IMPROVING ANT SWARM OPTIMIZATION WITH EMBEDDED VACCINATION FOR OPTIMUM REDUCTS GENERATION

LUSTIANA PRATIWI
DR. CHOO YUN HUOY
DR. AZAH KAMILAH BINTI DRAMAN @ MUDA
NOOR AZILAH BINTI DRAMAN@MUDA

(11th International Conference on Hybrid Intelligent Systems 2011 (HIS 2011), 5-8 December 2011, Melaka)

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Improving Ant Swarm Optimization with Embedded Vaccination for Optimum Reducts Generation

[1]Lustiana Pratiwi, [2]Yun-Huoy Choo, [3]Azah Kamilah Muda, [4]Noor Azilah Muda
Faculty of Information and Communication Technology,
Universiti Teknikal Malaysia Melaka (UTeM),
Melaka, Malaysia
[1]lustiana@gmail.com, [2]huoy@utem.edu.my, [3]azah@utem.edu.my, [4]azilah@utem.edu.my

Abstract— Ant Swarm Optimization refers to the hybridization of Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO) algorithms to enhance optimization performance. It is used in rough reducts calculation for identifying optimally significant attributes set. This paper proposes a hybrid ant swarm optimization algorithm by using immunity to discover better fitness value in optimizing rough reducts set. Unlike a conventional PSO/ACO algorithm, this hybrid algorithm shows improvement of the classification accuracy in its generated rough reducts to solve NP-Hard problem. This paper has evaluated the immune algorithm in 12 common benchmark dataset to evaluate the performance of rough reducts-based on attribute reduction. The results show that immune ant swarm algorithm is very competitive in terms of fitness value, number of iterations, and classification accuracy to produce a better optimization technique and more accurate results in rough reducts generation. The results also show that immune ant swarm optimization provides a slight increase in accuracy when compared to the differential evolution variant.

Keywords-component; rough reducts; particle swarm optimization; ant colony optimization; immunity; ant swarm optimization

I. INTRODUCTION

In the concept of rough set theory, reducts is an important attribute set which can discern all discernible objects by the original of information system. Given an information system $IS = (U, A)$, let $B \subseteq A$, where U is a non-empty finite set of objects called the universe and A- is a non-empty finite set of attributes, such that $a : U \rightarrow V_a$ for every $a \in A$. A reduct of A is a minimal set of attributes $B \subseteq A$ such that all attributes $a \in A - B$ are dispensable and an associated equivalence of indiscernibilty relation denoted by $IND(B)$

$$IND(B) = \{(x, x') \in U^2 | \forall_a \in B \; a(x) = a(x')\}, \quad (1)$$

and $IND(B)$ is called the *B-indiscernibillity* relation. Then an attribute a, is said to be dispensable in $B \subseteq A$ if $IND(B) = IND(B - \{a\})$. Otherwise, the attribute is indispensable in B[1], [2], [3], [4], and [5].

Reducts are such subsets that are minimal, and do not contain any dispensable attributes. The set of all reducts of an information system IS is denoted $RED(IS)$ or simply RED. Reducts calculation has great importance in features selection analysis. It enables the calculation of absolute reduction as well as relative reduction with core. However, the computational complexity of generating optimal reducts is very high . Since the search space increase exponentially with the number of attributes, finding the optimal reducts, a minimal reducts with minimal cardinality of attributes among all reducts is a NP-hard problem [2].

Formally, the minimum attribute reduction problem is a nonlinearly constrained combinatorial optimization problem. Hence, global optimization methods could be used to solve it. Therefore, many existing optimization technique have been proposed to solve the optimal reducts problem, such as Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO) and Immunity. This paper has analyzed the results from the previous study and suggested an improved solution by proposing a rough reducts optimization using a hybrid approach of PSO/ACO. This solution is also enhanced with embedded vaccination process which can significantly reduce iteration complexity and increase the fitness value, when PSO does not possess the ability to improve upon the quality of the solutions as the number of generations is increased.

The rest of the paper is organized as follows. Section II presents some related works on the optimization approach in attribute reduction problem. Section III describes the improved ant swarm optimization technique to enhance the previous studies with higher possibility of finding a minimum reduction with immunity. The effectiveness of proposed algorithm is demonstrated, compared to the other variants, and the computation results are discussed in Section IV, and finally, in section V outlines the conclusions and the future work.

