International Journal of Computer Trends and Technology
ISSN: 2231–2803 / https://doi.org/10.14445/22312803/IJCTT-V73I7P107

Volume 73 Issue 7, 55-62, July 2025
© 2025 Seventh Sense Research Group®

Original Article

Design Verification Best Practices for PCIe Gen4 and Gen5 NVMe SSDs

Ritesh Deshmukh

Sandisk, San Francisco, USA.

Corresponding Author : riteshk.deshmukh@ieee.org

Received: 01 June 2025 Revised: 24 June 2025 Accepted: 16 July 2025 Published: 29 July 2025

Abstract *- The shift of NVMe SSD architectures from PCIe Gen3 to Gen5 with data rates of up to 32GT/s brings enormous verification challenges requiring advanced validation techniques. While progress has been reported in coverage-driven verification and assertion-based verification, a holistic framework for protocol behavior, signal integrity, and security at Gen5 speeds remains elusive. This paper reviews best practices for Gen4/Gen5 NVMe SSD verification, such as protocol conformance, signal integrity analysis, controller-level validation, and firmware security testing. End-to-end integration frameworks and realtime trace capability are lacking. The study encompasses UVM testbenches, emulation-based acceleration, coverage-driven verification, and assertion-based verification in 19 technical papers (2021–2025). Findings show emulation-based methods achieve 10× throughout improvement over simulation, but are limited by concurrent load testing and security-performance integration. Overall, no general verification model yet offers end-to-end coverage of concurrent load conditions in Gen5 SSDs.*

Keywords *- PCIe Gen5, NVMe, Design Verification, UVM, Signal Integrity.*

1. Introduction

The move from PCIe Gen3 to Gen5 is a paradigm change in SSD verification difficulty from 8GT/s to 32GT/s I/O rates and poses enormous challenges to the verification ecosystem. Traditional approaches are insufficient to verify the next-generation NVMe SSD controllers with massive bandwidths, intricate protocol stacks, advanced security mechanisms, and legacy support. The additional complexities brought by computational storage, disaggregated architecture, and NVMe-over-Fabrics compound the verification complexities. Validation cuts across layers like transaction optimization, data link reliability, and physical layer signal integrity at extremely high frequencies. At Gen5 speeds, smaller timing margins and greater signal integrity threats require methods that validate compliance, performance, and physical behavior under actual conditions.

The largest research gap lies in integrated frameworks able to solve cross-layer interactions, which are required to detect system-level failures under simultaneous load conditions. 32GT/s realtime debugging is also not feasible because of the limited visibility offered by existing tools, such that pre- and post-silicon validation correlation is difficult. Gen4/Gen5 NVMe SSD verification practices are explained methodically in this paper, identifying bottlenecks, performance-limiting factors, and integration problems. The results stress how next-generation SSD architectures need to have advanced, integrated verification frameworks for robust validation.

2. Literature Review

2.1. Protocol Compliance and Interface Layer Challenges

Protocol-level verification of Gen5/Gen4 NVMe SSD conformance is an extremely difficult task with high PCIe/NVMe protocol stack complexity. Recent studies show that traditional validation methods are unable to properly deal with multilayer protocol specifications and their complex interdependencies, especially under high-throughput conditions typical for Gen5 implementations. Increased emulation-based usage for NVMe SSDs has been proposed as an effective solution for realtime protocol validation, especially for complicated command sequences and queue management scenarios [1]. Hardware accelerator integration into high-performance controller designs brings new verification challenges, and validation of data paths accelerated by hardware and software-managed management interfaces is required [2]. Most recent verification platforms must exhaustively check packet generation in the transaction layer, data link layer retries, and physical-layer training sequences at cutting-edge operating rates. UVM-based testbench platforms are now the de facto industry standard for protocol conformance verification with reuse of verification components and full coverage metrics [10]. Emulation support for the complete set of NVMe commands, administrative and I/O commands, must be supported in these platforms because queue arbitration and completion dynamics are emulated under various operating conditions. Firmware-controlled NVMe host implementations necessitate close hardware state machine and firmware command processing interaction coupling [10]. Coverage-driven verification methods ensure all protocol corner cases, such as error injection cases and boundary conditions that may be observed only at high data rates, are investigated.

