International Journal of Engineering and Science Applications
Vol 8 No. 1 May 2021
ISSN 2406-9833

IJEScA

A Comparative Study between Dynamic Programming and Artificial Atom Algorithm for Traveling Salesman Problem

A Chandra*, A Naro

Study Program Industrial Engineering, Engineering Faculty, Mercu Buana University
Raya Meruya Selatan Street No.1, Kembangan, West Jakarta 11650

*Corresponding author: agung.chandra@mercubuana.ac.id

ABSTRACT
Dynamic Programming (DP) algorithm is a powerful technique of exact method that can produce optimal solutions. In this paper, the Dynamic Program is compared with Artificial Atom Algorithm (A^3) which is a new heuristic - metaheuristic method. Both algorithms of DP and A^3 have been tested on case study of 9 locations in the West Jakarta. Based on the case study of small size locations, the results show that the distance of DP and A^3 method is the same, but the sequence is different.

Keywords:Dynamic Programming (DP), A^3, exact, heuristic, metaheuristic

1. INTRODUCTION

In this covid 19 pandemic era, it is common for every business firm to learn all aspects of expenditures including distribution aspect which is one of the supply chain performance categories. Distributing goods from warehouse to customers need a shorter distance, a shorter time, and more accurate shipment. Lowering cost of distribution can be achieved by shortening distance that will decrease the fuel consumption, and as a result the environment will be greener [1]. The problem of finding the shortest Hamiltonian cycle is called Traveling Salesman Problem (TSP) and this problem is NP Hard problem [2], [3], [4]. There are two techniques that can be used to optimize solutions: the exact methods and the approximate methods. A small size, some medium size, and a large size with a specific structure may be solved by exact algorithm and it is unwise to solve problems by using metaheuristic algorithm when exact algorithm is found to be an efficient method. Dynamic programming, branch and bound, branch and cut and A* family of search algorithms are exact methods. Heuristic algorithm and approximation algorithm are approximate methods [5].

Dynamic programming is a very powerful technique to solve a particular class of problems, and optimal solutions to the sub-problems contribute to the optimal solution of the given solution [6]. Dynamic programming turns a suitable recursive description of a process into a method to produce an optimal solution, and also called recursive optimization [7].

Metaheuristic is a part of heuristic algorithm that find good solutions on a large size problem instances in a reasonable time and obtain accepatable performance at acceptable costs in a wide range of problems. Artificial Atom Algorithm (A^3) is a new metaheuristic method and proved to be the best solution when compared to Genetic Algorithm (GA), Particle Swarm Algorithm (PSO) and Artificial Bee Colony Algorithm (ABC) [8]. The same result was also reported that A^3 has the best solution for metaheuristic algorithm compared to Genetic Algorithm (GA), Simulated Annealing (SA), Tabu Search (TS), Ant Colony Algorithm (ACA), Particle Swarm Optimization (PSO) and Artificial Bee Colony (ABC) [9]. In this paper, a comparative study will be tested on two methods: Exact algorithm – Dynamic programming (DP) as a powerful method and Metaheuristic algorithm – Artificial Atom Algorithm (A^3). The study is conducted in finding a better solution between two algorithms in which one of them will produce the shortest route for truck that delivers goods from warehouse to all customers in the West Jakarta.

2. METHODOLOGY

The Objective of both algorithm is getting the sum of distances in kilometres by traveling all customer and visiting once and then returning to warehouse. The distance between each customer in this study is assumed to be symetric with TSP which means if the truck travels from customer A to customer B, the distance will be the same from customer B to customer A. There is no congestion.

A. Dynamic Programming Algorithm

Dynamic programming (DP) is an alternative search strategy that is exhaustive search, slower than greedy search but gives the optimal solution. DP view a problem as consisting of subproblem that aims to solve the main problem by solving some subproblems [10].

