

ANT COLONY OPTIMIZATION (ACO) FOR GRAPH COLORING PROBLEM

FATMA MOHAMMED JABBER, KHA WLA HUSSEIN ALI, AND KAREEM RADHI HASSAN

Computer Science Dept., College of Science, Basrah University

Abstract

We describe an artificial Ant Colony Optimization (ACO) capable of solving the graph coloring problem. Ants of the artificial colony are able to generate successively shorter feasible tours by using information accumulated in the form of pheromone trail deposited of the edge of the colonies ant. Computer simulations demonstrate that the artificial ant colony is capable of generating good solutions. It is the successful use natural metaphor to design an optimization algorithm. We generated randomly 20 graphs of 100 nodes, the performance is a best than any other available heuristic techniques such that genetic algorithm and others, so the average of coloring of a graphs is less than any other algorithms. We achieve the goal the goal of the problem

Keywords: Ant colony optimization, artificial life, graph coloring problem

/ /

100

20

1- INTRODUCTION

Ant algorithm were inspired the observation of real ant colonies> Ants are social insects, that is, insects that live in colonies and where behavior is directed more to the survival of the colony as a whole than to that of single individual component of the colony. An important and foraging behavior and how ants can find the shortest path between food sources to the nest and vice versa, ants deposit on the ground substance called pheromone, forming in this' way pheromone trail. The pheromone trial allows the ants to find their back to the food source [

Dorigo, 1992].

Real ants are capable of finding the shortest path from a food source to the nest without using visual cues. Also, they are capable of adapting to changes in the environment, for example, finding a new shortest path once the old one, is no longer feasible due to a new obstacle. Consider Fig. 1A: ants are moving on a straight line that connects a food source to their nest. It is well known that the primary means for ants to form and maintain the line is a pheromone trail. Ants deposit certain amount of pheromone while walking, and each ant probabilistically prefers to follow a direction rich in pheromone. This elementary behavior of real ants can be used to explain how they can find the shortest path that reconnects a broken line after the sudden appearance of an unexpected obstacle has interrupted the initial path (Fig. 1B). In fact, once the obstacle has appeared, those ants which are just in front of the obstacle cannot continue to follow the pheromone trail and therefore they have to choose between turning right or left. In this situation we can expect half the ants to choose to turn right and the other half to turn left. A very similar situation can be found on the other side of the obstacle (Fig. 1 C). It is interesting to note that those ants which choose, by chance, the shorter path around the obstacle will more rapidly reconstitute the interrupted pheromone trail compared, the those which choose the longer path. Thus, the shorter path will receive a greater amount of pheromone per time unit and in turn a larger number of ants will choose the shorter path. Due to this positive feedback (autocatalytic) process, all the ants will rapidly choose the shorter path (Fig. 1D) [Dorigo, 1993]. The most interesting aspect of this autocatalytic process is that finding the shortest path around the obstacle seems to be an emergent property of the interaction between the obstacle shape and ants distributed

2- Related Work

ACO algorithms show similarities with some optimization, learning and simulation approaches like heuristic graph search. Monte Carlo simulation, neural networks, and evolutionary computation. These similarities are briefly discussed in the following.

Heuristic graph search: In ACO algorithms each ant performs an heuristic graph search in the space of the components of a solution: ants take biased probabilistic decisions to chose the next component to move to, where the bias is given by an heuristic evaluation function which favors components which are perceived as more promising.

Monte Carlo simulation: ACO algorithms can be interpreted as parallel replicated Monte Carlo systems. Monte Carlo systems are general stochastic simulation systems, that is, techniques performing repeated sampling experiments on the model of the system under consideration by making use of stochastic component in the state sapling and \ or transition rules. Analogously, in ACO algorithms the ants sample the problems solution space by repeatedly applying a stochastic decision policy until a feasible solution of the considered problem is built. Each ant "experiment" allows to adaptively modify the local statistical knowledge on the problem structure (i.e., the pheromone trails).

