



Comparative Performance and Communication Overhead Analysis of Post-Quantum Cryptographic Algorithms for Quantum-Safe Systems

José A. Montenegro¹, Ruben Rios²

¹Network, Information and Computer Security (NICS) Lab, Universidad de Málaga, Málaga 29071, Spain — jmmontes@uma.es.

²Network, Information and Computer Security (NICS) Lab, Universidad de Málaga, Málaga 29071, Spain — ruben.rdp@uma.es

Received:- 21/01/2026, Revised:- 09/03/2026, Accepted:- 06/04/2026, Published:- 29/04/2026

ABSTRACT

Post-quantum cryptography offers a promising safeguard for communication systems in the era of quantum-computing threats, it is essential to consider the computational efficiency and communication load of these systems when implementing them on the ground. This study compared the performance of ML-KEM and ML-DSA post-quantum algorithms to both the classical RSA-4096 and ECDSA-P256 algorithms on benchmarks such as execution time, variability, public key size, size-expansion ratio, communication overhead and signature/ciphertext size. The performance of ML-KEM variants was significantly better than RSA-4096 on the key generation, encapsulation, and decapsulation operations, while the size of the ciphertext and communication overhead were larger for the higher levels of the parameters. The advantages of ML-DSA variants were observed as their key generation was faster and the verification was comparable with ECDSA-P256 but the signing was slower and the size of the signature was much larger. The communication overhead of ML-KEM ranged between 50.00% and 206.25% over the classical counterparts, while that of ML-DSA ranged from 3,308.45% to 6,416.90% over the classical counterparts. Increasing security parameters also showed greater communication and computation costs for stronger configurations. The results highlight the need to carefully consider multiple factors beyond execution speed when selecting post-quantum algorithms for migration to quantum-safe systems, including performance efficiency, cryptographic object size, and communication requirements. These results give quantitative justification for choosing the application-specific processing capacity, bandwidth, security, constraints, and deployment conditions for the selection of standardized post-quantum mechanisms.

Keywords: Post-quantum cryptography; Key encapsulation; Digital signatures; Communication overhead; Quantum-safe systems

1. Introduction

With the advent of quantum computing, researchers are looking at post-quantum cryptography (PQC), an area that is increasingly dominating research efforts. As quantum computing advances, the long-held security assumptions of widely used public-key systems are being called into question, making post-quantum cryptography (PQC) a focal area of research. RSA and elliptic-curve cryptography are examples of algorithms where mathematical problems may be easily solved by powerful quantum computers, putting at risk confidentiality, authentication, and digital trust infrastructures. This is both a cryptographic and a systems-engineering need for developing alternatives to quantum-resistant cryptography. In the realm of connected environments where devices, gateways, cloud platforms and communication protocols need to maintain security for extended lifetimes, migration poses a particular challenge (Adere et al., 2026). More general evaluations also view PQC as one such mitigation measure for assuring the safety of today's digital ecosystems against potential future quantum-based attacks (Beullens et al., 2021).

This transition has been facilitated by the National Institute of Standards and Technology's standardization process. The second round of evaluation narrowed down the list of candidates by analyzing implementation characteristics, suitability for practical deployment, security, and efficiency. In the second round of analysis, the candidates were reduced further, and evidence was further bolstered for standardized quantum-resistant mechanisms (Alagic et al., 2022; Alagic et al., 2020). One of the main results was the formalisation of module-lattice-based key encapsulation, known as ML-KEM, which offers a set of standard parameters for quantum-safe key establishment (Avanzi et al., 2024). These advancements are moving beyond algorithm selection and towards gaining insight into the consequences of implementing standardized PQC mechanisms in real communication systems.

But migration isn't just about swapping out the classical algorithms for quantum-safe ones. The first challenge for organizations is to identify situations that may have cryptographic dependencies, understand the interoperability of the existing services, and implement transition mechanisms without impacting existing services (Barker et al., 2021a). Relevant guidance for PQC migration highlights the need to inventory all cryptographic assets and consider constraints before widespread adoption (Barker, 2021b). Fewer recent migration frameworks also indicate that factors such as the dependency of applications, protocols, hardware, and software can impact the complexity of transitions and deployment risk (Hasan et al., 2024). Staged deployment and operational impact evaluation prior to production rollout further emphasizes the need for preparatory guidance on quantum-safe adoption (Newhouse et al., 2023). Some hybrid solutions have also been suggested to maintain the compatibility of classical security with post-quantum security in the transition period (Barbosa et al., 2024). Evidence, however, is needed for both security properties and computational and communication costs (Aydeger et al., 2024) for strategic migration.

These deployment concerns are particularly important in resource-constrained and networked environments. IoT devices can be constrained to a small processing capability, memory, energy capacity, and communication bandwidth, which makes cryptographic overhead an important implementation constraint (Kumari et al., 2022b). Research on secure IoT architectures has revealed computation, memory usage, payload size, and protocol integration as important challenges in the quantum transition era (Schöffel et al., 2022). The same issues apply to industrial communication systems, as security mechanisms need to be integrated with features of real-time performance and compatibility (Paul & Guerin, 2020). Transitional certificate structures for TLS 1.3 show that post-quantum authentication may have significant impacts on the composition of handshake and communication volume (Paul et al., 2022) at the protocol layer. Studies related to MQTT also indicate that the overhead of authentication and communication efficiency in lightweight messaging systems need to be considered for PQC adoption (Samandari & Gritti, 2023).

Computation and communication need to be taken into account during comparative evaluation of PQC algorithms. The sizes of the public key, ciphertext and signature, and the execution time affect the efficiency of processing and the amount of data exchanged. Even with efficient computational execution, larger cryptographic objects can require greater bandwidth and communication overhead. This trade-off is fundamental in quantum-safe system design, because it is important to consider that using an algorithm just for its speed could be at the expense of high costs of transmission. The present study thus compares the security of chosen standardized post-quantum key-encapsulation and digital-signature schemes with their classical counterparts with a particular focus on their operation-specific timings, communication overhead, cryptographic object size, and parameter scaling. This evidence can assist in making informed deployment decisions for quantum-safe communication environments and infrastructures.

