World Academy of Science, Engineering and Technology
International Journal of Mathematical and Computational Sciences
Vol:8, No:11, 2014

Solution Economic Power Dispatch Problems by an Ant Colony Optimization Approach

Navid Mehdizadeh Afroozi, Khodakhast Isapour, Mojtaba Hakimzadeh, Abdolmohammad Davodi

Open Science Index, Mathematical and Computational Sciences Vol:8, No:11, 2014 publications.waset.org/9999969/pdf

***Abstract*—**The objective of the Economic Dispatch(ED) Problems of electric power generation is to schedule the committed generating units outputs so as to meet the required load demand at minimum operating cost while satisfying all units and system equality and inequality constraints. This paper presents a new method of ED problems utilizing the Max-Min Ant System Optimization. Historically, traditional optimizations techniques have been used, such as linear and non-linear programming, but within the past decade the focus has shifted on the utilization of Evolutionary Algorithms, as an example Genetic Algorithms, Simulated Annealing and recently Ant Colony Optimization (ACO). In this paper we introduce the Max-Min Ant System based version of the Ant System. This algorithm encourages local searching around the best solution found in each iteration. To show its efficiency and effectiveness, the proposed Max-Min Ant System is applied to sample ED problems composed of 4 generators. Comparison to conventional genetic algorithms is presented.

***Keywords*—**Economic Dispatch (ED), Ant Colony Optimization, Fuel Cost, Algorithm.

I. Introduction

IN an electrical power system, a continuous balance must be maintained between electrical generation and varying load demand, while the system frequency, voltage levels, and security also must be kept constant. Furthermore, it is desirable that the cost of such generation be minimal [1], [2].

Scarcity of energy resources, increasing power generation cost, and ever growing demand for electric energy necessitate optimal economic dispatch (ED) in today's electric power systems. ED operation is performed at the energy management center every few minutes to allocate the optimal real power generation to the committed generators in a power network in such a way that the cost of operation becomes minimum while all other operating constraints are satisfied [1].

To solve these problems, various algorithmic and heuristic approaches have been suggested or investigated by power engineers, including Lagrangian relaxation [1], gradient method [2], linear programming and dynamic programming [3], interior point method [4], etc. More recently, heuristic techniques, such as genetic algorithms [5], simulated annealing [6], evolutionary computing [7], and tabu-search [8] have also been intensively investigated.

During current years, great interest has developed in algorithms inspired by the observation of natural phenomena to help solve complex computational problems. In this paper, we introduce Max-Min Ant System Algorithm [9], an imported version of basic Ant System [10] of the family algorithms: Ant Colony Optimization (ACO) [11], which was inspired by the observation of ant colonies.

MAX-MIN Ant System has been applied to the Traveling Salesman Problem [12] and to the Flow Shop Problem [13]. MAX-MIN Ant System (MMAS) proved to be useful in guiding the local search algorithms.

Navid Mehdizadeh Afroozi, Khodakhast Isapour, Mojtaba Hakimzadeh and Abdolmohammad Davodi are with Department of Electrical Engineering, Ahram Branch, Islamic Azad University, Ahram, Iran (e-mail: mehdizadeh.n@yahoo.com)

II. Ant Colony Optimization

A. General Concept

The Ant Colony Optimization (ACO) [14] is a meta-heuristic to solve combinatorial optimization problems, is motivated by the behavior of real ant colonies. When ants attempt to find short paths between their nest and food sources, they communicate indirectly by using pheromone (pheromone trail) to mark the decisions they made when building their respective paths. Within ACO algorithms, the optimization problem is represented as a complete weighted graph G = (N,A) with N being the set of nodes and A the set of edges fully connecting the nodes N. In the Travelling Salesman Problem (TSP) application, edges have a cost associated (e.g. their length) and the problem is to find a minimal-length closed tour that visits all the nodes once and only once. In order to solve the problem, random walks of a fixed number of ants through the graph take place. The transition probabilities of each ant are governed by two parameters associated to the edges of the graph: the pheromone values (or pheromone trail) τ_{ij}, representing the learned desirability of choosing node j when in node i. A pheromone trail is a more global type of information, and visibility values, defined as the inverse of the distance between two nodes i and j: $n_{ij} = \frac{1}{d_{ij}}$ where d_{ij} is the distance between these two nodes.