Neural networks: Ant colonies, being composed of numerous concurrently and locally interacting units, can be seen as "connectionist" systems [Bishop, 1995, Hertz, 1991]. From a structural point of view, the parallel between the ACO meta-heuristic and a generic neural network is obtained by putting each state I visited by ants in correspondence with a neuron I,

and the problem specific neighborhood structure of state I in correspondence with the set of synaptic-like propagating through the neural network and modifying the strength of the synaptic-like inter-neuron connections. Signals (ants) are locally propagated by means of a stochastic transfer function and the more a synapse is used, the more the connection between its two end-neurons is reinforced.

Evolutionary computation: there are some general similarities between the ACO meta-heuristic and evolutionary computation (EC). Both approaches use a population of individuals which represent problem solutions, and in both approaches the knowledge about the problem collected by the population is used to stochastically generate a new population of individuals. A main difference is that in EC algorithms all the knowledge about the problem is contained in the current population, while in ACO a memory of past performance is maintained under the form of pheromone trails [Fogel, 1995, Goldberg, 1989].

behavior. Although all ants move at approximately the same speed and deposit a pheromone trail approximately the same rate, it is a fact that it takes longer to counter obstacles on their longer side on their shorter side which makes the pheromone trail accumulate quicker on the shorter side. It is the ants preference for higher pheromone trail levels which makes this accumulation still quicker on the shorter path [Dorigo, 1999].

