Weights Come Almost for Free in Ramp Threshold Cryptography

Aarushi Goel* Mingyuan Wang†

Abstract

A weighted threshold access structure assigns a weight to each party and authorizes a subset of parties whenever their total weight exceeds a prescribed threshold. Most existing approaches to designing multiparty cryptographic protocols in the weighted threshold setting, however, either incur complexity that grows with the total weight assigned to the parties or remain far from concretely efficient.

Toward *concretely efficient* weighted threshold primitives, we take a different route and relax to the *ramp* setting, in which two thresholds are prescribed: a subset whose total weight falls below an α fraction of the total is unauthorized, while a subset whose total weight exceeds a β fraction is authorized. Exploiting the gap between these two thresholds, we obtain the following results.

1. Given any weighted ramp threshold access structure with privacy threshold α and reconstruction threshold β, where $0 < \alpha < \beta < 1$, we present a weight compression technique that reduces the total weight to $\approx n/(2(\beta-\alpha))$ while preserving the authorized and unauthorized sets. Combined with the standard player-virtualization paradigm, this yields concretely efficient weighted threshold cryptographic protocols with only a multiplicative overhead of $\approx 1/(2(\beta-\alpha))$ over their unweighted counterparts. Previously, the best known compression technique in this setting, due to Benhamouda et al. [ITC'23], achieved total weight $\approx 5n/(\beta-\alpha)$.
2. Player virtualization has an inherent drawback: a party's share size and workload grow in proportion to its (compressed) weight, so the parties in which the most trust is placed also bear the largest computational and communication costs. Thus, although the *overall* overhead is $O(1)$, the cost borne by a single party can still be $O(n)$ times that in the unweighted setting. Our second result removes this imbalance. Building on our compression technique, we construct a linear secret-sharing scheme in which every party receives a share of size $\widetilde{O}(1)$, independent of its weight, for a total share size of $\widetilde{O}(n)$, nearly matching the $O(n)$ complexity of standard unweighted threshold secret sharing.

*Rutgers University aarushi.goel@rutgers.edu
†University of Delware mywang@udel.edu

Contents

1 Introduction **3**

1.1 Our Results 4

1.2 Applications to Silent Threshold Cryptography 6

2 Technical Overview **7**

2.1 Improved Weight Compression 8

2.2 Making Individual Share Size $\widetilde{O}(1)$ 11

3 Improved Weight Compression for Weighted Ramp Threshold Access Structures **15**

4 Linear Secret Sharing for Weighted Ramp Threshold Access Structure with Individual Share Size $O(\log n)$ **19**

AI Statement **23**

References **23**

1 Introduction

A weighted threshold access structure generalizes the standard threshold access structure by assigning a weight to each party: a subset of parties is authorized if its aggregate weight exceeds a prescribed threshold, and unauthorized otherwise. Such access structures capture settings in which parties hold heterogeneous levels of trust, stake, or influence, and they arise naturally in applications such as proof-of-stake blockchains [KRDO17], federated learning [ZMGN20, XLL+24, KNK+25, AIdK23], and electronic voting [BVF+25].

Motivated by these applications, several recent works have studied multiparty cryptographic primitives in the weighted threshold setting, including weighted secure multiparty computation, weighted threshold signatures, weighted (batch) threshold encryption, and weighted verifiable random functions [BW06, GJM+23, DPTX25, GSW26, FG26, ABD+26]. These constructions, however, are either concretely inefficient [BW06, GSW26, FG26] or restricted to small weights, incurring complexity that grows with the total weight assigned to the parties [GJM+23, DPTX25, ABD+26], which can be as large as n^n in the worst case [Mur71, Hås94]. The only exception is a recent line of SNARK-based constructions for weighted threshold signatures [DCX+23, GJM+24, GGW24]; these, however, apply only to signatures and require modifying the underlying signature-verification algorithm, which limits their generality and applicability. This state of affairs raises a natural question: can one construct concretely efficient weighted cryptographic systems, possibly under reasonable relaxations?

Leveraging the Ramp Threshold Setting. A common relaxation of threshold access structures is the *ramp* setting, in which a single threshold is replaced by separate *privacy* and *reconstruction* thresholds: a set of parties whose total weight lies below the privacy threshold is unauthorized, a set whose total weight exceeds the reconstruction threshold is authorized, and no guarantee is imposed on sets whose total weight falls between the two.

This relaxation is attractive for two reasons. First, it is typically a reasonable assumption in practice. Second, the efficiency gains it offers are substantial, and as a result the ramp relaxation is used pervasively. In secret sharing, it is what allows shares to be smaller than the secret itself [Rab89], circumventing the well-known barrier that each share of a perfect threshold scheme must be at least as large as the secret. In secure computation, packed secret sharing [FY92] inherently relies on the ramp relaxation and underlies all known constant-overhead MPC protocols [BGJK21, GPS21, GJMW24, HPW26], which help remove a multiplicative factor of $\Omega(n)$. In the weighted setting, the ramp relaxation allows one to sample a small committee that acts on behalf of the entire universe of parties; protocols then need only be run among the committee, whose size is independent of the total weight, which significantly reduces overhead. This simple idea underlies many practically deployed weighted cryptosystems (e.g., [MRV+21]) and is inherently impossible in the sharp threshold setting.

The recent work of Benhamouda et al. [BHS23] systematically exploited this relaxation and showed that the weights in a weighted ramp threshold access structure can be substantially compressed. In particular, given a weighted ramp threshold access structure with (fractional) privacy threshold α and reconstruction threshold β, where $0 < \alpha < \beta < 1$, they obtain a new weighted ramp threshold access structure with total weight approximately $5n/(\beta-\alpha)$ that preserves authorization: every set that was authorized remains authorized, and every set that was unauthorized remains unauthorized. Combined with the standard *player virtualization* paradigm [Sha79], this yields substantially more efficient constructions of multiparty cryptographic primitives in the weighted ramp threshold setting.

It remains unclear, however, whether this compression is optimal or whether further improvements are possible. Moreover, player virtualization has an inherent drawback: it replaces each party by a

number of virtual parties proportional to its weight, so the computation and communication costs of a party grow with its weight. For instance, compressing the total weight to $O(n)$ reduces the *overall overhead* to $O(1)$, yet a party with large weight may still incur an *individual overhead* of $\Omega(n)$. Thus, parties assigned greater weight, precisely because they are more trusted, must also bear proportionally greater costs. This imbalance persists across all concretely efficient weighted constructions (for small weights) [GJM+23, DPTX25, ABD+26],[1] raising the question of whether trust must inherently be tied to workload.[2] This naturally leads to the main questions we ask in this work:

Can we further improve the efficiency of cryptographic primitives in the weighted ramp threshold setting? Moreover, can we achieve a balanced per-party cost that is independent of the weight, and ideally $\widetilde{O}(1)$?

We answer both questions affirmatively. We first give an improved weight-compression technique for weighted ramp threshold access structures, strengthening the result of Benhamouda et al. [BHS23]. Building on this compression, we then construct a new linear secret-sharing scheme in which every party receives a share of size $\widetilde{O}(1)$, independent of its weight.

1.1 Our Results

We now describe our results and their implications in more detail.

Result I: An Improved Weight Compression Technique. We improve upon the weight compression technique of Benhamouda et al. [BHS23] and obtain the following result.

(Informal) Theorem 1. *Consider a weighted ramp threshold access structure with weights $(w_1, \ldots, w_n)$ and privacy and reconstruction thresholds α and β, respectively, where $0 < \alpha < \beta < 1$. Then, for any $\varepsilon > 0$ such that $\alpha + \varepsilon < \beta - \varepsilon$, there is an efficient algorithm that produces a new weighted ramp threshold access structure with weights $(w'_1, \ldots, w'_n)$ of total weight $W' \leqslant n/(4\varepsilon)$ and thresholds $\alpha' = \alpha + \varepsilon$ and $\beta' = \beta - \varepsilon$, such that every originally unauthorized set remains unauthorized and every originally authorized set remains authorized.*

Taking ε arbitrarily close to $(\beta - \alpha)/2$ gives $W' \leqslant n/(2(\beta - \alpha))$. For the natural choice $\alpha = 1/3$ and $\beta = 2/3$, the total compressed weight is thus arbitrarily close to $3n/2$.

Combined with the standard player-virtualization paradigm, this compression gives a generic transformation from ramp threshold cryptographic primitives to their weighted counterparts, with multiplicative overhead approximately $1/(2(\beta - \alpha))$ relative to the underlying unweighted construction. We highlight a few examples below.

- *Weighted MPC with Constant Overhead.* Order-C MPC protocols achieve total communication [GPS21, GPS22, EGPS22, BGH+23, EGP+23],[3] and in some cases total computation [DIK+08, DIK10,

[1] The only exceptions are constructions based on Yao's garbling technique [BW06, GSW26, FG26], which garble a Boolean circuit that sums the weights of the participating parties.

[2] Another drawback of player virtualization arises when weights change dynamically, as is typical in stake-based blockchains: each party's setup must then provision for the worst case, in which its weight may be as large as $\Omega(n)$. This incurs additional overhead in constructions such as weighted threshold encryption with silent setup [GKPW24, WW26]. We discuss this in more detail in Section 1.2.

[3] While most of these works are in the honest majority setting, [EGP+23] in fact supports the dishonest-majority setting under a ramp-threshold access structure. Since our weight-compression technique applies to arbitrary weighted ramp-threshold access structures, it can be combined directly with their result as well.

BGJK21, GJMW24, HPW26], linear in the size of the function being evaluated, with no multiplicative dependence on the number of parties. As discussed above, these constructions crucially rely on packed Shamir secret sharing [FY92], which inherently operates in the ramp threshold setting. Since our compression outputs an access structure that is itself in the ramp setting, applying player virtualization after compression yields weighted variants of these Order-C protocols with only $O(1)$ overhead.

- *Weighted Threshold Cryptography.* Constructing weighted threshold cryptographic primitives, often with additional features, has attracted significant attention recently, including threshold encryption with silent setup [GKPW24, WW26], batch threshold decryption [ABD+26], and threshold verifiable random functions [DPTX25].[4] In the ramp setting, our compression yields concretely efficient weighted constructions of all these primitives.

Overall, our compression improves the bound $W' \leqslant 5n/(\beta - \alpha)$ of Benhamouda et al. [BHS23] by a factor of approximately 10.[5]

Result II: Linear Secret Sharing with Individual Share Size $\widetilde{O}(1)$. Next, building upon our first result, we design a new linear secret sharing scheme for weighted ramp threshold access structures in which every party receives a share of size $\widetilde{O}(1)$.

(Informal) Theorem 2. *Consider a weighted ramp threshold access structure with weights $(w_1, \ldots, w_n)$ and privacy and reconstruction thresholds α and β, respectively, where $0 < \alpha < \beta < 1$. There exists a linear secret sharing scheme realizing this access structure in which each party receives a share of size $O(\log n)$, and hence the total share size is $O(n \log n)$.*

The hidden constant is larger than in our first result: approximately $32/(\beta - \alpha)^2$ when the weight compression from (Informal) Theorem 1 is applied first, and $8/(\beta - \alpha)^2$ when the input weights already sum to $O(n)$ (see Remark 3). We expect there is considerable room for improvement here.

Compared to naïvely applying player virtualization to the compressed access structure of (Informal) Theorem 1, this construction incurs only an additional multiplicative $O(\log n)$ overhead in the total share size. This modest overhead is outweighed by a significant advantage: every party now receives a share of the same size, independent of both its original and its compressed weight. We now discuss the consequences of this result.

