J. Appl. Comput. Mech., 5(4) (2019) 696-703
DOI: 10.22055/JACM.2019.27219.1384

ISSN: 2383-4536
jacm.scu.ac.ir

M&MoCS
Shahid Chamran
University of Ahvaz

Journal of
Applied and Computational Mechanics

Research Paper

The Urban Path Routing Adjustable Optimization by Means of Wavelet Transform and Multistage Genetic Algorithm

Seid Miad Zandavi[1], Hamoon Pourmirzaagha[2], Alireza Yekrangi Sendi[2], Ershad Sadeghi Toosi[3]
Mostafa Zakariapour[2]

[1] Department of Aerospace Engineering, Sharif University of Technology, 1458889694 Tehran, Iran
[2] Department of Mechanical Engineering, Ramsar Branch, Islamic Azad University, Ramsar, Iran
[3] Department of Physics, Neka Branch, Islamic Azad University, Neka, Iran

Received October 01 2018; Revised January 11 2019; Accepted for publication January 19 2019.
Corresponding author: Alireza Yekrangi Sendi, yekrangi_ali@yahoo.com

Abstract. This paper introduces the optimization algorithm to improve search rate in urban path routing problems using viral infection and local search in urban environment. This algorithm operates based on two different approaches including wavelet transform and genetic algorithm. The variables proposed by driver such as degree of difficulty and difficulty traffic are of the essence in this technique. Wavelet transform as the first part of proposed algorithm derives edges risk. Finally, multistage genetic algorithm operates to find the optimal solution which is defined as the shortest path. The proposed algorithm is applied to the case study. The performances of the algorithm is investigated by comparing with other methods.

Keywords: Adjustment Parameter, Multistage Genetic Algorithm, Routing Optimization, Wavelet Transform.

1. Introduction

Various algorithms are employed with multimodal time including Dijkstra's and Floyed-Warshall algorithms to solve the shortest path problem in deterministic graphs in which edge weights are independent, fixed and predetermined values. If edge weights vary dynamically in these algorithms, one may not find the optimal solution since one should analyze total graph again with the least change in graph. For this reason, the researchers consider algorithms for shortest path problems in dynamic graphs, i.e. the graphs in which edge weights and/or graph structure varies dynamically over the time [1]. Moreover, these algorithms in which edge weights are stochastic variables for the stochastic graphs lack high efficiency. Thus some algorithms are considered to solve shortest path problem in stochastic graphs. Frank [2] was the first one who started to study on stochastic grids. The probabilistic distribution of edge weights should be already clear in graph in these techniques while in many real situations the edge probabilistic distribution is not clear. Beigy and Meybodi have proposed the algorithm to solve shortest path problem between two nodes used in stochastic graphs [3]. Misra and Oommen have proposed the algorithm to find the shortest path among an origin, other nodes designed and all pairs of graph nodes in a stochastic graph [4].

The cost of edges is supposedly independent from each other in the graph. In many real applications, edge costs are not completely independent from each other; as traffic changes in a part of grid, traffic also varies in adjacent sections. Burton [5] and Waller and Ziliaskopoulos [6] explored subject of the probabilistic shortest path problem with correlated edges under the conditions considering with and without traffic.

The method suggested in this paper operates according to wavelet transform and genetic algorithm and driver receives presented variables such as difficulty order and traffic. Then the risk wavelet is transformed to acquire edges and finally the shortest path was achieved by multistage genetic algorithm. The key leverage is to find the shortest urban path considering different difficulty orders probably occurred in traffic.

2. Proposed Model

The given technique will be introduced as follows in this study: 1) to determine main criteria to select a path, 2) to propose a transparent definition for each of criteria, 3) to introduce suggested quality metric in evaluation technique through wavelet transform and genetic algorithm.

2.1 Statistical Weight for any Edge

In order to calculate statistical weight for any edge, one should extract periodic parameters for traffic data. Regarding this, argument space could be converted into frequency space by wavelet transform in traffic signal. The wavelet transform is superior than other techniques, such as Fourier's transform in that the place was identified where a certain frequency occurs in argument space in wavelet transform. This is highly important when periodic traffic parameters was extracted. We can drive all basic functions in the wavelet transform from transition and a function scale (mother wavelet).