Objectives:

- To perform a performance comparison of the computational performance of selected post-quantum and classical cryptographic algorithms in terms of key generation, encapsulation, decapsulation, signing, and verification operations.
- To measure communication overhead in terms of public-key size, ciphertext or signature size, size-expansion ratio, and percentage over classical communication.
- To explore the impact of different levels of post-quantum parameters on the computational vs communication complexity of quantum-safe systems.

2. Materials and Methods

2.1 Study Design

To compare the computational performance and communication overhead of the considered PQ algorithms with the classical cryptographic algorithms, the methodology used was the comparative experimental benchmarking design. Two types of functions were considered: key encapsulation mechanisms and digital-signature algorithms. The computational performance was measured by the execution time, the public-key size, the size of the ciphertext or the signature, the size of the transmitted artifact, the ratio of size expansion and the ratio of the percentage of the communication overhead.

2.2 Dataset and Algorithm Selection

The study used the “Comparative Performance Dataset: Post-Quantum Cryptography Standards vs. Classical Cryptographic Baselines.” The 24 observations in the dataset were for algorithm-operation variables, such as algorithm family, cryptographic function, operation type, number of iterations in the benchmark, mean execution time, standard deviation, size of the public key, size of the artifact, size of the transmitted data, classical baseline, size-expansion ratio, and percentage of communication overhead. The key-establishment group consisted of a classical baseline, RSA-4096 by using OAEP-SHA256, and three post-quantum baselines: ML-KEM-512, ML-KEM-768 and ML-KEM-1024. The selected digital signature schemes were ECDSA-P256 as the classical digital signature and as the post-quantum digital signature, ML-DSA-44, ML-DSA-65 and ML-DSA-87 (Mendonça et al., 2026).

2.3 Computational Performance Evaluation

The execution time in milliseconds was used to measure the computational efficiency. Key generation, encapsulation and decapsulation were considered for key establishment algorithms. Key generation, signing and verification were evaluated for digital-signature algorithms. The variability in the repetitions of a benchmark was described as standard deviation. The lowest mean execution times meant that the same cryptographic function had a higher computational efficiency. The average of execution times was reported as:

$$\bar{T} = \frac{1}{n} \sum_{j=1}^n T_j$$

where T_j represents the execution time of the j -th benchmark iteration and n represent the total number of iterations.

2.4 Communication Overhead Analysis

The overhead of communication was measured by the size of the cryptographic objects sent in a secure communication. The size of the public keys was taken for all algorithms; for key-establishment schemes, the size of the ciphertext was used; for digital-signature schemes, the size of the signature was used. The relative communication overhead was defined as:

$$CO(\%) = \frac{S_{PQC} - S_{Classical}}{S_{Classical}} \times 100$$

where S_{PQC} represents the transmitted size of the post-quantum artefact and $S_{Classical}$ represents the corresponding

classical baseline size. The size-expansion ratio was calculated as:

$$SER = \frac{S_{PQC}}{S_{Classical}}$$

Values greater than 1 meant there was more communication payload than the classical comparator.

2.5 Security-Parameter and Performance-Overhead Trade-Off Analysis

The impact of the post-quantum parameter levels was tested individually for ML-KEM-512, ML-KEM-768, and ML-KEM-1024, and for ML-DSA-44, ML-DSA-65 and ML-DSA-87. Comparisons of execution time, public-key size, size of the ciphertext or signature and communication overhead were considered. To determine if there were any computational costs associated with increased communication workload, a combined trade-off analysis was also conducted to compare execution time versus transmitted artifact size.

2.6 Statistical Analysis

Differences among the algorithms were summarized using descriptive and comparative statistical analyses. For the key establishment group and the digital-signature group, the following were compared: mean execution time, standard deviation, public-key size, artifact size, size-expansion ratio, and communication-overhead percentage. The relative differences between the post-quantum algorithms and the classical (conventional) baselines were also computed. Since the data did not consist of individual iterations' measurements, descriptive statistics, ratios, percentage differences, and graphical comparisons were used and inferential hypothesis testing was limited.

2.7 Data Quality and Ethical Considerations

Before the analysis of the data, the consistency of the algorithm names, cryptographic functions, operation labels, timing units and byte-size measurements was checked. Values of operations with no associated cryptographic artifacts were not considered missing experimental observations, nor were unobserved values imputed. There were no clinical records, human subjects, animals, personally identifiable information or other sensitive data involved in the study, only computational benchmark data. Hence, ethical approval and informed consent were not necessary at the time of the study.

3. Results

3.1 Comparative Computational Performance of Key-Establishment Algorithms

Significant differences were observed among the computational performance of the classical RSA-4096 baseline and the three ML-KEM parameter sets for key generation, encapsulation and decapsulation. As shown in Table 1, RSA-4096 required 417.1840 ms for key generation, whereas ML-KEM-512, ML-KEM-768, and ML-KEM-1024 required only 0.0351, 0.0654, and 0.0693 ms, respectively. For ML-KEM-512, ML-KEM-768, and ML-KEM-1024, relative to RSA-4096 the key generation was about 11,885.6 times faster, 6,379.0 times faster, and 6,020.0 times faster, respectively.

The post-quantum schemes have been also favored by encapsulation. RSA-4096 required 0.1243 ms, compared with 0.0248 ms for ML-KEM-512, 0.0386 ms for ML-KEM-768, and 0.0489 ms for ML-KEM-1024. The biggest operational contrast after key generation was decapsulation which took 4.0958 ms for RSA-4096, while the range for ML-KEM was 0.0249–0.0498 ms. The results are shown in Figure 1 on a log-scale to avoid the large differences in key-generation time.

