Efficient Non-Interactive Zero Knowledge Arguments for Set Operations*

Prastudy Fauzi[1], Helger Lipmaa[1], and Bingsheng Zhang[2]

[1] University of Tartu, Estonia
[2] National and Kapodistrian University of Athens, Greece

Abstract. We propose a non-interactive zero knowledge *pairwise multi-set sum equality test (PMSET)* argument in the common reference string (CRS) model that allows a prover to show that the given committed multisets $\mathbb{A}_j$ for $j \in \{1, 2, 3, 4\}$ satisfy $\mathbb{A}_1 \uplus \mathbb{A}_2 = \mathbb{A}_3 \uplus \mathbb{A}_4$, i.e., every element is contained in $\mathbb{A}_1$ and $\mathbb{A}_2$ exactly as many times as in $\mathbb{A}_3$ and $\mathbb{A}_4$. As a corollary to the PMSET argument, we present arguments that enable to efficiently verify the correctness of various (multi)set operations, for example, that one committed set is the intersection or union of two other committed sets. The new arguments have constant communication and verification complexity (in group elements and group operations, respectively), whereas the CRS length and the prover's computational complexity are both proportional to the cardinality of the (multi)sets. We show that one can shorten the CRS length at the cost of a small increase of the communication and the verifier's computation.
Keywords. Multisets, non-interactive zero knowledge, set operation arguments.

1 Introduction

One of the most common tasks undertaken to achieve active security (i.e., security against malicious participants) in various cryptographic protocols is to construct an efficient zero knowledge proof that the committed (or encrypted) messages sent by various parties belong to correct sets. For example, some of the most efficient e-voting protocols [15,17] and e-auction protocols [38] are secure only if the voters (resp., bidders) have committed to inputs from a certain range. Because of such reasons, range proofs — where the prover aims to convince the verifier that the committed message belongs to some public range — have been widely studied in cryptographic literature. There are many well-known efficient range proofs, both interactive [9,38,35,10,13] and non-interactive [42,14,21].

However, in many applications it is not sufficient to prove that the inputs belong to a continuous range, since the valid input set may be an arbitrary (polynomial-size) set of integers. Moreover, often the same party has to commit to related inputs many times, and the whole protocol is secure only if the committed input sets satisfy some set-theoretic relations. E.g., in an approval

* First eprint version, January 31, 2014

Paper	Operation	RO	|CRS|	Prov comp	Ver comp	Comm
[32]	zero-knowledge sets	yes	$\Theta(k)$	$\Theta(k)$	$\Theta(1)$	$\Theta(1)$
[19]	committed subset of disjoint sets	yes	-	$\Omega(k)$	$\Omega(k)$	$\Omega(k)$
[33]	set intersection, set union	yes	-	$O(k)$	$O(k)$	$O(k)$
[31]	set intersection	yes	$\Theta(1)$	$\Theta(k)$	$\Theta(k)$	$\Theta(k)$
This paper	PMSET, committed subset, set intersection, set union, set difference, zero-knowledge sets, accumulator, ...	no	$\Theta(k)$	$\Theta(k)$	$\Theta(1)$	$\Theta(1)$

Table 1. Performance comparison of NIZK for set operations

e-voting protocol, one could first to be asked to commit to a set $\mathbb{A}$ of all approved candidates, and in the second round (based on the outcome of the first round) to a certain subset $\mathbb{B}$ of $\mathbb{A}$. One could interpret $\mathbb{A}$ and $\mathbb{B}$ as multisets, where a voter is allowed to distribute a limited number of points between the set of all candidates. To achieve active security, the voter must prove in particular that $\mathbb{B} \subseteq \mathbb{A} \subseteq \mathbb{U}$, where $\mathbb{U}$ is the set of all candidates. Moreover, in any concrete application, it can also be required to lower and upper bound the cardinality of $\mathbb{A}$ and $\mathbb{B}$. For instance, in the case of approval voting, the voter may only have a number of votes to spend, but may be required to vote at least once. Similarly, in a combinatorial auction, a bidder may bid up to a certain number items, but might be required to bid at least once to continue in the next round.

Similar issues arise in many other applications, and thus a lot of work has been done in constructing efficient zero knowledge proofs for (multi)set-theoretic operations. However, practically all existing (multi)set-theoretic zero knowledge proofs [19,33,31] require at least linear communication in the size of the committed sets. This is not acceptable in many applications where the cardinality of the underlying sets is large. See Table 1 for a brief comparison, and App. A for a longer comparison. (App. A also compares the current work with [32].)

Moreover, all existing efficient set-theoretic zero-knowledge proofs are interactive, which makes them less useful in practice. While they can be made non-interactive in the random oracle model by using the Fiat-Shamir heuristic [22], it is well-known that such a heuristic is not a proof [12,26]. Thus, a better approach is to build non-interactive zero knowledge (NIZK) proofs in the common reference string (CRS) model. See Sect. 2 for more preliminaries on NIZK proofs and arguments (i.e., computationally sound proofs). For the rest of this introduction, we recall that sublinear NIZK proofs can only be (a) computationally sound, and (b) cannot be based on standard (falsifiable) assumptions [25]. Thus, following a long line of contemporary cryptographic research [28,14,36,24,5,3,21,37], we will construct NIZK arguments that are sound under some knowledge assumptions.

Our Contibutions. We tackle the task of constructing efficient (multi)set-theoretic NIZK arguments in a modular way. First, we design an efficient pairing-based NIZK argument for a certain multiset relation. Second, we show that the proposed argument can be used to construct efficient NIZK arguments for a plethora of other (multi)set relations.

More precisely, recall that if $\mathbb{A}$ is a multiset, then every element a of the universe $\mathbb{U}$ belongs to $\mathbb{A}$ with some multiplicity $\mathbf{1}_{\mathbb{A}}(a) \geq 0$. (Multiplicity 0 means that a does not belong to $\mathbb{A}$.) In particular, $\mathbb{A}_1 \uplus \mathbb{A}_2$ is a multiset that has as many copies of any element a as $\mathbb{A}_1$ and $\mathbb{A}_2$ put together, $\mathbf{1}_{\mathbb{A}_1 \uplus \mathbb{A}_2}(a) = \mathbf{1}_{\mathbb{A}_1}(a) + \mathbf{1}_{\mathbb{A}_2}(a)$ for each $a \in \mathbb{U}$. See Sect. 2 for more preliminaries on multisets.

We propose a non-interactive *pairwise multiset sum equality test* (PMSET) argument, where the prover has committed to four multisets $\mathbb{A}_1$, $\mathbb{A}_2$, $\mathbb{A}_3$ and $\mathbb{A}_4$, and aims to prove in zero knowledge that $\mathbb{A}_1 \uplus \mathbb{A}_2 = \mathbb{A}_3 \uplus \mathbb{A}_4$. That is, for all $a \in \mathbb{U}$, $\mathbf{1}_{\mathbb{A}_1}(a) + \mathbf{1}_{\mathbb{A}_2}(a) = \mathbf{1}_{\mathbb{A}_3}(a) + \mathbf{1}_{\mathbb{A}_4}(a)$. Moreover, for some public constants k_j, this argument guarantees the verifier that $|\mathbb{A}_j| \leq k_j$.

Briefly, the intuition behind our new PMSET argument is as follows. The prover first commits to a succinct encoding of each $\mathbb{A}_j$. More precisely, $\mathbb{A}_j \subset \mathbb{Z}_p$ is encoded as $\chi_{\mathbb{A}_j}(\sigma)$, where $\chi_{\mathbb{A}_j}(X) := \prod_{a \in \mathbb{A}_j}(X - a)$ (with correct multiplicities), and σ is a secret key. The prover commits to $\chi_{\mathbb{A}_j}(\sigma)$ for $j \in \{1, 2, 3, 4\}$. After that, the prover creates a succinct NIZK argument that $\chi_{\mathbb{A}_1}(\sigma)\chi_{\mathbb{A}_2}(\sigma) = \chi_{\mathbb{A}_3}(\sigma)\chi_{\mathbb{A}_4}(\sigma)$, where $\chi_{\mathbb{A}_j}(X)$ is a degree $\leq k_j$ polynomial. The real argument is more complicated, since it has to include several extra values to allow for both the soundness and the zero knowledge part of the security proof to go through:

(i) to achieve computational soundness, every group element in the argument is accompanied by a knowledge component,
(ii) to achieve zero knowledge, the argument contains independent random commitments D_j to all 4 multisets $\mathbb{A}_j$. In the simulation, the simulator sets D_j to be equal to random group elements, and simulates the NIZK arguments that D_j commit to the original sets $\mathbb{A}_j$.

(See Sect. 4 for details.) The argument can be verified by using a small number of computations of a bilinear map.

By relying on suitable cryptographic hardness assumptions, from a successful verification it follows that $\chi_{\mathbb{A}_1}(X)\chi_{\mathbb{A}_2}(X) = \chi_{\mathbb{A}_3}(X)\chi_{\mathbb{A}_4}(X)$, and thus the two polynomials $\chi_{\mathbb{A}_1}(X)\chi_{\mathbb{A}_2}(X)$ and $\chi_{\mathbb{A}_3}(X)\chi_{\mathbb{A}_4}(X)$ have the same set of roots with the same multiplicities. Thus, if the PMSET argument verifies, then the verifier is convinced that the prover knows multisets $\mathbb{A}_j$, such that $\mathbb{A}_1 \uplus \mathbb{A}_2 = \mathbb{A}_3 \uplus \mathbb{A}_4$. Moreover, since $\chi_{\mathbb{A}_j}(X)$ is a degree $\leq k_j$ polynomial, the verifier is also convinced that $|\mathbb{A}_j| \leq k_j$.

