



Performance and Signature-Size Trade-Offs in Lattice-Based Post-Quantum Threshold Ring Signatures

Arun Kumar B.S.

Department of Computer Science Engineering, Alliance University. Email: arun.bs.kumar@gmail. ORCID iD: 0009-0002-3259-4950

Received:- 14/06/2026, Revised:- 20/07/2026, Accepted:- 04/09/2026, Published:- 21/09/2026

ABSTRACT

Using the LoTRS benchmark dataset, this study has investigated the computational performance and signature-size trade-offs of lattice-based post-quantum threshold ring signatures (LoTRS). The analysis was performed on 14 benchmark configurations, the main evaluation being based upon eight threshold ring-signature settings obtained by setting $(N=32)$ and $(N=100)$ and by varying the threshold. The performance was evaluated by the key generation, key aggregation, signing and verification times and communication and storage overheads by the signature, public-key, secret-key and ring public-key sizes. The figure indicates that the number of threshold and ring size have a significant impact on signing and verification costs. The signing time was 97.81 to 923.60 ms and verification time was 33.95 to 311.00 ms for the main configurations. In comparison, the size of the signature grew comparatively slowly from 42.91 to 54.82 KB. The most significant gains with respect to configuration were in key-aggregation time and ring public-key size. Additional analyses using correlation and regression also showed a high correlation between threshold level and computational cost and a high correlation between ring size and signature size. It is seen that LoTRS provides a relatively moderate growth in the signature size while the computational overhead has increased significantly, thus pointing to a key scalability trade-off for practical implementation of post-quantum cryptography.

Keywords: post-quantum cryptography; lattice-based cryptography; threshold ring signatures; signature size; computational performance

1. Introduction

As quantum computing advances rapidly, fears have grown that traditional public-key cryptographic systems will not be secure in the future. With sufficiently powerful quantum computers, algorithms that rely on integer factorization and discrete logarithms are susceptible to being broken, making it necessary to have cryptographic mechanisms that are secure in the "post-quantum" age. Lattice-based cryptography is one of the few alternatives that has received a lot of attention due to its solid foundations, efficiency and flexibility to build complex cryptographic primitives. There are several areas where lattice assumptions have already been used to power practical post-quantum ring signatures and compact signature schemes, as well as privacy-preserving transaction mechanisms (Alberto Torres et al., 2018; Esgin et al., 2019; Lu et al., 2019). They have also paved the way for deniable and revocable ring-signature constructions for privacy-sensitive applications (Gao et al., 2019; Wen et al., 2023).

Ring signatures are a way of letting a particular member sign for a group without showing the other members. This property allows for anonymous and decentralized authentication, without using a central group manager. Threshold ring signatures generalise this idea to make it so that only a number of co-operating participants produce a valid signature. These kinds of builds are, therefore, able to incorporate signer anonymity and distributed authorization. The theory of threshold ring signatures has been solidified by formal definitions and post-quantum security models, and later on, better efficiency and security notions have been developed (Haque et al., 2022; Munch-Hansen et al., 2021). Moreover, logarithmic-size constructions have been demonstrated to realize threshold functionality using smaller representations of the signatures present a broader interest in the design of threshold ring signatures using codes (Haque et al., 2022; Ren et al., 2022).

What makes lattice-based threshold ring signatures particularly appealing is the fact that they are quantum-resistant, anonymous, and collaborative with a single cryptographic framework. Recent efficient constructions of ring signatures have shown that lattice-based designs give rise to practical instantiations without compromising privacy properties that are strong enough to be considered useful (Yuen et al., 2021). More efficient communication overhead can be achieved with compact constructions based on learning-with-errors assumptions (Chatterjee et al., 2021). There has also been related work on multi-signature over a lattice that proves that distributed signing can be achieved efficiently with limited interaction and structured commitments (Damgård et al., 2022; Fleischhacker et al., 2022). Recent advances have enhanced the efficiency of two-round multi-signatures and simulation techniques (Chen, 2023; Fleischhacker et al., 2023).

Even with these developments, there are still challenges to real-world application, including a balance between security and computational and communication efficiency. The size of the ring and threshold level significantly impact signing time, verification time, and key aggregation costs and a high threshold size will create a larger signature size that will require more transmission and storage costs. Hence, it is important to consider the trade-offs between the computational complexity and signature size in practical lattice-based ring-signature systems (Chatterjee et al., 2021; Haque et al., 2022; Lu et al., 2019).

It studies and compares the performance and signature-size trade-offs of lattice-based threshold ring signatures with various ring-size and threshold parameters. In particular, it considers the signing time, verification time, key aggregation cost, and signature size to see how altering the ring and threshold parameters affects the efficiency of computation and communication.

2. Materials and Methods

2.1 Dataset and Source

The lattice-based post-quantum structured threshold ring signature benchmark dataset with the LoTRS implementation artifact was utilized for this study. The data analyzed was contained in the data.csv file, which lists encoded-size and performance values for 14 different parameter configurations (Jagganath & Saarinen, 2026). The dataset consists of five configurations for ($N=32$), five for ($N=100$) and four auxiliary configurations for ($N=1$). The main threshold ring-signature configurations are ($N \geq 2$) and ($T \geq 2$) while ($T=1$) and ($N=1$) are used as comparative controls. All of the benchmarks were run on a Linux x86_64 box.