The Wavelet functions may be acquired by adding two transform and scale functions over mother wavelet as in eq. (1):

$$\psi_{a,b}(x) = \left(1/\sqrt{|a|}\right)\psi\left[(x-a)/b\right] \quad a,b \in R \tag{1}$$

In this expression, ψ denotes mother wavelet, a expresses scale parameter, and b is transform parameter. The coefficient $a^{-1/2}$ is added to normalize various scales. Scale parameter is proportional to frequency inverse; namely, as scale becomes smaller, frequency goes higher and vice versa. Wavelet transform operation can be summarized in five steps as follows:

1. Selecting the given mother wavelet and dividing it by a section at signal onset.
2. Computing similarity among wavelet and this part in signal: $C = W\psi$ .As this figure is greater, the similarity rate will be higher.
3. Moving wavelet to right side and repeating steps 1 and 2 to cover signal totally. In fact, the researchers have identified where a certain frequency has occurred in signal. As it is implied, one can calculate signal frequency by wavelet scale. Therefore, any part in signal is more similar to wavelet at this scale which includes frequency nearer to the relevant frequency to that scale at that part.
4. Changing wavelet scale and repeating Steps 1-3; in order to determine location for other frequencies.
5. Repeating steps 1-4 for all scales.

It was found that the periodic traffic parameters are as follows after wavelet transform over traffic sample data for one year:

- Parameter with daily frequency period (T_D) will repeat after 24 hours.
- Parameter with weekly period (T_W) will repeat after a week.
- Parameter with annual frequency period (T_Y) will repeat after a year.

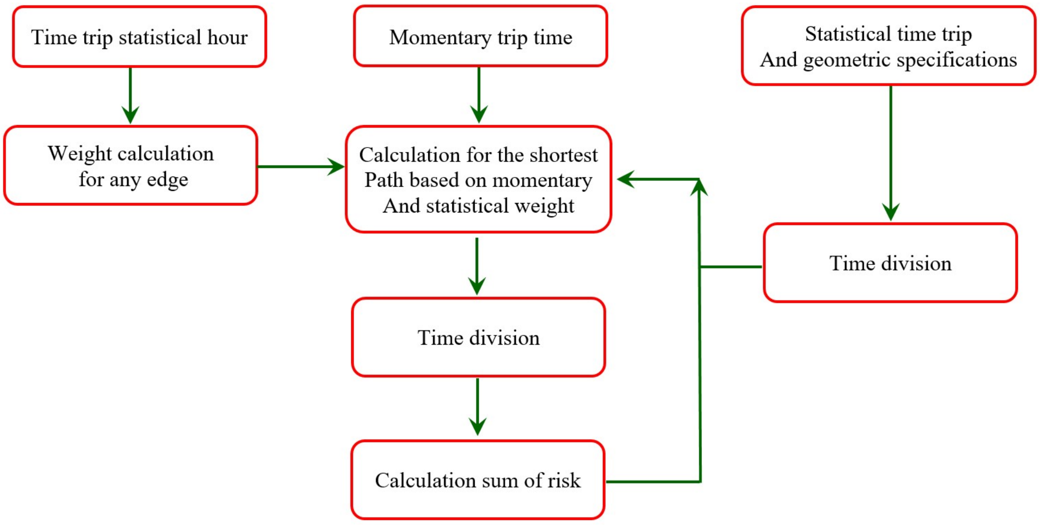


Fig. 1. Execution trend for suggested technique to risk calculation

The researchers employed this technique mainly in regression analysis and formed equation coordinates as in eq. (2):

$$T_i = a_i T_{D_i} + a_{i+1} T_{W_i} + a_{i+2} T_{Y_i} \tag{2}$$

In these equations, T_i is trip time momentary observation at edge e_i (using mobile and fixed metrics); T_{D_i} denotes edge trip time at momentary observation hour during last year and T_{W_i} expresses mean time for edge trip at momentary observation day in the week during last year and T_{Y_i} implies mean time for edge trip at momentary observation during last year and a_i, a_{i+1}, and a_{i+2} are unknown items show each of parameters. Number of these equations is the same as quantity of edges in the grid in which momentary trip time is visible. The researchers finally estimated the unknown items based on eq. (3) using Least

Squares technique.

$$L = AX \rightarrow \hat{X} = (A^T A)^{-1} A^T L \tag{3}$$

Statistical trip period was predicted in each of edges for future hours after estimating weight for each of traffic parameters.