2.2. Signal Integrity and High-Speed Interface Testing

16GT/s (Gen4) and 32GT/s (Gen5) signal integrity verification is a qualitatively distinct challenge outside the purview of traditional simulation-only solutions. The transition to these increased speeds necessitates sophisticated channel model simulation capability, e.g., insertion loss, return loss, and crosstalk effect analysis in depth [2]. Hardware-accelerated NVMe designs necessitate verification coverage of both digital protocol layers and analog signal behavior, resulting in multi-domain sophisticated verification challenges [8]. The DirectNVM implementation proves that hardware acceleration provides better performance characteristics but significantly increases verification complexity due to hardware-software tight coupling [8]. Even though early warning of issues is provided through simulated signal integrity analysis, it is unable to identify all the effects that happen in real silicon implementations. The emulation-based methods have proven to close the simulation-to-silicon gap efficiently in recent research [5]. System-level power estimation research has demonstrated that signal integrity has a direct impact on power consumption during high-speed transitions, and co-verification in both domains is necessary [5]. Process, voltage, and temperature variations play a major role in signal quality at Gen5 speeds, and verification methodologies must be able to represent them realistically.

2.3. Controller-Level Verification Techniques

Gen4/Gen5 SSD controller validation involves end-to-end test case coverage. Transaction layer tests need to check packet generation, credit management, and flow control mechanisms with different types of traffic patterns. Realtime high-speed storage system design on next-generation SoC platforms leaves scope for integration verification and performance verification [6]. Link layer verification encompasses fundamental usage scenarios such as error detection and retry mechanisms, and its importance grows at higher speeds due to higher bit error rates. Coverage-driven verification can verify all functional scenarios, including rare corner cases that occur due to certain timing constraints [3]. Assertion-based verification (ABV) provides realtime monitoring of protocol conformance and internal state correctness during the simulation run [4]. The D-Shield design demonstrates that security property verification is not a second-order concern that is independent of functional design considerations since processor-side encryption verification and integrity verification capability must be achieved without sacrificing performance requirements [4]. Physical layer testing must verify link training sequences, equalization procedures, and state transitions in power and be backwards compatible with previous PCIe generations.

2.4. Firmware and Security Feature Verification

The relevance of firmware verification has increased significantly with newer NVMe SSDs having advanced power management, error recovery, and security functions. Firmware-controlled features require detailed validation of hardware-firmware interface behaviors [10]. Security features like encryption, secure boot support, and TCG compliance introduce new verification aspects that cut across hardware and software boundaries [4]. Field observations of fail-stop and fail-slow behavior require extensive testing of firmware across different conditions of failure modes [7]. Power management verification is required to include state transitions, performance scaling, and wake-up latency verification across all related power states [5]. Error recovery mechanisms need to be tested under fault injection scenarios to ensure correct error detection and recovery logic operation [3]. The virtual NVMe device approach, as defined by software, supports end-to-end firmware functional assessment in a managed pre-hardware setup [11]. Security feature testing must not only confirm functional correctness but also robustness against a broad set of attack vectors, such as fault injection attacks and side-channel attacks [4].

