GENERALIZED NET MODELS FOR THE PROCESS OF HIBRID ANT COLONY OPTIMIZATION

Stefka Fidanova and Krassimir Atanassov*

Abstract: Ant Colony Optimization has been successfully used 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. Vary often the ant algorithms are combined with local search procedures to improve achieved results. These are so called hybrid algorithms. In this paper an example of generalized net model for the process of hybrid ant colony optimization is constructed. The presented example is the first one on this theme and it will be a basis for future authors research.

Key words: generalized nets, combinatorial optimization, ant colony optimization.

2000 Mathematics Subject Classification: 68T20, 68Q85, 90C27.

1 Introduction

Combinatorial optimization is a branch of optimization. Its domain is optimization problems whose 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 [9], Vehicle Routing [10], Minimum Spanning Tree [7], Constrain Satisfaction [6], Knapsack Problem [4], 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) [3].

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
    Local Search Procedure;
    Update-pheromone-trails;
end while

Figure 1: Pseudocode for ACO

Very often ACO algorithms are hybridized by Local Search (LS) procedures. We thus try to improve the achieved results. The main concept of LS is searching the local neighborhood of the current solution [8]. The LS procedure perturbs (mutate) a given solution to generate different neighborhoods using a move generation mechanism. A move generation is a transition from a solution S to another one $S' \in I(S)$ in one step (iteration). The returned solution S' may not be optimal, but it is the best solution in its local neighborhood $I(S)$. A local optimal solution is a solution with the local optimal value of the objective function. Than the S' is compared with S and the new current solution $\overline{S}$ is the better of them.

The pheromone trail update rule is applied on $\overline{S}$ given by:

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

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

We represent the functioning of hybrid ACO algorithm by Generalized Net (GN) in the next section. Thus we visualize it behavior and apprehending.

2 GN-model

The Generalized Net (GN) [1, 2] is an extension of Petri nets. GN contains tokens, which are transferred from place to place. Every token enters the net with an initial characteristic. During each transfer, the tokens receives new characteristic. So, they accumulate their "history". This is the first essential difference with the Petry nets. Every GN-place has at most one arc entering and at most one arc leaving it. The places with no entering arcs are called input places for the net and those with no leaving arcs are called output places. The

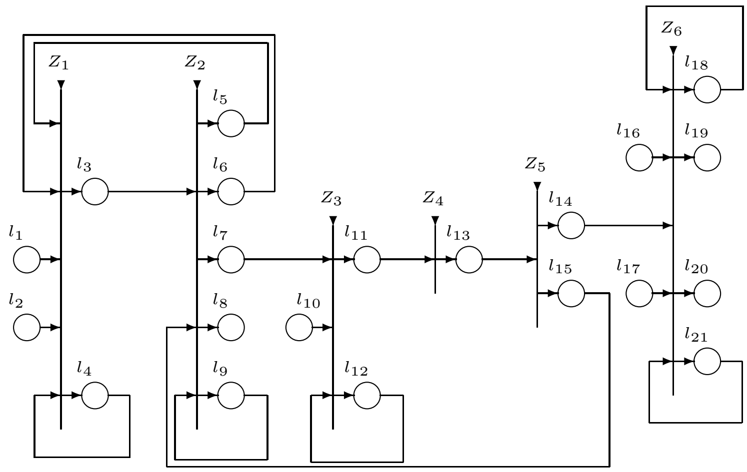

Figure 2: GN net model for ACO

second basic difference between GN and Petry nets is the "place-transition" relation. In GN transition may contain m input and n output places, where $m, n \geq 1$.

The present Generalized Net is an extension of the GN from [5]. We shall keep all notations from [5] so the reader to have possibility to compare both models.

We shall construct a GN having 6 transitions, 20 places and five types ($\alpha, \beta, \gamma, \varepsilon$ and μ) 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",

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$\rangle$",

l_{10} – with the initial characteristic

"function generating a mutation",

l_{16} – with the initial characteristic

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

l_{17} – with the initial characteristic

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

The tranzition Z_1 represents the solution construction as follows:

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

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

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".

The tranzition Z_2 represents end conditions for current iteration and for the hall aloorithm as follows:

$$Z_2 =< \{l_3, l_9, l_{15}\}, \{l_5, l_6, l_7, l_8, l_9\},$$

$$\begin{array}{c|ccccc} & l_5 & l_6 & l_7 & l_8 & l_9 \\ \hline l_3 & W_{3,5} & W_{3,6} & W_{3,7} & false & W_{3,9} \\ l_9 & false & false & false & W_{9,8} & W_{9,9} \\ l_{15} & false & false & false & false & true \end{array} >,$$

where
$W_{3,5} =$ "the current iteration is not finished",
$W_{3,6} = W_{3,7} = W_{3,9} = \neg W_{3,5} \vee \neg W_{9,8}$,
$W_{9,8} =$"truth-value of expression $C_1 \vee C_2 \vee C_3$ is $true$",
$W_{9,9} = \neg W_{9,8}$,
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 boundary is know, then the current results are close (e.g., less than 5%) to it".