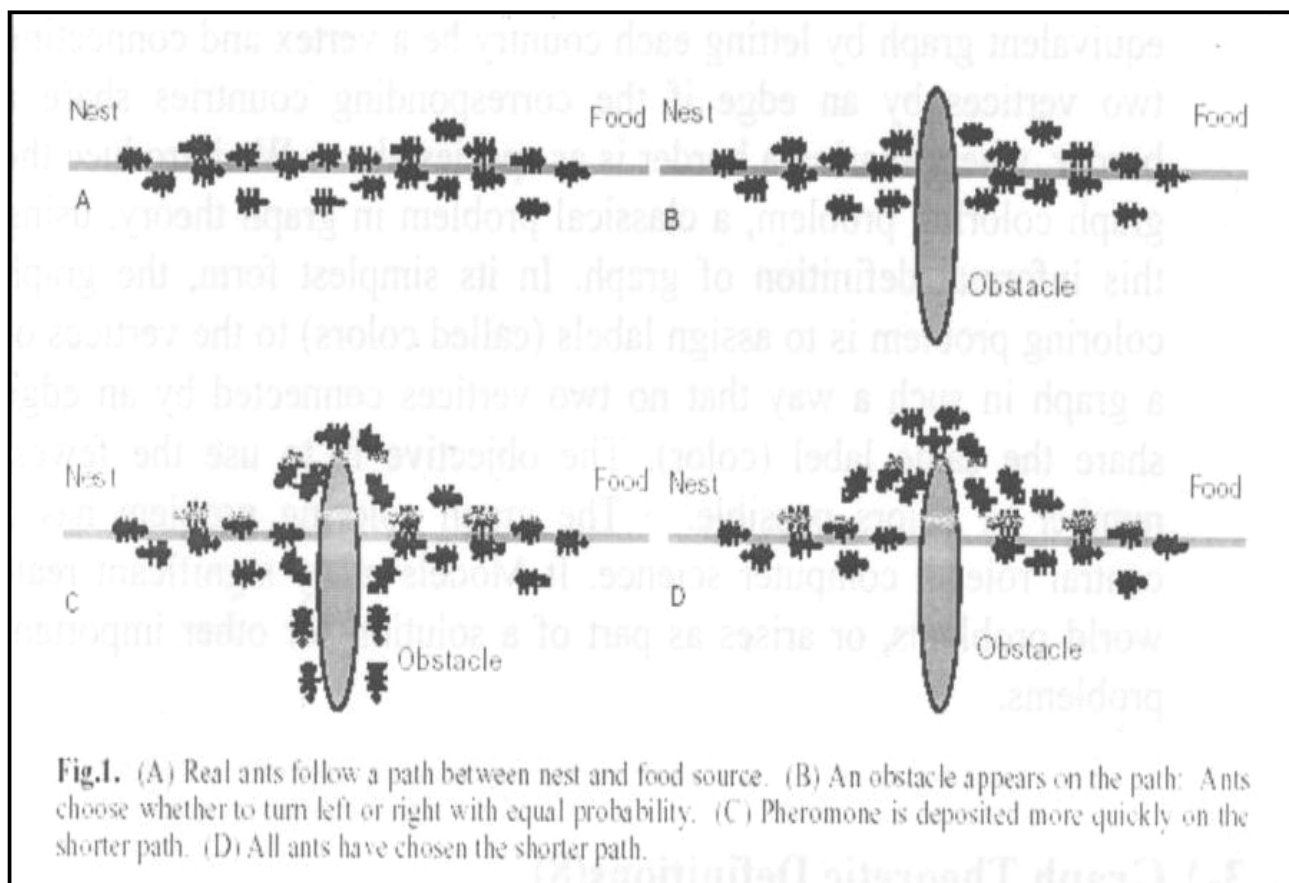


Fig.1. (A) Real ants follow a path between nest and food source. (B) An obstacle appears on the path: Ants choose whether to turn left or right with equal probability. (C) Pheromone is deposited more quickly on the shorter path. (D) All ants have chosen the shorter path.

In general a Graph Coloring Problem (GCP), given a graph $G = (N, E)$, N is the number of nodes, E is vertices, X -coloring of G is a mapping $C: N \rightarrow [1, \dots, X]$, such that $C(i) \neq C(j)$ if $(i, j) \in E$. The GCP is the problem of finding a coloring of the graph G so that the number X of colors used is minimum, that means the objective is to use the fewest numbers of colors possible, so we describe an ant colony optimization to solve this problem [Dorigo, 1999].

3- GRAPH COLORING PROBLEM(7,S)

Let us say you are asked to help out a cartographer or a map-maker with her map-coloring problem. It wants to color the countries on a map. It doesn't matter which color a country is assigned, as long as its color is different from that of all bordering countries. If two countries meet only at a single point, they do not count as sharing a border and hence can be made the same color. The cartographer is poor and can't afford many crayons, so the idea is to use as few colors as possible. In 1852 Francis Guthrie, while trying to color the map of countries of England, noticed that four colors sufficed. Subsequently, he conjectured that 4 colors are enough to color any map. Successive efforts made to prove Guthrie's 4-color conjecture led to the development of much of graph theory.

Some of which are connected by lines! called edges. A vertex in a graph models some physical entity or abstract concept. An edge, which joins exactly two vertices, represents the relationship or association between the respective entities or concepts. The map of countries mentioned above can, for instance, be converted into an equivalent graph by letting each country be a vertex and connecting two vertices by an edge if the corresponding countries share a border, where sharing a border is as species above. We introduce the graph coloring problem, a classical problem in graph theory, using this informal definition of graph. In its simplest form, the graph coloring problem is to assign labels (called colors) to the vertices of a graph in such a way that no two vertices connected by an edge share the same label (color). The objective is to use the fewest number of colors possible. The graph coloring problem has a central role in computer science. It models many significant real-world problems, or arises as part of a solution for other important problems

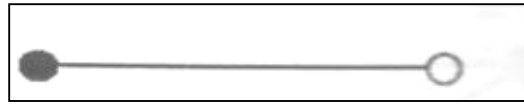
3-1 Graph Theoretic Definitions(8)

The graph coloring problem A vertex coloring of a graph, or simply Coloring for short, is an assignment of colors to the vertices such that no two adjacent vertices are assigned the same color. Alternatively, a coloring is a partition of the vertex set into a collection of vertex-disjoint independent sets. Each independent set in such a partition is called a color class. The graph coloring problem is to find a vertex coloring for a graph using the minimum number of colors possible. A X -coloring of a graph G is a coloring of G using X colors. The minimum possible value for X is called the chromatic number of G , denoted as $\chi(G)$. A coloring with the fewest possible number of colors (a X -coloring) is called an optimal coloring ..

3-2 Graph Theoretic Notation:

A graph G is a pair (V, E) of a set of vertices V and a set of edges E . The edges are unordered pairs of the form (i, j) where $i, j \in V$. Two vertices i and j are said to be adjacent if and only if $(i, j) \in E$ and Non-adjacent otherwise. And can show some examples for graph coloring

problem in follow.



