Fuzzy Population-Based Meta-Heuristic Approaches for Attribute Reduction in Rough Set Theory

Mafarja Majdi, Salwani Abdullah, Najmeh S. Jaddi

Abstract—One of the global combinatorial optimization problems in machine learning is feature selection. It concerned with removing the irrelevant, noisy, and redundant data, along with keeping the original meaning of the original data. Attribute reduction in rough set theory is an important feature selection method. Since attribute reduction is an NP-hard problem, it is necessary to investigate fast and effective approximate algorithms. In this paper, we proposed two feature selection mechanisms based on memetic algorithms (MAs) which combine the genetic algorithm with a fuzzy record to record travel algorithm and a fuzzy controlled great deluge algorithm, to identify a good balance between local search and genetic search. In order to verify the proposed approaches, numerical experiments are carried out on thirteen datasets. The results show that the MAs approaches are efficient in solving attribute reduction problems when compared with other meta-heuristic approaches.

Keywords—Rough Set Theory, Attribute Reduction, Fuzzy Logic, Memetic Algorithms, Record to Record Algorithm, Great Deluge Algorithm.

I. INTRODUCTION

AS the real word data is increasing very fast, the tools for use in the various knowledge fields (acquisition, storage, retrieval, maintenance, etc.) must develop in the same way to combat this growth. Knowledge can be defined as usable information and it is only valuable when it can be used efficiently and effectively [1]. With this increase of information amount, feature selection becomes a mandatory task of a learning process [2]. Feature selection studies how to select a minimal subset of features from a problem domain while retaining a suitably high accuracy in representing the original features. The optimal subset is determined by both relevancy and redundancy aspects. An attribute is said to be relevant if a decision is depending on it, otherwise it is irrelevant. Whilst, an attribute can be considered to be redundant if it is highly correlated with other attributes. Removing redundant and misleading attributes can improve the performance and efficiency of learning algorithms and to enhance the comprehensibility of the constructed models [3].

Over the past twenty years, rough set theory (RST) [4], [5] has attracted attention of many researchers over the world and has been applied to many domains [6]. Attribute reduction in rough set theory has been recognized as an important feature selection method since RST is a valid mathematical tool to

M. Mafarja is with the Department of Computer Science, Faculty of Information Technology, Birzeit University, Birzeit, Palestine.
S. Abdullah, N. S. Jaddi are with Data Mining and Optimisation Research Group (DMO), Center for Artificial Intelligence Technology, Universiti Kebangsaan Malaysia, 43600 UKM, Bangi Selangor, Malaysia (e-mail: salwani@ukm.edu.my).

handle imprecision, uncertainty and vagueness [6]-[8]. It's well known that attribute reduction is a NP-hard problem [9]. Examples of meta-heuristic approaches applied to attribute reduction are such as genetic algorithm in [10]-[12], ant colony [2], [10], simulated annealing algorithm [10], tabu search [13], scatter search [14], great deluge [15], [19], composite neighbourhood structure [16], hybrid variable neighbourhood search algorithm [17], and exponential monte-carlo [41]. Further reading about attribute reduction problems can be found in [20]-[25].

Memetic algorithms (MAs) which originally proposed by [26], are class of stochastic global search heuristic for global optimization in which a combination between the Evolutionary Algorithms-based (EA) approaches and local search techniques by using EA to perform exploration and local search to perform exploitation. Exploration is needed to ensure that every part in the search space searched enough to provide a reliable estimate of the global optimum. Exploitation is important since it concentrates the search effort around the best solutions found so far to derive better solutions by searching their neighbourhood structures. MAs for feature selection have been proposed recently in [27], [28]. In a MA, a population of possible solutions called chromosomes is generated. Two chromosomes are selected (based on their fitness value); GA operators (crossover and mutation) are applied on the selected chromosomes. The local search is applied on each chromosome in order to find better solutions in their neighboring region. At this point, a new population is evolved. The process is repeated until a stopping criterion has been achieved. On the other hand, the local search that has been used to improve the EAs is modified by employing a fuzzy logic controller in order to control the parameters within each algorithm intelligently.

In this paper, two fuzzy memetic-based algorithms, called GA- Fuzzy-RRT and GA- Fuzzy-GDA, are introduced. The main aim of this research is to investigate the efficiency of hybridizing the GA with local search algorithms (RRT and GDA) in determining (near) optimal feature subsets or rough set reducts. The local search algorithms were enhanced by employing the fuzzy logic to intelligently control the parameter in each algorithm. Furthermore, in contrast to the available attribute reduction methods that only report the numbers of generated attributes, we evaluated the quality of the generated subsets of attributes in terms of the number of generated rules (descriptive patterns) and the classification accuracy.

The remainder of this paper is organized as follows; Section II provides a brief introduction on rough set theory. Section III

discusses the proposed Memetic Algorithm (MA) for the attribute reduction problem. Section IV presents the experimental setup. Experimental results and analysis are presented in Section V. Finally, concluding remarks on the effectiveness of the proposed techniques and potential future research aspects are discussed in Section VI.

II. PRELIMINARIES

Let I = (U, A) be an information system, where U is a non-empty set of finite objects called the universe of discourse; A is a non-empty set of attributes. With every attribute $a \in A$, a set of its values (V_a) is associated. For a subset of attributes P $\subseteq$ A there is a an associated equivalence relation IND (P), which is called an indiscernibility relation. The relation IND (P) can be defined as:

$$IND(P) = \{ (x, y) \in U^2 | \forall a \in P, a(x) = a(y) \} \qquad (1)$$

If $(x, y) \in IND(P)$, then x and y are indiscernible by attributes from P. The equivalence classes of the P-indiscernibility relation are denoted $[x]_P$. The indiscernibility relation is the mathematical basis of the rough set theory. In rough set theory, the lower and upper approximations are two basic operations. For a subset $X \subseteq U$. X can be approximated using only information contained within P by constructing the P-lower approximation donated as $\underline{P}X$, is the set of all elements of U, which can be certainly classified as elements of X based on the attribute set P. The P-upper approximation of X, denoted as $\overline{P}X$, which can be possibly classified as elements of X based on the attribute set P. these two definitions can be expressed as:

$$\underline{P}X = \{x \mid [x]_P \subseteq X\} \qquad (2)$$

$$\overline{P}X = \{x \mid [x]_P \cap X \neq \varphi\} \qquad (3)$$

Definition 1 (Dependency Degree): Let P, Q $\subseteq$ A, the dependency degree k is defined by:

$$k = \gamma p(D) = \frac{|POS_P(Q)|}{|U|} \qquad (4)$$

where |Y| is the cardinality of Y. $POS_P(Q)$ called positive region, is defined by:

$$POS_P(Q) = \bigcup_{X \in U/Q} \underline{P}X \qquad (5)$$

The positive region contains all objects of U that can be uniquely classified to blocks of the partition U/Q using the knowledge in attributes P.

For P, Q $\subset$ A, it is said that Q depends on P in a degree of k $(0 \leq k \leq 1)$ denoted P $\Rightarrow$k Q,

If k = 1, we say that Q depends totally on P, if k < 1, we claim that Q depends partially on P, and if k = 0, we say that Q does not depend on P.

One of the major applications of rough set theory is to find the minimal reducts by eliminating the redundant attributes from original sets, without any information loss [4], [5]. The reduction of attributes can be achieved by comparing the dependency degrees of the generated subsets so that the reduced set has the same dependency degree of the original set [10]. A reduct is formally defined as a subset R of minimal cardinality of the conditional attribute set C such that $\gamma_R(D) = \gamma_C(D)$ where D is a decision system.

Definition 2 (Reduct): Let R be a subset of C, then R is said to be a reduct if:

The intersection of all reduced subsets is called the core:

$$r_R(D) = r_C(D) \wedge \forall R' \subset R, r_{R'}(D) < r_C(D) \qquad (6)$$

$$Core(\Re) = \bigcap_{R \in \Re} R \qquad (7)$$

The core contains all those attributes that cannot be removed from the dataset without introducing more contradictions to the dataset. In the process of attribute reduction, a set $\Re_{min} \subseteq \Re$ of reducts with minimum cardinality is searched for:

$$\Re_{min} = \{R \in \Re : |R| \leq |S|, \forall S \in \Re \} \qquad (8)$$

It is obvious that finding all possible reducts is a time consuming process, and moreover it is applicable only with small datasets. It is meaningless to calculate all reducts aiming to find only one minimal. To improve the performance of the above method an alternative strategy is required for large datasets.

III. ATTRIBUTE REDUCTION WITH MEMETIC ALGORITHM

MAs are population-based meta-heuristics search methods. In MAs, a population of solutions for the problem at hand is generated. Each of these solutions is called individual in the MA terminology. Recent studies on MAs have revealed their success on a wide variety of real world problems. Particularly, they not only converge to high quality solutions, but also search more efficiently than their conventional counterparts [26], [29]. Fig. 1 shows a pseudo code for a simple memetic algorithm adopted from [30].

MAs for feature selection have been proposed in [27], [28] and have shown superior performance over GAs and other search methods. Nonetheless, due to the inefficient nature of sequential LS [28] or random bit climbing LS [27] used in these existing methods, a large amount of redundant computation is incurred on evaluating the fitness of feature subsets. This makes them less attractive, particularly on problems with large feature size. In this work, we propose novel hybrid feature selection algorithms using memetic framework.

```
Memetic Algorithm Template
Begin
    INITIALIZE population;
    EVALUATE each candidate;
    Repeat Until (TERMINIATION CONDITION) Do
        SELECT parents;
        RECOMBINE to produce offspring;
        MUTATE offspring;
        IMPROVE offspring via Local Search;
        EVALUATE offspring;
        SELECT individuals for next generation;
    endDo
End
```

Fig. 1 Pseudo code for a simple memetic algorithm

A. Genetic Algorithms

Genetic Algorithms (GAs) have been developed by J. Holand in the 1970s to achieve the goal of understanding the adaptive process of the natural systems [31]. The traditional GA uses a population of solutions in solving a given problem; each solution is represented by a chromosome with a length of m where m is the number of attributes in the dataset [32]. Usually a binary representation is used to represent the solution. In the binary representation, the bit '1' implies a selected attribute while the bit '0' implies an excluded attribute.

In the general template of GA, firstly the initial population is randomly generated, where a random scheme is operated to decide the number of 1-bits in each chromosome and the places where those 1-bits will be located inside the chromosome, after the initialization step; the quality of the chromosome will be calculated by a fitness function. Then two parents will be selected to apply the GAs operators: crossover and mutation. The selection process can be either randomly or using a mechanism (e.g. Roulette Wheel Selection (RWS), Tournament Selection) [33]. The crossover operator will be applied on the selected parents in order to allow the search to look in diverse directions for attractive solutions to be combined in a single child to generate a new population. Then mutation operator will be applied to alter one or more components of the new child. This process (selection, crossover, and mutation) continues until the stopping criterion is satisfied.

B. Chromosome Representation

In this work, each candidate feature subset in the population is encoded as a binary string called chromosome of length m which is the number of condition attributes in the dataset. So that each bit encodes a single feature (as shown in Fig. 2). A bit of '1' implies the corresponding feature is selected while the '0' means that the feature is excluded. The maximum allowable number of 1's in each chromosome is m. At the start of the search, a population size of p is randomly initialized.