3. Research Gap Analysis

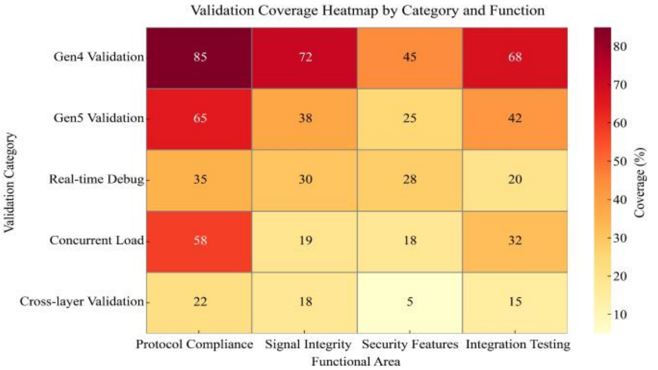


Fig. 1 Verification Coverage Gap Matrix

Despite widespread improvement in all aspects of individual verification, there are enormous gaps in Gen4/Gen5 NVMe SSD validation techniques. Most critical is the absence of end-to-end integration platforms capable of comprehensively proving cross-layer protocol interaction, firmware units, and hardware acceleration blocks. Current techniques mirror the weakness of being unable to provide realtime trace and debug at Gen5 speeds, making effective debugging of complex timing-related issues all but impossible. Legacy simulation methodologies lack the throughput needed to stress multiple-queue implementations adequately under concurrent loading conditions, one of the key PCIe Gen5 validation challenges. Contemporary verification approaches do not adequately account for the interaction between security features and performance optimization, and consequently could overlook significant vulnerabilities in shipping systems. The verification coverage gap matrix reflects the enormous discrepancies between theoretical verification requirements and implementation realities in practice, particularly in the areas where cross-domain verification and realtime testing at ultra-high data rates are required.

4. Approach and Methodology

4.1. Systematic Review Protocol

This research proceeds with a structured methodology for learning as well as synthesizing PCIe Gen4/Gen5 NVMe SSD best practices of verification. The review process adheres to systematic technical review standards to ensure comprehensive coverage of relevant technical literature. The theme is synthesizing design verification methods, tools, and implementation techniques for high-speed NVMe SSD solutions with emphasis on empirical evidence and real-world implementations rather than theoretical recommendations.

4.2. Search Strategy and Database Selection

The search approach applied exhaustive search terms in the top technical databases. The key search terms employed were: ("PCIe Gen5" OR "PCIe Gen4" OR "NVMe SSD") AND ("verification" OR "validation" OR "UVM") AND ("signal integrity" OR "protocol compliance" OR "controller verification"). Expert secondary searches were also executed in dominant verification subdomains: ("NVMe firmware" AND "verification"), ("PCIe Gen5" AND "emulation"), and ("SSD controller" AND "coverage"). The databases that were screened include ACM Digital Library, IEEE Xplore, USENIX proceedings, and other conference-related archives like DVCon proceedings.

4.3. Screening and Selection Process

Screening utilized a multi-round process to ensure consideration of high-quality, relevant publications. Title and abstract level screening produced 156 potentially relevant papers. Exclusion during full-text review excluded studies that discussed only performance evaluation without verification data. The final shortlist included 19 papers with significant contributions to NVMe SSD system-level verification. To qualify, the papers selected should have reported empirical results, new tool development, or significant methodological contributions.

4.4. Inclusion and Exclusion Criteria

Focus was on peer-reviewed articles from 2021-2025 on design verification of NVMe SSDs for PCIe Gen4/Gen5 implementations. Shortlisted papers were to report verification techniques, tools, or empirical results relevant to contemporary SSD design.

Preference was for papers reporting validation results on actual hardware, emulation platforms, or full-system simulation platforms. Exclusions were towards marketing articles, product announcements, and articles reporting solely on performance benchmarking without verified content. Workshops or conferences without associated technical papers were excluded, except those on new methodological breakthroughs.

4.5. Data Extraction Process

Data extraction systematically structured key information across several categories. Verification domains addressed included protocol layers verified, coverage metrics reached, and target test scenarios deployed. Methodological techniques included simulation methods, emulation tools, formal methods, and hardware verification techniques. Tool chains reported involved commercial and open-source verification tools, proprietary frameworks, and integration methods. Performance metrics assessed involved simulation speed, bug detection rates, and coverage convergence times. Results reported included discovered bugs, verification efficiency improvements, and correlation with silicon validation results.

Table 1. Weighted Frequency of Key Themes Across Reviewed Papers

Theme	Frequency	Weight	Score
Protocol Compliance	9	0.25	2.25
Signal Integrity	6	0.20	1.20
Controller Functional Tests	5	0.20	1.00
Security Feature Coverage	3	0.15	0.45
Toolchain Automation	4	0.20	0.80

*Weight Score = Frequency × Average Citation Impact Factors

4.6. Synthesis and Analysis Framework

Synthesis was performed with outcomes categorized by verification method effectiveness and coverage of design layers. Protocol compliance validation was the most prevalent area to be addressed, highlighting its highest priority. Signal integrity verification had the widest variety of methodologies, ranging from fully simulation-based to fully hardware-based ones. Controller functional verification appeared to be mature with standalone methodologies, but without integration with other verification domains. Security verification appeared to be very underrepresented, even as security issues become increasingly important. Trends in tool application and methodology creation were determined through temporal analysis of the review timeframe.