Fig(2)Example 1: 2-coloring of graph

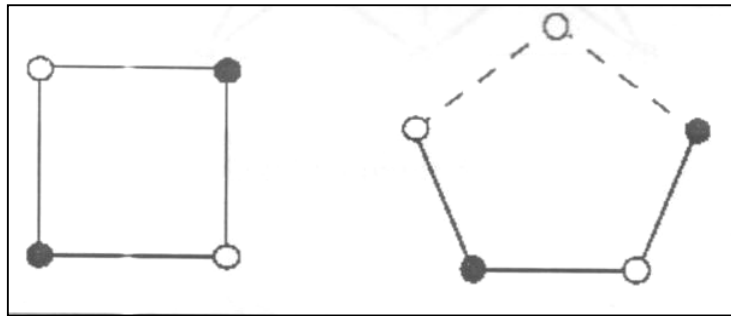


Fig (3): Example 2: 2-coloring of a graph1 and 3-Coloring of graph2

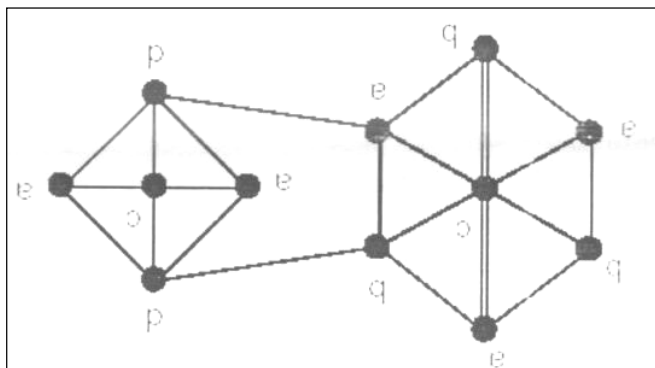


Fig (4): Example 3: 4-coloring of a graph.(Colors = a, b, c,d)

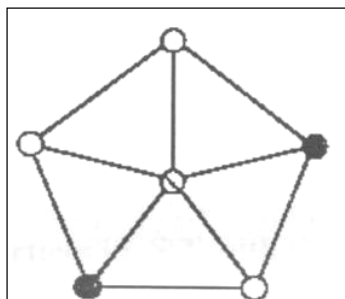


Fig (5) Example 4: 4-coloring of a graph

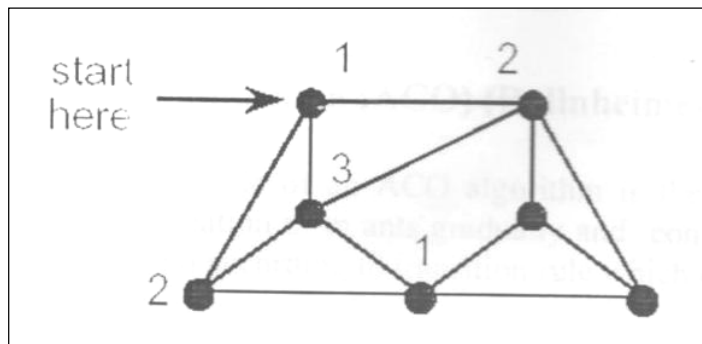


Fig (6) Example 5: A 2-color

3-3 APPLICATIONS OF GRAPH COLORING

(Coleman, More, 1983)

1. Transmitters 311d channel assignment
2. Fast register assignment
3. final exam scheduling

Vertices = classroom sections (over all courses) Two sections are adjacent if 3 student in both

4. cartography
- 5.

3-4 ALGORITHM FOR GRAPH COLORING PROBLEM

The algorithm for graph coloring problem can show in figure (7) (David ,2003).

```

Void color(graph G)
_ {
Computer array XS=V(X)
  for (T= 2^n - 1; T ≥ 0; T -- )
    {if (T<S AND x[s\T]=1 AND XITI=X[S]-1)
      {color all vertices in S\T with the same new color:
        }
    }
}

```

Fig (7) algorithm for graph coloring problem

3- Ant Colony Optimization (ACO)(Bullnheimer,1998)

The basic operation mode of an ACO algorithm is the following in each iteration, a population of ants gradually and concurrently build solution to the problem according to transition rule which depends on the Heuristic and pheromone trail information available. Ants can release pheromone while **building** the solutions (on line step by step pheromone trail updating). Once they have been generated and evaluated, on line delayed pheromone trail updating positively reinforcing the edges traveled with an amount of pheromone directly dependent on the solution quality or both. Then all the pheromone trail suffer from evaporation. Moreover some learning actions can be performed from global perspective, such as observing the quality of all the solution generated and updating an additional Pheromone trail only in some of them or applying local search procedure to the solutions generated by the ant and depositing additional Pheromone. In both cases, the daemon replace the online delayed Pheromone updating and the process is called **Offline Pheromone trail updating**.

