Advancements in Distributed RSA Key Generation: Enhanced Biprimality Tests

ChihYun Chuang[1], IHung Hsu[1], and TingFang Lee[2]

[1] AMIS, Taipei, Taiwan
{chihyun,glen}@maicoin.com
[2] Division of Biostatistics, NYU Langone Health
Ting-Fang.Lee@nyulangone.org

Abstract. This work re-evaluates the soundness guarantees of the Boneh-Franklin biprimality test (2001) for Blum integers. Under the condition $\gcd(pq, p+q-1) = 1$, we show that the test accepts a non-RSA modulus with probability at most 1/4. This is a refinement of the previously established 1/2 bound and holds for all cases except the specific instance where $p = q = 3$. We further demonstrate that this 1/4 bound is tight, thereby halving the number of test iterations required to achieve equivalent soundness. This directly reduces computational and communication overhead in distributed RSA generation protocols.

Additionally, we propose a generalized biprimality test based on the Lucas sequence. In the worst case, the acceptance probability of the proposed test is at most $1/4 + 1.25/(p_{\min} - 3)$, where $p_{\min}$ is the smallest prime factor of N. To validate our approach, we implemented the variant Miller-Rabin test, the Boneh-Franklin test, and our proposed test, performing pairwise comparisons of their effectiveness. Both theoretical analysis and simulations indicate that the proposed test is generally more efficient than the Boneh-Franklin test in detecting cases where N is not an RSA modulus. Furthermore, this test is applicable to generating RSA moduli for arbitrary odd primes.

A distributed RSA modulus verification protocol that incorporates our test is also introduced. The protocol is secure against semi-honest adversaries for general odd primes. For Blum integers, it also offers security against malicious adversaries. We analyze its efficiency and compatibility with existing distributed RSA protocols against semi-honest adversaries, including those of Boneh-Franklin and Burkhardt et al. (CCS 2023). Our protocol offers competitive performance while enhancing soundness and generality in cryptographic applications [3].

1 Introduction

The RSA cryptosystem [38] remains a cornerstone of public key cryptography. Traditionally, RSA key generation involves creating two large, distinct secret primes, p and q, whose product $N = pq$ forms the public modulus. However,

[3] https://eprint.iacr.org/2024/2072

generating these keys centrally introduces a single point of failure. Multi-Party Computation (MPC) offers a robust solution by enabling multiple parties to collectively compute N using their private inputs (i.e., shares of p and q) while preserving the confidentiality of these inputs. MPC-based RSA generation has become a foundational tool for constructing advanced cryptographic primitives, such as threshold homomorphic encryption [24, 27], time-lock puzzles [1, 32, 39], accumulators [6, 9, 31], and verifiable delay functions [8, 18, 22, 28, 36, 41].

The primary objective in distributed RSA modulus generation is to devise a secure protocol for n parties, resilient against up to $t < n$ colluding adversaries. The protocol should output a random and valid RSA modulus $N = pq$, where p, q are distinct primes of a specified size, such that an adversary learns nothing beyond N, ensuring the privacy of p and q. Such protocols typically involve two phases: **(a) Prime Candidate Sieving**: participants generate a potential RSA modulus N that does not divide by a prime less than a predetermined integer $p_{\min}$; and **(b) Biprimality test**: the candidate N is repeatedly tested by a biprimality test. If N is rejected by the biprimality test, then the process starts over. Current state-of-the-art sieving techniques often employ the Chinese Remainder Theorem (CRT) to efficiently generate candidates N free of small prime factors [16, 40]. For biprimality testing, variants of the Miller-Rabin primality test (cf. [15, Section 3.2]) and Boneh-Franklin's biprimality test (cf. Theorem 1) have been commonly used.

A notable limitation is that MPC protocols for these tests often restrict candidate primes to $p \equiv q \equiv 3 \pmod 4$. The Boneh-Franklin test, in its original analysis [10], has a worst-case soundness error (i.e., probability of accepting a non-RSA modulus) of at most $1/2$. While the Miller-Rabin primality test exhibits a worst-case error of $1/4$ for testing individual numbers [14, 37], its average-case performance for testing prime candidates (e.g., for p, q both 1024-bit) can achieve errors below 2^{-67} in just two iterations [19,20]. Deriving similarly strong average-case bounds for the Boneh-Franklin test remains an open question [17, 20]. Consequently, relying on its worst-case bound means the Boneh-Franklin test requires substantially more iterations (e.g., 67) to achieve comparable assurance, increasing verification costs. Burkhardt et al. [15] demonstrated enhanced efficiency using a variant Miller-Rabin test[4], though its single MPC execution can be costlier than one Boneh-Franklin iteration.

However, leveraging the Miller-Rabin test studied by Burkhardt et al.'s, in practical distributed RSA generation faces challenges. Firstly, it often assumes p and q are of equal bit-length [15, Input assumptions]. Secondly, its strong average-case soundness relies on p and q being chosen uniformly at random, an assumption not met by most efficient distributed RSA generation algorithms [10, 16,17,20,23,40], which typically produce p and q from more complex distributions (e.g., sums of uniform random variables). Thus, directly applying these average-case results in practical protocols requires careful justification.

[4] The variant Miller-Rabin test they used is a special case of the original Miller-Rabin test. See Section 5.1.

While extensive research has advanced prime candidate sieving, biprimality testing within distributed protocols has seen comparatively less focus on new alternatives or broader applicability. This paper addresses these gaps by investigating the following questions:

Which existing test, Boneh-Franklin or variant Miller-Rabin, offers superior advantages for distributed RSA moduli generation? More importantly, can we design more efficient or general biprimality tests?

1.1 Our contribution

Our work introduces significant advancements in biprimality testing, focusing on improved efficiency, tighter security analyses, and relaxed constraints for RSA modulus generation. We also provide a high-level technical overview in Section 1.2, before formalizing our results. Our primary contributions are:

A Refined Analysis of the Boneh-Franklin Test. We demonstrate that, in the worst-case scenario, the probability of the Boneh-Franklin test accepting a non-RSA modulus is 1/4, a tighter bound than the previously established 1/2. The theoretical underpinnings for this improved bound are detailed in our Technical Overview. Crucially, we identify the necessary and sufficient conditions on the factors p and q that lead to this worst case (cf. Corollary 1) and show that infinitely many such pairs (p, q) exist.

A Novel Lucas Biprimality Test. Inspired by classical Lucas primality tests, we propose a new biprimality test. For odd integers p, q, let $N = pq$. We define an exponent[5] $e_4 := (p + [\frac{-1}{p}])(q + [\frac{-1}{q}])/4$, where $\left[\frac{\cdot}{\cdot}\right]$ is the Jacobi symbol, such that $\gcd(N, e_4) = 1$. For a Lucas sequence U_k with the initial condition $U_0 = 0, U_1 = 1$, parameters P, Q, and discriminant $D := P^2 - 4Q$, we prove that there exists a pair P, Q satisfying $\gcd(N, 2QD) = 1$, $\left[\frac{-D}{p}\right] = \left[\frac{-D}{q}\right] = -1$, and $\left[\frac{Q}{N}\right] = 1$ for which the term $U_{e_4} \not\equiv 0 \pmod N$ if and only if N is not a valid RSA modulus.

Advantages of the Proposed Lucas Test. (1) Enhanced Detection Efficiency: Our study, supported by both theoretical analysis (cf. Section 5.2) and empirical results (cf. Section 5.1, Table 2), reveals that our Lucas test generally outperforms the Boneh-Franklin test by detecting non-RSA moduli with fewer iterations, despite both tests having nearly identical computational complexity per iteration (cf. Table 3). This efficiency reduces overall computational and communication costs. Additionally, Table 2 indicates that, when p and q are selected from a specific distribution, the Lucas test likely offers better security than the variant Miller-Rabin test. (2) Relaxed Prime Constraints: A significant advantage of our Lucas test is its ability to operate without the common restriction

[5] This formulation aligns with the Boneh-Franklin $e_4 = (p-1)(q-1)/4$ when $p \equiv q \equiv 3 \pmod 4$.

$p \equiv q \equiv 3 \pmod 4$ often imposed in distributed RSA key generation protocols. This broadens its applicability. While some systems, like the KMOV variant by Boudabra et al. [12], target $p \equiv q \equiv 1 \pmod 4$, our work contributes to a more theoretically complete framework for biprimality testing across diverse prime types.

Secure Protocols and Efficiency Gains. (1) Semi-Honest Security: We propose a Lucas-based protocol secure against semi-honest adversaries (cf. Theorem 3). The main challenge, ensuring privacy via indistinguishable views for the simulator, is addressed using refined methods, particularly for cases beyond $p \equiv q \equiv 3 \pmod 4$ where direct application of Boneh-Franklin's strategy is insufficient. (2) Malicious Security and Protocol Optimization: For the malicious setting, we adapt the framework of Chen et al. [16, Protocol 5.2] by integrating our Lucas test, initially for the $p \equiv q \equiv 3 \pmod 4$ case. Our analysis provides a more detailed proof within this framework and identifies potential technical refinements to Chen et al.'s protocol (cf. Section 6.11). Notably, our tighter $1/4$ worst-case soundness error for the Boneh-Franklin test directly translates to improved efficiency in such protocols, reducing the required number of iterations from approximately $2.5s$ to $\lceil 1.475s \rceil$ to achieve a failure probability of less than 2^{-s} (cf. Section 4.4), where s is the security parameter and $\lceil \cdot \rceil$ is the ceiling function.

Comprehensive Comparison and Validation. A comparative summary of the Miller-Rabin, Boneh-Franklin, and our Lucas test is provided in Table 1. Our Lucas-based protocol for $p \equiv q \equiv 3 \pmod 4$ cases is highly competitive. For other prime congruences, our Lucas test is recommended for generating RSA moduli due to its relaxed constraints and robust detection. Rigorous empirical analysis, including benchmarking against competing methods, was performed to validate our proposed test. The implementation code is publicly available for reproducibility at [6].

1.2 Technical Overview

First, we provide a high-level description of the biprimality tests considered in this paper. Let H and G be two sets whose definitions depend on p and q. While H is a subset of G, their distinct definitions are key that we have $G = H$ if and only if p and q are distinct primes. The two sets can then be utilized to construct a biprimality test. By randomly selecting an element g from G and verifying whether $g \in H$. If $g \notin H$, then N is not an RSA modulus. This value $\beta = |H|/|G|$ can be considered the soundness error for non-RSA moduli $N = pq$.

Our proof strategy for establishing soundness is analogous for both the Boneh-Franklin refinement and our proposed Lucas test: we compute the cardinalities of G and H, and then show that for any odd integers p, q, the set H always a subset of G. Consequently, if $|G| = |H|$, then one has $G = H$.

[6] https://github.com/lukakusilk/Three-biprimality-test-comparison

Table 1: Ranking Features of Three Tests: A Comparative Overview

Method	Boneh-Franklin	Variant Miller-Rabin	Proposed test
The worst case excluding special conditions	$1/2 \to 1/4$	$1/4$	$1/4 + 1.25/(p_{\min} - 3)$
Exceptional	$p = q = 3$	$p, q \leq 9$	$p_{\min} < 11$
Extra assumption	$\gcd(pq, e_4) = 1$	equal-length[1]	$\gcd(pq, e_4) = 1$[2]
Detecting of non-RSA moduli	*3*	*2*	*1*
MPC Protocol efficiency	*1*	*3*	*1*
Local computation efficiency	*1*	*3*	*2*
Leakage	*No*	*No*	Blum:*No*[3]
RSA Moduli Type	Blum	Blum	Arbitrary

The numbers in the table represent rankings. The section under the bold heading presents a comparison of the protocols in the semi-honest model (cf. Section 6.10). **The worst case excluding special conditions** is derived from Theorem 1, 2, and Lemma 14. **Exceptional** means that the exclusion of the worst-case scenario. **Extra assumption** means the additional conditions required by each test. The ranking for **Detecting of non-RSA moduli** comes from the Table 2. The ranking for **MPC protocol efficiency** comes from the Section 5.3. Finally, the ranking for **Local computation efficiency** is based on the comparison of local computations in Section 5.3, and Protocol 5, 6, and 7. For a discussion of **Leakage**, please refer to the last paragraph of Section 4.2.
The Blum moduli in the **RSA Moduli Type** require the condition $p \equiv q \equiv 3 \pmod 4$. $p_{\min}$ is the smallest prime factor of $N = pq$.
[1] The condition of equal-length for primes p, q implies that $\gcd(pq, e_4) = \gcd(pq, p + q - 1) = 1$.
[2] When considering $p \equiv 1 \pmod 4$ (or $q \equiv 1 \pmod 4$), we additionally assume that p (or q) is not a perfect square.
[3] If N is not a Blum integer, then this leakage is discussed in Remark 2.

Next, let us explain why the soundness error in the worst-case can be improved. In the original Boneh-Franklin's proof [10, Lemma 4.1], the condition $\gcd\left(pq, p + q - 1\right) = \gcd\left(pq, e_4\right) = 1$ was not assumed. However, this omission allowed for the existence of non-RSA moduli N, (i.e., $p = p_1^{d_1}, q = p_2^{d_2}$, $d_1 > 0$, and $q \equiv 1 \pmod{p_1^{d_1 - 1}}$, where p_1, p_2 are distinct primes) which would still pass the test. To address this issue, the assumption $\gcd\left(pq, p + q - 1\right) = 1$ was introduced to exclude these pathological cases[7]. However, in the original proof (i.e., they proved $H = \mathrm{BF}(N, e_4) \subsetneq G(N) = G$), the condition $\gcd\left(pq, p + q - 1\right) = 1$ was not easy to apply directly. Here

$$\mathrm{BF}(N, e_4) := \left\{ g \in \mathbb{Z}_N^\times \mid g^{e_4} \equiv \pm 1 \pmod N \right\} \subset G(N) := \left\{ g \in \mathbb{Z}_N^\times \;\middle|\; \left[\frac{g}{N}\right] = 1 \right\}.$$

To effectively leverage the conditions $\gcd\left(pq, p + q - 1\right) = 1$, we adopted an alternative approach based on two key insights. This enabled us to derive an accurate counting formula for $\mathrm{BF}(N, e_4)$.

- The oddness of e_4 (i.e., which holds when $p \equiv q \equiv 3 \pmod 4$) implies that the mapping $g \mapsto -g$ is a bijective on the relevant sets. Consequently, we

[7] Another method involves multiple verifications of an exponential operation in $\left(\mathbb{Z}_N[x]/(x^2 + 1)\right)^\times / \mathbb{Z}_N^\times$.

have

$$\left|\{g \in \mathbb{Z}_N^\times \mid g^{e_4} \equiv \pm 1 \pmod N\}\right| = 2\left|\{g \in \mathbb{Z}_N^\times \mid g^{e_4} \equiv 1 \pmod N\}\right|.$$

- We decompose the problem of counting solutions to $\{g \in \mathbb{Z}_N^\times \mid g^{e_4} \equiv 1 \pmod N\}$ into counting solutions modulo each prime power factor $p_i^{r_i}$ of N using CRT (cf. Section 6.1). This involves analyzing sets $\mathrm{BF}(p_i^{r_i}, e_4) \subset (\mathbb{Z}/p_i^{r_i}\mathbb{Z})^\times$. Moreover, the number of e_4-roots of 1 in a cyclic group $(\mathbb{Z}/p_i^{r_i}\mathbb{Z})^\times$ can be easily derived (i.e., the condition $\gcd(pq, p+q-1) = 1$ is used here). More details can be found in Lemma 1.

When N is not square-free, analyzing the quotient $|\mathrm{BF}(N, e_4)|/|G(N)|$ is relatively straightforward. However, when N is square-free, a more careful analysis is required to understand how the ratio changes. In the worst-case scenarios, such as $N = p_1p_2p_3$ and $N = p_1p_2p_3p_4$, we found that the soundness error in the worst-case is $1/4$ instead of $1/2$. For example, consider the case $p = p_1$ and $q = p_2p_3$. We can assume that $p_1 \equiv p_2 \equiv 3 \pmod 4$, and $p_3 \equiv 1 \pmod 4$, and $p_i - 1 = 2^{k_i}d_i$, where d_i is odd for all $1 \le i \le 3$, and $k_1 = k_2 = 1$, $k_3 \ge 2$. Then Lemma 3 and Lemma 13 give us

$$\frac{|\mathrm{BF}(N, e_4)|}{|G(N)|} = \frac{2\prod_{i=1}^3 \gcd(e_4, d_i)}{2^{-1}\prod_{i=1}^3 (p_i - 1)} \le \frac{4d_1d_2d_3}{2^{k_1+k_2+k_3}d_1d_2d_3} \le \frac{1}{4}.$$

In conclusion, the main difference between this approach and the original proof is that the original method only demonstrated that $\mathrm{BF}(N, e_4)$ is a subgroup of $G(N)$, without providing any insight into the relative size. In contrast, our method accurately computes their exact counts.

To develop a protocol analogous to Boneh-Franklin, we consider two sets

$$\mathrm{LPBP}(D, N, e_4) := \left\{ (P, Q) \;\middle|\; \begin{array}{l} 0 \le P, Q < N, \gcd(Q, N) = 1, \\ P^2 - 4Q = D \pmod N, \\ (\alpha\beta^{-1})^{e_4} = \pm 1 \pmod{N\mathcal{O}_D} \end{array} \right\},$$

and

$$\mathcal{Z}^\epsilon(D, N) := \left\{ (P, Q) \;\middle|\; \begin{array}{l} P^2 - 4Q = D \pmod N, \\ \left[\frac{Q}{N}\right] = \epsilon, \gcd(Q, N) = 1, \\ 0 \le P, Q < N \end{array} \right\}.$$

Here α, β are the two distinct roots of the quadratic polynomial $x^2 - Px + Q$, and $\mathcal{O}_D$ represents the ring of integers of the quadratic field extension $\mathbb{Q}(\sqrt{D})$. The relation between $\mathrm{LPBP}(D, N, e_4)$ and the Lucas sequence can be found in Section 2.2. For studying these two sets, we can apply the same strategy of $\mathrm{BF}(N, e_4)$ to the proposed Lucas test, which is more complex in proving counting the two sets.

Additionally, another key point is proving that, the set $\mathrm{LPBP}(D, N, e_4)$ is always a subset $\mathcal{Z}^{+1}(D, N)$ for any odd integers N and an integer D with

$\left[\frac{-D}{N}\right] = 1$, and $\left[\frac{-D}{p}\right] = -1$. In the original Boneh-Franklin paper, this was straightforward because $p \equiv q \equiv 3 \pmod 4$, and e_4 is odd. This allowed the result $\mathrm{BF}(N, e_4) \subset G(N)$ to be easily derived from the following observation:

$$\left[\frac{g}{N}\right] = \left[\frac{g}{N}\right]^{e_4} = \left[\frac{g^{e_4}}{N}\right] = \left[\frac{\pm 1}{N}\right] = 1.$$

However, in our case, $\alpha\beta^{-1}$ does not belong to $\mathbb{Z}_N$, so this trick must be applied with caution. In our study (cf. Proposition 2), we found that when $\left(\alpha\beta^{-1}\right)^{e_4} \equiv \pm 1 \pmod{N\mathcal{O}_D}$, the representative of β^{2e_4} modulo $N\mathcal{O}_D$ can be chosen in $\mathbb{Z}_N$. We can then express $Q^{e_4} = (\alpha\beta)^{e_4}$ as $\left(\alpha\beta^{-1}\right)^{e_4} \cdot (\beta^{2e_4})$, where all three representatives in $\mathcal{O}_D/N\mathcal{O}_D$ belong to $\mathbb{Z}_N$, and apply the same method to complete the proof.

We also explain why the Lucas test offers advantages in detecting non-RSA moduli. According to the counting formula of non-perfect-square N (cf. Theorem 1, Proposition 1), the sizes of $|G|$ in the Lucas test and the Boneh-Franklin test are nearly identical when p_i are sufficiently large for all i. However, for $|H|$, the Boneh-Franklin test (resp. Lucas test) results in a count $2\prod_i \gcd(e_4, p_i - 1)$ (resp. $\prod_i \big(\gcd(e_4, p_i - 1) - 1\big) + \prod_i \gcd(e_4, p_i - 1)$). This observation shows that in most cases, it is likely to find a p_i such that $\gcd(e_4, p_i - 1) = 1$. Consequently, the size of $|H|$ in the Boneh-Franklin test is twice that of the Lucas test. As a result, the Lucas test often achieves nearly twice the probability of detecting that N is not an RSA modulus when randomly selecting elements from G, and p_i sufficiently large for all i. Practically, ensuring that N has no small prime factors p_i is straightforward via trial division, a necessary step in any efficient distributed RSA moduli generation protocol.

Next, the proposed protocol against semi-honest adversaries for the Lucas test closely resembles the Boneh-Franklin protocol, with the key distinction being that, for cases where $p \equiv 1 \pmod 4$ or $q \equiv 1 \pmod 4$, it is essential to select a D that satisfies the condition $\left[\frac{-D}{N}\right] = 1$, and $\left[\frac{-D}{p}\right] = -1$. This requires computing $\left[\frac{-D}{p}\right] = -1$. As proposed in [26], although not proven in detail, this can be done by first jointly generating s, then jointly computing and publishing $s^2 p \pmod D$ thus obtaining $[\frac{p}{D}]$. The desired value can be computed using the basic rules of the Legendre symbol (cf. π_{Leg}). In the next step, participants use their respective secrets concerning p and q to jointly compute $(\alpha\beta^{-1})^{e_4}$. For the GCD test, we verify $\gcd(N, e_4) = \gcd\big(N, p[\frac{-1}{q}] + q[\frac{-1}{p}] + [\frac{-1}{N}]\big) = 1$. The parties $\mathcal{P}_i$ then jointly generate a random number r, which is used in an MPC multiplication to compute $r\big(p[\frac{-1}{q}] + q[\frac{-1}{p}] + [\frac{-1}{N}]\big)$.

We now elucidate the rationale for imposing the supplementary condition $\left[\frac{-D}{p}\right] = \left[\frac{-D}{q}\right] = -1$ in our proposed test (cf. Theorem 2). In traditional approaches (e.g., Miller–Rabin primality test) for handling N where $p \equiv 1 \pmod 4$ or $q \equiv 1 \pmod 4$, one typically uses the factorization $p - 1 = 2^k d$ with $k \geq 2$ and odd d. It requires multiple MPC rounds to iteratively divide by 2 until d is found, which leads to non-constant execution time. To address this, we leverage Lucas sequences with carefully chosen P, Q, such that for a prime p, the $\left(p - \left[\frac{D}{p}\right]\right)$-th

term is divisible by p, where $D = P^2 - 4Q$ and $\left[\frac{\cdot}{\cdot}\right]$ denotes the Jacobi symbol. By selecting D with $\left[\frac{-D}{p}\right] = -1$, we get $p - \left[\frac{D}{p}\right] = p+1 = 2d$ (i.e., $\left[\frac{D}{p}\right] = \left[\frac{-D}{p}\right]$ when $p \equiv 1 \pmod 4$), thereby optimizing the need for additional MPC rounds, because D can be computed in parallel.

Regarding security proof for the protocol, to simulate the transcript of proposed Lucas protocol, we carefully construct a method to generate a uniform distribution over $L = \{P \in \mathbb{Z}_N \mid [\frac{P^2-D}{N}] = 1\}$. In the scenario considered by Boneh-Franklin, they use $a^2(-1)^b$ to simulate g, where $a \in \mathbb{Z}_N^\times$. They utilize $b = 0$ or 1 to control $(a^2(-1)^b)^{e_4}$. In our case, the situation is more complex as $\sqrt{D}$ maybe not belong $\mathbb{Z}_N$. Therefore, we modify the selection of $a \in \left\{ \left(\frac{v+w\sqrt{D}}{v-w\sqrt{D}}\right) | v^2 - w^2 D \in \mathbb{Z}_N^\times \text{ for all } v, w \in \mathbb{Z}_N \right\}$. We then prove that this construction can produce the desired uniform distribution of the set L (cf. Proposition 4).

In the malicious setting, we follow the methodology proposed by Chen et al.'s protocol [16, Protocol 5.2] to design our protocol. First, we define a biprimality functionality and prove that it can be realized by a maliciously secure version of the Lucas biprimality test. The protocol essentially consists of two parts: a semi-honest version of the proposed Lucas test, and a Schnorr-like verification protocol to ensure that the test was executed correctly. The soundness error of the biprimality test combines with that of the Schnorr-like protocol, resulting in a total error of

$$\gamma := \frac{5}{8} + \frac{0.625}{p_{\min} - 3},$$

meaning the test must be repeated approximately $\lceil s/\log_2(\gamma^{-1}) \rceil$ times to reduce the total soundness error to at most 2^{-s}. Here s is a statistical parameter ensuring that the probability that N is not an RSA modulus is less than 2^{-s}. A detailed analysis of the proof led us to observe a few points in Chen et al.'s protocol that could potentially be improved. Possible improvements would be in Section 4.4.

1.3 Related work

The generation of RSA moduli in a distributed manner was first pioneered by Boneh and Franklin [10]. They introduced an efficient protocol to test if an integer $N = pq$ is a valid RSA modulus (i.e., the product of two distinct primes) without revealing the factors p and q. This protocol is secure in the semi-honest adversarial model, assuming an honest majority. Their test exhibits perfect completeness, always accepting valid RSA moduli, and has a soundness error of at most 1/2 (i.e., it accepts non-RSA moduli with probability at most 1/2). Their paper detailed two principal variants of this biprimality test: one involving multiple evaluations of $\gcd(pq, (p-1)(q-1)) = 1$, and another based on repeated exponentiations within the group $\left(\mathbb{Z}_N[x]/(x^2+1)\right)^\times / \mathbb{Z}_N^\times$. Subsequent mainstream approaches have often focused on variants that incorporate conditions like $\gcd(pq, p+q-1) = 1$ to handle specific types of non-RSA moduli. Building on the theme of distributed primality testing, Algesheimer et al. [2]

proposed a distributed Miller–Rabin test achieving semi-honest security against a dishonest majority. This line of work was followed by several papers that adapt the Miller-Rabin test for designing biprimality tests [15, 20]. In the analysis of average-case soundness error, Damgård et al. [19] established an upper bound for the Miller-Rabin test. For Lucas-based tests, Einsele et al. [21] provided a corresponding upper bound for strong Lucas pseudoprimes. For articles addressing the optimization of RSA moduli candidates and proposing a more secure security model, Burkhardt et al.'s paper [15] underwent a comprehensive review.

1.4 Structure of the Paper

Section 2 introduces fundamental mathematical background and notation, including Lucas sequences. Key properties of the Chinese Remainder Theorem and the Jacobi symbol, along with an introduction to their relevant applications, are detailed in Appendices 6.1 and 6.2. A novel and refined proof for the Boneh-Franklin test is presented in Section 3.

Section 4 is dedicated to our proposed Lucas biprimality test. We present the test itself, the protocol we constructed based on it, and provide proofs of security against both semi-honest for arbitrary odd integers p, q and malicious attackers for $p \equiv q \equiv 3 \pmod 4$.

Section 5 offers a comparative data analysis of the Miller-Rabin, Boneh-Franklin, and Lucas tests. This includes an examination of the complexity of our proposed Lucas protocol in the semi-honest model. Furthermore, this section provides a theoretical analysis comparing the Boneh-Franklin and Lucas tests. Finally, it presents experimental data from biprime generation scenarios.

2 Preliminaries

Basic notations. Let $\mathbf{P}$ be the set of all primes, $\mathbb{N}$ be the nature numbers, and $\mathbb{Z}$ be the ring of integers. For a finite set S, $|S|$ means the cardinality of S. Let $\mathbb{Z}_N$ be the additive group of order N, and $\mathbb{Z}_N^\times$ be the multiplicative group in $\mathbb{Z}_N$. Moreover, $|\mathbb{Z}_N^\times| = \phi(N)$, where ϕ is the Euler's totient function. For an interval $\mathcal{I}$, we set $\mathbf{P}(\mathcal{I}) := \{p \in \mathbf{P} \mid p \in \mathcal{I}\}$. The greatest common divisor of two positive integers x and $y \in \mathbb{N}$ is denoted by $\gcd(x, y)$. Let $[\![a]\!]_m$(resp. $[\![a]\!]_\mathbb{Z}$) be the secure additive sharing of value a in the integer domain $\mathbb{Z}_m$ (resp. $\mathbb{Z}$). That is each of the participants, $\{\mathcal{P}_i\}_{i=1}^n$, has their own secret $\mathfrak{a}_i \in \mathbb{Z}_m$(resp. $\mathfrak{a}_i \in \mathbb{Z}$) such that $\sum_{i=1}^n \mathfrak{a}_i \equiv a \pmod m$ (resp. $\sum_{i=1}^n \mathfrak{a}_i = a$). Given a finite set S, the notation $a \leftarrow S$ denotes that a is sampled uniformly at random from S.

For clarity, we present symbols that have been introduced in earlier studies. Given two odd positive integers p, q, set $e_4(= e_4(p, q)) := \frac{1}{4}\big(p + \left[\frac{-1}{p}\right]\big)\big(q + \left[\frac{-1}{q}\right]\big)$.

Here $\left[\frac{\cdot}{\cdot}\right]$ is the Jacobi symbol (cf. Section 6.2). For odd integers p, q, we set

$$\begin{aligned}
\mathrm{MR}(p) &:= \{g \in \mathbb{Z}_p^\times \mid g^{(p-1)/2} \equiv \pm 1 \pmod{p}\},\\
\mathrm{BF}(N, e_4) &:= \left\{g \in \mathbb{Z}_N^\times \mid g^{e_4} \equiv \pm 1 \pmod{N}\right\},\\
G(N) &:= \left\{g \in \mathbb{Z}_N^\times \;\middle|\; \left[\frac{g}{N}\right] = 1\right\},\\
\mathcal{Z}^\epsilon(D, N) &:= \left\{(P, Q) \;\middle|\; \begin{array}{l} P^2 - 4Q = D \pmod{N},\\ \left[\frac{Q}{N}\right] = \epsilon, \gcd(Q, N) = 1,\\ 0 \le P, Q < N \end{array}\right\},
\end{aligned}$$

for $\epsilon \in \{\pm 1\}$, $\mathcal{Z}(D, N) = \cup_{\epsilon \in \{\pm 1\}} \mathcal{Z}^\epsilon(D, N)$, and

$$\mathrm{LPBP}(D, N, e_4) := \left\{(P, Q) \;\middle|\; \begin{array}{l} 0 \le P, Q < N, \gcd(Q, N) = 1,\\ P^2 - 4Q = D \pmod{N},\\ (\alpha\beta^{-1})^{e_4} = \pm 1 \pmod{N\mathcal{O}_D} \end{array}\right\}.$$

Here α, β are the two distinct roots of the quadratic polynomial $x^2 - Px + Q$, and $\mathcal{O}_D$ represents the ring of integers of the quadratic field extension $\mathbb{Q}(\sqrt{D})$. If $p \equiv q \equiv 3 \pmod 4$, the set $\mathrm{BF}(N, e_4)$ (resp. $\mathrm{LPBP}(D, N, e_4)$) is a subgroup (resp. subset) of $G(N)$ (resp. $\mathcal{Z}^{+1}(D, N)$) (cf. Proposition 2).