- *Weighted MPC.* Since MPC can be built from any linear secret sharing scheme [CDM00], our second result immediately yields weighted MPC protocols in which each party's cost is independent of its weight. It is unclear to us, however, whether the packing techniques that shave a factor of $\Omega(n)$ off Shamir-based protocols can be applied to an arbitrary linear secret sharing scheme. Without such a gadget, our second result offers no meaningful improvement over the first for MPC: the resulting protocol has total complexity $O(n \log n \cdot |C|)$ for a circuit C, whereas combining (Informal) Theorem 1 with Order-C MPC already achieves $O(|C|)$ in total, with per-party cost that is imbalanced but never worse. We leave this as an exciting future research direction.

[4] As discussed above, threshold signatures (with silent setup) already admit SNARK-based schemes [DCX+23, GJM+24, GGW24] whose weighted variants are as efficient as their unweighted counterparts. These schemes, however, change the verification algorithm. If one insists on retaining a standard verification algorithm, such as that of Schnorr [Sch90] or BLS [BLS01] signatures, our compression again yields concretely efficient weighted variants.

[5] The technical approach underpinning [BHS23] can be pushed to get $W' \leqslant n/(\beta - \alpha)$, but not further, we discuss this in detail in Section 2.

- *Weighted Threshold Cryptography.* Threshold cryptosystems are typically constructed from any linear secret sharing scheme,[6] since they rely only on linearity and not on the additional properties of Shamir's scheme, such as multiplication-friendliness or packing. Our second result therefore immediately yields balanced variants of these threshold cryptosystems, in which every party's cost is $O(\log n)$, independent of its weight.

1.2 Applications to Silent Threshold Cryptography

In many of the motivating applications of threshold cryptography, both the *universe of parties* and *their respective weights* change over time. In a proof-of-stake blockchain, for instance, validators join and leave at every epoch, and the stake of each validator, hence its weight, also changes constantly. In the classical secret-sharing-based approach to threshold cryptography, any such change requires re-running the setup: the parties must engage in a fresh distributed key generation to produce shares of the new secret key, an interactive and expensive procedure that is at odds with the pace at which weights change.

Threshold cryptography with *silent setup* [GJM+24, GKPW24, WW26] was introduced precisely to remove this cost. In silent threshold encryption, for instance, each party independently publishes a public key once and for all, with no interaction; the encryptor then chooses, at encryption time, the set of parties and the threshold for that ciphertext, and any authorized subset can decrypt. Because the encryptor picks the committee, *changes to the universe come for free*: a new party simply publishes a key, and a departing party is simply omitted from future ciphertexts.

Weight changes, however, are a different matter when weights are *realized via player virtualization*.[7] Under virtualization, a party of weight w'_j must publish w'_j keys, one per virtual party, and it must do so at setup time, *before its future weights are known.* It is therefore forced to provision for the worst case. Even after applying the compression of (Informal) Theorem 1, which bounds the *total* compressed weight by $O(n)$, a single party's compressed weight may still be $\Omega(n)$ (consider a party holding a constant fraction of the total stake). Thus, to support arbitrary future weight vectors, every party must publish $\Omega(n)$ keys. Consequently, although (Informal) Theorem 1 guarantees an overall overhead of $O(1)$ when the weights are fixed, in the dynamic setting it only truly yields an (overall) overhead of $\Omega(n)$.

This is where our second result comes in. Suppose, for the moment, that we had a silent threshold encryption scheme in which the encryptor may specify *an arbitrary monotone policy*, as long as it is *realizable by a linear secret sharing scheme of bounded size.* Then a weight update would change only the policy, which is chosen at encryption time, and would require no new setup at all. The only remaining question would be how large a sharing scheme each party must be prepared for, and this is exactly what (Informal) Theorem 2 answers. Recall that our scheme gives every party a share of size $O(\log n)$, and that this bound holds uniformly over all weight vectors: it depends on n and on the gap $\beta - \alpha$, but not on the particular weights. Likewise, the sharing matrix[8] has $O(n \log n)$ rows for every weight vector. Silent weighted threshold encryption with dynamic weights thus reduces to silent monotone-policy encryption for linear secret sharing schemes of these fixed dimensions.

[6] The notable exceptions are schemes with silent setup, which we discuss next.

[7] We note that this does not apply to threshold signatures: as discussed above, the existing silent threshold signature schemes [DCX+23, GJM+24, GGW24] handle weights natively inside a SNARK rather than through virtualization, and so are unaffected.

[8] Recall that a linear secret sharing scheme over a field $\mathbb{F}$ among n parties is specified by a sharing matrix $M \in \mathbb{F}^{R \times W}$ together with a labeling that assigns each of the R rows to one of the parties. To share $s \in \mathbb{F}$, one samples a random vector $\vec{v} \in \mathbb{F}^W$ whose first coordinate is s and hands each party the coordinates of $M\vec{v}$ corresponding to its rows. A party's share size is the number of rows labeled with that party. Without loss of generality $W \leqslant R$, so if every party's share has size at most t, then M has at most nt rows and width at most nt.

Fortunately, the recent work of Waters and Wu [WW26] provides precisely such a scheme. Their construction supports any policy described by a linear secret sharing scheme over n parties in which each party's share consists of at most t field elements, and all of its parameters depend only on the bounds n and t, never on the particular policy. The policy enters only through a deterministic, non-interactive preprocessing step that anyone can run on the published keys to derive the encryption key for that policy. We import their result as follows.

(Imported) Theorem 1 ([WW26, Corollary 4.9])**.** *Under appropriate assumptions in prime-order pairing groups, there is a distributed monotone-policy encryption scheme with silent setup for the class of policies described by a linear secret sharing scheme over n parties in which each party's share consists of at most t field elements (so that the sharing matrix has at most nt rows and width at most nt), with the following efficiency:*

- *the common reference string consists of $O\big((nt)^2\big)$ group elements;*
- *each party's public key consists of $O(t)$ group elements, together with an aggregation hint of $O\big(nt^2\big)$ group elements;*
- *for any policy φ in the class, the aggregation key consists of $O(nt)$ group elements and the aggregated encryption key of 6 group elements, both computed deterministically and non-interactively from the parties' public keys and hints;*
- *a ciphertext consists of 4 group elements together with a short tag, and a party's partial decryption consists of $O(t)$ group elements.*

Instantiating Theorem 1 with the sharing scheme of (Informal) Theorem 2, for which $t = O(\log n)$ for all weight vectors, yields the following.

(Informal) Theorem 3. *Under appropriate assumptions in prime-order pairing groups, there is a silent-setup weighted threshold encryption scheme in the ramp setting, for any privacy and reconstruction thresholds $0 < \alpha < \beta < 1$, that supports* dynamic weight changes *with the following efficiency:*

- Setup. *The common reference string consists of $O\big(n^2 \log^2 n\big)$ group elements. Each party publishes, once and independently of its current or future weight, a public key of $O(\log n)$ group elements together with an aggregation hint of $O\big(n \log^2 n\big)$ group elements.*
- Weight updates. *A change of weights requires no new setup and no interaction: one re-runs the deterministic preprocessing on the published keys, which outputs an aggregation key of $O(n \log n)$ group elements and an encryption key of 6 group elements for the new weight vector.*
- Encryption and decryption. *A ciphertext consists of 4 group elements together with a short tag, and each party's partial decryption consists of $O(\log n)$ group elements.*

Notably, every parameter is within an $O\big(\log^2 n\big)$ factor of the corresponding parameter of the unweighted scheme of Theorem 1 (i.e., with $t = 1$), so the overhead of supporting dynamic weights is only $\widetilde{O}(1)$ (where ciphertexts and encryption keys are exactly the same size).

2 Technical Overview

This section presents the main ideas behind (Informal) Theorems 1 and 2.

2.1 Improved Weight Compression

We start with (Informal) Theorem 1. We are given weights $w_1, \ldots, w_n$ with total weight $W = \sum_i w_i$ and thresholds $(t, T) = (\alpha \cdot W,\ \beta \cdot W)$, so that a set S is unauthorized when $\mathrm{wt}(S) \leqslant t$ and authorized when $\mathrm{wt}(S) \geqslant T$. We wish to replace them by a new (hopefully much lighter) set of weights $w'_1, \ldots, w'_n$ and thresholds (t', T') that preserve authorization: every set authorized (resp. unauthorized) under the original structure should remain authorized (resp. unauthorized) under the new one. We additionally insist that the w'_i be integers, so that the compressed structure can still be realized via player virtualization.

The natural starting point is to normalize the weights,

$$w_1,\ \ldots,\ w_n \quad \Longrightarrow \quad p_1 = \frac{w_1}{W},\ \ldots,\ p_n = \frac{w_n}{W},$$

so that $\sum_i p_i = 1$, and then rescale by a new total weight W' of our choosing:

$$w'_1 = W' \cdot p_1,\ \ldots,\ w'_n = W' \cdot p_n.$$

Of course, there is no reason for $W' \cdot p_i$ to be an integer, so the values must be adjusted. Earlier work [BHS23] takes the natural *local* approach of rounding each coordinate on its own, say $w'_i = \lfloor W' \cdot p_i \rfloor$.

It remains to choose W' so that correctness and soundness survive the rounding. Suppose we set the new privacy threshold to $t' = \lfloor \alpha \cdot W' \rfloor$, then soundness comes for free: every unauthorized S satisfies

$$\sum_{i \in S} \lfloor W' \cdot p_i \rfloor \ \leqslant\ \left\lfloor \sum_{i \in S} W' p_i \right\rfloor \ =\ \left\lfloor W' \cdot \sum_{i \in S} p_i \right\rfloor \ \leqslant\ \lfloor \alpha \cdot W' \rfloor.$$

Correctness is where the rounding costs us. For an authorized S, each coordinate loses up to 1, so

$$\sum_{i \in S} \lfloor W' \cdot p_i \rfloor \ >\ \sum_{i \in S} \left(W' \cdot p_i - 1 \right) \ \geqslant\ \beta \cdot W' - |S| \ \geqslant\ \beta \cdot W' - n,$$

and this must still clear t'. It therefore suffices that

$$\beta \cdot W' - n \geqslant \alpha \cdot W' \qquad \Longrightarrow \qquad W' \ \geqslant\ \frac{n}{\beta - \alpha}.$$

The resulting total weight is upper bounded by W', which introduces an overhead of $1/(\beta - \alpha)$.

Remark 1 (Overhead of [BHS23])**.** *The formal statement of [BHS23] uses $W' = \frac{5n}{\beta - \alpha}$, giving an overhead of $5/(\beta - \alpha)$. The constant is not optimized there, and as we just saw, the same local rounding already delivers $1/(\beta - \alpha)$. More generally, one may ask that the compressed structure retain a gap of its own (for instance, to allow packed secret sharing), say with thresholds $(t', T') = \left(\alpha \cdot W',\ (\alpha + \delta) \cdot W'\right)$ for some $\delta \in (0, \beta - \alpha)$. Repeating the calculation above, correctness will require $\beta \cdot W' - n \geqslant (\alpha + \delta) \cdot W'$, so that*

$$W' \ \geqslant\ \frac{n}{\beta - \alpha - \delta}$$

suffices, i.e., an overhead of $1/(\beta - \alpha - \delta)$.

This overhead, however, appears to be where local rounding stops. Taking floors commits a one-sided error: it only ever undercounts, by up to 1 per party, and nothing prevents all n of those losses from landing inside a single authorized set, so the gap $(\beta - \alpha) \cdot W'$ must absorb a deficit of n. Taking ceilings gives the mirror image: correctness becomes free and soundness pays the same n. One might hope that a smarter

rule helps. Rounding to the nearest integer, for instance, cuts the per-party error to $1/2$. However, this also makes the error two-sided, so the gap must now absorb $n/2$ from below *and* $n/2$ from above,

$$\underbrace{\frac{n}{2}}_{\text{correctness}} + \underbrace{\frac{n}{2}}_{\text{soundness}} \leqslant (\beta - \alpha) \cdot W',$$

which is the condition $W' \geqslant n/(\beta - \alpha)$ all over again. It seems that we have hit a barrier here.

