. Feitos, and A. Signoretti, "Reinsel: A class-based mechanism for feature selection in ensemble of classifiers," *Applied Soft Computing*, vol. 12, no. 8, pp. 2517 – 2529, 2012.

[8] K. Manimala, K. Selvi, and R. Ahila, "Hybrid soft computing techniques for feature selection and parameter optimization in power quality data mining," *Applied Soft Computing*, vol. 11, no. 8, pp. 5485 – 5497, 2011.

[9] R. Nock and M. Sebban, "Sharper bounds for the hardness of prototype and feature selection," in *Algorithmic Learning Theory*, ser. Lecture Notes in Computer Science, H. Arimura, S. Jain, and A. Sharma, Eds. Springer Berlin Heidelberg, 2000, vol. 1968, pp. 224–238.

[10] S. C. Yusta, "Different metaheuristic strategies to solve the feature selection problem," *Pattern Recognition Letters*, vol. 30, no. 5, pp. 525 – 534, 2009.

[11] P. Pudil, J. Novoviov, and P. Somol, "Feature selection toolbox software package," *Pattern Recognition Letters*, vol. 23, no. 4, pp. 487 – 492, 2002.

[12] M. ElAlami, "A filter model for feature subset selection based on genetic algorithm," *Knowledge-Based Systems*, vol. 22, no. 5, pp. 356 – 362, 2009.

[13] S. Nemati, M. E. Basiri, N. Ghasem-Aghaee, and M. H. Aghdam, "A novel acoga hybrid algorithm for feature selection in protein function prediction," *Expert Systems with Applications*, vol. 36, no. 10, pp. 12 086 – 12 094, 2009.

[14] S. M. Vieira, J. M. Sousa, and T. A. Runkler, "Two cooperative ant colonies for feature selection using fuzzy models," *Expert Systems with Applications*, vol. 37, no. 4, pp. 2714 – 2723, 2010.

[15] M. Sebban and R. Nock, "A hybrid filter/wrapper approach of feature selection using information theory," *Pattern Recognition*, vol. 35, no. 4, pp. 835 – 846, 2002.

[16] K. Thangavel and A. Pethalakshmi, "Dimensionality reduction based on rough set theory: A review," *Applied Soft Computing*, vol. 9, no. 1, pp. 1 – 12, 2009.

[17] A. Verikas, M. Bacauskiene, D. Valincius, and A. Gelzinis, "Predictor output sensitivity and feature similarity-based feature selection," *Fuzzy Sets and Systems*, vol. 159, no. 4, pp. 422 – 434, 2008.

[18] C. Degang and Z. Suyun, "Local reduction of decision system with fuzzy rough sets," *Fuzzy Sets and Systems*, vol. 161, no. 13, pp. 1871 – 1883, 2010.

[19] R. Jensen and Q. Shen, "New approaches to fuzzy-rough feature selection," *Fuzzy Systems, IEEE Transactions on*, vol. 17, no. 4, pp. 824–838, Aug 2009.

[20] Y. Chen, D. Miao, and R. Wang, "A rough set approach to feature selection based on ant colony optimization," *Pattern Recognition Letters*, vol. 31, no. 3, pp. 226 – 233, 2010.

[21] N. Suguna and K. Thanushkodi, "A novel rough set reduct algorithm for medical domain based on bee colony optimization," *Journal of Computing*, vol. 2, no. 6, pp. 49–54, June 2010.

[22] X. Wang, J. Yang, X. Teng, W. Xia, and R. Jensen, "Feature selection based on rough sets and particle swarm optimization," *Pattern Recognition Letters*, vol. 28, no. 4, pp. 459 – 471, 2007.

[23] J. Wróblewski, "Finding minimal reducts using genetic algorithms," in *Proccedings of the second annual join conference on infromation science*, 1995, pp. 186–189.

[24] J. R. Anaraki and M. Eftekhari, "Rough set based feature selection: A review," in *Information and Knowledge Technology (IKT), 2013 5th Conference on*, May 2013, pp. 301–306.

[25] R. Jensen and Q. Shen, "Fuzzy-rough data reduction with ant colony optimization," *Fuzzy Sets and Systems*, vol. 149, no. 1, pp. 5 – 20, 2005.