2.2 Risk Calculation for any Edge

Risk for any edge indicates probability for occurring unpredicted events such as accidents of vehicles and failure of routes in calculating trip time at previous section. They have not paid due attention to path risk in exposing to predict events in best path algorithms at dynamic transportation network [7]. The researchers have supposedly considered the best path in terms of time as a target so that we presented general technique for risk calculation in Fig. 1.

The researchers considered two edges along with each other to calculate risk for any edge: 1) probability for unpredicted events occurs in an edge, 2) impact for unpredicted events occur in an edge.

$$\text{Risk}_{e_i} = \text{T}_i + (x - 0.001) \times (\text{T}_{avg} - \text{T}_i) / (0.05 - 0.001) \tag{4}$$

2.3 Path length

If $path\_length_1$ expresses this property ($i = 1$) and path j is the shortest length among candidate paths from first generation to present generation then it will be $path\_length_1^j = 1$. This relation will be expressed in Eq. (5) for other existing paths in present generation.

$$path\_length_1^k = (LV_{\max} - LV_k) / (LV_{\min} - LV_k) \tag{5}$$

where $LV_{\min}$ and $LV_{\max}$ are the best and worst length value in total populations, respectively. Similarly, this property can be extended to path length. If $path\_length_2$ expresses this property ($i = 1$) and path j is the shortest length among candidate paths from first generation to present generation then it will be $path\_length_2^j = 1$. This relation will be shown in Eq. (6) for other existing paths at present generation.

$$path\_length_1^k = (TV_{\max} - TV_k) / (TV_{\min} - TV_k) \tag{6}$$

where $TV_{\min}$ and $TV_{\max}$ are the best and worst time period in total populations, respectively. In order to calculate time for passing through each of paths and crossroads, time lag function was used for passing through path- prepared volume for each of passages in Tehran city [8] as well as the related model to time lag in each of crossroads with traffic lights and without traffic lights.

2.4 Path Density

The well-trained expert for path density will identify all of our main routes between two flags to determine the start and end points at path density. In other words, with respect to intensity in path density the expert determined it. The expert has identified all paths located between two flags by considering the coefficients at any density level (depending on intensity in path density). Two paths with flag mark will determine path density as equal to weighted mean (based on length) in both ends of flag at path servicing levels. All vehicles will supposedly move smoothly before joining and after separating from path density queue. Weighted coefficients used in this study are *very heavy traffic jam*, *traffic jam*, *traffic jam on move*, and *smooth motion* being valued 1.0, 0.64, 0.40, and 0 respectively. In other words, we calculate C_p value as density of the path related to p-path for various intensities in path density.

2.5 Path Difficulty

The difficulty for any path may be function of type and nature of the given path. If $Diff_X$ expresses difficulty of a path in this case trivial value (0-value) denotes ideal driving conditions in the given path; 1- value indicates the highest difficulty order for passing freely from x-path. Allocation of various weights is an appropriate method to determine path difficulty. In this regard, highway with negative gradient, highway without gradient, highway with positive gradient, first-order artery outside central limit, first-order artery inside central limit, second order-artery, pick-up and delivery type, and local access are suggested in terms of difficulty in any paths considering suggested weight ranging from 0 to 1.

2.6 Initialization

The initial properties will be calculated for path length and time for moving through the path in any candidate path with rang [0, 1]. If the existing passages in a path is supposed as a part of that path, so that any part of our path is located between two crossroads. Then, any part will include various values in terms of initial properties for path density and difficulty order. Whereas the researchers intended to calculate initial properties perfectly for any candidate path thus necessarily it is needed to introduce a method for integrating the related properties in any part in order to achieve certain value for path density and difficulty order specifications at any candidate path.

