ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIV, Issue V, May 2025

Evaluating the Impact of Hello Interval Timer on OSPF Performance for Real-Time Applications Using OPNET

Dr.A.V.N.Chandra Sekhar, T. Vinay, G.Nageswara Rao

Sasi Institute of Technology and Engineering, Tadepalligudem, A.P, 534101

DOI: https://doi.org/10.51583/IJLTEMAS.2025.140500003

Received: 30 April 2025; Accepted: 06 May 2025; Published: 29 May 2025

Abstract: In modern communication networks, various routing protocols differ in architecture, adaptability, processing delays, and convergence capabilities. For real-time applications, where nodes are often distributed randomly, Open Shortest Path First (OSPF) stands out as a widely adopted dynamic routing protocol due to its efficient dissemination of network topology information. The performance of OSPF in such scenarios heavily depends on parameters like convergence time, bandwidth utilization, and scalability. One critical parameter, the **Hello Interval timer**, influences the speed and stability of OSPF neighbor discovery and maintenance. This paper evaluates the impact of modifying the Hello Interval timer on OSPF protocol performance in real-time application environments. Simulations are conducted using OPNET Modeler 14.5 to analyze how different Hello Interval settings affect network convergence and reliability.

Keywords: Open Shortest Path First (OSPF), Hello Interval, Real-Time Networks, OPNET Simulation

I. Introduction

Mobile Ad-hoc Networks (MANETs) are self-configuring networks consisting of wireless nodes that operate without any centralized control. In MANETs, each node functions both as a host and as a router, forwarding packets to other nodes as needed. These networks feature highly dynamic topologies due to frequent changes in node positions. The ease of deployment and high mobility make MANETs a compelling subject for research.

Among the key challenges in MANETs are **Quality of Service (QoS)** and **routing**. In any internetwork, routing refers to the process of transferring data from a source to a destination. In MANETs, routing protocols are vital and can be classified into three categories: **Reactive**, **Proactive**, and **Hybrid** protocols.

Performance evaluation is a crucial aspect of protocol development, as it provides insights that guide the design and implementation of network solutions across various applications.

Section II presents the literature review. Section III discusses the routing protocols used in MANETs. Section IV provides an overview of the OSPF routing protocol. Section V outlines the methodology adopted for this study. Section VI introduces the OPNET Modeler used for simulation. Section VII describes the practical implementation of OSPF using OPNET, and Section VIII presents the simulation results and analysis.

II. Literature Review

Only a limited number of studies in the existing literature focus on optimizing OSPF performance for wireless networks. Wollman and Barsoum [4] recommend adjusting default timer values and increasing the use of route aggregation to address specific performance issues associated with OSPF. Baker [5] proposes modifications to OSPF for MANETs within the framework of OSPFv3 for IPv6. His suggestions include support for movement across different areas, the definition of a new interface type specific to mobile ad hoc networks, improved link metric handling, and enhanced scalability by limiting the number of neighbor relationships. He also addresses the integration of IPv4 and IPv6.

Researchers at Bolt, Beranek, and Newman (BBN) have developed a proprietary version known as “Radio OSPF,” which utilizes side information available at wireless routers. This version aims to eliminate the need for OSPF Hello messages in neighbor discovery and modifies designated router flooding behavior [6]. Additionally, Thomas R. et al. [7] describe strategies to adapt OSPF for wireless environments in a scalable and efficient manner.

Routing Protocols of MANET

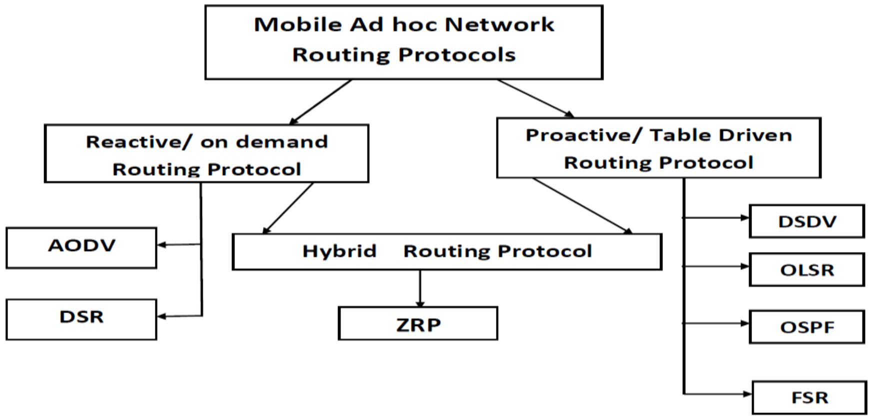
Routing protocols in Mobile Ad-hoc Networks (MANETs) are generally classified into three categories: **Proactive**, **Reactive**, and **Hybrid** routing protocols, as illustrated in Figure 1.

