EDGE RECOMBINATION WITH EDGE SENSITIVITY IN THE TSP PROBLEM

***Jarosław Kęsy*[1], *Zbigniew Domański*[1, 2]**

[1]*Institute of Mathematics and Computer Science, Czestochowa University of Technology*
[2]*Polonia University, Czestochowa*

Abstract. We use an evolutionary algorithm with a modified version of edge recombination crossover to search near optimal tours in the Traveling Salesman Problem (TSP). We introduce so called edge sensitivity to estimate the importance of having a given edge within an optimal solution. In order to test this approach we used the data files gr24, bays29 and gr48. Results we obtained show that edge sensitivities allow us to find solutions quicker than other similar algorithms with an edge recombination crossover.

1. Introduction

Optimisation is an important question in wide range of complex technological and scientific problems. The question is how to determine optima of a real-valued specific function, called fitness, which is a quality measure related to a given problem. Complex optimisation problems have a high-dimensional search space and then classical optimisation techniques are useless. In contrary, evolutionary algorithms represent very powerful approach to study such a question. They consist of different types of algorithms. Examples are simulated annealing [1], genetic algorithms [2], evolutionary programming [3] and genetic programming [4].

The evolution program is a probabilistic algorithm which maintains a population of individuals. Each individual is implemented as some data structure, and it represents a potential solution to the given problem. Each of them is evaluated to give some measure of its fitness. Then, from the current population, better individuals are selected by the selection operators. The selected solutions produce new candidate solutions by applying crossover and mutation operators. New solutions replace some of the old ones and the process is repeated until the termination criteria are met. After some number of generations the program converges and the best individual represents a near-optimal solution. Recombination and mutation operators play important role in a process of finding optima. Crossover combines the features of two parents to form two similar offspring by swapping corresponding segments of the parents. It enables an information exchange between different potential solutions. The mutation operator arbitrarily alters solutions and then introduces some extra variability into the population.

In some optimisation problems a special form of the crossover operator is employed. This so called recombination crossover mixes pieces of multiple promising sampled sub-solutions and composes solutions by combining them.

In this approach, we develop an *edge sensitivity matrix* from the current population, where an edge is a link or connection between two nodes - cities in a string. We then sample nodes of a new string according to the edge sensitivity matrix. We will call this method the *edge sensitivity based sampling algorithm* (ESBSA). We tested the algorithm in the Traveling Salesman Problem (TSP) [5], a typical, well-known optimization problem which consists of finding the shortest closed tour connecting all cities in a map. Even though the TSP is conceptually very simple it was proved to be NP-hard [6]. Hence, polynomial - time algorithms for finding optimal tours are unlikely to exist. Because exact solutions are almost impossible to obtain the aim is to find near-optimal solutions.

2. Edge Recombination Crossover and Edge Sensitivity Matrix

Edges encode important information in the TSP: they carry distances. The position of a city in a tour is not important because tours are circular. The objective function for minimisation is the total length of edges which constitute a legal tour, so the basic building blocks are edges. The general observation, based on experimental results, is that a good crossover operator should extract edge information from parents as much as possible. A class of operators that directly emphasises edges relays on different modifications of the two - parent edge recombination crossover operator ER [7]. The principal idea behind the ER crossover is that the offsprings are built almost exclusively from the edges present in both parents. The ER transfers more than 95% of the edges from the parents to the single offspring and widely explores the information on edges in a tour.

In our approach the sampling procedure is based on a quantity measuring how a given edge in a tour is sensitive to be broken and replaced by two other edges. Consider an edge $e(i,j)$ connecting cities i and j. The salesman can go from i to j directly or, perhaps he can improve his tour by going from i to k first and then, from k to j. It means that the edge $e(i,j)$ is broken and replaced by two other edges $e(i,k)$ and $e(k,j)$. Such situation arises when a third city k is injected between cities i and j. The distance between cities i and j changes from $d(i,j)$ to $d(i,j)+ d(i,j)$ and the difference is equals to