Given that $p \equiv q \equiv 3 \pmod 4$ and a perfect square D, we are interested in studying the following two quantities: $\beta_{\mathrm{Lucas}}(D, N, e_4) := \frac{|\mathrm{LPBP}(D,N,e_4)|}{|\mathcal{Z}^{+1}(D,N)|}$, and $\beta_{\mathrm{BF}}(N, e_4) := \frac{|\mathrm{BF}(N,e_4)|}{|G(N)|}$, where two quantities, viewed as soundness error, are used to evaluate the proportion of randomly selected elements in the set of denominators that pass the test when $N = pq$ is not an RSA modulus. These values always belong to the range $[0, 1]$, and the smaller the value, the easier it is to determine that p and q are not an RSA modulus. In fact, Proposition 1 and Proposition 2 implies that $\beta_{\mathrm{Lucas}}(D, N, e_4)$ is independent of the chosen of perfect squares D, if $p \equiv q \equiv 3 \pmod 4$. For simplicity, when we write $\beta_{\mathrm{Lucas}}(N, e_4) = \beta_{\mathrm{Lucas}}(1, N, e_4)$.

2.1 Two Mathematical Results

Lemma 1. [5, Lemma 2.1] *Let G be a cyclic group and d an integer. There are exactly* $\gcd(d, |G|)$ *dth-root of* 1 *in G.*

Lemma 2 (Hensel's Lemma). [34, Theorem 2.23 or 2.24] *Let $f(x)$ be a polynomial with integer coefficients. If p is a prime number and a is an integer such that $f(a) \equiv 0 \pmod{p^j}$, and $f'(a) \not\equiv 0 \pmod p$ then, there exists an integer $t \pmod p$ such that $f(a + tp^j) \equiv 0 \pmod{p^{j+1}}$.*

2.2 Lucas Pseudo-primes

We recall Lucas sequence and some results [5]. Let P and Q be integers and $D := P^2 - 4Q$. The Lucas sequence (U_k, V_k) that is associated with the parameters

P, Q are defined as, for $k \geq 0$,

$$\begin{cases} U_{k+2} = PU_{k+1} - QU_k; \\ V_{k+2} = PV_{k+1} - QV_k, \end{cases} \quad \text{with} \quad \begin{cases} U_0 = 0, U_1 = 1; \\ V_0 = 2, V_1 = P. \end{cases}$$

It is well known that $U_{p-\left[\frac{D}{p}\right]} \equiv 0 \pmod{p}$ for any prime $p \nmid 2QD$. For the Lucas sequence [5, Section 3], (U_k, V_k) associated with P, Q and $P^2 - 4Q \neq 0$, we have the general formula: for all $k \in \mathbb{N}$,

$$U_k = \frac{\alpha^k - \beta^k}{\alpha - \beta},\ V_k = \alpha^k + \beta^k,$$

where α, β are two distinct roots of the polynomial $x^2 - Px + Q$. Let $\mathcal{O}_D$ be the ring of integers of a quadratic field $\mathbb{Q}(\sqrt{D})$, and $\tau := \alpha\beta^{-1}$. If $N \nmid 2QD$, then we have, for $k \in \mathbb{N}$,

$$N \mid U_k \text{ if and only if } \tau^k \equiv 1 \pmod{N\mathcal{O}_D}. \tag{1}$$

Given an element $u + v\sqrt{D} \in \mathbb{Q}(\sqrt{D})$, the norm map is given by $\mathbf{N}(u + v\sqrt{D}) = u^2 - v^2 D \in \mathbb{Q}$. When $x \in \mathcal{O}_D$, the norm $\mathbf{N}(x) \in \mathbb{Z}$. Consider the multiplicative group of norm 1 elements denoted by $\widehat{\left(\mathcal{O}_D/N\right)}$ in a free $\mathbb{Z}/N\mathbb{Z}$-algebra of rank 2. This group is the image of the set

$$\{x \in \mathcal{O}_D \mid \mathbf{N}(x) \equiv 1 \pmod{N}\}$$

by the canonical map $\mathcal{O}_D \to \mathcal{O}_D/N$.

2.3 The Security Model

We analyze the security of our protocol against static, rushing semi-honest and malicious adversaries under the standard definition of stand-alone, secure multiparty computation with abort (cf. Goldreich [35, Section 7.5.1]; Katz [30, Definition 1]), as detailed in Section 6.4.

3 Refine Boneh-Franklin Biprimality Testing

We establish two key results regarding the Boneh-Franklin test: first, that its tightest worst-case soundness error is 1/4; and second, the necessary and sufficient conditions on p and q for this worst-case to occur. The formula for the size of $\mathrm{BF}(N, e_4)$ is provided below.

Lemma 3. *Let $p \equiv q \equiv 3 \pmod{4}$ with $\gcd\left(pq, e_4\right) = 1$. Assume that $N = pq = \prod_{i=1}^{s} p_i^{r_i}$, where p_i is prime for all i, then we have*

$$|\mathrm{BF}(N, e_4)| = 2 \cdot \prod_{i=1}^{s} \gcd(e_4, d_i).$$

Here $p_i - 1 = 2^{k_i} d_i$ with $2 \nmid d_i$ for all $1 \leq i \leq s$.

Proof. Since e_4 is odd, we have

$$|\{g \in \mathbb{Z}_N^\times \mid g^{e_4} \equiv 1 \pmod N\}| = |\{g \in \mathbb{Z}_N^\times \mid g^{e_4} \equiv -1 \pmod N\}|$$

by the bijective map $g \mapsto -g$, which implies that

$$|\mathrm{BF}(N, e_4)| = 2 \cdot |\{g \in \mathbb{Z}_N^\times \mid g^{e_4} \equiv 1 \pmod N\}|.$$

According to Lemma 8, we reduce the problem to count the cardinality of e_4-th roots of 1 in $(\mathbb{Z}/p_i^{r_i}\mathbb{Z})^\times$ which are cyclic groups for all i [29, Theorem 3, Chapter 4], since N is odd. Combining this fact, $\gcd(pq, e_4) = 1$, and Lemma 1, one has the number of e_4-th roots of 1 in the group $(\mathbb{Z}/p_i^{r_i}\mathbb{Z})^\times$ is

$$\gcd(e_4, p_i^{r_i-1}(p_i - 1)) = \gcd(e_4, 2^{k_i} d_i) = \gcd(e_4, d_i).$$

The above discussion implies the desired result. □

The proofs of Theorem 1 and Theorem 2 address non-RSA moduli N by considering three primary categories: 1) N is a perfect square; 2) N is square-free (and not an RSA modulus, implying N is a product of $s \geq 3$ distinct primes); and 3) N is neither a perfect square nor square-free.

Our proof for square-free non-RSA moduli (cf. category 2) analyzes cases based on the number of distinct prime factors, s, particularly when $s \geq 3$. For category 3), the analysis incorporates both the number of distinct prime factors s and the exponents r_i of these primes (where at least one $r_i \geq 2$).

We also recall that for an integer $N = pq$ where p, q are integers satisfying $p \equiv q \equiv 3 \pmod 4$, $\mathrm{BF}(N, e_4)$ forms a subgroup of $G(N)$.

Theorem 1 (Boneh-Franklin biprimality test). *Let $p \equiv q \equiv 3 \pmod 4$, and $\gcd(pq, e_4) = 1$, where $e_4 = (p-1)(q-1)/4$. Assume that $N := pq$. If p, q are both distinct primes, then we have $\mathrm{BF}(N, e_4) = G(N)$. For the other cases, we have $|\mathrm{BF}(N, e_4)| \leq |G(N)|/4$, except for the case $p = q = 3$.*

Proof. Recall that $p_i - 1 = 2^{k_i} d_i$ with odd d_i for all i as the same notations in the Lemma 3. At first, consider the case p, q are distinct primes, which implies $e_4 = d_1 d_2$ and $k_1 = k_2 = 1$. From Lemma 13, the proof of this case is completed by the following equality:

$$|\mathrm{BF}(N, e_4)| = 2\gcd(e_4, d_1) \cdot \gcd(e_4, d_2) = 2d_1 d_2 = \phi(N)/2.$$

Consider case 1): N is a perfect square, which implies $r_1 \geq 2$. Lemma 3 and Lemma 13 imply that

$$\begin{aligned}\beta_{\mathrm{BF}}(N, e_4) =& \frac{|\mathrm{BF}(N, e_4)|}{|G(N)|} = \frac{2\prod_{i=1}^s \gcd(e_4, d_i)}{\prod_{i=1}^s p_i^{r_i-1}(p_i - 1)} \leq \frac{2\prod_{i=1}^s d_i}{\prod_{i=1}^s p_i^{r_i-1}(p_i - 1)} \\ =& \frac{2\prod_{i=1}^s 2^{-k_i}}{\prod_{i=1}^s p_i^{r_i-1}} < 2^{1-1} \cdot 5^{-1} = \frac{1}{5},\end{aligned}$$

except for the case $p = q = 3$.

Consider the case 3). The condition non-perfect-square means that $s \geq 2$. If not, $s = 1$, then $N = p_1^{r_1}$. Since $p \equiv q \equiv 3 \pmod 4$, which implies that $N \equiv 1 \pmod 4$ and $p_1 \equiv 3 \pmod 4$, and r_1 is even, which gives a contradiction. Meanwhile, non square-free N implies that there exists i such that $r_i \geq 2$. Now, one has

$$\beta_{\mathrm{BF}}(N, e_4) = \frac{4\prod_{i=1}^{s} \gcd(e_4, d_i)}{\prod_{i=1}^{s} p_i^{r_i-1}(p_i - 1)} \leq 2^{-k_1-\ldots-k_s+2} \left(\prod_{i=1}^{s} p_i^{r_i-1}\right)^{-1}.$$

If there exists $p_i \geq 5$ with $r_i \geq 2$ then

$$\beta_{\mathrm{BF}}(N, e_4) \leq 2^{2-1-1} \cdot 5^{-1} = 1/5.$$

Additionally, if $s \geq 3$, then

$$\beta_{\mathrm{BF}}(N, e_4) \leq 2^{2-1-1-1} \cdot 3^{-1} = 1/6.$$

Therefore, we only consider the case $N = 3^{r_1}p_2$ with $r_1 = 2$ due to

$$\beta_{\mathrm{BF}}(N, e_4) \leq 2^{2-1-1} \cdot 3^{-2} = 1/9 \text{ as } r_1 \geq 3.$$

As for the case $s = 2$, then $p_2 \equiv 1 \pmod 4$ since $N \equiv 1 \pmod 4$. This case also implies that

$$\beta_{\mathrm{BF}}(N, e_4) \leq 2^{2-1-2} \cdot 3^{-1} = 1/6.$$

In conclusion, when N is non-square-free with $s \geq 2$, and $\beta_{\mathrm{BF}}(N, e_4) \leq 1/6$.

When N is square-free, the case 2), we consider the case $s = 3$. Because $p \equiv q \equiv 3 \pmod 4$, two elements of the set $\{p_1, p_2, p_3\}$ are 3 module 4 and one of it is 1 module 4, which gives the bound

$$\beta_{\mathrm{BF}}(N, e_4) \leq 2^{-k_1-\ldots-k_s+2} = 2^{2-1-1-2} = 1/4.$$

For all $s \geq 4$, we have $\beta_{\mathrm{BF}}(N, e_4) \leq 2^{-k_1-\ldots-k_s+2} \leq 2^{-2}$, since $k_i \geq 1$ for all i. □

Based on the proof of Theorem 1, we can establish the following sufficient and necessary conditions for the worst-case scenario to occur.

Corollary 1. *Assume that the assumption of Theorem 1 holds. The equality* $|\mathrm{BF}(N, e_4)| = |G(N)|/4$ *is true if and only if one of the two situations occurred without considering the symmetry of p and q. 1).* $s = 3$, $p = p_1p_2$, $q = p_3$, $\gcd(e_4, p_1 - 1) = (p_1 - 1)/4$, *and* $\gcd(e_4, p_i - 1) = (p_i - 1)/2$ *for* $i \in \{2, 3\}$, *where* $p_1 \equiv 5 \pmod 8$, *and* $p_2 \equiv p_3 \equiv 3 \pmod 4$*; 2).* $s = 4$, $\gcd(e_4, p_i - 1) = (p_i - 1)/2$, $p = p_1p_2p_3$, *and* $q = p_4$, *where* $p_i \equiv 3 \pmod 4$ *for all* $1 \leq i \leq s$.

The bound in the result of Theorem 1 is tight. Taking $p_1 = 3, p_2 = 5$, and $p_3 \equiv 23 \pmod{420}$, Dirichlet Theorem[8] says that there are infinitely many $N = (p)q = (p_1p_2)p_3$ such that $|\mathrm{BF}(N, e_4)| = |G(N)|/4$, given that $\gcd(N, e_4) = \gcd(15q, 7(q - 1)) = 1$ and $\gcd(420, 23) = 1$.

[8] If $\gcd(a, n) = 1$, then there exists infinite prime x with $x \equiv a \pmod n$ [33, Corollary 13.8].

4 The Lucas Biprimality Test

This section introduces a novel test for identifying RSA moduli, applicable to odd integers p and q where $\gcd\left(pq, (p+[\frac{-1}{p}])(q+[\frac{-1}{q}])\right) = 1$. We subsequently provide a distributed protocol for this test that is secure against semi-honest adversaries. Additionally, a protocol resilient to malicious adversaries is presented, specifically for the $p \equiv q \equiv 3 \pmod 4$ scenario. All necessary properties of the Jacobi symbol are detailed in Appendix 6.2.

4.1 A Lucas Biprimality Testing

The proof for the Lucas biprimality test follows a similar methodology to that of the Boneh-Franklin test. We first derive formulas for the cardinalities of $\mathrm{LPBP}(D, N, e_4)$ and $\mathcal{Z}^{+1}(D, N)$. Subsequently, we analyze the worst-case upper bound of their quotient. The analysis begins with an examination of the special case where $N = p^r$.

The key to calculating the cardinality of the set $\mathcal{Z}^{\epsilon}(D, p^r)$ is to first compute the result for the base case $r = 1$. Once this is obtained, the more general case for $r > 1$ can be addressed using the standard technique of Hensel's Lemma (cf. Lemma 2). For the $r = 1$ case, we introduce two auxiliary subsets, S^{+1} and S^{-1}, defined for $\epsilon \in \{\pm 1\}$ as:

$$S^{\epsilon} = \left\{ 1 \le i \le \frac{p-1}{2} \;\middle|\; \left[\frac{i^2 + D/4}{p}\right] = \epsilon, \text{ and } i^2 \not\equiv \frac{-D}{4} \pmod p \right\}.$$

The relationship between $\mathcal{Z}^{+1}(D, p)$ and S^{+1} is established in Equation (4). We then compute the values of two expressions, $|S^{+1}| + |S^{-1}|$ and $|S^{+1}| - |S^{-1}|$. Solving this system yields the individual cardinalities $|S^{+1}|$ and $|S^{-1}|$, from which we deduce the cardinality of $\mathcal{Z}^{\epsilon}(D, p)$.

Lemma 4. *Let p be an odd prime, and D be an element of $\mathbb{Z}_p^{\times}$, then for $\epsilon \in \{\pm 1\}$,*

$$|\mathcal{Z}^{\epsilon}(D, p^r)| = \begin{cases} \left(\frac{1+\epsilon}{2}\right) p^{r-1} \left(p - \left[\frac{D}{p}\right] - 1\right), & \text{if } 2 \mid r; \\ p^{r-1} \left(\frac{(p - \left[\frac{D}{p}\right] - 1) - \epsilon}{2}\right), & \text{if } 2 \nmid r. \end{cases}$$

Proof. In the case where $2 \mid r$, the condition $\left[\frac{Q}{p^r}\right] = 1$ always holds, which implies that $|\mathcal{Z}^{+1}(D, p^r)| = |\mathcal{Z}(D, p^r)|$. Therefore, we can focus on studying the cardinality of the set $\mathcal{Z}(D, p^r)$. In the special case $r = 1$ for the set $\mathcal{Z}(D, p^r)$, it is sufficient to consider the cardinality of the set $\{P \in \mathbb{Z}_p | P^2 = D + 4Q, 0 < Q < p\}$. Note that the equation $x^2 = D$ has two (resp. zero) solutions in $\mathbb{Z}_p$ if $\left[\frac{D}{p}\right] = 1$(resp. $\left[\frac{D}{p}\right] = -1$), there are $\frac{p-1}{2} - \frac{1+\left[\frac{D}{p}\right]}{2}$ values of Q such that $x^2 = D + 4Q$ has two distinct solutions. Additionally, there is one value of Q (specifically $Q = \frac{-D}{4}$) for which $x^2 = D + 4Q$ has a single solution. Thus, the

total number of solutions is given by $(\frac{p-1}{2} - \frac{1+\left[\frac{D}{p}\right]}{2}) \cdot 2 + 1 = p - \left[\frac{D}{p}\right] - 1$. For $r > 1$, the desired result can be obtained using Hensel's lemma. The detail proof can be found in Proposition 3.

As for the case $2 \nmid r$, we first consider the case $r = 1$ and $\epsilon = 1$. Then we can assume that $Q = Q'^2$. It implies that $\mathcal{Z}^{+1}(D, p)$, which is equal to

$$\left\{ (P, Q') \;\middle|\; \begin{array}{l} (P/2)^2 = (Q')^2 + D/4 \pmod p, \gcd(Q', p) = 1, \\ 0 \le P < p, 1 \le Q' \le (p-1)/2. \end{array} \right\}.$$

Now, for counting the above set, we study the following sum

$$\sum_{i=1}^{(p-1)/2} \left[\frac{i^2 + D/4}{p}\right] = \frac{-1 - \left[\frac{D}{p}\right]}{2} \text{ (by Lemma 16)},$$

which gives us the relation

$$|S^{-1}| = |S^{+1}| + \left(1 + \left[\frac{D}{p}\right]\right)/2, \tag{2}$$

where $S^{\epsilon} = \left\{ 1 \le i \le \frac{p-1}{2} \;\middle|\; \left[\frac{i^2+D/4}{p}\right] = \epsilon, i^2 \not\equiv \frac{-D}{4} \pmod p \right\}$ (i.e., if there exists i such that $i^2 \equiv -D/4 \pmod p$, then $\left[\frac{i^2+D/4}{p}\right] = 0$).

Note that $|S^{+1}|+|S^{-1}|$ depends on whether exist i such that $i^2 \equiv -\frac{D}{4} \pmod p$. Specifically,

$$|S^{+1}| + |S^{-1}| = (p-1)/2 - \left(1 + \left[\frac{-D}{p}\right]\right)/2. \tag{3}$$

Moreover, for each $i \in S^{+1}$, we can find two distinct solutions for $(x/2)^2 \equiv i^2 + D/4 \pmod p$. If $\left[\frac{-(D/4)}{p}\right] = \left[\frac{-D}{p}\right] = 1$, then an additional solution can be found (i.e., $(0, \frac{-D}{4}) \in \mathcal{Z}^{+1}(D, p)$). Therefore,

$$|\mathcal{Z}^{+1}(D, p)| = 2 \cdot |S^{+1}| + \frac{1 + \left[\frac{-D}{p}\right]}{2}. \tag{4}$$

Combining (2), (3), and (4) gives that

$$\left|\mathcal{Z}^{+1}(D, p)\right| = \frac{p - \left[\frac{D}{p}\right] - 2}{2}.$$

Furthermore, combining Proposition 3, one has

$$\left|\mathcal{Z}^{-1}(D, p)\right| = |\mathcal{Z}(D, p)| - \left|\mathcal{Z}^{+1}(D, p)\right| = p - \left[\frac{D}{p}\right] - 1 - \left(\frac{p - \left[\frac{D}{p}\right] - 2}{2}\right) = \frac{p - \left[\frac{D}{p}\right]}{2}.$$

The proof is complete by Hensel's Lemma for the general case $r \ge 2$ (cf. Lemma 15). □

The counting formula for a general N is provided below.

Proposition 1. *Let D be an integer and $N := \prod_{i=1}^{s} p_i^{r_i}$ be a positive integer with $\gcd(N, 2D) = 1$. Write $S = S_0 \cup S_1$, where $S_j := \{i \mid r_i \equiv j \pmod 2, 1 \le i \le s\}$. Then, one has, if N is not a perfect square in $\mathbb{Z}$,*

$$|\mathcal{Z}^{+1}(D,N)| = \left(\frac{\prod_{i\in S} p_i^{r_i-1}}{2}\right)\left(\prod_{i\in S_0}\left(p_i - \left[\frac{D}{p_i}\right] - 1\right)\right) \cdot \left(\prod_{i\in S_1}\left(p_i - \left[\frac{D}{p_i}\right] - 1\right) + (-1)^{|S_1|}\right).$$

Otherwise, if N is a perfect square,

$$|\mathcal{Z}^{+1}(D,N)| = \prod_{i\in S} p_i^{r_i-1}\left(p_i - \left[\frac{D}{p_i}\right] - 1\right).$$

Proof. If N is a perfect square, we obtain the desired result from Lemma 4 and Lemma 9. If N is not a square, from Lemma 9 then we have

$$|\mathcal{Z}^{+1}(D,N)| = \left(\prod_{i\in S_0} |\mathcal{Z}^{+1}(D,p_i^{r_i})|\right)\left(\sum_{\epsilon_1\cdot\ldots\cdot\epsilon_{|S_1|}=1}\prod_{i\in S_1}|\mathcal{Z}^{\epsilon_i}(D,p_i^{r_i})|\right).$$

Using Lemma 4, we only need to prove

$$\sum_{\epsilon_1\cdot\ldots\cdot\epsilon_{|S_1|}=1}\prod_{i\in S_1}|\mathcal{Z}^{\epsilon_i}(D,p_i^{r_i})| = \left(\prod_{i\in S_1} p_i^{r_i-1}\right)\left(\prod_{i\in S_1}\left(p_i - \left[\frac{D}{p_i}\right] - 1\right) + (-1)^{|S_1|}\right)\Big/2.$$

This proof can be concluded through mathematical induction on the cardinality of $|S_1|$. When $|S_1| = 1$, it follows that ϵ must equal 1, leading to the desired result. Assuming that $|S_1| = k$, the equality is satisfied. Let $A_i = p_i - \left[\frac{D}{p_i}\right] - 1$. Then, when $|S_1| = k+1$, applying $|\mathcal{Z}^{-1}| = |\mathcal{Z}| - |\mathcal{Z}^{+1}|$, and Proposition 3, we have

$$\begin{aligned}
&\sum_{\epsilon_1\cdot\ldots\cdot\epsilon_{k+1}=1}\prod_{i\in S_1}|\mathcal{Z}^{\epsilon_i}(D,p_i^{r_i})| \\
=&|\mathcal{Z}^{-1}(D,p_{k+1}^{r_{k+1}})|\cdot\sum_{\epsilon_1\cdot\ldots\cdot\epsilon_k=-1}\prod_{i=1}^{k}|\mathcal{Z}^{\epsilon_i}(D,p_i^{r_i})| + |\mathcal{Z}^{+1}(D,p_{k+1}^{r_{k+1}})|\cdot\sum_{\epsilon_1\cdot\ldots\cdot\epsilon_k=1}\prod_{i=1}^{k}|\mathcal{Z}^{\epsilon_i}(D,p_i^{r_i})| \\
=&\frac{\left(\prod_{i=1}^{k+1} p_i^{r_i-1}\right)\left(A_{k+1}+1\right)\left(2\prod_{i=1}^{k}A_i - \left(\prod_{i=1}^{k}A_i + (-1)^k\right)\right)}{4} \\
&+\frac{\left[\prod_{i=1}^{k+1} p_i^{r_i-1}\right]\left[A_{k+1}-1\right]\left[\prod_{i=1}^{k}A_i + (-1)^k\right]}{4} \\
=&\left(\prod_{i\in S_1} p_i^{r_i-1}\right)\left(\prod_{i\in S_1}A_i + (-1)^{|S_1|}\right)\Big/2 = \left(\prod_{i\in S_1} p_i^{r_i-1}\right)\left(\prod_{i\in S_1}\left(p_i - \left[\frac{D}{p_i}\right] - 1\right) + (-1)^{|S_1|}\right)\Big/2.
\end{aligned}$$

□

Next, we study the cardinality of the set LPBP and then prove it is a subset of $\mathcal{Z}^{+1}$.

Proposition 2. *Let p, q be positive odd integers, $N = pq = \prod_{i=1}^{s} p_i^{r_i}$, and D be an integer in $\mathbb{Z}$ with $\gcd(2D, N) = 1$, and $\left[\frac{-D}{p}\right] = \left[\frac{-D}{q}\right] = -1$. Then we have the set $\mathrm{LPBP}(D, N, e_4)$ is a subset of $\mathcal{Z}^{+1}(D, N)$. Furthermore assuming $\gcd(N, e_4) = 1$, its cardinality is given by*

$$|\mathrm{LPBP}(D, N, e_4)| = \prod_{i=1}^{s} \big(\gcd(e_4, d_i) - 1\big) + \prod_{i=1}^{s} \gcd(e_4, d_i).$$

Here $p_i - \left[\frac{D}{p_i}\right] = 2^{k_i} d_i$ with $2 \nmid d_i$ for all $1 \le i \le s$.

Proof. For sake of proving $\mathrm{LPBP}(D, N, e_4) \subseteq \mathcal{Z}^{+1}(D, N)$, we need to prove that taking any pair $(P, Q) \in \mathrm{LPBP}(D, N, e_4)$ then one has $(\alpha\beta^{-1})^{e_4} \equiv \pm 1 \pmod{N\mathcal{O}_D}$, where α, β are two distinct roots of the polynomial $x^2 - Px + Q$, which implies that $\left[\frac{Q}{N}\right] = 1$. Note that the representative of $(\alpha\beta^{-1})^{e_4}$ modulo $N\mathcal{O}_D$ is ± 1, which can be viewed as an element in $\mathbb{Z}_N^\times$, and $Q = \alpha\beta \in \mathbb{Z}_N^\times$ imply that the canonical representative of β^{2e_4} modulo $N\mathcal{O}_D$ belongs to $\mathbb{Z}_N^\times$. Because e_4 is odd, we have

$$\left[\frac{Q}{N}\right] = \left[\frac{Q}{N}\right]^{e_4} = \left[\frac{(\alpha\beta)^{e_4}}{N}\right] = \left[\frac{\beta^{2e_4}}{N}\right]\left[\frac{(\alpha\beta^{-1})^{e_4}}{N}\right].$$

Next, the goal is to prove that when condition $(\alpha\beta^{-1})^{e_4} \equiv \pm 1 \pmod{N}$ holds, one has $\beta^{e_4} \equiv Y\sqrt{D} \pmod{N\mathcal{O}_D}$ for some $Y \in \mathbb{Z}_N$. Recall that $(\alpha\beta^{-1})^{e_4} \equiv \pm 1 \pmod{N}$, then $\alpha^{e_4} = \pm\beta^{e_4}$, which implies that $(P + \sqrt{D})^{e_4} = \pm(P - \sqrt{D})^{e_4} \pmod{N\mathcal{O}_D}$. Now, consider the case $(P+\sqrt{D})^{e_4} = -(P-\sqrt{D})^{e_4} \pmod{N\mathcal{O}_D}$. Write $(P + \sqrt{D})^{e_4} = A + B\sqrt{D} \pmod{N\mathcal{O}_D}$, where $A = \sum_{\substack{i=0:\\ 2\nmid i}}^{e_4} \binom{e_4}{i} P^i D^{(e_4-i)/2}$ and $B = \sum_{\substack{i=0:\\ 2|i}}^{e_4} \binom{e_4}{i} P^i D^{(e_4-1-i)/2}$. Then $-(P-\sqrt{D})^{e_4} = -A+B\sqrt{D} \pmod{N\mathcal{O}_D}$. Therefore, the equality $A + B\sqrt{D} \equiv -A + B\sqrt{D} \pmod{N\mathcal{O}_D}$ gives us $A \equiv 0 \pmod{N\mathcal{O}_D}$, since N is odd. Now, we have

$$\beta^{e_4} = \left(\frac{P + \sqrt{D}}{2}\right)^{e_4} \equiv \frac{B(\sqrt{D})}{2^{e_4}} \pmod{N\mathcal{O}_D},$$

which implies that $\beta^{2e_4} \equiv 2^{-2e_4} B^2 D \pmod{N}$. In conclusion, when $(\alpha\beta^{-1})^{e_4} \equiv -1 \pmod{N\mathcal{O}_D}$, we have

$$\left[\frac{Q}{N}\right] = \left[\frac{\beta^{2e_4}}{N}\right]\left[\frac{(\alpha\beta^{-1})^{e_4}}{N}\right] = \left[\frac{2^{-2e_4} B^2 D}{N}\right]\left[\frac{-1}{N}\right] = \left[\frac{D}{N}\right]\left[\frac{-1}{N}\right] = 1.$$

Similarly, when $\alpha^{e_4} = \beta^{e_4}$, we have $\beta^{e_4} \equiv 2^{-e_4} A \pmod N$, which gives us

$$\left[\frac{Q}{N}\right] = \left[\frac{\beta^{2e_4}}{N}\right]\left[\frac{(\alpha\beta^{-1})^{e_4}}{N}\right] = \left[\frac{2^{-2e_4}A^2}{N}\right]\left[\frac{1}{N}\right] = 1.$$

The proof of the cardinality of $\mathrm{LPBP}(D, N, e_4)$ can be found in [5, Section 1.4]. □

Finally, the soundness error, denoted β_{Lucas}, is estimated as follows.

