Faster ABE for Turing Machines from Circular Evasive LWE

Valerio Cini* and Hoeteck Wee

[1] Bocconi University, Milan, Italy
[2] NTT Research, Sunnyvale, CA, USA

Abstract. We present a new ABE for Turing machines, where
- encryption runs in time $O(T)$ and produces a ciphertext of size $O(T)$ for time T computation;
- key generation produces a key of size $O(1)$;
- security relies on the (public-coin) circular evasive LWE assumption.

In fact, we obtain an ABE for a simple and more general model of computation which we refer to as iterated (local) computation. We improve on the recent construction of Agrawal, Kumari, and Yamada (CRYPTO 24) in three ways: (i) faster encryption and smaller ciphertext size; (ii) smaller key size, and (iii) weaker assumptions, eliminating the need for circular tensor LWE.

1 Introduction

In key-policy attribute-based encryption (ABE) [SW05, GPSW06], ciphertexts ct are associated with an attribute x and a message μ and keys sk with a predicate f, and decryption returns μ when x satisfies f (i.e., $f(x) = 0$). We require security against unbounded collusions, so that an adversary that sees a ciphertext along with secret keys for an arbitrary number of predicates learns nothing about μ as long as x satisfies none of these predicates.

Much of the early works on ABE focus on the setting where the predicate f is represented by a circuit, for which remarkable progress has been made [GPSW06, GVW13, BGG+14, HLL23, Wee24]. However, the circuit model has two major theoretical and practical drawbacks:

- First, the input size is fixed ahead of time, which fails to capture many real-world applications involving computation on data of unbounded size, such as network monitoring and logging, pattern matching in gene sequences, and processing tax returns.
- Second, we incur worst-case running time on every input.

To overcome these limitations, a line of works [Wat12, GKP+13, AS17, AMY19a, GWW19, AMY19b, LL20, Wee21, HLL24, AKY24, DHM+24] studied ABE for uniform models of computation, such as deterministic finite automata (DFA) as well as the much more challenging model of Turing machines (TM), which is the uniform analogue of polynomial-sized circuits.

ABE for TM. The state of the art for ABE for Turing machines, without relying on heavy machinery such as indistinguishability obfuscation, is the recent work of Agrawal, Kumari and Yamada (AKY24) [AKY24], where

- encryption takes as additional input a bound 1^T on the running time of the TM computation on the attribute x, and runs in time $O(T^3)$ and produces a ciphertext of size $O(T^2)$;
- key generation takes as input a TM M, and runs in time $O(|M|^2)$ and produces a key of size $O(|M|^2)$;
- decryption takes time $O(T \cdot |M|)$;
- set-up does not impose a bound on either T or $|M|$.

The security of the scheme relies on the circular evasive LWE and tensor LWE assumptions [Wee22, Tsa22, HLL23], inherited from the state-of-the-art ABE for circuits. We briefly note that we do have constructions from weaker assumptions for either (i) log-space Turing machines, from either the k-Lin assumption in bilinear groups [LL20], or the evasive LWE assumption over lattices [HLL24], or (ii) bounded collusions, from the LWE assumption [AS17].

* Work done while at NTT Research.

reference	Time(Enc)	|ct|	Time(KeyGen)	|sk|
HLL24 [HLL24]	$T \cdot 2^S$	$T \cdot 2^S$	$\|M\|$	$\|M\|$
AKY24 [AKY24]	T^3	T^2	$\|M\|^2$	$\|M\|^2$
this work	T	T	$\|M\|$	1

Fig. 1. Lattice-based ABE schemes for Turing machines (TM) running in time T and space S. We ignore $\mathsf{poly}(\lambda)$ factors as well as the overhead from specifying the TM and the ciphertext attribute. Here, $|M|$ denotes size of M as a circuit, which is roughly linear in the number of states. All of these schemes achieve decryption time $O(T \cdot |M|)$.

1.1 Our Results

We present a new ABE for TM, where

- encryption runs in time $O(T)$ and produces a ciphertext of size $O(T)$;
- key generation produces a key of size $O(1)$;
- security relies on the (public-coin) circular evasive LWE assumption;
- as before, decryption takes time $O(T \cdot |M|)$, and set-up does not impose a bound on either T or $|M|$.

In fact, we obtain an ABE for a simple and more general model of computation which we refer to as iterated (local) computation. Our TM ABE essentially matches the state-of-the-art for the non-uniform setting[3] [HLL23, BGG+14], which also achieves linear-time encryption and ciphertext size, as well as $O(1)$-sized keys.

Comparison with AKY24. We improve on AKY24 in three ways: (i) faster encryption and smaller ciphertext size; (ii) smaller key size, and (iii) weaker assumptions, eliminating the need for circular tensor LWE (see Fig 1). Our construction departs significantly from that of AKY24, which in turn follows a blueprint from [AMVY21, AMY19a]. In particular, we do not rely on the use of CP-ABE, thereby avoiding the large key sizes and the use of circular tensor LWE in AKY24. We proceed to provide a brief overview of our construction, deferring a more technical overview to Section 2. An overview of AKY24 can be found in Appendix A.

Our approach. At a high level, we show how to adapt the BGGHSNVV14 ABE for circuits [BGG+14] –and its generalization to unbounded-depth circuits in HLL23 [HLL23]– to the setting of TM computation. Recall BGGHSNVV14 encodings of $\mathbf{z} = (z_1, \ldots, z_\ell) \in \{0,1\}^\ell$, which are LWE samples of the form:

$$\mathbf{s}(\mathbf{A} - \mathbf{z} \otimes \mathbf{G}) = (\mathbf{s}(\mathbf{A}_1 - z_1\mathbf{G}), \ldots, \mathbf{s}(\mathbf{A}_\ell - z_\ell\mathbf{G})),\ \mathbf{s} \in \mathbb{Z}_q^n, \mathbf{A} \in \mathbf{Z}_q^{n \times \ell m}$$

Moreover, we can homomorphically evaluate any circuit on these encodings. We introduce dual encodings, which are LWE samples of the form:

$$(\mathbf{s}_1(\mathbf{A}_0 - z_1\mathbf{G}), \ldots, \mathbf{s}_\ell(\mathbf{A}_0 - z_\ell\mathbf{G})),\ \mathbf{s}_1, \ldots, \mathbf{s}_\ell \in \mathbb{Z}_q^n, \mathbf{A}_0 \in \mathbf{Z}_q^{n \times m}$$

The key advantage of dual encodings is that the size of the matrix $\mathbf{A}_0$ is independent of $|\mathbf{z}|$ (so that ABE set-up is independent of both T and $|M|$); the downside is that they only support very simple homomorphic computation, namely projections. Our construction uses both types of encodings:

(i) BGGHSNVV14 encodings of short strings of length $1 + \log |M| \leq 1 + \lambda$, for which we need to perform complex computation, and
(ii) dual encodings of long strings of length $O(T)$ for simple computation (i.e., projections), as well as
(iii) new techniques to switch between both types of encodings of the same string (i.e., no computation).

[3] A Turing machine computation running in time T can be simulated by circuits of size and depth $\tilde{O}(T)$.

At a high level, in a TM computation of time T and space S, we have a long work tape of length $S \approx T$, and we iteratively read 1 bit and write 1 bit from/to the work tape, while updating the state of the TM, over a sequence of T time steps. (WLOG, we work with "oblivious" TM, where the read and write locations only depend on the time step. We also assume the TM input is written on the work tape.) Moreover, the computation in each step is "local", which we can think of as applying a function f (i.e., the TM transition function) of input and output length $1 + \log |M|$, comprising 1 bit from the work tape, and $\log |M|$ bits for the TM state. In a bit more detail,

- we use $\mathbf{z}_t \in \{0,1\}^{S+\log |M|}$ to denote the concatenation of the TM state and the work tape at time t, for $t = 0, 1, \ldots, T$;
- we use $\mathbf{z}^R_{t-1} \in \{0,1\}^{\log |M|+1}$ to denote the read bit at time t and the TM state at time $t-1$, which we can derive from $\mathbf{z}_{t-1}$;
- we use $\mathbf{z}^W_t$ to denote the output of the TM transition function applied to $\mathbf{z}^R_{t-1}$, which specifies the write bit at time t as well as the TM state at time t.

Next, we describe how decryption works in our scheme. We start with a dual encoding of $\mathbf{z}_0$ which we derive from the ciphertext and we iteratively compute a dual encoding of $\mathbf{z}_t$. That is, at time t:

1. we derive a BGGHSNVV14 encoding of $\mathbf{z}^R_{t-1} \in \{0,1\}^{\log |M|+1}$ starting from a dual encoding of $\mathbf{z}^R_{t-1}$, which we can derive from the dual encoding of $\mathbf{z}_{t-1}$ via a simple projection;
2. we homomorphically evaluate the TM transition function on the encoding of $\mathbf{z}^R_{t-1}$ to derive an encoding of $\mathbf{z}^W_t$;
3. we derive a dual encoding of $\mathbf{z}_t$ starting from the dual encoding of $\mathbf{z}_{t-1}$ and the encoding of $\mathbf{z}^W_t$.

