Determination of the shortest path in the university campus of Serres using the Dijkstra and Bellman-Ford algorithms

Dimitrios Botsis
Adjunct Assistant Professor, Department of Surveying and Geoinformatics Engineering, International Hellenic University, Campus of Serres, 621 24 Terma Magnisias, Serres

Eleftherios Panagiotopoulos
Lecturer, Department of Surveying and Geoinformatics Engineering, International Hellenic University, Campus of Serres, 621 24 Terma Magnisias, Serres

+302321049389, jimbotsis@gmail.com

ΠΕΡΙΛΗΨΗ: Στην παρούσα ερευνητική εργασία παρουσιάζεται η εφαρμογή των αλγόριθμων Dijkstra και Bellman-Ford στο χώρο των εγκαταστάσεων της πανεπιστημιούπολης Σερρών του Διεθνούς Πανεπιστημίου Ελλάδος, για την προσέγγιση των συντομότερων διαδρομών από επιλεγμένες θέσεις, σε διάφορα κτίρια. Οι εγκαταστάσεις του εκπαιδευτικού ιδρύματος αναπαραστάθηκαν με τη μορφή γράφου, του οποίου οι κορυφές είναι επιλεγμένες θέσεις και κτίρια, ενώ οι ακμές είναι οι διαδρομές, με τα βάρη τους να αντιπροσωπεύουν τα μήκη τους. Η λήψη αποφάσεων για τη δημιουργία των βέλτιστων διαδρομών και ο σχεδιασμός χαρτών με αποτυπωμένες τις προτεινόμενες διαδρομές, αποτελούν ένα πολύπλοκο πρόβλημα το οποίο μπορεί να επιλυθεί με την εφαρμογή αλγόριθμων συντομότερων μονοπατιών. Συμπερασματικά, η εφαρμογή των αλγόριθμων Dijkstra και Bellman-Ford έδειξε ότι είναι αποτελεσματικοί για την προσέγγιση των βέλτιστων – συντομότερων διαδρομών και μπορούν να εφαρμοστούν με επιτυχία στο σχεδιασμό πολυπλοκότερων διαδρομών. Τέλος η εφαρμογή δύο διαφορετικών αλγορίθμων που προσεγγίζουν το ίδιο πρόβλημα επιτρέπει την επιβεβαίωση της εγκυρότητας των αποτελεσμάτων τους.

Λέξεις κλειδιά: αλγόριθμος Dijkstra, αλγόριθμος Bellman-Ford, γράφοι, συντομότερη διαδρομή, χάρτες διαδρομών.

ABSTRACT: This research paper presents the application of the Dijkstra and Bellman-Ford algorithms at the premises of the campus of the International Hellenic University in Serres, for the approximation of the shortest routes from selected locations to different buildings. The facilities of the educational institution were depicted in the form of a graph, whose nodes are the selected locations and buildings, while the edges are the paths with their weights representing their lengths. Making decisions to create optimal routes and designing maps with the proposed routes is a complex problem that can be solved by applying a shorter path algorithm. In conclusion, the specific application of the Dijkstra and Bellman-Ford algorithms has proven to be effective in approaching optimal - shortest routes and can be successfully implemented to even more sophisticated routes. Finally, the application of two different algorithms approaching the same problem enables the validity of their results.

Keywords: Dijkstra algorithm, Bellman-Ford algorithm, graphs, shorter path, route maps.

I. INTRODUCTION

The problem of identifying the shortest paths is one of the most researched combined optimization problems [1] with implementation in many different fields, such as business plans, social networking tools, computer science transport networks, the generation of maps and applications with paths, etc. The algorithms of finding the shortest paths were initially developed to improve computer networks, due to the fact that the concept of computer networking is based on a large extent in the graph theory [2].

Ahuja et al., in 1990 [3], in an extensive research paper, presented several algorithms for identifying the shortest paths, which were based on the Dijkstra algorithm using different data structures. The Dijkstra algorithm has many variations, however, it's principal structure permits the calculation of the shortest path between the selected nodes, in a guided graph or not, with negatively weighted edges [4,5,6]. There are cases where the Dijkstra algorithm is used to configure more complex models to calculate the shortest paths [7].

The Dijkstra algorithm was proposed in 1959 by the Dutch computer scientist Edsger Wybe Dijkstra, from which it took his name [8,9] and together with Bellman-Ford, they constitute the most generally accepted as well as widespread algorithms for discovering the shortest paths [10,11]. Accordingly, the Bellman-Ford algorithm was named after Richard Bellman and Lester Ford Jr. and calculates the shortest paths from a source node to the other nodes of a graph, at a slower time than the equivalent Dijkstra algorithm [5,12]. The main advantage of the Bellman-Ford algorithm is its ability to handle graphs with negative edge weights. In 2012, Meghanathan [13] reviewed the Dijkstra and Bellman-Ford algorithms to approximate the shortest graph paths and concluded that the time complexity for the Dijkstra algorithm is classified as O(|E|*log|V|), while corresponding to the Bellman-Ford algorithm is O (| E | | V |). Where V is related to the vertices or nodes of the graph and E corresponds to the edges.

There are many applications of the Dijkstra algorithm, such as optimizing the routes of an urban rail network, in order to present the framework of the development of future smart cities based on the shortest route [14], designing the evacuation of buildings during critical and dangerous incidents, identifying the shortest route that someone could follow in order to leave the building in the shortest possible time [11]. More specifically Dijkstra's algorithm has been applied in many cases to calculate the shortest paths in a building faster evacuation [15] - [16] -

[17]. Li et al. in 2016 [18] used the Dijkstra algorithm to simulate the optimal aisle network for classrooms evacuation in a primary school campus and Choi and Chi, in 2019 [19] create a model with the same method which locate the fastest, safest and finally optimal paths for a building evacuation. Finally Dijkstra algorithm is a useful technique when looking for the faster path between a single source and a single destination node [20,21].

A special and remarkable application of the Dijkstra and Bellman-Ford algorithms is the implementation of printed or digital navigation maps as well as navigation apps in the university communities. All university members have to move from one place to another for various reasons, such as lectures' attendance, meetings' participation, administrative affairs conduction, parking, books' lending and/or study in the library of the campus, events attendance in the amphitheater, dining, as well as rest / relaxation in green areas and/or in the canteen. Thus, for the effortless and punctual transfer, a proper choice of routes must be accomplished, especially in the case of large university communities. Selecting the optimal routes is a problem of identifying the shortest paths, where the cost may be the length of the route or the time needed to access it.

