**Chapter 2 Literature – Technology Review**

Traffic forecasting for road networks is a critical component of urban planning, transportation management, and intelligent transport systems. Accurate traffic forecasts facilitate optimized route planning, reduced travel times, and enhanced safety. This literature review explores the foundational algorithms and recent advancements relevant to traffic forecasting, emphasizing the application of Dijkstra’s algorithm within the context of road network navigation and optimization.

## 2.1. Literature Review

**A Weighted Undirected Graph**

A graph G = (V, E) consists of:

- A set of vertices

- A set of edges

Each edge e connects a pair of vertices {u,v}, where u and v may be the same or different. The vertices u and v are called the endpoints of edge e. If u and v are the same, e is called a self loop. Edges connecting the same pair of vertices are called parallel.

In a finite graph, both V and E are finite sets. In an undirected graph, the order of the vertices in an edge does not matter.

**Paths**

A path is a sequence of edges such that:

1. Each pair of consecutive edges and share a common endpoint, for 0 ≤ i <n.

2. If is not a self loop, it shares one endpoint with and another endpoint with (except for the first edge and the last edge .

**Weighted Graphs**

In a weighted graph, each edge e is assigned a length l(e) . The length of a path P= is the sum of the lengths of its edges:

**Additional Definitions**

- A path is simple if no vertex appears more than once.

- A graph is connected if, for every pair of vertices u and v in V, there exists a path that starts at u and ends at v.

- A circuit is a path where the start and end vertices are the same.

- A connected graph which has no circuits is called a tree.

For an example of a weighted graph, consider Figure 2.1, which includes:

- The set of vertices: {A,B,C,D,E,F}

- The set of edges: {}

In this graph:

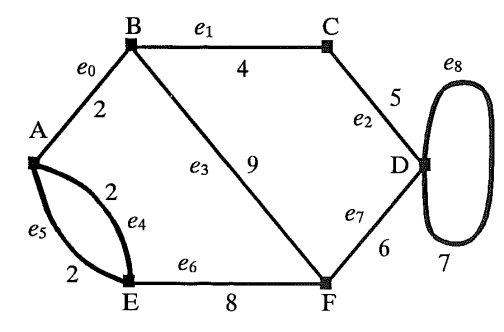
- Edges and are parallel.

- Edge is a self loop.

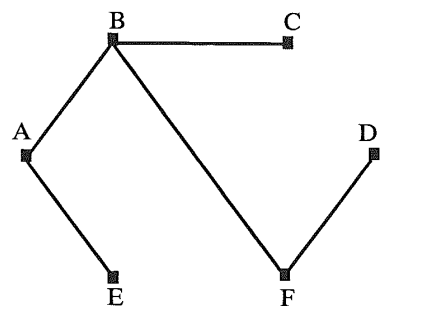
The length of the simple path is 2 + 9 + 6 = 17 .

The path forms a simple circuit.

Additionally, the graph depicted in Figure 2.2 is a tree.



**Fig. 2.1. A weighted non-directed graph**



**Fig. 2.2. A tree**

**A Weighted Directed Graph**

In a directed graph, the endpoints of an edge have a specific order. The first endpoint is called the start vertex and the second endpoint is called the end vertex. An edge is considered directed from its start vertex to its end vertex.

- Parallel edges: Edges with the same start and end vertices.

- Antiparallel edges: If u v and is directed from u to v while is directed from v to u, then and are antiparallel.

A directed path is a sequence of edges ,,...., such that the end vertex of is the start vertex of for 0 ≤ i < n. The length of a directed path and a simple directed path are defined similarly to those in an undirected case.

A directed graph (V, E) is called strongly connected if for every pair of vertices u and v in V, there exists a directed path from u to v and a directed path from v to u. A weighted, directed, strongly connected graph is known as a network.

**Geographic Information System (GIS)**

Geographic Information Systems (GIS) have been advancing since the 1970s. GIS serves as a crucial tool for location mapping, visualizing dynamic conditions, and aiding decision-making processes [6-8]. Geospatial data are instrumental in monitoring responses to accidents. During the response phase, GIS enables the analysis of real-time data, facilitating visualization and automation for more efficient decision-making. Research in GIS has concentrated on areas like shortest path analysis [9, 10]. This highlights the significant potential of GIS applications to reduce response times if geospatial information is utilized at the initial stages of accident response.

**Web service**

A web service is an internet-based technology defined by the W3C as "a software system designed to support interoperable machine-to-machine interaction over a network" [11].

**Web GIS**

Web GIS initially offered only client access through browsers, incorporating basic GIS functions to minimize system costs. Browsers are an optimal choice because they can extensively utilize GIS, reduce software expenses, simplify operations, and provide a user-friendly interface [12, 13].