We use the same $\mathbf{A}_0 \in \mathbb{Z}_q^{n\times m}$ for the dual encoding and the same $\mathbf{A} \in \mathbb{Z}_q^{n\times(\log |M|+1)m}$ for the BGGHSNVV14 encodings[4] for all t, but a fresh $(\mathbf{A}_0, \mathbf{A})$ per TM. To carry out steps 1 and 3, we provide techniques to switch between encodings and dual encodings of the same string, as well as dual encodings of the same string with respect to different LWE secrets. Switching between encodings blows up the error by a polynomial factor, which we can accommodate using the HLL23 noise reduction techniques.

So far, we have not specified how we choose the LWE secrets for the dual encodings of $\mathbf{z}_0, \ldots, \mathbf{z}_T$. Sampling them uniformly at random would yield an ABE scheme with encryption time and ciphertext size $O(ST)$, that admits a relatively straightforward proof of security based on circular evasive LWE. On the other hand, reusing the same LWE secrets across $\mathbf{z}_0, \ldots, \mathbf{z}_T$ yields encryption time and ciphertext size $O(T)$, but a scheme that is broken. The most technically involved part of this work lies in carefully correlating the distributions of LWE secrets across the dual encodings so as to achieve $O(T)$ encryption time while preserving security; in addition, we require a fairly intricate proof of security to carry out reductions with correlated secrets.

2 Technical Overview

2.1 Iterated Computation

We consider iterated computation with locality L defined as follows:

- The function is specified by $f : \{0,1\}^L \to \{0,1\}^L$;
- The input is $\mathbf{x} \in \{0,1\}^\ell$, a time bound T, a space bound S, and a pair of functions $\mathsf{read}, \mathsf{write} : [T] \to \binom{S}{L}$ where $\binom{S}{L}$ denotes all subsets of $[S]$ of size L.

The computation is defined iteratively over T steps as follows:

- Think of $\mathbf{z}_t \in \{0,1\}^S, t = 0, 1, \ldots, T$ as the "work tape" at the t'th step, where we initialize $\mathbf{z}_0 := \mathbf{x}\|0^{S-\ell}$.
- Given $\mathbf{z} \in \{0,1\}^S, \Gamma \subseteq [S]$, we write $\mathbf{z}[\Gamma] \in \{0,1\}^{|\Gamma|}$ to denote the substring of $\mathbf{z}$ indexed by Γ. For convenience, we will preserve indices when projecting on substrings, so that for any $\Gamma, \Lambda \subset [S]$, one has $\mathbf{z}[\Gamma][\Lambda] = \mathbf{z}[\Lambda][\Gamma] = \mathbf{z}[\Gamma \cap \Lambda]$.

[4] In the body of the paper, we call this matrix $\mathbf{D}$

– For $t \in [T]$, we update $\mathbf{z}_{t-1}$ to $\mathbf{z}_t$ as follows:

$$\mathbf{z}_t[\mathsf{write}(t)] := f(\mathbf{z}_{t-1}[\mathsf{read}(t)])$$
$$\mathbf{z}_t[\overline{\mathsf{write}(t)}] := \mathbf{z}_{t-1}[\overline{\mathsf{write}(t)}]$$

That is, in the t'th step, we read L entries of $\mathbf{z}_{t-1}$ in positions $\mathsf{read}(t)$, apply the function f, and update the L entries in positions $\mathsf{write}(t)$. The remaining entries outside $\mathsf{write}(t)$ remain unchanged.

Finally, we accept if $\mathbf{z}_T[1] = 0$. Also, we write $\mathsf{IC}(f, \mathbf{x}, 1^T) := \mathbf{z}_T[1]$.

ABE for iterated computation. Our main result is an ABE for iterated computation where

$$|\mathsf{mpk}| = O(L), \quad |\mathsf{ct}| = O(S + T \cdot L), \quad |\mathsf{sk}| = O(L),$$
$$\mathsf{Time}(\mathsf{Enc}) = O(S + T \cdot L), \quad \mathsf{Time}(\mathsf{KeyGen}) = O(L + \mathsf{size}(f)),$$
$$\mathsf{Time}(\mathsf{Dec}) = O(S + T \cdot \mathsf{size}(f)).$$

with $O(\cdot)$ that hides factors polynomial in λ (and thus $\log T, \log S$).

Turing machines. It is fairly easy to see that iterated computation captures (oblivious) Turing machines (OTM), where IC work tape = (OTM input tape, OTM work tape, OTM state). This means that locality L captures (oblivious) Turing machines with 2^{L-1} states and alphabet $\{0, 1\}$. In particular, at time $t \in [T]$:

– $\mathbf{z}_{t-1}$ would specify the input tape, work tape, and state u_{t-1} of the Turing machine (for oblivious Turing machines, the position of the pointer is a fixed function of t);
– $\mathbf{z}_{t-1}[\mathsf{read}(t)]$ returns (u_{t-1}, σ_t) where σ_t is the symbol being read, and $f(u_{t-1}, \sigma_t) = (u_t, \sigma'_t)$ where u_t is the new state, and σ'_t is the symbol that will be written. We then update $\mathbf{z}_t$ as follows: the entries in positions $\mathsf{write}(t)$ are (u_t, σ'_t), and the remaining entries are the same as in $\mathbf{z}_{t-1}$.

That is, TM can be captured by iterated computation where $L \leq \lambda$, and also, we have $S \leq T$. (We can convert any TM to an oblivious one with a $O(\log T)$ overhead in running time [PF79].)

ABE for Turing machines. Putting everything together, we obtain an ABE for Turing machines where:

$$|\mathsf{mpk}| = O(1), \quad |\mathsf{ct}| = O(T), \quad |\mathsf{sk}| = O(1).$$
$$\mathsf{Time}(\mathsf{Enc}) = O(T), \quad \mathsf{Time}(\mathsf{KeyGen}) = O(|M|),$$
$$\mathsf{Time}(\mathsf{Dec}) = O(T \cdot |M|).$$

with $O(\cdot)$ that hides factors polynomial in λ (and thus $\log T, \log S$, and L).

2.2 ABE Warm-up

As a warm-up, we present an ABE for iterated computation where

$$|\mathsf{mpk}| = O(L), \quad |\mathsf{ct}| = O(\boxed{T \cdot S}), \quad |\mathsf{sk}| = O(1).$$

Moreover,

$$\mathsf{Time}(\mathsf{Enc}) = O(\boxed{T \cdot S}), \quad \mathsf{Time}(\mathsf{KeyGen}) = O(L + \mathsf{size}(f)), \quad \mathsf{Time}(\mathsf{Dec}) = O(\boxed{T(S + \mathsf{size}(f))}).$$

where the boxed terms indicate the differences from our main result.

Attribute encodings. We begin with a recap of BGGHNSVV attribute encoding, which for an attribute $\mathbf{x} = (x_1, \ldots, x_\ell) \in \{0,1\}^\ell$ is given by:

$$\mathbf{s}_0(\mathbf{A}_1 - x_1\mathbf{G}), \ldots, \mathbf{s}_0(\mathbf{A}_\ell - x_\ell\mathbf{G}),\ \mathbf{s}_0 \leftarrow \mathbb{Z}_q^n, \mathbf{A}_i \leftarrow \mathbb{Z}_q^{n\times m}$$

Here, we introduce the "dual" encoding, given by

$$\mathbf{s}_1(\mathbf{A}_0 - x_1\mathbf{G}), \ldots, \mathbf{s}_\ell(\mathbf{A}_0 - x_\ell\mathbf{G}),\ \mathbf{s}_i \leftarrow \mathbb{Z}_q^n, \mathbf{A}_0 \leftarrow \mathbb{Z}_q^{n\times m}$$

In particular, the size of the "public matrix" $\mathbf{A}_0$ is independent of ℓ, which is the first step towards achieving small public parameters and secret key sizes in our ABE scheme.

We write the two encodings more compactly as

$$[\![\mathbf{s}_0, \mathbf{A}, \mathbf{x}]\!] \coloneqq (\mathbf{I}_\ell \otimes \mathbf{s}_0)\mathbf{A} - \mathbf{x}^\top \cdot \mathbf{s}_0\mathbf{G} \in \mathbb{Z}_q^{\ell\times m}$$
$$[\![\mathbf{S}, \mathbf{A}_0, \mathbf{x}]\!] \coloneqq \mathbf{S}\mathbf{A}_0 - \mathsf{diag}(\mathbf{x}) \cdot \mathbf{S}\mathbf{G} \in \mathbb{Z}_q^{\ell\times m}$$

where $\mathbf{A} \in \mathbb{Z}_q^{\ell n\times m}, \mathbf{S} \in \mathbb{Z}_q^{\ell\times n}$. We use $[\![\cdots]\!]$ for both encodings to reduce notational clutter; the matrix dimensions determine which encoding is used.