Theorem 2. *Let p, q be odd integers, $\gcd\left(pq, e_4\right) = 1$. Set $N = pq$. Assume that D is an integer in $\mathbb{Z}$ with $\gcd(2D, N) = 1$, and $\left[\frac{-D}{p}\right] = \left[\frac{-D}{q}\right] = -1$. If p, q are both distinct primes, then we have $\mathrm{LPBP}(D, N, e_4) = \mathcal{Z}^{+1}(D, N)$. For the remainder cases, set $p_{\min}$ be the minimal prime factor of N. Assume $p_{\min} \geq 11$, then we have*

$$\beta_{\mathrm{Lucas}}(D, N, e_4) = \frac{|\mathrm{LPBP}(D, N, e_4)|}{|\mathcal{Z}^{+1}(D, N)|} < \frac{1}{4} + \frac{1.25}{p_{\min} - 3}.$$

Proof. This proof will distinguish several cases, and the reader may refer to Theorem 1. Consider the case p, q are distinct primes. Set $p_1 = p$ and $p_2 = q$. Recall that $p_i - \left[\frac{D}{p_i}\right] = 2d_i$ for all i. Thus, one has $e_4 = \left(p - \left[\frac{D}{p}\right]\right)\left(q - \left[\frac{D}{q}\right]\right)/4 = d_1 d_2$. Now, we only need to prove that $|\mathcal{Z}^{+1}(D, N)| = |\mathrm{LPBP}(D, N, e_4)|$, because Proposition 2 says that $\mathrm{LPBP}(D, N, e_4)$ is a subset of $\mathcal{Z}^{+1}(D, N)$. The proof can be completed by the following equality:

$$\begin{aligned}|\mathrm{LPBP}(D, N, e_4)| =& \left(\gcd(e_4, d_1) - 1\right) \cdot \left(\gcd(e_4, d_2) - 1\right) + \gcd(e_4, d_1) \cdot \gcd(e_4, d_2) \\ =& (d_1 - 1)(d_2 - 1) + d_1 d_2 = \frac{(2d_1 - 1)(2d_2 - 1) + 1}{2} = |\mathcal{Z}^{+1}(D, N)|.\end{aligned}$$

Consider the case perfect square N. Proposition 1, and Proposition 2 imply that, for all $p_i \geq 7$,

$$\begin{aligned}&\beta_{\mathrm{Lucas}}(D, N, e_4) \leq \left(\frac{2}{\prod_{i=1}^{s} p_i^{r_i - 1}}\right)\left(\frac{\prod_{i=1}^{s} 2^{-k_i}(p_i - 1)}{\prod_{i=1}^{s}(p_i - 2)}\right) \\ &\leq \left(\frac{2}{\prod_{i=1}^{s} p_i^{r_i - 1}}\right)\left(\prod_{i=1}^{s}\left(\frac{1}{2} + \frac{1}{2(p_i - 2)}\right)\right) \leq \left(\frac{2}{7}\right)\left(\frac{1}{2} + \frac{1}{10}\right) = \frac{6}{35}.\end{aligned}$$

Note that

$$\begin{aligned}&\prod_{i \in S_0}\left(p_i - \left[\frac{D}{p_i}\right] - 1\right)\left(\prod_{i \in S_1}\left(p_i - \left[\frac{D}{p_i}\right] - 1\right) + (-1)^{|S_1|}\right) \\ &\geq \prod_{i \in S_0}(p_i - 2)\left(\prod_{i \in S_1}(p_i - 2) - 1\right).\end{aligned}$$

Regarding the case of non-square-free (i.e., there exists an i such that $r_i \geq 2$) and non-perfect-square N (i.e., $|S_1| \geq 1$), Proposition 1, and Proposition 2 say that, for all $p_i \geq 11$,

$$\beta_{\text{Lucas}}(D, N, e_4) \leq \left(\frac{4}{\prod_{i=1}^{s} p_i^{r_i-1}}\right)\left(\frac{\prod_{i=1}^{s} 2^{-k_i}(p_i-1)}{\prod_{i=1}^{s}(p_i-2) - \prod_{i\in S_0}(p_i-2)}\right)$$
$$\leq \left(\frac{4}{\prod_{i=1}^{s} p_i^{r_i-1}}\right)\left(\frac{\prod_{i=1}^{s}\left(\frac{1}{2}+\frac{1}{2(p_i-2)}\right)}{1-\left(\prod_{i\in S_1}(p_i-2)\right)^{-1}}\right) \leq \left(\frac{4}{11}\right)\left(\frac{\frac{1}{2}+\frac{1}{18}}{1-9^{-1}}\right) = \frac{5}{22}.$$

When N is square-free. Consider the case $s = 3$. Then there exists one of $\{p_1, p_2, p_3\}$ is $4 \mid p_i - \left[\frac{D}{p_i}\right]$. If not, for all $1 \leq i \leq 3$, $p_i - \left[\frac{D}{p_i}\right] = 2d_i$ with odd d_i hold, which is equivalent to $p_i \equiv -\left[\frac{D}{p_i}\right] \pmod 4$. Since $s = 3$, we can assume without loss of generality that $p = p_1$ and $q = p_2p_3$. For such q and the assumption $\left[\frac{-D}{q}\right] = -1$, we have

$$q \equiv \left[\frac{D}{p_2p_3}\right] = \left[\frac{D}{q}\right] = -\left[\frac{-1}{q}\right] \pmod 4 = \begin{cases} 1, \text{ if } q \equiv 3 \pmod 4; \\ 3, \text{ if } q \equiv 1 \pmod 4. \end{cases}$$

It gives a contradiction. Therefore, applying Lemma 17, we obtain that

$$\beta_{\text{Lucas}}(D, N, e_4) < \frac{1}{4}\left(\frac{\prod_{i=1}^{3}(p_i-1)}{\prod_{i=1}^{3}(p_i-2)-1}\right) < \frac{1}{4}\left(\frac{(p_{\min}-1)^3}{(p_{\min}-2)^3-1}\right)$$
$$\leq \frac{1}{4}\left(\frac{(p_{\min}-1)^4}{(p_{\min}-2)^4-1}\right), \text{ for all } p_{\min} \geq 3.$$

Similarly, as $s = 4$, we have

$$\beta_{\text{Lucas}}(D, N, e_4) < \frac{1}{4}\left(\frac{(p_{\min}-1)^4}{(p_{\min}-2)^4-1}\right).$$

When $s \geq 5$, applying the following fact

$$\frac{\prod_{i=1}^{s}(p_i-1)}{\prod_{i=1}^{s}(p_i-2)-1} < \left(\frac{\prod_{i=1}^{4}(p_i-1)}{\prod_{i=1}^{4}(p_i-2)-1}\right)\left(\frac{\prod_{i=5}^{s}(p_i-1)}{\prod_{i=5}^{s}(p_i-2)-1}\right),$$

and Lemma 18 with $j = 5$, we arrive that, for $s \geq 5$,

$$\beta_{\text{Lucas}}(D, N, e_4) \leq 2^{2-k_1-\ldots-k_s}\frac{\prod_{i=1}^{s}(p_i-1)}{\prod_{i=1}^{s}(p_i-2)-1} < \frac{1}{4}\left(\frac{(p_{\min}-1)^4}{(p_{\min}-2)^4-1}\right).$$

Lastly, we have

$$\frac{1}{4}\left(\frac{(p_{\min}-1)^4}{(p_{\min}-2)^4-1}\right) = \frac{1}{4} + \frac{1}{4}\left(\frac{(p_{\min}-1)^4-(p_{\min}-2)^4+1}{(p_{\min}-2)^4-1}\right)$$
$$= \frac{1}{4} + \frac{1}{4}\left(\frac{4}{(p_{\min}-2)-1} + \frac{2}{(p_{\min}-2)^2+1}\right)$$
$$< \frac{1}{4} + \frac{1}{4}\left(\frac{4}{(p_{\min}-2)-1} + \frac{2}{(p_{\min}-2)^2-1}\right) < \frac{1}{4} + \frac{1.25}{p_{\min}-3}.$$

□

The condition $\left[\frac{-D}{p}\right] = -1$ is unsatisfiable when p is a perfect square. Nevertheless, the likelihood of randomly selecting a perfect square for p is minimal. Moreover, in such instances where p is a perfect square, $N = pq$ would not constitute a valid RSA modulus.

4.2 The Lucas Biprimality Test in the Semi-Honest Setting

We propose a protocol based on Theorem 2 and provide its security proof under the semi-honest adversarial model. First, we define the target functionality and then present its realization, denoted as $\pi^{\mathcal{S}}_{\mathrm{RSA}}$.

Functionality 1 $\mathcal{F}^{\mathcal{S}}_{\mathrm{RSA}}(n)$

Inputs: Each party $\mathcal{P}_i$ has a public number $N = pq$, $p \pmod 4, q \pmod 4$, shares $[p]_{\mathbb{Z}}$ and $[q]_{\mathbb{Z}}$, where each share satisfies $\mathfrak{p}_1 \equiv p \pmod 4$, $\mathfrak{q}_1 \equiv q \pmod 4$, and $\mathfrak{p}_i \equiv \mathfrak{q}_i \equiv 0 \pmod 4$ for all $2 \leq i \leq n$.

Outputs:
If $p \equiv q \equiv 3 \pmod 4$:

- If $p \neq q$ are both primes and $\gcd(N, e_4) = 1$, then each party receives $(\texttt{RSAModulus}, \phi)$.
- Otherwise, each party receives $(\texttt{NonRSAModulus}, \{\mathfrak{p}_i, \mathfrak{q}_i\}_{i=1}^n)$.

Else:

- If $p \neq q$ are both primes and $\gcd(N, e_4) = 1$, then each party receives $\left(\texttt{RSAModulus}, \left\{\left[\frac{D_k}{p}\right]\right\}_{D_k \in S_{\min}}\right)$, where

$$S_{\min} := \left\{ D_k \in \mathbf{P}([3, D_{\min}]) \middle| \left[\frac{-D_k}{N}\right] = 1 \right\},$$

  and $D_{\min}$ is the minimal odd prime such that $\left[\frac{-D_{\min}}{p}\right] = -1$ and $\left[\frac{-D_{\min}}{N}\right] = 1$.
- Otherwise, each party receives $(\texttt{NonRSAModulus}, \{\mathfrak{p}_i, \mathfrak{q}_i\}_{i=1}^n)$.

In order to design a protocol to securely compute $\mathcal{F}^{\mathcal{S}}_{\mathrm{RSA}}$, we need functionality $\mathcal{F}_{\mathrm{Leg}}$ to compute the quadratic symbol $\left[\frac{-D}{p}\right]$. A realization of $\mathcal{F}_{\mathrm{Leg}}$ by protocol π_{Leg} is provided later in this section.

Functionality 2 $\mathcal{F}_{\text{Leg}}(n)$

Inputs: Each party $\mathcal{P}_i$ has a share $[p]_{\mathbb{Z}}$, $p \pmod 4$, and a prime D with $\gcd(D, p) = 1$.

Outputs: Each party $\mathcal{P}_i$ receives the value $\left[\frac{-D}{p}\right]$.

The Lucas biprimality test protocol, $\pi^{\mathcal{S}}_{\text{RSA}}$, consists of two main parts: first, verifying that $\gcd(e_4, N) = 1$, and second, performing the exponentiation test described in Theorem 2. The likelihood that N is an RSA modulus increases with each successful iteration of the exponentiation test.

Protocol 1 Lucas Biprimality Test $\pi^{\mathcal{S}}_{\text{RSA}}(n, \kappa)$

Inputs: Each party $\mathcal{P}_i$ has $p \pmod 4$, $q \pmod 4$, N and $[p]_{\mathbb{Z}}, [q]_{\mathbb{Z}}$, where each share satisfies $\mathfrak{p}_1 \equiv p \pmod 4$, $\mathfrak{q}_1 \equiv q \pmod 4$, and $\mathfrak{p}_i \equiv \mathfrak{q}_i \equiv 0 \pmod 4$ for all $2 \leq i \leq n$.

Outputs: $\left(\texttt{RSAModulus}, \left\{\left[\frac{D_k}{p}\right]\right\}_{D_k \in S_{\min}}\right)$ or $(\texttt{NonRSAModulus}, \{\mathfrak{p}_i, \mathfrak{q}_i\}_{i=1}^n)$.

Select an appropriate positive integer D:

1. If $p \equiv q \equiv 3 \pmod 4$, parties set $D = 1$, $S_{\min} := \phi$, and go to the step 5.
2. Else, parties find the minimal k such that $\left[\frac{-D_k}{N}\right] = 1$, where $D_1 = 3, D_2 = 5, D_3 = 7, \ldots$ is the odd prime number sequence.
3. The party $\mathcal{P}_i$ sends $([p]_{\mathbb{Z}}, p \pmod 4, D_k)$ to $\mathcal{F}_{\text{Leg}}$ to obtain $\left[\frac{-D_k}{p}\right]$ and adds D_k to $S_{\min}$.
4. If $\left[\frac{-D_k}{p}\right] = -1$ then parties set $D = D_k$. Else parties find next k such that $\left[\frac{-D_k}{N}\right] = 1$ and restart from step 3.

Exponential verification: For $1 \leq j \leq \kappa$:

5. Parties agree on a random $P_j \in \mathbb{Z}_N$ and let $Q_j := (P_j^2 - D)/4$. If $\gcd(N, Q_j) \neq 1$, then broadcast $\mathfrak{p}_i, \mathfrak{q}_i$ and output $(\texttt{NonRSAModulus}, \{\mathfrak{p}_i, \mathfrak{q}_i\}_{i=1}^n)$.
6. If $\left[\frac{Q_j}{N}\right] \neq 1$, then restart from the previous step.
7. The party $\mathcal{P}_1$ sets $y_{1,j} := (\alpha_j \beta_j^{-1})^{(N + \mathfrak{p}_1\left[\frac{-1}{q}\right] + \mathfrak{q}_1\left[\frac{-1}{p}\right] + \left[\frac{-1}{N}\right])/4}$ and the other parties set $y_{i,j} := (\alpha_j \beta_j^{-1})^{(\mathfrak{p}_i\left[\frac{-1}{q}\right] + \mathfrak{q}_i\left[\frac{-1}{p}\right])/4}$ for all $2 \leq i \leq n$, where α_j and β_j are two roots of the polynomial $x^2 - P_j x + Q_j$. Party $\mathcal{P}_i$ sends $y_{i,j}$ to $\mathcal{F}_{\text{Shuffle}}$ and then obtain u_j.
8. All parties check $u_j \equiv \pm 1 \pmod{N\mathcal{O}_D}$. If the check fails then they broadcast $\mathfrak{p}_i, \mathfrak{q}_i$ and return $(\texttt{NonRSAModulus}, \{\mathfrak{p}_i, \mathfrak{q}_i\}_{i=1}^n)$.

GCD Test

9. Each party randomly generates shares $[r]_N$. They send $\left([r]_N, \left[p[\frac{-1}{q}]+q[\frac{-1}{p}]+\left[\frac{-1}{N}\right]\right]_N\right)$ to $\mathcal{F}_{\text{ModMul}}$ to obtain $[z]_N$.
10. Each party broadcasts his share $\mathfrak{z}_i$ of $[z]_N$, then they check if $\gcd(N, z) = 1$. If the check fails they broadcast $\mathfrak{p}_i, \mathfrak{q}_i$ and return $(\texttt{NonRSAModulus}, \{\mathfrak{p}_i, \mathfrak{q}_i\}_{i=1}^n)$.

If all verification pass, then output $\left(\texttt{RSAModulus}, \left\{\left[\frac{D_k}{p}\right]\right\}_{D_k \in S_{\min}}\right)$.

In practical applications, if we set $p_{\min} = 179$, then 41 iterations are required to ensure the soundness error is less than 2^{-80}. A security proof of $\pi_{\text{RSA}}^{\mathcal{S}}$ under the semi-honest adversary model is provided below.

Theorem 3. *Let p and q be odd integers, $N = pq$, and D be an integer with $\left[\frac{-D}{p}\right] = \left[\frac{-D}{q}\right] = -1$, and $\gcd(D, N) = 1$. The inputs to $\mathcal{P}_i$ are given as*

$$(N, [p]_{\mathbb{Z}}, [q]_{\mathbb{Z}}),$$

*where each share satisfies $\mathfrak{p}_1 \equiv p \pmod 4$, $\mathfrak{q}_1 \equiv q \pmod 4$, and $\mathfrak{p}_i \equiv \mathfrak{q}_i \equiv 0 \pmod 4$ for all $2 \le i \le n$. If $p_{min} \ge 11$, then the Protocol $\pi_{\text{RSA}}^{\mathcal{S}}$ $(n-1)$-*privately computes *the functionality $\mathcal{F}_{\text{RSA}}^{\mathcal{S}}$ in the $\mathcal{F}_{\text{Shuffle}}$, $\mathcal{F}_{\text{ModMul}}$-hybrid model.*

Proof. **Correctness**. Assuming $p < q$ are both primes(i.e., the case $p > q$ is similar) with $\gcd(N, e_4) = 1$, we show that such p and q do not output $(\texttt{NonRSAModulus}, \{p_i, q_i\}_{i=1}^n)$ with overwhelming probability. Note that for any $1 \le j \le \kappa$,

$$\begin{aligned}
&\Pr[\gcd(Q_j, N) = 1] = \Pr[(P_j^2 - D)/4 \in \mathbb{Z}_N^\times] \\
\ge& 1 - \frac{N - \phi(N)}{\phi(N)/4} \ge 1 - \frac{4(p+q-1)}{\phi(N)} \ge 1 - 4\frac{2p-1}{q^2-1} \\
\ge& 1 - \frac{16p}{q^2} \ge 1 - 2^{-\log_2 q + |\log_2 p - \log_2 q| + 4}, \qquad (5)
\end{aligned}$$

which implies that such p, q will pass all tests in step 5 with overwhelming probability(cf. Remark 1). For the check of step 8, by Theorem 2, we have $u_j \equiv \pm 1 \pmod{N\mathcal{O}_D}$ for all $1 \le j \le \kappa$. Using the similar argument as in (5), we may assume $r \in \mathbb{Z}_N^\times$ which implies

$$\gcd(N, z) = \gcd(N, e_4) = 1.$$

The output of $\pi_{\text{RSA}}^{\mathcal{S}}$ is $\left(\texttt{RSAModulus}, \left\{\left[\frac{D_k}{p}\right]\right\}_{D_k \in S_{\min}}\right)$. In the case where $\gcd(N, e_4) \ne 1$, we have $\gcd(N, z) > 1$, and both $\pi_{\text{RSA}}^{\mathcal{S}}$ and $\mathcal{F}_{\text{RSA}}^{\mathcal{S}}$ output $(\texttt{NonRSAModulus}, \{\mathfrak{p}_i, \mathfrak{q}_i\}_{i=1}^n)$. When p and q are not distinct primes but $\gcd(N, e_4) = 1$, the probability of exponential test pass is not greater than $\frac{1}{4} + \frac{1.25}{p_{\min}-3}$, according to Theorem 2. Hence the probability of $\pi_{\text{RSA}}^{\mathcal{S}}$ outputting $\left(\texttt{RSAModulus}, \left\{\left[\frac{D_k}{p}\right]\right\}_{D_k \in S_{\min}}\right)$ is bounded by $(\frac{1}{4} + \frac{1.25}{p_{\min}-3})^\kappa$.

Privacy. Let $\mathcal{P}^*$ be the set of corrupt parties. We show that a simulator $\mathcal{S}$ can be constructed to simulate the transcript of $\pi_{\mathrm{RSA}}^{\mathcal{S}}$. If the input of $\mathcal{S}$ is

$$(\mathcal{P}^*, N, \{\mathfrak{p}_i, \mathfrak{q}_i\}_{i\in\mathcal{P}^*}, \texttt{NonRSAModulus}, \{\mathfrak{p}_i, \mathfrak{q}_i\}_{i=1}^n),$$

then $\mathcal{S}$ only needs to follow the honest parties' strategy to simulate the view of the protocol. Therefore, we consider the case $\mathcal{S}$ is given the input

$$\left(\mathcal{P}^*, N, \{\mathfrak{p}_i, \mathfrak{q}_i\}_{i\in\mathcal{P}^*}, \texttt{RSAModulus}, \left\{\left[\frac{D_k}{p}\right]\right\}_{D_k\in S_{\min}}\right).$$

1: For all $1 \le j \le \kappa$, $\mathcal{S}$ randomly samples $v_j, w_j \in \mathbb{Z}_N$ with $\gcd(v_j^2 - w_j^2 D, N) = 1$, $b_j \in \{0,1\}$, and sets $a_j = \frac{v_j + w_j\sqrt{D}}{v_j - w_j\sqrt{D}}$, $P_j' \in \mathbb{Z}_N$ such that the two roots of polynomial $x^2 - P_j'x + Q_j'$ are $\beta_j' := \frac{\sqrt{D}}{a_j^2\cdot(-1)^{b_j}-1}$ and $\alpha_j' := \beta_j' + \sqrt{D}$.
2: The simulator $\mathcal{S}$ randomly generates $z' \in \mathbb{Z}_N$, and it's additive shares $[z']_N$.
3: The adversary $\mathcal{S}$ outputs

$$\left(\mathcal{P}^*,\ N, \{\mathfrak{p}_i, \mathfrak{q}_i\}_{i\in\mathcal{P}^*}, \{P_j', (-1)^{b_j}\}_{i=1}^{\kappa}, [z']_N, \{z_i'\}_{i=1}^n\right).$$

First, we argue that $P_j' \in \mathbb{Z}_N$ with overwhelming probability. Note that

$$\begin{aligned} P_j' =& \alpha_j' + \beta_j' = \frac{2\sqrt{D}(v_j - w_j\sqrt{D})^2}{(v_j + w_j\sqrt{D})^2\cdot(-1)^{b_j} - (v_j - w_j\sqrt{D})^2} + \sqrt{D} \\ =& \left(\frac{v_j^2 + w_j^2 D}{2v_j w_j}\right)^{1-2b_j} D^{b_j} \in \mathbb{Z}_N. \end{aligned}$$

Secondly, we show that the distribution of $(P_j', (-1)^{b_j})$ generated by the simulator is indistinguishable from the distribution of the real-world transcript $(P_j, u_j) = (P_j, (\alpha_j\beta_j^{-1})^{e_4})$. Note that $(\alpha_j'\beta_j'^{-1})^{e_4} = ((\beta_j' + \sqrt{D})\beta_j'^{-1})^{e_4} = (a_j^2(-1)^{b_j})^{e_4}$. Due to the symmetry between p and q, we only need to consider proving

$$(a_j^2)^{e_4} \equiv 1 \pmod{p\mathcal{O}_D}.$$

Since p, q are odd primes and e_4 is odd, we have

1. If $\left[\frac{D}{p}\right] = -1$, we have, recalling that $\mathbf{N}$ is the norm map from $\mathcal{O}_D$ to $\mathbb{Z}$,

$$\left(\frac{v_j + w_j\sqrt{D}}{v_j - w_j\sqrt{D}}\right)^{2e_4} \equiv \left(\mathbf{N}\left(\frac{v_j + w_j\sqrt{D}}{v_j - w_j\sqrt{D}}\right)\right)^{(q+[\frac{-1}{q}])/2} \equiv 1 \pmod{p\mathcal{O}_D}.$$

2. If $\left[\frac{D}{p}\right] = 1$ (i.e., $\sqrt{D} \in \mathbb{Z}_p^\times$), Euler theorem says that

$$\left(\frac{v_j + w_j\sqrt{D}}{v_j - w_j\sqrt{D}}\right)^{(p+[\frac{-1}{p}])(q+[\frac{-1}{q}])/2} \equiv 1 \pmod{p}.$$

Therefore, $(\alpha'_j \beta'^{-1}_j)^{e_4} \equiv (-1)^{b_j} \pmod{N\mathcal{O}_D}$ by CRT. Note that the distribution of P'_j produced by the simulator $\mathcal{S}$ at the step 1. Proposition 4 says that the distributions of P_j and P'_j are identical. Lastly, $\gcd(N, e_4) = 1$ implies that $(p[\frac{-1}{q}] + q[\frac{-1}{p}] + [\frac{-1}{N}]) \in \mathbb{Z}_N^\times$, and $z \equiv r(p[\frac{-1}{q}] + q[\frac{-1}{p}] + [\frac{-1}{N}]) \pmod{N}$ is uniformly distributed in $\mathbb{Z}_N$. We conclude that the joint distribution of the outputs generated by $\mathcal{S}$ and $\mathcal{F}_{\text{RSA}}^{\mathcal{S}}$, and of the view and output of an execution $\pi_{\text{RSA}}^{\mathcal{S}}$ are indistinguishable. □

Remark 1. In the practical scenario (e.g., [16]), distributed RSA moduli protocols generate $p = \sum_{i=1}^{n} \mathfrak{p}_i$ and $q = \sum_{i=1}^{n} \mathfrak{q}_i$, where $\mathfrak{p}_i$ and $\mathfrak{q}_i$ are uniformly sampled from $[0, 2^{\ell - \log_2 n}]$, with ℓ being the security parameter. This implies $\max\{p, q\}$ is at most ℓ-bits and

$$\mathbb{P}[\min\{p, q\} \text{ is larger than } (\ell - \log_2 n - 80)\text{-bits }] \geq 1 - 2^{-80n}.$$

Therefore, $|\log_2 p - \log_2 q| \leq 80 + \log_2 n$ (i.e., $2^{-\log_2 q + |\log_2 p - \log_2 q| + 4}$ is negligible) with overwhelming probability.

Remark 2. Compared to the Boneh-Franklin protocol, our proposed protocol requires sampling an integer D that satisfies the specific conditions $\left[\frac{-D}{N}\right] = 1$, and $\left[\frac{-D}{p}\right] = -1$. Notably, when $p \equiv q \equiv 3 \pmod 4$, D can be directly chosen as 1, mirroring the Boneh-Franklin case and introducing no additional leakage. In other scenarios, while an integer D satisfying $\left[\frac{-D}{N}\right] = 1$ can often be found without revealing information about p or q, the probability that this D also satisfies $\left[\frac{-D}{p}\right] = -1$ is approximately $1/2$, under the heuristic that $p \pmod D$ is somewhat uniformly distributed in $\mathbb{Z}_D$. Revealing that a candidate D fails the latter condition (i.e., learning the Jacobi symbols $\left[\frac{-D}{p}\right]$ or $\left[\frac{-D}{q}\right]$) could potentially leak some information. The information leakage can be mitigated by limiting the search for a suitable D to a small constant, k (e.g., 5). If all k trials are unsuccessful (i.e. the estimated probability is approximately $1/2^k$.), the protocol aborts and is re-initiated with new, independent secret primes p and q. This approach strictly bounds the information leakage about any single prime p to at most k bits, as information from a failed instance becomes irrelevant for subsequent executions.

For completeness, we provide protocol π_{Leg}, which securely realizes Functionality $\mathcal{F}_{\text{Leg}}$ (cf. Proposition 5). A similar protocol for computing the Legendre symbol was proposed in [26], but a detailed security proof was not provided.

Protocol 2 Legendre symbol $\pi_{\text{Leg}}(n)$

Inputs: Each party $\mathcal{P}_i$ has $[p]_{\mathbb{Z}}$, $p \pmod 4$, and a prime D with $\gcd(D, p) = 1$.

Outputs: $\left[\frac{-D}{p}\right]$.

1. Each party randomly sample $\mathfrak{s}_i \in \mathbb{Z}_D$ sends $(\mathfrak{s}_i, \mathfrak{s}_i, D)$ to $\mathcal{F}_{\text{ModMul}}$ to obtain $[s^2]_D$.

2. Each party sends $([s^2]_D, \mathfrak{p}_i \pmod D, D)$ to $\mathcal{F}_{\text{ModMul}}$ to obtain $[s^2 p]_D$.
3. Each party opens $[s^2 p]_D$. If $\gcd(s^2 p, D) \neq 1$, then restarts to the step 1. Otherwise, output

$$\begin{cases} -\left\lfloor \frac{s^2 p}{D} \right\rfloor, & \text{if } p \equiv 3 \pmod 4 \text{ and } D \equiv 1 \pmod 4; \\ \left\lfloor \frac{s^2 p}{D} \right\rfloor, & \text{otherwise.} \end{cases}$$

4.3 The Lucas Biprimality Test in the Malicious Setting

This section investigates the Lucas biprimality test against malicious adversaries, focusing specifically on the case $p \equiv q \equiv 3 \pmod 4$, which corresponds to $D = 1$. This parameter regime is notable for its cryptographic relevance and distinct structural properties. Our analysis employs the biprimality test functionality $\mathcal{F}_{\text{BI}}^{\mathcal{M}}$, as formalized by Chen et al. [16, Functionality 4.2].

Functionality 3 $\mathcal{F}_{\text{BI}}^{\mathcal{M}}(n)$

Inputs: Each party $\mathcal{P}_i$ has a public number N, shares $[p]_{\mathbb{Z}}$ and $[q]_{\mathbb{Z}}$ with $\mathfrak{p}_i, \mathfrak{q}_i \geq 0$.

Outputs:
If all the following conditions are satisfied, then $\mathcal{F}_{\text{BI}}^{\mathcal{M}}$ send the message `BlumInteger` to the adversary $\mathcal{S}$:

1. All parties agree on the value of N;
2. $N = p \cdot q$;
3. $p \neq q$ are both primes;
4. $p \equiv q \equiv 3 \pmod 4$;
5. $\gcd(N, e_4) = 1$;
6. $\mathfrak{p}_i \geq 0$ and $\mathfrak{q}_i \geq 0$ for all i.

If $\mathcal{S}$ responds with `proceed`, then output `BlumInteger` to all parties. If $\mathcal{S}$ responds with `cheat`, or if any of the previous conditions are false, then output $\{(\mathfrak{p}_i, \mathfrak{q}_i)\}_{i=1}^n$ directly to $\mathcal{S}$, and output `NonBlumInteger` to all parties.

Compared to the protocol by Chen et al., our approach incorporates an explicit check for $p \equiv q \equiv 3 \pmod 4$. Furthermore, we relax the requirement for p and q; instead of needing them to be confined by a fixed upper bound M (i.e., $0 < \mathfrak{p}_i, \mathfrak{q}_i < M$), our analysis only assumes $\mathfrak{p}_i, \mathfrak{q}_i > 0$. While this broader condition on p, q might theoretically include small primes if $p_{\min}$ is not enforced, this is not a concern in our setting as the target functionality is assumed to output p, q forming a Blum integer. Thus, an explicit upper bound M is unnecessary for our protocol's security. Concretely, the removal of M is justified by our adjustment of the protocol coefficients (cf. the second part in Section 4.4).

The Lucas biprimality test is adapted for the malicious setting by leveraging techniques introduced in Chen et al. [16, Protocol 5.2]. A key observation is that the test's soundness error is closely tied to the parameter $p_{\min}$. Consequently, to ensure an overall soundness error bounded by $2^{-\kappa}$, where κ is the security parameter, the number of iterations in our proposed protocol $\pi_{\mathrm{BI}}^{\mathcal{M}}$ is adjusted accordingly, based on $p_{\min}$ and κ. In practical applications, if we set $p_{\min} = 233$, then $\lceil 1.489\kappa \rceil$ iterations are required to ensure the soundness error is less than $2^{-\kappa}$.

