

## **EFFECT OF NEGATIVE CAMPAIGN STRATEGY OF ELECTION ALGORITHM IN SOLVING OPTIMIZATION PROBLEM**

(Kesan Strategi Kempen Negatif dalam Al-Khwarizmi Pemilihan  
dalam Menyelesaikan Masalah Pengoptimuman)

HAMZA ABUBAKAR, SARATHA SATHASIVAM\* & SHEHAB ABDULHABIB ALZAEEMI

### *ABSTRACT*

Election algorithm (EA) is an optimization technique based on minimization and coalition operations to solve competition among neurons. The Election algorithm gives the best individual of the population by enhancing both minimization and coalition operations while local search gives the best local solutions by testing all neighbouring solutions. Negative campaign mechanism is one of the most important mechanism in EA for its impact on the diversification and overcoming premature convergence of the entire search space towards optimal searching. The challenging task lies in selecting the appropriate negative campaigning operator that leads to optimal searching in a reasonable amount of time. The decision then becomes more difficult and needs more trial and error to find the best negative campaigning operator. This paper investigates the effect of negative campaign operators in enhancing the performance of EA based on the Travelling Salesman Problem (TSP). New negative campaign operator has been proposed based on selecting the best voter to be replaced. Experiments were conducted on the TSP to evaluate the proposed methods. The proposed mechanism was compared with other negative campaign operators. The result reveals the significant enhancement of the EA performance based on the proposed method in TSP problem.

*Keywords:* negative campaign strategy; random supporters; furthest supporters; nearest supporters

### *ABSTRAK*

Al-Khwarizmi pemilihan (EA) adalah teknik pengoptimuman berdasarkan pengoptimuman dan gabungan operasi untuk menyelesaikan persaingan di antara neuron. EA menghasilkan individu populasi terbaik dengan meningkatkan peminimuman dan operasi gabungan manakala pencarian tempatan memberikan penyelesaian tempatan yang terbaik dengan menguji kesemua penyelesaian jiranan. Mekanisme kempen negatif adalah satu daripada mekanisme terpenting dalam EA kerana kesannya terhadap kepelbagaian dan mengatasi penumpuan pramatang pada keseluruhan ruang carian ke arah pencarian yang optimum. Tugas yang mencabar terletak pada pemilihan operator kempen negatif yang sesuai yang mengarah kepada pencarian optimum dalam jangka masa yang sewajarnya. Keputusan itu kemudian menjadi lebih sukar dan memerlukan lebih banyak cuba jaya untuk mencari pengendali kempen negatif yang terbaik. Artikel ini menyelidik pengaruh pengendalian kempen negatif dalam meningkatkan prestasi EA berdasarkan kepada masalah Jurujual Mengembara (TSP). Pengendali kempen negatif baharu telah dicadangkan berdasarkan pemilihan pemilih terbaik untuk digantikan. Eksperimen dilakukan terhadap masalah TSP untuk menilai kaedah yang dicadangkan. Mekanisme yang dicadangkan dibandingkan dengan pengendali kempen negatif yang lain. Hasil menunjukkan peningkatan dalam prestasi EA yang signifikan berdasarkan kepada kaedah yang dicadangkan untuk masalah TSP.

*Kata kunci:* strategi kempen negatif; penyokong rawak; penyokong paling jauh; penyokong terdekat

## 1. Introduction

Optimization is the process of finding an alternative approach with the highest efficiency by maximizing desired factors (Blum *et al.* 2011). Optimization problems are often identified in industries, academic research and application technology. However, the optimization problem is becoming significantly bigger and more complicated. Furthermore, the constraint is becoming more unjustified, the computational complexity becomes heavier, the inherent factors such as imprecision in quantification, the indeterminacy of mechanics inevitably are becoming more serious, preceded by rapid scientific advances and technological innovation. Several complications exist in these systems that could not be addressed by conventional optimization approaches. Research on novel optimization algorithms is always needed in the comparative field. According to the “No Free Lunch” (NFL) theorem, the averaged over all optimization problems, without re-sampling, all optimization algorithms perform equally well (Adam *et al.* 2019). The NFL theorem determines that for every algorithm, any increased efficiency over one class of problems is compensated by success over another class. Therefore, the performance of an algorithm relies on the problem it wants to address. The NFL Theorem indicates that it is difficult to look for a general algorithm for all optimization problems, and it also implies that an optimization algorithm will be more advanced in some optimization problems than others (Rouder *et al.* 2016).