II. RELATED WORK

Various optimization techniques have been suggested to solve the crucial problem in rough reduct; gaining interest in metaheuristic solutions of NP-hard problem, such as genetic algorithm (GA), ant colony optimization (ACO), artificial immune system (AIS), and more recently particle swarm optimization (PSO). Some researchers have proposed stochastic methods for inducing rough reducts [6]. Some studies used genetic algorithms to find minimal reducts [7]. Combination of genetic algorithm and greedy algorithm to generate short reducts was also proposed. However, it used

highly time-consuming operations and cannot assure the optimality of the resulting reducts subset.

Bjorvand applied genetic algorithms to compute approximate reducts [8]. He followed Wro'blewski's work as a foundation, and has suggested several variations and practical improvements in both speed and the quality of approximation. To obtain a good initial population for the GA, Bjorvand included the attribute core in all candidates. Zhai also proposed an integrated feature extraction approach based on rough set theory and genetic algorithms [9]. Based on this approach, a prototype feature extraction system has been established and illustrated in an application for the simplification of product quality evaluation. The results show that it can remarkably reduce the cost and time consumed on product quality evaluation without compromising the overall specifications of the acceptance tests [9] and [10].

Another commonly used optimization technique in data mining strategy is particle swarm optimization (PSO) technique. PSO was proposed by Kennedy and Eberhart [11] to be applied to difficult combinatorial optimization problem which aims to find reducts of minimal cardinality for attribute subsets [12], [13], and [14]. In [12] and [14] PSO is applied to find reducts of minimal cardinality. Like classical genetic algorithms, the particle's position is a binary representation of the attribute subsets. The fitness function and position-updating strategy are also the key factors in PSO for feature selection, which needed to be improved.

On the other hand, B. Yue et al has discovered the best feature combinations in an efficient way to observe the change of the positive region as the particle proceed through the search space in PSO [13]. A comparison between PSO and GA was presented, where PSO does not have genetic operators like crossover and mutation. The search is influenced by the speed of the particles and the processing time has reduced as compared to other approaches [15]. Subsequently, better results were obtained by using PSO algorithm, as reported in [13] and [15].

Chen et al. [16] and Zeng et al. [17] verified and demonstrated some experiments to provide efficient solution to find the minimal features subset using ant colony optimization (ACO). ACO has been verified again by [18] to provide competitive solutions efficiently to deal with attribute reduction in rough set theory. This algorithm has the features to update the pheromone trails of the edges connecting every two different attributes of the best-so-far solution. The pheromone values were limited between the upper and lower trail limits and a rapid procedure was used to construct candidate solutions. Due to its pheromone update rule and solution construction process, the proposed algorithm has the ability to identify solutions with very small cardinality rapidly [18].

Dongyi Ye et al [19] presented a new attribute reduction algorithm by using a combination of binary particle swarm optimization and the vaccination approach. The experimental result has remarkably improved local search from the immune selection mechanism and outperformed some recent global optimization techniques.

III. IMMUNE ANT SWARM OPTIMIZATION ALGORTIHM FOR ROUGH REDUCTS (IASORR)

The term "reducts" corresponds to a wide class of concepts. What typifies all of them is that they are used to reduce information (decision) systems by removing redundant attributes. Given an information system $IS = (U, A)$, a reducts is a minimal set of attributes $B \subseteq A$ such that $IND(B) = IND(A)$, where $IND(B) = IND(A)$ are the indiscernibility relations defined by B and A, respectively by Pawlak and Skowron in [20].

The intersection of all reducts is called a core. Intuitively, reducts is a minimal set of attributes from A that preserves the original classification defined by A. The improvement in rough reducts optimization using PSO/ACO with vaccination is based on the common characteristics of both PSO and ACO algorithm, like, survival as a colony by sharing information locally and globally in the swarm between particles (ants).

A. Ant Swarm Optimization for Rough Reducts (ASORR)

Both PSO and ACO adapt swarm intelligence metaheuristics which is based on population global search and co-operative biologically inspired algorithm motivated by social analogy [21]. PSO was inspired by real life social behavior of bird flocking or fish schooling, while ACO imitates foraging behavior of real life ants. PSO still has the problems of dependency on initial point and parameters, difficulty in finding their optimal design parameters, and the stochastic characteristic of the final outputs for local searching [23].

On the other hand, ACO has positive feedbacks for rapid discovery of good solutions and a simple implementation of pheromone-guided will improve the performance of PSO. Thus in this study, a simple pheromone-guided mechanism is explored to improve the performance of PSO method for optimization of rough reducts [24].