Implement the algorithm ACS (Ant colony / system) as follows:-

***Transition rule**. The destination node S is chosen as follows:

$$S = \begin{cases} \arg \max_{u \in J_k(r)} \{ [T_{ru}]^\alpha \cdot [\eta_{ru}]^\beta \} & , \text{if } q < q_0 \\ S & , \text{otherwise} \end{cases} \quad \text{--(1)}$$

with q being a random value uniformly in $[0, 1]$, $q_0 \in [0, 1]$ being a parameter defining the balance exploitation-biased exploration, and with S being a random node selected according to the probability distribution given by the AS (Ant System) transition rule:-

Transition rule (AS): The destination node S for an ant K located in node r is randomly chosen according to the following probability distribution:-

$$p_k(r, s) = \begin{cases} \frac{[T_{rs}]^\alpha \cdot [\eta_{rs}]^\beta}{\sum_{u \in j_k(r)} [T_{ru}]^\alpha \cdot [\eta_{ru}]^\beta} & , \text{if } s \in j_k(r) \\ 0 & , \text{otherwise} \end{cases} \quad \text{--(2)}$$

with T_{rs} being the pheromone trail of edge (r,s) , η_{rs} being the heuristic value, $j_k(r)$ being the set of nodes that remain to be visited by ant k , and

With a and B being parameter weighting the relative importance of pheromone trail and heuristic information.

"Online step-by-step updating rule :Each time an ant travel all edge .it is made in the way.

$$T_{rs} \leftarrow (1 - P).T_{rs} + p.\Delta T_{rs}$$

in this research $\Delta T_{ij} = T_0$

"**Offline pheromone updating** :in this case ,the deposit of pheromone is done by the daemone only considering a single ant the one who generated the global best solution ,S_{global-best}:

$$T_{rs} \leftarrow (1 - P).T_{rs} + p.\Delta T_{rs} \quad \text{where}$$

$$\Delta T_{rs} = \begin{cases} f(C(S_{global-best})), & \text{if } (r, s) \in S_{global-best} \\ 0, & \text{otherwise} \end{cases} \quad \text{---(3)}$$

There are many different ways to translate the above principle into a computational system to solve **graph coloring problem** in our *colony system* (ACS) an artificial ant k in node r choose the node s to move to among those which do not belong to its working memory lk by applying the following ACO algorithm for graph coloring problem.

5-ACO ALGORITHM FOR GRAPH COLORING PROBLEM

we implement the algorithm on the graph coloring problem ,we must acquire the main condition that neighboring nodes not is at the same coloring. we mean that lk contains all the neighboring of the ant k .

-Give an initial pheromone value T_0 to each edge. 2-for $k= 1$ to m do

(in parallel){ k is the no. of nodes}

*place ant k in an initial node r g. { r is "the color of the graph"} * include r in lk (tabue list of ant k keeping a record of he visited nodes).

* while ($am k$ not in target node) do

-select the next node to visit , $S \sim lk$ according to transition rule.

include S in lk . { S is the another color of the graph}

optional :online step-by-step updating of the pheromone trail T_{rs} of the traveled edge

- 3- **optional:** for $k > 1$ to m do
 "Evaluate the solution generated by ant k .Sr.
 *for each edge $(r,s) \ll S_i$..apply the online delayed
 pheromone trail updating rule. 4-
 Evaporate pheromone.
 5- optional. perform the daemon actions.
 6-it~stop condition is satisfied) then give the global best solution found as output and
 stop ,Else go to step 2.