The Traveling Salesman Problem (TSP) is probably the most successful search problem in combinatorial optimization, which gained a great deal of attention from researchers in the field of combinatorial optimization due to its practical applications in commercial, industrial and engineering fields. The salesman begins to move from an arbitrarily defined station and returns to the station after reaching  $n$  nodes. The task is to decrease the salesman's overall distance (Hameed *et al.* 2017). The mathematical problems of the TSP were dealt with in the 1800s by the Irish mathematician Sir William Rowan Hamilton and the British mathematician Thomas Pennington Kirkman (Matai *et al.* 2010). The problem was formulated by Karl Menger in 1930 for the first time (Chiang & Wahid 2014). A salesman and a list of towns have been given in TSP. The seller had to travel to visit all the cities to sell his goods and return to the city from which he started. The aim is to reach all the cities with a minimum total distance and return to the starting point. This problem reflects many problems, such as the TSP with pick-up and delivery, TSP time windows, vehicle routing problem (VRP), china postman, multi-depot TSP, multiple travelling salesman problems (MTSP) and others. The extensive use of TSP in other discipline and its ability to solve everyday problems, despite optimization the problems like scheduling (Yuliastuti *et al.* 2017), clustering (Wissink 2019), minimization and maximization versions of the quadratic travelling salesman problem (Oswin *et al.* 2017). An enhanced lower bound for the Time-Dependent Travelling Salesman Problem (Adamo *et al.* 2020), The travelling salesman problem and adiabatic quantum computation (Kieu 2019). These types of TSP problem can be modified and resolved using existing algorithms and can be used as a reference point for testing new algorithms. TSP has gained a great deal of attention and motivated scholars to study it in recent years.

## 2. Election Algorithm (EA)

This section presents the working principle of the Election algorithm (EA). Figure 1 shows the flowchart of the working principle of the EA. It is a metaheuristics algorithm inspired by the socio-political phenomenon of presidential election conducted by the majority of the country in the world introduced by (Emami & Derakhshan 2015). This EA is an evolutionary algorithm that uses as a source of inspiration from the socio-political competition between candidates. Inspired by the random and evolutionary search strategy, EA relies on an intelligent search by implementing 3 iterative operators i.e. positive advertisement, negative advertisement and

coalition. Each of the operators comprises individual that can be effectively divided into candidate and voters. Similar to the actual electoral system where a candidate must be initially selected from the party and the best candidate will end up with the most votes. In this situation, the candidate will assert dominance and influence to their supporters (voter) and increase the chances of the candidate to win the election. Interestingly, this algorithm provides partitions in a solution space where each partition is represented in terms of party and coordinated by one candidate. Each party will optimize both voters and candidate until the election day. In this paper, we will utilize EA to find the optimal assignment for Travelling Salesman Problem that minimizes the cost function.

In general, the global optimization can be demonstrated below (without the loss of generality minimization problem being considered as;

$$\max |f_{TSP}| \quad (1)$$

$$f_{TSP} = \prod_{i=1}^{N-1} \alpha_i \quad (2)$$

where  $f_{TSP}$  is the eligibility function (Objectives),  $N-1$  describes as the number of cities in TSP and  $\alpha_i$  designates the number of cities measured by the EA.

Election algorithm (EA) is originally designed to solve the continuous optimization problem (Emami & Derakhshan 2015). However, EA can be discretized to solve a combinatorial optimization problem, such as the Travelling Salesman Problem (Dorigo & Gambardella 1997). The main idea in this article is to demonstrate that EA can be used to optimize TSP and in a reasonable period. several algorithms have been added to the TSP, including reliable, heuristic, and metaheuristic, and there has been so much attempt to find a better solution. Exact approaches to optimizing the TSP are used effectively only for comparatively small problems, however, based on different strategies they can guarantee optimality. Such techniques apply algorithms that produce a lower as well as an upper limit on the true minimum value of the problem example. If the upper and lower borders coincide, there is proof for reaching optimality.

