Rational Modular Encoding in the DCR Setting: Non-Interactive Range Proofs and Paillier-Based Naor-Yung in the Standard Model

Julien Devevey[1], Benoît Libert[2,1], and Thomas Peters[3]

[1] ENS de Lyon, Laboratoire LIP (U. Lyon, CNRS, ENSL, Inria, UCBL), France
[2] CNRS, Laboratoire LIP, France
[3] FNRS and UCLouvain, ICTEAM, Belgium

Abstract. Range proofs allow a sender to convince a verifier that committed integers belong to an interval without revealing anything else. So far, all known non-interactive range proofs in the standard model rely on groups endowed with a bilinear map. Moreover, they either require the group order to be larger than the range of any proven statement or they suffer from a wasteful rate. Recently (Eurocrypt'21), Couteau *et al.* introduced a new approach to efficiently prove range membership by encoding integers as a modular ratio between small integers. We show that their technique can be transposed in the standard model under the Composite Residuosity (DCR) assumption. Interestingly, with this modification, the size of ranges is not a priori restricted by the common reference string. It also gives a constant ratio between the size of ranges and proofs. Moreover, we show that their technique of encoding messages as bounded rationals provides a secure standard model instantiation of the Naor-Yung CCA2 encryption paradigm under the DCR assumption.

Keywords. Range proofs, NIZK, standard model, Naor-Yung.

1 Introduction

Zero-knowledge proofs [36] make it possible for a prover to convince a verifier about the truth of a statement while revealing nothing else. Since their introduction, they have been used in countless cryptographic protocols to protect users' privacy or to hedge against malicious adversaries. In many situations, it is desirable to have non-interactive zero-knowledge (NIZK) proofs comprised of a single message from the prover to the verifier. In the non-interactive setting, NIZK proofs necessarily rely on a common reference string generated by some trusted party. While the Fiat-Shamir paradigm [32] allows for non-interactive proofs without a trusted setup in the random oracle model, it is known to only provide heuristic arguments in terms of security.

In the standard model, NIZK proofs are known to exist for all NP languages under well-studied assumptions [7,6,40,56]. For specific languages, however, much more efficient constructions are often possible, by dispensing with the need for an expensive Karp reduction.

Efficient NIZK constructions exist in the context of range proofs [11], where a prover convinces a verifier that a committed value belongs to a specific interval. Range proofs served as a building block of a number of cryptographic protocols, including anonymous credentials or e-cash [15], auction protocols [51], e-voting [38], and many more. Recently, they also served as crucial components of cryptocurrencies [53,12], where transaction amounts are private and only appear in committed [53] or encrypted [12] form. Range proofs then come into play to ensure that the committed/encrypted value lives in the correct range instead of being, e.g., slightly larger than the order of the message space.

A widely used approach [10,49,38] proceeds by committing to integers [35,28], rather than finite field elements. By withholding the order $|\mathcal{M}|$ of the message space, it forces the prover argue over the integers in order to demonstrate that a committed integer fits in a range $[0, B]$, where $B \in \mathbb{Z}$ may be larger than $|\mathcal{M}|$.

Recently, Couteau *et al.* [24] suggested an elegant technique that surprisingly emulates the properties of integer commitments in the discrete logarithm setting over public-order groups. The core idea of their construction is to view each Pedersen commitment [55] $C = g^m \cdot h^r$ as committing to the rounded rational $\lfloor x/c \rceil \in \mathbb{Z}$, where x and c are small-magnitude integers $x, c \in \mathbb{Z}$ such that $m = x \cdot c^{-1} \bmod q$, where q is the group order. This approach yields instantiations in class groups and under lattice assumptions. In the discrete-log setting, it outperforms the BulletProof technique [13] for a wide range of parameters. It also enables either computational or statistical soundness (whereas integers commitments only offer computational soundness).

In this paper, we consider their approach in the Composite Residuosity setting [54], where we highlight several advantages when proving range membership of Paillier-encrypted values.

1.1 Our Contribution

Range Proofs. We provide the first *unbounded* non-interactive range proof with constant rate in the standard model. The rate is defined in the standard way, as the ratio between the length of the witness and the total length of commitments and proofs. By "unbounded", we mean that a fixed-size common reference string makes it possible to commit to arbitrarily large integers.[4] In the standard model, it is also the first non-interactive candidate that does not rely on pairing-friendly groups. Instead, we can prove security under the standard Composite Residuosity (DCR) and Learning-with-Errors (LWE) [57] assumptions. While our construction provides statistical soundness (and computational zero-knowledge), it can be turned into a dual-mode NIZK system – where soundness/zero-knowledge can be either statistical or computational depending on the configuration of the CRS – at the cost of sacrificing unboundedness.

In either case, we obtain space-efficient proofs consisting of a constant number of Damgård-Jurik [29] ciphertexts. Asymptotically, the communication cost

[4] It is tempting to believe that Groth-Sahai proofs achieve unboundedness. In the full version of this paper, we explain why it is not the case.

is dominated by $O(\lambda^{3-O(1)} + \log B)$ bits, where B is the range size, which is on par with constructions based on integer commitments [49,38,25] in the random oracle model. In comparison, standard-model solutions based on Groth-Sahai proofs [41] cost $O(\lambda \cdot \log B)$ per proof.

Our unbounded range proof makes it possible to prove that a Paillier ciphertext decrypts to a modular ratio $M = x \cdot c^{-1} \bmod N^{\zeta}$, for some $\zeta \in \mathbb{N}$ and bounded integers $x, c \in \mathbb{Z}$ such that $\lfloor x/c \rceil \in \mathbb{Z}$ belongs to a range $[0, B]$. As a second contribution, we show that this encoding technique can be used to instantiate the Naor-Yung CCA2-secure encryption paradigm [52].

DCR-based Instantiation of Naor-Yung in the Standard Model. We give a Σ-protocol proving plaintext equalities between Paillier ciphertexts encrypted under distinct moduli, which restores the soundness of a Σ-protocol used by Fouque and Pointcheval [33]. Recently, Devevey *et al.* [30, Appendix E] showed that the Σ-protocol of [33, Section 4.2] does not provide soundness as a cheating prover can exploit the distinct moduli to prove false statements. This invalidates the proof[5] that the DCR-based threshold cryptosystem of [33] provides IND-CCA2 security in the random oracle model. Devevey *et al.* [30] suggested to fix the problem by additionally proving that the plaintext is smaller than both Paillier moduli. While efficient range proofs (e.g., [38,25,13]) can solve this problem in the random oracle model, we do not know how to instantiate them in the standard model via the Fiat-Shamir paradigm. To achieve standard-model security by exploiting correlation-intractable hash functions as in [19,56], we show that no range proof is actually necessary if the decryption algorithm is modified and "undoes" the rational modular encoding of Couteau *et al.* [24].

We show that the modified decryption algorithm can be combined with the correlation-intractable hash functions of [19,56] so as to instantiate the scheme in the standard model. As a result, we obtain a new construction of a non-interactive threshold CCA2-secure cryptosystem without pairings. Devevey *et al.* [30] recently proposed such a construction under the DCR and LWE assumptions. Our scheme provides several advantages over their construction. It notably inherits a property of the Damgård-Jurik system [29], which makes it possible to encrypt very long messages[6] for a fixed size public key comprised of an RSA modulus N. Variable-length plaintexts can even be encrypted by flexibly choosing an integer $\zeta > 1$, depending on the message length, and working over $\mathbb{Z}^*_{N^{\zeta+1}}$. In the threshold setting, the key generation phase requires to set a bound on the maximal value of ζ. However, this constraint disappears in the centralized (i.e., non-threshold) case, where we can CCA2-encrypt variable-length messages using a fixed-size public key without using hybrid encryption. To our knowledge,

[5] We are not aware of any effective attack. Only the proof of IND-CCA2 security in the ROM is affected.

[6] A common approach to encrypt long messages is to use hybrid encryption. However, it makes it harder to prove properties about encrypted data in zero-knowledge. It also destroys the additive homomorphic homomorphic properties that we retain when we discard ciphertext components that ensure chosen-ciphertext security. The latter property is useful in the context of voting protocols [5].

this useful property of the Damgård-Jurik cryptosystem was never preserved in the chosen-ciphertext setting (at least in the standard model).

We believe that, even in the random oracle model, properly instantiating Naor-Yung under the DCR assumption is important. For example, it provides a convenient way to encrypt arbitrarily long messages with a fixed-size public key while preserving the possibility of efficiently proving properties (e.g., range membership) about encrypted data, which would be difficult using hybrid encryption. It also provides a "voting-friendly" encryption scheme – in the terminology of [5] – in the sense that the keys/ciphertexts of the threshold CCA2-secure system can be publicly mapped to the keys/ciphertexts of an embedded additively homomorphic encryption scheme.

1.2 Technical Overview

Our range proofs depart from all known standard-model candidates [14,58], which are based on Groth-Sahai proofs [41] and proceed by breaking the committed integers into bits. To our knowledge, this approach either restricts committed integers to be smaller than the group order, or they are inherently stuck with a somewhat wasteful rate $O(1/\lambda)$ caused by bit-by-bit comparisons (as discussed in the full version of this paper). In the discrete-log setting, the construction of Couteau *et al.* [24] also requires the group order to be sufficiently larger than the maximal magnitude of committed integers.

To avoid this a priori bound on the range of committed values, we leverage a property of the Damgård-Jurik cryptosystem in that the CRS only consists of an RSA modulus $N = pq$. The prover commits to an integer in a range $[0, B]$ by having the prover first choose a sufficiently large $\zeta \geq 1$ such that $B < N^{\zeta}$ exactly as in the Damgård-Jurik encryption scheme. Following the approach of Kiayias *et al.* [44], we can obtain a constant rate as the ratio between the size of the proof and that of witnesses becomes constant (actually, less than 20) for a large $\zeta \in \mathsf{poly}(\lambda)$. Unlike our main construction, our dual-mode variant requires a CRS that fixes an integer $\zeta \geq 1$ once-and-for-all.

In order to prove security in the standard model, we build on recent progress on instantiations of the Fiat-Shamir paradigm. Canetti *et al.* [16] and Peikert and Shiehian [56] showed that Fiat-Shamir can provide soundness in the standard model under the Learning-With-Errors (LWE) assumption [57], which yields *correlation intractable* (CI) hash functions [17] for efficiently searchable relations.

Correlation intractability for a relation R requires the infeasibility of finding x such that $(x, H_k(x)) \in R$ given a random hashing key k. It guarantees soundness by preventing a cheating prover's first message a from being hashed into a challenge $\mathsf{Chall} = H_k(a)$ admitting a valid response z. Canetti *et al.* [19] showed that CI hash functions for efficiently searchable relations suffice when Fiat-Shamir is applied to *trapdoor Σ-protocols*. These are Σ-protocols that assume a CRS and where an efficiently computable function $\mathsf{BadChallenge}$ can identify (on input of a trapdoor τ_{Σ}, the false statement x and the prover's first message a) the only challenge Chall such that an accepting transcript (a, Chall, z)

exists for some z. Libert *et al.* [47] (based on earlier observations from [21,50]) showed that the group structure of Paillier allows $\mathsf{BadChallenge}$ to identify bad challenges within an exponentially large challenge space, thus eliminating the need for parallel repetitions to ensure soundness.

Here, we also achieve soundness without parallel repetitions by exploiting the group structure of $\mathbb{Z}^*_{N^{\zeta+1}}$. However, our $\mathsf{BadChallenge}$ functions additionally solve integer linear programming instances with a constant number of variables. They also apply the technique of Fouque, Stern and Wackers [34], which decodes Paillier-decrypted values into rational numbers. In our variant of Couteau *et al.*'s range proof [24], the prover first sends DCR-based commitments to integers $\{x_i\}_{i=0}^3$ such that $1 + 4x_0(B - x_0) = \sum_{i=1}^3 x_i^2$ over $\mathbb{Z}$ (recall that, for any positive integer y, there exist $\{x_i \in \mathbb{Z}\}_{i=1}^3$ such that $1 + 4y = \sum_{i=1}^3 x_i^2$, as observed in [38]). Our $\mathsf{BadChallenge}$ function first computes $\{\tilde{x}_i\}_{i=0}^3$ by decrypting Paillier ciphertexts. Following Fouque *et al.* [34], it then runs Gauss' algorithm to compute pairs $(x_i, c_i) \in [-B^*, B^*] \times [0, C]$ such that $\tilde{x}_i = x_i \cdot c_i^{-1} \bmod N^\zeta$ for each i. If no such decomposition exists for a given index $i \in [0, 3]$, the corresponding $\tilde{x}_i$ determines the only bad challenge that can admit a valid response element z_i. We show that this bad challenge is computable by solving an integer linear programming instance $\mathbf{A} \cdot \mathbf{t} \leq \mathbf{b}$ with 3 variables and 8 constraints. By the definition of the language, we know that the solution $\mathbf{t}$ is unique if the statement is false. Moreover, Lenstra's algorithm [45] allows computing it in polynomial time as the number of variables is fixed.

If all decrypted elements $\{\tilde{x}_i\}_{i=0}^3$ can be represented as pairs of integers $(x_i, c_i) \in [-B^*, B^*] \times [0, C]$ such that $\tilde{x}_i = x_i \cdot c_i^{-1} \bmod N^\zeta$, our $\mathsf{BadChallenge}$ function determines if such representations exist for a common denominator $c = c_i$ for each i. If not all $\tilde{x}_i$ have such a representation with $x_i \in [-B^*, B^*]$, then we know that no response elements $\{z_i\}_{i=0}^3$ will simultaneously satisfy all verification equations for the *same* challenge. In this case, the language definition implies that at most one challenge can satisfy all these verification equations and we can identify this bad challenge by solving an integer linear program with 9 variables. In the last case, the prover's first message commitments decrypt to elements $\{\tilde{x}_i \in \mathbb{Z}_{N^\zeta}\}_{i=0}^3$ that all admit a representation $(x'_i, c) \in [-B^*, B^*] \times [0, C]$ such that $\tilde{x}_i = x'_i \cdot c^{-1} \bmod N^\zeta$. In this case, if the statement is false, the unique bad challenge is determined by the last verification equation and it is computable by solving a simple modular equation.

Our Paillier-based instantiation of Naor-Yung uses exactly the same Σ-protocol as in [33, Section 4.2]. We prove that its soundness is restored if we introduce a post-processing step in the (distributed) decryption mechanism. Each decryption server computes its partial decryption exactly as in the threshold variant of Damgård-Jurik [29] (as in [33], this is done without interaction among servers). When partial decryptions are combined together, we first compute a Paillier/Damgård-Jurik plaintext $M \in \mathbb{Z}_{N^\zeta}$. Using Gauss' algorithm as suggested by Fouque *et al.* [34], we then decode M as a modular ratio $M = x \cdot c^{-1} \bmod N^\zeta$ for small-magnitude $x, c \in \mathbb{Z}$ before outputting the rounded rational $\lfloor x/c \rceil \in \mathbb{Z}$ as a plaintext. We show that this modified decryption algorithm can be safely combined with the Σ-protocol in [33] as it ensures that both Paillier ciphertexts lead to the same plaintext $\lfloor x/c \rceil \in \mathbb{Z}$. In the case $\zeta = 1$, given two Paillier ciphertexts $\mathsf{ct}_1 = (1+N_1)^{\mathsf{Msg}} \cdot r_1^{N_1} \bmod N_1^2$ and $\mathsf{ct}_2 = (1+N_2)^{\mathsf{Msg}} \cdot r_2^{N_2} \bmod N_2^2$, the protocol of [33] guarantees the existence of $\bar{c} \in [0, C]$ and $\bar{m} \in [-R, R]$ such that $\mathsf{ct}_1^{\bar{c}} = (1+N_1)^m \cdot w_1^{N_1} \bmod N_1^2$ and $\mathsf{ct}_2^{\bar{c}} = (1+N_2)^m \cdot w_2^{N_2} \bmod N_2^2$, for some $w_1 \in \mathbb{Z}_{N_1}^*$, $w_2 \in \mathbb{Z}_{N_2}^*$. While there is no guarantee that $m \cdot \bar{c}^{-1} \bmod N_1$ equals $m \cdot \bar{c}^{-1} \bmod N_2$, we know from [34] that they both decode to the same pair $(m, \bar{c}) \in [-R, R] \times [0, C]$ as long as $2RC < N$ when we run Gauss' algorithm. This ensures plaintext equality when the decryption algorithm outputs $\lfloor m/\bar{c} \rceil$.