C. Fuzzy Local Search

Local search is a method of searching a small area around a solution and adopting a better solution if one can be found. The search begins with randomly selected solution. The

process is repeated for the new feature and the algorithm continues until a local optimum is found. In this research, two local search methods are hybridized with genetic algorithm (GA). A fuzzy based record to record to record travel algorithm (FRRT) and a fuzzy based great deluge algorithm (FGDA). In these two approaches, the fuzzy logic controller is employed to control the parameter of each algorithm. Fuzzy Logic has been widely used with many real world applications since being introduced by [34] in 1965. For example, [35] have proposed three new techniques for fuzzy rough set feature selection based on the use of fuzzy transitive similarity relations. Also in scheduling and timetabling applications, fuzzy evaluation functions have been utilised in a number of different applications. The fuzzy systems are generally consist of four components; an input fuzzifier, a knowledge base (rule base), an interfaces engine and defuzzification inference (Fig. 3). The rules have a main role of linking the input and output variables (in `IF -THEN' form), they are utilised to depict the response of the system relatively in terms of linguistic variables (words) than the mathematical formulae.
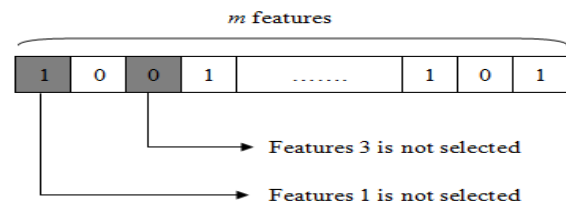

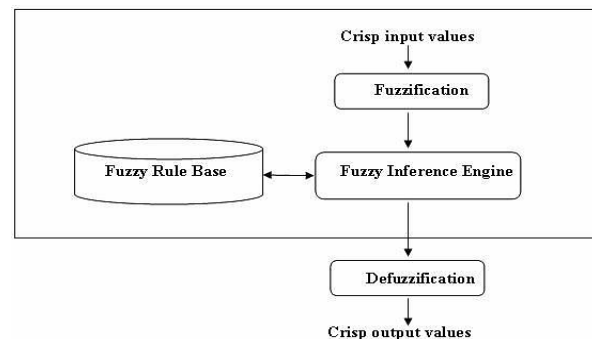Fig. 2 Chromosome Representation as a binary string

Fig. 3 Structure of a fuzzy Logic Model

The `IF' part of the rule is mentioned as the `antecedent' and the `THEN' part is mentioned as the `consequent'. The number of inputs and outputs and as well as the desired behaviour of the system have direct impact on the number of rules. After the rules are generated, the system can be seen as a non-linear mapping from inputs to outputs. More details about simple treatment can be found in [36] and complete treatment in [37].

In proposed approaches, the controller takes two inputs; the trial solution (Sol_{trial}) and the best solution (Sol_{best}) that is connected to general terms: low, medium and high (corresponding to fuzzy sets meanings). A rule set which linking the input variables (Sol_{trial} and Sol_{best}) with the single output variable is built. For each of these inputs and output, three symmetric and triangular-shaped membership functions are defined and evenly distributed on the appropriate universe

of discourse (Fig. 4). A membership function gives the degree of membership of an input value to every fuzzy set. The input
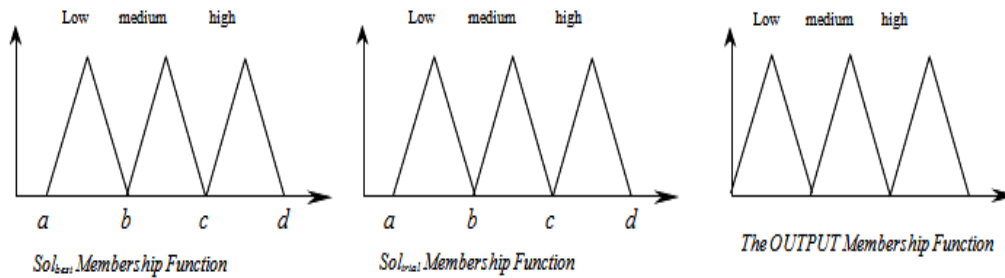
may belong to more than one fuzzy set.

Fig. 4 Membership Functions

Depending on the membership functions, the `fuzzifier' calculates the grade of membership in each input variable for every rule. For example, in Rule 1, the membership grade is calculated for the Sol_{best} in the fuzzy set low and for the Sol_{trial} in the medium fuzzy set. In dividing the intervals of the Sol_{best} and Sol_{trial} membership functions we depend the same criteria in MGDAR; where a three equaled intervals were established between the quality of the initial solution and the maximum dependency degree 1 (i.e. a represents the quality of the initial solution, d = 1.) Later with these fuzzified values, the inference engine calculates the antecedent's of Rule 1 by employing the suitable fuzzy operators consequent to the AND or OR connectives. Later the implication operator is applied by the inference engine to the rule, to get the fuzzy to be mounted over the output variable. Here the inference is implemented by shortening the output membership function at the level equivalent to the calculated level of truth in the rule's antecedent. Then the defuzzification step is then executed to translate the final fuzzy output into a crisp value. A common form of this process is called 'Center of Area' (COA), as it depends on the concept of identifying the area under a scaled membership function, as:

$$\int_{\min A}^{\max A} \mu(x_i)\cdot x_i \Big/ \int_{\min A}^{\max A} \mu(x_i) \qquad (9)$$

D.GA-Fuzzy-RRT

The genetic algorithm starts by generating population of solutions, which consists of 100 randomly generated solutions. The population is evaluated then (by calculating the dependency degree of each solution) and the best solution among the population is assigned to Sol_{best}. Then, two solutions (which called parents) are selected based on roulette wheel selection (RWS). After that, these two parents undergo through genetic operators (crossover and mutation) to generate two new offspring. Since this work proposes a memetic algorithm, the next step is applying a local search algorithm (RRT in this work) on the two generated offspring which will be used as initial solution for the SA process (individually).

