SOFTWARE REQUIREMENTS SPECIFICATIONS

FOR
Net-Route: Implementation and Comparative Study
22nd Nov, 2023

Prepared By

Specialization	SAP ID	Name
AI & ML	500090966	Abhishek Joshi
AI & ML	500091867	Nihar
AI & ML	500090994	Naman Jain
AI & ML	500094133	Suraj Singh Bhandari

Department of Informatics Cluster
School Of Computer Science

Under the guidance of

Dr. Kingshuk Srivastava

UNIVERSITY OF PETROLEUM & ENERGY STUDIES,
DEHRADUN- 248007. Uttarakhand

Table of Contents

Revision History

Date	**Change**	**Reason for Changes**	**Mentor Signature**

1. INTRODUCTION

In today's fast-paced and interconnected modern world, efficient network routing stands as a hub in the seamless flow of data between devices and systems. Network routing is the most useful or we can say unsung hero of digital connectivity, the transfer of data packets from their source to their destination, ensuring smooth or uninterrupted communication across the ever-growing digital landscape.
This project embarks on a journey to create a network routing system that not only seeks to find the optimal or near-to-optimal path for the data packets but we are going to endeavor to comprehensively compare and contrast the various approaches employed by different routing algorithms. It is an exploration of the routing algorithms within the context of modern network routing.

The backbone of this project is data, which serves as the informational foundation upon which the network routing system is built. So, our data is primarily in the form of text files, representing the essence of the seamless integration and knowledge about network topology. The text files encapsulate a wealth of information regarding the arrangement, configuration, and interconnections of network elements, thus offering a comprehensive view of the digital ecosystem. This vital data empowers the system to make informed decisions about the optimal pathways for data packet transfer.

The core objective of our project is two fold:

1. Efficiency: We aim to develop a network routing system that can identify and implement the most efficient paths for data packets. This is achieved through a thorough analysis of network topology data.

2. Comparative Analysis: We are not content with merely achieving efficiency; we strive to understand how different routing algorithms approach the challenge of optimizing data transmission. Our project will offer a comprehensive comparison of various routing methods, shedding light on their respective strengths and weaknesses, and ultimately aiding in the development of best practices.

In conclusion, our endeavor represents a significant step forward in the field of network routing. By utilizing data as the cornerstone, we aim to create a system that not only ensures efficient and reliable data transfer but also contributes to the broader knowledge base of routing methodologies. In doing so, we are poised to play a pivotal role in the ongoing evolution of digital connectivity, making it even more seamless and efficient for all.

1.1 Purpose of the Project

In today's advanced world where communication depends totally on networking, the role of routing algorithms plays a vital role in the fast and smooth transfer of data from one source to destination.
Hence, this project aims to implement various algorithms to provide an optimal or near-to-optimal path and conduct a comparative study between different algorithms.

1.2 Target Beneficiary

The main beneficiaries of our project, which focuses on using algorithms like Dijkstra, Bellman-Ford, and A* algorithms, are network administrators and engineers trying to roadmap systems and improve network performance, students and researchers studying computer science and communication studies will find it valuable for educational and research purposes. In addition, software developers working on routing algorithms, especially for GPS navigation and other applications, can benefit from our work. If our project is open source, it can also serve the broader open-source community, providing free and easy-to-use routing solutions for various applications.

1.3 Project Scope

The scope of the project includes the use of Dijkstra, Bellman-Ford, A* algorithms and comparative analysis to find the shortest path in network routing. This also focuses on performance analysis through metrics such as transfer rate. The project aims to provide clear and accurate documentation, and for the network has been accessed by administrators, engineers, students and researchers. While the primary focus is on basic algorithms, the project also enables future developments, which adapt to evolving network needs and challenges

1.4 References

1. M. S. Nassr, J. Jun, S. J. Eidenbenz, A. A. Hansson and A. M. Mielke, "Scalable and Reliable Sensor Network Routing: Performance Study from Field Deployment," IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications, Anchorage, AK, USA, 2007, pp. 670-678, doi: 10.1109/INFCOM.2007.84. (Link to Document)

2. Wang, Junfeng & Miao, Yiming & Zhou, Ping & Hossain, M. Shamim & Rahman, Sk Md Mizanur. (2016). A Software Defined Network Routing in Wireless Multihop Network. Journal of Network and Computer Applications. 85. 10.1016/j.jnca.2016.12.007. (Link To Document)

3. Madkour, Amgad & Aref, Walid & Rehman, Faizan & Rahman, Abdur & Basalamah, Saleh. (2017). A Survey of Shortest-Path Algorithms. (Link To Document)

4. P. C. Reddy and P. C. S. Reddy, "Performance Analysis of Adhoc Network Routing Protocols," 2006 International Symposium on Ad Hoc and Ubiquitous Computing, Mangalore, India, 2006, pp. 186-187, doi: 10.1109/ISAHUC.2006.4290671. (Link To Document)

2. PROJECT DESCRIPTION

2.1 Data/ Data structure

1. HashMap
2. Map (using HashMap)
3. HashSet
4. Set (using HashSet)
5. ArrayList
6. List (using ArrayList)
7. Graph (using Map)

2.2 SWOT Analysis

Strength:
1. Promotes efficient data transfer.
2. This is independent of the network size to a very large network.

Weakness:
1. Implementation of this algorithm can be complex and requires deep knowledge.
2. Error handling may be a little complex to implement.

