KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2007
Cluj-Napoca (Romania), June 6–8, 2007, pp. 185–192

SENSITIVE ANT SYSTEMS IN COMBINATORIAL OPTIMIZATION

CAMELIA CHIRA(1), CAMELIA-M. PINTEA(1), AND D. DUMITRESCU(1)

Abstract. A new model based on the robust *Ant Colony System (ACS)* is introduced. The proposed *Sensitive ACS (SACS)* model extends ACS using the sensitive reaction of ants to pheromone trails. Each ant is endowed with a pheromone sensitivity level allowing different types of responses to pheromone trails. The SACS model facilitates a good balance between search exploitation and search exploration. Both ACS and SACS models are implemented for solving the *NP*-hard *Generalized Traveling Salesman Problem.* Comparative tests illustrate the potential and efficiency of the proposed metaheuristic.

1. Introduction

Metaheuristics are good strategies in terms of efficiency and solution quality for problems of realistic size and complexity. Regarded as strategic problem solving frameworks, metaheuristics are widely recognized as one of the most powerful approaches for combinatorial optimization problems. The most representative metaheuristics include genetic algorithms, simulated annealing, tabu search and ant colony [6].

The aim of this paper is to design a new metaheuristic based on *Ant Colony System (ACS)* [3] for solving combinatorial optimization problems. The introduced model is called *Sensitive ACS (SACS)* and uses different reactions of sensitive ants to pheromone trails. This technique promotes both search exploitation and search exploration for complex problems. The ACS and SACS models are implemented for solving the *Generalized Traveling Salesman Problem (GTSP)* [7, 8]. Numerical experiments indicate the potential of the introduced SACS model.

2. Ant Colony Systems

An ant algorithm is a system based on agents which simulate the natural behavior of ants including mechanisms of cooperation and adaptation. In [2] the use of this kind of system as a new metaheuristic was proposed in order to solve

2000 *Mathematics Subject Classification.* 90C59, 68T10.
Key words and phrases. metaheuristics, pattern recognition, stigmergy.

combinatorial optimization problems. This new metaheuristic has been shown to be both robust and versatile in the sense that it has been successfully applied to a range of different combinatorial optimization problems.

Ant algorithms are based on the following main ideas:

- Each path followed by an ant is associated with a candidate solution for a given problem.
- When an ant follows a path, the amount of pheromone deposited on that path is proportional to the quality of the corresponding candidate solution for the target problem.
- When an ant has to choose between two or more paths, the path(s) with a larger amount of pheromone has(have) a greater probability of being chosen by ants. As a result, ants eventually converge to a short path which hopefully represents the optimum or a near-optimum solution for the target problem.

Well known and robust algorithms include *Ant Colony System (ACS)* [3] and $\mathcal{MAX}-\mathcal{MIN}$ Ant System [10]. *Ant Colony System (ACS)* metaheuristics is a particular class of ant algorithms. The insects behavior is replicated to search the space. While walking between their ant nest and the food source, ants deposit a substance called *pheromone*. In the future every ant can direct its search according to the amount of this hormone on the ground.

3. The Generalized Traveling Salesman Problem

Let $G = (V, E)$ be an n-node undirected graph whose edges are associated with non-negative costs. Let $V_1, ..., V_p$ be a partition of V into p subsets called *clusters*. The cost of an edge $(i, j) \in E$ is $c(i, j)$.

The *generalized traveling salesman problem* (*GTSP*) refers to finding a minimum-cost tour H spanning a subset of nodes such that H contains exactly one node from each cluster V_i, $i \in \{1, ..., p\}$. The problem involves two related decisions: choosing a node subset $S \subseteq V$, such that $|S \cap V_k| = 1$, for all $k = 1, ..., p$ and finding a minimum cost Hamiltonian in S (the subgraph of G induced by S).

Such a cycle is called a *Hamiltonian tour*. The *GTSP* is called *symmetric* if and only if the equality $c(i, j) = c(j, i)$ holds for every $i, j \in V$, where c is the cost function associated to the edges of G.

The *GTSP* has several applications to location and telecommunication problems [4, 5, 7].

4. Ant Colony System for solving GTSP

An *Ant Colony System* for solving the *GTSP* is introduced. Let $V_k(y)$ denote the node y from the cluster V_k. The *ACS* algorithm for solving the *GTSP* works as follows.

Initially the ants are placed in the nodes of the graph, choosing randomly the *clusters* and also a random node from the chosen cluster

At iteration $t+1$ every ant moves to a new node from an unvisited *cluster* and the parameters controlling the algorithm are updated.

Each edge is labeled by a trail intensity. Let $\tau_{ij}(t)$ is the trail intensity of the edge (i,j) at time t. An ant decides which node is the next move with a probability that is based on the distance to that node (i.e. cost of the edge) and the amount of trail intensity on the connecting edge. The inverse of distance from a node to the next node is known as the *visibility*, $\eta_{ij} = \frac{1}{c_{ij}}$.