Notation for distance:

Cost (or distance) of going from stage k, stage i to stage k+1, state j is: *d (k, i, k+1, j)*

Notation for minimum cost from a node to the end:

$V(k,i) = min_j\ (d(k,i,k+1,j) + V(k+1,j))$

The above formula is a recursion formula which means the current step is a base for the next step [11].

B. Artificial Atom Algorithm(A^3)

A^3 is a new nature inspired metaheuristics optimization method and developed by A.E. Yildirim [8]. A^3 is inspired by chemical compounding processes and developed by modeling of chemical ionic bond and covalent bond processes. The most important feature of A^3 is that A^3 examine the effect of parameter values on the result separately. There are three important concepts for A^3 which are electrons, atoms and atoms set. Each parameter value is represented by electrons and has an effect on the solution. Atoms consist of electron and means candidate solutions. [8], [12].

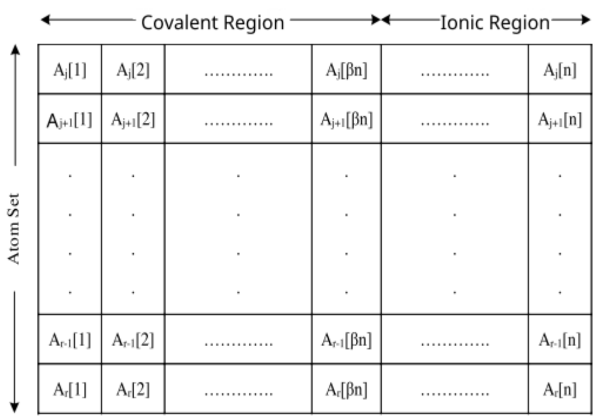

Fig 1. A Representation of Atom set [8]

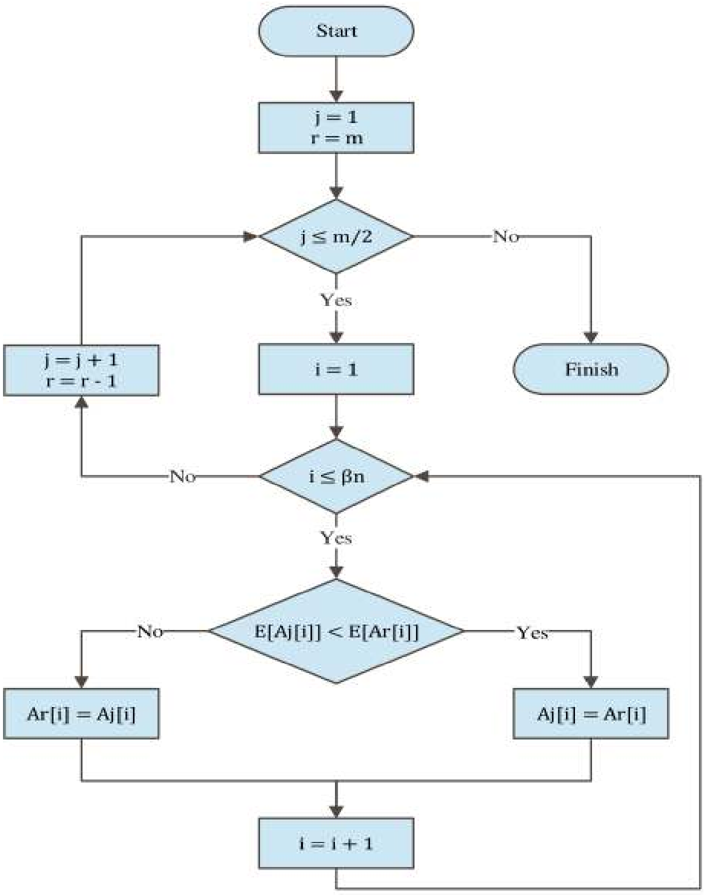

Fig.2. Covalent bond operator [8]