Many researchers have suggested similar algorithms to solve the TSP. Such techniques are based on Lagrangian relaxation (Zamani & Lau 2010), branch-and-cut (Dumitrescu *et al.* 2010), branch and bound (Mataija *et al.* 2016). TSP heuristics typically fall within two groups called tour construction heuristics that produce scratch tours and tour optimization heuristics that use basic local search heuristics to upgrade existing tours. A few of the well-known heuristic techniques are gravitational emulation local search (Rostami *et al.* 2015), random gravitational emulation (Nodehi *et al.* 2016), Black holes (Hatamlou 2018), genetic algorithm (Moreno 2008), discrete spider Monday search (Akhand *et al.* 2020), firefly algorithm (Jati *et al.* 2013) and simulated annealing (Wang *et al.* 2013). To arrive quickly at useful solutions, heuristics approximation methods, are usually necessary, but as a rule, they do not provide an estimate of the quality of the solutions found. Depending on whether a heuristic constructs a new tour or attempts to improve an existing itinerary, it is referred to as an opening (or construction) or improvement process. Metaheuristics are methods using heuristics to glance for a sub-optimal solution with a reasonable computational time. The general plan is to create consistency between local adjustments and a high-level strategy through a guided search of the solution space; they optimize problems (Vashisht & Choudhury 2013).

In short, though attempting to reduce computational time, meta-heuristics try optimality. Their main objective is to search the space for near-optimal solutions in an efficient way. Since meta-heuristic methods are very powerful to avoid optimal local values, they are considering as one of the strongest algorithms to solve combinatorial optimization problems. The limitation recognized Election algorithm as recognized by (Emami & Derakhshan 2015) is the use of the adverse Euclidean distance metric measure in negative campaign operator as well as during the formation of the original parties. The Euclidean distance measure adversely affects the computation velocity of the algorithm. This study focused on comparing the different negative campaign operators in searching for lowest total distance and the shortest computational time in searching for an optimal solution to TSP.

### 3. Research approach

The EA is an effective search and evolutionary strategy that uses an intelligent search mechanism by implementing advertisement and coalition operation to optimize a given problem using the best idea to thrive. In this paper, EA will be applying in optimizing TSP which is one of the most important discrete optimization problems.

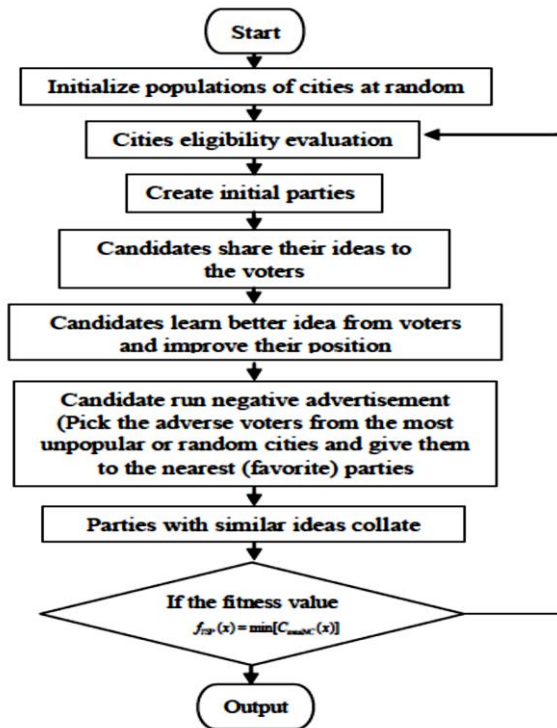


Figure 1: Flowchart of the Election algorithm

#### Stage 1: Initialization

A random population  $N_{POP}$  of individuals that consist of voters and candidates (TSP assignment)  $S_i \in [S_1, S_2, S_3, \dots, S_N]$  are initialized. The state of each individual corresponds to the

possible assignment for TSP problem. The EA procedure begins by splitting the entire population into parties.

