Leveraging Small Message Spaces for CCA1 Security in Additively Homomorphic and BGN-type Encryption

Benoît Libert

Zama, France

Abstract. We show that the smallness of message spaces can be used as a checksum allowing to hedge against CCA1 attacks in additively homomorphic encryption schemes. We first show that the additively homomorphic variant of Damgård's Elgamal provides IND-CCA1 security under the standard DDH assumption. Earlier proofs either required non-standard assumptions or only applied to hybrid versions of Damgård's Elgamal, which are not additively homomorphic. Our security proof builds on hash proof systems and exploits the fact that encrypted messages must be contained in a polynomial-size interval in order to enable decryption. With 3 group elements per ciphertext, this positions Damgård's Elgamal as the most efficient/compact DDH-based additively homomorphic CCA1 cryptosystem. Under the same assumption, the best candidate so far was the lite Cramer-Shoup cryptosystem, where ciphertexts consist of 4 group elements. We extend this observation to build an IND-CCA1 variant of the Boneh-Goh-Nissim encryption scheme, which allows evaluating 2-DNF formulas on encrypted data. By computing tensor products of Damgård's Elgamal ciphertexts, we obtain product ciphertexts consisting of 9 elements (instead of 16 elements if we were tensoring lite Cramer-Shoup ciphertexts) in the target group of a bilinear map. Using similar ideas, we also obtain a CCA1 variant of the Elgamal-Paillier cryptosystem by forcing λ plaintext bits to be zeroes, which yields CCA1 security almost for free. In particular, the message space remains exponentially large and ciphertexts are as short as in the IND-CPA scheme. We finally adapt the technique to the Castagnos-Laguillaumie system.

Keywords: Additively homomorphic encryption · BGN encryption · CCA1 security · Standard assumptions

1 Introduction

It is well known that homomorphic encryption schemes cannot withstand adaptive chosen-ciphertext attacks as they are inherently malleable. However, they can still satisfy the notion of non-adaptive chosen-ciphertext (a.k.a. IND-CCA1) security, where the adversary is only given access to a decryption oracle before the challenge phase. While weaker, IND-CCA1 security is still a meaningful and desirable security property. In particular, it guarantees security under chosen-ciphertext key-recovery attacks, meaning that an adversary cannot reconstruct

the secret key by observing decryptions of maliciously generated ciphertexts.

In the context of fully homomorphic encryption, CCA1 security turns out to be very difficult to achieve and even hardly compatible with bootstrapping and its approach of revealing an encryption of the secret key. The best solutions so far [8] either rely on non-standard knowledge assumptions, or they do not achieve compactness in the number of input ciphertexts.

In contrast with FHE schemes, we do have practical realizations of linearly homomorphic encryption (LHE) that are proven CCA1-secure under standard assumptions like the Decision Diffie-Hellman or the Composite Residuosity assumption. However, the most efficient candidates are obtained by downgrading an IND-CCA2 scheme and removing the components that ensure non-malleability. For example, the lite Cramer-Shoup cryptosystem [17] (dubbed "lite-CS" in the following) and its natural Composite Residuosity analogue [18,7] are obtained by eliminating the hash function that allows tying ciphertext components together. As a consequence, the resulting CCA1 schemes are not significantly more efficient than their CCA2 counterparts: The ciphertext size is identical and the number of modular exponentiations is almost the same as well. Yet, one would intuitively expect a larger efficiency gap between constructions satisfying the two security notions, at least if the homomorphic property is required in the CCA1 setting. A natural question to ask is then the following: *Can we achieve IND-CCA1 security in homomorphic schemes in a more efficient way than we can get IND-CCA2 security under the same standard assumption in the standard model?* Recently, Schäge [47] showed the existence of strong barriers to the provable IND-CCA1 security of Elgamal [26] and Paillier [44]. The question nevertheless remains open for some other LHE schemes.

As previously alluded to, currently known CCA1 FHE constructions [8] either rely on non-falsifiable assumptions or suffer from a lack of compactness (i.e., the size of evaluated ciphertexts grows with the number of input ciphertexts). Even for degree-2 functions, we are not aware of a CCA1 extension of the Boneh-Goh-Nissim cryptosystem [5] in the literature. A related question is: *How efficiently and under which assumptions can we obtain CCA1 security beyond linear homomorphic operations?*

In this paper, we provide positive answers to the above questions and prove the CCA1 security of several schemes where encrypted messages are - either naturally for correctness reasons or for the sake of getting the proof to work - restricted to live in a sparse subset of larger ambient space.

1.1 Our Contributions

Revisiting Damgård's Elgamal. We first provide a new proof of CCA1 security for the additively homomorphic version of Damgård's Elgamal encryption scheme [20]. Our proof stands in the standard model under the standard DDH assumption. Previous proofs under the same assumption were given for hybrid variants of the scheme [22,36], which are not additively homomorphic.

So far and although the scheme has been around for 3 decades, its additively homomorphic variant was only known to be secure under the knowledge

of assumption [20] or under other interactive assumption [33,23,40,3] that are much stronger than DDH. A result of Lipmaa [40] further shows that proving the CCA1 security of Damgård's Elgamal under the sole DDH assumption is impossible. However, the result of [40] only holds when messages are encoded as group elements. Here, we bypass the impossibility result of [40] by leveraging the fact that messages are encoded as integers in a polynomial-size interval.

Our security proof relies on hash proof systems [17,18] like earlier results under DDH [22,36]. A crucial difference is that these rely on additional secret-key components (like authenticated secret-key encryption and key derivation functions) which help the security reduction reject invalid ciphertexts, but break the linear homomorphism of the scheme. In contrast, we prove the CCA1 security of the original homomorphic system without introducing any additional component. To do this, we realize the ciphertext-integrity check by testing whether the decrypted plaintext belongs to a sufficiently small interval. Using the properties of hash proof systems [17,18], we show that malformed ciphertexts are very unlikely to decrypt to a plaintext in the legitimate message space. In Elgamal-type homomorphic schemes, the need to restrict the message space to a small interval is usually viewed as a limitation. Here, we use it as a leverage for CCA1 security.

We thus prove (33 years after its invention) that Damgård's Elgamal is actually the most efficient DDH-based CCA1 additively homomorphic candidate. With only 3 group elements per ciphertext and 3 modular exponentiations, it improves upon the lite-CS system [17] by 25% in terms of ciphertext size and encryption cost. These improvements are amplified when the scheme is used as a building block for the homomorphic evaluation of degree-2 functions.

A CCA1 BGN Cryptosystem. As a second contribution, we build a CCA1 variant of the Boneh-Goh-Nissim (BGN) cryptosystem [5], which allows evaluating 2-DNF formulas on encrypted data. Our construction is obtained by adapting Freeman's BGN [29] - which is itself an adaptation of [5] to prime-order groups - and relies on the same Symmetric eXternal Diffie-Hellman (SXDH) assumption in pairing-friendly groups. The main difference with [29] is that, while Freeman's multiplication algorithm uses a pairing to compute a tensor product of Elgamal encryptions, we compute a tensor product of ciphertexts in Damgård's Elgamal.

We note that an IND-CCA1 BGN-type encryption scheme could also be obtained from the lite Cramer-Shoup scheme. The corresponding multiplication algorithm would use the pairing in a similar way to compute a tensor product of lite-CS ciphertexts. However, the multiplication algorithm would output depth-1 ciphertexts consisting of 16 elements of the target group $\mathbb{G}_T$. By using Damgård's Elgamal instead, we obtain depth-1 ciphertexts comprised of only 9 elements of $\mathbb{G}_T$. Also, the multiplication algorithm only computes 9 pairings instead of 16. We thus reduce the cost of moving from IND-CPA to IND-CCA1 security from a factor 4 down to a factor 2.25. Indeed, under the SXDH assumption, Freeman's CPA-secure BGN [29] requires 4 elements of $\mathbb{G}_T$ in depth-1 ciphertexts. In the CCA1 setting, a lite-CS-based construction would cost 16 elements of $\mathbb{G}_T$ (vs 9 elements of $\mathbb{G}_T$ in our variant of BGN).

Extensions to Large Message Spaces. Our technique of exploiting restricted message spaces is not limited to discrete-logarithm-based schemes where plaintexts live in a polynomial-size interval. The legitimate message space only needs to be sufficiently sparse in a larger additive group over which the homomorphism is defined. Modulo slight adjustments, the technique carries over to the Elgamal-Paillier combination suggested by Camenisch and Shoup [7]. By modifying their scheme and artificially[1] forcing λ message bits to be zeroes, we obtain a CCA1 variant with shorter ciphertexts than in any previous variant of the same scheme. In terms of bandwidth, we thus lose λ plaintext bits but this still leaves room for exponentially large messages since the modulus N must have a super-linear length in the security parameter λ anyway.

For this construction, our proof additionally relies on an assumption (introduced by Hofheinz in [34]) saying that, on input of an RSA modulus $N = pq$, it is infeasible to compute a Paillier encryption $c = (1 + N)^m \cdot r^N \bmod N^2$ of a message $m \in \mathbb{Z}_N$ that is not co-prime with N. While this assumption is less standard than the Composite Residuosity assumption (DCR), it has been standing for a decade. To our knowledge, we thus provide the first proof of CCA1 security under non-tautological assumptions for a variant of the additively homomorphic Elgamal-Paillier where ciphertexts are as short as in the CPA case.

It is actually possible to avoid the non-standard assumption and only rely on the standard DCR assumption at the expense of further reducing the size of the message space $[0, B]$ from $B = \lfloor 2^{-\lambda} \cdot N \rfloor$ down to $B < \lfloor 2^{-\lambda} \cdot \min(p, q) \rfloor$. Concretely, if N has the recommended 3072-bit size at the 128-bit security level, the DCR assumption alone suffices to imply CCA1 security as long as legitimate plaintexts are restricted to live in a 1408-bit interval.

We finally adapt the latter construction to the Castagnos-Laguillaumie (CL) system [9], which relies on class groups of imaginary quadratic fields. The CL scheme resembles Elgamal-Paillier (in particular, it involves a hidden-order group that factors as a product of a DDH-hard subgroup and another subgroup where computing discrete logarithms is easy) and similarly allows encrypting exponentially large messages. One advantage over Elgamal-Paillier is that its message space can have prime order, thus ensuring that all encrypted messages are invertible. Even when messages are smaller than the plaintext modulus by only λ bits, this allows us to only rely on a classical subgroup membership assumption.

While our adaptations of Elgamal-Paillier and Castagnos-Laguillaumie are not quite identical to the original systems, they feature a similar efficiency. Indeed, the ciphertext size is exactly the same as in the underlying IND-CPA schemes and the message space remains exponentially large. CCA1 security is achieved by the simple action of checking that the plaintext belongs to the proper interval upon decryption (and restricting the number of homomorphic operations since correctness is only guaranteed as long as the resulting plaintext dwells in the legal range) rather than by introducing additional computations.

[1] We say "artificially" because the restriction is not imposed by the functionality of the scheme, but is rather an artifact of the proof of CCA1 security.

1.2 Technical Overview

In Damgård's Elgamal (called DEG for short in the next sections), ciphertexts are of the form $(c_0, c_1, c_2) = (g^m \cdot h^r, g_1^r, g_2^r)$, where the public key contains generators $(g, g_1, g_2 = g_1^x, h = g_1^y) \in \mathbb{G}^4$ for a secret key $(x, y) \in \mathbb{Z}_p^2$, where p is the order of a cyclic group $\mathbb{G}$. Decryption proceeds by testing if $c_2 = c_1^x$ and, if so, outputting the discrete logarithm $m = \log_g(c_0 \cdot c_1^{-y})$. Intuitively, $c_2 = g_2^r$ serves as a "proof" that the sender knows the encryption exponent r and ensures a form of plaintext awareness [4]. Several works [22,36] consider a variant where the public key is computed as $h = g_1^{-x_1} g_2^{-x_2}$, for a secret key $(x_1, x_2) \in \mathbb{Z}_p^2$, and decryption proceeds by computing $m = \log_g(c_0 \cdot c_1^{x_1} c_2^{x_2})$. We also consider this variant but our security proof easily extends to the original variant.

Like [22,36], our proof builds on ideas from Cramer-Shoup [17]. A difference is that [22,36] rely on a symmetric authenticated encryption scheme to ensure ciphertext integrity in the same way as in the Kurosawa-Desmedt cryptosystem [37]. They encrypt m using an authenticated secret-key encryption where the secret key K derived from h^{-r} using a key derivation function. An invalid ciphertext $(c_0, c_1, c_2) = (\mathsf{E}_K(m), g_1^r, g_2^{r'})$ (where $r \neq r'$) is rejected because $c_0 = \mathsf{E}_K(m)$ can only be valid if the adversary manages to forge a valid c_0 for a random secret key $K = \mathrm{KDF}(c_1^{x_1} \cdot c_2^{x_2})$ since $c_1^{x_1} \cdot c_2^{x_2} = h^{-r} \cdot g_2^{(r'-r) \cdot x_2}$ is uniformly random in the adversary's view (of which x_2 is independent). Here, we do not introduce any authenticated encryption scheme or key derivation function. Instead, we use an integrity check based on the smallness of decrypted messages. For an invalid ciphertext $(c_0, c_1, c_2) = (c_0, g_1^r, g_2^{r'})$, we still use the property that $c_0 \cdot c_1^{x_1} \cdot c_2^{x_2} = c_0 \cdot h^{-r} \cdot g_2^{(r'-r) \cdot x_2}$ is uniformly distributed in $\mathbb{G}$ conditionally on the adversary's view. However, we use the property that $\log_g(c_0 \cdot c_1^{x_1} \cdot c_2^{x_2})$ lands in the polynomial-size message space $\mathcal{M} = [0, B]$ with negligible probability $(B+1)/p$. Our sanity check is just to verify that there exists an integer $m \in [0, B]$ such that $c_0 \cdot c_1^{x_1} \cdot c_2^{x_2} = g^m$, which we prove sufficient to ensure CCA1 security.

In the challenge phase, we further exploit the entropy of x_2 to encrypt the challenge message m_ρ, where $\rho \in \{0, 1\}$ is a random bit. To do this, we need the conditional distribution of x_2 to be uniform in order to use $(r' - r) \cdot x_2$ as a one-time pad. One issue is that each rejected decryption query allows the adversary to eliminate one candidate x_2, so that x_2 is only uniform in a set of size $p - Q$ after Q queries. To preserve the uniformity of x_2 until the challenge phase, we use a sequence of games where we gradually replace the real decryption oracle by an oracle that does not use x_2, but only $\log_{g_1}(g_2)$ and $\log_{g_1}(h)$ (which reveal nothing about x_2 since they are completely determined by the public key).

In order to obtain a BGN-type cryptosystem, we generate a DEG key pair in each source group of a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ and we compute fresh ciphertexts (at depth 0) by computing a DEG ciphertext in both $\mathbb{G}_1$ and $\mathbb{G}_2$. In order to multiply two depth-0 ciphertexts, we follow Freeman's approach [29] which uses the pairing to compute a tensor product between the $\mathbb{G}_1$-component of one input ciphertext and the $\mathbb{G}_2$-component of the second input ciphertext. In the decryption algorithm, the only sanity check is to reject ciphertexts that do not decrypt to a message in a pre-determined polynomial-size interval $[0, B]$.

The proof of CCA1 security follows the blueprint of our proof for DEG. It first uses the SXDH assumption to tamper with the distribution of the challenge ciphertext. Next, it gradually modifies the decryption oracle and reaches a game where all decryption queries are answered using a secret key that is information-theoretically determined by the public key. At this point, since the real secret key is not used until the challenge phase, we can make the most of its entropy to perfectly hide the encrypted message.

Our adaptation of Elgamal-Paillier is similar to the CPA-secure variant of [7]. It has ciphertexts of the form $(c_0, c_1) = (g^r \bmod N^2, (1+N)^m \cdot h^r \bmod N^2)$, where $N = pq$ is an RSA modulus, g generates the subgroup of $2N$-th residues in $\mathbb{Z}^*_{N^2}$ and $h = g^{2x} \bmod N^2$. The difference with [7] is that the decryptor rejects all ciphertexts where $c_1 \cdot c_0^{-2x} \neq (1+N)^m \bmod N^2$ for any $m \in [0, B]$.

The security proof has the same skeleton as its DEG analogue with the difference that it relies on an additional assumption to eliminate a corner case. It first invokes the Composite Residuosity (DCR) assumption to replace c_0 by a random quadratic residue in the challenge phase. Then, it uses a sub-sequence games where the decryption oracle is gradually modified to reveal nothing about $x \bmod N$. To do this, we need to rely on an assumption introduced in [34] so as to argue that the adversary cannot create a ciphertext (c_0, c_1) where c_0 is of the form $c_0 = (1+N)^{\alpha_0} \cdot r_0^N \bmod N^2$, for some $r_0 \in \mathbb{Z}^*_N$ and $\alpha_0 \in \mathbb{Z}_N$ such that $\gcd(\alpha_0, N) \neq 1$. The reason is that, assuming that the first $i-1$ queries did not reveal anything about $x \bmod N$, we need to make sure that $c_1 \cdot c_0^{-2x} \bmod N^2$ has a uniformly distributed component in the subgroup $\langle 1+N \rangle$ (which would not be the case if $\gcd(\alpha_0, N) \neq 1$ at the i-th query).

In the case of the Castagnos-Laguillaumie cryptosystem [9,10], we do not need any non-standard assumption since messages can be defined modulo a prime p, so that there is no way to encrypt a non-invertible element.

1.3 Related Work

The first CCA1 LHE system can be traced back to the work of Damgård [20], where its security was shown under the knowledge-of-exponent assumption. Bellare and Palacio [4] proved it plaintext-aware (PA1) under a slightly different knowledge assumption. Gjøsteen subsequently proved [33] its CCA1 security under a more classical assumption stating that DDH remains hard when the distinguisher is given access to a static[2] DDH oracle before receiving its DDH challenge. This assumption is nevertheless interactive and still non-standard.