1) Particle Swarm Optimization

In PSO, particles as candidate solutions of a population, simultaneously coexist and evolve based on knowledge sharing with neighboring particles. Each particle generates a solution using directed velocity vector, while flying through the problem search space. Each particle modifies its velocity to find a better solution (position) by applying its own flying experience for the best position memory found in the earlier flights and experience of neighboring particles as the best-found solution of the population [21].

Each particle's movement is the composition of an initial random velocity and two randomly weighted influence; individuality, the tendency to return to the particle's best position P_{best}, and sociality, the tendency to move forwards the best previous position of the neighborhood's G_{best}. Particles update their positions and velocities as shown below

$$v_{t+1}^i = w_t v_t^i + c_1 r_1 \left(p_t^i - x_t^i \right) + c_2 r_2 (p_t^g - x_t^i), \quad (2)$$

$$x_{t+1}^i = \begin{cases} 1, rand() < sig(v_{t+1}^i) \\ 0, rand() \geq sig(v_{t+1}^i) \end{cases}, \quad (3)$$

$$sig(x) = \frac{1}{1+e^{-x}} \qquad (4)$$

where x_t^i represents the current position of particle i in solution space and subscript t indicates an iteration count; p_t^i is the best-found position of particle i up to iteration count t and represents the cognitive contribution to the search velocity v_t^i. Each component of v_t^i can be clamped to the range $[-v_{max}, v_{max}]$ to control excessive roaming of particles outside the search space; p_t^g is the global best-found position among all particles in the swarm up to iteration count t and forms the social contribution to the velocity vector; r_1 and r_2 are random numbers uniformly distributed in the interval $(0, 1)$, while c_1 and c_2 are the cognitive and social scaling parameters, respectively; w_t is the particle inertia, which is reduced dynamically to decrease the search area in a gradual fashion by Shi et al in [25]. The variable w_t is updated as

$$w_t = (w_{max} - w_{min}) * \frac{t_{max} - t}{t_{max}} + w_{min}, \qquad (5)$$

where, w_{max} and w_{min} denote the maximum and minimum of w_t respectively; t_{max} is a given number of maximum iterations. Particle i flies toward a new position according to (1) and (2). In this way, all particles P of the swarm find their new positions and apply these new positions to update their individual best p_t^i points and global best p_t^g of the swarm. This process is repeated until iteration count $t = t_{max}$ (a user-defined stopping criterion is reached).

Given an information system $IS = (U, A)$, $A = (C \cup D)$, where C is a non-empty finite set of condition attributes and D is a non-empty finite set of decision attributes, such that $RED \subseteq C$. In the algorithm of PSO, P denotes the number of particles in the population; $f(x_t^i)$ represents the objective function value of particle i at position x [20] and calculated as

$$f(x_t^i) = \alpha * \gamma_{x_t^i}(D) + \beta * \frac{|C| - |x_t^i|}{|C|}, \qquad (6)$$

where $\gamma_{x_t^i}(D)$ is the classification quality of particle condition attribute set x_t^i, which contains the reducts RED, and relative to decision table D, defined as follows

$$\gamma_{x_t^i}(D) = \frac{r_{RED}}{r_C} \text{ [22]}, \qquad (7)$$

where r_{RED} represents a degree of dependency of RED on D and r_C represents a degree of dependency of C on D. $|x_t^i|$ is the '1' number of the length of selected feature subset or the number of attributes for particle x_t^i, while population of solutions P is at iteration count t. $|C|$ is the total number of condition attributes. α and β parameters correspond to importance of classification quality and subset length, $\alpha \in [0,1]$ and $\beta = 1 - \alpha$ [21].

2) Ant Colony Optimization (ACO)

ACO studies the concept of "the emergent collective intelligence of groups of simple agents". As [26] discussed ACO algorithm was motivated by ants social behavior. Dre'o, et al and Socha in [21] found that ants behave as social insects that directly more toward the survival of the colony as a whole than that of a single individual of the colony. Indirect co-operative foraging process of ants is very interesting behavior to be adopted in searching problem of PSO.