2.2 Experimental Parameters

The main experimental variables were ring size (N) and threshold (T). For ($N=32$), threshold values of 1, 4, 8, 16,

and 32 were evaluated, while ($N=100$) used threshold values of 1, 5, 10, 25, and 50. Four other ($N=1$) configurations with thresholds of 2, 4, 8 and 16 were kept as multisignature controls. The performance differences were studied mainly as a function of ring size and threshold, while the underlying lattice dimension was fixed ($d=128$).

2.3 Computational Performance Variables

Key-generation time (`keygen_ms`), key-aggregation time (`kagg_ms`), signing time (`sign_ms`), and verification time (`verify_ms`) were used to evaluate computational efficiency, in milliseconds. The signing and verification measurements were repeated for each principal benchmark configuration. Mean, median, and standard deviation values were used to summarize the signing and verification results, when available. Both average and variance in computational costs were measured using these below measures as the values of (N) and (T) increased.

2.4 Signature and Key Size Variables

The following criteria were used to measure communication and storage overhead: signature size (`sig_bytes`), public-key size (`pk_bytes`), secret-key size (`sk_bytes`), and ring public-key size (`ring_bytes`). Signature size was considered the main size-related measure since this directly impacts the cost of the scheme's communication. The size of the public key was also studied to evaluate the scalability of the full public-key system in terms of ring size and threshold.

2.5 Data Analysis

Computational performance and encoded-size measurements were summarized over configurations using descriptive statistics. Comparisons were made for the different ring size and threshold levels, absolute as well as percentage changes were calculated to investigate the scaling behavior. To analyze the association between ring size, threshold, execution time and signature size, correlation analysis was applied. A regression analysis was performed when it was deemed appropriate to quantify the relationship between (N), (T) and the principal performance outcomes. The primary trade-off analysis involved a comparison of communication to the signature size to see if there were a proportional increase in communication overhead when the computational overhead was increased. As a complement to the examination of key-aggregation time and ring public-key size was also looked at.

3. Results

3.1 Characteristics of Benchmark Configurations

A total of 14 configurations were used in the full benchmark grid, all fixing the lattice size to ($d=128$). Ring sizes included ($N=1$), ($N=32$), and ($N=100$), with threshold values ranging from ($T=1$) to ($T=50$). The main set of LoTRS analysis consisted of eight threshold ring-signature configurations both with ($N \geq 2$) and ($T \geq 2$): four with ($N=32$) and four with ($N=100$). The other six configurations were for ring-signature-only or multisignature control options. In the eight main configurations, the mean signing time varied between 97.81 and 923.60 ms, the mean verification time was between 33.95 and 311.00 ms, and the mean size of the signature was between 42.91 and 54.82 KB. Key-generation time varied only slightly, while key aggregation, signing, verification and ring-key size had significant variability as illustrated in Table 1.

Table 1. Descriptive Summary of LoTRS Benchmark Measurements

Measure	Full Grid ($n = 14$), Mean	Full Grid Range	Main LoTRS ($n = 8$), Mean	Main LoTRS Range
Key generation (ms)	8.34	8.11–8.64	8.35	8.21–8.64

Key aggregation (ms)	37.00	0.57–224.20	62.48	7.17–224.20
Signing time (ms)	206.25	40.44–923.60	305.23	97.81–923.60
Verification time (ms)	73.49	14.15–311.00	110.60	33.95–311.00
Signature size (bytes)	45,554	37,704–54,815	48,741	42,906–54,815
Public-key size (bytes)	9,636	9,636–9,636	9,636	9,636–9,636
Ring-key size (bytes)	7,624,502	19,261–48,161,096	13,147,832	1,237,320–48,161,096

3.2 Signing Performance Across Ring Sizes and Thresholds

Both threshold level and ring size were found to be positively associated with signing time. In the case of ($N=32$), the mean signing time goes from 97.81 ms at ($T=4$) to 268.20 ms at ($T=32$), which is a rise of roughly 174.2%. For ($N=100$), the signing time rose more steeply from ($T=5$), from 226.00 ms to ($T=50$), with 923.60ms, thus by around 308.7%. Figure 1 also shows that the higher the threshold, the more expensive it is to sign, with the ($N=100$) configurations much closer to the 1000ms limit.