Gjøsteen [33] gave a Composite Residuosity instantiation of his general gap subgroup membership problem,[3] but the resulting assumption is also interactive (similar assumptions were considered in [3]). In contrast, our modification of Elgamal-Paillier only requires non-interactive assumptions.

Lipmaa [40] showed that an interactive assumption, which is equivalent to the

[2] Here, "static" means that one input of the DDH oracle is a fixed group element, as in the Strong Diffie-Hellman assumption of [1].

[3] The corresponding DCR analogue of DEG is similar to the IND-CPA variant of [7].

CCA1 security of DEG, is provably not implied by DDH. As discussed before, our proof of CCA1 security does not contradict his impossibility result because the latter assumes[4] that messages are group elements. The recent work of Schäge [47] rules out the provable CCA1 security of Elgamal and Paillier/Damgård-Jurik under standard assumptions via a wide class of reductions. Our proofs evade his impossibility results, which apply to homomorphic schemes where valid ciphertexts are publicly recognizable.

We note that the results of [40] imply the security of DEG and Elgamal in the generic group model [48]. In the algebraic group model [30], Elgamal was shown CCA1-secure under a q-type assumption when messages are group elements.

In the standard model, LHE constructions based on standard assumptions (e.g., DDH, DCR, QR) were - sometimes implicitly - proposed in [17,18,7]. Unfortunately, their complexity is roughly the same as that of their CCA2 siblings. For example, lite-CS ciphertexts are as long as Cramer-Shoup ciphertexts.

In the context of FHE, Loftus *et al.* [41] introduced a knowledge assumption in lattices in order to build a CCA1 candidate by ensuring a form of plaintext awareness (PA1). They also considered the notion of ciphertext-verification attacks [35] (where the adversary has access to ciphertext-validity oracle after the challenge phase) and showed that their scheme is vulnerable to such attacks. They finally gave concrete CCA1 key-recovery attacks against earlier somewhat homomorphic schemes. Several CCA1 attacks [41,51,14,19,28] were subsequently reported against leveled FHE schemes.

Canetti *et al.* [8] described three constructions of FHE schemes with CCA1 security. Their leveled realizations based on identity-based multi-key FHE and obfuscation only provide compactness with respect to the size of evaluated circuits, but not with respect to the number of input ciphertexts. Their third construction applies the Naor-Yung paradigm [43] using composable zero-knowledge SNARKs, which inherently rely on non-falsifiable assumptions. Similar constructions based on multi-key FHE were concurrently proposed in [50].

Canetti *et al.* [8, Appendix A] also suggested a different approach (i.e., which does not proceed by downgrading a CCA2 system so as to make it homomorphic) of building CCA1 LHE schemes from a form of single-key functional encryption for linear functions. They gave a concrete instantiation from DDH. Still, the resulting LHE is less efficient than the lite Cramer-Shoup construction.

To our knowledge, all CCA1 homomorphic encryption schemes so far are either limited to perform homomorphic additions, or they suffer from some lack of compactness, or they rely on non-falsifiable assumptions. Even among schemes that can only evaluate depth-one circuits, we are not aware of any prior mention of a CCA1 variant of BGN [5].

Li and Micciancio [39] introduced a strengthened notion of IND-CPA security (called IND-CPA^D security) where the adversary is given access to a decryption

[4] More precisely, the second direction in the proof of equivalence of [40, Lemma 1] does no longer work if DEG ciphertexts are of the form $(g^m \cdot h^r, g_1^r, g_2^r)$, for a small $m \in [0, B]$, since the reduction cannot simulate the computational oracle of the assumption using the decryption oracle for DEG.

oracle that decrypts honestly generated ciphertexts. While their notion does not imply CCA1 security (as the decryption oracle cannot be queried on maliciously generated ciphertexts), it is not achieved by approximate homomorphic encryption schemes like [16]. Recently, it was shown [13,15] that exact FHE schemes may fail to provide IND-CPA^D security as well when their correctness is not guaranteed to hold with overwhelming probability. Viand *et al.* [49] considered IND-CPA^D security as an ingredient allowing to build verifiable FHE schemes satisfying a form of IND-CCA1 security using SNARKs.

Several works [45,6,12,27,2,42] considered security notions that ensure a meaningful form of non-malleable/CCA2 security in homomorphic encryption. Prabhakaran and Rosulek [45] and Chase *et al.* [12] considered notions of HCCA security and controlled malleability (CM-CCA), respectively, which are restricted to enable unary transformations on encrypted data. Boneh *et al.* [6] suggested a notion of targeted malleability and realized it using the Naor-Yung paradigm and succinct arguments. While their constructions are also CCA1-secure, they are not known to be instantiable in the standard model without knowledge assumptions. Targeted malleability [6] and HCCA/CM-CCA security [45,12] both require non-malleability [25] beyond a set of allowed transformations with the difference that the former is applicable to FHE.

Emura *et al.* [27] introduced the concept of keyed-homomorphic encryption, where homomorphic operations can only be performed using a dedicated evaluation key which, by itself, does not enable decryption. Their model requires IND-CCA2 security when the evaluation key is secret and preserves IND-CCA1 security otherwise. They proposed keyed-homomorphic LHE realizations based on various standard assumptions. Lai *et al.* [38] considered the design of keyed-homomorphic FHE schemes using indistinguishability obfuscation. Sato *et al.* [46] showed how to dispense with obfuscation, but their approach still requires a CCA1 FHE scheme to begin with.

Akavia *et al.* [2] introduced the notion of functional re-encryption security (funcCPA), which they proved strictly stronger than CPA security and yet achievable by FHE schemes. In their model, the adversary has access to an oracle that inputs a ciphertext C and a function f in some family, and returns a fresh encryption of $f(\mathsf{Decrypt}(SK, C))$. Their notion is not known to imply CCA1 security as the adversary never obtains any decryption. Dodis *et al.* [24] showed that it is closer to CPA than to CCA security by showing a black-box construction of (non-homomorphic) funcCPA-secure encryption scheme from a CPA-secure one. In the same paper [24, Lemma 8], they showed that funcCPA security (in a possibly stronger form dubbed funcCPA+) is actually implied by CCA1 security. The schemes discussed in this paper are thus also secure the funcCPA+ sense.

Manulis and Nguyen [42] suggested a notion of vCCA2 security that strengthens CCA1 security by means of a verification mechanism allowing to link an evaluated ciphertext to its input ciphertexts. In their notion, post-challenge decryption queries are only allowed on ciphertexts that are verifiably not derivatives of the challenge ciphertext. They showed that vCCA2 is achievable by combining a simulation-extractable SNARK and a standard FHE.

2 Background

NOTATIONS. When S is a finite set, we sometimes denote by $U(S)$ the uniform distribution on S. We also denote by $x \xleftarrow{R} S$ the action of sampling x from $U(S)$.

2.1 Hardness Assumptions

We first recall the definition of the Decision Diffie-Hellman problem.

Definition 1. *In a cyclic group $\mathbb{G}$ of prime order p, the* **Decision Diffie-Hellman Problem** *(DDH) is to distinguish the distributions (g, g^a, g^b, g^{ab}) and (g, g^a, g^b, g^c), with $a, b, c \xleftarrow{R} \mathbb{Z}_p$. The Decision Diffie-Hellman assumption is the intractability of DDH for any PPT distinguisher.*

In the case of Damgård's Elgamal, we can rely on the DDH assumption in standard groups without a bilinear map. Our BGN-type encryption scheme uses asymmetric bilinear maps $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ over groups of prime order p. We will work in Type-3 asymmetric pairings, where we have $\mathbb{G}_1 \neq \mathbb{G}_2$ so as to allow the DDH assumption to hold in both $\mathbb{G}_1$ and $\mathbb{G}_2$. This assumption is called *Symmetric eXternal Diffie-Hellman* (SXDH) assumption and it implies that no isomorphism between $\mathbb{G}_1$ and $\mathbb{G}_2$ be efficiently computable.

In our Paillier-based construction, we need two assumptions. The first one is the standard Composite Residuosity assumption, which is recalled below.

Definition 2 ([44]). *Let $N = pq$ for primes p, q. The* **Decision Composite Residuosity** *(DCR) assumption states that the distributions $\{x = z^N \bmod N^2 \mid z \xleftarrow{R} \mathbb{Z}_N^*\}$ and $\{x \mid x \xleftarrow{R} \mathbb{Z}_{N^2}^*\}$ are computationally indistinguishable.*

The second assumption,[5] which was introduced by Hofheinz [34], is less standard and posits the hardness of computing a Paillier encryption of a non-zero integer that is not co-prime to N without knowing the factorization of N.

In Definition 3, $\mathcal{K}$ is a probabilistic algorithm that inputs a security parameter λ and outputs an RSA modulus $N = pq$ for large primes p, q. In addition, $\mathcal{D}(\cdot)$ denotes the deterministic decryption algorithm of Paillier's cryptosystem, which takes as input an element $c \in \mathbb{Z}_{N^2}^*$ and outputs the unique $\alpha \in \mathbb{Z}_N$ such that $c = (1+N)^\alpha \cdot \beta^N \bmod N^2$ for some $\beta \in \mathbb{Z}_N^*$.

Definition 3 ([34, Assumption 4.4]). *The* **Composite Non-Invertibility** *assumption says that, for any PPT algorithm $\mathcal{A}$, we have*

$$\mathbf{Adv}_{\mathcal{A}}^{\text{noninv}}(\lambda) := \Pr\left[N \leftarrow \mathcal{K}(1^\lambda),\ c \leftarrow \mathcal{A}(1^\lambda, N)\ :\ c \in \mathbb{Z}_{N^2}^*\ \wedge\ 1 < \gcd(\mathcal{D}(c), N) < N\right] \leq \mathsf{negl}(\lambda),$$

where $\mathcal{K}$ (resp. $\mathcal{D}(\cdot)$) denotes the key generation algorithm (resp. decryption function) of Paillier's cryptosystem.

[5] To our knowledge, this assumption was not given a name so far. We thus chose to call it "Composite Non-Invertibility" assumption.

2.2 Additively Homomorphic and BGN-type Encryption

A depth-one (a.k.a. BGN-type) homomorphic cryptosystem [5] is a public-key encryption scheme allowing to non-interactively compute 2-DNF formulas on encrypted data. For convenience, the syntax that we use allows the encryptor to directly create depth-1 ciphertexts that are distributed as outputs of the multiplication algorithm. Such a scheme is a tuple $(\mathsf{Keygen}, \mathsf{Encrypt}_0, \mathsf{Encrypt}_1, \mathsf{Decrypt}_0, \mathsf{Decrypt}_1, \mathsf{Add}_0, \mathsf{Add}_1, \mathsf{Multiply})$ of efficient algorithms with the following syntax:

Keygen: is a randomized algorithm that inputs a security parameter 1^λ and a message length 1^t. It outputs a key pair (PK, SK). The public key PK contains the description of a plaintext space $\mathcal{M}$ and ciphertext spaces $\mathcal{CT}_d$ for each depth $d \in \{0,1\}$.

$\mathsf{Encrypt}_d$ $(d \in \{0,1\})$: is a randomized algorithm that takes as input a plaintext $m \in \mathcal{M}$ and a public key PK. It outputs a fresh ciphertext $C \in \mathcal{CT}_d$ at depth $d \in \{0,1\}$.

$\mathsf{Decrypt}_d$ $(d \in \{0,1\})$: is a deterministic algorithm that inputs a secret key SK and a depth-d ciphertext $C \in \mathcal{CT}_d$. It outputs either a plaintext $m \in \mathcal{M}$ or a rejection symbol $\perp$ indicating an invalid C.

Add_d $(d \in \{0,1\})$: is a (possibly randomized) algorithm that inputs a public key PK and two depth-d ciphertexts $C_1, C_2 \in \mathcal{CT}_d$. It outputs a new depth-d ciphertext $C \in \mathcal{CT}_d$.

Multiply: is a (possibly randomized) algorithm that takes as input a public key PK and ciphertexts $C_1, C_2 \in \mathcal{CT}_0$. It outputs a depth-1 ciphertext $C \in \mathcal{CT}_1$.

We will consider schemes that are circuit-private in the sense that evaluated ciphertexts are statistically indistinguishable from fresh ciphertexts (at the same depth) encrypting the same message. We recall the formal definitions of correctness and circuit-privacy in Supplementary Material A.1.

3 Proof of CCA1 Security for Damgård's Additively Homomorphic Elgamal under the DDH assumption

We give a new proof of CCA1 security for the additively homomorphic variant of Damgård's Elgamal encryption scheme, which is recalled hereunder.

3.1 The DEG Scheme

$\mathsf{Keygen}(1^\lambda, 1^t)$: Given a security parameter $\lambda \in \mathbb{N}$ and a desired message length $t = O(\log \lambda)$,

1. Choose a cyclic group $\mathbb{G}$ of prime order $p > 2^{l(\lambda)}$, for some polynomial $l : \mathbb{N} \to \mathbb{N}$. Choose generators $g, g_1, g_2 \xleftarrow{R} \mathbb{G}$.
2. Choose $x_1, x_2 \xleftarrow{R} \mathbb{Z}_p$ and compute $h = g_1^{-x_1} g_2^{-x_2}$.

Return the key pair (PK, SK) consisting of $SK := (x_1, x_2) \in \mathbb{Z}_p^2$ and

$$PK := \left(\mathbb{G},\ g,\ g_1,\ g_2,\ h,\ B = 2^t\right),$$

where B defines the message space $\mathcal{M} = [0, B]$.

Encrypt(PK, m)**:** Given a public key PK and a message m consisting of an integer in the interval $\mathcal{M} = [0, B]$, do the following:

1. Choose $r \xleftarrow{R} \mathbb{Z}_p$ and compute

$$c_0 = g^m \cdot h^r \qquad c_1 = g_1^r \qquad c_2 = g_2^r$$

2. Output the ciphertext $C = (c_0, c_1, c_2)$.

Decrypt(SK, C)**:** Given $SK = (x_1, x_2) \in \mathbb{Z}_p^2$ and $C = (c_0, c_1, c_2)$,

1. Compute $M = c_0 \cdot c_1^{x_1} \cdot c_2^{x_2}$.
2. If there exists an integer $m \in [0, B]$ such that $M = g^m$, return m. Otherwise, return $\perp$.

3.2 New Security Proof

Theorem 1. *The scheme provides IND-CCA1 security in the standard model under the DDH assumption. For any CCA1 adversary $\mathcal{A}$ making at most Q decryption queries, there is a DDH distinguisher $\mathcal{B}$ such that*

$$\mathbf{Adv}_{\mathcal{A}}^{\mathrm{CCA1}}(\lambda) \leq \mathbf{Adv}_{\mathcal{B}}^{\mathrm{DDH}}(\lambda) + \frac{1 + Q \cdot (B+1)}{2^\lambda}$$

Proof. The proof considers a sequence of hybrid games. For each index i, we call W_i the event that the adversary $\mathcal{A}$ wins in Game_i.

Game_0: This is the real IND-CCA1 security game. In the challenge phase, $\mathcal{A}$ chooses messages $m_0, m_1 \in [0, B]$ and obtains a challenge ciphertext

$$c_1^\star = g_1^r, \qquad c_2^\star = g_2^r, \qquad c_0^\star = g^{m_\rho} \cdot h^r,$$

where $\rho \xleftarrow{R} \{0, 1\}$ is a random bit chosen by the challenger. When the adversary $\mathcal{A}$ halts, it outputs a bit $\rho' \in \{0, 1\}$ and wins if $\rho' = \rho$. Its advantage is $\mathbf{Adv}_{\mathcal{A}}^{\mathrm{CCA1}}(\lambda) := |\Pr[W_0] - 1/2|$.

Game_1: We modify the generation of the challenge ciphertext. In the challenge phase, $\mathcal{A}$ outputs $m_0, m_1 \in [0, B]$. The challenge is computed as

$$c_1^\star = g_1^r, \qquad c_2^\star = g_2^r, \qquad c_0^\star = g^{m_\rho} \cdot c_1^{\star -x_1} \cdot c_2^{\star -x_2}.$$

for a random $r \xleftarrow{R} \mathbb{Z}_p$. Clearly, Game_1 is identical to Game_0 from the adversary's view and we have $\Pr[W_1] = \Pr[W_0]$.

Game_2: We modify the distribution of the challenge, which is now computed as

$$c_1^\star = g_1^r, \qquad c_2^\star = g_2^{r'}, \qquad c_0^\star = g^{m_\rho} \cdot c_1^{\star -x_1} \cdot c_2^{\star -x_2}.$$

for random $r \xleftarrow{R} \mathbb{Z}_p$, $r' \xleftarrow{R} \mathbb{Z}_p \setminus \{r\}$. Under the DDH assumption, Game_2 is indistinguishable from Game_1 and $|\Pr[W_2] - \Pr[W_1]| \leq \mathbf{Adv}^{\mathrm{DDH}}(\lambda) + \frac{1}{p}$.

We now consider a sub-sequence of games where we gradually modify the decryption oracle. For convenience, $\mathsf{Game}_{2.0}$ is defined as being identical to Game_2.

$\mathsf{Game}_{2.i}$ $(1 \leq i \leq Q)$: In these games, we modify the key generation phase where the challenger computes $g_2 = g_1^\omega$ for a random $\omega \xleftarrow{R} \mathbb{Z}_p$ and defines an alternative secret key $SK' := (\omega, z)$, where $z = x_1 + \omega \cdot x_2$ is such that $h = g_1^{-z}$. In the first i decryption queries, the challenger uses the following modified decryption algorithm:

$\mathsf{Decrypt}'(SK', C)$: On input of $C = (c_0, c_1, c_2)$, return $\perp$ if $c_2 \neq c_1^\omega$. Otherwise (i.e., if $c_2 = c_1^\omega$), compute $M = c_0 \cdot c_1^z$. If there exists $m \in [0, B]$ such that $M = g^m$, return m. Otherwise, return $\perp$.

