IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
PP 63-67

Travelling Salesman Problem using Differential Evolutionary Algorithm

Vanita G.Tonge
[1]ARajiv Gandhi College of Engineering Research and Technology

Abstract: *This paper presents an efficient and powerful population based algorithm that is Differential evolutionary algorithm which has been successfully apply on travelling salesman problem. The travelling salesman problem is one of the large classes of "NP Hard "optimization problem. It has applications in science and engineering field. An efficient solution to this problem reduces travelling costs and the objective of this problem is based on the applications used. For the travelling salesman problem shortest distance is an objective function. In this paper we have used recursive DE algorithm to implement the travelling salesman problem. Recursive DE improves the time complexity of algorithm and gives improved result.*
Keywords: *Differential Evolution, TSP, Mutation, Crossover operator, Optimization.*

I. Introduction

The Travelling Salesman problem (TSP) or Hamiltonian tour is a type of classic old problem and one of the benchmark in Computer Science and Operations Research. Heuristic optimization algorithms such as genetic algorithm, tabu search and ant colony optimization have been widely used to find optimal solution in travelling salesman problem. Standard GA have a good performance for finding the promising regions of the search space, they are not so successful at determining local minimum in terms of convergence speed. In order to overcome these disadvantages of GA in numeric optimization problem, Differential evolution algorithm has been introduced by Storn and Price.

Differential evolution algorithm is a new heuristic approach mainly having three advantages; finding the true global minimum of a multi modal search space regardless of the initial parameter values, fast convergence, and using a few control parameters. DE algorithm is population based algorithm similar to GA .it consist similar operators like crossover, mutation and selection. There is one difference between DE and GA that DE is rely on mutation and GA is rely on crossover. The method of differential evolution can be applied to real-valued problems over a continuous space with much more ease than a genetic algorithm.

The idea behind the method of differential evolution is that the difference between two vectors yields a difference vector which can be used with a scaling factor to traverse the search space. As in the genetic algorithms beginning a random population is chosen, equally over the problem space, and to create the next generation. There are mainly three reasons why TSP has been attracted the attention of many researcher's and remains an active research area. First, a large number of real-world problems can be modelled by TSP. Second, it was proved to be NP-Complete problem [1]. Third, NP-Complete problems are intractable in the sense that no one has found any really efficient way of solving them for large problem size. Also, NP-complete problems are known to be more or less equivalent to each other; if one knew how to solve one of them one could solve the lot.

The problems which are NP-Complete are presumed to have no complete solutions which can be implemented in polynomial time and are therefore good candidates for evolutionary algorithms.Due to the NP-Complete nature of the N-Queen and travelling salesman problems, the differential evolution and genetic algorithm are used for optimizing the performance of the problems. Furthermore the interest in NP-Complete problems is due to the theoretical nature of the NP-Complete problem which states that any problem in NP can be reduced to an instance of an NP-Complete problem in polynomial time. This means that if we can find an efficient way to solve these combinatorial problems using evolutionary algorithms we should be able to apply these same algorithms to many different problems which have no fast algorithmic approach.[2]The paper is organized as follows. Section 2 contains a brief explanation of travelling salesman problem. Section 3 presents a basic of Differential Evolution algorithm. Section 4 describes the implementation of TSP problem using DE. Section 5 presents a simulation results and discussion.

II. Travelling Salesman Problem

The travelling salesman problem (TSP) has been an early proving ground for many approaches to combinatorial optimization, including classical local optimization techniques as well as many of the more recent variants on local optimization, such as simulated annealing, tabu search, neural networks, and genetic algorithms [1]. The travelling salesman problem consist one salesman and set of cities. The salesman has to visit every city starting from home town and back to the same city. The travelling salesman problem is one of the large classes of “NP Hard “optimization problem. It has applications in science and engineering field. For example, in the manufacture of a circuit board, it is important to determine the best order in which a laser will drill thousands of holes. An efficient solution to this problem reduces production costs for the manufacturer. It is defined a Hamilton circuit is a circuit that uses every vertex at once.

Obtaining a solution to the problem of a salesman visiting n cities while minimizing the total distance travelled is one of the most studied combinatorial optimization problems [10]. It is called combinatorial optimization because the optimum solution consists of a certain combination of variables from the finite pool of all possible variables e.g. optimal arrangement, grouping, ordering, or selection of discrete objects in finite number [11]. The TSP problem is well known to belong to the class of Non-deterministic Polynomial hard (NP-hard) problems [11]. It is described as follows:

TSP= {(G,f,t):G=(V,E) a complete graph,
f is a function($V \times V \rightarrow Z$)
t Є Z

G is a graph that contains a travelling salesman tour with cost that does not exceed t. V is a vertex and E represents edges. The objective function describes as follows set of n cities along with distance between them. A salesman starts and terminates his tour in home city but the rule is that he/she must visit a city at once and aim is to determine shortest path. Evaluation of objective function is completed in polynomial time thus TSP belongs to NP problem. The TSP can be defined on a complete undirected graph G=(V,E) if it is symmetric or on a directed graph G= (V,A) if it is asymmetric