Recoding attribute encodings. In our construction, we want to be able to switch the LWE secret and matrix in a dual attribute encoding, namely

$$\mathbf{x}, [\![\mathbf{S}, \mathbf{A}_0, \mathbf{x}]\!] \mapsto [\![\mathbf{S}', \mathbf{A}_0', \mathbf{x}]\!]$$

and where $\mathbf{x}$ is not known in advance. We show how to do the switch given

$$(\mathbf{C}, \mathbf{C}') := (\mathbf{S}\mathbf{B}, \mathbf{S}\mathbf{W} + \mathbf{S}'\mathbf{G})$$
$$\mathbf{K} \leftarrow \mathbf{B}^{-1}\big(\mathbf{A}_0\mathbf{G}^{-1}(\mathbf{W}) - \mathbf{W}\mathbf{G}^{-1}(\mathbf{A}_0')\big)$$

where $\mathbf{B}, \mathbf{W} \leftarrow \mathbb{Z}_q^{n\times m}$. Looking ahead, $(\mathbf{C}, \mathbf{C}')$ goes into our ciphertext, and $\mathbf{K}$ goes into the secret key. As a warm-up, observe that for $\ell = 1$, we have:

$$\mathbf{s}'(\mathbf{A}_0' - x\mathbf{G}) = \overbrace{\mathbf{s}\mathbf{B}}^{\mathbf{C}} \cdot \overbrace{\mathbf{B}^{-1}(\mathbf{A}_0\mathbf{G}^{-1}(\mathbf{W}) - \mathbf{W}\mathbf{G}^{-1}(\mathbf{A}_0'))}^{\mathbf{K}}$$
$$+(\overbrace{\mathbf{s}\mathbf{W} + \mathbf{s}'\mathbf{G}}^{\mathbf{C}'})(\mathbf{G}^{-1}(\mathbf{A}_0') - x\mathbf{I}) - \mathbf{s}(\mathbf{A}_0 - x\mathbf{G})\mathbf{G}^{-1}(\mathbf{W})$$

More generally,

$$[\![\mathbf{S}', \mathbf{A}_0', \mathbf{x}]\!] = \mathbf{C} \cdot \mathbf{K} + \mathbf{C}' \cdot \mathbf{G}^{-1}(\mathbf{A}_0') - \mathsf{diag}(\mathbf{x}) \cdot \mathbf{C}' - [\![\mathbf{S}, \mathbf{A}_0, \mathbf{x}]\!] \cdot \mathbf{G}^{-1}(\mathbf{W})$$

where $\mathsf{diag}(\mathbf{x}) \in \mathbb{Z}^{|\mathbf{x}|\times|\mathbf{x}|}$ denotes the matrix with entries of $\mathbf{x}$ on the diagonal, and zeros everywhere else. As before, we introduce the following compact notation:

$$[\![\mathbf{S} \mapsto \mathbf{S}']\!]_{\mathbf{B},\mathbf{W}} := (\mathbf{S}\mathbf{B}, \mathbf{S}\mathbf{W} + \mathbf{S}'\mathbf{G})$$
$$[\![\mathbf{A}_0 \mapsto \mathbf{A}_0']\!]_{\mathbf{W}} := \mathbf{A}_0\mathbf{G}^{-1}(\mathbf{W}) - \mathbf{W}\mathbf{G}^{-1}(\mathbf{A}_0')$$

We can write down analogous recodings from an attribute encoding to a dual attribute encoding and vice versa. Details are presented in Section 4 and Appendix C.

Decryption invariant. We begin with the decryption invariant in our scheme. Fix an input $(\mathbf{x}, 1^T)$ and a function f. At step t, for $t = 0, \ldots, T$, the decryptor computes a dual attribute encoding of $\mathbf{z}_t \in \{0,1\}^S$ given by

$$\tilde{\mathbf{C}}_t := [\![\mathbf{S}_t, \mathbf{A}_0, \mathbf{z}_t]\!] = \mathbf{S}_t\mathbf{A}_0 - \mathsf{diag}(\mathbf{z}_t) \cdot \mathbf{S}_t\mathbf{G} \tag{1}$$

where $\mathbf{S}_0, \mathbf{S}_1, \ldots, \mathbf{S}_T \in \mathbb{Z}_q^{S\times n}$ is associated with the ciphertext and $\mathbf{A}_0 \leftarrow \mathbb{Z}_q^{n\times m}$ with the secret key (that is, sampled during encryption and key generation, respectively).

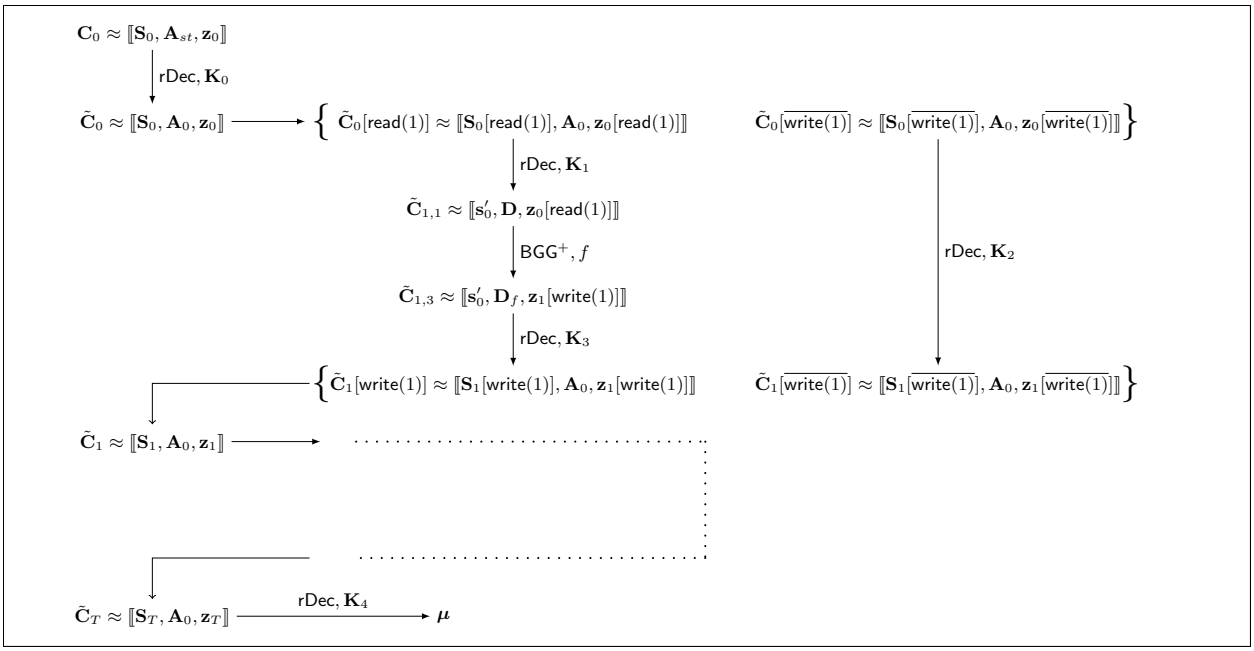

Fig. 2. Summary of how decryption works in our warm-up construction.

How to update the encodings. We want to derive $[\![\mathbf{S}_t, \mathbf{A}_0, \mathbf{z}_t]\!]$ from $[\![\mathbf{S}_{t-1}, \mathbf{A}_0, \mathbf{z}_{t-1}]\!]$. Recall

$$\mathbf{z}_t[\mathsf{write}(t)] := f(\mathbf{z}_{t-1}[\mathsf{read}(t)])$$
$$\mathbf{z}_t[\overline{\mathsf{write}(t)}] := \mathbf{z}_{t-1}[\overline{\mathsf{write}(t)}]$$

Henceforth, we also use $\mathbf{S}[\Gamma]$, for $\Gamma \subseteq [S]$ to denote the submatrix of $\mathbf{S}$ whose rows are indexed by Γ.

encoding of $\mathbf{z}_t[\overline{\mathsf{write}(t)}]$*:* We simply compute the recoding

$$\begin{aligned}&[\![\mathbf{S}_{t-1}[\overline{\mathsf{write}(t)}], \mathbf{A}_0, \mathbf{z}_{t-1}[\overline{\mathsf{write}(t)}]]\!]\\ \mapsto\ &[\![\mathbf{S}_t[\overline{\mathsf{write}(t)}], \mathbf{A}_0, \mathbf{z}_{t-1}[\overline{\mathsf{write}(t)}]]\!] = [\![\mathbf{S}_t[\overline{\mathsf{write}(t)}], \mathbf{A}_0, \mathbf{z}_t[\overline{\mathsf{write}(t)}]]\!]\end{aligned}$$

using $[\![\mathbf{S}_{t-1}[\overline{\mathsf{write}(t)}] \mapsto \mathbf{S}_t[\overline{\mathsf{write}(t)}])]\!]_{\mathbf{B}_2, \mathbf{W}_2}$ and ${\mathbf{B}_2}^{-1}([\![\mathbf{A}_0 \mapsto \mathbf{A}_0]\!]_{\mathbf{W}_2})$. In other words, we "update" the secret from $\mathbf{S}_{t-1}[\overline{\mathsf{write}(t)}]$ to $\mathbf{S}_t[\overline{\mathsf{write}(t)}]$, while keeping the attribute $\mathbf{z}_{t-1}[\overline{\mathsf{write}(t)}] = \mathbf{z}_t[\overline{\mathsf{write}(t)}]$ unchanged.

encoding of $\mathbf{z}_t[\mathsf{write}(t)]$*:* The idea is to first recode $[\![\mathbf{S}_{t-1}[\mathsf{read}(t)], \mathbf{A}_0, \mathbf{z}_{t-1}[\mathsf{read}(t)]]\!]$ to an intermediate BGG$^+$ attribute encoding

$$[\![\mathbf{s}'_{t-1}, \mathbf{D}, \mathbf{z}_{t-1}[\mathsf{read}(t)]]\!],$$

where $\mathbf{s}'_{t-1} \in \mathbb{Z}_q^n$ is associated with the ciphertext and $\mathbf{D} \leftarrow \mathbb{Z}_q^{Ln \times m}$ with the secret key. Next, we homomorphically evaluate f to derive

$$[\![\mathbf{s}'_{t-1}, \mathbf{D}_f, f(\mathbf{z}_{t-1}[\mathsf{read}(t)])]\!] = [\![\mathbf{s}'_{t-1}, \mathbf{D}_f, \mathbf{z}_t[\mathsf{write}(t)]]\!]$$

Finally, we recode to $[\![\mathbf{S}_t[\mathsf{write}(t)], \mathbf{A}_0, \mathbf{z}_t[\mathsf{write}(t)]]\!]$.