Gao et al., in 2014 [22], used the Dijkstra and Bellman-Ford algorithms to track the shortest paths between Carnegie Mellon University buildings in Silicon Valley, USA. Agarana et al., in 2016 [23], expounded the implementation of the Dijkstra algorithm for finding the shortest paths in a tram transportation network between several locations on an extended hypothetical campus. They used traveling time as the cost, knowing the distances between the various nodes and assuming that the tram will move at an average speed of 50 km/h. Lateef et al., in 2017 [24], suggested an algorithm for the development of a navigation application in the buildings and premises of a Nigerian university, geared to provide assistance to a new visitor at the university. The proposed algorithm that was used to identify the shortest routes from the university's principal administration building to all other buildings, was the Dijkstra.

Sadavare and Kulkarni [25], in 2012, applied the algorithms of Dijkstra as well as Floyd-Warshall and Bellman-Ford in order to simulate shorter paths, aiming to infrastructures development such as cable television, electricity and telephone networks or water pipelines. In their research, they shaped a graph where the vertices represented specific locations in the city and the edges imprinted the roads or routes between the vertices. The cost of construction or the length of ducts or cables was established as the weight of the edges.

If there was the possibility of transportation without any obstacle, then obviously a straight line would be the shortest route between two nodes. However, in the real world, as on campuses, there are buildings, green areas, obstacles, etc. that make it impossible to move always straight ahead not to mention that it is more difficult to find the shortest route, especially when the environment is unknown. The purpose of the particular research is to present an optimization model, that deals with the selection of the optimum (shorter) routes on the campus of the International Hellenic University in Serres.

II. METHODOLOGY

A. Research area

The scope of the research is the campus of the International Hellenic University, located in Serres, specifically at the southern edge of the city. The given campus occupies an area of approximately 196.5 hectares, of which about 144.4 stremmas with establishments are available for use [26]. More specifically, the implementation of the Dijkstra and Bellman-Ford algorithms for identifying the shortest paths was conducted in the aforementioned area, where facilities and configured paths already exist, that is the 144.4 stremmas. Fig. 1 presents a topographic diagram of the campus of the International Hellenic University in Serres, appearing the northern area, in which the facilities have been already developed, as well as the southern region, which is available for future exploitation.

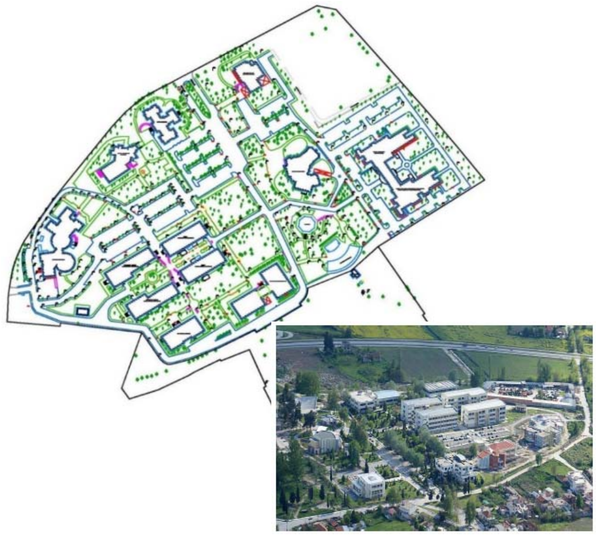


Figure 1 – The campus of the International Hellenic University in Serres

B. The implementation of the Dijkstra and Bellman-Ford algorithms

B1. The structure of the graph

A graph G consists of two sets, specifically the set of nodes or vertices V(G) and the sum of lines E(G) and is represented as follows:

$$G = (V, E) \quad (1)$$

Each edge of the graph connects two nodes and is represented as a pair $(x_i, x_j) \in E(G)$ [27]. A sequence of successive nodes, which are connected to each other via edges, is searched in order to find the shortest paths. The node that constitutes the starting point for a path in the graph, is called the source node (V_s) and the endpoint of the path is called the destination node (V_d). The route or the path (P) is the sequence of the nodes connecting the source node to the destination node:

$$P = V_1 \rightarrow V_2 \rightarrow \cdots \rightarrow V_n \quad (2)$$

The weight of the edges (w) is also called cost, while it receives exclusively non-negative values. It must be noted that the Bellman-Ford algorithm has the ability to handle graphs with negative edge weights. However, there is no physical meaning on the representation of the paths' length with negative values, therefore, the present research does not use the specific feature of the Bellman-Ford algorithm. Finally, the distance (D) is the sum of the weights of the edges for each possible path connecting the source node to the destination node and is mathematically expressed by the Eq. 3.

$$D = w(P) = \sum_{i=1}^{n-1} w(V_i, V_{i+1}) \quad (3)$$

After checking the sums of the edge weights from all the possible paths between the source node and the destination node, then the smallest sum, which is called the minimum cost, is chosen:

$$D_{min}(V_s \rightarrow V_d) = min\,\{w(P)\} \quad (4)$$

The Dijkstra and Bellman-Ford algorithms calculate the shortest paths in a directed or non-directional graph with non-negative edges.

B2. The configuration of the graph

The used methodology for the configuration of the graph includes the determination of 22 nodes or vertices corresponding to 1 central entrance gate, 15 building establishments and finally 6 central points, which were selected having as the criterion the spatial planning of the campus facilities. Specifically, it was estimated that there are 6 points of more frequent passage on the possible paths on the campus premises, based on the locations of the buildings on the area. Table 1 analyzes the nodes of the graph with the necessary explanation for each one.