In order to obtain a trapdoor Σ-protocol, our $\mathsf{BadChallenge}$ function appeals again to Lenstra's algorithm and solves an integer linear programming instance with a constant number of variables/constraints. When it comes to proving CCA2-security in the standard model, we need to turn the Σ-protocol into a one-time simulation-sound[7] NIZK proof system [59]. For this purpose, we could use a construction put forth by Devevey *et al.* [30] but it would unfortunately ruin the length-flexible property of the scheme. If we were to combine it with our trapdoor Σ-protocol showing plaintext equalities, the public key would inherently bound the size of the message space. To avoid this problem, we build a new DCR-based construction that compiles any trapdoor Σ-protocol into a one-time simulation-sound NIZK argument. Unlike the solution of [30, Section 3], simulation-soundness is achieved by augmenting the CRS with a number of bits that does not depend on the underlying trapdoor Σ-protocol.

1.3 Related Work

Range proofs were introduced by Brickell *et al.* [11] and receive continuous attention [22,14,10,49,39,20,25,37] since then. So far, known solutions have been following two main approaches.

The first approach proceeds by breaking integers into bits or small digits [11,3,29,14,39,37,13], which allows communicating a logarithmic (in the range size) number of group elements [14,39,37,13]. This technique is usually implemented using homomorphic commitment schemes over groups of public prime order, while the optimized versions of [14,39,37] require pairings. Within this line of work, Bulletproof [13] obtains the best communication complexity via a clever recursive proof technique and can be realized over standard (i.e., non-pairing-friendly) discrete-logarithm-hard groups. Unfortunately, it is not known to be instantiable in the standard model without interaction.

The second approach [10,49,38,25] relies on integer commitments over groups of hidden order. This approach is often preferred for very large ranges (which arise in applications like anonymous credentials [15], where range elements may be comprised of thousands of bits) where it tends to be more efficient. Also, it does not require the maximal range length to be known ahead of time, when the

[7] In short, one-time simulation-soundness means that seeing a simulated proof for a false statement of its choice does not help the adversary prove a new false statement.

commitment key is set up. Using homomorphic integer commitments, any range $[\alpha, \beta]$ can be proven by exploiting the homomorphic properties of the commitment scheme and demonstrating that $X - \alpha \in [0, \beta - \alpha]$. Indeed, working over the integers allows showing that $X - \alpha$ and $\beta - X$ are both positive by expressing them as a sum of squares. The idea to rely on square decompositions over the integers dates back to [11]. The square decomposition method was improved by Lipmaa [49] by relying on the Lagrange decomposition of any positive integer as a sum of four squares. Groth [38] observed any positive integer of the form $4Y + 1$, for some $Y \in \mathbb{Z}$, can be more efficiently expressed as a sum of three squares. Further efficiency and security improvements were given in [25]. In this second approach, the underlying integer commitment scheme builds on [35,28] and is usually instantiated using RSA groups. Couteau *et al.* [25] showed that its security relates to a slight variant of the RSA assumption rather than the less standard Strong RSA assumption.

Very recently, Couteau *et al.* [24] managed to reconcile the advantages of both approaches. Their core technique converts any (homomorphic) commitment scheme over groups of (public) prime order into a *bounded* integer commitment scheme. While the conversion does not completely preserve the homomorphic property, it allows committing to bounded-range integers by interpreting them as rounded rationals. It also allows reviving the square decomposition method so as to prove integer relations holding over public ranges. As a result, their range proof consists of a public-coin 3-move interactive protocol that only communicates a constant number of elements. It can be instantiated using standard Pedersen commitments [55] in prime-order groups as long as the group order is large enough to represent the bounded integers. Their technique also applies under lattice assumptions and in class groups. In the latter instantiation, it also inherits the unbounded property of solutions based on hidden-order groups.

We note that a generic transformation due to Ciampi *et al.* [23, Section 4.2] can be used to turn a slight modification (where the first-message group elements are not hashed) of Couteau *et al.*'s discrete-log-based range proof [24] into a trapdoor Σ-protocol, and thus obtain a non-interactive variant in the standard model. However, since the transformation of [23] only applies to Σ-protocols with small challenge space, it has to be repeated $O(\lambda)$ times in parallel to achieve negligible soundness error. In contrast, we achieve soundness without parallel repetitions as in [47]. Moreover, applying [23] to build a non-interactive variant of [24] would still require to fix the maximal cardinality of ranges ahead of time. As it turns out, none of the existing range proofs (even in the bounded case where the CRS depends on $\log(\beta-\alpha)$) in the standard model features proofs comprised of a constant number of element of the base ring/group.

The first non-interactive CCA-secure threshold cryptosystems date back to the work of Shoup and Gennaro [60] who gave DDH-based realizations in the random oracle model. Fouque and Pointcheval [33] gave a generic construction and a DDH-based instantiation using the Naor-Yung paradigm. Until the recent years, all non-interactive solutions in the standard model were pairing-based [8,48]. Boneh *et al.* gave a generic technique [9] to transform any IND-CCA secure encryption scheme into a non-interactive threshold system using fully homomorphic encryption. Using correlation-intractable hash functions, Devevey *et al.* [30] recently obtained constructions under the DCR and LWE assumptions in the adaptive corruption setting. Back in 1999, Canetti and Goldwasser [18] showed that chosen-ciphertext security was achievable in the standard model by allowing decryption servers to interact with one another. Their approach was subsequently extended to handle adaptive adversaries [43,1].

2 Background

Let S be a finite set. Then, $a \hookleftarrow U(S)$ means that a is sampled according to the uniform distribution over S. $|a|$ is the bit-length of a.

2.1 Hardness Assumptions

We first recall Paillier's Composite Residuosity assumption and its variant considered by Damgård and Jurik.

Definition 2.1 ([54,29]). *Let integers $N = pq$ and $s > 1$ for primes p, q. The s-**Decision Composite Residuosity** (s-DCR) assumption states that the distributions $\{x = w^{N^s} \bmod N^{s+1} \mid w \hookleftarrow U(\mathbb{Z}_N^\star)\}$ and $\{x \mid x \hookleftarrow U(\mathbb{Z}_{N^{s+1}}^\star)\}$ are computationally indistinguishable.*

Lemma 2.2 (Adapted from [29]). *Let $s = \mathsf{poly}(\lambda)$. Then s-DCR is equivalent to $1-\mathsf{DCR}$, with a security loss $\leq s$.* (The proof is straightforward.)

2.2 Correlation Intractable Hash Functions

We consider unique-output efficiently searchable relations [16].

Definition 2.3. *A relation $R \subseteq \mathcal{X} \times \mathcal{Y}$ is* **searchable** *in time T if there exists a function $f : \mathcal{X} \to \mathcal{Y}$ which is computable in time T and such that, if there exists y such that $(x, y) \in R$, then $f(x) = y$.*

Let $\lambda \in \mathbb{N}$ a security parameter. A hash family with input length $n(\lambda)$ and output length $m(\lambda)$ is a collection $\mathcal{H} = \{h_\lambda : \{0,1\}^{s(\lambda)} \times \{0,1\}^{n(\lambda)} \to \{0,1\}^{m(\lambda)}\}$ of keyed functions induced by efficient algorithms $(\mathsf{Gen}, \mathsf{Hash})$, where $\mathsf{Gen}(1^\lambda)$ outputs a key $k \in \{0,1\}^{s(\lambda)}$ and $\mathsf{Hash}(k, x)$ computes $h_\lambda(k, x) \in \{0,1\}^{m(\lambda)}$.

Definition 2.4. *For a relation ensemble $\{R_\lambda \subseteq \{0,1\}^{n(\lambda)} \times \{0,1\}^{m(\lambda)}\}$, a hash function family $\mathcal{H} = \{h_\lambda : \{0,1\}^{s(\lambda)} \times \{0,1\}^{n(\lambda)} \to \{0,1\}^{m(\lambda)}\}$ is R-**correlation intractable** if, for any probabilistic polynomial time (PPT) adversary $\mathbb{A}$, we have* $\Pr\left[k \leftarrow \mathsf{Gen}(1^\lambda)),\ x \leftarrow \mathcal{A}(k) : (x, h_\lambda(k, x)) \in R\right] = \mathsf{negl}(\lambda)$.

Peikert and Shiehian [56] described a correlation-intractable hash family for any searchable relation (in the sense of Definition 2.3) defined by functions f of bounded depth.

2.3 Trapdoor Σ-protocols

Canetti *et al.* [19] considered a definition of Σ-protocols that slightly differs from the usual formulation [26].

Definition 2.5 (Adapted from [19,2]). *Let a language* $\mathcal{L} = (\mathcal{L}_{\mathsf{zk}}, \mathcal{L}_{\mathsf{sound}})$ *associated with two NP relations* $\mathcal{R}_{\mathsf{zk}}, \mathcal{R}_{\mathsf{sound}}$*. A 3-move interactive proof system* $\Pi = (\mathsf{Gen}_{\mathsf{par}}, \mathsf{Gen}_{\mathcal{L}}, \mathsf{P}, \mathsf{V})$ *in the common reference string model is a Gap* Σ*-protocol for* $\mathcal{L}$ *if it satisfies the following conditions:*

- **3-Move Form:** P *and* V *both take as input* $\mathsf{crs} = (\mathsf{par}, \mathsf{crs}_{\mathcal{L}})$*, with* $\mathsf{par} \leftarrow \mathsf{Gen}_{\mathsf{par}}(1^\lambda)$ *and* $\mathsf{crs}_{\mathcal{L}} \leftarrow \mathsf{Gen}_{\mathcal{L}}(\mathsf{par}, \mathcal{L})$*, and a statement* x *and proceed as follows: (i)* P *takes in* $w \in \mathcal{R}_{\mathsf{zk}}(x)$*, computes* $(\mathbf{a}, st) \leftarrow \mathsf{P}(\mathsf{crs}, x, w)$ *and sends* $\mathbf{a}$ *to the verifier; (ii)* V *sends back a random challenge* Chall *from the challenge space* $\mathcal{C}$*; (iii)* P *finally sends a response* $\mathbf{z} = \mathsf{P}(\mathsf{crs}, x, w, \mathbf{a}, \mathsf{Chall}, st)$ *to* V*; (iv) On input of* $(\mathbf{a}, \mathsf{Chall}, \mathbf{z})$*,* V *outputs* 1 *or* 0*.*
- **Completeness:** *If* $(x, w) \in \mathcal{R}_{\mathsf{zk}}$ *and* P *honestly computes* $(\mathbf{a}, \mathbf{z})$ *for a challenge* Chall*,* $\mathsf{V}(\mathsf{crs}, x, (\mathbf{a}, \mathsf{Chall}, \mathbf{z}))$ *outputs* 1 *with probability* $1 - \mathsf{negl}(\lambda)$*.*
- **Special zero-knowledge:** *There is a PPT simulator* ZKSim *that inputs* crs*,* $x \in \mathcal{L}_{\mathsf{zk}}$ *and a challenge* $\mathsf{Chall} \in \mathcal{C}$*. It outputs* $(\mathbf{a}, \mathbf{z}) \leftarrow \mathsf{ZKSim}(\mathsf{crs}, x, \mathsf{Chall})$ *such that* $(\mathbf{a}, \mathsf{Chall}, \mathbf{z})$ *is computationally indistinguishable from a real transcript with challenge* Chall *(for* $w \in \mathcal{R}_{zk}(x)$*).*
- **Special soundness:** *For any CRS* $\mathsf{crs} = (\mathsf{par}, \mathsf{crs}_{\mathcal{L}})$ *obtained as* $\mathsf{par} \leftarrow \mathsf{Gen}_{\mathsf{par}}(1^\lambda)$*,* $\mathsf{crs}_{\mathcal{L}} \leftarrow \mathsf{Gen}_{\mathcal{L}}(\mathsf{par}, \mathcal{L})$*, any* $x \notin \mathcal{L}_{\mathsf{sound}}$*, and any first message* $\mathbf{a}$ *sent by* P*, there is at most one challenge* $\mathsf{Chall} = f(\mathsf{crs}, x, \mathbf{a})$ *for which an accepting transcript* $(\mathsf{crs}, x, \mathbf{a}, \mathsf{Chall}, \mathbf{z})$ *exists for some third message* $\mathbf{z}$*. The function* f *is called the "bad challenge function" of* Π*. That is, if* $x \notin \mathcal{L}_{\mathsf{sound}}$ *and the challenge differs from the bad challenge, the verifier never accepts.*

Definition 2.5 is taken from [19] and relaxes the standard special soundness property in that extractability is not required. Instead, it considers a bad challenge function f, which may not be efficiently computable. Canetti *et al.* [19] define *trapdoor* Σ-protocols as Σ-protocols where the bad challenge function is efficiently computable using a trapdoor. Here, we use a definition where the CRS and the trapdoor may depend on the language.

The common reference string $\mathsf{crs} = (\mathsf{par}, \mathsf{crs}_{\mathcal{L}})$ consists of a fixed part par and a language-dependent part $\mathsf{crs}_{\mathcal{L}}$ which is generated as a function of par and a language parameter $\mathcal{L} = (\mathcal{L}_{\mathsf{zk}}, \mathcal{L}_{\mathsf{sound}})$.

Definition 2.6 (Adapted from [19]). *A* Σ*-protocol* $\Pi = (\mathsf{Gen}_{\mathsf{par}}, \mathsf{Gen}_{\mathcal{L}}, \mathsf{P}, \mathsf{V})$ *with bad challenge function* f *for a trapdoor language* $\mathcal{L} = (\mathcal{L}_{\mathsf{zk}}, \mathcal{L}_{\mathsf{sound}})$ *is a* **trapdoor Σ-protocol** *if it satisfies the properties of Definition 2.5 and there exist PPT algorithms* $(\mathsf{TrapGen}, \mathsf{BadChallenge})$ *with the following properties.*

- $\mathsf{Gen}_{\mathsf{par}}$ *inputs* $\lambda \in \mathbb{N}$ *and outputs public parameters* $\mathsf{par} \leftarrow \mathsf{Gen}_{\mathsf{par}}(1^\lambda)$*.*
- $\mathsf{Gen}_{\mathcal{L}}$ *is a randomized algorithm that, on input of public parameters* par*, outputs the language-dependent part* $\mathsf{crs}_{\mathcal{L}} \leftarrow \mathsf{Gen}_{\mathcal{L}}(\mathsf{par}, \mathcal{L})$ *of* $\mathsf{crs} = (\mathsf{par}, \mathsf{crs}_{\mathcal{L}})$*.*

- $\mathsf{TrapGen}(\mathsf{par}, \mathcal{L}, \tau_{\mathcal{L}})$ *takes as input public parameters* par *and a membership-testing trapdoor* $\tau_{\mathcal{L}}$ *for the language* $\mathcal{L}_{\mathsf{sound}}$*. It outputs a common reference string* $\mathsf{crs}_{\mathcal{L}}$ *and a trapdoor* $\tau_{\Sigma} \in \{0,1\}^{\ell_\tau}$*, for some* $\ell_\tau(\lambda)$*.*
- $\mathsf{BadChallenge}(\tau_{\Sigma}, \mathsf{crs}, x, \mathbf{a})$ *takes in a trapdoor* τ_{Σ}*, a CRS* $\mathsf{crs} = (\mathsf{par}, \mathsf{crs}_{\mathcal{L}})$*, an instance* x*, and a first prover message* $\mathbf{a}$*. It outputs a challenge* Chall*.*

In addition, the following properties are required.