ACO algorithm was implemented in the second stage. ACO works as a local search to attain rapidly feasible solution space, where the ants apply pheromone-guided mechanism to update the positions found by the particles in the previous stage. The ants also use their capability to locate their food resources found by their mates and proven those behaviors to stimulate the optimization of ant foraging behavior in ACO [27]. The proposed ACO algorithm from the previous research handles P ants equal to the number of particles in PSO and it is proven in mathematical concept that ACO helps PSO not only to efficiently perform global exploration for rapidly attaining the feasible solution space in [21]. Each ant i generates a solution z_t^i around p_t^g the global best-found position among all particles in the swarm up to iteration count t as

$$z_t^i = \mathcal{N}(p_t^g, \sigma), \qquad (8)$$

In (8), the algorithm generates components of solution vector z_t^i, which satisfy Gaussian distributions [21], [24], and [28] with mean p_t^g and standard deviation σ, where, initially at $t = 1$ value of $\sigma = 1$ and is updated at the end of each iteration as $\sigma = \sigma \times d$, where, d is a parameter in (0.25, 0.997) [21] to produce better position value within this boundaries and if $\sigma < \sigma_{min}$ then $\sigma = \sigma_{min}$, where, σ_{min} is a parameter in (10^{-4}, 10^{-2}).

In the proposed algorithm, to compute the objective function value of $f(z_t^i)$, it uses z_t^i and replace the position of x_t^i the current position of particle i in the swarm if $f(z_t^i) < f(x_t^i)$ as $x_t^i = z_t^i$ and $f(x_t^i) = f(z_t^i)$ [21]. This simple pheromone-guided mechanism considers, there is highest density of trails (single pheromone spot) at the global best solution p_t^g of the swarm at any iteration $t + 1$ in each stage of ACO implementation and all ants P search for better solutions in the neighborhood of the global best solution p_t^g [21]. In the beginning of the search process, ants explore larger search area in the neighborhood of p_t^g due to the high value of standard deviation σ and intensify the search around p_t^g.

The implementation of ACO in the second stage of the proposed PSO/ACO approach was based on the studies by Shelokar in [20]. It has been proven that PSO was able to discover reasonable quality solutions much faster than other evolutionary algorithms. However, PSO does not possess the ability to improve upon the quality of the solutions as the number of generations is increased. In our application of PSO/ACO approach, a simple pheromone-guided search

mechanism of ant colony was implemented which acted locally to synchronize positions of the particles in PSO to attain the feasible domain of the objective function [20] faster.

B. Immnuity System

The biological immune system, consisting of the innate and adaptive immune systems, is an effective and efficient defense mechanism against infections [23]. The important characters of immunity system are diversity of antibodies and immunity memory. To ensure the diversity of antibodies during the evolvement, high affinity and low concentration antibodies are stimulated, but low affinity and high concentrations antibodies are restrained [23].

Immunity memory will be a part of memory cells in immunity system, which is retained during the reaction between the invasion of antigen and antibody. It is the certainty mapping of selected some antibodies in a given probability of α, where $0 < \alpha < 1$, and only the high affinity antibodies are breed and mutated, but the low-affinity antibodies still exist in the immune system, and gradually are expelled [23].

According to the different important degrees of attribute, the particles are vaccinated by the abstracted bacteria. Theoretically, the more the weight of an attribute is greater, the more the attribute is more important and it is greater that the probability of attribute is selected [23]. Otherwise, it is smaller which the probability of attribute is selected. So we can vaccinate each particle by the prior knowledge and the more optimization particles will be generated. For mutation probability of a bit in a particle can be defined as [23]:

$$\eta = k_1(w_i - p_i)^2 - k_2, \tag{9}$$

$$p_i = \begin{cases} p_i, rand() \geq \eta \\ !p_i, rand() < \eta \end{cases}, \tag{10}$$