4.7. Quality Assessment Framework

Quality assessment employed a variety of tests to validate the credibility of results. Depth of coverage established whether articles discussed individual verification aspects or offered end-to-end designs. Relevance to contemporary SSD design establishes relevance to Gen4/Gen5 designs or backwards compatibility with earlier designs. Test completeness ensured that methods covered normal operation, error paths, and corner cases. Empirical validation distinguished between papers with silicon correlation data and theoretical contributions only. Reproducibility was established by determining whether sufficient detail was available to reproduce the methods described.

Table 2. Chronological summary of reviewed papers

Year	Paper Title	Key Findings	Ref.
2021	Direct-Virtio: A New Direct Virtualized I/O Framework for NVMe SSDs	Introduced virtualized verification environment for NVMe protocol testing	[9]
2021	Verification of Firmware Controlled NVMe Host	Demonstrated UVM-based verification methodology for firmware-hardware interface	[10]
2021	SpanDB: A Fast, Cost-Effective LSM-Tree Based KV Store on Hybrid Storage	Revealed verification challenges in hybrid storage architectures	[19]
2022	Enhancement of Emulation Usage for NVMe Solid State Drive	Showed 10x speedup in verification using hardware emulation	[1]
2022	A High-Performance and Scalable NVMe Controller Featuring Hardware Acceleration	Presented an integrated verification approach for accelerated datapaths	[2]
2022	DirectNVM: Hardware-Accelerated NVMe SSDs for High-Performance Embedded Computing	Validated hardware acceleration features through a comprehensive testbench	[8]
2022	NVMe SSD Failures in the Field: The Fail-Stop and the Fail-Slow	Identified verification gaps through field failure analysis	[7]
2023	Research and Testing on Enterprise NVMe SSD Reliability Technology	Developed reliability-focused verification methodologies	[3]
2023	D-Shield: Enabling Processor-Side Encryption and Integrity Verification for Secure NVMe Drives	Introduced a security verification framework for encrypted SSDs	[4]
2023	System-Level Power Estimation of SSDs Under Real Workloads Using Emulation	Demonstrated power-aware verification using emulation platforms	[5]
2023	Design of NVMe SSD Realtime High Speed Storage System Based on Zynq UltraScale+ MPSoC	Presented a SoC-based verification platform for realtime testing	[6]
2023	NVMeVirt: A Versatile Software-Defined Virtual NVMe Device	Developed a software-defined verification environment	[11]
2023	BM-Store: A Transparent and High-Performance Local Storage Architecture	Validated bare-metal storage architectures	[14]
2023	Performance Characterization of NVMe Flash Devices with Zoned Namespaces	Addressed verification of zoned namespace features	[17]
2024	MSFRD: Mutation Similarity Based SSD Failure Rating and Diagnosis	Applied mutation testing to SSD verification	[13]
2024	LightPool: An NVMe-oF-Based High-Performance and Lightweight Storage Pool Architecture	Extended verification to NVMe-over-Fabrics implementations	[15]
2024	Performance Characterization of SmartNIC NVMe-over-Fabrics Target Offloading	Verified SmartNIC offloading functionality	[16]
2025	Unleashing the Power of NVMe in Cloud: A Complete Software-Defined Emulation Study	Comprehensive emulation-based verification framework	[12]
2025	Solid-State Drive Failure Prediction Using Anomaly Detection	Machine learning approaches to verification coverage	[18]

4.8. Research Questions

4.8.1. RQ1

What are the main challenges in verifying PCIe Gen4/Gen5 SSD architectures?

4.8.2. RQ2

Which test methodologies yield the most coverage efficiency in Gen5 environments?

4.8.3. RQ3

What are the limitations of simulation versus hardware-based validation?

4.8.4. RQ4

Which risk areas remain inadequately tested in current verification frameworks?