Protocol 3 Malicious Lucas Biprimality Test $\pi_{\mathrm{BI}}^{\mathcal{M}}(n, \kappa, p_{\min})$

Inputs: Each party $\mathcal{P}_i$ has N and $[p]_{\mathbb{Z}}, [q]_{\mathbb{Z}}$ with $0 \leq \mathfrak{p}_i, \mathfrak{q}_i$, where each share satisfies $\mathfrak{p}_1 \equiv \mathfrak{q}_1 \equiv 3 \pmod 4$, and $\mathfrak{p}_i \equiv \mathfrak{q}_i \equiv 0 \pmod 4$ for all $2 \leq i \leq n$.

Outputs: `BlumInteger` or `NonBlumInteger`

1. Let $\kappa_{\min} := \left\lceil \left(\log_2 \frac{8p_{\min}-24}{5p_{\min}-10} \right)^{-1} \kappa \right\rceil$. Party $\mathcal{P}_i$ samples $\tau_{i,j} \leftarrow \mathbb{Z}_{2^{2\kappa-1}n^3 N \kappa_{\min}}$ for $1 \leq j \leq \kappa_{\min}$, and sends $\big(\texttt{commit}, i, (\mathfrak{p}_i, \mathfrak{q}_i, \{\tau_{i,j}\}_{j=1}^{\kappa_{\min}})\big)$ to functionality $\mathcal{F}_{\mathrm{ComCompute}}$.
2. Each party $\mathcal{P}_i$ sends `sample` to $\mathcal{F}_{\mathrm{Zero}}(2^{\kappa-3}nN)$ and receives r_i in response.
3. For $1 \leq j \leq \kappa_{\min}$, the parties invoke $\mathcal{F}_{\mathrm{CT}}(\mathcal{Z}^{+1}(1, N))$ to obtain $\{(P_j, Q_j)\}_{j=1}^{\kappa_{\min}}$.
4. Party $\mathcal{P}_1$ sets $y_{1,j} := (\alpha_j \beta_j^{-1})^{r_1 + (\mathfrak{p}_1 \left[\frac{-1}{q}\right] + \mathfrak{q}_1 \left[\frac{-1}{p}\right] + 6)/4}$, and the other parties set $y_{i,j} := (\alpha_j \beta_j^{-1})^{r_i + (\mathfrak{p}_i \left[\frac{-1}{q}\right] + \mathfrak{q}_i \left[\frac{-1}{p}\right])/4}$ for all $2 \leq i \leq n$, $1 \leq j \leq \kappa_{\min}$. Here α_j and β_j are two roots of the polynomial $x^2 - P_j x + Q_j$. Party $\mathcal{P}_i$ sends $(\texttt{commit}, i, \{y_{i,j}\}_{j=1}^{\kappa_{\min}}, \{1, \ldots, n\})$ to $\mathcal{F}_{\mathrm{Com}}$.
5. Party $\mathcal{P}_i$ sends $(\texttt{decommit}, i)$ to $\mathcal{F}_{\mathrm{Com}}$ and receives $\{y_{i',j}\}_{j=1}^{\kappa_{\min}}$ for $i' \neq i$.
6. The parties output `NonBlumInteger` and halt if there exists $1 \leq j \leq \kappa_{\min}$ such that
$$(\alpha_j \beta_j^{-1})^{(N-5)/4} \cdot \prod_{i=1}^{n} y_{i,j} \not\equiv \pm 1 \pmod N.$$
7. For $1 \leq j \leq \kappa_{\min}$, each party $\mathcal{P}_i$ computes
$$\gamma_{i,j} = (\alpha_j \beta_j^{-1})^{\tau_{i,j}} \pmod N,$$
and broadcasts $\{\gamma_{i,j}\}_{j=1}^{\kappa_{\min}}$.
8. All parties send `flip` to $\mathcal{F}_{\mathrm{CT}}(\{0,1\}^{\kappa_{\min}})$ and then obtain an agreed-upon random bit vector $\vec{c} = (c_i)$ of length $\kappa_{\min}$.
9. For $1 \leq j \leq \kappa_{\min}$, the party $\mathcal{P}_1$ computes
$$\zeta_{1,j} = \tau_{1,j} + c_j \cdot \big(r_1 - (\mathfrak{p}_1 + \mathfrak{q}_1 - 6)/4\big),$$
and every other party $\mathcal{P}_i$ for $2 \leq i \leq n$ computes
$$\zeta_{i,j} = \tau_{i,j} + c_j \cdot \big(r_i - (\mathfrak{p}_i + \mathfrak{q}_i)/4\big).$$
They all broadcast the values they have computed to one another.

10. The parties halt and output `NonBlumInteger` if there exists any $1 \leq j \leq \kappa_{\min}$ such that
$$\prod_{i=1}^{n}(\alpha_j\beta_j^{-1})^{\zeta_{i,j}} \not\equiv \prod_{i=1}^{n}\gamma_{i,j}\cdot y_{i,j}^{c_j} \pmod N.$$
11. Let C be a circuit computing $\pi_{\text{VerifyBI}}(N,\vec{c},\{\cdot,\cdot,\cdot,\zeta_{i,*}\}_{i\in\{1,\ldots,n\}})$; that is, let it be a circuit representation of Algorithm π_{VerifyBI} with the public values N, $\vec{c}$, and ζ hardcoded. The parties send $(\texttt{compute}, 1, \{1,\ldots,n\}, C)$ to $\mathcal{F}_{\text{ComCompute}}$, and in response they all receive z or `VerifyFail`. If receive `VerifyFail`, or if $\mathcal{F}_{\text{ComCompute}}$ aborts, then the parties halt and output `NonBlumInteger`.
12. The parties halt and output `BlumInteger` if $\gcd(z,N) = 1$, or halt and output `NonBlumInteger` otherwise.

Algorithm π_{VerifyBI} (cf. [16, Algorithm 5.3]) is employed to verify the relationships between $\mathfrak{p}_i, \mathfrak{q}_i, \tau_{i,j}, \zeta_{i,j}, \vec{c}$, N, thereby confirming the correctness of $\mathfrak{p}_i$ and $\mathfrak{q}_i$ and the consistency of the protocol transcript. If all checks pass, then π_{VerifyBI} outputs $z := r\big(-1+\sum_{i=1}^{n}(\mathfrak{p}_i+\mathfrak{q}_i)\big)$ for some random r, which isused for the GCD test.

Algorithm 0 $\pi_{\text{VerifyBI}}(N,\vec{c},\{\mathfrak{p}_i,\mathfrak{q}_i,\{\tau_{i,j}\}_{j=1}^{\kappa_{\min}},\{\zeta_{i,j}\}_{j=1}^{\kappa_{\min}}\}_{i=1}^{n})$

---

1. Sample $r \leftarrow \mathbb{Z}_N$ and compute
$$z = r\cdot\left(-1+\sum_{i=1}^{n}(\mathfrak{p}_i+\mathfrak{q}_i)\right) \pmod N.$$
2. Return z if and only if it holds that
$$\begin{aligned}
N &= \left(\sum_{i=1}^{n}\mathfrak{p}_i\cdot\sum_{i=1}^{n}\mathfrak{q}_i\right)\\
&\wedge \sum_{i=1}^{n}\mathfrak{p}_i \equiv \sum_{i=1}^{n}\mathfrak{q}_i \equiv 3 \pmod 4\\
&\wedge \big(\mathfrak{p}_i \geq 0\big)\wedge\big(\mathfrak{q}_i\geq 0\big) \text{ for all } 1\leq i\leq n\\
&\wedge \Big(\tau_{1,j} = \zeta_{1,j} + c_j\cdot\big(r_1-(\mathfrak{p}_1+\mathfrak{q}_1-6)\big)/4\Big) \text{ for all } 1\leq j\leq\kappa_{\min}\\
&\wedge \Big(\tau_{i,j} = \zeta_{i,j} + c_j\cdot\big(r_i-(\mathfrak{p}_i+\mathfrak{q}_i)/4\big)\Big) \text{ for all } 2\leq i\leq n \text{ and } 1\leq j\leq\kappa_{\min}.
\end{aligned}$$

If any part of the above predicate does not hold, output `VerifyFail`.

Theorem 4. *Let $\kappa > 0$ be a security parameter. The inputs to all participants $\{\mathcal{P}_i\}_{i=1}^{n}$ are given as $(N,[p]_{\mathbb{Z}},[q]_{\mathbb{Z}})$. If $p_{min} \geq 11$, then the Protocol $\pi_{\text{BI}}^{\mathcal{M}}$ $(n-1)$-*securely computes *the functionality $\mathcal{F}_{\text{BI}}^{\mathcal{M}}$* with abort *in the $\mathcal{F}_{\text{ComCompute}}$, $\mathcal{F}_{\text{Zero}}$, $\mathcal{F}_{\text{CT}}$, $\mathcal{F}_{\text{Com}}$-hybrid model. Here $p_{\min}$ is the minimal prime factor of N.*

Proof. Let $\mathcal{A}$ be a real-world adversary, and let $\mathcal{P}^*$ (resp. $\overline{\mathcal{P}^*}$) denote the set of corrupted (resp. honest) parties. We construct the simulator $\mathcal{S}$ that operates as follows:

1. The simulator $\mathcal{S}$ simulates $\mathcal{F}_{\text{ComCompute}}$ to obtain $(\texttt{commit}, i, (\mathfrak{p}'_i, \mathfrak{q}'_i, \{\tau_{i,j}\}_{j=1}^{\kappa_{\min}}))$ for each $i \in \mathcal{P}^*$. Here $\kappa_{\min} := \left\lceil \left(\log_2 \frac{8p_{\min}-24}{5p_{\min}-10}\right)^{-1} \kappa \right\rceil$.
2. The simulator $\mathcal{S}$ sends $\{(N, \mathfrak{p}'_i, \mathfrak{q}'_i)\}_{i\in\mathcal{P}^*}$ to the functionality $\mathcal{F}_{\text{BI}}^{\mathcal{M}}$ to obtain $\texttt{BlumInteger}$ or $(\texttt{NonBlumInteger}, \{\mathfrak{p}_i, \mathfrak{q}_i\}_{i=1}^n)$. If $\mathcal{S}$ receives $(\texttt{NonBlumInteger}, \{\mathfrak{p}_i, \mathfrak{q}_i\}_{i=1}^n)$, then $\mathcal{S}$ follows the honest parties' strategies in $\pi_{\text{BI}}^{\mathcal{M}}$ with shares $\{\mathfrak{p}_i, \mathfrak{q}_i\}_{i\in\overline{\mathcal{P}^*}}$. Otherwise, $\mathcal{S}$ proceeds with the following steps.
3. The simulator $\mathcal{S}$ follows the strategy of functionality $\mathcal{F}_{\text{Zero}}(2^{\kappa-3}nN)$, samples $\{r_i\}_{i=1}^n$, and sends $\{r_i\}_{i\in\mathcal{P}^*}$ to $\mathcal{A}$.
4. For each $1 \le j \le \kappa_{\min}$, the simulator $\mathcal{S}$ randomly samples $v_j, w_j \in \mathbb{Z}_N$ such that $\gcd(v_j^2 - w_j^2 D, N) = 1$, and $b_j \in \{0,1\}$. It then sets $a_j = \frac{v_j + w_j\sqrt{D}}{v_j - w_j\sqrt{D}}$, $P_j \in \mathbb{Z}_N$ such that the two roots of the polynomial $x^2 - P_j x + Q_j$ are
$$\beta_j := \frac{\sqrt{D}}{a_j^2 \cdot (-1)^{b_j} - 1} \quad \text{and} \quad \alpha_j := \beta_j + \sqrt{D}.$$
The simulator $\mathcal{S}$ sends (P_j, Q_j) to $\mathcal{A}$ for each $1 \le j \le \kappa_{\min}$.
5. The simulator $\mathcal{S}$ simulates $\mathcal{F}_{\text{Com}}$ to receive $(\texttt{commit}, i, \{y_{i,j}\}_{j=1}^{\kappa_{\min}}, \{1, \ldots, n\})$ from $\mathcal{A}$ for each $i \in \mathcal{P}^*$.
6. Let $y'_{i,j}$ denote the value of, for $i \in \mathcal{P}^*$,
$$\begin{cases} (\alpha_j\beta_j^{-1})^{r_i + (\mathfrak{p}'_i\left[\frac{-1}{q}\right] + \mathfrak{q}'_i\left[\frac{-1}{p}\right] + 6)/4}, & \text{if } i = 1; \\ (\alpha_j\beta_j^{-1})^{r_i + (\mathfrak{p}'_i\left[\frac{-1}{q}\right] + \mathfrak{q}'_i\left[\frac{-1}{p}\right])/4}, & \text{if } 2 \le i \le n. \end{cases}$$
Let i' be the minimal number such that $\mathcal{P}_{i'} \in \overline{\mathcal{P}^*}$. The simulator $\mathcal{S}$ randomly samples $r'_i \leftarrow \mathbb{Z}_{2^{\kappa-2}nN}$ and sets $y_{i,j} := (\alpha_j\beta_j^{-1})^{r'_i}$ for each $i \in \overline{\mathcal{P}^*} \setminus \{i'\}$ and $1 \le j \le \kappa_{\min}$. Additionally, for $1 \le j \le \kappa_{\min}$, $\mathcal{S}$ sets $y_{i',j}$ be the value such that
$$(\alpha_j\beta_j^{-1})^{(N-5)/4} \prod_{i\in\overline{\mathcal{P}^*}} y_{i,j} \prod_{i\in\mathcal{P}^*} y'_{i,j} \equiv (-1)^{b_j} \pmod N,$$
and sends $(\texttt{decommitted}, i, \{y_{i,j}\}_{j=1}^{\kappa_{\min}})$ to $\mathcal{A}$ for each $i \in \overline{\mathcal{P}^*}$.
7. The simulator $\mathcal{S}$ checks
$$(\alpha_j\beta_j^{-1})^{(N-5)/4} \prod_{i=1}^{n} y_{i,j} \equiv \pm 1 \pmod N$$
for each $1 \le j \le \kappa_{\min}$. If any check fails, then $\mathcal{S}$ sends $\texttt{cheat}$ to the functionality $\mathcal{F}_{\text{BI}}^{\mathcal{M}}$ and outputs the output of $\mathcal{A}$.
8. For each $1 \le j \le \kappa_{\min}$, the simulator $\mathcal{S}$ randomly samples $c_j \leftarrow \{0,1\}$ and $\zeta_{i,j} \leftarrow \mathbb{Z}_{2^{2\kappa-1}n^3N\kappa_{\min}}$ for each $i \in \overline{\mathcal{P}^*}$. It sends $\{\gamma_{i,j}\}_{j=1}^{\kappa_{\min}} := \{(\alpha_j\beta_j^{-1})^{\zeta_{i,j}} \cdot y_{i,j}^{-c_j}\}_{j=1}^{\kappa_{\min}}$ for each $i \in \overline{\mathcal{P}^*}$ to $\mathcal{A}$ and receives $\{\gamma_{i,j}\}_{j=1}^{\kappa_{\min}}$ from $\mathcal{A}$ for each $i \in \mathcal{P}^*$.

9. The simulator $\mathcal{S}$ simulates $\mathcal{F}_{\mathrm{CT}}$, and sends $\{c_j\}_{j=1}^{\kappa_{\min}}$ to $\mathcal{A}$.
10. The simulator $\mathcal{S}$ sends $\{\zeta_{i,j}\}_{j=1}^{\kappa_{\min}}$ to $\mathcal{A}$ for each $i \in \overline{\mathcal{P}^*}$ and receives $\{\zeta_{i,j}\}_{j=1}^{\kappa_{\min}}$ for each $i \in \mathcal{P}^*$ from $\mathcal{A}$.
11. The simulator $\mathcal{S}$ checks

$$\prod_{i=1}^{n}(\alpha_j\beta_j^{-1})^{\zeta_{i,j}} \equiv \prod_{i=1}^{n}\gamma_{i,j}\cdot y_{i,j}^{c_j} \pmod N$$

for each $1 \le j \le \kappa_{\min}$. If any check fails, then $\mathcal{S}$ sends `cheat` to $\mathcal{F}_{\mathrm{BI}}^{\mathcal{M}}$and outputs the output of $\mathcal{A}$.
12. The simulator $\mathcal{S}$ simulates $\mathcal{F}_{\mathrm{ComCompute}}$ to obtain $(\texttt{compute}, 1, \{1,\ldots,n\}, C)$, if C is the circuit described in $\pi_{\mathrm{BI}}^{\mathcal{M}}$ step 11 and $(\tau_{i,j}, \zeta_{i,j}, c_j, \mathfrak{p}_i', \mathfrak{q}_i')$ satisfy the equations

$$\begin{cases}\tau_{i,j} = \zeta_{i,j} + c_j \cdot \big((r_i - (\mathfrak{p}_i' + \mathfrak{q}_i' - 6)/4\big), & \text{if } i = 1;\\ \tau_{i,j} = \zeta_{i,j} + c_j \cdot \big((r_i - (\mathfrak{p}_i' + \mathfrak{q}_i')/4\big), & \text{if } 2 \le i \le n,\end{cases}$$

for each $i \in \mathcal{P}^*$ and $1 \le j \le \kappa_{\min}$, then randomly samples $z \leftarrow \mathbb{Z}_N$ and sends $(\texttt{result}, 1, z)$ to $\mathcal{A}$. Otherwise, $\mathcal{S}$ sends `cheat` to $\mathcal{F}_{\mathrm{BI}}^{\mathcal{M}}$ and outputs the output of $\mathcal{A}$.
13. The simulator $\mathcal{S}$ sends `proceed` to $\mathcal{F}_{\mathrm{BI}}^{\mathcal{M}}$ and outputs the output of $\mathcal{A}$.

We proceed with the proof by considering two cases, depending on the value obtained by the simulator $\mathcal{S}$ from $\mathcal{F}_{\mathrm{BI}}^{\mathcal{M}}$ in Step 2.

Case 1: $\mathcal{S}$ obtains `BlumInteger` from $\mathcal{F}_{\mathrm{BI}}^{\mathcal{M}}$.

To demonstrate the indistinguishability of the real and ideal world distributions, we employ the following hybrid worlds, where $\mathcal{H}_i$ denotes the joint distribution of the simulator's output and $\mathcal{F}_{\mathrm{BI}}^{\mathcal{M}}$'s output.

Hybrid 0: The ideal world $\mathcal{H}_0$.

Hybrid 1: This hybrid world $\mathcal{H}_1$ is the same as $\mathcal{H}_0$, except that:

(1) The simulator $\mathcal{S}_1$ is given the honest parties' shares $\{\mathfrak{p}_i, \mathfrak{q}_i\}_{i\in\overline{\mathcal{P}^*}}$ as auxiliary input.

(2) In Step 4, $\mathcal{S}_1$ randomly samples (P_j, Q_j) from $\mathcal{Z}^{+1}(1, N)$ for each $1 \le j \le \kappa_{\min}$.

(3) In Step 6, the values $y_{i,j}$ are randomly sampled from $\mathbb{Z}_N^{\times}$ such that

$$\begin{aligned}&(\alpha_j\beta_j^{-1})^{(N-5)/4}\prod_{i\in\overline{\mathcal{P}^*}} y_{i,j}\prod_{i\in\mathcal{P}^*} y_{i,j}'\\ \equiv&(\alpha_j\beta_j^{-1})^{(\sum_{i\in\mathcal{P}^*}\mathfrak{p}_i' + \sum_{i\in\overline{\mathcal{P}^*}}\mathfrak{p}_i - 1)(\sum_{i\in\mathcal{P}^*}\mathfrak{q}_i' + \sum_{i\in\overline{\mathcal{P}^*}}\mathfrak{q}_i - 1)/4} \pmod N,\end{aligned}$$

for each $i \in \overline{\mathcal{P}^*}$ and $1 \le j \le \kappa_{\min}$.

(4) In Step 7, the check is replaced by, for each $1 \le j \le \kappa_{\min}$,

$$\begin{aligned}&(\alpha_j\beta_j^{-1})^{(N-5)/4}\prod_{i=1}^{n} y_{i,j}\\ \equiv&(\alpha_j\beta_j^{-1})^{(\sum_{i\in\mathcal{P}^*}\mathfrak{p}_i' + \sum_{i\in\overline{\mathcal{P}^*}}\mathfrak{p}_i - 1)(\sum_{i\in\mathcal{P}^*}\mathfrak{q}_i' + \sum_{i\in\overline{\mathcal{P}^*}}\mathfrak{q}_i - 1)/4} \pmod N.\end{aligned}$$

Hybrid 2: This hybrid world $\mathcal{H}_2$ is the same as $\mathcal{H}_1$, except that the values $y_{i,j}$ in Step 6 are defined as follows:

$$\begin{cases} (\alpha_j\beta_j^{-1})^{r_i+(\mathfrak{p}_i'\left[\frac{-1}{q}\right]+\mathfrak{q}_i'\left[\frac{-1}{p}\right]+6)/4}, & \text{if } i=1, \\ (\alpha_j\beta_j^{-1})^{r_i+(\mathfrak{p}_i'\left[\frac{-1}{q}\right]+\mathfrak{q}_i'\left[\frac{-1}{p}\right])/4}, & \text{if } 2\le i\le n, \end{cases}$$

for each $1 \le j \le \kappa_{\min}$.

Hybrid 3: This hybrid world $\mathcal{H}_3$ is the same as $\mathcal{H}_2$, except that in Step 8, the simulator $\mathcal{S}_3$ randomly samples $\tau_{i,j} \leftarrow \mathbb{Z}_{2^{2\kappa-1}n^3N\kappa_{\min}}$ and sets $\gamma_{i,j} := (\alpha_j\beta_j^{-1})^{\tau_{i,j}}$ and

$$\zeta_{i,j} := \begin{cases} \tau_{i,j} - c_j \cdot \big((r_i - (\mathfrak{p}_i + \mathfrak{q}_i - 6)/4\big), & \text{if } i=1, \\ \tau_{i,j} - c_j \cdot \big((r_i - (\mathfrak{p}_i + \mathfrak{q}_i)/4\big), & \text{if } 2\le i\le n \end{cases}$$

for each $1 \le j \le \kappa_{\min}$ and each $i \in \overline{\mathcal{P}^*}$.

Hybrid 4: The real world $\mathcal{H}_4$.

$\mathcal{H}_0 \overset{\text{c}}{\equiv} \mathcal{H}_1$: The distributions of $\mathcal{H}_0$ and $\mathcal{H}_1$ are identical; an analogous proof establishing this indistinguishability is provided in the privacy proof of Theorem 3.

$\mathcal{H}_1 \overset{\text{c}}{\equiv} \mathcal{H}_2$: We first observe that since N is an RSA modulus, $(\alpha_j\beta_j^{-1})^{e_4} \equiv \pm 1 \pmod N$ for $1 \le j \le \kappa_{\min}$. This implies that the order of $\alpha_j\beta_j^{-1}$ is at most $2e_4$ for all j. To bound the statistical distance SD between $\mathcal{H}_1$ and $\mathcal{H}_2$, we employ Proposition 6 with $A = 2e_4$ and $B = 2^{\kappa-3}nN$, and Proposition 8 with $A = 2e_4$ and $B = 2^{\kappa-2}nN$, yielding:

$$\text{SD}(\mathcal{H}_1, \mathcal{H}_2) \le \frac{n}{4\lfloor(2^{\kappa-2}nN+1)/(2e_4)\rfloor} + \frac{n}{4\lfloor(2^{\kappa-2}nN)/(2e_4)\rfloor} \le \frac{1}{2^{\kappa+1}}.$$

$\mathcal{H}_2 \overset{\text{c}}{\equiv} \mathcal{H}_3$: Proposition 7 demonstrates that the statistical distance between the distributions of $\zeta_{i,j}$ generated in **Hybrid 2** and **Hybrid 3** is at most $\frac{n\kappa_{\min}}{2^{\kappa+1}n\kappa_{\min}} < \frac{1}{2^{\kappa+1}}$, by setting $A = 2^{\kappa-2}n^2N$ and $B = 2^{\kappa+1}n\kappa_{\min}$. We note that $-A \le c_j\big(r_i - (\mathfrak{p}_i + \mathfrak{q}_i)/4\big) \le A$ for all $1 \le i \le n$ and $1 \le j \le \kappa_{\min}$, given that $\mathfrak{p}_i, \mathfrak{q}_i \ge 0$ and $N = (\sum_{i=1}^n \mathfrak{p}_i) \cdot (\sum_{i=1}^n \mathfrak{q}_i)$. Consequently, the joint distributions of $(\gamma_{i,j}, c_j, \zeta_{i,j})$ generated in **Hybrid 2** and **Hybrid 3** are indistinguishable, and

$$\text{SD}(\mathcal{H}_2, \mathcal{H}_3) \le \frac{1}{2^{\kappa+1}}.$$

$\mathcal{H}_3 \overset{\text{c}}{\equiv} \mathcal{H}_4$: Note that the view of the real-world adversary $\mathcal{A}$ in **Hybrid 3** and in the real world is identical. Additionally, $\mathcal{S}_3$ sends `cheat` to $\mathcal{F}_{\text{BI}}^{\mathcal{M}}$ if and only if the honest parties output 0. Hence, the distributions of $\mathcal{H}_3$ and $\mathcal{H}_4$ are identical.

Case 2: $\mathcal{S}$ obtains $(\texttt{NonBlumInteger}, \{\mathfrak{p}_i, \mathfrak{q}_i\}_{i=1}^n)$ from $\mathcal{F}_{\text{BI}}^{\mathcal{M}}$.

Since $\mathcal{S}$ possesses the honest parties' shares and faithfully follows $\pi_{\text{BI}}^{\mathcal{M}}$, it suffices to prove that the parties output `NonBlumInteger` except with negligible probability in the real execution. Note that all check conditions of $\mathcal{F}_{\text{BI}}^{\mathcal{M}}$, except condition 3 (i.e., that $p \ne q$ and both p and q are primes), are directly checked

in $\mathcal{F}_{\mathrm{BI}}^{\mathcal{M}}$. It remains to show that the parties output `NonBlumInteger` in a real execution except with negligible probability if $(\sum_{i\in\mathcal{P}^*}\mathfrak{p}'_i+\sum_{i\in\overline{\mathcal{P}^*}}\mathfrak{p}_i)\cdot(\sum_{i\in\mathcal{P}^*}\mathfrak{q}'_i+\sum_{i\in\overline{\mathcal{P}^*}}\mathfrak{q}_i)$ is not a Blum integer.

We now consider the probability that $\mathcal{A}$ passes both Step 7 and Step 10. Consider the check in Step 10. Let $\delta_{1,j}$ and $\delta_{2,j}$ be the offsets such that

$$(\alpha_j\beta_j^{-1})^{(N-5)/4}\prod_{i\in\overline{\mathcal{P}^*}}y_{i,j}\prod_{i\in\mathcal{P}^*}y'_{i,j}\cdot(\alpha_j\beta_j^{-1})^{\delta_{1,j}}\equiv\pm1\pmod N,$$

$$(\alpha_j\beta_j^{-1})^{\delta_{2,j}}\cdot\prod_{i=1}^{n}\gamma_{i,j}\equiv\prod_{i=1}^{n}(\alpha_j\beta_j^{-1})^{\tau_{i,j}}\pmod N$$

for all $1\le j\le\kappa_{\min}$.

Combining this with the check on $\zeta_{i,j}$ in $\mathcal{F}_{\mathrm{BI}}^{\mathcal{M}}$, we have

$$(\alpha_j\beta_j^{-1})^{\delta_{2,j}}\equiv(\alpha_j\beta_j^{-1})^{c_j\cdot\delta_{1,j}}\pmod N$$

for all $1\le j\le\kappa_{\min}$. Since the adversary $\mathcal{A}$ must choose $\delta_{1,j}$ and $\delta_{2,j}$ before it learns c_j, for each j, the probability that adversary $\mathcal{A}$ passes Step 10 in $\pi_{\mathrm{BI}}^{\mathcal{M}}$ with $\delta_{1,j}\neq0$ is at most $\frac{1}{2}$. Using Theorem 2, the probability that $\mathcal{A}$ must set non-zero $\delta_{1,j}$ is $\frac{1}{4}+\frac{1.25}{p_{\min}-3}$ for each j. Therefore, the probability that $\mathcal{A}$ passes Step 7 and Step 10 is at most

$$\frac{1}{4}+\frac{1.25}{p_{\min}-3}+\left(1-(\frac{1}{4}+\frac{1.25}{p_{\min}-3})\right)\cdot\frac{1}{2}=\frac{5}{8}+\frac{0.625}{p_{\min}-3}$$

for each j. We have

$$\begin{aligned}&\Pr[\pi_{\mathrm{BI}}^{\mathcal{M}}\text{ outputs }\texttt{BlumInteger}\mid N\text{ is not a Blum integer}]\\&\le\left(\frac{5}{8}+\frac{0.625}{p_{\min}-3}\right)^{\kappa_{\min}}\le\frac{1}{2^{\kappa}}.\end{aligned}\tag{6}$$

Finally, considering **Case 1** (which utilizes the triangle inequality) and **Case 2**, for any non-uniform probabilistic polynomial-time distinguisher $\mathcal{D}$, it follows that:

$$\Big|\Pr\Big[\mathcal{D}\big(\{\mathrm{REAL}_{\pi_{\mathrm{BI}}^{\mathcal{M}},\mathcal{A},\mathcal{P}^*}(\lambda,\{(\mathfrak{p}_i,\mathfrak{q}_i,N)\}_{i=1}^n,\mathbf{aux})\}_{\lambda,\mathfrak{p}_i,\mathfrak{q}_i,N,\mathbf{aux}},\mathbf{aux}\big)\Big]$$
$$-\Pr\Big[\mathcal{D}\big(\{\mathrm{IDEAL}_{\mathcal{F}_{\mathrm{BI}}^{\mathcal{M}},\mathcal{S},\overline{\mathcal{P}^*}}(\lambda,\{(\mathfrak{p}_i,\mathfrak{q}_i,N)\}_{i=1}^n,\mathbf{aux})\}_{\lambda,\mathfrak{p}_i,\mathfrak{q}_i,N,\mathbf{aux}},\mathbf{aux}\big)\Big]\Big|\le\frac{1}{2^{\kappa}}.$$

□

The functionality $\mathcal{F}_{\mathrm{ComCompute}}$ (cf. [16, Functionality A.4]) allows parties to first commit to their respective inputs x_i. Subsequently, the parties can jointly select a public function f, and the functionality then outputs the result $f(x_1,\ldots,x_n)$.

Functionality 4 $\mathcal{F}_{\text{ComCompute}}(n)$

Input Commitment: Upon receiving $(\texttt{commit}, \texttt{id}, x)$ from party $\mathcal{P}_i$, if $\texttt{id}$ is a fresh value, then store $(\texttt{value}, \texttt{id}, i, x)$ in memory, and send $(\texttt{committed}, \texttt{id}, i)$ to all other parties.