**Stage 2: Create initial parties and their supporters**

Select the best 6% of the population which is an appropriate value to serve as the initial number of candidates in forming the initial parties. Thus, the number of individuals in the population that is selected as the initial party's candidate is defined as,

$$\alpha_C = C_r \times N_{POP} \tag{1}$$

where  $\alpha_C$  is the total number of candidates,  $C_r$  is the candidate rate,  $N_{POP}$  is the total population. The remaining persons are the total number of initial supporters of the mentioned candidate. The remaining are considered as the total number of such candidates' supporters in the solution space, which can be described as follows,

$$N_V = N_{POP} - \alpha_C \tag{2}$$

where,  $N_V$  is the total number of voters in the solution space.

**Stage 3: Electoral Parties encoding for TSP**

The TSP can be conveniently expressed in two different ways by a set of integers. For instance, with seven cities, the first route is through the string  $v = 5\ 4\ 3\ 6\ 1\ 7\ 2$ , which means that the tour goes from 5 to 4 and then from 3 to 6, and from 6 returns to 1 (Moon *et al.* 2002) and (Bontoux *et al.* 2010). The second way to express the TSP is through loop notation presented in (Vashisht & Choudhury 2013), with an integer string  $v = b_1 b_2 \dots b_n$ , where the tour travels from two  $i$  to city  $v = b_i$ . That is, the integer  $v_1 = 5\ 4\ 3\ 6\ 1\ 7\ 2$  means that the tour travels from city 1 to city  $v = b_1 = 5$ , city 5 to city  $v = b_4 = 6$ , city 6 to city  $v = b_6 = 7$ , city 7 to city  $v = b_7 = 2$ , city 2 to city  $v = b_2 = 4$ , city 3 to city  $v = b_3 = 3$  and city 5 to city  $v = b_5 = 1$ . It should be noticed that not every feasible sequence here reflects a legal tour in which a legal tour is a one-time tour of each city and returns to the first city. An unauthorized tour that reflects disjoint phases is conceivable, for example,  $v_2 = 4\ 7\ 5\ 3\ 1\ 6\ 2$  implies that tour goes from city 1 to city 4 and returns to city 1.

The first is used to implement the proposed algorithm on the TSP and the first component is set to the city. Therefore, a computer must generate a predetermined number of cities randomly. The use of a random design at this point leads to solutions that have an unusual structure in a feasible space.

Table 1: Coding of a tour

1	3	5	6	4	7	2	4
---	---	---	---	---	---	---	---

**Stage 4: Eligibility evaluation**

All the variable will undergo eligibility evaluation ( $f_{TSP}$ ). Each of the correct variables which result in correct TSP will be “awarded”. During the eligibility evaluation, the number of correct

TSP will represent the eligibility of the persons (variable). The objective function of the EA is as follows:

$$f_{TSP} = \sum_{i=0}^m \alpha_i = \alpha_1(x) + \alpha_2(x) + \alpha_3(x) + \dots + \alpha_{N_{var}}(x) \quad (3)$$

where  $\alpha_1, \alpha_2, \dots, \alpha_{N_{var}}$  is the variable measure by EA and  $N_{var}$  is the number of variables present in the search space. Specifically, the role of the eligibility function is to evaluate the goodness of each variable in predicting the correct position.

### Stage 5 Advertising Campaign

#### Positive advertisement

The process of modelling EA positive mechanism, the vector variable of a candidate in the solution space is randomly selected. Random numbers are sampled to pick the position of vector variables (voters) to be substituted. The selection rate is denoted by  $\lambda_s = rand \in [0,1]$ . The number of variable vector values that are transferred from a candidate toward its supporters is represented in Equation (6) modelled in (Abubakar *et al.* 2020).

$$\alpha_s = \lambda_s \times S_C \quad (4)$$

where  $\alpha_s$  is defined as the number of sampled vector variables to be replaced  $\lambda_s$  define the selection rate and  $S_C$  described the total number of vector variables of the candidate in the solution space. In an EA, party eligibility Euclidean metric between a candidate and its relevant supporters, the effectiveness of advertisement varies. Positive advertisement happens, whereby the candidates that seem to have an excellent plan and idea if the decision was taken by the voters is to be influenced, the number of its supporters will increase and the chances of increasing the quality of the party's plans will increase.

