International Journal of Advanced Technology & Engineering Research (IJATER)
www.ijater.com

ANT COLONY OPTIMIZATION:EFFICIENT WAY TO FIND SHORTEST PATH

Kapil Agrawal, Dept. of Computer Science, Jagannath University, Jaipur, India, kapilagrawal.408@gmail.com
Renu Bagoria, Dept. of Computer Science, Jagannath University, Jaipur, India, renu.bagoria@jagannathuniversity.org

Abstract

We describe an artificial ant colony capable of finding the shortest path. Ants of the artificial colony are able to generate successively shorter feasible path by using information accumulated in the form of a pheromone trail deposited on the edges of the graph. We describe the problem of Dijkstra's algorithm that solves the single-source shortest-path problem when all edges have non-negative weight. We try to solve that problem using Ant colony optimization. Ant colony optimization is already used in too many areas from graph related problems to the medical problem and study of Genomics.

Index Terms—Ant colony optimization, Dijkstra Algorithm, Finding shortest Path using Ant Colony optimization

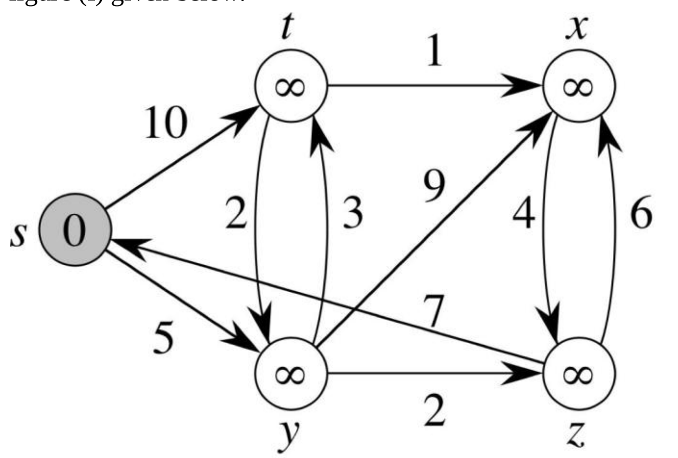
I. INTRODUCTION TO DIJKSTRA'S ALGORITHM

Djikstra's algorithm solves the problem of finding the shortest path from a point in a graph to a destination. It figure out that one can find the shortest paths from a given source to all points in a graph in the same time, hence this problem is sometimes called the single-source shortest paths problem. It is greedy algorithm that starts at the source vertex, s, it grows a tree, T, that ultimately spans all vertices reachable from S. Vertices are added to T in order of distance i.e., first S, then the vertex closest to S, then the next closest, and so on

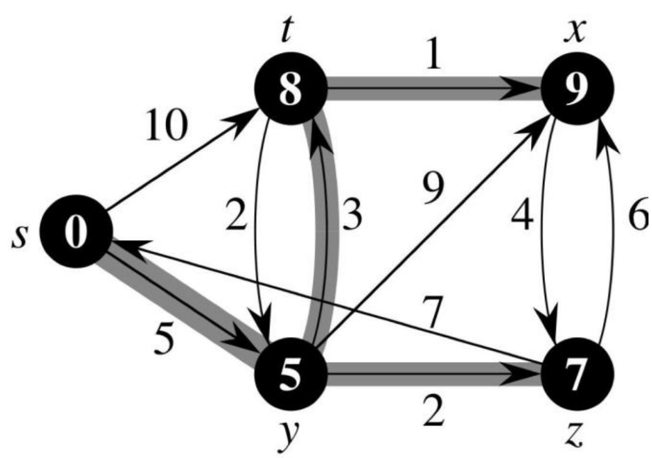
```
dist[s]←0
        (distance to source vertex is zero)
for all v∈V–{s}
        do dist[v]←∞
        (set all other distances to infinity)
S←∅
        (S, the set of visited vertices is initially empty)
Q←V
        (Q, the queue initially contains all vertices)
while Q ≠∅
        (while the queue is not empty)
        do u←mindistance(Q,dist)
(select the element of Q with the min. distance)
        S←S∪{u}

(add u to list of visited vertices)
        for all v∈neighbors[u]
                do if dist[v] > dist[u] + w(u, v)
                (if new shortest path found)
                then d[v]←d[u] + w(u, v)
        (set new value of shortest path)
        (if desired, add traceback code)
return dist
```

To demonstrate Dijkstra's algorithm look at the example figure (I) given below.



(i)



(ii)

We have source vertex s and out task is to find out the shortest path to all vertexes from source vertex s.

Start from source vertex s and look at the the neighbors of s. from s to y it costs 5, while s to t it costs 10. so pick up path from s to y. Now vertex y have three neighbors that are z with cost 2, x with cost 9 and t with cost 3. Cost to reach at z from s via y is 7 while from other nodes it is higher so pick up path {s,y,z} with cost 7. Applying the same algorithm to reach at each vertex and will get final result as show in figure (ii).