Table 1 – The nodes of the configured campus graph

Nodes	Interrelation – Illustration of the nodes
1	Central entrance gate
2	Management building
3	Library
4	Multi-purpose building
5	Central point between the Administration building, the Library and the Conference center
6	Central point between the Library and the Conference center
7	Student club
8	Central point between the Student club, the multi-purpose building and the building of Economics
9	Conference center
10	Department of Computer Engineering
11	Laboratory of Management and Economics
12	School of Management and Economics
13	Central point between the Conference center and the Canteen
14	Central point between the Conference center, the Canteen and the Topography building
15	School of Technology Applications
16	Central point between the school of Economics and Management
17	School of technological Appliance
18	Canteed
19	Laboratories of experiments
20	Laboratory of Engineering
21	Outdoor amphitheater
22	Department of Topography and Geoinformatics

For the calculation of the weights, the distances that a visitor spends in order to travel from one building to another was measured. These distances are not the lines that link the nodes of the graph, but the paths in the campus area. Essentially, it has been assumed that the transition from one building to another is followed by defined (logical) routes (walkways and streets) rather than accessing green areas or blocks. The total paths representing the graph edges were measured in the design environment and are listed in Table 2. It must be noted that the graph is non-directional, and consequently, the movement between the nodes can be done and vice versa, that is 1 → 2 or equivalent 2 → 1.

Table 2 – Edges weights (distances between nodes)

Edges	Weights	Edges	Weights
1—2	85	9—21	144
1—3	130	10—14	78
1—5	101	10—22	78
1—6	180	11—12	80
2—3	167	11—15	126
2—4	82	11—16	64
2—5	111	12—13	124
2—8	185	12—16	54
2—12	192	12—17	114
2—13	212	12—20	146
3—5	119	12—21	244
3—6	135	13—14	129
3—9	160	13—16	137
4—5	146	13—17	123
4—7	135	13—18	53
4—8	127	13—19	239
4—13	240	13—20	139
5—6	87	13—21	127
5—8	167	14—18	232
5—9	101	14—21	109
5—12	173	14—22	78
5—13	100	15—16	64

Edges	Weights	Edges	Weights
5—17	317	15—17	75
6—9	111	15—19	123
6—10	148	15—20	177
6—14	134	16—17	54
7—8	101	16—19	104
7—11	128	16—20	162
8—9	223	17—19	120
8—11	86	17—20	110
8—12	87	17—21	203
8—13	204	18—19	244
8—16	71	18—20	144
9—10	209	18—21	85
9—13	87	19—20	150
9—14	130	20—21	154
9—18	76	21—22	142

Fig. 2 shows the shape of the graph in the background of the topographic diagram of the campus facilities and in the next Fig. 3, the graph with the weights of the edges is presented.

B3. The development of the Dijkstra and Bellman-Ford algorithms

The development and implementation of the Dijkstra and Bellman-Ford [28] algorithms were conducted in the MATLAB programming language environment. The data in both algorithms is input in the form of a table:

$$P_{ij} = \begin{bmatrix} P_{1-1} & P_{1-2} & \dots & P_{1-21} & P_{1-22} \\ P_{2-1} & P_{2-2} & \dots & P_{2-21} & P_{2-22} \\ \dots & \dots & \dots & \dots & \dots \\ P_{21-1} & P_{21-2} & \dots & P_{21-21} & P_{21-22} \\ P_{22-1} & P_{22-2} & \dots & P_{22-21} & P_{22-22} \end{bmatrix} \quad (5)$$

The values of the table are the distances between the nodes (ie the weights of the edges) as measured and presented in Table 2, while when two nodes are not interconnected, the value 0 is entered in the corresponding field. The Dijkstra and Bellman-Ford algorithms are implemented according to relations 6 and 7:

$$[cost \quad rute] = dijkstra(G, V_s, V_d) \quad (6)$$

$$[cost \quad rute] = Bellman - Ford(G, V_s, V_d) \quad (7)$$

Where: cost: is the minimum cost, ie, in this case, the total length of the shortest route, rute: is the sequence of the nodes defining the resulting path, Vs: source node (starting point) and Vd: destination node (endpoint).

The models of the Dijkstra and Bellman-Ford algorithms can be applied to all possible paths derived from the graph formed for the campus of the International Hellenic University in Serres. In this case, different paths were chosen to simulate the two models, while the choice was focused on the more complex ones, thus, demonstrating the effectiveness of the Dijkstra and Bellman-Ford algorithms.

III. RESULTS

Table 3 lists the results of all Dijkstra and Bellman-Ford algorithms applications for finding the shortest paths on the campus of the International Hellenic University in Serres. It must be noted that each calculated route is valid also vice versa, with the same cost and the same (yet inverse) node sequence.

Table 3 – Results of the Dijkstra and Bellman-Ford algorithms

α/α	Source node	Destination node	Min cost	Route
	Vs	Vd	cost	rute
1	1	7	302	1-2-4-7
2	1	17	324	1-5-13-17
3	1	15	392	1-5-12-16-15
4	1	19	432	1-5-12-16-19
5	1	21	328	1-5-13-21
6	1	9	202	1-5-9
7	1	22	392	1-6-14-22
8	2	15	310	2-12-16-15
9	2	10	346	2-5-6-10
10	2	9	212	2-5-9
11	3	7	384	3-2-4-7
12	3	22	347	3-6-14-22
13	3	19	450	3-5-12-16-19
14	4	11	213	4-8-11
15	4	18	293	4-13-18
16	4	10	381	4-5-6-10
17	7	9	324	7-8-9
18	7	20	334	7-8-16-20
19	7	10	503	7-8-5-6-10
20	9	11	288	9-13-16-11
21	9	19	326	9-13-19
22	10	3	283	10-6-3
23	10	17	330	10-14-13-17
24	11	3	372	11-12-5-3
25	11	18	254	11-16-13-18
26	12	3	292	12-5-3
27	15	4	262	15-16-8-4
28	15	10	405	15-17-13-14-10
29	16	3	346	16-12-5-3
30	16	7	172	16-8-7
31	16	10	344	16-13-14-10
32	17	2	300	17-16-12-2
33	17	3	342	17-13-5-3
34	18	7	358	18-13-8-7
35	19	2	350	19-16-12-2
36	20	4	360	20-12-8-4
37	21	2	338	21-13-5-2
38	22	4	445	22-14-6-5-4
39	22	12	331	22-14-13-12

Fig. 4, 5, 6, 7, 8 and 9 present the results of the Dijkstra and Bellman-Ford algorithms for determining the shortest paths from campus gate (source node 1) to various destination nodes (individual buildings 20, 15, 18, 22, 11 and 19).