[26] J. Xiang, X. Han, F. Duan, Y. Qiang, X. Xiong, Y. Lan, and H. Chai, "A novel hybrid system for feature selection based on an improved gravitational search algorithm and k-nn method," *Applied Soft Computing*, vol. 31, pp. 293 – 307, 2015.

[27] S. M. Vieira, L. F. Mendona, G. J. Farinha, and J. M. Sousa, "Modified binary {PSO} for feature selection using {SVM} applied to mortality prediction of septic patients," *Applied Soft Computing*, vol. 13, no. 8, pp. 3494 – 3504, 2013.

[28] Z. Xu, G. Huang, K. Q. Weinberger, and A. X. Zheng, "Gradient boosted feature selection," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014, pp. 522–531.

[29] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.

[30] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and regression trees*. CRC press, 1984.

[31] Z. Pawlak, "Rough sets," *International Journal of Computer & Information Sciences*, vol. 11, no. 5, pp. 341–356, 1982.

[32] J. Komorowski, Z. Pawlak, L. Polkowski, and A. Skowron, "Rough sets: A tutorial," in *Rough-Fuzzy Hybridization: A New Trend in Decision Making*, S. K. Pal and A. Skowron, Eds. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1998, pp. 3–98.

[33] M. Eusuff, K. Lansey, and F. Pasha, "Shuffled frog-leaping algorithm: a memetic meta-heuristic for discrete optimization," *Engineering Optimization*, vol. 38, no. 2, pp. 129–154, 2006.

[34] Q. Duan, S. Sorooshian, and V. Gupta, "Effective and efficient global optimization for conceptual rainfall-runoff models," *Water Resources Research*, vol. 28, no. 4, pp. 1015–1031, 1992.

[35] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Neural Networks, 1995. Proceedings., IEEE International Conference on*, vol. 4, Nov 1995, pp. 1942–1948 vol.4.

[36] A. S. Reddy and K. Vaisakh, "Environmental constrained economic dispatch by modified shuffled frog leaping algorithm," *Journal of Bioinformatics and Intelligent Control*, vol. 2, no. 3, pp. 216–222, 2013.

[37] A. M. Radzikowska and E. E. Kerre, "A comparative study of fuzzy rough sets," *Fuzzy Sets and Systems*, vol. 126, no. 2, pp. 137 – 155, 2002.

[38] S. Kamyab, M. Eftekhari, and J. R. Anaraki, "A novel rough set based dissimilarity measure and its application in multimodal optimization," in *Artificial Intelligence and Signal Processing (AISP), 2012 16th CSI International Symposium on*, May 2012, pp. 180–185.

[39] R. Jensen and Q. Shen, *Computational intelligence and feature selection: rough and fuzzy approaches*. John Wiley & Sons, 2008, vol. 8.

[40] M. Lichman, "UCI machine learning repository," 2013. (Online). Available: http://archive.ics.uci.edu/ml.

[41] A. Tsanas, M. A. Little, C. Fox, and L. O. Ramig, "Objective automatic assessment of rehabilitative speech treatment in parkinson's disease," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 22, no. 1, pp. 181–190, 2014.

[42] B. A. Johnson, "High-resolution urban land-cover classification using a competitive multi-scale object-based approach," *Remote Sensing Letters*, vol. 4, no. 2, pp. 131–140, 2013.

[43] B. Johnson and Z. Xie, "Classifying a high resolution image of an urban area using super-object information," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 83, pp. 40–49, 2013.

[44] P. Van Der Putten and M. van Someren, "Coil challenge 2000: The insurance company case," *Published by Sentient Machine Research, Amsterdam. Also a Leiden Institute of Advanced Computer Science Technical Report*, vol. 9, pp. 1–43, 2000.

[45] I. Guyon, S. Gunn, A. Ben-Hur, and G. Dror, "Result analysis of the nips 2003 feature selection challenge," in *Advances in neural information processing systems*, 2004, pp. 545–552.

[46] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: An update," *SIGKDD Explor. Newsl.*, vol. 11, no. 1, pp. 10–18, Nov. 2009.

[47] K.-C. Wong, C.-H. Wu, R. K. Mok, C. Peng, and Z. Zhang, "Evolutionary multimodal optimization using the principle of locality," *Information Sciences*, vol. 194, pp. 138 – 170, 2012.