6-RESULTS AND PARAMETERS

the parameters considered here are those the effect dire' I.' he computation of the
 probability of the numbers of colors in the graph:

- α :the relative importance of trail, $\alpha \in [20]$;
- β :the relative importance of visibility. $\beta \in [20]$;
- P :trail persistence. $0 \leq p \leq 1$ ($1-p$ can be interpreted as trail
 evaporation);
- Q :a constant related to the quantity of trail laid b (ant .

The number m of ants has always $b \geq n$
 nodes in the graph.

On 10 randomly generated graphs of 100 nodes (vertices). We tested several values for
 each parameters in order to achieve some statistical information about the
 average evolution, the default value of the parameters was ($\alpha = 1, \beta = 1, p = 0.5, Q = 100$),
 in each experiment only one of the values was changed ,except for α and β
 ,which have been tested over different sets values, the values tested over
 different sets values , the values tested were $\alpha \in \{0.0.5.1. \sim .5\}$. $\beta \in \{0.1.2.5\}$.
 $p \in \{0.3.0.5.0.7.0.9,0.999\}$ }and $Q \in \{1.1 00 .1000\}$ we run the algorithm ten times
 using the best parameters set .results are shown in table I .parameter Q is not
 shown because its influence to be legible

	Best parameters	Average result	Best result
ACO	$\alpha = 1, \beta = 5, \rho = 0.99;$	15.05	14.05

SO, the future works can be used to the hybrid techniques between genetic and ant algorithm to
 achieve the best optimization

7-CONCLUSION

The key to the application of ACO to the graph coloring problem is to identify an appropriate representation for the problem (to be represented as a graph searched by many artificial Ants), and an appropriate heuristic that defines the vertices in the ants mediated by the pheromone trail deposited on the graph edges will generate good, and often optimal problem solutions. We applied ACO to the graph to be colored

We saw that algorithmic must trail and error to achieve the best parameters for the (GCP), This mean that we can't access the best solution at first application (If ACO, we take time, so in the future work, the best parameters by using GA (genetic algorithm) then application of ACO.

REFERENCES

- 1- Assefaw Hadish Gebremedhin, (1999), " Parallel Graph Coloring:", Thesis, Candidatus Scientiarum Dept. of Informatics, University of Bergen, Norway.
- 2- Bullnheimer, B., Hartl, R. F., & Strauss, C. (1988), " Applying the Ant System to the Vehicle Routing Problem", in S. Voss., S. Martello, I. H. Osman, & C. Roucairol (Eds.), (pp. 109 -120), Boston, Kluwer.
- 3- Coleman T. F. and J. More, (1983), "Estimation of sparse Jacobian Matrices and Graph Coloring Problems ", SIAM J. Numer. Anal., 20 (1),187.
- 4- David Eppstein, (2003), " Small Maximal Independent Sets and Faster Exact Graph Coloring" , Dept., of Information and Computer Science University of California.
- 5- Dorigo, M, (1992), " Optimization, Learning and Natural Algorithms Unpublished Doctoral Dissertation ", Politecnico di Milano, Dipartimento Elettronica, Italy.
- 6- Dorigo, M, ((1993) , " Parallel Ant System', An experimental study, Unpublished Manuscript.
- 7- Dorigo, M.,"& Di Caro, G.(1999), " The Ant Colony Optimization Meta- Heuristic", UK, McGraw-Hill.
- 8- Dorigo Marco, (1996), " Ant Colonies for the Traveling Salesman Problem", IRIDIA, University Libre de Bruxelles, Aveue Franklin Roosevelt 50, CP 194/6, 1050, Bruxelles, Belguim, Mdorigo @ulb.ac.ab.
- 9- Finocchi, A, Paconesi, and R. Sivestri, (2002), " Experimental analysis of simple, distributed Vertex Coloring algorithms", in Proceedings of the Thirteenth ACVM-SIAM on Discrete of the (SODA 02), San Francisco.
- 10- S . M. Bishop, " Neural Networks for Pattern Recognition" (1995), Oxford University Press,
- 11- J. Hertz, A. Krogh, and R. G. Palmer,(1991), "Introduction to the Theory of Neural Computation', Addison Wesley.
- 12- D. B. Fogel, (1995), " Evolutionary Computation ", (1995), IEEE Press.
- 13- D. E. Goldberg, (1989), "Genetic Algorithms in Search,