TELKOMNIKA, Vol.13, No.3, September 2015, pp. 1029~1036
ISSN: 1693-6930, accredited A by DIKTI, Decree No: 58/DIKTI/Kep/2013
DOI: 10.12928/TELKOMNIKA.v13i3.1806

Application of Ant Colony Algorithm in Multi-objective Optimization Problems

Juan Li*, Xianghong Tian
School of Computer Engineering,Jinling Institute of Technology, Nanjing 211169,Jiangsu,China
*Corresponding author, e-mail: iamlj6@163.com

Abstract

In actual application and scientific research, multi-objective optimization is an extremely important research subject. In reality, many issues are related to the simultaneous optimization under multi-objective conditions. The research subject of multi-objective optimization is getting increasing attention. In order to better solve some nonlinear, restricted complex multi-objective optimization problems, based on the current studies of multi-objective optimization and evolutionary algorithm, this paper applies the ant colony algorithm to multi-objective optimization, and proves through experiments that multi-objective ant colony algorithm can converge the real Pareto front of the standard test function more quickly and accurately, and can also maintain the distributivity of the better solution.

Keywords*: ant colony algorithm, multi-objective optimization, pareto optimal set*

1. Introduction

Originated from the design, planing scale, project adjustment and other essential decision issues of many complex systems in real life, multi-objective optimization has always been one of the important subjects of engineering practice and scientific research. In the aspects of computing science, decision science and operation science, there once appeared much certainty, randomization methods specialized for multi-objective optimization [1]. In recent year, along with the improvement in the calculating speed and capability of calculation devices, the application of intelligent evolutionary algorithms in multi-objective optimization, as with genetic algorithm, genetic planing and genetic program designing, etc., has gained wide confirmation, which is mainly because these evolutionary algorithms process intelligence features of self-adaptivity, self-directed learning and self-organizing [2].

In terms of multi-objective optimization, usually there are conflicts and restrains among different targets of the optimization problem. Therefore, in order to maintain balance among these targets, the research aim of the algorithm is to try best to locate the optimal set near the actual Pareto front. Considering the followed decision step, the solutions in the solution set should be distributed as evenly as possible to increase the diversity of the possible solution. Considering the effect of actual application, the time period of searching the Pareto solution set should be as short as possible [3]. Most of the current studies adopt genetic algorithm to explore multi-objective optimization, and the amount of studies adopting emerging intelligence algorithm, such as ant colony algorithm, to explore multi-objective optimization is quite limited. With characteristics of implicit parallelism and intelligence, ant colony algorithm is quite suitable for optimization. It can be seen from the current research results that ant colony algorithm has good performance. It is proved by practice that the application of ant colony algorithm in solving single-objective problems is very successful [4]. However, there are still a lot of problems to solve in the application of ant colony algorithm in multi-objective optimization. Fields worthy of exploration include how to select the initial ant colony, how to construct Pareto optimal solution set, how to set the parameters of any colony algorithm, how to conduct simulation experiment and the verification of related theories, etc[5].

This paper first explained the basic principles of multi-objective problems and ant colony algorithm in detail, based on which, it provided the complete procedure of multi-objective ant colony optimization, and with the simulation, test and analysis of the standard test function, and

Received March 28, 2014; Revised June 12, 2015; Accepted June 28, 2015

the evaluation of performance measurement function, it proved the excellent performance and advantage of ant colony algorithm in aspects such as uniformity of solution distribution, etc.

2. Introduction to Multi-objective Optimization

Most of the engineering and science problems are multi-objective optimization problem (MOP), which requires simultaneous optimization of several objects. However, usually these objects are contradictory. That is, the improvement of one object might worsen other object. It is often impossible to optimize all the objects. Therefore, one can only try best to coordinate each object to optimize all the objects, moreover, the optimal of such problems often consists of the Pareto optimal with large, even endless amount [6].