Computation: Upon receiving $(\texttt{compute}, \texttt{id}, \texttt{input-ids}, f)$ from all parties, where $\texttt{id}$ is a fresh, agreed upon value, and where $\texttt{input-ids}$ is a vector of IDs such that for every $1 \leq i \leq |\texttt{input-ids}|$ there exists in memory a record of the form $(\texttt{value}, \texttt{input-ids}_i, *, *)$, and where f is the description of a function that takes as input the values associated with the IDs in $\texttt{input-ids}$ and produces as output an n-tuple of values, if the parties disagree upon the function f or the vector $\texttt{input-ids}$, then abort, and otherwise:

1. Let x be a vector of the same length as $\texttt{input-ids}$ such that for $1 \leq i \leq |\texttt{input-sids}|$, there exists in memory a record of the form $(\texttt{value}, \texttt{input-ids}_i, *, v)$ such that $x_i = v$.
2. Compute $(y_1, \ldots, y_n) := f(x)$, and then send $(\texttt{result}, \texttt{id}, y_i)$ to each party $\mathcal{P}_i$.

The functionality $\mathcal{F}_{\text{Zero}}$ (cf. [16, Functionality A.3.]) takes as input a number of parties n and a bound B, and outputs shares r_i for each party such that $\sum_{i=1}^{n} r_i = 0$.

Functionality 5 $\mathcal{F}_{\text{Zero}}(n, B)$

Inputs: Each party $\mathcal{P}_i$ has input $\texttt{sample}$.

Outputs:

1. Uniformly sample $x_{i,j} \leftarrow \{-B, -B+1, \ldots, B\}$ for $1 \leq i, j \leq n$ such that $x_{i,j} + x_{j,i} = 0$.
2. Each party $\mathcal{P}_i$ receives the value $r_i := \sum_{j=1}^{n} x_{i,j}$.

We employ standard coin-tossing (cf. [16, Functioanlity A.1.]) and commitment (cf. [16, Functioanlity A.2.]) functionalities.

Functionality 6 $\mathcal{F}_{\mathrm{CT}}(n, S)$

Inputs: Each party $\mathcal{P}_i$ has input `flip`.

Outputs: Each party $\mathcal{P}_i$ receives the value c, where c is uniformly sampled from the set S.

Functionality 7 $\mathcal{F}_{\mathrm{Com}}(n)$

Commit: On receiving $(\texttt{commit}, \texttt{id}, x, J)$ from party $\mathcal{P}_i$, where $J \subseteq \{1, \ldots n\}$ and $x \in \{0,1\}^*$, if `id` is a fresh value, then store $(\texttt{commitment}, \texttt{id}, x, J, i)$ in memory and send $(\texttt{committed}, \texttt{id}, i)$ to each party $\mathcal{P}_j$ for $j \in J$.

Decommit: On receiving $(\texttt{decommit}, \texttt{id})$ from $\mathcal{P}_i$, if a record of the form $(\texttt{commitment}, \texttt{id}, x, J, i)$ exists in memory, then send $(\texttt{decommitted}, \texttt{id}, x)$ to every party $\mathcal{P}_j$ for $j \in J$.

4.4 Technical Observations and Potential Enhancements for Chen et al.'s Protocol

This section offers several suggestions, notably an optimization for the number of iterations in the Chen et al. protocol [16, Protocol 5.2]. This refinement stems from our updated worst-case soundness error for the Boneh-Franklin test: while the original test yielded a false positive with at most $1/2$ probability, our new analysis (cf. Theorem 1) reduces this probability to $1/4$. Therefore, assuming the adversary always cheats and considering an honest execution would not have yielded a false positive, the probability of producing a positive outcome in the jth iteration (i.e., Steps 4–6 and 8–11) is upper-bounded by $5/8$ (cf. the inequality (6)). Consequently, the probability that the adversary succeeds across all $\lceil 1.475s \rceil$ (i.e., the original result gives $2.5s$) iterations is at most $(5/8)^{1.475s} < 2^{-s}$. A brief description of our suggestion is provided below. A revising protocol from Chen et al. can be found in Section 6.11.

1. **Lack of Congruence Verification for p and q.**
   The VarifyBiprime algorithm by Chen et al. [16, Algorithm 5.3] does not enforce the condition that both prime factors p and q of $N = pq$ are congruent to 3 (mod 4). This omission presents a vulnerability: an adversary could select primes $p \equiv q \equiv 1 \pmod 4$, potentially causing specific variants of the Boneh-Franklin test (i.e., or its associated soundness analysis, which often assumes $p \equiv q \equiv 3 \pmod 4$) to behave unexpectedly or fail to provide its intended security guarantees. Indeed, as detailed in Lemma 10, infinitely many RSA moduli $N = pq$ with $p \equiv q \equiv 1 \pmod 4$ consistently pass certain

Boneh-Franklin test configurations, even if those configurations are primarily analyzed for 3 (mod 4) primes.

To address this, one possible enhancement is to augment the verification procedure with an explicit check for $p \equiv q \equiv 3 \pmod 4$. An alternative approach, applicable if using encoding components (e.g., shares $\mathfrak{p}_i, \mathfrak{q}_i$ of p, q), could be to enforce constraints directly at the encoding stage, such as requiring primary shares $\mathfrak{p}_1, \mathfrak{q}_1 \equiv 3 \pmod 4$ while other shares $\mathfrak{p}_i, \mathfrak{q}_i$ for $i > 1$ satisfy $\mathfrak{p}_i, \mathfrak{q}_i \equiv 0 \pmod 4$.

2. **Mismatch Between Proof Structure and Commitment Definition.** The security proof follows a structure analogous to Schnorr's protocol. Consequently, the definition of the responses $\zeta_{i,j}$ in the challenge-response phase (cf. Step 10) must correctly incorporate the randomness r_i used in the commitment phase (cf. Step 4). Specifically, for commitments $\chi_{i,j}$ (e.g., $\gamma_j^{r_i-(p_i+q_i-6)/4}$ or $\gamma_j^{r_i-(p_i+q_i)/4}$), the corresponding responses must be defined using the actual exponent containing r_i (i.e., $\zeta_{i,j} = \tau_{i,j} + c_j(r_i - (p_i + q_i)/4)$ or $\tau_{i,j} + c_j(r_i - (p_i + q_i - 6)/4)$ respectively). A definition of $\zeta_{i,j}$ that omits r_i (e.g., based on $-(p_i + q_i - 6)/4$ or $-(p_i + q_i)/4$) would not faithfully represent a Schnorr-like proof of knowledge for $\chi_{i,j}$, even if the formal verification equation in Step 10 remains unchanged.

   Moreover, the introduction of r_i in the exponent necessitates an adjustment to the sample space of each blinding factor $\tau_{i,j}$. Since the quantity that $\tau_{i,j}$ must blind becomes either $c_j(r_i - (p_i + q_i)/4)$ or $c_j(r_i - (p_i + q_i - 6)/4)$, the dominant term is now r_i rather than the prime shares p_i and q_i. As a result, the distribution of $\tau_{i,j}$ must be chosen to statistically hide r_i, not merely the smaller shares. Accordingly, in our proposed protocol $\pi_{\mathrm{BI}}^{\mathcal{M}}$ and the corresponding functionality $\mathcal{F}_{\mathrm{BI}}^{\mathcal{M}}$, we remove the upper bound M and eliminate the assumption that $p_i, q_i \leq M$. In contrast, in our revised version of Chen et al.'s protocol (cf. Section 6.11), we retain the bound M and enlarge the population of $\tau_{i,j}$ from $\mathbb{Z}_{2^{s+1}M}$ to $\mathbb{Z}_{\lceil 1.475s \rceil \cdot n 2^{s+1}(n^2 2^{2\kappa+s-3}+M)}$ to accommodate the additional randomness introduced by r_i.

3. **Improper Dependency of Security Parameters.**
   Intuitively, the output of $\mathcal{F}_{\mathrm{Zero}}$ should depend on the number of participants n, since each party receives the sum of a matrix row, which aggregates inputs from all n participants. As the number of participants increases, the potential range of these row sums naturally grows. Consequently, both the sample space of $\mathcal{F}_{\mathrm{Zero}}$ and the choice of the security parameter $\tau_{i,j}$ should scale with n, rather than being determined solely by a fixed upper bound on the shared values.

   Indeed, according to Proposition 6, ensuring that the output distribution of $\mathcal{F}_{\mathrm{Zero}}$ is statistically close to that of the random vector $U := (U_1, \ldots, U_{n-1}, -\sum_{i=1}^{n-1} U_i)$, where U_i is uniformly distribution on the set $\mathbb{Z}_{\phi(N)}$ for all $1 \leq i \leq n-1$, the population must be adjusted from $2^{2\kappa+s}$ to $n2^{2\kappa+s-3}$ (cf. Proposition 6).

4. **Sampling considerations in Boneh-Franklin Simulation.**
   In the Boneh–Franklin test, all parties agree on a common base $g \in \mathbb{Z}_N^\times$ such that $\left[\frac{g}{N}\right] = 1$. Each participant then computes $\chi_i \equiv g^{x_i} \pmod N$

for $1 \leq i \leq n$, where the exponent x_i is deterministically derived from the party's share in the protocol.
For the simulation in the security proof, Chen et al. [16] propose sampling each χ_i uniformly from $\mathbb{Z}_N^\times$. However, this approach treats χ_i as independent of g, which contradicts the actual protocol behavior where each χ_i must be a specific power of g. This dependency is essential: for example, if $g = 1$, then $\chi_i = 1$ for all $1 \leq i \leq n$, whereas a uniform sample from $\mathbb{Z}_N^\times$ would likely yield different values, possibly with $\left[\frac{\chi_i}{N}\right] = -1$, violating the protocol's correctness.
To ensure a faithful simulation, we instead sample exponents r_i' uniformly from $\mathbb{Z}_{2^{2\kappa+s-2}n}$ and set $\chi_i \equiv g^{r_i'} \pmod{N}$ for $1 \leq i \leq n$. This produces values that are statistically close to uniform over the subgroup generated by g, while preserving the necessary algebraic relation to g.

5 Implementation, Benchmarks, and Evaluation

This section presents an experimental evaluation of our proposed Lucas biprimality test in comparison with established methods, namely the Boneh-Franklin test and a relevant variant of the Miller-Rabin test.

In Section 5.3, we conduct a comparative analysis of cryptographic protocols: specifically, the protocol employing Burkhardt et al.'s variant Miller-Rabin test [15, FIGURE 6.1], a protocol based on the Boneh-Franklin test (as exemplified in [23]), and our proposed Lucas-based protocol. Subsequently, Section 5.4 details our independent implementations of both the Boneh-Franklin test and our own protocol, presenting runtime performance data from executions conducted on a laptop.

5.1 Comparing the effectives of Three Tests

We begin by recalling the variant Miller-Rabin test [15]. We then proceed to determine which of the three tests under consideration Boneh-Franklin, the variant Miller-Rabin, or our proposed Lucas test is more effective in detecting non-RSA moduli. For this analysis, consider $N = pq$ with $p \equiv q \equiv 3 \pmod 4$, and let $f \in \{p, q\}$. The algorithm for the variant Miller-Rabin test is as follows:

1. Uniformly sample an element $v \in \mathbb{Z}_N^\times$ [9] (i.e., in [15], v is chosen in $\mathbb{Z}_N$).
2. Compute $\gamma = v^{\frac{f-1}{2}} \pmod N$.
3. If $\gamma \equiv \pm 1 \pmod f$, then output **probably prime**. Otherwise output **composite**.

The biprimality test proposed in [15, 20] applies the variant Miller-Rabin test separately to $f \in \{p, q\}$. Therefore, for any $N = pq$ with $p \equiv q \equiv 3 \pmod 4$ and

[9] We narrow the selection range of v from $\mathbb{Z}_N$ to $\mathbb{Z}_N^\times$ because an element $v \in \mathbb{Z}_N \backslash \mathbb{Z}_N^\times$ will let the test output composite even when f is prime.

$\gcd(N, e_4) = 1$ the probability that N passes the process is (cf. Lemma 14)

$$\beta_{\mathrm{MR}}(p) := \frac{|\mathrm{MR}(p)|}{\phi(p)} = 2\left(\prod_{p_i|p} \frac{\gcd(d_i, \frac{p-1}{2})}{p_i^{r_i-1}(p_i-1)}\right).$$

In particular, when $p = q$ is prime, such an RSA modulus candidate p, q will always pass this algorithm's test with 100% certainty. Therefore, we recommend incorporating a check to verify whether N is a perfect square to exclude this case.

We simplify the formula to compare the tests pairwise and analyze the ratios across three scenarios. Let $\mathbf{1}_{\mathbb{P}}(\cdot)$ be the indicator function of positive integers (i.e., $\mathbf{1}_{\mathbb{P}}(0) = 0$).

- **Variant Miller-Rabin vs. Boneh-Franklin Test:**

$$\frac{\beta_{\mathrm{BF}}(N, e_4)}{\beta_{\mathrm{MR}}(p)\beta_{\mathrm{MR}}(q)} = \left(\frac{1}{\mathbf{1}_{\mathbb{P}}(\sqrt{N})+1}\right)\left(\prod_{\substack{p_i|p \\ p_i\nmid q}} \frac{\gcd(e_4, d_i)}{\gcd(d_i, \frac{p-1}{2})}\right)$$
$$\cdot\left(\prod_{\substack{p_i|q \\ p_i\nmid p}} \frac{\gcd(e_4, d_i)}{\gcd(d_i, \frac{q-1}{2})}\right)\left(\prod_{p_i|\gcd(p,q)} \frac{(p_i-1)\gcd(e_4, d_i)}{p_i \gcd(d_i, \frac{p-1}{2}) \gcd\left(d_i, \frac{q-1}{2}\right)}\right).$$

- **Lucas Test vs. Boneh-Franklin Test**

$$\frac{\beta_{\mathrm{Lucas}}(N, e_4)}{\beta_{\mathrm{BF}}(N, e_4)} = \left(\frac{\prod_{i=1}^{s}(\gcd(e_4, d_i)-1) + \prod_{i=1}^{s}\gcd(e_4, d_i)}{2\prod_{i=1}^{s}\gcd(e_4, d_i)}\right)$$
$$\cdot\left(\frac{\prod_{i=1}^{s}(p_i-1)}{\prod_{i\in S_0}(p_i-2)\left(\prod_{i\in S_1}(p_i-2) + \mathbf{1}_{\mathbb{P}}(|S_1|)(-1)^{|S_1|}\right)}\right). \quad (7)$$

- **Lucas Test vs. Variant Miller-Rabin Test**

$$\frac{\beta_{\mathrm{Lucas}}(N, e_4)}{\beta_{\mathrm{MR}}(p)\beta_{\mathrm{MR}}(q)} = \left(\frac{\prod_{i=1}^{s}(\gcd(e_4, d_i)-1) + \prod_{i=1}^{s}\gcd(e_4, d_i)}{2\prod_{p_i|p}\gcd(\frac{p-1}{2}, d_i)\prod_{p_i|q}\gcd(\frac{q-1}{2}, d_i)}\right)$$
$$\cdot\left(\frac{\left(\prod_{i=1}^{s}(p_i-1)\right)\left(\prod_{p_i|\gcd(p,q)}(1-p_i^{-1})\right)}{\prod_{i\in S_0}(p_i-2)\left(\prod_{i\in S_1}(p_i-2) + \mathbf{1}_{\mathbb{P}}(|S_1|)(-1)^{|S_1|}\right)}\right).$$

Table 2 demonstrates that, among the three tests, the Lucas test slightly outperforms the variant Miller-Rabin test and significantly outperforms the Boneh-Franklin test in identifying non-RSA moduli.

5.2 Theoretical comparison between the Boneh-Franklin test and the Lucas test.

Starting with the formula (7), we attempt to explain the rationale behind Table 2. At first, when the product $\prod_{i=1}^{s}\left(\gcd(e_4, p_i-1)-1\right)$ vanishes, if $p_{\min}$ can be

Table 2: Pairwise comparison charts among the three tests.

Method	$\beta = \frac{\beta_{\mathrm{BF}}(N,e_4)}{\beta_{\mathrm{MR}}(p)\beta_{\mathrm{MR}}(q)}$	$\beta = \frac{\beta_{\mathrm{BF}}(N,e_4)}{\beta_{\mathrm{Lucas}}(N,e_4)}$	$\beta = \frac{\beta_{\mathrm{Lucas}}(N,e_4)}{\beta_{\mathrm{MR}}(p)\beta_{\mathrm{MR}}(q)}$
$\beta < 1$	0.08%	$< 0.01\%$	54.26%
$\beta = 1$	54.18%	0%	0%
$\beta > 1$	45.74%	$> 99.99\%$	45.74%

Count how many non-RSA moduli $N = pq$ with $p \equiv q \equiv 3 \pmod 4$, $\gcd(N, e_4) = 1$, and $\gcd(pq, p') = 1$ for all primes $p' \leq 541$ satisfy $\beta > 1$, $\beta = 1$ or $\beta < 1$, which run over all $3 \leq p < q \leq 1440003$.

controlled (or bounded) such that N lies in

$$\Gamma(p_{\min}) := \left\{ N = \prod_{i=1}^{s} p_i^{r_i} \mid p_i \geq p_{\min} \text{ for all } i, s < \frac{\ln(1/2 + 1/(p_{\min}-1))}{\ln(1 - 1/(p_{\min}-1))} \right\},$$

then $\beta_{\mathrm{BF}}(N, e_4)$ is necessarily greater than $\beta_{\mathrm{Lucas}}(N, e_4)$. In addition, the function $f(x) = \frac{\ln(1/2+1/x)}{\ln(1-1/x)} > 0$ is an increasing function for all $x > 5$ implying that the number of prime factors of N in $\Gamma(p_{\min})$ increases with $p_{\min}$. This also explains why in most of the cases in Table 2, there is always a high probability that $\beta_{\mathrm{BF}}(N, e_4) > \beta_{\mathrm{Lucas}}(N, e_4)$.

Lemma 5. *Let p, q be two positive integers with $p \equiv q \equiv 3 \pmod 4$, and $N = \prod_{i=1}^{s} p_i^{r_i}$ be the product of p, q such that there exists $\gcd(d_i, e_4) = 1$ for some $1 \leq i \leq s$. Here $p_i - 1 = 2^{k_i} d_i$ with $2 \nmid d_i$ for all $1 \leq i \leq s$. For any odd prime number $p_{\min}$, and all non-perfect square and non-RSA moduli $N \in \Gamma(p_{\min})$, we have*

$$\beta_{\mathrm{Lucas}}(N, e_4) < \beta_{\mathrm{BF}}(N, e_4).$$

Proof. If there exists $1 \leq i \leq s$ such that $\gcd(e_4, d_i) = 1$, then one has

$$\frac{\beta_{\mathrm{Lucas}}(N, e_4)}{\beta_{\mathrm{BF}}(N, e_4)} = \frac{\prod_{i=1}^{s}(p_i - 1)}{2\big(\prod_{i=1}^{s}(p_i - 2) + \mathbf{1}_{\mathbb{P}}(|S_1|)(-1)^{|S_1|}\prod_{i\in S_0}(p_i - 2)\big)}.$$

The condition $\beta_{\mathrm{Lucas}}(N, e_4) < \beta_{\mathrm{BF}}(N, e_4)$ is equivalent to

$$\prod_{i\in S}\left(1 - \frac{1}{p_i - 1}\right) + \mathbf{1}_{\mathbb{P}}(|S_1|)(-1)^{|S_1|}\prod_{i\in S_0}\left(1 - \frac{1}{p_i - 1}\right)\prod_{i\in S_1}\frac{1}{p_i - 1} > \frac{1}{2},$$

which implies that, for all $N \in \Gamma(p_{\min})$, $\big(\text{i.e., } s < \frac{\ln(1/2+1/(p_{\min}-1))}{\ln(1-1/(p_{\min}-1))}\big)$,

$$\begin{aligned}\prod_{i\in S}\left(1 - \frac{1}{p_i - 1}\right) &> \left(1 - \frac{1}{p_{\min} - 1}\right)^s \\ &> \frac{1}{2} + \frac{1}{p_{\min} - 1} > \frac{1}{2} - \mathbf{1}_{\mathbb{P}}(|S_1|)(-1)^{|S_1|}\prod_{i\in S_0}\left(1 - \frac{1}{p_i - 1}\right)\prod_{i\in S_1}\frac{1}{p_i - 1}.\end{aligned}$$

□

When $p_{\min} = 359$ (or $p_{\min} = 62017$ in our experimental scenario, respectively), it can be ensured that for N satisfying the conditions of Lemma 5 and having a bit length of up to 2086 bits (or 684319 bits, respectively), $\beta_{\text{Lucas}}(N, e_4) < \beta_{\text{BF}}(N, e_4)$ always holds.

Regarding the reverse direction of the inequality, we can identify the conditions for its validity by considering the following extreme scenarios.

Lemma 6. *Let p, q be two positive integers with $p \equiv q \equiv 3 \pmod 4$, and $N = \prod_{i=1}^{s} p_i^{r_i}$ be the product of p, q such that $p_i > 3$, $\gcd(e_4, p_i - 1) = d_i$, and $p_i \equiv 3 \pmod 4$ for all $1 \le i \le s$. For any non-perfect square N except for $s = |S_1| = 2$, we have $\beta_{\text{Lucas}}(e_4) > \beta_{\text{BF}}(N, e_4)$. In particular, $s = |S_1| = 2$, then $\beta_{\text{Lucas}}(e_4) = \beta_{\text{BF}}(N, e_4)$.*

Proof. Let and $p_i - 1 = 2^{k_i} d_i$ with $2 \nmid d_i$ for all $1 \le i \le s$. Since $\gcd(e_4, d_i) = d_i = 2^{-k_i}(p_i - 1)$ for all $1 \le i \le s$, we have

$$\frac{\beta_{\text{Lucas}}(N, e_4)}{\beta_{\text{BF}}(N, e_4)} = \frac{\prod_{i=1}^{s}(p_i - 1) + \prod_{i=1}^{s}(p_i - 1 - 2^{k_i})}{2\big(\prod_{i=1}^{s}(p_i - 2) + \mathbf{1}_{\mathbb{P}}(|S_1|)(-1)^{|S_1|}\prod_{i \in S_0}(p_i - 2)\big)}.$$

Note that $p_i \equiv 3 \pmod 4$, which implies that $k_i = 1$ for all $1 \le i \le s$. Therefore, the condition $\beta_{\text{Lucas}}(N, e_4) > \beta_{\text{BF}}(N, e_4)$ is equivalent to say that, letting $x_i := p_i - 2 > 1$ (i.e. if there exists one $p_i = 3$, then $\prod_{i=1}^{s}(x_i - 1) = 0$ vanishes),

$$\begin{aligned} 2 \cdot \mathbf{1}_{\mathbb{P}}(|S_1|)(-1)^{|S_1|} \prod_{i \in S_0} x_i &< \prod_{i=1}^{s}(x_i + 1) + \prod_{i=1}^{s}(x_i - 1) - 2\prod_{i=1}^{s}(x_i) \\ &= \begin{cases} 0, \text{ if } s = 1; \\ 2\left(e_{s-2} + e_{s-4} + \ldots + e_{s-2\lfloor s/2 \rfloor}\right), \text{otherwise,} \end{cases} \end{aligned}$$

where e_j is the sum of products of j variables x_i. Now, we show that for all $s \ge 1$, the above inequality holds. When $s = 1$, then $|S_1| = 1$, since N is not perfect square. Thus $-2 < 0$ means that the inequality holds. As for the case $s = 2$, then $|S_1| = 1$ or 2 (i.e., $|S_1| = s = 2$ gives us the equality), since N is not perfect square. If $|S_1| = 1$, then the inequality holds since the left side is negative. If $|S_0| = s \ge 3$, then N is a perfect square. Thus, we only need to consider the case $|S_0| = s - 1$. This is because in the other scenarios, the inequality clearly holds, since $\prod_{i \in S_0} x_i < e_{s-2}$. However, in the case where $S_0 = s - 1$, we have $N = pq \equiv 3 \pmod 4$, since $p_i \equiv 3 \pmod 4$ for all i. This contradicts $N = pq \equiv 1 \pmod 4$. □

For other scenarios not explicitly discussed above (e.g., $1 < \gcd(e, p_i - 1) < d_i$ for all $1 \le i \le s$), the inequality may hold in either direction. However, our analysis of the governing formulas suggests that the probability of $\beta_{\text{Lucas}}(N, e_4) < \beta_{\text{BF}}(N, e_4)$ is generally higher. Therefore, the empirical results presented in Table 2 align with this theoretical expectation.

In Lemma 6, we only consider the case where N has no prime factor 3. This restriction is justified because practical RSA moduli generation protocols typically eliminate the prime 3 via initial trial division.

It is noteworthy that since Table 2 enforces a minimum $p_{\min}$ of 541, any value of N for which a boundary condition such as $\beta = 1$ might occur must be greater than $p_{\min}^3 \sim 1.58 \times 10^8$. This large scale for N provides an explanation for the absence of observed cases where $\beta = 1$.

5.3 Computational Cost Comparison of Three Tests

Burkhardt et al. [15] demonstrated that their variant Miller-Rabin protocol exhibits superior efficiency over the Boneh-Franklin protocol, which is presented by Frederiksen et al. [23] when compared at the same security level. Their analysis highlighted that the Boneh-Franklin test typically requires more iterations to achieve equivalent soundness, a consequence of its original 1/2 worst-case soundness error. To systematically evaluate the effectiveness of these established protocols alongside our proposed Lucas test, we adopt the comparative framework and terminology from Burkhardt et al.'s work (i.e., detailed further in our Section 6.10), with key performance metrics summarized in Table 3.

Our evaluation indicates that, in terms of computational cost per iteration, the Boneh-Franklin test is the most efficient of the three. However, our Lucas test also demonstrates strong efficiency, closely comparable to Boneh-Franklin's, with the primary performance difference stemming from local computations. Considering contemporary computational capabilities, this operational gap between the Boneh-Franklin and Lucas tests is nearly negligible in many practical scenarios.

5.4 Implementation in the Semi-honest Setting

Our experiment is composed of three components:

1. **Generate an RSA modulus candidate**: Utilizing the CRT-Sampling protocol [16, Protocol 4.4] generates N, $\mathfrak{p}_i$, $\mathfrak{q}_i$, and $\{\mathfrak{p}_i \pmod 4, \mathfrak{q}_i \pmod 4\}_{i=1}^n$ satisfying $p = \sum_i \mathfrak{p}_i \equiv 3 \pmod 4$, $q = \sum_i \mathfrak{q}_i \equiv 3 \pmod 4$, and $\gcd(N, p') = 1$ for all primes $p' \leq B$. Other RSA moduli generation protocols can also be utilized (i.e., [10, 16, 17, 20, 23, 40]). Meanwhile, set a parameter $p_{\min}$ to check that no prime smaller than $p_{\min}$ dividing $N = pq$. In our case, $p_{\min} = 62017$. For $N = 2048$ (resp. 3072) bits, passing this check implies approximately a 0.0767% (resp. 0.0341%) probability that both p and q are prime. This is based on DeBruijn's formula [13]: for a k bit integer p,

$$\Pr(p \in \mathbf{P} | \text{ trial division up to } B) \sim 2.57 \cdot \ln B \cdot k^{-1}.$$

   Like most experiments, our MPC multiplication with secret-sharing is proposed by Gennaro et al. [25, Figure 2], assuming an honest majority.
2. **A biprimality test**: We continue checking the exponential conditions required by both biprimality tests until the soundness error is reduced to 2^{-80}. To be more precise, either Protocol 5 or Protocol 6 may be iterated to a predetermined number of iterations. In the event that candidate N is identified as a non-RSA modulus, the procedure reverts to step 1.

Table 3: Single Execution Comparison for Three Tests

Method	Boneh-Franklin	Variant Miller-Rabin	Proposed test
Basis selection (local)	$g \leftarrow \mathbb{Z}_N, \left[\frac{g}{N}\right] = 1$	$v \leftarrow \mathbb{Z}_N$	$P \leftarrow \mathbb{Z}_N, \left[\frac{P^2-1}{N}\right] = 1$
Exponential test (local)	$\mathcal{P}_1 : g^{(N-(\mathfrak{p}_i+\mathfrak{q}_i)+1)/4}$ $\mathcal{P}_i : g^{(-\mathfrak{p}_i-\mathfrak{q}_i)/4}$	$\mathcal{P}_1 : v^{(\mathfrak{f}_i-1)/2}$ $\mathcal{P}_i : v^{\mathfrak{f}_i/2}$	$\mathcal{P}_1 : (\alpha\beta^{-1})^{(N-(\mathfrak{p}_i+\mathfrak{q}_i)+1)/4}$ $\mathcal{P}_i : (\alpha\beta^{-1})^{(-\mathfrak{p}_i-\mathfrak{q}_i)/4}$
Exponential test (MPC)	$g^{e_4} \leftarrow$Shuffle	Mul-to-Add $\left[v^{(f-1)/2}\right] \leftarrow$ Divisible $[y_{+1}], [y_{-1}] \leftarrow [y_{+1} \cdot y_{-1}] \leftarrow$Mult	$(\alpha\beta^{-1})^{e_4} \leftarrow$ Shuffle
GCD test (MPC)	$[r] \leftarrow$RandomSample $[r \cdot (p+q-1)] \leftarrow$Mult	*None*[1]	$[r] \leftarrow$RandomSample $[r \cdot (p+q-1)] \leftarrow$Mult

The calculations of the Variant Miller-Rabin test need to be executed for both $f = p$ and $f = q$. The process above the bold line requires 40 iterations to achieve a soundness error not greater than 2^{-80}. **Basis selection** refers to the conditions of the basis for exponential calculations. We consider the semi-honest model; hence, the basis is determined by $\mathcal{P}_1$. In **Exponential test(local)**, $\mathcal{P}_i$ represents $\mathcal{P}_2, \ldots \mathcal{P}_n$, and α, β are the two roots of the polynomial $x^2 + Px + (P^2 - 1)/4$. In **Exponential test(MPC)**, the Shuffle protocol outputs the product of shares. Mul-to-Add refers to the conversion of multiplicative shares to additive shares. The output of Divisible $y_{\pm 1}$ indicates whether $v^{(f\pm 1)/2} \equiv 0 \pmod{f}$. Mult denotes the MPC multiplication between additive shares. RandomSample outputs a random element from a specified set.

[1] The Variant Miller-Rabin test needs to confirm that $p \neq q$.

3. **Verify** $\gcd\left(pq, (p + \left[\frac{-1}{p}\right])(q + \left[\frac{-1}{p}\right])\right) = 1$: Sample an $r \in \mathbb{Z}_N^\times$, calculate $z = r\left(p\left[\frac{-1}{q}\right] + q\left[\frac{-1}{p}\right] + \left[\frac{-1}{N}\right]\right)$, and check $\gcd(pq, z) = 1$. If the check fails, return to step 1.