Our scheme so far. The scheme so far is as follows, omitting error terms in the ciphertext:

$\underline{\mathsf{mpk}}$:

$$\begin{aligned}\mathbf{A}_{st}, \mathbf{B}_0, \mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3, \mathbf{B}_4 &\leftarrow \mathbb{Z}_q^{n\times m},\\ \mathbf{W}_1, \mathbf{W}_2 &\leftarrow \mathbb{Z}_q^{n\times m},\\ \mathbf{W}_3 &\leftarrow \mathbb{Z}_q^{Ln\times m},\\ \mathbf{P} &\leftarrow \mathbb{Z}_q^{n\times m}\end{aligned}$$

$\underline{\mathsf{sk}}$:

$$\begin{aligned}\mathbf{A}_0 &\leftarrow \mathbb{Z}_q^{n\times m},\\ \mathbf{D} &\leftarrow \mathbb{Z}_q^{Ln\times m},\\ \mathbf{R} &\leftarrow \mathcal{D}_{\mathbb{Z}^{m\times m}},\\ \mathbf{K}_0 &\leftarrow \mathbf{B}_0^{-1}(\mathbf{A}_0 - \mathbf{A}_{st}),\\ \mathbf{K}_1 &\leftarrow (\mathbf{I}_L \otimes \mathbf{B}_1)^{-1}([\![\mathbf{A}_0 \mapsto \mathbf{D}]\!]_{\mathbf{W}_1}),\\ \mathbf{K}_2 &\leftarrow \mathbf{B}_2^{-1}([\![\mathbf{A}_0 \mapsto \mathbf{A}_0]\!]_{\mathbf{W}_2}),\\ \mathbf{K}_3 &\leftarrow (\mathbf{I}_L \otimes \mathbf{B}_3)^{-1}([\![\mathbf{D}_f \mapsto \mathbf{A}_0]\!]_{\mathbf{W}_3}),\\ \mathbf{K}_4 &\leftarrow \mathbf{B}_4^{-1}(\mathbf{P} + \mathbf{A}_0\mathbf{R})\end{aligned}$$

$\underline{\mathsf{ct}}$:

$$\begin{aligned}\mathbf{S}_0, \mathbf{S}_1, \ldots, \mathbf{S}_T &\leftarrow \mathbb{Z}_q^{S\times n},\\ \mathbf{s}'_0, \mathbf{s}'_1, \ldots, \mathbf{s}'_{T-1} &\leftarrow \mathbb{Z}_q^{n},\\ \mathbf{C}'_0 &\coloneqq [\![\mathbf{S}_0, \mathbf{A}_{st}, \mathbf{z}_0]\!],\\ \mathbf{C}_0 &\coloneqq \mathbf{S}_0\mathbf{B}_0,\\ (\mathbf{C}_{t,1}, \mathbf{C}'_{t,1}) &\coloneqq [\![\mathbf{S}_{t-1}[\mathsf{read}(t)] \mapsto \mathbf{s}'_{t-1}]\!]_{\mathbf{B}_1,\mathbf{W}_1},\\ (\mathbf{C}_{t,2}, \mathbf{C}'_{t,2}) &\coloneqq [\![\mathbf{S}_{t-1}[\overline{\mathsf{write}(t)}] \mapsto \mathbf{S}_t[\overline{\mathsf{write}(t)}]]\!]_{\mathbf{B}_2,\mathbf{W}_2},\\ (\mathbf{C}_{t,3}, \mathbf{C}'_{t,3}) &\coloneqq [\![\mathbf{s}'_{t-1} \mapsto \mathbf{S}_t[\mathsf{write}(t)]]\!]_{\mathbf{B}_3,\mathbf{W}_3}, \qquad t \in [T],\\ \mathbf{C}'_{T,4} &\coloneqq \mathbf{S}_T[1]\mathbf{P} + \boldsymbol{\mu} \cdot \lfloor q/2 \rfloor,\\ \mathbf{C}_{T,4} &\coloneqq \mathbf{S}_T[1]\mathbf{B}_4\end{aligned}$$

where the expression for the $[\![\ldots]\!], [\![\ldots]\!]_{\mathbf{B}_\gamma,\mathbf{W}_\gamma}$ in the ciphertext are given by

$$\begin{aligned}&\mathbf{S}_0\mathbf{A}_{st} - \mathsf{diag}(\mathbf{z}_0)\mathbf{S}\mathbf{A}_{st}\\ &(\mathbf{S}_{t-1}[\mathsf{read}(t)]\mathbf{B}_1, \mathbf{S}_{t-1}[\mathsf{read}(t)]\mathbf{W}_1 + \mathbf{1}_L^\top \cdot \mathbf{s}'_{t-1}\mathbf{G})\\ &(\mathbf{S}_{t-1}[\overline{\mathsf{write}(t)}]\mathbf{B}_2, \mathbf{S}_{t-1}[\overline{\mathsf{write}(t)}]\mathbf{W}_2 + \mathbf{S}_t[\overline{\mathsf{write}(t)}]\mathbf{G})\\ &(\mathbf{s}'_{t-1}\mathbf{B}_3, (\mathbf{I}_L \otimes \mathbf{s}'_{t-1})\mathbf{W}_3 + \mathbf{S}_t[\mathsf{write}(t)]\mathbf{G})\end{aligned}$$

respectively.

Decryption. Decryption iteratively computes the decryption invariant $[\![\mathbf{S}_t, \mathbf{A}_0, \mathbf{z}_t]\!]$ in (1):

- We compute $\tilde{\mathbf{C}}_0 = [\![\mathbf{S}_0, \mathbf{A}_0, \mathbf{z}_0]\!]$ using $\mathbf{C}'_0, \mathbf{C}_0, \mathbf{K}_0$.
- For $t = 1, \ldots, T$, we compute $\tilde{\mathbf{C}}_t$ using $\tilde{\mathbf{C}}_{t-1}$ along with $\mathbf{C}_{t,1}, \mathbf{C}'_{t,1}, \mathbf{K}_1, \mathbf{C}_{t,2}, \mathbf{C}'_{t,2}, \mathbf{K}_2, \mathbf{C}_{t,3}, \mathbf{C}'_{t,3}, \mathbf{K}_3$ as described earlier. At the end of this step, we recover $\mathbf{s}_T[1](\mathbf{A}_0 - 0 \cdot \mathbf{G})$ since $\mathbf{z}_T[1] = 0$.
- Compute $\mathbf{s}_T[1](\mathbf{P} + \mathbf{A}_0\mathbf{R})$ using $\mathbf{C}_{T,4}, \mathbf{K}_4$. Combined with $\mathbf{C}'_{T,4}, \mathbf{s}_T[1]\mathbf{A}_0$, and $\mathbf{R}$, we recover $\boldsymbol{\mu}$.

A summary is given in Fig. 2.

Noise growth. As written, the noise blows up multiplicatively for each step of the computation, which means q grows exponentially with T. To manage this, we use HLL23-style noise reduction techniques. In order to rely on such results, we will have to: (i) use short gaussian secrets instead of uniform random ones, (ii) add circular encodings of such secrets to the ciphertext, (iii) rely on circular (short-secret) evasive LWE instead of evasive LWE in the security proof. We ignore these modifications in the rest of the overview.

Proof overview. Informally, the evasive LWE assumption [Wee22, Tsa19] says that given a LWE sample $(\mathbf{B}, \mathbf{sB}+\mathbf{e})$, getting a gaussian pre-image $\mathbf{B}^{-1}(\mathbf{P})$ is no more useful than getting the product $(\mathbf{sB}+\mathbf{e}) \cdot \mathbf{B}^{-1}(\mathbf{P}) \approx \mathbf{sP}+\mathbf{e}'$. Here, we work with a (public-coin[5]) generalization of the assumption (similar to that in [HLL24]) where we start with LWE samples $(\mathbf{B}_\gamma, \mathbf{S}_\gamma\mathbf{B}_\gamma + \mathbf{E}_\gamma), \gamma = 0,1,2,3,4$, and we assert that getting gaussian pre-images $\mathbf{B}_\gamma^{-1}(\mathbf{P}_\gamma)$ is no more useful than getting the products $(\mathbf{S}_\gamma\mathbf{B}_\gamma + \mathbf{E}_\gamma) \cdot \mathbf{B}_\gamma^{-1}(\mathbf{P}_\gamma) \approx \mathbf{S}_\gamma\mathbf{P}_\gamma + \mathbf{E}'_\gamma$. Here, the marginal distribution of each matrix $\mathbf{S}_\gamma$ is uniform, but there may be overlapping rows amongst the matrices.