We actually work in a relaxation of the described model, by allowing $\chi_{\mathbb{A}_j}(X)$ to be *any* polynomial that has $\mathbb{A}_j$ as its null set (again, with correct multiplicities). This somewhat simplifies the argument. Moreover, it allows us to specify parameters k_j such that the prover can additionally convince the verifier that the cardinality of $\mathbb{A}_j$ is not larger than k_j. Thus, we automatically achieve the size-hiding property, required (in particular) in the case of zero-knowledge sets [39]. On the other hand, we can use the upper bound on $|\mathbb{A}_j|$ to guarantee, for exam-

ple, that a voter has approved at most k_j candidates. Without the mentioned relaxation, it seems that the cardinality of $\mathbb{A}_j$ would have to be exactly equal to k_j, where k_j is fixed during the CRS generation.

The length of the new argument is $\Theta(1)$ group elements, while the verifier's computation is dominated by $\Theta(1)$ cryptographic pairings. As a drawback, the CRS length is $\Theta(k^*)$, where $k^* = \max_j k_j$, and the prover's computational complexity is dominated by several k^*-wide bilinear-group multiexponentiations. Although multiexponentiations can be optimized by using the algorithms of Straus [43] and Pippenger [41], they are still costly.

We also provide a version of the PMSET argument that has a smaller CRS length but larger communication and verifier's computation. In the balanced version, all these parameters have complexity $\Theta(\sqrt{k})$. (The prover's computation is still linear in k — this *seems*, although we are not claiming it, to be necessary unless $\mathbb{A}_j$ have a specific structure that one can exploit.)

Applications. We finish the paper by showing how to use the PMSET argument to prove the correct execution of several (multi)set operations. Many applications are possible since any of the multisets $\mathbb{A}_j$ can be either public (e.g., in some applications we can choose $\mathbb{A}_j = \emptyset$ to be public) or committed to, and that we are given flexibility of choosing the values k_j for committed multisets. For example, we obtain arguments for $\mathbb{A}_1 \subseteq \mathbb{A}_2$, $\mathbb{A}_1 = \mathbb{A}_2 \setminus \mathbb{A}_3$, $\mathbb{A}_1 = \mathbb{A}_2 \cup \mathbb{A}_3$, $\mathbb{A}_1 = \mathbb{A}_2 \cap \mathbb{A}_3$, etc.

As another example, we can prove that $\mathbb{A}_1$ is a multiset obtained from $\mathbb{A}_2$ by increasing or decreasing the multiplicity of exactly one (public or committed) element by one. If that element is public, we obtain a dynamic accumulator [11].

Finally, we mention that one can construct a zap (two-message witness-indistinguishable argument, where the verifier's first message can be shared between many protocol executions, [20]) from the new NIZK argument by using standard techniques: basically, the *verifier* creates the CRS, and the prover then replies with the NIZK argument. Such a zap is secure in the standard model, without assuming the existence of a trusted third party who creates the CRS.

2 Preliminaries

Notation. Sets are denoted by blackboard bold uppercase letters as in $\mathbb{A}$. By $\deg(f)$, we denote the degree of the polynomial f. If $h = g^x$ in a group $\mathbb{G}$, then we write $x = \log_g h$. For a group $\mathbb{G}$, we utilize the fact that $\mathbb{G}^2 = \mathbb{G} \times \mathbb{G}$ is a group and thus aggressively use notation like $(g, h)^a$ or $(g_1, h_1) \cdot (g_2, h_2)$. Let NUPPT stand for non-uniform probabilistic polynomial time. A positive function $\varepsilon(\cdot)$ is negligible in its parameter if it decreases faster than the inverse of any polynomial, i.e., $\varepsilon(n) = n^{-\omega(1)}$. By κ, we denote the security parameter.

Sets And Multisets. Formally, a multiset is a 2-tuple $(\mathbb{A}, \mu_{\mathbb{A}})$ where $\mathbb{A}$ is some set and $\mu_{\mathbb{A}} : \mathbb{A} \to \mathbb{N}_{\geq 1}$ is a function from $\mathbb{A}$ to the set $\mathbb{N}_{\geq 1} = \{1, 2, 3, \dots\}$ of

positive natural numbers. The set $\mathbb{A}$ is called the underlying set of elements. For each a in $\mathbb{A}$ the multiplicity of a is the number $\mu_{\mathbb{A}}(a)$. If $\mathbb{A} \subseteq \mathbb{U}$ for some larger set $\mathbb{U}$, then one can extend $\mu_{\mathbb{A}}$ to $\mathbb{U}$, by defining $\mu_{\mathbb{A}}(a) = 0$ for $a \notin \mathbb{A}$. We denote this extended multiplicity function by $\mathbf{1}_{\mathbb{A}}$, and assume its existence implicitly, talking about a multiset $\mathbb{A}$ instead of a multiset $(\mathbb{A}, \mathbf{1}_{\mathbb{A}})$.

If $\mathbb{A}$ and $\mathbb{B}$ are sets, then $\mathbf{1}_{\mathbb{A}}(a) = 1$ if $a \in \mathbb{A}$ and $\mathbf{1}_{\mathbb{A}}(a) = 0$ if $x \notin \mathbb{A}$. If $\mathbb{A}$ and $\mathbb{B}$ are sets, then $\mathbf{1}_{\mathbb{A}\cap\mathbb{B}}(a) = \min\{\mathbf{1}_{\mathbb{A}}(a), \mathbf{1}_{\mathbb{B}}(a)\}$ and $\mathbf{1}_{\mathbb{A}\cup\mathbb{B}}(a) = \max\{\mathbf{1}_{\mathbb{A}}(a), \mathbf{1}_{\mathbb{B}}(a)\}$. We have that $\mathbb{A} \subseteq \mathbb{B}$ iff $\forall a$, $\mathbf{1}_{\mathbb{A}}(a) \leq \mathbf{1}_{\mathbb{B}}(a)$. The cardinality of a finite (multi)set $\mathbb{A}$ is $|\mathbb{A}| = \sum_{a\in\mathbb{U}} \mathbf{1}_{\mathbb{A}}(a)$.

Now, assume that $\mathbb{A}$ and $\mathbb{B}$ are multisets. The multiset sum $\mathbb{A} \uplus \mathbb{B}$ is defined so that $\mathbf{1}_{\mathbb{A}\uplus\mathbb{B}}(i) = \mathbf{1}_{\mathbb{A}}(i) + \mathbf{1}_{\mathbb{B}}(i)$ for all i, and the multiset difference $\mathbb{A} \setminus \mathbb{B}$ is defined so that $\mathbf{1}_{\mathbb{A}\setminus\mathbb{B}}(i) = \max(0, \mathbf{1}_{\mathbb{A}}(i) - \mathbf{1}_{\mathbb{B}}(i))$ for all i. In most of the cases, we just use common set-theoretic operations with multisets. For example, $a \in \mathbb{A}$ means that $\mathbf{1}_{\mathbb{A}}(a) \geq 1$.

Bilinear Groups. Let $\mathcal{G}_{\mathbf{bp}}(1^\kappa)$ be a bilinear group generator that outputs a description of a bilinear group $\mathsf{parm} := (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e}) \leftarrow \mathcal{G}_{\mathbf{bp}}(1^\kappa)$, s.t. p is a κ-bit prime, $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ are multiplicative cyclic groups of order p, $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is a bilinear map (pairing), s.t. $\forall a, b \in \mathbb{Z}_p$ and $g_z \in \mathbb{G}_z$, $\hat{e}(g_1^a, g_2^b) = \hat{e}(g_1, g_2)^{ab}$. If g_z generates $\mathbb{G}_z$ for $z \in \{1, 2\}$, then $\hat{e}(g_1, g_2)$ generates $\mathbb{G}_T$. Deciding the membership in $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$, group operations, the pairing $\hat{e}$, and sampling the generators are efficient, and the descriptions of the groups and group elements are $O(\kappa)$-bit long each. A cryptographic pairing is also required to satisfy some hardness assumptions (see later in this section).

Well-chosen asymmetric pairings (with no efficient isomorphism between $\mathbb{G}_1$ and $\mathbb{G}_2$) are much more efficient than symmetric pairings (where $\mathbb{G}_1 = \mathbb{G}_2$). For $\kappa = 128$, the current recommendation is to use an optimal (asymmetric) Ate pairing [30] over a subclass of Barreto-Naehrig curves [2,40]. In that case, at security level of $\kappa = 128$, an element of $\mathbb{G}_1/\mathbb{G}_2/\mathbb{G}_T$ can be represented in respectively 512/256/3072 bits.

(Λ, u) Trapdoor Commitment Scheme. A trapdoor commitment scheme is a randomized cryptographic primitive (in the common reference string model [8]) that takes a message and outputs a commitment and a trapdoor. It is required to have the following three security properties.

Computational binding: without access to the trapdoor, it is intractable to open the same commitment to two different messages.

Perfect hiding: the commitments of any two messages have the same distribution.

Trapdoor: given an access to the original message, the randomizer and the trapdoor, one can open the commitment to an arbitrary message.

Let $z \in \{1, 2\}$. Assume that $k > 0$ and $u \notin [0, k]$ are public parameters. Let $\Psi_{k,u} := [0, k] \cup \{u\}$. We use the following $([0, k], u)$ *trapdoor commitment scheme* from [21]. For $\mathsf{parm} \leftarrow \mathcal{G}_{\mathbf{bp}}(1^\kappa)$, $g_z \leftarrow_r \mathbb{G}_z \setminus \{1\}$ and the trapdoor $(\sigma, \alpha) \leftarrow \mathbb{Z}_p^2$

(with $\sigma \neq 0$), let the common reference string be $\mathsf{ck} = \left((g_z, g_z^{\alpha})^{\sigma^i}\right)_{i \in \Psi_{k,u}}$. The commmon reference string ck is made public, while the trapdoor (σ, α) is only used in security proofs. Define[3] $\mathsf{com}_{\mathsf{ck}}((a_0, \ldots, a_k); r) := \prod_{i=0}^{k} \left((g_z, g_z^{\alpha})^{\sigma^i}\right)^{a_i} \cdot \left((g_z, g_z^{\alpha})^{\sigma^u}\right)^r = (g_z, g_z^{\alpha})^{r\sigma^u + \sum_{i=0}^{k} a_i \sigma^i}$. The computation of com can be sped up by using efficient multi-exponentiations algorithms [43,41]. Groth [28] and Lipmaa [36] used a similar trapdoor commitment scheme, but with $u = 0$. (See also [27].) In our arguments, the case of an arbitrary u is more suitable, though we can also modify them to work in the case $u = 0$.