where w_i is the weight of the ith attribute which is mapped to [0,1] in particle P, and p_i is the new value of condition attribute. Where k_1 and k_2 are the adjustive parameters, $rand()$ is the random number of uniform distribution and $k_1 = 0.4$, $k_2 = 0.005$, $rand() \in [-1,1]$. If the difference between w_i and p_i is greater, the value η is greater, and the aberrance probability of p_i will be greater [23]. Otherwise, the probability of mutation is smaller and even never mutate as the algorithm progresses as shown in the following algorithms:

```
Step 1: Initialize Optimization

   1.1 Initialize algorithm constants t_max, P, and
       {0,1}^m is the m-dimensional Boolean particle
       space.
   1.2 Initialize the positive acceleration
       constants c_1,c_2 and MaxFit as the maximum
       fitness value.
   1.3 Calculate the inertia weight of each
       particle space in (5).
   1.4 Initialize randomly all particles'
       positions x_t^i velocities v_t^i.
```

```
   1.5 Evaluate all particles' objective function
       value f(x_t^i) in (6).
   1.6 Find f_t^best(p_t^best) = max{f(x_t^1),f(x_t^2),···,f(x_t^P)} and
       initialize p_t^g = p_t^best and Gbest_t = f_t^best(p_t^best).
```

```
Step 2: Perform Optimization

   While (t < t_max)
   2.1 Calculate all particles' velocities v_t^i in
       (2) and update particles' positions in (3).
   2.2 Vaccinate each particle in (9) and (10).
   2.3 Evaluate all particles objective function
       value f(x_t^i) in (6).
   2.4 Generate P solutions z_t^i using (8).
   2.5 Evaluate objective function value f(z_t^i) in
       (6) and if f(z_t^i) > f(x_t^i) then f(z_t^i) = f(x_t^i) and
       z_t^i = x_t^i
   2.6 Update particle best position if Pbest_t^i <
       f(x_t^i) then update Pbest_t^i = f(x_t^i) and p_t^i = x_t^i
   2.7 Find f_t^best(p_t^best) = max{Pbest_t^1,Pbest_t^2,···,Pbest_t^P} and
       if Pbest_t^i > the best fitness value f_t^best(p_t^best)
       in history then Gbest_t = f_t^best(p_t^best) and p_t^g =
       p_t^best
   2.8 If the Gbest_t continuously can't be changed
       and can't meet the 10^th termination
       conditions by several evolutionary
       generations, the program will go to (12)
       refresh the particles.
```

```
Step 3: Report best solution p^g as P_gbest global
best position of the swarm with objective function
value f(p^g)
```

According to the principle that the high affinity and low concentration antibodies are stimulated, but low affinity and high concentrations antibodies are restrained, the ith particle selected probability formulas based on concentration particle is as follows [23], [29], and [30]:

$$D(X_i) = \frac{1}{\sum_{j=1}^{N_0} |f(x_i) - f(x_j)|}, j = 1,2,\dots,N_0, \tag{11}$$

$$p(x_i) = \frac{\frac{1}{D(X_i)}}{\sum_{i=1}^{N_0} \frac{1}{D(X_i)}} = \frac{\sum_{j=1}^{N_0} |f(x_i) - f(x_j)|}{\sum_{i=1}^{N_0} \sum_{j=1}^{N_0} |f(x_i) - f(x_j)|}, i,j = 1,2,\dots,N_0. \tag{12}$$

IV. EXPERIMENTAL RESULTS

The performance of the proposed enhanced Ant Swarm algorithm for global optimization function has been tested on several well-unknown benchmark multimodal problems [13]. All the test functions are multimodal in nature. Because of the characteristics from previous work [31], it is difficult to seek for the global minima. Ant swarm optimization algorithm parameter settings used in all the simulations is given as: number of particles, $P = 10$; cognitive and social scaling parameters, $c_1 = 2$, $c_2 = 2$; maximum and minimum values of inertia weights, $w_{max} = 0.7$, $w_{min} = 0.4$; maximum number of iterations, $t_{max} = 100 * n$, n is the size of solution vector. Implementation of IASORR has been tested on12 datasets.

The experimental results are presented based on the number of reducts and iterations, fitness values, and the classification accuracy for performance analysis are shown in Table I and II. Three algorithms, which are Particle Swarm Optimization for Rough Sets-based Feature Selection (PSORSFS), Ant Swam Optimization for Rough Reducts (ASORR), and Immune Ant Swarm Optimization for Rough Reducts (IASORR); were implemented by using Naïve Bayes to extract rules from the data for rule induction in classification. Ten-fold cross validation was applied to estimate the classification accuracy.

Results reported in Table I are in term of, the numbers of reducts calculation (also shown graphically in Figure 1) and iterations, which has shown that IASORR has more optimal results for number of iterations than another two algorithms in most of datasets, but in average of attribute reduction, all three algorithms yield the same number of reducts. However, the number of reducts is not sufficient to reflect the performance, in terms of finding the optimal fitness value which is able to gain higher classification accuracy.

The fitness values of each algorithm (shown graphically in Figure 2) reported in Table II has shown that IASORR can achieve better values while the fitness value is proven to be affected by number of reducts. As shown in Table I, IASORR obtained better results than PSORSFS and ASORR by reducing 2 to 3 iterations in average for 10 independent runs. Table II shows the results in terms of classification accuracy where IASORR has induced more significant optimal solution with the generated fitness values. IASORR has achieved best fitness values in 8 datasets as compared to PSORSFS and ASORR, which only gained best results in 2 and 4 datasets respectively.