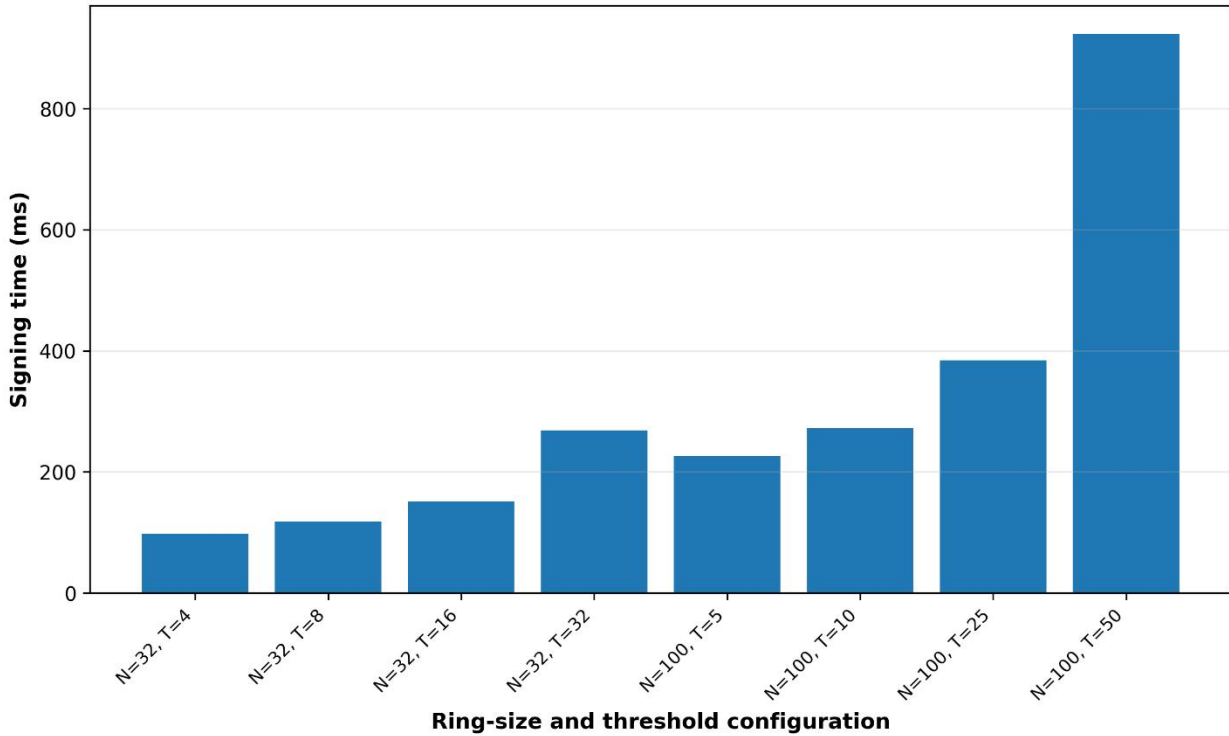


Figure 1. Signing Time Across Ring-Size and Threshold Configurations

3.3 Verification Performance

Verification showed a similar but less extreme scaling pattern than signing. For (N=32), mean verification time increased from 33.95 ms at (T=4) to 81.88 ms at (T=32), an increase of 141.2%. For (N=100), verification time rose from 76.56 ms at (T=5) to 311.00 ms at (T=50), corresponding to a 306.2% increase. Checking was always quicker than authentication in all major configurations. Both the signature and verification costs also went up as the ring and threshold configurations got bigger as indicated in Table 2.

Table 2. Signing and Verification Performance Across Principal LoTRS Configurations

N	T	Signing Mean (ms)	Signing SD (ms)	Signing Median (ms)	Verification Mean (ms)	Verification SD (ms)	Verification Median (ms)
32	4	97.81	49.62	89.92	33.95	1.97	33.73
32	8	118.20	82.06	89.11	42.36	5.32	40.79
32	16	151.40	102.40	118.70	54.56	3.94	53.22
32	32	268.20	180.50	203.50	81.88	5.46	79.94
100	5	226.00	115.00	183.90	76.56	4.35	75.44

10 0	1 0	272.70	137.90	242.70	102.70	3.67	102.20
10 0	2 5	383.90	226.80	295.80	181.80	7.99	180.40
10 0	5 0	923.60	713.20	718.70	311.00	11.07	310.70

3.4 Key Generation and Key Aggregation Performance

The key generation time was very stable for all the main configurations, ranging from 8.21 to 8.64 ms. By contrast, there was a significant increase in key-aggregation cost as the threshold was raised. In the case of $N=32$, the time taken for aggregation increased from 7.17 ms to 49.57 ms, which is about 591.4%. For ($N=100$), it rose from 24.10 ms at ($T=5$) to 224.20 ms at ($T=50$), an increase of approximately 830.3%. As shown in Figure 2, key generation is almost constant and key-aggregation time increases significantly with the threshold.