Our Idea: Global Rounding. We depart from the previous local rounding approach and consider rounding all of the weights *jointly*. We do so by first isolating a clean *sufficient but not necessary* condition. Call an integer vector $(w'_1, \ldots, w'_n)$ with $\sum_i w'_i = W'$ an *ε-deviation* rounding of $(p_1, \ldots, p_n)$ if

$$\forall S \subseteq [n], \qquad \left| \frac{1}{W'} \sum_{i \in S} w'_i \; - \; \sum_{i \in S} p_i \right| \leqslant \varepsilon. \tag{1}$$

Both $(p_i)_i$ and $(w'_i/W')_i$ are probability distributions over $[n]$, and Equation (1) asks that no event separate them by more than ε; that is, it says exactly that the *statistical distance* SD between the weight distribution before and after compression is at most ε. We note that this is more than we need: it also pins down the sets that are neither authorized nor unauthorized, about which we do not need to guarantee anything. However, this is a clean condition, and we will see that it significantly simplfies the problem.

Indeed, given such a vector, setting $t' = (\alpha + \varepsilon) \cdot W'$ and $T' = (\beta - \varepsilon) \cdot W'$ preserves both directions at once: an unauthorized S has $\sum_{i \in S} w'_i \leqslant (\alpha+\varepsilon) \cdot W' = t'$, and an authorized S has $\sum_{i \in S} w'_i \geqslant (\beta-\varepsilon) \cdot W' = T'$. The two thresholds describe a sharp threshold structure when $\varepsilon = (\beta - \alpha)/2$ and a ramp structure when $\varepsilon < (\beta - \alpha)/2$. So the question becomes: how large must W' be so that we are able to find new weights ensuring that $\varepsilon \leqslant (\beta - \alpha)/2$?

It turns out that this question has a clean answer, and that the optimal rounding is computed by a simple greedy rule: *Hamilton apportionment* [BY01]. Write $f_i \coloneqq W' \cdot p_i - \lfloor W' \cdot p_i \rfloor$ for the remainder of party i and $k \coloneqq W' - \sum_i \lfloor W' \cdot p_i \rfloor$ for the number of unassigned units (note that $\sum_i f_i = k$). The Hamilton apportionment works as follows (refer to Figure 1 for a pictorial example):

1. First, grants each party the integer part $\lfloor W' \cdot p_i \rfloor$ of its exact quota $W' \cdot p_i$ outright;
2. Next, hands the k still-unassigned units to the k parties with the *largest remainders* (in what follows, we denote this set as U).

Hamilton apportionment states that this is the *optimal* assignment. Indeed, it is not too hard to observe that: for every assignment (with total weight W') different from this rule, one may adjust it to get a smaller deviation (hence, not optimal). Now, observe that the deviation in Equation (1) is maximized by $S = U$ (similar to statistical distance, it is also maximized by $[n] \setminus U$). Therefore, the deviation $\varepsilon \cdot W'$ could be expressed as

$$E \coloneqq \sum_{i \in U} (1 - f_i) \;=\; \sum_{i \notin U} f_i \;=\; \frac{1}{2} \cdot \left\| w' - W' \cdot p \right\|_1.$$

Intuitively, this quantity is maximized (Lemma 1 formally proves this) when

$$f_1 = f_2 = \cdots = f_n = \frac{1}{2}. \tag{2}$$

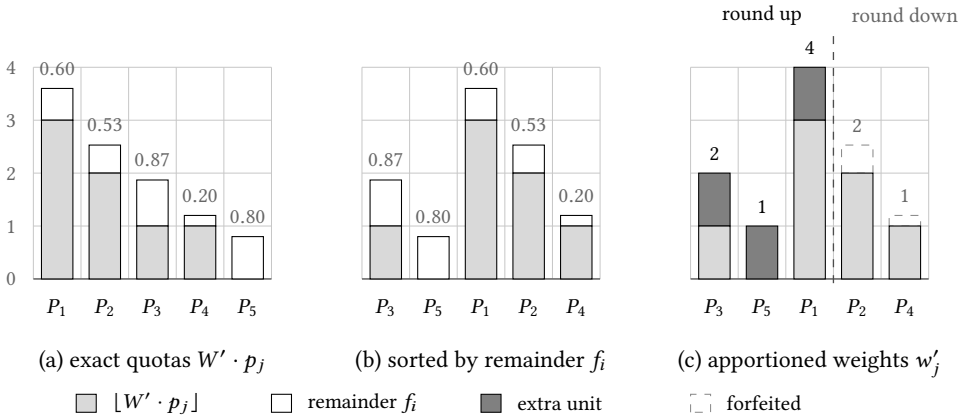


Figure 1: A pictorial example of the Hamilton apportion rule: (a) Party i is drawn as a column of height $W' \cdot p_i$, whose shaded part $\lfloor W' \cdot p_i \rfloor$ is granted outright. (b) The same columns, reordered by decreasing remainder f_i. (c) The final assignment: the k parties with the largest remainders are rounded up and the remaining $n - k$ are rounded down.

Therefore, we conclude that $E \leqslant n/4$ and

$$\varepsilon \leqslant \frac{n/4}{W'}.$$

Thus, it suffices to take

$$W' = \left\lceil \frac{n}{4\varepsilon} \right\rceil.$$

This gives a factor-2 improvement over the local rounding approaches.

Remark 2 (Searching for a smaller W')**.** *We note that, the bound $W' = \lceil n/(4\varepsilon) \rceil$ is an upper bound (based on the worst-case analysis Equation (2)). For any particular weight vector $p_1, \ldots, p_n$, this may be far from tight*[9] *and it is worth searching for a smaller value. One may simply run Hamilton apportionment for $W' = 1, 2, \ldots, \lceil n/(4\varepsilon) \rceil$ in turn and stop at the first value that works. We hereby note that two tests are available at each candidate.*

- The deviation test. *Compute the worst-case deviation $E = \frac{1}{2}\|w' - W' \cdot p\|_1$ of the apportionment directly and check whether $E \leqslant \varepsilon W'$. Including sorting, this takes $O(n \log n)$ time per candidate. As noted earlier, this test is sufficient but not necessary: it certifies all sets including those that are neither authorized nor unauthorized.*

- The exact test. *One can instead check exactly whether authorization is preserved (note that, even if the deviation test fails, authorization might still be preserved). At first glance, it might appear to require exhaustive search over all 2^n sets. However, that is not necessary as it is a* knapsack problem. *Let*

$$m(v) := \min\left\{\sum_{i \in S} w_i \ : \ \sum_{i \in S} w'_i \geqslant v\right\},$$
$$M(v) := \max\left\{\sum_{i \in S} w_i \ : \ \sum_{i \in S} w'_i \leqslant v\right\},$$

[9] Since we are choosing W', each $f_i = W' \cdot p_i - \lfloor W' \cdot p_i \rfloor$ can be very far from 1/2 based on the choice of W'; consequently, we do not expect Equation (2) to be tight.

each computed by a single pass of dynamic programming. Soundness holds if and only if

$$m(\lfloor t' \rfloor + 1) > \alpha W$$

and correctness holds if and only if

$$M(\lceil T' \rceil - 1) < \beta W.$$

This DP takes $O(n \cdot W') = O(n^2)$ time, so each candidate costs $O(n^2)$ time.

2.2 Making Individual Share Size $\widetilde{O}(1)$

We now turn to (Informal) Theorem 2. Our objective is a secret sharing scheme in which *every* party's share size is $\widetilde{O}(1)$, regardless of its weight, so that the overhead is $\widetilde{O}(1)$ in a non-amortized sense. We first describe a base construction achieving share size $O(\sqrt{n})$, and then show that the same ideas can be extended to bring the share size down to $O(\log n)$. Thanks to our weight compression, we assume throughout this section that the weights are positive integers with $W = O(n)$.

Making All Weights $O(\sqrt{n})$. Our first idea is the familiar square-root trick (as in, e.g., square-root ORAM [GO96]): split the parties into a *light* and a *heavy* class, and measure each class in its own unit. In particular, put $\eta := \lceil \sqrt{n} \rceil$ and bipartition the parties into

$$\mathcal{P}_1 := \{\, j \in [n] \;:\; w_j < \eta \,\}, \qquad \mathcal{P}_2 := \{\, j \in [n] \;:\; w_j \geqslant \eta \,\},$$

with associated scales $s_1 := 1$ and $s_2 := \eta$. Light parties are already light enough to virtualize directly; heavy parties, on the other hand, are *quantized with unit size η*, i.e., their weights are rounded down to a multiple of η. That is, we set

$$\forall j \in \mathcal{P}_1,\ w'_j := \left\lfloor \frac{K \cdot w_j}{s_1} \right\rfloor = K \cdot w_j \text{ and } \forall j \in \mathcal{P}_2,\ w'_j := \left\lfloor \frac{K \cdot w_j}{s_2} \right\rfloor = \left\lfloor \frac{K \cdot w_j}{\eta} \right\rfloor,$$

where K is a (constant) scalar to be decided later.

Observe that both classes now have small weights. A light party has $w'_j = K w_j < K\eta = O(\sqrt{n})$ because its weight was small to begin with (and no rounding occurs, as w_j is an integer). A heavy party has

$$w'_j \leqslant \frac{K \cdot w_j}{\eta} \leqslant \frac{K \cdot W}{\eta} = O\left(\sqrt{n}\right)$$

because the unit it is measured in is large.

It remains to see that quantization preserves authorization. For any set $S \subseteq [n]$, define

$$\text{scaled-wt}(S) := \underbrace{\left(\sum_{j \in S \cap \mathcal{P}_1} w'_j\right) \cdot s_1}_{\text{light parties' contribution}} + \underbrace{\left(\sum_{j \in S \cap \mathcal{P}_2} w'_j\right) \cdot s_2}_{\text{heavy parties' contribution}}.$$

We want to argue that scaled-wt(S) can be used in place of wt(S) for deciding whether S is authorized. As with local rounding before, the floor only ever undercounts, so scaled-wt$(S) \leqslant K \cdot \text{wt}(S)$ for every S; setting the new privacy threshold to $\alpha' \cdot K \cdot W$ with $\alpha' := \alpha$, every unauthorized set remains unauthorized.

For correctness, we must account for the loss due to the floor.

- There is no loss on a light party, since Kw_j is already an integer.
- On a heavy party, the floor discards at most one unit (of size η). Since $\eta \leqslant w_j$, this loss is at most a $1/K$ fraction of the party's scaled weight $K \cdot w_j$, namely,

$$K \cdot w_j - \eta \;<\; w'_j \cdot \eta \;\leqslant\; K \cdot w_j \qquad \Longrightarrow \qquad (K-1) \cdot w_j \;<\; w'_j \cdot \eta \;\leqslant\; K \cdot w_j.$$

Summing over all parties $j \in S$ therefore gives

$$(K-1) \cdot \mathsf{wt}(S) \;<\; \text{scaled-wt}(S) \;\leqslant\; K \cdot \mathsf{wt}(S). \tag{3}$$

Now set

$$K \;:=\; \left\lceil \frac{2}{\beta - \alpha} \right\rceil, \qquad \beta' \;:=\; \frac{\alpha + \beta}{2}, \qquad T' \;:=\; \beta' \cdot K \cdot W.$$

For an authorized S, the lower bound in Equation (3) gives

$$\text{scaled-wt}(S) \;>\; (K-1) \cdot \beta \cdot W \;\geqslant\; \frac{\alpha + \beta}{2} \cdot K \cdot W \;=\; T'.$$

Hence, authorized sets remain authorized, and we are left with a slack of $\beta' - \alpha' = (\beta - \alpha)/2$; half of the original, but still $\Omega(n)$ in absolute terms.