Let $\Lambda \subseteq \mathbb{Z}_p$. A bilinear group generator $\mathcal{G}_{\mathbf{bp}}$ is *Λ-PSDL (power symmetric discrete logarithm) secure* [36], if for any NUPPT adversary $\mathcal{A}$, the following probability is negligible in κ:

$$\Pr\left[\begin{array}{l} \mathsf{parm} := (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e}) \leftarrow \mathcal{G}_{\mathbf{bp}}(1^\kappa), g_1 \leftarrow_r \mathbb{G}_1 \setminus \{1\}, \\ g_2 \leftarrow_r \mathbb{G}_2 \setminus \{1\}, \sigma \leftarrow_r \mathbb{Z}_p : \mathcal{A}(\mathsf{parm}; (g_1^{\sigma^i}, g_2^{\sigma^i})_{i \in \Lambda}) = \sigma \end{array}\right].$$

For algorithms $\mathcal{A}$ and $X_{\mathcal{A}}$, we write $(y; y_X) \leftarrow (\mathcal{A}||X_{\mathcal{A}})(\sigma)$ if $\mathcal{A}$ on input σ outputs y, and $X_{\mathcal{A}}$ on the same input (including the random tape of $\mathcal{A}$) outputs y_X. Let $z \in \{1, 2\}$. Let $\Lambda \subset \mathbb{Z}_p$. $\mathcal{G}_{\mathbf{bp}}$ is *Λ-PKE (power knowledge of exponent) secure [28,36] in* $\mathbb{G}_z$ if for any NUPPT $\mathcal{A}$ there exists an NUPPT extractor $X_{\mathcal{A}}$, such that the following probability is negligible in κ:

$$\Pr\left[\begin{array}{l} \mathsf{parm} := (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e}) \leftarrow \mathcal{G}_{\mathbf{bp}}(1^\kappa), g_z \leftarrow_r \mathbb{G}_z \setminus \{1\}, (\alpha, \sigma) \leftarrow_r \mathbb{Z}_p^2, \\ \mathsf{crs} \leftarrow \left(\mathsf{parm}; ((g_z, g_z^{\alpha})^{\sigma^i})_{i \in \Lambda}\right), (c, \hat{c}; (a_i)_{i \in \Lambda}) \leftarrow (\mathcal{A}||X_{\mathcal{A}})(\mathsf{crs}) : \\ \hat{c} = c^{\alpha} \wedge c \neq \prod_{i \in \Lambda} g_z^{a_i \sigma^i} \end{array}\right].$$

Let $z = 1$. Consider a CRS ck that in particular specifies $g_2, \hat{g}_2 \in \mathbb{G}_2$. A commitment $(C, \hat{C}) \in \mathbb{G}_1^2$ is *valid*, if $\hat{e}(C, \hat{g}_2) = \hat{e}(\hat{C}, g_2)$. The case $z = 2$ is dual.

As shown in [21], the $([0, k], u)$ trapdoor commitment scheme is perfectly hiding, and computationally binding under the $\Psi_{k,u}$-PSDL assumption. Moreover, if the $\Psi_{k,u}$-PKE assumption holds, then for any NUPPT $\mathcal{A}$ that outputs a valid commitment C, there exists a NUPPT extractor that, given $\mathcal{A}$'s input together with $\mathcal{A}$'s random coins, extracts a valid opening of C.

Non-Interactive Zero Knowledge (NIZK). NIZK proofs [8] allow the prover to convince the verifier that some input x belongs to some **NP** language $\mathcal{L}$ in the manner that nothing else expect the truth of the statement is revealed. It is well-known that NIZK proofs do not exist without trusted setups unless $\mathbf{P} = \mathbf{NP}$. There are two popular approaches to deal with this. The first approach, the use of random oracle model, results often in very efficient protocols. It is well known [12,26] that some protocols that are secure in the random

[3] Here and in what follows, elements of the form $(g, g^{\alpha})^x$, where α is a secret random key, can be thought of as a *linear-only encoding* of x, see [5] for a discussion

oracle model are non-instantiable in the standard model, and thus the random oracle model is a heuristic at its best.

A better approach is to construct NIZK proofs in the common reference string (CRS) model [8]. Many verifiers can then later independently verify the proof, by having access to the same CRS. The proof has to be complete, sound and satisfy the zero-knowledge property. In practice, one is interested in proofs where both the proof length and verification time are sublinear in the statement size. Such *succinct* proofs cannot be statistically sound, and their soundness cannot be proven under falsifiable assumptions [25]. The latter means that one has to employ knowledge assumptions [16]. A computationally sound proof is also known as an argument. Succinct NIZK arguments have been proposed for languages like Circuit-SAT [28,36,24,3,37], Range [14,21], Set Partition, Subset Sum and Decision Knapsack [21]. While several of these arguments are efficient, they are all highly technical, and based on a careful combination of already complex basic arguments.

More formally, an NIZK argument for a language L consists of three algorithms, $\mathsf{Gen}_{\mathsf{crs}}$, Pro and Ver. The CRS generation algorithm $\mathsf{Gen}_{\mathsf{crs}}$ takes as input 1^κ (and possibly some other, public, language-dependent information) and outputs the prover's CRS crs_p, the verifier's CRS crs_v, and the trapdoor td. (The distinction between crs_p and crs_v is not important for security, but in many applications crs_v is much shorter.) The prover's algorithm Pro takes as an input crs_p together with a statement x and a witness w, and outputs an argument π. The verifier's algorithm Ver takes as an input crs_v together with a statement x and an argument π, and either accepts or rejects.

We expect the argument to be (i) perfectly complete (the honest verifier always accepts the honest prover), (ii) perfectly zero knowledge (there exists an efficient simulator who can, given x, crs_p and td, output an argument that comes from the same distribution as the argument produced by the prover), and (iii) computationally sound (if $x \notin L$, then an arbitrary NUPPT prover has only a negligible success in creating a satisfying argument). We refer to say [28,36] for formal definitions.

3 New Succinct Trapdoor Multiset Commitment Scheme

To succinctly commit to a multiset $\mathbb{A}$, we represent $\mathbb{A}$ as a null set (with multiplicities) of a polynomial. For a multiset $\mathbb{A} \subset \mathbb{Z}_p$, let $\chi_{\mathbb{A}}(X) := \prod_{a\in\mathbb{A}}(X-a)$, where every a has been counted with its multiplicity. For example, $\chi_{\{1,1,2\}}(X) = (X-1)^2(X-2)$.

Let $z \in \{1,2\}$, and let $k = |\mathbb{A}|$ (recall that $|\mathbb{A}|$ includes the multiplicities of all elements) and $u \notin [0,k]$ be again public parameters. To commit to a multiset $\mathbb{A}$, we use the $([0,k],u)$ trapdoor commitment scheme from [21]. Again, we first choose $\mathsf{parm} \leftarrow \mathcal{G}_{\mathbf{bp}}(1^\kappa)$ and $\alpha, \sigma \leftarrow_r \mathbb{Z}_p$, and then set $\mathsf{ck} \leftarrow (\mathsf{parm}, ((g_z, g_z^\alpha)^{\sigma^i})_{i\in\Psi_{k,u}})$ to be the common reference string. We then define $\mathsf{com}_{\mathsf{ck}}(\mathbb{A}; r) := \mathsf{com}_{\mathsf{ck}}(\chi_{\mathbb{A}}(\sigma); r)$. More precisely, the committer assumes that $\chi_{\mathbb{A}}(X) = \sum_{i=0}^{k} s_i X^i$ for some coefficients s_i, and then computes

$\mathsf{com}_{\mathsf{ck}}(\mathbb{A};r) := \prod_{i=0}^{k}\left((g_z,g_z^{\alpha})^{\sigma^i}\right)^{s_i}\cdot\left((g_z,g_z^{\alpha})^{\sigma^u}\right)^{r}$ for $r\leftarrow_r\mathbb{Z}_p$. The trapdoor is equal to $\mathsf{td}\leftarrow(\alpha,\sigma)$.

Theorem 1. *Suppose $z\in\{1,2\}$. The described trapdoor multiset commitment scheme is hiding and, under the $\Psi_{k,u}$-PSDL assumption, computationally binding. If the $\Psi_{k,u}$-PKE assumption holds in $\mathbb{G}_z$, then one can also extract the contents of the commitment.*

Proof. The proof follows [21]. Perfect Hiding: follows from the fact that if r is uniformly random in $\mathbb{Z}_p$, then $g_z^{\chi_{\mathbb{A}}(\sigma)+r\sigma^u}$ is a uniformly random element of $\mathbb{G}_z$ and thus does not depend on $\mathbb{A}$. Computational Binding: assume that an adversary can efficiently produce $(s_1,\ldots,s_k;r)$ and $(s'_1,\ldots,s'_k;r')$ with $s_i\neq s'_i$ for some i, such that $\log_{g_z} c=\sum_{i=0}^{k}s_i\sigma^i+r\sigma^u=\sum_{i=0}^{k}s'_i\sigma^i+r'\sigma^u$. Then $f(X)=\sum_{i=0}^{k}s_iX^i+rX^u$ and $f'(X)=\sum_{i=0}^{k}s'_iX^i+r'X^u$ are two different polynomials. Thus, $d(X)=f(X)-f'(X)$ is a non-zero polynomial such that $d(\sigma)=0$. By using efficient polynomial factorization [34], we can find all possible roots of d, and then find σ by comparing for each root x the value g_z^x with the given g_z^{σ} in ck.