Pseudo code for covalent bond operator is

i ← *1,2, ..., βn*

if E $[A_j [i]]$ *is better than* $E[A_r[i]]$

Copy value of $A_j [i]$ *to* $A_r[i]$

Else

Copy value of $A_r[i]$ *to* $A_j [i]$

When the operator of ionic bond is used instead of electrons in the ionic region, random electrons are incorporated into the atom set. The ionic bond operator algorithm is as follows:

Ionic bond (atomset, m, n, β)

j ← *, ...,m // m : number of atoms*

i ← *βn + 1,...,n // β: covalent rate*

// n: number of electrons

$A_j [i] \leftarrow L_i + \eta * (U_i - L_i)$

// $A_j [i] \in AtomSet$

// η: a random number generated between 0-1

// U_i*: Upper bound for* i^{th} *attribute*

// L_i *: Lower bound for* i^{th} *attribute*

The A^3 algorithm step is following Fig. 1.

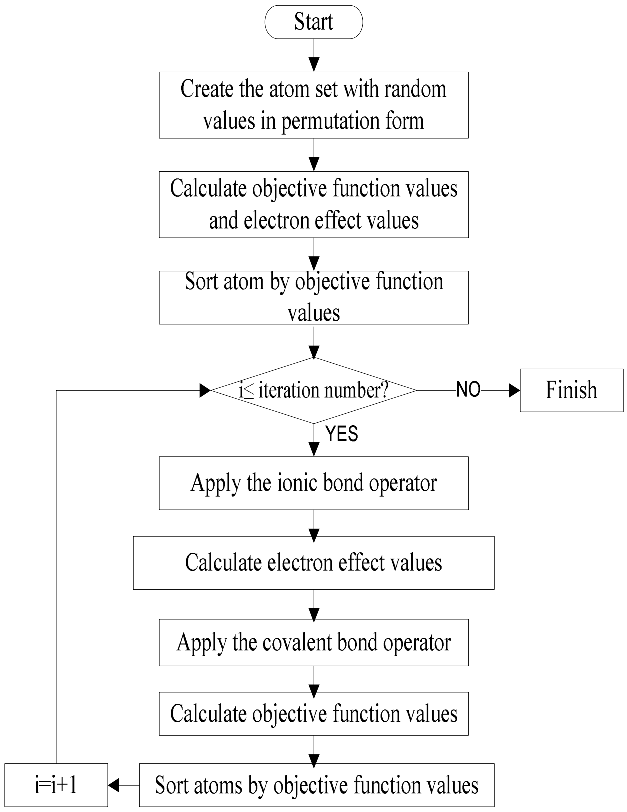

Fig 3. Artificial Atom Algorithm [8]

3. RESULTS AND DISCUSSION

There are nine locations where one of those locations is distribution center - DCH. The distribution center and the customer's locations are located in the West Jakarta. The route always starts to deliver goods from DCH to each location once and after that the truck returns to DCH. The distance between locations are as follow: starting from distribution center (DCH) and ending at distribution center (DCH) too. The distance between locations and the coordinates of locations are as follow:

Table 1. Distance between locations

	DCH	CG6	CPM	PRM	KTA	LMP	GMP	MTA	NSF
DCH	0	10	3	5	7	4	6	3	3
CG6	10	0	11	8	12	8	13	11	11
CPM	3	11	0	7	4	6	4	1	1
PRM	5	8	7	0	10	1	10	7	7
KTA	7	12	4	10	0	10	1	4	4
LMP	4	8	6	1	10	0	10	7	6
GMP	6	13	4	10	1	10	0	4	4
MTA	3	11	1	7	4	7	4	0	1
NSF	3	11	1	7	4	6	4	1	0

Table 2. The coordinates of locations

No	Location	X (peta)	Y (peta)
1	**DCH**	0.26	6.72
2	**CG6**	7.05	0.00
3	**CPM**	1.71	8.56
4	**PRM**	0.20	2.32
5	**KTA**	3.90	11.61
6	**LMP**	0.00	2.79
7	**GMP**	3.24	11.71
8	**MTA**	1.27	8.84
9	**NSF**	1.72	8.52

The route is DCH → NSF → CPM → MTA → GMP → KTA → CG6 → PRM → LMP → DCH.