Table 1. Computational performance of RSA-4096 and ML-KEM algorithms

Algorithm	Key generation, ms	Encapsulation, ms	Decapsulation, ms	KeyGen SD, ms	Encapsulation SD, ms	Decapsulation SD, ms
RSA-4096	417.1840	0.1243	4.0958	276.1657	0.1141	0.3539

ML-KEM-512	0.0351	0.0248	0.0249	0.0382	0.0116	0.0064
ML-KEM-768	0.0654	0.0386	0.0382	0.0933	0.0166	0.0110
ML-KEM-1024	0.0693	0.0489	0.0498	0.0395	0.0288	0.0189

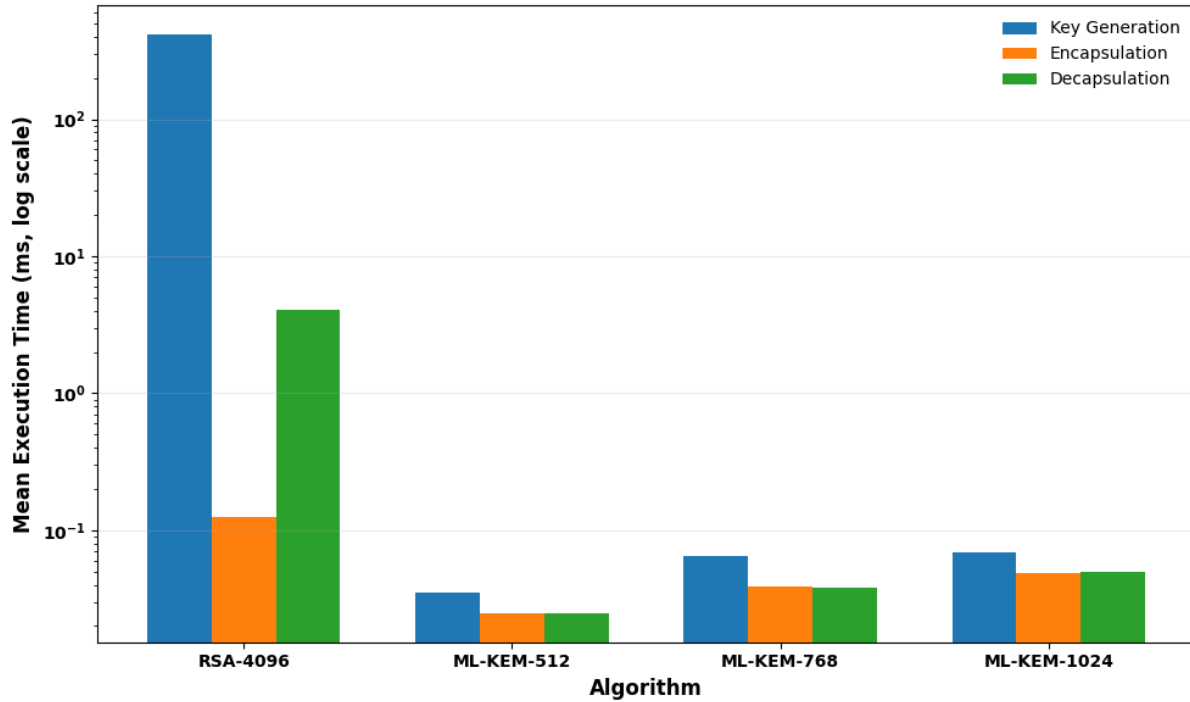


Figure 1. Comparative execution time of RSA-4096 and ML-KEM algorithms across key generation, encapsulation, and decapsulation operations

3.2 Comparative Computational Performance of Digital-Signature Algorithms

The signature algorithms had different computational profiles. Table 2 shows that ECDSA-P256 required 1.0086 ms for key generation, whereas ML-DSA-44, ML-DSA-65, and ML-DSA-87 required 0.0478, 0.2018, and 0.1843 ms, respectively. So, all three variants of the ML-DSA were faster than ECDSA-P256 in creating keys.

ECDSA-P256 had slightly better signing performance. Its mean signing time was 0.0800 ms, whereas ML-DSA-44, ML-DSA-65, and ML-DSA-87 required 0.1189, 0.1577, and 0.1933 ms, respectively. Verification was significantly faster for the ML-DSA-44 and ML-DSA-65 than for ECDSA-P256, on the other hand. The time it takes to verify an ML-DSA-44 signature was 0.0505 ms, while the time to verify an ML-DSA-65 signature was 0.0673 ms. ML-DSA-87 was verified in 0.0949 ms, almost the same as ECDSA-P256. Figure 2 shows performance comparison for the execution time of the three digital-signature operations.

Table 2. Computational performance of ECDSA-P256 and ML-DSA algorithms

Algorithm	Key generation, ms	Signing, ms	Verification, ms	KeyGen SD, ms	Signing SD, ms	Verification SD, ms
ECDSA-P256	1.0086	0.0800	0.0955	4.2723	0.6069	0.0301

ML-DSA-44	0.0478	0.1189	0.0505	0.0195	0.0673	0.0176
ML-DSA-65	0.2018	0.1577	0.0673	0.4533	0.0970	0.0144
ML-DSA-87	0.1843	0.1933	0.0949	0.2642	0.0799	0.0154

