# An Evolutionary Quantum Behaved Particle Swarm Optimization for Mining Association Rules

K. Indira1, S. Kanmani2, R. Jagan3, G. Balaji3, F. Milton Joseph3
1Research Scholar, Department of Computer Science and Engineering
2Professor, Department of Information Technology
3Final Year UG Student, Department of Information Technology
Pondicherry Engineering College, Puducherry, India.
induharini@gmail.com1, kanmani@pec.edu2, jaganit@pec.edu3, bala@pec.edu3, itsmemilton@gmail.com3

**Abstract**- In data mining, association rule mining is a popular and well researched method for discovering interesting relations between variables in large databases, which are meaningful to the users and can generate strong rules on the basis of these frequent patterns, which are helpful in decision support system. Quantum Particle Swarm Optimization (QPSO) is one of the several methods for mining association rules. It combines the aspects of traditional PSO philosophy and quantum mechanics. However, preventing the occurrence of local optima and improving the convergence speed is still a tedious task. In this paper, an Evolutionary Quantum behaved Particle Swarm Optimization (EQPSO) is presented with improved computational efficiency and has proper convergence. The proposed work introduces local search techniques into QPSO using Modified Shuffled Frog Leaping Algorithm (MSFLA) and depicts a systematic parameter adaptation by developing an Evolutionary State Estimation (ESE) and an Elitist Learning Strategy (ELS). The EQPSO implementation has comprehensively been evaluated on 5 different datasets taken up from the UCI Irvine repository. The performance of EQPSO is compared with Basic QPSO and the experimental results shows that the proposed system outperforms the existing algorithm quite significantly.

**Keywords**- Association Rule Mining, Elitist, Evolutionary State, Memetic, Particle Swarm Optimization, Quantum Behavior, Self-Adaptive, Shuffled Frog Leaping.

## 1 INTRODUCTION

Data Mining, the analysis step of Knowledge Discovery in Databases, is the practice of examining large pre-existing databases in order to generate new information. However, continuous increase in the amount of data stored in databases and types of databases, the extraction of critical hidden information from these databases has become tedious. Several methods such as classification, clustering and association rules have been used to deduce interferences from large databases. Among these methods, association rule mining is the most widely used method.

Association rules (AR) are usually required to satisfy a user-specified minimum support and a user-specified minimum confidence at the same time [1]. The association rule generation can be split up into steps: firstly, to find all frequent itemsets in a database, minimum support value is applied. Second, the rules are formed making use of these frequent itemsets and the minimum confidence constraints.

Apriori algorithm is one of the first known association rule mining algorithm. It uses a level wise search, where k-itemsets are used to explore (k+1)-itemsets, to mine frequent itemsets from transactional database for Boolean association rules. In order to improve the efficiency, a major step forward was the introduction of compact data structure, referred to FP-tree or frequent pattern tree and its association mining algorithm FP-growth. However, support calculation is possible only if the entire dataset is added to the tree. Then, an alternative of Apriori Itemset Generation called Dynamic Itemset Counting was introduced. In this algorithm, once the transactions are read, the itemsets are dynamically added and deleted. It relies on the fact that for an itemset, all of its subsets must also be frequent, so only those itemsets whose subsets are all frequent can be examined.

Later, evolutionary algorithms like Genetic Algorithm (GA) [2], Particle Swarm Optimization (PSO) [3] and a variant of PSO namely Quantum Particle Swarm Optimization (QPSO) were introduced. These algorithm are population based search methods and it moves from one set of points (population) to another set of points in a single iteration with likely improvement using set of control operators. However, the fundamental problem in the existing system is its imbalanced local and global search and also while trying to reduce the convergence time it gets trapped in the local optimal region when solving multimodal problems. These weaknesses have restricted their wider applications. These appealing goals in association rule mining research has been overcome in the proposed Evolutionary Quantum behaved Particle Swarm Optimization (EQPSO) that performs a diversified search over the entire search space for better convergence.

The paper is organized as follows. Section 2 presents the methodology and the aspects of mining association rules. Section 3 explains the experimental results and section 4 gives the conclusion.