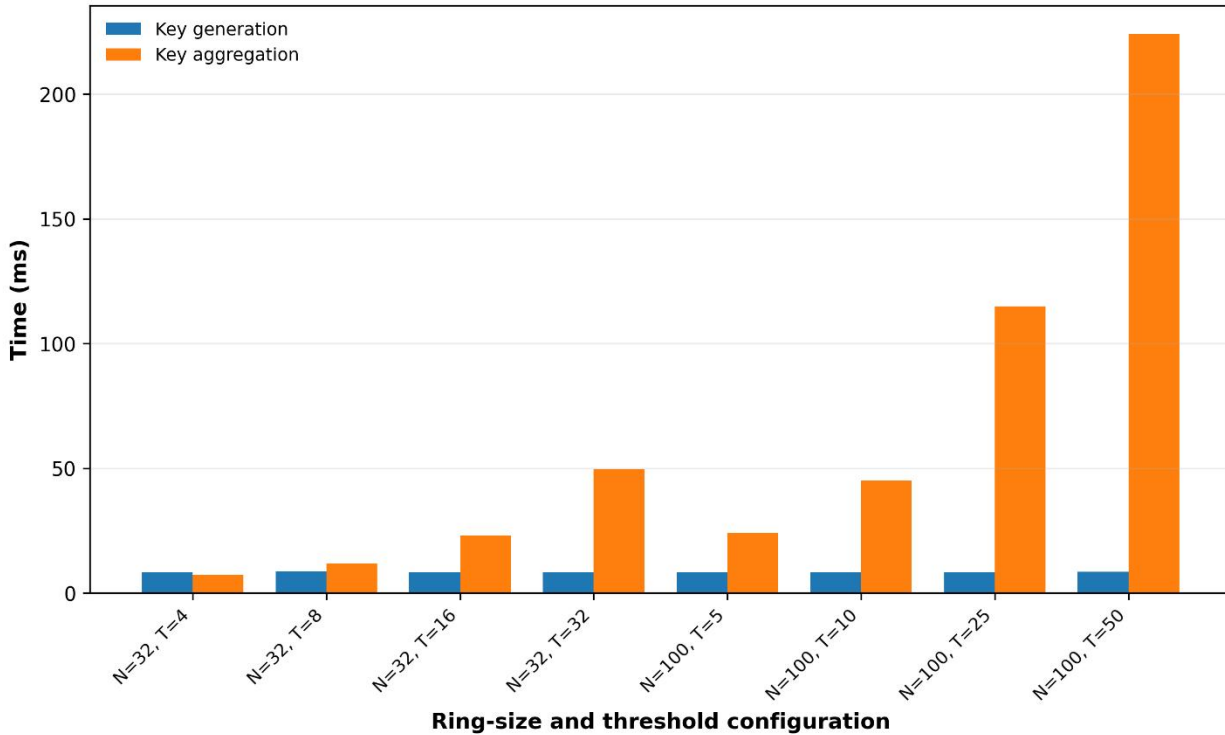


Figure 2. Key Generation and Key Aggregation Performance Across Configurations

3.5 Signature-Size Characteristics

The size of the signature was correlated with the size of the ring, but only changed in a moderate way with the threshold in the same-size ring. For ($N=32$), signature size ranged from 42,906 bytes at ($T=4$) to 44,124 bytes at ($T=32$). For ($N=100$), it ranged from 53,217 bytes at ($T=5$) to 54,815 bytes at ($T=50$). The personal-key and secret-key sizes were maintained at 9,636 and 32 bytes, respectively. Conversely, the size of the entire ring public-key design quadrupled as (N) and (T) grew, to about 45.93 MiB in the biggest setup. As indicated in Table 3, the growth of signature size was not quite as large as the growth of the complete ring-key structure.

Table 3. Signature and Key Sizes Across Principal LoTRS Configurations

N	T	Secret Key (bytes)	Public Key (bytes)	Ring Public Key (MiB)	Signature Size (bytes)	Signature Size (KiB)
32	4	32	9,636	1.18	42,906	41.90
32	8	32	9,636	2.35	43,254	42.24
32	16	32	9,636	4.70	43,745	42.72
32	32	32	9,636	9.41	44,124	43.09
100	5	32	9,636	4.59	53,217	51.97
100	10	32	9,636	9.19	53,586	52.33
100	25	32	9,636	22.96	54,282	53.01
100	50	32	9,636	45.93	54,815	53.53

3.6 Effect of Ring Size and Threshold on Signature Size

The impact of threshold growth on signature size was significantly lower than its impact on the computational performance. In the (N=32) series, signature size was increased by just 2.84% while signing time increased by 174.2% when the threshold was increased from 4 to 32. Likewise, a signature's size rose by ~3.00% and signing time by 308.7% when the value of (T) was raised from 5 to 50 for (N=100). The correlation of ring size with signature size was ($r=0.995$) while the correlation of threshold level with signature size was ($r=0.346$). This can be seen in Figure 3, where the within-ring signature-size growth is relatively shallow, but a distinction between the (N=32) and (N=100) configurations exists.