If we suppose N_n as number of the related parts to the studied path proposed for L_x and ρ_x show length and density for the related path to x-part, then we can calculate W_x the weight relating to any part in candidate path as in Eq. (7):

$$W_x = L_x \Big/ \left(\sum_{x=1}^{N_n} L_x \right) \tag{7}$$

Thus, we can compute path density for total suggested path in Eq. (8).

$$CV_k = \sum_{x=1}^{N_n} W_x \times \rho_x \tag{8}$$

whereas we have $\sum_{x=1}^{N_n} W_x = 1$ and $0 < \rho_x < 1$, path density value will also locate within range [0,1] for k^{th} candidate path (CV_k).

If *path_length₃* expresses initial property in path density for any path then we can calculate it for the existing paths in present generation in Eq. (9).

$$path\_length_3^k = (CV_{\max} - CV_k)/(CV_{\min} - CV_k) \tag{9}$$

where $CV_{\min}$ and $CV_{\max}$ are the best and the worst path density value in total populations respectively. Similarly, the researchers employed the given technique to calculate difficulty order value ($path\_length^k{}_4$) related to each of the existing candidate paths in any generation. The researchers derived optimal solution for any criterion by using Dijkstra's algorithm independently in this study. Thus, the ideal path will be one for which we can call the related descriptive vector as ideal and we have obtained it by using Dijkstra's algorithm independently.

In order to measure quality in the best suggested compatible response in which the suggested metric not to be influenced by measurement scale from the existing targets, therefore if we call the related descriptive vector for that path as ideal, we can use metric independent from weighted l_p norm scale in Eq. (10) as

$$l_p(z;P,w) = \left(\sum_{i=1}^{q} w_i^P \left[\frac{\left| z_i^* - z_i \right|}{z_i^*} \right]^p \right)^{1/P} \tag{10}$$

Weighted vector $W = \begin{bmatrix} w_1 & w_2 & \dots & w_q \end{bmatrix}^T$ denotes the importance in each of criteria z_i introduced by the driver. $(P \geq 1)$ P value determines emphasis degree on biases among each of targets. As this value is greater, the higher emphasis will be on greatest biases.

Whereas all biases are important in the problem, so the value ($p=1$) in Eq. (11) is expressed.

$$l_p(z;1,w) = \sum_{i=1}^{q} w_i \left[\frac{\left| z_i^* - z_i \right|}{z_i^*} \right] \tag{11}$$

As a result, distance from z_i to z^*_i derived from Eq. (11) denotes distance from each of z_i responses from correspondent ideal response z^*_i; namely, if $z^*_i = z_i$, this distance is zero.

3. Multistage Genetic Algorithm

They operate on chromosomes with fixed length by encoding and creating initial population in classic genetic algorithm. Length of any chromosome varies in the suggested genetic algorithm. This issue applies if length of any chromosome does not exceed from maximum N (number of the existing nodes in network). Thus, a chromosome is a sequence from positive integers that refer to unique code for each of nodes (cross points) show a path from origin node to destination node. The researchers intend to create a path (chromosome) using topologic data stored in the given network so that the given node is available with no loop (Iterated passage from a node). Cost function is calculated to evaluate any path based on selected goals weighting technique Sum of weighted Global Ratios (SWGR). If R_k denotes k^{th} candidate path in population, the objective function for the optimization algorithm is formulated by (12) and (13).

$$f(R_k) = \sum_{i=1}^{k} W_i \times path\_length_i^k \tag{12}$$

$$\sum_{i=1}^{4} W_i = 1 \tag{13}$$

in which W_i denotes importance in each of target functions for driver. Design variables to evaluate paths ($path\_length_i^k$) are length (LV) and time period (TV), from which these variables define the shortest path. Having used suggested genetic algorithm by tournament selection without replacement with size-2, two paths are randomly selected by stochastic number generator with the selected uniform distribution and a path with higher fitness. Needless to say that two paths should not be identical. n-point cross over have been employed in the suggested genetic algorithm.