Now, by running software Matlab 2015, Intel R Core (TM) i5 – 7200 U CPU@2.5 GHz 32 bit ACPIx64 for ten times, the results are the same with the total distance for Dynamic Programming algorithm is 35 kilometres with time less than one second.

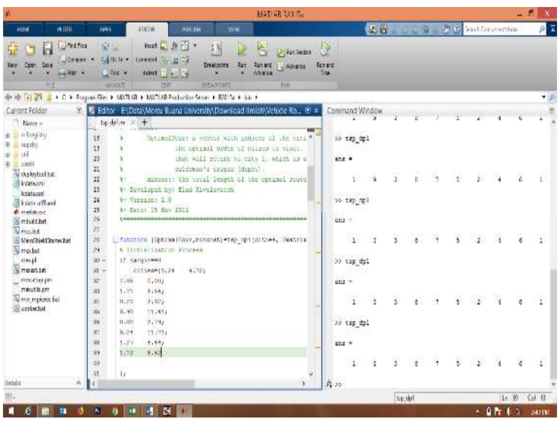

Fig 4. Route by Dynamic Programming Matlab code

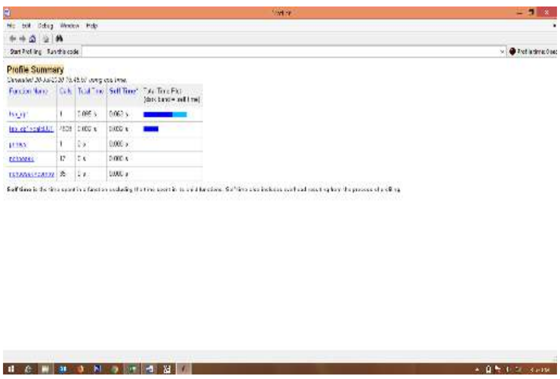

Fig 5. Time needed to produce the result by DP matlab

Next, running Artificial Atom Algorithm (A^3) for ten times, the iteration is various from 6 to 8 locations, and the results are the same for each running. The graph, time, and distance is as follows:

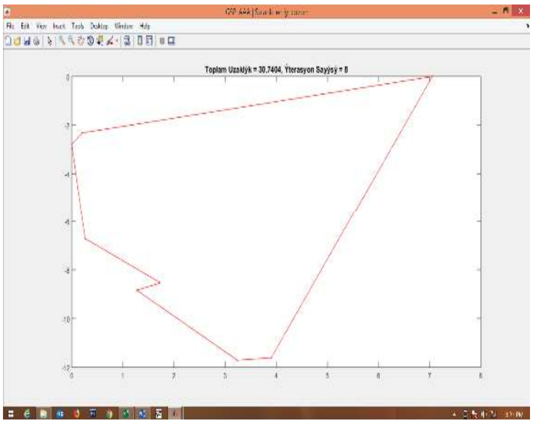

Fig 6. Graph for A3 algorithm

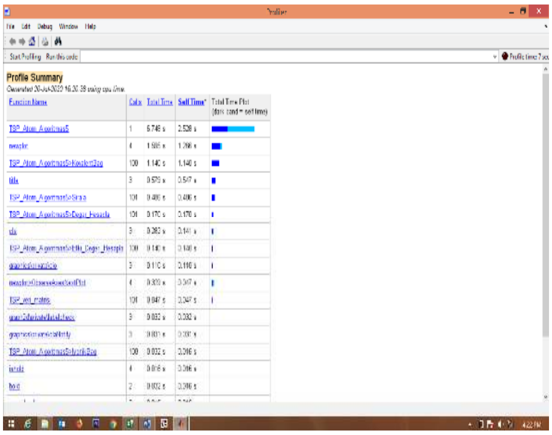

Fig 6. Time to produce the result of A3 algorithm

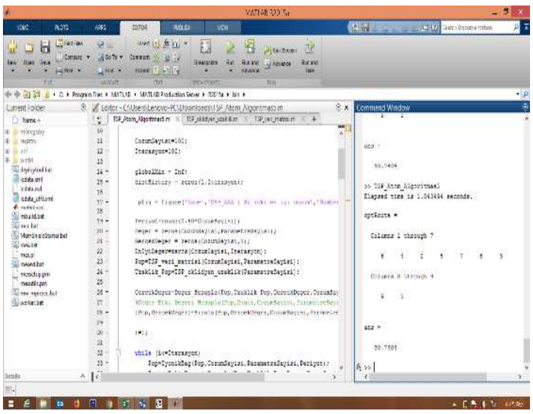

Fig 7. The result of A3 matlab code

The route is DCH → LMP → PRM → CG6 → KTA → GMP → MTA → CPM → NSF → DCH with total distance is 35 kilometres. From both DP and A^3 algorithm, the total distances are the same, 35 kilometres, but the sequence of route for each algorithm is different.

Table 3. Summary Sequence and Distance

Algorithm	Sequence of Route										Distance (KM)
DP	DCH	NSF	CPM	MTA	GMP	KTA	CG6	PRM	LMP	DCH	35
A3	DCH	LMP	PRM	CG6	KTA	GMP	MTA	CPM	NSF	DCH	35

In this small size case, one algorithm is not better than other one, in other words, both have equal solutions and the next research needs further analysis for medium size or large size – larger than fifty locations

4. CONCLUSIONS

In summary, for the case study of small size locations, both Exact method – Dynamic Programming algorithm and Heuristic method – Artificial Atom Algorithm produce 35 kilometre distance from distribution center to eight customer's location. The difference is the sequence of route. Larger size locations needs to be analyzed for next research.

References

[1] Dukic, G., Cesnik V., Opetuk T, 1985. Order Picking Methods and Technologies for Greener Warehouseing, *Strojarstvo* Vol. 52 (1), p.23-31.

[2] Chandra, A., and Setiawan, B, 2018. Optimizing the Distribution Routes Using Vehicle Routing Method," *Jurnal Manajemen Transportasi dan Logistik,* Vol.05 no.02, Juli 2018.

[3] G. Clarke and J.W. Wright,"Scheduling of Vehicles from a Central Depot to a number of Delivery Points," *Operations Research,* Vol.12 no.04, p.568-581, 1964.

[4] S. Dasgupta, C.H. Papadimitriou, U.V. Vazirani. (2006). *Algorithm.* McGraw Hill.

[5] E.G. Talbi. (2009). *Metaheuristics: From Design to Implementation.* John Wiley and Sons. New Jersey.

[6] A. Singhal and P. Pandy,"Traveling Salesman Problem by Dynamic Programming Algorithm," *International Journal of Scientific Engineering and Applied Science* Vol.2, Issue 1, January 2016.

[7] D.L. Applegate, W.J. Cook, S. Dash, D.S. Johnson. (2014). *A Practical Guide to Discrete Optimization: Dynamic Programming.* Compbook.

[8] A.E. Yildirim and A. Karci,"Application of Traveling Salesman Problem for 81 provinces in Turkey Using Artificial Atom Algorithm," *7th International Conference on Advanced Technologies,* April 28 to May 1, 2018.

[9] A.E. Yildirim,"Yapay Atom Algoritmasi ve Ayrik Problemlere Uygulanmasi," *Ph.D . Thesis*, Institute of Natural and Applied Science, Malatya , Turkey, February 2018.

[10] USC Linguistic. (2015). *Dynamic Programming: Matlab Code.* University of Southern California.

[11] Q.N. Nguyen. (May 2020). *Traveling Salesman Problem and Bellman-Held-Karp Algorithm.* White paper.

[12] A. Chandra and A. Naro,"A Comparative Study of Optimization Methods". *Scientific Journal of King Faisal University (submitted).* 2019