Record-to-Record Travel algorithm (RRT) is a local search algorithm was originally proposed by [38]. It's a variant of simulated annealing (SA) algorithm. It is similar to SA in the structure and differs in the mechanism of accepting non-

improving solutions. It has the advantage that it depends only on one parameter which is the value of the RECORD-DEVIATION [38]. The algorithm begins with a randomly generated solution. Initially, the best solution, Sol_{best}, is set as Sol, and the Record is set as the fitness value of the best solution. In the while loop, a trial solution is generated randomly (Sol_{trial}). The best and the trial solutions are considered to be inputs to the fuzzy logic controller (FLC) in order to compute the DEVIATION value. If the Sol_{trial} is better than the best solution in hand (Sol_{best}), it will be accepted. Then, the Sol_{best} will be updated to be Sol_{trial}, and the record will be $f(Sol_{trial})$. If the Sol_{trial} and the Sol_{best} have the same qualities, then the solution with the smallest number of attributes is accepted, and Sol_{best} and the record will be updated. Otherwise, the difference between the two solutions is calculated, if it is smaller than the deviation parameter DEVIATION, then it will be accepted as a nonimproving solution. The algorithm stops when the termination criterion (the number of generations is) is satisfied.

E. GA-Fuzzy-GDA

In this work, we introduce a hybridisation between GA and an enhanced great deluge algorithm (GDA). An enhancement was made on the modified great deluge for attribute reduction (MGDAR) which was proposed by [19]. A fuzzy Logic controller is used to intelligently control the updating scheme (β) of the 'water level' based on the quality of the produced solutions. By using the intelligent controller, we can update the level by applying different β values through the search process instead of using one updating scheme or three static updating schemes (as proposed in MGDAR). The search space is divided into equaled three intervals; each one represents a fuzzy set (low, medium and high) in the fuzzy logic system.

In the proposed approach, the algorithm starts with applying the GA operators on the selected solutions, then the enhanced GDA algorithm is applied to enhance the selected solutions. In the enhanced GDA, a better solution is always been accepted. If the quality of the trial solution is equal to the quality of the best solution so far, then the solution with the lower cardinality is accepted, and the best solution is updated. A worse solution is accepted if the quality of the trial solution, $f(Sol_{trial})$ is higher than the *level*. Then the level is updated by automatically by the fuzzy logic controller. The process continues until the

termination criterion is met. In this work, the termination criterion is set as the number of iterations.

IV. EXPERIMENTAL SETUP

In this section, we present an experimental study of GA-Fuzzy-RRT and GA-Fuzzy-GDA on commonly used benchmark datasets. In particular we used 13 well-known UCI datasets from the small and medium sizes [10], [39] as shown in Table I. The proposed algorithm was programmed using Java and performed on Intel Pentium 4, 2.33 GHz computer.

In the proposed methods, we employed a population size of 30 and a number of generations of 100. In our experimental setup, we employ crossover and mutation probabilities of 0.6 and 0.4, respectively. Roulette wheel selection [33] is used for selection.

TABLE I
UCI DATA SETS

Datasets	No of Attributes	No. of Objects
M-of-N	13	1000
Exactly	13	1000
Exactly2	13	1000
Heart	13	294
Vote	16	300
Credit	20	1000
Mushroom	22	8124
LED	24	2000
Letters	25	26
Derm	34	366
Derm2	34	358
WQ	38	521
Lung	56	32

V. RESULTS AND ANALYSIS

A. Comparing GA-Fuzzy-RRT with GA-Fuzzy-GDA

This section includes a comparison between the proposed approaches to check the effectiveness of Hybridizing different local searches with GA on the number of minimal reducts and the classification accuracy and the number of generated rules.

1. Number of Minimal Attributes (Reducts)

The search performances of the proposed methods (GA-Fuzzy- RRT and GA-Fuzzy- GDA) on the 13 datasets considered are summarized in Table II. The superscripts in parentheses represent the number of runs that achieved the minimal reducts, while the number of attributes without superscripts means that the method could obtain only that number of attributes in all runs. Note that in these experiments, we run our experiments for 20 times for each dataset.

2. Classification Accuracy and Number of Rules

We have carried out further experiments to find the accuracy of the classification and generate the number of rules for all datasets based on the obtained reducts. Classification was performed using the Standard Voter algorithm found in the ROSETTA library [40]. The independent tests were performed with the Voting parameter set to Simple. All other

parameters were set to the default values. The data presented in Table III show the classification accuracy as a percentage.

TABLE II
RESULTS OBTAINED FROM GA-FUZZY-RRT AND GA-FUZZY-GDA

Datasets	Number of Attributes	GA-Fuzzy-RRT	GA-Fuzzy-GDA
M-of-N	13	6	6
Exactly	13	6	6
Exactly2	13	10	10
Heart	13	6	$6^{(18)}7^{(2)}$
Vote	16	8	8
Credit	20	$8^{(19)}9^{(1)}$	8
Mushroom	22	4	4
LED	24	$5^{(19)}6^{(1)}$	5
Letters	25	8	8
Derm	34	$6^{(17)}7^{(3)}$	$6^{(16)}7^{(4)}$
Derm2	34	$8^{(8)}9^{(12)}$	$8^{(8)}9^{(12)}$
WQ	38	13	$12^{(7)}13^{(13)}$
Lung	56	$4^{(14)}5^{(6)}$	$4^{(12)}5^{(8)}$

TABLE III
CLASSIFICATION ACCURACY AND NUMBER OF RULES GENERATED BY GA-FUZZY-RRT AND GA-FUZZY-GDA

Datasets	GA-Fuzzy-RRT		GA-Fuzzy-GDA	
	No. of Rules	Accuracy %	No. of Rules	Accuracy %
M-of-N	64	100	64	100
Exactly	64	100	64	100
Exactly2	606	57	606	57
Heart	261	0	514	14
Vote	134	63	134	63
Credit	1760	8	873	6
Mushroom	601	100	537	100
LED	30	100	10	100
Letters	459	0	436	0
Derm	5356	19	6358	24
Derm2	4311	78	5480	86
WQ	9215	30	9209	37
Lung	448	100	458	100

Even though there is a decrease in the classification accuracy for the Credit dataset, this decrease is insignificant at the same time GA-Fuzzy-GDA was able to outperform GA-Fuzzy-RRT in terms of number of attribute. Indeed, GA-Fuzzy-GDA may sometimes discover subsets of similar size to those selected by GA-Fuzzy-RRT but demonstrates an increase in the classification accuracy, this manly happen because the reducts may contain different attributes.