Figure 2 – The graph on the background of the topographic diagram of the facilities of the International Hellenic University in Serres

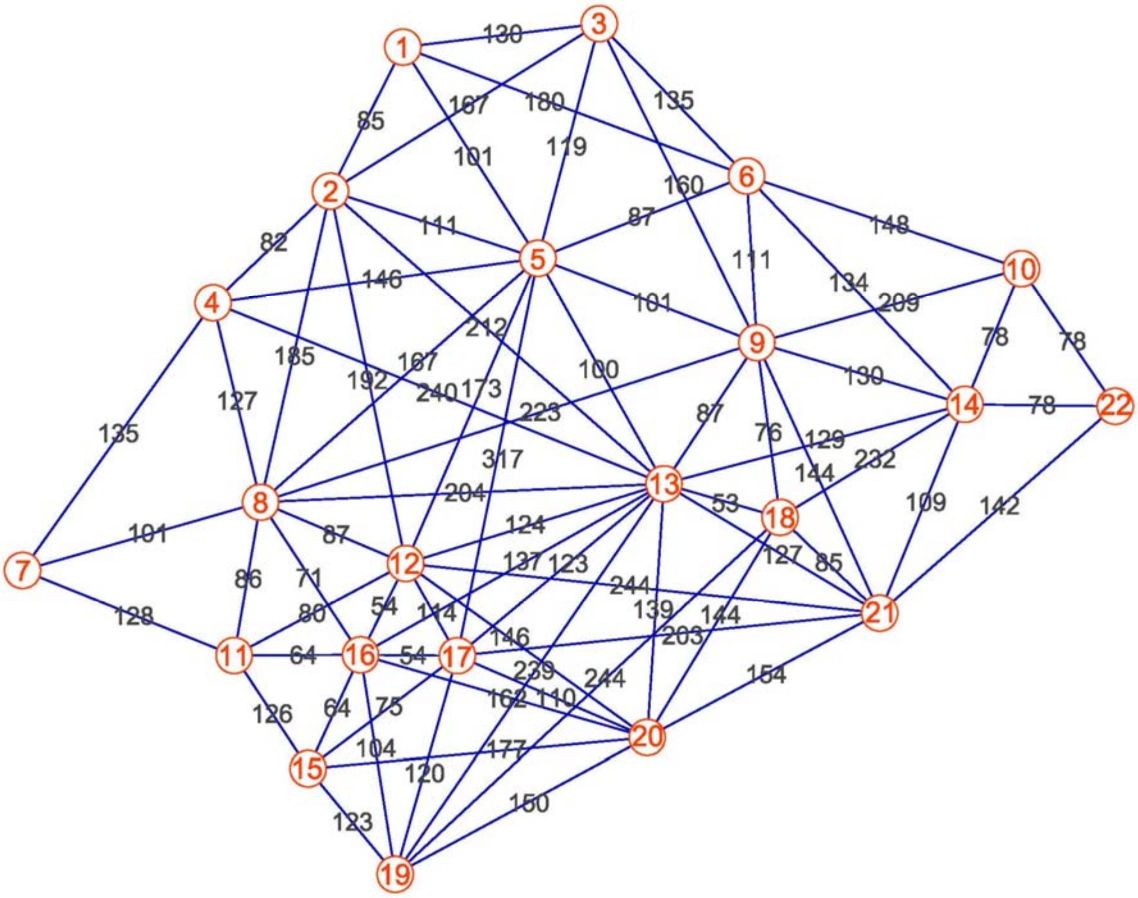
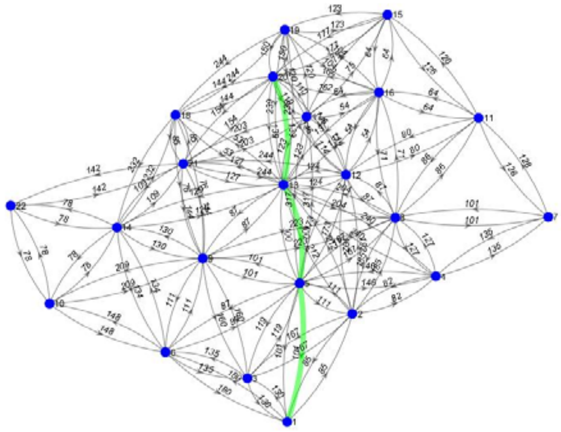
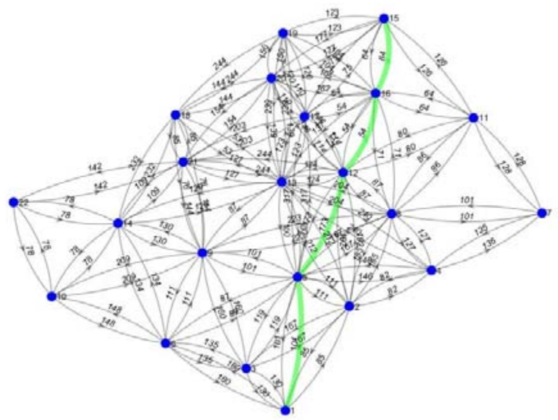


Figure 3 – The Graph for the premises of the International Hellenic University in Serres



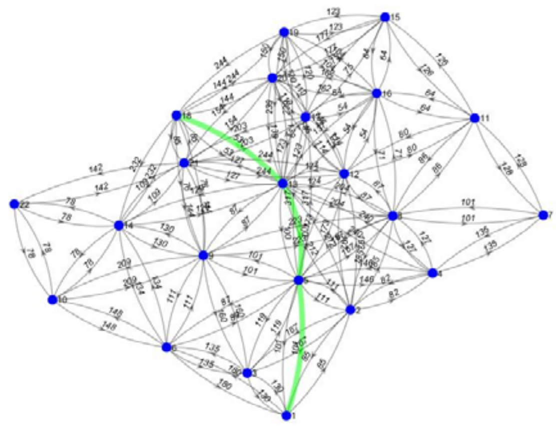
Route 1-5-13-20

Figure 4 – Outcome from the algorithms implementation - path from the gateway (node 1) to destination 20



