Frontiers in Computer Science and Artificial Intelligence
DOI: 10.32996/fcsai
Journal Homepage: www.al-kindipublisher.com/index.php/fcsai

FCSAI
AL-KINDI CENTER FOR RESEARCH AND DEVELOPMENT

| RESEARCH ARTICLE

Security-Performance Analysis of Post-Quantum Blockchain for Smart Cities: An Adaptive Edge-Batched ML-DSA Framework

Ahmed Chalak Shakir
Dean, College of Computer Science and Information Technology, University of Kirkuk, Kirkuk, Iraq
Corresponding Author: Ahmed Chalak Shakir, **E-mail:** ahmedchalak2020@gmail.com

| ABSTRACT

Blockchain technologies are widely used to ensure the integrity and traceability of data in smart city services for the Internet of Things (IoT). To address this issue, switching to post-quantum cryptography increases network costs, a factor that is frequently ignored when assessing the strength of cryptographic methods. NIST-standardized ML-DSA signatures can range from 2,420 to 4,627 bytes, depending on the specific signature, which may seem large given the need to protect sensor records. In this study, the costs across the various layers are investigated, and an Adaptive Edge-batched ML-DSA platform-independent Quantum-Resistant Smart City Blockchain Traffic (QRSCTV) system is proposed. Supervised device-to-edge trust-domain routine telemetry is divided into Merkle batches and then committed to the blockchain with a single ML-DSA signature. Public device-level non-repudiation is not included in the batching process; it is performed only when necessary. A closed-form solution is developed for transaction expansion, normalized block capacity, constrained-link fragmentation, batching delay, and batch selection, accounting for the service-level agreement (SLA). A discrete-event simulation is offered for the model, with Poisson arrival rates varying from 10 to 1000 transactions per second and latency budgets ranging from 50 to 500 milliseconds, and the simulation is easy to reproduce. When the application transaction size is 256 bytes, direct ML-DSA-44 produces a 2,676-byte normalized transaction, allowing for 391 transactions within a 1-MiB budget. AEB-MLDSA uses 70.56 bytes per record and has a normalized capacity of 3210 transactions (that is, 97.1% fewer bytes of authentication per record than direct ML-DSA-44). Moreover, the usefulness of the simulation depends on the workload: if the traffic is light, then small batches or direct signing should be used; if it is telemetry, then a high-rate signature can be efficiently distributed within a bounded delay. The results show that the move from quantum to post-quantum blockchain in smart cities must be handled at both the network architecture and cryptographic levels.

| KEYWORDS

blockchain; smart cities; Internet of Things; post-quantum cryptography; ML-DSA; edge computing; network security; adaptive batching; crypto agility

|ARTICLE INFORMATION

ACCEPTED: 3 September 2026 **PUBLISHED:** 4 October 2026 **DOI:** 10.32996/fcsai.2026.5.10.5

1. Introduction

Smart city platforms are a collection of transportation, energy, public safety, environmental monitoring, utilities, health services and municipal information systems. The base traffic consists of a lot of small sensors and controllers transmitting short records with relatively high frequencies. The records may affect physical operations or subsequent audits, so blockchain has been extensively explored as a means to maintain a tamper-proof history and distribute trust, and in some architectures, edge computing is also part of this setup to allow for limited endpoints to offload validation and aggregation tasks to the local infrastructure rather than a remote cloud or ledger for each operation [1], [2]

The movement of post-quantum modifies the economics of this design. In FIPS 204, NIST has approved the Module-Lattice-Based Digital Signature Algorithm (ML-DSA). The three sets of parameters used are standardized and generate signatures of 2,420, 3,309 and 4,627 bytes, respectively [3]. These sizes are easily accommodated in many traditional application scenarios, but they are large

Copyright: © 2026 the Author(s). This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) 4.0 license (https://creativecommons.org/licenses/by/4.0/). Published by Al-Kindi Centre for Research and Development, London, United Kingdom.

for applications where the application record is only a few hundred bytes. These additional bytes impact not just ledger storage, but also ledger fragmentation, transmission airtime, buffer pressure, block occupancy, propagation traffic, and validator-side ledger data processing.

This effect is especially noticeable on low power links. RFC 4944 specifies an IEEE 802.15.4 frame with a 127-octet physical-layer maximum, and an example of security overhead that leaves 81 octets at the MAC layer [4].For that example, the signature of just one ML-DSA-44 is many times bigger than the payload being secured. Recent evaluations of quantum-safe blockchain and post-quantum networking come to the same conclusion: migration expenses, device limitations, protocol compatibility, scalability, and operational transition are pragmatic concerns, in addition to the fact that the cryptographic algorithms have been standardized [5], [6], [7].

A direct migration would provide a strong public attribution system, but also imply that all data related to smart cities is equally compelling and encodes equal latency. Therefore, there are differences between the semantics of a parking occupancy update, an air-quality sample, a billing event, an emergency alarm, and a remote actuator command with respect to their security. One of the points that the engineering question is about is whether a post-quantum signature should replace a classical signature. It's also where the signature should be placed, how much should be created, and where actual public non-repudiation of the records is necessary.

We consider a question in this study by Adaptive Edge-Batched ML-DSA (AEB-MLDSA). Routine telemetry is authenticated inside a managed device-to-edge trust domain, bundled into a Merkle batch, and expressed at the boundary of the blockchain via a single ML-DSA-signed commitment. Records that must be independently publicly attributed do not go through the batch, but use direct ML-DSA. The number of transactions in the batch varies depending on the transaction arrival rate and the allowed latency budget. The contribution is deliberately system-oriented; that is, no new lattice signature is introduced. Rather, the paper examines the location and granularity of a current standardized signature and calculates the ensuing network trade-offs. embedding it.

There are four contributions to the study. It first presents the idea of a cross-layer problem where the size of the cryptographic signature, the fragmentation of the ledger with weak links, and the number of bytes used for the ledger come together. Second, it defines a dual-path design to distinguish the routine: authenticated telemetry from records to be explicitly signed by their signature services, for post-quantum security. Thirdly, it builds an analytical model for the transaction expansion, the signature amortization, block capacity, the fragmentation, the batching delay, and the SLA aware batch selection. Finally, it validates the model with a uniform discrete-event simulation, separating its assumptions from its inputs, so that the study can be reproduced or re-parameterized in a different setting of the blockchain.

2. Literature Review

2.1 Post-Quantum Signatures and Blockchain Migration

Quantum-safe blockchain research has moved beyond the question of whether quantum computing matters to the more practical question of how migration can be carried out without destabilizing deployed systems. [8] in their survey post-quantum and quantum blockchain approaches and show that quantum threats exist for signatures, key management, wallet security, and protocol design. [7] highlights a quantum-safe blockchain framework and underlines implementation and adoption limitations. [6] extend the analysis to layers above the cryptographic primitive and suggest that the migration process has an impact on the validation of transactions, propagation, interoperability, governance, and performance of the system, rather than just the cryptographic primitive itself. A recent systematic review of the post-quantum security of IoT comes to the same conclusion and also highlights that implementation overhead and deployment constraints remain a research need [9].