The results in Table III show that GA-Fuzzy-GDA demonstrates an increase in the classification accuracy performance in four out of thirteen datasets (i.e. Heart, Derm, Derm2 and WQ) when compared with GA-Fuzzy-RRT.Some of these increases are significant; like 14% in Heart dataset and 8% in Derm2 dataset. In terms of number of generated rules, from Table III, GA-Fuzzy-GDA outperforms GA-Fuzzy-RRT significantly in some datasets. For example, in Credit dataset, 873 rules were generated when the reducts produced by GA-Fuzzy-GDA were used, while 1760 rules were generated by using GA-Fuzzy-RRT reducts. In some datasets, the difference in the number of generated rules is notable but not significant, i.e. in Mushroom dataset 537 rules

were generated by using GA-Fuzzy-GDA reducts, while 601 were generated in the case of GA-Fuzzy-RRT.

TABLE IV
COMPARISON WITH THE STATE-OF-ART APPROACHES 1

Datasets	GA-Fuzzy-RRT	GA-Fuzzy-GDA	GD-RSAR	TSAR	SimRSAR	AntRSAR	ACOAR
M-of-N	6	6	$6^{(10)} 7^{(10)}$	6	6	6	6
Exactly	6	6	$6^{(7)} 7^{(10)}8^{(3)}$	6	6	6	6
Exactly2	10	10	$10^{(14)}11^{(6)}$	10	10	10	10
Heart	6	$6^{(18)}7^{(2)}$	$9^{(4)}10^{(16)}$	6	$6^{(29)} 7^{(1)}$	$6^{(18)} 7^{(2)}$	6
Vote	8	8	$9^{(17)}10^{(3)}$	8	$8^{(15)} 9^{(15)}$	8	8
Credit	$8^{(19)} 9^{(1)}$	8	$11^{(11)}12^{(9)}$	$8^{(13)} 9^{(5)} 10^{(2)}$	$8^{(18)} 9^{(1)} 11^{(1)}$	$8^{(12)} 9^{(4)} 10^{(4)}$	$8^{(16)}9^{(4)}$
Mushroom	4	4	$4^{(8)} 5^{(9)}6^{(3)}$	$4^{(17)} 5^{(3)}$	4	4	4
LED	$5^{(19)} 6^{(1)}$	5	$8^{(14)}9^{(6)}$	5	5	$5^{(12)} 6^{(4)} 7^{(3)}$	5
Letters	8	8	$8^{(7)}9^{(13)}$	$8^{(17)} 9^{(3)}$	8	8	8
Derm	$6^{(17)} 7^{(3)}$	$6^{(16)} 7^{(4)}$	$12^{(14)}13^{(6)}$	$6^{(14)} 7^{(6)}$	$6^{(12)} 7^{(8)}$	$6^{(17)} 7^{(3)}$	6
Derm2	$8^{(13)} 9^{(7)}$	$8^{(8)} 9^{(12)}$	$11^{(14)}12^{(6)}$	$8^{(2)} 9^{(14)} 10^{(4)}$	$8^{(3)} 9^{(7)}$	$8^{(3)} 9^{(17)}$	$8^{(4)}9^{(16)}$
WQ	13	$12^{(7)}13^{(13)}$	$15^{(14)}16^{(6)}$	$12^{(1)} 13^{(13)} 14^{(6)}$	$13^{(16)} 14^{(4)}$	$12^{(2)} 13^{(7)} 14^{(11)}$	$12^{(4)}13^{(12)}14^{(4)}$
Lung	$4^{(14)} 5^{(6)}$	$4^{(12)} 5^{(8)}$	$4^{(5)} 5^{(2)} 6^{(13)}$	$4^{(6)} 5^{(13)} 6^{(1)}$	$4^{(7)} 5^{(12)} 6^{(1)}$	4	4

TABLE V
COMPARISON WITH THE STATE-OF-ART APPROACHES 2

Datasets	GA-Fuzzy-RRT	GA-Fuzzy-GDA	EMC-FS	IS-CNS	HVNS-AR	GenRSAR	CHH_RSAR	SSAR
M-of-N	6	6	6	6	6	$6^{(6)}7^{(12)}$	$6^{(11)}7^{(9)}$	6
Exactly	6	6	6	6	6	$6^{(10)}7^{(10)}$	$6^{(13)}7^{(7)}$	6
Exactly2	10	10	10	10	10	$10^{(9)}11^{(11)}$	10	10
Heart	6	$6^{(18)}7^{(2)}$	$\mathbf{5^{(3)}}6^{(17)}$	6	6	$6^{(18)}7^{(2)}$	6	6
Vote	8	8	8	8	8	$8^{(2)}9^{(18)}$	8	8
Credit	$8^{(19)} 9^{(1)}$	8	8	$8^{(10)}9^{(9)} 10^{(1)}$	$8^{(7)}9^{(6)} 10^{(7)}$	$10^{(6)}11^{(14)}$	$8^{(10)}9^{(7)} 10^{(3)}$	$8^{(9)} 9^{(8)} 10^{(3)}$
Mushroom	4	4	4	4	4	$5^{(1)}6^{(5)}7^{(14)}$	4	$4^{(12)} 5^{(8)}$
LED	$5^{(19)} 6^{(1)}$	5	5	5	5	$6^{(1)}7^{(3)}8^{(16)}$	5	5
Letters	8	8	8	8	8	$8^{(8)}9^{(12)}$	8	$8^{(5)} 9^{(15)}$
Derm	$6^{(17)} 7^{(3)}$	$6^{(16)} 7^{(4)}$	6	$6^{(18)} 7^{(2)}$	$6^{(16)} 7^{(4)}$	$10^{(6)}11^{(14)}$	6	6
Derm2	$8^{(13)} 9^{(7)}$	$8^{(8)} 9^{(12)}$	$\mathbf{8^{(19)}}9^{(1)}$	$8^{(4)}9^{(16)}$	$8^{(5)}9^{(12)}10^{(3)}$	$10^{(4)}11^{(16)}$	$8^{(5)}9^{(5)}10^{(10)}$	$8^{(2)} 9^{(18)}$
WQ	13	$12^{(7)}13^{(13)}$	$\mathbf{12^{(17)}}14^{(3)}$	$12^{(2)}13^{(8)}14^{(10)}$	$12^{(3)}13^{(6)}14^{(8)} 15^{(3)}$	16	$12^{(13)}14^{(7)}$	$13^{(4)} 14^{(16)}$
Lung	$4^{(14)} 5^{(6)}$	$4^{(12)} 5^{(8)}$	4	$4^{(17)} 5^{(3)}$	$4^{(16)} 5^{(4)}$	$6^{(8)}7^{(12)}$	$4^{(10)} 5^{(7)} 6^{(3)}$	4