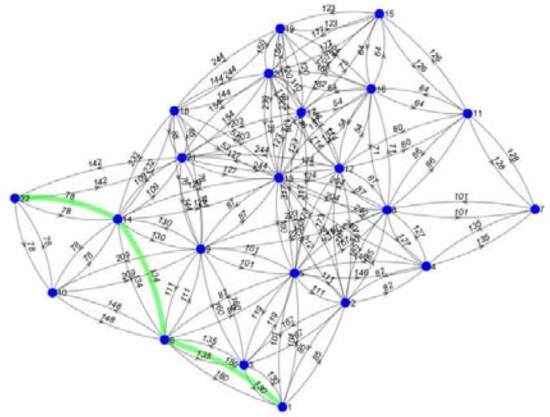
Route 1-5-12-16-15

Figure 5 – Outcome from the algorithms implementation - path from the gateway (node 1) to destinations 15



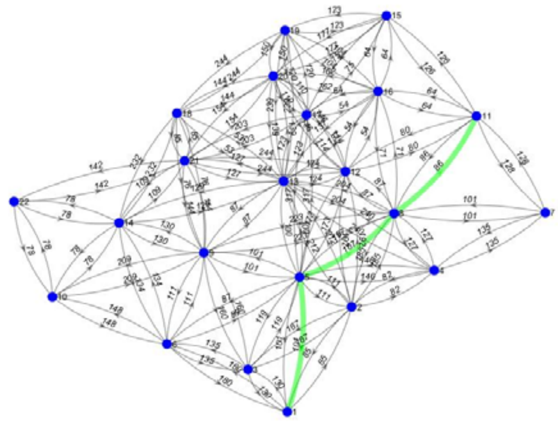
Route 1-5-13-18

Figure 6 – Outcome from the algorithms implementation - path from the gateway (node 1) to destinations 18



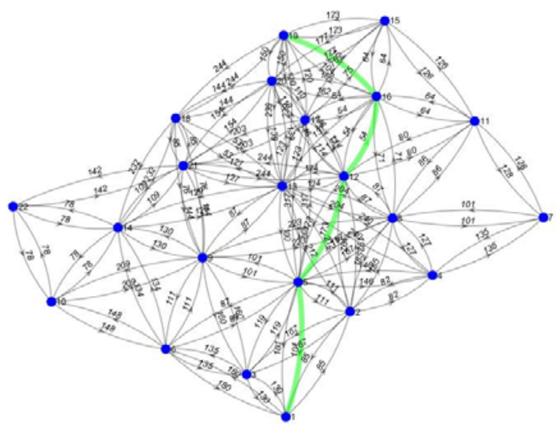
Route 1-6-14-22

Figure 7 – Outcome from the algorithms implementation - path from the gateway (node 1) to destinations 22



Route 1-5-8-11

Figure 8 – Outcome from the algorithms implementation - path from the gateway (node 1) to destinations 11



Route 1-5-12-16-19

Figure 9 – Outcome from the algorithms implementation - path from the gateway (node 1) to destinations 19

Fig. 10 and 11 illustrate some of the shortest paths, resulting from the implementation of the Dijkstra and Bellman-Ford algorithms to the campus topographic background.

