GENERALIZED NET MODELS OF THE PROCESS OF ANT COLONY OPTIMIZATION.

Stefka Fidanova[1] **and Krassimir Atanassov**[2]

[1] IPP – Bulgarian Academy of Sciences, Acad. G. Bonchev Str., Bl. 25A, Sofia-1113, Bulgaria

e-mail: *stefka@parallel.bas.bg*

[2] CLBME – Bulgarian Academy of Sciences, Acad. G. Bonchev Str., Bl. 105, Sofia-1113, Bulgaria

e-mail: *krat@bas.bg*

Abstract: Ant Colony Optimization has been used successfully to solve hard combinatorial optimization problems. This metaheuristic method is inspired by the foraging behavior of ant colonies, which manage to establish the shortest routes to feeding sources and back. In this paper a generalized net model of the process of ant colony optimization is constructed. The present model is the first one on this theme and it will be a basis of next authors research.

1 Introduction

Combinatorial optimization is a branch of optimization. Its domain is optimization problems which set of feasible solutions is discrete or can be reduced to a discrete one, and the goal is to find the best possible solution. A combinatorial optimization problem consists of objective function, which needs to be minimized or maximized, and constraints. Examples of optimization problems are Traveling Salesman Problem [7], Vehicle Routing [8], Minimum Spanning Tree [6], Constrain Satisfaction [5], Knapsack Problem [3], etc. They are NP-hard problems and in order to obtain solution close to the optimality in reasonable time, metaheuristic methods are used. One of them is Ant Colony Optimization (ACO) [4].

Real ants foraging for food lay down quantities of pheromone (chemical cues) marking the path that they follow. An isolated ant moves essentially at random but an ant encountering a previously laid pheromone will detect it and decide to follow it with high probability and thereby reinforce it with a further quantity of pheromone. The repetition of the above mechanism represents the auto-catalytic behavior of a real ant colony where the more the ants follow a trail, the more attractive that trail becomes.

The ACO algorithm uses a colony of artificial ants that behave as cooperative agents in a mathematical space where they are allowed to search and reinforce pathways (solutions) in order to find the optimal ones. The problem is represented by graph and the ants walk on the graph to construct solutions. The solutions are represented by paths in the graph. After the initialization of the pheromone trails, the ants construct feasible solutions, starting from random nodes, and then the pheromone trails are updated. At each step the ants compute a set of feasible moves and select the best one (according to some probabilistic rules) to continue the rest of the tour. The structure of the ACO algorithm is shown by the pseudocode below. The transition probability $p_{i,j}$, to choose the node j when the current node is i, is based on the heuristic information $\eta_{i,j}$ and the pheromone trail level $\tau_{i,j}$ of the move, where $i, j = 1, \ldots, n$.

$$p_{i,j} = \frac{\tau_{i,j}^a \eta_{i,j}^b}{\sum_{k \in Unused} \tau_{i,k}^a \eta_{i,k}^b},$$

where $Unused$ is the set of unused nodes of the graph.

The higher the value of the pheromone and the heuristic information, the more profitable it is to select this move and resume the search. In the beginning, the initial pheromone level is set to a small positive constant value τ_0; later, the ants update this value after completing the construction stage. ACO algorithms adopt different criteria to update the pheromone level.

```
Hybrid Ant Colony Optimization
Initialize number of ants;
Initialize the ACO parameters;
while not end-condition do
    for k=0 to number of ants
        ant k starts from random node;
        while solution is not constructed do
            ant k selects higher probability node;
        end while
    end for
    Update-pheromone-trails;
end while
```

Figure 1: Pseudocode for ACO

The pheromone trail update rule is given by:

$$\tau_{i,j} \leftarrow \rho \tau_{i,j} + \Delta \tau_{i,j},$$

where $\rho \in (0, 1)$ models evaporation in the nature and $\Delta \tau_{i,j}$ is new added pheromone which is proportional to the quality of the solution.