The last $Q - i$ decryption queries are answered by running the original decryption algorithm as in Game_2.

In Lemma 1, we prove that $|\Pr[W_{2.i}] - \Pr[W_{2.(i-1)}]| \leq (B+1)/2^\lambda$, so that the two games are statistically close.

Game_3: This game is identical to $\mathsf{Game}_{2.Q}$ except that the challenge ciphertext is computed by choosing $r, u \xleftarrow{R} \mathbb{Z}_p$, $r' \xleftarrow{R} \mathbb{Z}_p \setminus \{r\}$ and computing

$$c_1^\star = g_1^r, \qquad c_2^\star = g_2^{r'}, \qquad c_0^\star = g^u \cdot h^r$$

To see that Game_3 is perfectly indistinguishable from $\mathsf{Game}_{2.Q}$, we note that $\mathsf{Game}_{2.Q}$ computes $(c_1^\star, c_2^\star) = (g_1^r, g_2^{r'})$ and

$$c_0^\star = g^{m_\rho} \cdot c_1^{-x_1} \cdot c_2^{-x_2} = g^{m_\rho} \cdot h^r \cdot g_2^{x_2 \cdot (r - r')}$$

where x_2 perfectly hides m_b. Indeed, the real secret key $SK = (x_1, x_2)$ is not used at all in $\mathsf{Game}_{2.Q}$ since all decryption queries are answered using $SK' = (\omega, z)$. So, the adversary's view is the same as if the choice of x_2 was postponed to the challenge phase at which point the challenger would sample $x_2 \xleftarrow{R} \mathbb{Z}_p$ and define $SK = (z - \omega \cdot x_2, x_2)$. Hence, $\Pr[W_3] = \Pr[W_{2.Q}]$.

In Game_3, the challenge ciphertext is completely independent of m_ρ and we have $\Pr[W_3] = 1/2$. By the triangle inequality, we obtain the stated upper bound for $\mathbf{Adv}_{\mathcal{A}}^{\mathrm{CCA1}}(\lambda) = |\Pr[W_0] - 1/2|$. □

Lemma 1. *For each $i \in [0, Q]$, $\mathsf{Game}_{2.i}$ is statistically indistinguishable from $\mathsf{Game}_{2.(i-1)}$. We have $|\Pr[W_{2.i}] - \Pr[W_{2.(i-1)}]| \leq (B+1)/2^\lambda$.*

Proof. For each $i \in [Q]$, $\mathsf{Game}_{2.i}$ only differ from $\mathsf{Game}_{2.(i-1)}$ in the i-th decryption query, where $\mathsf{Game}_{2.i}$ uses algorithm $\mathsf{Decrypt}'$ whereas $\mathsf{Game}_{2.(i-1)}$ uses the original $\mathsf{Decrypt}$ algorithm. So, the two games are identical from the adversary's view unless the i-th decryption query involves a ciphertext that would not have been rejected in $\mathsf{Game}_{2.(i-1)}$ but gets rejected in $\mathsf{Game}_{2.i}$.

For any well-formed ciphertext $C = (c_0, c_1, c_2)$ (i.e., such that $c_2 = c_1^{\omega}$), both decryption algorithms output the same result. On a ciphertext (c_0, c_1, c_2) such that $(c_1, c_2) = (g_1^{r_1}, g_2^{r_2})$ for distinct $r_1 \neq r_2$, $\mathsf{Decrypt}'$ always returns $\perp$ and we just have to assess the probability that $\mathsf{Decrypt}$ does not output $\perp$ as well. For such a ciphertext, $\mathsf{Decrypt}$ computes

$$\begin{aligned} M = c_0 \cdot c_1^{x_1} \cdot c_2^{x_2} &= c_0 \cdot (g_1^{r_1})^{x_1} \cdot (g_2^{r_2})^{x_2} \\ &= c_0 \cdot (g_1^{x_1} \cdot g_2^{x_2})^{r_1} \cdot g_2^{(r_2 - r_1) \cdot x_2} = c_0 \cdot h^{-r_1} \cdot g_2^{(r_2 - r_1) \cdot x_2} \end{aligned} \tag{1}$$

In the right-hand-side member of (1), we note that $c_0 \cdot h^{-r_1}$ is completely determined by the ciphertext. However, we claim that $g_2^{(r_2 - r_1) \cdot x_2}$ is uniformly distributed in $\mathcal{A}$'s view since the first $i-1$ decryption queries are answered using $\mathsf{Decrypt}'$. In $\mathsf{Game}_{2.(i-1)}$, $\mathcal{A}$'s view is exactly the same as if the challenger was postponing the choice of x_2 until the moment where $\mathcal{A}$ has submitted its i-th decryption query. In more details, the challenger could equivalently generate $g_2 = g_1^{\omega}$ and $h = g_1^{-z}$ in the key generation phase and answer the first $i-1$ decryption queries using $SK' = (\omega, z)$. Only at the moment where $\mathcal{A}$ sends its i-th decryption query $C = (c_0, c_1, c_2)$, the challenger would sample $x_2 \xleftarrow{R} \mathbb{Z}_p$ uniformly and define the real secret key $SK = (x_1, x_2) = (z - \omega \cdot x_2, x_2)$ to be used in the last $Q - i + 1$ decryption queries. Therefore, if the i-th decryption query $C = (c_0, c_1, c_2)$ involves a ciphertext such that $r_1 \neq r_2$, the product in the rightmost member of (1) is uniformly distributed in $\mathbb{G}$ since x_2 is drawn uniformly in $\mathbb{Z}_p$ *after* the choice of c_0, $r_1 = \log_{g_1}(c_1)$ and $r_2 = \log_{g_2}(c_2)$ by $\mathcal{A}$. The probability that $\log_g(M)$ lands in the polynomial-size interval $[0, B]$ is thus at most $(B+1)/p < (B+1)/2^{\lambda}$. Therefore, except with probability smaller than $(B+1)/2^{\lambda}$, $\mathsf{Decrypt}$ returns $\perp$ and agrees with $\mathsf{Decrypt}'$. □

The proof of Theorem 1 extends to the original variant of Damgård's Elgamal [20], where the modified decryption algorithm $\mathsf{Decrypt}'$ (which answers all decryption queries in $\mathsf{Game}_{2.Q}$) is used in the real scheme, and not just in the security proof. The details are given in Supplementary Material B.

4 An IND-CCA1 BGN Cryptosystem

At a high level, the scheme is reminiscent of the LWE-based construction of Gentry, Halevi and Vaikuntanathan [31]. However, the security proof is very different and relies on the Cramer-Shoup techniques, which are not known to provide chosen-ciphertext security under the LWE assumption. For this reason, we are currently unable to obtain a CCA1 variant of [31].

4.1 Description

To simplify the presentation of our variant of BGN, we will use the implicit representation of group elements. For a matrix $\mathbf{M}$ over $\mathbb{Z}_p$, we will use the notations $[\mathbf{M}]_1 = g_1^{\mathbf{M}}$, $[\mathbf{M}]_2 = g_2^{\mathbf{M}}$ and $[\mathbf{M}]_T = e(g_1, g_2)^{\mathbf{M}}$, where $g_1 \in \mathbb{G}_1$ and $g_2 \in \mathbb{G}_2$ are pre-determined generators. As in [29], we rely on the pairing to compute matrix products in the exponent. For matrices $\mathbf{A}, \mathbf{B}$ of compatible dimensions, the pairing operation between matrices of group elements $g_1^{\mathbf{A}}$ and $g_2^{\mathbf{B}}$ is written $[\mathbf{A} \cdot \mathbf{B}]_T = [\mathbf{A}]_1 \cdot [\mathbf{B}]_2 = e(g_1^{\mathbf{A}}, g_2^{\mathbf{B}})$.

Keygen$(1^\lambda, 1^t)$**:** Given a security parameter $\lambda \in \mathbb{N}$ and a desired message length $t = O(\log \lambda)$,

1. Choose asymmetric pairing-friendly cyclic groups $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ of prime order $p > 2^{l(\lambda)}$, for some polynomial $l : \mathbb{N} \to \mathbb{N}$. Choose generators $g_1 \in \mathbb{G}_1$ and $g_2 \in \mathbb{G}_2$.
2. Choose vectors $\boldsymbol{a} = (a_1, a_2, a_3)^\top \xleftarrow{R} (\mathbb{Z}_p^*)^3$, $\boldsymbol{b} = (b_1, b_2, b_3)^\top \xleftarrow{R} (\mathbb{Z}_p^*)^3$ and compute
$$[\boldsymbol{a}]_1 := g_1^{\boldsymbol{a}} \in \mathbb{G}_1^3, \qquad [\boldsymbol{b}]_2 := g_2^{\boldsymbol{b}} \in \mathbb{G}_2^3$$
3. Choose $x_1, x_2, y_1, y_2 \xleftarrow{R} \mathbb{Z}_p$ uniformly conditionally on
$$\langle (x_1, x_2, 1), \boldsymbol{a} \rangle = 0 \qquad \text{and} \qquad \langle (y_1, y_2, 1), \boldsymbol{b} \rangle = 0. \tag{2}$$
Define $\boldsymbol{x} = (x_1, x_2, 1)^\top$ and $\boldsymbol{y} = (y_1, y_2, 1)^\top$.

Return the key pair (PK, SK) where
$$PK := \big((\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T),\ g_1,\ g_2,\ [\boldsymbol{a}]_1,\ [\boldsymbol{b}]_2,\ B = 2^t\big)$$
and $SK := (\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{Z}_p^3 \times \mathbb{Z}_p^3$.

Encrypt$_d(PK, m)$**:** To encrypt $m \in \mathcal{M} = [0, B]$ at depth $d \in \{0, 1\}$, do the following:

1. If $d = 0$, choose $r, s \xleftarrow{R} \mathbb{Z}_p$ and compute
$$[\boldsymbol{c}]_1 = r \cdot \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}_1 + m \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}_1, \qquad [\boldsymbol{d}]_2 = s \cdot \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}_2 + m \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}_2$$
Then, return $C = \big([\boldsymbol{c}]_1, [\boldsymbol{d}]_2\big) \in \mathbb{G}_1^3 \times \mathbb{G}_2^3$.
2. If $d = 1$, choose $r_1, r_2, r_3 \xleftarrow{R} \mathbb{Z}_p$, $s_1, s_2, s_3 \xleftarrow{R} \mathbb{Z}_p$ and compute
$$[\mathbf{c}]_T = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}_1 \cdot \begin{bmatrix} r_1 & r_2 & r_3 \end{bmatrix}_2 + \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix}_1 \cdot \begin{bmatrix} b_1 & b_2 & b_3 \end{bmatrix}_2 + m \cdot \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}_T \tag{3}$$
and return $C = [\mathbf{c}]_T \in \mathbb{G}_T^{3\times 3}$.

Decrypt$_d(SK, C)$**:** To decrypt a ciphertext C at level $d \in \{0,1\}$ using the secret key $SK = (\boldsymbol{x}, \boldsymbol{y}) = \big((x_1, x_2, 1), (y_1, y_2, 1)\big)^\top \in \mathbb{Z}_p^3 \times \mathbb{Z}_p^3$, do the following:

1. If $d = 0$, return $\perp$ if $C \notin \mathbb{G}_1^3 \times \mathbb{G}_2^3$. Otherwise, let $C = \big([\boldsymbol{c}]_1, [\boldsymbol{d}]_2\big)$ and compute

$$[M]_1 = \boldsymbol{x}^\top \cdot [\boldsymbol{c}]_1 \in \mathbb{G}_1, \qquad [N]_2 = \boldsymbol{y}^\top \cdot [\boldsymbol{d}]_1 \in \mathbb{G}_2$$

   If there exists $m \in [0, B]$ such that $[M]_1 = m \cdot [1]_1$ and $[N]_2 = m \cdot [1]_2$, return m. Otherwise, return $\perp$.
2. If $d = 1$, return $\perp$ if $C \notin \mathbb{G}_T^{3\times 3}$. Otherwise (i.e., if $C = [\mathbf{c}]_T \in \mathbb{G}_T^{3\times 3}$), compute

$$[M]_T = \boldsymbol{x}^\top \cdot [\mathbf{c}]_T \cdot \boldsymbol{y} = \begin{bmatrix} x_1 & x_2 & 1 \end{bmatrix}^\top \cdot [\mathbf{c}]_T \cdot \begin{bmatrix} y_1 \\ y_2 \\ 1 \end{bmatrix}$$

   If there exists an integer $m \in [0, B]$ such that $[M]_T = m \cdot [1]_T$, return m. Otherwise, return $\perp$.

Multiply(PK, C_1, C_2)**:** Given two ciphertexts $C_1 = \big([\boldsymbol{c}_1]_1, [\boldsymbol{d}_1]_2\big) \in \mathbb{G}_1^3 \times \mathbb{G}_2^3$ and $C_2 = \big([\boldsymbol{c}_2]_1, [\boldsymbol{d}_2]_2\big) \in \mathbb{G}_1^3 \times \mathbb{G}_2^3$ at depth 0, return $\perp$ if C_1 and C_2 do not parse properly. Otherwise, choose $r_1, r_2, r_3, s_1, s_2, s_3 \xleftarrow{R} \mathbb{Z}_p$, compute

$$[\mathbf{c}]_T = [\boldsymbol{c}_1]_1 \cdot [\boldsymbol{d}_2^\top]_2 + \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}_1 \cdot \begin{bmatrix} r_1 & r_2 & r_3 \end{bmatrix}_2 + \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix}_1 \cdot \begin{bmatrix} b_1 & b_2 & b_3 \end{bmatrix}_2 \tag{4}$$

and return $[\mathbf{c}]_T \in \mathbb{G}_T^{3\times 3}$.

Add$_d(PK, C_1, C_2)$**:** Given the public key PK and two ciphertexts C_1 and C_2, do the following:

1. If $d = 0$, return $\perp$ if C_1 and C_2 cannot be parsed as $C_1 = \big([\boldsymbol{c}_1]_1, [\boldsymbol{d}_1]_2\big) \in \mathbb{G}_1^3 \times \mathbb{G}_2^3$ and $C_2 = \big([\boldsymbol{c}_2]_1, [\boldsymbol{d}_2]_2\big) \in \mathbb{G}_1^3 \times \mathbb{G}_2^3$. Otherwise, choose $r, s \xleftarrow{R} \mathbb{Z}_p$ and compute

$$C' = \big([\boldsymbol{c}']_1, [\boldsymbol{d}']_2\big) = \big([\boldsymbol{c}_1]_1 + [\boldsymbol{c}_2]_1, [\boldsymbol{d}_1]_2 + [\boldsymbol{d}_2]_2\big)$$

   Then, return $C = \big([\boldsymbol{c}]_1, [\boldsymbol{d}]_2\big) \in \mathbb{G}_1^3 \times \mathbb{G}_2^3$ where

$$[\boldsymbol{c}]_1 = [\boldsymbol{c}']_1 + r \cdot \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}_1 \qquad [\boldsymbol{d}]_2 = [\boldsymbol{d}']_2 + s \cdot \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}_1$$

2. If $d = 1$, return $\perp$ if C_1 and C_2 cannot be parsed as $[\mathbf{c}_1]_T \in \mathbb{G}_T^{3\times 3}$ and $[\mathbf{c}_2]_T \in \mathbb{G}_T^{3\times 3}$. Otherwise, choose $r_1, r_2, r_3, s_1, s_2, s_3 \xleftarrow{R} \mathbb{Z}_p$ and compute

$$[\mathbf{c}]_T = [\boldsymbol{c}_1]_1 + [\boldsymbol{c}_2]_1 + \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}_1 \cdot \begin{bmatrix} r_1 & r_2 & r_3 \end{bmatrix}_2 + \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix}_1 \cdot \begin{bmatrix} b_1 & b_2 & b_3 \end{bmatrix}_2 \tag{5}$$

   and return $[\mathbf{c}]_T \in \mathbb{G}_T^{3\times 3}$.

Correctness. The correctness of decryption algorithms is straightforward due to orthogonality conditions (2). The correctness of addition and multiplication algorithms is shown in Supplementary Material A.2.

Efficiency. The multiplication algorithm only requires 9 pairing evaluations since the second and third terms of (4) are computable as exponentiations in $\mathbb{G}_T$ using the randomizers $r_1, r_2, r_3, s_1, s_2, s_3 \in \mathbb{Z}_p$. For the same reason, the second and third terms of (5) are computable using exponentiations and encrypting a ciphertext at depth 1 does not require any pairing evaluation in (3).

4.2 Security

We now prove security under the SXDH assumption. In the proof of Theorem 2, we assume that the challenge ciphertext is always computed at depth 0 since one can always turn a depth-0 ciphertext into a depth-1 encryption of the same message. This can be done by performing a multiplication with a depth-0 ciphertext encrypting 1 and re-randomizing the resulting depth-1 ciphertext so as to have an element of $\mathbb{G}_T^{3\times 3}$ that is distributed as a fresh depth-1 ciphertext.

Theorem 2. *The scheme provides IND-CCA1 security in the standard model under the SXDH assumption. For any CCA1 adversary $\mathcal{A}$ making at most Q decryption queries, there exist distinguishers $\mathcal{B}_1$ and $\mathcal{B}_2$ against the DDH assumption in $\mathbb{G}_1$ and $\mathbb{G}_2$ such that*

$$\mathbf{Adv}_{\mathcal{A}}^{\mathrm{CCA1}}(\lambda) \leq \mathbf{Adv}_{\mathcal{B}_1}^{\mathrm{DDH}_1}(\lambda) + \mathbf{Adv}_{\mathcal{B}_2}^{\mathrm{DDH}_2}(\lambda) + \tfrac{1+Q\cdot(B+1)}{2^{\lambda-1}}$$

Proof. The proof considers a sequence of games. For each i, we call W_i the event that the adversary wins and successfully guesses the challenger's bit in Game_i.