Networking studies report closely related concerns. [5] identify larger keys and signatures, protocol compatibility, implementation overhead, and side-channel exposure as barriers to post-quantum deployment in Internet and cellular protocols. NIST guidance on crypto agility takes an operational view of the transition: systems should be able to replace algorithms or cryptographic configurations without unacceptable disruption [10]. AEB-MLDSA follows this principle by treating signing mode, parameter set, batch cap, and timeout as configurable controls rather than fixed protocol constants. The publication of RFC 9964, which specifies ML-DSA for JOSE and COSE, further shows that post-quantum signatures are moving into concrete network serialization and transport profiles rather than remaining purely cryptographic building blocks [11]. Table 1 illustrates the ML-DSA parameter sets beside the corresponding public-key and signature sizes.

Table 1.FIPS 204 ML-DSA signature-size parameters *[11]*.

ML-DSA parameter set	NIST security category	Public key (bytes)	Signature (bytes)
ML-DSA-44	2	1,312	2,420
ML-DSA-65	3	1,952	3,309
ML-DSA-87	5	2,592	4,627

2.2 Edge and Blockchain Security in Smart Cities

Edge computing and blockchain are commonly combined in smart-city research because the two technologies solve different parts of the same systems problem. The ledger provides a shared integrity and audit layer, while the edge reduces dependence on remote cloud infrastructure and keeps computationally expensive operations closer to constrained devices. Bandwidth, latency, energy consumption, and security are coupled design goals for blockchain-edge IoT systems [1]. In the same way, [2] propose individual processing on top of an overlay blockchain in a lightweight smart-city structure; on the other hand, [12], [13], [14] explore applications of blockchain in securing smart city networks by preserving privacy and enabling trust.

This hierarchy creates an attractive migration point for post-quantum cryptography, but it also introduces a trust decision. When an edge gateway signs a batch root, the blockchain receives quantum-resistant evidence of what the gateway attested. It does not receive a public signature from every individual device. This distinction matters for safety-critical control, financial settlement, legal evidence, and cross-domain accountability. Any batching mechanism must therefore state explicitly when gateway attestation is acceptable and when direct device signing remains necessary.

2.3 Recent Post-Quantum IoT and Blockchain Systems

Post-quantum blockchain for IoT is already an active research area. [15] combine post-quantum authentication with consortium blockchain for cross-domain IoT procedures. [16] constructs a scalable lattice-based non-interactive multi-signature design for IoT-enabled blockchain systems, while [17] combine a post-quantum-secure blockchain into a dynamic IoT intrusion-detection architecture. Other studies by, [18] investigate a post-quantum blockchain design for smart-grid SCADA resilience, while [19] combine post-quantum blockchain with a learning framework for smart IoT systems.

These studies establish the feasibility and relevance of quantum-resistant IoT blockchains, but their main questions differ from the one considered here. They focus on authentication protocols, multi-signature design, intrusion detection, or application-specific architectures. The narrower gap addressed by AEB-MLDSA is network-centric: it quantifies how the byte footprint of standardized ML-DSA signatures affects small smart-city records and constrained links, and then adapts the granularity of signing to the observed traffic rate and latency requirement.

3. Methodology

3.1 System Model, Threat Model, and Design Goals

The modeled system has four rational layers: constrained smart-city IoT devices, a reliable edge security gateways, a blockchain network, and applications that consume ledger records. Devices produce an unidentified application body of size S_u bytes. The device-to-edge trajectory may use a specific technology, for example IEEE 802.15.4 (Montenegro et al., 2007); however, the gateway-to-blockchain network is expected to have higher capacity; the gateway will validate authentication and freshest details, constructs Merkle leaves from accepted records, merges leave into batches, and provides an ML-DSA signature on the batch commitment (National Institute of Standards and Technology [NIST], 2024). The blockchain contain either an edge-attested batch record and a direct device-signed transaction with adequate information to verify membership. As shown in Figure1. the proposed AEB-MLDSA and architecture of the two security paths.

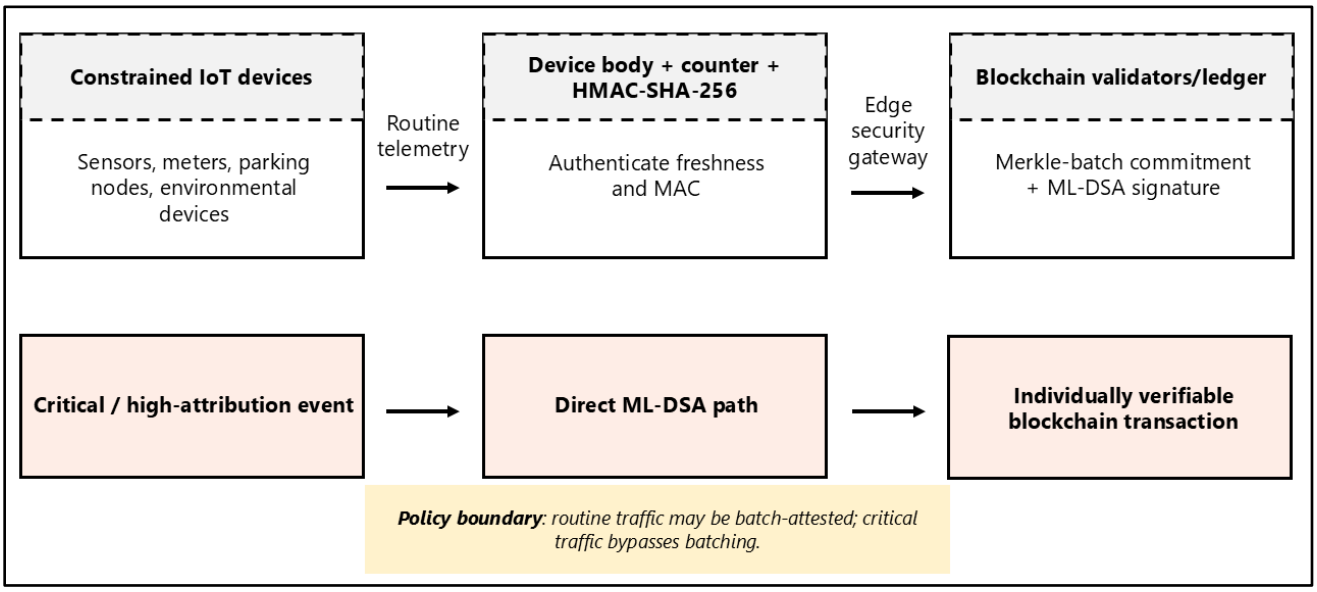


Figure 1.Architecture of the proposed AEB-MLDSA framework and its two security paths.

The model is deliberately platform-independent. It does not assume Ethereum gas accounting, Hyperledger Fabric endorsement policies, a particular consensus algorithm, or a fixed block interval. Instead, an adjustable byte budget S_b is used to study capacity, keeping the examination focused on communication and storage pressure and avoiding presenting a platform-independent calculation as measured blockchain throughput.

The attacker has the capability to eavesdrop on network traffic, inject forged records, modify packets, inspect blockchain data, and replay previously observed messages. A future adversary is expected to have quantum capability sufficient to threaten conventional public-key signatures [20], and the model does not consider that the intruder may attack ML-DSA, SHA-256, or HMAC-SHA-256. Edge gateways are trustworthy in batched mode; gateway compromise is therefore treated as a residual risk rather than concealed by the architecture itself.

