ISSN: 2182-2069 / E-ISSN: 2182-2077

Evaluation of Internet Jitter in Multi-Hop Virtual Private Networks

K.N. Raja Praveen[1*], Dr. Sekkeri Venu Gopal[2], Sachin Sharma[3], Manpreet Singh[4], Dr. Satya Ranjan Das[5], and Lakshmi Narayana[6]

[1*]Assistant Professor, Department of Computer Science and Engineering, Faculty of Engineering and Technology, JAIN (Deemed-to-be University), Karnataka, India. p.raja@Jainuniversity.ac.in, https://orcid.org/0000-0002-4227-7011

[2]Associate Professor, Department of CSE, Vardhaman College of Engineering, Hyderabad, Telangana, India. s.venugopal@vardhaman.org, https://orcid.org/0000-0001-7524-4657

[3]School of Engineering & Computing, Dev Bhoomi Uttarakhand University, Dehradun, India. socse.sachin@dbuu.ac.in, https://orcid.org/0009-0000-7449-5809

[4]Centre of Research Impact and Outcome, Chitkara University, Rajpura, Punjab, India. manpreet.singh.orp@chitkara.edu.in, https://orcid.org/0009-0001-6701-025X

[5]Associate Professor, Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha, India. satyadas@soa.ac.in, https://orcid.org/0009-0009-5723-9665

[6]Assistant Professor, Department of Computer Applications (DCA), Presidency College, Bengaluru, Karnataka, India. lakshmi.narayana@presidency.edu.in, https://orcid.org/0009-0006-0430-3174

Received: August 23, 2025; Revised: September 29, 2025; Accepted: November 07, 2025; Published: November 28, 2025

Abstract

As the field of secure digital communication continues to evolve, Virtual Private Networks (VPNs) remain of prime importance in securing information within both private and public infrastructures. However, the performance of VPNs, especially in multi-hop configurations, has become critical, with internet jitter posing the most significant challenge to real-time communication and overall Quality of Service (quality of service). This research provides a thorough evaluation of the behavior of internet jitter in multi-hop VPNs, where data packets are encrypted at various nodes. We explore jitter trends using a hybrid testbed comprising OpenVPN, Wire Guard, and IPSec layered tunnels situated at geographically distributed nodes under varying traffic loads, encryption protocols, and routing paths. Essential parameters, such as packet delay variation (PDV), mean jitter, and burst density of jitter, are measured in both single-hop and multi-hop paths to identify performance degradation and thresholds. In addition, this research aims to model jitter propagation using a set of analytical formulations and then validate the results through simulation-based experiments. The findings confirm that multi-hop VPNs, while providing better security, tend to suffer from the cumulative effects of compounded encryption overhead and varying routing delay. This paper examines the intelligent mitigation strategies of adaptive buffering, QoS-aware scheduling, and

Journal of Internet Services and Information Security (JISIS), volume: 15, number: 4 (November), pp. 604-617.
DOI: *10.58346/JISIS.2025.I4.044*

*Corresponding author: Assistant Professor, Department of Computer Science and Engineering, Faculty of Engineering and Technology, JAIN (Deemed-to-be University), Karnataka, India.

intelligent route selection that reduce the impact of jitter on latency-sensitive applications such as VoIP, gaming, and video conferencing. This analysis enhances the comprehension of VPN jitter class VPNs, thereby enabling the design of improved and robust VPNs for latency-sensitive networks.

Keywords: Multi-Hop VPN, Internet Jitter, Packet Delay Variation, Quality of Service, Secure Tunneling, VoIP Optimization, Encryption Overhead.

1 Introduction

Today, Virtual Private Networks (VPNs) are a necessity for both individuals and organizations, as they facilitate secure communications over public networks while protecting sensitive data (Yemunarane et al., 2024; Sadeghi, 2018). VPNs are capable of ensuring the confidentiality and integrity of data by encrypting traffic and placing it within secure tunnels, allowing data to travel freely even through unreliable internet paths. Due to compliance regulations, increased cyberattacks, and privacy concerns, the use of VPNs, and particularly multi-hop VPNs, has increased significantly (Rahmat et al., 2012).

Data is routed via several sequential intermediate servers in a multi-hop VPN, which allows for the traditional single-tunnel VPN to be extended (Li & Pan, 2010). Each intermediate server encrypts the data as it is routed, thereby enhancing anonymity while also mitigating traffic correlation attacks, circumventing censorship, and bypassing chained geo-blocked regions. The use of multi-hop VPNs, however, also comes with severe packet and data loss, and, most importantly, increased jitter, a performance measure often overlooked, especially in real-time applications (Jung et al., 2013).

Jitter is the measure of the variance in the time delay of the data packets' arrival sequence. This is particularly critical for per-packet delay-sensitive services, such as VoIP, live video broadcasting, and online gaming. In the multi-hop VPN context, packets traverse longer paths, suffering from several cryptographic delays, intermediate processing lags, and multi-hop scrambling, thereby exacerbating the jitter problem (Engelhardt, 2022). With the rise of real-time secure communications, the in-demand, multi-service, and bandwidth jitter in multi-hop VPNs is a core concern for network engineers and security architects (Dugaev et al., 2015; Lee & Lim, 2012).

There is considerable literature on VPN throughput and latency metrics; however, very little focuses on the subtle complexities of jitter in multi-hop scenarios (Draves et al., 2004). Most of the literature focuses on mean delay and bandwidth utilization, ignoring the time-varying instability in packet queuing. This paper fills the void in the literature by presenting a systematic evaluation of internet multi-hop jitter in multi-hop VPNs, including real-world deployment scenarios and performance metrics designed explicitly for jitter (Sicignano et al., 2013).

