Proceedings of the 2012 International Conference on Industrial Engineering and Operations Management
Istanbul, Turkey, July 3 – 6, 2012

A Development of Hybrid Cross Entropy-Tabu Search Algorithm for Travelling Repairman Problem

Budi Santosa and Muchammad Aminuddin
Department of Industrial Engineering
Surabaya Institute of Technology
Indonesia

Abstract

Travelling Repairman Problem (TRP) is one variant of the Travelling Salesman Problem (TSP) with the objective function minimizing the total waiting time for customers served by the repairman. Finding TRP solution is very important because it is often encountered in real world cases. In the TRP, a customer requires a fast service that he was satisfied. TRP is a combinatorial problem that requires a long computing time especially for large problems. For such problems, metaheuristics approaches are more efficient than the exact approach i.e mixed integer programming. Cross Entropy is one of relatively new metaheuristics techniques that has shown good results to solve some combinatorial optimization problems. Meanwhile, Tabu Search algorithm is often used for the completion of combinatorial optimization problems with great results. In this paper the hybrid both methods is used to obtain better results in solving the Travelling Repairman Problem.

Keywords
Travelling Repairman Problem, combinatorial, Cross Entropy, Tabu Search, Hybrid Cross Entropy-Tabu Search

1. Introduction

Traveling Repairman Problem (TRP) is one variant of the Traveling Salesman Problem (TSP) with different objective function. Travelling Repairman Problem is also known as the Minimum Latency Problem, Traveling Deliveryman Problem, or TSP with cumulative costs [5]. The purpose of the Travelling Repairman Problem is to minimize the total waiting time for customers that served by a repairman. While, the TSP objective function is minimizing the total time or distance traveled by the salesman. Solving the TRP is very important because in the real cases it is often found that a customer requesting a service, which is urgent and must be served.

Somes methods have been developed to solve Travelling Repairman Problem. However, it is still a little research being done to solve Travelling Repairman Problem using the metaheuristics. Recent research mostly solve this problem using exact methods and heuristic methods, including Tabu Search [4], Polynomial Time Algorithms [16], Exact Algorithm [15], Lagrangian Relaxation [8,9] (Rocha et al.), and Approximation Algorithm[1,2]. The research of this problem with metaheuristics method first performed using GRASP + VND algorithm [11] and the last is done using Improved Genetic Algorithm [3]. This paper applies relatively new metaheuristics to solve the problem to have faster computational time. Cross Entropy (CE) was chosen because this algorithm has shown good results in solving some optimization problems such as job scheduling, orienteering problem, and crew scheduling [10, 12, 13, 14].

The research not only developing algorithm CE for TRP, but also developing a Tabu Search algorithm which will be hybrid with CE so it will created a new algorithm that is Hybrid Cross-Entropy Tabu Search Algorithm (CE-TS) to solve TRP in order to get better results. In its application, the CE algorithm is very depending on the number of samples generated to improve optimal results. However, computational time will be longer by increasing the number of samples generated. CE-TS algorithm developed to improve the optimal results obtained by the CE algorithm without increasing the number of samples generated by setting the initial generation of CE through the Tabu Search algorithm so that the initial sample matrix generated by CE is already a good sample from the results of Tabu Search. Tabu Search algorithm is chosen to be hybrid with CE because its advantage with the Tabu List can keep the search process does not trapped on a local optimum that appeared on a previous search [6]. The size of tabu list can be changed so that we can determine how many best results of Tabu Search will be used for the initial solution of CE.