2 GN-model

We shall construct a Generalized Net (GN, see [1, 2]) having 3 transitions, 16 places and four types (α, β, γ and ε) of tokens (see Fig. 2). These tokens enter, respectively, places l_1 with the initial characteristic

"n-dimensional vector with elements – the couples of the ants coordinates",

where n is the number of used ants.

l_2 – with the initial characteristic

"$\langle m$-dimensional vector of heuristics with elements – the graph vertices or

l-dimensional vector of heuristics with elements – the graph arcs; objective function$\rangle$",

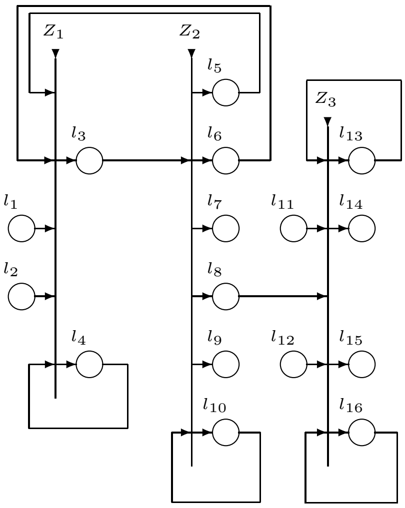

Figure 2: GN net model for ACO

where m is the number of the nods of the graph of the problem and l is the number of the arcs of the graph.

l_{11} – with the initial characteristic

"the graph structure with m vertices's and l arcs",

l_{12} – with the initial characteristic

"initial data for the places and quantities of the pheromones".

$$Z_1 =< \{l_1, l_2, l_4, l_5, l_6\}, \{l_3, l_4\},$$

$$\begin{array}{c|cc} & l_3 & l_4 \\ \hline l_1 & true & false \\ l_2 & false & true \\ l_4 & false & true \\ l_5 & true & false \\ l_6 & true & false \end{array} > .$$

Token α from places l_1, l_5 or l_6 enters place l_3 with a characteristic

"vector of current transition function results $\langle \varphi_{1,cu}, \varphi_{2,cu}, ..., \varphi_{n,cu} \rangle$",

while token ε stays only in place l_4 obtaining the characteristic

"new m-dimensional vector of heuristics with elements – the graph vertices or,

new l-dimensional vector of heuristics with elements – the graph arcs".

$$Z_2 =< \{l_3, l_{10}\}, \{l_5, l_6, l_7, l_8, l_9, l_{10}\},$$

$$\begin{array}{c|cccccc} & l_5 & l_6 & l_7 & l_8 & l_9 & l_{10} \\ \hline l_3 & W_{3,5} & W_{3,6} & W_{3,7} & false & W_{3,9} & W_{3,10} \\ l_{10} & false & false & true & false & W_{10,9} & W_{10,10} \end{array} >,$$

where
$W_{3,5} =$ "the current iteration is not finished",
$W_{3,6} = W_{3,10} = \neg W_{3,5} \vee \neg W_{10,9}$,
$W_{10,7} =$ "the current best solution is worse than the global best solution",
$W_{10,9} =$"truth-value of expression $C_1 \vee C_2 \vee C_3$ is *true*",
$W_{10,10} = \neg W_{10,9}$,
where C_1, C_2 and C_3 are the following end-conditions:
C_1 – "computational time (maximal number of iterations) is achieved",
C_2 – "number of iterations without improving the result is achieved",
C_3 – "if the upper/lower bound is known, then the current results are close (e.g., less than 5%) to the bound".

Token α from place l_3 enters place l_5 with a characteristic

$$\text{“}\langle S_{1,cu}, S_{2,cu}, ..., S_{n,cu}\rangle\text{”},$$

where $S_{i,cu}$ is the current partial solution for the current iteration, made by i-th ant ($1 \leq 1 \leq n$).