To sum up, authorized and unauthorized sets are now defined as

- **Authorized:** $\text{scaled-wt}(S) \geqslant \beta' \cdot K \cdot W$.
- **Unauthorized:** $\text{scaled-wt}(S) \leqslant \alpha' \cdot K \cdot W$,

where $0 < \alpha' < \beta' < 1$ are constants.

Realizing the Scaled Threshold. It remains to see how to realize the scaled threshold access structure. Write

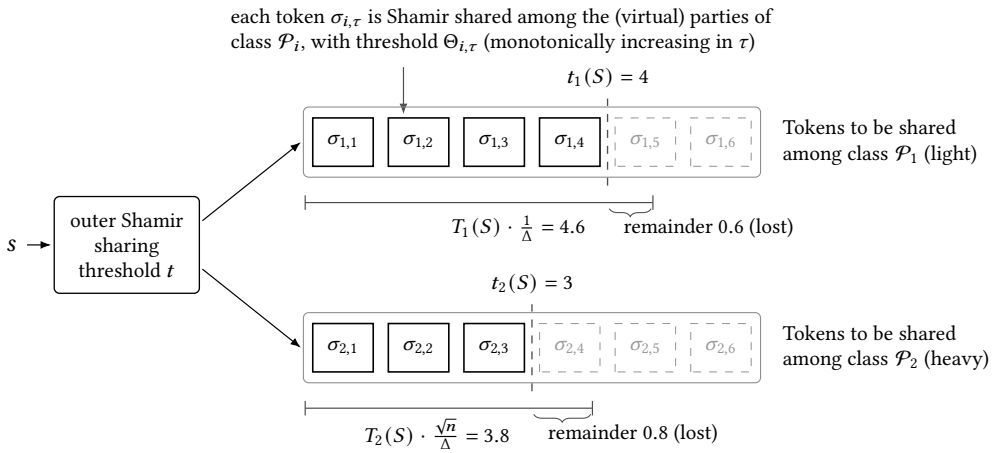
$$T_i(S) \;:=\; \sum_{j \in S \cap \mathcal{P}_i} w'_j \qquad \text{so that} \qquad \text{scaled-wt}(S) \;=\; \underbrace{s_1 \cdot T_1(S)}_{\text{light}} \;+\; \underbrace{s_2 \cdot T_2(S)}_{\text{heavy}}.$$

Each of $s_1 \cdot T_1(S)$ and $s_2 \cdot T_2(S)$ is easy to handle on its own: we have already brought all the weights down to $O(\sqrt{n})$, so virtualization inside $\mathcal{P}_i$ introduces only a $O(\sqrt{n})$ overhead. Jointly, however, the problem becomes highly non-trivial. In particular, it is not clear how to *add* the contributions of the two classes, as they are counted in different units, $s_1 = 1$ and $s_2 = \eta$.

The crucial idea is to measure both contributions in a common unit Δ, to be determined. Let us (again) approximate $\text{scaled-wt}(S)$ by

$$\Bigg(\underbrace{\left\lfloor \frac{s_1 \cdot T_1(S)}{\Delta} \right\rfloor}_{t_1(S)} + \underbrace{\left\lfloor \frac{s_2 \cdot T_2(S)}{\Delta} \right\rfloor}_{t_2(S)} \Bigg) \cdot \Delta \;=\; N(S) \cdot \Delta, \qquad N(S) := t_1(S) + t_2(S).$$

The quantity $N(S)$ counts *how many units of* Δ the light and the heavy parties jointly contribute. Both $t_1(S)$ and $t_2(S)$ are now integers in the same unit, so they can simply be *added*; if this count is high enough the set should be authorized, and if it is low enough it should be unauthorized. To materialize this, two concrete issues remain.



$$N(S) = t_1(S) + t_2(S) = 7, \text{ and } \Delta \cdot N(S) \leqslant \text{scaled-wt}(S) = 8.4\,\Delta < \Delta \cdot N(S) + \underbrace{2\Delta}_{\text{loss}}$$

Figure 2: A concrete example of our two-layer construction. The secret s is shared once by an outer Shamir sharing among all $2m$ tokens with threshold t; each token $\sigma_{i,\tau}$ is then shared again among the virtual parties of its own class $\mathcal{P}_i$, with inner threshold $\Theta_{i,\tau} = \lceil \tau\Delta/s_i \rceil$ for all $\tau \in [m]$. Since these thresholds monotonically increase in τ, a set S recovers a *prefix* of each row (solid) and learns nothing about the rest (dashed). The measuring bar below each row is the class's true contribution $s_i T_i(S)/\Delta$: it reaches only partway into the next token, and that remainder is discarded. Each class therefore loses less than Δ, so $N(S)$ underestimates scaled-wt$(S)/\Delta$ by at most 2 (units of Δ).

- The floors introduce a further loss, which we must account for.
- How do we design a mechanism in which a set S obtains an amount of information proportional to $N(S)$, so that a threshold access structure can be imposed on it?

The first issue is relatively easy to handle. Each floor discards less than one unit, i.e. less than Δ in absolute terms, so with two classes the total loss is less than 2Δ. We have a slack of $(\beta' - \alpha') \cdot K \cdot W$ left over from the previous step, so it suffices to take

$$\Delta \leqslant \frac{(\beta' - \alpha') \cdot K \cdot W}{2}.$$

For the second, our answer is to encode the count in *unary*. We give class $\mathcal{P}_i$ a supply of m *tokens* $\sigma_{i,1}, \ldots, \sigma_{i,m}$ and arrange that S obtains exactly the first $t_i(S)$ tokens, i.e., the prefix $\sigma_{i,1}, \ldots, \sigma_{i,t_i(S)}$. The number of tokens held by S across both classes is then exactly $N(S)$, so a single *outer Shamir sharing* of the secret s among all $2m$ tokens, with threshold

$$t \approx \frac{\alpha' \cdot K \cdot W}{\Delta},$$

tests whether $N(S) \geqslant t$. See Figure 2 for an illustration.

Producing the prefix is exactly where the *inner threshold gadgets* come in. Declare that

$$\sigma_{i,\tau} \text{ certifies the event “class } i \text{ contributes at least } \tau \cdot \Delta\text{”},$$

which is a *threshold condition inside the class.* Converting $\tau \cdot \Delta$ into the class's own unit, we share $\sigma_{i,\tau}$ among the virtual parties of $\mathcal{P}_i$ with threshold

$$\Theta_{i,\tau} \;:=\; \left\lceil \frac{\tau \cdot \Delta}{s_i} \right\rceil.$$

By the guarantee of Shamir's scheme, S recovers $\sigma_{i,\tau}$ if and only if $T_i(S) \geqslant \Theta_{i,\tau}$; and since $T_i(S)$ and τ are both integers,

$$T_i(S) \geqslant \left\lceil \frac{\tau\Delta}{s_i} \right\rceil \iff T_i(S) \geqslant \frac{\tau\Delta}{s_i} \iff \frac{s_i \cdot T_i(S)}{\Delta} \geqslant \tau \iff t_i(S) \geqslant \tau,$$

so the tokens S obtains from class i are exactly the first $t_i(S)$ of them, as required.

It remains to fix the parameter Δ, and here we argue that Δ should be set as large as possible. Observe that a class needs one token for every Δ it could possibly contribute, so $m = \lceil KW/\Delta \rceil$, and each party holds w'_j inner shares for each of the m tokens of its class; the larger Δ is, the smaller m is, and hence the smaller the share size. In other words, the more *fine-grained* we need to be (i.e., the smaller Δ), the more *cost* we must pay. We therefore take the largest permitted value,

$$\Delta \;:=\; \frac{(\beta' - \alpha') \cdot K \cdot W}{2},\; m \;=\; \left\lceil \frac{K \cdot W}{\Delta} \right\rceil = \left\lceil \frac{2}{\beta' - \alpha'} \right\rceil = O(1),\; t \;:=\; \left\lfloor \frac{\alpha' \cdot K \cdot W}{\Delta} \right\rfloor + 1,$$

so that the total quantization loss 2Δ exactly exhausts the available slack.

Correctness and soundness now follow by inspection. If S is authorized then $\text{scaled-wt}(S) \geqslant \beta' KW$, so $\Delta \cdot N(S) > \beta' KW - 2\Delta = \alpha' KW$ and hence $N(S) \geqslant t$; if S is unauthorized then $\Delta \cdot N(S) \leqslant \text{scaled-wt}(S) \leqslant \alpha' KW$ and hence $N(S) \leqslant t - 1$. Finally, a party $j \in \mathcal{P}_i$ participates only in the m inner sharings of its own class, playing w'_j virtual parties in each, so its share consists of

$$m \cdot w'_j \;=\; O(1) \cdot O\left(\sqrt{n}\right) \;=\; O\left(\sqrt{n}\right)$$

field elements. This completes the $\sqrt{n}$ construction.

From $\sqrt{n}$ to $\widetilde{O}(1)$. Nothing in the base construction relies on there being exactly two classes. Suppose instead we take ℓ classes at geometrically increasing scales with ratio R,

$$\mathcal{P}_i = \{\, j : R^{i-1} \leqslant w_j < R^i \,\}, \qquad s_i = R^{i-1}, \qquad R^\ell > W.$$

Every step goes through verbatim. The one hypothesis to recheck is $s_i \leqslant w_j$ for $j \in \mathcal{P}_i$, which is exactly how the classes are defined, and which is what makes the quantization error proportional to each party's own weight rather than additive; the sandwich bound Equation (3) is therefore unchanged. Two quantities do move with R. Within a class, $w_j < R^i = R \cdot s_i$ gives

$$w'_j \;=\; \left\lfloor \frac{K \cdot w_j}{s_i} \right\rfloor \;\leqslant\; K \cdot R,$$

and across classes the quantization loss becomes $\ell \cdot \Delta$ rather than 2Δ, so $\Delta = (\beta' - \alpha')KW/\ell$ and $m = \lceil \ell/(\beta' - \alpha') \rceil = \Theta(\ell)$. The share size is therefore

$$m \cdot w'_j \;=\; O\left(\frac{\ell \cdot R}{(\beta - \alpha)^2}\right), \qquad \text{subject to } R^\ell \geqslant W = O(n).$$

The two factors pull against each other: a larger R means fewer classes, but heavier parties within each. The base construction is the case $R = \sqrt{n}$, $\ell = 2$; writing $\ell = \log_R n$, the cost is

$$O\left(\frac{R \log n}{\log R}\right),$$

which is minimized when $R = 2$.[10] In particular, this gives $\ell = O(\log n)$, $w'_j \leqslant 2K = O(1)$, and share size

$$m \cdot w'_j \;=\; O\left(\frac{\log n}{(\beta - \alpha)^2}\right) \;=\; O(\log n)$$

per party. This completes the overview of our construction.

3 Improved Weight Compression for Weighted Ramp Threshold Access Structures

Given a set of party weights with total weight W in the ramp-threshold setting, this section presents a method for reassigning the parties new weights whose total weight W' is substantially smaller. We simultaneously adjust the privacy and reconstruction thresholds so as to preserve the original access guarantees: every set that is unauthorized under the original weights remains unauthorized with respect to the new privacy threshold, while every set that is authorized under the original weights remains authorized with respect to the new reconstruction threshold.

Choosing New Weights. Let $(w_1, \ldots, w_n)$ be the original set of weights such that $\sum_{i \in [n]} w_i = W$ and let $t = \alpha W$ be the privacy threshold and $T = \beta W$ be reconstruction threshold, where $0 < \alpha < \beta < 1$. We set the new total weight $W' \coloneqq \left\lceil \frac{n}{4\varepsilon} \right\rceil$, for any $0 < \varepsilon < \frac{\beta - \alpha}{2}$. We will now construct new positive integer weights $w'_1, \ldots, w'_n$ whose sum is exactly W'. We will set the new privacy threshold as $t' = (\alpha + \varepsilon)W'$ and the new reconstruction threshold as $T' = (\beta - \varepsilon)W'$.