2. Mathematical Model of *Travelling Repairman Problem* (TRP)

The mathematical model for Travelling Repairman Problem refers to [5] :

$$Min \quad n * \sum_{i=1}^{n}\sum_{j=1}^{n} C_{ij}X_{ij} - \sum_{i=1}^{n}\sum_{j=1}^{n} C_{ij}Y_{ij} \dots\dots\dots\dots\dots\dots\dots\dots (1)$$

Subject to.

$$\sum_{j=1}^{n} X_{ij} = 1 \quad \forall\ (i = 1, \dots, n) \dots\dots\dots\dots\dots\dots\dots\dots (2)$$

$$\sum_{j=1}^{n} X_{ji} = 1 \quad \forall\ (i = 1, \dots, n) \dots\dots\dots\dots\dots\dots\dots\dots (3)$$

$$\mu_i - \mu_j + (n-1)X_{ij} + (n-3)X_{ji} \leq (n-2) \ \forall\ (i, j = 2, \dots, n) \dots\dots\dots\dots (4)$$

$$\mu_j \leq (n-2) + (3-n)X_{1j} + X_{j(n+1)} \ \forall\ (j = 2, \dots, n) \dots\dots\dots (5)$$

$$\mu_j \geq (n-3)X_{j(n-1)} - X_{j1} + 2 \ \ \forall\ (j = 2, \dots, n) \dots\dots\dots (6)$$

$$Y_{ij} \leq (n-1) * X_{ij} \ \ \forall\ (i, j = 1, \dots, n) \dots\dots\dots\dots\dots\dots (7)$$

$$Y_{ij} \leq \mu_i \ \ \forall\ (i, j = 1, \dots, n) \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (8)$$

$$X_{ij} \in \{0,1\} \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (9)$$

$$\mu_i \geq 0 \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (10)$$

$$Y_{ij} \geq 0, \quad Y_{ij} \ \ integer \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (11)$$

Notes :

X_{ij} = 1 if *repairman* travels arc (i, j)
0 otherwise

Y_{ij} = μ_i if $X_{ij} = 1$
0 otherwise

μ_i = denotes node position in the *tour*

Objective function of TRP is minimizing overall total of customer waiting time (1). Constraints (2) and (3) are the assignment constraints, which guarantee that only one arc arriving and one arc leaving any node so that one node only is visited once. Constraint (4) corresponds to the lifted subtour elimination.In this contraint, μ_i denotes position of the node. Constraints (5) and (6) are lifted the bound constraints on the position of μi. Constrain (7) denotes that Y_{ij} will have positive value if node *j* visited right after node *i*. Y_{ij} in constraint (8) denotes position of customer on the tour.

The difference between the TRP with the TSP on the formulation above is that the distance between one node to another node is not added directly to obtain the final result. For example, the distance from the initial node to first node that visited should be added n-1 times as the influence of waiting time of all other customers. Thus, the variable Y_{ij} will equal to 0 if the arc between i and j are not part of the route or has n-k+1 value if the arc is in position k on the route [7].

The constraints will be accommodated in a Hybrid Cross Entropy-Tabu Search Algorithm (CE-TS) consist of the objective function in constraint (1), constraints (2) and (3) to ensure that all customers should be visited by repairman exactly once, and constraints (4) to (8) to get the value of Yij, which denotes the position of customers in route from the repairman intial city. Y_{ij} values will be the key factor to calculate the influence of another customer waiting time on customer waiting time j.

Objective function of TRP refers to the constrain (1) has been accommodated in the CE-TS hybrid algorithms through an evaluation. Objective function count the influence of another customer waiting time which can be determined from the value Y_{ij}. Y_{ij} values have been accommodated in the algorithm by setting the index that will be a reduction coefficient in the objective function. Constraints (2) and (3) have been accommodated in the CE-TS hybrid algorithms through the generation mechanism of the sample route. The route generation mechanism used in CE and TS respectively and on CE-TS hybrid algorithms is different. In the CE algorithm, the generation of routes using probability transition matrix uses random numbers for route selection with roulette wheel selection mechanism.

3. Development of Algorithm

At this section, the development of Cross Entropy algorithm and Tabu Search algorithms produce a new algorithm which is a hybrid of CE and TS that can be applied to the Traveling Repairman Problem (TRP).

3.1 Cross Entropy Algorithm (CE) for Travelling Repairman Problem

Cross Entropy algorithm used to solve NP-hard optimization problems by generating a number of *N* candidate solutions by a specific mechanism (in TRP using node transition mechanism) and each candidate solution objective function will be evaluated. From the generated candidate solutions, we choose the number ρ x N elite samples to update the parameters for the generation of solutions in the next iteration so that the candidate solution in the next iteration better than previous iteration

The proposed algorithm can be explained in details below [16].