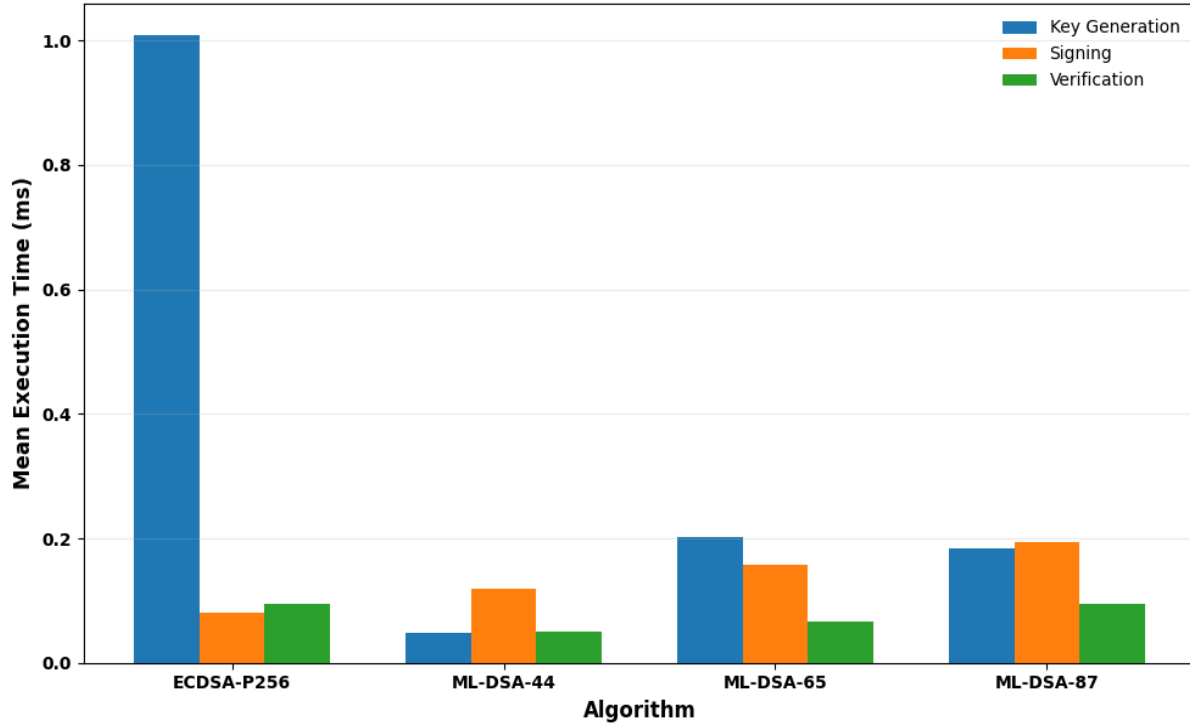


Figure 2. Comparative execution time of ECDSA-P256 and ML-DSA algorithms across key generation, signing, and verification operations

3.3 Cryptographic Object Sizes and Communication Overhead

The communication-cost analysis showed significant larger cryptographic objects that need to be transmitted for the post-quantum schemes, especially for ML-DSA. The baseline for the RSA-4096 used an 800-byte public key and produced an encapsulation artifact of 512 bytes. The baseline for RSA-4096 required an 800-byte public key and produced a 512-byte encapsulation artifact. ML-KEM-512 had the same public-key of 800 bytes, but generated 768 bytes of ciphertext. The public-key size grew to 1,184 bytes and the ciphertext size to 1,088 bytes with ML-KEM-768, and to 1,568 bytes each for the public-key and the ciphertext with ML-KEM-1024.

The size expansion ratio of ML-KEM-512 was 1.50 and the communication overhead was 50.00% when compared to RSA-4096. ML-KEM-768 increased these values to 2.125 and 112.50%, while ML-KEM-1024 reached 3.0625 and 206.25%, respectively.

However, the signature schemes showed communication expansion that is much higher. ECDSA-P256 produced a 71-byte signature, whereas ML-DSA-44, ML-DSA-65, and ML-DSA-87 produced signatures of 2,420, 3,309, and 4,627 bytes. Corresponding size-expansion ratios were 34.0845, 46.6056, and 65.1690. Communication overhead therefore increased from 3,308.45% for ML-DSA-44 to 4,560.56% for ML-DSA-65 and 6,416.90% for ML-DSA-87. In both cryptographic groups, there is a growth in the size of the artefacts that were transmitted, as shown in Figure 3. Table 3 shows Cryptographic object size and communication overhead.

Table 3. Cryptographic object size and communication overhead

Algorithm	Public key, bytes	Ciphertext/signature, bytes	Classical baseline, bytes	Size-expansion ratio	Communication overhead, %
RSA-4096	800	512	512	1.0000	0.00
ML-KEM-512	800	768	512	1.5000	50.00
ML-KEM-768	1,184	1,088	512	2.1250	112.50
ML-KEM-1024	1,568	1,568	512	3.0625	206.25
ECDSA-P256	178	71	71	1.0000	0.00
ML-DSA-44	1,312	2,420	71	34.0845	3,308.45
ML-DSA-65	1,952	3,309	71	46.6056	4,560.56
ML-DSA-87	2,592	4,627	71	65.1690	6,416.90