Game_0: This is the real IND-CCA1 game. In the challenge phase, $\mathcal{A}$ chooses messages $m_0, m_1 \in [0, B]$ and can choose to obtain a challenge ciphertext at depth 0 or at depth 1. At depth 0, $\mathcal{A}$ obtains a ciphertext of the form

$$[\boldsymbol{c}^\star]_1 = r \cdot \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}_1 + m_\rho \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}_1, \qquad [\boldsymbol{d}^\star]_2 = s \cdot \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}_2 + m_\rho \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}_2$$

for random $r, s \xleftarrow{R} \mathbb{Z}$, where $\rho \xleftarrow{R} \{0,1\}$ is a random bit chosen by the challenger. At depth 1, the challenge ciphertext is of the form

$$[\mathbf{c}^\star]_T = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}_1 \cdot \begin{bmatrix} r_1 & r_2 & r_3 \end{bmatrix}_2 + \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix}_1 \cdot \begin{bmatrix} b_1 & b_2 & b_3 \end{bmatrix}_2 + m_\rho \cdot \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}_T$$

for random $r_1, r_2, r_3, s_1, s_2, s_3 \xleftarrow{R} \mathbb{Z}_p$. When $\mathcal{A}$ terminates, it outputs a bit $\rho' \in \{0,1\}$ and wins if $\rho' = \rho$. Its advantage is $\mathbf{Adv}_{\mathcal{A}}^{\mathrm{CCA1}} := |\Pr[W_0] - 1/2|$.

Game_1: We modify the generation of the challenge ciphertext. In the challenge phase, the adversary outputs $m_0, m_1 \in [0, B]$. The challenger first computes

$$\begin{bmatrix} c_1^\star \\ c_2^\star \end{bmatrix}_1 = r \cdot \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}_1, \qquad \begin{bmatrix} d_1^\star \\ d_2^\star \end{bmatrix}_2 = s \cdot \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}_2 \tag{6}$$

where $r, s \xleftarrow{R} \mathbb{Z}_p$. A depth-0 challenge ciphertext is then computed by setting

$$\begin{aligned} [c_3^\star]_1 &= -x_1 \cdot [c_1^\star]_1 - x_2 \cdot [c_2^\star]_1 + m_\rho \cdot [1]_1, \\ [d_3^\star]_2 &= -y_1 \cdot [d_1^\star]_2 - y_2 \cdot [d_2^\star]_2 + m_\rho \cdot [1]_2 \end{aligned} \tag{7}$$

and defining $[\boldsymbol{c}^\star]_1 = [\ c_1^\star \mid c_2^\star \mid c_3^\star\ ]_1$, $[\boldsymbol{d}^\star]_2 = [\ d_1^\star \mid d_2^\star \mid d_3^\star\ ]_2$. This change is only conceptual since $[\boldsymbol{c}^\star]_1$ and $[\boldsymbol{d}^\star]_2$ have the same value as in Game_0. Clearly, Game_1 is identical to Game_0 from $\mathcal{A}$'s view and $\Pr[W_1] = \Pr[W_0]$.

Game_2: We modify the distribution of the challenge ciphertexts. Instead of computing depth-0 ciphertexts as per (6), the challenger now computes

$$\begin{bmatrix} c_1^\star \\ c_2^\star \end{bmatrix}_1 = \begin{bmatrix} r \cdot a_1 \\ r' \cdot a_2 \end{bmatrix}_1, \qquad \begin{bmatrix} d_1^\star \\ d_2^\star \end{bmatrix}_2 = \begin{bmatrix} s \cdot b_1 \\ s' \cdot b_2 \end{bmatrix}_2 \tag{8}$$

where $r, s \xleftarrow{R} \mathbb{Z}_p$ and $r' \xleftarrow{R} \mathbb{Z}_p \setminus \{r\}$, $s' \xleftarrow{R} \mathbb{Z}_p \setminus \{s\}$. Then, $[c_3^\star]_1$ and $[d_3^\star]_2$ are computed from $[c_1^\star \mid c_2^\star]_1$ and $[d_1^\star \mid d_2^\star]_2$ as in (7). Since the only change is the distribution of $[\ c_1^\star \mid c_2^\star\ ]_1$ and $[\ d_1^\star \mid d_2^\star\ ]_2$, a simple reduction shows that Game_2 is indistinguishable from Game_1 as long as the DDH assumption holds in both $\mathbb{G}_1$ and $\mathbb{G}_2$. We have

$$|\Pr[W_2] - \Pr[W_1]| \leq \mathbf{Adv}^{\mathrm{DDH}_1}(\lambda) + \mathbf{Adv}^{\mathrm{DDH}_2}(\lambda) + \tfrac{2}{p}.$$

We now consider a sub-sequence of games where we gradually modify the decryption oracle. For convenience, $\mathsf{Game}_{2.0}$ is defined as being identical to Game_2.

$\mathsf{Game}_{2.i}$ $(1 \leq i \leq Q)$: In these games, we modify the key generation phase where the challenger only chooses the vectors $\boldsymbol{a} = (a_1, a_2, a_3), \boldsymbol{b} = (b_1, b_2, b_3) \xleftarrow{R} \mathbb{Z}_p^3$. The choice of $\boldsymbol{x} = (x_1, x_2, 1)$, $\boldsymbol{y} = (y_1, y_2, 1)$ such that $\langle \boldsymbol{x}, \boldsymbol{a} \rangle = 0$, $\langle \boldsymbol{y}, \boldsymbol{b} \rangle = 0$ is postponed until the i-th decryption query (note that we can afford to explicitly use the discrete logarithms $\boldsymbol{a}, \boldsymbol{b}$ since we are done with the SXDH assumption at this point). At the outset of the game, the challenge defines an alternative secret key $SK' = (\mathbf{Z}, \mathbf{W}) \in (\mathbb{Z}_p^{2\times 3})^2$ consisting of matrices

$$\mathbf{Z} = \begin{pmatrix} a_3 & 0 & -a_1 \\ 0 & a_3 & -a_2 \end{pmatrix}, \qquad \mathbf{W} = \begin{pmatrix} b_3 & 0 & -b_1 \\ 0 & b_3 & -b_2 \end{pmatrix},$$

which form bases of the linear subspaces $\boldsymbol{a}^\perp = \{\boldsymbol{c} \in \mathbb{Z}_p^3 \mid \langle \boldsymbol{c}, \boldsymbol{a} \rangle = 0\}$ and $\boldsymbol{b}^\perp = \{\boldsymbol{d} \in \mathbb{Z}_p^3 \mid \langle \boldsymbol{d}, \boldsymbol{b} \rangle = 0\}$, respectively. In the first i decryption queries, the challenger uses the following modified decryption algorithms:

$\mathsf{Decrypt}'_0(SK', C)$: Given a depth-0 $C = ([\boldsymbol{c}]_1, [\boldsymbol{d}]_2) \in \mathbb{G}_1^3 \times \mathbb{G}_2^3$, compute

$$[\mathbf{M}]_1 = \mathbf{Z} \cdot [\mathbf{c}]_1 \in \mathbb{G}_1^2, \qquad [\mathbf{M}]_2 = \mathbf{W} \cdot [\mathbf{d}]_2 \in \mathbb{G}_2^2$$

If there exists $m \in [0, B]$ such that

$$[\mathbf{M}]_1 = -m \cdot \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}_1, \qquad [\mathbf{M}]_2 = -m \cdot \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}_2 \tag{9}$$

return m. Otherwise, return $\perp$.

$\mathsf{Decrypt}'_1(SK', C)$: On input of a depth-1 $C = [\mathbf{c}]_T \in \mathbb{G}_T^{3\times 3}$, compute

$$[\mathbf{M}]_T = \mathbf{Z} \cdot [\mathbf{c}]_T \cdot \mathbf{W}^\top \in \mathbb{G}_T^{2\times 2}.$$

If there exists $m \in [0, B]$ such that

$$[\mathbf{M}]_T = m \cdot \begin{bmatrix} a_1 \cdot b_1 & a_1 \cdot b_2 \\ a_2 \cdot b_1 & a_2 \cdot b_2 \end{bmatrix}_T \tag{10}$$

return m. Otherwise, return $\perp$.

At the $(i+1)$-th query, the challenger samples $\boldsymbol{x} = (x_1, x_2, 1)$, $\boldsymbol{y} = (y_1, y_2, 1)$ for random $x_1, x_2, y_1, y_2 \xleftarrow{R} \mathbb{Z}_p$ satisfying the constraint $\langle \boldsymbol{x}, \boldsymbol{a} \rangle = \langle \boldsymbol{y}, \boldsymbol{a} \rangle = 0$. The last $Q - i$ decryption queries are then answered by running the original decryption algorithm as in Game_2.

Lemma 2 shows that $|\Pr[W_{2.i}] - \Pr[W_{2.(i-1)}]| \leq 2(B+1)/2^\lambda$, thus proving the statistical indistinguishability of the two games.

Game_3: This game is identical to $\mathsf{Game}_{2.Q}$ except that the challenge ciphertext is now generated without using the plaintext m_ρ at all. A depth-0 ciphertext is simulated by computing

$$\begin{bmatrix} c_1^\star \\ c_2^\star \end{bmatrix}_1 = \begin{bmatrix} r \cdot a_1 \\ r' \cdot a_2 \end{bmatrix}_1, \qquad \begin{bmatrix} d_1^\star \\ d_2^\star \end{bmatrix}_2 = \begin{bmatrix} s \cdot b_1 \\ s' \cdot b_2 \end{bmatrix}_2 \tag{11}$$

for random $r, s \xleftarrow{R} \mathbb{Z}_p$ and $r' \xleftarrow{R} \mathbb{Z}_p \setminus \{r\}$, $s' \xleftarrow{R} \mathbb{Z}_p \setminus \{s\}$, and then

$$[c_3^\star]_1 = r \cdot [a_3]_1 + u \cdot [1]_1, \qquad [d_3^\star]_2 = s \cdot [b_3]_2 + v \cdot [1]_2 \tag{12}$$

with $u, v \xleftarrow{R} \mathbb{Z}_p$. We claim that Game_3 is perfectly indistinguishable from $\mathsf{Game}_{2.Q}$. Indeed, the real secret key $SK = \big((x_1, x_2, 1) \mid (y_1, y_2, 1)\big)^\top$ is not used at all until the challenge phase in $\mathsf{Game}_{2.Q}$ since all decryption queries are answered using $SK' = (\mathbf{Z}, \mathbf{W})$. The choice of x_2 and y_2 is thus postponed until the challenge phase when the challenger samples $x_2, y_2 \xleftarrow{R} \mathbb{Z}_p$, defines

$$x_1 = (-a_2 \cdot x_2 - a_3)/a_1, \qquad y_1 = (-b_2 \cdot y_2 - b_3)/b_1$$

and uses $SK = \big((x_1, x_2, 1) \mid (y_1, y_2, 1)\big)^\top$ to compute a depth-0 challenge ciphertext as per (11)-(12). In this case, $\mathsf{Game}_{2.Q}$ computes a challenge ciphertext whose $\mathbb{G}_1$ and $\mathbb{G}_2$ components are of the form

$$\begin{aligned}
[c_1^\star]_1 &= r \cdot [a_1]_1 \\
[c_2^\star]_1 &= r \cdot [a_2]_1 + (r' - r) \cdot [a_2]_1 \\
[c_3^\star]_1 &= -x_1 \cdot [c_1^\star]_1 - x_2 \cdot [c_2^\star]_1 + m_\rho \cdot [1]_1 \\
&= r \cdot [a_3]_1 + x_2 \cdot (r - r') \cdot [a_2]_1 + m_\rho \cdot [1]_1
\end{aligned} \tag{13}$$

and

$$\begin{aligned}
[d_1^\star]_2 &= s \cdot [b_1]_2 \\
[d_2^\star]_2 &= s \cdot [b_2]_2 + (s' - s) \cdot [b_2]_2 \\
[d_3^\star]_2 &= -y_1 \cdot [d_1^\star]_2 - y_2 \cdot [d_2^\star]_2 + m_\rho \cdot [1]_1 \\
&= s \cdot [b_3]_2 + y_2 \cdot (s - s') \cdot [b_2]_2 + m_\rho \cdot [1]_2,
\end{aligned} \tag{14}$$

respectively. Since $r' \neq r$, $s' \neq s$ and $x_2, y_2 \sim U(\mathbb{Z}_p)$ are independent of $\mathcal{A}$'s view until the challenge phase, the ciphertext distributions (13) and (14) are identical to the challenge ciphertext distribution (11)-(12), which is used in Game_3. This implies $\Pr[W_3] = \Pr[W_{2.Q}]$.

In Game_3, the challenge ciphertext is totally independent of m_ρ and we have $\Pr[W_3] = 1/2$. By combining all inequalities, we obtain the stated upper bound for $\mathbf{Adv}_{\mathcal{A}}^{\mathrm{CCA1}}(\lambda) = |\Pr[W_0] - 1/2|$. □

Lemma 2. *For each $i \in [0, Q]$, $\mathsf{Game}_{2.i}$ is statistically indistinguishable from $\mathsf{Game}_{2.(i-1)}$. We have $|\Pr[W_{2.i}] - \Pr[W_{2.(i-1)}]| \leq 2(B+1)/2^\lambda$.*

Proof. For each $i \in [Q]$, $\mathsf{Game}_{2.i}$ only differ from $\mathsf{Game}_{2.(i-1)}$ in the i-th decryption query, where $\mathsf{Game}_{2.i}$ uses algorithm $\mathsf{Decrypt}'_0$ and $\mathsf{Decrypt}'_1$ while $\mathsf{Game}_{2.(i-1)}$ uses the original decryption algorithm. The two games are thus identical from $\mathcal{A}$'s view unless the i-th decryption query involves a ciphertext that would be rejected in one of the two games and not in the other one. We first remark that any ciphertext that is not rejected by $\mathsf{Game}_{2.i}$ is not rejected by $\mathsf{Game}_{2.(i-1)}$ either. Indeed, the real secret key $SK = \big((x_1, x_2, 1), (y_1, y_2, 1)\big)$ is obtained by taking linear combination of the rows of $\mathbf{Z}$ and $\mathbf{W}$ for random coefficients $\alpha_1, \alpha_2, \beta_1, \beta_2 \in \mathbb{Z}_p$ such that $\alpha_1 \cdot a_1 + \alpha_2 \cdot a_2 = -1$ and $\beta_1 \cdot b_1 + \beta_2 \cdot b_2 = -1$. Namely,

$$(x_1, x_2, 1) = (\alpha_1, \alpha_2) \cdot \begin{pmatrix} a_3 & 0 & -a_1 \\ 0 & a_3 & -a_2 \end{pmatrix}, \qquad (y_1, y_2, 1) = (\beta_1, \beta_2) \cdot \begin{pmatrix} b_3 & 0 & -b_1 \\ 0 & b_3 & -b_2 \end{pmatrix}$$

This implies that any depth-1 ciphertext that is accepted by $\mathsf{Decrypt}'_1$ in $\mathsf{Game}_{2.i}$ is also accepted by $\mathsf{Decrypt}$ in $\mathsf{Game}_{2.(i-1)}$. We are left with assessing the probability that, in $\mathsf{Game}_{2.i}$, $\mathsf{Decrypt}'_1$ rejects a ciphertext that would not have been rejected in $\mathsf{Game}_{2.(i-1)}$.

If the i-th query is a depth-1 ciphertext $[\mathbf{c}]_T \in \mathbb{G}^{3\times 3}$, let $[\mathbf{M}]_T \in \mathbb{G}^{2\times 2}$ the

matrix obtained by $\mathsf{Decrypt}_1'$ in (10). We note that $\mathsf{Game}_{2.i}$ rejects if there exists no $m \in [0, B]$ such that

$$[\mathbf{M}]_T := \mathbf{Z} \cdot [\mathbf{c}]_T \cdot \mathbf{W}^\top = m \cdot \begin{bmatrix} a_1 \cdot b_1 & a_1 \cdot b_2 \\ a_2 \cdot b_1 & a_2 \cdot b_2 \end{bmatrix}_T \tag{15}$$

while $\mathsf{Game}_{2.(i-1)}$ only rejects if there exists no $m \in [0, B]$ such that

$$[M]_T := (\alpha_1, \alpha_2)^\top \cdot \underbrace{\left(\mathbf{Z} \cdot [\mathbf{c}]_T \cdot \mathbf{W}^\top\right)}_{= [\mathbf{M}]_T} \cdot \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} = m \cdot [1]_T \tag{16}$$

For the matrix $[\mathbf{M}]_T \in \mathbb{G}_T^{2\times 2}$ obtained by $\mathsf{Decrypt}_1'$ at the i-th decryption query in $\mathsf{Game}_{2.i}$, we assess the probability of the event bad that there exists $m \in [0, B]$ satisfying (16) given that no such m satisfies (15). Let us parse $[\mathbf{M}]_T$ as

$$[\mathbf{M}]_T := \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix}_T$$

Let bad_m the event bad occurs for a fixed $m \in [0, B]$. Then, from (16), we see that bad_m implies

$$\alpha_1 \cdot (m_1 \cdot \beta_1 + m_2 \cdot \beta_2) + \alpha_2 \cdot (m_3 \cdot \beta_1 + m_4 \cdot \beta_2) = m \tag{17}$$

However, $(\alpha_1, \alpha_2) \in \mathbb{Z}_p^2$ is only chosen after $\mathcal{A}$ has submitted its i-th decryption query (which determines $[\mathbf{M}]_T$) and it is sampled uniformly in the affine subspace

$$\{(\alpha_1, \alpha_2) \in \mathbb{Z}_p^2 \mid \alpha_1 \cdot a_1 + \alpha_2 \cdot a_2 = -1\}.$$

Therefore the probability that (17) holds is at most $1/p$ if the vector

$$\boldsymbol{m}_\beta \triangleq (m_1 \cdot \beta_1 + m_2 \cdot \beta_2, m_3 \cdot \beta_1 + m_4 \cdot \beta_2)$$

is linearly independent of (a_1, a_2) since, in this case, there is only one pair $(\bar{\alpha}_1, \bar{\alpha}_2)$ satisfying both (17) and $\bar{\alpha}_1 \cdot a_1 + \bar{\alpha}_2 \cdot a_2 = -1$. If we now assume that $\boldsymbol{m}_\beta$ is co-linear with (a_1, a_2), then bad_m implies

$$(m_1 \cdot \beta_1 + m_2 \cdot \beta_2, m_3 \cdot \beta_1 + m_4 \cdot \beta_2) = -m \cdot (a_1, a_2) \tag{18}$$

If we call $\mathsf{bad}_{m,\beta}$ the event that the equalities (18) hold, we have shown that $\Pr[\mathsf{bad}_m \mid \neg\mathsf{bad}_{m,\beta}] = 1/p$ since $\neg\mathsf{bad}_{m,\beta}$ implies that $\boldsymbol{m}_\beta$ is linearly independent of (a_1, a_2). We now claim that $\Pr[\mathsf{bad}_{m,\beta}] \leq 1/p$ unless

$$\begin{pmatrix} m_1 & m_2 \\ m_3 & m_4 \end{pmatrix} = m \cdot \begin{pmatrix} a_1 \cdot b_1 & a_1 \cdot b_2 \\ a_2 \cdot b_1 & a_2 \cdot b_2 \end{pmatrix}, \tag{19}$$

which would contradict the hypothesis that (15) does not hold.