- **CRS indistinguishability:** *For any* $\mathsf{par} \leftarrow \mathsf{Gen}_{\mathsf{par}}(1^\lambda)$*, and any trapdoor* $\tau_{\mathcal{L}}$ *for the language* $\mathcal{L}$*, an honestly generated* $\mathsf{crs}_{\mathcal{L}}$ *is computationally indistinguishable from a CRS produced by* $\mathsf{TrapGen}(\mathsf{par}, \mathcal{L}, \tau_{\mathcal{L}})$*. Namely, for any* aux *and any PPT distinguisher* $\mathcal{A}$*, we have*

$$\begin{aligned}\mathbf{Adv}^{\text{indist-}\Sigma}_{\mathcal{A}}(\lambda) := |\Pr[&\mathsf{crs}_{\mathcal{L}} \leftarrow \mathsf{Gen}_{\mathcal{L}}(\mathsf{par}, \mathcal{L}) : \mathcal{A}(\mathsf{par}, \mathsf{crs}_{\mathcal{L}}) = 1] \\ &- \Pr[(\mathsf{crs}_{\mathcal{L}}, \tau_{\Sigma}) \leftarrow \mathsf{TrapGen}(\mathsf{par}, \mathcal{L}, \tau_{\mathcal{L}}) : \mathcal{A}(\mathsf{par}, \mathsf{crs}_{\mathcal{L}}) = 1]| \leq \mathsf{negl}(\lambda).\end{aligned}$$

- **Correctness:** *There exists a language-specific trapdoor* $\tau_{\mathcal{L}}$ *such that, for any instance* $x \notin \mathcal{L}_{\mathsf{sound}}$ *and all pairs* $(\mathsf{crs}_{\mathcal{L}}, \tau_{\Sigma}) \leftarrow \mathsf{TrapGen}(\mathsf{par}, \mathcal{L}, \tau_{\mathcal{L}})$*, we have* $\mathsf{BadChallenge}(\tau_{\Sigma}, \mathsf{crs}, x, \mathbf{a}) = f(\mathsf{crs}, x, \mathbf{a})$.

Note that the $\mathsf{TrapGen}$ algorithm does not take a specific statement x as input, but only a trapdoor $\tau_{\mathcal{L}}$ allowing to recognize elements of $\mathcal{L}_{\mathsf{sound}}$.

2.4 Trapdoor *Σ*-Protocol Showing Composite Residuosity

We recall a standard Σ-protocol that allows proving that an element of $\mathbb{Z}^*_{N^{\zeta+1}}$ is a N^ζ-th residue. In [47], it was shown that the latter protocol is a trapdoor Σ-protocol showing that an element of $\mathbb{Z}^*_{N^2}$ is a composite residue.

Namely, let $\mathcal{L}^{\mathsf{DCR}} := \{x \in \mathbb{Z}^*_{N^{\zeta+1}} \mid \exists w \in \mathbb{Z}^\star_N : x = w^{N^\zeta} \bmod N^{\zeta+1}\}$, the language of N^ζ-th residues, for some integer $\zeta > 1$, where $N = pq$ is an RSA modulus. We assume that the challenge space is $\{0, \ldots, 2^\lambda - 1\}$ and that $p, q > 2^{l(\lambda)}$, for some polynomial $l : \mathbb{N} \to \mathbb{N}$ such that $l(\lambda) > \lambda$ for any sufficiently large $\lambda \in \mathbb{N}$. The condition $p, q > 2^\lambda$ will ensure that the difference between any two challenges be co-prime with N.

In order to obtain a $\mathsf{BadChallenge}$ function that identifies bad challenges for elements $x \notin \mathcal{L}^{\mathsf{DCR}}$, [47] uses an observation from Lipmaa [50], which shows that the factorization of N allows computing bad challenges even if $\gcd(x, N) > 1$.

$\mathsf{Gen}_{\mathsf{par}}(1^\lambda)$: Given the security parameter λ, define $\mathsf{par} = \{\lambda\}$.

$\mathsf{Gen}_{\mathcal{L}}(\mathsf{par}, \mathcal{L}^{\mathsf{DCR}})$: Given public parameters par and the description of a language $\mathcal{L}^{\mathsf{DCR}}$, consisting of an RSA modulus $N = pq$ with primes p and q such that $p, q > 2^{l(\lambda)}$, for some polynomial $l : \mathbb{N} \to \mathbb{N}$ such that $l(\lambda) > \lambda$, define the language-dependent $\mathsf{crs}_{\mathcal{L}} = \{N\}$. The global CRS is $\mathsf{crs} = (\{\lambda\}, \mathsf{crs}_{\mathcal{L}})$.

$\mathsf{TrapGen}(\mathsf{par}, \mathcal{L}^{\mathsf{DCR}}, \tau_{\mathcal{L}})$: Given par, the description of a language $\mathcal{L}^{\mathsf{DCR}}$ that specifies an RSA modulus N and a membership-testing trapdoor $\tau_{\mathcal{L}} = (p, q)$ consisting of the factorization of $N = pq$, output the language-dependent $\mathsf{crs}_{\mathcal{L}} = \{N\}$ which defines $\mathsf{crs} = (\{\lambda\}, \mathsf{crs}_{\mathcal{L}})$ and the trapdoor $\tau_{\Sigma} = (p, q)$.

$\mathsf{P}\big(\mathsf{crs}, x, w\big) \leftrightarrow \mathsf{V}(\mathsf{crs}, x)$**:** Given a crs, a statement $x = w^{N^\zeta} \bmod N^{\zeta+1}$, P (who has the witness $w \in \mathbb{Z}_N^\star$) and V interact as follows:

1. P chooses a random $r \hookleftarrow U(\mathbb{Z}_N^*)$ and sends $a = r^{N^\zeta} \bmod N^{\zeta+1}$ to V.
2. V sends a random challenge $\mathsf{Chall} \hookleftarrow U(\{0, \ldots, 2^\lambda - 1\})$ to P.
3. P computes the response $z = r \cdot w^{\mathsf{Chall}} \bmod N$ and sends it to V.
4. V checks if $a \cdot x^{\mathsf{Chall}} \equiv z^{N^\zeta} \pmod{N^{\zeta+1}}$ and returns 0 otherwise.

$\mathsf{BadChallenge}\big(\mathsf{par}, \tau_\Sigma, \mathsf{crs}, x, a\big)$**:** Given $\tau_\Sigma = (p, q)$, (Damgård-Jurik) decrypt x and a to obtain $\alpha_x = \mathcal{D}_{\tau_\Sigma}(x) \in \mathbb{Z}_{N^\zeta}$, $\alpha_a = \mathcal{D}_{\tau_\Sigma}(a) \in \mathbb{Z}_{N^\zeta}$.

1. If $\alpha_a = 0$, return $\mathsf{Chall} = 0$.
2. If $\alpha_a \neq 0$, let $d_x = \gcd(\alpha_x, N^\zeta)$, which lives in the set $\{p^i q^j \mid 0 \leq i < \zeta,\ 0 \leq j < \zeta\} \cup \{p^i q^\zeta \mid 0 \leq i < \zeta\} \cup \{p^\zeta q^j \mid 0 \leq j < \zeta\}$. Then,
   a. If $1 < d_x < N^\zeta$, return $\perp$ if d_x does not divide $N^\zeta - \alpha_a$.
   b. Otherwise, the congruence $\alpha_a + \mathsf{Chall} \cdot \alpha_x \equiv 0 \pmod{\frac{N^\zeta}{d_x}}$ has a unique solution $\mathsf{Chall}' = -\alpha_x^{-1} \cdot \alpha_a \in \mathbb{Z}_{N^\zeta/d_x}$ since $\gcd(\alpha_x, N^\zeta/d_x) = 1$. If $\mathsf{Chall}' \in \mathbb{Z}_{N^\zeta/d_x} \setminus \{0, \ldots, 2^\lambda - 1\}$, return $\perp$. Else, return $\mathsf{Chall} = \mathsf{Chall}'$.

In [47], it is shown that the above construction is a trapdoor Σ-protocol with large challenge space. By applying [56], this implies compact NIZK arguments (i.e., without using parallel repetitions to achieve negligible soundness error) for the language $\mathcal{L}^{\mathsf{DCR}}$ assuming that the LWE assumption holds.

Lemma 2.7 ([47]). *The above protocol is a trapdoor Σ-protocol for $\mathcal{L}^{\mathsf{DCR}}$.*

2.5 Encoding and Decoding Bounded Rationals in $\mathbb{Z}_N$

In [34], Fouque *et al.* suggested a technique that allows computing over rational numbers when they are encrypted using Paillier. The idea is to encode a rational r/s, for co-prime integers $(r, s) \in [-R, R] \times [0, S]$, as the modular ratio $r \cdot s^{-1} \bmod N$. They showed that, as long as, $2RS < N$, it is possible to recover (r, s) from $t = r \cdot s^{-1} \bmod N$ using Gauss' lattice reduction algorithm in dimension 2.

Let an RSA modulus and bounds R, S. Let $r, s \in \mathbb{Z}$ such that $-R \leq r \leq R$, $0 < s \leq S$, $\gcd(r, s) = 1$ and $\gcd(s, N) = 1$. Let the rational $t = r/s \in \mathbb{Q}$

Define the encoding $\mathcal{E}(t) := t' = r \cdot s^{-1} \bmod N$. To decode it and recover $t \in \mathbb{Q}$ from t', consider the lattice

$$\Lambda := \{(x, y) \in \mathbb{Z}^2 : x = y \cdot t' \bmod N\} = \{(x, y) \in \mathbb{Z}^2 : s \cdot x = y \cdot r \bmod N\}.$$

A particular basis of Λ is formed by the vectors $(N, 0)$ and $(t', 1)$. Since s is invertible over $\mathbb{Z}_N$, the vector $(r, s) \in \mathbb{Z}^2$ also lives in Λ. To recover co-prime integers $(r, s) \in \mathbb{Z}^2$ such that $t' = r \cdot s^{-1} \bmod N$, one can run Gauss' algorithm on input of the initial basis $\vec{u} = (N, 0)$, $\vec{v} = (t', 1)$ to compute a minimal vector of Λ. A result of Vallée [61] ensures that the number of iterations is at most $3 + \log_{1+\sqrt{2}} \max(\|\vec{u}\|, \|\vec{v}\|)$ in the worst case.

Fouque *et al.* proved that the decoding procedure is correct and pointed out that it carries over when computations take place modulo N^ζ for $\zeta > 1$.

Lemma 2.8 ([34, Theorem 1]). *If $t' = r \cdot s^{-1} \bmod N$, $-R \leq r \leq R$, and $0 < s \leq S$, then Gauss' algorithm uniquely recovers r and s if $2RS < N$.*

2.6 Paillier Decryption of (Rounded) Rationals

We first describe a variant of Paillier's cryptosysem used by Fouque, Stern and Wackers [34] to perform homomorphic operations over rational numbers. While the encryption algorithm is identical to that of Paillier/Damgård-Jurik [54,29], the message space is restricted to a specific interval and the decryption algorithm runs Gauss' lattice reduction algorithm in dimension 2. In fact, we modify the decryption algorithm of [34] to make sure that it outputs an integer instead of a rational. In addition, we follow a suggestion of Damgård and Jurik [29] and assume that the message space is *not* a priori bounded by the public key. Instead, it can be flexibly adjusted by the encryption algorithm.

In the following, we let $\ell_M \in \mathsf{poly}(\lambda)$ denote the message length, which can be dynamically determined at encryption time. We also denote by $\mathrm{abs} : \mathbb{Z} \to \mathbb{N}$ the absolute value function defined as $\mathrm{abs}(x) = x \cdot (x \geq 0) + (-x) \cdot (x < 0)$. Letting $C = 2^\lambda - 1$, the encryptor will fix $R > 2^\lambda \cdot (M + 1)$, where $M = 2^{\ell_M} - 1$ is the largest possible message, and choose ζ in such a way that $2RC < N^\zeta$. After having obtained $\widetilde{\mathsf{Msg}} \in \mathbb{Z}_{N^\zeta}$ from the decryption algorithm of Damgård-Jurik, the receiver will be able to apply Lemma 2.8 so as to decode $\widetilde{\mathsf{Msg}}$ as the ratio $m \cdot c^{-1} \bmod N^\zeta$ between bounded rationals $-R \leq m \leq R$ and $1 \leq c \leq C$.

Keygen(1^λ) : Given a security parameter, choose an RSA modulus $N = pq$ such that $p, q > 2^{l(\lambda)}$, for some polynomial $l : \mathbb{N} \to \mathbb{N}$ with $l(\lambda) \geq \lambda$, and an integer $\zeta \geq 1$. The public key is $\mathsf{pk} = N$ and the secret key is $\mathsf{sk} = (p, q)$.

Encrypt$\big(\mathsf{pk}, \mathsf{Msg}\big)$: To encrypt $\mathsf{Msg} \in \{0,1\}^{\ell_M}$, interpret it as a positive integer in $[0, M]$, where $M = 2^{\ell_M} - 1$. Set $\zeta > 1$ as a small integer such that $N^\zeta \geq 2^{2\lambda+1} M$. Then, choose $r \hookleftarrow U(\mathbb{Z}_N^*)$ and compute

$$(\mathsf{ct}, \ell_M) = \Big((1+N)^{\mathsf{Msg}} \cdot r^{N^\zeta} \bmod N^{\zeta+1}, \ell_M \Big) .$$

Decrypt$(\mathsf{sk}, (\mathsf{ct}, \ell_M))$: Given $(\mathsf{ct}, \ell_M) \in \mathbb{Z}^*_{N^{\zeta+1}} \times \mathbb{N}$ and $\mathsf{sk} = (p, q)$. Compute $\widetilde{\mathsf{Msg}} \in \mathbb{Z}_{N^\zeta}$ by running the Damgård-Jurik decryption algorithm, denoted $\mathcal{D}_{\mathsf{sk}}(\mathsf{ct})$. Then, using Gauss' algorithm, find the unique $(m, c) \in \mathbb{Z}^2$ such that $-R \leq m \leq R$, $1 \leq c \leq C$ and $\widetilde{\mathsf{Msg}} = m \cdot c^{-1} \bmod N^\zeta$. If no such pair exists, return $\perp$. Otherwise, return $\mathsf{Msg} = \mathrm{abs}(\lfloor m/c \rceil)$, where $m/c \in \mathbb{Q}$.

In the decryption algorithm, the absolute value is used to enforce positiveness. The scheme is identical to [34], except that it outputs a positive integer rather than a rational. This decoding method will be applied in our instantiation of Naor-Yung. In our non-interactive range proof of Section 3, we will also use the scheme as a perfectly binding extractable commitment with an extraction algorithm $\mathsf{Decrypt}'$ where $\mathsf{Msg}' = \lfloor m/c \rceil$, without absolute values.

3 Constant-Rate Unbounded Non-Interactive Range Proofs in the Standard Model

This section presents a range proof where a fixed-size common reference string containing an RSA modulus $N = pq$ allows committing to arbitrarily large integers. We note that, after having committed to an integer, the committer is bound to a specific modulus $N^{\zeta+1}$ and all subsequent proofs related to this commitment are restricted to ranges smaller than a certain bound. Still, the CRS and the underlying algebraic structure do not have to be scaled with the size of the committed integers.