## 2 METHODOLOGY

### 2.1 Association Rule Mining

Association rule mining is a popular and well researched method for discovering frequent patterns, associations and correlations among items which are meaningful to the users and can generate strong rules on the basis of these frequent patterns, which help in decision support system. Typically, the relationship will be in the form of IF(X) THEN(Y), such that $X\cap Y = \emptyset$, where X part is called *Antecedent* and Y part is called *Consequent*. [4]

The two parameters that indicate the importance of association rules are *support* and *confidence*. The support implies frequency of occurrence of the rule in the whole database. Support of a rule is the ratio of the number of times the antecedent occurs to the total dataset size, and is given by

$$supportX=No.\ of\ transactions\ containing\ XTotal\ No.of\ transactions \tag{1}$$

Confidence means the strength of implication. Confidence of a rule is the ratio of the number of times the full rule occurs to the total dataset size, and is given by

$$confidenceX\rightarrow Y=support(\ XUY\ )support(X) \tag{2}$$

### 2.2 Mining AR based on QPSO

As per the experimental results of Van Den Bergh [5], Particle Swarm Optimization is not a global convergence-guaranteed optimization algorithm. Therefore, Sun *et* al. [6],[7] proposed a global convergence-guaranteed search algorithm called QPSO, whose performance is superior to the PSO.

As per traditional PSO, the state of a particle is depicted by its position vector xi and velocity vector vi, which defines the *trajectory* of the particle. The movement of the particle is along a defined trajectory as it follows Newtonian Mechanics. According to Heisenberg's Uncertainty Principle, the velocity and position of a quantum behaved particle cannot be determined simultaneously. Hence the term trajectory has no meaning in quantum mechanics. The dynamic behavior of the particle is widely divergent from that of the particle in classical PSO systems.

The particles move according to the following iterative equations [8],[9]:

$xt+1=p+\ \beta*mbest-xt*ln1u$, if $k\geq 0.5$
$xt+1=p-\ \beta*mbest-xt*ln1u$, if $k<0.5$

(3)

where β is called contraction-expansion coefficient [10] in the range (0,1), u and k are uniformly distributed random numbers in the range (0,1).

p is the local attractor to guarantee QPSO convergence
$p=(c1*pid+c2*pgd)/(c1+c2)$ (4)
where c1 and c2 are acceleration coefficients (usually c1=1.82 and c2=1.97) [11]

Mean best (mbest) of the population is defined as the mean of the best positions of all particles.

mbest=1Mi=1MPi=1Mi=1MPi1, 1Mi=1MPi2,….., 1Mi=1MPid

(5)

where i represents the index of the best particle among all the particles in the swarm.

The outline of basic Quantum Particle Swarm Optimizer is as follows:

Step1: Initialize the swarm
Step2: Evaluate the fitness of every particle in the swarm
Step3: Update local best (pbest)
Step4: Update global best (gbest)
Step5: Calculate Mean best position (mbest) using equation(5)
Step6: Update the particle position using equation(3)
Step7: Terminate if the condition is met else go to step2

### 2.3 Mining AR based on Adaptive QPSO (AQPSO)

QPSO is mainly conducted by four key parameters important for the speed, convergence and efficiency of the algorithm: the contraction-expansion coefficient (β), mean-best position (mbest) and two acceleration coefficients (c1 and c2). Contraction-expansion coefficient is a convergence factor used to balance between exploration and exploitation by using previous flying experience of the particles. Mean-best position is replaced with Weighted mean-best position to determine whether a particle is elitist or not. Acceleration parameters are typically two positive constants, called the cognitive c1 and social parameter c2.

The Adaptive QPSO algorithm consists of a systematic parameter adaptation scheme by developing an Evolutionary State Estimation (ESE) and an Elitist Learning Strategy (ELS).

### 2.3.1 Evolutionary State Estimation

During a QPSO process, the characteristics of population distribution not only vary with respect to the generation number but also with the evolutionary state. At an initial stage, the distribution of population is dispersive in nature as the particles may be scattered in various areas. However, the particles would cluster together and converge to a locally or globally optimal area.

An ESE approach is devised based on the search behavior and the characteristics of the distribution of the population of the QPSO. By calculating the mean distance of each particle to all other particles, the distribution information can be developed. During the convergence state, since the global best tends to be surrounded by the swarm, probably the mean distance between the global best particle to the other particles will be minimum. On the other hand, the mean distance between will be maximum in the jumping out state, since the global best is expected to be away from the crowding swarm. During every generation, the information about the distribution of population will be taken into account by the ESE approach as follows:

**Step 1:** The mean distance between the particle *i* to the other particles is given by-

*di=1N-1j=1,j≠iNk=1D(xik-xjk)2*

(6)

where N and A are the population size and the number of attributes respectively.