The suggested cross over composition is not similar to the customary technique so that at least one gene (node) should be common in two selected paths for crossover composition in the suggested technique rather than origin and destination nodes while we no longer need to identical position in common nodes (similar to the customary technique). To this end, initially the researchers find common node(s) in both of paths and execute *n*-point crossover based on number on cross points (*n*). Having selected stochastic number generator with uniform distribution to implement randomly mutation operator on the given paths for two genes (nodes), the researchers execute stochastic routing operation based on modified Dijkstra's algorithm to replace former path with new path. It is not allowed to pass iteratively through a gene (node) leading to create loop in path at structure of each of paths while we may create loop in path after execution of *n*-point crossover and mutation operators. Therefore, any path should necessarily be recovered to create the existing path without loop after *n*-point crossover and mutation operators.

After arranging chromosomes in ascending order and selection D members from the best ones in parent group in the studied method, we will transfer $\lambda . D$ members (where λ is a stochastic number and stochastic generator with uniform distribution within range [0, 1]) of this group directly to the next generation that is called elitism. It should be implied that the researchers select D members based on the minimum qualification criterion so that these members include at least fifty percent fitness in best member in this population [9].

Minimal Generation Gap (MGG) method [10] was utilized to change in generation to develop the model. In any change process in population we replace the nearest distance for each of paths by *n*-point crossover technique and cost function assessment and the nearest distance criterion in this method. The researchers select randomly two paths in the existing population as parent without replacement. n-point crossover technique (where n is number of common nodes between two paths) was employed to create two children from two parents. The researchers selected the fittest path among 2 parents and their related 2 children and another member among 3 rest members (except the fittest path) by Roulette Wheel Selection technique. Independent metric was employed from l_p norm weighted scale with P=1 shown in Eq. (14).

$$l_p(z;1,w) = \sum_{i=1}^{q} w_i \left[\frac{\left| z_i^{parent} - z_i^{child} \right|}{z_i^{parent}} \right] \tag{14}$$

Furthermore, virus theory was included to develop this algorithm. The researchers designated part and or some parts of a highway and path of first-order artery and or second- order artery as virus in the suggested technique by considering traffic behavior in various types of city passages in trip from any traffic zone to other traffic zone and stored them in spatial database. Such virus populations form to take trip from a traffic zone to another in addition to path populations. Virus encoding is similar to encoding a path; in other words, it is a sequence in positive integers each of them refers to a cross point.

The virus should not include origin and destination nodes. Only those paths including virus form as initial population. If two similar paths emerge in population one of them will be deleted. Each of emerging paths should be available with no loop in them. In this way, the path changes stochastically from origin node to the beginning node in virus (path 1) and from the end of virus to destination point (path 2) by Dijkstra's algorithm. Then, the final path is composed of path 1, virus, and path 2. This process continues for emerging initial population in genetic algorithm.

The local search is another extension relating to proposing a suitable technique to execute local search based on viral influence called as local infection. Given path is divided into some pieces with equal lengths (d) in the suggested method. If we suppose $N=\{N_1, N_2, \ldots, N_n\}$ as a set of all elements placed on path R_k with equal distances from each other then N_i stochastically is selected from Set N as (N_s) and draw a circle with radius r ($r \leq d$). If v'_k is viral and located totally in the circle we select the point to become closer to N_s.

After v'_k selection if It is a node in path nearer to start point in virus (a) v'_k along with N_{before} and N_{after} among set N to two nodes before and after I_t; also I_{t-1} and I_{t+1} are the nearest nodes to N_{before} and N_{after}, respectively, then path routing changes based on Dijkstra's algorithm (according to length criterion) from I_{t-1} to node at start point in virus (a) and from node at end point in virus (b) to I_{t+1} in order to approach to paths $R_{I_{t-1}\rightarrow a}$ and $R_{b\rightarrow I_{t-1}}$. Then we analyze them whether they have loop or not. Table 1 shows general trend in the designed Genetic Algorithm.

4. Results

The researchers implemented all these operations on a network with 6000 edges and 4000 nodes. A modified Dijkstra's algorithm was employed to create a path based on *d-Heap, s* with d=2 and memory management to exposure to great implementation network. In other words, the researchers weight stochastically each of existing edges in the network with uniform distribution for any time by emerging path. Fig. 2 indicates calculation within time interval (day, week, and year).