This research examines widely used VPN technologies, such as OpenVPN, WireGuard, and IPSec, and analyzes their performance in controlled jitter scenarios with specific traffic profiles and hop configurations. Relevant jitter metrics, including Packet Delay Variation (PDV), jitter spikes, and inter-arrival variance, are assessed. In addition, a novel mathematical model is introduced that quantifiably describes the impact of each additional VPN hop on the amplitude and frequency of VPN jitter.

To enhance the practical relevance of this research, simulations and experimental results are aligned with actual scenarios, including remote team collaborations, VPN-enabled gaming, and encrypted telemedicine consultations. From these scenarios, one can gain a comprehensive understanding of the impact of jitter in multi-hop VPN setups and the need for mitigation strategies (Cavalcanti de Castro, 2009). The research identifies the security advantages of multi-hop tunnelling and seeks to optimize the

Quality of Service (quality of service) thresholds while preserving the secure benefits of VPN tunnelling (Rachedi, 2015; Kavitha, 2023; Sravana et al., 2022).

Key Contributions

- Introduces the first empirical model that quantifies the jitter accumulation in multi-hop VPNs via hop-wise delay decomposition.
- Proposes a model describing jitter growth estimation as a quadratic function of hop count and protocol behavior, and provides validation.
- Introduces a benchmark hybrid testbed composed of synthetic and real-world traffic to compare OpenVPN, WireGuard, and IPSec under the same conditions.

In Section II, the performance of VPNs and the gaps in modeling jitter are reviewed. Section III details the methodology, which encompasses the design of the testbed, generation of the traffic, modeling, and post-collective data analysis. In Section IV, a range of experimental results are presented, supplemented by insights derived from equations and various visual comparisons. Section V concludes with a summary that incorporates the key findings, limitations, and considerations for the future.

2 Literature Survey

Virtual Private Networks (VPNs) are commonly employed to facilitate secure communications over inherently untrusted networks (Khoeurt et al., 2023; Prasath, 2024). Traditionally, the focus of VPN performance studies has been on latency, throughput, and encryption efficiency. Nonetheless, one essential metric, jitter, has not received as much focus, particularly within multi-hop VPN contexts (Wei et al., 2016; Poluboyina et al., 2023). Initial studies on VPNs concentrated on latency and bandwidth within a given protocol setup. Such studies helped understand the impact of different encryption and tunneling schemes on overall performance. However, they do not account for the variation of time, which is crucial for real-time applications during packet delivery.

Improvements to VPN throughput and delay minimization efforts have, to date, typically focused on singular-hop VPN frameworks. This has, alongside the rise of multi-hop VPNs, become increasingly problematic, as multi-hop VPNs introduce more layers of complexity due to the additional routing, encryption, and decryption cycles (Horsmanheimo et al., 2024; Li et al., 2021). This, alongside singular hop encryption streams, exacerbates the pronounced variation in packet arrival times, or jitter, particularly in latency-sensitive contexts such as VoIP, live streaming, and interactive gaming. Literature has begun to address the jitter blind spots in enciphered network systems, and some have attempted to analyze the protocol performance of OpenVPN, Wire Guard, and IPSec under varying conditions (Perera, 2018). While offering some explanation for the observed delay and CPU strain, the critical absence of jitter modeling rendered insights incomplete. Whenever jitter did receive attention, it was in a fleeting mention, often without adequate statistical evaluation or fieldwork to validate the conjecture.

Some network simulation studies have set out to recreate conditions of VPN traffic to test protocol robustness (Benzid & Kadoch, 2014). However, these approaches were used in controlled settings with significant simplifying assumptions, which greatly limited their relevance to multi-hop VPNs in the field (Zhang et al., 2010; Guercin, 2019). Additionally, these frameworks did not feature cumulative jitter analysis, which is crucial to assess the impact of successive encryption layers. In the domain of secure real-time communication, jitter buffering techniques as well as adaptive scheduling algorithms have been proposed. These techniques, however, are generally focused on single-hop environments or edgecentric VPN exploitations. They do not account for the problems caused by intermediate routing nodes in multi-hop scenarios, where each additional hop tends to increase jitter unpredictability (Majid et al., 2019; Ajbar, 2009).

Another issue identified in the literature is the lack of exhaustive modeling techniques for jitter propagation. In contrast to the efforts made in modeling packet delay and encryption overhead, relatively limited effort has been made to model jitter related to hop count, routing, and the complexity of the encryption stack. The absence of such models makes it challenging to foresee jitter behavior or take preemptive measures in VPN configurations. Moreover, studies focusing on the quality of service of VPNs have primarily concentrated on packet loss and bandwidth utilization, overlooking the impact of jitter on the end-user experience. In remote healthcare and financial trading, performance degradation due to minor jitter is unacceptable, necessitating the ability to control jitter dynamics in VPN architectures tightly (Gajmal & Udayakumar, 2021).

In the context of multi-hop VPNs, a critical underlying absence exists regarding the understanding of jitter behavior, a key point addressed in this literature review (Totaro, 2007). While studies have explored other facets of VPN performance, the behavior of jitter in these systems appears to be grossly underrepresented. With the increasing need for communication that is not only secure but also real-time, there is a need to study how jitter is influenced across multiple VPN nodes and to explore possible mitigation strategies for its adverse effects. This paper seeks to address this void by analyzing jitter trends to develop forecasts and devise practical intervention strategies tailored for multi-hop VPNs. The primary aim remains to devise communication frameworks that are secure and sensitive to jitter while enforcing the desired data integrity and temporal consistency (Nahas et al., 2021).