The more distinctive feature of ACO is the management of pheromone trails that are used, in conjunction with the objective function, to construct new solutions. Informally, the pheromone trails are used for exploration and exploitation. Exploration representing the probabilistic choice of the components used to construct a solution. A higher probability is given to elements with a strong pheromone trail. Exploitation is based on the choice of the component that maximizes a blend of pheromone-trail values and partial objective function evaluations. The mathematical formulations of the ACO algorithms presented in this paper named Ant System (AS) and Max-Min Ant System (MMAS) are given in the following sections.

Open Science Index, Mathematical and Computational Sciences Vol:8, No:11, 2014 publications.waset.org/9999969/pdf

B. Power Balance Constraint

Ant System (AS) [15] is the original and most simplistic ACO algorithm. The decision policy used within AS is as follows: The probability with which ant k, currently at node i, chooses to go to node j is given [10] by:

$$p_{ij}^{k}(t) = \frac{\left[\tau_{ij}(t)\right]^{\alpha} \cdot \left[n_{ij}\right]^{\beta}}{\sum_{l \in J_i^k} \left[\tau_{il}(t)\right]^{\alpha} \cdot \left[n_{il}\right]^{\beta}} \quad \text{If j } \epsilon J_i^k \text{ and 0 if } j \notin J_i^k \tag{1}$$

J_i^k : is the feasible neighborhood of ant k, that is, the set of nodes which ant k has not yet visited. $\tau_{ij}(t)$: is the concentration of pheromone associated with edge (i,j) in iteration t. n_{ij} : is the inverse of the length of the edge known as visibility. α and β: are parameters that control the relative importance of pheromone intensity versus visibility.

Upon conclusion of an iteration (i.e. each ant has generated a solution) the pheromone on each edge is updated, according to the following formula:

$$\tau_{ij}(t) = \rho \tau_{ij}(t) + \Delta \tau_{ij}(t) \tag{2}$$

where ρ is the coefficient representing pheromone persistence ($0 \le \rho < 1$), and $\Delta\tau_{ij}$, is a function of the solutions found at iteration t, given by:

$$\Delta \tau_{ij} = \sum_{k=1}^{n} \Delta \tau_{ij}^{k}(t) \tag{3}$$

n: number of ants

$\Delta\tau_{ij}^{k}$: is the quantity per unit of length of pheromone addition laid on edge (i,j) by the k[th] ant at the end of iteration t, is given by:

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

where $T^k(t)$ is the tour done by ant k at iteration t, $L^k(t)$, is its length and Q is a constant parameter, used for defining to be of high quality solutions with low cost.

C. Max-Min Ant System

Max-Min Ant System (MMAS) [16], is a direct improvement over AS. The solutions in MMAS are constructed in exactly the same way as in AS, that is, the selection probabilities are calculated as in Equation (1).

The main modifications by MMAS with respect to AS are the following:

(i). To exploit the best solutions found, after each iteration only one single ant is allowed to add pheromone

To avoid search stagnation, the allowed range of the pheromone trail strengths is limited to interval $[\tau_{\min}(t), \tau_{\max}(t)]$, that is $\tau_{\min}(t) \le \tau_{ij}(t) \le \tau_{\max}(t)$

(ii). The pheromone trails are initialized to the upper trail limit, which causes a higher exploitation at the start of the algorithm. The upper bound, $\tau_{\max}(t)$, is given by:

$$\tau_{\max}(t) = \frac{1}{(1-\rho)Cost_{opt}(t)} \tag{5}$$

where $Cost_{opt}(t)$, is the optimal solution value for a specific problem. The lower bound $\tau_{\min}(t)$, is given by:

$$\tau_{\min}(t) = \frac{\tau_{\max}(t)(1-\sqrt[n]{p_{best}})}{(\lambda-1)\sqrt[n]{p_{best}}} \tag{6}$$

where p_{best} is the probability of creating the global-best solution. This parameter is defined from the user. If p_{best}=1 then $\tau_{\min}(t)$ =0. Also if p_{best} is very small there is a probability to use $\tau_{\min}(t) > \tau_{\max}(t)$. In the case we set $\tau_{\min}(t) = \tau_{\max}(t)$ and this algorithm uses only this heuristic information for solving the problem. n is the number of decision points and λ is the average number of edges at each decision point.