In order to determine efficiency in population size proportional to efficiency in the suggested general genetic algorithm, the maximum quantity in generation production was determined as 500 and *n*-point crossover and mutation rate as 0.1. Population size may impact on efficiency of general genetic algorithm as a guiding factor by lower rates for *n*-point crossover and mutation. Fig. 3 implies the best, worst, and average cost functions from 25 independent executions for various population size. With respect to the results, the researchers may observe the population size greater than 50 while population rise has no effect on efficiency in general genetic algorithm. The researchers operated in two ways to achieve the best rate in *n*-point crossover and mutation: The researchers considered mutation rate as 0, change in crossover rate [0.1-0.9], crossover rate as 0, and change in mutation rate [0.1-0.9]. The maximum numbers considered in generation production (1000) and population size as 60 are selected based on ref [11]. Thus, the suitable crossover and mutation rates are 0.7 and 0.2 respectively. In summary, the parameters of genetic algorithms are listed in Table 2.

Table 1. Genetic Algorithm General Trend

```
General GA()
    Set DBMAP
    Set nodes
    Set condition
    Set destination
    Set routes
    Set populations
    Initialization
    Gen=1
    While Gen < Gen_max
        Gen.ran()
        Gen.Fitness()
        Gen.elitism()
        individuals.rand(2)
        Gen.n-pointcrossover()
        Gen.mutation()
        Gen.repair()
MGG()
    Gen.evaluation= 1
    While Gen.evaluation < Gen.evaluation max
        Gen.Fitness()
VG()
    Gen.evaluation= 1
    While Gen.evaluation < Gen.evaluation max
        Gen.Fitness()
        MGG()
LG()
    Gen.evaluation= 1
    While Gen.evaluation < Gen.evaluation um
        Gen.Fitness()
        MGG()
        VG()
```

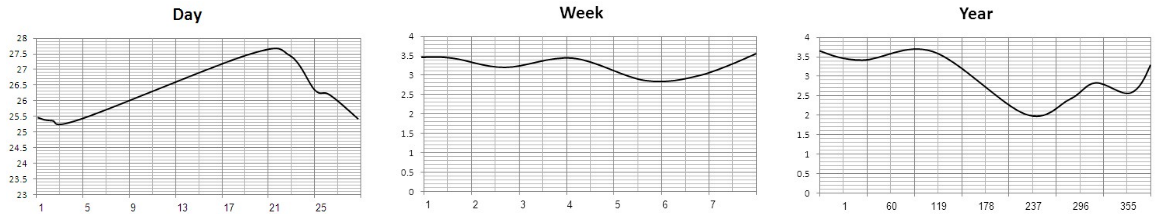


Fig. 2. Time charts for daily, weekly and annual trips

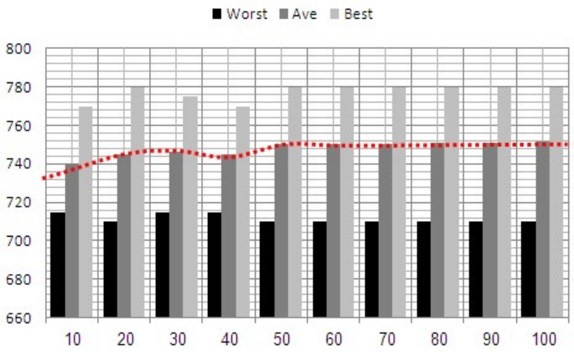


Fig. 3. Comparison among the best, worst, and average fitness in 200- population size in 25 independent executions

Table 2. The parameters of Genetic Algorithm

Parameters	Value
Number of Generation	1000
Population Size	60
Crossover	0.7
Mutation	0.2

Fig. 4 indicates how to change in average fitness and the best average individual cost within 25 independent executions with population size 60 and crossover (0.7) and mutation rates (0.2) along with weights introduced for any criterion. Further, Fig. 5 shows the results for genetic algorithm while it moves from different stages of our method (see Table I) to reach the optimal solution with parameter adjustment.