$$\Delta_k(i,j) = d(i,k) + d(k,j) - d(i,j)$$

Considering only this particular choice of city k has little meaning. From the value of $\Delta_k(i,j)$ it is hard to see that injecting this city in position between cities i and j is better than keeping the edge $e(i,j)$ unbroken. However, adding all possible Δ's, *i.e.* evaluating we obtain the quantity $\sum_{k=1,\dots,N} \Delta_k(i,j)$ which enable us to see how sensitive is the edge $e(i,j)$ in the face of replacement by one of the other available edges. Higher values of $S_{i,j}$ indicate high sensitivity of the edge $e(i,j)$ to be locally broken. Finally we can construct the sensitivity matrix S by defining its elements as follows

$$S_{i,j} = \sum_{k=1}^{N} \Delta_k(i,j) - d(i,j)$$

Here, we extracted the distance in order to make a difference between edges having the same sensitivity. As an example consider the TSP problem with the distances $d(i,j)$ given by the following matrix D:

$$D = \begin{bmatrix} 0 & 2 & 1 & 2 & 3 & 6 & 4 & 2 & 1 \\ 2 & 0 & 5 & 4 & 2 & 1 & 5 & 1 & 4 \\ 1 & 5 & 0 & 8 & 2 & 2 & 2 & 1 & 3 \\ 2 & 4 & 8 & 0 & 3 & 1 & 3 & 1 & 4 \\ 3 & 2 & 2 & 3 & 0 & 5 & 8 & 5 & 2 \\ 6 & 1 & 2 & 1 & 5 & 0 & 1 & 2 & 7 \\ 4 & 5 & 2 & 3 & 8 & 1 & 0 & 2 & 5 \\ 2 & 1 & 1 & 1 & 5 & 2 & 2 & 0 & 3 \\ 1 & 4 & 3 & 4 & 2 & 7 & 5 & 3 & 0 \end{bmatrix}$$

Then, the corresponding sensitivity matrix S has the form:

$$S = \begin{bmatrix} 0 & 25 & 35 & 27 & 21 & -14 & 11 & 2 & 40 \\ 25 & 0 & -2 & 10 & 34 & 39 & 4 & 1 & 13 \\ 35 & -2 & 0 & -30 & 34 & 29 & 34 & 1 & 23 \\ 27 & 10 & -30 & 0 & 26 & 41 & 26 & 1 & 15 \\ 21 & 34 & 34 & 26 & 0 & 5 & -20 & 5 & 39 \\ -14 & 39 & 29 & 41 & 5 & 0 & 45 & 2 & -16 \\ 11 & 4 & 34 & 26 & -20 & 45 & 0 & 2 & 9 \\ 18 & 31 & 31 & 33 & -3 & 22 & 27 & 0 & 16 \\ 40 & 13 & 23 & 15 & 39 & -16 & 9 & 16 & 0 \end{bmatrix}$$

The edge sensitivity based on sampling procedure enable us to build the offspring by choosing the more sensitive edges first, then those less sensitive. We applied it in the framework of the greedy algorithm. In the case of edge failure (*i.e.* being left with a city without a continuing edge) we made a random selection.

3. Numerical computations

We tested the sensitivity edge matrix approach with the following well - known data files: 24 cities gr24, 29 cities bays29 and 48 cities gr48 employing the (μ,λ) evolution strategy. 24 cities gr24 and 48 cities gr48 were used in the study of TSP with different crossovers defined for the path representation: partially mapped (PMX) [8], order (OX) [9], cycle (CX) [10] and for crossovers defined for permutation representation: the enhanced edge recombination operator (eER) [11], EDA [8] or EHBSA [12]. For each of the files gr24, bays29 and gr48 we tried to find the solution ten times. The population size was 60 and 120. The maximum number of runs was 10000 for gr24, 50000 for bays29 and 100000 for gr48. Results are shown in Table 1, Table 2 and Table 3, respectively. NOpt is the number of runs in which the algorithm succeeded in finding the optimum tours or evaluation maximum number of trials is reached.