To see this, we first note that, if $\mathbf{M} \in \mathbb{Z}_p^{2\times 2}$ has full rank, this is straightforward since there is only one pair $(\bar{\beta}_1, \bar{\beta}_2)$ satisfying (18) and a random pair of the

affine subspace $\{(\beta_1, \beta_2) \in \mathbb{Z}_p^2 \mid \beta_1 \cdot b_1 + \beta_2 \cdot b_2 = -1\}$ satisfies $(\beta_1, \beta_2) = (\bar{\beta}_1, \bar{\beta}_2)$ with probability $1/p$ (recall that (β_1, β_2) is only sampled after $\mathcal{A}$ has submitted its i-th decryption query, which defines $\mathbf{M}$). If $\mathbf{M}$ has rank 1 but (m_1, m_2) and (m_3, m_4) are linearly independent of (b_1, b_2), there is similarly at most one pair $(\bar{\beta}_1, \bar{\beta}_2)$ satisfying (18) and $b_1 \cdot \beta_1 + b_2 \cdot \beta_2 = -1$ and this pair is sampled with probability $1/p$. Finally, if $\mathbf{M}$ has rank 1 but (m_1, m_2) and (m_3, m_4) are co-linear with (b_1, b_2), we can only have (18) and $b_1 \cdot \beta_1 + b_2 \cdot \beta_2 = -1$ if (19) holds.

Consequently, if the i-th decryption query is a depth-1 ciphertext, we have $\Pr[\mathsf{bad}_m] \leq \Pr[\mathsf{bad}_m \mid \neg\mathsf{bad}_{m,\beta}] + \Pr[\mathsf{bad}_{m,\beta}] \leq 2/p$.

We obtain a similar bound in the simpler case of depth-0 ciphertexts. Such a ciphertext creates a discrepancy between the two games at the i-th decryption query if there exists no $m \in [0, B]$ satisfying (9) but there exists $m \in [0, B]$ such that $(x_1, x_2, 1) \cdot [\mathbf{c}]_1 = m \cdot [1]_1$ and $(y_1, y_2, 1) \cdot [\mathbf{d}]_2 = m \cdot [1]_2$. We only look at the decryption operations in $\mathbb{G}_1$ since the treatment of ciphertext components in $\mathbb{G}_2$ is similar. Let us parse $[\mathbf{M}]_1 = \mathbf{Z} \cdot [\mathbf{c}]_1$ as $[\mathbf{M}]_1 = \left[\begin{smallmatrix} m_1 \\ m_2 \end{smallmatrix}\right]_1$. If (m_1, m_2) is linearly independent of (a_1, a_2), for any fixed $m \in [0, B]$, there is a unique pair (α_1, α_2) satisfying $\alpha_1 m_1 + \alpha_2 m_2 = m$ and $\alpha_1 a_1 + \alpha_2 a_2 = -1$ and this pair is sampled with probability $1/p$. If (m_1, m_2) and (a_1, a_2) are colinear, we can only have the equalities $\alpha_1 m_1 + \alpha_2 m_2 = m$ and $\alpha_1 a_1 + \alpha_2 a_2 = -1$ if $(m_1, m_2) = -m \cdot (a_1, a_2)$, which would contradict the hypothesis that no $m \in [0, B]$ satisfies (9). For depth-0 ciphertexts, we thus find $\Pr[\mathsf{bad}_m] \leq 1/p$.

We finally obtain $\Pr[\mathsf{bad}] \leq \sum_{m \in [0,B]} \Pr[\mathsf{bad}_m] \leq 2(B+1)/p$ by taking a union bound over all possible plaintexts $m \in [0, B]$. □

5 A CCA1 Variant of Elgamal-Paillier

We now show that a slight variant of the Elgamal-Paillier cryptosystem of [7] can also be proven IND-CCA1 while preserving its additive homomorphism.

Besides the Composite Residuosity assumption, the security proof relies on the Composite Non-Invertibility assumption [34] when the message space $[0, B]$ is as large as $B = \lfloor N \cdot 2^{-\lambda} \rfloor$. As of today, we do not know if the Composite Non-Invertibility assumption is strictly necessary or if it is an artifact of the proof. Clearly, an adversary that would be able to break this assumption would also break the CCA1 security of Paillier's original scheme [44]. However, the same implication does not appear to hold in general for the present variant of Elgamal-Paillier. For example, we just need the DCR assumption when $B \approx N^{1/2} \cdot 2^{-\lambda}$.

5.1 Description

As in [7, Section 3], we assume that N is a safe-prime product. The main difference with [7] is that the actual message space is required to be smaller than N by λ bits. Since we usually need a 3072-bit modulus N at the 128-bit security level, we only lose 128 out of 3072 bits (or less than 5%) when we set $\lambda = 128$. Asymptotically, the size of the message space remains super-exponential in λ since RSA moduli have to be of size $\lambda^3/\mathrm{polylog}(\lambda)$ to resist factorization attacks.

Keygen$(1^\lambda, 1^t)$**:** On input of a security parameter $\lambda \in \mathbb{N}$ and a message length $t \in \mathsf{poly}(\lambda)$,

1. Choose a safe-prime product $N = pq$ for large primes $p = 2p' + 1$ and $q = 2q' + 1$ such that $p, q > 2^{l(\lambda)}$, for some polynomial $l : \mathbb{N} \to \mathbb{N}$ satisfying $t + \lambda < 2(l + 1) < \log N$, and where p', q' are also prime.
2. Choose $g \xleftarrow{R} \mathbb{Z}^*_{N^2}$ and $x \xleftarrow{R} [0, N\cdot(N-1)/4]$. Compute $h = g^{4N\cdot x} \bmod N^2$.

Return the key pair (PK, SK) where $SK := x \in \mathbb{Z}$ and

$$PK := (N,\ g,\ h,\ B)\,,$$

where $B = 2^t < \lfloor N \cdot 2^{-\lambda} \rfloor$ is an integer defining $\mathcal{M} = [0, B]$.

Encrypt(PK, m)**:** Given a public key PK and a message $m \in \mathcal{M}$,

1. Choose $r \xleftarrow{R} [0, (N - 1)/4]$ and compute

$$c_0 = g^{2N\cdot r} \bmod N^2 \qquad\qquad c_1 = (1 + N)^m \cdot h^r \bmod N^2$$

2. Output the ciphertext $C = (c_0, c_1)$.

Decrypt(SK, C)**:** Given $SK = x \in \mathbb{Z}$ and $C = (c_0, c_1)$, return $\perp$ if $c_0 \notin \mathbb{Z}^*_{N^2}$ or $c_1 \notin \mathbb{Z}^*_{N^2}$. Otherwise, conduct the following steps:

1. Compute $M = c_1 \cdot c_0^{-2x} \bmod N^2$.
2. If there exists an integer $m \in [0, B]$ such that $M = (1 + N)^m \bmod N^2$, return m. Otherwise, return $\perp$.

Recall that, in the subgroup of $\mathbb{Z}^*_{N^2}$ generated by $1 + N$, computing m from $M = (1 + N)^m \bmod N^2$ is straightforward since $(1 + N)^m = 1 + mN \bmod N^2$.

In terms of computation, the scheme is as efficient as the CPA-secure variant of [7] since the sanity check at step 2 of Decrypt has a negligible impact on the decryption time. This means that CCA1 security comes essentially for free.

5.2 Security

We now prove CCA1 security under the DCR and Composite Non-Invertibility assumptions. The proof relies on the following lemma, of which the proof is standard and omitted.

Lemma 3. *Let integers $\ell, L > 0$ such that $L > \ell$. The statistical distance between the distributions $D_1 = \{x \bmod \ell \mid x \xleftarrow{R} \mathbb{Z}_L\}$ and $D_2 = \{x \xleftarrow{R} \mathbb{Z}_\ell\}$ is bounded by $\Delta\big(D_1, D_2\big) \leq \frac{\min(\ell, 2(L \bmod \ell))}{L}$. In particular, for $L \geq \ell \cdot 2^\lambda$, the statistical distance is at most $2^{-\lambda}$.*

Theorem 3. *The scheme provides IND-CCA1 security in the standard model under the DCR assumption and the Composite Non-Invertibility assumption. For any CCA1 adversary $\mathcal{A}$ making at most Q decryption queries, there exist a DCR distinguisher $\mathcal{B}_1$ and a Composite Non-Invertibility algorithm $\mathcal{B}_2$ such that*

$$\mathbf{Adv}^{\mathrm{CCA1}}_{\mathcal{A}}(\lambda) \leq \mathbf{Adv}^{\mathrm{DCR}}_{\mathcal{B}_1}(\lambda) + Q \cdot \mathbf{Adv}^{\mathrm{noninv}}_{\mathcal{B}_2}(\lambda) + (Q + 1) \cdot 2^{-\lambda+1} \tag{20}$$

Proof. The proof considers a sequence of hybrid games where W_i denotes the event that the adversary wins and outputs $\rho' = \rho$ in Game_i.

Game_0: This is the real IND-CCA1 security game. In the challenge phase, the adversary chooses messages $m_0, m_1 \in [0, B]$ and obtains a challenge

$$c_0^\star = g^{2N \cdot r} \bmod N^2 \qquad\qquad c_1^\star = (1+N)^{m_\rho} \cdot h^r \bmod N^2,$$

where $\rho \xleftarrow{R} \{0,1\}$ is a random bit chosen by the challenger. When the adversary $\mathcal{A}$ halts, it outputs a bit $\rho' \in \{0,1\}$ and wins if $\rho' = \rho$. Its advantage is $\mathbf{Adv}_{\mathcal{A}}^{\mathrm{CCA1}}(\lambda) := |\Pr[W_0] - 1/2|$.

Game_1: We modify the generation of the challenge ciphertext. In the challenge phase, the adversary outputs $m_0, m_1 \in [0, B]$. The challenge ciphertext is

$$c_0^\star = g^{2N \cdot r} \bmod N^2 \qquad\qquad c_1^\star = (1+N)^{m_\rho} \cdot c_0^{\star 2x} \bmod N^2,$$

for a random $r \xleftarrow{R} [0, (N-1)/4]$. Game_1 is identical to Game_0 from $\mathcal{A}$'s view since $c_1^\star$ has the same value either way. We have $\Pr[W_1] = \Pr[W_0]$.

Game_2: We change again the generation of the challenge ciphertext. The challenger now samples a random N-th residue $z = z_0^N \bmod N^2$, for some $z_0 \xleftarrow{R} \mathbb{Z}_N^*$, and computes

$$c_0^\star = z^2 \bmod N^2 \qquad\qquad c_1^\star = (1+N)^{m_\rho} \cdot c_0^{\star 2x} \bmod N^2,$$

We claim that Game_2 is statistically indistinguishable from Game_1 since the distributions of $c_0^\star$ in the two games are statistically close. This follows from the fact that the subgroup of $2N$-th residues in $\mathbb{Z}_{N^2}^*$ is a cyclic group of order $p'q'$, of which $g^{2N} \bmod N^2$ is a generator with overwhelming probability. In Game_2, $c_0^\star$ is thus a sample from the distribution $\{g^{2N \cdot r} \bmod N^2 \mid r \xleftarrow{R} \mathbb{Z}_{p'q'}\}$, which is within distance $2^{-\lambda}$ from $\{g^{2N \cdot r} \bmod N^2 \mid r \xleftarrow{R} [0, (N-1)/4]\}$ by Lemma 3. Therefore, we have $|\Pr[W_2] - \Pr[W_1]| \leq 2^{-\lambda}$.

Game_3: We modify the distribution of the challenge ciphertext, which is now computed by choosing $z \xleftarrow{R} \mathbb{Z}_{N^2}^*$ and computing

$$c_0^\star = z^2 \bmod N^2 \qquad\qquad c_1^\star = (1+N)^{m_\rho} \cdot c_0^{\star 2x} \bmod N^2, \tag{21}$$

The only change w.r.t. Game_2 is the distribution of z, which is no longer an N-th residue. Under the DCR assumption, Game_3 is indistinguishable from Game_2 and we have $|\Pr[W_3] - \Pr[W_2]| \leq \mathbf{Adv}^{\mathrm{DCR}}(\lambda)$.

Game_4: This game is like Game_3 except that the challenger makes use of the factorization of $N = pq$ and rejects all queries $(c_0, c_1) \in (\mathbb{Z}_{N^2}^*)^2$ such that c_0 is a Paillier encryption of an element α such that $1 < \gcd(\alpha, N) < N$. We define bad as the event that $\mathcal{A}$ queries such a ciphertext for decryption. From $\mathcal{A}$'s view, Game_4 is identical to Game_3 until bad happens: i.e., we have $W_3 \wedge \neg\mathsf{bad} \Leftrightarrow W_4 \wedge \neg\mathsf{bad}$. Lemma 4 shows that, under the Composite Non-Invertibility assumption, $\Pr[\mathsf{bad}]$ is negligible and the challenger does not reject a ciphertext that would not have been rejected in Game_3. Concretely, Lemma 4 implies $|\Pr[W_4] - \Pr[W_3]| \leq \Pr[\mathsf{bad}] \leq Q \cdot \mathbf{Adv}_{\mathcal{A}}^{\mathrm{noninv}}(\lambda)$.

Game_5: This game is identical to Game_4 except that the secret key x is initially sampled as $x \xleftarrow{R} \mathbb{Z}_{Np'q'}$. We claim that Game_5 is statistically indistinguishable from Game_4. In Game_4, the challenge ciphertext (21) is of the form

$$\begin{aligned} c_0^\star &= (1+N)^\alpha \cdot g^{2N\cdot\beta} \bmod N^2 \\ c_1^\star &= (1+N)^{m_\rho + 2\alpha\cdot(x \bmod N)} \cdot g^{4N\cdot\beta\cdot(x \bmod p'q')} \bmod N^2, \end{aligned} \tag{22}$$

for some $\alpha \in \mathbb{Z}_N$, $\beta \in \mathbb{Z}_{p'q'}$. By the Chinese Remainder Theorem, $\mathcal{A}$'s view of the secret key x in (22) is completely determined by $x \bmod Np'q'$. The same holds for possibly malformed ciphertexts (c_0, c_1) sent by $\mathcal{A}$ in its decryption queries since the order of $c_0 \in \mathbb{Z}_{N^2}^*$ is at most $\lambda(N^2) = 2Np'q'$ and the decryption oracle computes $M = c_1 \cdot c_0^{-2\cdot x} \bmod N^2$, where $c_0^{-2\cdot x} \bmod N^2$ is completely determined by $x \bmod Np'q'$ and c_0. The distinguishing advantage of $\mathcal{A}$ between Game_4 and Game_5 can then be bounded by the statistical distance between the distributions $\{x \bmod Np'q' \mid x \xleftarrow{R} [0, N\cdot(N-1)/4]\}$ and $\{x \xleftarrow{R} \mathbb{Z}_{Np'q'}\}$, which is smaller than $(p'+q')/p'q' < 2^{-\lambda}$ by Lemma 3. Hence, we have $|\Pr[W_5] - \Pr[W_4]| \leq 2^{-\lambda}$.

We now consider a sub-sequence of games where we gradually modify the decryption oracle. For convenience, $\mathsf{Game}_{5.0}$ is defined as being identical to Game_5.

$\mathsf{Game}_{5.i}$ $(1 \leq i \leq Q)$: In these games, we modify the key generation phase where the challenger initially computes $h = g^{4N\cdot\beta_x}$, for a random $\beta_x \xleftarrow{R} \mathbb{Z}_{p'q'}$, and defines an alternative secret key $SK' := (\beta_x, p, q)$. In the first i decryption queries, the challenger uses the following modified decryption algorithm:

$\mathsf{Decrypt}'(SK', C)$: On input of $C = (c_0, c_1)$, return $\perp$ if $c_0^{2p'q'} \neq 1 \bmod N^2$. Otherwise, compute $M = c_1 \cdot c_0^{-2\beta_x} \bmod N^2$. If there exists $m \in [0, B]$ such that $M = (1+N)^m \bmod N^2$, return m. Otherwise, return $\perp$.