Proactive Routing Protocols

Proactive routing protocols, also known as table-driven protocols, are based on traditional link-state approaches developed for both wired and wireless networks. In these protocols, each node maintains routing information for every other node in the network, regardless of whether communication is needed. Route creation and maintenance are handled through periodic and

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIV, Issue V, May 2025

event-triggered updates. Periodic updates occur at regular time intervals, while event-triggered updates are initiated when network changes occur.



Although proactive protocols are not bandwidth-efficient—since control messages are continuously exchanged regardless of data transmission needs—they offer low-latency route availability. This makes them suitable for environments with frequent and numerous data sessions. Their main advantage is the immediate availability of routing information, which helps in establishing sessions quickly. However, the downside is the overhead of maintaining routing tables and exchanging control messages, even when routes are not in use. Proactive protocols perform particularly well in highly dynamic topologies where reduced delay is essential. Common examples include **OSPF**, **OLSR**, and **DSDV**.

Reactive Routing Protocols

Reactive or on-demand routing protocols create routes only when they are needed by the source node. When a node requires a route to another node, it initiates a route discovery process across the network. This process continues until a valid route is found or all possible options are exhausted. Once a route is established, it is maintained as long as it is valid or required. Reactive protocols reduce overhead compared to proactive ones but may introduce delays during the route discovery phase.

Hybrid Routing Protocols

Hybrid routing protocols combine the strengths of both proactive and reactive approaches to optimize performance. Typically, they use a proactive strategy to maintain routes to nearby nodes (within a predefined zone) and a reactive strategy for nodes outside that zone. This combination helps balance overhead and latency, making them suitable for larger and more scalable network environments. An example of a hybrid routing protocol is the **Zone Routing Protocol (ZRP)**.

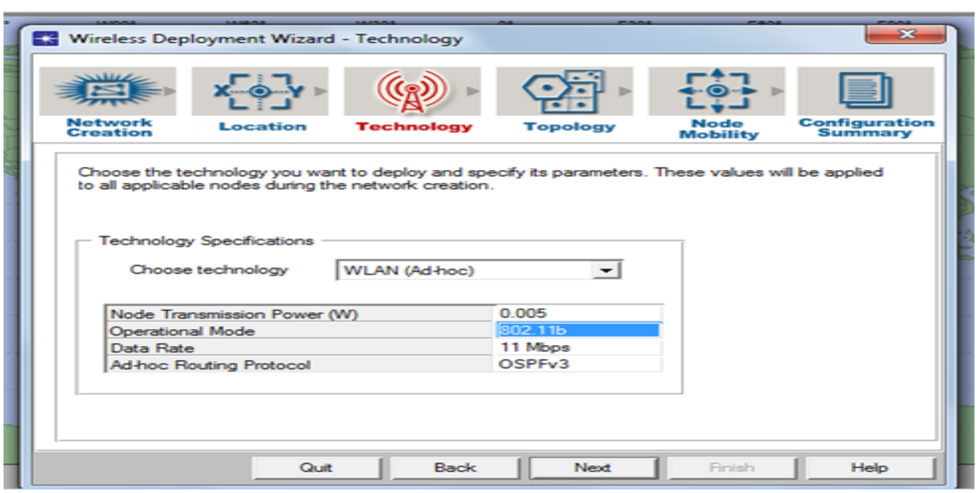


Fig 2 Wireless Network Deployment model

III. Methodology

Validating a new framework or protocol in a networked environment can be achieved through several approaches, including **mathematical modeling**, **simulation**, and **test-bed emulation**. Mathematical modeling is the fastest among these methods; however, it becomes less accurate and often impractical when dealing with complex systems involving multiple variables and dynamic interactions.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIV, Issue V, May 2025

Simulation, on the other hand, enables detailed modeling of network behavior, often at the packet level, allowing for in-depth analysis of protocol performance under various conditions. It offers a good balance between realism and flexibility, making it a widely accepted approach for network research.