Let positive integers B, $C = 2^{\lambda} - 1$, $B^* = 2^{\lambda} BC$ and $\zeta \geq 1$ satisfying the conditions $2^{2\lambda+3} B^2 C^2 < N^{\zeta}$. Let $\mathcal{L}_{\mathsf{range}}^{B,B^*,C} = (\mathcal{L}_{\mathsf{zk}}^{B}, \mathcal{L}_{\mathsf{sound}}^{B,B^*,C})$ be

$$\mathcal{L}_{\mathsf{zk}}^{B} := \Big\{ \mathsf{ct} \in \mathbb{Z}_{N^{\zeta+1}}^{*} \mid \exists x \in [0, B],\ w \in \mathbb{Z}_N^{\star}\ :\ \mathsf{ct} = (1+N)^{x} \cdot w^{N^{\zeta}} \bmod N^{\zeta+1} \Big\}$$

$$\begin{aligned} \mathcal{L}_{\mathsf{sound}}^{B,B^*,C} := \Big\{ & \mathsf{ct} \in \mathbb{Z}_{N^{\zeta+1}}^{*} \mid \exists x \in [0, B^*],\ c \in [1, C],\ w \in \mathbb{Z}_N^{\star}\ : \\ & \mathsf{ct} = (1+N)^{x \cdot c^{-1} \bmod N^{\zeta}} \cdot w^{N^{\zeta}} \bmod N^{\zeta+1}\ \wedge\ \lfloor x/c \rceil \in [0, B] \Big\}. \end{aligned}$$

To prove membership, we will have the prover generate auxiliary commitments $\{C_i\}_{i=1}^{3}$ and rely on $\bar{\mathcal{L}}_{\mathsf{range}}^{B,B^*,C} = (\bar{\mathcal{L}}_{\mathsf{zk}}^{B}, \bar{\mathcal{L}}_{\mathsf{sound}}^{B,B^*,C})$ such that

$$\begin{aligned} \bar{\mathcal{L}}_{\mathsf{zk}}^{B} := \Big\{ & (\mathsf{ct}, \{C_i\}_{i=1}^{3}) \in (\mathbb{Z}_{N^{\zeta+1}}^{*})^4\ \mid\ \exists x_0, x_1, x_2, x_3 \in [0, B], \\ & \exists\ s_0, s_1, s_2, s_3 \in \mathbb{Z}_N^{*}\ :\ 1 + 4(B - x_0)x_0 = x_1^2 + x_2^2 + x_3^2 \\ & \wedge\ (1+N)^{B} \cdot \mathsf{ct}^{-1} = (1+N)^{x_0} \cdot s_0^{N^{\zeta}} \bmod N^{\zeta+1} \\ & \wedge\ C_i = (1+N)^{x_i} \cdot s_i^{N^{\zeta}} \bmod N^{\zeta+1} \quad \forall i \in [3] \quad \Big\}, \end{aligned}$$

$$\begin{aligned} \bar{\mathcal{L}}_{\mathsf{sound}}^{B,B^*,C} := \Big\{ & (\mathsf{ct}, \{C_i\}_{i=1}^{3}) \in (\mathbb{Z}_{N^{\zeta+1}}^{*})^4\ \mid\ \exists x_0, x_1, x_2, x_3 \in [-B^*, B^*], \\ & \exists\ s_0, s_1, s_2, s_3, \tau \in \mathbb{Z}_N^{*},\ c \in [1, C]\ : \\ & \wedge\ \big((1+N)^{B} \cdot \mathsf{ct}^{-1}\big)^{c} = (1+N)^{x_0} \cdot s_0^{N^{\zeta}} \bmod N^{\zeta+1} \\ & \wedge\ C_i^{c} = (1+N)^{x_i} \cdot s_i^{N^{\zeta}} \bmod N^{\zeta+1} \quad \forall i \in [3] \\ & \wedge\ (1+N)^{c} = \prod_{i=1}^{3} C_i^{x_i} \cdot \mathsf{ct}^{-4x_0} \cdot \tau^{N^{\zeta}} \bmod N^{\zeta+1} \Big\}, \end{aligned} \tag{1}$$

In Lemma 3.2, we show that $(\mathsf{ct}, \{C_i\}_{i=1}^{3}) \in \bar{\mathcal{L}}_{\mathsf{sound}}^{B,B^*,C}$ implies $\mathsf{ct} \in \mathcal{L}_{\mathsf{sound}}^{B,B^*,C}$, which in turn implies $\mathsf{Decrypt}'(\mathsf{sk}, (\mathsf{ct}, |B^*|)) \in [0, B]$, where $\mathsf{sk} = (p, q)$ and $N = pq$.

Gen$_{\mathsf{par}}(1^{\lambda})$: Given the security parameter λ, define $\mathsf{par} = \{\lambda\}$.

Gen$_{\mathcal{L}}(\mathsf{par}, \mathcal{L}_{\mathsf{range}}^{B,B^*,C})$: Given public parameters par as well as a description of a language pair $\mathcal{L}_{\mathsf{range}}^{B,B^*,C}$, consisting of an RSA modulus $N = pq$ with primes $p, q > 2^{l(\lambda)}$, for some polynomial $l : \mathbb{N} \to \mathbb{N}$ such that $l(\lambda) > \lambda$, define the language-dependent CRS $\mathsf{crs}_{\mathcal{L}} = \{N\}$. The global CRS is $\mathsf{crs} = (\{\lambda\}, \mathsf{crs}_{\mathcal{L}})$.

TrapGen$(\mathsf{par}, \mathcal{L}_{\mathsf{range}}^{B,B^*,C}, \tau_{\mathcal{L}})$: This algorithm is identical to $\mathsf{Gen}_{\mathcal{L}}(\mathsf{par}, \mathcal{L}_{\mathsf{range}}^{B,B^*,C})$, except that it also outputs the trapdoor $\tau_{\Sigma} = (p, q)$.

$\mathbf{P}\big(\mathsf{crs}, \vec{x}, \vec{w}\big) \leftrightarrow \mathbf{V}(\mathsf{crs}, \vec{x})$: On input of a CRS crs, a statement $\mathsf{ct} \in \mathcal{L}_{\mathsf{zk}}^{B}$, the prover P (who has $\vec{w} = (x, w) \in [0, B] \times \mathbb{Z}_N^{\star}$) and V interact as follows:

1. P computes $x_1, x_2, x_3 \in [0, B+1]$ such that $1 + 4x(B-x) = \sum_{i=1}^{3} x_i^2$ over $\mathbb{Z}$. Then, P sets $C_0 = (1+N)^B \cdot \mathsf{ct}^{-1} \bmod N^{\zeta+1}$, $x_0 = B - x$ and $s_0 = w^{-1} \bmod N$. It randomly picks $s_1, s_2, s_3 \hookleftarrow U(\mathbb{Z}_{N^\zeta}^*)$ and computes
$$C_i = (1+N)^{x_i} \cdot s_i^{N^\zeta} \qquad \forall i \in [3].$$
Next, to show that $(\mathsf{ct}, \{C_i\}_{i=1}^3) \in \bar{\mathcal{L}}_{\mathsf{zk}}^{B}$, it chooses $\sigma \hookleftarrow U(\mathbb{Z}_N^*)$, $r_i \hookleftarrow U([0, B^*])$ and $\alpha_i \hookleftarrow U(\mathbb{Z}_{N^\zeta}^*)$ for each $i \in [0, 3]$, to compute
$$R_i = (1+N)^{r_i} \cdot \alpha_i^{N^\zeta} \bmod N^{\zeta+1} \qquad \forall i \in [0,3]$$
$$R = \sigma^{N^\zeta} \cdot C^{4 \cdot r_0} \cdot \prod_{i=1}^{3} C_i^{-r_i} \bmod N^{\zeta+1}.$$
and send $(R, \{R_i\}_{i=0}^3, \{C_i\}_{i=1}^3)$ to V.
2. V sends a random challenge $\mathsf{Chall} \hookleftarrow U(\{0, \ldots, 2^\lambda - 1\})$ to P.
3. P computes the response
$$\tau = \sigma \cdot \big(s_0^{4 \cdot x_0} \cdot \prod_{i=1}^{3} s_i^{x_i}\big)^{\mathsf{Chall}} \bmod N$$
$$z_i = r_i + \mathsf{Chall} \cdot x_i, \qquad t_i = \alpha_i \cdot s_i^{\mathsf{Chall}} \bmod N \qquad \forall i \in [0,3]$$
and fails if there exists $i \in [0,3]$ such that $z_i \notin [0, B^*]$. Otherwise, it sends $(\tau, \{(z_i, t_i)\}_{i=0}^3)$ to V.
4. V sets $C_0 = (1+N)^B \cdot \mathsf{ct}^{-1} \bmod N^{\zeta+1}$. It accepts iff $z_i \in [0, B^*]$ for each $i \in [0,3]$ and the following equations hold:
$$R_i \equiv (1+N)^{z_i} \cdot t_i^{N^\zeta} \cdot C_i^{-\mathsf{Chall}} \pmod{N^{\zeta+1}} \qquad \forall i \in [0,3],$$
$$R \equiv \prod_{i=1}^{3} C_i^{-z_i} \cdot \mathsf{ct}^{4 \cdot z_0} \cdot \tau^{N^\zeta} \cdot (1+N)^{\mathsf{Chall}} \pmod{N^{\zeta+1}}. \tag{2}$$

$\mathsf{BadChallenge}\big(\mathsf{par}, \tau_\Sigma, \mathsf{crs}, \mathbf{x}, \mathbf{a}\big)$: Given the statement $\mathbf{x} = \mathsf{ct} \in \mathbb{Z}_{N^\zeta}$, the message $\mathbf{a} = (R, \{R_i\}_{i=0}^3, \{C_i\}_{i=1}^3)$ and the trapdoor $\tau_\Sigma = (p, q)$, return $\perp$ if $\mathsf{Decrypt}'_{\tau_\Sigma}(\mathsf{ct}) \in [0, B]$. Otherwise, do the following.

1. Let $C_0 = (1+N)^B \cdot \mathsf{ct}^{-1} \bmod N^{\zeta+1}$. For each index $i \in [0,3]$, compute $\tilde{x}_i = \mathcal{D}_{\tau_\Sigma}(C_i) \in \mathbb{Z}_{N^\zeta}$ using the Damgård-Jurik decryption algorithm. Also, compute $r = \mathcal{D}_{\tau_\Sigma}(R) \in \mathbb{Z}_{N^\zeta}$ and $r_i = \mathcal{D}_{\tau_\Sigma}(R_i) \in \mathbb{Z}_{N^\zeta}$ for each $i \in [0,3]$. Then, for each $i \in [0,3]$, run Gauss' algorithm to compute $x_i \in [-B^*, B^*]$ and $c_i \in [0, C]$ such that $\tilde{x}_i = x_i \cdot c_i^{-1} \bmod N^\zeta$.

2. If there exists $i \in [0,3]$ such that no pair $(x_i, c_i) \in [-B^*, B^*] \times [0, C]$ satisfies $\tilde{x}_i = x_i \cdot c_i^{-1} \bmod N^\zeta$, let $j \in [0,3]$ the smallest such index. Compute $(z_j, \mathsf{Chall}_j, k_j) \in \mathbb{Z}^3$ such that

$$\begin{aligned} r_j &= z_j - \tilde{x}_j \cdot \mathsf{Chall}_j + k_j \cdot N^\zeta \\ 0 &\le z_j \le B^* \\ 0 &\le \mathsf{Chall}_j \le 2^\lambda - 1 \\ 0 &\le k_j \le 2^\lambda \end{aligned} \tag{3}$$

This can be achieved by replacing the first equality by inequalities

$$z_j - \tilde{x}_j \cdot \mathsf{Chall}_j + k_j \cdot N^\zeta \le r_j, \qquad -z_j + \tilde{x}_j \cdot \mathsf{Chall}_j - k_j \cdot N^\zeta \le -r_j$$

and solving an integer linear programming instance with 8 constraints and 3 variables $(z_j, \mathsf{Chall}_j, k_j) \in \mathbb{Z}^3$ using Lenstra's algorithm [45]. If a solution is found (in which case, it is unique), return $\mathsf{Chall} = \mathsf{Chall}_j$.

3. For each $i \in [0,3]$, let $(x_i, c_i) \in [-B^*, B^*] \times [0, C]$ such that $\{\tilde{x}_i\}_{i=0}^3$ satisfy $\tilde{x}_i = x_i \cdot c_i^{-1} \bmod N^\zeta$. Then, let $c \triangleq \mathrm{lcm}(c_0, c_1, c_2, c_3)$. Check if $c \in [0, C]$ and there exist integers $x'_0, x'_1, x'_2, x'_3 \in [-B^*, B^*]$ such that $\tilde{x}_i = x'_i \cdot c^{-1} \bmod N^\zeta$ for each $i \in [0,3]$. If no such $\{x'_i\}_{i=0}^3$ and c exist, find the (unique) integer vector $(z_0, z_1, z_2, z_3, \mathsf{Chall}, k_0, k_1, k_2, k_3) \in \mathbb{Z}^9$ such that $0 \le \mathsf{Chall} \le 2^\lambda - 1$ and

$$\forall j \in [0,3] : \begin{cases} r_j = z_j - \tilde{x}_j \cdot \mathsf{Chall} + k_j \cdot N^\zeta \\ 0 \ \le z_j \le B^* \\ 0 \ \le k_j \le 2^\lambda \end{cases}$$

This is done by replacing equalities by pairs of inequalities and solving an integer linear programming instance with 9 variables and 26 constraints. If this vector exists, return the corresponding $\mathsf{Chall} \in [0, 2^\lambda - 1]$.

4. Let $c \in [0, C]$ and $\{x'_i \in [-B^*, B^*]\}_{i=0}^3$ such that $\tilde{x}_i = x'_i \cdot c^{-1} \bmod N^\zeta$. Let $d_x = \gcd(4\tilde{x}\tilde{x}_0 - \sum_{i=1}^3 \tilde{x}_i^2 + 1, N^\zeta)$, where $\tilde{x} = B - \tilde{x}_0 \bmod N^\zeta$, and compute

$$\mathsf{Chall}_0 \triangleq (r + \sum_{i=1}^{3} \tilde{x}_i \cdot r_i - 4\tilde{x} \cdot r_0) \cdot (4\tilde{x} \cdot \tilde{x}_0 - \sum_{i=1}^{3} \tilde{x}_i^2 + 1)^{-1} \bmod \frac{N^\zeta}{d_x}.$$

If $\mathsf{Chall}_0 \in \{0, \ldots, 2^\lambda - 1\}$, return $\mathsf{Chall} = \mathsf{Chall}_0$. Otherwise, return $\mathsf{Chall} = \perp$.

The $\mathsf{BadChallenge}$ function computes the bad challenge (which is unique when the statement is false) using Lenstra's algorithm [45] that runs in polynomial time since the number of variables is fixed. For an instance with t constraints, each of binary encoding length $O(s)$, the algorithm requires $O(st + s^2)$ arithmetic operations on s-bit numbers.

Completeness. As long as $z_i \in [0, B^*]$ for all $i \in [0,3]$ when P computes its response at step 3, i.e., P does not abort, we have

$$\prod_{i=1}^{3} C_i^{-z_i} \cdot \mathsf{ct}^{4\cdot z_0} \cdot \tau^{N^\zeta} \cdot (1+N)^{\mathsf{Chall}}$$

$$= \prod_{i=1}^{3} C_i^{-r_i} \cdot \mathsf{ct}^{4\cdot r_0} \cdot \big(\prod_{i=1}^{3}(1+N)^{x_i} s_i^{N^\zeta}\big)^{-x_i \mathsf{Chall}} \cdot \big((1+N)^{x} w^{N^\zeta}\big)^{4x_0 \mathsf{Chall}}$$

$$\cdot\, \sigma^{N^\zeta} \cdot \big(w^{-4\cdot x_0} \cdot \prod_{i=1}^{3} s_i^{x_i}\big)^{N^\zeta \cdot \mathsf{Chall}} \cdot (1+N)^{\mathsf{Chall}} \mod N^{\zeta+1}$$

$$= \prod_{i=1}^{3} C_i^{-r_i} \cdot \mathsf{ct}^{4\cdot r_0} \cdot (1+N)^{-\mathsf{Chall}\cdot \sum_{i=1}^{3} x_i^2} \cdot (1+N)^{4\cdot x_0 \cdot \mathsf{Chall}\cdot x}$$

$$\cdot\, \sigma^{N^\zeta} \cdot (1+N)^{\mathsf{Chall}} \mod N^{\zeta+1}$$

$$= \prod_{i=1}^{3} C_i^{-r_i} \cdot \mathsf{ct}^{4\cdot r_0} \cdot \sigma^{N^\zeta} \mod N^{\zeta+1} \;=\; R$$

$$(1+N)^{z_i} \cdot t_i^{N^\zeta} \equiv (1+N)^{r_i + \mathsf{Chall}\cdot x_i} \cdot \alpha_i^{N^\zeta} \cdot s_i^{\mathsf{Chall}\cdot N^\zeta} \equiv R_i \cdot C_i^{\mathsf{Chall}} \pmod{N^{\zeta+1}},$$

Finally, P only aborts with probability at most $4 \cdot 2^{-\lambda}$.

Special zero-knowledge. We first describe a simulator $\mathsf{ZKSim}_B^{\text{range}}$ before showing that a simulated transcript produced by $\mathsf{ZKSim}_B^{\text{range}}(\mathsf{crs}, \vec{x}, \mathsf{Chall})$ is *computationally* indistinguishable from a real transcript generated from a statement-witness pair $(\vec{x}, \vec{w}) \in \mathcal{R}_B^{\text{range}}$ when the challenge is Chall.

Given $\mathsf{crs} = (\{\lambda\}, \mathsf{crs}_{\mathcal{L}})$, an element $\vec{x} = \mathsf{ct} \in \mathbb{Z}_{N^{\zeta+1}}^*$ of the language $\mathcal{L}^{B,B^*,C}$ and a challenge $\mathsf{Chall} \in [0, C]$, $\mathsf{ZKSim}_B^{\text{range}}(\mathsf{crs}, \vec{x}, \mathsf{Chall})$ proceeds as follows: First, it sets $C_0 = (1+N)^B \cdot \mathsf{ct}^{-1} \mod N^{\zeta+1}$ and randomly picks $s_1, s_2, s_3 \hookleftarrow U(\mathbb{Z}_{N^\zeta}^*)$ in order to compute an encryption $C_i = s_i^{N^\zeta} \mod N^{\zeta+1}$ of 0 for each $i \in [3]$. Then, the simulator uniformly picks elements of the response $\mathbf{z}$ as $z_i \hookleftarrow [0, B^*]$, $t_i \hookleftarrow \mathbb{Z}_N^*$, for all $i \in [0,3]$, and $\tau \hookleftarrow \mathbb{Z}_N^*$. Finally, it computes the remaining components $(R, \{R_i\}_{i=0}^3)$ of the first prover message $\mathbf{a}$ in such a way that satisfy the verification equations (2).