The scheme was implemented using the Golang programming language and its native "math/big" library. To ensure that the probability of accepting a non-RSA modulus is at most 2^{-80}, we configured both biprimality tests to perform 40 iterations. Experiments were conducted using moduli N of 2048 and 3072 bits, involving $2, 3$, and 4 parties. All programs ran single-threaded on a 13-inch MacBook Pro (2022) equipped with an Apple M2 processor and 16GB of LPDDR5 RAM. The resulting execution times are presented in Table 4.

Table 4: The mean ± standard deviation of execution time (in seconds) for our methods and the competing method.

	$N = 2048$		$N = 3072$	
	Proposed test	Boheh-Franklin	Proposed test	Boheh-Franklin
$n = 2$	18.84 ± 18.50	20.47 ± 19.64	117.59 ± 119.24	109.66 ± 119.97
$n = 3$	33.01 ± 35.36	43.46 ± 42.68	174.59 ± 200.88	169.81 ± 161.44
$n = 4$	59.67 ± 60.12	64.16 ± 61.07	232.81 ± 249.38	274.67 ± 273.64

Our experiments demonstrate that both the Lucas test and the Boneh-Franklin test achieve efficient average execution times. Notably, when N was not an RSA modulus in our test instances, both tests consistently identified this within a single iteration. This suggests that overall performance variations in RSA modulus generation are likely influenced more by the success rate of the initial p and q candidate selection than by inherent differences in the single-run detection speed of these two biprimality tests for clear non-RSA cases.

Regarding computational complexity within MPC protocols (i.e., as detailed in Section 5.3), our Lucas test and the Boneh-Franklin test are comparable, with both generally outperforming the variant Miller-Rabin test proposed by Burkhardt et al. [15]. A key advantage is that the most efficient Prime Candidate Sampling methods [16, 40] are not directly compatible with Burkhardt et al.'s approach, as these methods do not guarantee equal bit-lengths for p and q, which is a requirement for that Miller-Rabin variant. For instance, Chen et al. [16] restrict shares $\mathfrak{p}_i$, $\mathfrak{q}_i$ to $[0, 2^{\ell-\log_2 n}]$, while Guilhem et al. [40] use $[2^{\ell-1}, 2^{\ell-1+80}]$, where ℓ is related to the bit-lengths of p and q. Consequently, to accommodate its specific requirements, integrating the variant Miller-Rabin test may incur additional overhead in the prime generation phase compared to the more flexible Boneh-Franklin and Lucas tests.

Furthermore, exhaustive experiments (cf. Table 2) indicate that, on average, the Lucas test achieves the best soundness error. In practical terms, if a variant Miller-Rabin test requires two iterations to reach a specific target error rate, our proposed Lucas test typically achieves a comparable or superior error rate within the same number of iterations. In conclusion, given that the difference in local computational overhead between the Boneh-Franklin and Lucas tests is negligible, our proposed Lucas test stands as a highly competitive alternative.

References

1. Abadi, A., Ristea, D., Murdoch, S.J.: Delegated time-lock puzzle. arXiv preprint arXiv:2308.01280 (2023) 1
2. Algesheimer, J., Camenisch, J., Shoup, V.: Efficient computation modulo a shared secret with application to the generation of shared safe - prime products. In: Yung, M. (ed.) Advances in Cryptology — CRYPTO 2002. pp. 417 – 432. Springer Berlin Heidelberg, Berlin, Heidelberg (2002) 1.3
3. Apostol, T.M.: Introduction to analytic number theory (1976), `https://api.semanticscholar.org/CorpusID:118086024` 6.2
4. Apostol, T.M.: Introduction to analytic number theory. Springer Science & Business Media (2013) 1, 2, 11, 12
5. Arnault, F.: The rabin-monier theorem for lucas pseudoprimes. Math. Comput. **66**, 869–881 (04 1997). https://doi.org/10.1090/S0025-5718-97-00836-3 1, 2.2, 4.1, 6.5
6. Benaloh, J., de Mare, M., Accumulators, O.W.: A decentralized alternative to digital signatures. In: Advances in Cryptology-Proceedings of Eurocrypt. vol. 93 (1994) 1
7. Benaloh, J.: Secret sharing homomorphisms: keeping shares of a secret secret. In: Proceedings on Advances in cryptology—CRYPTO '86. vol. LNCS 263, pp. 251–260 (01 1987) 6.9

8. Boneh, D., Bonneau, J., Bünz, B., Fisch, B.: Verifiable delay functions. In: Annual international cryptology conference. pp. 757–788. Springer (2018) 1
9. Boneh, D., Bünz, B., Fisch, B.: Batching techniques for accumulators with applications to iops and stateless blockchains. In: Advances in Cryptology–CRYPTO 2019: 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18–22, 2019, Proceedings, Part I 39. pp. 561–586. Springer (2019) 1
10. Boneh, D., Franklin, M.: Efficient generation of shared rsa keys. Journal of the ACM **48** (12 2001). https://doi.org/10.1145/502090.502094 1, 1.2, 1.3, 5.4, 6.10, 6.10
11. Boneh, D., Shoup, V.: A Graduate Course in Applied Cryptography (2023), `https://crypto.stanford.edu/~dabo/cryptobook/BonehShoup_0_6.pdf` 5
12. Boudabra, M., Nitaj, A.: A new rsa variant based on elliptic curves. Cryptography (2023). https://doi.org/10.3390/cryptography7030037 1.1
13. Bruijn, de, N.: On the number of uncancelled elements in the sieve of eratosthenes. Proceedings of the Koninklijke Nederlandse Akademie van Wetenschappen: Series A: Mathematical Sciences **53**(5-6), 803–812 (1950) 5.4
14. Buhler, J., Stevenhagen, P.: Algorithmic number theory. Lattices, number fields, curves and cryptography. Reprint of the 2008 hardback ed. Cambridge University Press (01 2011) 1
15. Burkhardt, J., Damgård, I., Frederiksen, T., Ghosh, S., Orlandi, C.: Improved distributed rsa key generation using the miller-rabin test. CCS '23: Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security pp. 2501–2515 (2023). https://doi.org/10.1145/3576915.3623163 1, 1.3, 5, 5.1, 5.3, 5.4, 6.10, 6.10, 6.10, 5
16. Chen, M., Doerner, J., Kondi, Y., Lee, E., Rosefield, S., Shelat, A., Cohen, R.: Multiparty generation of an rsa modulus. Journal of Cryptology **35** (04 2022). https://doi.org/10.1007/s00145-021-09395-y 1, 1.1, 1.2, 1, 4.3, 4.3, 4.3, 4.3, 4.3, 4.3, 4.4, 5.4, 5.4, 6.11
17. Chen, M., Hazay, C., Ishai, Y., Kashnikov, Y., Micciancio, D., Riviere, T., Shelat, A., Venkitasubramaniam, M., Wang, R.: Diogenes: Lightweight scalable rsa modulus generation with a dishonest majority. In: 2021 IEEE Symposium on Security and Privacy (SP). pp. 590–607. IEEE (2021) 1, 5.4
18. Chvojka, P.: Private coin verifiable delay function. Cryptology ePrint Archive (2023) 1
19. Damgård, I., Landrock, P., Pomerance, C.: Average case error estimates for the strong probable prime test. Mathematics of Computation - Math. Comput. **61**, 177–177 (09 1993). https://doi.org/10.2307/2152945 1, 1.3
20. Damgård, I., Mikkelsen, G.: Efficient, robust and constant-round distributed rsa key generation. In: Theory of Cryptography. pp. 183–200. Springer Berlin Heidelberg (02 2010). https://doi.org/10.1007/978-3-642-11799-2_12 1, 1.3, 5.1, 5.4
21. Einsele, S., Paterson, K.: Average case error estimates of the strong lucas test. Designs, Codes and Cryptography pp. 1–38 (01 2024). https://doi.org/10.1007/s10623-023-01347-w 1.3
22. Ephraim, N., Freitag, C., Komargodski, I., Pass, R.: Continuous verifiable delay functions. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 125–154. Springer (2020) 1
23. Frederiksen, T., Lindell, Y., Osheter, V., Pinkas, B.: Fast Distributed RSA Key Generation for Semi-honest and Malicious Adversaries: 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19–23, 2018, Proceedings, Part II, pp. 331–361. Springer International Publishing (07 2018). https://doi.org/10.1007/978-3-319-96881-0_12 1, 5, 5.3, 5.4, 5

24. Friedman, O., Marmor, A., Mutzari, D., Scaly, Y.C., Spiizer, Y., Yanai, A.: Tiresias: Large scale, maliciously secure threshold paillier. Cryptology ePrint Archive (2023) 1
25. Gennaro, R., Rabin, M.: Simplified vss and fast-track multiparty computations with applications to threshold cryptography. Proc. of 17th PODC (06 1998). https://doi.org/10.1145/277697.277716 5.4
26. Grassi, L., Rechberger, C., Rotaru, D., Scholl, P., Smart, N.: Mpc-friendly symmetric key primitives. In: CCS '16: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. pp. 430–443 (10 2016). https://doi.org/10.1145/2976749.2978332 1.2, 4.2
27. Hazay, C., Mikkelsen, G.L., Rabin, T., Toft, T., Nicolosi, A.A.: Efficient rsa key generation and threshold paillier in the two-party setting. Journal of Cryptology **32**, 265–323 (2019) 1
28. Hoffmann, C., Hubáček, P., Kamath, C., Krňák, T.: (verifiable) delay functions from lucas sequences. Cryptology ePrint Archive (2023) 1
29. Ireland, K., Rosen, M.I.: A classical introduction to modern number theory, vol. 84. Springer Science & Business Media (01 1990) 3, 7
30. Katz, J.: On achieving the" best of both worlds" in secure multiparty computation. In: Proceedings of the thirty-ninth annual ACM symposium on Theory of computing. pp. 11–20 (2007) 2.3, 6.4
31. Khedr, W.I., Khater, H.M., Mohamed, E.R.: Cryptographic accumulator-based scheme for critical data integrity verification in cloud storage. IEEE Access **7**, 65635–65651 (2019) 1
32. Malavolta, G., Thyagarajan, S.A.K.: Homomorphic time-lock puzzles and applications. In: Annual International Cryptology Conference. pp. 620–649. Springer (2019) 1
33. Montgomery, H.L., Vaughan, R.C.: Multiplicative Number Theory I: Classical Theory. Cambridge Studies in Advanced Mathematics, Cambridge University Press (2006). https://doi.org/10.1017/CBO9780511618314 8
34. Niven, I., Zuckerman, H.S., Montgomery, H.L.: An introduction to the theory of numbers. Wiley, New York, fifth edition. edn. (1991) 2, 6.5
35. Oded, G.: Foundations of Cryptography: Volume 2, Basic Applications. Cambridge University Press, USA, 1st edn. (2009) 2.3, 6.4, 6.4
36. Pietrzak, K.: Simple verifiable delay functions. In: 10th innovations in theoretical computer science conference (itcs 2019). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2018) 1
37. Rabin, M.: Probabilistic algorithm for testing primality. Journal of Number Theory **12**, 128–138 (02 1980). https://doi.org/10.1016/0022-314X(80)90084-0 1
38. Rivest, R., Shamir, A., Adleman, L.: A method for obtaining digital signatures and public-key cryptosystems. Commun. ACM **26**, 96–99 (01 1983). https://doi.org/10.1145/359340.359342 1
39. Rivest, R.L., Shamir, A., Wagner, D.A.: Time-lock puzzles and timed-release crypto. 1996 Technical Report (1996) 1
40. Delpech de Saint Guilhem, C., Makri, E., Rotaru, D., Tanguy, T.: The return of eratosthenes: Secure generation of rsa moduli using distributed sieving. In: Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. pp. 594–609 (2021) 1, 5.4, 5.4
41. Wesolowski, B.: Efficient verifiable delay functions. In: Advances in Cryptology–EUROCRYPT 2019: 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19–23, 2019, Proceedings, Part III 38. pp. 379–407. Springer (2019) 1

Appendix 6 Mathematical Tools and Supplementary Proofs and Protocols

This section begins by introducing the Chinese Remainder Theorem (CRT) as utilized throughout this paper. We then apply the CRT to demonstrate how the core sets in both the Boneh-Franklin test (see Lemma 3) and our proposed Lucas test (see Proposition 1) can be decomposed into components modulo prime factors p_i. This decomposition is a standard application of the CRT. For the reader's convenience, we subsequently introduce several properties of the Jacobi symbol and related lemmas that are used in this article.

Further details are provided as follows: Section 6.3 details the number of elements in the set related to $\mathrm{MR}(p)$ for the Miller-Rabin test when $p \equiv 3 \pmod 4$. The security model adopted in our work is described in Section 6.4. Section 6.5 includes supplementary proofs pertaining to the Lucas test. Subsequently, Section 6.6 examines the distribution consistency required for Theorem 3. A security proof for the π_{Leg} protocol is then provided in Section 6.7. Furthermore, Section 6.9 covers auxiliary protocols utilized within our main protocol against semi-honest adversaries. Finally, necessary lemmas concerning statistical distance are presented in Section 6.8.

For ease of comparison, Section 6.10 summarizes the three RSA moduli generation protocols discussed. Lastly, a revised version of the Chen et al. protocol is offered for convenient reference in Section 6.11.

6.1 The CRT and its applications

We start by illustrating the Chinese Remainder Theorem (CRT) and demonstrating some of its applications.

Lemma 7. [29, Theorem 3, Chapter 4] *Let* $N = \prod_{i=1}^{s} p_i^{r_i}$ *be the prime decomposition of* N*. Then we have the isomorphism* f_{CRT} *defined by*

$$\begin{aligned} f_{CRT} : \quad \mathbb{Z}_N^{\times} \quad &\longrightarrow \mathbb{Z}_{p_1^{r_1}}^{\times} \times \cdots \times \mathbb{Z}_{p_s^{r_s}}^{\times} \\ x \pmod N &\mapsto \left(x \pmod{p_1^{r_1}}, \ldots, x \pmod{p_s^{r_s}}\right). \end{aligned}$$

Here $\mathbb{Z}_{p_i^{r_i}}^{\times}$ *is a cyclic group of order* $p_i^{r_i-1}(p_i-1)$*.* $\mathbb{Z}_{2^a}^{\times}$ *is cyclic of order* 1 *and* 2 *for* $a = 1$ *and* 2*, respectively. If* $a \geq 3$*, then it is the product of two cyclic groups, one of order* 2*, the other of order* 2^{a-2}*.*

Lemma 8. *Let* $p \equiv q \equiv 3 \pmod 4$*, and* $\gcd\left(pq, e_4\right) = 1$*, where* $e_4 = (p-1)(q-1)/4$*. Assume that* $N := pq = \prod_{i=1}^{s} p_i^{r_i}$*. Then*

$$|\mathrm{BF}(N, e_4)| = \prod_{i=1}^{s} |\mathrm{BF}(p_i^{r_i}, e_4)|.$$

Proof. Through the map f_{CRT}, we only need to prove that $x^{e_4} \equiv 1 \pmod{N}$ if and only $x^{e_4} \equiv 1 \pmod{p_i^{r_i}}$ for all $1 \leq i \leq s$. This is directly proven due to the fact $N \mid x^{e_4} - 1$ if and only if $p_i^{r_i} \mid x^{e_4} - 1$ for all $1 \leq i \leq s$. □

Lemma 9. *Let D be an integer and $N := \prod_{i=1}^{s} p_i^{r_i}$ be a positive integer with $\gcd(N, 2D) = 1$. Then, for all $\epsilon \in \{-1, 1\}$, one has*

$$|\mathcal{Z}^{\epsilon}(D, N)| = \sum_{\epsilon_1 \cdots \epsilon_s = \epsilon} \prod_{i=1}^{s} |\mathcal{Z}^{\epsilon_i}(D, p_i^{r_i})|.$$

Proof. It is clearly, $P^2 - 4Q \equiv D \pmod{N}$ is solvable, then $P^2 - 4Q \equiv D \pmod{p_i^{r_i}}$ for all $1 \leq i \leq s$. Conversely, CRT says that if $P^2 - 4Q \equiv D \pmod{p_i^{r_i}}$ for all $1 \leq i \leq s$, then $P^2 - 4Q \equiv D \pmod{N}$. Moreover, $\gcd(Q, N) = 1$ if and only if $\gcd(Q, p_i^{r_i}) = 1$ for all $1 \leq i \leq s$. Consider the condition of the Jacobi symbol. As for the case N is non-square, given a fixed set $\epsilon_1, \ldots, \epsilon_s$ such that $\epsilon_1 \cdots \epsilon_s = \epsilon$, where $\epsilon_i \in \{-1, 1\}$ for all $1 \leq i \leq s$, the map f_{CRT} guarantees that for any Q_i with $\left[\frac{Q_i}{p_i^{r_i}}\right] = \epsilon_i$ for all $1 \leq i \leq s$, there exists a $Q \equiv Q_i \pmod{p_i^{r_i}}$ such that $\left[\frac{Q}{N}\right] = \epsilon$. The case square N is straightforward, since $\left[\frac{Q}{N}\right] = 1$ for all $\gcd(Q, N) = 1$. The proof is complete by the above discussion. □

Lemma 10. *Let p_1, p_2, p_3 be distinct prime numbers with $p_3 = (p_2 - 1)p' + 1$, and $p' \mid p_1 - 1$. Here p' is a prime number. Assume that $p = p_1$, $q = p_2 p_3$, $e = (p-1)(q-1)/4$, and $N = pq$. Then for all $g \in \mathbb{Z}_N^{\times}$, we have*

$$g^e \equiv 1 \pmod{N}.$$

Proof. Since $p' \mid p - 1$, then for any $g \in \mathbb{Z}_N^{\times}$, we have

$$g^e \equiv g^{(p-1)(p_2 p_3 - 1)/4} \equiv 1 \pmod{p},$$

$$g^e \equiv g^{(p-1)(p_2-1)((p_2-1)p' + 1 + p')/4} \equiv 1 \pmod{p_2},$$

and

$$g^e \equiv g^{p'(p_2-1)((p-1)/p')((p_2-1)p' + 1 + p')/4} \equiv 1 \pmod{p_3}.$$

The above equalities, and Lemma 7 implies the desired result. □

To satisfy the $\gcd(N, e) = 1$ constraints, we first fix $a = p_2 - 1$ and choose a prime p' such that both p_2 and $p_3 = a \cdot p' + 1$ are prime. As long as there exists a pair (a, p') satisfying the above conditions, then by Dirichlet's theorem, there are infinitely many primes of the form $p_1 = 4kp' + 1$ such that: $p' \mid p_1 - 1$, $\gcd(a, p_1) = 1$, and $\gcd(p_1, ap' + 1 + p') = 1$, because a, p' have been fixed, which implies that the desired result $\gcd(N, e) = 1$. As a concrete example, we may simply choose $p_2 = 5$ and $p' = 3$.

6.2 The Jacobi Symbol and Related Consequences

This section lists the properties and some results of the Jacobi symbol.

Definition 1 (Legendre symbol). [4, Section 9.2] *Let p be an odd prime. We define Legendre symbol $\left[\frac{a}{p}\right]$ as follows:*

$$\left[\frac{a}{p}\right] = \begin{cases} 0 & \text{if } a \equiv 0 \pmod{p}; \\ +1 & \text{if } a \text{ is a quadratic residue modulo } p; \\ -1 & \text{if } a \text{ is a quadratic nonresidue modulo } p. \end{cases}$$

Definition 2 (Jacobi symbol). [4, Section 9.7] *If N is a positive odd integer with prime factorization $N = \prod_{i=1}^{s} p_i^{r_i}$, then the Jacobi symbol $\left[\frac{a}{N}\right]$ is defined for all integers a by the equation*

$$\left[\frac{a}{N}\right] = \prod_{i=1}^{s} \left[\frac{a}{p_i}\right]^{r_i},$$

where $\left[\frac{a}{p_i}\right]$ is the Legendre symbol.

Lemma 11. [4, Theorem 9.9, Theorem 9.10] *If N_1 and N_2 are odd positive integers, we have*

(a) $\left[\frac{ab}{N_1}\right] = \left[\frac{a}{N_1}\right]\left[\frac{b}{N_1}\right]$;
(b) $\left[\frac{a}{N_1 N_2}\right] = \left[\frac{a}{N_1}\right]\left[\frac{a}{N_2}\right]$;
(c) $\left[\frac{a}{N_1}\right] = \left[\frac{b}{N_1}\right]$ *whenever* $a \equiv b \pmod{N_1}$;
(d) $\left[\frac{c^2 b}{N_1}\right] = \left[\frac{b}{N_1}\right]$ *whenever* $\gcd(c, N_1) = 1$;
(e) $\left[\frac{-1}{N_1}\right] = (-1)^{(N_1-1)/2}$;
(f) $\left[\frac{-2}{N_1}\right] = (-1)^{(N_1^2-1)/8}$.

Lemma 12 (Reciprocity law for Jacobi symbols). [4, Theorem 9.11] *If N_1 and N_2 are two coprime odd numbers, then*

$$\left[\frac{N_1}{N_2}\right]\left[\frac{N_2}{N_1}\right] = (-1)^{\frac{(N_1-1)(N_2-1)}{4}}.$$

Lemma 13. *Let $N = \prod_{i=1}^{s} p_i^{r_i}$ be a positive odd integer. Then, the cardinality of $G(N)$ is given by:*

$$|G(N)| = \begin{cases} \phi(N) & \text{if } N \text{ is a perfect square;} \\ \phi(N)/2 & \text{if } N \text{ is not a perfect square.} \end{cases}$$

Proof. First, we prove the case where N is a perfect square. The proof is complete by the below observation

$$|G(N)| = \left|\left\{a \in \mathbb{Z}_N \middle| \left[\frac{a}{N}\right] = 1\right\}\right| = \left|\mathbb{Z}_N^\times\right| = \phi(N).$$

For the case where N is not a perfect square, we claim that there exists an $a \in \mathbb{Z}_N^\times$ such that $\left[\frac{a}{N}\right] = -1$. Without loss of generality, we assume r_1 is odd, which means that there exists $1 \leq a' < p_1$ such that $\left[\frac{a'}{p_1}\right] = -1$ (cf. [3, Theorem 9.1]). CRT implies that the integer a can be constructed by the following system of congruence equations. That is $a \equiv 1 \pmod{p_i}$ for all $i \geq 2$, and $a \equiv a' \pmod{p_1}$. Notice that

$$\left[\frac{a}{N}\right] \sum_{x \in \mathbb{Z}_N^\times} \left[\frac{x}{N}\right] = \sum_{x \in \mathbb{Z}_N^\times} \left[\frac{ax}{N}\right] = \sum_{x \in \mathbb{Z}_N^\times} \left[\frac{x}{N}\right],$$

which gives that $\sum_{x \in \mathbb{Z}_N^\times} \left[\frac{x}{N}\right] = 0$. Therefore, this equality gives us that the size of $G(n)$ is half of $\mathbb{Z}_N^\times$. □

6.3 Variant Miller-Rabin Test

For completeness, we provide the formula for the number of variants of the Miller-Rabin test, which proof is similar to Theorem 1.

Lemma 14. *Let* $p = \prod_{i=1}^{s} p_i^{r_i} \equiv 3 \pmod 4$. *Then*

$$|\mathrm{MR}(p)| = 2 \prod_{i}^{s} \gcd\big((p-1)/2, d_i\big).$$

Proof. Since $(p-1)/2$ is odd, we have

$$|\{g \in \mathbb{Z}_p^\times \mid g^{(p-1)/2} \equiv 1 \pmod p\}| = |\{g \in \mathbb{Z}_p^\times \mid g^{(p-1)/2} \equiv -1 \pmod p\}|,$$

which implies that

$$|\mathrm{MR}(p)| = 2 \cdot |\{g \in \mathbb{Z}_p^\times \mid g^{(p-1)/2} \equiv 1 \pmod p\}|.$$

Similar to Lemma 3, we consider the problem of counting the cardinality of $\frac{(p-1)}{2}$-th roots of 1 in $(\mathbb{Z}/p_i^{r_i}\mathbb{Z})^\times$ using CRT. Combining the fact $(\mathbb{Z}/p_i^{r_i}\mathbb{Z})^\times$ is cyclic, $\gcd(p, (p-1)/2) = 1$, and Lemma 1, one has the number of $\frac{(p-1)}{2}$-th roots of 1 in the group $(\mathbb{Z}/p_i^{r_i}\mathbb{Z})^\times$ is

$$\gcd((p-1)/2, p_i^{r_i-1}(p_i-1)) = \gcd((p-1)/2, d_i).$$

The above discussion implies the desired result. □

6.4 The Security Model

Our security analysis considers static and rushing adversaries. In the static model, the adversary must choose the set of corrupted parties before the protocol execution commences and cannot alter this selection. Conversely, a rushing adversary can delay messages from corrupted parties within a given round until all messages from honest parties in that round have been received. We assume a standard communication model where n parties exchange messages in synchronized rounds via pairwise private and authenticated channels, and additionally have access to a broadcast channel.

Semi-honest adversaries follow the protocol specifications honestly but attempt to learn as much information as possible from the messages they receive from other parties. For this model, we adopt the definition provided in Goldreich [35, Definition 7.5.1], detailed as follows.

Let $f : (\{0,1\}^*)^n \to (\{0,1\}^*)^n$ be an $n-$ary functionality, where $f_i(x_1, \ldots, x_n)$ denotes the i-th element of $f(x_1, \ldots, x_n)$. For $I = \{i_1, \ldots i_t\} \subset \{1, \ldots, n\}$, we let $f_I(x_1, \ldots, x_n)$ denote the subsequence $f_{i_1}(x_1, \ldots, x_n), \ldots, f_{i_t}(x_1, \ldots, x_n)$. Let Π be an n-party protocol for computing f. The view of the $i-$th party during an execution of Π on $\boldsymbol{x} = (x_1, \ldots, x_n)$, denoted $\text{VIEW}_i^{\Pi}(\boldsymbol{x})$, is $(x_i, r_i, m_{i_1}, \ldots, m_{i_\ell})$, where r_i represents the outcome of the $i-$th party's internal coin tosses, and m_{i_j} represents the $j-$th message it has received. For $I = \{i_1, \ldots, i_t\}$, we let $\text{VIEW}_I^{\Pi}(\boldsymbol{x}) := (I, \text{VIEW}_{i_1}^{\Pi}(\boldsymbol{x}), \ldots, \text{VIEW}_{i_t}^{\Pi}(\boldsymbol{x}))$.

Definition 3. *We say that Π* t-privately computes *f if there exists a probabilistic polynomial-time algorithm, denoted S, such that for every $I \subseteq \{1, \ldots, n\}$ with $|I| \leq t$, it holds that*

$$\begin{aligned} &\{(S(I, (x_{i_1}, \ldots, x_{i_t}), f_I(\boldsymbol{x})), f(\boldsymbol{x}))\}_{\boldsymbol{x} \in (\{0,1\}^*)^n} \\ \stackrel{\text{c}}{\equiv} &\{(\ \text{VIEW}_I^{\Pi}(\boldsymbol{x}),\ \text{OUTPUT}^{\Pi}(\boldsymbol{x}))\}_{\boldsymbol{x} \in (\{0,1\}^*)^n}. \end{aligned}$$

Here $\text{OUTPUT}^{\Pi}(\boldsymbol{x})$ denotes the output sequence of all parties during the execution represented in $\text{VIEW}_I^{\Pi}(\boldsymbol{x})$, and $\stackrel{\text{c}}{\equiv}$ is computationally indistinguishable of two distribution ensembles.

We consider security with abort against malicious adversaries according to the definition presented by Katz [30, Definition 1], a definition grounded in the framework of Goldreich [35]. In the malicious adversarial model, corrupted parties may deviate from the protocol specification in an arbitrary manner.

The Real Model. At the beginning of a real execution of a protocol, each party $\mathcal{P}_i$ holds the security parameter 1^λ and its input x_i. Let $f : (\{0,1\}^*)^n \to (\{0,1\}^*)^n$ be an $n-$ary functionality. The adversary $\mathcal{A}$ takes as input 1^λ, the set $I \subset \{1, \ldots, n\}$ of corrupted parties, the inputs of the corrupted parties, and an auxiliary input **aux**. The interaction of $\mathcal{A}$ with a protocol Π defines a random variable $\text{REAL}_{\Pi,\mathcal{A},I}(\lambda, \vec{x}, \textbf{aux})$ whose value is determined by the coin tosses of the adversary and the honest players. This random variable contains the output of the adversary (which may be an arbitrary function of its view) as well as the

outputs of the uncorrupted parties. We let $\mathrm{REAL}_{\Pi,\mathcal{A},I}$ denote the distribution ensemble $\{\mathrm{REAL}_{\Pi,\mathcal{A},I}(\lambda,\vec{x},\mathbf{aux})\}_{k\in\mathbb{N},(\vec{x},\mathbf{aux})\in\{0,1\}^*}$.

The ideal model. Here the parties interact with a trusted party implementing f. Each honest party $\mathcal{P}_i$ holds an input x_i as before; the adversary $\mathcal{A}'$ is again given 1^λ, the set I of corrupted parties, the inputs of all the corrupted parties, and an auxiliary input **aux**. Each honest party $\mathcal{P}_i$ sets $x'_i = x_i$ and sends x'_i to the trusted party; each corrupted party $\mathcal{P}_j$ sends an arbitrary input x'_j to the trusted party as directed by $\mathcal{A}'$. In case some corrupted party $\mathcal{P}_j$ does not send an input, x'_j is set to a default value. The trusted party computes $(y_1,\ldots,y_n) \leftarrow f(x'_1,\ldots,x'_n)$, choosing a uniformly random tape for f in case it is randomized. Then, the trusted party sends the outputs $\{y_i\}_{i\in I}$ to $\mathcal{A}'$ first. The adversary can then decide whether to abort the trusted party, or whether to allow it to continue. In the former case, the trusted party sends the special symbol $\perp$ to all honest parties as their output, where $\perp$ is assumed not to lie in the range of f. In the latter case, the trusted party sends the correct output y_i to each honest party $\mathcal{P}_i$.

The interaction of $\mathcal{A}'$ with the trusted party defines a random variable $\mathrm{IDEAL}_{f,\mathcal{A}',I}(\lambda,\vec{x},\mathbf{aux})$ whose value is determined by the random coins of the adversary and those used by the trusted party in evaluating f. This random variable contains the output of $\mathcal{A}'$ (which may be an arbitrary function of its view) as well as the outputs of the uncorrupted parties. We let $\mathrm{IDEAL}_{f,\mathcal{A}',I}$ denote the distribution ensemble $\{\mathrm{IDEAL}_{f,\mathcal{A}',I}(\lambda,\vec{x},\mathbf{aux})\}_{\lambda\in\mathbb{N},(\vec{x},\mathbf{aux})\in\{0,1\}^*}$.