- **The ideal fractional allocation:** If fractional weights were allowed, the natural way to preserve the normalized weights exactly would be to set the ith new weight as $v_i \coloneqq W' \cdot \frac{w_i}{W}$. Indeed, this would guarantee $\sum_{i=1}^n v_i = W'$ and $\frac{v_i}{W'} = \frac{w_i}{W}$. The only obstacle, however, is that the v_i's need not be integers.
- **Accounting for fractional parts of v_i:** We know that $v_i = \lfloor v_i \rfloor + f_i$, where $f_i \in [0, 1)$ is the fractional part of v_i. Since $\sum_i v_i = W'$ is an integer, $k \coloneqq \sum_{i=1}^n f_i = W' - \sum_{i=1}^n \lfloor v_i \rfloor$ is also an integer. It follows that $\sum_{i=1}^n \lfloor v_i \rfloor = W' - k$. Thus, if every v_i is first rounded down, the resulting total weight is exactly $W' - k$. We must therefore distribute exactly k additional units in order to obtain integer weights whose total is W'.

[10] To be precise, it is minimized at $R = e$, and one could set R to be a non-integer. This does not change the asymptotics, however.

- **Largest-remainder rounding:** We use the classical largest-remainder rounding, also called the *Hamilton apportionment rule* [BY01]. That is, we round up exactly the k coordinates having the largest fractional remainders and round all remaining coordinates down. In particular, let $I \subseteq [n]$ consist of the k indices with the largest fractional parts f_i, breaking ties arbitrarily. We define

$$w'_i := \begin{cases} \lfloor v_i \rfloor + 1, & i \in I, \\ \lfloor v_i \rfloor, & i \notin I. \end{cases}$$

  As a result, we get $\sum_{i=1}^{n} w'_i = \sum_{i=1}^{n} \lfloor v_i \rfloor + k = (W' - k) + k = W'$.

- **Why do we need $\varepsilon < \frac{\beta-\alpha}{2}$:** We require $\varepsilon < \frac{\beta-\alpha}{2}$ to ensure that the new privacy and reconstruction thresholds remain properly ordered, i.e., $t' = W'(\alpha + \varepsilon) < W'(\beta - \varepsilon) = T'$. If this condition were violated, then the new privacy threshold would exceed the new reconstruction threshold, and the resulting parameters would no longer define a valid ramp-threshold structure.

Preserving Authorized and Unauthorized Sets. We next show, through a sequence of lemmas, that for any $\varepsilon > 0$, the above construction preserves the authorized and unauthorized sets of the original weighted access structure under the corresponding new thresholds. We first prove in Lemmas 1 to 3 that, for every subset[11] $S \subseteq [n]$,

$$\left| \sum_{i \in S} \left(\frac{w'_i}{W'} - \frac{w_i}{W} \right) \right| \leqslant \varepsilon.$$

We then show in Lemmas 4 and 5 that this bound implies preservation of all originally unauthorized and authorized sets, respectively. Together these lemmas give us Theorem 4.

Lemma 1 (Bounding the total rounding error)**.** *For each $i \in [n]$, let $e_i := w'_i - v_i$. Then, it holds that $\sum_{i=1}^{n} |e_i| \leqslant \frac{n}{2}$.*

Proof of Lemma 1. Let us reindex the parties, solely for the purpose of this analysis, so that $f_1 \geqslant f_2 \geqslant \cdots \geqslant f_n$. Thus the first k values are rounded upward and the remaining $n - k$ values are rounded downward. In other words,

$$e_i = \begin{cases} 1 - f_i, & 1 \leqslant i \leqslant k, \\ -f_i, & k < i \leqslant n. \end{cases} \qquad \text{and} \qquad \sum_{i=1}^{n} |e_i| = \sum_{i=1}^{k} (1 - f_i) + \sum_{i=k+1}^{n} f_i.$$

Since $\sum_i f_i = k$, we have,

$$\sum_{i=k+1}^{n} f_i = k - \sum_{i=1}^{k} f_i \qquad \text{and} \qquad \sum_{i=1}^{k} (1 - f_i) = k - \sum_{i=1}^{k} f_i.$$

Combining the above, we get $\sum_{i=1}^{n} |e_i| = 2(k - \sum_{i=1}^{k} f_i)$. We next lower bound $\sum_{i=1}^{k} f_i$. Since $f_1, \ldots, f_k$ are the k largest fractional parts, their average is at least the average of all n fractional parts. Therefore, when $k > 0$,

$$\frac{1}{k} \sum_{i=1}^{k} f_i \geqslant \frac{1}{n} \sum_{i=1}^{n} f_i = \frac{k}{n}.$$

[11] To preserve correctness and soundness, it would suffice to establish this property only for sets S that are authorized or unauthorized under the original ramp access structure; for the remaining sets, which fall in the gap between the two thresholds, nothing needs to be guaranteed. Our argument is nonetheless strong enough to cover every $S \subseteq [n]$.

Therefore, $\sum_{i=1}^{k} f_i \geqslant \frac{k^2}{n}$. Note that, we are not accounting for the case when $k = 0$, because in that case all v_i values are already integers and the bound $\sum_{i=1}^{n} |e_i| = 0 \leqslant \frac{n}{2}$ trivially holds. For $k > 0$, the above lower bound on $\sum_{i=1}^{k} f_i$, yields

$$\sum_{i=1}^{n} |e_i| = 2(k - \sum_{i=1}^{k} f_i) \leqslant 2\left(k - \frac{k^2}{n}\right) = \frac{2k(n-k)}{n}.$$

We know that, for any real $k \in [0, n]$, $k(n-k)$ is maximized at $k = n/2$, where its value is $n^2/4$. Hence, we can conclude that $\sum_{i=1}^{n} |e_i| \leqslant \frac{n}{2}$. This completes the proof of Lemma 1.

□

Lemma 2 (Bounding the subset rounding error). *For each $i \in [n]$, let $e_i := w'_i - v_i$. Then, for every $S \subseteq [n]$, it holds that, $|\sum_{i \in S} e_i| \leqslant \frac{n}{4}$.*

Proof of Lemma 2. Let $E^+ := \sum_{i:e_i>0} e_i$ be the total positive rounding error and $E^- := -\sum_{i:e_i<0} e_i$ be the magnitude of the total negative rounding error. Since $\sum_i w'_i = \sum_i v_i = W'$, it holds that $\sum_{i=1}^{n} e_i = 0$ and hence $E^+ = E^-$. Also, $\sum_{i=1}^{n} |e_i| = E^+ + E^- = 2E^+$. From Lemma 1, it then follows that

$$E^+ = E^- = \frac{1}{2}\sum_{i=1}^{n} |e_i| \leqslant \frac{n}{4}.$$

For any arbitrary subset $S \subseteq [n]$, the largest possible positive value of its rounding error $\sum_{i \in S} e_i$ is obtained by including all coordinates with positive error and excluding all coordinates with negative error. Therefore, $\sum_{i \in S} e_i \leqslant E^+ \leqslant \frac{n}{4}$. Similarly, the most negative possible value of this rounding error has magnitude at most E^-, and thus $\sum_{i \in S} e_i \geqslant -E^- \geqslant -\frac{n}{4}$. As a result, the absolute value of this rounding error is bounded by $|\sum_{i \in S} e_i| \leqslant \frac{n}{4}$. This concludes the proof of Lemma 2. □

Lemma 3 (Bounding the error in normalized subset weights). *For any $\varepsilon > 0$ and any subset $S \subseteq [n]$, it holds that $\left|\sum_{i \in S}\left(\frac{w'_i}{W'} - \frac{w_i}{W}\right)\right| \leqslant \varepsilon$.*

Proof of Lemma 3. For each $i \in [n]$, let $p_i := \frac{w_i}{W}$ and $q_i := \frac{w'_i}{W'}$. Since $W = \sum_{i=1}^{n} w_i$, we get $p_i \geqslant 0$ and $\sum_{i=1}^{n} p_i = 1$. Similarly, for the new weights, since $W' = \sum_{i=1}^{n} w'_i$, we get $q_i \geqslant 0$ and $\sum_{i=1}^{n} q_i = 1$. Now, for any $S \subseteq [n]$, we have

$$\left|\sum_{i \in S}(q_i - p_i)\right| = \left|\sum_{i \in S}\left(\frac{w'_i}{W'} - p_i\right)\right| = \frac{1}{W'}\left|\sum_{i \in S}(w'_i - W' \cdot p_i)\right| = \frac{1}{W'}\left|\sum_{i \in S} e_i\right|.$$

From Lemma 2, we get $|\sum_{i \in S} q_i - p_i| \leqslant \frac{n}{4W'}$. Moreover, since $W' = \left\lceil \frac{n}{4\varepsilon} \right\rceil \geqslant \frac{n}{4\varepsilon}$, it implies that $\frac{n}{4W'} \leqslant \varepsilon$. Thus, we have shown that for any $S \subseteq [n]$, it holds that $|\sum_{i \in S} q_i - p_i| \leqslant \varepsilon$. This completes the proof of Lemma 3. □

Lemma 4 (Preservation of unauthorized sets). *If for every $S \subseteq [n]$, it holds that $\left|\sum_{i \in S}\left(\frac{w'_i}{W'} - \frac{w_i}{W}\right)\right| \leqslant \varepsilon$, then it must hold that*

$$\sum_{i \in S} w_i \leqslant \alpha W \quad \Longrightarrow \quad \sum_{i \in S} w'_i \leqslant (\alpha + \varepsilon) W'.$$

Proof of Lemma 4. Let S be unauthorized under the original weights, i.e., $\sum_{i\in S} w_i \leqslant \alpha W$. Moreover, since $\left|\sum_{i\in S}\left(\frac{w'_i}{W'} - \frac{w_i}{W}\right)\right| \leqslant \varepsilon$, we get

$$\sum_{i\in S}\frac{w'_i}{W'} \leqslant \sum_{i\in S}\frac{w_i}{W} + \varepsilon \leqslant \alpha + \varepsilon \implies \sum_{i\in S} w'_i \leqslant (\alpha+\varepsilon)W' = t'.$$

Thus every originally unauthorized set remains unauthorized. This concludes the proof of Lemma 4. □

Lemma 5 (Preservation of authorized sets)**.** *If for every $S \subseteq [n]$, it holds that $\left|\sum_{i\in S}\left(\frac{w'_i}{W'} - \frac{w_i}{W}\right)\right| \leqslant \varepsilon$, then it must hold that*

$$\sum_{i\in S} w_i \geqslant \beta W \quad \implies \quad \sum_{i\in S} w'_i \geqslant (\beta-\varepsilon)W'.$$

Proof of Lemma 5. Let S be authorized under the original weights, i.e., $\sum_{i\in S} w_i \geqslant \beta W$. Moreover, since $\left|\sum_{i\in S}\left(\frac{w'_i}{W'} - \frac{w_i}{W}\right)\right| \leqslant \varepsilon$, we get

$$\sum_{i\in S}\frac{w'_i}{W'} \geqslant \sum_{i\in S}\frac{w_i}{W} - \varepsilon \geqslant \beta - \varepsilon \implies \sum_{i\in S} w'_i \geqslant (\beta-\varepsilon)W' = T'.$$

Thus every originally authorized set remains authorized. This concludes the proof of Lemma 5. □

Our First Result. Combining Lemmas 3 to 5 yields the following theorem.