III. Suggested MMAS for Economic Dispatch

A. Problem Formulation

The objective of the economic load dispatch problem is to minimize the function cost of the production units, (7), under the construction power balance, (8), and also under construction of the technical limits of the generators, (9), [5]:

$$Min \quad F_T(\mathbf{P}) = \sum_{i=1}^{n} F_i(P_i) \tag{7}$$

$$\sum_{i=1}^{n} P_i = P_D \tag{8}$$

$$P_{i,\min} \le P_i \le P_{i,\max} \tag{9}$$

where n: the number of running (on line) thermal units, P_i: is the output power of i-unit in MW, P: is the vector that contains all the P_i , $F_i(P_i)$: is the production cost P_i , MW, in thousands of drachmas per hour (kGrd/h), $P_{i,\min}$ and $P_{i,\max}$: are the power limitations of the i-unit in MW, P_D : total load demand in MW

In (7), the generation cost function $F_i(P_i)$ is usually expressed as a quadratic polynomial:

$$F_i(P_i) = a_i + b_i P_i + c_i P_i^2 \tag{10}$$

Open Science Index, Mathematical and Computational Sciences Vol:8, No:11, 2014 publications.waset.org/9999969/pdf

where a_i, b_i, c_i are cost coefficients of generator i.

B. Algorithm Description

One of the units, of our problem, is called 'reference unit' and is defined from the beginning. For every generator the area of its thermal limits is divided in discrete values. This division can be done in various ways. In this paper we can divide all fragments in equal number of sub-fragments. So for every generator (except the reference generator that has been defined from the beginning) we do not have a continuous fragment of power but a discrete definite set depending on the separation that has taken place. We suppose that all generators should function within their limit except the reference machine that can exceed its limits, and it can take some penalty. In the case that there is a violation of the maximum production limit for the reference generator (which we give the indicator N) then:

$$dif = P_N - P_{N,\max}.$$

The total cost (7), is calculated including a penalty parameter $F_{tot,\max} dif$. In the solution presented in the paper no measures are taken in case that the reference generator is 'under working', violating the lowest limit. A relative penalty is suggested, $F_{tot,\max} dif2$ where $dif2 = P_{N,\min} - P_N$.

The algorithm works like this: every ant starts from the first generator and selects a power level for that machine and this is repeated until it reaches the last generator which is the reference generator and it is responsible for the power balance and takes continuous values (Fig. 1).

At the end the total cost is calculated in order to decide whether the solution is satisfactory. Before presenting the algorithm we will present the mathematical model for MAX-MIN Ant System solution for the specific problem.

C. Mathematical Model

1- Transition Rule: lets an ant k is in generator i and it must choose a power level j for it, according to the probability distribution, called a random-proportional rule:

$$\text{if j } \epsilon\ J_i^k \quad p_{ij}^{\ k}(t) = \frac{\left[\tau_{ij}(t)\right]^{\alpha} \cdot \left[n_{ij}\right]^{\beta}}{\sum_{l \in J_i^k} \left[\tau_{il}(t)\right]^{\alpha} \cdot \left[n_{il}\right]^{\beta}} \qquad (11)$$

where J_i^k : is the list of the power levels that corresponds to generator I, n_{ij} : is the visibility. In the classical problem TSP this is defined as the inverse of the distance between two cities, $n_{ij} = \frac{1}{d_{ij}}$

So, it could be also used here the inverse of the cost for the particular power level:

$$n_{ij}(t) = \frac{1}{F_i(P_{ij})}$$

2- Pheromone Update Rule: $\tau_{ij}(t)$ is the pheromone quantity that is found in edge that connects every generator (except the reference generator) with power level. Because the algorithm that was used is the MMAS some notifications must be made concerning the pheromone calculation:

(i). Renewal of pheromone takes place from every ant in every iteration. Either from the one that has found the global best solution (global best ant) or from the one that has found the best solution in iteration (iteration best ant). These two mechanisms can be combined.