Under restrain conditions, multi-objective optimization is also called multi-objective planing, which can be briefly described as: searching for a vector set consists of decision variables, which can both meet the assigned restraint, and optimize each sub-object function. In which, the sub-object function is the mathematical description of performance standard evaluation. Generally, these performance indexes are contradictory. Therefore, “multi-objective optimization” is to seek for a solution that makes the performance indexes represented by all the sub-object functions are relatively good solutions acceptable for decision makers. The mathematical description of performance standard evaluation is:

$$\begin{cases} \min/\max\ f(x) \\ s.t. g_j(X) \le 0,\ j = 1, 2, \cdots, m \end{cases} \tag{1}$$

In the equation, $f(x) = [f_1(x), \cdots, f_N(x)]^T$ refers to object function vector, in which, $N \ge 2$ is object function sum, $g(x) = [g_1(x), \cdots, g_m(x)]^T$ is m restraints, x is D dimension decision variables [7].

MO problems require the optimization of a set of object function. Its solution is not a single dot, but the set of a group of dots, which is called Pareto optimal set.

Pareto optimal set cannot be controlled by any solutions in the possible solution set. Suppose x^* is a possible solution, if x^* is the Pareto optimal instead of the inferior solution, when and only when there is no X (decision variable in the feasible region) make $x \succ x^*$.

The set consists of all the non-inferior solutions is called the Pareto optimal set of the MOP, and is also called the non-inferior solution set or valid solution set, defined as $P = \{x \in X \,|\, \neg\exists x' \in X, x' \succ x\}$.

The curved surface where the Pareto optimal is located is called the Pareto front, A, B and C as shown in Figure 1 are all the Pareto optimal, and the curved surface where they are located is the Pareto front.

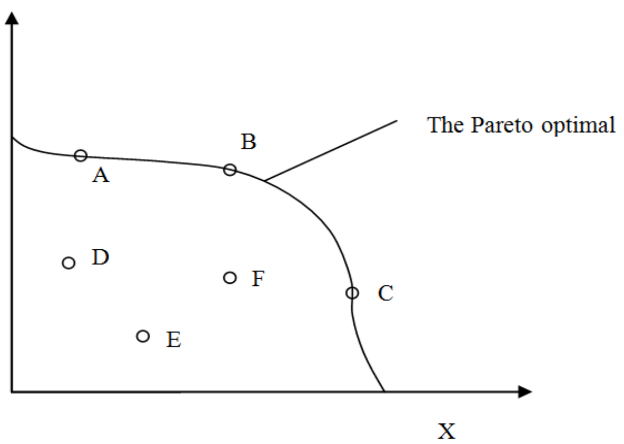


Figure 1. The pareto optimal of multi-objective optimization problem

The set gained through multi-objective optimization is the Pareto optimal set. However, considering the reality, it is necessary to combine the decision maker's subjective requirements for the optimized object with the calculating process to select the solution that best meets the decision makers requirements from the non-inferior solution set, that is, the "preferred solution" [8].

3. Basic Principle of Ant Colony Algorithm

Ant colony algorithm (ACA) is the simulation algorithm put forward through simulating the behavior of ant searching for food. With massive careful observation, bionicists find that ants communicate with each other through the material called pheromone. During the movement, ants can leave this material on the road it passed, and they can also perceive this material, thus to guide their direction. Therefore, the community behavior of a large amount of ants has the phenomenon of positive feedback: the more a tour being passed by the ants, the bigger the probability of other ants choosing this tour. The basic principle of ACA is: simulating the real ant colony cooperation process based on the study of food searching behavior of real ant colony. The algorithm finds the shortest tour to achieve the optimization through constructing the solution tour together by several ants, and leaving and exchanging pheromone on the solution tour to improve the quality f the solution [9].