Moreover, the suggested algorithm was compared Our Method (OM) in Fig. 6 with Tabu search [12], Variable neighborhood Search [13], Record-to-Record Travel [14], and hybrid developmental algorithm [15].

The results signify the suggested technique could not derive very appropriate responses for standard problems. The evaluation model in each of path routing criteria according to deriver's attitude improve search rate in urban path routing problems to find the shortest path. On the other hand, and by comparing with other meta-heuristic algorithm the researchers concluded proposed algorithm has competitive results among others.

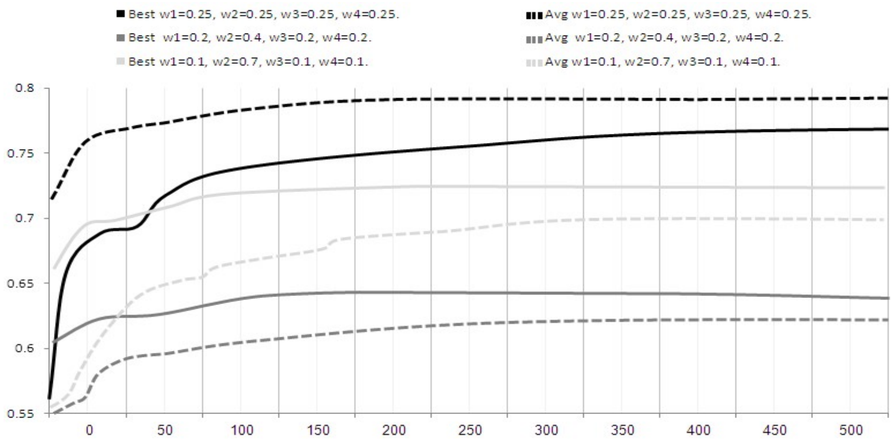
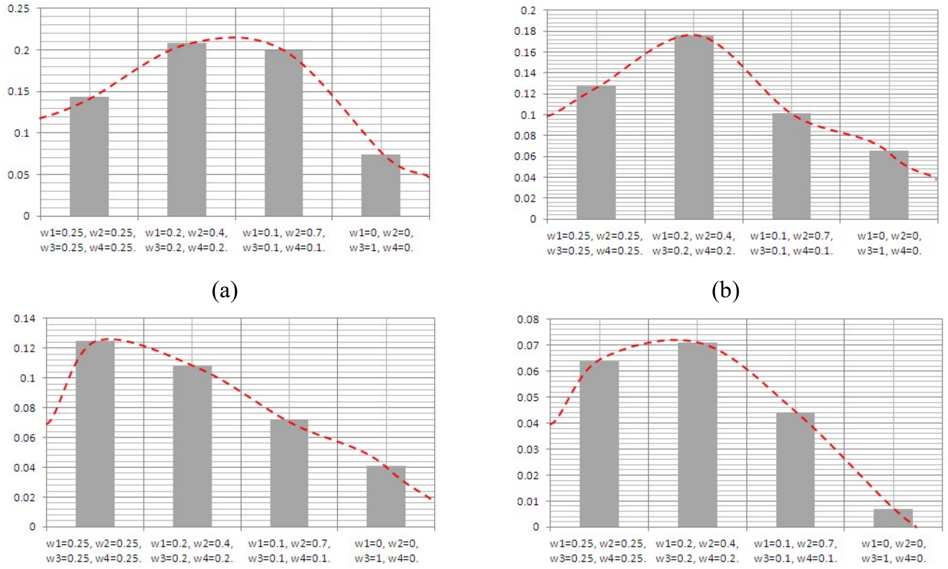


Fig. 4. Change in average fitness and the best average in 25 independent executions with population size 60



(a) (b)

(c) (d)

Fig. 5. The numerical results derived from execution of general genetic algorithm (a) up to ideal distance; (b) using MGG in generation change; (c) using theory of viruses and viral infection (d) of viruses and viral infection along with local search