Routine telemetry contains a newness value and a 32-byte HMAC-SHA-256 tag on the local link; this offers efficient authenticity as well as integrity within the managed edge domain; however, it fails to provide public third-party non-repudiation since the gateway also knows the shared secret [21]. Whenever independent public attribution is needed, the transaction takes the direct ML-DSA flow [3]. The framework is developed to maintain post-quantum integrity across the blockchain boundary by decreasing the communication overhead for high-rate telemetry, checking that batching delay is within an SLA, preserving direct signing for high-criticality activities, and maintaining crypto-agility as standards and device capacities evolve.

3.2 Proposed AEB-MLDSA Framework

AEB-MLDSA utilizes two different security paths. In the batched path, the device produces its application record, adds a counter or timestamp and an HMAC-SHA-256 tag, and transfers the record to the authorized edge gateway. After verification, the gateway hashes the approved record into a Merkle leaf. A group of leaves is collected into a Merkle tree, and the gateway signs the 32-byte root with ML-DSA. The signed batch commitment is then submitted to the blockchain. In the direct path, an essential transaction is signed with ML-DSA without waiting for aggregation. Direct signing is selected when public device-level non-repudiation is needed, the gateway trust domain is restricted, or batching would provide minimal benefit due to low traffic.

Let $C \in [0,1]$ denote normalized data criticality, $N \in \{0,1\}$ represent whether public device-level non-repudiation is required, $G \in \{0,1\}$ represent gateway trust availability, and $Q \in [0,1]$ denote a policy-defined quantum-migration priority. A simple policy score is written as:

$$R = w_c C + w_q Q + w_n N, \quad \text{where } w_c + w_q + w_n = 1 \quad (1)$$

The score is an input defined by the administrator rather than a standard security metric. The operational mode is then selected by the following rule:

$$\text{Mode} = \text{direct ML-DSA if } N = 1,\ G = 0, \text{ or } R \geq \tau; \text{ otherwise edge-batched ML-DSA} \quad (2)$$

This rule forbids a performance improvement from silently compromising the attribution semantics of a critical record. Municipal organizations may utilize distinct weights or thresholds when applying the same framework. Figure 2. shows the flow of transactions that results from signing directly with ML-DSA to implementing SLA-aware adaptive batching.

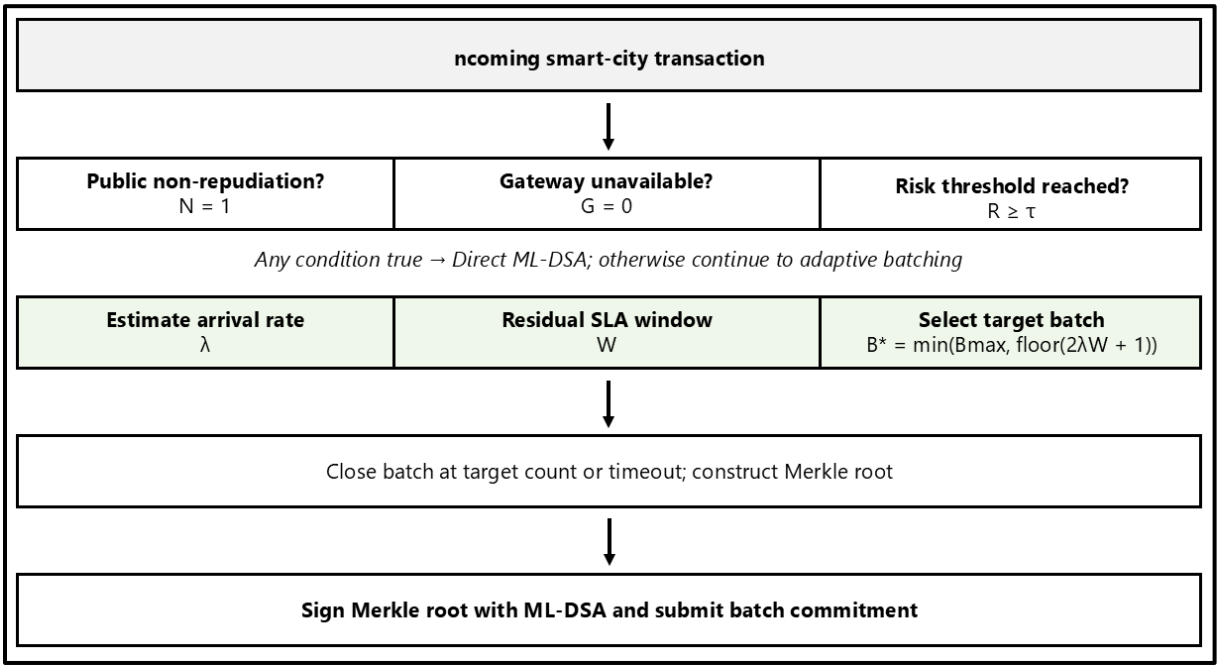


Figure 2.Transaction-handling and SLA-aware batch-selection flow for AEB-MLDSA.

3.3 Analytical Communication Model

As a post-transaction that is properly signed, the normalizing byte size includes the unsigned application body plus the signature:

$$S_{direct} = S_u + S_{sig} \quad (3)$$

Within the batched path, all local records retain a 32-byte HMAC tag. An individual ML-DSA signature, a 32-byte Merkle root, and M bytes of batch metadata are disseminated via B records. The compounded authentication overhead per transaction is therefore as follows:

$$O_{AEB}(B) = S_{HMAC} + (S_{sig} + S_{root} + M) / B \quad (4)$$

The transaction-equivalent byte cost utilized for normalized block-capacity contrast becomes:

$$S_{AEB}(B) = S_u + O_{AEB}(B) \quad (5)$$

For a block-byte budget S_b, the number of normalized transactions that are possible to be fit within the budget is:

$$N_{block} = \text{floor}(S_b / S_{tx}) \quad (6)$$

For a restricted link with L utilizable bytes per protected frame, the approximate number of segments is:

$$F = \text{ceil}(S_{link} / L) \quad (7)$$

An evaluation leverages L = 81 bytes as the secured payload example provided in RFC 4944 [4]. It offers a comparative baseline compared to a universal IEEE 802.15.4 payload size; a deployment evaluation needs to supplement the exact adaptation-layer and security overhead expenses of the targeted stack.

3.4 SLA-Aware Adaptive Batching

Batching Batching reduces signature bytes and produces waiting time. For a consistent arrival rate λ and an ideal fixed batch of B transactions, the mean waiting time can be measured as:

$$D_{batch} \approx (B - 1) / (2\lambda) \quad (8)$$

If L_{SLA} represents the end-to-end latency and L_0 indicates the non-batching processing and transport delay, the residual batching window is $W = \max(L_{SLA} - L_0, 0)$. Fixing the preceding approximation for B gives the expected batch size:

$$B^* = \min(B_{max}, \text{floor}(2\lambda W + 1)) \quad (9)$$

The execution process also applies a hard timeout equal to W from the first record in the batch. A batch therefore ends either when B^* records have arrived or whenever the timeout has expired. This detail is essential under low traffic due to the observed batch can be much smaller than the analytical goal. A reduction in authentication bytes with respect to direct signing is measured as:

$$\text{Reduction}(\%) = 100 \times (S_{sig} - O_{AEB}(B)) / S_{sig} \quad (10)$$

3.5 Evaluation Design and Reproducibility

The evaluation integrates deterministic analytical computation alongside a discrete-event simulation. Consistent parameters are retained separately with modeling assumptions; the parameter values for the ML-DSA signature sizes were extracted from FIPS 204 [3], whereas its constrained-link reference was taken from RFC 4944 [4]. The preset unsigned transaction is defined as 256 bytes, with the sensitivity measured at 128, 512, 1,024, and with 2,048 bytes; where the normalized block-byte limit is 1 MiB, while the batched mode uses a 32-byte HMAC-SHA-256 tag, a 32-byte Merkle root, and 16 bytes of batch metadata. ML-DSA-44 is utilized as the standard post-quantum signature since it has the smallest signature compared to the other three standardized ML-DSA parameter sets. The default/range values of the evaluation parameters are illustrated in Table 2.