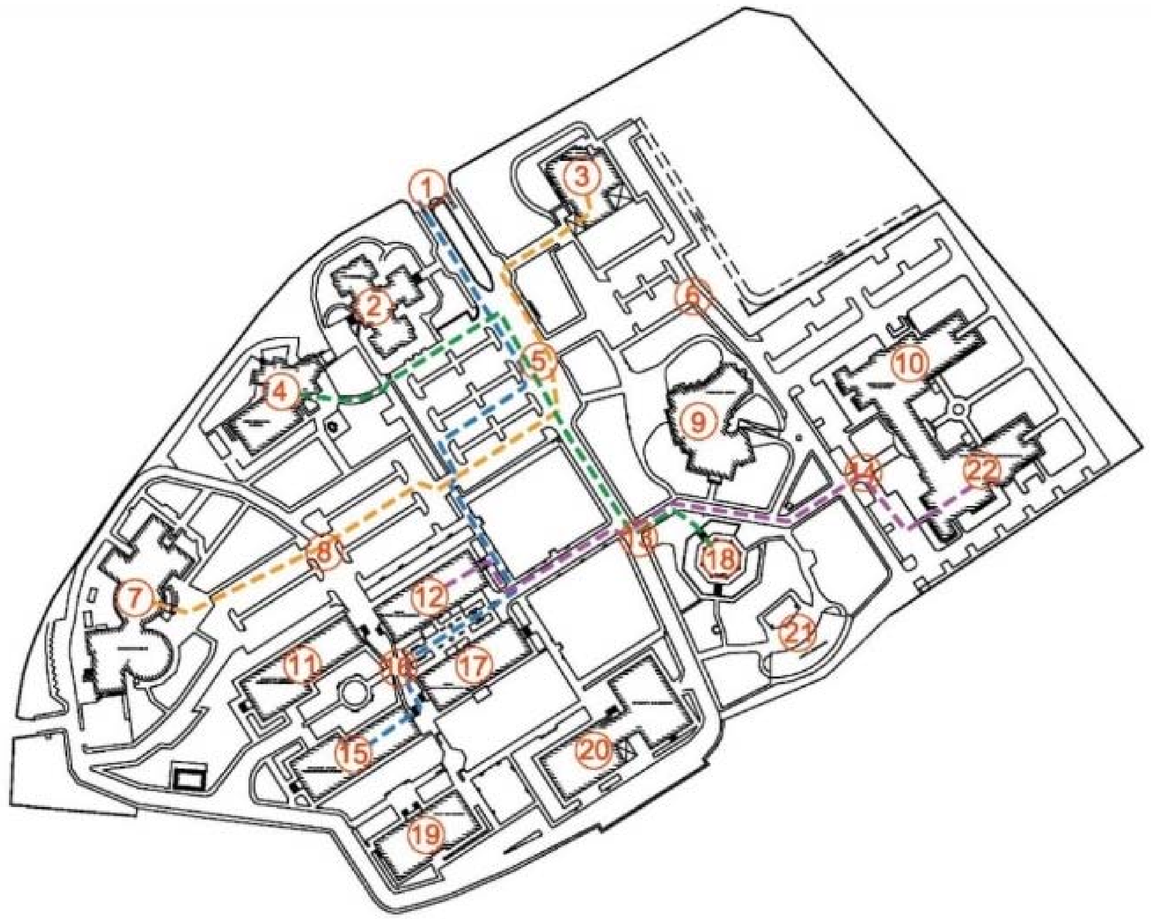


Figure 10 – Routes 1-15: blue, 7-3: orange, 4-18: green and 22-12 purple

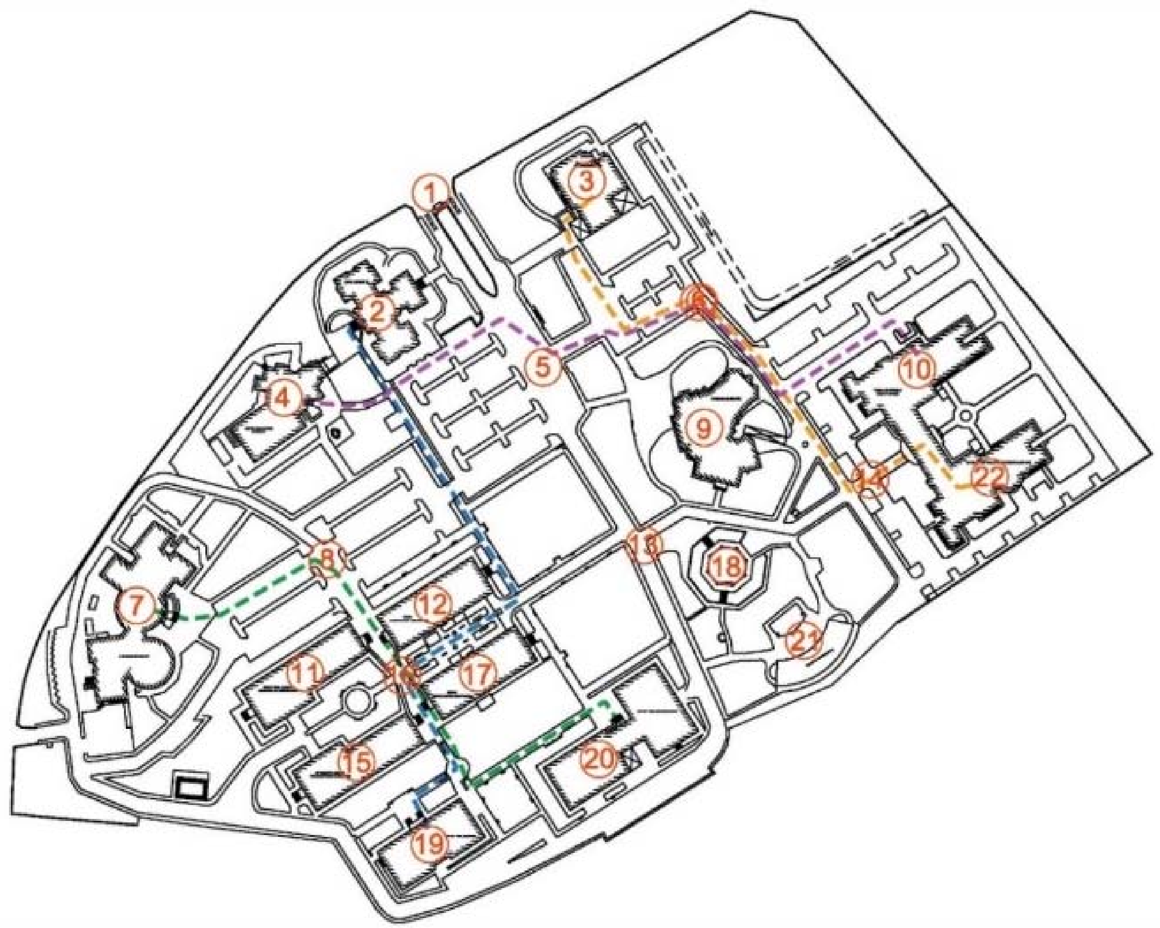


Figure 11 – Routes 19-2: blue, 3-22: orange, 7-20: green and 4-10 purple

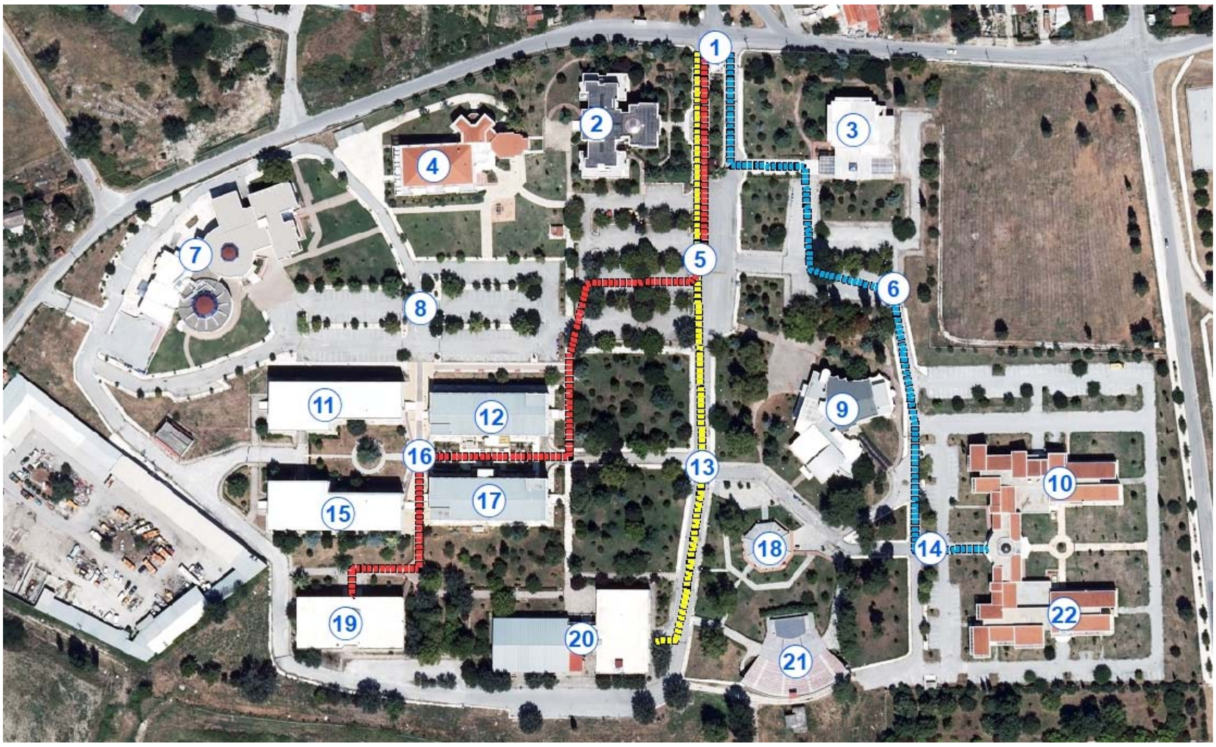


Figure 12 – Campus map with routes from the gate to specific buildings: 1 (Central entrance gate) - 20 (Laboratory of Engineering): yellow, 1 (Central entrance gate) - 19 (Laboratories of experiments): red and 1 (Central entrance gate) - 22 (Department of Topography and Geoinformatics): blue

Figure 13 – Campus map with accessible routes: 7 (Student club) - 20 (Laboratory of Engineering): yellow, 3 (Library) - 17 (School of technological Appliance): red and 4 (Multi-purpose building) - 10 (Department of Computer Engineering): blue

The data in Table 3 exhibit that from the simulated routes the largest one is between nodes 7 and 10 with a length of 503 m and a node sequence of 7-8-5-6-10. The smallest route is between nodes 16 and 7 with a length (cost) of 172 m and a sequence of nodes 16-8-7, while the average length of the routes amounts to 332.5 m.

The mapping of certain paths in the topographic diagram of campus facilities (Fig. 10 and 11) shows the convenience of graphically illustrating the results.

Particular mention should be given to the routes identified from the main entrance gate of the campus to various buildings, since the entry into the university area is the most important starting point for the search of various points, services, buildings, etc. Thus, the larger routes simulated by the main entrance of the academic institution are: (1) 1-19, with a length of 432 m and a sequence of nodes 1-5-12-16-19, 2) 1-15, with a length of 392 m and a sequence of nodes 1-5- 12-16-15 and 3) 1-22, with a length of 392 m and a node sequence of 1-6-14-22.

Obviously, the routes between neighboring nodes were not simulated as the distances between them are small, while the node sequence will consist exclusively of the source node and the destination node. Additionally, such a calculation is not relevant for the approximation of the shortest paths at the premises of the campus of the International Hellenic University in Serres.

IV. CONCLUSIONS

Identifying the shortest routes to specific destinations is an issue that every visitor can encounter, especially in a new and unknown area. This issue appears to be greater when the search for shorter paths is related to professional activities, such as for members of a university community. There are cases where the areas occupied by university facilities are quite extensive with many buildings, thus the existence of appropriate maps, instructions or applications to find the shortest path is totally necessary.

