COMPARISON OF DIFFERENT HEURISTIC, METAHEURISTIC, NATURE BASED OPTIMIZATION ALGORITHMS FOR TRAVELLING SALESMAN PROBLEM SOLUTION

[1]KIRTI PANDEY, [2]PALLAVI JAIN

[1]Shri Vaishnav Institute of Technology & Science, Indore, India
[2]Department of Computer Science Engineering, SVITS, Indore, India
Email: pandey.kirti30@gmail.com, pallavijain02@gmail.com

Abstract- The Traveling Salesman Problem (TSP) is widely studied problems in combinatorial optimization. Many researches were made for that type of problem. This problem is known to be the NP- hard problems that cannot be solve by conventional mathematical approach. It is also a sub problem of many areas. Different approaches are performing on it as a general test bench to find solution for NP-had problem. In this paper we study seven approaches like Genetic algorithm, Ant colony optimization, Simulated annealing, Tabu search algorithm, Self organize migration algorithm, Intelligent Water Drop and Memetic algorithm which used previously to get a solution which is optimal and comparison of these algorithm to determine certain aspects like nature of algorithm, time, number of iteration to solve this problem. Routing is always an issue in today's world where a time efficient research is going onto it. Also we are going to present a new concept which is Genetic algorithm based, and it will give the solution to get best route in this problem within efficient time without repeating nodes.

Keywords:-TSP, NP-Hard problem, Routing, Combinatorial optimization.

I. INTRODUCTION

Generally in TSP, a map of cities is given to the salesman and he has to visit all the cities only once to complete a tour such that the tour length is the shortest among all possible tours for this map of city. The weights assigned to the edges of a finite complete graph G (V, E) and the objective is to find the best route to get time efficient result.
Travelling salesman problem need an efficient solution procedure to get optimized solution. One can either use heuristic, meta-heuristic or natural inspired algorithm to solve this problem with no any knowledge that particular solution given by algorithm is require result or not. In this research paper Genetic algorithm, Ant colony optimization, Simulated annealing, Tabu search algorithm, Self organize map algorithm, Intelligent Water Drop and mimetic algorithm will be describe .Mechanism of each algorithm is also explain and comparison among all above is done with table representation in terms of number of cities, cost ,time ,tour length, number of iteration .
This research is based on some other literature that proposed by other researcher.

As the development of research method, heuristics algorithm, meta-heuristics algorithm and natural inspired algorithm

This research is based on some other literature that proposed by other researcher.
As the development of research method, heuristics algorithm, meta-heuristics algorithm and natural inspired algorithm was introduced to improve heuristics algorithm

II. RECENT SCENARIO

1. Genetic Algorithm

Genetic algorithm is different than conventional algorithm. It is heuristic search method. It is proposed in 1960 by John Holland. This algorithm is start with initial solution by selecting population randomly. Each element of population is called chromosome. Chromosome is a string of symbol. Chromosomes themselves are composed of genes. The real value of a control parameter, encoded in a gene, is called an allele.

Working of Genetic algorithm:
Assuming an initial random population produced and evaluated, genetic evolution takes place by means of basic genetic operators:
1) Parent selection;
2) Crossover;
3) Mutation.
4) Termination

GA uses three operations to maintain population that evolve from one generation to another. The first operation is "Selection" operation which is inspired by the principle of 'Survival of the Fittest'. The search begins from a randomly generated population that evolves over successive generations (iterations).A of the solutions. Each time two solutions are chosen as parent solutions by selection process based on fitness function. The second operation is the "Crossover" operation, which is inspired by mating in biological populations. The crossover operator inherits features of good surviving designs from the parent population into the future population, which will have better fitness value on

average. The third operation is "Mutation" which causes diversity in population characteristics. It causes local modifications to the new generation randomly. The new generation is identical to the parent except one or more changes made by mutation process. Repeat selection, crossover and mutation operations to produce more new solutions until the population size of the new generation is the same as that of the old one. The iteration then starts from the new population. Since better solutions have a larger probability to be selected for crossover and the new solutions produced carry the features of their parents, it is hoped that the new generation will be better than the old one. The procedure continues until the number of generations is reached to n or the solution quality cannot be easily improved.

Pseudo Code of Genetic Algorithm

```
begin procedure GA
generate populations and fitness function
evaluate population
while (termination criteria not meet)
{
{
while (best solution not meet)
{
Crossover
mutation
evaluate
}
}
post-process results and output
end GA procedure
```

2. Ant Colony Optimization

The ACO is inspired by the food search behavior of real ants and their ability in finding the optimum paths. It is a population-based general search technique for the solution of difficult combinatorial optimization problems other ants follow one of the paths at random, also laying pheromone trails. Since the ants on the shortest path lay pheromone trails faster, this path gets reinforced with more pheromone, making it more appealing to future ants. The ants become increasingly likely to follow the shortest path since it is constantly reinforced with a larger amount of pheromones [8]. The pheromone trails of the longer paths evaporate.