Trapdoor: given td, ck, $(\mathbb{A},r)$, $(C,C')=\mathsf{com}_{\mathsf{ck}}(\mathbb{A};r)$ and $\mathbb{A}'$, one can compute r' such that $(C,C')=\mathsf{com}_{\mathsf{ck}}(\mathbb{A}';r')$ by using the fact that $\log_{g_z} C=\sum s_i\sigma^i+r\sigma^u=\sum s'_i\sigma^i+r'\sigma^u$.

Extraction: follows straightforwardly from the $\Psi_{k,u}$-PKE assumption. □

4 New Pairwise Multiset Sum Equality Test Argument

In a *pairwise multiset sum equality test (PMSET)* argument, the prover aims to convince the prover, that he knows how to open given four commitments C_j to four multisets $\mathbb{A}_j$, for $j\in\{1,2,3,4\}$, such that $\mathbb{A}_1\uplus\mathbb{A}_2=\mathbb{A}_3\uplus\mathbb{A}_4$, where in both sides, the multiplicities of all elements are summed up. That is, we have $\mathbf{1}_{\mathbb{A}_1}(i)+\mathbf{1}_{\mathbb{A}_2}(i)=\mathbf{1}_{\mathbb{A}_3}(i)+\mathbf{1}_{\mathbb{A}_4}(i)$ for all $i\in\mathbb{Z}_p$. In addition to that, one can also upperbound $|\mathbb{A}_j|$ by some public value k_j.

The intuition of the new PMSET argument is as follows. The prover commits to $\mathbb{A}_j$, for $j\in\{1,2,3,4\}$, by using the multiset commitment scheme of Sect. 3. After that, the prover creates a short NIZK argument to show that

$$\chi_{\mathbb{A}_1}(\sigma)\chi_{\mathbb{A}_2}(\sigma)=\chi_{\mathbb{A}_3}(\sigma)\chi_{\mathbb{A}_4}(\sigma)\ . \tag{1}$$

If one does not randomize the commitments, the use of the trapdoor commitment scheme from [21] makes the corresponding NIZK argument relatively (but not completely) straightforward. To take into account the fact that the commitment scheme is randomized, we let the prover also to create a crib E that enables the verifier to verify Eq. (1) on committed elements.

Moreover, due to technical reasons, the prover also has to add extra elements (D_j,Δ_j), $j\in\{1,2,3,4\}$, to the argument. These elements make it possible for the simulator to simulate the NIZK argument, and are necessary since the

commitments C_j are a part of the statement (i.e., the input of the prover) and not a part of the NIZK argument. Here, D_j is basically an alternative random commitment to $\mathbb{A}_j$, while Δ_j is an element that makes it possible to verify that D_j was created correctly. In the simulation, D_j are chosen uniformly and at random, and Δ_j will be set so that the verification still accepts. Such a design also increases the compatibility of our argument; namely the four multisets to be proven can be arbitrarily committed in either $\mathbb{G}_1$ or $\mathbb{G}_2$. This allows the prover to freely compose our arguments for some complex (multi)set relations. Without loss of generality, in the remaining of this section, we assume that all the commitments in the statement are in $\mathbb{G}_1$.

Thus, in the new argument, the prover creates new random commitments D_j to $\mathbb{A}_j$ for $j \in \{1, 2, 3, 4\}$, together with Δ_j and the crib E. Since we will use a knowledge assumption, all elements have an accompanying knowledge component.

By relying on suitable assumptions, from Eq. (1) we obtain that $\chi_{\mathbb{A}_1}(X)\chi_{\mathbb{A}_2}(X) = \chi_{\mathbb{A}_3}(X)\chi_{\mathbb{A}_4}(X)$, and thus in particular $\chi_{\mathbb{A}_1}(X)\chi_{\mathbb{A}_2}(X)$ and $\chi_{\mathbb{A}_3}(X)\chi_{\mathbb{A}_4}(X)$ have the same roots with the same multiplicities. Therefore, the verifier is convinced that $\mathbb{A}_1 \uplus \mathbb{A}_2 = \mathbb{A}_3 \uplus \mathbb{A}_4$ (and due to the use of a knowledge assumption, that the prover actually knows all four multisets).

We relax the multiset commitment scheme of Sect. 3 slightly, by allowing $\chi_{\mathbb{A}_j}(X)$ to be any polynomial that has $\mathbb{A}_j$ as its null set (with correct multiplicities). This relaxation allows us to achieve the following property. Recall that the cardinality of a multiset counts the multiplicities of its elements, $|\mathbb{A}| = \sum_a \mathbf{1}_A(a) = \deg \chi_{\mathbb{A}}(X)$. In the new PMSET argument, one sets an upper bound k_j to the cardinality of the multiset $\mathbb{A}_j$, $|\mathbb{A}_j| \leq k_j$, before creating the CRS. Hence, $\chi_{\mathbb{A}_j}(X) = \sum_{i=0}^{k_j} s_{ji} X^i$ for some coefficients s_{ji}. As we will see later, setting different k_j to related values makes it possible to design interesting variations of the PMSET argument.

We do not know how to achieve such flexibility without the relaxation of the previous paragraph: without it, the committed polynomial $\chi_{\mathbb{A}_j}$ has to be monic, and thus in the committed subset argument one has to check that a specific coefficient of $\chi_{\mathbb{A}_j}$ is equal to 1. This would mean that the cardinality of $\mathbb{A}_j$ has to be known before even creating the CRS. In our case, one just has an upper bound on $|\mathbb{A}_j|$, and thus our arguments are *size-hiding* which allows to build zero-knowledge sets [39].

We note that we have another complication. We divide the commitment scheme into two partial commitment schemes as follows $(\mathsf{com}^1_{\mathsf{ck}}(\mathbb{A}; r), \mathsf{com}^2_{\mathsf{ck}}(\mathbb{A}; r)) \leftarrow \mathsf{com}_{\mathsf{ck}}(\mathbb{A}; r)$. (Thus, com^2 is the knowledge component of the commitment scheme.) Only $\mathsf{com}^1_{\mathsf{ck}}(\mathbb{A}_j; r_j)$ is given as a part of the statement. To obtain soundness, it is necessary that the prover generates $\mathsf{com}^2_{\mathsf{ck}}(\mathbb{A}_j; r_j)$ as a part of the argument.

We now give a formal definition of the new PMSET argument $(\mathsf{Gen}_{\mathsf{crs}}, \mathsf{Pro}, \mathsf{Ver})$. Here, the statement is $(C_j)_{j=1}^4$ where $C_j = \mathsf{com}^1_{\mathsf{ck}}(\mathbb{A}_j; r_j)_{j=1}^4$. On the other hand, the witness is $(\mathbb{A}_j, r_j)_{j=1}^4$. Note that most of the elements g_i^j that are used by the prover or the verifier include a secret component in their

exponent and thus they are computed based on the elements that are a part of the CRS. To avoid filling the variable namespace, we will not assign special variable names for all those elements.

CRS generation $\mathsf{Gen}_{\mathsf{crs}}(1^\kappa, k_1, k_2, k_3, k_4)$**:**

Set $\mathsf{parm} := (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e}) \leftarrow_r \mathcal{G}_{\mathbf{bp}}(1^\kappa)$; Set $g_1 \leftarrow_r \mathbb{G}_1 \setminus \{1\}$ and $g_2 \leftarrow_r \mathbb{G}_2 \setminus \{1\}$; Set $\sigma, \alpha, \beta_1, \beta_2, \beta_3, \beta_4, \eta, \gamma \leftarrow_r \mathbb{Z}_p$ with $\sigma \neq 0$; Set $k^* \leftarrow \max(k_1, k_2, k_3, k_4)$; Set $u \leftarrow k^* + 1$;

For $j \in \{1,2,3,4\}$: Let $z = 1$ if $j \in \{1,3\}$ and $z = 2$ if $j \in \{2,4\}$; Set $\mathsf{ck}_j \leftarrow (((g_z, g_z^{\beta_j})^{\sigma^i})_{i \in \Psi_{k_j,u}})$; Set $\mathsf{ck} \leftarrow ((g_1, g_1^{\alpha})^{\sigma^i})_{i \in \Psi_{k^*,u}}$;

Output

$$\mathsf{crs}_p \leftarrow \left(\mathsf{parm}, \mathsf{ck}, \mathsf{ck}_1, \mathsf{ck}_2, \mathsf{ck}_3, \mathsf{ck}_4, ((g_2, g_2^{\eta})^{\sigma^{i+u}})_{i=0}^{k^*}, (g_2, g_2^{\eta})^{\sigma^{2u}}\right) ,$$

$$\mathsf{crs}_v \leftarrow \left(\mathsf{parm}, g_1, g_2^{\gamma}, g_2^{\sigma^u}, g_1^{\beta_1}, g_1^{\beta_3}, g_2, g_2^{\beta_2}, g_2^{\beta_4}, g_2^{\eta}\right) ,$$

$$\mathsf{td} \leftarrow (\sigma, \alpha, \beta_1, \beta_2, \beta_3, \beta_4, \eta, \gamma) .$$

Prover $\mathsf{Pro}(\mathsf{crs}_p; (C_j)_{j=1}^4; (\mathbb{A}_j, r_j)_{j=1}^4)$**:**

For $j \in \{1,2,3,4\}$:

(i) Write $\chi_{\mathbb{A}_j}(X) = \sum_{i=0}^{k_j} s_{ji} X^i$;
(ii) Set $C'_j \leftarrow \mathsf{com}^2_{\mathsf{ck}}(\mathbb{A}_j; r_j)$;
(iii) Set $r'_j \leftarrow_r \mathbb{Z}_p$;
(iv) Set $(D_j, D'_j) \leftarrow \mathsf{com}_{\mathsf{ck}_j}(\mathbb{A}_j; r'_j)$;
(v) Set $(\Delta_j, \Delta'_j) \leftarrow (g_1, g_1^{\gamma})^{r_j - r'_j}$;

Set

$$\begin{aligned}(E, E') \leftarrow & \prod_{i=0}^{k_1} \left((g_2, g_2^{\eta})^{\sigma^{i+u}}\right)^{r'_2 s_{1i}} \cdot \prod_{i=0}^{k_2} \left((g_2, g_2^{\eta})^{\sigma^{i+u}}\right)^{r'_1 s_{2i}} \cdot \\ & \prod_{i=0}^{k_3} \left((g_2, g_2^{\eta})^{\sigma^{i+u}}\right)^{-r'_4 s_{3i}} \cdot \prod_{i=0}^{k_4} \left((g_2, g_2^{\eta})^{\sigma^{i+u}}\right)^{-r'_3 s_{4i}} \cdot \\ & \left((g_2, g_2^{\eta})^{\sigma^{2u}}\right)^{r'_1 r'_2 - r'_3 r'_4} ;\end{aligned}$$

Output $\pi \leftarrow \left((C'_j, \Delta_j, \Delta'_j, D_j, D'_j)_{j=1}^4, E, E'\right)$;

Verifier $\mathsf{Ver}(\mathsf{crs}_v; (C_j)_{j=1}^4; \pi)$**:** Accept if

(a) Verify knowledge components:
- For $j \in \{1,2,3,4\}$, $\hat{e}(\Delta'_j, g_2) =^? \hat{e}(\Delta_j, g_2^{\gamma})$,
- For $j \in \{1,2,3,4\}$, $\hat{e}(C'_j, g_2) =^? \hat{e}(C_j, g_2^{\alpha})$,
- $\hat{e}(D'_1, g_2) =^? \hat{e}(D_1, g_2^{\beta_1})$, $\hat{e}(g_1, D'_2) =^? \hat{e}(g_1^{\beta_2}, D_2)$, $\hat{e}(D'_3, g_2) =^? \hat{e}(D_3, g_2^{\beta_3})$, $\hat{e}(g_1, D'_4) =^? \hat{e}(g_1^{\beta_4}, D_4)$,
- $\hat{e}(g_1, E') =^? \hat{e}(g_1^{\eta}, E)$,

(b) Verify that C_j and D_j commit to the same multisets:
- For $j \in \{1,3\}$, $\hat{e}(C_j / D_j, g_2) =^? \hat{e}(\Delta_j, g_2^{\sigma^u})$;
- For $j \in \{2,4\}$, $\hat{e}(C_j, g_2) / \hat{e}(g_1, D_j) =^? \hat{e}(\Delta_j, g_2^{\sigma^u})$;

(c) Verify that $A_1 \uplus A_2 =^? A_3 \uplus A_4$: $\hat{e}(g_1, E) =^? \hat{e}(D_1, D_2)/\hat{e}(D_3, D_4)$. Otherwise, reject.

Theorem 2. *The argument of the current subsection is a perfectly complete and perfectly zero-knowledge argument that the prover knows how to open C_j as a multiset $\mathbb{A}_j$ for $j \in \{1,2,3,4\}$, such that $\mathbb{A}_1 \uplus \mathbb{A}_2 = \mathbb{A}_3 \uplus \mathbb{A}_4$ and $|\mathbb{A}_j| \leq k_j$ for $j \in \{1,2,3,4\}$. Let $\Psi_{k^*,u,2u} := [0,k^*] \cup [u,k^*+u] \cup \{2u\}$. Moreover:*

- *If the $\Psi_{k^*,u,2u}$-PSDL, the $\Psi_{k_1,u}$-PKE and $\Psi_{k_3,u}$-PKE assumption in $\mathbb{G}_1$, the $\Psi_{k_2,u}$-PKE and the $\Psi_{k_4,u}$-PKE and the $([u,u+k^*] \cup \{2u\})$-PKE assumption in $\mathbb{G}_2$ hold, then it is computationally sound.*
- *If the $\Psi_{k_1,u}$-PKE assumption and the $\Psi_{k_3,u}$-PKE assumption hold in $\mathbb{G}_1$ and the $\Psi_{k_2,u}$-PKE assumption and the $\Psi_{k_4,u}$-PKE assumption hold in $\mathbb{G}_2$, then it is an argument of knowledge.*

We remark that to simplify the claim, one can combine the the different PKE assumptions into one (stronger than necessary) PKE assumption, but we preferred to state precise assumptions. For example, $(\Psi_1 \cup \Psi_2)$-PKE implies both Ψ_1-PKE and Ψ_2-PKE, but the opposite direction does not necessarily hold.

Proof. Let $h = \hat{e}(g_1, g_2)$. Completeness: It is easy to see that if the prover is honest, then all the equations but the last one hold. For the very last equation, note that since $(\sum_{i=0}^{k_1} s_{1i}\sigma^i)(\sum_{i=0}^{k_2} s_{2i}\sigma^i) = \prod_{i\in\mathbb{A}_1}(\sigma - i) \cdot \prod_{i\in\mathbb{A}_2}(\sigma - i) = \prod_{i\in\mathbb{A}_1\uplus\mathbb{A}_2}(\sigma - i) = \prod_{i\in\mathbb{A}_3\uplus\mathbb{A}_4}(\sigma - i) = \cdots = (\sum_{i=0}^{k_3} s_{3i}\sigma^i)(\sum_{i=0}^{k_4} s_{4i}\sigma^i)$, we get $\log_h \hat{e}(D_1, D_2) = \log_h \hat{e}\left(g_1^{\sum_{i=0}^{k_1} s_{1i}\sigma^i + r'_1\sigma^u}, g_2^{\sum_{i=0}^{k_2} s_{2i}\sigma^i + r'_2\sigma^u}\right) = \left(\sum_{i=0}^{k_1} s_{1i}\sigma^i + r'_1\sigma^u\right)\left(\sum_{i=0}^{k_2} s_{2i}\sigma^i + r'_2\sigma^u\right) = \chi_{\mathbb{A}_1\uplus\mathbb{A}_2}(\sigma) + \sum_{i=0}^{k_1} r'_2 s_{1i}\sigma^{i+u} + \sum_{i=0}^{k_2} r'_1 s_{2i}\sigma^{i+u} + r'_1 r'_2 \sigma^{2u}$, and analogously $\log_h \hat{e}(D_3, D_4) = \chi_{\mathbb{A}_3\uplus\mathbb{A}_4}(\sigma) + \sum_{i=0}^{k_3} r'_4 s_{3i}\sigma^{i+u} + \sum_{i=0}^{k_4} r'_3 s_{4i}\sigma^{i+u} + r'_3 r'_4 \sigma^{2u}$. Thus, $\log_h(\hat{e}(D_1, D_2)/\hat{e}(D_3, D_4)) = \left(\sum_{i=0}^{k_1} r'_2 s_{1i}\sigma^{i+u} + \sum_{i=0}^{k_2} r'_1 s_{2i}\sigma^{i+u}\right) - \left(\sum_{i=0}^{k_3} r'_4 s_{3i}\sigma^{i+u} + \sum_{i=0}^{k_4} r'_3 s_{4i}\sigma^{i+u}\right) + (r'_1 r'_2 - r'_3 r'_4)\,\sigma^{2u} = \log_h E$.

Zero-Knowledge: In the real execution, the variables C_j, D_j, Δ_j, and E are distributed randomly, modulo the last verification equation. Moreover, C'_j, D'_j, Δ'_j, and E' are such that the verification equations on line (a) hold.

The simulator, who knows td but does not know the witness, will simulate the proof as follows.

1. Let $D_1 \leftarrow g_1^{\beta_1^*}$, $D_2 \leftarrow g_2^{\beta_2^*}$, $D_3 \leftarrow g_1^{\beta_3^*}$, $D_4 \leftarrow g_2^{\beta_4^*}$ for $\beta_1^*, \beta_2^*, \beta_3^*, \beta_4^* \leftarrow_r \mathbb{Z}_p$.
2. For $j \in \{1,2,3,4\}$, set $\Delta_j \leftarrow \left(C_j g_1^{-\beta_j^*}\right)^{1/\sigma^u}$. It is obvious that $\hat{e}(C_j/D_j, g_2) = \hat{e}\left(C_j g_1^{-\beta_j^*}, g_2\right) = \hat{e}\left(\Delta_j, g_2^{\sigma^u}\right)$ for $j \in \{1,3\}$ and $\hat{e}(C_j, g_2)/\hat{e}(g_1, D_j) = \hat{e}\left(C_j g_1^{-\beta_j^*}, g_2\right)\hat{e}(g_1, g_2)^{\beta_j^*}/\hat{e}(g_1, D_j) = \hat{e}\left(\Delta_j, g_2^{\sigma^u}\right)$ for $j \in \{2,4\}$.
3. Choose E so that the last verification equation holds, that is, $E \leftarrow g_2^{\beta_1^*\beta_2^* - \beta_3^*\beta_4^*}$. Clearly, $\hat{e}(D_1, D_2)/\hat{e}(D_3, D_4) = \hat{e}(g_1, g_2)^{\beta_1^*\beta_2^* - \beta_3^*\beta_4^*} = \hat{e}(g_1, E)$.

4. Now, set $C'_j \leftarrow C_j^{\alpha}, \Delta'_j \leftarrow \Delta_j^{\gamma}, D'_j \leftarrow D_j^{\beta_j}$ for $j \in \{1,2,3,4\}$, and $E' \leftarrow E^{\eta}$. Such a choice satisfies the verification equations on line (a).
5. Finally, let $\pi \leftarrow ((C'_j, \Delta_j, \Delta'_j, D_j, D'_j)_{j=1}^4, E, E')$.