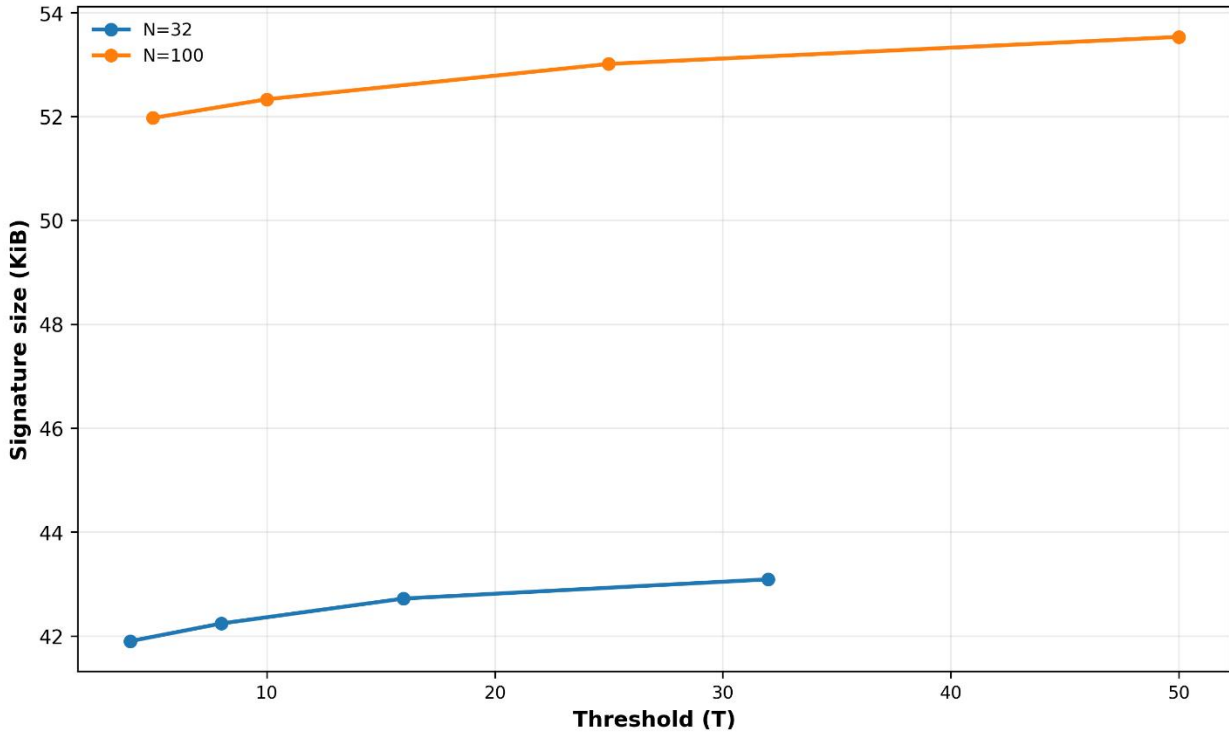


Figure 3. Signature Size Across Ring-Size and Threshold Configurations

3.7 Performance–Signature-Size Trade-Off

The trade-off analysis revealed that the increase in computational overhead was significantly higher than that of the signature size with an increase in threshold complexity. At $(N=32, T=4)$ the scheme took 2.33 ms of signing time per KiB of signature, and 6.22 ms/KiB at $(N=32, T=32)$. For $(N=100)$, this ratio increased from 4.35 ms/KiB at $(T=5)$ to 17.25 ms/KiB at $(T=50)$. The same trend was noticed in verification cost. In this way, the bigger configurations were subject to significantly higher execution costs without a proportional increase in the final signature. The configuration with the highest computation costs $(N=100, T=50)$ had the lowest signature (53.53 KiB) but took 923.60 ms to sign and 311.00 ms to verify, as shown in Table 4.

Table 4. Performance–Signature-Size Trade-Off Across Principal Configurations

N	T	Signing (ms)	Verification (ms)	Signature (KiB)	Signing ms/KiB	Verific ation ms/Ki B	Ring PK (MiB)
32	4	97.81	33.95	41.90	2.33	0.81	1.18
32	8	118.20	42.36	42.24	2.80	1.00	2.35
32	16	151.40	54.56	42.72	3.54	1.28	4.70

32	32	268.20	81.88	43.09	6.22	1.90	9.41
100	5	226.00	76.56	51.97	4.35	1.47	4.59
100	10	272.70	102.70	52.33	5.21	1.96	9.19
100	25	383.90	181.80	53.01	7.24	3.43	22.96
100	50	923.60	311.00	53.53	17.25	5.81	45.93

3.8 Overall Scalability Pattern

In general, threshold level was positively correlated with signing time ($r=0.875$), verification time ($r=0.849$), and key-aggregation time ($r=0.893$). Performance in signing and verifying also had a strong relationship ($r=0.980$). Signature size was found to be most related to ring size ($r=0.995$). In simple regression models with (N) and (T) as predictors, the models accounted for about 90.9 percent of the variation in signing time, 93.2 percent of the variation in verification time and 99.9 percent of the variation in signature size in the eight major configurations. Such models were taken to be descriptive scaling relations but not population-level inferential estimates due to the small number of benchmark configurations.

The general trend is thus that the rise in (N) and more so (T) significantly increases computational and ring-key overhead, but final signature size increases much more slowly. This trade-off is visually summarized in Figure 4 by the dramatic increase in the cost of signing and verification in comparison with the relatively limited increase in signature size.