In term of classification accuracy, IASORR has achieved best accuracy readings in 4 datasets while ASORR in only 1 datasets. On the other hand, PSORSFS lines on top of the three techniques which has gained best accuracy readings in 7 datasets. Thus, based on the experimental results, the IASORR technique has better performances, both in gaining higher fitness value and better quality of reducts as compared to its opponents.
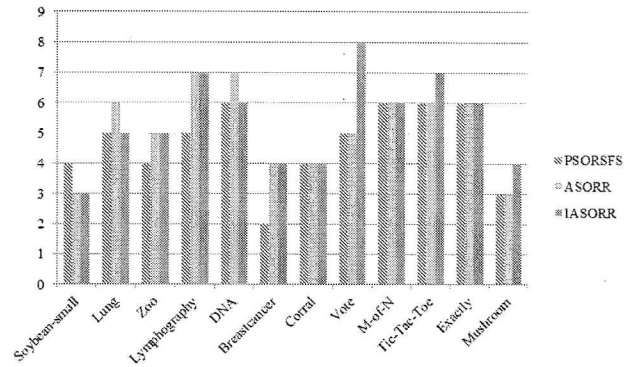

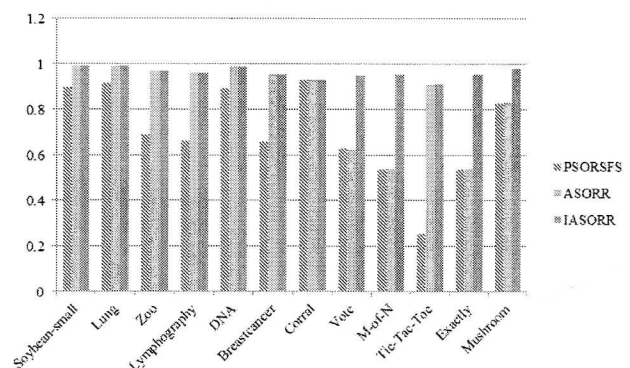Figure 1: Performance comparison in term of no. rough reducts set

Figure 2: Performance comparison in term of fitness values

TABLE I. IASORR EXPERIMENTAL RESULTS ON NO. OF REDUCTS AND ITERATIONS

No.	Dataset	Number of Condition Attributes	Instances	No. of Reducts			No. of Iteration		
				PSORSFS	ASORR	IASORR	PSORSFS	ASORR	IASORR
1	Soybean-small	35	47	4	3	3	86	100	78
2	Lung	56	32	5	6	5	90	100	77
3	Zoo	16	101	4	5	5	100	100	100
4	Lymphography	18	148	5	7	7	100	100	100
5	DNA	57	318	6	7	6	100	100	100
6	Breastcancer	9	699	2	4	4	100	100	100
7	Corral	6	64	4	4	4	100	100	100
8	Vote	16	300	5	5	8	100	100	100
9	M-of-N	13	1000	6	6	6	100	100	100
10	Tic-Tac-Toe	9	958	6	6	7	100	100	100
11	Exactly	13	1000	6	6	6	100	100	100
12	Mushroom	22	8124	3	3	4	100	100	100
Average				5	5	5	98	100	97

TABLE II.   IASORR EXPERIMENTAL RESULTS ON FITNESS VALUE AND CLASSIFICATION ACCURACY (%)