Since all verifications are satisfied and π comes from the correct distribution, then the simulation has been successful and the argument is perfect zero-knowledge.

Computational soundness: Assume that an adversary $\mathcal{A}$ can break the soundness assumption. We construct another adversary $\mathcal{A}_{psdl}$ that breaks the $\Psi_{k^*,u,2u}$-PSDL assumption as follows.

Assume that all the required knowledge assumptions hold. Therefore, we can extract the following values:

- For $j \in \{1,2,3,4\}$, by the $\Psi_{k_j,u}$-PKE assumption in $\mathbb{G}_1$, from (C_j, C'_j) the adversary obtains a polynomial $f_j(X) = \sum_{i=0}^{k_j} s_{ji}X^i + r_j X^u$, such that $C_j = g_1^{f_j(\sigma)}$.
- For $j \in \{1,2,3,4\}$, by the $\{0\}$-PKE assumption in $\mathbb{G}_2$, from (Δ_j, Δ'_j) the adversary obtains δ_j such that $\Delta_j = g_1^{\delta_j}$. (Note that the $\{0\}$-PKE assumption follows from the $\Psi_{k_j,u}$-PKE assumption.)
- For $j \in \{1,2,3,4\}$: let $z = 1$ for $j \in \{1,3\}$ and $z = 2$ for $j \in \{2,4\}$. By the $\Psi_{k_j,u}$-PKE assumption in $\mathbb{G}_z$, from (D_j, D'_j) the adversary obtains a polynomial $f'_j(X) = \sum_{i=0}^{k_j} s'_{ji}X^i + r'_j X^u$, such that $D_j = g_z^{f'_j(\sigma)}$.
- By the $([u, u+k^*] \cup \{2u\})$-PKE assumption in $\mathbb{G}_2$, from (E, E') the adversary obtains a polynomial $\hat{f}(X) = \sum_{i=0}^{k^*} \hat{s}_i X^{u+i} + \hat{r}X^{2u}$, such that $E = g_2^{\hat{f}(\sigma)}$.

If any extraction does not succeed, then $\mathcal{A}_{psdl}$ aborts (this happens with a negligible probability). Assume now that $\mathcal{A}_{psdl}$ does not abort.

Since for $j \in \{1,3\}$, $\hat{e}(C_j/D_j, g_2) = \hat{e}(\Delta_j, g_2^{\sigma^u})$ holds, we have $f_j(\sigma) - f'_j(\sigma) = \delta_j \sigma^u$. Therefore, if for some i, j, $s_{ji} \neq s'_{ji}$ or $\delta_j \neq t_j - r_j$ we have a non-zero polynomial $d(X) := f_j(X) - f'_j(X) - \delta_j X^u$, such that $d(\sigma) = 0$. Note that $\sigma \neq 0$, so $\mathcal{A}_{psdl}$ uses an efficient polynomial factorization algorithm [34] to find all roots of $d(X)$, and then tests for which root x it holds that (say) $g_1^x = g_1^{\sigma}$. Thus, $\mathcal{A}_{psdl}$ has found σ and broken the $\Psi_{k^*,u}$-PSDL assumption (and thus also the $\Psi_{k^*,u,2u}$-PSDL assumption).

Analogously, $\mathcal{A}_{psdl}$ can break the $\Psi_{k^*,u}$-PSDL assumption if for some i, $s_{ji} \neq s'_{ji}$ or $\delta_j \neq t_j - r_j$ in the case $j \in \{2,4\}$.

Assuming that the adversary did not already break the $\Psi_{k^*,u}$-PSDL assumption, we now have that for all $j \in \{1,2,3,4\}$, (D_j, D'_j) and (C_j, C'_j) commit to the same set, let it be $\mathbb{A}_j$.

Finally, due to the last verification equation, we have $\hat{f}(\sigma) = f'_1(\sigma)f'_2(\sigma) - f'_3(\sigma)f'_4(\sigma)$. This means that, defining $d(X) := f'_1(X)f'_2(X) - f'_3(X)f'_4(X) - \hat{f}(X) = \left(\sum s'_{1i}X^i\right)\left(\sum s'_{2i}X^i\right) - \left(\sum s'_{3i}X^i\right)\left(\sum s'_{4i}X^i\right) + \sum_{i=0}^{k^*} c_i X^{i+u} + c'X^{2u}$ for some coefficients c_i and c', we have $d(\sigma) = 0$.

Since $\mathcal{A}$ succeeded in cheating, it must be the case that $d(X)$ is a non-zero polynomial. But in this case, $\mathcal{A}_{psdl}$ has obtained a non-zero polynomial $d(X)$ where $d(\sigma) = 0$ for some unknown σ. $\mathcal{A}_{psdl}$ uses an efficient polynomial factorization algorithm [34] to find all roots of $d(X)$, and then tests for which

root x it holds that (say) $g_1^x = g_1^\sigma$. Thus, $\mathcal{A}_{psdl}$ has found σ and broken the $\Psi_{k^*,u,2u}$-PSDL assumption.

Thus, (D_j, D'_j) commit to the sets $\mathbb{A}_j$ such that $\mathbb{A}_1 \uplus \mathbb{A}_2 = \mathbb{A}_3 \uplus \mathbb{A}_4$. We have already established before that (C_j, C'_j) and (D_j, D'_j) commit to the same values. The claim follows.

ARGUMENT OF KNOWLEDGE: follows from the last claim of Thm. 1. □

Clearly, the communication complexity of this argument is $\Theta(1)$ group elements and the verifier's computational complexity is dominated by $\Theta(1)$ pairings. The verifier's CRS length contains the parameters parm and $\Theta(1)$ group elements. On the other hand, the prover's CRS length, the CRS computation, and the prover's computation are $\Theta(k)$ group elements or operations respectively. Once again, the computation can be sped up by using efficient multi-exponentiation algorithms [43,41].

Balancing. One can design a balanced version of the new subset argument as follows. Let $k = |\mathbb{A}_1 \uplus \mathbb{A}_2|$. Partition both $\mathbb{A}_1$ and $\mathbb{A}_2$ into $\approx \sqrt{k}$ subsets $\mathbb{A}_{1i}$ and $\mathbb{A}_{2i}$, so that $|\mathbb{A}_{1i} \uplus \mathbb{A}_{2i}| \approx \sqrt{k}$. Partition $\mathbb{A}_3$ and $\mathbb{A}_4$ in a similar way, so that $\mathbb{A}_{1i} \uplus \mathbb{A}_{2i} = \mathbb{A}_{3i} \uplus \mathbb{A}_{4i}$. Now, the PMSET argument that $\mathbb{A}_1 \uplus \mathbb{A}_2 = \mathbb{A}_3 \uplus \mathbb{A}_4$ is just equal to the concatenation of $\sqrt{k}$ PMSET arguments that $\mathbb{A}_{1i} \uplus \mathbb{A}_{2i} = \mathbb{A}_{3i} \uplus \mathbb{A}_{4i}$. Clearly, in this balanced version, the CRS length, the verifier's computation, and the communication are $\Theta(\sqrt{k})$, that is, sublinear in k. On the other hand, the prover's computational complexity is still $\Theta(k)$. However, $\Theta(k)$ total work is clearly a lower bound for arbitrary sets $\mathbb{A}_j$.

5 Applications

Next, we show how to apply the new PMSET argument to construct arguments for standard (multi)set operations, such as intersections, unions, and complements. In such arguments, the prover wants to convince the verifier that its three committed (multi)sets $\mathbb{A}, \mathbb{B}, \mathbb{C}$ satisfy relations like $\mathbb{A} \subseteq \mathbb{B}$, $\mathbb{A} = \mathbb{B} \cap \mathbb{C}$, $\mathbb{A} = \mathbb{B} \cup \mathbb{C}$ or $\mathbb{A} = \mathbb{B} \setminus \mathbb{C}$. We first note that one can clearly modify the PMSET argument so that to allow any subset of $\{\mathbb{A}, \mathbb{B}, \mathbb{C}, \mathbb{D}\}$ to be publicly known sets (e.g., $\mathbb{C} = \emptyset$). This just means that canonical commitments of the public sets are included to the CRS. One has to obviously take care about including only the correct knowledge components to the CRS, but we omit further discussion because of the lack of space.

In what follows, let $\mathbb{U}$ be some publicly known universal set. For efficiency reasons, it is required that $\mathbb{U}$ is not too large; this is usually not a too restrictive assumption. In fact, in many cases $\mathbb{U}$ has been fixed by the application and one has to verify among other things that all sets belong to $\mathbb{U}$. E.g., in the case of e-voting, $\mathbb{U}$ can be the set of all candidates, and in the case of e-auctions, $\mathbb{U}$ can be the set of bids (or in combinatorial auctions, the set of all auctioned goods).

Is-a-Sub(multi)set argument. Clearly, $\mathbb{A} \subseteq \mathbb{B}$ (i.e., $\mathbf{1}_{\mathbb{A}}(a) \leq \mathbf{1}_{\mathbb{B}}(a)$ for all $a \in \mathbb{U}$) iff $\mathbb{A} \uplus \mathbb{C} = \emptyset \uplus \mathbb{B}$, for some (committed) multiset $\mathbb{C}$. Thus, the prover simply provides a commitment to $\mathbb{C}$ as a part of the is-a-subset argument, and then directly utilizes the PMSET argument.