Theorem 4. *Let $(w_1,\ldots,w_n)$ be a set of positive integer weights with total weight W and let $t = \alpha W$ and $T = \beta W$ be ramp thresholds, where $0 < \alpha < \beta < 1$. For any $0 < \varepsilon < \frac{\beta-\alpha}{2}$, there exists an efficient algorithm that produces nonnegative integer weights $(w'_1,\ldots,w'_n)$ with total weight $W' = \left\lceil \frac{n}{4\varepsilon} \right\rceil$, such that, for every $S \subseteq [n]$,*

$$\begin{aligned}
\sum_{i\in S} w_i \leqslant \alpha W \quad &\implies \quad \sum_{i\in S} w'_i \leqslant (\alpha+\varepsilon)W', \text{ and} \\
\sum_{i\in S} w_i \geqslant \beta W \quad &\implies \quad \sum_{i\in S} w'_i \geqslant (\beta-\varepsilon)W'.
\end{aligned}$$

Thus, every unauthorized set in the original weighted access structure remains unauthorized with respect to the new lower threshold $t' = (\alpha+\varepsilon)W'$, and every authorized set remains authorized with respect to the new upper threshold $T' = (\beta-\varepsilon)W'$.

Setting ε to be arbitrarily close to $\frac{\beta-\alpha}{2}$, results in the total weight of the new weighted access structure to be $W' = \left\lceil \frac{n}{2(\beta-\alpha)} \right\rceil$.

Weighted Ramp Threshold Cryptosystems via Player Virtualization. The above weight-reduction procedure can be combined with the standard player-virtualization paradigm to obtain concretely efficient weighted ramp-threshold cryptosystems. In particular, given a weighted ramp-threshold access structure, we can first apply the above procedure to obtain a new weighted ramp-threshold access structure with total weight $W' = \left\lceil \frac{n}{4\varepsilon} \right\rceil$, while preserving all originally authorized and unauthorized sets under the corresponding shifted thresholds. We can then apply player virtualization to any existing threshold cryptosystem

supporting the resulting ramp thresholds: party P_i is represented by w'_i virtual players, and the underlying threshold cryptosystem is executed among the resulting W' virtual players.

Since the total number of virtual players is only $W' = \left\lceil \frac{n}{4\varepsilon} \right\rceil$, this will only incur an aggregate overhead of at most[12] $1/(4\varepsilon)$ relative to the ramp threshold access structure in the unweighted setting. For instance, when $\alpha = 1/3$ and $\beta = 2/3$, this only introduces a constant overhead of $3/2$. This is substantially more efficient than applying player virtualization directly to the original weighted access structure, whose total weight W may be as large as $O(n^n)$ in the worst case [Mur71, Hås94].

4 Linear Secret Sharing for Weighted Ramp Threshold Access Structure with Individual Share Size $O(\log n)$

We now present our second result. We will present our construction first. The formal theorem is stated as Theorem 5. Without loss of generality, we assume that

$$W = w_1 + \cdots + w_n = O(n).$$

Indeed, we may first invoke Theorem 4 to replace the given weights by a new set of weights whose total weight is linear in n. Doing so increases α and decreases β by an arbitrarily small constant of our choosing, so both remain constants in $(0, 1)$ with $\alpha < \beta$. We further assume that $w_j \geqslant 1$ for every $j \in [n]$, since parties of weight 0 may simply be discarded.

Our Construction. To share a secret $s \in \mathbb{F}$, the dealer proceeds as follows.

1. **Partition the Parties.** Let $\ell \in \mathbb{N}$ be the smallest integer such that $2^\ell > W$; since $W = O(n)$, we have $\ell = \lfloor \log_2 W \rfloor + 1 = O(\log n)$. Split the parties into the ℓ classes

$$\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_\ell \subseteq [n], \qquad \mathcal{P}_i := \left\{ j \in [n] \;\middle|\; 2^{i-1} \leqslant w_j < 2^i \right\}.$$

   As $1 \leqslant w_j \leqslant W < 2^\ell$ for every j, each party lies in exactly one class.

2. **Scale Each Class.** Let $K := \left\lceil \frac{2}{\beta - \alpha} \right\rceil$; as α, β are constants, so is K. For every $i \in [\ell]$ and every $j \in \mathcal{P}_i$, set

$$w'_j := \left\lfloor \frac{K \cdot w_j}{2^{i-1}} \right\rfloor.$$

   Since $2^{i-1} \leqslant w_j < 2^i$ for $j \in \mathcal{P}_i$, these scaled weights satisfy

$$\forall j \in [n], \qquad K \leqslant w'_j \leqslant 2K = O(1).$$

3. **Outer Secret Sharing.** Set

$$\alpha' := \alpha, \qquad \beta' := \frac{\alpha + \beta}{2}, \qquad \Delta := \frac{(\beta' - \alpha') \cdot K \cdot W}{\ell},$$

$$m := \left\lceil \frac{K \cdot W}{\Delta} \right\rceil = \left\lceil \frac{\ell}{\beta' - \alpha'} \right\rceil, \qquad t := \left\lfloor \frac{\alpha' \cdot K \cdot W}{\Delta} \right\rfloor + 1 = \left\lfloor \frac{\alpha' \cdot \ell}{\beta' - \alpha'} \right\rfloor + 1.$$

[12] Accounting for the ceiling, the precise bound is $1/(4\varepsilon) + 1/n$; we suppress the additive $1/n$ term.

Additionally, for all $i \in [\ell]$ and $\tau \in [m]$,

$$\Theta_{i,\tau} := \left\lceil \frac{\tau \cdot \Delta}{2^{i-1}} \right\rceil.$$

Share s using Shamir's secret sharing among $\ell \cdot m$ parties with threshold t, obtaining the *outer shares* $\{\sigma_{i,\tau}\}_{i\in[\ell],\tau\in[m]}$. These outer shares *will not* be given to any (virtual) party.

4. **Inner Secret Sharing.** For every pair $(i, \tau) \in [\ell] \times [m]$, share $\sigma_{i,\tau}$ using Shamir's secret sharing among the virtual parties of $\mathcal{P}_i$ with threshold $\Theta_{i,\tau}$, where party $j \in \mathcal{P}_i$ plays the role of w'_j virtual parties and hence receives w'_j (inner) shares.

Reconstruction is the natural one: from the inner shares parties hold, they first recover every outer share that they are able to recover, and then reconstruct s from these outer shares.

We have the following theorem about our construction. The rest of the section proves this theorem.

Theorem 5. *For any set of weights $w_1, \ldots, w_n$ and threshold ($t = \alpha \cdot W$ and $T = \beta \cdot W$). Our construction is a linear secret sharing scheme realizing this weighted ramp threshold access structure with individual share size is $O(\log n)$.*

Efficiency. We argue efficiency first since it is straightforward. A party $j \in \mathcal{P}_i$ takes part in the inner sharing of the m outer shares $\sigma_{i,1}, \ldots, \sigma_{i,m}$ of its own class only, and there plays the role of w'_j virtual parties. Its share therefore consists of

$$m \cdot w'_j \leqslant \left\lceil \frac{\ell}{\beta' - \alpha'} \right\rceil \cdot 2K = O\left(\frac{\ell}{(\beta - \alpha)^2} \right) = O(\log n)$$

inner shares (i.e., field elements), so the total share size is $O(n \log n)$ field elements. Observe that, for Shamir's secret sharing, any field $\mathbb{F}$ with $|\mathbb{F}| > \max\left\{\ell \cdot m,\ \max_{i\in[\ell]} \sum_{j\in\mathcal{P}_i} w'_j\right\}$ may be used. Since $\ell \cdot m = O\left(\log^2 n\right)$ and $\sum_{j\in[n]} w'_j \leqslant 2K \cdot n = O(n)$, a field of size $O(n)$ suffices.

Correctness and Soundness. We next argue correctness and soundness. Let Γ denote the original ramp access structure. We show that every set of parties that is authorized in Γ reconstructs the secret, and that every unauthorized set in Γ learns nothing about it. This is established by the following sequence of lemmas.

Lemma 6 (Scaling preserves the access structure)**.** *For a set $S \subseteq [n]$, define*

$$\text{scaled-wt}(S) := \sum_{i=1}^{\ell} \left(\sum_{j \in S \cap \mathcal{P}_i} w'_j \right) \cdot 2^{i-1},$$

and let Γ' be the ramp access structure in which

- *S is unauthorized if* $\text{scaled-wt}(S) \leqslant \alpha' \cdot (K \cdot W)$*, and*
- *S is authorized if* $\text{scaled-wt}(S) \geqslant \beta' \cdot (K \cdot W)$.

Then every set that is authorized (resp., unauthorized) in the original weighted ramp access structure Γ remains authorized (resp., unauthorized) in Γ'. Moreover, $\beta' - \alpha' = (\beta - \alpha)/2$ is again a constant in $(0, 1)$.

Proof of Lemma 6. Fix $j \in \mathcal{P}_i$. Since $x - 1 < \lfloor x \rfloor \leqslant x$ for all x,

$$\frac{K w_j}{2^{i-1}} - 1 < w'_j \leqslant \frac{K w_j}{2^{i-1}} \qquad \Longrightarrow \qquad K \cdot w_j - 2^{i-1} < w'_j \cdot 2^{i-1} \leqslant K \cdot w_j.$$

As $j \in \mathcal{P}_i$ gives $2^{i-1} \leqslant w_j$, this yields

$$(K-1) \cdot w_j \;<\; w'_j \cdot 2^{i-1} \;\leqslant\; K \cdot w_j. \tag{4}$$

Summing Equation (4) over all $j \in S$, grouped by class, gives for every $S \subseteq [n]$

$$(K-1) \cdot \sum_{j \in S} w_j \;<\; \text{scaled-wt}(S) \;\leqslant\; K \cdot \sum_{j \in S} w_j. \tag{5}$$

Unauthorized sets. If S is unauthorized in Γ, then $\sum_{j \in S} w_j \leqslant \alpha \cdot W$, and the upper bound in Equation (5) gives

$$\text{scaled-wt}(S) \leqslant K \cdot \sum_{j \in S} w_j \leqslant \alpha \cdot (K \cdot W) = \alpha' \cdot (K \cdot W).$$

Authorized sets. If S is authorized in Γ, then $\sum_{j \in S} w_j \geqslant \beta \cdot W$, and the lower bound in Equation (5) gives

$$\text{scaled-wt}(S) > (K-1) \cdot \beta \cdot W \geqslant \beta \cdot (K \cdot W) - W \geqslant \beta \cdot (K \cdot W) - \frac{\beta - \alpha}{2} \cdot (K \cdot W) = \beta' \cdot (K \cdot W),$$

using $\beta \leqslant 1$ in the second inequality and $K \geqslant 2/(\beta - \alpha)$ in the third. □

By Lemma 6 it suffices from now on to analyse the scheme with respect to the new ramp access structure Γ'. For $S \subseteq [n]$ and $i \in [\ell]$ define

$$T_i(S) := \sum_{j \in S \cap \mathcal{P}_i} w'_j, \qquad t_i(S) := \left\lfloor \frac{2^{i-1} \cdot T_i(S)}{\Delta} \right\rfloor, \qquad N(S) := \sum_{i=1}^{\ell} t_i(S),$$

so that $\text{scaled-wt}(S) = \sum_{i=1}^{\ell} 2^{i-1} \cdot T_i(S)$. The next lemma identifies $N(S)$ as the number of outer shares available to S.

Lemma 7 (Recoverable outer shares)**.** *For every $S \subseteq [n]$ and every $i \in [\ell]$ we have $t_i(S) \in \{0, 1, \ldots, m\}$, and for every $\tau \in [m]$,*

$$S \text{ can recover } \sigma_{i,\tau} \qquad \Longleftrightarrow \qquad \tau \leqslant t_i(S).$$

Consequently the set

$$R(S) := \{(i, \tau) \in [\ell] \times [m] \;:\; \tau \leqslant t_i(S)\}$$

of outer shares recoverable by S has size exactly $|R(S)| = N(S)$.