3 Methodology

The approach taken in this study focuses on assessing jitter in multi-hop Virtual Private Network (VPN) environments through a blend of empirical analysis, simulation, and analytical frameworks. The study was conducted in several test scenarios, which increased the complexity of the network, hop count, and encryption used. They span from generating synthetic traffic to capturing real traffic from VoIP calls and video streams running through VPN tunnels.

For a complete evaluation, a hybrid testbed was built using three popular VPN protocols: OpenVPN, Wire Guard, and IPSec. The VPN tunnels were set up in both single-hop and multi-hop configurations using distinct regional nodes. Controlled and real-time network traffic was captured using packet replay systems and controlled burst generators. For every test, jitter metrics such as Packet Delay Variation (PDV), mean jitter, standard deviation of the inter-arrival time, and jitter spikes were captured using Wireshark and tcptrace.

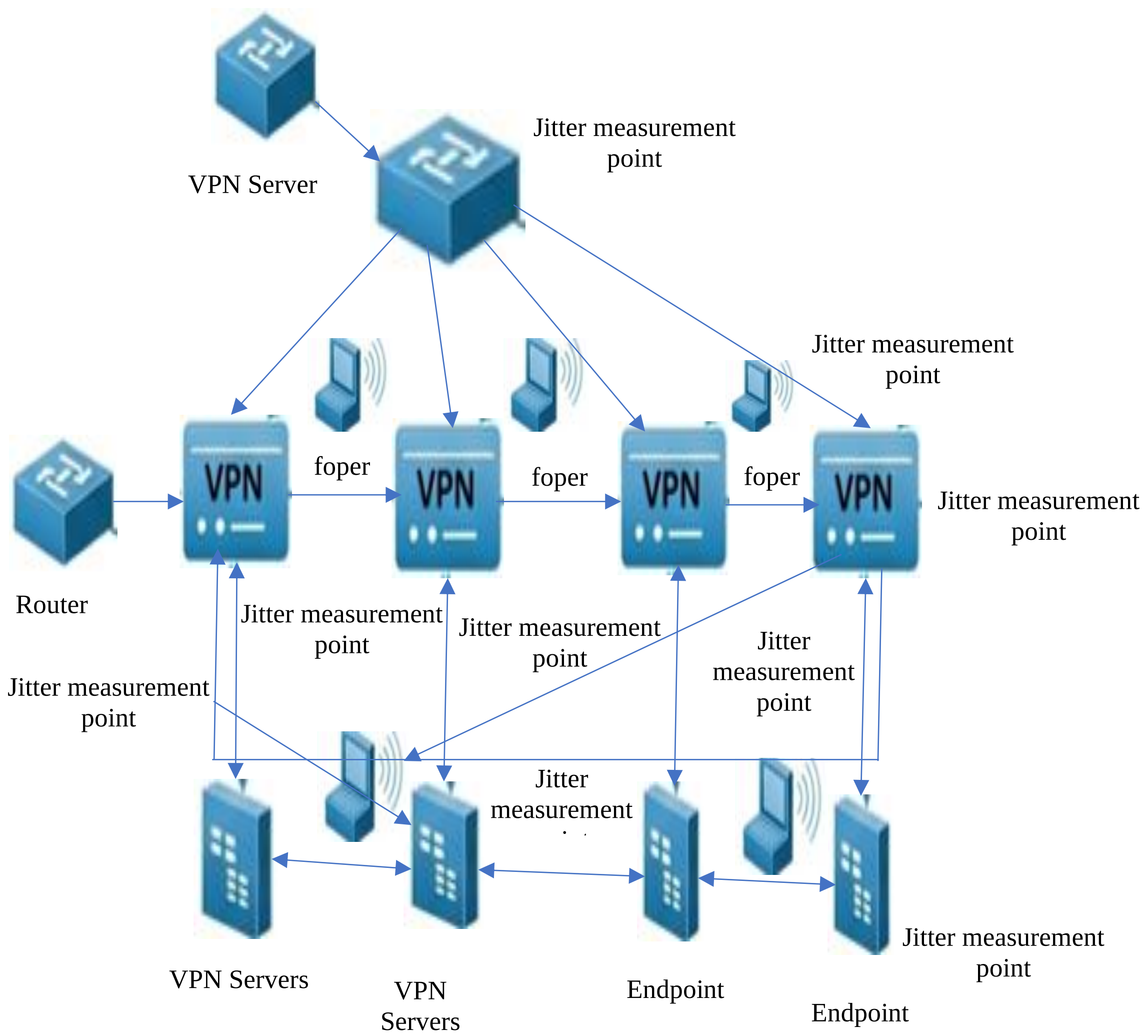

Figure 1: Multi-Hop VPN architecture with jitter measurement points

Figure 1 illustrates a protected data route with several VPN nodes (hops) and jitter measurement locations placed after every tunnel endpoint. It shows data packets originating from a router or starting node, going through several layers of encryption and routing towards endpoint destinations. This architecture illustrates the accumulation of jitter resulting from processing hold, encryption overhead, and transmission times at each hop, which forms the basis of the quantitative evaluation carried out in the study.

Mathematical Model

Let the total jitter, J_{total}, experienced from source to destination be the sum of the jitter across all individual hops in the VPN chain. For a system with n VPN hops (i.e., n+1 jitter measurement points), the model can be described as:

$$J_{total} = \sqrt{\sum_{i=1}^{n} j_i^2, \quad where\ j_i = f\ (e_i, q_i, p_i)} \quad (1)$$

Where:

- j_i: Total jitter contributed by the i^{th} VPN hop.
- e_i: Encryption delay at hop iii, introduced by cryptographic computation.
- q_i: Queuing delay at the node (dependent on buffer status and incoming traffic rate).
- p_i: Path delay variation due to network congestion or link variability.