3.1. Ant System

Ant system is the earliest ant colony algorithm. It approximate searching process is as follows:

At the initialization stage, m ants are randomly put in the city, the initial value of the pheromones on each tour are same, suppose $\tau_{ij}(0)=\tau_0$ as the pheromone initial value , then $\tau_0 = m/L_m$, L_m refers to the distance of the tour constructed by nearest neighbor heuristics. Then, ant $k(k=1,2,\cdots m)$ selects the city as the next transfer destination according to random ratio. The selection probability is:

$$p_{ij}^k(t)=\begin{cases}\dfrac{[\tau_{ij}(t)]^\alpha[\eta_{ij}(t)]^\beta}{\sum\limits_{s\in allowed_k}[\tau_{ij}(t)]^\alpha[\eta_{ij}(t)]^\beta}, j\in allowed_k\\ 0, \text{otherwise}\end{cases}$$

In which, τ_{ij} is the pheromone on (i,j), $\eta_{ij}=1/d_{ij}$ is the heuristic factor of the city i - city j transfer, $allowed_k$ is the next city set that ant k are allowed to visit.

In order to prevent the ant from visiting the already visited city, it adopts tabu list $tabu_k$ to record the cities that have already been visited by ant k. When passing t, all the ants finish the first circulation. Calculate the distance of the tour visited by each ant, and reserve the shortest distance, meanwhile, update the pheromone on each side. The first is the volatilization of the pheromone. The second is the ants releasing pheromone on the sides they are passing through, the equation is as follows:

$\tau_{ij}=(1-\rho)\tau_{ij}$, in which ρ is the pheromone volatilization coefficient, and $0<\rho\le 1$.

$\tau_{ij}=\tau_{ij}+\sum_{k=1}^{m}\Delta\tau_{ij}^k$, in which, $\Delta\tau_{ij}^k$ is the pheromone released on the side by the k numberant, defined as:

$$\Delta t_{ij}^k=\begin{cases}1/d_{ij}, \text{if side}(i,j) \text{ is on tour } T^k\\ 0, \text{ otherwise}\end{cases} \quad (2)$$

According to(2), the smaller the tour distance d_{ij} constructed by the ant , the more pheromone gained from each side on the tour, and it is more likely to be selected by other ants in future iteration.

After completing the first circulation, empty the tabu list, turn back to the initial city, and prepare for the next circulation. Massive simulation experiments shows that when solving small-scale TSP problems, the performance of the ant system is not bad. It can quickly find the optimized solution. However, along with the increase of the scale, the performance of AS algorithm weakens seriously, and is easy to stagnate. Therefore, a large amount of improved algorithm for its drawbacks appeared [10].

3.2. Elite Ant System

Elite ant system first put forward by Dorigo, et. al. is the improvement of the basic AS algorithm. Its concept is to provide extra pheromone for the optimized tour after each circulation. The ant that finds this solution is called the elite ant.

This best-so-far tour is T^{bs}. The extra enhancement for T^{bs} is gained through increasing the pheromone with the amount of e / L^{bs} of each side on the T^{bs}. In which, e is a parameter, which decides the weight of the T^{bs}, L^{bs} is the length of T^{bs}. Then, the updated equation of pheromone is:

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \sum_{k=1}^{m} \Delta\tau_{ij}^{k}(t) + e\Delta\tau_{ij}^{bs}(t) \tag{3}$$

In which, the definition method of $\Delta\tau_{ij}^{k}(t)$ is the same as before, the definition of $\Delta\tau_{ij}^{bs}(t)$ is:

$$\Delta\tau_{ij}^{bs}(t) = \begin{cases} 1/L^{bs}, & if\,(i,j) \in T^{bs} \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

The results of Dorigo, et. al.'s paper shows that using the elite strategy and selecting a proper e can not only make the AS algorithm gain a better solution, but also reduce the iteration amount [11].