Let $f^{(1)}, \ldots, f^{(Q)}$ denote the Q key queries made by the adversary, and we use $(\mathbf{K}_0^{(j)}, \ldots, \mathbf{K}_4^{(j)}, \mathbf{A}_0^{(j)}, \mathbf{D}^{(j)}, \mathbf{R}^{(j)})$ to denote the j'th key. In the proof, we want to show that given $\mathsf{mpk}, \mathsf{ct}$ together with the following quantities

$$
\begin{array}{lll}
 & \mathbf{A}_0^{(j)}, \mathbf{D}^{(j)}, \mathbf{R}^{(j)} & \\
\tilde{\mathbf{C}}_0^{(j)} & = [\![\mathbf{S}_0, \mathbf{A}_0^{(j)}, \mathbf{z}_0^{(j)}]\!] & \text{instead of } \mathbf{C}_0 \cdot \mathbf{K}_0^{(j)} \\
\tilde{\mathbf{C}}_{t,1}^{(j)} & = [\![\mathbf{s}'_{t-1}, \mathbf{D}^{(j)}, \mathbf{z}_{t-1}^{(j)}[\mathsf{read}(t)]]\!] & \mathbf{C}_{t,1}, \mathbf{K}_1^{(j)} \\
\tilde{\mathbf{C}}_t^{(j)} & = [\![\mathbf{S}_t, \mathbf{A}_0^{(j)}, \mathbf{z}_t^{(j)}]\!] & \mathbf{C}_{t,2}, \mathbf{K}_2^{(j)}, \mathbf{C}_{t,3}, \mathbf{K}_3^{(j)} \\
\tilde{\mathbf{C}}_{T,4}^{(j)} & = \mathbf{z}_T^{(j)}[1]\mathbf{S}_T[1]\mathbf{GR}^{(j)} & \mathbf{C}_{T,4}, \mathbf{K}_4^{(j)}
\end{array}
$$

are pseudorandom.[6] We have the following hybrid sequence:

- $\mathsf{H}_{\mathsf{pre}}$: pre-condition of the evasive LWE assumption
- H_0: replace $\mathbf{C}'_0, \mathbf{C}_0, \tilde{\mathbf{C}}_0^{(j)}, \mathbf{C}_{1,1}, \mathbf{C}'_{1,1}, \mathbf{C}_{1,2}, \mathbf{C}'_{1,2}$, with random
- H'_0: replace $\tilde{\mathbf{C}}_{1,1}^{(j)}, \mathbf{C}_{1,3}, \mathbf{C}'_{1,3}$ with random
- H_1: replace $\tilde{\mathbf{C}}_1^{(j)}, \mathbf{C}_{2,1}, \mathbf{C}'_{2,1}, \mathbf{C}_{2,2}, \mathbf{C}'_{2,2}$, with random
- ...

More generally, in H_t, we want to replace

$$
\begin{aligned}
&\tilde{\mathbf{C}}_t^{(j)}, \mathbf{C}_{t+1,1}, \mathbf{C}'_{t+1,1}, \mathbf{C}_{t+1,2}, \mathbf{C}'_{t+1,2} \\
&= [\![\mathbf{S}_t, \mathbf{A}_0^{(j)}, \mathbf{z}_t^{(j)}]\!], [\![\mathbf{S}_t[\mathsf{read}(t+1)] \mapsto \mathbf{s}'_t]\!]_{\mathbf{B}_1, \mathbf{W}_1}, [\![\mathbf{S}_t[\overline{\mathsf{write}(t+1)}] \mapsto \mathbf{S}_{t+1}[\overline{\mathsf{write}(t+1)}]]\!]_{\mathbf{B}_2, \mathbf{W}_2}
\end{aligned}
$$

with random. Here, we will rely on LWE with $\mathbf{S}_t$ as the LWE secret.

Similarly, in H'_t, we want to replace

$$
\begin{aligned}
&\tilde{\mathbf{C}}_{t,1}^{(j)}, \mathbf{C}_{t,3}, \mathbf{C}'_{t,3}, \\
&= [\![\mathbf{s}'_{t-1}, \mathbf{D}^{(j)}, \mathbf{z}_{t-1}^{(j)}[\mathsf{read}(t)]]\!], [\![\mathbf{s}'_{t-1} \mapsto \mathbf{S}_t[\mathsf{write}(t)]]\!]_{\mathbf{B}_3, \mathbf{W}_3}
\end{aligned}
$$

with random. Here, we will rely on LWE with $\mathbf{s}'_{t-1}$ as the LWE secret.

2.3 Getting to $O(S+T)$ complexity

The main efficiency bottleneck of the above construction is that a fresh secret $\mathbf{S}_t \in \mathbb{Z}_q^{S \times n}$ is used for each computation step $t \in [T]$. This leads to ciphertext size, encryption time and decryption time at least linear in $T \cdot S$.

We show how to modify the scheme to achieve encryption time and ciphertext size $O(S+TL)$, and decryption time $O(S+T \cdot \mathsf{size}(f))$.

First idea. Suppose we set $\mathbf{S}_0 = \mathbf{S}_1 = \cdots = \mathbf{S}_T$. This way, we no longer have to update the entries outside $\mathsf{write}(t)$. We could then omit $\mathbf{C}_{t,2}, \mathbf{C}'_{t,2}$ from the ciphertext (and $\mathbf{K}_2$ from the secret key) and achieve encryption time and ciphertext size $O(S+TL)$, and decryption time $O(S+T \cdot \mathsf{size}(f))$. Unfortunately, the security proof and the scheme are both broken: the earlier proof crucially relies on the fact that $\mathbf{S}_0, \mathbf{S}_1, \ldots, \mathbf{S}_T$ are independent, and moreover, $[\![\mathbf{S}_0, \mathbf{A}_0, \mathbf{z}_0]\!], [\![\mathbf{S}_0, \mathbf{A}_0, \mathbf{z}_1]\!]$ leaks partial information about $\mathbf{S}_0$ whenever $\mathbf{z}_0 \neq \mathbf{z}_1$.

[5] Public-coin suffices, for the same reason as the schemes in [Wee22, HLL23]; see section Section 6 for details.

[6] The distribution on the left hand side and the one obtained by "appropriately multiplying" the elements on the right hand side can be shown to be statistically close. See Section 6 and Appendix D for more details.

Our strategy. Instead, we use fresh randomness for

$$\mathbf{S}_1[U_1], \mathbf{S}_2[U_2], \ldots, \mathbf{S}_T[U_T]$$

for some sets $U_1, U_2, \ldots, U_T$ where $\mathsf{write}(t) \subseteq U_t$ to be determined. We would then modify $(\mathbf{C}_{t,2}, \mathbf{C}'_{t,2})$ in the ciphertext as follows: we replace $\overline{\mathsf{write}(t)}$ with $\overline{\mathsf{write}(t)} \cap U_t$ to obtain

$$(\mathbf{C}_{t,2}, \mathbf{C}'_{t,2}) \coloneqq [\![\mathbf{S}_{t-1}[\overline{\mathsf{write}(t)} \cap U_t] \mapsto \mathbf{S}_t[\overline{\mathsf{write}(t)} \cap U_t]]\!]_{\mathbf{B}_2, \mathbf{W}_2}$$

As long as $|U_1|, \ldots, |U_T| = O(L)$, we would achieve the desired encryption, ciphertext size, and decryption time.