Is-a-Set argument. A committed multiset $\mathbb{A}$ is a set (i.e., $\mathbf{1}_{\mathbb{A}}(a) \leq 1$ for all a) if $\mathbb{A} \subseteq \mathbb{U}$. Thus, for example to show that $\mathbb{A} \subseteq \mathbb{B}$ where $\mathbb{A}$ and $\mathbb{B}$ are both sets, one has to show that $\mathbb{A} \subseteq \mathbb{B}$ and $\mathbb{B} \subseteq \mathbb{U}$ by using the argument from the previous paragraph. Note that having an upper bound on $|\mathbb{C}|$ effectively enforces an lower bound on $|\mathbb{A}|$.

Multiset-Sum argument. Multiset sum is trivial, as $\mathbb{C} = \mathbb{A} \uplus \mathbb{B}$ iff $\mathbb{A} \uplus \mathbb{B} = \mathbb{C} \uplus \emptyset$.

Set-Intersection-And-Union argument. Set intersection and union are closely related. Suppose the prover wants to show that the given four committed sets $\mathbb{A}, \mathbb{B}, \mathbb{C}, \mathbb{D} \subseteq \mathbb{U}$ satisfy $\mathbb{C} = \mathbb{A} \cap \mathbb{B}$ and $\mathbb{D} = \mathbb{A} \cup \mathbb{B}$. For this it is sufficient to show that $\mathbb{A} \uplus \mathbb{B} = \mathbb{C} \uplus \mathbb{D}$, $\mathbb{C} \subseteq \mathbb{A}$, $\mathbb{C} \subseteq \mathbb{B}$ and that $\mathbb{A}$, $\mathbb{B}$ and $\mathbb{D}$ are sets. Really, if $\mathbb{A}$, $\mathbb{B}$ and $\mathbb{D}$ are sets, and $\mathbb{C} \subseteq \mathbb{A}$ then also $\mathbb{C}$ is a set. Thus, for all a, $\mathbf{1}_{\mathbb{A}}(a), \mathbf{1}_{\mathbb{B}}(a), \mathbf{1}_{\mathbb{C}}(a), \mathbf{1}_{\mathbb{D}}(a) \in \{0, 1\}$. If $\mathbf{1}_{\mathbb{A}}(a) = \mathbf{1}_{\mathbb{B}}(a) = 0$, then also $\mathbf{1}_{\mathbb{C}}(a) = \mathbf{1}_{\mathbb{D}}(a) = 0$. If $\mathbf{1}_{\mathbb{A}}(a) = \mathbf{1}_{\mathbb{B}}(a) = 1$, then $\mathbf{1}_{\mathbb{C}}(a) + \mathbf{1}_{\mathbb{D}}(a) = 2$. But since $\mathbb{C}$ and $\mathbb{D}$ are sets, then $\mathbf{1}_{\mathbb{C}}(a) = \mathbf{1}_{\mathbb{D}}(a) = 1$. If $\mathbf{1}_{\mathbb{A}}(a) = 0$ and $\mathbf{1}_{\mathbb{B}}(a) = 1$ (the opposite case is similar), then $\mathbf{1}_{\mathbb{C}}(a) + \mathbf{1}_{\mathbb{D}}(a) = 1$. But since $\mathbb{C} \subseteq \mathbb{A}$, $\mathbf{1}_{\mathbb{C}}(a) = 0$ and $\mathbf{1}_{\mathbb{D}}(a) = 1$. Thus, $\mathbb{C} = \mathbb{A} \cap \mathbb{B}$ and $\mathbb{D} = \mathbb{A} \cup \mathbb{B}$.

Set-Difference argument. To show that committed sets $\mathbb{A}, \mathbb{B}, \mathbb{C} \subseteq \mathbb{U}$ satisfy $\mathbb{A} = \mathbb{B} \setminus \mathbb{C}$ (i.e., $\mathbf{1}_{\mathbb{A}}(a) = \max(0, \mathbf{1}_{\mathbb{B}}(a) - \mathbf{1}_{\mathbb{C}}(a))$ for all a), the prover shows (by using the set-intersection-and-union argument from the previous paragraph) that $\mathbb{A} \cap \mathbb{C} = \emptyset$ and $\mathbb{A} \cup \mathbb{C} = \mathbb{B} \cup \mathbb{C}$. Since $\emptyset$ is not committed to, one can somewhat simplify the resulting argument (e.g., one does not have to verify that $\emptyset \subseteq \mathbb{A}$).

Accumulators. We can extend the applications to the case of cryptographic accumulators [4], where given committed $\mathbb{S}$ and a public k, one has to present a short proof of either $k \in \mathbb{S}$ or $k \notin \mathbb{S}$. In this case, one is traditionally not interested in privacy, but the proofs should be sound. More precisely, given $k \in \mathbb{S}$, we can give a PMSET argument that $\{k\} \cup \mathbb{S}' = \mathbb{S}$ for some committed multiset $\mathbb{S}'$. Similarly, given $k \notin \mathbb{S}$, we can give a PMSET argument that $\{k\} \cup \mathbb{S}'' = \mathbb{U} \setminus \mathbb{S}$ for some committed multiset $\mathbb{S}''$. In both cases, one can additionally use an is-a-set argument to show that $\mathbb{S}$ (or $\mathbb{S}''$, in the $k \notin \mathbb{S}$ case) is a set. This also means that we can implement a dynamic accumulator [11], by first showing that $k \in \mathbb{S}$ (or $k \notin \mathbb{S}$) and then using commitment to $\mathbb{S}'$ as the accumulator for $\mathbb{S} \setminus \{k\}$ (resp., commitment to $\mathbb{S} \cup \{k\}$ as the accumulator for $\mathbb{S} \cup \{k\}$).

Acknowledgments. The first two authors were supported by the Estonian Research Council, and European Union through the European Regional Development Fund. The third author was supported by Project FINER, Greek Secretariat of Research and Technology, and by ERC project CODAMODA.

References

1. Abe, M. (ed.): ASIACRYPT 2010, LNCS, vol. 6477. Springer, Heidelberg
2. Barreto, P.S.L.M., Naehrig, M.: Pairing-Friendly Elliptic Curves of Prime Order. In: Preneel, B., Tavares, S.E. (eds.) SAC 2005. LNCS, vol. 3897, pp. 319–331. Springer, Heidelberg
3. Ben-Sasson, E., Chiesa, A., Genkin, D., Tromer, E., Virza, M.: SNARKs for C: Verifying Program Executions Succinctly and in Zero Knowledge. In: Canetti, R., Garay, J. (eds.) CRYPTO (2) 2013. LNCS, vol. 8043, pp. 90–108. Springer, Heidelberg
4. Benaloh, J., de Mare, M.: One-Way Accumulators: A Decentralized Alternative to Digital Signatures. In: Helleseth, T. (ed.) EUROCRYPT 1993. LNCS, vol. 765, pp. 274–285. Springer, Heidelberg
5. Bitansky, N., Chiesa, A., Ishai, Y., Ostrovsky, R., Paneth, O.: Succinct Non-interactive Arguments via Linear Interactive Proofs. In: Sahai, A. (ed.) TCC 2013. LNCS, vol. 7785, pp. 315–333. Springer, Heidelberg
6. Blanton, M., Aguiar, E.: Private and oblivious set and multiset operations. In: Youm, H.Y., Won, Y. (eds.) ASIACCS 2012. pp. 40–41. ACM
7. Blaze, M. (ed.): FC 2002, LNCS, vol. 2357. Springer, Heidelberg
8. Blum, M., Feldman, P., Micali, S.: Non-Interactive Zero-Knowledge and Its Applications. In: STOC 1988. pp. 103–112. ACM Press (May 2–4, 1988)
9. Boudot, F.: Efficient Proofs That a Committed Number Lies in an Interval. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 431–444. Springer, Heidelberg
10. Camenisch, J., Chaabouni, R., shelat, a.: Efficient Protocols for Set Membership and Range Proofs. In: Pieprzyk, J. (ed.) ASIACRYPT 2008. LNCS, vol. 5350, pp. 234–252. Springer, Heidelberg
11. Camenisch, J., Lysyanskaya, A.: Dynamic Accumulators and Application to Efficient Revocation of Anonymous Credentials. In: Yung, M. (ed.) CRYPTO 2002. LNCS, vol. 2442, pp. 61–76. Springer, Heidelberg
12. Canetti, R., Goldreich, O., Halevi, S.: The Random Oracle Methodology, Revisited. In: Vitter, J.S. (ed.) STOC 1998. pp. 209–218
13. Chaabouni, R., Lipmaa, H., shelat, a.: Additive Combinatorics and Discrete Logarithm Based Range Protocols. In: Steinfeld, R., Hawkes, P. (eds.) ACISP 2010. LNCS, vol. 6168, pp. 336–351. Springer, Heidelberg
14. Chaabouni, R., Lipmaa, H., Zhang, B.: A Non-Interactive Range Proof with Constant Communication. In: Keromytis, A. (ed.) FC 2012. LNCS, vol. 7397, pp. 179–199. Springer, Heidelberg
15. Cramer, R., Gennaro, R., Schoenmakers, B.: A Secure and Optimally Efficient Multi-Authority Election Scheme. In: Fumy, W. (ed.) EUROCRYPT 1997. LNCS, vol. 1233, pp. 103–118. Springer, Heidelberg
16. Damgård, I.: Towards Practical Public Key Systems Secure against Chosen Ciphertext Attacks. In: Feigenbaum, J. (ed.) CRYPTO 1991. LNCS, vol. 576, pp. 445–456. Springer, Heidelberg
17. Damgård, I., Jurik, M.: A Generalisation, a Simplification and Some Applications of Paillier's Probabilistic Public-Key System. In: Kim, K. (ed.) PKC 2001. LNCS, vol. 1992, pp. 119–136. Springer, Heidelberg
18. D'Arco, P., Gonzalez Vasco, M.I., Pérez del Pozo, A.L., Soriente, C.: Size-Hiding in Private Set Intersection: Existential Results and Constructions. In: Mitrokotsa, A., Vaudenay, S. (eds.) AFRICACRYPT 2012. LNCS, vol. 7374, pp. 378–394. Springer, Heidelberg