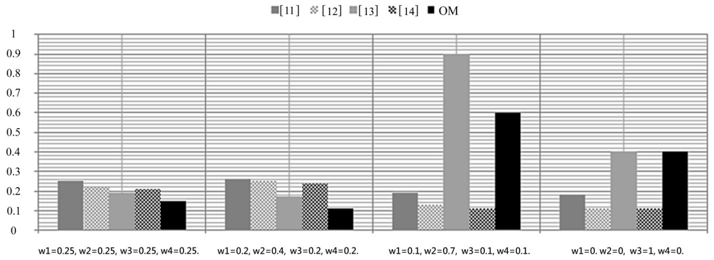


Fig. 6. Comparison between algorithms in distance from ideal point

5. Conclusion

The researchers proposed a comprehensive method to improve search rate in urban optimal path routing problems using viral infection and local search in urban. The proposed approach is performed based on two different schemes including wavelet transform and genetic algorithm. Wavelet transform as the first part of proposed algorithm derives edges risk; secondly, multistage genetic algorithm operates to find the optimal solution which is defined as the shortest path. The researchers presented and discussed the results from any improvement phase for this method from general genetic algorithm to the suggested method within four testing formats in order to prove efficiency in comprehensive suggested technique. Numerical results show that the proposed algorithms have a competitive performance in solving path routing problems.

Conflict of Interest

The author(s) declared no potential conflicts of interest with respect to the research, authorship and publication of this article.

Funding

The author(s) received no financial support for the research, authorship and publication of this article.

References

[1] McQuillan, J., Richer, I., & Rosen, E., The new routing algorithm for the ARPANET, *IEEE Transactions on Communications*, *28*(5), 1980, 711-719.
[2] Frank, H., Shortest paths in probabilistic graphs, *Operations Research*, *17*(4), 1969, 583-599.
[3] Beigy, H., & Meybodi, M. R., Utilizing distributed learning automata to solve stochastic shortest path problems, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *14*(05), 2006, 591-615.
[4] Misra, S., & Oommen, B. J., Dynamic algorithms for the shortest path routing problem: learning automata-based solutions, *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *35*(6), 2005, 1179-1192.
[5] Burton, D., On the inverse shortest path problem, *Département de Mathématique, Faculté des Sciences, Facultés Universitaires Notre-Dame de la Paix de Namur,* 1993.
[6] Waller, S. T., & Ziliaskopoulos, A. K., On the online shortest path problem with limited arc cost dependencies, *Networks: An International Journal*, *40*(4), 2002, 216-227.
[7] Pritchard, D., & Thurimella, R., Fast computation of small cuts via cycle space sampling, *ACM Transactions on Algorithms (TALG)*, *7*(4), 2011, 46.
[8] T. time-volume function Studies, Tehran Comprehensive Transportation and Traffic Studies (TCTTS), a division of the Municipality of Tehran, Tehran, 1995.
[9] Zitzler, E., *Evolutionary algorithms for multiobjective optimization: Methods and applications*, ed. 63, Ithaca: Shaker, 1999.
[10] Satoh, H., Minimal generation gap model for GAs considering both exploration and exploitation, In *Proc. of the 4th International Conference on Fuzzy Logic, Neural Nets and Soft Computing*, 1996, 494-497.
[11] Pourtakdoust, S. H., & Zandavi, S. M., A hybrid simplex non-dominated sorting genetic algorithm for multi-objective optimization, *International Journal of Swarm Intelligence & Evolutionary Computation*, *5*(3), 2016, 1-11.
[12] Gouveia, L., Simonetti, L., & Uchoa, E., Modeling hop-constrained and diameter-constrained minimum spanning tree problems as Steiner tree problems over layered graphs, *Mathematical Programming*, *128*(1-2), 2011, 123-148.
[13] Hao, H., & Barooah, P., Stability and robustness of large platoons of vehicles with double-integrator models and nearest neighbor interaction, *International Journal of Robust and Nonlinear Control*, *23*(18), 2013, 2097-2122.
[14] Alam, A., Fuel-efficient heavy-duty vehicle platooning, Doctoral dissertation, KTH Royal Institute of Technology, 2014.
[15] Lenzen, C., & Patt-Shamir, B., Fast routing table construction using small messages, In *Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing,* 2013, 381-390.