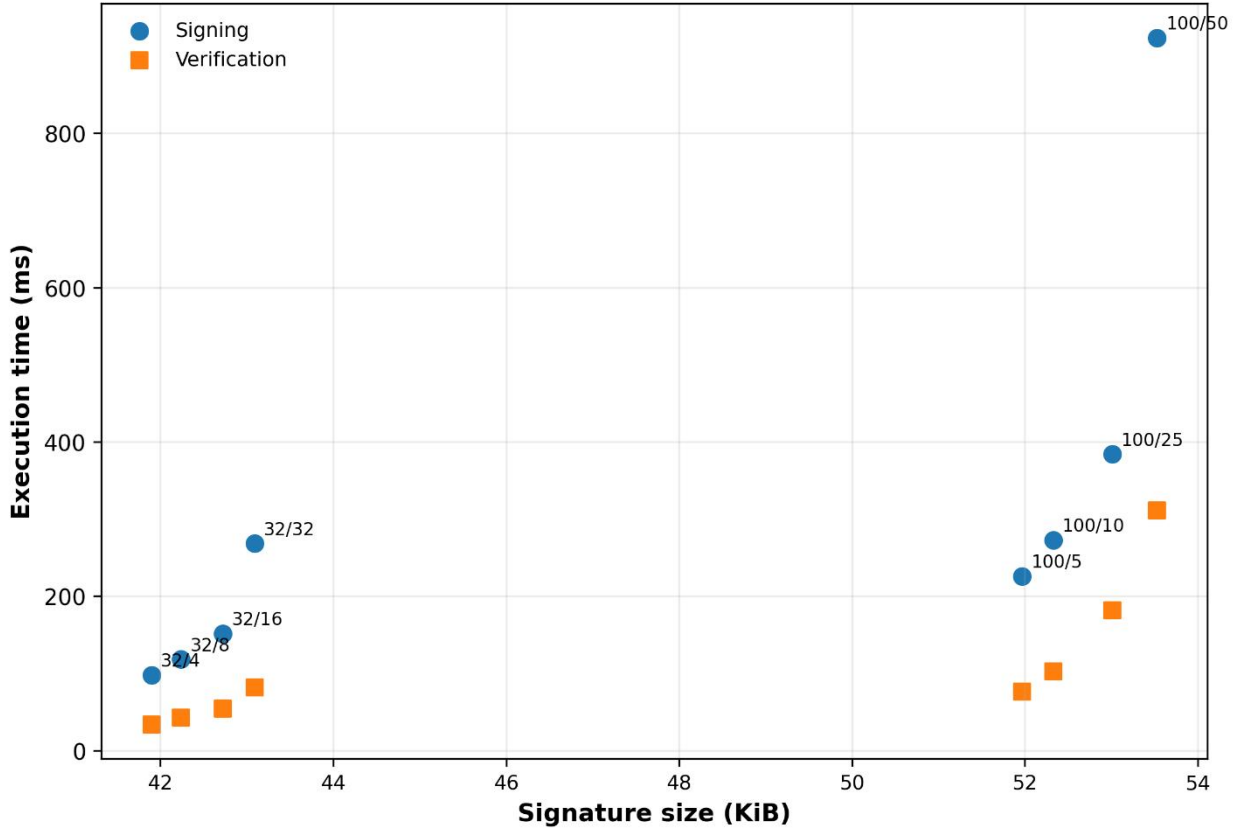


Figure 4. Signing and Verification Performance Versus Signature Size

4. Discussion

The results show a clear computational trade-off due to the increase in ring size and threshold in the LoTRS benchmark. Signing time exhibited the greatest growth with the larger ($N=100$) configurations, and verification time followed suit but at a lower cost. The signature size, too, grew as the configurations grew; but the growth was much lower than the growth of the signing and verification costs. This trend aligns with recent research that focuses on compact threshold and ring-signature designs and that is mindful of the deployment constraints imposed by computational overhead (Wang et al., 2024; Wu et al., 2025). The results are also comparable with the design vision of LoTRS that aims to create efficient lattice-based threshold ring signatures with practicality for post-quantum applications (Jagganath et al., 2026).

Signing, verification and ring public-key costs were significantly impacted by the increase in ring size. The more public-key components and cryptographic relations to process, the more arithmetic operations and memory are required, and larger rings need more. This aligns with lattice-based ring-signature research, where ring structure and proof complexity are crucial factors for efficiency (Zhang et al., 2025). More generally, lattice-based digital signatures also suffer from such considerations; the size of the lattices and the degree of polynomial operations directly affect the computational complexity (Ducas et al., 2018). So, efficient lattice implementations rely on parameter and arithmetic optimisation as well as on the security strength (Albrecht et al., 2018).

Signing and verification time was especially sensitive to threshold size. The cost of producing and validating signatures grew as (T) grew, due to the extra computation/signature aggregation needed for the signer. This is aligned with the recent research work on threshold signatures, where reducing communication and limiting interaction are significant design considerations because threshold interactions can incur significant overhead (Boschini et al., 2025; Chairattana-Apirom et al., 2025). The observed increase in aggregation cost also mirrors the difficulty of aggregating multiple contributions of cryptographic operations in a structured threshold setting.

One of the important conclusions was that the expansion of the signature is much slower than the increase in the computational cost. This indicates that raising (N) and (T) comes with a higher cost in processing time than a higher cost in final signature transmission size. This kind of behavior is good if the communication bandwidth is

limited while computational resources are relatively plentiful. The pursuit of compactness continues to be a key goal for contemporary lattices and ring-signatures, evident in logarithmic-size threshold constructions and sign-then-prove designs (Wang et al., 2024; Zhang et al., 2025). Post-quantum schemes like qTESLA and the masking-based lattice constructions exhibit a similar focus on compactness and ease of implementation (Bindel et al., 2018; Esgin et al., 2024).

The experimental results suggest that lattice-based threshold ring signatures can be used in applications where privacy and distributed authorization are both needed. It can be used for decentralized platforms, privacy-preserving authentication, collaborative access control, blockchain use cases, and electronic voting. To show the practical relevance of this cryptographic model, rejection-free threshold ring signatures have already been proposed for receipt-free cloud e-voting (Wu et al., 2025). Generally, lattice-based cryptography is appealing due to its post-quantum security guarantees and flexibility of implementation. Recent studies also focused on implementation aspects and indicate the need for optimizing lattice signatures for resource-poor systems such as IoT (Rowank, 2025).

The drawback of the present study is the comparatively small number of benchmark configurations, use of only one implementation of LoTRS, and measurements taken on one computational environment. As such, these observed relationships do not necessarily represent the universal performance features of all lattice-based threshold ring signatures, but instead the scaling patterns inherent in their specific implementation. Absolute execution times might vary depending on the hardware architecture, the compiler settings, the set of parameters, or the implementation optimizations.