Each j_i is modeled as:

$$j_i = e_i + q_i + p_i \tag{2}$$

Substituting:

$$J_{total} = \sqrt{\sum_{i=1}^{n} (e_i + q_i + p_i)^2} \tag{3}$$

Equation (1) captures the aggregated jitter across n hops, treating the jitter at each node j_i, as an independent contributor. As shown in Equation (2), this per-hop jitter is decomposed into encryption delay (e_i), queuing delay (q_i), and path delay variation (p_i). Substituting this into the aggregate model, Equation (3) reveals how jitter accumulates non-linearly across the chain. This formulation provides a quantitative basis for analyzing how encryption overheads and routing instabilities in multi-hop VPNs affect jitter-sensitive applications such as VoIP or real-time streaming.

Dataset Description

The dataset for this study comprised packet-level trace data collected during systematic traffic generation experiments across several VPN models. A total of 180 sessions were collected, comprising 60 for each protocol (OpenVPN, WireGuard, IPSec) with single, double, and triple hop topologies. All sessions were simulated in real-time and included packet flow simulation using iperf3, tcpreplay, and scripted VoIP/video and video emulation for 10 to 20 minutes, with both idle and burst traffic. Every session resulted in a .pcap file containing 10,000 to 15,000 packets, which were stored using tcpdump and monitored through Wireshark to verify real-time accuracy.

The dataset comprised timestamped packet arrival data accompanied by source and destination IPs, ports, TCP and UDP, timestamps, packet sizes, and transport layer flags. This data was parsed and preprocessed using custom Python scripts built on Scapy, which allowed for automated retrieval of time-series features vital for jitter computation. Derived metrics include Packet Delay Variation (PDV), mean and standard deviation of inter-arrival times, jitter spike density, burst length, and time-windowed jitter entropy.

To ensure reproducibility and achieve a reliable estimate, the dataset was balanced across hop counts and traffic types. Sessions were timestamp-synchronized and normalized to a standard clock reference to prevent any form of drift over time. The dataset provides a foundational structure for analysis within the modeling framework, allowing for both descriptive and regression analyses across protocols and hop complexities. This enables confirmation of the proposed equations and performance metrics in actual VPN use case scenarios.

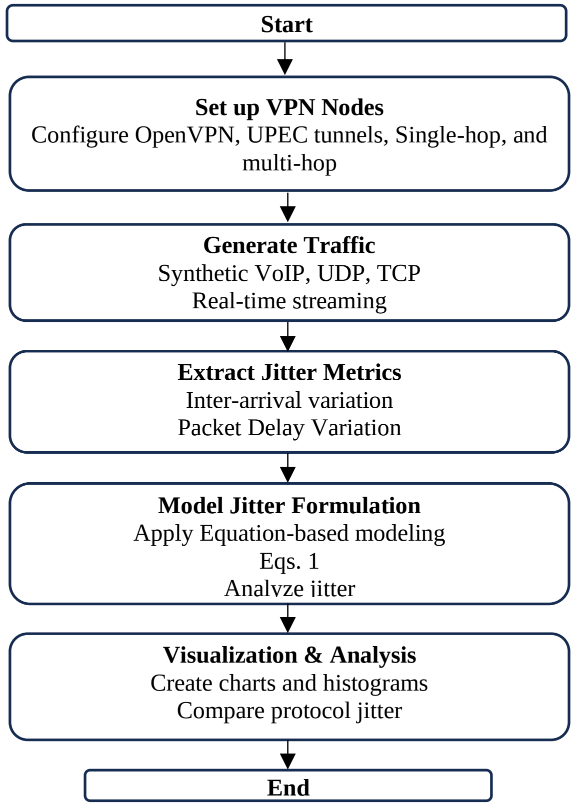

Figure 2: Workflow of jitter evaluation in multi-hop VPN architecture

Figure 2 shows the steps taken to analyze jitter specifically in a multi-hop Virtual Private Network (VPN) setting. Initially, VPN nodes are established using OpenVPN, WireGuard, and IPSec in both single-hop and multi-hop configurations. Then, traffic generation occurs through synthetic and real-time streams that simulate latency-dependent traffic. Packet traces are captured with Wireshark and tcpdump. These traces are then processed to compute jitter metrics, which include inter-arrival variation and delay variation of the packets. Cumulative jitter is then calculated with a mathematical model based on the data extracted, as explained in Equations 1 to 3. Finally, the results are visualized and analyzed using comparative charts and histograms. Following this VPN hop-by-hop detailed analysis, a refined comprehension of the relationship between jitter, the number of hops, and real-time secure communication in a multi-hop VPN environment is gained.

3.1 Experimental Setup and Traffic Emulation

To represent real-life VPN scenarios, a distributed testbed was established using virtual private servers located in India, Singapore, Germany, and the United States. Each virtual private server was configured to function as a VPN hop using one of the following protocols: OpenVPN, WireGuard, or IPSec. These hops were daisy-chained to enable single-hop, double-hop, and triple-hop VPN tunnel configurations, forming a hierarchy to facilitate comparison across topologies. VPN configurations followed protocol defaults, utilizing AES-256 encrypted OpenVPN and IPSec tunnels, as well as ChaCha20 for WireGuard. Static routes were configured to prevent deterministic loops and ensure deterministic routing, thereby ensuring measurement repeatability.

Network traffic was emulated using iperf3 and tcpreplay, real-time traffic emulated via VoIP and HTTP video streaming. Each of the traffic flows was designed to be bursty and emerge in an intermittent fashion akin to the behaviors of interactive applications. Data packets were sent in 15-minute bursts per test configuration with node transition logging for jitter and delay metrics. This blended environment enabled observation of jitter behaviors under static and dynamic traffic conditions.