1. Determination of Parameters
   Determine the parameters of CE: smoothing parameter (α), elite sample proportion (ρ), and number of population in each iteration (N).

2. Generation of Transition Probability Matrix
   The value of each cell of Transition Probability matrix P in the first stage is set to $\frac{1}{n-1}$, and the values of all diagonal matrix is 0.

3. Generating *N* route as a candidate solution
   N routes are generated based on transition matrix *P*.

4. Calculation of Customer Waiting Time
   Calculation of total customers waiting time for each route generated before. Use these values as fitness measure.

5. Elite Samples Choice
   Select *ρ x n* elite sample corresponds to the best fitness values.

6. Matriks Transition Update
   Transition matrix update is conducted to obtain the transition matrix to generate a solution on the next iteration.

7. Stopping Criteria Checking
   Stopping Criteria determined to decide whether iteration continues or not. Here the number of maximum iteration is used.

3.2 Hybrid Cross Entropy-Tabu Search Algorithm (CE-TS) for *Travelling Repairman Problem*

Best solution obtained by CE in considerable time is very dependent on the number of samples generated. However, this would be inefficient because the computational time would be much longer. Thus, CE-TS hybrid algorithm was developed with the aim to improve the best solution without increasing the number of generated samples. The procedure is generating initial sample for CE using the best result of Tabu Search. CE-TS is a combination of CE and Tabu Search. Tabu Search algorithm plays a significant role in generating the best routes in the Tabu List as the result of iterations of Tabu Search. Tabu list contains good routes which will be initial elite sample for CE.

CE-TS Algorithm for *Travelling Repairman Problem* in general is explained in details.

1. Determination of Parameters

Determine the parameters of CE-TS: smoothing parameter (α), elite sample proportion (ρ), number of population in each iteration (N). Tabu List Member Count (number of route generated by *Tabu Search* and saved in *Tabu List*), and itmax_tabu (maximum iteration of *Tabu Search*).

2. Generation of candidat solution of *Tabu Search* using neighbourhood selection
   Initially, generate one randomized route, then copy it as many as n . Do the neighborhood selection so it will produce n candidate solutions.

3. Choosing the best solution from candidate solution
   Candidate solution will be evaluated one by one and then sort ascendingly. The best route will be evaluated whether can enter into Tabu List or not.

4. Updating Tabu List
   Best route from step 3 will be evaluated whether it is better than existing route in the Tabu List or not. If this route is better than existing route in the Tabu List, then the route will enter into the Tabu List to change worse route in the Tabu List. Best route is used for generating candidate solution in the next iteration. However, if the generated route is not better than existing route in the Tabu List, then solution will not be entered into Tabu List and will generate randomize route for the neighborhood selection at the next iteration.

5. Checking Stopping Criteria of Tabu Search
   Stopping Criteria of Tabu Search is maximum iteration determined in the step 2. If stopping criteria is reached, then route in the Tabu List will be inputted to the CE. Otherwise, then go back to step 3 and iteration will be continued until maximum iteration is met.

6. Tabu List Route Input as initial elite sample of CE
   This is transition step between TS and CE where result from TS will be used as input for CE. Tabu List consists of best routes from the TS iteration and then will be used to generate elite sample P. In this way, transition matrix P in the start of iteration already consists of good samples.
7. Generate N route as a initial candidate solution
   N routes are generated based on transition matrix P.

8. Fitness Function Calculation
   Calculation of total customers waiting time for each route generated before. Use these values as fitness measure.

9. Elite Sample Selection
   Select $\rho \, x \, n$ elite sample corresponds to the best fitness values.

10. Transition Matrix Update
    Transition matrix update to obtain transition matrix to generate solution on the next iteration.