A. Comparison with the State of the Art

The results of our approaches and the results from the state-of-the art methods are reported in Tables IV and V. The entries in these tables represent the number of attributes in the minimal reducts obtained by each method. The superscripts in parentheses represent the number of runs that achieved the minimal reducts. The number of attribute without superscripts means that the method could obtain this number of attribute for all runs. The proposed approaches are compared with other rough set attribute reduction methods i.e. Tabu Search (TSAR) by [13], Ant Colony Optimization (AntRSAR) by [10], [39], Genetic Algorithm (GenRSAR) by [10], [39], Simulated Annealing (SimRSAR) by [10], Ant Colony Optimization (ACOAR) by [2], Scatter Search (SSAR) by [14], Great Deluge algorithm (GD-RSAR) by [15], Composite Neighbourhood Structure for Attribute Reduction (IS-CNS) by [16], Hybrid variable neighbourhood search algorithm (HVNS-AR) by [17], a Constructive Hyper-Heuristics (CHH_RSAR) by [18], and an Exponential Monte-Carlo algorithm (EMC-FS) by [41].

From the results, it can be seen that GA-Fuzzy-GDA is comparable with the other approaches since it performs better than some approaches in some datasets. It is better than AntRSAR on five datasets; and better than SSAR on five datasets (ties on four datasets). GA-Fuzzy-GDA outperforms TSAR in seven datasets. Our method outperforms IS-CNS, HVNS-AR, CHH_RSAR on 3, 3, and 5 instances, respectively. Here, we are interested to compare our approach with relevant local search approaches for attribute reduction (i.e., GD-RSAR and SimRSAR) and a population based approach (GenRSAR) in order to examine the performance of hybridizing a local search with GA. These two local search methods are selected because they have the same structure as GDA with a difference in accepting worst solutions.

The results showed that our approach is able to obtain better results on all datasets when compare with the GD-RSAR. GA-Fuzzy-GDA is also performed better than SimRSAR in six datasets, and ties on 7 datasets. On the other hand, GA-Fuzzy-GDA showed a good performance when compared with GenRSAR, and it was able to get better results in all datasets. As seen from the previous discussion, GA-Fuzzy-GDA shows promising performance when compared with other available methods, we believe that the strength of this method comes from the improvement of the search capability which was

added to GA by combining the local search. In addition, the Novel modification on the used local search, by employing the fuzzy logic controller to control the parameters, improved the performance of the proposed approach.

TABLE VI
COMPARISON WITH THE STATE-OF-ART APPROACHES IN TERMS OF ACCURACY AND NUMBER OF RULES

Datasets	GA-Fuzzy-GDA		GA		Johnson		Holt's	
	No. of Rules	Accuracy %	No. of Rules	Accuracy %	No. of Rules	Accuracy %	No. of Rules	Accuracy %
M-of-N	64	100	64	100	64	100	26	63
Exactly	64	100	64	100	64	100	26	66
Exactly2	606	57	606	57	606	57	26	71
Heart	514	14	7485	17	261	0	66	69
Vote	134	63	494	63	134	63	48	87
Credit	873	6	151133	21	887	2	83	69
Mushroom	537	100	1	100	1	100	119	100
LED	10	100	1788	100	10	100	27	76
Letters	436	0	4278	0	23	0	27	0
Derm	6358	24	57336	38	320	0	189	57
Derm2	5480	86	63254	75	310	14	135	61
WQ	9209	37	99903	25	465	2	132	61
Lung	458	100	3986	100	26	100	158	100

B. Classification Accuracy and Number of Generated Rules

We carried out further experiment to find the accuracy of the classification and generate the number of rules for all datasets based on the obtained reducts that were generated by applying the proposed approaches. Classification was performed using the Standard Voter algorithm found in the ROSETTA library [40]. ROSETTA is a rough set toolkit that is used for analyzing data. It can be found at http://www.lcb.uu.se/tools/rosetta/resources.php. The independent tests were performed with the Voting parameter set to Simple. All other parameters were set to the default values. We compare our results with the results obtained from the methods that exist in ROSETTA system; genetic algorithm, Johnson's algorithm and Holt's 1R algorithm [40] in terms of the classification accuracy and the number of generated rules. The data presented in Table VI shows the classification accuracy as a percentage along with the number of rules.