We now prove the computational indistinguishability between the transcripts generated by $\mathsf{ZKSim}_B^{\text{range}}$ and real transcripts, which are faithfully computed from $\vec{w} \in \mathcal{R}_B^{\text{range}}(\vec{x})$. We first observe that a simulated transcript $(\mathbf{a}, \mathsf{Chall}, \mathbf{z})$ is computationally indistinguishable from an hybrid transcript where, instead of encrypting 0 in the computation of $\{C_i\}_{i=1}^3$, we encrypt $\{x_i\}_{i=1}^3$ such that $1 + 4x(B-x) = \sum_{i=1}^3 x_i^2$ and $x_0 = B - x$ over $\mathbb{Z}$, as in the real protocol. In this hybrid transcript, however, we still compute $(R, \{R_i\}_{i=0}^3)$ and $\mathbf{z}$ as in the simulation. A simple reduction shows that the probability to distinguish between simulated transcripts and hybrid transcripts is at most 3 times the advantage of an adversary against the semantic security of Damgård-Jurik (and thus the ζ-DCR assumption). Finally, we show that the distributions of hybrid and real transcripts for $(\vec{x}, \vec{w}) \in \mathcal{R}_B^{\text{range}}$ and the challenge Chall are statistically close (assuming that we use a deterministic algorithm to compute the Lagrange decomposition of $1 + 4x(B - x) \geq 0$) into the sum of *3* squares). This follows from standard arguments. By relying on the generalized Paillier isomorphism we can and split the analysis. Over the "randomness" modulo N, the distributions are the same because each (t_i, α_i) are in one-to-one relation for $i \in [0, 3]$, as well as (τ, σ). Since the x_i's are constant, the distributions "over the plaintext" modulo N^ζ are statistically close because the statistical distance between the z_i-variables is negligible.

More precisely, the ciphertexts $\{C_i\}_{i=0}^3$ have exactly the same distribution in the hybrid and the real transcripts. Now, let $\psi \colon \mathbb{Z}_{N^\zeta} \times \mathbb{Z}_N^\star \mapsto \mathbb{Z}_{N^{\zeta+1}}^\star$ denote the generalized Paillier isomorphism. Let also $(r_i, \alpha_i) := \psi^{-1}(R_i)$, for all $i \in [0, 4]$, and $(r, \alpha) := \psi^{-1}(R)$ of an hybrid transcript. We thus have, for all $i \in [0, 3]$,

$$r_i \equiv z_i - \mathsf{Chall} \cdot x_i \pmod{N^\zeta} \qquad \alpha_i \equiv t_i \cdot s_i^{-\mathsf{Chall}} \pmod{N},$$

where $x_0 = B - x \bmod N^\zeta$ and $s_0 = w^{-1} \bmod N$, as well as

$$r \equiv 4z_0(B - x_0) - \textstyle\sum_{i \in [3]} z_i x_i + \mathsf{Chall} \pmod{N^\zeta},$$

and $\alpha \equiv w^{4z_0} \cdot \prod_{i \in [3]} s_i^{-z_i} \cdot \tau \pmod{N}$. For α and $\{\alpha_i\}_{i \in [0,3]}$, The congruences in the multiplicative group $\mathbb{Z}_N^*$ show that, given w and $\{(z_i, s_i)\}_{i \in [0,3]}$, there is a one-to-one relation between α and τ, and between α_i and t_i, for all $i \in [0, 3]$. Then, their distributions are the same as those of the real distributions. (Note that α in the real distribution is also random due to σ.) We are thus left with analyzing the distributions over the additive group $\mathbb{Z}_{N^\zeta}$. For all $i \in [0, 3]$, the congruences on the r_i ensure that, unless $z_i \in [0, CB]$ (which occurs with negligible probability $2^{-\lambda}$), we have $0 \leq r_i = z_i - \mathsf{Chall} \cdot x_i \leq B^*$. That means that, over the integers, we have to show that the statistical distance between $U([0, B^*])$ (which is the distribution of the hybrid z_i) and $\mathsf{Chall} \cdot x_i + U([0, B^*])$ (which is the distribution of the real z) is negligible. Since $x_i \cdot \mathsf{Chall} \leq BC \leq 2^{-\lambda} B^*$, it is actually bounded by $2^{-\lambda}$. Finally, since $1 + 4x(B - x) = \sum_{i=1}^3 x_i^2$ and $x_0 = B - x$ in both transcripts, we can rewrite the hybrid r as $r = 4r_0(B - x_0) - \sum_{i \in [3]} r_i x_i \bmod N^\zeta$, which, given the x_i, is a deterministic function evaluated on independent statistically-closed distributions.

Lemma 3.1 ([24]). *Let integers $n, d \in \mathbb{Z}$, $B \geq 2$ and $x = \lfloor \frac{n}{d} \rceil$. If there exist $x_1, x_2, x_3 \in \mathbb{Q}$ such that $1 + 4 \cdot \frac{n}{d} \cdot \left(B - \frac{n}{d}\right) = \sum_{i=1}^3 x_i^2$, then we have $x \in [0, B]$.*

Lemma 3.2. *The above construction is a trapdoor Σ-protocol for $\bar{\mathcal{L}}^{B,B^*,C}$ assuming that $2^{2\lambda+3} B^2 C^2 < N^\zeta$, for any $\lambda \geq 1$.*

Proof. We first prove that $(\mathsf{ct}, \{C_i\}_{i=1}^3) \in \bar{\mathcal{L}}^{B,B^*,C}$ ensures that $\mathsf{ct} \in \mathcal{L}^{B,B^*,C}$.

Indeed, letting $\gamma = c^{-1} \bmod N^\zeta$ and $k \in \mathbb{Z}$ such that $\gamma \cdot c + k \cdot N^\zeta = 1$, the first four equations of (1) imply

$$\begin{aligned} C_i &= (1 + N)^{x_i \cdot \gamma} \cdot (s_i^\gamma \cdot C_i^k)^{N^\zeta} \bmod N^{\zeta+1}, \qquad \forall i \in [0, 3] \\ &= (1 + N)^{x_i \cdot (c^{-1} \bmod N^\zeta)} \cdot \tilde{s}_i^{N^\zeta} \bmod N^{\zeta+1} \end{aligned}$$

for some $\tilde{s}_i \in \mathbb{Z}_N^*$, and thus $\mathsf{ct} = (1+N)^{B - x_0 \cdot (c^{-1} \bmod N^\zeta)} \cdot \tilde{s}_0^{-N^\zeta} \bmod N^{\zeta+1}$. Hence, the ciphertexts $(\mathsf{ct}, \{C_i\}_{i=1}^3)$ decrypt to $\tilde{x} = B - x_0 \cdot c^{-1} \bmod N^\zeta$ and $\{\tilde{x}_i = x_i \cdot c^{-1} \bmod N^\zeta\}_{i=1}^3$. Then, decrypting the last equation of (1) implies

$$c = \sum_{i=1}^{3} \left(\frac{x_i}{c}\right) \cdot x_i - 4x_0 \cdot \left(B - \frac{x_0}{c}\right) \bmod N^\zeta.$$

If we multiply both members of the latter equation by c, we obtain

$$c^2 + 4(Bc - x_0)x_0 = \sum_{i=1}^{3} x_i^2 \bmod N^\zeta. \tag{4}$$

The latter equality holds over $\mathbb{Z}$ if we represent it over $[-N^\zeta/2, N^\zeta/2]$. Indeed, the absolute value the left-hand-side member is bounded by $C^2 + 4(BC + B^*)B^* = C^2 + 4(BC)^2(1+2^\lambda)2^\lambda \leq 2^{\lambda+3}B^2C^2 < N^\zeta/2$ and the right-hand-side member is bounded by $3{B^*}^2 = 3 \cdot 2^{2\lambda}B^2C^2 < N^\zeta/2$. If we divide both members by c^2 over the rationals, we obtain

$$1 + 4\Big(B - \frac{x_0}{c}\Big) \cdot \frac{x_0}{c} = \sum_{i=1}^{3} \left(\frac{x_i}{c}\right)^2 \quad \text{over } \mathbb{Q}.$$

By Lemma 3.1, this in turn implies $\lfloor x_0/c \rceil \in [0, B]$ and thus $B - \lfloor x_0/c \rceil \in [0, B]$.

We now prove that $\mathsf{BadChallenge}$ output the correct result when the prover sends commitments $\{C_i\}_{i=1}^3$ such that $(\mathsf{ct}, \{C_i\}_{i=1}^3) \notin \bar{\mathcal{L}}^{B,B^*,C}$. For a given first message $\mathbf{a} = (R, \{R_i\}_{i=0}^3, \{C_i\}_{i=1}^3)$ sent by the prover, $\mathsf{BadChallenge}$ obtains $r, \{r_i\}_{i=0}^3 \in \mathbb{Z}_{N^\zeta}$ and $\{x_i\}_{i=0}^3 \in \mathbb{Z}_{N^\zeta}$ at step 1. It only stops at step 2 if there exists $i \in [0,3]$ such that C_i decrypts to a value $\tilde{x}_i \in \mathbb{Z}_{N^\zeta}$ which has no representation $\tilde{x}_i = x_i \cdot c_i^{-1} \bmod N^\zeta$ with $(x_i, c_i) \in [-B^*, B^*] \times [0, C]$. In this case, only one pair $(\mathsf{Chall}_i, z_i) \in [0, C] \times [0, B^*]$ can satisfy the first verification equation of (2). Indeed, if we had distinct such pairs $(\mathsf{Chall}_i, z_i), (\mathsf{Chall}_i, z_i') \in [0, C] \times [0, B^*]$ with $\mathsf{Chall}_i' \neq \mathsf{Chall}_i$, we would have $C_i^{\mathsf{Chall}_i - \mathsf{Chall}_i'} = (1+N)^{z_i - z_i'} \cdot (t_i/t_i')^{N^\zeta} \bmod N^{\zeta+1}$ and thus $\tilde{x}_i = (z_i - z_i') \cdot (\mathsf{Chall}_i - \mathsf{Chall}_i')^{-1} \bmod N^\zeta$. Hence, the unique valid pair $(\mathsf{Chall}_i, z_i) \in [0, C] \times [0, B^*]$ that can satisfy the first equation (2) can be found by applying Gauss' algorithm. Note that $\mathsf{BadChallenge}$ might output $\mathsf{Chall} \neq \perp$ when no bad challenge exists at all.[8] However, $\mathsf{BadChallenge}$ only needs to find the bad challenge when it exists. When there is no bad challenge, the Fiat-Shamir hash function can output arbitrary values without hurting soundness.

If step 3 is reached, each plaintext in $\{\tilde{x}_i \in \mathbb{Z}_{N^\zeta}^*\}_{i=0}^3$ is decoded as a pair $(x_i, c_i) \in [-B^*, B^*] \times [0, C]$ such that $\tilde{x}_i = x_i \cdot c_i^{-1} \bmod N^\zeta$. We then define $c \triangleq \mathrm{lcm}(c_0, c_1, c_2, c_3)$ and distinguish two cases:

[8] This can happen when more than one $\{\tilde{x}_i\}_{i=0}^3$ has no valid representation $(x_i, c_i) \in [-B^*, B^*] \times [0, C]$, in which case they can possibly determine incompatible bad challenges.

(a) $c \notin [0, C]$ or $c \in [0, C]$ but there exist no integers $x'_0, x'_1, x'_2, x'_3 \in [-B^*, B^*]$ such that $\tilde{x}_i = x'_i \cdot c^{-1} \bmod N^\zeta$ for each $i \in [0, 3]$.
(b) $c \in [0, C]$ and there exist integers $x'_0, x'_1, x'_2, x'_3 \in [-B^*, B^*]$ such that we have $\tilde{x}_i = x'_i \cdot c^{-1} \bmod N^\zeta$ for each $i \in [0, 3]$.

In case (a), we observe from the first four verification equations (2) that a valid response $(\tau, \{(z_i, t_i)\}_{i=0}^3)$ can exist for at most one $\mathsf{Chall} \in [0, 2^\lambda - 1]$. This unique challenge value can be determined by solving an integer linear program and finding $(z_0, z_1, z_2, z_3, \mathsf{Chall}, k_0, k_1, k_2, k_3) \in \mathbb{Z}^9$ satisfying (4).

We are left with case (b). In order to satisfy the verification equations (2), the challenge-response pair $(\mathsf{Chall}, (\tau, \{(z_i, t_i)\}_{i=0}^3))$ must satisfy

$$z_i = r_i + \tilde{x}_i \cdot \mathsf{Chall} \mod N^\zeta \qquad r = -\sum_{i=1}^{3} \tilde{x}_i \cdot z_i + 4\tilde{x}z_0 + \mathsf{Chall} \mod N^\zeta.$$

Letting $\tilde{x} = B - \tilde{x}_0 \bmod N$, the above implies

$$\mathsf{Chall} \cdot (4\tilde{x} \cdot \tilde{x}_0 - \sum_{i=1}^{3} \tilde{x}_i^2 + 1) = r + \sum_{i=1}^{3} \tilde{x}_i \cdot r_i - 4\tilde{x} \cdot r_0 \mod N^\zeta, \tag{5}$$

Observe that we cannot have $4\tilde{x} \cdot \tilde{x}_0 - \sum_{i=1}^3 \tilde{x}_i^2 + 1 = 0 \bmod N^\zeta$ as this would imply $4\tilde{x} \cdot x'_0 - \sum_{i=1}^3 \tilde{x}_i \cdot x'_i + c = 0 \bmod N^\zeta$, which would mean that

$$(1+N)^c \cdot \prod_{i=1}^{3} C_i^{-x'_i} \cdot \mathsf{ct}^{4x'_0} \bmod N^{\zeta+1}$$

is an N^ζ-th residue in $\mathbb{Z}^*_{N^{\zeta+1}}$. Since we are in case (b), this would contradict the hypothesis $(\mathsf{ct}, \{C_i\}_{i=1}^3) \notin \bar{\mathcal{L}}^{B, B^*, C}$.

From the inequality $4\tilde{x} \cdot \tilde{x}_0 - \sum_{i=1}^3 \tilde{x}_i^2 + 1 \neq 0 \bmod N^\zeta$, we are guaranteed that $d_x = \gcd(4\tilde{x}\tilde{x}_0 - \sum_{i=1}^3 \tilde{x}_i^2 + 1, N^\zeta) < N^\zeta$ and (5) then yields

$$\mathsf{Chall} \cdot (4\tilde{x} \cdot \tilde{x}_0 - \sum_{i=1}^{3} \tilde{x}_i^2 + 1) = r + \sum_{i=1}^{3} \tilde{x}_i \cdot r_i - 4\tilde{x} \cdot r_0 \mod \frac{N^\zeta}{d_x}. \tag{6}$$

Since $\gcd(4\tilde{x} \cdot \tilde{x}_0 - \sum_{i=1}^3 \tilde{x}_i^2 + 1, N^\zeta / d_x) = 1$, equation (6) has a unique solution $\mathsf{Chall}_0 \in \mathbb{Z}_{N^\zeta/d_x}$. Since $N^\zeta/d_x > \min(p, q) > 2^\lambda$, we have $\mathsf{Chall} = \mathsf{Chall} \bmod N^\zeta/d_x$ for any $\mathsf{Chall} \in \{0, 1, \ldots, 2^\lambda - 1\}$, meaning that $\mathsf{BadChallenge}$ returns the correct result by outputting Chall_0 whenever $\mathsf{Chall}_0 \in \{0, \ldots, 2^\lambda - 1\}$. □

Compiling the Σ-protocol into multi-theorem NIZK. The trapdoor Σ-protocol immediately implies a single-theorem NIZK construction via the Fiat-Shamir transform when we apply the CI hash function of [56]. In order to obtain NIZK proofs in the multi-theorem setting, we could apply the compiler of [46, Appendix B]. One issue is that the latter proceeds by encrypting the Σ-protocol's first prover message using an equivocable lossy encryption system [4]. Unfortunately, while Paillier can serve as an equivocable lossy encryption scheme (as observed in [42]), we would lose the unbounded property of the range proof if we were to use it. The reason is that the CRS should contain a lossy/injective Paillier public key component that should be longer than messages to be encrypted.