In the proof, for $t \geq 1$: instead of proving pseudorandomness of $\tilde{\mathbf{C}}_t^{(j)} = [\![\mathbf{S}_t, \mathbf{A}_0^{(j)}, \mathbf{z}_t^{(j)}]\!]$, it suffices to prove pseudorandomness of

$$\tilde{\mathbf{C}}_t^{(j)}[U_t] = [\![\mathbf{S}_t[U_t], \mathbf{A}_0^{(j)}, \mathbf{z}_t^{(j)}[U_t]]\!]$$

That is, in $\mathsf{H}_t, t \geq 1$, we want to replace

$$\begin{aligned}
&\tilde{\mathbf{C}}_t^{(j)}[U_t], \mathbf{C}_{t+1,1}, \mathbf{C}'_{t+1,1}, \mathbf{C}_{t+1,2}, \mathbf{C}'_{t+1,2} \\
&= [\![\mathbf{S}_t[U_t], \mathbf{A}_0^{(j)}, \mathbf{z}_t^{(j)}[U_t]]\!], \\
&\quad [\![\mathbf{S}_t[\mathsf{read}(t+1)] \mapsto \mathbf{s}'_t]\!]_{\mathbf{B}_1, \mathbf{W}_1} \\
&\quad [\![\mathbf{S}_t[\overline{\mathsf{write}(t+1)} \cap U_{t+1}] \mapsto \mathbf{S}_{t+1}[\overline{\mathsf{write}(t+1)} \cap U_{t+1}]]\!]_{\mathbf{B}_2, \mathbf{W}_2}
\end{aligned}$$

with random. Moreover, pseudorandomness needs to hold against a distinguisher that gets as auxiliary input

$$\left.\begin{array}{l} \mathbf{S}_{t'+1}[\mathsf{read}(t'+2)] \cdot (\mathbf{B}_1 \mid \mathbf{W}_1) \\ \mathbf{S}_{t'+1}[\overline{\mathsf{write}(t'+2)} \cap U_{t'+2}] \cdot (\mathbf{B}_2 \mid \mathbf{W}_2) \end{array}\right\} \quad \text{for } t' \geq t$$

In the reduction, we also need to simulate

$$(\mathbf{C}_{t'+2,1}, \mathbf{C}'_{t'+2,1}), (\mathbf{C}_{t'+2,2}, \mathbf{C}'_{t'+2,2}), (\mathbf{C}_{t'+1,3}, \mathbf{C}'_{t'+1,3}), t' \geq t$$

which expand to:

$$\begin{array}{l}
(\mathbf{S}_{t'+1}[\mathsf{read}(t'+2)]\mathbf{B}_1, \mathbf{S}_{t'+1}[\mathsf{read}(t'+2)]\mathbf{W}_1 + \mathbf{1}_L^\top \cdot \mathbf{s}'_{t'+1}\mathbf{G}) \\
(\mathbf{S}_{t'+1}[\overline{\mathsf{write}(t'+2)} \cap U_{t'+2}]\mathbf{B}_2, \mathbf{S}_{t'+1}[\overline{\mathsf{write}(t'+2)} \cap U_{t'+2}]\mathbf{W}_2 + \mathbf{S}_{t'+2}[\overline{\mathsf{write}(t'+2)} \cap U_{t'+2}]\mathbf{G}) \\
(\mathbf{s}'_{t'}\mathbf{B}_3, (\mathbf{I}_L \otimes \mathbf{s}'_{t'})\mathbf{W}_3 + \mathbf{S}_{t'+1}[\mathsf{write}(t'+1)]\mathbf{G})
\end{array}$$

respectively. To do this,

- we rely on the auxiliary input to handle the first two rows;
- for the third row, we sample $\mathbf{S}_{t'+1}[U_{t'+1}]$ at random (which are by construction fresh randomness)

The challenge and our fix. The challenge is that the auxiliary input might leak information about the LWE secret $\mathbf{S}_t$ due to correlations between $\mathbf{S}_t$ and $\mathbf{S}_{t'+1}$. A natural starting point is to set $U_{t'+1} \coloneqq \mathsf{write}(t'+1) \cup \mathsf{read}(t'+2)$, so that the auxiliary inputs are independent of $\mathbf{S}_t$. The problem is that $[\![\mathbf{S}_t[U_t], \mathbf{A}_0^{(j)}, \mathbf{z}_t^{(j)}[U_t]]\!]$ would then depend on $\mathbf{S}_t[\mathsf{read}(t+1)]$ and $\mathsf{read}(t+1) \nsubseteq U_{t+1}$.

Instead, we set $U_t \coloneqq \mathsf{write}(t) \cup \mathsf{read}(t)$ for all t. This means that $\overline{\mathsf{write}(t'+2)} \cap U_{t'+2} \subseteq \mathsf{read}(t'+2)$, so we can replace the above auxiliary input with:

$$\left.\begin{array}{l} \mathbf{S}_{t'+1}[\mathsf{read}(t'+2)] \cdot (\mathbf{B}_1 \mid \mathbf{W}_1) \\ \mathbf{S}_{t'+1}[\mathsf{read}(t'+2)] \cdot (\mathbf{B}_2 \mid \mathbf{W}_2) \end{array}\right\} \quad \text{for } t' \geq t$$

Next, observe that we can simulate $\mathbf{S}_{t+1}, \ldots, \mathbf{S}_T$ given $\mathbf{S}_t[\overline{\mathsf{read}(t+1)}]$, so we can also simulate the auxiliary input given

$$\mathbf{S}_t[\overline{\mathsf{read}(t+1)}] \cdot (\mathbf{B}_1 \mid \mathbf{W}_1)$$
$$\mathbf{S}_t[\overline{\mathsf{read}(t+1)}] \cdot (\mathbf{B}_2 \mid \mathbf{W}_2)$$

At this point, our goal is to replace

$$[\![\mathbf{S}_t[\mathsf{write}(t) \cup \mathsf{read}(t)], \mathbf{A}_0^{(j)}, \mathbf{z}_t^{(j)}[\mathsf{write}(t) \cup \mathsf{read}(t)]]\!]$$
$$\mathbf{S}_t[\mathsf{read}(t+1)] \cdot (\mathbf{B}_1 \mid \mathbf{W}_1)$$
$$\mathbf{S}_t[\overline{\mathsf{write}(t+1) \cap \mathsf{read}(t+1)}] \cdot (\mathbf{B}_2 \mid \mathbf{W}_2)$$

with random, given as auxiliary input $\mathbf{S}_t[\overline{\mathsf{read}(t+1)}] \cdot (\mathbf{B}_1 \mid \mathbf{W}_1)$ and $\mathbf{S}_t[\overline{\mathsf{read}(t+1)}] \cdot (\mathbf{B}_2 \mid \mathbf{W}_2)$. We stress that we cannot give out $\mathbf{S}_t[\overline{\mathsf{read}(t+1)}]$ in the auxiliary input since that could leak information about $\mathbf{S}_t[\mathsf{write}(t) \cup \mathsf{read}(t)]$. In fact, by the LWE assumption with secret $\mathbf{S}_t$, we can prove the following stronger (and simpler) statement: we can replace

$$[\![\mathbf{S}_t, \mathbf{A}_0^{(j)}, \mathbf{z}_t^{(j)}]\!]$$
$$\mathbf{S}_t[\mathsf{read}(t+1)] \cdot (\mathbf{B}_1 \mid \mathbf{W}_1)$$
$$\mathbf{S}_t[\overline{\mathsf{write}(t+1) \cap \mathsf{read}(t+1)}] \cdot (\mathbf{B}_2 \mid \mathbf{W}_2)$$

with random, given auxiliary input $\mathbf{S}_t[\overline{\mathsf{read}(t+1)}] \cdot (\mathbf{B}_1 \mid \mathbf{W}_1)$ and $\mathbf{S}_t[\overline{\mathsf{read}(t+1)}] \cdot (\mathbf{B}_2 \mid \mathbf{W}_2)$.

Final step. At the end, we use the fact that $\mathbf{z}_T^{(j)}[1] = 1$ (i.e., the j'th query is unauthorized). Namely, by the LWE assumption with secret $\mathbf{S}_T$, we replace $[\![\mathbf{S}_T, \mathbf{A}_0^{(j)}, \mathbf{z}_T^{(j)}]\!]$, $\mathbf{z}_T^{(j)}[1]\mathbf{S}_T[1]\mathbf{G}\mathbf{R}^{(j)}$, $\mathbf{S}_T[1]\mathbf{B}_4$ and $\mathbf{S}_T[1]\mathbf{P} + \lfloor q/2 \rfloor$ with random.

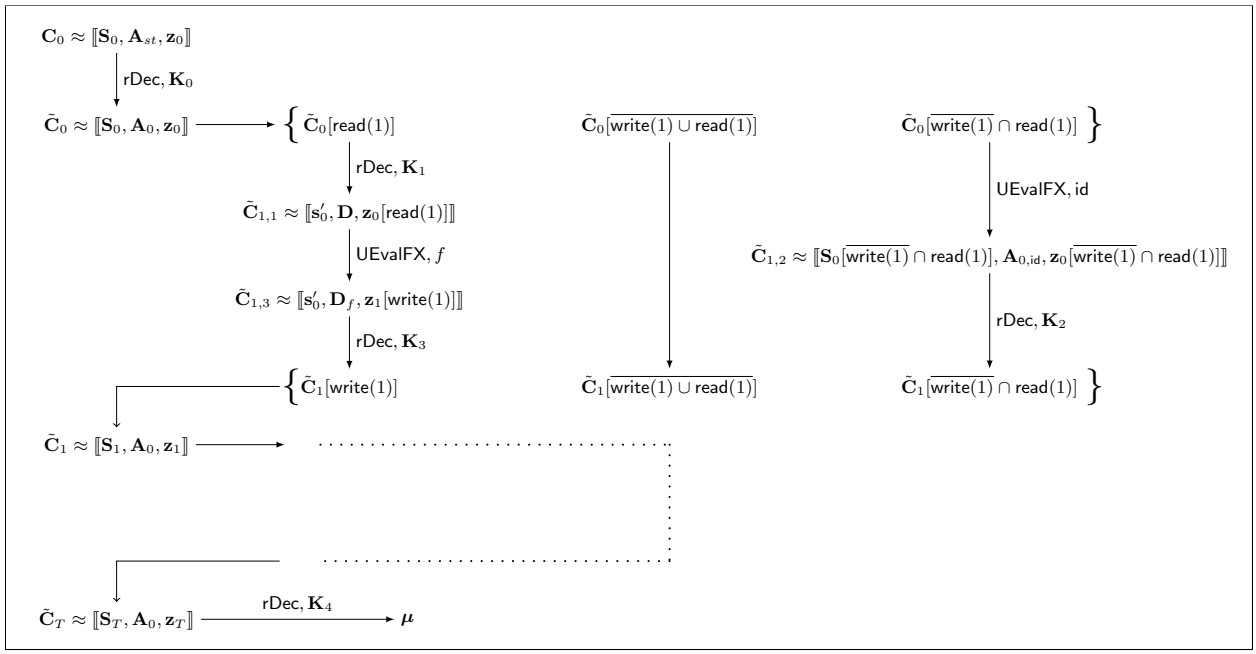

Fig. 3. Summary of how decryption works in our scheme, where UEvalFX and rDec are defined in Sections 3 and 4.