To model this goal, we represented eligibility distance coefficient ( $\psi_e$ ) as follows,

$$\psi_e = \frac{1}{dist(M_e, M_v) + 1} \quad (5)$$

where  $M_e$  and  $M_v$  designated the eligibility of candidate  $e$  and voter  $s$ , respectively. EA applied the Euclidean metric to measures the distance between the vector variable  $M_e$  and  $M_v$  in the solution space (Abubakar *et al.* 2020). In EA, the advertising mechanism, after selecting  $\alpha_s$  and measured  $\psi_e$  the vector values of identified vector variables from the candidate are multiplied in coefficient  $\psi_e$  and the replaced with the identified vector of the associated voters. In other words, given  $e_{iold}$  be the value of  $i$ th chosen vector variables of the voters before advertising advances, then after a campaign, the updated value of the identified vector is given as follows,

$$e_{inew} = \psi_e \times e_{iold} \quad (6)$$

Based on the Eq. (33) in the campaign process, the near supporters are much more influenced by their associated candidate than by other followers.

### Negative advertisement

In the implementations of EA, contrast advertisement is used among different negative campaigning strategy. Candidates, by their campaign of resistance, seek to fascinate the members of other parties towards themselves. This leads to an upsurge in support of the popular parties and a decline in popularity of the marginalized parties. The difference in eligibility between the voters and the candidates is measured at first in the defender group by applying the Euclidean metric as follows,

$$dist(r_i, t_j) = \|r_i - t_j\| = \sqrt{\sum_{j=1}^{M_d} (E_{r_i} - E_{t_j})^2} \quad (7)$$

where  $M_d$  described the number of all voters in the defender party,  $t_i$  defined the candidates of defender party and  $r_i$  defined the  $i$ th supporter of the defender party.  $E_{r_i}$  is the eligibility of candidate  $t_i$  and  $E_{t_j}$  is the eligibility of candidate  $i$ th supporter.

### Stage 5: Coalition

Candidates confederate if they shared the same ideas; In EA, sometimes two or more parties with the same ideas and goals in solution space can come together to create a new party. Until a condition of termination is met, three different operators, positive advertising, negative advertising and coalition will be applied to update the population. Like the process of candidate coalition, a candidate will form a partnership with an individual (voter and candidate) from another party. In this case, the parties will exist co-dependently with each other. The effect of both candidates from both parties within the same coalition will be computed based on Equation (8).

### Stage 6: Election Day

Until a condition of termination is satisfied, three different operators, positive advertising, negative advertising and coalition will be applied to update the population. Ultimately a candidate who gets the most votes will declare himself the winner and is equal to the best solution found for the problem of optimization and search. If the termination criteria for Stage 3-5 is satisfied, the election will be conducted to evaluate the final eligibility of all the candidate. If the  $f_{TSP} = m + n$  candidate will be elected else repeat stage 3-5 until the number of iterations is reached.

## 4. Experimental Setup, Results and Discussions

In this paragraph, the consistency comparison between the proposed algorithms is recorded. We incorporated all EA with various negative campaign operators in C++ on a Core i5 2.30GHz

processor PC and 8 GB RAM with Windows 8. Every EA of specific negative campaign operator is conducted on TSP 10 times for statistical comparison. In this case, the population of 1000 is used, the iteration is set at 1000, we defined the total number of cities as 50 and the number of constrained cities covered is 10, 20, 30 and 40.