Fortunately, multi-theorem NIZK proofs can be achieved (with computational zero-knowledge and statistical soundness) by adapting the Feige-Lapidot-Shamir compiler using correlation intractable hash functions. The OR trick of [31] builds multi-theorem NIZK proofs by showing OR statements of the form "either the statement is true OR some component of the CRS is in the image of a pseudorandom generator." Here, we can instantiate their approach using a DCR-based PRG. Recall that the DCR assumption immediately implies a length-doubling PRG that maps a seed $s \in \mathbb{Z}_N^*$ to $y = s^N \bmod N^2$. Here, we can apply the trapdoor Σ-protocol of [47] (which is recalled in Section 2.4) together with the OR Σ-protocols of [26] to prove that "either the range statement is true OR the CRS component $y \in \mathbb{Z}_{N^2}^*$ is an N-th residue." In the real construction, the CRS contains a uniformly random $y \sim U(\mathbb{Z}_{N^2}^*)$ so as to obtain statistical soundness. In the simulation, y is sampled as a composite residue and its N-th root allows simulating proofs. Using this approach, since the zero-knowledge property is only computational, we can obtain adaptive soundness by hashing the statement together with the prover's first message when the Fiat-Shamir transform is applied (as observed in [23, Theorem 4]).

In Section 4.2, we will apply a similar instantiation of the FLS paradigm to obtain one-time simulation-soundness in our DCR-based variant of Naor-Yung.

Dual-Mode Range Proofs/Arguments. If we give up unboundedness, we can obtain statistically zero-knowledge or even dual-mode range arguments as follows. The CRS initially chooses $\zeta > 1$ and a modulus N such that committed integers always live in a range $[0, B]$ for which $2^{3\lambda+1}B < N^\zeta$. The CRS is augmented with an element $g \in \mathbb{Z}_{N^{\zeta+1}}^*$ that is chosen as an N^ζ-th residue in the zero-knowledge setting (and uniformly over $\mathbb{Z}_{N^{\zeta+1}}^*$ in the soundness setting).

Then, each occurrence of $1 + N$ is replaced by g in the Σ-protocol. The DCR assumption immediately implies the indistinguishability of CRS distributions for the soundness and zero-knowledge settings. Moreover, our simulator $\mathsf{ZKSim}_B^{\mathrm{range}}$ produces statistically indistinguishable transcripts as it computes $\{C_i\}_{i=1}^3$ as dual-mode (or lossy) encryption of 0 instead of random elements modulo $N^{\zeta+1}$.

Achieving Constant Rate. Let $x \in [0, B]$ and $N^{\zeta'-1} \leq B \leq N^{\zeta'}$, for some integer ζ', and where only N is fixed by the CRS. We now assess the ratio between the input size and the proof size assuming that $n := |N|$. We see the witness x as a $|B|$-bit string since the zero-knowledge property requires a commitment whose message space contains $[0, B]$. For simplicity we assume that $\zeta = 2\zeta' + 1$ since our proof system requires $2^{2\lambda+3}B^2C^2 < N^\zeta$.

Since the commitment ct to x is a ciphertext over $\mathbb{Z}_{N^{\zeta+1}}^*$, we have

$$\frac{|\mathsf{ct}|}{|B|} \leq \frac{(\zeta + 1)n}{m} \leq \frac{(2\zeta' + 2)n}{(\zeta' - 1)n} = 2 + \frac{4}{(\zeta' - 1)} \downarrow 2.$$

The range proof π for x consists of $\{C_i\}_{i=1}^3, \{R_i\}_{i=0}^3, R$, each of size $(\zeta+1)n$, and of $\tau, \{(z_i, t_i)\}_{i=0}^3$, where $|\tau| = n$ and $|(z_i, t_i)| = (m + 3\lambda + 1) + n \leq (\zeta+1)n$, for each $i = 0$ to 3. The total proof size amounts to $12(\zeta+1)n + n$ and

$$\frac{|\pi|}{|\mathsf{ct}|} \leq \frac{12(\zeta+1)n + n}{(\zeta+1)n} = 12 + \frac{1}{2\zeta' + 2} \downarrow 12,$$

leading to a total rate of $|\pi|/|B| \leq (24(\zeta'+1)+1)/(\zeta'-1) \leq 73$ for $\zeta' > 1$, which goes down to 24 when ζ grows. If the OR trick is used in the multi-theorem case, it is easy to see that the asymptotic rate remains unchanged as the OR-branch involving the N-th residue only adds a component of size at most $4n$.

4 Instantiating Naor-Yung under the DCR Assumption

In this section, we show that decoding Paillier plaintexts as rounded rationals provides a secure instantiation of Naor-Yung under the DCR assumption. We first give a trapdoor Σ-protocol showing plaintext equalities before upgrading it into a one-time simulation-sound NIZK argument.

4.1 A Trapdoor Σ-Protocol Showing Plaintext Equalities Between Paillier Ciphertexts for Distinct Moduli

We now give a trapdoor Σ-protocol showing that two ciphertexts decrypt to the same plaintext in the encryption scheme of Section 2.6. Let $N_1 = p_1 q_1$ and $N_2 = p_2 q_2$ be RSA moduli. Let $C = 2^\lambda - 1$ and let also the languages

$$\begin{aligned}
\mathcal{L}_{\mathsf{zk}}^{\text{eq-dcr}} := \big\{(\mathsf{ct}_1, \mathsf{ct}_2, \ell_M) \in \mathbb{Z}^*_{N_1^\zeta} \times \mathbb{Z}^*_{N_2^\zeta} \times \mathbb{N} \ \mid\ &\exists m \in [0, M], \\
w_1 \in \mathbb{Z}^\star_{N_1},\ w_2 \in \mathbb{Z}^\star_{N_2}\ :\ &\mathsf{ct}_1 = (1+N_1)^m \cdot w_1^{N_1^\zeta} \bmod N_1^{\zeta+1} \\
\wedge\ &\mathsf{ct}_2 = (1+N_2)^m \cdot w_2^{N_2^\zeta} \bmod N_2^{\zeta+1}\big\},
\end{aligned}$$

$$\begin{aligned}
\mathcal{L}_{\mathsf{sound}}^{\text{eq-dcr}} := \big\{(\mathsf{ct}_1, \mathsf{ct}_2, \ell_M) \in \mathbb{Z}^*_{N_1^{\zeta+1}} \times \mathbb{Z}^*_{N_2^{\zeta+1}} \times \mathbb{N} \mid &\exists m \in [-R, R],\ \bar{c} \in [0, C], \\
w_1 \in \mathbb{Z}^\star_{N_1},\ w_2 \in \mathbb{Z}^\star_{N_2}\ :\ &\mathsf{ct}_1^{\bar{c}} = (1+N_1)^m \cdot w_1^{N_1^\zeta} \bmod N_1^{\zeta+1} \\
\wedge\ &\mathsf{ct}_2^{\bar{c}} = (1+N_2)^m \cdot w_2^{N_2^\zeta} \bmod N_2^{\zeta+1}\big\},
\end{aligned}$$

where $M = 2^{\ell_M} - 1$ and $\zeta \geq 1$ is the smallest integer such that

$$2RC < 2^{\lambda+1} R < \min(N_1^\zeta, N_2^\zeta)$$

with $R > 2^\lambda (C+1)(M+1)$. Note that $\mathcal{L}_{\mathsf{zk}}^{\text{eq-dcr}} \subset \mathcal{L}_{\mathsf{sound}}^{\text{eq-dcr}}$ since $M < R$.

We note that, for any pair of ciphertexts $((\mathsf{ct}_1, \ell_M), (\mathsf{ct}_2, \ell_M))$ such that $(\mathsf{ct}_1, \mathsf{ct}_2, \ell_M) \in \mathcal{L}_{\mathsf{sound}}^{\text{eq-dcr}}$, the decryption algorithms of Section 2.6 for N_1 and N_2 output the same $\mathsf{Msg} = \mathrm{abs}(\lfloor m/\bar{c} \rceil)$. Indeed, there exist $u_1, v_2, u_2, v_2 \in \mathbb{Z}$ with

$|u_1| < N_1^\zeta$ and $|u_2| < N_2^\zeta$ such that $u_1 \cdot \bar{c} + v_1 \cdot N_1^\zeta = 1$ and $u_2 \cdot \bar{c} + v_2 \cdot N_2^\zeta = 1$, which implies

$$\mathsf{ct}_1 = (1+N_1)^{u_1 \cdot m} \cdot (w_1^{u_1} \cdot \mathsf{ct}_1^{v_1})^{N_1^\zeta} \bmod N_1^{\zeta+1},$$
$$\mathsf{ct}_2 = (1+N_2)^{u_2 \cdot m} \cdot (w_2^{u_2} \cdot \mathsf{ct}_2^{v_2})^{N_2^\zeta} \bmod N_2^{\zeta+1}.$$

Since $u_1 = \bar{c}^{-1} \bmod N_1^\zeta$ and $u_2 = \bar{c}^{-1} \bmod N_2^\zeta$, the decryption algorithm necessarily outputs $\mathsf{Msg} = \lfloor m/\bar{c} \rceil$ in both cases.

We assume that the challenge space is $\{0, \ldots, C\}$, where $C = 2^\lambda - 1$, and that $p, q > 2^{l(\lambda)}$, for some polynomial $l : \mathbb{N} \to \mathbb{N}$ such that $l(\lambda) > \lambda$ for any sufficiently large $\lambda \in \mathbb{N}$. We now give a trapdoor Σ-protocol proving membership of $\mathcal{L}^{\text{eq-dcr}}_{\mathsf{sound}}$.

$\mathsf{Gen}_{\mathsf{par}}(1^\lambda)$: Given the security parameter λ, define $\mathsf{par} = \{\lambda\}$.

$\mathsf{Gen}_{\mathcal{L}}(\mathsf{par}, \mathcal{L}^{\text{eq-dcr}})$: Given public parameters par and a language description $\mathcal{L}^{\text{eq-dcr}}$, consisting of RSA moduli $N_1 = p_1 q_1$ and $N_2 = p_2, q_2$ with primes $p_1, q_1, p_2, q_2 > 2^{l(\lambda)}$, for some polynomial $l : \mathbb{N} \to \mathbb{N}$ such that $l(\lambda) > \lambda$, define the language-dependent CRS $\mathsf{crs}_{\mathcal{L}} = \{N_1, N_2\}$. The global CRS is $\mathsf{crs} = (\{\lambda\}, \mathsf{crs}_{\mathcal{L}})$.

$\mathsf{TrapGen}(\mathsf{par}, \mathcal{L}^{\text{eq-dcr}}, \tau_{\mathcal{L}})$: This algorithm is identical to $\mathsf{Gen}_{\mathcal{L}}(\mathsf{par}, \mathcal{L}^{\text{eq-dcr}})$, except that it also outputs the trapdoor $\tau_\Sigma = (p_1, q_1, p_2, q_2)$.

$\mathsf{P}(\mathsf{crs}, \vec{x}, \vec{w}) \leftrightarrow \mathsf{V}(\mathsf{crs}, \vec{x})$: On input of a common reference string crs, a statement $\vec{x} = (\mathsf{ct}_1, \mathsf{ct}_2, \ell_M) \in \mathbb{Z}^*_{N_1^{\zeta+1}} \times \mathbb{Z}^*_{N_2^{\zeta+1}} \times \mathbb{N}$, the prover P (who has $\vec{w} = (m, w_1, w_2) \in [0, M] \times \mathbb{Z}^\star_{N_1} \times \mathbb{Z}^\star_{N_2}$) and the verifier V interact as follows:

1. P chooses $a \hookleftarrow U([0, R])$, $r_1 \hookleftarrow U(\mathbb{Z}^*_{N_1})$, $r_2 \hookleftarrow U(\mathbb{Z}^*_{N_2})$ and sends
$$A_1 = (1+N_1)^a \cdot r_1^{N_1^\zeta} \bmod N_1^{\zeta+1}, \qquad A_2 = (1+N_2)^a \cdot r_2^{N_2^\zeta} \bmod N_2^{\zeta+1}.$$
2. V sends back a random challenge $\mathsf{Chall} \hookleftarrow U(\{0, \ldots, 2^\lambda - 1\})$.
3. P aborts if $a + \mathsf{Chall} \cdot m \notin [0, R]$. Otherwise, it sends V the response
$$z = a + \mathsf{Chall} \cdot m, \quad z_1 = r_1 \cdot w_1^{\mathsf{Chall}} \bmod N_1^\zeta, \quad z_2 = r_2 \cdot w_2^{\mathsf{Chall}} \bmod N_2^\zeta$$
4. V checks if $z \in [0, R]$ and accepts iff the following conditions hold:
$$A_1 \cdot \mathsf{ct}_1^{\mathsf{Chall}} \equiv z_1^{N_1^\zeta} \cdot (1+N_1)^z \pmod{N_1^{\zeta+1}},$$
$$A_2 \cdot \mathsf{ct}_2^{\mathsf{Chall}} \equiv z_2^{N_2^\zeta} \cdot (1+N_2)^z \pmod{N_2^{\zeta+1}}$$

$\mathsf{BadChallenge}(\mathsf{par}, \tau_\Sigma, \mathsf{crs}, \mathbf{x}, \mathbf{a})$: Given $\mathbf{x} = (\mathsf{ct}_1, \mathsf{ct}_2, \ell_M) \in \big(\mathbb{Z}^*_{N_1^{\zeta+1}}\big)^2 \times \mathbb{N}$, the message $\mathbf{a} = (A_1, A_2) \in \big(\mathbb{Z}^*_{N_1^{\zeta+1}}\big)^2$ and the trapdoor $\tau_\Sigma = (p_1, q_1, p_2, q_2)$,

1. Using $\mathsf{sk}_1 = (p_1, q_1)$, decrypt ct_1 and A_1 using Paillier's decryption algorithm to obtain $m_1 \in \mathbb{Z}_{N_1^\zeta}$ and $a_1 \in \mathbb{Z}_{N_1^\zeta}$. Likewise, use $\mathsf{sk}_2 = (p_2, q_2)$ to compute $m_2 \in \mathbb{Z}_{N_2^\zeta}$ and $a_2 \in \mathbb{Z}_{N_2^\zeta}$ by decrypting ct_2 and A_2.

2. Find an integer vector $(z, \mathsf{Chall}, k_1, k_2) \in \mathbb{Z}^4$ satisfying

$$\begin{aligned} a_1 &= z - m_1 \cdot \mathsf{Chall} + k_1 \cdot N_1^{\zeta} \\ a_2 &= z - m_2 \cdot \mathsf{Chall} + k_2 \cdot N_2^{\zeta}, \\ 0 &\leq \mathsf{Chall} \leq 2^{\lambda} - 1 \\ 0 &\leq k_1 \leq 2^{\lambda} \\ 0 &\leq k_2 \leq 2^{\lambda} \end{aligned} \tag{7}$$

This can be achieved by replacing the equalities by inequality pairs

$$\forall b \in \{1, 2\} : \begin{cases} z - m_b \cdot \mathsf{Chall} + k_b \cdot N_b^{\zeta} & \leq a_b, \\ -z + m_b \cdot \mathsf{Chall} - k_b \cdot N_b^{\zeta} & \leq -a_b \end{cases}$$

and running Lenstra's algorithm [45] to solve an integer linear programming instance with 10 constraints and 4 variables.

If a suitable $(z, \mathsf{Chall}, k_1, k_2) \in \mathbb{Z}^4$ is found (in which case, Chall is uniquely determined), output the corresponding Chall. Otherwise, return $\perp$.

Again, Lenstra's algorithm [45] allows computing the unique bad challenge (when it exists) in polynomial time since the number of variables is fixed.

Lemma 4.1. *The construction is a trapdoor Σ-protocol for* $(\mathcal{L}_{\mathsf{zk}}^{\text{eq-dcr}}, \mathcal{L}_{\mathsf{sound}}^{\text{eq-dcr}})$.

Proof. We first show the completeness and special zero-knowledge properties.