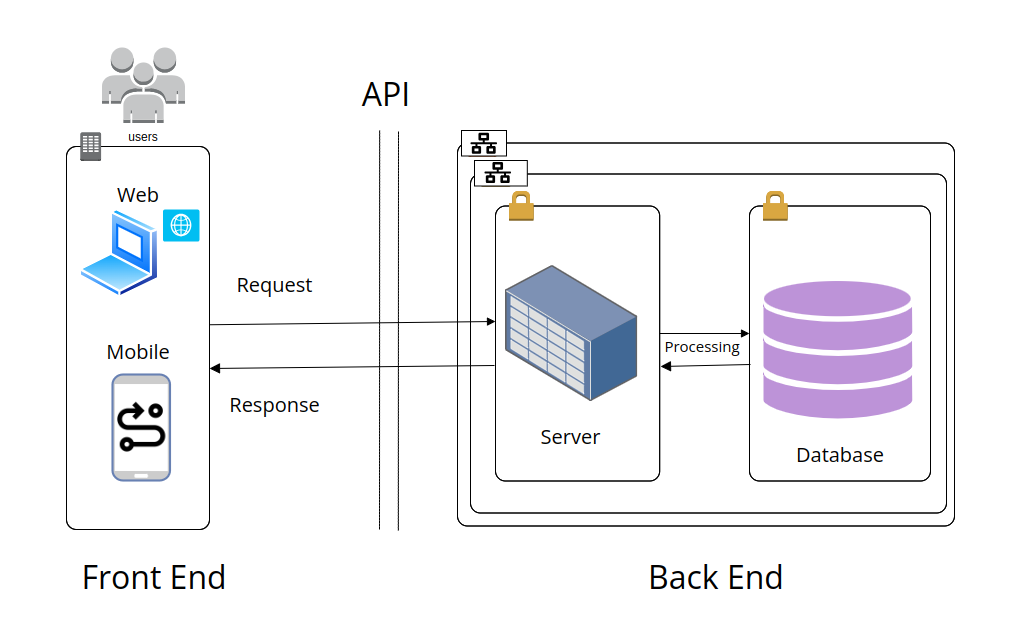
**System components**

The web framework is based on a three-tier architecture comprising the client layer, middleware layer, and database layer (Figure 2.3). These components work together to provide a unified interface for data consultation, requests, and decision-making for users. The database is accessed through the Internet, allowing users to interact with the data without needing to know the database's physical location. Users can consult, add, and modify the data as needed. The following sections will discuss these components in more detail.

The application utilizes PostgreSQL as its database management system, enhanced with the geospatial extension PostGIS. In addition to supporting relational queries, PostGIS enables users to perform spatial queries.

Typically, there are two basic approaches to deploying this GIS application on the Internet: server-side and client-side [14]. Web-based GIS users can utilize a web browser to navigate maps and perform basic spatial analysis. In other words, for a web-based GIS application, users enter specifications such as location or search requests on a web page to set up their environment for mapping or searching. The user's requests are sent to clients via HTML forms. These forms are processed by the Apache Web Server, which then passes the request to the GIS server. The GIS server subsequently queries the database.

The described system is designed to support two types of users: system administrators and teleoperators. The system administrator is responsible for updating the databases (user accounts and layers) and ensuring all functionalities are operational. Teleoperators handle the entry, editing, and deletion of all other data through a distributed Graphical User Interface (GUI). The system uses two types of data: geographic and semantic. The geographic database includes the region's road network map, monitoring stations, and all necessary geographic information to generate an output GIS on the web. The semantic data, maintained in a relational database, contains information about routing and traffic flows.



**Figure 2.3**

## 2.3. Technology Review

**Dijkstra’s Algorithm and Its Relevance**

Dijkstra’s algorithm, introduced by Edsger W. Dijkstra in 1959, is a seminal method for finding the shortest paths in a graph[4, 5] . It has been extensively utilized across various domains such as transportation networks, computer science, and urban planning due to its efficiency and robustness. This algorithm calculates the shortest path between nodes in a graph with non-negative weights, making it particularly suitable for road networks.

Several studies highlight the effectiveness of Dijkstra’s algorithm in optimizing routes. For instance, Botsis and Panagiotopoulos applied this algorithm to the campus of the International Hellenic University in Serres, Greece, determining the shortest routes between various locations and buildings[2]. Their findings confirmed the algorithm's efficiency in solving complex routing problems, providing a strong motivation for its use in traffic forecasting and road network navigation .

**Choosing Dijkstra’s Algorithm over Bellman-Ford**

The Bellman-Ford algorithm, developed by Richard Bellman and Lester Ford Jr., also calculates shortest paths in a graph and can handle graphs with negative edge weights . However, it operates with a higher time complexity (O(|E||V|)), making it less efficient compared to Dijkstra’s algorithm, which has a time complexity of O(|E| log |V|) . Given the extensive and complex road network of London and the University of Roehampton, the efficiency of Dijkstra’s algorithm renders it a more practical choice.

While the Bellman-Ford algorithm's ability to handle negative weights is advantageous in certain scenarios, this feature is unnecessary for navigating London’s road network, where negative edge weights do not exist. Thus, the additional complexity of Bellman-Ford is unwarranted for this application .