This research paper demonstrates that the campus facilities (i.e. the International Hellenic University in Serres) can be configured as a graph, to which the Dijkstra and Bellman-Ford algorithms are applied and the requested shortest paths, with accuracy and speed, are extracted every time. This dissertation, as shown in Fig. 10 and 11, could be the basis for the creation of printed or digital maps with instructions for finding the shortest paths in the university area. For example, Fig. 12 and 13 show maps of the campus with the shortest routes between specific destinations. Furthermore, the specific study could be leveraged for the development of an application with these capabilities, as in the case of Gao et al., 2014 and Lateef et al., 2017.

The main advantage of this research is the fact that two simple algorithms such as the Dijkstra and Bellman-Ford can be applied, with some assumptions or admissions, for estimating the shortest paths on campus. Simultaneously, the simulations of the Dijkstra and Bellman-Ford algorithms, through the computing environment of a mathematical programming language, provide great speed for the calculations of the shortest paths, from each point in the university area. Even though the Dijkstra algorithm is mentioned in the literature as faster than the Bellman-Ford algorithm, in the particular case, no differences arose concerning the computation speed as well as the results output. Possibly, time differences are more distinct in applications of algorithms with more complex graphs.

Another capability that emerges from the specific research is the creation of new proposed routes, with small interventions in the infrastructure of the green areas or small tiers, in order to optimize some complex paths. In particular, if the existing campus infrastructure appears a rather complex route, requiring unnecessary traffic around non-accessible areas, it is possible with a small intervention to become shorter.

In general, the successful application of the Dijkstra and Bellman-Ford algorithms to identify the shortest paths in campus outdoors generates the prerequisites for expanding also to the interiors of buildings with instructions for the shortest paths between halls, offices, workshops, administrative departments, etc. Eventually, it must be highlighted that both algorithms in each simulation of the desired paths yielded the same results, which substantiates their proper development and implementation.

V. REFERENCES

[1] ''A faster algorithm for the single source shortest path problem with few distinct positive lengths'', J.B. Orlin, K. Madduri, K. Subramani, M. Williamson, Journal of Discrete Algorithms, Vol. 8, Nr. 2, 2010, pp. 189–198. https://doi.org/10.1016/j.jda.2009.03.001.

[2] ''Routing Choice Modelling using Fuzzy Logic and Adaptive Neuro-Fuzzy'', K.D. Subodh, M. Deval, S.A. Shriniwa, P.S. Ajit, K.S. Ashoke, Modern Traffic and Transportation, Engineering Research, Vol. 2, Nr. 4, 2013, pp. 91–99.

[3] ''Faster algorithms for the shortest path problem'', K.R. Ahuja, K. Mehlhorn, B.J. Orlin, E.R. Tarjan, Journal of the ACM, Vol. 37, Nr. 2, 1990, pp. 213–223. https://doi.org/10.1145/77600.77615.

[4] ''Data Structures and Algorithms: Concepts, Techniques and Applications'', G.A.V. Pai, Tata McGraw-Hill Education Private Limited, New Delhi, India, 2010.

[5] ''On the History of the Shortest Path Problem'', A. Schrijver, Documenta Mathematica, 2010, pp. 172.