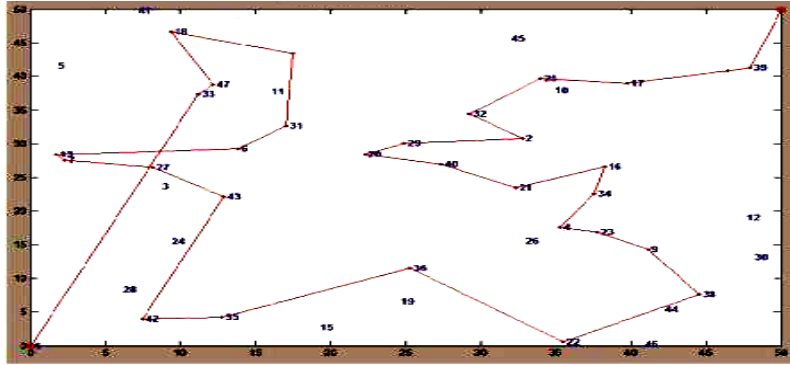


Figure 2: TSP with random supporters

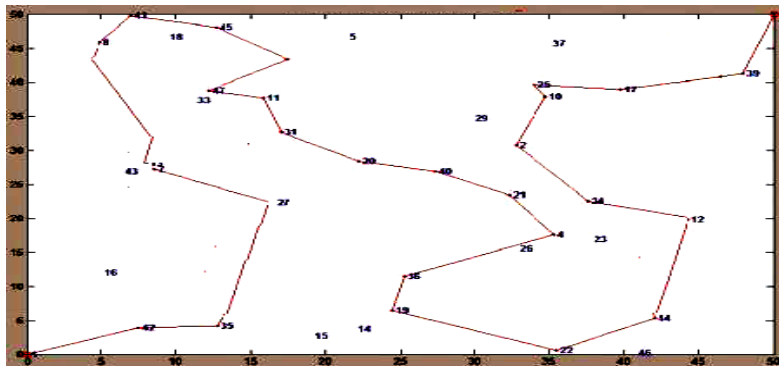


Figure 3: TSP with Nearest supporters

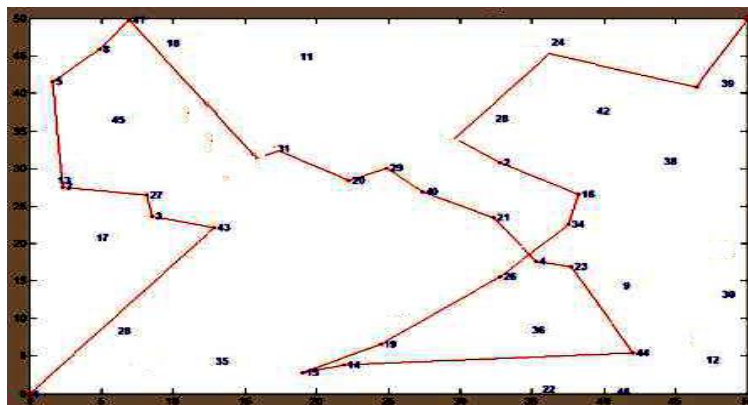


Figure 4. TSP with Furthest supporters



Table 2: Mean of minimum tour distance for different negative campaign operators of EA

No. of city visited ( $N$ )	Random supporters	Nearest supporters	Furthest supporters
10	115.1791	119.6361	126.8613
20	184.3624	188.0269	194.9301
30	232.1828	247.0380	238.5520
40	265.7966	328.0730	271.7210

Table 3: Mean iteration for different negative campaign operator of EA

No. of city visited ( $N$ )	Random supporters	Nearest supporters	Furthest supporters
10	37.6	37.5	18.5
20	190.8	356.9	149.6
30	574.6	651.1	266.4
40	894.4	943.6	450.3

Table 4: Mean computational time for different negative campaign operators of EA

No. of city visited ( $N$ )	Random supporters	Nearest supporters	Furthest supporters
10	11.4	12.2	5.1
20	12.2	13.2	6
30	15.1	15.1	7
40	16.8	16.88	8.1

The main focus of this study is comparing the negative campaign operators of EA in searching the lowest total distance and the shortest computational time of TSP. The numbers are the cities toured and the red line is the tour's route. The computational result of TSP with using different negative campaign operator of EA has been displayed in Figure 2, Figure 3 and Figure 4. In Figure 2 the performance of TSP using random supporters has been demonstrated, which revealed that the route crossing through other cities significantly affect the optimality of the result generated and increase the computational time of the EA. The similar crossing was also observed by nearest supporters in Figure 3. The TSP performance based on furthest supporters was displayed in Figure 4, which trends display non-crossing path that random supporters and the nearest supporters have generated.