Table 2. Evaluation parameters.

Parameter	Default/range
Unsigned transaction S_u	256 B; sensitivity: 128-2,048 B
Block-byte budget S_b	1 MiB
ML-DSA signatures	2,420 / 3,309 / 4,627 B
HMAC tag	32 B
Merkle root	32 B
Batch metadata M	16 B
Batch cap B_max	256
Arrival-rate range	10-1,000 tx/s
Latency budgets	50, 100, 250, 500 ms
Non-batch latency L_0	10 ms
Monte Carlo arrivals	50,000 transactions per scenario
Random seed	20260913

For every combination of arrival rate and latency budget, 50,000 arrivals are obtained from exponentially distributed inter-arrival times, resulting in a process called a Poisson. The randomized seed is set to 20260913, and the target B* is measured using the Equation (9); the simulated batch ends when the target count is gained, or the remaining SLA window terminates. The simulation stores the per-transaction waiting time and the realized batch size, while the authentication overhead is calculated depending on the realized mean batch over the target. The study provides normalized transactions, authentication bytes per 1-MiB byte budget, estimates of batching delay, and constrained-link fragments. It does not state CPU time, consensus throughput, gas cost, or energy measurements, as those quantities demand a particular blockchain configuration and hardware test environment.

4. Results and Findings

4.1 Direct Post-Quantum Signature Expansion

The default 256-byte example makes the byte-level overhead of direct post-quantum signing directly observable. A simple 64-byte classical-signature reference generates a 320-byte standardized transaction. However, Direct ML-DSA-44 extends the same application record to 2,676 bytes, ML-DSA-65 to 3,565 bytes, while ML-DSA-87 to 4,883 bytes within a 1-MiB byte budget; the corresponding standardized capacities are 391, 294, and 214 transactions. Conversely, the AEB-MLDSA with a B = 64 results in an amortized authentication interval of 70.56 bytes and generates a normalized transaction of 326.56 bytes. Table 3. below shows detailed comparisons of transaction size, normalized capacity, and fragmentation counts.

Table 3. Default 256-byte transaction comparison.

Mode	Auth. B	Total B	Tx / 1 MiB	Fragments
Classical 64-B reference	64.00	320.00	3,276	4
ML-DSA-44	2,420.00	2,676.00	391	34
ML-DSA-65	3,309.00	3,565.00	294	45
ML-DSA-87	4,627.00	4,883.00	214	61
ML-DSA-44 + classical	2,484.00	2,740.00	382	34
AEB, B=64	70.56	326.56	3,210	4

Table 3. demonstrates the substantial growth in transaction size presented by direct ML-DSA signatures, with overhead expanding across ML-DSA-44, ML-DSA-65, and ML-DSA-87. Conversely, the AEB-MLDSA configuration with B = 64 reduces the amortized

authentication cost to 70.56 bytes while preserving the transaction size near the classical reference. Figure 3. below shows the corresponding authentication overhead per normalized transaction.

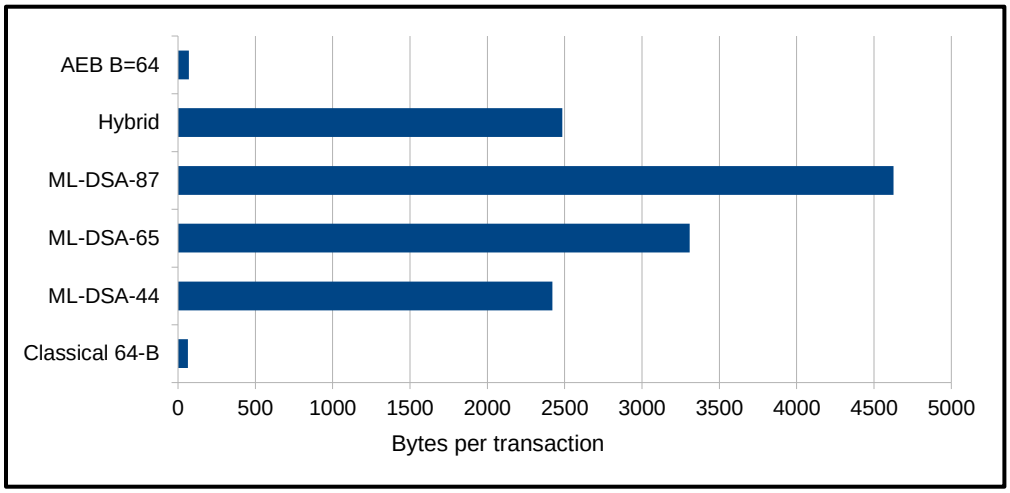


Figure 3. Authentication overhead per normalized transaction. The AEB value uses a 64-transaction batch and one ML-DSA-44 signature per batch.

4.2 Block-Capacity Sensitivity to Transaction Size

The post-quantum penalty becomes more severe when the application load is minimal, as the signature is a standard-size field. For an unsigned transaction size of 128 bytes, a 1-MiB cost carries 5,461 classical-reference transactions, 411 direct ML-DSA-44 transactions, and 5,280 AEB-MLDSA normalized transactions at B = 64. At 2,048 bytes, the same budget carries 496 classical-reference transactions, 494 AEB transactions, and 234 direct ML-DSA-44 transactions. The relative benefit of batching therefore lowers as the application payload grows, but it stays most robust for the small, repetitive records common in smart-city telemetry. Figure 4. shows the interpretation in normalized block capacity with the size of the unsigned transaction across the modes under evaluation.

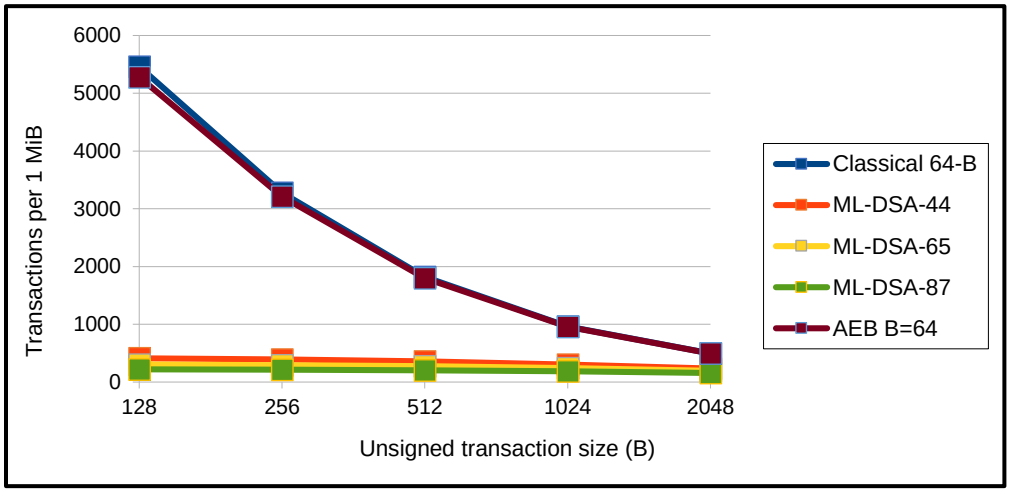


Figure 4. Normalized block capacity as a function of unsigned application-transaction size.