Proof of Lemma 7. The outer share $\sigma_{i,\tau}$ is shared among the virtual parties of $\mathcal{P}_i$ with threshold $\Theta_{i,\tau}$, and the parties of $S \cap \mathcal{P}_i$ jointly hold exactly $T_i(S)$ of these virtual shares. By the correctness and privacy of Shamir's scheme, S can recover $\sigma_{i,\tau}$ if and only if $T_i(S) \geqslant \Theta_{i,\tau}$. Since $T_i(S)$ and τ are integers,

$$T_i(S) \geqslant \Theta_{i,\tau} = \left\lceil \frac{\tau \cdot \Delta}{2^{i-1}} \right\rceil \iff T_i(S) \geqslant \frac{\tau \cdot \Delta}{2^{i-1}} \iff \frac{2^{i-1} \cdot T_i(S)}{\Delta} \geqslant \tau \iff t_i(S) \geqslant \tau.$$

As for the range of $t_i(S)$: it is non-negative by definition, and the upper bound in Equation (5) applied to $S = [n]$ gives $2^{i-1} \cdot T_i(S) \leqslant \text{scaled-wt}([n]) \leqslant K \cdot W$, whence $t_i(S) \leqslant \lfloor KW/\Delta \rfloor \leqslant m$. Therefore, for each fixed i, exactly $|\{\tau \in [m] : \tau \leqslant t_i(S)\}| = t_i(S)$ outer shares of the i-th class are recoverable; summing over i gives $|R(S)| = N(S)$. □

The next two lemmas establish the relation between Γ' and $N(S)$.

Lemma 8 (Sandwich bound)**.** *For every $S \subseteq [n]$,*

$$\Delta \cdot N(S) \;\leqslant\; \text{scaled-wt}(S) \;<\; \Delta \cdot N(S) + (\beta' - \alpha') \cdot K \cdot W. \tag{6}$$

Proof of Lemma 8. By the definition of $t_i(S)$ we have $\Delta \cdot t_i(S) \leqslant 2^{i-1} \cdot T_i(S) < \Delta \cdot \big(t_i(S) + 1\big)$ for every $i \in [\ell]$. Summing over $i \in [\ell]$ and using $\text{scaled-wt}(S) = \sum_i 2^{i-1} T_i(S)$ gives

$$\Delta \cdot N(S) \;\leqslant\; \text{scaled-wt}(S) \;<\; \Delta \cdot N(S) + \ell \cdot \Delta,$$

and the claim follows from $\ell \cdot \Delta = (\beta' - \alpha') \cdot K \cdot W$. □

Lemma 9 (Threshold gap)**.** *Let $S \subseteq [n]$. If S is authorized in Γ', then $N(S) \geqslant t$; if S is unauthorized in Γ', then $N(S) \leqslant t - 1$.*

Proof of Lemma 9. Suppose first that $\text{scaled-wt}(S) \geqslant \beta' \cdot K \cdot W$. The right-hand inequality of Equation (6) gives

$$\Delta \cdot N(S) \;>\; \text{scaled-wt}(S) - (\beta' - \alpha') \cdot K \cdot W \;\geqslant\; \alpha' \cdot K \cdot W,$$

so $N(S) > \alpha' KW/\Delta$; as $N(S)$ is an integer, $N(S) \geqslant \lfloor \alpha' KW/\Delta \rfloor + 1 = t$.

Suppose instead that $\text{scaled-wt}(S) \leqslant \alpha' \cdot K \cdot W$. The left-hand inequality of Equation (6) gives $\Delta \cdot N(S) \leqslant \alpha' \cdot K \cdot W$, so $N(S) \leqslant \alpha' KW/\Delta$ and, again by integrality, $N(S) \leqslant \lfloor \alpha' KW/\Delta \rfloor = t - 1$. □

The next lemma routinely argues that the inner shares held by S reveal nothing beyond the outer shares that S can already recover (i.e., $R(S)$).

Lemma 10 (View simulation)**.** *For every $S \subseteq [n]$ there is a simulator Sim_S that, on input $\{\sigma_{i,\tau}\}_{(i,\tau) \in R(S)}$, outputs a distribution identical to the joint distribution of the shares held by the parties of S.*

Proof of Lemma 10. The dealer's randomness consists of the randomness of the outer sharing together with that of $\ell \cdot m$ mutually independent inner sharings, one per pair (i, τ). Conditioned on the outer shares $\{\sigma_{i,\tau}\}$, the shares held by S are therefore independent across pairs (i, τ), and for each pair they consist of the $T_i(S)$ virtual shares of the (i, τ)-th inner sharing assigned to $S \cap \mathcal{P}_i$. The simulator produces them pair by pair.

If $(i, \tau) \in R(S)$, then Sim_S knows $\sigma_{i,\tau}$ and samples these shares by running the inner sharing of $\sigma_{i,\tau}$ honestly. If $(i, \tau) \notin R(S)$, then $T_i(S) < \Theta_{i,\tau}$ by Lemma 7, so by the perfect privacy of Shamir's scheme these $T_i(S)$ shares are distributed uniformly over $\mathbb{F}^{T_i(S)}$ independently of the value being shared; accordingly Sim_S samples them uniformly at random. The resulting distribution is identical to the view of S. □

Proof of Theorem 5. We are now ready to prove correctness and soundness of our construction. Let S be authorized in Γ. Lemma 6 states that it is authorized in Γ', so Lemma 9 gives $N(S) \geqslant t$ and Lemma 7 shows that the parties of S recover at least t outer shares. By the correctness of the outer Shamir sharing they reconstruct s.

Let S be unauthorized in Γ. Lemma 6 states that it is unauthorized in Γ', so Lemma 9 together with Lemma 7 gives $|R(S)| = N(S) \leqslant t - 1$. By the perfect privacy of the outer Shamir sharing, $\{\sigma_{i,\tau}\}_{(i,\tau) \in R(S)}$ is distributed independently of s, and by Lemma 10 the view of S is a randomized function of these shares alone; hence the view of S is independent of s. □

Finally, we remark on the concrete efficiency of our second construction.

Remark 3 (The constant hidden in $O(\log n)$)**.** *It is instructive to make the constant in Theorem 5 explicit, since it depends on the gap $\beta-\alpha$. A party $j \in \mathcal{P}_i$ holds $m \cdot w'_j$ field elements, and the two factors are controlled by the gap in two different ways.*

- *The scaling factor $K = \lceil 2/(\beta-\alpha) \rceil$ bounds the scaled weights, $w'_j \leqslant 2K$. It is set so that rounding each heavy party's weight down to a multiple of its class scale loses at most a $1/K$ fraction of its weight, and this loss must fit inside half of the gap (Lemma 6); hence $K \approx 2/(\beta-\alpha)$.*
- *The number of tokens per class is $m = \lceil \ell/(\beta'-\alpha') \rceil = \lceil 2\ell/(\beta-\alpha) \rceil$. Each of the ℓ classes may lose up to one unit Δ to the floor in Lemma 8, so $\ell\Delta$ must fit inside the remaining half of the gap, forcing $\Delta \approx (\beta-\alpha)KW/(2\ell)$ and thus $m = KW/\Delta \approx 2\ell/(\beta-\alpha)$.*

Multiplying the two bounds, the share size of every party is at most

$$m \cdot w'_j \;\leqslant\; \left\lceil \tfrac{2\ell}{\beta-\alpha} \right\rceil \cdot 2 \left\lceil \tfrac{2}{\beta-\alpha} \right\rceil \;\approx\; \frac{8\,\ell}{(\beta-\alpha)^2}, \qquad \ell = \lfloor \log_2 W \rfloor + 1.$$

The gap is therefore paid for quadratically, once through the scaling and once through the number of tokens. For the natural choice $\alpha = 1/3$ and $\beta = 2/3$ we have $K = 6$, $w'_j \leqslant 12$, and $m = 6\ell$, so every party holds at most $72\,\ell$ field elements.

If the input weights are not already $O(n)$ in total, one first applies Theorem 4. Doing so with $\varepsilon = (\beta-\alpha)/4$ yields $W \leqslant \lceil n/(\beta-\alpha) \rceil$ while replacing the gap $\beta-\alpha$ by $(\beta-\alpha)/2$; the bound above then becomes $\approx 32\,\ell/(\beta-\alpha)^2$ with $\ell \leqslant \log_2\big(n/(\beta-\alpha)\big) + 2$. The factor of 4 is the price of reserving half of the original gap for compression and half for the secret sharing scheme, and one may trade these off differently.

AI Statement

The results in this paper were developed through a series of interactions with AI (Claude Fable 5.1). In particular, the key ideas of using Hamilton apportionment ((Informal) Theorem 1) and two-level Shamir secret sharing ((Informal) Theorem 2) were both proposed by AI. AI also assisted with the preparation of the manuscript. The author takes full responsibility for the correctness of the results.

References

[ABD+26] Amit Agarwal, Kushal Babel, Sourav Das, Babak Poorebrahim Gilkalaye, Arup Mondal, Benny Pinkas, Peter Rindal, and Aayush Yadav. Weighted batched threshold encryption with applications to mempool privacy. In *2026 IEEE Symposium on Security and Privacy*, pages 2462–2481, San Francisco, CA, USA, May 18–20, 2026. IEEE Computer Society Press. doi:10.1109/SP63933.2026.00175. 3, 4, 5

[AIdK23] Muhammad Nadeem Ali, Muhammad Imran, Muhammad Salah ud din, and Byung-Seo Kim. Low rate ddos detection using weighted federated learning in sdn control plane in iot network. *Applied Sciences*, 13(3), 2023. URL: https://www.mdpi.com/2076-3417/13/3/1431, doi:10.3390/app13031431. 3

[BGH+23] Gabrielle Beck, Aarushi Goel, Aditya Hegde, Abhishek Jain, Zhengzhong Jin, and Gabriel Kaptchuk. Scalable multiparty garbling. In Weizhi Meng, Christian Damsgaard Jensen, Cas

Cremers, and Engin Kirda, editors, *ACM CCS 2023: 30th Conference on Computer and Communications Security*, pages 2158–2172, Copenhagen, Denmark, November 26–30, 2023. ACM Press. doi:10.1145/3576915.3623132. 4

[BGJK21] Gabrielle Beck, Aarushi Goel, Abhishek Jain, and Gabriel Kaptchuk. Order-C secure multiparty computation for highly repetitive circuits. In Anne Canteaut and François-Xavier Standaert, editors, *Advances in Cryptology – EUROCRYPT 2021, Part II*, volume 12697 of *Lecture Notes in Computer Science*, pages 663–693, Zagreb, Croatia, October 17–21, 2021. Springer, Cham, Switzerland. doi:10.1007/978-3-030-77886-6_23. 3, 5

[BHS23] Fabrice Benhamouda, Shai Halevi, and Lev Stambler. Weighted secret sharing from wiretap channels. In Kai-Min Chung, editor, *ITC 2023: 4th Conference on Information-Theoretic Cryptography*, volume 267 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 8:1–8:19, Aarhus, Denmark, June 6–8, 2023. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik. doi:10.4230/LIPIcs.ITC.2023.8. 3, 4, 5, 8

[BLS01] Dan Boneh, Ben Lynn, and Hovav Shacham. Short signatures from the Weil pairing. In Colin Boyd, editor, *Advances in Cryptology – ASIACRYPT 2001*, volume 2248 of *Lecture Notes in Computer Science*, pages 514–532, Gold Coast, Australia, December 9–13, 2001. Springer Berlin Heidelberg, Germany. doi:10.1007/3-540-45682-1_30. 5

[BVF+25] Samuel Breckenridge, Dani Vilardell, Andrés Fábrega, Amy Zhao, Patrick McCorry, Rafael Solari, and Ari Juels. B-privacy: Defining and enforcing privacy in weighted voting, 2025. URL: https://arxiv.org/abs/2509.17871, arXiv:2509.17871. 3

[BW06] Amos Beimel and Enav Weinreb. Monotone circuits for monotone weighted threshold functions. *Information Processing Letters*, 97(1):12–18, 2006. 3, 4