No.	Dataset	Fitness Value				Classification Accuracy (%)			
		PSORSFS	ASORR	IASORR	Best Values	PSORSFS	ASORR	IASORR	Best Values
1	Soybean-small	0.8971	0.9917	0.9929	IASORR	100.0000	100.0000	100.0000	PSORSFS
2	Lung	0.9152	0.9891	0.9914	IASORR	87.5000	87.5000	90.6250	IASORR
3	Zoo	0.6891	0.9688	0.9684	ASORR	98.0198	98.0198	98.0198	PSORSFS
4	Lymphography	0.6652	0.9631	0.9611	ASORR	83.7838	81.7568	83.7838	PSORSFS
5	DNA	0.8927	0.9879	0.9896	IASORR	34.5912	34.2767	32.7044	PSORSFS
6	Breastcancer	0.6607	0.9556	0.9556	ASORR	96.1373	96.1373	96.1373	PSORSFS
7	Corral	0.9333	0.9333	0.9333	PSORSFS	84.3750	84.3750	84.3750	PSORSFS
8	Vote	0.6319	0.6237	0.9500	IASORR	92.7833	92.4167	93.6667	IASORR
9	M-of-N	0.5385	0.5385	0.9538	IASORR	95.7000	95.7000	95.7000	PSORSFS
10	Tic-Tac-Toe	0.2575	0.9147	0.9145	ASORR	69.2119	69.6242	70.9812	IASORR
11	Exactly	0.5385	0.5385	0.9538	IASORR	68.8000	68.8000	68.8000	PSORSFS
12	Mushroom	0.8316	0.8343	0.9811	IASORR	79.7717	87.5677	87.5677	ASORR
	Average	0.7043	0.8533	0.9621	IASORR	82.5562	83.0145	83.5301	IASORR

V.   CONCLUSION

An improved approach of rough reducts optimization based on PSO/ACO with vaccination has been presented. Although the approach is quite simple in structure, it is feasible to solve NP-Hard problem and accelerate the search process. The main contribution of this research lies in the thoroughly an application of ant swarm optimization technique in depth as compared to the past research in attribute reduction. By combining PSO and ACO algorithms with embedded immunity process, the proposed approach is expected to be able to generate better optimal rough reducts, where PSO algorithm performs the global exploration which can effectively reach the optimal or near optimal solution.

This study is also expected to enhance the optimization ability by defining a suitable fitness function with vaccination process to increase the competency in attribute reduction. Thus, the experimental results have also provided further justification on previous studies which have implemented PSO in attribute reduction domain. The initial results of IASORR are promising in most of the tested datasets. Hence, future works are to test on the enhanced IASORR algorithm in various domains to validate its performance in yielding the optimal reducts set.

ACKNOWLEDGMENT

REFERENCES

[1]   Pawlak, Zdzislaw, et al., "Rough Sets," in Communications of the ACM, vol. 38, pp. 88—95, 1995.

[2]   Wang, X., Xu, R., and Wang, W., "Rough Set Theory: Application in Electronic Commerce Data Mining," IEEE Computer Society, Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence (WI'04), 2004.

[3]   Barceló, A. Juan, "Computational Intelligence in Archaeology", Idea Group Inc (IGI), 2008.

[4]   Wang, J., Meng, G., and Zheng, X., "The Attribute Reduce Based on Rough Sets and SAT Algorithm", IEEE Intelligent Information Technology Application, vol. 1, Shanghai, 2008.

[5]   Pawlak, Z., et al., Transactions on Rough Sets VII: Part II, vol. 4400. Berlin: Springer, 2007, pp. 372.

[6]   Bazan, J., et al., "Rough Set Algorithms in Classification Problem", Rough set methods and applications: new developments in knowledge discovery in information systems, Studies in Fuzziness and Soft Computing. Heidelberg, Germany: Physica-Verlag, 2000, pp. 49—88.

[7]   Wroblewski, J., "Finding Mininmal Reducts Using Genetic Algorithm," ICS Research Report 16/95. Warszawa, Polska : Warsaw University of Technology, 1995.

[8]   Bjorvand, A.T. and Komorowski, J., "Practical Applications of Genetic Algorithms for Efficient Reduct Computation," 15th IMACS World Congress on Scientific Computation, Modelling and Applied Mathematics, vol. 4. Wissenschaft & Technik Verlag, 2007, pp. 601—606.

[9]   Zhai, L.Y., Koo, L.P., and Fok, S.C., "Feature Extraction Using Rough Set Theory and Genetic Algorithms - An Application for The Simplification of Product Quality Evaluation". Journal Computers and Industrial Engineering Tarrytown, vol. 43. NY: Pergamon Press, Inc., 2002.

[10]   Skowron, A., et al., "RSES 2.2 User's Guide. Institute of Mathematics". Warsaw : Warsaw University, 2005.

[11]   Kennedy, J. and R.C., Eberhart., "Particle Swarm Optimization," Proceedings of IEEE International Conference of Neural Networks, pp. 1942—1948. Perth: IEEE, 1955.

[12]   Wang, X., et al., "Feature Selection Based on Rough Sets and Particel Swarm Optimization," New York : Elsevier Science Inc., vol. 28, 2007.