4.3 Constrained-Link Fragmentation Pressure

The fragmentation increases variation at the access link. Utilizing the 81-byte secured-payload instance with RFC 4944, a 256-byte application record needs about 34 fragments when it carries a direct ML-DSA-44 signature, 45 fragments with ML-DSA-65, and 61 fragments with ML-DSA-87. The AEB device-to-edge ink delivers the 256-byte application body along with a 32-byte HMAC tag, which requires approximately four fragments. At 128 bytes, the comparison includes two fragments for the AEB path versus 32 for direct ML-DSA-44. These calculations are deliberately approximate since real stacks add headers and may employ compression, while the order-of-magnitude impact remains clear. The typical fragmentation pressures for the transaction sizes and security modes considered are shown in Figure 5.

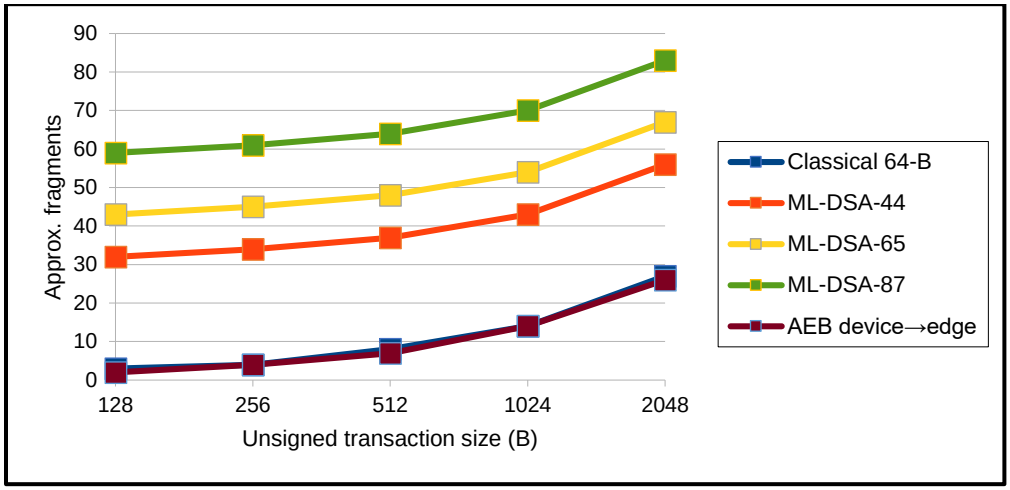


Figure 5. Approximate secured IEEE 802.15.4 fragmentation pressure for direct signatures and the AEB device-to-edge path.

4.4 Batch-Size Trade-off

The batch size is the main control variable in the suggested design; along with ML-DSA-44, the Merkle root, the signature, and 16-byte metadata field contribute 2,468 shared bytes for each batch. When B = 8, the amortized authentication cost is 340.50 bytes per transaction; at B = 16, it is 186.25 bytes; at B = 32, it is 109.13 bytes; at B = 64, it is 70.56 bytes; and at B = 128, it equals 51.28 bytes. The curve shows diminishing returns as the 32-byte per-device HMAC term is never amortized. As shown in Figure 6. the authentication overhead decreases as the batch size grows.

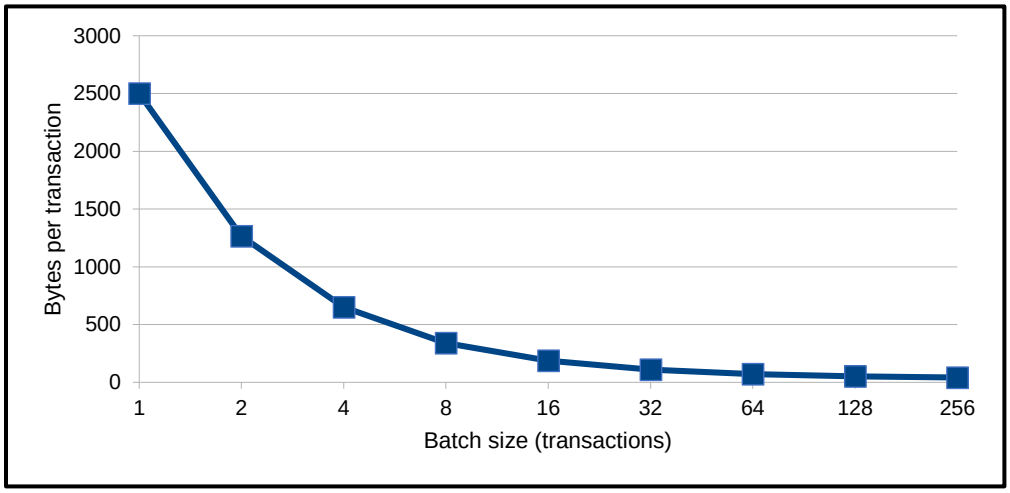


Figure 6. Amortized authentication overhead versus batch size for ML-DSA-44.

For the standard B = 64 case, the authentication-byte reduction compared to direct ML-DSA-44 is 97.1%. Normalized block capacity increases from 391 to 3,210 transactions, indicating an 8.21-fold increase. In contrast to the 64-byte classical footprint concern, AEB-MLDSA maintains around 98% of normalized capacity in this case. This comparison refers to byte capacity; it cannot be read as a claim of identical trust semantics, as batched mode depends on gateway attestation for routine telemetry.

4.5 Adaptive Batching Under Latency Constraints

The analytical controller raises the target batch while the traffic rate or the allowed delay budget increases. At 100 tx/s with a 100-ms SLA, Equation (9) presents a target batch of 19 transactions. At 1,000 tx/s under the exact SLA, the target rises to 181. The timeout remains critical as low-rate streams usually fail to attain the target before their latency budget ends. Figure 7. shows the relationship between the transaction arrival rate, the latency budget, and the resulting target batch size.