Each time unit evaporation takes place. This is to stop the intensity trails increasing unbounded. The rate evaporation is denoted by ρ, and its value is between 0 and 1.

To favor the selection of an edge that has a high pheromone value, τ, and high visibility value, η a probability function ${p^k}_{iu}$ is considered. ${J^k}_i$ are the unvisited neighbors of node i by ant k and $u \in {J^k}_i, u = V_k(y)$, being the node y from the unvisited cluster V_k. This probability function is defined as follows:

$$(1) \qquad {p^k}_{iu}(t) = \frac{[\tau_{iu}(t)][\eta_{iu}(t)]^{\beta}}{\Sigma_{o \in {J^k}_i}[\tau_{io}(t)][\eta_{io}(t)]^{\beta}},$$

where β is a parameter used for tuning the relative importance of edge cost in selecting the next node. ${p^k}_{iu}$ is the probability of choosing $j = u$, where $u = V_k(y)$ is the next node, if $q > q_0$ (the current node is i). q is a random variable uniformly distributed over $[0,1]$ and $0 \leq q_0 \leq 1$. If $q \leq q_0$ the next node j is chosen as follows:

$$(2) \qquad j = argmax_{u \in J_i^k}\{\tau_{iu}(t)[\eta_{iu}(t)]^{\beta}\},$$

After each transition the trail intensity is updated using the local correction rule:

$$(3) \qquad \tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \rho\tau_0.$$

Only the ant that generate the best tour is allowed to *globally* update the pheromone. The global update rule is applied to the edges belonging to the *best tour*. The correction rule is

$$(4) \qquad \tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \rho\Delta\tau(t),$$

where $\Delta\tau(t)$ is the inverse cost of the best tour.

The *ACS* for *GTSP* algorithm shown in the following, computes for a given time $time_{max}$ a (sub-)optimal solution for the given problem.

5. Proposed Sensitive Ant Colony System Model

The proposed *Sensitive Ant Colony System (SACS)* emphasizes a more robust and flexible system obtained by considering that not all ants react in the same way to pheromone trails. Within the proposed model, each ant is endowed with a pheromone sensitivity level denoted by PSL which is expressed by a real number in the unit interval [0, 1]. Extreme situations are:

- If $PSL = 0$ the ant completely ignores stigmergic information (the ant is 'pheromone blind');
- If $PSL = 1$ the ant has maximum pheromone sensitivity.

Small PSL values indicate that the ant will normally choose very high pheromone levels moves (as the ant has reduced pheromone sensitivity). These ants are more independent and can be considered environment explorers. They have the potential to autonomously discover new promising regions of the solution space. Therefore, search diversification can be sustained.

Ants with high PSL values will normally choose any pheromone marked move. Ants of this category are able to intensively exploit the promising search regions already identified. In this case the ant's behavior emphasizes search intensification.

During their lifetime the ants may improve their performance by learning. This process translates to modifications of the pheromone sensitivity. The PSL value can increase or decrease according to the search space topology encoded in the ant's experience.

6. Sensitive Ant Colony System for solving GTSP

The proposed SACS model for solving *GTSP* is described. Two ant colonies are involved. Each ant is endowed with a pheromone sensitivity level (PSL). Ants of the first colony have small PSL values indicating that they normally choose very high pheromone level moves. These sensitive-explorer ants are called *small PSL-ants (sPSL)*. They autonomously discover new promising regions of the solution space to sustain search diversification. Ants of the second colony have high PSL values. These sensitive-exploiter ants called *high PSL-ants (hPSL)* normally choose any pheromone marked move. They intensively exploit the promising search regions already identified by the first ant colony.

SACS for solving *GTSP* works as follows:

Step 1. Initially the ants are placed randomly in the nodes of the graph.

Step 2. At iteration $t + 1$ every *sPSL-ant* moves to a new node and the parameters controlling the algorithm are updated. When an ant decides which node is the next move it does so with a probability that is based on the distance to that node and the amount of trail intensity on the connecting edge. At each time unit evaporation takes place. This is to stop the intensity trails increasing unbounded. In order to stop ants visiting the same node in the same tour a tabu list is maintained. This prevents ants visiting nodes they have previously visited.

To favor the selection of an edge that has a high pheromone value, τ, and high visibility value, η a function ${p^k}_{iu}$ is considered. ${J^k}_i$ are the unvisited neighbors of node i by ant k and $u \in {J^k}_i$. ${p^k}_{iu}$ is the probability of choosing $j = u$ as the next node if $q > q_0$ (the current node is i). If $q \leq q_0$ the next node j is chosen as in Equation 2.