(ii). First of all $\tau_{ij}(t)$ is the quantity of pheromone on the edge that connects machine i with its power level j. At the beginning this quantity should be equal to $\tau_{\max}$, but since this has not been calculated yet we have to use a large value τ_0 and after the first iteration we must set all pheromone trails equal to $\tau_{\max}$. The resulting pheromone update rule is:

$$\tau_{ij}(t+1) = \rho\tau_{ij}(t) + \Delta\tau_{ij}(t) \qquad (12)$$

where

$$\Delta\tau_{ij}(t) = \begin{cases} 1/Cost^*(t), & \text{if } (i,j) \in T^*(t) \\ 0, & \text{if } (i,j) \notin T^*(t) \end{cases}$$

$Cost^*(t)$ is either the global best solution so far ($Cost_{best}(t)$), or the best solution during the current iteration ($Cost_{iter}(t)$) and $T^*(t)$ is the list that keeps track for the best solution. ρ with $0 \leq \rho \leq 1$ is the evaporation coefficient

After this step it is checked if the pheromone trails are within the limits $\tau_{\min}$, $\tau_{\max}$ and finally the pheromone is updated according to the following relationship:

$$\tau_{ij} \leftarrow \begin{cases} \tau_{\min} & \text{if } \tau_{ij} < \tau_{\min} \\ \tau_{\max} & \text{if } \tau_{ij} > \tau_{\max} \\ \tau_{ij} & \text{otherwise} \end{cases} \qquad (13)$$

The upper bound $\tau_{\max}(t)$ and the lower bound $\tau_{\min}(t)$ are given by (5) and (6) respectively.

(iii). Smoothing of pheromone (optional step): When the algorithm converges, the following mechanism can be activated, that increases pheromone levels depending on the difference from $\tau_{\max}$, so that the selection possibility of trails with low pheromone levels is minimized. This can take place as following:

$$\tau_{ij}^*(t) = \tau_{ij} + \delta(\tau_{\max}(t) - \tau_{ij}(t)) \qquad (14)$$

where $\tau_{ij}^*(t)$ and $\tau_{ij}(t)$ are the pheromone trails before and after smoothing, δ is a parameter set by the user and $0 \leq \delta \leq 1$.

Open Science Index, Mathematical and Computational Sciences Vol:8, No:11, 2014 publications.waset.org/9999969/pdf

For δ =1 there is a reinitialization of pheromone levels and for δ =0, this mechanism becomes inactivate.

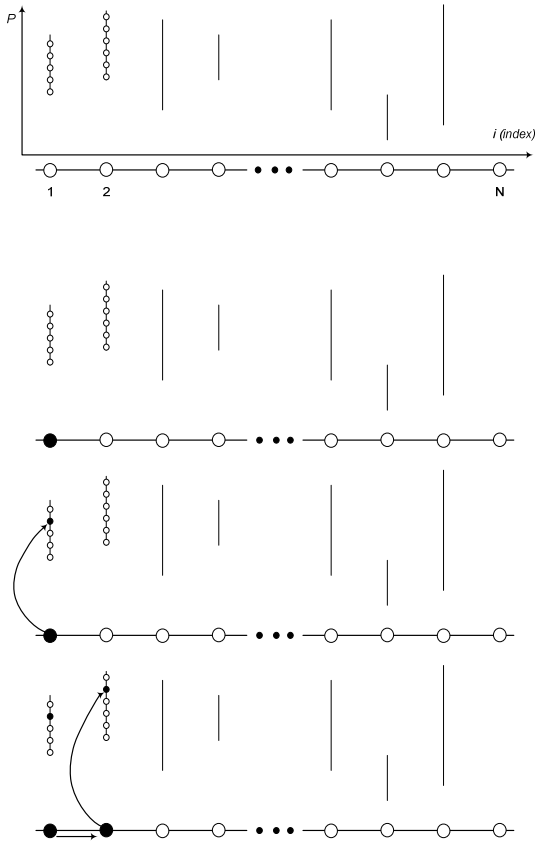


Fig. 1 Movement of an ant between machines and power levels

D. Application of Suggested Algorithm in ED Problem

The basic mode of operation, in steps, of the MAX-MIN Ant System is as follows:

Step 1. Define (discrete) power level for every generator and for the reference generator also. For every generator and for every power lever we calculate the specific production cost of the particular power quantity and the visibility $\eta_{ij}(t)$ (which can de defined as $\eta_{ij}(t) = 1/F_i(P_{ij})$ or possible with a more compatible-suitable expression with the problem $\eta_{ij}(t) = P_{ij}/F_i(P_{ij})$. Define the pheromone, giving it a very large value, in all edges that connect every generator with the power level respectively. Define the number of ants and the number of iterations.

Step 2. For every ant and for every engine select a power level based on the transition rule until the reference generator is reached.