Table 1. Results of gr24

Max number of trials = 10000, Crossover = 0.7, Mutation = 0.3, Solution = 1272			
Population size = 60		Population size = 120	
Min length of tour	NOpt	Min length of tour	NOpt
1272	23	1272	129
1272	189	1272	286
1272	47	1272	76
1272	48	1272	245
1272	286	1272	208
1272	61	1272	297
1272	54	1272	740
1272	82	1272	289
1272	101	1272	61
1272	156	1272	245

Table 2. Results of bays29

Max number of trials = 50000, Crossover = 0.7, Mution = 0.3, Solution = 2020			
Population size = 60		Population size = 120	
Min length of tour	NOpt	Min length of tour	NOpt
2020	3390	2020	57
2020	434	2020	798
2020	496	2020	412
2020	33786	2020	1735
2020	213	2020	732
2020	107	2020	1177
2020	859	2020	1350
2020	2288	2020	9857
2020	31133	2020	850
2020	4163	2020	299

Table 3. Results of gr48

Max number of trials = 100000, Crossover = 0.7, Mution = 0.3, Solution = 5046			
Population size = 60		Population size = 120	
Min length of tour	NOpt	Min length of tour	NOpt
5046	82523	5055	100000
5074	100000	5107	100000
5074	100000	5074	100000
5067	100000	5063	100000
5046	51407	5093	100000
5063	100000	5074	100000
5074	100000	5046	46255
5063	100000	5055	100000
5072	100000	5125	100000
5074	100000	5115	100000

Conclusions

In this approach, we have proposed new edge recombination crossover operator based on the edge sensitivity mechanism of sampling from the current population. The results showed that it worked fairly well with a smaller size of population on the test problems used. It also worked better than well-known traditional two parent recombination operators. There are many opportunities for further research related to the proposed approach. The effect of the size of population N on the performance of the algorithm must be carefully investigated. Applying ESBSA to other problems, such as job-shop scheduling problems, remain for future work.

References

[1] Kirkparick S, Gelatt C.D., Jr., Vecchi M.P., Science 1983, 220, 671.

[2] Holland J.H., Adaptation in Natural nad Artificial Systems, University of Michigan Press, Ann Arbor 1975.

[3] Fogel D.B., Evolutionary Computation - Toward a New Philosophy of Machine Intelligence, IEEE, Piscataway 1995.

[4] Koza J.R., Genetic Programming: On the Programming of Computers by Means of Natural Selection, MIT Press, Cambridge 1992.

[5] Michalewicz Z., Genetics Algorithms + Data Structures = Evolution Programs, Springer - Verlag, Berlin, Heidelberg 1996.

[6] Garey M., Johnson D., Computers and Intractability, W.H. Freeman, San Francisco 1979.

[7] Whitley D., Starkweather T., Fuquay D.A., Scheduling Problems and Traveling Salesman: The Genetic Edge Recombination Operator, Proc. 3th Int. Conf. on Genetic Algorithms, J. Schaffer (Editor), Morgan Kaufmann Publishers 1988, 133-140.

[8] Robles V., Miguel P.D., Larranaga P., Solving the traveling salesman problem with EDAs, Estimation of Distribution Algorithms, eds P. Larranaga, J.A. Lozano, Kluwer Academic Publishers 2002, Chapter 10, 211-229.

[9] Oliver I., Smith D., Holland J., A study of permutation crossover operators on the travel salesman problem, Proc. 2nd Int. Conf. on Genetic Algorithms, Lavrence Erlbaum Associates, Hillsdale 1987, 224-230.

[10] Starkweather T., McDaniel S., Mathias K., Whitley D., Whitley C., A comparison of genetic sequence operators, Proc. 4th Int. Conf. on Genetic Algorithms, Morgan Kaufmann Publishers 1991, 69-76.

[11] Goldberg D.E., Genetic algorithms in search, optimization and machine learning, Addison - Wesley Publishing Company 1989.

[12] Tsutsui S., Goldberg D.E., Pelikan M., Solving Sequence Problems by Building and Sampling Edge Histograms, IlliGAL Report No. 2002024, August 2002, http//www-illigal.ge.uiuc.edu