Scheme:

Construct ant solutions

Define attractiveness τ, based on experience from previous solutions

Define specific visibility function, η, for a given problem (e.g. distance)

Pseudo Code of Ant Colony Optimization

```
Initialize the base attractiveness, τ, and visibility, η, for each edge;
for i < IterationMax do:
for each ant do:
choose probabilistically (based on previous equation) the next state to move
into;
add that move to the tabu list for each ant;
repeat until each ant completed a solution;
end;
for each ant that completed a solution do:
update attractiveness τ for each edge that the ant traversed;
end;
if (local best solution better than global solution)
save local best solution as global solution;
end;
end;
```

III. SIMULATED ANNEALING ALGORITHM

Simulated Annealing (SA) is the oldest probabilistic meta-heuristic algorithm and one of the first algorithms having ability to avoid being trapped in local minima. It was first presented by Kirkpatrick, Gelatt and Vecchi in 1983.

A. Inspiration

Simulated Annealing is inspired by the process of annealing in metallurgy. In this process a material is heated and slowly cooled into solid crystal state with minimum energy and larger crystal size to reduce defects in metallic structures. The heat increases the energy of the atoms allowing them to move freely, and the slow cooling schedule allows a new low energy level to be discovered. This annealing process requires the careful control of temperature and cooling rate.

B. Metaphor

Each set of a solution represents a different internal energy of the system. Heating the system, results in a relaxation of the acceptance criteria of the samples taken from the search space. As the system is cooled, the acceptance criteria of samples are narrowed to focus on improving movements. Once the system has cooled, the configuration will represent a sample at or close to a global optimum.

Pseudo Code of Simulated Annealing

```
Procedure SA
begin
iteration t 0
initial value of temperature Trandom choice of xc
evaluation of xc
repeat
repeat
choice xn from neighbors of xc using assumed way
if f (xn) <f (xc) {the new tour is shortest} then xc xn
else if random [0,1]<exp{−(f(xn)−f(xc))/T }
then xc xn
until (end condition)
T g (T, t)
```

t t+1
until (stop criterion)
end

In the above pseudo code of SA 'end condition' allows to stop the internal loop when temperature equilibrium is reached, which means that probability distribution of choosing a new solution is close to the Boltzman distribution. SA often assumes

IV. INTELLIGENT WATER DROPS (IWD)

The IWDs algorithm is based on the natural water drops in the rivers as explain before. Flowing Waters will denote as intelligent water drops in this algorithm. Each of the intelligent water drops will have Amount of soil and the visited city which. Path in a river also contains some soil in the riverbed, this soil denoted as where i is start of a path and j is the end of a path.IWD algorithm need graph as the input problem, this means the problem have to represent in graph. Iteration in IWD algorithm is defined by user, which in each iteration IWD will have a tour to visit the entire city (each IWD will have their own iteration). Below is the IWD algorithm that specified by Hamed Shah-Hosseini

Pseudo Code of Intelligent Water Drops
1. Initialization of static parameters.
2. Initialization of dynamic parameters.
3. Spread the IWDs randomly on the nodes of the graph as their first visited nodes.
4. Add the node just visited to visited node list of each IWD.
5. Repeat steps 5.1 to 5.4 for those IWD with partial solution.
5.1For the IWD residing in node i, choose the next node j, which does not
violate any constraints of the problem and is not in the visited node
list of the IWD.
5.2For each IWD moving from node i to node j, update its velocity.
5.3Compute the soil.
5.4Update the path soil and IWD soil.
6. Find the current iteration-best solution.
7. Update soils on the path of current iteration-best solution.
8. Update the total-best solution by the current iteration-best solution if the
current iteration-best solution is better.
9. Increment the iteration counter.
10. Stops with total-best solution

5. Tabu Search

Tabu search (TS) is a neighborhood search method which employs "intelligent" search and flexible memory technique to avoid being trapped at local optimum. Speed up the search process. Moves are selected intelligently and Use tabus to restrict the search space and avoid cyclic behavior. Glover (1986) introduces tabu search as a "metaheuristic" superimposed on another heuristic. Glover (1989) provides a full description of the method.

Pseudo Code of Tabu search
Step 1 (Initialization)
(A) Select a starting solution **x**now □ **X**.
(B) **x**best = **x**now, best_cost = **c**(**x**best).
(C) Set the history record H empty.
Step 2 (Choice and termination)
Determine Candidate_N (**x**now) as a subset of N (H, **x**now).
Select **x**next from Candidate_N (**x**now) to minimize c (H, **x**).
Terminate by a chosen iteration cut-off rule.
Step 3 (Update)
Re-set **x**now = **x**next.
If c (**x**now) < best_cost, perform Step 1(B).
Update the history record H.
Return to Step 2.