Test-bed emulation involves implementing the proposed protocol or framework on real hardware at a small scale. While this method provides realistic results, it is costly and frequently faces unexpected engineering challenges.

In this study, **simulation** was selected as the preferred method due to its cost-effectiveness and ability to provide repeatable and controlled testing environments. The simulations were conducted using **OPNET Modeler**, which offers comprehensive tools for modeling and analyzing network protocols and behaviors.

OPNET Modeler

OPNET (Optimized Network Engineering Tools) is a leading commercial discrete event simulator widely used in both industry and academia [8]. It is built on object-oriented principles and utilizes a hierarchical modeling structure to simulate complex network behaviors. In OPNET, a **network model** is composed of **node models**, which in turn consist of **processes**, **transmitters**, and **receivers**.

Each process model represents the behavior of a network node using a **state transition diagram**, where transitions are triggered by specific conditions or events occurring during the network's operation. The OPNET library offers a rich set of predefined components including routers, switches, and both fixed and mobile wireless workstations, making it a flexible tool for a wide range of simulations.

OPNET integrates the **C programming language** with its state transition diagrams and introduces a specialized language called **Proto-C** for designing and implementing process models. Additionally, **C++** can be used to extend existing models for more customized behavior.

For analysis and troubleshooting, OPNET provides a built-in debugging tool known as the **OPNET Debugger (ODB)**, which enables users to trace packet flows and monitor the movement of mobile nodes in the simulated environment. This makes OPNET a powerful platform for modeling, testing, and refining network protocols and architectures.

OPNET-Based Practical OSPF

Simulations were conducted using **OPNET Modeler version 14.5** to evaluate the impact of the **Hello Interval timer** on OSPF performance in wireless networks. The primary performance metrics analyzed in this study are **end-to-end delay** and **throughput**.

Delay refers to the total end-to-end delay experienced by packets received at the Wireless LAN MAC layer of all WLAN nodes and forwarded to the higher layers. This includes delays due to medium access at the source MAC, the reception of fragmented frames, and transmission through the access point (AP), if AP functionality is enabled.

Throughput represents the total number of bits per second (bps) successfully forwarded from the wireless LAN layers to the higher layers across all WLAN nodes in the network.

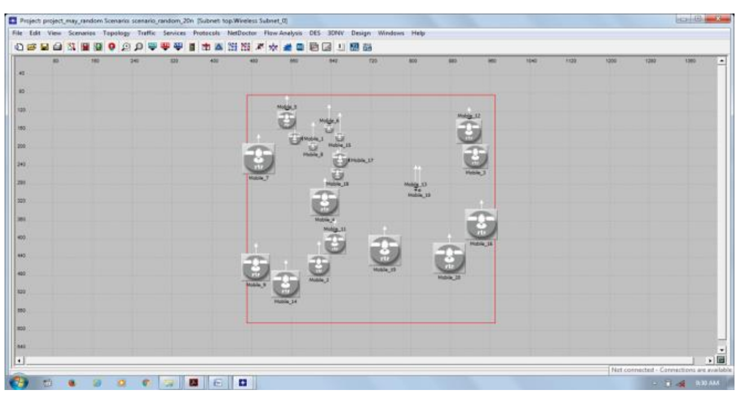
To analyze OSPF behavior under varying conditions, the simulation environment was set within a **500m x 500m** area. Two different node deployment scenarios were used:

A **small-scale network** consisting of **20 nodes**, and

A **medium-scale network** with **50 nodes**.

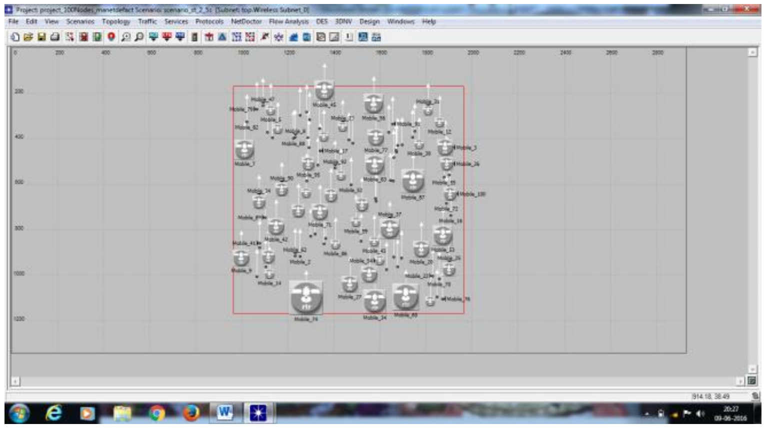
Node placement strategies for both network sizes are illustrated in the following figure.