11. Checking Stopping Criteria
    If the number of maximum iteration is met, then stop. Otherwise, go to step 7.

4. Experiments and Results

Experiments were conducted using three data sets: Eil51 (51 cities), KroA100 (100 cities), and KroA150 (150 cities) These data sets were obtained from TSPLIB 95 which is available online. The parameters for CE are α (smoothing parameter), ρ (the proportion of elite samples), and N (number of samples). For the CE-TS, there are several additional parameters such as number of routes stored into Tabu List and maximum number of iterations for TS. The resulting total customer waiting time (latency), computational time, and number of iteration required to find the optimum solution are compared between CE and CE-TS. In addition, the total waiting time of both algorithms are also compared with those produced by Approximation Algorithm (AA).

The results of the experiments on three data sets are shown in Tables 1, 2 and 3 for CE, CE-TS and AA.

Table 1 Comparison of the Experimental Results of CE and CE-TS on Eil51

Algoritma	CE		CE-TS		AA
	Average	Best	Average	Best	
Waiting Time	10.537	10.352	10.488,6	10.406	14.638
Computing time	3.655	3.313	4.362	4.126	
Iterations	37	34	37	35	
Gap	38,92% better		39,56% better		

Table 2 Comparison of the Experimental Results of CE and CE-TS on KroA100

Algoritma	CE		CE-TS		AA
	Average	Best	Average	Best	
Waiting Time	1.168.967	1.123.700	1.183.367	1.173.100	1.307.340
Computing time	41.008	39.581	41.720	41.208	
Iterations	91	90	91	89	
Gap	11,84% better		10,48% Better		

Table 3 Comparison of the Experimental Results of CE on data set KroA150 with AA

Algoritma	CE		CE-TS		AA
	Average	Best	Average	Best	
Waiting Time	2.391.900	2.391.900	2.567.267	2.481.700	2.494.782
Computing time	250,733.33	249,360	229,020	216,390	
Iterations	220	220	194	183	
Gap	4,3% better		2,9% worse		

5. Analysis

For Eil 51 and KroA100 data sets, experimental results shows that CE and CE-TS produced better results than AA for both average and best results. This is because CE-TS initial good routes generated by the TS in the Tabu list provides a significant influence on the CE-TS algorithm to find best solution. However, computational time of CE is relatively faster than CE-TS. Because the CE-TS must pass 2 parts of algorithm: the generation of CE-TS good initial solutions and finding the final optimal or near optimal solution. While, comparisson between CE and CE-TS for the case of 100 cities showed that the CE gives best and average total customer waiting time better than CE-TS. This is because the route result of Tabu Search is not good enough to proceed into CE-TS algorithm. In terms of computational time, CE algorithm also needs a little faster computing time. However, if it is compared with small data size, the computational time between the CE and CE-TS for the problem of 100 cities is not too different though with the same CE parameters.

In the case of KroA150 data set, the comparison of total customer waiting time obtained in CE algorithm indicates that CE algorithm gives better results than AA. CE-TS algorithm provides worse results than those produced by AA. From these results we can see that CE-TS algorithm is not reliable enough to solve large size-problems. From 3 replications, the CE algorithm indicates that it is obtained more stable results than CE-TS algorithm On CE-TS algorithm, it needs few replications, but highly risks to get stuck in local optimal despite it has receive good initial samples from TS algorithm. CE-TS algorithm needs better computational time than the CE, whereas in CE-TS there is spesific generation samples mechanism with Tabu Search which takes a certain computation. This is because the prematur convergence of CE-TS algorithm. However, prematur convergence makes CE-TS algorithm could potentially get stuck in local optimum.

Overall Performance Analysis

In the experiments, the number of samples (N) = 10000 is used for all data types. By large sample size, it allows more extensive exploration of the solution. In this way it is enable to obtain better results yet still spend reasonable computational time. The elite sample is about 10% of total sample which is big enough to prevent prematur convergence. The value of α used in the algorithm CE and CE-TS is 0.6. The value of α that are too large (for example 0.8 or 0.9) will give a larger proportion of elite sample empirical probability so that CE will reach convergence quickly. Elite samples taken in any iteration is not always good because samples generation at the initial iteration is still random. The value of α used 0.6 will give enough proportion for the transition matrix at the previous iteration so that exploration space still potential to get better results than the previous iteration.