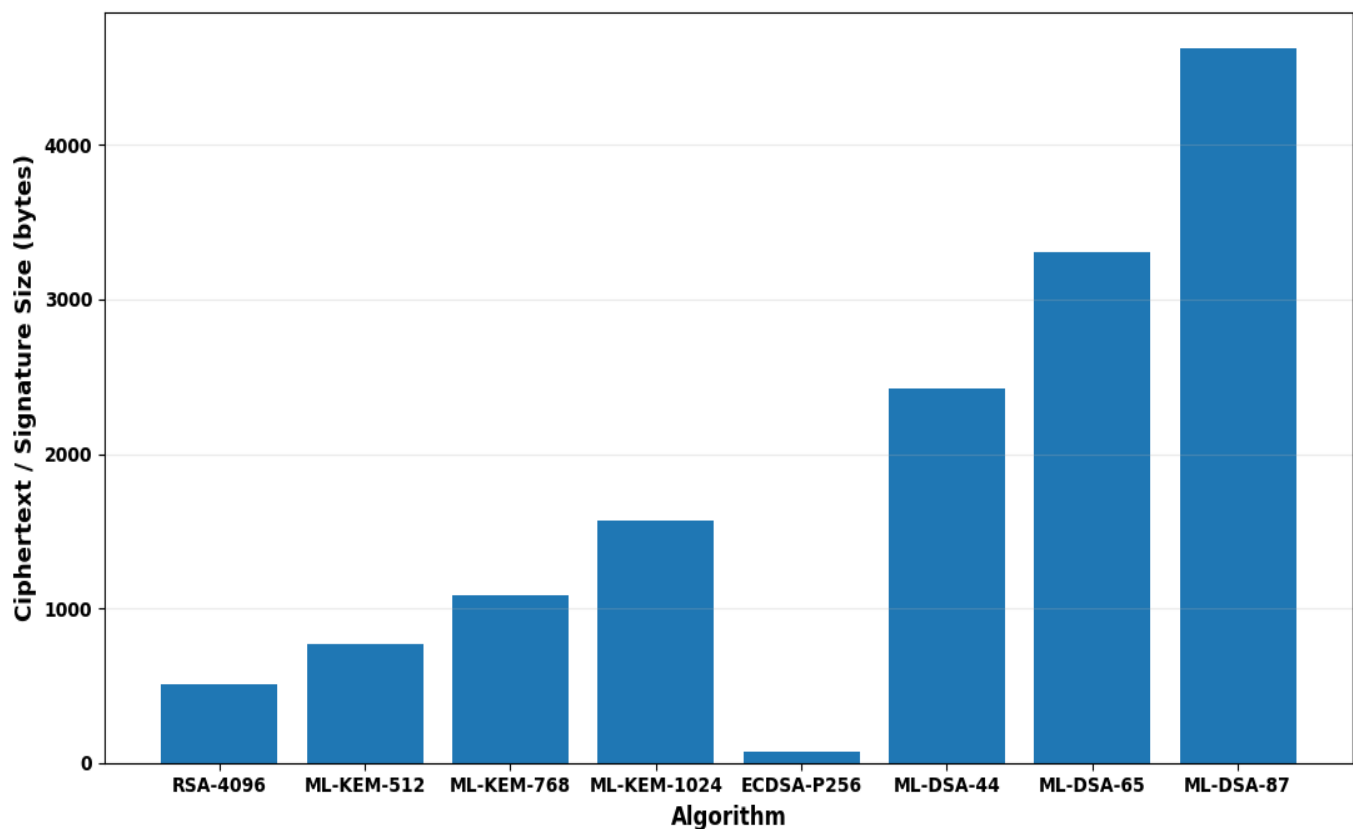


Figure 3. Comparison of transmitted ciphertext and signature sizes for classical and post-quantum cryptographic algorithms

3.4 Relative Computational Performance Against Classical Baselines

Table 4 also quantified the difference in computation of the operation-specific comparison between the post-quantum algorithms and their classical counterparts. ML-KEM-512 only needed 19.95% of the time needed for encapsulation with RSA-4096 and 0.61% of the time to decapsulate with RSA-4096. ML-KEM-768 required 31.05% and 0.93% of the corresponding RSA-4096 times, while ML-KEM-1024 required 39.34% and 1.22%, respectively.

The variants of ML-DSA generated quicker keys than ECDSA-P256, while the latter signed faster. ML-DSA-44 required 148.63% of the ECDSA signing time, ML-DSA-65 required 197.13%, and ML-DSA-87 required 241.63%. The verification time of ML-DSA-44, ML-DSA-65 and ML-DSA-87 was 52.88%, 70.47% and 99.37% of the time used by ECDSA-P256 respectively.

Table 4. Relative execution-time ratios of post-quantum algorithms versus classical baselines

Algorithm	Operation	PQC time, ms	Classical time, ms	PQC/Classical ratio	Relative interpretation
ML-KEM-512	Key generation	0.0351	417.1840	0.000084	Faster
ML-KEM-512	Encapsulation	0.0248	0.1243	0.1995	Faster
ML-KEM-512	Decapsulation	0.0249	4.0958	0.0061	Faster
ML-KEM-768	Key generation	0.0654	417.1840	0.000157	Faster
ML-KEM-768	Encapsulation	0.0386	0.1243	0.3105	Faster
ML-KEM-768	Decapsulation	0.0382	4.0958	0.0093	Faster
ML-KEM-1024	Key generation	0.0693	417.1840	0.000166	Faster
ML-KEM-1024	Encapsulation	0.0489	0.1243	0.3934	Faster
ML-KEM-1024	Decapsulation	0.0498	4.0958	0.0122	Faster
ML-DSA-44	Key generation	0.0478	1.0086	0.0474	Faster
ML-DSA-44	Signing	0.1189	0.0800	1.4863	Slower
ML-DSA-44	Verification	0.0505	0.0955	0.5288	Faster
ML-DSA-65	Key generation	0.2018	1.0086	0.2001	Faster
ML-DSA-65	Signing	0.1577	0.0800	1.9713	Slower
ML-DSA-65	Verification	0.0673	0.0955	0.7047	Faster

ML-DSA-87	Key generation	0.1843	1.0086	0.1827	Faster
ML-DSA-87	Signing	0.1933	0.0800	2.4163	Slower
ML-DSA-87	Verification	0.0949	0.0955	0.9937	Approximately equivalent

3.5 Security-Parameter Scaling and Performance-Communication Trade-Off

For each family of parameters, as the strength was increased, the communication artifacts grew larger, and the execution times were generally greater. The size of the public-key was increased by 96.0% and the size of the ciphertext was increased by 104.17% when the size of the ML-KEM was changed from ML-KEM-512 to ML-KEM-1024, as seen in Table 5. The key-generation time and the time needed for encapsulation grew by 97.44% and 97.18% respectively, while the time needed for decapsulation grew by 100.0%.

The key-generation time for the ML-DSA family had an even larger increase. Key-generation time ranged from a 285.56% rise in ML-DSA-44, to an increase of 62.57% in signing time and 87.92% in verification time for ML-DSA-87. The size of the public key grew up 97.56% and signature grew up 91.20%.