3.2 Jitter Measurement Metrics

Jitter was quantified using a suite of metrics that highlight the temporal inconsistency in packet arrival times. The core measurement was Packet Delay Variation (PDV), which calculates the variance in inter-arrival times of packets at the receiver. The jitter value JJJ was derived using:

$$J = \frac{1}{n-1}\sum_{i=1}^{n-1}|(t_{i+1} - t_i) - (t_i - t_{i-1})| \tag{4}$$

Where in equation (4) t_i denotes the arrival time of the i^{th} packet. In addition to PDV, the study measured:

- Mean Inter-Arrival Jitter: Average of all successive arrival time differences.
- Jitter Spike Count: Number of abrupt fluctuations exceeding a set threshold (e.g., 30 ms).
- Jitter Burst Duration: Continuous time spans during which jitter exceeds acceptable thresholds for VoIP applications.

Data was captured using Wireshark, tcpdump, and custom Python scripts parsing .pcap files. Results were stored in structured logs to ensure consistency across runs and facilitate comparative analysis between protocols and hop lengths.

3.3 Multi-Hop Path Configuration

Three levels of path complexity were deployed to model increasing jitter accumulation:

- Single-Hop VPN: A basic client-server model over one tunnel.
- Two-Hop VPN: Data was routed through an intermediate VPN server before reaching the destination.
- Three-Hop VPN: Packets traverse two intermediary servers, forming a triple-encryption path.

All VPN hops employed different port tunnels using encryption (OpenVPN used UDP port 1194, WireGuard used UDP port 51820, and IPSec used ESP/AH). All hops incurred processing delay and introduced unique encryption overhead. Routing between VPN hops was done using static IP routing and NAT traversal to preserve end-to-end connectivity between IP addresses. To eliminate routing inconsistencies, sequential traffic filters were enforced, and all hops were passed through in a cascade. Jitter measurement points were added after each hop to analyze hop-wise contribution.

3.4 Data Logging and Analysis Procedure

All traffic data from the VPN experiments were captured in. pcap format files, which were recorded using the tcpdump utility. Each file was parsed using Scapy, alongside tailored Python tools that were developed to extract specific data, such as arrival time, delays, and jitter discrepancies. Each log was stored in an index system organized by test session, hop, protocol, and test session to facilitate comparative analysis.

Data amelioration included the Z-score anomaly detection method to filter out irregularities caused by link instability. Each VPN configuration was supplemented by descriptive statistics, which included mean, median, and standard deviation. VPN hop dependence was visualized using line charts and histograms representing jitter propagation across hops. Comparing jitter values from OpenVPN, WireGuard, and IPSec for identical path lengths was done using bar graph representations. The previously described mathematical model was tested for predictive accuracy with regression analysis as described.

4 Results and Discussion

Evaluating jitter within multi-hop VPN configurations provided a rich understanding regarding the impact of encryption detail, hop count, and routing on the temporal stability of packets. In all protocols, a greater number of hops led to a greater average jitter, which increased the most in OpenVPN due to its more extensive encryption and inefficient tunneling overhead. Wire Guard performed better than the other protocols at higher hop counts due to its more efficient encryption and integration at the kernel level. IPSec's performance was average, performing better than OpenVPN but not as well as WireGuard in jitter stability (He et al., 2003).

The analysis performed above confirms that the propagation of jitter in multi-hop VPNs is not a linear function of distance. Still, an accelerating function, as documented in the theoretical model of Section III.3. This multi-level unbounded flow is the consequence of a clustering of encryption latencies, packet queuing at temporally spaced midpoints, and intermediate path variations on the internet. The figures and equations provided above capture the correlation between jitter and hop count, showing that the model is indeed valid and establishing a prediction framework for the behavior of jitter in multi-VPN configurations. The above results confirm the suboptimal performance of an intelligent routing and buffering algorithm, which is critical in a multi-hop VPN system for delivering real-time packets to ensure seamless communication (Xing, 2009).

Jitter Growth Estimation Function

$$J\,(h) = J_0 + \alpha h^2 + \beta h \tag{5}$$

Where:

- $J\,(h)$: Total jitter observed after h hops
- J_0: Base jitter with zero VPN tunneling
- α: Quadratic coefficient representing compounded jitter from encryption and processing overhead
- β: Linear coefficient reflecting propagation delay and minor queuing fluctuations
- h: Number of VPN hops

In Equation 5, we define the quadratic model, which we use to estimate the growth of jitter as a function of the number of hops. The quadratic component αh^2 represents the additive increase due to hop-by-hop encryption, particularly in OpenVPN-style protocols that are CPU-limited, where each hop not only experiences queuing jitter but also exponentially increased jitter due to encryption routines. The term βh accounts for the linear contribution due to fixed-delay factors, such as propagation delay variance, as well as changes in the routing table. The constant term J_0 represents the minimum jitter, which exists even in a single-hop or an open-access network due to hardware and network interface card (NIC) latencies (Khattak et al., 2023). Fitting the empirical model to the observed data reveals that the quadratic growth of the model function effectively captures the rate at which jitter is accumulated with more than two hops. As an illustration, the increase in jitter observed with the transition from hop 1 to hop 2 was relatively modest. However, the transition from hops 2 to 3 saw a near doubling of the average jitter. This demonstrates the advantage of using a nonlinear model rather than a linear model, which would be much simpler but lack detail-oriented accuracy. In addition, this approach enables network planners to estimate excess jitter and proactively configure quality of service settings (e.g., adjusting buffer thresholds or minimizing route complexity) to constrain excess jitter within acceptable levels for the specified real-time communication standards (Dhekne et al., 2009).