3.3. Maximum-Minimum Ant System

The maximum-minimum ant system(MMAS)is the one of the best ACO algorithms for solvingTSP problems so far. On the basis of AS, MMAS mainly conducts the following improvement: (1) Prevent the algorithm from converging to the local optimal too early, it limited the possible exohormone concentration of each tour within $[\tau_{\min}, \tau_{\max}]$, values surpass this range will be forcefully set as $\tau_{\min}$ or $\tau_{\max}$, which can effectively avoid the information volume of certain tour being too larger than that of other tours, thus to avoid all the ants gathering to the same tour. (2) Emphasize the utilization of the optimized solution. Aftereachiteration, only the information on the tour where the optimized solution is located will be updated, which enables to better use the former information. (3) The initial value of the pheromone is set as the upper limit of its range. At the beginning of the algorithm, when ρ is smaller, the algorithm is more capable of finding the batter solution. After all the ants completing one iteration, update all the information on the tour according to Equation (5):

$$\tau_{ij}(t+1) = (1-\rho)\cdot\tau_{ij}(t) + \Delta\tau_{ij}^{best}(t), \rho \in (0,1) \tag{5}$$

$$\Delta\tau_{ij}^{best} = \begin{cases} \dfrac{1}{L^{best}}, & \text{if side}(i,j) \text{ is included in the optimized tour} \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

The allowed updated tour can be the global optimized solution, or the optimized solution of this iteration. The practice proved that gradual increase of the usage frequency of the global optimized solution enables the algorithm to perform better [12].

3.4. Ant Colony Optimization Based on Rank

Ant system based on rank (ASRANK) is the improvement of AS algorithm. Its concept is: after every iteration, the tour passed by the ants will be ranked from small to large, that is, $L^1(t) \le L^2(t) \le \cdots L^m(t)$. The tour will be weighted according to its length, the shorter the length, the bigger the weight [13]. The weight of the global optimized solution is w, the weight of the number optimized solution is $\max\{0, w-r\}$, and then the pheromone update rule of ASRANK is:

$$\tau_{ij}(t+1) = (1-\rho)\cdot\tau_{ij}(t) + \sum_{r=1}^{w-1}(w-r)\cdot\Delta\tau_{ij}^{r}(t) + w\cdot\Delta\tau_{ij}^{gb}(t), \rho \in (0,1) \quad (7)$$

$$\text{In which,} \Delta\tau_{ij}^{r}(t) = 1/L^{r}(t), \Delta\tau_{ij}^{gb}(t) = 1/L^{gb}$$

3.5. Any Colony System

Any colony system(ACS)is the improved ant colony algorithm put forward by Dorigo, et. al. There are three differences between it and AS: (1) It adopts the different tour selection rule, which can better use the searching experience accumulated by the ant. (2) The volatilization and resealing of pheromone will be only conducted on the side of the best-so-far tour. That is, after every iteration, only the best ant so far is allowed to release pheromone. (3) Except for the global pheromone update rule, it also adopts the local pheromone update rule.

In ACS, ant k located in city i selects city j as the next city to be visited according to the pseudo random ratio rule. The tour selection rule is provided by the following equation:

$$j = \begin{cases} \arg\max\left\{\tau_{il}[\eta_{il}]^{\beta}\right\}, & \text{if} \quad q \le q_0 \\ J, & \text{otherwise} \end{cases} \quad (8)$$

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{\left[\tau_{ij}(t)\right]^{\alpha}\left[\eta_{ij}(t)\right]^{\beta}}{\sum\limits_{s \subset allowed_k}\left[\tau_{ij}(t)\right]^{\alpha}\left[\eta_{ij}(t)\right]^{\beta}} & if \quad j \in allowed_k \\ 0 & else \end{cases} \quad (9)$$

In which, q is a random variable evenly distributed in $[0,1]$, $q_0 (0 \le q_0 \le 1)$ is a parameter. J is a random variable generated by the probability distribution provided by Equation (9) (in which, $\alpha = 1$).