The results show that both algorithm families suffered from a communication penalty when their post-quantum parameter was increased. Figure 4 combines the computational time and artifact size, and clearly indicates that there is a tendency for larger parameter variants to require more communication, with some variations requiring more computational cost than others depending on the specific operation being used, KEM and signature.

Table 5. Scaling from the lowest to highest post-quantum parameter configuration

Algorithm family	Metric	Lowest configuration	Highest configuration	Percentage increase
ML-KEM	Key generation time, ms	0.0351	0.0693	97.44
ML-KEM	Encapsulation time, ms	0.0248	0.0489	97.18
ML-KEM	Decapsulation time, ms	0.0249	0.0498	100.00
ML-KEM	Public-key size, bytes	800	1,568	96.00
ML-KEM	Ciphertext size, bytes	768	1,568	104.17
ML-DSA	Key generation time, ms	0.0478	0.1843	285.56
ML-DSA	Signing time, ms	0.1189	0.1933	62.57
ML-DSA	Verification time, ms	0.0505	0.0949	87.92

ML-DSA	Public-key size, bytes	1,312	2,592	97.56
ML-DSA	Signature size, bytes	2,420	4,627	91.20

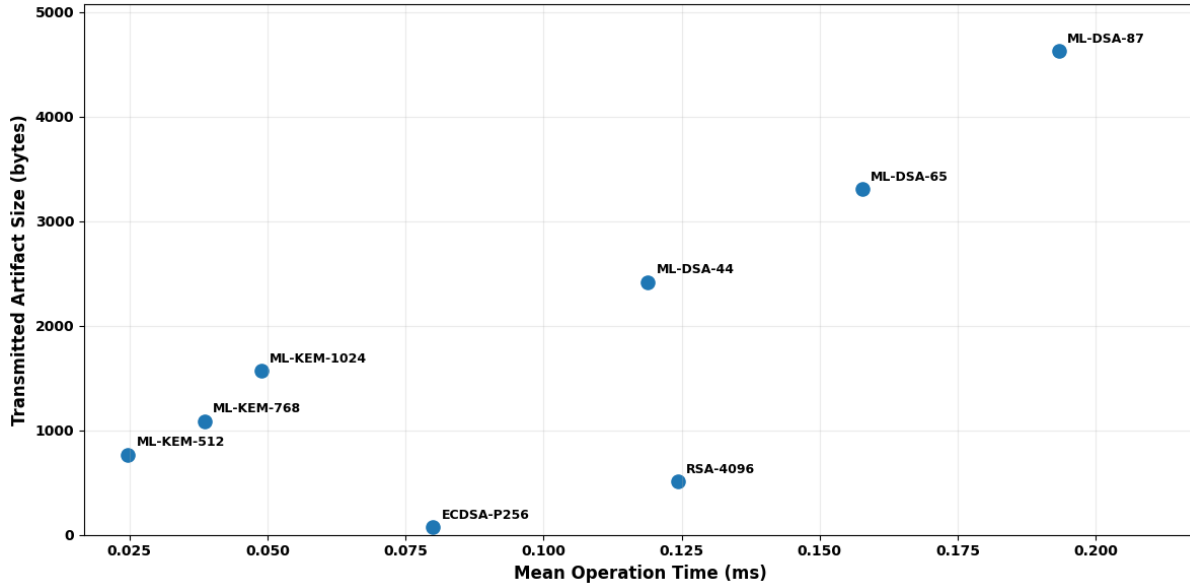


Figure 4. Performance–communication trade-off between operation time and transmitted cryptographic artifact size across classical and post-quantum algorithms

3.6 Summary of Results

The results showed that there was a clear trade-off between computational efficiency and communication overhead of the post-quantum cryptographic algorithms. Key generation, as well as encapsulation and decapsulation, consistently outperformed RSA-4096, with ML-KEM variants showing a better performance and ML-DSA variants showing a faster key generation performance with a competitive performance in verification, but slower signing performance compared to ECDSA-P256. As the post-quantum parameter levels increased, the overheads of communication also showed significant increases, in the case of ML-KEM from 50.00% to 206.25% and ML-DSA signature overhead from 3,308.45% to 6,416.90% compared to the classical baselines. The overall outcome is that the communication payloads of the cryptographic operations needed for post-quantum algorithms are significantly larger, but that this can be balanced by obtaining a significant computational advantage.

4. Discussion

The results of the present invention reveal that algorithms based on Quantum cryptography can offer computational benefits over classical algorithms and can also put increased communications demands. The family of the ML-KEM demonstrated good results: key generation, encapsulation, and decapsulation were faster than with RSA-4096. This behavior is in line with the benchmarking results indicating that lattice-based cryptography can provide practical execution times on constrained and embedded platforms, where the execution efficiency varies based on the hardware and software configuration (Halak et al., 2024). In the automotive security context, post-quantum schemes were shown to be viable to execute in real time (Ravi et al., 2020). From these findings, it can be concluded that computational costs are unlikely to be the primary obstacle to implementing ML-KEM in quantum-safe systems.

The results of the digital signature were more complex. The performance of ML-DSA variants in generating keys was also faster than ECDSA-P256, and it had competitive verification performance, while it had slower signed performance. This is a trade-off that is relevant when signing occurs regularly or when latency is of concern. Previous research on signature schemes for resource-limited IoT has focused on the fact that the generation/verification cost of the signature is not independent, and that the size of the transmitted signature should be taken into account. (Behnia

& Yavuz, 2021) Lightweight authentication studies also reveal that lattice-based constructions can be computationally feasible without sacrificing any overhead due to increased size of cryptographic objects and protocol integration (Kumari et al., 2022a). As a result, ML-DSA might be appropriate for certain systems where verifying is the most important operation, but this will depend on the nature of operations, their direction, message frequency, and deployment context.