Dijkstra's algorithm runs in O(|E|lg|V|) time.

II. INTRODUCTION TO ANT COLONY OPTIMIZATION

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. An analogy with the way ant colonies functions has suggested the definition of new computing paradigm that is Ant System. The main characteristic of this model are

Positive feedback that accounts for rapid discovery of new solutions

Distributed computation to avoids for premature convergence and

The use of constructive greedy heuristic that helps find acceptable solutions in the early stage of the search process.

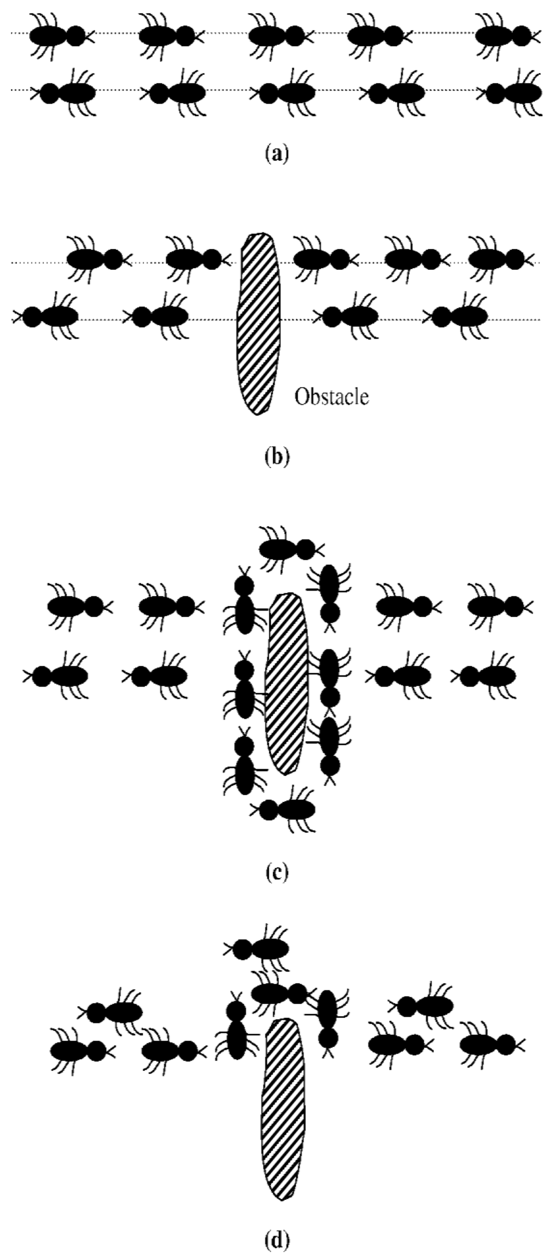
There are three ideas from natural ant behavior that we have transferred to our artificial ant colony:

(i) the preference for paths with a high pheromone level,

(ii) the higher rate of growth of the amount of pheromone on shorter paths

(iii) the trail mediated communication among ants.

In the (a) part of figure given below 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 a certain amount of pheromone while walking, and each ant probabilistically prefers to follow a direction rich in pheromone. While in (b) 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. In (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 to 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. In (d) 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 behavior .



(iii)

III. EXISTING WORK IN ANT COLONY OPTIMIZATION

Too many research take place in last 20 year in ant colony. Some area of applications for Ant Colony optimization are

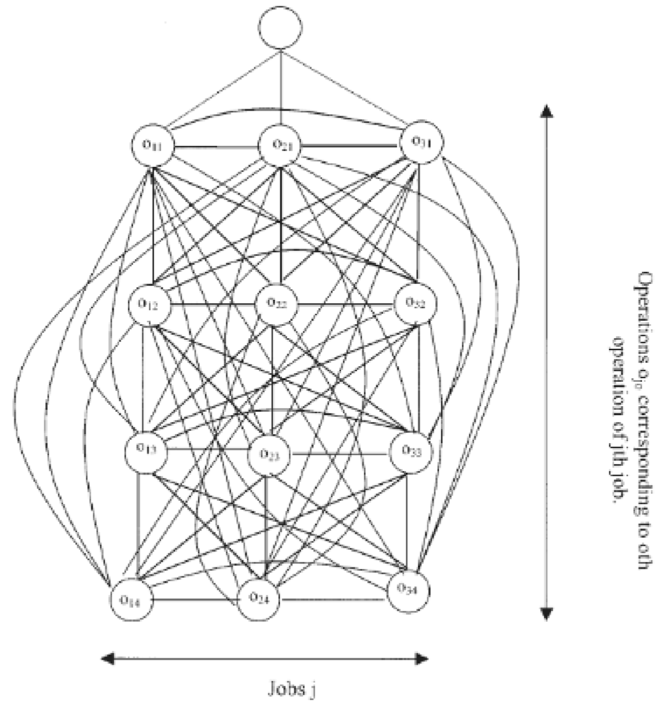
1. colony used to solve the well known traveling salesman problem (TSP) in which each ant gets a start city. Beginning from this city, the ant chooses the next city according to algorithm rules. After visiting all customer cities exactly once, the ant returns to the start city. The ants might travel concurrently or in sequence. Each ant deposits some amount of pheromone on his path. The amount of pheromone depends on the quality of the ant's path: a shorter path usually results in a greater amount of pheromone.