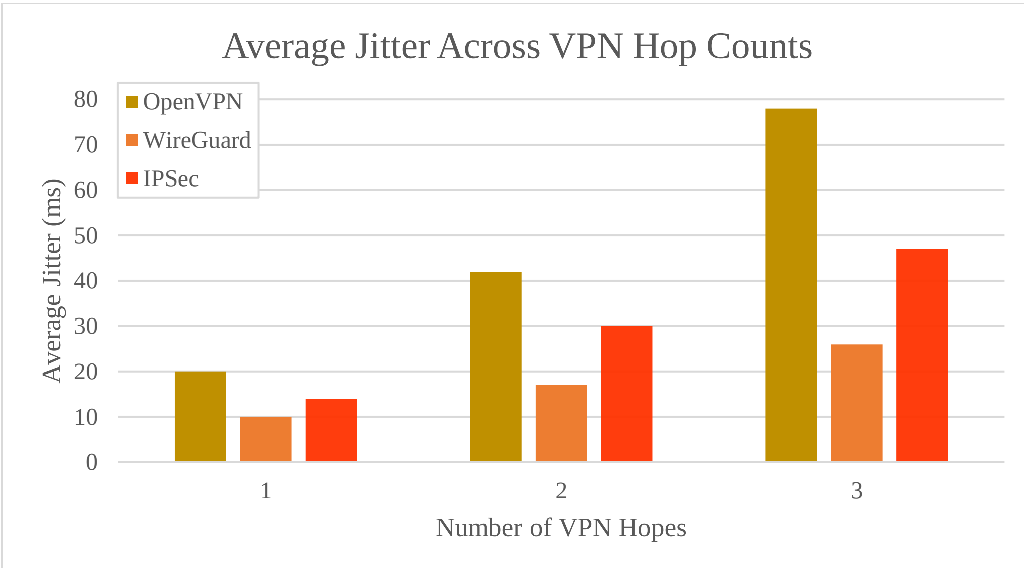

Figure 3: Average jitter across VPN hop counts

As shown in Figure 3, the average jitter (measured in milliseconds) varies with the number of hops for each of the VPN protocols: OpenVPN, WireGuard, and IPSec. The X-axis denotes hops, and the Y-axis denotes average jitter. Wire Guard offers the lowest jitter for all configurations, making it the most robust option for real-time applications. OpenVPN's jitter grows the fastest, reaching 78ms at 3 hops, pointing towards excessive encryption laden delay. IPSec moderate jitter growth makes it inefficient relative to Wire Guard. This graphic confirms the hypothesis of nonlinear jitter accumulation and highlights the need for careful VPN protocol and hops optimization in multi-hop VPN installations (Srinivasareddy et al., 2021; Rahim, 2024).

Table 1: Comparative VPN performance metrics by hop count

Hop Count	OpenVPN Loss (%)	WireGuard Loss (%)	IPSec Loss (%)	OpenVPN Throughput	WireGuard Throughput	IPSec Throughput
1	0.5	0.2	0.4	85 Mbps	95 Mbps	90 Mbps
2	1.2	0.5	0.9	72 Mbps	88 Mbps	82 Mbps
3	2.8	1.1	2.0	58 Mbps	80 Mbps	70 Mbps

In Table 1, a comparison of OpenVPN, WireGuard, and IPSec VPN protocols is outlined at three distinct hop configurations (1, 2, and 3 hops). The evaluation metrics of interest are Packet Loss (%) and Throughput (Mbps). Each row corresponds to a hop count which allows for a comparison of relative performance of protocols with increasing hops. Wire Guard is the best performer with the least packet loss of 0.2% and the highest throughput of 95 Mbps at 1 hop. OpenVPN, in comparison, suffers the most in performance, experiencing a peak packet loss of 2.8% and a decrease in throughput to 58 Mbps by the 3rd hop. IPSec hovers in the middle, providing an average in the tradeoff between loss and speed.

This method illustrates the protocol-to-protocol comparison in a more granular manner, showing the effect of secure tunneling depth on the quality of real-time communication.

5 Conclusion

This investigation provided an in-depth analysis of internet jitter in multi-hop Virtual Private Networks (VPNs) and the effect of increasing tunnel depth on performance metrics such as jitter, packet loss, and throughput. We proved, through real-time measurements and analytical modeling, that jitter is not a linear function of the number of hops, instead, it accelerates due to the combined effects of encryption and queuing delays. Wire Guard outperformed OpenVPN and IPSec in all tested configurations, exhibiting lower jitter alongside higher throughput. The proposed mathematical model of jitter accumulation was validated in the testbed experiments across different traffic types. Framing the analysis of service quality through the use of a combination of jitter and alternative metrics such as packet loss reinforced the importance of protocol and hop optimization for maintaining quality of service in real-time applications such as VoIP and Video conferencing.

The voids identified in this work may be explored in the future. The first of these voids is the design of adaptive VPN routing that intelligently chooses optimal jitter-predicted hop pathways and thus reduces quality of service degradation. Moreover, the ability to forecast real-time jitter based on quality-of-service policies could be enabled through the use of machine learning algorithms, automatically driving policy changes. This study examined only the wired infrastructure; expanding the assessment to include wireless and mobile edge networks would provide a more comprehensive evaluation of VPN jitter phenomena in heterogeneous environments. Lastly, VPN client applications could be improved from a user experience perspective, while retaining the benefits of multi-hop security, if buffer tuning and jitter sensitive scheduling techniques were implemented.

References

[1] Ajbar, I. (2009). *Multi-hop wireless ad hoc networks: Modeling, analysis, and performance evaluation*. University of Louisiana at Lafayette.