The major constraint in the current analysis was communication overhead. The ML-KEM overhead for the 512-bit and 1024-bit versions is 50.00% and 206.25% respectively, when compared to RSA-4096. The overhead for ML-DSA signatures for the different sizes is 3,308.45% and 6,416.90% respectively when compared to ECDSA-P256. The results are consistent with protocol-level studies demonstrating that the handshake size and bytes can be significantly larger in TLS contexts when using post-quantum authentication (Sikeridis et al., 2020). The larger size of key, message, and signature objects are also significant factors in communication and memory overhead in evaluations of TLS 1.3 on constrained embedded systems, such as those in Tasopoulos et al., 2022. These gains can be especially significant in real-world deployments where bandwidth, lossy channels, and limited power are critical.

The scaling analysis also revealed that as the post-quantum parameters go up, the computational cost and communication size are also raised. In ML-KEM, the size of the ciphertext has been more than doubled between the lowest and the highest configuration tested, and has been more than doubled in ML-DSA signature size. This behaviour is an illustration of the tradeoff between security parameters and system resources. Research on the topic of key exchange for IoT in industry has also focussed on choosing the methods that are quantum-safe based on the devices, frequency of communication and constraints faced (Gupta, 2023). Vehicular authentication research also shows that a lattice-based security is feasible if the protocol design takes into account the bandwidth and time constraints (Dharminder & Mishra, 2020).

There are numerous system-level consequences other than latency and byte extension. The more cryptographic operations that are performed and the larger the message exchanges, the more energy the device uses, particularly if there are repeated secure sessions. It has been found that, on constrained hardware, the handshake complexity and algorithm choice can have significant impact on power usage during evaluation of post-quantum TLS (Tasopoulos et al., 2023). There are also some examples of full post-quantum DTLS implementations in IoT environments, which suggest that, in such cases, communication overhead and fragmentation may be important in the transmission of large authentication objects through lightweight protocols (McLoughlin et al., 2023). From these observations, it would seem that the present communication size results could carry over to other implications in the field of energy and reliability in actual deployments.

The results also are significant to layered and industrial communication architectures. To preserve practical communication performance, quantum-safe authentication protocols for IoTs are being increasingly designed to incorporate cryptographic security along with design considerations to the extent of implementing the protocol. The security of quantum-safe authentication protocols is being increasingly combined with the design choices that account for implementation considerations in order to preserve practical communication performance (Prajwal et al., 2024). Likewise, in layered IoT architectures, security mechanisms of authentication and integrity must support a mix of devices and intermediate gateways (Samandari & Gritti, 2024). It is therefore important that algorithm selection is considered as a multi-dimensional engineering decision, instead of picking the fastest cryptographic primitive.

ML-KEM has a favourable profile as the ciphertext size grows, while the ML-DSA is efficient in verification but has very large signatures. As a result, the study also validates a trade-off approach for the deployment of post-quantum. Different parameters may be necessary for systems with tight bandwidth limitations as compared to systems governed by computation. These findings should be extended to future evaluations through raw iteration-level measurements, multiple hardware platforms, real network conditions, and full protocol handshakes to understand the impact of the performance size trade-offs on end-to-end quantum-safe communication.

Conclusion

This study compares the computational performance and communication overhead of post-quantum cryptographic algorithms to the well-known classical ones, with a focus on ML-KEM and ML-DSA in the cryptographic functions. The results indicated that the variants of ML-KEM significantly improved on key generation, encapsulation and decapsulation compared to RSA-4096, with the variants being highly efficient for quantum-safe key establishment. But as the values of the parameters became larger, this benefit was offset by increasingly large increment in the size of the ciphertext and the amount of communication overhead. The key generation and verification rates of ML-DSA variants were also improved over ECDSA-P256, and the sizes of signatures were also significantly larger, while

signing operations were slower. Hence, the most significant deployment constraint was communication overhead, especially for ML-DSA, where the signature expansion in the classical baseline was far less. The security-parameter scaling also showed that higher security parameter leads to higher computation load as well as data volume sent. The results in this work emphasize that the choice of post-quantum migration should not be based on the execution speed only, but on an overall evaluation of the computational cost and efficiency, communication cost, level of security, and system constraints. The deployment of ML-DSA seems to be a challenge in bandwidth-limited applications, while ML-KEM looks particularly promising when it comes to key establishment in a quantum-safe manner. Future studies should focus on a wider range of realistic use cases, including energy consumption, complete implementation of the protocols in realistic communication situations and different hardware configurations, in order to evaluate the results on real network conditions in different parts of the world and under different heterogeneous communication scenarios.