**Step 2:** The Evolutionary Factor (f) is given by-

$$f=dg-dmindmax-dmin\in[0,1] \quad (7)$$

where dg is the di of the globally best particle, dmax is the maximum di value and dmin is the minimum di value

**Step 3:** The Evolutionary Factor (f) is used to identify the evolutionary state that the particle undergoes in each generation. Based on its value, its entire range can be classified into four sets S1, S2, S3 and S4 representing the states Exploration, Exploitation, Convergence and Jumping Out respectively.

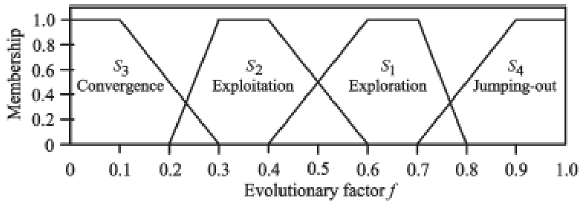

Fig. 1. Fuzzy membership functions for the four evolutionary states [15]

**Step 4: Adaptation of Acceleration Factors**

Based on the particle’s current evolutionary state, the acceleration factors c1 and c2 should be dynamically adapted. Parameter c1 represents the “self-cognition” that helps in exploring local niches by pulling the particle to its own historical best position and it also maintains the diversity of the swarm. Parameter c2 represents the “social influence” that helps in faster convergence by pushing the swarm to converge to the current globally best position.

In this paper, the acceleration factors c1 and c2 are initialized to 1.82 and 1.97 respectively [11], which satisfied the convergence condition of the particles ω=(c1+c2)/2 -1. As c1 < c2, the particles will converge faster to the global optimal position than the local optimal position of each particle. The adaptation control strategy is mentioned in the below table I

Table 1
Adaptation Control Strategy of c1 and c2

| State | Strategy | c1 | c2 |
|---|---|---|---|
| Exploration | Strategy 1 | Increase | Decrease |
| Exploitation | Strategy 2 | Increase slightly | Decrease slightly |
| Convergence | Strategy 3 | Increase slightly | Increase slightly |
| Jumping Out | Strategy 4 | Decrease | Increase |

**Strategy 1:** In this state, the particles must explore as many optima as possible. Increasing c1 helps the particle to search for the maximum number of possible solutions, instead of crowding at the same region. The particle searches for its historical best position by decreasing c2, instead of trying to find new solution.

**Strategy 2:** It is important to the particles to make use of the local information and group towards possible local optimal niches by making use of its historical best position. Slowly increasing c1 can emphasize the search around pBesti. Slowly decreasing c2 avoids the deception of a local optimum.

**Strategy 3:** It is important to find the global optimal region by the swarm. Slowly increasing c2 guides the other particles to this region. For a faster convergence of the particle, c2 is decreased to avoid possible local searches.

**Strategy 4:** During the convergence state, as soon as the particle finds itself crowded in the local optimal region, it jumps out to a new optimal region. To obtain this goal, a small value of c1 with relatively larger value of c2 is maintained.

### 2.3.2 Adaption of Contraction-Expansion Coefficient

In order to balance between exploration and exploitation, the parameter contraction-expansion coefficient (β) is used. The path of a particle depends upon the local best and global best positions. Moreover, contraction-expansion coefficient is an important convergence factor, the larger the β value, the faster the convergence of QPSO. Based on the results of stochastic simulations, the performance of QPSO can be relatively improved by linearly decreasing the value of β from 1.0 at the beginning of the search to 0.5 at the end of the search to balance between exploration and exploitation. [12]

### 2.3.3 Elitist Learning Strategy

To determine whether a quantum particle is elitist or not is an important strategy in QPSO. This can be done by dynamically adjusting the value of mean-best (mbest) position. The elitism is usually associated with the particle's fitness value, in any evolutionary algorithm, the greater the fitness, the more important the particle is. Based on the fitness value of the particle, in descending order, a rank can be assigned to the particle. According to their rank, the particles are assigned with a weight coefficient ($\alpha$i). The value of weight coefficient will be higher, if the particle is nearer to the best solution.