Further research would be useful to test larger ring sizes and other thresholds to get more detailed scalability profiles. A comparison of the efficiency of LoTRS to other lattice-based threshold signatures, ring signatures, and post-quantum digital signatures would also help to better understand the efficiency of LoTRS. Besides runtime, researchers should measure memory use, energy use, and how well parallelization is used, and how resistant the implementation is to side-channel attacks. Analyses like these would help to characterize more fully the practicality of use in distributed and resource-constrained post-quantum settings.

5. Conclusion

This work provides a comprehensive analysis of the performance and signature size trade-offs of lattice-based threshold ring signatures on various ring size and threshold settings, using the benchmark dataset LoTRS. The analysis revealed that the computational effort rises significantly with the ring size and threshold, and the biggest impact on computation time is on signing and verification time. The increase in signing time was more pronounced than the increase in verification time, especially for the large ($N=100$) sizes, which suggests that with higher thresholds, the computational cost of signature generation is greater. The signature size, however, grew fairly slowly, showing that the communication overhead of the final signature was fairly stable even as its computational complexity grew significantly. However, ring public-key size increased significantly as the size of the parameters increased, pointing to an extra storage and transmission consideration for practical application. As a whole, the results have shown that a lattice-based threshold RS can be a practical distributed authorization system that offers a desirable balance of computational efficiency, signer anonymity, and post-quantum security, though careful system configuration is required to achieve this. For some applications that require low latency, the smaller and moderate configurations may be more suitable, while for some applications that require higher anonymity sets or threshold participation, the larger configurations may be suitable. Benchmarking should be continued for more parameters, hardware platform, and alternative post-quantum threshold signature constructions, and additional security aspects at the implementation level should be taken into account, as should memory consumption and energy efficiency.

References

1. Alberto Torres, W. A., Steinfeld, R., Sakzad, A., Liu, J. K., Kuchta, V., Bhattacharjee, N., Au, M. H., & Cheng, J. (2018). Post-Quantum One-Time Linkable Ring Signature and Application to Ring Confidential Transactions in Blockchain (Lattice RingCT v1.0). In W. Susilo & G. Yang (Eds.), *Information Security and Privacy* (Vol. 10946, pp. 558–576). Springer International Publishing. https://doi.org/10.1007/978-3-319-93638-3_32
2. Albrecht, M. R., Curtis, B. R., Deo, A., Davidson, A., Player, R., Postlethwaite, E. W., Virdia, F., & Wunderer, T. (2018). Estimate All the {LWE, NTRU} Schemes! In D. Catalano & R. De Prisco (Eds.), *Security and Cryptography for Networks* (Vol. 11035, pp. 351–367). Springer International Publishing.