At the $(i+1)$-th decryption query, the challenger samples $\alpha_x \xleftarrow{R} \mathbb{Z}_N$ and defines $x \in \mathbb{Z}_{Np'q'}$ such that $\alpha_x = x \bmod N$ and $\beta_x = x \bmod p'q'$. Then, it uses $SK = x$ to answer the last $Q - i$ decryption queries via the original decryption algorithm as in Game_5.

In Lemma 5, we prove that $|\Pr[W_{5.i}] - \Pr[W_{5.(i-1)}]| \leq 1/2^{\lambda-1}$, so that the two games are statistically close.

Game_6: This game is identical to $\mathsf{Game}_{5.Q}$ except that the challenge ciphertext is computed by sampling $\alpha \xleftarrow{R} \mathbb{Z}_N$, $\beta \xleftarrow{R} \mathbb{Z}_{p'q'}$ and computing

$$\begin{aligned} c_0^\star &= (1+N)^\alpha \cdot g^{2N\cdot\beta} \bmod N^2 \\ c_1^\star &= (1+N)^u \cdot g^{4N\cdot\beta\cdot(x \bmod p'q')} \bmod N^2, \end{aligned} \tag{23}$$

for a random $u \xleftarrow{R} \mathbb{Z}_N$. We claim that Game_6 is statistically indistinguishable from $\mathsf{Game}_{5.Q}$. To see this, we note that all decryption queries are answered using $SK' = x \bmod p'q'$ in $\mathsf{Game}_{5.Q}$. This implies that $x \bmod N$ is perfectly

independent of $\mathcal{A}$'s view until the challenge phase since it is only chosen when $\mathcal{A}$ has declared its challenge messages $m_0, m_1 \in [0, B]$. Now, recall that the challenge ciphertext of $\mathsf{Game}_{5.Q}$ is of the form

$$\begin{aligned} c_0^\star &= (1+N)^\alpha \cdot g^{2N\cdot\beta} \bmod N^{\zeta+1} \\ c_1^\star &= (1+N)^{m_\rho + 2\alpha\cdot(x \bmod N)} \cdot g^{4N\cdot\beta\cdot(x \bmod p'q')} \bmod N^2. \end{aligned} \tag{24}$$

for random $\alpha \xleftarrow{R} \mathbb{Z}_N$, $\beta \xleftarrow{R} \mathbb{Z}_{p'q'}$. With overwhelming probability $\varphi(N)/N \geq 1 - 2^{-\lambda}$, we have $\gcd(\alpha, N) = 1$. Since $\gcd(2, N) = 1$ and given that the distribution of $x \bmod N$ conditionally on $x \bmod p'q'$ is uniform over $\mathbb{Z}_N$, so is the term $m_\rho + 2\alpha\cdot x \bmod N$ in the expression of $c_1^\star$ in (24). This shows that the ciphertext distributions and (23) and (24) are perfectly indistinguishable unless $\gcd(\alpha, N) \neq 1$. Therefore, we have $|\Pr[W_6] - \Pr[W_{5.Q}]| \leq 2^{-\lambda}$.

In Game_6, the challenge ciphertext is perfectly independent of m_ρ, so that $\Pr[W_6] = 1/2$. By combining the above, we obtain the claimed upper bound on the adversary's advantage $\mathbf{Adv}_{\mathcal{A}}^{\mathrm{CCA1}}(\lambda) = |\Pr[W_0] - 1/2|$. □

Lemma 4. *Under the Composite Non-Invertibility assumption,* Game_4 *is indistinguishable from* Game_3 *and we have* $|\Pr[W_4] - \Pr[W_3]| \leq Q \cdot \mathbf{Adv}_{\mathcal{A}}^{\mathrm{noninv}}(\lambda)$

Proof. From $\mathcal{A}$'s view, Game_4 is identical to Game_3 until the event bad that $\mathcal{A}$ makes a decryption query (c_0, c_1) for which $c_0 \in \mathbb{Z}_{N^2}^*$ is a Paillier ciphertext $c_0 = (1+N)^\gamma \cdot \delta^N \bmod N^2$ such that $1 < \gcd(\gamma, N) < N$ (note that this implies $\gamma \neq 0$, so that $c_0^{2p'q'} \neq 1 \bmod N^2$). However, this would contradict the Composite Non-Invertibility assumption. Assuming that bad occurs with non-negligible probability in Game_3, we can build a simple reduction $\mathcal{B}$ that breaks the assumption with probability $\Pr[\mathsf{bad}]/Q$.

Initially, $\mathcal{B}$ receives $N = pq$ from its Composite Non-Invertibility challenger and uses N to generate the key pair (PK, SK) as specified by Game_3 (this can be done without knowing the factorization of N). At the outset of the game, $\mathcal{B}$ draws a random index $i^\star \xleftarrow{R} [Q]$ as a guess for the first occurrence of event bad. Then, $\mathcal{B}$ starts interacting with $\mathcal{A}$ as in Game_3. The first $i^\star - 1$ decryption queries are answered exactly as in Game_3. At the $i^\star$-th query (c_0, c_1), $\mathcal{B}$ halts and sends c_0 to its Composite Non-Invertibility challenger.

By construction, if c_0 is a Paillier encryption of a non-invertible $\gamma \in \mathbb{Z}_N$, then $\mathcal{B}$ succeeds against its challenger. Since the index $i^\star \in [Q]$ is chosen independently of $\mathcal{A}$'s view, it happens to be the index of the first occurrence of bad with probability $\Pr[\mathsf{bad}]/Q$. Since $|\Pr[W_4] - \Pr[W_3]| \leq \Pr[\mathsf{bad}]$, we obtain the stated inequality of the lemma. □

Lemma 5. *For each* $i \in [0, Q]$, $\mathsf{Game}_{5.i}$ *is statistically indistinguishable from* $\mathsf{Game}_{5.(i-1)}$. *We have* $|\Pr[W_{5.i}] - \Pr[W_{5.(i-1)}]| \leq 1/2^{\lambda-1}$.

Proof. The two games are identical from $\mathcal{A}$'s view unless the i-th decryption query involves a ciphertext that gets rejected in $\mathsf{Game}_{5.i}$, but not in $\mathsf{Game}_{5.(i-1)}$.

For any ciphertext $C = (c_0, c_1)$ such that $c_0^{2p'q'} = 1 \bmod N^2$, both decryption

algorithms output the same result since the action of $x \in \mathbb{Z}_{Np'q'}$ on $c_0^2 \bmod N^2$ only depends on $x \bmod p'q'$. On a ciphertext (c_0, c_1) such that $c_0^{2p'q'} \neq 1 \bmod N^2$, $\mathsf{Decrypt}'$ always returns $\perp$ and we need to assess the probability that $\mathsf{Decrypt}$ does not output $\perp$ as well. If such a malformed ciphertext $C = (c_0, c_1)$ is involved in the i-th decryption query, we can write c_0 as $c_0 = \zeta_0 \cdot (1+N)^{\alpha_0} \cdot g^{2N \cdot \beta_0} \bmod N^2$ and c_1 as $c_1 = \zeta_1 \cdot (1+N)^{\alpha_1} \cdot g^{2N \cdot \beta_1} \bmod N^2$, for some elements ζ_0, ζ_1 of order at most 2 in $\mathbb{Z}_{N^2}^*$ and some arbitrary $\alpha_1 \in \mathbb{Z}_N$, $\alpha_0 \in \mathbb{Z}_N \setminus \{0\}$, $\beta_0, \beta_1 \in \mathbb{Z}_{p'q'}$. Due to the change introduced in Game_4, we can assume that $\gcd(\alpha_0, N) = 1$ since, otherwise, C would already be rejected in $\mathsf{Game}_{5.(i-1)}$ and the two games would proceed identically. Then, in $\mathsf{Game}_{5.(i-1)}$, $\mathsf{Decrypt}$ computes

$$\begin{aligned} M &= c_1 \cdot c_0^{-2 \cdot x} \bmod N^2 \\ &= c_1 \cdot (1+N)^{-\alpha_0 \cdot (2x \bmod N)} \cdot g^{-2N \cdot \beta_0 \cdot (x \bmod p'q')}, \\ &= \zeta_1 \cdot (1+N)^{\alpha_1 - \alpha_0 \cdot (2x \bmod N)} \cdot g^{-2N \cdot (-\beta_1 + 2\beta_0 \cdot (x \bmod p'q'))} \end{aligned} \tag{25}$$

and we can assume that $\zeta_1 = 1$ and $\beta_1 = 2\beta_0 \cdot x \pmod{p'q'}$ since both decryption algorithms return $\perp$ if M is not in the subgroup $\langle 1+N \rangle$. In (25), we note that $\alpha_x \triangleq x \bmod N$ is sampled uniformly in $\mathbb{Z}_N$ *after* $\mathcal{A}$ has submitted its i-th decryption query (c_0, c_1) and thus after α_0 and α_1 have been fixed. Consequently, in the right-hand-side member of (25), $(1+N)^{-\alpha_0 \cdot (2x \bmod N)} \bmod N^2$ is independent of $\mathcal{A}$'s view and ensures that $(1+N)^{\alpha_1 - \alpha_0 \cdot (2x \bmod N)}$ is uniformly distributed in the subgroup $\langle 1+N \rangle$ since $\gcd(2\alpha_0, N) = 1$ and $x \bmod N$ is independent of $x \bmod p'q'$ by the CRT. Therefore, if the i-th decryption query $C = (c_0, c_1)$ involves a ciphertext such that $c_0^{2p'q'} \neq 1 \bmod N^2$, the product (25) has a uniformly distributed component $(1+N)^{\alpha_1 - \alpha_0 \cdot (2x \bmod N)} \bmod N^2$ in the subgroup $\langle 1+N \rangle$. The probability that $\alpha_1 - \alpha_0 \cdot (2x \bmod N) \bmod N$ falls into the interval $[0, B] = [0, 2^t]$ is then at most $(B+1)/N < 1/2^\lambda + 1/N$. Except with probability smaller than $1/2^\lambda + 1/N < 1/2^{\lambda-1}$, $\mathsf{Decrypt}$ thus also returns $\perp$ at the i-th decryption query. □

5.3 Avoiding the Composite Non-Invertibility Assumption

Interestingly, if we further restrict the message space $[0, B]$ in such a way that $B < 2^{-\lambda} \cdot \min(p, q)$, it is possible to prove security under the sole DCR assumption. This requires to adapt the proof of Lemma 5 in the following way.

If the adversary makes its i-th decryption query on a ciphertext (c_0, c_1) such that $c_0 = \zeta_0 \cdot (1+N)^{\alpha_0} \cdot g^{2N \cdot \beta_0} \bmod N^2$ with $\alpha_0 = k \cdot p$, for some $k \in \mathbb{Z}_q$, the product $\alpha_0 \cdot 2x \bmod N$ still has a uniformly distributed component $2x\alpha_0 \bmod q$ in $\mathbb{Z}_q^*$ (note that $\gcd(\alpha_0, q) = 1$ if $\alpha_0 = k \cdot p$ for non-zero k). Then, the distribution of $\alpha_1 - \alpha_0 \cdot 2x \bmod q$ is also uniform in $\mathbb{Z}_q$ since x is chosen after α_0 and α_1. The real $\mathsf{Decrypt}$ algorithm only accepts (c_0, c_1) when $(\alpha_1 - \alpha_0 \cdot 2x \bmod N) \in [0, B]$, which implies $(\alpha_1 - \alpha_0 \cdot 2x \mod q) \in [0, B]$ since $q | N$ and $B < q$. Since the distribution of $\alpha_1 - \alpha_0 \cdot 2x \mod q$ is uniform over $\mathbb{Z}_q$, it falls into the forbidden interval $[0, B]$ with probability $(B+1)/q < 2^{-\lambda+1}$, as required.

The rest of the proof remains unchanged, except that we can remove Game_4

from the sequence of games in the proof of Theorem 3.

We then have to choose $p, q > 2^{l(\lambda)}$ so that $l > t + \lambda$, which reduces the size of the message space by a factor ≈ 2. Concretely, for a 3072-bit modulus N and with $\lambda = 128$, we can CCA1-encrypt 1408-bit messages without relying on any other assumption than DCR.

5.4 Open Questions

An interesting open question is to prove the CCA1 security of the scheme in Section 5.1 under the sole DCR assumption and without sacrificing more than 50% of the message space.

Another interesting open problem is to extend the proof to a variant of the scheme based on the Damgård-Jurik technique [21]. The larger message space would be $[0, B]$, where $B < \lfloor 2^{-\lambda} \cdot N^s \rfloor$ for some $s > 1$, while ciphertexts would live in $(\mathbb{Z}^*_{N^{s+1}})^2$. Unfortunately, we do not know how to adapt the proof to this case since the adversary can always encrypt a multiple of N and we can no longer adapt the proof of Lemma 5.

6 A CCA1 Variant of Castagnos-Laguillaumie

In this section, we adapt our variant of Elgamal-Paillier to the framework of Castagnos and Laguillaumie [9]. It allows a message space of large prime order and removes the need for the non-standard composite non-invertibility assumption used in Section 5, even when messages are as large as possible. We also obtain a tighter security reduction by avoiding the linear security loss (20) of Theorem 3 in the number Q of decryption queries.

We actually present a CCA1 variant of the scheme suggested in [10, Section 3.2], which was itself obtained from [9] by adapting ideas from the Elgamal-Paillier scheme of [7].

We first recall the CL framework, as described in [10,11], which abstracts away the specific number theoretic instantiation introduced in [9]. In the chosen-ciphertext setting, we need to use the variant of [11] that extends [10] so as to work in a larger ambient group $\hat{G}$ whose elements are efficiently recognizable.

Definition 4 (Generator of a DDH group with an easy DL group). *Let* $\mathsf{GenGroup} = (\mathsf{Gen}, \mathsf{Solve})$ *a pair of efficient algorithms where* Gen *is a group generator algorithm taking as inputs parameters* λ *and* μ *and outputting a tuple* $(p, \hat{s}_{\max}, g, f, g_p, \hat{G}, F, \hat{G}^p)$*, where*

- $(\hat{G}, \cdot)$ *is a group of order* $p \cdot \hat{s}$*, for some integer* $\hat{s}$*, and where* p *is a* μ*-bit prime such that* $\gcd(p, \hat{s}) = 1$*. Moreover, elements of* $\hat{G}$ *are efficiently recognizable and* Gen *only outputs an upper bound* $\hat{s}_{\max}$ *on* $\hat{s}$*.*
- $\hat{G}^p = \{x^p \mid x \in \hat{G}\}$ *and* F *are the subgroups of order* $\hat{s}$ *and* p*, respectively, in* $\hat{G}$ *(so that* $\hat{G} \simeq F \times \hat{G}^p$*).*
- $\hat{G}$ *contains a cyclic subgroup* G *of order* $p \cdot s$*, for some* $s | \hat{s}$*. Since* $\gcd(p, s) = 1$*,* $G \simeq F \times G^p$*, where* $G^p = \{x^p \mid x \in G\}$*.*

- f *and* g_p *are generators of* F *and* G^p*, respectively, so that their product* $g \triangleq g_p \cdot f$ *generates* G*.*
- Solve *is a deterministic polynomial time algorithm that solves the discrete logarithm problem in* F*.*

As in [10,11], we rely on a subgroup membership assumption that can be seen as a special case of the general subgroup assumption defined in [32] and captures that the uniform distribution on G is computationally indistinguishable from the uniform distribution on G^p.

Definition 5 (HSM assumption). *Let* $\mathsf{GenGroup} = (\mathsf{Gen}, \mathsf{Solve})$ *the algorithms of Definition 4. The* **Hard Subgroup Membership** *(HSM) assumption says that no PPT adversary can distinguish the distributions*

$$D_0 = \big\{(p, \hat{s}_{\max}, g, f, g_p, \hat{G}, F, \hat{G^p}, Z) \mid (p, \hat{s}_{\max}, g, f, g_p, G, F, G^p) \leftarrow \mathsf{Gen}(1^\lambda, 1^\mu);\ x \hookleftarrow \mathcal{D};\ Z = g^x\big\},$$

and

$$D_1 = \big\{(p, \hat{s}_{\max}, g, f, g_p, \hat{G}, F, \hat{G^p}, Z) \mid (p, \hat{s}_{\max}, g, f, g_p, G, F, G^p) \leftarrow \mathsf{Gen}(1^\lambda, 1^\mu);\ x \hookleftarrow \mathcal{D}_p;\ Z = g_p^x\big\},$$

where $\mathcal{D}$ *(resp.* $\mathcal{D}_p$*) denotes a distribution over* $\mathbb{Z}$ *such that* $\{g^x \mid x \hookleftarrow \mathcal{D}\}$ *and* $\{g_p^x \mid x \hookleftarrow \mathcal{D}_p\}$ *is within statistical distance* $2^{-\lambda}$ *from the uniform distribution over* G *(resp.* G^p*).*

Using class groups of imaginary quadratic fields, Castagnos *et al.* [11] provide an efficient instantiation where the HSM assumption is believed to hold.

6.1 Description

Based on the above framework, we can build an IND-CCA1 variant of the scheme in [10, Figure 2.a] as follows.

Besides the interval check in the decryption algorithm, the main difference is that the secret key and the encryption exponent are sampled from a uniform distribution over a sufficiently large interval (which simplifies the security proof) while [10] obtains shorter keys sampled from a discrete Gaussian distribution.

Keygen$(1^\lambda, 1^t)$**:** Given a security parameter $\lambda \in \mathbb{N}$ and a message length $t \in \mathsf{poly}(\lambda)$, set $\mu \geq t + \lambda$ and conduct the following steps.