3 Preliminaries

Notations. We use boldface lower case for row vectors (e.g. $\mathbf{r}$) and boldface upper case for matrices (e.g. $\mathbf{R}$). For integral vectors and matrices (i.e., those over $\mathbb{Z}$), we use the notation $\|\mathbf{r}\|$, $\|\mathbf{R}\|$ to denote the maximum absolute value over all the entries. We use $v \leftarrow D$ to denote a random sample from a distribution D, as well as $v \leftarrow S$ to denote a uniformly random sample from a set S. We use $\approx_s$ and $\approx_c$ as the abbreviation for statistically close and computationally indistinguishable. We denoted by $\mathcal{D}_{\mathbb{Z}^m,\chi}$ the (centered) discrete Gaussian distribution over $\mathbb{Z}^m$ with parameter χ, i.e., the distribution over $\mathbb{Z}^m$ where for all $\mathbf{x}$, $\Pr[\mathbf{x}] \propto e^{-\pi\cdot(x_1^2+\cdots+x_m^2)/\chi^2}$. Given $\mathbf{M}_1 \in \mathbb{Z}^{n\times m}, \mathbf{M}_2 \in \mathbb{Z}^{n\times h}, \mathbf{M}_3 \in \mathbb{Z}^{t\times m}$, we will denote by $(\mathbf{M}_1 \mid \mathbf{M}_2) \in \mathbb{Z}^{n\times(m+h)}$ and $(\mathbf{M}_1 \cup \mathbf{M}_3) \in \mathbb{Z}^{(n+t)\times m}$ their horizontal and vertical concatenation respectively. For $\mathbf{M} = (\mathbf{M}_1 \cup \ldots \cup \mathbf{M}_L) \in \mathbb{Z}^{Ln\times m}$, with $\mathbf{M}_i \in \mathbb{Z}_q^{n\times m}$, we denote by $\mathsf{diag}_L(\mathbf{M}) \in \mathbb{Z}^{Ln\times Lm}$ the matrix with $(\mathbf{M}_i)_{i\in[L]}$ on the diagonal, and zeros elsewhere, and by $\mathsf{flat}_L(\mathbf{M}) = (\mathbf{M}_1 \mid \ldots \mid \mathbf{M}_L)$. We omit the subscript L when L equals the number of rows of $\mathbf{M}$.

Trapdoor sampling algorithms and relative properties can all be found in Appendix B.

3.1 Lattices background

Lemma 1 (Noise Flooding [GKPV10]). *Let $n \in \mathbb{N}$. For any real $\chi > \omega(\sqrt{\log n})$, and any $\mathbf{c} \in \mathbb{Z}^n$, it holds $\mathsf{SD}(\mathcal{D}_{\mathbb{Z}^n,\chi}, \mathcal{D}_{\mathbb{Z}^n,\chi} + \mathbf{c}) \le \|\mathbf{c}\|/\chi$. In particular, if $\chi \ge \lambda^{\omega(1)} \cdot \|\mathbf{c}\|$, one has $\mathcal{D}_{\mathbb{Z}^n,\chi} \approx_s \mathcal{D}_{\mathbb{Z}^n,\chi} + \mathbf{c}$.*

Homomorphic Computation on Matrices. The following is an extension of [BGG+14] to circuits of unbounded depth.

Lemma 2 (Unbounded Homomorphic Evaluation [HLL23, Theorem 12, 13]). *Let $f : \{0,1\}^L \to \{0,1\}^L$ and $\mathbf{x} \in \{0,1\}^L$. Suppose*

$$\mathbf{C} = \begin{bmatrix} \widetilde{\mathbf{A}} \\ \mathbf{s}\widetilde{\mathbf{A}} + \mathbf{e}_{\mathsf{fhe}} \end{bmatrix} - \mathsf{bits}(\mathbf{s}') \otimes \mathbf{G},$$
$$\mathbf{c}_{\mathsf{circ}} = \mathbf{s}'(\mathbf{D}_{\mathsf{circ}} - (1 \mid \mathsf{bits}(\mathbf{C})) \otimes \mathbf{G}) + \mathbf{e}_{\mathsf{circ}},$$
$$\mathbf{c}_{\mathsf{att}} = (\mathbf{I}_L \otimes \mathbf{s}')\mathbf{D}_{\mathsf{att}} - \mathbf{x}^\top \cdot \mathbf{s}'\mathbf{G} + \mathbf{e}_{\mathsf{att}},$$

where $\widetilde{\mathbf{A}} \leftarrow \mathbb{Z}_q^{(n-1)\times nm\lceil\log q\rceil}, \mathbf{D}_{\mathsf{circ}} \leftarrow \mathbb{Z}_q^{n\times(n^2m\lceil\log q\rceil^2+1)m}, \mathbf{D}_{\mathsf{att}} \leftarrow \mathbb{Z}_q^{Ln\times m}, \mathbf{s}' = (\mathbf{s} \mid -1) \in \mathbb{Z}^n$ and $\|\mathbf{e}_{\mathsf{att}}\|, \|\mathbf{s}\| \le B_{\mathsf{HLL}}, \|\mathbf{e}_{\mathsf{fhe}}\|, \|\mathbf{e}_{\mathsf{circ}}\| \le 2^{-\lambda}B_{\mathsf{HLL}}$. There are two efficient algorithms

$$\mathsf{UEvalF}(\mathbf{D}_{\mathsf{att}}, \mathbf{D}_{\mathsf{circ}}, f) = \mathbf{D}_f \qquad \mathsf{UEvalFX}(\mathbf{D}_{\mathsf{att}}, \mathbf{D}_{\mathsf{circ}}, \mathbf{c}_{\mathsf{att}}, (\mathbf{c}_{\mathsf{circ}}, \mathbf{C}), f, \mathbf{x}) = \mathbf{c}_{f,\mathbf{x}}$$

satisfying, with all but negligible probability

$$\mathbf{c}_{f,\mathbf{x}} = (\mathbf{I}_L \otimes \mathbf{s}')\mathbf{D}_f - f(\mathbf{x})^\top \cdot \mathbf{s}'\mathbf{G} + \mathbf{e}_{f,\mathbf{x}}, \quad \text{and} \quad \|\mathbf{e}_{f,\mathbf{x}}\| \le 2^{-\lambda}B_{\mathsf{HLL}}.$$

Proof (sketch). Given an attribute encoding $\mathbf{c}_{\mathsf{att}}$ of $\mathbf{x}$ with noise norm B_{HLL}, one can first apply the bootstrapping algorithm from HLL, [HLL23, Theorem 12], to reduce the noise to level $B_{\mathsf{HLL}} \cdot 2^{-\lambda}$. Given this encoding of $\mathbf{x}$ with small noise, one can perform the unbounded evaluation procedure from HLL, [HLL23, Theorem 13], to obtain an encoding $\mathbf{c}_{f,\mathbf{x}}$ of $f(\mathbf{x})$ with noise level B_{HLL}. Running the HLL bootstrapping algorithm once again yields an encoding of $f(\mathbf{x})$ with noise level $B_{\mathsf{HLL}} \cdot 2^{-\lambda}$ as required. □

It is convenient to define a notational short-hand for the circular encoding, namely:

$$\odot[\![\mathbf{s}', \mathbf{D}_{\mathsf{circ}}]\!] := \begin{cases} \hat{\mathbf{C}} := \begin{bmatrix} \widetilde{\mathbf{A}} \\ \mathbf{s}\widetilde{\mathbf{A}} + \hat{\mathbf{e}} \end{bmatrix} + \mathsf{bits}(\mathbf{s}') \otimes \mathbf{G} \\ \hat{\mathbf{C}}' := \mathbf{s}'(\mathbf{D}_{\mathsf{circ}} - \mathsf{bits}(\hat{\mathbf{C}}) \otimes \mathbf{G}) + \hat{\mathbf{e}}' \end{cases}$$

where $\mathbf{s}' = (\mathbf{s} \mid -1)$, for $\mathbf{s} \leftarrow \mathcal{D}_{\mathbb{Z}^{n-1},\chi}$, $\mathbf{D}_{\mathsf{circ}} \in \mathbb{Z}_q^{n\times(n^2m\lceil\log q\rceil^2+1)m}$ and we sample $\widetilde{\mathbf{A}} \leftarrow \mathbb{Z}_q^{(n-1)\times nm\lceil\log q\rceil}, \hat{\mathbf{e}} \leftarrow \mathcal{D}_{\mathbb{Z}^{nm\lceil\log q\rceil},\chi}, \hat{\mathbf{e}}' \leftarrow \mathcal{D}_{\mathbb{Z}^{(n^2m\lceil\log q\rceil^2+1)m},\chi}$.