If $W_{3,6} = true$ it splits to three tokens α, α' and α'' that enter places l_6 – token α – with a characteristic

"new n-dimensional vector with elements – the couples of the new ants coordinates",

place l_8 – token α' – with the last α-characteristic, and place l_{10} – token α'' – with a characteristic

"⟨the best solution for the current iteration; its number⟩".

Token α'' can enter place l_9 only when $W_{10,9} = true$ and there it obtains the characteristic

"the best achieved result".

In place l_7 one of the two tokens from place l_{10} enters, which has the worst values as a current characteristic, while in place l_{10} the token containing the best values as a current characteristic stays.

$$Z_3 =< \{l_8, l_{11}, l_{12}, l_{13}, l_{16}\}, \{l_{13}, l_{14}, l_{15}, l_{16}\}, \begin{array}{c|cccc} & l_{13} & l_{14} & l_{15} & l_{16} \\ \hline l_8 & false & false & false & true \\ l_{11} & true & false & false & false \\ l_{12} & false & false & false & true \\ l_{13} & W_{13,13} & W_{13,14} & false & false \\ l_{16} & false & false & W_{16,15} & W_{16,16} \end{array} >,$$

where
$W_{13,14} = W_{16,15} =$"truth-value of expression $C_1 \vee C_2 \vee C_3$ is *true*",
$W_{13,13} = W_{16,16} = \neg W_{13,14}$.

Tokens γ from place l_{11} and β from place l_{12} with above mentioned characteristics enter, respectively, places l_{13} and l_{16} without any characteristic.

Token α from place l_8 enters place l_{16} and unites with token β (the new token is again β) with characteristic

"value of the pheromone updating function with respect of the values of

the objective function".

Tokens β and γ enter, respectively, places l_{14} and l_{15} without any characteristics.

3 Conclusion

In this paper, we address the modelling of the process of ant colony optimization method by generalized net. The aim of this representation is to study in detail the methodology and relationships between the processes. Thereby we can see the weaknesses of the method and to improve it implementation. In future work, other generalized net models of different ant colony applications will be developed and analyzed . Our aim is to help to the optimization algorithm developer to understand in detail the behavior of their algorithms and to improve it if it is possible.

References

[1] Atanassov K., Generalized Nets, World Scientific, Singapore, New Jersey, London, 1991.

[2] Atanassov, K., On Generalized Nets Theory. Prof. M. Drinov Publishing House, Sofia, 2007.

[3] Fidanova S., Evolutionary Algorithm for Multiple Knapsack Problem, Int. Conference Parallel Problems Solving from Nature, Real World Optimization Using Evolutionary Computing, ISBN No 0-9543481-0-9,Granada, Spain, 2002.

[4] Dorigo M., Gambardella L.M., Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem, IEEE Transactions on Evolutionary Computation 1, 53–66, 1997.

[5] Lessing L., Dumitrescu I., Stutzle T. A Comparison Between ACO Algorithms for the Set Covering Problem. ANTS Workshop 2004, pp.1 – 12.

[6] Reiman M., Laumanns M., A Hybrid ACO algorithm for the Capacitated Minimum Spanning Tree Problem, In proc. of First Int. Workshop on Hybrid Metahuristics, Valencia, Spain, 2004, pp.1 –10.

[7] Stutzle T. Dorigo M., ACO Algorithm for the Traveling Salesman Problem, In K. Miettinen, M. Makela, P. Neittaanmaki, J. Periaux eds., Evolutionary Algorithms in Engineering and Computer Science, Wiley, pp. 163 – 183, 1999.

[8] Zhang T., Wang S., Tian W., Zhang Y., ACO-VRPTWRV: A New Algorithm for the Vehicle Routing Problems with Time Windows and Re-used Vehicles based on Ant Colony Optimization, Sixth International Conference on Intelligent Systems Design and Applications, IEEE press, 2006, pp. 390 – 395.