Completeness. Given $\vec{w} \in \mathcal{R}_{zk}^{\text{eq-dcr}}(\vec{x})$, P computes $(\mathbf{a}, \mathbf{z})$ for a challenge Chall such that $\mathsf{V}(\mathsf{crs}, \vec{x}, (\mathbf{a}, \mathsf{Chall}, \mathbf{z})) = 1$ as long as P does not abort at step 3 of the interactive protocol. Therefore, an honest run of the protocol always leads to a valid transcript except if $a + \mathsf{Chall} \cdot m \notin [0, R]$ which occurs with probability at most $2^{-\lambda}$ since $\mathsf{Chall} \cdot m \leq CM < 2^{\lambda + \ell_M}$ and $R > 2^{2\lambda + \ell_M}$.

Special zero-knowledge. The simulator ZKSim proceeds in a standard way. It that inputs $\mathsf{crs} = (\{\lambda\}, \mathsf{crs}_{\mathcal{L}})$, a statement $\vec{x} = (\mathsf{ct}_1, \mathsf{ct}_2, \ell_M) \in \mathcal{L}_{\mathsf{zk}}^{\text{eq-dcr}}$ and a challenge $\mathsf{Chall} \in \{0, \ldots, 2^{\lambda} - 1\}$. First, the simulator $\mathsf{ZKSim}(\mathsf{crs}, \vec{x}, \mathsf{Chall})$ picks $z \hookleftarrow U([0, R])$ as well as $z_1 \hookleftarrow U(\mathbb{Z}_{N_1}^*)$ and $z_2 \hookleftarrow U(\mathbb{Z}_{N_2}^*)$. Then, it computes $A_1 = z_1^{N_1^{\zeta}} \cdot (1 + N_1)^z \cdot \mathsf{ct}_1^{-\mathsf{Chall}} \bmod N_1^{\zeta+1}$, as well as

$$A_2 = z_2^{N_2^{\zeta}} \cdot (1 + N_2)^z \cdot \mathsf{ct}_2^{-\mathsf{Chall}} \bmod N_2^{\zeta+1},$$

and outputs $(\mathbf{a}, \mathbf{z})$, where $\mathbf{a} = (A_1, A_2)$ and $\mathbf{z} = (z, z_1, z_2)$. We turn to showing that $(\mathbf{a}, \mathsf{Chall}, \mathbf{z})$ is statistically indistinguishable from a real transcript computed using the witness $\vec{w} = (m, w_1, w_2) \in [0, M] \times \mathbb{Z}_{N_1}^{\star} \times \mathbb{Z}_{N_2}^{\star}$ (i.e., $\vec{w} \in \mathcal{R}_{zk}^{\text{eq-dcr}}(\vec{x})$) and with challenge Chall. For each $i \in \{1, 2\}$, let $\psi_i \colon \mathbb{Z}_{N_i^{\zeta}} \times \mathbb{Z}_{N_i}^{\star} \mapsto \mathbb{Z}_{N_i^{\zeta+1}}^{\star}$

denote the generalized Paillier isomorphism. By applying $\{\psi_i^{-1}\}_{i=1}^2$ to compute $(a_1, r_1) := \psi_1^{-1}(A_1)$ and $(a_2, r_2) := \psi_2^{-1}(A_2)$ for a simulated transcript $((A_1, A_2), \mathsf{Chall}, (z, z_1, z_2))$, we find

$$a_1 \equiv z - \mathsf{Chall} \cdot m \pmod{N_1^\zeta} \qquad r_1 \equiv z_1 \cdot w_1^{-\mathsf{Chall}} \pmod{N_1},$$
$$a_2 \equiv z - \mathsf{Chall} \cdot m \pmod{N_2^\zeta} \qquad r_2 \equiv z_2 \cdot w_2^{-\mathsf{Chall}} \pmod{N_2}.$$

The congruences on the left ensure that, unless $z \in [0, CM]$ (which occurs with negligible probability $2^{-\lambda}$), we have $0 \leq a_1 = z - \mathsf{Chall} \cdot m = a_2 \leq R$. Given Chall, the distributions of $\{(z_i, r_i)\}_{i=1}^2$ over the multiplicative rings are exactly the same between the real and the simulated transcripts. Finally, we show that, over the integers, the statistical distance between $U([0, R])$ (which is the distribution of the simulated z) and $\mathsf{Chall} \cdot m + U([0, R])$ (in the real z) is negligible. Since $m \cdot \mathsf{Chall} \leq MC < 2^{\lambda + \ell_M} < 2^{-\lambda} R$, it is actually bounded by $2^{-\lambda}$.

Special soundness. Let us assume two transcripts $((A_1, A_2), \mathsf{Chall}, (z, z_1, z_2))$ and $\big((A_1, A_2), \mathsf{Chall}', (z', z_1', z_2')\big)$ that both satisfy the verification equations with $z, z' \in [0, R]$ and $\mathsf{Chall} \neq \mathsf{Chall}'$ for a given first message (A_1, A_2) sent by the prover. We assume w.l.o.g. that $0 \leq \mathsf{Chall}' < \mathsf{Chall} \leq 2^\lambda - 1$. This implies that $\bar{c} = \mathsf{Chall} - \mathsf{Chall}' \in [0, 2^\lambda - 1]$ and $\bar{z} = z - z' \in [-R, R]$ satisfy the congruences $\mathsf{ct}_1^{\bar{c}} \equiv (z_1/z_1')^{N_1^\zeta} (1 + N_1)^{\bar{z}} \pmod{N_1^{\zeta+1}}$ and

$$\mathsf{ct}_2^{\bar{c}} \equiv (z_2/z_2')^{N_2^\zeta} (1 + N_2)^{\bar{z}} \pmod{N_2^{\zeta+1}},$$

which implies $(\mathsf{ct}_1, \mathsf{ct}_2) \in \mathcal{L}_{\mathsf{sound}}^{\text{eq-dcr}}$. This shows that, for any first message (A_1, A_2) sent by the prover, only one bad challenge can exist if $(\mathsf{ct}_1, \mathsf{ct}_2) \notin \mathcal{L}_{\mathsf{sound}}^{\text{eq-dcr}}$.

CRS indistinguishability. The distribution of the CRS output by $\mathsf{TrapGen}$ is exactly the same as the distribution of the CRS output by $\mathsf{Gen}_{\mathcal{L}}$.

BadChallenge correctness. Let a false statement $\vec{x} \notin \mathcal{L}_{\mathsf{sound}}^{\text{eq-dcr}}$. Special soundness ensures the existence of at most one bad challenge for any given $\mathbf{a}$. Lenstra's algorithm can efficiently determine if the bad challenge exists since it can solve the integer feasibility problem in polynomial time when the number of variables is fixed. Moreover, whenever an admissible integer solution $(z, \mathsf{Chall}, k_1, k_2) \in \mathbb{Z}^4$ exists (in which case it is unique), it is efficiently computable from the decrypted values (m_1, m_1, a_1, a_2). □

4.2 New Construction of One-Time Simulation-Sound NIZK Arguments from Trapdoor Σ-Protocols

In this section, we aim at one-time simulation soundness without imposing a bound on the plaintext space in the centralized version our scheme of Section 4.3. To this end, we cannot use the constructions of [47,30] because they follow an idea from [27] and encrypt the prover's first message using a DCR-based lossy encryption scheme [4]. Unfortunately, the latter's public key should be larger than the first prover message in the underlying trapdoor Σ-protocol.

We describe a new one-time simulation-sound argument which departs from [46,47,30] in that it does not proceed by encrypting the first prover message of the trapdoor Σ-protocol. Instead, it uses an OR proof [26] inspired by the FLS technique [31]. In order to achieve one-time simulation-soundness, we introduce a twist and program the CRS $(u, v) \in (\mathbb{Z}_{N^2}^*)^2$ in such a way that $u^{\mathsf{VK}} \cdot v$ is a composite residue for exactly one VK.

- A trapdoor Σ-protocol $\Pi^{(1)} = (\mathsf{Gen}_{\mathsf{par}}^{(1)}, \mathsf{Gen}_{\mathcal{L}}^{(1)}, \mathsf{P}^{(1)}, \mathsf{V}^{(1)})$ for an NP language $\mathcal{L}$. This protocol should satisfy the properties of Definition 2.6. We assume that $\Pi^{(1)}$ has challenge space $\mathcal{C} = \{0,1\}^\lambda$, where λ is the security parameter. In addition, the function $\mathsf{BadChallenge}^{(1)}$ should be computable within time $T_1 \in \mathsf{poly}(\lambda)$ for any input $(\tau, \mathsf{crs}^{(1)}, x, a_1)$.
- A strongly unforgeable one-time signature scheme $\mathsf{OTS} = (\mathcal{G}, \mathcal{S}, \mathcal{V})$ with verification keys in $\{0,1\}^L$, where $L \in \mathsf{poly}(\lambda)$.
- An RSA modulus $N = pq$, for large primes $p, q > 2^L$.
- A trapdoor Σ-protocol $\Pi^{(0)} = (\mathsf{Gen}_{\mathsf{par}}^{(0)}, \mathsf{Gen}_{\mathcal{L}}^{(0)}, \mathsf{P}^{(0)}, \mathsf{V}^{(0)})$ for the language $\mathcal{L}^{\mathsf{DCR}} := \{x \in \mathbb{Z}_{N^2}^* \mid \exists w \in \mathbb{Z}_N^\star : x = w^N \bmod N^2\}$. We assume that the function $\mathsf{BadChallenge}^{(0)}$ is computable within time $T_0 \in \mathsf{poly}(\lambda)$ for any input $(\tau, \mathsf{crs}^{(0)}, x, a_0)$. This protocol can be instantiated as in Section 2.4
- A correlation intractable hash family $\mathcal{H} = (\mathsf{Gen}, \mathsf{Hash})$ for the class $\mathcal{R}_{\mathsf{CI}}$ of relations that are efficiently searchable within time T.

$\mathsf{Gen}_{\mathsf{par}}(1^\lambda)$: Run $\mathsf{par} \leftarrow \mathsf{Gen}_{\mathsf{par}}^{(1)}(1^\lambda)$ and output par.

$\mathsf{Gen}_{\mathcal{L}}(\mathsf{par}, \mathcal{L})$: Given public parameters par and a language $\mathcal{L}$, the CRS is generated as follows.

1. Generate a CRS $\mathsf{crs}_{\mathcal{L}}^{(1)} \leftarrow \mathsf{Gen}_{\mathcal{L}}^{(1)}(\mathsf{par}, \mathcal{L})$ for the trapdoor Σ-protocol $\Pi^{(1)}$.
2. Choose the description of a one-time signature scheme $\mathsf{OTS} = (\mathcal{G}, \mathcal{S}, \mathcal{V})$ with verification keys in $\{0,1\}^L$, where $L \in \mathsf{poly}(\lambda)$.
3. Choose an RSA modulus $N = pq$, for primes $p, q > 2^L$, where $L \in \mathsf{poly}(\lambda)$ is the verification key length of OTS. Then, choose $u_0, v_0 \hookleftarrow \mathbb{Z}_N^*$ and compute $u = u_0^N \bmod N^2$, $v = v_0^N \bmod N^2$.
4. Generate a CRS $\mathsf{crs}^{(0)} \leftarrow \mathsf{Gen}_{\mathcal{L}}^{(0)}(\mathsf{par}, \mathcal{L}^{\mathsf{DCR}})$ for $\Pi^{(0)}$, where $\mathcal{L}^{\mathsf{DCR}}$ is associated with $N = pq$.
5. Generate a key $k \leftarrow \mathsf{Gen}(1^\lambda)$ for a correlation intractable hash function with output length λ.

Output the language-dependent CRS $\mathsf{crs}_{\mathcal{L}} := \big(N, u, v, \mathsf{crs}^{(0)}, \mathsf{crs}_{\mathcal{L}}^{(1)}, k\big)$ and the simulation trapdoor $\tau_{\mathsf{zk}} := (u_0, v_0)$. The global common reference string consists of $\mathsf{crs} = (\mathsf{par}, \mathsf{crs}_{\mathcal{L}}, \mathsf{OTS})$.

$\mathsf{P}(\mathsf{crs}, x, w, \mathsf{lbl})$: To prove a statement $x \in \mathcal{L}$ for a label $\mathsf{lbl} \in \{0,1\}^*$ using the witness w, generate a one-time signature key pair $(\mathsf{VK}, \mathsf{SK}) \leftarrow \mathcal{G}(1^\lambda)$. Then,

1. Compute $(a_1, st) \leftarrow \mathsf{P}^{(1)}(\mathsf{crs}_{\mathcal{L}}^{(1)}, x, w)$. Then, generate a simulated proof $(a_0, \mathsf{Chall}_0, z_0) \in \mathbb{Z}_{N^2}^* \times \{0,1\}^\lambda \times \mathbb{Z}_N^*$ that $(u^{\mathsf{VK}} \cdot v) \in \mathcal{L}^{\mathsf{DCR}}$. Namely, choose random elements $z_0 \hookleftarrow U(\mathbb{Z}_N^*)$, $\mathsf{Chall}_0 \hookleftarrow U(\{0,1\}^\lambda)$ and compute $a_0 = z_0^N \cdot (u^{\mathsf{VK}} \cdot v)^{-\mathsf{Chall}_0} \bmod N^2$.

2. Compute $\mathsf{Chall} = \mathsf{Hash}\big(k, (x, a, \mathsf{VK})\big) \in \{0,1\}^\lambda$, where $a = (a_0, a_1)$, and set $\mathsf{Chall}_1 = \mathsf{Chall} \oplus \mathsf{Chall}_0$.
3. Compute $z_1 = \mathsf{P}^{(1)}(\mathsf{crs}_{\mathcal{L}}^{(1)}, x, w, a_1, \mathsf{Chall}_1, st)$ by executing the prover of $\Pi^{(1)}$. Define $z = (z_0, z_1, \mathsf{Chall}_0, \mathsf{Chall}_1)$.
4. Generate $sig \leftarrow \mathcal{S}(\mathsf{SK}, (x, a, z, \mathsf{lbl}))$ and output $\vec{\pi} = \big(\mathsf{VK}, (a, z), sig\big)$.

$\mathbf{V}(\mathsf{crs}, x, \vec{\pi}, \mathsf{lbl})$: Given a statement x, a label lbl as well as a purported proof $\vec{\pi} = \big(\mathsf{VK}, (a, z), sig\big)$, return 0 if $\mathcal{V}(\mathsf{VK}, (x, a, z, \mathsf{lbl}), sig) = 0$. Otherwise,

1. Write $z = (z_0, z_1, \mathsf{Chall}_0, \mathsf{Chall}_1)$ and return 0 if any of these does not parse properly or if $\mathsf{Hash}\big(k, (x, a, \mathsf{VK})\big) \neq \mathsf{Chall}_0 \oplus \mathsf{Chall}_1$.
2. If $\mathsf{V}^{(1)}(\mathsf{crs}_{\mathcal{L}}^{(1)}, x, a_1, \mathsf{Chall}_1, z_1)) = 1$ and $a_0 \cdot (u^{\mathsf{VK}} \cdot v)^{\mathsf{Chall}_0} = z_0^N \bmod N^2$, return 1. Otherwise, return 0.

Theorem 4.2. *The above construction is a one-time simulation-sound NIZK argument if: (i)* OTS *is a strongly unforgeable one-time signature; (ii) The* DCR *assumption holds; (iii) The hash function is correlation-intractable for efficiently searchable relations.* (The proof is given in the full version of this paper.)

4.3 A DCR-Based CCA2-Secure Threshold Cryptosystem from the Naor-Yung Paradigm

The syntax and the security definitions of threshold PKE schemes are recalled in the full version of this paper. Using the tools of Section 4.1 and Section 4.2, we obtain the following variant of the threshold encryption scheme in [33].

We assume that the key generation step chooses a value ζ' that determines a maximal length of encrypted messages (note that this is only necessary in the threshold setting and not in the centralized version of the scheme). However, the encryptor can still choose $\zeta \leq \zeta'$ in a flexible way at encryption time.

For simplicity, we first describe the non-robust version of the scheme, where decryption servers do not provide a proof that partial decryptions are correctly generated. However, robustness can be achieved in a modular way as in [30].

$\mathbf{Keygen}(1^\lambda, 1^B, 1^t, 1^n)$: On input of a security parameter λ, a maximal bitlength $B \in \mathsf{poly}(\lambda)$ of encrypted messages, a number of servers $n \in \mathsf{poly}(\lambda)$, and a threshold $t \in \mathsf{poly}(\lambda)$, conduct the following steps.