In Table 2, 3 and 4, the average of minimum tour distance, average iteration and average computational time was recorded. It is observed that EA with random supporters displays excellent in achieving the minimum distance for all four separate limited numbers of visited towns. Relatively, EA has not produced a good compromise with the closest supporter and furthest supporters since they tend to make the roads that cross each other as shown in Figure 2 and Figure 4. Essentially, a route with a minimum total distance travelled should not have two paths crossing each other as the overall crossing path distance is greater than the non-crossing paths. The EA contributes less iteration and computation time with the furthest

supporters. This is due to each negative operator's ability to contribute their optimal solutions and increase the likelihood of searching for the optimal solution.

## 5. Conclusion

This paper explores EA's success in a TSP problem with various negative campaign operators. The result shows that EA was better at reducing the total distance by using a random operator. Moreover, the experimental results also demonstrate that EA shows great promise in reducing the computational cycle with the furthest supporter's operators. It still depends on how the question is represented and how the correct operator is used to produce a good solution for the TSP using EA. The creation of innovative-negative campaign operators for the problem of travelling salesmen problem and satisfiability optimization may be the focus of future research. We are also planning to implement EA for the next step in addressing other combinatorial problems such as satisfaction. Thus, the result can also be used to compare with EA with other algorithms when doing TSP to see which algorithm has a better performance.

## Acknowledgement

This research is supported by the Research University Grant (RUI) by Universiti Sains Malaysia (1001/PMATHS/8011131).

## References

- Abubakar H., Rijal S., Sabri M., Masanawa S. A. & Yusuf S. 2020. Modified election algorithm in hopfield neural network for optimal random  $k$  satisfiability representation. *International Journal for Simulation and Multidisciplinary Design Optimization* **11**(6): 1-13.
- Adam S. P., Alexandropoulos S. A. N., Pardalos P. M. & Vrahatis M. N. 2019. No free lunch theorem: A review. *Springer Optimization and Its Applications* **145**: 57-82.
- Adamo T., Ghiani G. & Guerriero E. 2020. An enhanced lower bound for the Time-Dependent Travelling Salesman Problem. *Computers and Operations Research* **113**, 104795.
- Akhand M. A. H., Ayon S. I., Shahriyar S. A., Siddique N. & Adeli H. 2020. Discrete spider monkey optimization for travelling salesman problem. *Applied Soft Computing* **86**,105887.
- Blum C., Puchinger J., Raidl G. R. & Roli A. 2011. Hybrid metaheuristics in combinatorial optimization: A survey. *Applied Soft Computing Journal* **11**(6): 4135-4151.
- Bontoux B., Artigues C. & Feillet D. 2010. A memetic algorithm with a large neighborhood crossover operator for the generalized traveling salesman problem. *Computers and Operations Research* **37**(11): 1844-1852.
- Chieng H. H. & Wahid N. 2014. A performance comparison of genetic algorithm's mutation operators in n-cities open loop travelling salesman problem. In Herawan T., Ghazali R. & Deris M. (eds.) *Recent Advances on Soft Computing and Data Mining. Advances in Intelligent Systems and Computing*: 89-97. Cham, Switzerland: Springer.
- Dorigo M., & Gambardella L. M. 1997. Ant colonies for the travelling salesman problem. *BioSystems* **43**: 73-81
- Dumitrescu I., Ropke S., Cordeau J. F. & Laporte G. 2010. The traveling salesman problem with pickup and delivery: Polyhedral results and a branch-and-cut algorithm. *Mathematical Programming* **121**(2): 269-305.
- Emami H. & Derakhshan F. 2015. Election algorithm: A new socio-politically inspired strategy. *AI Communications* **28**(3): 591-603.
- Hameed W. M. & Kanbar B. A. 2017. A comparative study of crossover operators for genetic algorithms to solve travelling salesman problem. *International Journal of Research-Granthaalayah* **5**(2): 284-291.
- Hatamlou A. 2018. Solving travelling salesman problem using black hole algorithm. *Soft Computing* **22**(24): 8167-8175.
- Jati G. K. & Manurung R. 2013. Discrete firefly algorithm for traveling salesman problem: A new movement scheme. In *Swarm Intelligence and Bio-Inspired Computation*, pp. 295-312.
- Kieu T. D. 2019. The travelling salesman problem and adiabatic quantum computation: an algorithm. *Quantum Information Processing* **18**(3), 90.
- Matai R., Singh S. & Lal M. 2010. Traveling salesman problem: An overview of applications, formulations, and solution approaches. In *Traveling Salesman Problem, Theory and Applications*. <https://doi.org/10.5772/12909>.
- Mataija M., Rakamarić Šegić M., & Jozić F. 2016. Solving the travelling salesman problem using the branch and bound method. *Zbornik Veleučilišta u Rijeci* **4**(1): 259-270.
- Moon C., Kim J., Choi G. & Seo Y. 2002. An efficient genetic algorithm for the traveling salesman problem with