III. Differential Evolutionary algorithm

“Evolutionary Computation (EC) uses ideas inspired from the behavior of community-based animals such as ants, bees and birds. Furthermore researchers proposed algorithms with variants of DE.”.[13] The differential evolution algorithm is a population based optimization tool. It is used by several researchers for various numerical problems to find optimal solution from a set of constraint. “The space complexity of DE is as compared to other optimization algorithm this feature is used to extend DE for large scale problems”[14]. It was introduced by Storn and Prince in 1996. The main steps of De algorithm is given as follows:

Initialization
Evolution
Repeat
Mutation
Recombination
Evolution
Selection
Until(termination criteria are met)

Fig.1 Procedural steps of DE algorithm

In the first step we initialize the population randomly. Initial population consists of a collection of individual. The size of the population is not fixed. It is user defined value and relies on a problem. Evolution means to implement the objective function of a problem. Find out the values for every individual and select one as best solution and called it as target object. Two out of three critical parameters associated with the original DE algorithm namely, CR and F are adaptively changed instead of taking fixed values to deal with different classes of problems [3]. Crossover, mutation and selection successively applied on the population. For each target vector a donar vector is generated. F is a scaling factor used in generation of donar vector. A target vector is mixed with donar vector to generate a trial vector here CR factor that is crossover factor is used to select one of vector between target and donar vector then selection is applied on trial and target vector based on fitness value . After selection population is updated and procedure is repeated until the criterion is met.

IV. Implementation of TSP problem using DE

Differential evolution algorithm is started with a set of solutions (chromosomes/individuals) called as population. Population is a randomly generated string of numbers which represent the number in a sequence, number represents the city. The objective of the TSP problem is to find an optimal tour such that cost of that tour is minimum.

4.1 Fitness function

The fitness function to calculate the tour is f(x) =1/d, where d is the distance of the tour represented by a chromosome. It is the total length of the tour from the first city to the last city moving according to the order of the city in a individual. If the cities are represented with x and y coordinates in 2D coordinate system, then we calculate the distance between them according to the equation [4]:

$$D= (x1-x2)^2+(y1-y2)^2 \quad (1)$$

4.2 Forward and backward Transformation

DE algorithm is only capable of handling continuous variables; we required to transfer data from discrete solution to continue solution and vice versa for the operation of the DE internal mutation schema. This approach was highly effective in solving a permutation flow shop scheduling problem. The forward transformation method transforms discrete variables to continuous variable for internal representation of vector value. The backward transformation values transform the continues variable into discrete value for evaluating the objective function

Selection

After deciding the method of encoding, the decision for selection technique is to be made. From various selection methods such as Roulette-wheel Selection [6], Tournament selection method [7] in this work tournament selection method is used. Selection is used to evaluate each individual and keep only the fittest ones among them. Selected individual and some less fir ones could be selected according to a small probability and others are removed from the current population [8]

4.3 Tournament selection operator

Tournament selection is one of the most popular selection operators. Tournaments are small competitions among the individuals. In tournament selection, an individual passes into the next generation if it is better fitted than opponents randomly selected from the population. Tournament size N_{tour} is the selection parameter [9].

Procedure : Tournament selection

While population size < pop_size
do
Set the tournament size, Ts
Pick Ts random individuals from the population
From those Ts individuals, pick one with the best fitness
If the same individual chosen as both parents, discard the second one
End While
Return chromosome with the best fitness among Ts chromosomes
End Procedure

The proposed DE algorithm in order to improve the solution is given below. The outline is given below.

Initial phase

1. Population Generation: An initial number of discrete trial solutions are generated for the initial population.
2. Evaluation of initial population is done and best one is selected as target vector based on fitness value.

Conversion

1. Forward Transformation: This conversion scheme transforms the parent solution into the required continuous solution.

DE Strategy

1. Tournament selection operator is used to select two random vectors for mutation. Mutation transforms the parent solution into Donar solution by using inbuilt mutation scheme.
2. Using crossover probability trial vector is generated out of target and donar vector
3. position based crossover operator is used to improve the fitness value.
4. Backward Transformation: This conversion schema transforms the continuous child solution into a discrete solution.

V. Experimentation

Numerical experiments have been provided in order to demonstrate the effectiveness and efficiency of the proposed DE approach. To solve the problem using the proposed DE, four important parameters have to consider .they are maximum generation, population size, crossover probability mutation probability. The encoding used for the differential evolution's approach to the traveling salesman problem also used arrays of permutations, however an additional level of abstraction was used to allow us to perform addition, subtraction and multiplication on permutations.