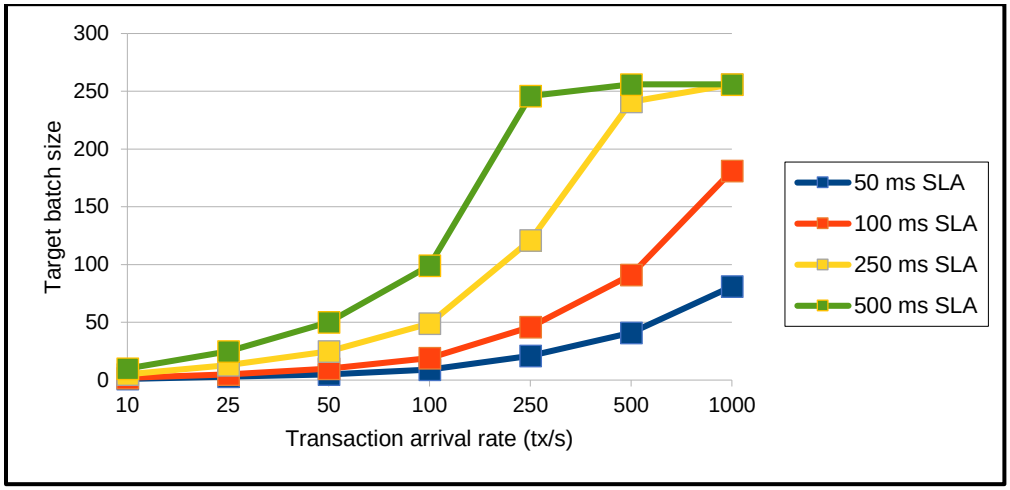


Figure 7. Analytical SLA-aware target batch size over the evaluated transaction-arrival rates.

The discrete-event simulation indicates that the real batch can be materially smaller than the target. With λ = 100 tx/s and a 100-ms SLA, the target value is 19, but the mean discovered batch is about 10.0 because the 90-ms residual timeout frequently closes the batch first. The mean batching delay is 49.5 ms, where the mean authentication overhead is about 278.5 bytes. At λ = 1,000 tx/s within the same SLA, the mean realized batch is about 90.4 records, the mean delay is 45.7 ms, and the mean overhead is approximately 59.3 bytes. Given a 250-ms SLA at 1,000 tx/s, the realized mean batch reaches about 238 records and mean authentication overhead falls to roughly 42.4 bytes. The selected simulation results are summarized in Table 4.

Table 4. Selected discrete-event simulation results.

λ (tx/s)	SLA	Mean batch	Mean wait	Mean auth.
10	100 ms	1.59	37.4 ms	1581.9 B
100	100 ms	10.01	49.5 ms	278.5 B
1,000	100 ms	90.42	45.7 ms	59.3 B
1,000	250 ms	238.10	119.9 ms	42.4 B

Figure 8. also shows the simulated average authentication overhead across various transaction arrival rates and SLA requirements.

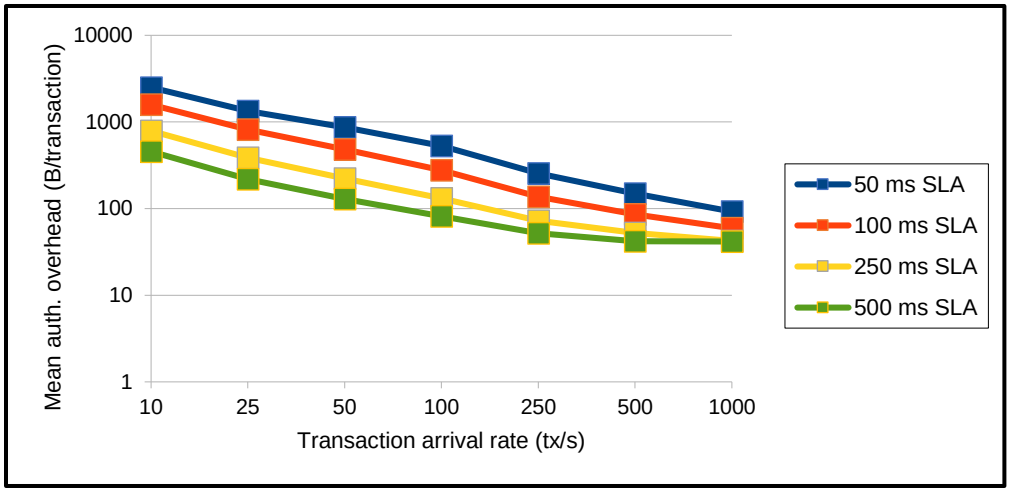


Figure 8. Simulated mean authentication overhead under SLA-constrained batching. The vertical axis uses a logarithmic scale to show both sparse- and high-rate regimes clearly.

The mean batching delay under the same arrival rate and SLA conditions is shown in Figure 9.

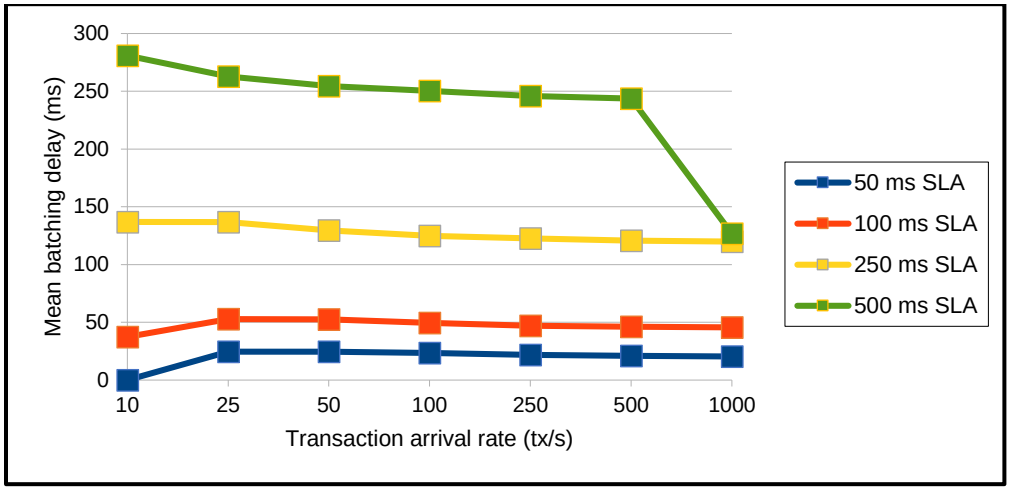


Figure 9. Simulated mean batching delay. The timeout keeps waiting time bounded when the target batch is not reached.

4.6 Security and Performance Discussion

Regarding blockchain limitations, each AEB batch is secured by an ML-DSA signature, while the Merkle root secures the selection transaction set: modifying any committed leaf alters the root and eliminates the gateway signature. The freshest field is checked before the leaf is produced, enabling rejection of simple replay attempts before they are preserved in the permanent ledger. The HMAC tag safeguards device-to-edge integrity and authenticity under the shared-key assumption.

The attribution presents the key security trade-off. A direct ML-DSA signature can be confirmed against a certain device public key, enabling public non-repudiation. A symmetric MAC cannot preserve this property as the verifier also knows the authentication secret. Therefore, the AEB-MLDSA enables verified, gateway-authenticated device provenance in batched mode rather than independent, device-level public non-repudiation. Records utilized for financial settlement, legal evidence, privileged administration, safety-critical actuation, or cross-domain approval should typically utilize direct ML-DSA.

The gateway therefore serves as a critical component. A hacked gateway may verify fraudulent batch content until its key is invalidated. Practical deployments ought to secure the gateway private keys in hardware-backed storage where possible, applying secure or measured boot, retaining audit logs, rotating keys by epoch, and supporting rapid revocation. The authentication keys for devices should be unique for each device or derived from device-specific secrets. Systems with significant assurance can use the same batching principle for threshold or multi-gateway verification, although the extension lies outside the domain of the present evaluation.