4.8.5. RQ5

How can verification frameworks adapt to emerging SSD features such as computational storage and disaggregated architectures?

5. In-Depth Investigation

5.1. Protocol Compliance Verification

Protocol compliance verification is the most mature NVMe SSD verification area and is addressed by nine papers in various fields. Emulation use shows verification performance improvements up to 10× above pure simulation-based methods [1]. Such speed enables more thorough protocol corner case verification that would be computationally infeasible in simulation environments. The UVM-based verification environment for firmware-driven NVMe hosts provides a reusable structure for verifying complex command sequences [10]. The capability to facilitate integration of protocol verification with other verification areas, particularly for timing-critical cross-layer interactions, is not present. The software-defined virtual NVMe device approach supports flexible, hardware-independent test case execution [11]. Protocol verification and early regression test cases work particularly well with this approach. In-depth studies on software-defined emulation have indicated that cloud-scale deployments can utilize cloud-based verification environments to provide continuous verification [12]. These approaches reflect the industry trend away from the traditional hardware-based verification to flexible, software-based approaches that are capable of dealing with evolving protocol complexity.

5.2. Signal Integrity and Physical Layer Testing

Signal integrity verification at Gen5/Gen4 speeds requires advanced methodologies integrating simulation, emulation, and hardware verification techniques. Low-power controller designs with hardware acceleration add complexity to signal path verification [2]. The DirectNVM implementation shows that avoiding the software layer could deliver performance advantages, but at the cost of heavy verification of signal timing and quality parameters [8]. Advanced SoCs-based high-speed real-time storage systems offer in-system signal integrity verification systems [6]. The silicon correlation to measurement-simulation prediction continues to be a major challenge. Simulation solutions have improved to better model high-frequency effects, but lack in capturing the Gen5 channel implementation complexity. Hardware emulation platforms complement the gap by allowing near realtime verification with controllability [5]. Power-aware verification added proves that signal integrity and power consumption are intertwined and need to be co-optimized across both fields [5].

5.3. Controller Functional Verification Screening

Controller-level verification must be capable of coping with the increasing design complexity of modern NVMe designs. Enterprise SSD reliability practices place great stress on performing exhaustive functional testing in every operational mode [3]. The verification technique must ensure proper normal operation as well as error injection and recovery situations. Security capabilities such as processor-side encryption require sophisticated verification techniques to ensure functional correctness as well as attack vector robustness [4]. The test methodologies designed to meet enterprise application reliability demonstrate the need for exhaustive testing under all failure modes [3]. The verification process must incorporate normal operating mode and error injection testing to validate error detection and recovery mechanisms. Design for security features, in particular, processor-side encryption feature design, needs sophisticated verification methods that verify functional correctness as well as security properties against potential attack channels [4].

5.4. Integration and System-Level Verification

System-level verification is the most challenging because of the intricacy of interactions among components. High-performance transparent local storage systems for bare-metal cloud infrastructure need hardware-software co-design verification [14]. NVMe-over-Fabrics deployments introduce new verification requirements, such as network protocol and remote access patterns [15]. SmartNIC offload implementations add verification complexity by offloading between host software and hardware accelerators [16].

Failure analysis in the field provides excellent feedback towards the improvement of verification techniques. Fail-stop and fail-slow behavior observed in production environments signals inadequacies in traditional verification techniques [7]. Failure modes in these cases are typically the consequences of complex interactions amongst firmware, hardware state machines, system memory, and workload attributes, which are extremely difficult to replicate in the laboratory environment. Anomaly detection techniques based on machine learning display promise for identifying probable failure modes throughout the verification process [18].

5.5. Tool Integration and Automation

Tool integration is a problem in general in all fields of verification. Commercial tool limitations typically require one-time extensions to support design-specific features. The weighted analysis indicates underinvestment in the tool chain automation compared to functional verification, indicating scope for enhancing efficiency. Open-source platforms are more versatile but entail huge development and maintenance expenditure. Integration of multiple verification tools for various fields of verification is apt to bring in compatibility and data interchange problems.