1. Generate $(p, \hat{s}_{\max}, g, f, g_p, \hat{G}, F, \hat{G^p}) \leftarrow \mathsf{Gen}(1^\lambda, 1^\mu)$.
2. Choose $x \xleftarrow{R} [0, 2^\lambda \cdot \hat{s}_{\max} \cdot p]$ and compute $h = g_p^x$.

Return the key pair (PK, SK) where $SK := x$ and

$$PK := \big(p, \hat{s}_{\max}, g, f, g_p, G, F, G^p, B\big),$$

where $B = 2^t$ defines the message space $\mathcal{M} = [0, B]$.

Encrypt(PK, m)**:** Given a public key PK and a message $m \in \mathcal{M}$,

1. Choose $r \xleftarrow{R} [0, 2^\lambda \cdot \hat{s}_{\max}]$ and compute
$$c_0 = g_p^r \qquad\qquad c_1 = f^m \cdot h^r$$
2. Output the ciphertext $C = (c_0, c_1)$.

Decrypt(SK, C)**:** Given $SK = x \in \mathbb{Z}$ and $C = (c_0, c_1)$, return $\perp$ if $c_0 \notin \hat{G}$ or $c_1 \notin \hat{G}$. Otherwise, conduct the following steps:

1. Compute $M = c_1 \cdot c_0^{-x}$.
2. Return $\perp$ if $M^p \neq 1$. Otherwise, compute $m = \mathsf{Solve}(f, M)$ and return $\perp$ if $m \notin [0, B]$. Otherwise, return m.

6.2 Security

Theorem 4. *The scheme provides IND-CCA1 security in the standard model under the HSM assumption. For any CCA1 adversary $\mathcal{A}$ making at most Q decryption queries, there exist an HSM distinguisher $\mathcal{B}$ such that*

$$\mathbf{Adv}_{\mathcal{A}}^{\mathrm{CCA1}}(\lambda) \leq \mathbf{Adv}_{\mathcal{B}}^{\mathrm{HSM}}(\lambda) + (Q+2) \cdot 2^{-\lambda+1}$$

Proof. The proof considers a sequence of hybrid games where W_i denotes the event that the adversary wins and outputs $\rho' = \rho$ in Game_i.

Game_0: This is the real IND-CCA1 security game. When the adversary chooses messages $m_0, m_1 \in [0, B]$ in the challenge phase, it obtains a ciphertext

$$c_0^\star = g_p^r \qquad\qquad c_1^\star = f^{m_\rho} \cdot h^r,$$

where $\rho \xleftarrow{R} \{0,1\}$ is the challenger's random bit. When $\mathcal{A}$ terminates, it outputs a bit $\rho' \in \{0,1\}$ and wins if $\rho' = \rho$. Its advantage is defined as $\mathbf{Adv}_{\mathcal{A}}^{\mathrm{CCA1}}(\lambda) := |\Pr[W_0] - 1/2|$.

Game_1: We change the generation of the challenge ciphertext. When $\mathcal{A}$ outputs messages $m_0, m_1 \in [0, B]$, the challenge ciphertext is computed as

$$c_0^\star = g_p^r \qquad\qquad c_1^\star = f^{m_\rho} \cdot c_0^{\star x},$$

for a random $r \xleftarrow{R} [0, 2^\lambda \cdot \hat{s}_{\max}]$. Game_1 is identical to Game_0 from $\mathcal{A}$'s view and we have $\Pr[W_1] = \Pr[W_0]$. Also, by Lemma 3, the distribution of $c_0^\star$ is at most $2^{-\lambda}$ apart from the uniform distribution over G^p.

Game_2: We change the distribution of the challenge ciphertext. In the challenge phase, the challenger now picks $r \xleftarrow{R} [0, 2^\lambda \cdot \hat{s}_{\max} \cdot p]$ and computes

$$c_0^\star = (g_p \cdot f)^r \qquad\qquad c_1^\star = f^{m_\rho} \cdot c_0^{\star x},$$

Under the HSM assumption, Game_2 is indistinguishable from Game_1 and a straightforward reduction shows that $|\Pr[W_2] - \Pr[W_1]| \leq \mathbf{Adv}^{\mathrm{HSM}}(\lambda)$. We note that, by Lemma 3, the distribution of $c_0^\star$ is within distance $2^{-\lambda}$ from the uniform distribution over G.

Game_3: We change the generation of the challenge ciphertext while keeping its distribution statistically unchanged. The challenger now samples $r \xleftarrow{R} [0, 2^\lambda \cdot \hat{s}_{\max}]$, $u \xleftarrow{R} \mathbb{Z}_p$, and computes

$$c_0^\star = f^u \cdot g_p^r \qquad\qquad c_1^\star = f^{m_\rho} \cdot c_0^{\star x},$$

Lemma 6 shows the distribution of $c_0^\star$ is statistically the same as in Game_2, so that the two games are statistically indistinguishable.

At this point, we are done with reductions from computational assumptions. We can thus afford to use a challenger that runs in super-polynomial time without affecting the efficiency of the reduction considered in earlier game transitions.

Game_4: This game is like Game_3 except that, at the outset of the game, the challenger explicitly computes the orders $\hat{s}$ and $\hat{s} \cdot p$ of the groups $\hat{G}^p$ and $\hat{G}$ (which it can do in sub-exponential time). Also, the secret key x is sampled as $x \xleftarrow{R} \mathbb{Z}_{\hat{s} \cdot p}$, which makes h perfectly uniform over G^p since s divides $\hat{s}$. We claim that Game_4 is statistically indistinguishable from Game_3. In Game_3, the challenge ciphertext is of the form

$$c_0^\star = f^u \cdot g_p^r, \qquad\qquad c_1^\star = f^{m_\rho + x \cdot u} \cdot h^r, \tag{26}$$

with $u \sim U(\mathbb{Z}_p)$, $r \sim U([0, 2^\lambda \cdot \hat{s}_{\max}])$. The information that (26) reveals about x is completely determined by $x \bmod s \cdot p$ (and thus by $x \bmod \hat{s} \cdot p$ since s divides $\hat{s}$). The information revealed by decryptions of malformed ciphertexts (c_0, c_1) only depends on $x \bmod \hat{s} \cdot p$ since the order of c_0 is at most $\hat{s} \cdot p$ and the decryption oracle computes $M = c_1 \cdot c_0^{-x}$, where c_0^{-x} is completely determined by $x \bmod \hat{s} \cdot p$ and c_0. The distinguishing advantage between Game_4 and Game_3 is at most the statistical distance between the distributions $\{x \bmod \hat{s} \cdot p \mid x \xleftarrow{R} [0, 2^\lambda \cdot \hat{s}_{\max} \cdot p]\}$ and $\{x \xleftarrow{R} \mathbb{Z}_{\hat{s} \cdot p}\}$, which is smaller than $\hat{s} \cdot p / (2^\lambda \cdot \hat{s}_{\max} \cdot p) < 2^{-\lambda}$ by Lemma 3. Hence, we have $|\Pr[W_3] - \Pr[W_2]| \leq 2^{-\lambda}$.

We now use a sub-sequence of games where we gradually modify the decryption oracle. The sub-sequence starts with $\mathsf{Game}_{4.0}$, which is identical to Game_4.

$\mathsf{Game}_{4.i}$ $(1 \leq i \leq Q)$: In these games, we modify the key generation phase where the challenger initially computes $h = g_p^{\beta_x}$, for a random $\beta_x \xleftarrow{R} \mathbb{Z}_{\hat{s}}$, and defines an alternative secret key $SK' := (\beta_x, p, \hat{s})$. The first i decryption queries are answered using the following modified decryption algorithm:

$\mathsf{Decrypt'}(SK', C)$: On input of $C = (c_0, c_1)$, return $\perp$ if $c_0^{\hat{s}} \neq 1$. Otherwise, compute $M = c_1 \cdot c_0^{-\beta_x}$ and return $\perp$ if $M^p \neq 1$. If $M^p = 1$, compute $m = \mathsf{Solve}(f, M)$ and return m if $m \in [0, B]$. Otherwise, return $\perp$.

At the $(i+1)$-th decryption query, the challenger samples $\alpha_x \xleftarrow{R} \mathbb{Z}_p$ and defines $x \in \mathbb{Z}_{\hat{s} \cdot p}$ such that $\alpha_x = x \bmod p$ and $\beta_x = x \bmod \hat{s}$. Then, $SK = x$ is used to answer the last $Q - i$ decryption queries using the decryption algorithm of Game_4.

In Lemma 7, we prove that $|\Pr[W_{4.i}] - \Pr[W_{4.(i-1)}]| \leq 1/2^{\lambda-1}$, so that the two games are statistically close.

Game_5: This game is identical to $\mathsf{Game}_{5.Q}$ except that the challenge ciphertext is computed by sampling $u, v \xleftarrow{R} \mathbb{Z}_p$, $r \xleftarrow{R} [0, 2^\lambda \cdot \hat{s}_{\max}]$ and computing

$$c_0^\star = f^u \cdot g_p^r \qquad\qquad c_1^\star = f^v \cdot h^r, \tag{27}$$

instead of

$$c_0^\star = f^u \cdot g_p^r \qquad\qquad c_1^\star = f^{m_\rho + x \cdot u} \cdot h^r, \tag{28}$$

as in $\mathsf{Game}_{4.Q}$. We claim that Game_5 is perfectly indistinguishable from $\mathsf{Game}_{4.Q}$. The reason is that all decryption queries are answered using the alternative secret key $SK' = (x \bmod \hat{s}, p, \hat{s})$ in $\mathsf{Game}_{4.Q}$. This implies that $x \bmod p$ is perfectly independent of $\mathcal{A}$'s view until the challenge phase since its choice is postponed until the moment where $\mathcal{A}$ has declared the challenge messages $m_0, m_1 \in [0, B]$. Since $\gcd(u, p) = 1$ and given that $x \bmod p$ is uniformly distributed over $\mathbb{Z}_p$ conditionally on $x \bmod \hat{s}$, the conditional distribution of $m_\rho + x \cdot u \bmod p$ is also uniform over $\mathbb{Z}_p$ in the expression of $c_1^\star$ in (28). This shows that the ciphertext distributions (28) and (27) are perfectly indistinguishable and we have $\Pr[W_5] = \Pr[W_{4.Q}]$.

In Game_5, the challenge ciphertext is totally independent of m_ρ and we have $\Pr[W_5] = 1/2$. We then obtain the stated upper bound on the adversary's advantage $\mathbf{Adv}_{\mathcal{A}}^{\mathrm{CCA1}}(\lambda) = |\Pr[W_0] - 1/2|$. □

Lemma 6. Game_2 *is statistically indistinguishable from* Game_3*. Concretely, we have* $|\Pr[W_3] - \Pr[W_2]| \leq 2^{-\lambda+1}$.

Proof. In both games, the distribution of $c_0^\star$ is within distance $2^{-\lambda}$ from the uniform distribution over G.

In Game_2, we have $c_0^\star = (g_p \cdot f)^r$ where $r \xleftarrow{R} [0, 2^\lambda \cdot \hat{s}_{\max} \cdot p]$ and the product $g_p \cdot f$ has order $s \cdot p$. So, we can apply Lemma 3 with $L = 2^\lambda \cdot \hat{s}_{\max} \cdot p$ and $\ell = s \cdot p$ to argue that the distribution of $c_0^\star$ is within statistical distance $2^{-\lambda}$ from $\{c_0^\star = (g_p \cdot f)^r \mid r \xleftarrow{R} [0, s \cdot p]\}$. By the CRT, the latter distribution is the same as

$$\{c_0^\star = g_p^r \cdot f^u \mid r \xleftarrow{R} \mathbb{Z}_s,\ u \xleftarrow{R} \mathbb{Z}_p\}.$$

By applying Lemma 3 again with $L = 2^\lambda \cdot \hat{s}_{\max}$ and $\ell = s$, the above distribution of $c_0^\star$ is within distance $2^{-\lambda}$ from that in Game_3. □

Lemma 7. *For each* $i \in [0, Q]$, $\mathsf{Game}_{4.i}$ *is statistically indistinguishable from* $\mathsf{Game}_{4.(i-1)}$*. We have* $|\Pr[W_{4.i}] - \Pr[W_{4.(i-1)}]| \leq 2^{-\lambda+1}$.

Proof. The two games are identical from $\mathcal{A}$'s standpoint unless its i-th decryption query involves a ciphertext that gets rejected in $\mathsf{Game}_{4.i}$, but would not have been rejected in $\mathsf{Game}_{4.(i-1)}$.

For any ciphertext $C = (c_0, c_1)$ such that $c_0 \in \hat{G}^p$ (i.e., $c_0^{\hat{s}} = 1$), both decryption oracles output the same result since the action of $x \in \mathbb{Z}_{\hat{s}\cdot p}$ on (c_0, c_1) is completely determined by $x \bmod \hat{s}$. On a ciphertext (c_0, c_1) such that $c_0^{\hat{s}} \neq 1$, $\mathsf{Decrypt}'$ always returns $\perp$ and we just need to assess the probability that $\mathsf{Decrypt}$ returns something else. If the i-th query involves a ciphertext $C = (c_0, c_1)$ where c_0 has a non-trivial component in F, we can write $(c_0, c_1) = (\mu_0 \cdot f^{\alpha_0} \cdot g_p^{\beta_0}, \mu_1 \cdot f^{\alpha_1} \cdot g_p^{\beta_1})$ for some arbitrary $\alpha_0, \alpha_1 \in \mathbb{Z}_p$ and $\beta_0, \beta_1 \in \mathbb{Z}_s$ and $\mu_0, \mu_1 \in \hat{G}^p \setminus G^p$. Then, in $\mathsf{Game}_{4.(i-1)}$, the i-th query is answered using $\mathsf{Decrypt}$ which computes

$$M = c_1 \cdot c_0^{-x} = \mu_1 \cdot \mu_0^{-(x \bmod \hat{s})} \cdot f^{(\alpha_1 - \alpha_0 \cdot x \bmod p)} \cdot g_p^{(\beta_1 - \beta \cdot x \bmod s)}, \tag{29}$$

where $\alpha_x \triangleq x \bmod p$ is sampled uniformly in $\mathbb{Z}_p$ *after* the choice of (c_0, c_1) by $\mathcal{A}$ and thus after (α_0, α_1) have been fixed. Therefore, in the right-hand-side member of (29), $f^{\alpha_1 - \alpha_0 \cdot (x \bmod p)}$ is completely independent of $\mathcal{A}$'s view and uniformly distributed in the subgroup F since $\gcd(\alpha_0, p) = 1$ and $x \bmod p$ is independent of $x \bmod \hat{s}$ by the CRT. Hence, if $c_0^{\hat{s}} \neq 1$ at the i-th query, $M = c_1 \cdot c_0^{-x}$ has a uniformly distributed component in the subgroup F.

If $M^p \neq 1$, both decryption algorithms output $\perp$ and we only need to consider the case $M^p = 1$. Then, the probability that $\mathsf{Solve}(f, M) \in [0, B] = [0, 2^t]$ is bounded by $(B+1)/p < 1/2^\lambda + 1/p < 1/2^{\lambda-1}$. Except with probability at most $1/2^{\lambda-1}$, $\mathsf{Decrypt}$ thus agrees with $\mathsf{Decrypt}'$ and also returns $\perp$ at the i-th decryption query. □

Acknowledgments. The author is grateful to Thomas Peters and the anonymous reviewers for very useful comments.