https://doi.org/10.1007/978-3-319-98113-0_19

3. Bindel, N., Akleylek, S., Alkim, E., Barreto, P. S., Buchmann, J., Eaton, E., Gutoski, G., Kramer, J., Longa, P., & Polat, H. (2018). Submission to NIST’s post-quantum project: Lattice-based digital signature scheme qTESLA. https://drive.google.com/file/d/1ay1TbVXUJ0ZJAFkR_yDSq1nMGtZ5XhXu/view
4. Boschini, C., Kaviani, D., Lai, R. W., Malavolta, G., Takahashi, A., & Tibouchi, M. (2025). Ringtail: Practical two-round threshold signatures from learning with errors. 2025 IEEE Symposium on Security and Privacy (SP), 149–164. <https://ieeexplore.ieee.org/abstract/document/11023447/>
5. Chairattana-Apirom, R., Tessaro, S., & Zhu, C. (2025). Partially Non-interactive Two-Round Lattice-Based Threshold Signatures. In K.-M. Chung & Y. Sasaki (Eds.), *Advances in Cryptology – ASIACRYPT 2024* (Vol. 15487, pp. 268–302). Springer Nature Singapore. https://doi.org/10.1007/978-981-96-0894-2_9
6. Chatterjee, R., Garg, S., Hajiabadi, M., Khurana, D., Liang, X., Malavolta, G., Pandey, O., & Shiehian, S. (2021). Compact Ring Signatures from Learning with Errors. In T. Malkin & C. Peikert (Eds.), *Advances in Cryptology – CRYPTO 2021* (Vol. 12825, pp. 282–312). Springer International Publishing. https://doi.org/10.1007/978-3-030-84242-0_11
7. Chen, Y. (2023). DualMS : Efficient Lattice-Based Two-Round Multi-signature with Trapdoor-Free Simulation. In H. Handschuh & A. Lysyanskaya (Eds.), *Advances in Cryptology – CRYPTO 2023* (Vol. 14085, pp. 716–747). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-38554-4_23
8. Damgård, I., Orlandi, C., Takahashi, A., & Tibouchi, M. (2022). Two-Round n -out-of- n and Multi-Signatures and Trapdoor Commitment from Lattices. *Journal of Cryptology*, 35(2), 14. <https://doi.org/10.1007/s00145-022-09425-3>
9. Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schwabe, P., Seiler, G., & Stehlé, D. (2018). Crystals-dilithium: A lattice-based digital signature scheme. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 238–268.
10. Esgin, M. F., Espitau, T., Niot, G., Prest, T., Sakzad, A., & Steinfeld, R. (2024). Plover : Masking-Friendly Hash-and-Sign Lattice Signatures. In M. Joye & G. Leander (Eds.), *Advances in Cryptology – EUROCRYPT 2024* (Vol. 14657, pp. 316–345). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-58754-2_12
11. Esgin, M. F., Steinfeld, R., Sakzad, A., Liu, J. K., & Liu, D. (2019). Short Lattice-Based One-out-of-Many Proofs and Applications to Ring Signatures. In R. H. Deng, V. Gauthier-Umaña, M. Ochoa, & M. Yung (Eds.), *Applied Cryptography and Network Security* (Vol. 11464, pp. 67–88). Springer International Publishing. https://doi.org/10.1007/978-3-030-21568-2_4
12. Fleischhacker, N., Herold, G., Simkin, M., & Zhang, Z. (2023). Chipmunk: Better Synchronized Multi-Signatures from Lattices. *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, 386–400. <https://doi.org/10.1145/3576915.3623219>
13. Fleischhacker, N., Simkin, M., & Zhang, Z. (2022). Squirrel: Efficient Synchronized Multi-Signatures from Lattices. *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, 1109–1123. <https://doi.org/10.1145/3548606.3560655>
14. Gao, W., Chen, L., Hu, Y., Newton, C. J. P., Wang, B., & Chen, J. (2019). Lattice-based deniable ring signatures. *International Journal of Information Security*, 18(3), 355–370. <https://doi.org/10.1007/s10207-018-0417-1>
15. Haque, A., Krenn, S., Slamanig, D., & Striecks, C. (2022). Logarithmic-Size (Linkable) Threshold Ring Signatures in the Plain Model. In G. Hanaoka, J. Shikata, & Y. Watanabe (Eds.), *Public-Key Cryptography – PKC 2022* (Vol. 13178, pp. 437–467). Springer International Publishing. https://doi.org/10.1007/978-3-030-97131-1_15
16. Jagganath, N., Esgin, M. F., Steinfeld, R., Sakzad, A., Saarinen, M.-J. O., & Liu, D. (2026). LoTRS: Practical Post-Quantum Structured Threshold Ring Signatures from Lattices. *Cryptology ePrint Archive*. <https://eprint.iacr.org/2026/974>
17. Jagganath, N., & Saarinen, M.-J. (2026). LoTRS: Practical Post-Quantum Structured Threshold Ring Signatures from Lattices. <https://doi.org/10.5281/zenodo.22807126>
18. Lu, X., Au, M. H., & Zhang, Z. (2019). Raptor: A Practical Lattice-Based (Linkable) Ring Signature. In R. H. Deng, V. Gauthier-Umaña, M. Ochoa, & M. Yung (Eds.), *Applied Cryptography and Network Security* (Vol. 11464, pp. 110–130). Springer International Publishing. https://doi.org/10.1007/978-3-030-21568-2_6

19. Munch-Hansen, A., Orlandi, C., & Yakoubov, S. (2021). Stronger Notions and a More Efficient Construction of Threshold Ring Signatures. In P. Longa & C. Ràfols (Eds.), *Progress in Cryptology – LATINCRYPT 2021* (Vol. 12912, pp. 363–381). Springer International Publishing. https://doi.org/10.1007/978-3-030-88238-9_18
20. Ren, F., Xiu, H., Ji, C., Zheng, D., & Wu, Z. (2022). A New Code-Based Linkable Threshold Ring Signature Scheme. *Wireless Communications and Mobile Computing*, 2022(1), 3150714. <https://doi.org/10.1155/2022/3150714>
21. Rowank, B. (2025). Efficient CRYSTALS-Dilithium: Enhancing Speed and Security for IoT in the Quantum Age [PhD Thesis, Nara Institute of Science and Technology]. <https://naist.repo.nii.ac.jp/record/2001631/files/R019498.pdf>
22. Wang, H., Tao, Y., & Zhang, R. (2024). Threshold ring signature: Generic construction and logarithmic size instantiation. *Cybersecurity*, 7(1), 46. <https://doi.org/10.1186/s42400-024-00233-9>
23. Wen, J., Bai, L., Yang, Z., Zhang, H., Wang, H., & He, D. (2023). LaRRS: Lattice-based revocable ring signature and its application for VANETs. *IEEE Transactions on Vehicular Technology*, 73(1), 739–753.
24. Wu, C., Zhou, Y., Zhang, F., Du, Y., & Lin, Q. (2025). An efficient rejection-free threshold ring signature from lattices and its application in receipt-free cloud e-voting. *Computer Standards & Interfaces*, 94, 104008.
25. Yuen, T. H., Esgin, M. F., Liu, J. K., Au, M. H., & Ding, Z. (2021). DualRing: Generic Construction of Ring Signatures with Efficient Instantiations. In T. Malkin & C. Peikert (Eds.), *Advances in Cryptology – CRYPTO 2021* (Vol. 12825, pp. 251–281). Springer International Publishing. https://doi.org/10.1007/978-3-030-84242-0_10
26. Zhang, R., Luo, P., & Huang, Q. (2025). SPRING: Sign-then-prove ring signatures from lattices with compactness, extensibility, and efficiency. *Computer Standards & Interfaces*, 104084.