From the results, we can observe that GA-Fuzzy-GDA is able to produce 100% classification accuracy for five datasets (i.e. M-of-N, Exactly, Mushroom, LED, and lung). The accuracy produced when using GA-Fuzzy-GDA is between 57% to 86% on three datasets and four of the datasets with accuracy in between 6% to 37% (excluding 0% on Letters dataset). GA-Fuzzy-GDA is able to obtain the best result in one dataset (i.e. Derm2 dataset) and comparable with the other approaches in the other datasets. For M-of-N and Exactly datasets that are nearly at the same size, it can obtain the same results as the other approaches. Fig. 5 illustrates the classification accuracy from all approaches. Holt's algorithm shows good results in comparison of the other two methods in the literature i.e. GA and Johnson algorithms. Our approach is better than GA and Johnson in most of the cases. Our competitor is the Holt's algorithm. In comparison with our approach, Holt's outperforms our approach on five datasets (i.e. Exactly2 (71%), Heart (69%), Vote (87%), Credit (69%), Derm (57%) and WQ (61%)). On the other hand, GA-Fuzzy-GDA outperforms Holt's on 4 datasets (i.e. M-of-N (100%), Exactly (100%), LED (100%) and Derm2 (86%)). Note that GA-Fuzzy-GDA outperforms Holt's in the two datasets (i.e., LED and Derm2) which are the most complex datasets based on the number of the number of attributes i.e. 24 and 34 attributes, respectively.
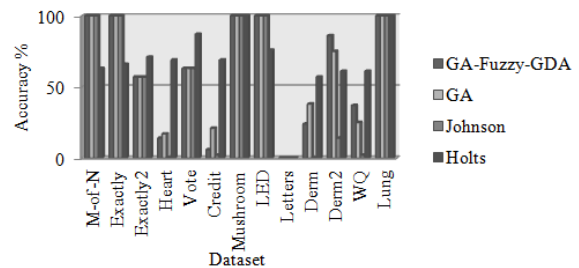

Fig. 5 Classification Accuracy

Fig. 6 shows the comparison between GA-Fuzzy-GDA and other approaches in terms of number of generated rules. Note that, the lower number of the generated rules shows the performance of the algorithm is better than the others. From Fig. 6, we can see the number of rules generated by the reducts produced by GA-Fuzzy-GDA algorithm is lower when compared to GA algorithm in most of the tested datasets, and show the same performance with Johnson algorithm. However, Holt's algorithm in most of the cases is better than GA-FUZZY-GDA algorithm except on LED dataset. Clearly, the above results show how the selected attributes by each method affect the performance of the classification accuracy and the number of the generated rules, since different attributes may be contained in each subset. That is, we can see if the selected attributes from the GA-FUZZY-GDA method are used, it causes a different accuracy percentage and different number of generated rules compared with those that can be obtained using the attributes selected from other approaches. This improves the efficiency of GA-FUZZY-

GDA in selecting the most related attributes that enhance the performance of the learning algorithm.
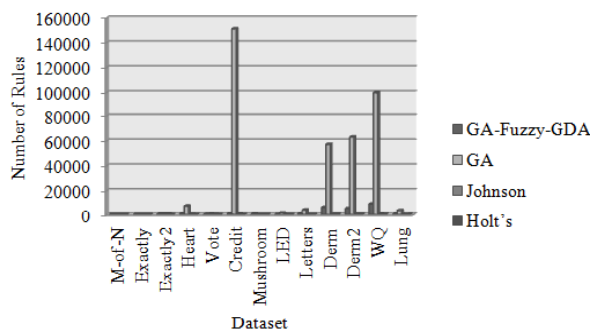

Fig. 6 Number of generated rules

VI. CONCLUSION AND FUTURE WORK

In this paper, two attribute reduction methods were proposed. To the best of our knowledge, this is the first such algorithm that aims at this problem domain. Two local search algorithms were hybridized with GA to form the MA framework. The first method was the fuzzy record to record travel algorithm where a fuzzy logic controller was embedded in RRT to control the single parameter that controls the acceptance of the worst solutions. The second local search was the fuzzy controlled great deluge algorithm.

The performance of the proposed algorithms was tested on standard benchmark datasets and comparison results were presented. The preliminary results showed that GA-Fuzzy-GDA and GA-Fuzzy-RRT were comparable because sometimes GA-Fuzzy-RRT outperforms GA-Fuzzy-GDA and vice versa. Experimental results had showed that our approach was able to produce three best-known results in the literature and comparable with other approaches for the rest of the datasets. We had used ROSETTA to find out the classification accuracy and the number of generated rules, when using the features that are selected using our approach. In terms of the classification accuracy, it can be said that our approach is comparable with the other methods existing in ROSETTA, and it can outperform some methods in terms of the number of generated rules as well.