Overall, CE and CE-TS algorithm has provided better results than AA. However, CE-TS algorithm is not enough reliable to solve large size problems. This is because CE-TS initial sample generation is set by entering best sample from TS as an initialization for the CE. But for large problem to get initial sample which is good enough still needs big sample size. As a conssequence, it needs longer computational time. For large size problem, TS is not able to produce a good results because of possible solutions is too many. Solutions generation in the TS is very random and no specific parameters, just relies on local search strategy so that large size problems might require many iterations to get a good solution. It becomes inefficient because it will take a long time on TS iterations.

6. Conclusion

In this paper Hybrid Cross Entropy-Tabu Search (CE-TS) Algorithm is developed successfully for solving Traveling Repairman Problem (TRP). We compared the results of CE-TS, CE and Approximation Approach. CE-TS produces total customer waiting time better than those produced by CE in small-size problems. For large-size problem, CE algorithm produced better total customer waiting time. CE and CE-TS produces an overall performance better than the Approximation Algorithm. CE-TS has shortcomings of long computational time. In these two algorithms a number of N-solutions should be generated in each iteration. CE and CE-TS performances are very dependent on the number of generated samples, and the greater number of samples requires longer computational time.

References

[1]. Archer, A., Levin, A. & Williamson, D. P. 2008. A faster, better approximation algorithm for the minimum latency problem. *SIAM Journal on Computing*, 37**,** 1472-1498.

[2]. Archer, A. & Williamson, D. P. 2003. Faster approximation algorithms for the minimum latency problem. *Proceedings of the fourteenth annual ACM-SIAM symposium on Discrete algorithms.* Baltimore, Maryland: Society for Industrial and Applied Mathematics.

[3]. Bang, B. H. & Nghia, N. D. 2010. Improved genetic algorithm for minimum latency problem. *Proceedings of the 2010 Symposium on Information and Communication Technology.* Hanoi, Viet nam: ACM.

[4]. Dewilde, T., Cattrysse, D., Coene, S. & CR, F. Year. Heuristics for the Traveling Repairman Problem with Profits. *In*., 34.

[5]. Ezzine, I., Semet, F. & Chabchoub, H. Year. New Formulations For The Traveling Repairman Problem. *In*, 2010. Citeseer.

[6]. Glover, F. & Laguna, M. 1998. *Tabu search*, Kluwer Academic Pub.

[7]. Lechmann, M. 2009. The traveling repairman problem.

[8]. Rocha, A., Fernandes, E. & Soares, J. Year. Solving the Traveling Repairman problem with differentiated waiting times through Lagrangian relaxation. *In*.: Citeseer, 972-99841.

[9]. Rocha, M. & Neves, J. 2004. Preventing premature convergence to local optima in genetic algorithms via random offspring generation. *Multiple Approaches to Intelligent Systems***,** 127-136.

[10]. Rubinstein, R., & Kroese., D. 2004. The cross-entropy method: A unified approach to combinatorial optimization, Monte-Carlo simulation, and machine-learning. *Springer-Verlag Berlin Heidelberg*.

[11]. Salehipour, A., Sorensen, K., Goos, P. & Braysy, O. 2008. An efficient GRASP+ VND metaheuristic for the traveling repairman problem. *Working Papers*.

[12]. Santosa, B and Hardiansyah, N., 2010**,** Cross Entropy Method for Solving Generalized Orienteering Problem**,** *iBusiness*, 2, 342-347

[13]. Santosa, B, Budiman, MA, and Wiratno, S.E, 2011, A Cross Entropy-Genetic Algorithm for m-Machines No-Wait Job-Shop Scheduling Problem, JILSA, Vol. 3 No.3.

[14]. Santosa, B. & Willy, P. 2011. *Metoda Metaheuristik : Konsep dan Implementasi,* Surabaya, Guna Widya.
[15]. Wu, B., Huang, Z. & Zhan, F. 2004. Exact algorithms for the minimum latency problem. *Information Processing Letters*, 92**,** 303-309.
[16]. Wu, B. Y. 2000. Polynomial time algorithms for some minimum latency problems. *Information Processing Letters,* 75**,** 225-229.