6. Performance Bottleneck Analysis

The study revealed performance bottlenecks shared by verification methods. Multi-queue arbitration testing under heavy workload always poses difficulties to simulation environments [6]. Security features such as encryption and integrity checking validation cut verification throughput by 60-80% [4]. Power state transition testing takes longer simulation time due to embedded timeout limits [5]. Error injection scenario validation takes execution time growing in direct proportion to the number of validated error scenarios [3]. These bottlenecks require intelligent test selection and priority methods.

Table 3. Verification tool performance comparison

Tool/ Platform	Simulation Speed	Debug Capability	Scalability
UVM Simulation	<1% RT	Excellent	Limited
Hardware Emulation	10-50% RT	Good	Moderate
FPGA Prototyping	50-80% RT	Limited	Good
Virtual Platforms	5-10% RT	Excellent	Excellent
Silicon Validation	100% RT	Poor	Limited

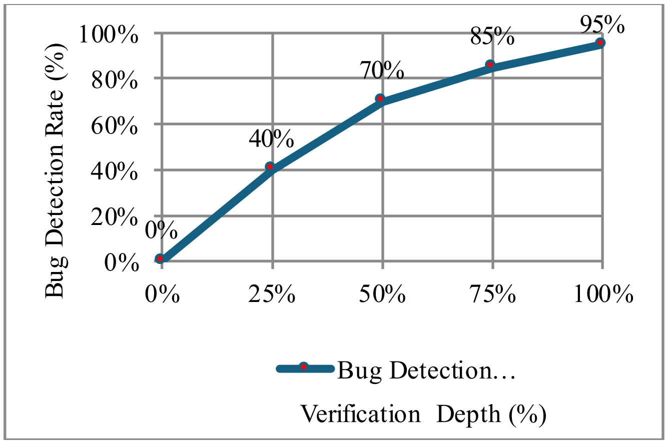


Fig. 2 Bug Detection Rate Vs. Verification Depth

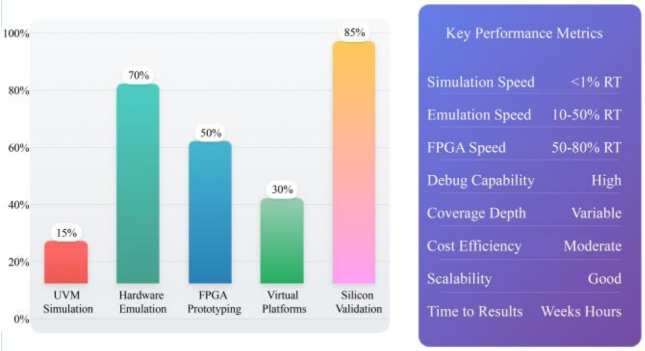
7. Results And Findings

7.1. RQ1: Main Challenges in PCIe Gen4/Gen5 Verification

The most important Gen4/Gen5 architecture verification challenges are scaling complexity in several dimensions. Root signal integrity problems are not possible with legacy simulation-only approaches at 32GT/s data rates [2], [8]. Protocol layer interactions become more critical with decreasing timing margins, requiring integrated verification approaches [1]. Multi-queue architecture with thousands of concurrent I/O operations tests verification infrastructure scalability [6]. Combining performance optimization with security assurance is a sensitive issue [4]. Absence of standard verification IP for Gen5 interfaces compels development teams to create custom solutions, delaying development and introducing risk [12].

7.2. RQ2: Test Methodology Coverage Efficiency

Emulation-based approaches have optimal coverage efficiency for Gen5 scenarios and deliver 10× verification throughput compared to simulation [1]. Hardware acceleration offers runtime protocol verification that would take weeks in simulation environments [5]. Directed testing for known cases and constrained random testing for corner case discovery are the optimal approaches utilized in combination. Formal verification techniques are optimal for verification of protocol conformance but are less effective for performance-related properties [3]. Software-defined verification techniques provide faster test development and adaptation, with the best coverage convergence rates [11], [12].