The ACS global pheromone update rule is:

$$\tau_{ij} = (1-\rho)\tau_{ij} + \rho\Delta\tau_{ij}^{bs}, \forall(i,j) \in T^{bs} \quad (10)$$

$$\Delta\tau_{ij}^{bs} = 1/C^{bs} \quad (11)$$

The ACS local pheromone update rule defines:

During the process of tour construction, whenever the ant passes a side (i, j), it will immediately use this rule to update the pheromone on the side:

$$\tau_{ij} = (1-\rho)\tau_{ij} + \xi\tau_0 \quad (12)$$

In which, ξ and τ_0 are two parameters, ξ meets $0 < \xi < 1$, τ_0 is the initial value of the pheromone amount. The function of local update is that whenever the ant passesside (i, j), the pheromone τ_{ij} on this side will reduce, thus to reduce the probability of other ants selecting this side [14].

4. The Design of Multi-objective Ant Colony Optimization

The sub-objects of the MOP are contradictory. The improvement of one sub-object might weaken the performance of another object or other objects. In other words, it is impossible to optimize several sub-objects at the same time. Instead, it can only coordinate and compromise among them. The realization procedure of multi-objective ant colony optimization is all follows [15]:

(1) Even the randomly generated initial ant colony pop with a scale of N, calculate each ant objective function $f_i(x), i=1,2,\cdots k$, and the restrain function $e_j, j=1,2=\cdots,J$ in the pop.

(2) Separate the non-feasible solution set $(X_{imp}=\{x\in POP \mid e(x)>0\})$ from the feasible solution set $(X_f=\{x\in POP \mid e(x)\le 0\})$.

(3) Iterate the feasible solution set X_f in the initial solution.

(4) Initialize the external set BP, its initial value is the non-control solution in all the feasible solutions of pop. That is, $X_f=\{x\in POP \mid e(x)\le 0\}, BP=\{x\in X_f \mid \exists x'\in X_{f'},\ X'<X\}$.

(5) Set the iteration time $N_c=0$.

(6) make $i=1$.

(7) A random number p within the range of [0,1] is generated, compare it with parameter p_0, p_0 is a parameter within the range of [0,1]. When $p\le p_0$, let the current ant i to optimize based on the guidance of the global optimized experience, when $p>p_0$, let ant i to optimize through pheromone exchange. It can be seen that, the large the p_0, the large the probability of adopting the global optimized experience.

(8) Move ant i within its activity range, and add a random perturbation ϕ on its final position. Reevaluation ant i, calculate its objective function and restrain function.

(9) Update the optimized experience set BP, if ant i is feasible solution, and is non-control to set BP. Then include ant i into set BP, and delete the solutions in the set that are controlled by i.

(10) $i=i+1$, if $i\le N$, then turn to step (7).

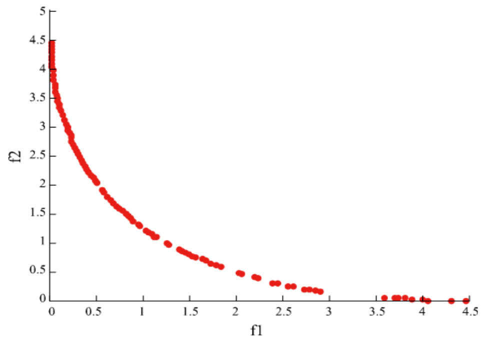
(11) $t=t+1$, if t is smaller than the maximum iteration time, turn to step (6), otherwise, end the algorithm.