2. Feature Selection (FS) is extensive and spread across many fields, including document classification, data mining, object recognition, biometrics, remote sensing and computer vision

3. The ACORSES algorithm (ACO based algorithm) for solving Area Traffic Control problem differs from approaches in that new ant colony is generated at each cycle with the assistance of the best solution of the previous information. Moreover, the best solution that is obtained from the previous evaluation is saved to prevent being trapped in bad local optimum.

4. Ant is also used to solve scheduling problem for flexible manufacturing systems. To do so, A directed graph-based approach is used to represent the whole process. Ants move from one node to another node and progressively move towards the final node.

5. Ant colony is also used for Traffic congestion forecasting algorithm. A road agent installed at each intersection coordinates with its neighboring road agents based on the pheromone-communications model to adaptively respond to dynamically arising congestion and forecasts congestion a few minutes ahead.



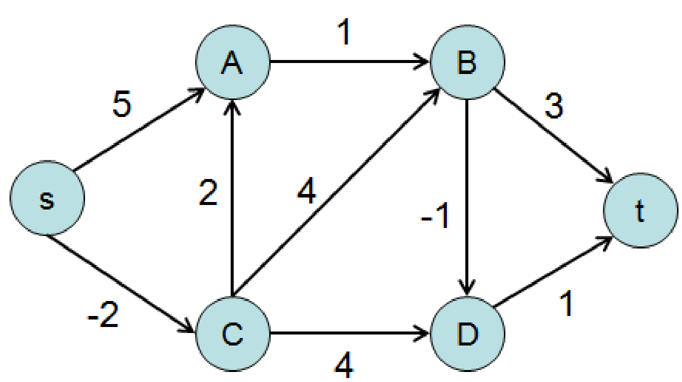
(iv)

IV. PROBLEM WITH DIJKSTRA'S ALGORITHM

Although dijkstra's algorithm is good to find single source shortest path to all vertices but it also have some shortcomings that are

1. Dijkstra's algorithm do not work with negative weights

2. Dijkstra's algorithm do not apply for graphs that have negative cycles like in the figure below we have negative cycle and negative weight also.



(v)

3. The major disadvantage of Dijkstra's algorithm is the fact that it does a blind search by there by consuming a lot of time waste of necessary resources.

4. Dijkstra's algorithm have complexity O(|E|lg|V|) where E is set of edges and V is set of vertices.

V. CONCLUSION

In that paper have mainly focused on one particular limitation of Dijkstra's algorithm that is complexity. It depends upon the edges and vertices of the graph. So far we have seen so many applications of ant colony optimization, TSP is an example that depicts that ant colony optimization is useful in solving graph related problem. With the help of Ant colony optimization algorithm we improve Dijkstra's algorithm by reducing the time complexity. We try to improve execution time of Dijkstra's algorithm.

REFERENCES

[1] Marco Dorigo, Luca Maria Gambardella "Ant colonies for the traveling salesman problem" TR/IRIDIA/1996-3

[2] Melissa Yan Dijkstra's algorithm

[3] Merin Puthuparampil Report Dijkstra’s Algorithm

[4] http://www.cs.indiana.edu/~achauhan//Teaching/B403/LectureNotes/10-graphalgo.html

[5] Graph Algorithms: Introduction to Algorithm design and Analysis

[6] Ivan Brezina Jr., Zuzana Čičková "Solving the Travelling Salesman Problem Using the Ant Colony Optimization"

[7] Hamidreza Rashidy Kanan, Karim Faez and Mehdi Hosseinzadeh "Face Recognition System Using Ant Colony Optimization-Based Selected Features"

[8] Soner Haldenbilen, Ozgur Baskan and Cenk Ozan "An Ant Colony Optimization Algorithm for Area Traffic Control"

[9] R Kumar, M K Tiwari and R Shankar2 "Scheduling of fexible manufacturing systems: an ant colony optimization approach"

[10] Satoshi Kurihara "Traffic-Congestion Forecasting Algorithm Based on Pheromone Communication Model Pheromone Communication Model"

[11] http://www.csupomona.edu/~ftang/courses/CS241/notes/graph%20algorithms%201.htm