Definition 4. *Let f be an n-party randomized functionality, and Π be an n-party protocol. Then Π* t-securely computes f with abort *if for any PPT adversary $\mathcal{A}$ there exists a PPT adversary $\mathcal{A}'$ such that for any $I \subseteq \{1,\ldots,n\}$ with $|I| \le t$:*

$$REAL_{\Pi,\mathcal{A},I} \overset{\mathrm{c}}{\equiv} IDEAL_{f,\mathcal{A}',I}.$$

6.5 Missing Proofs of Section 4

When D is not a square, the result have already been provided in [5, Section 5]. Here, we extend this result to general integers D.

Proposition 3. *Let D be an integer and $N := \prod_{i=1}^{s} p_i^{r_i}$ be a positive integer with* $\gcd(N, 2D) = 1$. *Then we have* $|\mathcal{Z}(D,N)| = \prod_{i=1}^{s} p_i^{r_i-1}\left(p_i - \left[\frac{D}{p_i}\right] - 1\right)$.

Proof. Similarly, applying CRT, we only consider the case $\mathcal{Z}(D,p^r)$. When $r=1$, in the beginning proof of Lemma 4 gives us $|\mathcal{Z}(D,p)| = p_i - \left[\frac{D}{p_i}\right] - 1$. When $r \ge 2$, we first the case where $Q \equiv -D/4 \pmod{p}$, and need to compute the cardinality of the set

$$\left\{(P,Q) \;\middle|\; \begin{array}{l} P^2 \equiv 0 \pmod{p}, \\ \gcd(Q,p^r) = 1, 0 \le P, Q < p^r \end{array}\right\}. \tag{8}$$

The number of solution (P,Q) is p^{r-1}, which form is $(P,Q) = \big(tp, ((tp)^2 - D)/4\big)$, where $t \in \mathbb{Z}_{p^{r-1}}$.

For the case $Q \not\equiv -D/4 \pmod{p}$, we consider $Q = a + tp$, where $t \in \mathbb{Z}_{p^{r-1}}$ and

$$a \in \mathcal{T} := \left\{ Q \not\equiv -D/4 \in \mathbb{Z}_p^{\times} \;\middle|\; x^2 \equiv D + 4Q \mod p \text{ is solvable} \right\}.$$

For each $Q \in \mathbb{Z}_{p^r}$ with $Q \not\equiv -D/4 \pmod{p}$, if $f_Q(x) := x^2 - 4Q - D \equiv 0 \pmod{p}$ has m solutions in $\mathbb{Z}_p$, and $f'_Q(a) \not\equiv 0 \pmod{p}$ for all $a \in \mathcal{T}$. Therefore, by Lemma 2, $f_Q(x) \equiv 0 \pmod{p^r}$ also has m solutions in $\mathbb{Z}_{p^r}$. Since $t \in \mathbb{Z}_{p^{r-1}}$ is arbitrary, then the number of solutions for this case is $p^{r-1}(|\mathcal{Z}(D,p)| - 1)$. Therefore, the total number of solutions is $p^{r-1}(|\mathcal{Z}(D,p)| - 1) + p^r = p^{r-1}(|\mathcal{Z}(D,p)|)$. □

This part completes the proof of the Lemma 4.

Lemma 15. *Let p be an odd prime, and D be an element of $\mathbb{Z}_p^{\times}$. Then we have, for any $r \geq 1$ and $\epsilon \in \{\pm 1\}$,*

$$|\mathcal{Z}^{\epsilon}(D,p^r)| = p^{r-1} \cdot |\mathcal{Z}^{\epsilon}(D,p)|.$$

Proof. When $r = 1$, the desired result have been proved in the proof of Lemma 4. Here, we only consider the case $2 \nmid r$, because $\mathcal{Z}^{+1}(D,N) = \mathcal{Z}(D,N)$ as $2 \mid r$, which result can be obtain by Proposition 3. Assume $\epsilon = 1$, since we have $|\mathcal{Z}(D,p^r)| = p^{r-1} \cdot |\mathcal{Z}(D,p)|$ by Proposition 3 and $|\mathcal{Z}^{-1}(D,p^r)| = |\mathcal{Z}^{(}D,p^r)| - |\mathcal{Z}^{+1}(D,p^r)|$. When $\left[\frac{-D/4}{p}\right] = 1$ holds, one has $\big(tp, ((tp)^2 - D)/4\big) \in \mathcal{Z}^{+1}(D,p^r)$ for $t \in \mathbb{Z}_{p^{r-1}}$, which implies that the cardinality of the set (8) is p^{r-1}. Using the same trick as in the Proposition 3, express Q as $a + tb$, where $t \in \mathbb{Z}_{p^{r-1}}$, and

$$a \in \left\{ Q \not\equiv -D/4 \in \mathbb{Z}_p^{\times} \;\middle|\; x^2 \equiv D + 4Q \mod p \text{ is solvable }, \left[\frac{Q}{p}\right] = 1 \right\}.$$

Notice that $\left[\frac{Q}{p^r}\right] = \left[\frac{Q}{p}\right] = 1$, since r is odd. Therefore, For each $Q \in \mathbb{Z}_{p^r}$, if $x^2 \equiv D + 4Q \mod p$ and $\left[\frac{Q}{p}\right] = 1$ has m solutions, then $x^2 \equiv D + 4Q \mod p^r$ and $\left[\frac{Q}{p^r}\right] = 1$ also has m solutions by Lemma 2. Since $t \in \mathbb{Z}_{p^{r-1}}$ is arbitrary, then the number of solutions of $\mathcal{Z}^{+1}(D,p^r)$ is $p^{r-1}\big(|\mathcal{Z}^{+1}(D,p)| - 1\big)$. For the case where $\left[\frac{-D/4}{p}\right] = -1$, there are no $(P,Q) \in \mathcal{Z}^{+1}(D,p^r)$ with $Q \equiv -D/4 \pmod{p}$. Consequently, the number of solutions of $|\mathcal{Z}^{+1}(D,p^r)|$ is given by $p^{r-1}|\mathcal{Z}^{+1}(D,p)|$ following the same reasoning as above. □

Some lemmas are used in Theorem 4.

Lemma 16. *Let p be an odd prime and $D \not\equiv 0 \pmod{p}$. Then*

$$\sum_{i=1}^{(p-1)/2} \left[\frac{i^2 + D}{p}\right] = \frac{-1 - \left[\frac{D}{p}\right]}{2}.$$

Proof. First, we prove that

$$\sum_{i=1}^{p} \left[\frac{i^2 + D}{p} \right] = -1.$$

According to Euler's criterion (cf. [34, Theorem 3.1]), the above considering sum can be written as

$$\sum_{i=1}^{p} (i^2 + D)^{\frac{p-1}{2}}.$$

Since $\mathbb{Z}_p^\times$ is a cyclic group, there exists a generator g, which induces that

$$\sum_{i=1}^{p-1} i^k \pmod p = \sum_{i=0}^{p-2} g^{ik} \pmod p = \begin{cases} 0, \text{ if } p-1 \nmid k; \\ -1, \text{ if } p-1 \mid k. \end{cases}$$

Therefore, applying this fact and expending $(i^2 + D)^{\frac{p-1}{2}}$, one has

$$\sum_{i=1}^{p} \left[\frac{i^2 + D}{p} \right] \equiv \sum_{\ell=0}^{\frac{p-1}{2}} \binom{\frac{p-1}{2}}{\ell} D^\ell \sum_{i=1}^{p} i^{p-1-2\ell} \equiv D^{(p-1)/2} \sum_{i=1}^{p} 1 + \sum_{i=1}^{p} i^{p-1} \equiv -1 \pmod p.$$

Notice that

$$\left| \sum_{i=1}^{p} \left[\frac{i^2 + D}{p} \right] \right| \leq p,$$

which implies that $\sum_{i=1}^{p} \left[\frac{i^2 + D}{p} \right] = -1$ or $p-1$. However, if $\sum_{i=1}^{p} \left[\frac{i^2 + D}{p} \right] = p-1$, then we must have $p-1$ terms equal to 1 and exactly 1 term $a^2 + D \equiv 0 \pmod p$ with $a \equiv -a \pmod p$, which implies that $a \equiv 0 \pmod p$, since p is odd. Therefore, one has $D \equiv 0 \pmod p$, which gives us a contradiction.

The proof is completed by the above fact and the following observation.

$$2 \sum_{i=1}^{\frac{p-1}{2}} \left[\frac{i^2 + D}{p} \right] = \sum_{i=1}^{p-1} \left[\frac{i^2 + D}{p} \right] = -1 - \left[\frac{D}{p} \right].$$

□

Lemma 17. *Let $p_i > 3$ be distinct primes and $s \geq 1$. Then*

$$\frac{\prod_{i=1}^{s} (p_i - 1)}{\prod_{i=1}^{s} (p_i - 2) - 1} \leq \frac{(p_{\min} - 1)^s}{(p_{\min} - 2)^s - 1}.$$

Here $p_{\min} := \min\limits_{1 \leq i \leq s} \{p_i\}$.

Proof. Observe that

$$\frac{\prod_{i=1}^{s}(p_i-1)}{\prod_{i=1}^{s}(p_i-2)-1}=\left(\frac{\prod_{i=1}^{s}(p_i-1)}{\prod_{i=1}^{s}(p_i-2)}\right)\left(\frac{\prod_{i=1}^{s}(p_i-2)}{\prod_{i=1}^{s}(p_i-2)-1}\right).$$

Since $(p_i-1)/(p_i-2)$ is a decreasing function for p_i, we have

$$\prod_{i=1}^{s}\left(\frac{p_i-1}{p_i-2}\right)\leq\frac{(p_{\min}-1)^s}{(p_{\min}-2)^s}.$$

The proof is completed by the facts that $x/(x-1)$ is decreasing and $\prod_{i=1}^{s}(p_i-2)\geq(p_{\min}-2)^s$. □

Lemma 18. *Let $p_i>5$ be distinct primes and $s\geq 1$. Then for any $1\leq j\leq s$,*

$$\prod_{i=j}^{s}(p_i-1)<\prod_{i=j}^{s}2(p_i-2)-2^{s-j+1}.$$

Proof. For all $p_i\geq 5$, we have

$$\prod_{i=j}^{s}(p_i-1)+2^{s-j+1}\leq\prod_{i=j}^{s}\big((p_i-1)+2\big)=\prod_{i=j}^{s}(p_i+1)\leq\prod_{i=j}^{s}2(p_i-2).$$

□

6.6 The Identical Distribution of P_j, P'_j in Theorem 3

In this section, for an integer m, if $\sqrt{D}\in\mathbb{Z}_m^{\times}$, then $\pmod m$ refers to the module $m\mathbb{Z}$; otherwise, if $\sqrt{D}\notin\mathbb{Z}_m^{\times}$, $\pmod m$ refers to the module $m\mathcal{O}_D$. To investigate the distribution of P_j and P'_j, we will examine the relationship between $S_{\text{real}(m,b)}$ and $S_{\text{ideal}(m,b)}$ given an odd integer m and $b\in\{0,1\}$. Here

$$S_{\text{real}(m,b)}:=\left\{P\in\mathbb{Z}_m\;\middle|\;\left[\frac{(P^2-D)/4}{m}\right]=(-1)^b\right\},\text{ and}$$

$$S_{\text{ideal}(m,b)}:=\left\{\frac{2\sqrt{D}}{a^2(-1)^b-1}+\sqrt{D}\;\middle|\;a=\frac{v+w\sqrt{D}}{v-w\sqrt{D}},v,w\in\mathbb{Z}_m,\ v^2-w^2D\in\mathbb{Z}_m^{\times},a^2(-1)^b\not\equiv 1\pmod m\right\}.$$

Then we have

Lemma 19. *If p is an odd prime, and D is an integer with $\left[\frac{-D}{p}\right]=-1$, then we have $S_{\text{real}(p,b)}=S_{\text{ideal}(p,b)}$ for $b\in\{0,1\}$.*

Proof. For any $P' \in S_{\text{ideal}(p,b)}$, we have

$$\begin{aligned}(P'^2 - D)/4 =& \left(\frac{\sqrt{D}}{a^2(-1)^b - 1}\right)^2 + \frac{D}{a^2(-1)^b - 1} \\ =& D\left(\frac{(v^2 - w^2D)^2(-1)^b}{\left((v + w\sqrt{D})^2(-1)^b - (v - w\sqrt{D})^2\right)^2}\right)\end{aligned}$$

Therefore,

$$\left[\frac{(P'^2 - D)/4}{p}\right] = \begin{cases} \left[\frac{1/(v^2w^2)}{p}\right] = 1 & \text{, if b = 0;} \\ \left[\frac{-D/(v^2+w^2D)^2}{p}\right] = -1 & \text{, if b = 1.} \end{cases}$$

We derive $\left[\frac{(P'^2-D)/4}{p}\right] = (-1)^b$ and $S_{\text{real}(p,b)} \supseteq S_{\text{ideal}(p,b)}$. On the other hand, let P be an element in $S_{\text{real}(p,b)}$. We assume that there exists a belonging the set

$$\left\{\frac{v + w\sqrt{D}}{v - w\sqrt{D}}\middle|\ v, w \in \mathbb{Z}_p, v^2 - w^2D \in \mathbb{Z}_{p^\times}, \right. \\ \left. (v + w\sqrt{D})^2 \equiv (-1)^b(v - w\sqrt{D})^2 \pmod p\right\}$$

such that $a^2(-1)^b = \frac{P+\sqrt{D}}{P-\sqrt{D}} \not\equiv 1 \pmod p$. Then we have

$$P \equiv \frac{2\sqrt{D}}{\frac{P+\sqrt{D}}{P-\sqrt{D}} - 1} + \sqrt{D} \equiv \frac{2\sqrt{D}}{a^2(-1)^b - 1} + \sqrt{D} \pmod p,$$

which implies $S_{\text{real(p,b)}} \subseteq S_{\text{ideal(p,b)}}$. To prove the assumption, we split it into two cases.

Case1: $\left[\frac{D}{p}\right] = 1$ (i.e., $\sqrt{D} \in \mathbb{Z}_p^\times$).

Note that $\left[\frac{P^2-D}{p}\right] = 1$, because of P be an element in $S_{\text{real}(p,b)}$. Since the condition in Lemma gives $\left[\frac{-D}{p}\right] = -1$, we have $\left[\frac{-1}{p}\right] = -1$. Then one has

$$\left[\frac{(-1)^b(P + \sqrt{D})/(P - \sqrt{D})}{p}\right] = \left[\frac{(-1)^b(P + \sqrt{D})^2/(P^2 - D)}{p}\right]$$
$$= \left[\frac{(-1)^b(P^2 - D)}{p}\right] = \left[\frac{(-1)^b}{p}\right](-1)^b = 1.$$

There exists $t \in \mathbb{Z}_p^\times$ such that $t^2 \equiv (-1)^b\frac{P+\sqrt{D}}{P-\sqrt{D}} \pmod p$. Assume $t \not\equiv 1 \pmod p$, we take $(v, w) = (\frac{t+1}{t-1}\sqrt{D}, 1)$ and then $a^2 \equiv \left(\frac{v+w\sqrt{D}}{v-w\sqrt{D}}\right)^2 \equiv (-1)^b\frac{P+\sqrt{D}}{P-\sqrt{D}} \pmod p$. If $t = 1$, we set $(v, w) = (1, 0)$, then $a^2 = 1$.

Case2: $\left[\frac{D}{p}\right] = -1$.

If $b = 0$ (resp. $b = 1$), then we take $(v, w) = (\frac{P+\sqrt{4(P^2-D)}}{2}, 1) \in \mathbb{Z}_p \times \mathbb{Z}_p$ (resp. $(v, w) = (\frac{D+\sqrt{D(D-P^2)}}{P}, 1) \in \mathbb{Z}_p \times \mathbb{Z}_p$). Racall that $a = \frac{v+w\sqrt{D}}{v-w\sqrt{D}}$. Then, one has $a^2 \equiv (-1)^b \frac{P+\sqrt{D}}{P-\sqrt{D}} \pmod{p}$. □

Assume that p is an odd prime and $D \in \mathbb{Z}_p^\times$. Let

$$G := \left\{ (a, b) \;\middle|\; a, b \in \mathbb{Z}_p, a^2 - b^2 D \in \mathbb{Z}_p^\times \right\}.$$

Given $g_1 = (a_1, b_1), g_2 = (a_2, b_2) \in G$, define $g_1 * g_2 = (a_1a_2 + b_1b_2D, a_1b_2 + b_1a_2)$. Then G is a group with the identity $(1, 0)$, and its inverse of $g = (a, b)$ is $\big(a/(a^2 - b^2D), -b/(a^2 - b^2D)\big)$. Let

$$H := \left\{ \frac{a + b\sqrt{D}}{a - b\sqrt{D}} \in \mathbb{Z}_p(\sqrt{D}) \;\middle|\; a, b \in \mathbb{Z}_p, a^2 - b^2 D \in \mathbb{Z}_p^\times \right\},$$

which is also a group under the field multiplication. Here $\mathbb{Z}_p(\sqrt{D})$ is the fractional field of the ring $\{a + b\sqrt{D} \mid a, b \in \mathbb{Z}_p\}$. The inverse of any $h = \frac{a+b\sqrt{D}}{a-b\sqrt{D}} \in H$ is $\frac{a-b\sqrt{D}}{a+b\sqrt{D}}$, and the identity is 1.

Lemma 20. *Let p be an odd prime, and $D \in \mathbb{Z}_p^\times$. Consider a group homomorphism $f : G \to H$ defined by*

$$g = (a, b) \in G \mapsto \left(\frac{a + b\sqrt{D}}{a - b\sqrt{D}} \right)^2 \in H.$$

Then the set of $f(g)$ forms a subgroup of H, and $|\ker(f)| = 2p - 2$.

Proof. It is a subgroup can be verified directly using the definition. We omit this step. The map f is a group homomorphism, which can be verified by showing that for any $(a_1, b_1), (a_2, b_2) \in G$:

$$f(a_1, b_1)f(a_2, b_2) = \frac{a_1a_2 + b_1b_2D + (a_1b_2 + a_2b_1)\sqrt{D}}{a_1a_2 + b_1b_2D - (a_1b_2 + a_2b_1)\sqrt{D}} = f((a_1, b_1) * (a_2, b_2)).$$

Let $g = (a, b) \in G$ with $f(g) = 1$. Then $\left(\frac{a+b\sqrt{D}}{a-b\sqrt{D}}\right)^2 = 1$, which implies that $ab\sqrt{D} = 0$. Therefore $a = 0$ or $b = 0$. If $a = 0$ and $b \in \mathbb{Z}_p^\times$, then $f(g) = 1$. Similarly, if $b = 0$, then $a \in \mathbb{Z}_p^\times$, then $f(g) = 1$. In conclusion, the cardinality of kernel of f is $2p - 2$. □

Proposition 4. *If $N = pq$ is an odd RSA modulus, and D is an integer with $\left[\frac{-D}{p}\right] = \left[\frac{-D}{q}\right] = -1$, then we have $S_{\text{real}(N,0)} = S_{\text{ideal}(N,0)} \cup S_{\text{ideal}(N,1)}$. Furthermore, uniformly sampling $u, v \in \mathbb{Z}_N$, $b \in \{0, 1\}$ with $u^2 - v^2D \in \mathbb{Z}_N^\times$ and $a^2(-1)^b \not\equiv 1 \pmod{N}$ is equivalent to randomly selecting from the set $S_{\text{real}(N,0)}$.*

Proof. According to the CRT, we have

$$S_{\mathrm{real}(N,0)} = \left(S_{\mathrm{real}(p,0)} \times S_{\mathrm{real}(q,0)}\right) \cup \left(S_{\mathrm{real}(p,1)} \times S_{\mathrm{real}(q,1)}\right).$$

Similarly, one has

$$S_{\mathrm{ideal}(N,0)} = S_{\mathrm{ideal}(p,0)} \times S_{\mathrm{ideal}(q,0)}, \text{ and}$$

$$S_{\mathrm{ideal}(N,1)} = S_{\mathrm{ideal}(p,1)} \times S_{\mathrm{ideal}(q,1)}.$$

Thus, according to Lemma 19, there exists a bijective map from $S_{\mathrm{ideal}(N,0)} \cup S_{\mathrm{ideal}(N,1)}$ to $S_{\mathrm{real}(N,0)}$.

Notice that to ensure $S_{\mathrm{ideal}(N,b)}$ is well-defined, we need to assume $a^2(-1)^b \not\equiv 1 \pmod{N}$. Specifically, for any odd prime p satisfying $\left[\frac{-D}{p}\right] = -1$, then this condition is equivalent to $a^2 \equiv 1 \mod p$ and $b = 0$, which is also equivalent to $u = 0, w \in \mathbb{Z}_p^\times$ or $u \in \mathbb{Z}_p^\times, v = 0$. Let $T_N := \{(u,w) : u^2 - w^2 D \in \mathbb{Z}_N^\times\}$. Lemma 20 says that there is a surjective map f from T_p to the set $S_{\mathrm{ideal}(p,b)}$ for any $b \in \{0,1\}$ such that $|f^{-1}(x)| = 2p-2$ for all $x \in S_{\mathrm{ideal}(p,b)}$. This map induces a bijective map

$$T_p - \{u,v \mid uv = 0, (u,v) \neq (0,0)\} \times T_p \to S_{\mathrm{ideal}(p,0)} \times S_{\mathrm{ideal}(p,1)}.$$

In fact, the set $\{u,v \mid uv = 0, (u,v) \neq (0,0)\}$ is $f^{-1}(1)$.

Lastly, the CRT says that $T_N = T_p \times T_q$. Therefore, there exists a map g such that $|g^{-1}(x)| = (2p-2)(2q-2)$ for all $x \in S_{\mathrm{ideal}(N,0)} \cup S_{\mathrm{ideal}(N,1)}$. □

6.7 A security proof of Protocol $\pi_{\mathbf{Leg}}$

Proposition 5. *Protocol* π_{Leg} $(n-1)$-privately computes *the functionality* $\mathcal{F}_{\mathrm{Leg}}$ *in* $\mathcal{F}_{\mathrm{ModMul}}$*-hybrid model.*

Proof. We construct the simulator $\mathcal{S}$ to simulate the transcript of π_{Leg}. Suppose $\mathcal{S}$ is given input

$$\left(\mathcal{P}^*, \{\mathfrak{p}_i\}_{i\in\mathcal{P}^*}, p \pmod 4, D, \left[\frac{-D}{p}\right]\right).$$

1: $\mathcal{S}$ uniformly samples $s \in \mathbb{Z}_D^\times$ and $\mathfrak{s}_i \in \mathbb{Z}_D$ for $i \in \{1,\ldots,n\}$ such that $\sum_{i=1}^n \mathfrak{s}_i \equiv s \pmod D$.
2: $\mathcal{S}$ uniformly samples $\mathfrak{s}'_i \in \mathbb{Z}_D$ for $i \in \{1,\ldots,n\}$ such that $\sum_{i=1}^n \mathfrak{s}'_i \equiv s^2 \pmod D$.
3: $\mathcal{S}$ uniformly samples $r \in \mathbb{Z}_D^\times$ such that

$$\left[\frac{r}{D}\right] = \begin{cases} -\left[\frac{-D}{p}\right], & \text{if } p \equiv 3 \pmod 4 \text{ and } D \equiv 1 \pmod 4; \\ \left[\frac{-D}{p}\right], & \text{otherwise.} \end{cases}$$

4: $\mathcal{S}$ uniformly samples $\mathfrak{r}_i \in \mathbb{Z}_D$ for $i \in \{1,\ldots,n\}$ such that $\sum_{i=1}^n \mathfrak{r}_i \equiv r \pmod D$.

5: $\mathcal{S}$ outputs

$$(\{\mathfrak{p}_i\}_{i\in\mathcal{P}^*}, p \pmod 4, D, \{\mathfrak{s}_i\}_{i\in\mathcal{P}^*}, \{\mathfrak{s}'_i\}_{i\in\mathcal{P}^*}, \{\mathfrak{r}_i\}_{i\in\mathcal{P}^*}, \{\mathfrak{r}_i\}_{i\in\{1,\ldots,n\}\setminus\mathcal{P}^*})$$

Because $\mathcal{F}_{\text{Leg}}$ is a deterministic function, we only need to prove

$$\begin{aligned}&\left\{\mathcal{S}\Big(\mathcal{P}^*, \{\mathfrak{p}_i\}_{i\in\mathcal{P}^*}, p \pmod 4, D, \left[\frac{-D}{p}\right]\Big)\right\}\\ \overset{c}{\equiv}&\{\text{view}^{\pi_{\text{Leg}}}_{\mathcal{P}^*}(\mathcal{P}^*, \{\mathfrak{p}_i\}_{i\in\mathcal{P}^*}, p \pmod 4, D)\}\end{aligned}$$

for any $\mathcal{P}^* \subseteq \{1,\ldots,n\}, |\mathcal{P}^*| \leq n-1, \{\mathfrak{p}_i\}_{i=1}^n$ and prime D. In the beginning, fixed any $\{\mathfrak{p}_i\}_{i=1}^n$ and D, we claim that the output of

$$\mathcal{S}\left(\mathcal{P}^*, \{\mathfrak{p}_i\}_{i\in\mathcal{P}^*}, p \pmod 4, D, \left[\frac{-D}{p}\right]\right)$$

and the view

$$\text{view}^{\pi_{\text{Leg}}}_{\mathcal{P}^*}(\mathcal{P}^*, \{\mathfrak{p}_i\}_{i\in\mathcal{P}^*}, p \pmod 4, D)$$

are identical. Observe that

$$\begin{aligned}\left[\frac{p}{D}\right] &= \left[\frac{D}{p}\right]\cdot(-1)^{\frac{p-1}{2}\frac{D-1}{2}} = \left[\frac{-D}{p}\right]\cdot\left[\frac{-1}{p}\right]\cdot(-1)^{\frac{p-1}{2}\frac{D-1}{2}}\\ &= \left[\frac{-D}{p}\right]\cdot(-1)^{\frac{p-1}{2}}\cdot(-1)^{\frac{p-1}{2}\frac{D-1}{2}}\end{aligned}$$

implies that $\left[\frac{p}{D}\right] = \left[\frac{r}{D}\right]$. The facts that D is a prime, and s is uniformly randomly chosen from $\mathbb{Z}_D^\times$, which gives us the identical distribution between $\{s^2 p \mid s \in \mathbb{Z}_D^\times\}$ with $\left\{r \in \mathbb{Z}_D^\times \mid \left[\frac{r}{D}\right] = (-1)^{\frac{(p-1)(D-3)}{4}}\left[\frac{-D}{p}\right]\right\}$. Due to $|\mathcal{P}^*| < n$, $\mathfrak{s}_i, \mathfrak{s}'_i$ in the $\text{view}^{\pi_{\text{Leg}}}_{\mathcal{P}^*}(\mathcal{P}^*, N, \{\mathfrak{p}_i\}_{i\in\mathcal{P}^*}, p \pmod 4, D)$ and

$$\mathcal{S}\left(\mathcal{P}^*, N, \{\mathfrak{p}_i\}_{i\in\mathcal{P}^*}, p \pmod 4, D, \left[\frac{-D}{p}\right]\right)$$

are both independently and uniformly distributed in $\mathbb{Z}_D$. We conclude that for any $\mathcal{P}^* \subseteq \{1,\ldots,n\}, |\mathcal{P}^*| \leq n-1, \{\mathfrak{p}_i\}_{i=1}^n$, and prime D

$$\begin{aligned}&\left\{\mathcal{S}\left(\mathcal{P}^*, \{\mathfrak{p}_i\}_{i\in\mathcal{P}^*}, p \pmod 4, D, \left[\frac{-D}{p}\right]\right)\right\}\\ \overset{c}{\equiv}&(\{\mathfrak{p}_i\}_{i\in\mathcal{P}^*}, p \pmod 4, D, \{\mathfrak{s}_i\}_{i\in\mathcal{P}^*}, \{\mathfrak{s}'_i\}_{i\in\mathcal{P}^*}, \{\mathfrak{r}_i\}_{i\in\mathcal{P}^*}, \{\mathfrak{r}_i\}_{i\in\{1,\ldots,n\}\setminus\mathcal{P}^*})\\ \overset{c}{\equiv}&(\{\mathfrak{p}_i\}_{i\in\mathcal{P}^*}, p \pmod 4, D, \{\mathfrak{s}_i\}_{i\in\mathcal{P}^*}, \{\mathfrak{s}'_i\}_{i\in\mathcal{P}^*}, \{s^2\mathfrak{p}_i\}_{i\in\mathcal{P}^*}, \{s^2\mathfrak{p}_i\}_{i\in\{1,\ldots,n\}\setminus\mathcal{P}^*})\\ \overset{c}{\equiv}&\{\text{view}^{\pi_{\text{Leg}}}_{\mathcal{P}^*}(\mathcal{P}^*, \{\mathfrak{p}_i\}_{i\in\mathcal{P}^*}, p \pmod 4, D)\}.\end{aligned}$$

□

6.8 Some Lemmas of Statistical Distance

In this paper, we use the following statistical distance to prove Proposition 6, 7 for our malicious model, showing that the real view and the ideal view are indistinguishable.

Definition 5. [11, Definition 3.5] *Let X, Y be two random vectors that takes values in a finite set S. The statistical distance between X and Y is defined as*

$$\mathrm{SD}(X,Y) = \frac{1}{2}\sum_{s\in S} \left|\Pr(X=s) - \Pr(Y=s)\right|.$$

Here, we will study the statistical distance between the distributions of the sum of two random variables modulo a positive integer A. More formally, let $X = (X_1, \ldots, X_n)$ and $Y = (Y_1, \ldots, Y_n)$ be random vectors defined on a probability space. We define $X + Y \pmod A$ to mean the component-wise modulo $\big(X_1 + Y_1 \pmod A, \ldots, X_n + Y_n \pmod A\big)$. In addition, when S is a finite set, we define U_S to be the random variable uniformly distributed over S.