The sensitivity level is denoted by s and its value is randomly generated in $(0, 1)$. For *sPSL* ants s values are in $(0, s_0)$, where $0 \leq s_0 \leq 1$.

Step 3. The trail intensity is updated using the local rule as following.

$$\tau_{ij}(t+1) = s^2 \cdot \tau_{ij}(t) + (1-s)^2 \cdot \Delta\tau(t)\frac{1}{n}. \tag{5}$$

where n is the total number of the nodes.

Step 4. Step 2 and Step 3 are reconsidered by the *hPSL-ant* using the information of the *sPSL* ants. For *hPSL* ants s values are randomly chosen in $(s_0, 1)$.

Step 5. Only the ant that generates the best tour is allowed to *globally* update the pheromone. The global update rule is applied to the edges belonging to the *best tour*. The correction rule is Equation 4.

A run of the algorithm returns the shortest tour found. In the *SACS* algorithm for *GTSP* the implementation of the pheromone trail τ, in order to obtain more qualitative results comparing to the *ACS* for *GTSP* is improved.

The description of the *SACS* algorithm for *GTSP* is shown in Algorithm 1.

Algorithm 1. Sensitive Ant Colony System for GTSP

```
begin
  Set parameters, initialize pheromone trails
  Loop
    Place ant k on a randomly chosen node
    from a randomly chosen cluster
    Loop
      Each sPSL-ant incrementally build a solution (1)(2)
      A local pheromone updating rule (5)
      Each hPSL-ant incrementally build a solution (1)(2)
      A local pheromone updating rule (5)
    Until all ants have built a complete solution
    A global pheromone updating rule is applied (4)
  Until end_condition
end.
```

7. Numerical Experiments

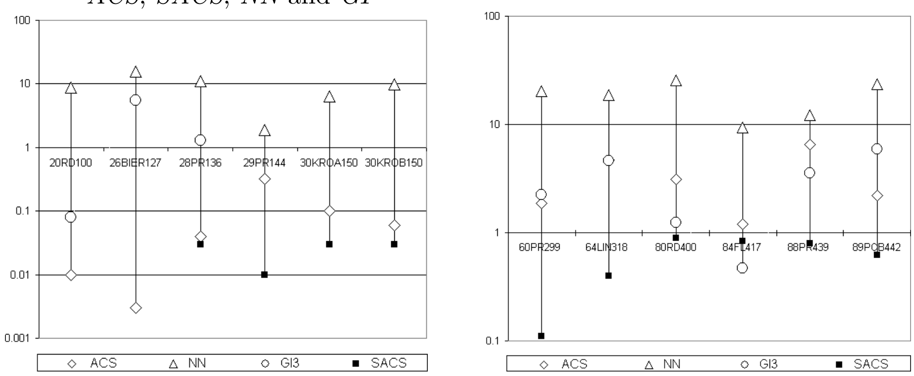
To evaluate the performance of the proposed model, the *SACS* algorithm for solving *GTSP* has been compared to the *ACS* algorithm, the *Nearest Neighbor (NN)* technique and the composite heuristic GI^3 [9]. Problems from *TSP* library [1] have been considered. *TSPLIB* provides optimal objective values for each of the problems. Several problems with Euclidean distances have been considered. Comparative results are shown in Table 2. To divide the set of nodes into subsets the procedure proposed in [4] has been used. This procedure sets the number of clusters $[n/5]$, identifies the m farthest nodes from each other, called centers, and assigns each remaining node to its nearest center.

The parameters used for both ant-based algorithms have been chosen as follows: τ_0=0.1, $\beta = 5$, $m = 10$, $\rho = 0.05$, $q_0 = 0.5$.

Besides the settings inherited from *ACS*, the *SACS* algorithm for *GTSP* uses an sensitivity parameter $s_0 = 0.5$. The sensitivity level of *hPSL* ants is considered to be distributed in the interval $(s_0, 1)$ while *sPSL* ants have the sensitivity level in the interval $(0, s_0)$.

All the solutions of *ACS* and *SACS* for *GTSP* are the average of five successively runs of the algorithm for each problem. Termination criteria is given by the $time_{max}$ the maximal computing time set by the user; in this case ten minutes. Figure 1 shows comparative computational results for solving the *GTSP* using the *ACS*, *SACS*, *NN* and GI^3.

Figure 1. Comparative Standard Deviation Average values for *ACS*, *SACS*, *NN* and GI^3



The *ACS* algorithm for *GTSP* performes well finding good solution in many cases. The test results clearly show that the newly introduced *SACS* algorithm outperforms the basic *ACS* model and obtains better results for most problems than those of the *NN* and GI^3.