References

1. Adere, K., Hailu, S., Tseghai, A., Haile, G., Tamirat, B., Bitew, W., ... & Alemu, W. (2026). Systematic review on securing IoT systems with post quantum cryptography emerging threats, countermeasures, and future research needs. *Discover Internet of Things*, 6(1), 14.
2. Alagic, G., Alagic, G., Apon, D., Cooper, D., Dang, Q., Dang, T., ... & Smith-Tone, D. (2022). Status report on the third round of the NIST post-quantum cryptography standardization process.
3. Alagic, G., Alperin-Sheriff, J., Apon, D., Cooper, D., Dang, Q., Kelsey, J., ... & Smith-Tone, D. (2020). Status report on the second round of the NIST post-quantum cryptography standardization process. US Department of Commerce, NIST, 2(69).
4. Avanzi, R., Bos, J., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., ... & Stehlé, D. (2024). Module-Lattice-Based Key-Encapsulation Mechanism Standard (No. 203).
5. Aydeger, A., Zeydan, E., Yadav, A. K., Hemachandra, K. T., & Liyanage, M. (2024, October). Towards a quantum-resilient future: Strategies for transitioning to post-quantum cryptography. In *2024 15th International Conference on Network of the Future (NoF)* (pp. 195-203). IEEE.
6. Barbosa, M., Connolly, D., Duarte, J. D., Kaiser, A., Schwabe, P., Varner, K., & Westerbaan, B. (2024). X-wing: The hybrid kem you've been looking for. *Cryptology ePrint Archive*.
7. Barker, W., Polk, W., & Souppaya, M. (2021a). Getting ready for post-quantum cryptography: Exploring challenges associated with adopting and using post-quantum cryptographic algorithms (No. NIST CSWP 15). US Department of Commerce.
8. Barker, W., Souppaya, M., & Newhouse, W. (2021b). Migration to post-quantum cryptography. NIST National Institute of, Standards and Technology and National Cybersecurity, Center of Excellence, 1-15.
9. Behnia, R., & Yavuz, A. A. (2021, December). Towards practical post-quantum signatures for resource-limited internet of things. In *Proceedings of the 37th Annual Computer Security Applications Conference* (pp. 119-130).
10. Beullens, W., D'Anvers, J. P., Hülsing, A. T., Lange, T., Panny, L., de Saint Guilhem, C., & Smart, N. P. (2021). Post-Quantum Cryptography: Current state and quantum mitigation.
11. Dharminder, D., & Mishra, D. (2020). LCPA: Lattice-based conditional privacy preserving authentication in vehicular communication. *Transactions on Emerging Telecommunications Technologies*, 31(2), e3810.
12. Gupta, D. S. (2023). PiLike: Post-quantum identity-based lightweight authenticated key exchange protocol for IIoT environments. *IEEE Systems Journal*, 18(1), 15-23.
13. Halak, B., Gibson, T., Henley, M., Botea, C. B., Heath, B., & Khan, S. (2024). Evaluation of performance, energy, and computation costs of quantum-attack resilient encryption algorithms for embedded devices. *IEEE access*, 12, 8791-8805.
14. Hasan, K. F., Simpson, L., Bae, M. A. R., Islam, C., Rahman, Z., Armstrong, W., ... & McKague, M. (2024). A framework for migrating to post-quantum cryptography: Security dependency analysis and case studies. *IEEE Access*, 12, 23427-23450.
15. Kumari, S., Singh, M., Singh, R., & Tewari, H. (2022a). A post-quantum lattice based lightweight authentication and code-based hybrid encryption scheme for IoT devices. *Computer Networks*, 217, 109327.

16. Kumari, S., Singh, M., Singh, R., & Tewari, H. (2022b). Post-quantum cryptography techniques for secure communication in resource-constrained Internet of Things devices: A comprehensive survey. *Software: Practice and Experience*, 52(10), 2047-2076.
17. McLoughlin, C., Gritti, C., & Samandari, J. (2023, May). Full post-quantum datagram TLS handshake in the Internet of Things. In *International Conference on Codes, Cryptology, and Information Security* (pp. 57-76). Cham: Springer Nature Switzerland.
18. Mendonça, S., Leithardt, V. R. Q., & Crocker, P. A. (2026). Comparative performance dataset: Post-quantum cryptography standards vs. classical cryptographic baselines (Version 1) [Data set]. Mendeley Data. <https://doi.org/10.17632/dvwwc65cf5.1>
19. Newhouse, W., Souppaya, M., Barker, W., Brown, C., Kampanakis, P., Manzano, M., ... & Lee, C. (2023). Migration to post-quantum cryptography: Preparation for considering the implementation and adoption of quantum safe cryptography (No. NIST Special Publication (SP) 1800-38 (Draft)). National Institute of Standards and Technology.
20. Paul, S., & Guerin, E. (2020, September). Hybrid OPC UA: enabling post-quantum security for the industrial internet of things. In *2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)* (Vol. 1, pp. 238-245). IEEE.
21. Paul, S., Kuzovkova, Y., Lahr, N., & Niederhagen, R. (2022, May). Mixed certificate chains for the transition to post-quantum authentication in TLS 1.3. In *Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security* (pp. 727-740).
22. Prajwal, C. P., Mohan, A. S., Reddy, D. N., & Nimmy, K. (2024, June). Quantum-safe authentication protocol leveraging qPUF for industrial internet of things. In *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)* (pp. 1-8). IEEE.
23. Ravi, P., Sundar, V. K., Chattopadhyay, A., Bhasin, S., & Easwaran, A. (2020, October). Authentication protocol for secure automotive systems: Benchmarking post-quantum cryptography. In *2020 IEEE International Symposium on Circuits and Systems (ISCAS)* (pp. 1-5). IEEE.
24. Samandari, J., & Gritti, C. (2023). Post-quantum authentication in the MQTT protocol. *Journal of cybersecurity and privacy*, 3(3), 416-434.
25. Samandari, J., & Gritti, C. (2024, August). Post-quantum authentication and integrity in 3-layer IoT architectures. In *2024 21st Annual International Conference on Privacy, Security and Trust (PST)* (pp. 1-11). IEEE.
26. Schöffel, M., Lauer, F., Rheinländer, C. C., & Wehn, N. (2022). Secure IoT in the era of quantum computers—Where are the bottlenecks?. *Sensors*, 22(7), 2484.
27. Sikeridis, D., Kampanakis, P., & Devetsikiotis, M. (2020). Post-quantum authentication in TLS 1.3: A performance study. *Cryptology ePrint Archive*.
28. Tasopoulos, G., Dimopoulos, C., Fournaris, A. P., Zhao, R. K., Sakzad, A., & Steinfeld, R. (2023, May). Energy consumption evaluation of post-quantum TLS 1.3 for resource-constrained embedded devices. In *Proceedings of the 20th ACM International Conference on Computing Frontiers* (pp. 366-374).
29. Tasopoulos, G., Li, J., Fournaris, A. P., Zhao, R. K., Sakzad, A., & Steinfeld, R. (2022, November). Performance evaluation of post-quantum TLS 1.3 on resource-constrained embedded systems. In *International Conference on Information Security Practice and Experience* (pp. 432-451). Cham: Springer International Publishing.