The results also illustrate why the framework is adjustable instead of simply batched. At 10 tx/s and a 100-ms SLA, the simulated mean batch size is only 1.59 transactions, and the mean authentication overhead amount remains about 1.58 kB per record. In that system, the operational sophistication of batching may lack justification. For hundreds or thousands of transactions per second, the same latency budget enables tens or hundreds of records to use a single signature. A smart-city gateway must thus be able to shift between small batches, large batches, and direct signing rather than imposing a single cryptographic granularity across the entire workload. Table 5. shows the qualitative security-performance trade-offs between the evaluated approaches.

Table 5. Qualitative security-performance trade-off.

Approach	PQ at device	Low link burden	Public device non-rep.	Gateway trust
Direct ML-DSA	Yes	No	Yes	Low
AEB-MLDSA	No	Yes	No	High
AEB direct path	Conditional	No	Yes	Low
Adaptive AEB	Conditional	Conditional	Conditional	Policy-based

4.7 Smart-City Deployment Analysis

A smart city comprises multiple security domains, and this variety is the primary reason to preserve both signing pathways. High-rate environmental sensing, road-surface telemetry, weather measurements, parking occupancy, and non-billing infrastructure observations are optimal nominees for batched mode when they remain within a managed municipal trust domain. In these situations, the engineering goal is to validate an enormous number of small records without attaching a multi-kilobyte public-key signature to each sample.

Additional records hold more vital attribution necessities; commands for remoting actuator commands, privileged administrator actions, inter-agency approvals, emergency alarms, billing or settlement events, and records likely used as independent legal

evidence should normally follow the direct ML-DSA path. The operational mode can subsequently be assigned at the data-class level instead of a permanent property of the device. A singular endpoint may transmit routine telemetry via AEB-MLDSA, then shift to direct signing for an essential state transition. Table 6. below illustrates smart city policy mapping (where applicable, subject to regulatory requirements).

Table 6. Illustrative smart-city policy mapping (subject to regulatory requirements).

Smart-city workload	Typical rate	Suggested mode	Reason
Parking / environmental telemetry	High	Batched	Strong amortization benefit
Traffic sensing	High	Batched	High volume; bounded latency
Smart metering telemetry	Moderate-high	Batched*	Depends on billing/evidence rules
Administrative approval	Low	Direct PQ	Public attribution
Emergency control	Low/event	Direct PQ	Safety / non-repudiation
Actuator command	Low-medium	Direct PQ	High consequence

The classification is determined, and a municipal authority or utility regulator can use individual metering records as billing evidence and demand direct public signatures, while the design principle that enhances performance must not unintentionally change the evidentiary meaning of the record.

Migration can be performed in stages: operators can primarily inventory existing signatures, traffic sizes, key lifetimes, endpoint capabilities, and latency classes, while edge gateways are updated to support ML-DSA, allowing the ledger to obtain quantum-resistant gateway validation while constrained devices keep their established local authentication. Devices of high criticality could transition to direct ML-DSA as firmware and hardware permit; this sequence is aligned with crypto-agility guidance since it prevents the concurrent substitution of each endpoint [10].

4.8 Additional Sensitivity and Ablation Analysis

Equation (4) isolates the cost per record corresponding to the fields shared by the batch. The 32-byte HMAC is paid within each device communication, while the 32-byte Merkle root, ML-DSA signature, and batch metadata are distributed. As B grows, the modeled authentication overhead approaches thus reside 32 bytes instead of zero; this reduces the bound that helps explain the diminishing returns in Fig. 6 when the shared term is small, and expanding the batch further produces only modest byte savings while continuing to expand the possible waiting time.

The outcome is more adaptive to batch composition compared to small changes in metadata size. When the metadata assumption is increased from 16 to 32 bytes, the B = 64 authentication overhead increases from about 70.56 to approximately 70.81 bytes; however, lowering the batch from 64 to 32 raises the overhead to about 109.13 bytes. In practical terms, timeout policy and traffic intensity are therefore more important than modest metadata growth.

Improved ML-DSA security classifications boost the overall benefit of aggregation, since the shared signature becomes larger. Participating in the signing with ML-DSA-65 or ML-DSA-87 becomes more expensive for small telemetry records, while AEB distributes that additional byte cost across the batch, and this does not indicate that the lower security classification must always be selected. A level of security level must be determined based on the required protection lifetime and risk profile; batching is a method to control performance after making that security decision.

The Poisson arrival model operates as the baseline rather than claiming that all smart-city traffic lacks memory; as smart meters can transmit data periodically, cameras might produce bursty event metadata, and emergency sensors could show correlated bursts. Bursty traffic can enhance batching effectiveness temporarily; in contrast, it may also generate queueing and downstream ledger conflicts. A production controller requires estimating λ within a moving window and may require queue-length feedback or hysteresis to avoid fluctuations between minimum and maximum batch sizes.

4.9 Limitations and Validity

The first limitation is that the study is based on analysis and simulation, which means it does not benchmark a live Ethereum, Tendermint, Hyperledger Fabric, or another blockchain execution method. The 1-MiB capacity measure accounts for normalized transaction bytes while excluding block headers, database indexing, smart-contract execution, consensus metadata, and implementation-specific encoding; it must be considered byte-capacity pressure, and not as measured end-to-end throughput.

The second is that the 81-byte constrained-link value serves as an RFC 4944 example of security overhead rather than a universal payload limit for every IEEE 802.15.4 network. Real protocol stacks can utilize different security settings, additional adaptation-layer fields, and header compression; as the fragmentation examination is thus comparative: it reveals how a multi-kilobyte signature alters the order of magnitude of frame count. A deployment study needs to replace the actual MTU and protocol overhead.

The third is that the default transaction size, metadata length, block-byte budget, Poisson arrivals, and 10-ms non-batching latencies are defined as modeling assumptions. Absolute smart-city workloads may show irregular, bursty, correlated, or event-driven behavior; future-based verification needs to replay traces from smart meters, municipal infrastructure, or traffic sensors and estimate the resulting batch-size distribution on an active blockchain and on suitable edge hardware.

Lastly, the fourth one, batched mode, is based on symmetric device authentication and trusted edge gateways. Such a compromise is not ideal for all possible applications, indicating why the direct path stays a component of the framework. This study focuses on ML-DSA as it provides a reliable and standardized baseline; other post-quantum signature families may be used interchangeably with the same equations by swapping S_sig and the related processing parameters; and NIST continues its activities in the standardization of post-quantum digital signatures, which underscores the importance of keeping the framework algorithm-agile instead of irreversibly binding it to a particular signature family [22].

5. Conclusion

Post-quantum migration may involve significant storage costs and communication overhead in blockchain-based smart-city IoT systems, since the signature will likely be larger than the sensor record it protects. This study examines the penalty utilizing a constrained-link fragmentation model and final ML-DSA signature sizes, followed by offering the AEB-MLDSA as a platform-independent approach to lower the cost; routine telemetry has been confirmed with a valid edge domain and verified by an ML-DSA-signed batch commitment, while essential records maintain direct post-quantum signing when public device-level non-repudiation is needed.

Given the typical 256-byte transaction, direct ML-DSA-44 provides a 2,676-byte normalized transaction and increases a 1-MiB byte budget to 391 transactions; using a 64-record AEB batch, amortized authentication overhead reduces to 70.56 bytes and normalized throughput grows to 3,210 transactions. The communication gain includes two distinctive costs: batching delay and gateway trust; the SLA-aware controller shows the secondary cost, while the simulation indicates that high-rate telemetry can effectively change the signatures with specified delay, whereas sparse traffic should employ direct signing or smaller batches.