Weighted Mean Best Position (or) Mainstream Though Point is defined as the mean of product of local best and weight coefficient of every particle in the swarm and it can be evaluated using the formula:

mt=(1Mi=1Mα1P1,1Mi=1Mα2P2,.....1Mi=1MαnPn)

(8)

where M is the population size and $\alpha$i is the weight coefficient (decreasing linearly from 1.5 to 0.5) [13]

The steps involved in Adaptive Quantum Particle Swarm Optimizer are as follows:

```
Step 1: Initialize the population
Step 2: Set c1 to 1.82 and c2 to 1.97
Step 3: Evaluate the fitness of each particle
Step 4: Update local best (pbest)
Step 5: Update global best (gbest)
Step 6: Calculate mean best position using equation (5)
Step 7: Compute evolutionary factor using the mean distance between every particle using equations (6)&(7)
Step 8: Estimate the evolutionary state of every particle
Step 9: Adaptively control the algorithmic parameter as mention in ESE
Step 10: Update the particle position using equation (3)
Step 11: Perform ELS once the particles get into convergence state
Step 12: Terminate if the condition is met else go to step3
```

## 2.4 Mining AR based on Memetic QPSO (MQPSO)

MQPSO introduces local search techniques into basic QPSO. It performs the search more precisely around possible solutions of the problem at hand. For the improvement of local search ability, a Modified Shuffled Frog Leaping Algorithm (MSFLA) can be implemented.

### 2.4.1 Modified Shuffled Frog Leaping Algorithm (MSFLA)

MSFLA partitions the entire swarm into sub-swarms. Each particle in the swarm is said to a *frog* and a sub-swarm is said to be a *memeplex*. The culture (idea) of frogs in each memeplex is treated to be different and each performs their own local search. Every frog has its own ideas and is also capable of influencing the idea of other frogs within a memeplex through a shuffling process by undergoing a series of memetic evolution. This process is continued until a defined convergence criteria is satisfied.

The initial population consists of a set of frogs formed randomly. In the descending order of the fitness value, the frogs are subjected to sorting. The sorted population is partitioned into m memeplex, in such a way that the first memeplex holds the first frog, the second memeplex holds the second frog, mth memeplex holds the mth frog, then (m+1)th memeplex holds the (m+1)th frog and this continues until all frogs are accommodated. By this way, every memeplex would contain n frogs. Based on the fitness value, the best and worst frog are identified as Xb and Xw respectively, within every memeplex. The global best frog with respect to the entire swarm is identified as Xg. At the end of every iteration, the frog with worst fitness is identified within every memeplex and is subjected to a position updation. [14]

The position updation of worst frog in every memeplex is calculated using-

$$Dj=rand*Xb-Xw+1Ni=1Nrand*(Xi-Xw) \tag{9}$$

$$\text{New Position }Xw=\text{Current Position }Xw+Dj \tag{10}$$

where rand is a random integer between 0 and 1, Xb is the position of best frog in the memeplex, Xw is the position of worst frog in the memeplex, Xi is the position of ith frog in the memeplex.

The steps involved in Memetic Quantum Particle Swarm Optimizer are as follows:

Step 1: Initialize the population

Step 2: Set c1 to 1.82 and c2 to 1.97

Step 3: Evaluate fitness of the individual particle (update pbest)

Step 4: Sorting of population in descending order in terms of fitness value

Step 5: Distribution of frog into M memeplexes

Step 6: Update local best, global best and mean best positions (separately for the entire swarm and for every memeplex)

Step 7: Update particle position (based on the corresponding memeplex's pbest, gbest and mbest) using equation (3)

Step 8: Iterative update of worst frog in each memeplexes using equations (9) & (10)

Step 9: Combining all frogs to form a new population

Step 10: Terminate if the condition is met else go to step 2

### 2.5 Mining AR based on EQPSO (AQPSO+MQPSO)

The steps involved in Evolutionary Quantum Behaved Particle Swarm Optimizer are as follows:

Step 1: Initialize the population
Step 2: Set c1 to 1.82 and c2 to 1.97
Step 3: Evaluate the fitness of each particle
Step 4: Sorting of population in descending order in terms of fitness value
Step 5: Distribution of frog into M memeplexes

```
Step 6: Compute evolutionary factor using the mean distance between every particle using equations (6)&(7)
Step 7: Estimate the evolutionary state of every particle
Step 8: Adaptively control the algorithmic parameter as mention in ESE
Step 9: Update local best, global best and mean best positions (separately for the entire swarm and for every memeplex)
Step10: Update the particle position (based on the corresponding memeplex's pbest, gbest and mbest) using equation (3)
Step11: Iterative update of worst frog in each memeplex using equation (9) & (10)
Step12: Perform ELS once the particles get into convergence state
Step13: Combine all frogs to form a new population
Step14: Terminate if the condition is met else go to step3
```

## 3 RESULTS AND DISCUSSION

The datasets namely Lenses, Car Evaluation, Post-Operative Patient Care, Haberman's Survival and Zoo used for the comparison of QPSO, AQPSO, MQPSO and EQPSO were taken up from the UCI Irvine repository [16]. Table 2 gives the details about the datasets used. To provide the uniformity, the existing and proposed algorithms were coded in Java. The efficiency of the algorithms are compared utilizing the output parameters mentioned in the below section.