Random deployment with 20 nodes



ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIV, Issue V, May 2025

Random deployment with 50 nodes



Simulation Environment

Area	1000m x 1000m
Nodes	20, 50
Node Placement Model	Random
Mobility Model	Random Way Point
Node Transmission Power	0.005
Operational mode	802.11b
Data rate	11Mbps
Simulation time	1000 sec
Hello interval	1,1.05,1.10,1.25,1.5,1,75,2.0 sec

IV. Results and Analysis

To assess the performance of the OSPF routing protocol, the key metric evaluated in this study is **end-to-end delay**.

Figure A illustrates the variation in delay as the **Hello Interval time** is adjusted in a network of **20 mobile nodes**. The results indicate that **delay decreases** when the Hello Interval is set to a **lower value**, and **increases** as the Hello Interval becomes **longer**. This trend suggests that more frequent Hello messages lead to quicker neighbor detection and faster convergence, thereby reducing communication delays in the network.

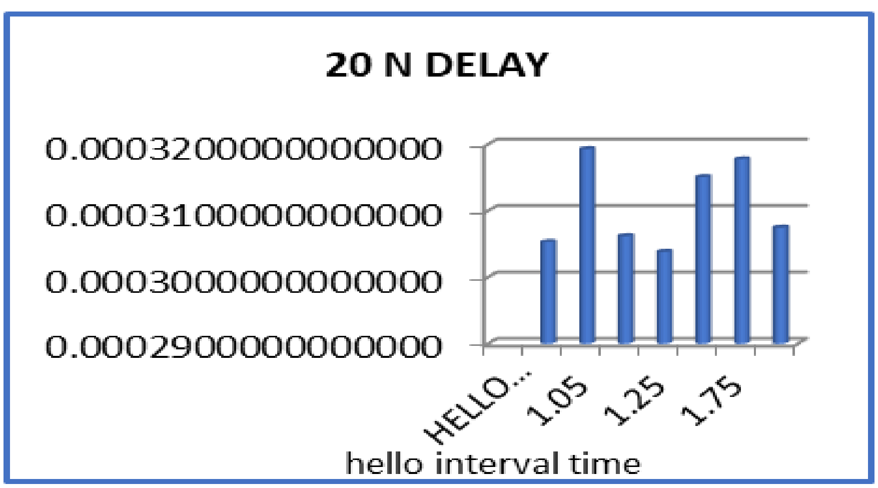


Fig a) Variation of delay with hello interval for 20 nodes

The variation in delay with different Hello Interval times for a network of 50 mobile nodes is shown in Figure B. The results show that delay is lowest when the Hello Interval time is short, and highest when the HelloInterval time is long.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIV, Issue V, May 2025

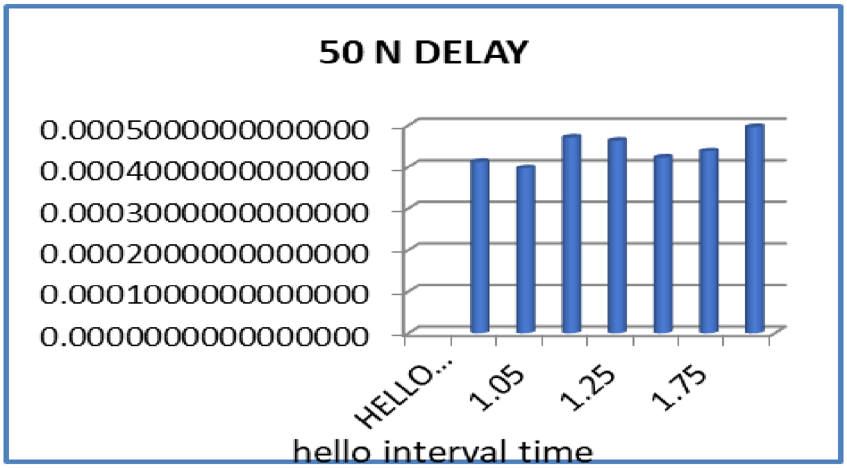


Fig b) Variation of delay with hello interval for 50 nodes

Throughput: Represents the total number of bits (in bits/sec) forwarded from wireless LAN layers to higher layers in all WLAN nodes of the network.