1. Generate two safe prime products $N_1 = p_1 q_1$ and $N_2 = p_2 q_2$ such that $p_i, q_i > 2^{l(\lambda)}$, for some polynomial $l : \mathbb{N} \rightarrow \mathbb{N}$, and primes $p_i = 2p_i' + 1$, $q_i = 2q_i' + 1$ for which p_i', q_i' are also prime for each $i \in \{1, 2\}$.
2. Choose an integer $\zeta' > 0$ such that $2^{B+2\lambda+1} < \min(N_1^{\zeta'}, N_2^{\zeta'})$.
3. Choose an integer d such that $d = 1 \bmod N_1^{\zeta'}$ and $d = 0 \bmod \lambda(N_1)$.
4. Choose a random polynomial $f[X] = \sum_{i=0}^{t-1} a_i X^i \in \mathbb{Z}_{N_1^{\zeta'} p_1' q_1'}[X]$ such that $a_0 = d \bmod N_1^{\zeta'} p_1' q_1'$.
5. Generate the CRS $\mathsf{crs}_{\mathcal{L}} := \big(N, u, v, \mathsf{crs}^{(0)}, \mathsf{crs}_{\mathcal{L}}^{(1)}, k\big)$ of a simulation-sound NIZK argument for the language $(\mathcal{L}_{\mathsf{zk}}^{\text{eq-dcr}}, \mathcal{L}_{\mathsf{sound}}^{\text{eq-dcr}})$ of Section 4.1, which is induced by the moduli N_1 and N_2.

The public key is $\mathsf{pk} = (N_1, N_2, \mathsf{crs}_{\mathcal{L}})$ and the secret key shares $\{\mathsf{sk}_i\}_{i=1}^n$ are defined as $\mathsf{sk}_i = f(i) \bmod N_1^{\zeta'} p_1' q_1'$ for each $i \in [n]$.

Encrypt$\big(\mathsf{pk}, \mathsf{Msg}\big)$: To encrypt $\mathsf{Msg} \in \{0,1\}^{\ell_M}$, return $\perp$ if $\ell_M > B$. Otherwise, interpret Msg as a positive integer in $[0, M]$, where $M = 2^{\ell_M} - 1$. Set $\zeta > 1$ as the smallest integer such that $\min(N_1^{\zeta}, N_2^{\zeta}) \geq 2^{2\lambda+1} M$. Then, choose $(r_1, r_2) \hookleftarrow U(\mathbb{Z}_{N_1}^* \times \mathbb{Z}_{N_2}^*)$ and compute

$$\mathsf{ct}_1 = (1+N_1)^{\mathsf{Msg}} \cdot r_1^{N_1^{\zeta}} \bmod N_1^{\zeta+1}, \quad \mathsf{ct}_2 = (1+N_2)^{\mathsf{Msg}} \cdot r_2^{N_2^{\zeta}} \bmod N_2^{\zeta+1}.$$

Then, using the empty label $\mathsf{lbl} = \varepsilon$, generate a simulation-sound NIZK argument $\vec{\pi} \leftarrow \mathsf{P}\big(\mathsf{crs}, (\mathsf{ct}_1, \mathsf{ct}_2, \ell_M), (\mathsf{Msg}, r_1, r_2), \mathsf{lbl}\big)$ that $(\mathsf{ct}_1, \mathsf{ct}_2, \ell_M) \in \mathcal{L}_{\mathsf{zk}}^{\text{eq-dcr}}$. Finally, output $\mathsf{ct} = (\mathsf{ct}_1, \mathsf{ct}_2, \ell_M, \vec{\pi})$.

PartDec$\big(\mathsf{sk}_i, \mathsf{ct}\big)$: Given a ciphertext $\mathsf{ct} = (\mathsf{ct}_1, \mathsf{ct}_2, \ell_M, \vec{\pi})$ and $\mathsf{sk}_i \in \mathbb{Z}_{N_1^{\zeta'} p_1' q_1'}$, the i-th server proceeds as follows.

1. If $\mathsf{V}(\mathsf{crs}, (\mathsf{ct}_1, \mathsf{ct}_2, \ell_M), \vec{\pi}, \mathsf{lbl}) = 0$, return $\perp$.
2. Compute $\mu_i = \mathsf{ct}_1^{2\Delta \cdot \mathsf{sk}_i} \bmod N_1^{\zeta+1}$, where $\Delta = n!$, and return (i, μ_i).

Combine$\big(\mathsf{pk}, \mathcal{S}, \{\mu_i\}_{i\in\mathcal{S}}, \mathsf{ct}\big)$**:** Let $R = 2^{\lambda} \cdot (M+1)$ and $C = 2^{\lambda} - 1$. If $\mathcal{S}$ contains less than t shares in $\mathbb{Z}^*_{N_1^{\zeta+1}}$, return $\perp$. Otherwise, do the following.

1. Define scaled Lagrange coefficients $\lambda_{0,i}^{\mathcal{S}} = \Delta \cdot \prod_{i' \in \mathcal{S}\setminus\{i\}} \frac{-i}{i-i'} \in \mathbb{Z}$ for each $i \in \mathcal{S}$ and compute $\mu_0 = \prod_{i\in\mathcal{S}} \mu_i^{2\cdot\lambda_{0,i}^{\mathcal{S}}} \bmod N_1^{\zeta+1}$, which should be of the form $\mu_0 = \mathsf{ct}_1^{4\Delta^2 f(0)} = \mathsf{ct}_1^{4\Delta^2 d} \bmod N_1^{\zeta+1}$.
2. Compute $\tilde{\mu} = L(\mu_0, N_1^{\zeta}) \cdot 4^{-1} \cdot (\Delta)^{-2} \bmod N_1^{\zeta}$, where $L(\cdot, N_1^{\zeta})$ extracts the discrete logarithm w.r.t. base $(1+N_1)$ of the elements modulo $N_1^{\zeta+1}$ that are congruent to 1 modulo N_1 as in [29]. Then, using Gauss' algorithm, find the unique $(m, c) \in \mathbb{Z}^2$ such that $-R \leq m \leq R$, $0 \leq c \leq C$ and $\tilde{\mu} = m \cdot c^{-1} \bmod N^{\zeta}$. If no such pair exists, return $\perp$. Otherwise, return $\mathsf{Msg} = \mathrm{abs}(\lfloor m/c \rceil)$, where the division is computed over $\mathbb{Q}$.

Theorem 4.3. *The scheme provides IND-CCA security under static corruptions if: (i) The* DCR *assumption holds; (ii)* Π^{OTSS} *is a one-time simulation-sound argument.* (The proof is given in the full version of this paper.)

Comparisons. Devevey *et al.* [30, Section 4] gave a non-interactive threshold CCA2-secure scheme based on DCR and LWE in the standard model. While they can prove security under adaptive corruptions, our scheme provides several advantages over [30] although we only prove static security.[9] In the robust version of the scheme, if we do not consider commitments to the secret key shares as being part of the public key (which is reasonable as the encryptor does not need

[9] Adaptive security is non-trivial to achieve when $t, n \in \mathsf{poly}(\lambda)$. In many applications like e-voting, one can expect the number of servers to be small (e.g., logarithmic in λ), in which case adaptive security can be achieved via complexity leveraging.

them), the public key size grows with $|N|$ instead of $|N^\zeta|$. Also, the scheme of [30] requires larger secret key shares, which grow super-linearly with the number of servers. Finally, our scheme allows the sender to adjust the block length by choosing ζ according to the message length.

The full version of this paper provides more comparisons.

Acknowledgements

Part of this research was funded by the French ANR ALAMBIC project (ANR-16-CE39-0006). This work is also partially supported by Indo French Center for the Promotion of Advanced Research (IFCPAR, project number: 6002-1). Thomas Peters is a research associate of the Belgian Fund for Scientific Research (F.R.S.-FNRS).

References

1. M. Abe and S. Fehr. Adaptively secure Feldman VSS and applications to universally-composable threshold cryptography. In *Crypto*, 2004.
2. G. Asharov, A. Jain, and D. Wichs. Multiparty computation with low communication, computation and interaction via threshold FHE. Cryptology ePrint Archive: Report 2011/613, 2012.
3. M. Bellare and S. Goldwasser. Verifiable partial key escrow. In *ACM-CCS*, 1997.
4. M. Bellare, D. Hofheinz, and S. Yilek. Possibility and impossibility results for encryption and commitment secure under selective opening. In *Eurocrypt*, 2009.
5. D. Bernhard, V. Cortier, O. Pereira, B. Smyth, and B. Warinschi. Adapting helios for provable ballot privacy. In *ESORICS*, 2011.
6. M. Blum, A. De Santis, S. Micali, and G. Persiano. Non-interactive zero-knowledge. *SIAM J. of Computing*, 1991.
7. M. Blum, P. Feldman, and S. Micali. Non-interactive zero-knowledge and its applications. In *STOC*, 1988.
8. D. Boneh, X. Boyen, and S. Halevi. Chosen ciphertext secure public key threshold encryption without random oracles. In *CT-RSA*, 2006.
9. D. Boneh, R. Gennaro, S. Goldfeder, A. Jain, S. Kim, P. Rasmussen, and A. Sahai. Threshold cryptosystems from threshold fully homomorphic encryption. In *Crypto*, 2018.
10. F. Boudot. Efficient proofs that a committed number lies in an interval. In *Eurocrypt*, 2000.
11. E. Brickell, D. Chaum, I. Damgård, and J. van de Graaf. Gradual and verifiable release of a secret. In *Crypto*. Springer, 1988.
12. B. Bünz, S. Agrawal, M. Zamani, and D. Boneh. Zether: Towards privacy in a smart contractworld. In *FC*, 2020.
13. B. Bünz, J. Bootle, D. Boneh, A. Poelstra, P. Wuille, and G. Maxwell. Bulletproofs: Short proofs for confidential transactions and more. In *IEEE S&P*, 2018.
14. J. Camenisch, R. Chaabouni, and A. shelat. Efficient protocols for set membership and range proofs. In *Asiacrypt*, 2008.
15. J. Camenisch and A. Lysyanskaya. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In *Eurocrypt*, 2001.

16. R. Canetti, Y. Chen, J. Holmgren, A. Lombardi, G. Rothblum, R. Rothblum, and D. Wichs. Fiat-Shamir: From practice to theory. In *STOC*, 2019.
17. R. Canetti, O. Goldreich, and S. Halevi. The random oracle methodology, revisted. *J. of the ACM*, 51(4), 2004.
18. R. Canetti and S. Goldwasser. An efficient threshold public key cryptosystem secure against adaptive chosen-ciphertext attacks. In *Eurocrypt*, 1999.
19. R. Canetti, A. Lombardi, and D. Wichs. Fiat-Shamir: From Practice to Theory, Part II (NIZK and Correlation Intractability from Circular-Secure FHE). Cryptology ePrint Archive: Report 2018/1248.
20. R. Chaabouni, H. Lipmaa, and B. Zhang. A non-interactive range proof with constant communication. In *Financial Cryptography*, 2012.
21. P. Chaidos and J. Groth. Making Sigma-protocols non-interactive without random oracles. In *PKC*, 2015.
22. A. Chan, Y. Frankel, and Y. Tsiounis. Easy come – easy go divisible cash. In *Eurocrypt*, 1998.
23. M. Ciampi, R. Parisella, and D. Ventury. On adaptive security of delayed-input Sigma protocols and Fiat-Shamir NIZKs. In *SCN*, 2020.
24. G. Couteau, M. Klooß, H. Lin, and M. Reichle. Efficient range proofs with transparent setup from bounded integer commitments. In *Eurocrypt*, 2021.
25. G. Couteau, T. Peters, and D. Pointcheval. Removing the strong RSA assumption from arguments over the integers. In *Eurocrypt*, 2017.
26. R. Cramer, I. Damgård, and B. Schoenmaekers. Proofs of partial knowledge and simplified design of witness hiding protocols. In *Crypto*, 1994.
27. I. Damgård. Efficient concurrent zero-knowledge in the auxiliary string model. In *Eurocrypt*, 2000.
28. I. Damgård and E. Fujisaki. A statistically-hiding integer commitment scheme based on groups with hidden order. In *Asiacrypt*, 2002.
29. I. Damgård and M. Jurik. A generalisation, a simplification and some applications of Paillier's probabilistic public-key system. In *PKC*, 2001.
30. J. Devevey, B. Libert, K. Nguyen, T. Peters, and M. Yung. Non-interactive CCA2-secure threshold cryptosystems: Achieving adaptive security in the standard model without pairings. In *PKC*, 2021.
31. U. Feige, D. Lapidot, and A. Shamir. Multiple non-interactive zero knowledge proofs based on a single random string (extended abstract). In *FOCS*, 1990.
32. A. Fiat and A. Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *Crypto*, 1986.
33. P.-A. Fouque and D. Pointcheval. Threshold Cryptosystems Secure against Chosen-Ciphertext Attacks. In *Asiacrypt*, 2001.
34. P.-A. Fouque, J. Stern, and G. Wacker. Cryptocomputing with rationals. In *FC*, 2002.
35. E. Fujisaki and T. Okamoto. Statistical zero knowledge protocols to prove modular polynomial relations. In *Crypto*, 1997.
36. S. Goldwasser, S. Micali, and C. Rackoff. The knowledge complexity of interactive proof systems. *SIAM Journal on Computing*, 1989.
37. A. Gonzalez and C. Ràfols. New techniques for non-interactive shuffle and range arguments. In *ACNS*, 2017.
38. J. Groth. Non-interactive zero-knowledge arguments for voting. In *ACNS*, 2005.
39. J. Groth. Efficient zero-knowledge arguments from two-tiered homomorphic commitments. In *Asiacrypt*, 2011.
40. J. Groth, R. Ostrovsky, and A. Sahai. New techniques for noninteractive zero-knowledge. *J. ACM*, 2012.

41. J. Groth and A. Sahai. Efficient non-interactive proof systems for bilinear groups. In *Eurocrypt*, 2008.
42. B. Hemenway, B. Libert, R. Ostrovsky, and D. Vergnaud. Lossy encryption: Constructions from general assumptions and efficient selective opening chosen ciphertext security. In *Asiacrypt*, 2011.
43. S. Jarecki and A. Lysyanskaya. Adaptively secure threshold cryptography: Introducing concurrency, removing erasures. In *Eurocrypt*, 2000.
44. A. Kiayias, N. Leonardos, H. Lipmaa, K. Pavlyk, and Q. Tang. Near optimal rate homomorphic encryption for branching programs. *Priv. Enhancing Technol.*, 2015.
45. H. Lenstra. Integer programming with a fixed number of variables. *Mathematics of Operations Research*, 8(4), 1983.
46. B. Libert, K. Nguyen, A. Passelègue, and R. Titiu. Simulation-sound arguments for LWE and applications to KDM-CCA2 security. In *Asiacrypt*, 2020.
47. B. Libert, K. Nguyen, T. Peters, and M. Yung. One-shot Fiat-Shamir-based NIZK arguments of composite residuosity in the standard model. Cryptology ePrint Archive: Report 2020/1334, 2020.
48. B. Libert and M. Yung. Non-interactive cca-secure threshold cryptosystems with adaptive security: New framework and constructions. In *TCC*, 2012.
49. H. Lipmaa. On Diophantine complexity and statistical zero-knowledge arguments. In *Asiacrypt*, 2003.
50. H. Lipmaa. Optimally sound sigma protocols under DCRA. In *FC*, 2017.
51. H. Lipmaa, N. Asokan, and V. Niemi. Secure vickrey auctions without threshold trust. In *Financial Cryptography*, 2002.
52. M. Naor and M. Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *STOC*, 1990.
53. S. Noether. Ring signature confidential transactions for monero. Cryptology ePrint Archive Report 2015/1098, 2015.
54. P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Eurocrypt*, 1999.
55. T. Pedersen. Non-interactive and information-theoretic secure verifiable secret sharing. In *Crypto*, 1991.
56. C. Peikert and S. Shiehian. Non-interactive zero knowledge for NP from (plain) Learning With Errors. In *Crypto*, 2019.
57. O. Regev. On lattices, learning with errors, random linear codes, and cryptography. In *STOC*, 2005.
58. A. Rial, M. Kohlweiss, and B. Preneel. Universally composable adaptive priced oblivious transfer. In *Pairing*, 2009.
59. A. Sahai. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In *FOCS*, 1999.
60. V. Shoup and R. Gennaro. Securing threshold cryptosystems against chosen ciphertext attack. In *Eurocrypt*, 1998.
61. B. Vallée. Gauss' algorithm, revisited. *J. of Algorithms*, 1991.