References

1. Abdalla, M., Bellare, M., Rogaway, P.: The oracle Diffie-Hellman assumptions and an analysis of DHIES. In: CT-RSA (2001). https://doi.org/10.1007/3-540-45353-9_12
2. Akavia, A., Gentry, C., Halevi, S., Vald, M.: Achievable CCA2 relaxation for homomorphic encryption. In: TCC (2022). https://doi.org/10.1007/978-3-031-22365-5_3
3. Armknecht, F., Katzenbeisser, S., Peter, A.: Group homomorphic encryption: characterizations, impossibility results, and applications. Designs, Codes and cryptography **67** (2013). https://doi.org/10.1007/S10623-011-9601-2
4. Bellare, M., Palacio, A.: Towards plaintext-aware public-key encryption without random oracles. In: Asiacrypt (2004). https://doi.org/10.1007/978-3-540-30539-2_4
5. Boneh, D., Goh, E.J., Nissim, K.: Evaluating 2-DNF formulas on ciphertexts. In: TCC (2005). https://doi.org/10.1007/978-3-540-30576-7_18

6. Boneh, D., Segev, G., Waters, B.: Targeted malleability: Homomorphic encryption for restricted computations. In: ITCS (2012). https://doi.org/10.1145/2090236.2090264
7. Camenisch, J., Shoup, V.: Practical verifiable encryption and decryption of discrete logarithms. In: Crypto (2003). https://doi.org/10.1007/978-3-540-45146-4_8
8. Canetti, R., Raghuraman, S., Richelson, S., Vaikuntanathan, V.: Chosen ciphertext secure fully homomorphic encryption. In: PKC (2017). https://doi.org/10.1007/978-3-662-54388-7_8
9. Castagnos, G., Laguillaumie, F.: Linearly homomorphic encryption from DDH. In: CT-RSA (2015). https://doi.org/10.1007/978-3-319-16715-2_26
10. Castagnos, G., Laguillaumie, F., Tucker, I.: Practical fully secure unrestricted inner product functional encryption modulo p. In: Asiacrypt (2018). https://doi.org/10.1007/978-3-030-03329-3_25
11. Castagnos, G., Laguillaumie, F., Tucker, I.: A tighter proof for CCA secure inner product functional encryption: Genericity meets efficiency. Theoretical Computer Science (914) (2022). https://doi.org/10.1016/J.TCS.2022.02.014
12. Chase, M., Kohlweiss, M., Lysyanskaya, A., Meiklejohn, S.: Malleable proof systems and applications. In: Eurocrypt (2012). https://doi.org/10.1007/978-3-642-29011-4_18
13. Checri, M., Sirdey, R., Boudguiga, A., Bultel, J.P., Choffrut, A.: On the practical CPAD security of exact and threshold FHE schemes and libraries. In: Crypto (2024). https://doi.org/10.1007/978-3-031-68382-4_1
14. Chenal, M., Tang, Q.: On key recovery attacks against existing somewhat homomorphic encryption schemes. In: Latincrypt (2014). https://doi.org/10.1007/978-3-319-16295-9_13
15. Cheon, J.H., Choe, H., Passelègue, A., Stehlé, D., Suvanto, E.: Attacks against the INDCPA-D security of exact FHE schemes. In: ACM-CCS (2024), to appear
16. Cheon, J.H., Kim, A., Kim, M., Song, Y.S.: Homomorphic encryption for arithmetic of approximate numbers. In: Asiacrypt (2017). https://doi.org/10.1007/978-3-319-70694-8_15
17. Cramer, R., Shoup, V.: A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In: Crypto (1998). https://doi.org/10.1007/BFB0055717
18. Cramer, R., Shoup, V.: Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In: Eurocrypt (2002). https://doi.org/https://doi.org/10.1007/3-540-46035-7_4
19. Dahab, R., Galbraith, S., Morais, E.: Adaptive key recovery attacks on NTRU-based somewhat homomorphic encryption schemes. In: ICITS (2015). https://doi.org/10.1007/978-3-319-17470-9_17
20. Damgård, I.: Towards practical public key systems secure against chosen ciphertext attacks. In: Crypto (1991). https://doi.org/10.1007/3-540-46766-1_36
21. Damgård, I., Jurik, M.: A generalisation, a simplification and some applications of Paillier's probabilistic public-key system. In: PKC (2001). https://doi.org/10.1007/3-540-44586-2_9
22. Desmedt, Y., Lipmaa, H., Phan, D.H.: Hybrid Damgård is CCA1-secure under the DDH assumption. In: CANS (2008). https://doi.org/10.1007/978-3-540-89641-8_2
23. Desmedt, Y., Phan, D.H.: A CCA Secure Hybrid Damgård's ElGamal Encryption. In: Provsec (2008). https://doi.org/10.1007/978-3-540-88733-1_5
24. Dodis, Y., Halevi, S., Wichs, D.: Security with functional re-encryption from CPA. In: TCC (2023). https://doi.org/10.1007/978-3-031-48618-0_10

25. Dolev, D., Dwork, C., Naor, M.: Non-malleable cryptography. In: STOC (1991). https://doi.org/10.1145/103418.103474
26. El Gamal, T.: A public key cryptosystem and a signature scheme based on discrete logarithms. In: Crypto (1984). https://doi.org/10.1007/3-540-39568-7_2
27. Emura, K., Hanaoka, G., Ohtake, G., Matsuda, T., Yamada, S.: Chosen ciphertext secure keyed-homomorphic public-key encryption. In: PKC (2013). https://doi.org/10.1007/978-3-642-36362-7_3
28. Fauzi, P., Norberg Hovd, M., Raddum, H.: On the IND-CCA1 security of FHE schemes. Cryptology ePrint Archive Report 2021/1624 (2021)
29. Freeman, D.: Converting pairing-based cryptosystems from composite-order groups to prime-order groups. In: Eurocrypt (2010). https://doi.org/10.1007/978-3-642-13190-5_3
30. Fuchsbauer, G., Kiltz, E., Loss, J.: The algebraic group model and its applications. In: Crypto (2018). https://doi.org/10.1007/978-3-319-96881-0_2
31. Gentry, C., Halevi, S., Vaikuntanathan, V.: A simple BGN-style encryption scheme from LWE. In: Eurocrypt (2010). https://doi.org/10.1007/978-3-642-13190-5_26
32. Gjøsteen, K.: Homomorphic cryptosystems based on subgroup membership problems. In: Mycrypt (2005). https://doi.org/10.1007/11554868_22
33. Gjøsteen, K.: A New Security Proof for Damgård's ElGamal. In: CT-RSA (2006). https://doi.org/10.1007/11605805_10
34. Hofheinz, D.: All-but-many lossy trapdoor functions. In: Eurocrypt (2012). https://doi.org/10.1007/978-3-642-29011-4_14
35. Joye, M., Quisquater, J.J., Yung, M.: On the power of misbehaving adversaries and security analysis of the original EPOC. In: CT-RSA (2001). https://doi.org/10.1007/3-540-45353-9_16
36. Kiltz, E., Pietrzak, K., Stam, M., Yung, M.: A new randomness extraction paradigm for hybrid encryption. In: Eurocrypt (2009). https://doi.org/10.1007/978-3-642-01001-9_34
37. Kurosawa, K., Desmedt, Y.: A new paradigm of hybrid encryption scheme. In: Crypto (2004). https://doi.org/10.1007/978-3-540-28628-8_26
38. Lai, J., Deng, R.H., Ma, C., Sakurai, K., Weng, J.: CCA-secure keyed-fully homomorphic encryption. In: PKC (2016). https://doi.org/10.1007/978-3-662-49384-7_4
39. Li, B., Micciancio, D.: On the security of homomorphic encryption on approximate numbers. In: Eurocrypt (2021). https://doi.org/10.1007/978-3-030-77870-5_23
40. Lipmaa, H.: On the CCA1-security of Elgamal and Damgård Elgamal. In: Inscrypt (2010). https://doi.org/10.1007/978-3-642-21518-6_2
41. Loftus, J., May, A., Smart, N., Vercauteren, F.: On CCA-secure fully homomorphic encryption. In: SAC (2011). https://doi.org/10.1007/978-3-642-28496-0_4
42. Manulis, M., Nguyen, J.: Fully homomorphic encryption beyond IND-CCA1 security: Integrity through verifiability. In: Eurocrypt (2024). https://doi.org/10.1007/978-3-031-58723-8_3
43. Naor, M., Yung, M.: Public-key cryptosystems provably secure against chosen ciphertext attacks. In: STOC (1990). https://doi.org/10.1145/100216.100273
44. Paillier, P.: Public-key cryptosystems based on composite degree residuosity classes. In: Eurocrypt (1999). https://doi.org/10.1007/3-540-48910-X_16
45. Prabhakaran, M., Rosulek, M.: Homomorphic encryption with CCA security. In: ICALP (2008). https://doi.org/10.1007/978-3-540-70583-3_54

46. Sato, S., Emura, K., Takayasu, A.: Keyed-fully homomorphic encryption without indistinguishability obfuscation. In: ACNS (2022). https://doi.org/10.1007/978-3-031-09234-3_1
47. Schäge, S.: New limits of provable security and applications to ElGamal encryption. In: Eurocrypt (2024). https://doi.org/10.1007/978-3-031-58737-5_10
48. Shoup, V.: Lower bounds for discrete logarithms and related problems. In: Eurocrypt (1997). https://doi.org/10.1007/3-540-69053-0_18
49. Viand, A., Knabenhans, C., Hithnawi, A.: Verifiable fully homomorphic encryption. arxiv preprint (2023), https://arxiv.org/abs/2301.07041
50. Yasuda, S., Kitagawa, F., Tanaka, K.: Constructions for the IND-CCA1 secure fully homomorphic encryption. In: CREST Crypto-Math Project (2017). https://doi.org/10.1007/978-981-10-5065-7_18
51. Zhang, Z., Plantard, T., Susilo, W.: On the CCA1-security of somewhat homomorphic encryption over the integers. In: ISPEC (2012). https://doi.org/10.1007/978-3-642-29101-2_24

Supplementary Material

A Deferred Material for the Scheme in Section 4

A.1 Correctness and Circuit Privacy Definitions

Algorithms $\mathsf{Add}_0, \mathsf{Add}_1$ and $\mathsf{Multiply}$ are used to define an evaluation algorithm Eval that takes as input a public key PK, a set of ciphertexts $(C_i)_{i\in[\ell]}$ at depth 0 or 1 and a function f of degree at most 2, and performs a sequence of additions/or multiplications to output an evaluated ciphertext $C \leftarrow \mathsf{Eval}(PK, (C_i)_{i\in[\ell]}, f)$.

We first recall the definition of circuit privacy since it will simplify the definition of correctness.

Circuit privacy. A depth-one homomorphic encryption scheme is *circuit-private* if, for any $\ell \in \mathbb{N}$, any messages $m_1, \ldots, m_\ell \in \mathcal{M}$, and any function $f : \mathcal{M}^\ell \to \mathcal{M}$ of degree ≤ 2 such that $f(m_1 \ldots, m_\ell) \in \mathcal{M}$, there exists a simulator Sim such that the following distributions are statistically close:

$$D_0 := \{(PK, SK, (m_i, C_i)_{i\in[\ell]}, C) \mid (PK, SK) \leftarrow \mathsf{Keygen}(1^\lambda, 1^t), \\ \forall i \in [\ell] : C_i \leftarrow \mathsf{Encrypt}_0(PK, m_i),\ C \leftarrow \mathsf{Eval}(PK, (C_i)_{i\in[\ell]}, f)\}$$

$$D_1 := \{(PK, SK, (m_i, C_i)_{i\in[\ell]}, C) \mid (PK, SK) \leftarrow \mathsf{Keygen}(1^\lambda, 1^t), \\ \forall i \in [\ell] : C_i \leftarrow \mathsf{Encrypt}_0(PK, m_i),\ C \leftarrow \mathsf{Sim}(PK, f(m_1, \ldots, m_\ell))\}$$

Correctness. A depth-one homomorphic encryption scheme is correct if, for any $\ell \in \mathbb{N}$, any messages $m_1, \ldots, m_\ell \in \mathcal{M}$, and any function $f : \mathcal{M}^\ell \to \mathcal{M}$ of

degree ≤ 2 such that $f(m_1 \ldots, m_\ell) \in \mathcal{M}$, there exists $d \in \{0,1\}$ such that

$$\Pr\left[f(m_1,\ldots,m_\ell) \neq \mathsf{Decrypt}_d(SK, C) \mid (PK, PK) \leftarrow \mathsf{Keygen}(1^\lambda, 1^t), \right.$$
$$\left. \forall i \in [\ell] : C_i \leftarrow \mathsf{Encrypt}_0(PK, m_i),\ C \leftarrow \mathsf{Eval}(PK, (C_i)_{i\in[\ell]}, f)\right] \leq \mathsf{negl}(\lambda)$$

In the definition of correctness, we assume that all input ciphertexts $(C_i)_{i\in[\ell]}$ are encrypted at depth 0 since, by the circuit privacy requirement, a depth-1 ciphertext C^1 that encrypts $m \in \mathcal{M}$ is statistically indistinguishable from the product two depth-0 ciphertexts C_1^0, C_2^0 which encrypt m and 1, respectively.

A.2 Correctness of Homomorphic Operations in Section 4

It is easy to see that outputs of the multiplication algorithm are distributed as fresh depth-1 encryptions of product messages. Namely, let two depth-0 ciphertexts $C_1 = ([\boldsymbol{c}_1]_1, [\boldsymbol{d}_1]_2)$ and $C_2 = ([\boldsymbol{c}_2]_1, [\boldsymbol{d}_2]_2)$ of the form

$$[\boldsymbol{c}_1]_1 = r_1 \cdot \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}_1 + m_1 \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}_1, \qquad [\boldsymbol{d}_1]_2 = s_1 \cdot \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}_2 + m_1 \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}_2$$

$$[\boldsymbol{c}_2]_1 = r_2 \cdot \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}_1 + m_2 \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}_1, \qquad [\boldsymbol{d}_2]_2 = s_2 \cdot \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}_2 + m_2 \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}_2$$

for some plaintexts $m_1, m_2 \in [0, B]$ such that $m_1 \cdot m_2 \in [0, B]$. Then,

$$[\boldsymbol{c}_1]_1 \cdot [\boldsymbol{d}_2^\top]_2 = r_1 \cdot s_2 \cdot \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}_1 \cdot \begin{bmatrix} b_1 & b_2 & b_3 \end{bmatrix}_2 + r_1 \cdot m_2 \cdot \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}_1 \cdot \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}_2$$
$$+ s_2 \cdot m_1 \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}_1 \cdot \begin{bmatrix} b_1 & b_2 & b_3 \end{bmatrix}_2 + m_1 \cdot m_2 \cdot \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}_T$$

so that $[\boldsymbol{c}_1]_1 \cdot [\boldsymbol{d}_2^\top]_2$ decrypts to $m_1 \cdot m_2$. Moreover, the second and third terms of (4) perfectly re-randomize $[\boldsymbol{c}_1]_1 \cdot [\boldsymbol{d}_2^\top]_2$ so as to obtain a product ciphertext that has the same distribution as a fresh encryption of $m_1 \cdot m_2$ at depth 1.

B The Case of the Original Homomorphic DEG

The scheme described in Section 3.1 is not quite identical to the original system described by Damgård [20], where the decryptor computes two exponentiations instead of a multi-exponentiation. We show that our proof carries over to the original scheme (with the message in the exponent), which is recalled hereunder.

$\mathsf{Keygen}(1^\lambda, 1^t)$: Given a security parameter $\lambda \in \mathbb{N}$ and a desired message length $t = O(\log \lambda)$,

1. Choose a cyclic group $\mathbb{G}$ of prime order $p > 2^{l(\lambda)}$, for some polynomial $l : \mathbb{N} \to \mathbb{N}$. Choose generators $g, g_1 \xleftarrow{R} \mathbb{G}$.
2. Choose $\omega, z \xleftarrow{R} \mathbb{Z}_p$ and compute $g_2 = g_1^{\omega}$ and $h = g_1^{-z}$.

Return the key pair (PK, SK) consisting of $SK := (\omega, z) \in \mathbb{Z}_p^2$ and

$$PK := \left(\mathbb{G},\ g,\ g_1,\ g_2,\ h,\ B = 2^t\right),$$

where B defines the message space $\mathcal{M} = [0, B]$.

Encrypt(PK, m)**:** Given a public key PK and a message m consisting of an integer in the interval $\mathcal{M} = [0, B]$, do the following:

1. Choose $r \xleftarrow{R} \mathbb{Z}_p$ and compute

$$c_0 = g^m \cdot h^r \qquad c_1 = g_1^r \qquad c_2 = g_2^r$$

2. Output the ciphertext $C = (c_0, c_1, c_2)$.

Decrypt(SK, C)**:** Given $SK = (\omega, z) \in \mathbb{Z}_p^2$ and $C = (c_0, c_1, c_2)$,

1. On input of $C = (c_0, c_1, c_2)$, return $\perp$ if $c_2 \neq c_1^{\omega}$. Otherwise, compute $M = c_0 \cdot c_1^z$.
2. If there exists an integer $m \in [0, B]$ such that $M = g^m$, return m. Otherwise, return $\perp$.

The main difference with the proof of Theorem 1 is that, between Game_0 and Game_1, we need to introduce a sub-sequence of games $\{\mathsf{Game}_{0.i}\}_{i=1}^{Q}$ where we gradually modify the decryption oracle to use the Decrypt algorithm of the scheme in Section 3.1 instead of the above one.

Theorem 5. *The scheme provides IND-CCA1 security in the standard model under the DDH assumption. For any CCA1 adversary $\mathcal{A}$ making at most Q decryption queries, there is a DDH distinguisher $\mathcal{B}$ such that*

$$\mathbf{Adv}_{\mathcal{A}}^{\mathrm{CCA1}}(\lambda) \leq \mathbf{Adv}_{\mathcal{B}}^{\mathrm{DDH}}(\lambda) + \frac{1 + 2 \cdot Q \cdot (B + 1)}{2^{\lambda}}$$

Proof. We only outline the changes compared to the proof of Theorem 1.

Game_0: This is the real IND-CCA1 security game.

We now consider a sub-sequence of hybrid games where $\mathsf{Game}_{0.0}$ is Game_0.

$\mathsf{Game}_{0.i}$ $(1 \leq i \leq Q)$**:** In these games, we modify the key generation phase where the challenger computes $g_2 = g_1^{\omega}$ and $h = g_1^{-z}$ for random $\omega, z \xleftarrow{R} \mathbb{Z}_p$. In addition, it initially chooses $x_2 \xleftarrow{R} \mathbb{Z}_p$ and sets $x_1 = z - \omega \cdot x_2$. It defines the alternative secret key $SK' := (x_1, x_2)$. In the first $Q - i$ decryption queries, the challenger runs the real decryption algorithm. In the last i decryption queries, it uses modified decryption algorithm below:

$\mathsf{Decrypt}'(SK', C)$: Given $C = (c_0, c_1, c_2)$, compute $M = c_0 \cdot c_1^{x_1} \cdot c_2^{x_2}$. If there exists $m \in [0, B]$ such that $M = g^m$, return m. Otherwise, return $\perp$.

The same argument as in the proof of Lemma 1 shows that, for each $i \in [0, Q]$, we have $|\Pr[W_{0.i}] - \Pr[W_{0.(i-1)}]| \leq (B+1)/2^\lambda$. Namely, $\mathsf{Game}_{0.i}$ only differs from $\mathsf{Game}_{0.(i-1)}$ in the $(Q-i+1)$-th decryption query, where $\mathsf{Game}_{0.i}$ starts answering decryption queries using $\mathsf{Decrypt}'$ but $\mathsf{Game}_{0.(i-1)}$ still uses $\mathsf{Decrypt}$. The probability that $\mathsf{Game}_{0.i}$ fails to reject a ciphertext that $\mathsf{Game}_{0.(i-1)}$ would reject is $(B+1)/2^\lambda$ since neither of these games uses x_2 before the $(Q-i+1)$-th query.

We next define Game_1 as being identical to $\mathsf{Game}_{0.Q}$. Then, from Game_1 onwards, the sequence of games is identical to the one of Theorem 1. □