The variation of Throughput with varying the number of Hello Interval time is shown in the Figure a) 20 nodes, b) 50 nodes

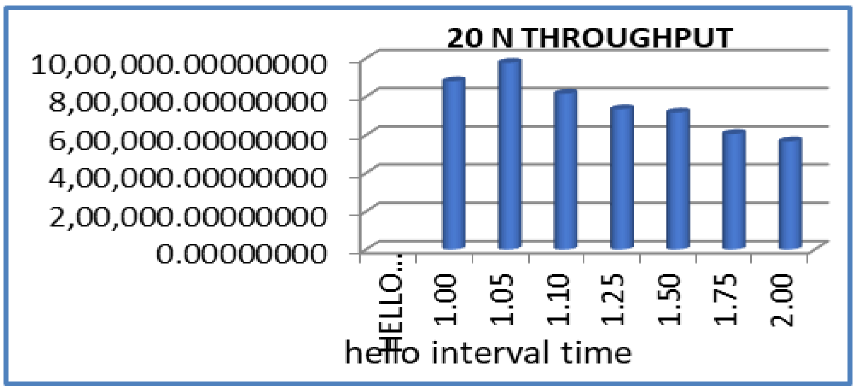


Fig a) Variation of Throughput with hello interval for 20 nodes

The variation of Throughput of 20 nodes with Hello Interval time of 20 mobile nodes is shown in the Figure a. It describes that throughput is maximum when Hello Interval time is minimum and comparatively minimum at maximum Hello Interval time.

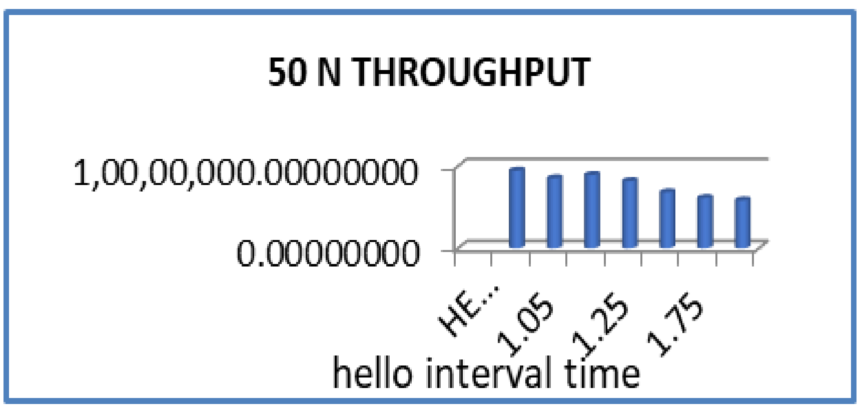


Fig b) Variation of Throughput with hello interval for 50 nodes

The variation of Throughput of 50 nodes with varying the Hello Interval time of 50 mobile nodes is shown in the Figure a. It describes that throughput is minimum when Hello Interval time is less and comparatively maximum at maximum simulation time.

Throughput: Throughput refers to the total number of bits per second (bps) successfully forwarded from the wireless LAN layer to the higher layers across all WLAN nodes in the network. The variation in throughput with different Hello Interval times is illustrated in Figure A for 20 nodes and Figure B for 50 nodes. The results demonstrate how changes in the Hello Interval setting impact the overall data transmission efficiency in both small and medium-sized networks.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIV, Issue V, May 2025

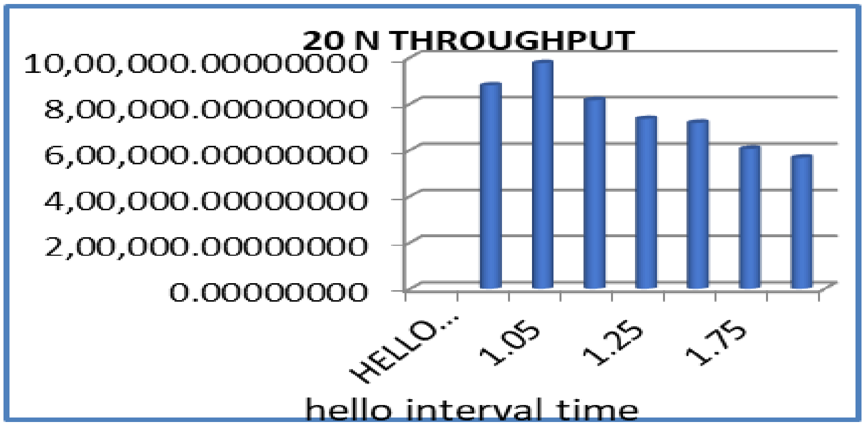


Fig a) Variation of Throughput with hello interval for 20 nodes

The variation in throughput for a network of **20 mobile nodes** with different **Hello Interval times** is shown in **Figure A**. The results indicate that **throughput is highest** when the HelloInterval time is **lowest**, and **decreases** as the Hello Interval time increases.