Lemma 21. *Suppose $X = (X_1, \ldots, X_n)$, $Y = (Y_1, \ldots, Y_n)$ are random vectors with X_i, Y_i are all independent random variable on the finite set S. Then*

$$\mathrm{SD}(X,Y) \leq \sum_{i=1}^{n} \mathrm{SD}(X_i, Y_i).$$

Proof. This is a standard application of the hybrid argument. We will briefly describe the approach. Consider $n-1$ random vectors $H_1, \ldots, H_n$, defined as follows:
$H_1 := (Y_1, X_2, \ldots, X_n), H_2 := (Y_1, Y_2, X_3, \ldots, X_n), \ldots, H_{n-1} = (Y_1, \ldots, Y_{n-1}, X_n)$.
Then one has

$$\mathrm{SD}(X,Y) \leq \mathrm{SD}(X,H1) + \mathrm{SD}(H_1,H_2) + \ldots + \mathrm{SD}(H_{n-1},Y) = \sum_{i=1}^{n} \mathrm{SD}(X_i,Y_i).$$

□

Lemma 22. *Let X, Z be independent random variables on the set $\{0, \ldots, A-1\}$, and U be the uniformly random variable on the set $\{0, \ldots, A-1\}$. Then*

$$\mathrm{SD}(X + Z \pmod A, U \pmod A) \leq \mathrm{SD}(X,U).$$

Proof. Note that the sum of two distributions $U + Z \pmod A$ is still U for any distribution Z. Therefore, we only need to prove that

$$\mathrm{SD}(X + Z \pmod A, U + Z \pmod A) \leq \mathrm{SD}(X,U).$$

From the definition of SD, one has

$$\begin{aligned}
&\mathrm{SD}(X+Z \pmod A, U+Z \pmod A)\\
=&\frac{1}{2}\sum_z \left|\sum_x \Pr(X=x)\Pr(Z=z-x) - \sum_x \Pr(U=x)\Pr(Z=z-x)\right|\\
\le&\frac{1}{2}\sum_z\sum_x |\Pr(X=x)-\Pr(U=x)|\Pr(Z=z-x)\\
=&\frac{1}{2}\sum_x |\Pr(X=x)-\Pr(U=x)|\sum_z \Pr(Z=z-x) = \mathrm{SD}(X \pmod A, U \pmod A).
\end{aligned}$$

□

Lemma 23. *Let $B > A > 0$ be positive integers. Write $2B+1 = QA+r$, where $0 \le r < A$. Then the set*

$$\{x \pmod A \mid -B \le x \le B\} = \bigcup_{i=0}^{A-1} [i].$$

Here $[i]$ is equivalent class of i of modulo A. Furthermore, there exist $A-r$ equivalence classes $[i]$ whose cardinality is Q, and the remaining r equivalence classes have cardinality $Q+1$.

Proof. The set $\{x \pmod A \mid -B \le x \le B\}$ is Q copies of a complete residue system modulo A. And there are an additional r distinct residue classes that will each have one more element. □

Lemma 24. *Assume that n, A, B are positive integers with $A < 2B+1$. Let $X = (X_1, \ldots, X_n)$ be the random vector with i.i.d. $X_i = U_B := U_{\{x \pmod A \mid -B \le x \le B\}}$, $Z = (Z_1, \ldots, Z_n)$ be a random vector with a random variable Z_i on the set $\{0, \ldots, A-1\}$, and $U = (U_1, \ldots, U_n)$ be a random vector with i.i.d. $U_i = U_{\{0,1,\ldots,A-1\}}$ for all $1 \le i \le n$. Then*

$$\mathrm{SD}\big(X+Z \pmod A, U\big) \le \frac{n}{4\lfloor (2B+1)/A \rfloor}.$$

Proof. If $A \mid (2B+1)$, then Lemma 23 says that U_B is the uniform random variable on the set $\{0, 1, \ldots, A-1\}$. Therefore, the $\mathrm{SD}\big(X+Z \pmod A, U\big) = 0$, since for all $1 \le i \le n$, $X_i + Z_i$ is also the uniform variable on the set $\{0, \ldots, A-1\}$. Therefore, we only need to consider the case $A \nmid (2B+1)$. Write $2B+1 = QA+r$ with $0 < r < A$. In particular, when $n = 1$, Lemma 22 says that

$$\mathrm{SD}(X_i + Z_i \pmod A, U_i) \le \mathrm{SD}(X_i, U_i).$$

Notice that $f(x) = \frac{x(A-x)}{QA+x}$ has maximal $A(\sqrt{Q+1} - \sqrt{Q})^2$ (i.e., the critical point is $A(\sqrt{Q(Q+1)} - Q)$) in the set $[0, A]$. Now, by the Lemma 23, we have

$$\begin{aligned}\mathrm{SD}(X_i, U_i) =& \frac{1}{2}\left[r\left(\frac{Q+1}{2B+1} - \frac{1}{A}\right) + (A-r)\left(\frac{1}{A} - \frac{Q}{2B+1}\right)\right] \\ =& \frac{r(A-r)}{(QA+r)A} \leq \frac{1}{(\sqrt{Q+1}+\sqrt{Q})^2} \leq \frac{1}{4Q}.\end{aligned}$$

Therefore, Lemma 21 implies that

$$\mathrm{SD}(X + Z \pmod A, U) \leq \frac{n}{4Q}.$$

□

Proposition 6. *Assume that n, A, B are positive integers with $A < 2B+1$. Let $X = (X_1, \ldots, X_n)$ be a random vector representing the output of $\mathcal{F}_{Zero}(n, B)$, and $U = (U_1, \ldots, U_{n-1}, -\sum_{i=1}^{n-1} U_i)$ be a random vector with $U_i = U_{\{0,1,\ldots,A-1\}}$ are i.i.d. for all $1 \leq i \leq n-1$. For any vector $s = (s_1, \ldots, s_n)$, where $s_i \in \{0, 1 \ldots, A-1\}$ for all $1 \leq i \leq n$. Then*

$$\mathrm{SD}\big(X + s \pmod A, U + s \pmod A\big) \leq \frac{n-1}{4\lfloor (2B+1)/A \rfloor}.$$

Here $\lfloor \cdot \rfloor$ is the floor function.

Proof. Note that the definition of $\mathcal{F}_{\mathrm{Zero}}(n, B)$ gives that $\sum_{i=1}^{n} X_i = 0$, which means $X_n = -\sum_{i=1}^{n-1} X_i$. For each $1 \leq i \leq n-1$, the random variable X_i can be regarded as the random variable $Z_i + U_B$ for some random variable Z_i. Here $U_B := U_{\{-B,\ldots,B\}}$. Let $f(x_1, \ldots, x_{n-1}) = (x_1, \ldots, x_{n-1}, -\sum_{i=1}^{n-1} x_i)$ be a injective function from $\mathbb{Z}_A^{n-1}$ to $\mathbb{Z}_A^n$. By the above observation, and the definition of statistical distance, we have

$$\begin{aligned}&\mathrm{SD}\big(X + s \pmod A, U + s \pmod A\big) \\ =&\mathrm{SD}\big(X, U\big), \text{ by Lemma 22} \\ =&\mathrm{SD}\big(f(X_1 \pmod A, \ldots, X_{n-1} \pmod A), f(U_1, \ldots, U_{n-1})\big) \\ =&\mathrm{SD}\left((X_1 \pmod A, \ldots, X_{n-1} \pmod A), (U_1, \ldots, U_{n-1})\right), \text{ by defintion of the statistical distance} \\ =&\mathrm{SD}\left((Z_1 + U_B \pmod A, \ldots, Z_{n-1} + U_B \pmod A)), (U_1, \ldots, U_{n-1})\right) \\ \leq& \frac{n-1}{4\lfloor (2B+1)/A \rfloor}, \text{ by Lemma 24.}\end{aligned}$$

□

Proposition 7. *Let A, B, n be positive integers with $B > n$ for all $n \geq 1$. Assume that $X = (X_1, \ldots, X_n)$, $X' = (X'_1, \ldots, X'_n)$ are n-dimensional random vectors, where $X_i = X'_i = U_{\{0,1,\ldots,AB-1\}}$ are i.i.d.. For any n-dimensional vector $s = (s_i)$ with $-A \leq s_i \leq A$ for all $1 \leq i \leq n$, the statistical distance between X and $X' - s$ is at most $\frac{n}{B}$.*

Proof. If $0 \le s_n \le A$, by the definition of statistical distance, we have

$$\begin{aligned}
&\mathrm{SD}(X, X'-s)) \\
&= \sum_{\substack{x=(x_i)\\ x_i\in\mathbb{Z}}} \frac{1}{2}\left|\Pr[X=x]-\Pr[X'=s+x]\right| \\
&= \sum_{x=(x_1,\ldots,x_{n-1})} \frac{1}{2}\left(\sum_{x_n=-s_n}^{-1} + \sum_{x_n=0}^{AB-s_n-1} + \sum_{x_n=AB-s_n}^{AB-1} \left|\prod_{i=1}^{n}\Pr[X_i=x_i]-\prod_{i=1}^{n}\Pr[X_i'=s_i+x_i]\right|\right) \\
&= \frac{1}{2}\sum_{x=(x_1,\ldots,x_{n-1})}\left(\frac{s_n}{AB}\prod_{i=1}^{n-1}\Pr[X_i'=s_i+x_i] + \frac{s_n}{AB}\prod_{i=1}^{n-1}\Pr[X_i=x_i]\right. \\
&\qquad\left. + \frac{AB-s_n}{AB}\left|\prod_{i=1}^{n-1}\Pr[X_i=x_i]-\prod_{i=1}^{n-1}\Pr[X_i'=s_i+x_i]\right|\right) \\
&= \frac{|s_n|}{AB} + \left(1-\frac{|s_n|}{AB}\right)\sum_{x=(x_1,\ldots,x_{n-1})}\frac{1}{2}\left(\left|\prod_{i=1}^{n-1}\Pr[X_i=x_i]-\prod_{i=1}^{n-1}\Pr[X_i'=s_i+x_i]\right|\right)
\end{aligned}$$

Similarly, if $-A \le s_n < 0$ we have

$$\begin{aligned}
&\mathrm{SD}(X, X'-s)) \\
&= \frac{|s_n|}{AB} + \left(1-\frac{|s_n|}{AB}\right)\sum_{x=(x_1,\ldots,x_{n-1})}\frac{1}{2}\left(\left|\prod_{i=1}^{n-1}\Pr[X_i=x_i]-\prod_{i=1}^{n-1}\Pr[X_i'=s_i+x_i]\right|\right)
\end{aligned}$$

Therefore, for computing $\mathrm{SD}(X, X'-s))$, we can consider the recursive sequence F_n defined as follows, $F_1 = \frac{|s_1|}{AB}$, and $F_n = \frac{|s_n|}{AB} + \left(1-\frac{|s_n|}{AB}\right)F_{n-1}$. It is easy to obtain the closed form is

$$F_n = 1-\prod_{i=1}^{n}\left(1-\frac{|s_i|}{AB}\right) = \sum_{i=1}^{n}\frac{|s_i|}{AB} - \sum_{1\le i<j\le n}\frac{|s_i||s_j|}{(AB)^2} + \ldots - (-1)^n\frac{\prod_{i=1}^{n}|s_i|}{(AB)^n}.$$

Notice that the Taylor expansion of $\ln(1-x) = -\sum_{k=1}^{\infty}\frac{x^k}{k}$ for $|x|<1$. Since $\sum_{i=1}^{n}|s_i| < AB$, one has

$$\sum_{k=1}^{\infty}\frac{\left(\sum_{i=1}^{n}|s_i|/(AB)\right)^k}{k} \ge \sum_{i=1}^{n}\sum_{k=1}^{\infty}\frac{\left(|s_i|/AB\right)^k}{k},$$

which implies that

$$1-\sum_{i=1}^{n}\frac{|s_i|}{AB} \le \prod_{i=1}^{n}\left(1-\frac{|s_i|}{AB}\right),$$

since taking ln to the both side, applying Taylor expansion of $\ln(1-x)$. Finally, because $|s_i| \leq A$ for all $1 \leq i \leq n$, the closed form gives us

$$\mathrm{SD}(X, X' - s)) \leq \sum_{i=1}^{n} \frac{|s_i|}{AB} \leq \frac{n}{B}.$$

□

Proposition 8. *Let A, B, n be positive integers with $B > n$ for all $n \geq 1$. Assume that $X = (X_1, \ldots, X_n)$, $X' = (X'_1, \ldots, X'_n)$ are n-dimensional random vectors, where $X_i = U_{\{0,1,\ldots,A-1\}}$ and $X'_i = U_{\{0,1,\ldots,B-1\}}$ are i.i.d.. The statistical distance between $X \pmod A$ and $X' \pmod A$ is at most $\frac{n}{4\lfloor B/A \rfloor}$.*

Proof. Using the same proof strategy as in Lemma 24, we only briefly outline the main idea of the proof. Let $B = QA + r$ with $0 \leq r < A$, by Lemma 21 the definition of statistical distance, we have

$$\mathrm{SD}\big(X, X'\big) \leq n \cdot \mathrm{SD}(X_1, X'_1) = \frac{nr(A-r)}{(QA+r)A} \leq \frac{n}{4Q}.$$

□

6.9 Missing Functionalities and Protocols

The functionality describes that each party $\mathcal{P}_i$ has two shares, $\mathfrak{x}_i$ and $\mathfrak{y}_i$, the functionality outputs $\mathfrak{z}_i$ where $[z]_N = [xy]_N$ and assigns to $\mathcal{P}_i$.

Functionality 8 Modular Multiplication $\mathcal{F}_{\mathrm{ModMul}}(n)$

Inputs: Each party $\mathcal{P}_i$ has shares $[x]_N$, $[y]_N$ and N.

Outputs: Each party has shares of $[z]_N = [x \cdot y]_N$, with uniformly random $\mathfrak{z}_i \in \mathbb{Z}_N$ for all $1 \leq i \leq n$.

The functionality below is to ensure that participants can learn $\prod_i y_i$ without revealing their own y_i. In our setting, we define the finite group G utilized by the functionality $\mathcal{F}_{\mathrm{Shuffle}}$ as follows:

$$G := \begin{cases} \mathbb{Z}_N^{\times}, \text{ if } \sqrt{D} \in \mathbb{Z}_N; \\ \left\{ a + b\sqrt{D} \mid a, b \in \mathbb{Z}_N, a^2 - b^2 D \in \mathbb{Z}_N^{\times} \right\}, \text{ otherwise.} \end{cases}$$

Note that the inverse of $x = v + w\sqrt{D} \in G$ is given by $x^{-1} = \frac{v - w\sqrt{D}}{v^2 - w^2 D} \in G$, and the identity is 1.

Functionality 9 $\mathcal{F}_{\text{Shuffle}}(n, G)$

Inputs: Each party $\mathcal{P}_i$ has y_i in a finite group G.

Outputs: Each party $\mathcal{P}_i$ receives $y := \prod_{i=1}^{n} y_i \in G$.

In the following protocol [7], each party splits their own input y_i into $n-1$ partitions and randomly send one share to other parties to avoid revealing their own input y_i. Every party will calculate the product of all obtained shares $\prod_i z_i$ and publish it. Eventually, we have $\prod_{i=1}^{n} z_i = \prod_{i=1}^{n} y_i$.

Protocol 4 Shuffle $\pi_{\text{Shuffle}}(n)$

Inputs: Each party $\mathcal{P}_i$ has $y_i \in G$.

Outputs: $\prod_{i=1}^{n} y_i \in G$.

1. Each party $\mathcal{P}_i$ randomly chooses $x_{i,j} \in G$ for all $1 \leq j \leq n$ such that $\prod_{j=1}^{n} x_{i,j} = 1$ (i.e., randomly chooses $x_{i,j}$ for $1 \leq j \leq n-1$ and $x_{i,n}^{-1} := \prod_{j=1}^{n-1} x_{i,j}$). Set $y_{i,1} := x_{i,1} \cdot y_i$ and $y_{i,j} := x_{i,j}$ for all $2 \leq j \leq n$. Send $y_{i,j}$ to the party $\mathcal{P}_j$ for all $1 \leq j \neq i \leq n$.
2. Each party $\mathcal{P}_i$ computes $z_i := \prod_{j=1}^{n} y_{j,i}$. Broadcast z_i to the other party $\mathcal{P}_j$.
3. Outputs $z := \prod_{i=1}^{n} z_i$.

If $G = \left\{ a + b\sqrt{D} \mid a, b \in \mathbb{Z}_N, a^2 - b^2 D \in \mathbb{Z}_N^{\times} \right\}$, the parties choose $x_{i,j}$ by randomly selecting $v_{i,j}, w_{i,j} \in \mathbb{Z}_N$ such that $v_{i,j}^2 - w_{i,j}^2 D \in \mathbb{Z}_N^{\times}$ and setting $x_{i,j} := (v_{i,j} + w_{i,j}\sqrt{D})$. In our setting, all inputs are norm 1 (i.e., $y_i = \alpha_i \beta_i^{-1}$) elements of G.

6.10 Three RSA Moduli Protocols in the Semi-honest Model

In this section, we rewrite the Lucas test using macros from [15] to facilitate comparison with the Boneh-Franklin test [10] and Burkhardt's et al.'s [15] protocols. Here, we always assume $p \equiv q \equiv 3 \pmod 4$. Finally, we note that an RSA modulus refers to N, which is the product of two distinct prime numbers. In contrast, a biprime refers to N being the product of any two prime numbers.

Protocol 5 Lucas Biprimality test type (n)

Inputs: Each party $\mathcal{P}_i$ has odd integers $[p]_{\mathbb{Z}}$, $[q]_{\mathbb{Z}}$, $D = 1$, and N.

Outputs:

1. Party $\mathcal{P}_1$ randomly chooses $0 \leq P < N$ such that $Q = (P^2 - D)/4$ and $\left[\frac{Q}{N}\right] = 1$. Send this P to the other parties.
2. Party $\mathcal{P}_1$ computes $v_1 := g^{(N-\mathfrak{p}_1-\mathfrak{q}_1+1)/4} \pmod N$, where $g := \frac{P-\sqrt{D}}{P+\sqrt{D}}$. The other parties compute $v_i := g^{-(\mathfrak{p}_i+\mathfrak{q}_i)/4} \pmod N$. Parties broadcast v_i to compute $v := \prod_{i=1}^{n} v_i \pmod N$. They then check if

$$v = \prod_{i=1}^{n} v_i \equiv 1 \pmod N.$$

   If the test fails, return to `NonBlumInteger`.
3. Parties verify $\gcd(N, e) = 1$ as follows:
   3.1 obtain $[r]_N \leftarrow$ `RandomSample`$(\mathbb{Z}_N)$.
   3.2 compute $[p]_N \leftarrow$ `Int-to-mod`$(\mathbb{Z}_N, [p]_{\mathbb{Z}})$ and $[q]_N \leftarrow$ `Int-to-mod`$(\mathbb{Z}_N, [q]_{\mathbb{Z}})$.
   3.3 call $[b]_N \leftarrow$ `Mult`$(\mathbb{Z}_N, [r]_N, [p]_N + [q]_N - 1)$.
   3.4 obtain $b \leftarrow$ `OpenAll`$(\mathbb{Z}_N, [b]_N)$. If $b \neq 1$ then output `NonBlumInteger`. Otherwise, output `BlumInteger`.

Below is Boneh-Franklin protocol [10], as cited from the version in [15, FIGURE 7.1].

Protocol 6 Boneh-Franklin biprimality protocol(n)

Inputs: Each party has $[p]_{\mathbb{Z}}$, $[q]_{\mathbb{Z}}$ and N.

Outputs:

1. Party $\mathcal{P}_1$ randomly chooses $g \in \mathbb{Z}_N^{\times}$ and $\left[\frac{g}{N}\right] = 1$. Send this g to the other parties.
2. Party $\mathcal{P}_1$ computes $v_1 := g^{(N-\mathfrak{p}_1-\mathfrak{q}_1+1)/4} \pmod N$. The other parties compute $v_i := g^{-(\mathfrak{p}_i+\mathfrak{q}_i)/4} \pmod N$. Parties broadcast v_i to compute $v := \prod_{i=1}^{n} v_i \pmod N$. They then check if

$$v = \prod_{i=1}^{n} v_i \equiv \pm 1 \pmod N.$$

   If the test fails, return to `NonBlumInteger`.
3. Parties verify $\gcd(N, e) = 1$ as follows:
   3.1 obtain $[r]_N \leftarrow$ `RandomSample`$(\mathbb{Z}_N)$.

3.2 compute $[p]_N \leftarrow$ `Int-to-mod`$(\mathbb{Z}_N, [p]_{\mathbb{Z}})$ and $[q]_N \leftarrow$ `Int-to-mod`$(\mathbb{Z}_N, [q]_{\mathbb{Z}})$.
3.3 call $[b]_N \leftarrow$ `Mult`$(\mathbb{Z}_N, [r]_N, [p]_N + [q]_N - 1)$.
3.4 obtain $b \leftarrow$ `OpenAll`$(\mathbb{Z}_N, [b]_N)$. If $b \neq 1$ then output `NonBlumInteger`. Otherwise, output `BlumInteger`.

Herein lies Burkhardt's protocol. For further details, please consult [15].

Protocol 7 Miller-Rabin biprimality protocol$(\kappa_{\mathrm{lenP}}, s, n)$

Inputs: Each party has $[p]_{\mathbb{Z}}$, $[q]_{\mathbb{Z}}$, P, Q and N. Here P and Q are primes satisfying $n^2 2^{2\kappa_P} < nP < Q$.

Outputs:

1. Let $G = \emptyset$, for $f \in \{p, q\}$:
   1.1 $\mathcal{P}_n$ uniformly samples $v \in \mathbb{Z}_N$ and broadcasts v.
   1.2 Compute $<\gamma>_N$ as follows: Party $\mathcal{P}_1$ sets $\gamma_1 := v^{\frac{f_1-1}{2}} \pmod N$. For $2 \leq i \leq n$, $\mathcal{P}_i$ sets $\gamma_i := v^{\frac{f_i}{2}} \pmod N$.
   1.3 Obtain $[\gamma]_N \leftarrow$ `Mul-to-Add`$(\mathbb{Z}_N, <\gamma>_N)$.
   1.4 Compute $[\gamma + 1]_N$ and $[\gamma - 1]_N$.
   1.5 For $\delta \in \{\gamma + 1, \gamma - 1\}$, compute

$$[y_\delta]_Q \leftarrow \texttt{Divisible}(\kappa_{\mathrm{lenP}}, s, \mathbb{Z}_P, \mathbb{Z}_Q, [\delta]_N, [f]_{\mathbb{Z}}).$$

   1.6 Compute $[y]_Q \leftarrow$ `Mult`$(\mathbb{Z}_Q, [y_{\gamma+1}]_Q, [y_{\gamma-1}]_Q)$.
   1.7 Reveal $y \leftarrow$ `OpenAll`$(\mathbb{Z}_Q, [y]_Q)$.
   1.8 If $y = 0$, set $G = G \cup \{f\}$.
2. If $G = \{p, q\}$ output `BlumInteger`, otherwise output `NonBlumInteger`.

The number of macros used in each test are summarized below.

Table 5: The number of macros in biprimality tests.

	# `Random -sample`	♯ `Int-to -mod`	♯ `Mult`	♯ `OpenAll`	♯ `Mult-to -add`
Boneh-Franklin [23]	1	2	1	1	0
Millier-Rabin [15]	≥ 2	4	≥ 6	≥ 4	2
Type-(I)	1	2	1	1	0

In addition to the aforementioned, Burkhardt's protocol includes other macros such as **Invert** and **Larger-domain**.

6.11 Adaptation of Chen's Biprime Test Protocol Based on Section 4.4

In this section, we present a modification of the protocol by Chen et al. [16, Protocol 5.2], building upon the enhancements detailed in Section 4.4. For ease of comparison, the notation employed in this section adheres to that of [16].

Protocol 8 Adaptation of Chen's Biprime Test $\pi_{\text{Biprime}}(M, n)$

This protocol is parametrized by an integer M and the number of parties n. In addition, there is a statistical parameter s. The parties have access to the $\mathcal{F}_{\text{CT}}$, $\mathcal{F}_{\text{Com}}$, $\mathcal{F}_{\text{ComCompute}}$, and $\mathcal{F}_{\text{Zero}}$ functionalities.

Input Commitment:

1. Upon receiving input $(\texttt{check-biprimality}, \mathsf{sid}, N, p_i, q_i)$ from the environment, each party $\mathcal{P}_i$ for $i \in [n]$ samples $\boldsymbol{\tau}_{i,j} \leftarrow \mathbb{Z}_{\lceil 1.475s \rceil n 2^{s+1}(n^2 2^{2\kappa+s-3}+M)}$ for $j \in [\lceil 1.475s \rceil]$ and commits to these values, along with its shares of p and q, by sending $(\texttt{commit}, \mathsf{GenSID}(\mathsf{sid}, i), (p_i, q_i, \boldsymbol{\tau}_{i,*})$ to $\mathcal{F}_{\text{ComCompute}}(n)$.

Boneh-Franklin Test:

2. Each party $\mathcal{P}_i$ for $i \in [n]$ sends $(\texttt{sample}, \mathsf{sid})$ to $\mathcal{F}_{\text{Zero}}(n, n2^{2\kappa+s-3})$ and receives $(\texttt{zero-share}, \mathsf{sid}, r_i)$ in response.
3. For $j \in [\lceil 1.475s \rceil]$, the parties invoke $\mathcal{F}_{\text{CT}}(n, \mathbb{J}_N)$, where $\mathbb{J}_N$ is the subdomain of $\mathbb{Z}_N^*$ that contains only values with Jacobi symbol 1. The parties define vector $\boldsymbol{\gamma}$ that contains the $\lceil 1.475s \rceil$ sampled values.
4. For every $j \in [\lceil 1.475s \rceil]$, party $\mathcal{P}_1$ computes

$$\boldsymbol{\chi}_{1,j} := \boldsymbol{\gamma}_j^{r_1-(p_1+q_1-6)/4} \pmod N$$

and every other party $\mathcal{P}_i$ for $i \in [2, n]$ computes

$$\boldsymbol{\chi}_{i,j} := \boldsymbol{\gamma}_j^{r_i-(p_i+q_i)/4} \pmod N.$$

5. Every $\mathcal{P}_i$ for $i \in [n]$ sends $(\texttt{commit}, \mathsf{GenSID}(\mathsf{sid}, i), \boldsymbol{\chi}_{i,*}, [n])$ to $\mathcal{F}_{\text{Com}}(n)$.
6. After being notified that all other parties are committed, each party $\mathcal{P}_i$ for $i \in [n]$ sends $(\texttt{decommit}, \mathsf{GenSID}(\mathsf{sid}, i))$ to $\mathcal{F}_{\text{Com}}(n)$, and in response receives $\boldsymbol{\chi}_{i',*}$ from $\mathcal{F}_{\text{Com}}(n)$ for $i' \in [n] \setminus \{i\}$.
7. The parties output $(\texttt{not-biprime}, \mathsf{sid})$ to the environment and halt if there exists $j \in [\lceil 1.475s \rceil]$ such that

$$\boldsymbol{\gamma}_j^{(N-5)/4} \cdot \prod_{i \in [n]} \boldsymbol{\chi}_{i,j} \not\equiv \pm 1 \pmod N.$$

Consistency Check and GCD Test:

8. For $j \in [\lceil 1.475s \rceil]$, each party $\mathcal{P}_i$ for $i \in [n]$ computes

$$\boldsymbol{\alpha}_{i,j} := \boldsymbol{\gamma}_j^{\boldsymbol{\tau}_{i,j}} \pmod N.$$

The parties all broadcast the values they have computed to one another.

9. The parties all send $(\texttt{flip}, \mathsf{sid})$ to $\mathcal{F}_{\mathrm{CT}}(n, \{0,1\}^{\lceil 1.475s \rceil})$ to obtain an agreed-upon random bit vector $\boldsymbol{c}$ of length $\lceil 1.475s \rceil$.
10. For $j \in [\lceil 1.475s \rceil]$, party $\mathcal{P}_1$ computes

$$\boldsymbol{\zeta}_{1,j} := \boldsymbol{\tau}_{1,j} + \boldsymbol{c}_j \cdot \big(r_i - (p_1 + q_1 - 6)/4\big),$$

and every other party $\mathcal{P}_i$ for $i \in [2, n]$ computes

$$\boldsymbol{\zeta}_{i,j} := \boldsymbol{\tau}_{i,j} + \boldsymbol{c}_j \cdot \big(r_i - (p_i + q_i)/4\big).$$

They all broadcast the values they have computed to one another.
11. The parties halt and output $(\texttt{not-biprime}, \mathsf{sid})$ if there exists any $j \in [\lceil 1.475s \rceil]$ such that

$$\prod_{i \in [n]} \boldsymbol{\gamma}_j^{\boldsymbol{\zeta}_{i,j}} \not\equiv \prod_{i \in [n]} \boldsymbol{\alpha}_{i,j} \cdot \boldsymbol{\chi}_{i,j}^{\boldsymbol{c}_j} \pmod{N}.$$

12. Let C be a circuit computing $\mathsf{VerifyBiprime}(N, M, \boldsymbol{c}, \{\cdot, \cdot, \cdot, \boldsymbol{\zeta}_{i,*}\}_{i \in [n]})$; that is, let it be a circuit representation of Algorithm $\mathsf{VerifyBiprime}$ with the public values $N, M, \boldsymbol{c}$, and $\boldsymbol{\zeta}$ hardcoded. The parties send $(\texttt{compute}, \mathsf{sid}, \{\mathsf{GenSID}(\mathsf{sid}, i)\}_{i \in [n]}, C)$ to $\mathcal{F}_{\mathrm{ComCompute}}(n)$, and in response they all receive $(\texttt{result}, \mathsf{sid}, z)$. If $z = \perp$, or if $\mathcal{F}_{\mathrm{ComCompute}}(n)$ aborts, then the parties halt and output $(\texttt{not-biprime}, \mathsf{sid})$.
13. The parties halt and output $(\texttt{biprime}, \mathsf{sid})$ to the environment if $\gcd(z, N) = 1$, or halt and output $(\texttt{not-biprime}, \mathsf{sid})$ otherwise.

Algorithm 0 $\mathsf{VerifyBiprime}(N, M, \boldsymbol{c}, \{(p_i, q_i, \boldsymbol{\tau}_{i,*}, \boldsymbol{\zeta}_{i,*})\}_{i \in [n]})$

---

1. Sample $r \leftarrow \mathbb{Z}_N$ and compute

$$z = r \cdot \left(-1 + \sum_{i \in [n]} (p_i + q_i)\right) \pmod{N}$$

2. Return z if and only if it holds that

$$\begin{aligned} N &= \sum_{i \in [n]} p_i \cdot \sum_{i \in [n]} q_i \\ &\wedge \sum_{i \in [n]} p_i \equiv \sum_{i \in [n]} q_i \equiv 3 \pmod 4 \\ &\wedge 0 \le p_i < M \wedge 0 \le q_i < M \text{ for all } i \in [n] \\ &\wedge \boldsymbol{\tau}_{1,j} = \boldsymbol{\zeta}_{1,j} + \boldsymbol{c}_j \cdot (p_1 + q_1 - 6)/4 \text{ for all } j \in [\lceil 1.475s \rceil] \\ &\wedge \boldsymbol{\tau}_{i,j} = \boldsymbol{\zeta}_{i,j} + \boldsymbol{c}_j \cdot (p_i + q_i)/4 \text{ for all } i \in [2, n] \text{ and } j \in [\lceil 1.475s \rceil] \end{aligned}$$

If any part of the above predicate does not hold, output $\perp$.