The overall implication is that the design of a quantum-safe blockchain for smart cities needs to be treated as a cross-layer architectural challenge rather than only a direct cryptographic substitution; trust placement, traffic intensity, signature size, constrained-link capacity, latency policy, and ledger growth should be evaluated together. The valuable next phase is to establish AEB-MLDSA in an authorized blockchain test environment, analyze standardized parameter settings on the edge devices, simulate authentic urban traffic traces, and investigate threshold post-quantum attestation to reduce reliance on a singular gateway.

Statements and Declarations

Funding: This research received no external funding. The APC, if applicable, will be funded by the author.

Conflicts of Interest: The author declares no conflict of interest.

Author Contributions: The sole author was responsible for conceptualization, methodology, analytical modeling, simulation design, analysis, writing, review, and editing.

Data Availability: No proprietary or human-subject dataset was used. Numerical results were generated from the disclosed analytical equations and simulation settings.

Code Availability: The reproducible analysis code and generated result tables can be supplied as supplementary material or made available by the corresponding author.

Ethics Statement: The study did not involve human participants, animals, clinical records, or personally identifiable data; ethics approval was therefore not required.

Publisher's Note: All claims expressed in this article are solely those of the author and do not necessarily represent those of the affiliated organization, or those of the publisher, editors, and reviewers.

References

[1] T. Nguyen, H. Nguyen, and T. Nguyen Gia, "Exploring the integration of edge computing and blockchain IoT: Principles, architectures, security, and applications," Jun. 01, 2024, *Academic Press*. doi: 10.1016/j.jnca.2024.103884.

[2] K. Guo, C. Zhan, M. Niu, X. Li, Z. Zheng, and A. Sharma, "An integrated IoT and blockchain lightweight framework for secure smart cities," *Discover Internet of Things*, vol. 6, no. 1, Dec. 2026, doi: 10.1007/s43926-025-00273-8.

[3] National Institute of Standards and Technology [NIST], "Module-lattice-based digital signature standard," Aug. 2024. doi: 10.6028/NIST.FIPS.204.

[4] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, "Transmission of IPv6 packets over IEEE 802.15.4 networks," 2007.

[5] S. Y. Chang and Q. Khan, "Post-Quantum Cryptography in Networking Protocols: Challenges, Solutions, and Future Directions," Feb. 01, 2026, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/cryptography10010012.

[6] E. Ishchukova, F. S. Sajadi, S. Petrenko, A. Petrenko, and A. Nekrasov, "Post-Quantum Transition in Blockchain Architectures: A Systematic Review of Cross-Layer Security, Performance, and Governance Constraints," Jun. 01, 2026, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/technologies14060367.

[7] A. Wicaksana, "A survey on quantum-safe blockchain security infrastructure.," *Comput. Sci. Rev.*, vol. 57, p. 100752, 2025, doi: 10.1016/J.COSREV.2025.100752.

[8] Z. Yang, H. Alfauri, B. Farkiani, R. Jain, R. Di Pietro, and A. Erbad, "A Survey and Comparison of Post-Quantum and Quantum Blockchains," *IEEE Communications Surveys and Tutorials*, vol. 26, no. 2, pp. 967–1002, 2024, doi: 10.1109/COMST.2023.3325761.

[9] K. Adere *et al.*, "Systematic review on securing IoT systems with post quantum cryptography emerging threats, countermeasures, and future research needs," Dec. 01, 2026, *Springer Nature*. doi: 10.1007/s43926-025-00275-6.

[10] E. Barker *et al.*, "Considerations for Achieving Crypto Agility," Gaithersburg, MD, 2026. doi: 10.6028/NIST.CSWP.39-upd1.

[11] M. Prorock and O. Steele, "RFC 9964 ML-DSA for JSON Object Signing and Encryption (JOSE) and CBOR Object Signing and Encryption (COSE)," 2026. doi: 10.17487/RFC9964.

[12] H. Aldawsari, "A blockchain-based approach for secure energy-efficient IoT-based Wireless Sensor Networks for smart cities," *Alexandria Engineering Journal*, vol. 126, pp. 1–7, Jul. 2025, doi: 10.1016/j.aej.2025.04.052.

[13] A. Iftikhar, K. N. Qureshi, F. B. Hussain, M. Shiraz, and M. Sookhak, "A blockchain based secure authentication technique for ensuring user privacy in edge based smart city networks.," *Journal of Network and Computer Applications*, vol. 233, p. 104052, 2025, doi: 10.1016/J.JNCA.2024.104052.

[14] R. Islam *et al.*, "Decentralized trust framework for smart cities: a blockchain-enabled cybersecurity and data integrity model," *Sci. Rep.*, vol. 15, no. 1, Dec. 2025, doi: 10.1038/s41598-025-06405-y.

[15] S. Liu, S. Rao, X. A. Wang, K. Tian, and Y. Wang, "PQ-BCDA: A post-quantum blockchain based cross-domain authentication scheme for Internet of Things.," *Internet Things*, vol. 33, p. 101737, 2025, doi: 10.1016/J.IOT.2025.101737.

[16] J. Zhang, Z. Liu, D. Yao, and J. Zhao, "Scalable post-quantum non-interactive multi-signature for IoT-enabled blockchain systems.," *Internet Things*, vol. 38, p. 101998, 2026, doi: 10.1016/J.IOT.2026.101998.

[17] Y. Huang, M. Ma, W. J. K. Raymond, and C.-O. Chow, "An adaptive intrusion detection system for the internet of things using large language models and post-quantum-secure blockchain.," *Computer Networks*, vol. 274, p. 111819, 2026, doi: 10.1016/J.COMNET.2025.111819.

[18] M. T. Naz, W. Elmedany, and M. Ali, "Securing SCADA systems in smart grids with IoT integration: A Self-Defensive Post-Quantum Blockchain Architecture.," *Internet of Things*, vol. 28, p. 101381, 2024, doi: 10.1016/J.IOT.2024.101381.

[19] T. G. Keshavamurthy, S. Guruprasad, K. N. Hareesh, M. V. C. Murthy, K. J. Ashwini, and K. S. Swathi, "Post quantum blockchain framework using probabilistic hidden state deep learning for smart IoT systems," *Discover Artificial Intelligence*, vol. 6, no. 1, p. 865, Dec. 2026, doi: 10.1007/s44163-026-01964-z.

[20] W. Barker, W. Polk, and M. Souppaya, "Getting Ready for Post-Quantum Cryptography: Exploring Challenges Associated with Adopting and Using Post-Quantum Cryptographic Algorithms," Gaithersburg, MD, Apr. 2021. doi: 10.6028/NIST.CSWP.04282021.

[21] M. Sönmez Turan and L. T. A. N. Brandão, "Keyed-Hash Message Authentication Code (HMAC): Specification of HMAC and Recommendations for Message Authentication," 2024. doi: 10.6028/NIST.SP.800-224.ipd.

[22] National Institute of Standards and Technology, "Post-quantum cryptography: Additional digital signature schemes," May 2026. doi: 10.6028/NIST.IR.8610.