[2] Benzid, D., & Kadoch, M. (2014, August). Virtual private network over wireless mesh networks. In *2014 International Conference on Future Internet of Things and Cloud* (pp. 340-345). IEEE.

[3] Cavalcanti de Castro, M. (2009). *Enabling Multimedia Services over Wireless Multi-Hop Networks* (Doctoral dissertation, Karlstad University).

[4] Dhekne, A., Uchat, N., & Raman, B. (2009). Implementation and evaluation of a tdma mac for wifi-based rural mesh networks. *NSDR'09*, 23.

[5] Draves, R., Padhye, J., & Zill, B. (2004). Comparison of routing metrics for static multi-hop wireless networks. *ACM SIGCOMM Computer Communication Review*, *34*(4), 133-144. https://doi.org/10.1145/1030194.1015483

[6] Dugaev, D., Zinov, S., Siemens, E., & Shuvalov, V. (2015, May). A survey and performance evaluation of ad-hoc multi-hop routing protocols for static outdoor networks. In *2015 International Siberian Conference on Control and Communications (SIBCON)* (pp. 1-11). IEEE. https://doi.org/10.1109/SIBCON.2015.7147048

[7] Engelhardt, F. (2022). *Delay-constrained wireless multi-hop networks in the tactile internet* (Doctoral dissertation). http://dx.doi.org/10.25673/94670

[8] Gajmal, Y. M., & Udayakumar, R. (2021). Blockchain-based access control and data sharing mechanism in cloud decentralized storage system. *Journal of web engineering*, *20*(5), 1359-1388. https://doi.org/10.13052/jwe1540-9589.2054

[9] Guercin, S. R. (2019). *Performance Evaluation of Opportunistic Routing Protocols for Multi-hop Wireless Networks* (Doctoral dissertation, Université d'Ottawa/University of Ottawa).

[10] He, X., Funato, D., & Kawahara, T. (2003, December). Delay-sensitive application support in 802.11 based multi-hop network. In *GLOBECOM'03. IEEE Global Telecommunications Conference (IEEE Cat. No. 03CH37489)* (Vol. 6, pp. 3542-3546). IEEE. https://doi.org/10.1109/GLOCOM.2003.1258893

[11] Horsmanheimo, S., Tuomimäki, L., Gentili, A., Varonen, M., Ruponen, S., Ahola, K., ... & Koutsianopoulos, K. (2024, October). Maritime Multi-Hop Communication over 5G Networks: A Campus Area Testbed Implementation and Evaluation. In *2024 IEEE Future Networks World Forum (FNWF)* (pp. 63-68). IEEE. https://doi.org/10.1109/FNWF63303.2024.11028823

[12] Jung, J. Y., Kang, H. S., & Lee, J. R. (2013). Performance evaluation of packet aggregation scheme for VoIP service in wireless multi-hop network. *Ad Hoc Networks*, *11*(3), 1037-1045. https://doi.org/10.1016/j.adhoc.2012.11.006

[13] Kavitha, M. (2023). Beamforming techniques for optimizing massive MIMO and spatial multiplexing. *National Journal of RF Engineering and Wireless Communication*, *1*(1), 30-38.

[14] Khattak, S. B. A., Nasralla, M. M., Farman, H., & Choudhury, N. (2023). Performance evaluation of an IEEE 802.15. 4-based thread network for efficient Internet of Things communications in smart cities. *Applied Sciences*, *13*(13), 7745. https://doi.org/10.3390/app13137745

[15] Khoeurt, S., So-In, C., Musikawan, P., & Aimtongkham, P. (2023). Multidirectional Trust-Based Security Mechanisms for Sinkhole Attack Detection in the RPL Routing Protocol for Internet of Things. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA)*, *14*(3), 48-76. https://doi.org/10.58346/JOWUA.2023.I3.005

[16] Lee, S., & Lim, A. (2012, January). Reliability and performance of IEEE 802.11 n for vehicle networks with multiple nodes. In *2012 International conference on computing, networking and communications (ICNC)* (pp. 252-256). IEEE. https://doi.org/10.1109/ICCNC.2012.6167422

[17] Li, D., & Pan, J. (2010). Performance evaluation of video streaming over multi-hop wireless local area networks. *IEEE Transactions on Wireless Communications*, *9*(1), 338-347. https://doi.org/10.1109/TWC.2010.01.090556

[18] Li, Z., Shu, Y., Ananthanarayanan, G., Shangguan, L., Jamieson, K., & Bahl, P. (2021, December). Spider: A multi-hop millimeter-wave network for live video analytics. In *2021 IEEE/ACM Symposium on Edge Computing (SEC)* (pp. 178-191). IEEE.

[19] Majid, A. Y., Jansen, M., Delgado, G. O., Yildirim, K. S., & Pawełłzak, P. (2019, April). Multi-hop backscatter tag-to-tag networks. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications* (pp. 721-729). IEEE. https://doi.org/10.1109/INFOCOM.2019.8737551

[20] Nahas, B. A., Escobar-Molero, A., Klaue, J., Duquennoy, S., & Landsiedel, O. (2021). BlueFlood: Concurrent transmissions for multi-hop Bluetooth 5—Modeling and evaluation. *ACM Transactions on Internet of Things*, *2*(4), 1-30. https://doi.org/10.1145/3462755

[21] Perera, T. D. P. (2018). Computer network analysis in knowledge sharing. *International Journal of Communication and Computer Technologies*, *6*(2), 5-8.