Opportunities:
1. Ongoing advancement in routing algorithms can provide opportunities for improving routing efficiency and adaptability.
2. Network Routing systems can play a crucial role in efficiently routing data to and from edge devices.

Threats:
1. This field is competitive with various routing protocols, so it makes it challenging to stand out.
2. Integrating a subpar solution with an existing system might be unsuitable in some cases.

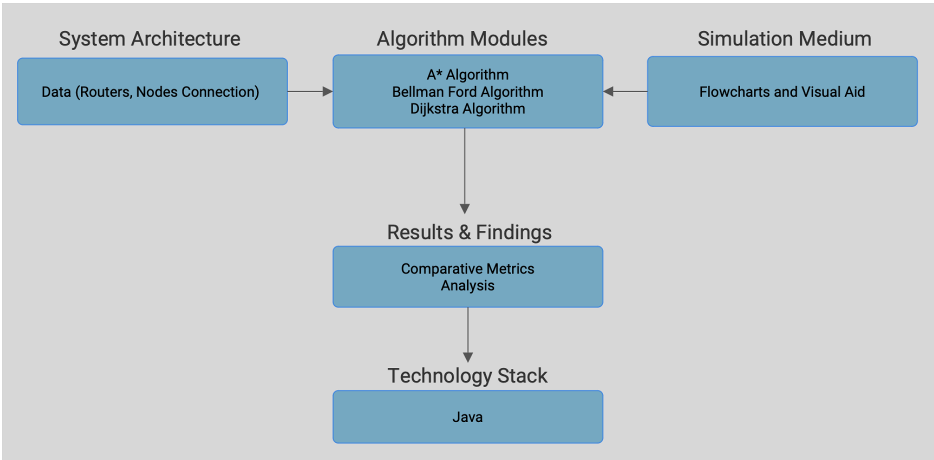
2.3 Project Features

Key features of the program include the use and comparative analysis of leading network routing algorithms, such as Dijkstra, Bellman-Ford, A*, and combining these algorithms to develop a comprehensive model of network data as information introducing and finding the shortest path to transfer data efficiently packets from the source to the destination. The program uses a data representation structured as a grid graph, where nodes are vertices and connections are edges. Performance metrics of packet transfer are measured to evaluate the efficiency of the algorithms. The program prioritizes documentation, ensures that the program is properly documented with code statements and external documentation files, and makes it accessible and manageable by users. Extensive testing is done to verify the efficiency and capability of the program if it can handle different network topologies. It discusses possible future advancements.

2.4 Design and Implementation Constraints

Project planning and implementation may face constraints related to resource availability, deadlines, and coordination issues. Limited hardware resources and processing capacity can affect program performance, especially for large network data sets. Time constraints can limit the depth of research and testing that can be done. Issues of compatibility with different operating systems and Java compilers can affect program portability. Moreover, unexpected changes in network topologies or real-time network dynamics can pose challenges for optimal routing decisions. Compliance with design best practices and algorithmic complexities must be balanced with these constraints to ensure successful project design and execution

2.5 Design diagram



3. SYSTEM REQUIREMENTS

3.1 User Interface

Hardware Requirements:

1. Operating System (OS): You can use Windows, macOS, or Linux (e.g., Ubuntu) based on your preference.
2. Processor: Intel Core 2 Duo, Athlon X2, or better
3. Memory: 8GB RAM
4. Storage: Sufficient storage space for code and dataset. SSDs are preferable for faster execution and data access.

Software Requirements:

1. Java Compiler: Choose a Java compiler like javac, jcg compatible with your selected operating system
2. Integrated Development Environment (IDE): You can use IDEs like Visual Studio Code or any other Java development environment that suits your preferences.

4. NON-FUNCTIONAL REQUIREMENTS

4.1 Performance requirements

Performance requirements are crucial for ensuring that your system meets certain performance criteria. Here are some examples:

1. Scalability: Define how the system's performance should scale as the network grows. This could include the maximum number of nodes or devices supported.
2. Reliability: Specify the system's expected uptime and the allowable downtime within a given time period.

4.2 Software Quality Attributes

Software quality attributes encompass various characteristics that contribute to the overall quality of the system. Here are some examples:

1. Reliability: Specify the system's ability to perform consistently and predictably under different scales.
2. Maintainability: Define how easily the system can be maintained and updated. This includes considerations for code readability, modularity, and documentation.
3. Portability: Define the system's ability to run on different platforms or environments.
4. Interoperability: Define the system's ability to work seamlessly with other systems.

APPENDIX A: GLOSSARY

Term	Description
Ad hoc	Created or done for a particular purpose or situation, often impromptu.
Anchorage	Typically refers to a mechanism used in routing protocols to establish a stable point or reference point for network nodes
Beneficiary	A person or entity that receives advantages or profits, especially from a will, insurance policy, or trust.
Cornerstone	A fundamental or essential part upon which something is built or developed.
Encapsulate	To enclose or surround something completely; to summarize or express the essential features of something.
Pivotal	Of crucial importance; Acting as the point or axis on which something revolves or depends.
Poised	Being in a state of balance or readiness; displaying composure and self-assurance.
Subpar	Below a standard or expectation; not reaching a satisfactory level of quality.
Symposium	A formal gathering or conference where experts discuss a particular topic, often in a series of presentations.
Ubiquitous	Present, appearing, or found everywhere; being or seeming to be everywhere at the same time.