5. Evaluation of the Performance of the Multi-objective Ant Colony Optimization

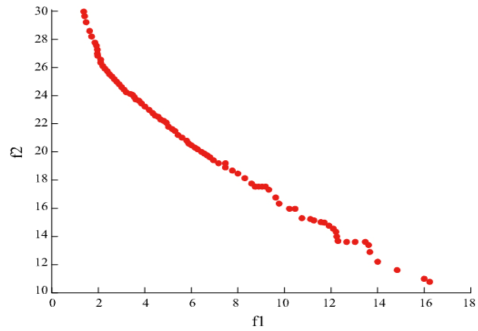
Table 1. 4 Standard Test Functions

Problem	Dimension	Range	Objective function	Convergence
SCH	1	[-103,103]	$f_1(x)=x^2$ $f_2(x)=(x-2)^2$	Convergent
POL	3	$[-\pi,\pi]$	$f_1(x)=[1+(A_1-B_1)^2+(A_2-B_2)^2]$ $f_2(x)=[(x_1+3)^2+(x_2+1)^2]$ $A_1=0.5\sin 1-\cos 1+2\sin 2-1.5\cos 2$ $A_2=0.5\sin x1-2\cos x1+\sin x2-1.5\cos x2$ $B_2=1.5\sin x1-\cos x1+2\sin 2-0.5\cos x2$	Non-convergent, non-continuous
ZDT1	30	[0,1]	$f_1(x)=x_1$ $f_2(x)=g(x)[1-\sqrt{x_1/g(x)}]$ $g(x)=1+9(\sum_{i=2}^{n}x_i)/(n-1)$	Convergent
ZDT3	30	[0,1]	$f_1(x)=1-\exp(-4x_1)\sin^6(4\pi x_1)$ $f_2(x)=g(x)[1-(f_1(x)/g(x))^2]$ $g(x)=1+9[(\sum_{i=2}^{n}x_i)/(n-1)]^{0.25}$	Convergent, continuous

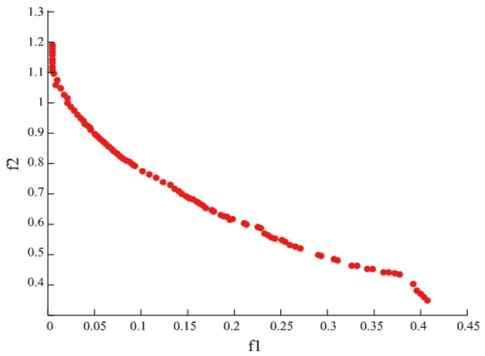
In order to test the performance of the algorithm, this paper selected 4 classical test functions to test the improved multi-objective ant colony optimization, which includes the most representative issues in ZDT test function. The selected test functions are shown in Figure 1. The simulation results are shown in Figure 2. All the test functions gained satisfying Pareto front. The main parameter values of the algorithm: ant amount=100, optimization time of pheromone=150, optimization time of global experience=5, ant step length=0.8, global experience step length=0.7, perturbationcoefficient=0.4, volatilization coefficient=0.01, niche radius =0.01, pheromone minimum =0.001, pheromone maximum=0.3, pheromone heuristic factor $\alpha = 1$, expectation heuristic factor $\beta = 3$.



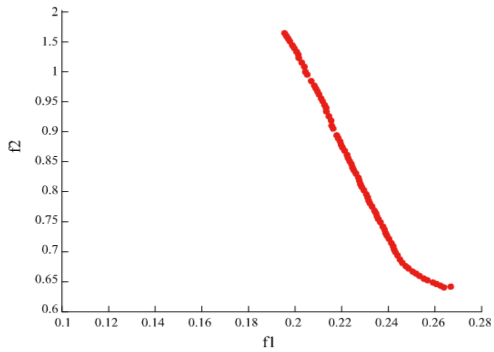
(a) Pareto front gained by test function SCH



(b) Pareto front gained by test function POL



(c) Pareto front gained by test function ZDT1



(d) Pareto front gained by test function ZDT3

Figure 2. Pareto front gained by test functions

It can be seen from Figure 2 that the non-inferior solution set gained through this algorithm is closest to the optimal set front, and maintains great advantage in the aspect of even distribution, which is mainly because this algorithm's selection of local and global optimized solution maintains the diversity of optimized solution, and is combined with the concepts of individual distance algorithm and non-inferior optimal control, which selects the optimal set accurately, to make it distribute evenly, therefore, performs relatively well.

6. Conclusion

No matter in the application of scientific research or engineering, multi-objective optimization is a very important research subject, because in many real life applications, it is often required to optimize many and often contradictory objects at the same time. In order to solve this multi-objective optimization problem, this paper has put forward the ant colony algorithm, and fully proved the efficiency and excellent performance of the multi-objective ant colony optimization with tests from its pivotal operators to standard test functions, and evaluation of performance measurement index.