[6] ''Calibration and validation of a simulation-based dynamic traffic assignment model for a large-scale congested network'', S. Shafiei, Z. Gu, M. Saberi, Simulation Modelling Practice and Theory, Vol. 86, 2018, pp. 169-186. https://doi.org/10.1016/j.simpat.2018.04.006.

[7] ''Energy and mobility conscious multipath routing scheme for route stability and load balancing in MANETs'', W.A. Jabbar, M. Ismail, R. Nordin, Simulation Modelling Practice and Theory, Vol. 77, 2017, pp. 245-271. http://dx.doi.org/10.1016/j.simpat.2017.07.001.

[8] ''A Note of Two Problems in Connexion with Graph'', E.W. Dijkstra, Numerische Mathematik, Vol. 1, 1959, pp. 23-33. https://doi.org/10.1007/BF01386390.

[9] ''An Interview with Edsger W. Dijkstra'', P. Frana, Communication of the ACM, Vol. 53, Nr. 8, 2010, pp. 41-47. https://doi.org/10.1145/1787234.1787249.

[10] ''Route Planning in Vanet By Comparative Study of Algorithms'', J. Shivani, J. Singh, International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 3, Nr. 7, 2013, pp. 682-689.

[11] ''The Utilization of Dijkstra's Algorithm to Assist Evacuation Route in Higher and Close Building'', N.A.M. Sabri, A.S.H. Basari, B. Husin, K.A.F.A. Samah, Journal of Computer Science, Vol. 11, Nr. 2, 2015, pp. 330–336. https://doi.org/10.3844/jcssp.2015.330.336.

[12] ''An Investigation of Dijkstra and Floyd Algorithms in National City Traffic Advisory Procedures'', A.K. Sangaiah, M. Han, S. Zhang, International Journal of Computer Science and Mobile Computing, Vol. 3, Nr. 2, 2014, pp. 124-138.

[13] ''Review of Graph Theory Algorithms'', N. Meghanathan, Department of Computer Science, Jackson State University, 2012.

[14] ''Railway Route Optimization System using Dijkstra's Method'', P. Pramod, D. Sunanda, International Journal on Recent and Innovation Trends in Computing and Communication, Vol. 2, Nr. 3, 2014, pp. 3435-3440.

[15] ''Using modified Dijkstra' s algorithm for critical path method in a project network'', N.R. Shankar, V. Sireesha, Int. J. Comput. Appl. Math., Vol. 5, Nr. 2, 2010, pp. 217–225.

[16] ''Comparison of Dijkstra' s shortest path algorithm with genetic algorithm for static and dynamic routing network'', Y. Sharma, S.C. Saini, M. Bhandhari, Int. J. Electron. Comput. Sci. Eng., Vol. 1, Nr. 2, 2012, pp. 416–425.

[17] Xu Y., Wang Z., Zheng Q., Han, Z., ''The application of Dijkstra's algorithm in the intelligent fire evacuation system'', in 4th

International Conference on Intelligent Human-Machine Systems and Cybernetics 2012, 1, pp. 3–6. doi:10.1109/IHMSC. 2012.7.

[18] ''The Trace Model: A model for simulation of the tracing process during evacuations in complex route environments'', W. Li, Y. Li, P. Yu, J. Gong, S. Shen, Simulation Modelling Practice and Theory, Vol. 60, 2016, pp. 108-121. https://doi.org/10.1016/j. simpat.2015.09.011.

[19] ''Optimal route selection model for fire evacuations based on hazard prediction data'', M. Choi, S. Chi, Simulation Modelling Practice and Theory, Vol. 94, 2019, pp. 321-333. https://doi.org /10.1016/j. simpat.2019.04.002.

[20] ''Modification of Dijkstra's algorithm for safest and shortest path during emergency evacuation'', K.A.F.A. Samah, B. Hussin, A.S.H. Basari, Applied Mathematical Sciences, Vol. 9, Nr. 31, 2015, pp. 1531–1541. http://dx.doi.org/10.12988/ams.2015. 49681.

[21] ''The application of the shortest path algorithm in the evacuation system'', T.Y. Wang, R. Huang, L. Li, W.G. Xu, J.G. Nie, In Information Technology, Computer Engineering and Management Sciences (ICM) 2011, 2 IEEE, 2011, pp. 250–253. doi: 10.1109/ICM.2011.119.

[22] ''Optimizing Paths Taken Across Campus With Respect to Distance and Weather'', H. Gao, D. Wu, B. Zhang, 21-393 Final Project, 2014.

[23] ''Application of Dijkstra algorithm to proposed tramway of a potential world class university'', M.C. Agarana, N.C. Omoregbe, M.O. Ogunpeju, Applied Mathematics, Vol. 7, 2016, pp. 496–503. https://doi.org/10.4236/am.2016.76045.

[24] Lateef U.O., Adedeji A., Kazeem R., Idowu P.A., ''Determination of the shortest path of a Nigerian university map using Dijkstra's algorithm'', 11th International Multi-Conference on ICT Applications, 2017, AICTRRA.

[25] ''A Review of Application of Graph Theory for Network'', A.B. Sadavare, R.V. Kulkarni, International Journal of Computer Science and Information Technologies, Vol. 3, Nr. 6, 2012, pp. 5296-5300.

[26] Kariotis G., Theodoridou L., Kariotou G., Panagiotopoulos E., ''From suspicion to coexistence - The Municipality of Serres and the T.E.I. campus Serres (1979-2009)'', Proceedings of the 1st Panhellenic Conference with international participation and theme: Local Communities and Higher Education Institutions: Coexistence for Sustainable Development, Rhodes, 2010 April 24-25.

[27] ''Machine Learning: A Bayesian and Optimization Perspective'', S. Theodoridis, Academic Press, Elsevier, 2015.

[28] ''Bellman Ford Algorithm, MATLAB Central File Exchange'', A. Rath, 2020, https://www.mathworks.com/matlabcentral/fileexchange/63069-bellman-ford-algorithm.

AUTHOR'S DATA

Dimitrios Botsis
Department of Surveying and Geoinformatics Engineering
International Hellenic University
Campus of Serres, 621 24 Terma Magnisias, Serres
e-mail: jimbotsis@gmail.com

Χωρογραφίες/ Τεύχος 1/ Αρ 1/ 2020 – σελ 33-42