- precedence constraints. *European Journal of Operational Research* **140**(3): 606-617.
- Moreno A. G. 2008. Solving travelling salesman problem in a simulation of genetic algorithms with DNA. *Information Theories & Applications* **15**(4): 357-363.
- Nodehi A. N., Fadaei M. & Ebrahimi P. 2016. Solving the traveling salesman problem using randomized gravitational emulation search algorithm. *Journal of Current Research in Science* **5** (2): 818.
- Oswin A., Fischer A., Fischer F., Meier J. F., Pferschy U., Pilz A. & Staněk R. 2017. Minimization and maximization versions of the quadratic travelling salesman problem. *Optimization* **66**(4): 521-546.
- Rostami A. S., Mohanna F., Keshavarz H. & Hosseinabadi A. R. 2015. Solving multiple traveling salesman problem using the gravitational emulation local search algorithm. *Applied Mathematics and Information Sciences* **9**(2): 1-11.
- Rouder J. N., Morey R. D., Verhagen J., Province J. M. & Wagenmakers E. J. 2016. Is there a free lunch in inference? *Topics in Cognitive Science* **8**(3): 520-547.
- Vashisht V. & Choudhury T. 2013. Open loop travelling salesman problem using genetic algorithm. *International Journal of Innovative Research in Computer and Communication Engineering* **1**(1):112-116.
- Wang Y., Tian D. & Li Y. H. 2013. An improved simulated annealing algorithm for travelling salesman problem. *International Journal of Online Engineering* **9**(4): 28-32.
- Wissink P. L. J. 2019. *The Probabilistic Travelling Salesman Off the Beaten Track*. The University of Edinburgh. <https://era.ed.ac.uk/handle/1842/35764>.
- Yuliasuti G. E., Mahmudy W. F. & Rizki A. M. 2017. Implementation of genetic algorithm to solving travelling salesman problem with time window (TSP-TW) for scheduling tourist destinations in Malang City. *Journal of Information Technology and Computer Science* **2**(1): 1-10.
- Zamani R. & Lau S. K. 2010. Embedding learning capability in Lagrangean relaxation: An application to the travelling salesman problem. *European Journal of Operational Research* **201**(1): 82-88.

*School of Mathematical Sciences*

*Universiti Sains Malaysia*

*11800 USM, Penang*

*MALAYSIA*

*E-mail: zeeham4u2c@yahoo.com, saratha@usm.my\*, shehab\_alzaeemi@yahoo.com*

Received: 5 April 2020

Accepted: 13 September 2020

---

\*Corresponding author