V. MEMETIC ALGORITHMS

Memetic Algorithms (MAs) were originally inspired by Models of adaptation in natural systems that combine evolutionary adaptation of population of individuals (GAs) with Individual learning (LS) within a lifetime (others consider the LS as development stage). Learning took the form of (problem specific) local search. Dawkin's concept of a meme which represents a unit of cultural evolution that can exhibit refinement hence the local search can be adaptive. Inspired by both Darwinian principles of natural evolution and Dawkins' notion of a meme, the term "Memetic Algorithm" (MA) was introduced by Moscato in his technical report in 1989 where he viewed MA as being close to a form of population-based hybrid genetic algorithm (GA) coupled with an individual learning procedure capable of performing local refinements.

Pseudo Code of Memetic search algorithm
Begin
initialize population;
for each individual do local-search individual;
repeat
for individual =1 to #crossovers do
select two parent individual1, individual2 \in
population randomly;
individual3:=crossover (individual1, individual2);
individual3:=local-search (individual3);
add individual i3 to population;
end for;
for individual=1 to #mutations do
select an individual of population randomly;
individual {m}:= mutate (individual);
individual {m}:= local-search (individual {m});

add individual {m} to population;
end for;
population: =select (population);
if population converged then
for each individual of bestpopulations do individual:
=local-search (mutate (individual);
endif
until terminate=true;
end

VI. SELF ORGANIZE MIGRATION ALGORITHM (SOMA)

The problem was first formulated as a mathematical problem in 1930 and is one of the most intensively studied problems in optimization. It is used as a benchmark for many optimization methods. Even though the problem is computationally difficult, a large number of heuristics and exact methods are known, so that some instances with tens of thousands of cities can be solved. The input data (a set of n WDS) are presented to the SOMA in a random order and a competition based on Euclidian distance is performed between neurons in the ring. The winner neuron is the neuron I with the minimum distance to the presenting city.

Pseudo Code of Memetic Self Organize Migrating Algorithm

Generate a random initial population, pop
While termination criteria not met
Compute fitness of each individual
Select leader (individual with highest fitness)
For each individual (i) of the population,
While not path length
Generate PRT_vector as illustrated in equation 9
Update position of particle as given in equation 10
Update fitness value
Select a new leader
End while
End while

Benefits and Limitation of previously studied papers:

(i) Genetic algorithm which is the Evolutionary Algorithm and it is behaves as a base of most of algorithm so solve such type of problem. It is easily mix with other algorithms to get better solution. When we have poor or no mathematical analysis is available it is more useful. It is having poor search capability for both discrete and continuous variables. But with these good qualities it also having limitation because it converges at local minima and dynamic change in particular problem is not accepts by this algorithm.

(ii) Ant colony optimization algorithm is best for distributed computing. It is robust type of algorithm and it is easily mixes with other algorithm to provide best solution. It is better algorithm than Simulated Annealing and Genetic Algorithm for Solution of Travelling Salesman Problem when Graph changes dynamically and it is able to change in real time. But some time it converse at local optimal solution.

(iii) Simulated Annealing is one of the oldest algorithm [12] which is not trapped at local minima. It is statically in nature. It is easy to code complex problem. Non linear ability of global optimality makes it versatile in nature. But it is time consuming algorithm. To get cost for particular problem it needs more computation. Unlike above algorithm it is not easily accumulate with other algorithms. Time need to execute this algorithm is inversely proportional quality of solution.

(iv) Tabu search algorithm is stochastic in nature, guaranteed to give optimal result always but it takes time that is time complexity is high. This algorithm is act like anticycling mechanism.

(v) Self organize migrating algorithm non linear programming algorithm. It give fast result initially after that become slow but having less complexity overall that is time is not go exponentially in it. Feasible solution is given by this algorithm[7]. Its main limitation is that its computation is expensive and it is not global in nature.

(vi) Memetic algorithm is having more iteration than genetic algorithm. It is hyper heuristic algorithm. But is efficient and better algorithm, safe because guaranteed to converge to an optimal result with reasonable amount of iteration regardless to initial population[6]. It is more accurate algorithm than tabu search algorithm.

(vii) Intelligent Water Drops algorithm is metaheuristic algorithm and used for continuous optimization. It is static and dynamic both in nature and gives more reachable solution for any problem. it is better than ant colony optimization in terms of time for computation it tales less time for more population.

Comparison Of Above Algorithm For The Solution Of Travelling Salesman Problem By Considering Following Parameter.