Table 2
Dataset Description

| Dataset | Attributes | Swarm Size | Classes |
|---|---|---|---|
| Lenses | 4 | 24 | 3 |
| Car Evaluation | 6 | 700 | 4 |
| Haberman's Survival | 3 | 310 | 2 |
| Post-operative Patient Care | 8 | 87 | 3 |
| Zoo | 16 | 101 | 7 |

### 3.1 Predictive Accuracy (PA)

The effectiveness of the rules mined is measured in terms of predictive accuracy. The mined rules must have high predictive accuracy and is given by

$$\textit{Predictive accuracy}= |X \;\&\; Y||X| \qquad (11)$$

Where |X&Y| is the number of records that satisfy both antecedent and consequent, |X| is the number of rules satisfying the antecedent.

The predictive accuracy of the association rule mined by the proposed EQPSO methodology is compared with the basic QPSO, AQPSO and MQPSO is plotted in fig2.

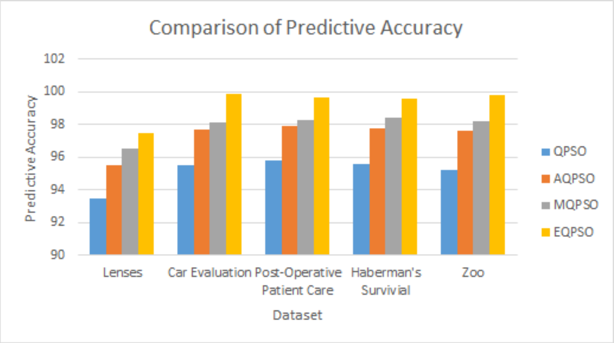

Fig. 2. Comparison of Predictive Accuracy

The predictive accuracy achieved by EQPSO is better compared to basic QPSO, AQPSO and MQPSO for all five datasets taken up for study.

### 3.2 Number of rules mined

This denotes the count of the rules generated above a certain PA. The number of rules mined by the proposed EQPSO methodology is compared with the basic QPSO, AQPSO and MQPSO is presented in table 3.

Table 3
Comparison of number of rules mined

| Datasets | QPSO | AQPSO | MQPSO | EQPSO |
|---|---|---|---|---|
| Lenses (24) | 18 | 21 | 19 | 21 |
| Car Evaluation (700) | 685 | 691 | 685 | 695 |
| Haberman's Survival (310) | 301 | 303 | 296 | 304 |
| Post-operative Patient Care (87) | 79 | 82 | 76 | 82 |
| Zoo (101) | 92 | 95 | 87 | 98 |

As a result, the proposed EQPSO generates more numbers of rules mined when compared to basic QPSO, AQPSO and MQPSO methodologies.

### 3.3 Laplace

Laplace is a confidence estimator that measure the interestingness of the generated rules. Rules with high confidence may occur by change, such spurious rules are detected.

$lapl(X\rightarrow Y)=supX\cup Y+1supX+2$

(12)

### 3.4 Conviction

Conviction measure the weakness of confidence. Conviction is infinite for logical implications (confidence 1) and is 1 if X and Y are independent

$$convX\rightarrow Y=1\text{-}supY1\text{-}conf(X\rightarrow Y) \qquad (13)$$

The interestingness of rules mined is also measured through Laplace and Conviction, and is summarized in table 4.

Table 4
Measures of Association Rule Mining with EQPSO

| Datasets | Laplace | Conviction |
|---|---|---|
| Lenses | 0.624 | infinity |
| Haberman's Survival | 0.6 | infinity |
| Car Evaluation | 0.627 | infinity |
| Post-operative Patient Care | 0.61 | infinity |
| Zoo | 0.589 | infinity |

## 4 CONCLUSION

The EQPSO algorithm is superior to the classical QPSO mainly in three aspects. Firstly, for a wider searching space of the algorithm and in order to generate different state of the particles, an uncertain system known as quantum theory can be deployed. Secondly, the introduction of the parameter local attractor in QPSO is a benchmark. The parameter is designed in such a way that c1 represents self-cognition and c2 represents social influence, hence a proper convergence can be attained when there is an equilibrium between these two coefficients. Lastly, the balancing between every memeplex is achieved by the introduction of mean-best position in every memeplex. In classical QPSO, the particle can fall into a local optima very easily in few iterations. In EQPSO, the average error is lowered by the introduction of mbest, so that without considering its colleague particles, a particle cannot converge fast, which makes the frequency lower than QPSO. Hence the performance of the algorithm is significantly improved by EQPSO. In future, EQPSO can emphasize an extensive study on the applications that involves more complex practical optimization problems, to completely estimate the performance and investigate the attributed associated to EQPSO.