[BY01] Michel L. Balinski and H. Peyton Young. *Fair Representation: Meeting the Ideal of One Man, One Vote*. Brookings Institution Press, 2 edition, 2001. 9, 16

[CDM00] Ronald Cramer, Ivan Damgård, and Ueli M. Maurer. General secure multi-party computation from any linear secret-sharing scheme. In Bart Preneel, editor, *Advances in Cryptology – EUROCRYPT 2000*, volume 1807 of *Lecture Notes in Computer Science*, pages 316–334, Bruges, Belgium, May 14–18, 2000. Springer Berlin Heidelberg, Germany. doi:10.1007/3-540-45539-6_22. 5

[DCX+23] Sourav Das, Philippe Camacho, Zhuolun Xiang, Javier Nieto, Benedikt Bünz, and Ling Ren. Threshold signatures from inner product argument: Succinct, weighted, and multi-threshold. In Weizhi Meng, Christian Damsgaard Jensen, Cas Cremers, and Engin Kirda, editors, *ACM CCS 2023: 30th Conference on Computer and Communications Security*, pages 356–370, Copenhagen, Denmark, November 26–30, 2023. ACM Press. doi:10.1145/3576915.3623096. 3, 5, 6

[DIK+08] Ivan Damgård, Yuval Ishai, Mikkel Krøigaard, Jesper Buus Nielsen, and Adam Smith. Scalable multiparty computation with nearly optimal work and resilience. In David Wagner, editor, *Advances in Cryptology – CRYPTO 2008*, volume 5157 of *Lecture Notes in Computer Science*, pages 241–261, Santa Barbara, CA, USA, August 17–21, 2008. Springer Berlin Heidelberg, Germany. doi:10.1007/978-3-540-85174-5_14. 5

[DIK10] Ivan Damgård, Yuval Ishai, and Mikkel Krøigaard. Perfectly secure multiparty computation and the computational overhead of cryptography. In Henri Gilbert, editor, *Advances in Cryptology – EUROCRYPT 2010*, volume 6110 of *Lecture Notes in Computer Science*, pages 445–465, French Riviera, May 30 – June 3, 2010. Springer Berlin Heidelberg, Germany. doi:10.1007/978-3-642-13190-5_23. 5

[DPTX25] Sourav Das, Benny Pinkas, Alin Tomescu, and Zhuolun Xiang. Distributed randomness using weighted VUFs. In Serge Fehr and Pierre-Alain Fouque, editors, *Advances in Cryptology – EUROCRYPT 2025, Part VII*, volume 15607 of *Lecture Notes in Computer Science*, pages 314–344, Madrid, Spain, May 4–8, 2025. Springer, Cham, Switzerland. doi:10.1007/978-3-031-91098-2_12. 3, 4, 5

[EGP+23] Daniel Escudero, Vipul Goyal, Antigoni Polychroniadou, Yifan Song, and Chenkai Weng. SuperPack: Dishonest majority MPC with constant online communication. In Carmit Hazay and Martijn Stam, editors, *Advances in Cryptology – EUROCRYPT 2023, Part II*, volume 14005 of *Lecture Notes in Computer Science*, pages 220–250, Lyon, France, April 23–27, 2023. Springer, Cham, Switzerland. doi:10.1007/978-3-031-30617-4_8. 4

[EGPS22] Daniel Escudero, Vipul Goyal, Antigoni Polychroniadou, and Yifan Song. TurboPack: Honest majority MPC with constant online communication. In Heng Yin, Angelos Stavrou, Cas Cremers, and Elaine Shi, editors, *ACM CCS 2022: 29th Conference on Computer and Communications Security*, pages 951–964, Los Angeles, CA, USA, November 7–11, 2022. ACM Press. doi:10.1145/3548606.3560633. 4

[FG26] Oriol Farràs and Miquel Guiot. Revisiting beimel-weinreb weighted threshold secret sharing schemes. In *2026 IEEE 39th Computer Security Foundations Symposium (CSF)*, pages 313–328. IEEE, 2026. 3, 4

[FY92] Matthew K. Franklin and Moti Yung. Communication complexity of secure computation (extended abstract). In *24th Annual ACM Symposium on Theory of Computing*, pages 699–710, Victoria, BC, Canada, May 4–6, 1992. ACM Press. doi:10.1145/129712.129780. 3, 5

[GGW24] Sanjam Garg, Aarushi Goel, and Mingyuan Wang. How to prove statements obliviously? In Leonid Reyzin and Douglas Stebila, editors, *Advances in Cryptology – CRYPTO 2024, Part X*, volume 14929 of *Lecture Notes in Computer Science*, pages 449–487, Santa Barbara, CA, USA, August 18–22, 2024. Springer, Cham, Switzerland. doi:10.1007/978-3-031-68403-6_14. 3, 5, 6

[GJM+23] Sanjam Garg, Abhishek Jain, Pratyay Mukherjee, Rohit Sinha, Mingyuan Wang, and Yinuo Zhang. Cryptography with weights: MPC, encryption and signatures. In Helena Handschuh and Anna Lysyanskaya, editors, *Advances in Cryptology – CRYPTO 2023, Part I*, volume 14081 of *Lecture Notes in Computer Science*, pages 295–327, Santa Barbara, CA, USA, August 20–24, 2023. Springer, Cham, Switzerland. doi:10.1007/978-3-031-38557-5_10. 3, 4

[GJM+24] Sanjam Garg, Abhishek Jain, Pratyay Mukherjee, Rohit Sinha, Mingyuan Wang, and Yinuo Zhang. hinTS: Threshold signatures with silent setup. In *2024 IEEE Symposium on Security and Privacy*, pages 3034–3052, San Francisco, CA, USA, May 19–23, 2024. IEEE Computer Society Press. doi:10.1109/SP54263.2024.00057. 3, 5, 6

[GJMW24] Sanjam Garg, Abhishek Jain, Pratyay Mukherjee, and Mingyuan Wang. Scalable multiparty computation from non-linear secret sharing. In Leonid Reyzin and Douglas Stebila, editors, *Advances in Cryptology – CRYPTO 2024, Part VIII*, volume 14927 of *Lecture Notes in Computer Science*, pages 384–417, Santa Barbara, CA, USA, August 18–22, 2024. Springer, Cham, Switzerland. doi:10.1007/978-3-031-68397-8_12. 3, 5

[GKPW24] Sanjam Garg, Dimitris Kolonelos, Guru-Vamsi Policharla, and Mingyuan Wang. Threshold encryption with silent setup. In Leonid Reyzin and Douglas Stebila, editors, *Advances in Cryptology – CRYPTO 2024, Part VII*, volume 14926 of *Lecture Notes in Computer Science*, pages 352–386, Santa Barbara, CA, USA, August 18–22, 2024. Springer, Cham, Switzerland. doi:10.1007/978-3-031-68394-7_12. 4, 5, 6

[GO96] Oded Goldreich and Rafail Ostrovsky. Software protection and simulation on oblivious rams. *Journal of the ACM (JACM)*, 43(3):431–473, 1996. 11

[GPS21] Vipul Goyal, Antigoni Polychroniadou, and Yifan Song. Unconditional communication-efficient MPC via hall's marriage theorem. In Tal Malkin and Chris Peikert, editors, *Advances in Cryptology – CRYPTO 2021, Part II*, volume 12826 of *Lecture Notes in Computer Science*, pages 275–304, Virtual Event, August 16–20, 2021. Springer, Cham, Switzerland. doi:10.1007/978-3-030-84245-1_10. 3, 4

[GPS22] Vipul Goyal, Antigoni Polychroniadou, and Yifan Song. Sharing transformation and dishonest majority MPC with packed secret sharing. In Yevgeniy Dodis and Thomas Shrimpton, editors, *Advances in Cryptology – CRYPTO 2022, Part IV*, volume 13510 of *Lecture Notes in Computer Science*, pages 3–32, Santa Barbara, CA, USA, August 15–18, 2022. Springer, Cham, Switzerland. doi:10.1007/978-3-031-15985-5_1. 4

[GSW26] Aarushi Goel, Swagata Sasmal, and Mingyuan Wang. Weighted cryptography with weight-independent complexity. In *Annual International Cryptology Conference*, pages 198–231. Springer, 2026. 3, 4

[Hås94] Johan Håstad. On the size of weights for threshold gates. *SIAM J. Discret. Math.*, 7(3):484–492, 1994. doi:10.1137/S0895480192235878. 3, 19

[HPW26] Aditya Hegde, Phuoc Van Long Pham, and Mingyuan Wang. Multiparty computation with minimal overhead without circuit transformation. In *Annual International Cryptology Conference*, pages 134–165. Springer, 2026. 3, 5

[KNK+25] Naeem Khan, Shibli Nisar, Muhammad Asghar Khan, Yasar Abbas Ur Rehman, Fazal Noor, and Gordana Barb. Optimizing federated learning with aggregation strategies: A comprehensive survey. *IEEE Open Journal of the Computer Society*, 6:1227–1247, 2025. doi:10.1109/OJCS.2025.3590102. 3

[KRDO17] Aggelos Kiayias, Alexander Russell, Bernardo David, and Roman Oliynykov. Ouroboros: A provably secure proof-of-stake blockchain protocol. In Jonathan Katz and Hovav Shacham, editors, *Advances in Cryptology – CRYPTO 2017, Part I*, volume 10401 of *Lecture Notes in Computer Science*, pages 357–388, Santa Barbara, CA, USA, August 20–24, 2017. Springer, Cham, Switzerland. doi:10.1007/978-3-319-63688-7_12. 3

[MRV+21] Silvio Micali, Leonid Reyzin, Georgios Vlachos, Riad S. Wahby, and Nickolai Zeldovich. Compact certificates of collective knowledge. In *2021 IEEE Symposium on Security and Privacy*, pages 626–641, San Francisco, CA, USA, May 24–27, 2021. IEEE Computer Society Press. doi:10.1109/SP40001.2021.00096. 3

[Mur71] Saburo Muroga. *Threshold Logic and Its Applications*. Wiley-Interscience, New York, 1971. 3, 19

[Rab89] Michael O Rabin. Efficient dispersal of information for security, load balancing, and fault tolerance. *Journal of the ACM (JACM)*, 36(2):335–348, 1989. 3

[Sch90] Claus-Peter Schnorr. Efficient identification and signatures for smart cards. In Gilles Brassard, editor, *Advances in Cryptology – CRYPTO'89*, volume 435 of *Lecture Notes in Computer Science*, pages 239–252, Santa Barbara, CA, USA, August 20–24, 1990. Springer, New York, USA. doi:10.1007/0-387-34805-0_22. 5

[Sha79] Adi Shamir. How to share a secret. *Communications of the Association for Computing Machinery*, 22(11):612–613, November 1979. doi:10.1145/359168.359176. 3

[WW26] Brent Waters and David J. Wu. Silent threshold cryptography from pairings: Expressive policies in the plain model. In Joan Daemen and Emmanuel Thomé, editors, *Advances in Cryptology – EUROCRYPT 2026, Part V*, volume 16545 of *Lecture Notes in Computer Science*, pages 517–547, Rome, Italy, May 10–14, 2026. Springer, Cham, Switzerland. doi:10.1007/978-3-032-25330-9_18. 4, 5, 6, 7

[XLL+24] Runhua Xu, Bo Li, Chao Li, James B. D. Joshi, Shuai Ma, and Jianxin Li. Tapfed: Threshold secure aggregation for privacy-preserving federated learning. *IEEE Transactions on Dependable and Secure Computing*, 21(5):4309–4323, 2024. doi:10.1109/TDSC.2024.3350206. 3

[ZMGN20] Huafei Zhu, Rick Siow Mong Goh, and Wee-Keong Ng. Privacy-preserving weighted federated learning within the secret sharing framework. *IEEE Access*, 8:198275–198284, 2020. doi:10.1109/ACCESS.2020.3034602. 3