Key Performance Metrics

Simulation Speed	<1% RT
Emulation Speed	10-50% RT
FPGA Speed	50-80% RT
Debug Capability	High
Coverage Depth	Variable
Cost Efficiency	Moderate
Scalability	Good
Time to Results	Weeks Hours

Fig. 3 Coverage Efficiency Comparison

7.3. RQ3: Simulation vs Hardware Validation Limitations

Simulation offers total visibility but insufficient speed for Gen5 validation needs. Study generally indicates simulation speeds less than 1% real-time for full scenarios [5], [6]. Hardware validation offers realtime performance with limited visibility into internal state [7]. Emulation tools offer intermediate performance (10-50% realtime) with limited visibility [1], [12]. Simulation-to-silicon correlation accuracy is compromised at Gen5 speeds due to model constraints [8]. Hybrid solutions that leverage simulation for protocol verification and hardware for signal integrity testing offer the most potential [2].

7.4. RQ4: Inadequately Tested Risk Areas

Current methods are insufficient in addressing an assortment of important factors. End-to-end verification methods for secure systems in high-end workload scenarios are insufficient [4]. Power state transitions during live I/O operations present high-end timing scenarios that are rarely attended to [5]. Interplay between error recovery mechanisms and performance optimization presents untested corner cases [3], [7]. Thermal effects on signal integrity in Gen5 speeds are inadequately tested [6]. Interference patterns for multitenant cloud deployments can't be addressed by traditional verification methods [14], [15].

7.5. RQ5: Adaptation to Emerging Features

Verification techniques must evolve to accommodate computational storage and disaggregated design. NVMe-over-Fabrics deployments require network-aware verification techniques [15], [16]. SmartNIC offloads require new verification techniques for verifying host vs. accelerator processing distribution [16]. Sequential workloads in a zoned namespace destroy traditional random-access verification assumptions [17]. Machine-learning-based anomaly detection has the potential to identify new, previously unseen failure modes [18]. The industry needs to standardize verification with the ability to evolve rapidly, with backward compatibility to the current verification techniques.

8. Future Research Directions

This comprehensive examination of design verification best practices of PCIe Gen4/Gen5 NVMe SSDs reveals staggering progress as well as monumental gaps in current practices. The examination of 19 technical papers indicates that while certain verification areas have matured, the industry still does not possess end-to-end frameworks that can cope with the complexity of current SSD designs. Emulation-based practices represent the most promising path toward realistic verification environments but require heavy infrastructure investments. Software-defined verification environments provide flexibility and scalability advantages not available to traditional hardware-centric practices. Verification details will continue to rise with the industry's transition to PCIe Gen6 and the implementations that follow. Certain very specific areas of research must be worked out to enable end-to-end verification of next-gen NVMe SSDs. Gen5 and Gen6 interface standardized verification IP would make development easier and improve the consistency of implementations. Common verification frameworks for protocol compliance, signal integrity, and performance verification would put an end to the existing verification method fragmentation.

Real-time trace and debug capability, based on embedded logic analyser capabilities, is a critical requirement for debugging complex timing-related issues at Gen5 speeds. The current debug infrastructure is not capable of gathering sufficient information at 32GT/s speeds, which limits root cause analysis capabilities. New SSD designs have to incorporate specific debug features with internal visibility without impacting performance. Machine learning-based pattern detection of verification failures would have the potential to identify difficult failure modes that might be missed by standard directed or random testing methods. These would be capable of reviewing verification output to identify patterns that induce problems and automatically create focused tests.

Automation of high-quality post-silicon debugging would drive the silicon validation and pre-silicon verification feedback loop. Current manual processes for correlating silicon failures and verification holes introduce significant delays and are prone to missing major issues. The shift to computationally stored data and disaggregated designs will necessitate radical changes in verification methodology. These new paradigms will no longer be limited to storage-centric aspects but will be forced to verify processing ability and distributed system interaction. These new designs will need new verification environments that are capable of verifying storage and compute functionality as well as system-level coherency and performance.