## REFERENCES

[1]. Min Wang, Qin Zou, Caihui Liu, *"Multidimensional Association Rule Mining Based on Adaptive Genetic Algorithm"*, IEEE International Conference on Uncertainity Reasoning and Knowledge Engineering, pp.150-153, 2011.

[2]. K.Indira, S.Kanmani, D.Gaurav Sethia, S.Kumaran, J.Prabhakar, *"Rule Acquisition in Data Mining Using a Self Adaptive Genetic Algorithm"*, Springer International Conference on Trends in Computer Science, Engineering and Information Technology CCSEIT 2011, CCIS 204, pp.171-178, 2011.

[3]. K.Indira, S.Kanmani, P.Prashanth, V.Harish, Konda Ramcharan Teja, R.Jeeva, *"Population Based Search Methods in Mining Association Rules"*, Third International Conference on Advances in Communication, Network and Computing – CNC 2012, LNCS 2012, pp.255-261.

[4]. S.Dehuri, S.Patnaik, A.Ghosh, R.Mall, *"Application of Elitist Multi-objective Genetic Algorithm for Classification Rule Generation"*, Applied Soft Computing pp.477-487, 2008.

[5]. Van Den Bergh, *"A Study of Particle Swarm Optimization Particle"*, A.P.Engelbrecht/Informational Sciences 176(2006), pp.937-971.

[6]. J. Sun, B. Feng, and W. Xu, *"Particle Swarm Optimization with Particles having Quantum Behavior"*, Proceedings on IEEE Congress on Evolutionary Computation, 2004, pp.325-331.

[7]. J. Sun, B. Fend and W. Xu, *"A Global Search Strategy of Quantum behaved Particle Swarm Optimization"*, Proceedings on IEEE Conference on Cybernetics and Intelligent Systems, 2004, pp.111-116.

[8]. Shoaf J. and Foster A., *"The Efficient Set GA for Stock Portfolios"*, in Proceedings of IEEE International Conference on Computational Intelligence, AK, pp.354-359, 1998.

[9]. Sun J., *"Particle Swarm Optimization with Particles Having Quantum Behavior"*, in Proceedings of Congress on Evolutionary Computation, vol.1, pp.325-331, 2004.

[10]. Sun J, Xu W, Feng B. Adaptive parameter control for quantum-behaved particle swarm optimization on individual level. In: Proceedings of IEEE international conference on systems, man and cybernetics, Big Island (HI, USA), 2005. p. 3049–54.

[11]. Yourui Huang, Liguo Qu and Chaoli Tang, *"Optimal Converage Scheme Based on QPSO in Wireless Sensor Networks"*, Journal of Networks, Volume7, No.9, September 2012.

[12]. Sun J., Xu W., and Liu J., *"Parameter Selection of Quantum-behaved Particle Swarm Optimization"*, in Proceedings of 1st International Conference on Advances in Natural Computation, Berlin, Vol.3, pp.543-552, 2005.

[13]. Maolong Xi., Jun Sun, and Wenbo Xu, *"An Improved Quantum-behaved Particle Swarm Optimization with Weighted Mean Best Position"*, IEEE Conference on Applied Mathematics and Computation 205 (2008), pp.751-759.

[14]. Jiuzhong Zhang, Xueming Ding, *"A Multi-Swarm Self Adaptive and Cooperative Particle Swarm Optimization"*, Engineering Applications of Artificial Intelligence 24, 2011, pp.958-967.

[15]. Zhan, Z-H. and Zhang, J. and Li, Y. and Chung, H.S-H, Adaptive Particle Swarm Optimization, IEEE Transactions on Systems Man, and Cybernetics – Part B: Cybernetics , 39(6), 2009, pp.1362-1381.

[16]. UCI C.J.Merz and P.M.Murphy, *"UCI Repository of Machine Learning Databases"*, University of California Irvine, Department of Information and Computer Science, http://kdd.ics.uci.edu, 1996.