Token α from place l_3 enters place l_5 with a characteristic

"$\langle S_{1,cu}, S_{2,cu}, ..., S_{n,cu} \rangle$",

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

If $W_{3,6} = true$ token α splits to three tokens $\alpha, \alpha_{cu_{it}}$ and δ_{it} that enter places l_6 – token α – with a characteristc

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

place l_7 – token $\alpha_{cu_{it}}$ – with the last α-characteristc, and place l_9 – token δ_{it}. This token unites with token δ (below we shall mark this token by δ instead of δ_1) in place l_9 that enters there after the end of the first iteration and that has as a current characteristc

"$\langle$the best solution for the current iteration; its number$\rangle$".

Token δ can enter place l_8 only when $W_{9,8} = true$ and there it obtains the characteristic

"the best achieved result".

The tranzition Z_3 represents local serch procedure as follows:

$$Z_3 =< \{l_7, l_{10}, l_{12}\}, \{l_{11}, l_{12}\},$$
$$\begin{array}{c|cc} & l_{11} & l_{12} \\ \hline l_7 & true & false \\ l_{10} & false & true \\ l_{12} & false & true \end{array} > .$$

Token α from place l_7 enters place l_{11} with a characteristic

"mutation $\langle S'_{1,cu}, S'_{2,cu}, ..., S'_{n,cu}\rangle$ of the current solution in respect to the initial

μ's characteristric".

while on first time-moment token μ enters place l_{12} and stays only there during the entire time of GN-functioning without any new characteristic.

The tranzition Z_4 represents iteration best solution as follows:

$$Z_4 =< \{l_{11}\}, \{l_{13}\}, \begin{array}{c|c} & l_{13} \\ \hline l_{11} & false \end{array} > .$$

Token α from place l_{11} enters place l_{13} with a characteristic

"$\langle \overline{S}_{1,cu}, \overline{S}_{2,cu}, ..., \overline{S}_{n,cu}\rangle$,"

where $\overline{S}_i$ is the better solution between solutions S_i and S'_i, $1 \le i \le n$.

The tranzition Z_5 represents the global best solution as follows:

$$Z_5 =< \{l_{13}\}, \{l_{14}, l_{15}\},$$
$$\begin{array}{c|cc} & l_{14} & l_{15} \\ \hline l_{13} & true & true \end{array} > .$$

Token α from place l_{13} splits to two tokens – α^* that enters place l_{15} with a characteristic

"S, where $S = Best(\overline{S}_{1,cu}, \overline{S}_{2,cu}, ..., \overline{S}_{n,cu})$",

where *Best* is the best solution according to the objective function, and token α that enters place l_{14} without any characteristic.

The tranzition Z_6 represents the pheromone updating as follows:

$$Z_6 =< \{l_{14}, l_{16}, l_{17}, l_{18}, l_{21}\}, \{l_{18}, l_{19}, l_{20}, l_{21}\}$$

,

$$\begin{array}{c|cccc} & l_{18} & l_{19} & l_{20} & l_{21} \\ \hline l_{14} & false & false & false & true \\ l_{16} & true & false & false & false \\ l_{17} & false & false & false & true \\ l_{18} & W_{18,18} & W_{18,19} & false & false \\ l_{21} & false & false & W_{21,20} & W_{21,21} \end{array} >,$$

where
$W_{18,19} = W_{21,20} =$"truth-value of expression $C_1 \vee C_2 \vee C_3$ is *true*",
$W_{18,18} = W_{21,21} = \neg W_{18,19}$.

Tokens γ from place l_{16} and β from place l_{17} with above mentioned characteristics enter, respectively, places l_{18} and l_{21} without any characteristic.

Token α from place l_{14} enters place l_{21} and unites with token β (the new token is again β) with characteristic

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

the objective function".

Tokens β and γ enter, respectively, places l_{20} and l_{19} without any characteristics.

The GN model of the hybrid ACO gives us detailed notation and better understanding of the algorithm behavior. This representation can suggest ideas for development and improvement of the method.

3 Conclusion

In this paper, we address the modelling of the process of hybrid ant colony optimization method by generalized net. The aim of this representation is to study in details the methodology and relationships between the processes. Thereby we can see the possibilities to improve the method and it implementation. The constructed GN for hybrid ACO gives us an idea that we can use dynamic mutation function instead of static, i. e. the characteristic of the token μ can be changed and we use the last one which is a novelty. Other authors use only static mutation. In future work, other generalized net models of different ant colony applications will be developed and analyzed . Our aim is to help 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] Dorigo M., L. M. Gambardella, IEEE Transactions on Evolutionary Computation 1,1997, 53-66.

[4] 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.

[5] Fidanova S., K. Atanassov. Issues on Intuitionistic Fuzzy Sets and Generalized Nets, Vol 7, 2008, 108-114.

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

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

[8] Schaffer A. A., M. Yannakakis, Simple Local Search Problems that are Hard to Solve. Society for Industrial Applied Mathematics Journal on Computing, Vol 20, 1991, 56-87.

[9] Stutzle T. M. Dorigo, J. Periaux eds., Evolutionary Algorithms in Engineering and Computer Science, Wiley, 1999, 163-183.

[10] Zhang T., S. Wang, W. Tian, Y. Zhang, 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, 390-395.

Institute for Parallel Processing
Bulgarian Academy of Science
Acad. G. Bonchev str. bl25A
1113 Sofia, Bulgaria
stefka@parallel.bas.bg

* Center for Biomedical Engineering
Bulgarian Academy of Sciences
Acad. G. Bonchev Str., Bl. 105
1113 Sofia, Bulgaria
krat@bas.bg