TABLE 1. SACS algorithm for solving GTSP versus other algorithms

Problem	Opt.val.	NN	GI^3	ACS	SACS
11EIL51	174	181	**174**	**174**	**174**
14ST70	316	326	**316**	**316**	**316**
16EIL76	209	234	**209**	**209**	**209**
16PR76	64925	76554	**64925**	**64925**	**64925**
20RAT99	497	551	**497**	**497**	**497**
20KROA100	9711	10760	**9711**	**9711**	**9711**
20KROB100	10328	**10328**	**10328**	**10328**	**10328**
20KROC100	9554	11025	**9554**	**9554**	**9554**
20KROD100	9450	10040	**9450**	**9450**	**9450**
20KROE100	9523	9763	**9523**	**9523**	**9523**
20RD100	3650	3966	3653	3650.4	**3650**
21EIL101	249	260	250	**249**	**249**
21LIN105	8213	8225	**8213**	8215.4	**8213**
22PR107	27898	28017	**27898**	27904.4	27899.2
22PR124	36605	38432	36762	36635.4	**36619.2**
26BIER127	72418	83841	76439	72420.2	**72418**
28PR136	42570	47216	43117	42593.4	**42582.2**
29PR144	45886	46746	**45886**	46033	45890
30KROA150	11018	11712	**11018**	11029	11021.2
30KROB150	12196	13387	**12196**	12203.6	12199.6
31PR152	51576	53369	51820	51683.2	**51628.6**
32U159	22664	26869	23254	22729.2	**22693**
39RAT195	854	1048	**854**	856.4	**854**
40D198	10557	12038	10620	10575.2	**10562.2**
40KROA200	13406	16415	**13406**	13466.8	13416.8
40KROB200	13111	17945	**13111**	13157.8	13127.4
45TS225	68345	72691	68756	69547.2	**68473.6**
46PR226	64007	68045	**64007**	64289.4	64131
53GIL262	1013	1152	1064	1015.8	**1015.4**
53PR264	29549	33552	29655	29825	**29603.2**
60PR299	22615	27229	23119	23039.6	**22640.6**
64LIN318	20765	24626	21719	21738.8	**20846.8**
80RD400	6361	7996	6439	6559.4	**6417.4**
84FL417	9651	10553	**9697**	9766.2	9731.8
88PR439	60099	67428	62215	64017.6	**60571.6**
89PCB442	21657	26756	22936	22137.8	**21790.6**

8. Conclusions

A new *ACS*- based model called *Sensitive Ant Colony System* is introduced. Within *SACS* ants are endowed with a pheromone sensitivity level. The proposed model emphasizes a more aggressive exploration of the search space facilitating the detection of promising search areas. The *SACS* algorithm is applied for solving GTSP.

The computational results concerning the *SACS* algorithm are good and competitive - in both solution quality and computational time - with the existing heuristics from the literature [9]. Compared with the basic *ACS* model, the *SACS* algorithm produces better results for many of the test cases used. The results can be potentially improved by considering different parameters settings.

References

[1] Bixby, B., Reinelt, G.: http://nhse.cs.rice.edu/softlib/catalog/tsplib.html (1995)
[2] Dorigo, M.: Optimization, Learning and Natural Algorithms. Ph.D thesis, Dipart.di Elettronica, Politecnico di Milano, Italy (1992)
[3] Dorigo, M., Gambardella, L.M.:Ant Colony System:A cooperative learning approach to the Traveling Salesman Problem.IEEE Trans.Evol.Comp.,1,53-66 (1997)
[4] Fischetti, M., Gonzales, J.J.S, Toth, P.: A Branch-and-Cut Algorithm for the Symmetric Generalized Travelling Salesman Problem. Oper.Res. 45, 3, 378-394 (1997)
[5] Fischetti, M., Gonzales, J.J.S, Toth, P.: The Generalized Traveling Salesman and Orienteering Problem. Kluwer (2002).
[6] Glover, F.W., Kochenberger, G.A.: Handbook of Metaheuristics. Kluwer (2002).
[7] Laporte, G., Nobert, Y.: Generalized traveling salesman problem through n sets of nodes: An integer programming approach. INFOR 21, 1, 61–75 (1983)
[8] Noon, C.E., Bean, J.C.: A Lagrangian based approach for the asymmetric generalized traveling salesman problem. Oper.Res. 39, 623–632 (1991)
[9] Renaud, J., Boctor, F.F.: An efficient composite heuristic for the Symmetric Generalized Traveling Salesman Problem. Euro. J. Oper.Res., 108, 3, 571–584 (1998)
[10] Stützle, T., Hoos, H.H.: The $\mathcal{MAX}-\mathcal{MIN}$ Ant System and local search for the traveling salesman problem. Proc. Int. Conf. on Evol. Comp., IEEE Press, Piscataway, NJ, 309–314 (1997)

(1) Faculty of Mathematics and Computer Science, Babes-Bolyai University, Kogalniceanu 1, RO-400084 Cluj-Napoca, Romania

E-mail address: {cchira,cmpintea,ddumitr}@cs.ubbcluj.ro