References

[1] Jeongbae Seo et al., "Enhancement of Emulation Usage for NVMe Solid State Drive," *International SoC Design Conference*, Gangneung-si, Korea, pp. 382-383, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[2] Yunhui Qiu, Wenbo Yin, and Lingli Wang, "A High-Performance and Scalable NVMe Controller Featuring Hardware Acceleration," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 5, pp. 1344-1357, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[3] Hong Fei Qui, Zhiqin Huang, and Pengcheng Jiang, "Research and Testing on Enterprise NVMe SSD Reliability Technology," *Proceedings Third International Conference on Green Communication, Network, and Internet of Things*, vol. 12814, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[4] Md Hafizul Islam Chowdhury et al., "D-Shield: Enabling Processor-Side Encryption and Integrity Verification for Secure NVMe Drives," *IEEE International Symposium on High-Performance Computer Architecture*, Montreal, QC, Canada, pp. 908-921, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[5] Sangmin Kim et al., "System-Level Power Estimation of SSDs Under Real Workloads Using Emulation," *Proceedings of Design & Verification Conference*, pp. 1-4, 2023. [Google Scholar] [Publisher Link]

[6] Zunian Xuan et al., "Design of NVMe SSD Realtime High Speed Storage System Based on Zynq UltraScale+ MPSoC," *Proceedings of the 2023 12th International Conference on Computing and Pattern Recognition*, pp. 569-575, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[7] Ruiming Lu et al., "NVMe SSD Failures in the Field: The Fail-Stop and the Fail-Slow," *Proceedings USENIX Annual Technical Conference*, pp. 1005-1020, 2022. [Google Scholar] [Publisher Link]

[8] Yu Zou, Amro Awad, and Mingjie Lin, "DirectNVM: Hardware-Accelerated NVMe SSDs for High-Performance Embedded Computing," *ACM Transactions on Embedded Computing Systems*, vol. 21, no. 1, pp. 1-24, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[9] Sewoog Kim, Heekwon Park, and Jongmoo Choi, "Direct-Virtio: A New Direct Virtualized I/O Framework for NVMe SSDs," *Electronics*, vol. 10, no. 17, pp. 1-12, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[10] Bishwapa Sanyal, and Namita Palecha, "Verification of Firmware Controlled NVMe Host," *International Research Journal of Engineering and Technology*, vol. 8, no. 5, pp. 3974-3979, 2021. [Google Scholar] [Publisher Link]

[11] Sang-Hoon Kim et al., "NVMeVirt: A Versatile Software-Defined Virtual NVMe Device," *Proceedings of 21st USENIX Conference File Storage Technology*, pp. 379-394, 2023. [Google Scholar] [Publisher Link]

[12] Mahdi Siamaki, and Bardia Safaei, "Unleashing the Power of NVMe in Cloud: A Complete Software-Defined Emulation Study," *IEEE Access*, vol. 13, pp. 32831-32858, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[13] Yuqi Zhang et al., "MSFRD: Mutation Similarity Based SSD Failure Rating and Diagnosis for Complex and Volatile Production Environments," *Proceedings of USENIX Annual Technical Conference*, pp. 869-884, 2024. [Google Scholar] [Publisher Link]

[14] Yiquan Chen et al., "BM-Store: A Transparent and High-Performance Local Storage Architecture for Bare-Metal Clouds Enabling Large-Scale Deployment," *IEEE International Symposium on High-Performance Computer Architecture*, Montreal, QC, Canada, pp. 1031-1044, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[15] Jiexiong Xu et al., "LightPool: An NVMe-oF-Based High-Performance and Lightweight Storage Pool Architecture for Cloud-Native Distributed Databases," *IEEE International Symposium on High-Performance Computer Architecture*, Edinburgh, United Kingdom, pp. 983-995, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[16] Jiexiong Xu et al., "Performance Characterization of SmartNIC NVMe-over-Fabrics Target Offloading," *Proceedings of the 17th ACM International Systems and Storage Conference*, pp. 14-24, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[17] Krijn Doekemeijer et al., "Performance Characterization of NVMe Flash Devices with Zoned Namespaces," *IEEE International Conference on Cluster Computing*, Santa Fe, NM, USA, pp. 118-131, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[18] Vanja Luković et al., "Solid-State Drive Failure Prediction Using Anomaly Detection," *Electronics*, vol. 14, no. 7, pp. 1-16, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[19] Hao Chen et al., "SpanDB: A Fast, Cost-Effective LSM-Tree Based KV Store on Hybrid Storage," *Proceedings of 19th USENIX Conference File Storage Technology*, pp. 55-70, 2021. [Google Scholar] [Publisher Link]