Step 3. Calculate the cost for all ants based on the division of power levels and save the best. It must be checked if it is really the best and if it does not violate the power levels of the reference machine. Two are the possibilities when a violation takes place:

1. Reject the solution and continue to next iteration
2. Keep it, note that is not valid, and if the algorithm ends and a non valid solution is suggested, then a restart can take place with some of the parameters changed or the cost equation changed, so that non-valid solutions to be rejected even more.

Step 4. Renew pheromone using the rule mentioned above (Pheromone update rule), and also $\tau_{max} - \tau_{min}$ levels.

Step 5. Repeat the procedure from Step 2 until a specific number of iterations is completed, or some criterion is satisfied (for example cost needs of power fall under a threshold that was already requested).

Optional step: When the algorithm seems to converge, smoothing of pheromone trail can take place.

IV. Simulation Results

To assess the efficiency and effectiveness of the proposed MAX-MIN Ant System, it has been applied to ED problem, with 4 generators. The results obtained are compared with conventional genetic algorithm [5]. Table I shows the 4 generators. Here the system demand is 50 MW. 'Reference unit' is the 4th generator.

TABLE I
DATA OF TEST SYSTEM WITH 4 GENERATORS

	Type	$P_{i,min}$	$P_{i,max}$	a	b	c	Cost
1	Steam	4.0	6.25	0.0	0.368	0.0	46.0
2	Nat.Gas	3.0	14.75	2.0938	0.24837	0.002270	123.0
3	Oil	3.0	12.28	0.3667	0.109	0.00425	46.0
4	Steam	16.0	25.00	0.4053	0.2210	0.000643	46.0

Fig. 2 illustrates the machine load results after 100 iterations. Fig. 3 illustrates the best cost in relation with different values of parameter β, when the other parameters of the problem remain constant.

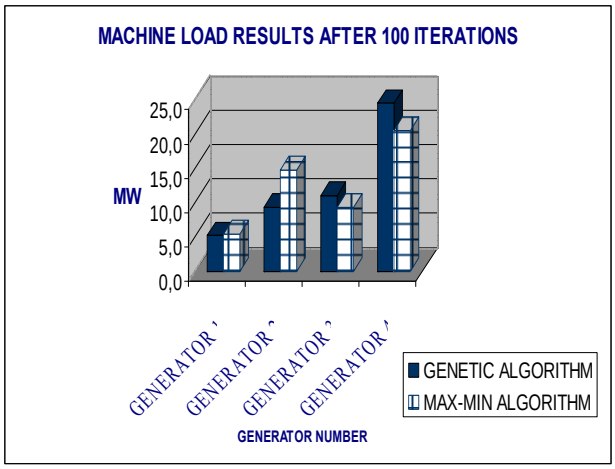


Fig. 2 Comparison of results using Genetic Algorithms and Max-Min Ant System after 100 iterations

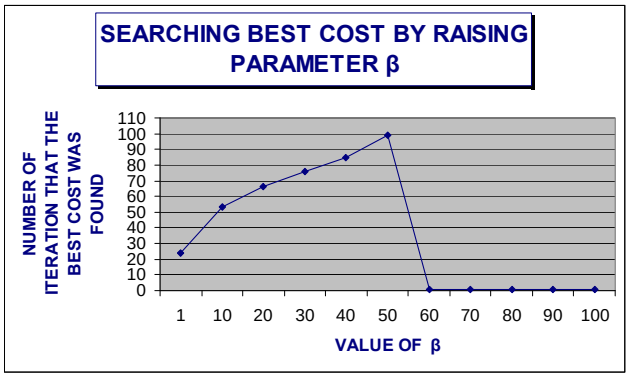


Fig. 3 Best cost in relation with parameter β

It was noticed that as β value became bigger the number of iterations that the optimal best cost was found became bigger.

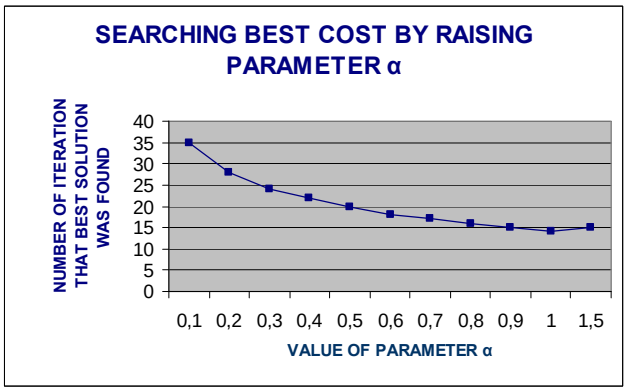


Fig. 4 Best cost in relation with parameter α

The program was tested for different values of parameter α, and keeping stable the other parameters of the problem. It was noticed that as the parameter α became bigger, best cost was found sooner, Fig. 4.