19. Dimitriou, T., Foteinakis, D.: A Zero Knowledge Proof for Subset Selection from a Family of Sets with Applications to Multiparty/Multicandidate Electronic Elections. LNCS, vol. 3416, pp. 100–111. Springer, Heildeberg
20. Dwork, C., Naor, M.: Zaps and Their Applications. In: FOCS 2000. pp. 283–293. IEEE Computer Society Press
21. Fauzi, P., Lipmaa, H., Zhang, B.: Efficient Modular NIZK Arguments from Shift and Product. In: Abdalla, M., Nita-Rotaru, C., Dahab, R. (eds.) CANS 2013. LNCS, vol. 8257, pp. 92–121. Springer, Heidelberg
22. Fiat, A., Shamir, A.: How to Prove Yourself: Practical Solutions to Identification and Signature Problems. In: Odlyzko, A.M. (ed.) CRYPTO 1986. LNCS, vol. 263, pp. 186–194. Springer, Heidelberg
23. Freedman, M.J., Nissim, K., Pinkas, B.: Efficient Private Matching and Set Intersection. In: Cachin, C., Camenisch, J. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 1–19. Springer, Heidelberg
24. Gennaro, R., Gentry, C., Parno, B., Raykova, M.: Quadratic Span Programs and NIZKs without PCPs. In: Johansson, T., Nguyen, P.Q. (eds.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 626–645. Springer, Heidelberg
25. Gentry, C., Wichs, D.: Separating Succinct Non-Interactive Arguments from All Falsifiable Assumptions. In: Vadhan, S. (ed.) STOC 2011. pp. 99–108. ACM Press
26. Goldwasser, S., Kalai, Y.T.: On the (In)security of the Fiat-Shamir Paradigm. In: FOCS 2003. pp. 102–113. IEEE, IEEE Computer Society Press
27. Golle, P., Jarecki, S., Mironov, I.: Cryptographic Primitives Enforcing Communication and Storage Complexity. In: Blaze [7], pp. 120–135
28. Groth, J.: Short Pairing-Based Non-interactive Zero-Knowledge Arguments. In: Abe [1], pp. 321–340
29. Henry, R., Goldberg, I.: All-but-k Mercurial Commitments and their Applications. Tech. Rep. 26, Centre for Applied Cryptographic Research (Dec 2012), available at `http://cacr.uwaterloo.ca/techreports/2012/cacr2012-26.pdf`
30. Hess, F., Smart, N.P., Vercauteren, F.: The Eta Pairing Revisited. IEEE Transactions on Information Theory 52(10), 4595–4602 (2006)
31. Jarecki, S., Liu, X.: Efficient Oblivious Pseudorandom Function with Applications to Adaptive OT and Secure Computation of Set Intersection. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 577–594. Springer, Heidelberg
32. Kate, A., Zaverucha, G.M., Goldberg, I.: Constant-Size Commitments to Polynomials and Their Applications. In: Abe [1], pp. 177–194
33. Kissner, L., Song, D.: Privacy-Preserving Set Operations. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 241–257. Springer, Heidelberg
34. Lenstra, A.K., Lenstra, Jr., H.W., Lovász, L.: Factoring Polynomials with Rational Coefficients. Mathematische Annalen 261, 513–534 (1982)
35. Lipmaa, H.: On Diophantine Complexity and Statistical Zero-Knowledge Arguments. In: Laih, C.S. (ed.) ASIACRYPT 2003. LNCS, vol. 2894, pp. 398–415. Springer, Heidelberg
36. Lipmaa, H.: Progression-Free Sets and Sublinear Pairing-Based Non-Interactive Zero-Knowledge Arguments. In: Cramer, R. (ed.) TCC 2012. LNCS, vol. 7194, pp. 169–189. Springer, Heidelberg
37. Lipmaa, H.: Succinct Non-Interactive Zero Knowledge Arguments from Span Programs and Linear Error-Correcting Codes. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013 (1). LNCS, vol. 8269, pp. 41–60. Springer, Heidelberg
38. Lipmaa, H., Asokan, N., Niemi, V.: Secure Vickrey Auctions without Threshold Trust. In: Blaze [7], pp. 87–101

39. Micali, S., Rabin, M.O., Kilian, J.: Zero-Knowledge Sets. In: FOCS 2003. pp. 80–91. IEEE, IEEE Computer Society Press
40. Pereira Geovandro, C.C.F., Simplício Jr., M.A., Naehrig, M., Barreto, P.S.L.M.: A Family of Implementation-Friendly BN Elliptic Curves. Journal of Systems and Software 84(8), 1319–1326 (2011)
41. Pippenger, N.: On the Evaluation of Powers and Monomials. SIAM J. Comput. 9(2), 230–250 (1980)
42. Rial, A., Kohlweiss, M., Preneel, B.: Universally Composable Adaptive Priced Oblivious Transfer. In: Shacham, H., Waters, B. (eds.) Pairing 2009. LNCS, vol. 5671, pp. 231–247. Springer, Heidelberg
43. Straus, E.G.: Addition Chains of Vectors. American Mathematical Monthly 70, 806–808 (1964)

A Related Work

Our multiset commitment scheme is a modification of the (what we call a FLZ) commitment scheme [21], which in turn is related to the polynomial commitment scheme of [32]. In [32], the authors proposed a commitment scheme for polynomials f, where instead of committing to the coefficients of f separately, one commits to $f(\sigma)$, where σ is a random key. Their commitment scheme is based on the fact that for any polynomial f, $x-i$ divides $f(x)-f(i)$. Our commitment scheme is somewhat more efficient than the one from [32], since [32] required the randomness r also to be a polynomial. Thus, one needs to generate $\deg(f)$ times more randomness, and the opening of the commitment is also more burdensome. While the need for a new commitment scheme was motivated by the applications considered in [32], it is not necessary in our *distinctively different* applications.

Also, based on their commitment scheme, [32] proposed an NIZK proof that a specific public element belongs to the committed subset, which they named zero knowledge sets. Henry and Goldberg [29] showed that this argument was insecure, and provided a secure improvement. However, both these constructions were interactive, and would either require a random oracle, or be less efficient to get non-interactiveness. We provide a non-interactive implementation without random oracles in our accumulator argument, which is as efficient as both [32] and [29].

The balanced version of our multiset commitment scheme is somewhat similar to the setting in the electronic voting protocol of Dimitriou and Foteinakis [19], which had K disjoint but same size sets $V_1, \cdots V_K$ with total cardinality $C = K \cdot |V_1|$, and a prover commits to S such that $S \subseteq V_i$ for some $i \in [1, K]$. We can directly compare when either $K = 1$ or $K = \sqrt{C} = |V_1|$. But in both cases Dimitriou and Foteinakis require a separate zero-knowledge proof for each candidate, hence the prover's computation, communication and verification are all $\omega(C)$, whereas we have either $\Theta(C)$ prover's computation, $\Theta(\sqrt{C})$ communication and $\Theta(\sqrt{C})$ verification (in the balanced version) or $\Theta(C)$ prover's computation, constant communication and constant verification (in the non-balanced version).

In terms of set operations, there is a lot of related research in the literature. We denote k to be an upper bound for the size of the client's and server's sets (or

the maximum of the two, if an upper bound is not required). Freedman, Nissim and Pinkas presented a two-party private matching and set intersection protocol [23], where the client inputs a private set $\mathbb{C}$, and the server inputs a private set $\mathbb{S}$; if $s_i \in \mathbb{S} \cap \mathbb{C}$, the client learns s_i, otherwise it learns a uniformly random value. The proposed 2-round protocol requires oblivious pseudorandom functions (OPRF) and is provably secure in the random oracle model, but requires $O(k)$ communication. Jarecki and Lim [31] improved upon this and used OPRF to get a 1-round protocol secure in the random oracle model, and a 2-round protocol secure in the CRS model, both cases having $O(k)$ communication. Both protocols reveal the size of the server's set.

Kissner and Song [33] proposed different privacy-preserving set operation protocols that employed the concept of multi-sets. For example, the set union operation is seen as simply the product of the polynomial representations of the two sets. They implement secure set intersection with a fixed and equal size for the client and server sets, using the fact that for random polynomials r, s, $\chi_{\mathbb{A}} r + \chi_{\mathbb{B}} s = \chi_{\mathbb{A}\cap\mathbb{B}} t$ with t having no roots from the universal set $\mathbb{U}$, except for a negligible probability. However, their protocols have $O(k)$ proof size, prover's computation and verification, with the overhead being a proof of correct polynomial multiplication. Moreover, they also have several operations on encrypted polynomials, such as derivatives to reduce duplicated elements of a multiset. These operations are costly, and we choose not to implement them as they will require a product argument as in [21].

There are several other results on private set intersections that are not directly comparable to ours. For example, Blanton and Aguiar [6] had more efficient set operations than the work stated above based on efficient parallelized multi-party operations, but it requires $n > 2$ parties while we focus on two-party protocols. D'Arco *et al.* [18] showed that unconditionally secure size-hiding set intersection is possible with the help of a trusted third party (TTP), given that the client and server have set cardinality at most k. However, the TTP sends output to the client and server based on their specific sets. This means that even for a fixed server set $\mathbb{V}$, the TTP is required for each new client set. Moreover, their 2-round, $O(k)$-communication protocol was only secure in the semi-honest model. Extending it to become a protocol secure against malicious adversaries, the proof size (that is dominated by proof of correct encryption for each of k Paillier ciphertexts) will also become $O(k)$.

We summarize in Table 1. Note that we only include results that either have non-interactive zero knowledge proofs, or can be made non-interactive using the Fiat-Shamir heuristic. None of the work discussed has 1 round (non-interactive), does not require a random oracle and has proof size sublinear in the set cardinality, whereas our set operations have constant-size proof and is secure in the CRS model.