REFERENCES

[1] R. Jensen and Q. Shen, "Fuzzy-rough sets for descriptive dimensionality reduction," presented at the Fuzzy Systems, 2002. FUZZ-IEEE'02. Proceedings of the 2002 IEEE International Conference on, 2002.
[2] L. Ke, Z. Feng, and Z. Ren, "An efficient ant colony optimization approach to attribute reduction in rough set theory," *Pattern Recognition Letters,* vol. 29, 2008, pp. 1351-1357.
[3] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, Third ed. Orlando, FL, USA Academic Press, Inc. , 2006.
[4] Z. Pawalk, "Rough sets: theoretical aspects of reasoning about data," ed: Dordrecht: Kluwer Academic Publishers, 1991.
[5] Z. Pawlak, "Rough Sets," *International Journal of Information and Computer Sciences,* vol. 11, 1982, pp. 341-356.
[6] Z. Pawlak, "Rough sets and data analysis," presented at the Fuzzy Systems Symposium, 1996. 'Soft Computing in Intelligent Systems and Information Processing', 1996.
[7] Z. Pawlak and A. Skowron, "Rudiments of rough sets," *Information sciences,* vol. 177, 2007, pp. 3-27.
[8] R. W. Swiniarski and A. Skowron, "Rough set methods in feature selection and recognition," *Pattern Recognition Letters,* vol. 24, 2003, pp. 833-849.
[9] A. Skowron and C. Rauszer, "The discernibility matrices and functions in information systems," *Intelligent decision support–Handbook of applications and advances of the rough set theory*, 1992, pp. 311–362.
[10] R. Jensen and Q. Shen, "Semantics-Preserving Dimensionality Reduction: Rough and Fuzzy-Rough-Based Approaches," *IEEE Trans. on Knowl. and Data Eng.,* vol. 16, 2004, pp. 1457-1471.
[11] J. G. Bazan, A. Skowron, and P. Synak, "Dynamic Reducts as a Tool for Extracting Laws from Decisions Tables," presented at the Proceedings of the 8th International Symposium on Methodologies for Intelligent Systems, 1994.
[12] J. Wroblewski, "Finding minimal reducts using genetic algorithms," presented at the 2nd Annual Joint Conf. on Information Sciences, Wrightsville Beach, NC, 1995.
[13] A.-R. Hedar, J. Wang, and M. Fukushima, "Tabu search for attribute reduction in rough set theory," *Soft Comput.,* vol. 12, 2006, pp. 909-918.
[14] J. Wang, A.-R. Hedar, G. Zheng, and S. Wang, "Scatter Search for Rough Set Attribute Reduction," presented at the Bio-Inspired Computing: Theories and Applications, 2007. BIC-TA 2007. Second International Conference on, 2007.
[15] S. Abdullah and N. S. Jaddi, "Great Deluge Algorithm for Rough Set Attribute Reduction," in *Database Theory and Application, Bio-Science and Bio-Technology.* vol. 118, Y. Zhang, A. Cuzzocrea, J. Ma, K.-i. Chung, T. Arslan, and X. Song, Eds., ed: Springer Berlin Heidelberg, 2010, pp. 189-197.
[16] S. K. Jihad and S. Abdullah, "Investigating composite neighbourhood structure for attribute reduction in rough set theory," presented at the Intelligent Systems Design and Applications (ISDA), 10th International Conference, 2010.
[17] Y. Z. Arajy and S. Abdullah, "Hybrid variable neighbourhood search algorithm for attribute reduction in Rough Set Theory," presented at the Intelligent Systems Design and Applications (ISDA), 2010.
[18] S. Abdullah, N. R. Sabar, M. Z. A. Nazri, H. Turabieh, and B. McCollum, "A constructive hyper-heuristics for rough set attribute reduction," presented at the Intelligent Systems Design and Applications (ISDA), 2010.
[19] M. Mafarja and S. Abdullah, "Modified great deluge for attribute reduction in rough set theory," presented at the Fuzzy Systems and Knowledge Discovery (FSKD), 2011.
[20] D. Koller and M. Sahami, "Toward Optimal Feature Selection," presented at the International Conference on Machine Learning 1996.
[21] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial Intelligence* vol. 97, 1997, pp. 273-324.
[22] G. H. John, R. Kohavi, and K. Pfleger, "Irrelevant Features and the Subset Selection Problem", 1994.
[23] R. Jensen and Q. Shen, *Computational Intelligence and Feature Selection: Rough and Fuzzy Approaches*: Wiley-IEEE Press, 2008.
[24] J. Dong, N. Zhong, and S. Ohsuga, "Using Rough Sets with Heuristics for Feature Selection," presented at the Proceedings of the 7th International Workshop on New Directions in Rough Sets, Data Mining, and Granular-Soft Computing, 1999.
[25] J. Bazan, H. S. Nguyen, S. H. Nguyen, P. Synak, and J. Wróblewski, "Rough set algorithms in classification problem," *Rough Set Methods and Applications. Physica Verlag,* 2000, pp. 49–88.
[26] P. Moscato, "On Evolution, Search, Optimization, Genetic Algorithms and Martial Arts - Towards Memetic Algorithms," California Instiyute of Technology, Pasadena, CA1989.
[27] C. Guerra-Salcedo, S. Chen, D. Whitley, and S. Smith, "Fast and accurate feature selection using hybrid genetic strategies," presented at the Genetic and Evolutionary Computation Conf, 1999.
[28] I. S. Oh, J. S. Lee, and B. R. Moon, "Hybrid genetic algorithms for feature selection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 26, 2004, pp. 1424-1437.
[29] N. Krasnogor, "Studies on the Theory and Design Space of Memetic Algorithms" PhD, University of the West of England, 2002, 2002.

[30] W. E. Hart, N. Krasnogor, and J. E. Smith, "Recent Advances in Memetic Algorithms," vol. 166, 2005.
[31] J. H. Holland, *Adaptation in natural and artificial systems*: MIT Press, 1992.
[32] J. Yang and V. G. Honavar, "Feature Subset Selection Using a Genetic Algorithm," *IEEE Intelligent Systems,* vol. 13, pp. 44-49, 1998.
[33] Z. Michalewicz, *Genetic algorithms+ data structures*: Springer, 1996.
[34] L. A. Zadeh, "Fuzzy sets," *Information and Control,* vol. 8, 1965, pp. 338-353.
[35] R. Jensen and Q. Shen, "New approaches to fuzzy-rough feature selection," *Fuzzy Systems, IEEE Transactions on,* vol. 17, 2009, pp. 824-838.
[36] E. Cox, *The fuzzy systems handbook: a practitioner's guide to building, using, and maintaining fuzzy systems*: Academic Press Professional, Inc., 1994.
[37] H. Zimmermann, *Fuzzy Set Theory and its Applications*: Kluwer Academic Publishers, Boston, 1996.
[38] G. Dueck, "New Optimization Heuristics The Great Deluge Algorithm and the Record-to-Record Travel," *Journal of Computational Physics,,* vol. 104, 1993, pp. 86-92.
[39] R. Jensen and Q. Shen, "Finding Rough Set Reducts with Ant Colony Optimization," in *Proceedings of the 2003 UK Workshop on Computational Intelligence*, ed, 2003, pp. 15–22
[40] A. Øhrn, "Discernibility and rough sets in medicine: tools and applications," Department of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway, 1999.
[41] S. Abdullah, N.R. Sabar, M.Z. Ahmad Nazri, M. Ayob. "An Exponential Monte-Carlo algorithm for feature selection problems", Computers & Industrial Engineering. vol 67 (2014), pp 160-167.