V. Conclusion

In this paper Max-Min Ant System is placed on solve the ED problem. An algorithm based on several modifications to AS which aim:

(i). to exploit more strongly the best solutions found during the search and to direct the ants search towards very high quality solutions and

(ii). to prevent premature convergence of the ants search. Among the main ideas introduce by MMAS, could be the utilization of pheromone trail limits to avoid premature convergence. The results obtained clearly shows the MMAS converges to the optimum solution. The massive parallel agent cooperation makes the ants able to jump over the local optimum ant to identify the right cluster easily; hence a good solution could be found.

References

[1] X. Guan, P.B. Luh, L. Zhang, Nonlinear approximation method in Lagrangian relaxation based algorithms for hydrothermal scheduling, *IEEE Trans. Power Systems*, Vol. 10, (2), pp. 772-778, 1995.

[2] A.J. Wood, B.F. Wollenberg, *Power Generation, Operation and Control*, John Wiley & Sons, New York 1996.

[3] A.M. Chebbo, M.R. Irving, Combined active and reactive dispatch, *Proc IEE, Pt.C*, (4), pp. 393-405, 1995.

[4] S. Granville, Optimal reactive dispatch through interior point methods, *IEEE Summer Meeting*, Paper No. 92, SM 416-8 PWRS, 1992.

[5] D.C. Walter and G.B. Sheble, "Genetic algorithm solution of economic dispatch with valve point loading," *IEEE Trans. Power Syst.*, vol. 8, no. 3, pp. 1325–1332, Aug. 1999.

[6] K.P. Wong and C.C. Fung, "Simulated annealing based economic dispatch algorithm," *Proc. Inst. Elect. Eng. C., Gen., Transm., Distrib.*, vol. 140, no. 6, pp. 505–519, Nov. 1993.

[7] N. Sinha, R. Chakrabarti, and P. K. Chattopadhyay, "Evolutionary programming techniques for economic load dispatch," *IEEE Trans. Evol. Comput.*, vol. 7, no. 1, pp. 83–94, Feb. 2003.

[8] W.M. Lin, F.S. Cheng, and M.T. Tsay, "An improved tabu search for economic dispatch with multiple minima," *IEEE Trans. Power Syst.*, vol. 17, no. 1, pp. 108–112, Feb. 2002.

[9] T. Stutzle, and H.H. Hoos, Max-Min Ant System, *Future Generation Computer Systems*, 16, pp.889-914, 2001.

[10] E. Bonabenn, M. Dorigo, G. Theraulaz, Swarm Intelligence from natural to Artificial systems, *Sante Fe Institute studies in the Sciences of complexity*, Oxford University Press, 1999.

[11] M. Dorigo, G. Di Caro, The Ant Colony Optimization Meta-Heuristic, In D.Come, M.Dorigo, and F.Glover, editors, *New Ideas in optimization*, Mc Graw-Hill, 2001.

[12] T. Stutzle, H. Hoos, The MAX-MIN Ant System and local search for the traveling Salesman Problem, *Proceedings of the IEEE International conference on Evolutionary Computaion* , ICEC '97, pp.309-314. 1997.

[13] T. Stutzle, An Ant Approach to the Flow Shop Problem, *Proceedings of the 6th European Congress on Intelligent Techniques and soft Computing (EUFIT '98)*, (3), Verlag Mainz, Aachem, pp.1560-1564, 1997.

[14] M. Dorigo, L.M. Gambardella, *Ant Algorithms for Discrete Optimization, Artificial Life*, volume 5, no.2, pp.137-172, 1998.

[15] M. Dorigo , V. Maniezzo, and K. Colorni, The ant system: optimization by a colony of cooperating agents, IEE Transactions on Systems, Man, and Cybernetics, Part B, *Cybernetics*, 26(1), pp. 29-44, 2000.

[16] T. Stutzle and H.H. Hoos, MAX-MIN Ant System, *Future Generation Computer Systems*, (16), pp.889-914, 2003.

Open Science Index, Mathematical and Computational Sciences Vol:8, No:11, 2014 publications.waset.org/9999969/pdf