Table (1)

Algorithm name	Nature of algorithm	Time	Convergence solution	Variable type	Number of iteration	heuristic
Genetic algorithm	Static	more	Local minima	Linear and non linear	high	heuristic
Self organize migation algorithm	Static	Less	Global minima	Non linear	More than other six algorithm	heuristic
Ant colony optimization	Dynamic	more	Local minima	Linear and non linear	Less than Genetic algorithm	Meta heuristic
Intelligent water drop	Static and Dynamic	less	Global minima	linear	Less than above three algorithm	Meta heuristic
Tabu search	stochastic	high	Global minima (not always gives)	Non linear	More	Meta heuristic
Memetic algorithm	Static	more	Global minima	Linear and non linear	More	Hyper heuristic
Simulated annealing	Static	more	Global minima	Non linear	More	Meta heuristic

VII. PROPOSED METHOD

In proposed method we developed an algorithm which is based on modified Genetic algorithm. Each algorithm which I studied in previous papers is either time efficient or gives the optimal solution. So in our propose paper we uses combination of any two above algorithms which embedded other new approach to solve the problem in efficient time with less number of iterations and give the best result. Our main focus is on reducing the problem of repetition of node in route finding for travelling salesman problem.

Outline of proposed Approach:

- Start with taking number of nodes, edges and weights assign to each edge joining nodes.
- Make the graph representation of problem.
- Find the sub graph of given problem by new approach.
- Applying genetic algorithm on sub graph with the help of tabu search algorithm to find the best route.

CONCLUSION

In this paper, different algorithm compared to benchmark the performance. According to study simulated annealing and tabu search algorithm are derivative free algorithm. Simulated annealing [10] is probabilistic metaheuristic algorithm and work on global optimization. Simulated annealing having slow convergence, it is good for non linear and linear variables but Ant Colony optimization is better than simulated annealing because of its slow convergence. Tabu search having less number of moves than simulated annealing hence having low cost as compare to simulated annealing. Half time taken by tabu search than simulated annealing but tabu some time not guaranteed to find optimal solution. Intelligent water drop algorithm is better than ant colony optimization because it is a fast algorithm.memetic algorithm provide higher success rate than tabu search.IWD algorithm has better performance in time/duration rather than ACO in finding the optimal solution as the numbers of cities growth. IWD algorithm sometimes cannot find the optimal solution; this is because of the random probability equation that used to find the next city to visit. In future we can mix above algorithms like Genetic algorithm and other above to get the solution of problems which are face by them individually.

REFRENCES

[1] "A comparative study of a class of Nature Inspired Algorithms", Proceedings of the 5th national conference: INDIA Com, March 10-11, 2011

[2] N. Sureja, B. Chawda, "Random Travelling Salesman problem using Genetic Algorithms," IFRSA"sInternational Journal Of Computing, Vol. 2, issue 2, April 2012

[3] M. Dorigo, V. Maniezzo, A. Colomi, "The Ant System: Optimization by a colony of cooperating agents", IEEE Transactions on Systems, Man and Cybematics-Part B,Vol. 26, No.1, 1996.

[4] Binitha, S.S. Siva sathya, "A survey of bio inspired optimization algorithm", IJSCE, Vol-2, issue-2, May 2012 .

[5] Zaid Ameen Abduljabbar, Mustafa S. Khalefa and Marzanah A. Jabar,"Comparison between Ant Colony and Genetic Algorithm Using Traveling Salesman Problem", International Journal of Soft Computing, Vol- 8, Year 2013.

[6] Č Zuzana ,B.Ivan, P.Juraj," Alternative Method for Solving Traveling Salesman Problem by Evolutionary Algorithm", Management Information Systems,Vol. 3 (2008), No. 1. and Telecommunications (SETIT), pp.639-642, March 2012.

[7] Gaifang Dong, William W. Guo," A Cooperative Ant Colony System and Genetic Algorithm for TSPs",Advances in Swarm IntelligenceLecture Notes in Computer Science Volume 6145, pp 597-604,2010.

[8] Osaba, E. Diaz, F.," Comparison of a memetic algorithm and a tabu search algorithm for the traveling salesman problem"IEEE Computer Science and Information Systems (FedCSIS), 2012 Federated Conference, 9 Sept. 2012.

[9] FFred Glover†, James P. Kelly, Manuel Laguna," Genetic algorithms and tabu search: Hybrids for optimization", Computers & Operations Research Volume 22, Issue 1, January 1995, Pages 111–134

[10] Jin-kao Hao , Jerôme Pannier," simulated Annealing and Tabu Search for Constraint Solving" Proceedings of the fifth international CONFERENCE in 1998.

[11] Shujia Liu,"A Powerful Genetic Algorithm for Travelling Salesman Problem" Sun Yat-sen University, Guangzhou 510006 China.

★★★