[13]   Yue, B., et al., "A New Rough Set Reduct Algorithm Based on Particle Swarm Optimization". Heidelberg: Springer Berlin, vol. 4527, 2007, pp. 397—406.

[14] Wang, X., et al., "Finding Minimal Rough Set Reducts with Particle Swarm Optimization," Heidelberg: Springer Berlin, vol. 3641, 2005, pp. 451—460.

[15] Bai, Q., "Analysis of Particle Swarm Optimization Algorithm," Computer and Information (CIS), vol. 3. pp. 180-184. CCSE, 2010.

[16] Chen, Yumin, Miao, Duoqian and Wang, Ruizhi, "A Rough Set Approach to Feature Selection Based on Ant Colony Optimization," Elsevier, vol. 31, 2010.

[17] Zeng, H., Huang, Y., and Zeng, X., "A New Approach of Attribute Reduction Based on Ant Colony Optimization." Tianjian: IEEE, 2009.

[18] Ke, L., Feng, Z., and Ren, Z., "An Efficient Ant Colony Optimization Approach to Attribute Reduction in Rough Set Theory," Pattern Recognition Letters, vol. 29. Elsevier, 2008.

[19] Ye, D., Chen, Z., and Liao, Jiankun, "A New Algorithm for Minimum Attribute Reduction Based on Binary Particle Swarm Optimization with Vaccination," Advances in Knowledge Discovery and Data Mining, vol. 4426. Berlin: Springer-Verlag, 2007, pp. 1029—1036.

[20] Pawlak, Z. and Skowron, A., "Rough Sets and Boolean Reasoning," Information Sciences, vol. 177. Elsevier, 2007, pp. 41—73.

[21] Shelokar, P.S., et al., "Particle Swarm and Ant Colony Algorithms Hybridized for Improved Continues Optimization," Applied Mathematics and Computation, vol. 188. Elsevier Inc., 2007, pp. 129—142.

[22] Lee, K.Y., and Park, J. B., "Application of Particle Swarm Optimization to Economic Dispatch Problem: Advantages and Disadvantages." IEEE: PSCE, 2006.

[23] Lin, W., Wu, Y., Mao, D., and Yu, Y., "Attribute Reduction of Rough Set Based on Particle Swarm Optimization with Immunity," IEEE Second International Conference on Genetic and Evoluationary Computing, 2008, pp. 14—17.

[24] Pratiwi, L., Yun-Huoy, Choo, and Muda, A.K., "An Empirical Study of Density and Distribution Functions for Ant Swarm Optimized Rough Reducts", Proc. Second International Conference (ICSECS'11), Communications in Computer and Information Science – Part 6, vol. 180, June 2011, pp. 590-604, Pahang.

[25] Yang, Q. and Wang, D., "An Improved Paricle Swarm Optimization." IEEE, 2009.

[26] Shi, Y. and Eberhart, R. C., "Empirical study of particle swarm optimization," Int. Congr. Evolutionary Computation, vol. 1. IEEE, 2001, pp. 101—106.

[27] Dorigo, M., Maniezzo, V., and Colorni, A., "The Ant System: Optimization by A Colony of Cooperating Agents", IEEE Transactions on Systems, Man, and Cybernetics - Part B, vol. 26, 1966, pp. 1—13.

[28] Kaveh, A. and Talataharib, S., "A Hybrid Particle Swarm and Ant Colony Optimization for Design Of Truss Structures," Asian Journal of Civil Engineering (Building and Housing), vol. 9. 2008, pp. 329—348.

[29] Jiao, W., and Liu, G.-b., "An Improved Particle Swarm Optimization Algorithm with Immunity," Second International Conference of Intelligent Computation Technology and Automation (ICICTA), IEEE Press, Oct. 2009, pp. 241-244, doi: 10.1109/ICICTA.2009.66.

[30] Afshinmanesh, F., Marandi, A., and Rahimi-Kian, A., "A Novel Binary Particle Swarm Optimization Method Using Artificial Immune System," The International Conference of EUROCON: Computer as a Tool, IEEE Press, Nov. 2005, pp. 217-220, doi: 10.1109/EURCON.2005.1629899.

[31] Pratiwi, L., Yun-Huoy, Choo, and Muda, A.K., "A Framework of Rough Reducts Optimization Based on PSO/ACO Hybridized Algorithms", 3rd Data Mining and Optimization (DMO'11), IEEE Press, June. 2011, pp. 153-159, doi: 10.1109/DMO.2011.5976520..