References

[1] Giacomo Filippo Porzio, Gianluca Nastasi, Valentina Colla, Marco Vannucci, Teresa Annunziata Branca. Comparison of Multi-objective Optimization Techniques Applied to Off-gas Management within An Integrated Steelwork. *Applied Energy*. 2014; 136(31): 1085-1097.

[2] Zorka Novak Pintarič, Zdravko Kravanja. Suitable Process Modelling for Proper Multi-Objective Optimization of Process Flow Sheets. *Computer Aided Chemical Engineering*. 2014; 33(6): 1387-1392.

[3] Cherif Larouci, Kamal Ejjabraoui, Pierre Lefranc, et al. Impact of the Control Constraint on A Buck Converter Design Based on Multiobjective Optimization Procedure. *International Journal of Power Electronics and Drive Systems*. 2012; 2(3): 257-266.

[4] Hossein Hemmatian, Abdolhossein Fereidoon, Ali Sadollah, Ardeshir Bahreininejad. Optimization of Laminate Stacking Sequence for Minimizing Weight and Cost Using Elitist Ant System Optimization. *Advances in Engineering Software*. 2013; 57(3): 8-18.

[5] RM Rizk-Allah, Elsayed M Zaki, Ahmed Ahmed El-Sawy. Hybridizing Ant Colony Optimization with Firefly Algorithm for Unconstrained Optimization Problems. *Applied Mathematics and Computation.* 2013; 224(1): 473-483.

[6] A Sadighmanesh, K Zare, M Sabahi. Distributed Generation Unit and Capacitor Placement for Multiobjective Optimization. *International Journal of Electrical and Computer Engineering*. 2012; 2(5): 615-620.

[7] Qi Zhang, Guangchun Yang, Rui Tang, et al. Multi Objective Optimization Algorithms Design Based on Support Vector Regression Metamodeling. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2013; 11(11): 6406-6412.

[8] KZ Gao, PN Suganthan, QK Pan, TJ Chua, TX Cai, CS Chong. Pareto-Based Grouping Discrete Harmony Search Algorithm for Multi-Objective Flexible Job Shop Scheduling. *Information Sciences*. 2014; 289(24): 76-90.

[9] Khalid M Salama, Alex A Freitas. Classification with Cluster-Based Bayesian Multi-Nets using Ant Colony Optimisation. *Swarm and Evolutionary Computation*. 2014; 18(10): 54-70.

[10] Alberto Cano, Juan Luis Olmo, Sebastián Ventura. Parallel Multi-Objective Ant Programming for Classification using GPUs. *Journal of Parallel and Distributed Computing*. 2013; 73(6): 713-728.

[11] Tianshun Huang. Optimization of Routing Protocol in Wireless Sensor Networks by Improved Ant Colony and Particle Swarm Algorithm. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2014; 12(10): 7486-7494.

[12] Askar Bagherinasab, Mahmoud Zadehbagheri, Saifulnizam Abdul Khalid, et al. Optimal Placement of DSTATCOM Using Hybrid Genetic and Ant Colony Algorithm to Losses Reduction. *International Journal of Applied Power Engineering*. 2013; 2(2): 53-60.

[13] Lenin Kanagasabai, B Ravindranath Reddy, M Surya Kalavathi. Optimal Power Flow using Ant Colony Search Algorithm to Evaluate Load Curtailment Incorporating Voltage Stability Margin Criterion. *International Journal of Electrical and Computer Engineering*. 2013; 3(5): 603-611.

[14] Chengming Qi. Vehicle Routing Optimization in Logistics Distribution Using Hybrid Ant Colony Algorithm. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2013; 11(9): 5308-5315.

[15] Juan Rada-Vilela, Manuel Chica, Óscar Cordón, Sergio Damas. A Comparative Study of Multi-Objective Ant Colony Optimization Algorithms for The Time and Space Assembly Line Balancing Problem. *Applied Soft Computing*. 2013; 13(11): 4370-4382.