The generation of the donor vector uses DE/best/1 in which the most fit member of the population is chosen as Xbest, and two other members are chosen using tournament selection operator that is Xb and Xc,

$D_i = X_{best} + F(X_b - X_c)$. (2)

According to the function the numbers are all added, subtracted and multiplied by the scaling factor F with no restrictions. The generation of the trial vector is a little more complex. Starting with the element at index 0 in the array we choose the city, A, from the donor vector or the previous generation based on the CR as is typical for differential evolution. For each selection after that the selection process is the same, but we select the city which follows A in the donor vector or previous generation, depending on CR.

In the case that this city is already present in the 'child' we choose the city following that city. This ensures that we do not create an invalid permutation but also adds to the computational complexity of the differential evolution's method. The values for the Differential Evolution's variables were set to CR = 0.9 and F = 0.5 based on tests to determine the optimal settings for those parameters. The differential evolution approach did not suffer from premature convergence at all.

The DE algorithm were tested for population size (N), mutation factor (F), and cross over factor (CR). First, each value of N was tested with population sizes of 5, 10, 25, 100, 500, and 1000. For mutation factor and crossover factor values were consider as F=0.1, 0.5, 0.7 and CR=0.2, 0.5, 0.6, 0.8. Next, each combination of N size, CR and F was given 10 trial runs recording the number of generations, running-time, and whether the solution discovered an optimal solution. The number of generations and runtime numbers were averaged. In most accounts differential evolution proved to find an optimal solution in fewer generations. For the experimentation we have considered Burma 14 problem which is given below.

burma14
16.47 96.10 16.47 94.44 20.09 92.54 22.39 93.37 25.23 97.24 22.00 96.05 20.47 97.02 17.20 96.29 16.30 97.38 14.05 98.12 16.53 97.38 21.52 95.59 19.41 97.13 20.09 94.55

Population size(N)	Mutation Factor(F)	Crossover Factor(CR)	Max Generation	Optimal Value	Known Optimal Value
150	0.1	0.2	100	20	14
300	0.3	0.4	250	16	
500	0.5	0.6	500	18	
1000	0.7	0.8	650	15	
550	0.1	0.7	500	15	
600	0.1	0.6	500	15	
600	0.1	0.4	500	15	
600	0.1	0.9	500	16	
600	0.1	0.8	500	14	

VI. Conclusion

In this paper we have proposed Differential Evolutionary algorithm for travelling salesman problem. The travelling salesman problem is one of the large classes of NP hard problem. While executing a proposed algorithm experimental trial has been taken. Variation of crossover factor and Mutation factor are considered. We got optimal result with the value of crossover factor and Mutation factor is 0.1 and 0.8 respectively.

References

[1]. David S. Johnson,Lyle A.McGeoch ,"The Travelling salesman Problem: A case Study in Local Optimization".november 20,1995

[2]. Brian Hegerty, Chih-Cheng Hung, and Kristen Kasprak," A Comparative Study on Differential Evolution and Genetic Algorithms for Some Combinatorial Problems",

[3]. A.K.Quin,P.N.Suganthan,"Self-adaptive Differential Evolution Algorithm for Numerical Optimization"0-7803-9363-5/05/$20.00@2005IEEE.

[4]. Thomas H.Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein,"Introduction to Algorithms Second Edition", PHI Learning

[5]. Godfrey C. Onwubalu, B.V. labu, "New Optimization Techniques in Engineering", published in springer

[6]. Goldberg, D.E. (1989) "Genetic Algorithms in search, optimization machine learning", Pearson Education. Kdjksdk

[7]. Rajasekaran S.,VijayaLakshmi Pai C.A.,"Neural Networks, Fuzzy Logic and Genetic Algorithms Snthesis and Applications",Prentice-Hall of India.

[8]. Mitchell M. "An Introduction to Genetic Algorithms", Prentice-Hall of India.

[9]. Vladimir Filipovi´c," FINE-GRAINED TOURNAMENT SELECTION OPERATOR IN GENETIC ALGORITHMS", Computing and Informatics, Vol. 22, 2003, ??–??

[10]. S. Chatterjee, et al., "Genetic Algorithms and Traveling Salesman Problems," European Journal of Operational Research, vol. 93, pp. 490-510, 1996.

[11]. R. L. Haupt and S. E. Haupt, Practical Genetic Algorithms, Second Edition ed.: John Wiley & Sons, Inc., 2004.

[12]. C. M. Reidys and P. F. Stadler, "Combinatorial Landscapes," SIAM review, Society for Industrial and Applied Mathematics, vol. 44, No. 1, pp. 3-54, 2002.
[13]. Vanita Tonge," Performance Improvement of Differential Evolutionary Algorithm: A Survey", International Journal of Current Engineering and Technology ISSN 2277 – 4106.
[14]. Vanita G. Tonge and Pravin Kulkarni," Permutation Flowshop Scheduling Problem Using Classical NEH, ILS-ESP Operator", B.K. Panigrahi et al. (Eds.): SEMCCO 2013, Part II, LNCS 8298, pp. 595–605, 2013. © Springer International Publishing Switzerland 2013