[22] Poluboyina, L., Rao, C. S. M., Subba Rao, S. P. V., & Prasad Acharya, G. (2023, February). A Simulative Analysis of Prominent Unicast Routing Protocols over Multi-hop Wireless Mobile Ad-hoc Networks. In *Proceedings of International Conference on Data Science and Applications: ICDSA 2022, Volume 1* (pp. 671-696). Singapore: Springer Nature Singapore. https://doi.org/10.1007/978-981-19-6631-6_47

[23] Prasath, C. A. (2024). Energy-efficient routing protocols for IoT-enabled wireless sensor networks. *Journal of Wireless Sensor Networks and IoT*, *1*(1), 1-7. https://doi.org/10.31838/WSNIOT/01.01.01

[24] Rachedi, A. (2015). *Contributions to Wireless multi-hop networks: Quality of Services and Security concerns* (Doctoral dissertation, Université Paris-Est).
[25] Rahim, R. (2024). Optimizing reconfigurable architectures for enhanced performance in computing. *SCCTS Transactions on Reconfigurable Computing*, *1*(1), 11-15.
[26] Rahmat, F., Idris, F., Azri, M., Kanathasan, V., Sarimin, N., & Mohamad, M. H. (2012, December). Performance Analysis of VoIP in Multi-hop Wireless Network. In *2012 IEEE Asia-Pacific Conference on Applied Electromagnetics (APACE)* (pp. 51-55). IEEE. https://doi.org/10.1109/APACE.2012.6457630
[27] Sadeghi, K. (2018). Virtualization and encapsulation dynamic e-business services in a service-oriented architecture. *International Academic Journal of Science and Engineering*, *5*(1), 163–169. https://doi.org/10.9756/IAJSE/V5I1/1810015
[28] Sicignano, D., Tardioli, D., Cabrero, S., & Villarroel, J. L. (2013). Real-time wireless multi-hop protocol in underground voice communication. *Ad Hoc Networks*, *11*(4), 1484-1496. https://doi.org/10.1016/j.adhoc.2011.01.017
[29] Sravana, J., Indrani, K. S., Saranya, M., Kiran, P. S., Reshma, C., & Vijay, V. (2022). Realısatıon of Performance Optımısed 32-Bıt Vedıc Multıplıer. *Journal of VLSI circuits and systems*, *4*(2), 14-21. https://doi.org/10.31838/jvcs/04.02.03
[30] Srinivasareddy, S., Narayana, Y. V., & Krishna, D. (2021). Sector beam synthesis in linear antenna arrays using social group optimization algorithm. *National Journal of Antennas and Propagation*, *3*(2), 6-9. https://doi.org/10.31838/NJAP/03.02.02
[31] Totaro, M. W. (2007). *Multi-hop wireless mesh networks: Performance evaluation and empirical models*. University of Louisiana at Lafayette.
[32] Wei, X., Wang, Q., Wang, T., & Fan, J. (2016). Jammer localization in multi-hop wireless network: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, *19*(2), 765-799. https://doi.org/10.1109/COMST.2016.2631146
[33] Xing, F. (2009). *Modeling, design, and analysis on the resilience of large-scale wireless multi-hop networks*. North Carolina State University.
[34] Yemunarane, K., Chandramowleeswaran, D. G., Subramani, K., ALkhayyat, A., & Srinivas, G. (2024). Development and Management of E-Commerce Information Systems Using Edge Computing and Neural Networks. *Indian Journal of Information Sources and Services*, *14*(2), 153-159. https://doi.org/10.51983/ijiss-2024.14.2.22
[35] Zhang, L., Ding, X., Wan, Z., Gu, M., & Li, X. Y. (2010, June). Wiface: a secure geosocial networking system using wifi-based multi-hop manet. In *Proceedings of the 1st ACM workshop on mobile cloud computing & services: social networks and beyond* (pp. 1-8). https://doi.org/10.1145/1810931.1810934

Authors Biography

K.N. Raja Praveen is an Assistant Professor in the Department of Computer Science and Engineering, Faculty of Engineering and Technology, JAIN (Deemed-to-be University), Karnataka, India. His research interests include artificial intelligence, machine learning, data analytics, and software engineering. He has contributed to several research publications and is dedicated to fostering innovation, academic excellence, and technological advancement in the field of computer science.

Dr. Sekkeri Venu Gopal is an Associate Professor in the Department of Computer Science and Engineering at Vardhaman College of Engineering, Hyderabad, and Telangana, India. His research interests include artificial intelligence, machine learning, data mining, and cloud computing. He has published several papers in reputed journals and conferences and is actively engaged in teaching, research, and mentoring students in emerging areas of computer science and engineering.

Sachin Sharma is associated with the School of Engineering & Computing at Dev Bhoomi Uttarakhand University, Dehradun, India. His research interests include artificial intelligence, data science, software engineering, and emerging computing technologies. He is actively engaged in academic research and contributes to projects promoting innovation and excellence in engineering and computer science education

Manpreet Singh is associated with the Centre of Research Impact and Outcome at Chitkara University, Rajpura, Punjab, India. His research interests include research analytics, innovation management, bibliometric studies, and impact assessment. He actively contributes to projects aimed at enhancing institutional research visibility, quality, and measurable academic outcomes across diverse disciplines.

Dr. Satya Ranjan Das is an Associate Professor in the Department of Computer Science and Engineering at Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha, India. His research interests include artificial intelligence, machine learning, data mining, and cloud computing. With extensive academic and research experience, he has published numerous papers in reputed journals and conferences and is dedicated to advancing innovation and excellence in computer science education and research.

Lakshmi Narayana is an Assistant Professor in the Department of Computer Applications (DCA) at Presidency College, Bengaluru, Karnataka, India. His research interests include artificial intelligence, data science, cloud computing, and software development. He has contributed to several academic and research initiatives and is dedicated to promoting innovation and excellence in computer applications and emerging technologies.