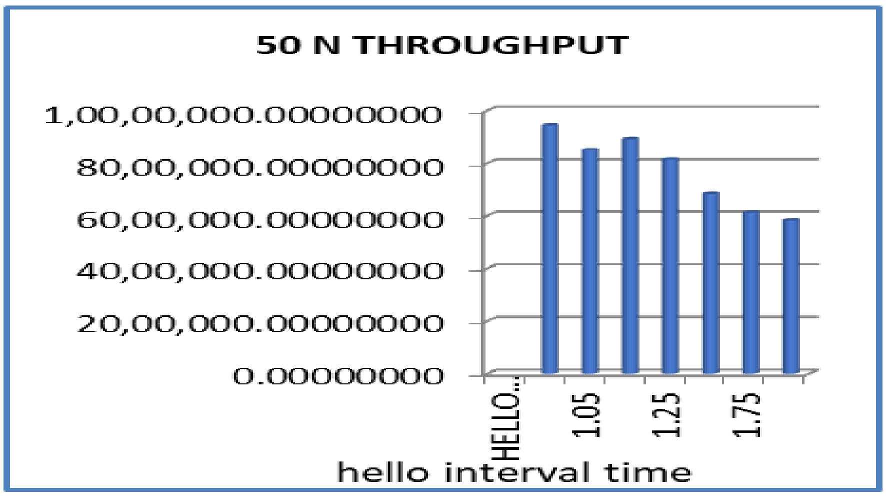


Fig b) Variation of Throughput with hello interval for 50 nodes

The variation in throughput for a network of 50 mobile nodes with different Hello Interval times is shown in Figure B. The results show that throughput is lower when the Hello Interval time is shorter, and higher when the Hello Interval time is longer, indicating improved stability and data flow at higher intervals in larger networks.

V. Conclusion & Future Scope.

The study demonstrates that the performance of the OSPF protocol, particularly in terms of delay and throughput, is significantly influenced by the **Hello Interval time**, with its impact varying based on **network size** and **node mobility**. Lower Hello Interval values improve responsiveness in smaller networks but may lead to overhead in larger ones, whereas higher intervals can enhance stability in medium to large-scale networks.

For future work, it is essential to develop adaptive mechanisms that dynamically adjust the Hello Interval based on network conditions such as size, topology changes, and traffic load. The integration of **Soft Computing Techniques**, such as fuzzy logic or machine learning algorithms, may provide an intelligent and efficient way to optimize Hello Interval settings, thereby enhancing the overall performance and scalability of OSPF in mobile ad hoc environments.

References

1. Wollman, D., & Barsoum, A. "Improving OSPF Performance in Wireless Networks through Timer Adjustment and Route Aggregation," IEEE Communications Surveys & Tutorials, vol. 10, no. 2, pp. 42–56, 2008.
2. Baker, F. "OSPF Extensions for Mobile Ad Hoc Networks," IETF Draft, Internet Engineering Task Force, 2004. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-baker-manet-ospf-extension
3. BBN Technologies. "Radio OSPF: A Routing Protocol for Wireless Routers," Technical Report, Bolt, Beranek, and Newman, 2006.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIV, Issue V, May 2025

4. Thomas, R., Gupta, P., & Singh, A. "Scalable Adaptation of OSPF for Wireless Ad Hoc Networks," Proceedings of the International Conference on Wireless Networks, 2009.Wollmn. W. and Y. Barsourn 'Overview of Open Shortest Path First Version 2 (OSPF "2)
5. Baker, F., "An outsider's view of MANET'' Internet l-Draft: draft-baker-manet-review-0l.ur (expired), Mach 2012.
6. Zavgen, J, Presentation at Panel Session on Adhoc Networks, IEEE Milcom Conference, October 1997
7. Thomas R.et .al "A WIRELESS INTERFACE TYPE FOR OSPF", 0-7803-814&8103617.W (C) 2003 IEEE