**Practical Applications of Dijkstra’s Algorithm**

Dijkstra’s algorithm has been successfully employed in various practical scenarios, such as optimizing urban rail networks and facilitating building evacuations, demonstrating its versatility and reliability [2]. These applications underscore the algorithm's suitability for traffic forecasting in road networks, including the complex and dynamic environment of London’s road infrastructure.

**Limitations of Existing Navigation Tools**

Current navigation tools, such as Google Maps, provide broad route planning services but often fall short in specific contexts, such as University of Roehampton campuses or intricate urban road networks[2, 5]. These tools may not always provide the most optimal routes or account for all the nuances of local road layouts .

This research aims to address these limitations by implementing a tailored approach using Dijkstra’s algorithm, specifically designed for the navigation needs of London’s road network and the University of Roehampton. By integrating on-site drone surveys for precise data collection and employing advanced software tools, the proposed system aims to deliver an efficient and user-friendly navigation solution.

**Advancing Current Research**

The study builds on previous research, such as the work by Botsis and Panagiotopoulos, which applied Dijkstra’s algorithm to campus navigation . By densifying the graph model and developing a web application, this research seeks to enhance user experience and improve the routing algorithm's effectiveness. This focused approach aims to fill existing gaps and extend the applicability of Dijkstra’s algorithm to more complex and dynamic traffic forecasting scenarios within road networks.

**2.4. Summary of Literature Review and Objectives**

**Findings:**

**- Dijkstra’s Algorithm:** Proven effective for finding shortest paths in graphs. Ideal for navigation in complex environments like university campuses and road networks [4, 5].

**- Real-World Applications:** Successfully used in optimizing campus routes (Botsis & Panagiotopoulos, 2020) and urban networks [2]. Highlights its suitability for detailed, constrained environments.

**- Limitations of Existing Tools:** Generic tools like Google Maps often fail in providing optimal routes for specific layouts and intricate networks [2, 5].

**Goal:** To reduce traffic congestion and enhance campus navigation through an optimized, tailored navigation system.

The literature underscores the robustness of Dijkstra’s algorithm in traffic forecasting and road network optimization. By addressing the limitations of existing navigation tools and tailoring solutions to specific contexts, this research aims to enhance route planning and navigation efficiency within London’s road network. Future work will focus on refining the algorithm’s implementation and expanding its applicability to broader urban transportation systems.

**References**

[2] K. Botsis and A. Panagiotopoulos. Application of dijkstra’s algorithm for the optimization of routes within the campus of the international hellenic university. *International Journal of Computer Applications*, 175(26):8–14, 2020.

[3] Y. Choi and S. Chi. Modeling the fastest and safest paths for building evacuation. *Safety Science*, 112:101–109, 2019.

[4] E.W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[5] Y. Li, Z. Peng, and Y. Huang. Simulation of optimal evacuation route planning in primary school based on dijkstra’s algorithm. *Journal of Safety Research*, 57:73–78, 2016.

1. M. Kataoka,"GIS for Homeland Security'",ESRl Press,Redlands,CA, 2007.  
   [7] AR. Pradhan, D.F. Laefer, W.J. Rasdorf. "Infrastructure management information system framework requirements for disasters'",ASCE,Journal of Computing in Civil Engineering,vol. 21,pp. 90-101,2007.  
   [8] M.I(. Lindell, C.S. Prater, "A hurricane evacuation management decision support system (EMDSS)"" Natural Hazards,vol. 40,pp. 627-634, 2007.
2. M.P. Kwan, 1. Lee, "Emergency response after 9/11: the potential of real-time 3d GIS for quick emergency response in micro-spatial environments'", Computing Environment and Urban Systems, vol. 29, pp. 93-113,2005.

[10] J. Han,Z. Yong,K.W. Dai,"The approach for shortest paths in fire succor based on component GIS technology'", SPIE-The International Society for Optical Engineering, Geoinformatics 2007, Geospatial Information Technology and Applications,vol. 6754,2007.

[11] Y.M. Fang, L.Y. Lin, C.H. Huang, T.Y. Chou, "An integrated information system for real estate agency-based on service-oriented architecture",Expert Systems with Applications,vol. 36,pp. 11039-11044, 2009.

[12] D. Nebert, A Whiteside, "OpenGIS catalogue services specification version2.0'", Technical report OGC 04-02Ir2, Open GIS Consortium Inc, 2004.  
[13] S. Tu,E. Normand,S. Kuchimanchi,V. Bizot,S. Shu,M. Abdelguerfi, and all. "Integrating web services into map image applications", In Computers and communications,2004. Proceedings of ISCC 2004. Ninth international symposium,pp. 44-49,2004.

[14] K.X. Tanga,N.M. Waters,'The internet,GIS and public participation in transportation planning'",Progress in Planning,vol. 64,pp. 7-62,2005.