The notation is extended naturally when $\mathbf{s}' \in \mathbb{Z}^n$ is replaced with a $\mathbf{S} \in \mathbb{Z}^{L\times n}$: $\odot[\![\mathbf{S}, \mathbf{D}_{\mathsf{circ}}]\!]$ will indicate the vertical concatenation of $\odot[\![\mathbf{S}[i], \mathbf{D}_{\mathsf{circ}}]\!]$, for each $i \in [L]$, with fresh randomness for each row of $\mathbf{S}$.

3.2 Computational Assumptions

The standard LWE and the circular short-secret LWE assumptions are recalled in Appendix B.

(Public-Coin) Circular Small-Secret Evasive LWE. Let Samp be a PPT algorithm that on input 1^λ and randomness coins, outputs

$$\mathbf{A}' \in \mathbb{Z}_q^{Jn\times m'}, (\mathbf{P}_\gamma)_{\gamma=0,\dots,4} \in \mathbb{Z}_q^{n\times t_0+\dots+t_4}, (\Gamma_\gamma)_{\gamma=0,\dots4} \subseteq [J], \mathsf{aux} \in \{0,1\}^*$$
$$\chi, \chi_{\mathsf{pre}}, \chi_{\mathsf{post}}, (\chi_\gamma)_{\gamma=0,\dots,4}$$

We define the following advantage functions

$$\mathsf{Adv}^{\mathsf{pre}}_{\mathcal{A}_0}(1^\lambda) := \Pr\left[\mathcal{A}_0\left(\begin{array}{c}\mathsf{flat}(\mathbf{S}^\dagger)\mathbf{A}'+\mathbf{e}'_{\mathsf{pre}},\\ (\mathbf{S}^\dagger[\Gamma_\gamma]\mathbf{B}_\gamma+\mathbf{E}_{\gamma,\mathsf{pre}}, \mathbf{S}^\dagger[\Gamma_\gamma]\mathbf{P}_\gamma+\mathbf{E}''_\gamma)_{\gamma=0,\dots,4},\\ \odot\llbracket\mathbf{S}^\dagger,\mathbf{D}_{\mathsf{circ}}\rrbracket,\\ \mathbf{A}', (\mathbf{B}_\gamma)_{\gamma=0,\dots,4}, \mathbf{D}_{\mathsf{circ}}, \mathsf{aux}, \mathsf{coins}\end{array}\right)=1\right]$$
$$-\Pr\left[\mathcal{A}_0\left(\begin{array}{c}\mathbf{c}',\\ (\mathbf{C}_\gamma, \mathbf{C}''_\gamma)_{\gamma=0,\dots,4},\\ \mathbf{C},\\ \mathbf{A}', (\mathbf{B}_\gamma)_{\gamma=0,\dots,4}, \mathbf{D}_{\mathsf{circ}}, \mathsf{aux}, \mathsf{coins}\end{array}\right)=1\right],$$

$$\mathsf{Adv}^{\mathsf{post}}_{\mathcal{A}_1}(1^\lambda) := \Pr\left[\mathcal{A}_1\left(\begin{array}{c}\mathsf{flat}(\mathbf{S}^\dagger)\mathbf{A}'+\mathbf{e}'_{\mathsf{post}},\\ (\mathbf{S}^\dagger[\Gamma_\gamma]\mathbf{B}_\gamma+\mathbf{E}_{\gamma,\mathsf{post}}, \mathbf{K}_\gamma)_{\gamma=0,\dots,4},\\ \odot\llbracket\mathbf{S}^\dagger,\mathbf{D}_{\mathsf{circ}}\rrbracket,\\ \mathbf{A}', (\mathbf{B}_\gamma)_{\gamma=0,\dots,4}, \mathbf{D}_{\mathsf{circ}}, \mathsf{aux}, \mathsf{coins}\end{array}\right)=1\right]$$
$$-\Pr\left[\mathcal{A}_1\left(\begin{array}{c}\mathbf{c}',\\ (\mathbf{C}_\gamma, \mathbf{K}_\gamma)_{\gamma=0,\dots,4},\\ \mathbf{C},\\ \mathbf{A}', (\mathbf{B}_\gamma)_{\gamma=0,\dots,4}, \mathbf{D}_{\mathsf{circ}}, \mathsf{aux}, \mathsf{coins}\end{array}\right)=1\right],$$

where $(\mathbf{A}', (\mathbf{P}_\gamma)_{\gamma=0,\dots,4}, (\Gamma_\gamma)_{\gamma=0,\dots,4}, \mathsf{aux}) \leftarrow \mathsf{Samp}(1^\lambda; \mathsf{coins})$,

$$\begin{array}{ll}\mathbf{S}^\dagger \leftarrow \mathcal{D}_{\mathbb{Z}^{J\times n},\chi}, & \mathbf{B}_\gamma \leftarrow \mathbb{Z}_q^{n\times m}, \text{ for } \gamma=0,\dots,4,\\ \mathbf{C}_\gamma \leftarrow \mathbb{Z}_q^{|\Gamma_\gamma|\times m}, & \mathbf{C}''_\gamma \leftarrow \mathbb{Z}_q^{|\Gamma_\gamma|\times t_\gamma}, \text{ for } \gamma=0,\dots,4,\\ \mathbf{c}' \leftarrow \mathbb{Z}_q^{m'}, & \mathbf{C} \leftarrow (\mathbb{Z}_q^{n\times nm\lceil\log q\rceil} \times \mathbb{Z}_q^{(n^2m\lceil\log q\rceil^2+1)m})^J\\ \mathbf{D}_{\mathsf{circ}} \leftarrow \mathbb{Z}_q^{n\times n\times(n^2m\lceil\log q\rceil^2+1)m}, & \end{array}$$

and

$$\begin{array}{ll}\mathbf{e}'_{\mathsf{pre}} \leftarrow \mathcal{D}_{\mathbb{Z}^{m'},\chi_{\mathsf{pre}}}, & \mathbf{E}_{\gamma,\mathsf{pre}} \leftarrow \mathcal{D}_{\mathbb{Z}^{|\Gamma_\gamma|\times m},\chi_{\mathsf{pre}}},\\ \mathbf{e}'_{\mathsf{post}} \leftarrow \mathcal{D}_{\mathbb{Z}^{m'},\chi_{\mathsf{post}}}, & \mathbf{E}_{\gamma,\mathsf{post}} \leftarrow \mathcal{D}_{\mathbb{Z}^{|\Gamma_\gamma|\times m},\chi_{\mathsf{post}}},\\ \mathbf{E}''_\gamma \leftarrow \mathcal{D}_{\mathbb{Z}^{|\Gamma_\gamma|\times t_\gamma},\chi_\gamma}, & \mathbf{K}_\gamma \leftarrow \mathbf{B}_\gamma^{-1}(\mathbf{P}_\gamma, \tau), \text{ for } \gamma=0,\dots,4.\end{array}$$

We say that the (public-coin) circular small-secret evasive LWE assumption holds if for every PPT Samp, $\mathcal{A}_1$, there exists another PPT $\mathcal{A}_0$ and a polynomial $p(\cdot)$ such that

$$\mathsf{Adv}^{\mathsf{pre}}_{\mathcal{A}_0}(1^\lambda) \geq \mathsf{Adv}^{\mathsf{post}}_{\mathcal{A}_1}(1^\lambda)/p(\lambda) - \mathsf{negl}(\lambda).$$

Remark 1 (single $\mathbf{B}$). We believe a small tweak to our construction following [Wee22] would allow us to base security on evasive LWE with a single matrix $\mathbf{B}$. The idea is as follows: replace each

$$\mathbf{S}_\gamma\mathbf{B}_\gamma, \mathbf{B}_\gamma^{-1}(\mathbf{P}_\gamma), \mathbf{B}_\gamma \in \mathbb{Z}_q^{n\times m}, \gamma \in \{0,1,2,3,4\}$$

in the scheme with:

$$\mathbf{S}_\gamma^+\mathbf{B}, \mathbf{B}^{-1}(\mathbf{P}_\gamma^+)$$

where $\mathbf{B} \leftarrow \mathbb{Z}_q^{5n\times 5m}$ so that

$$\mathbf{S}_\gamma^+\mathbf{B} \cdot \mathbf{B}^{-1}(\mathbf{P}_\gamma^+) = \mathbf{S}_\gamma\mathbf{P}_\gamma$$

Concretely, we pad $\mathbf{S}_\gamma$ horizontally with random values to obtain $\mathbf{S}_\gamma^+$, and we pad $\mathbf{P}_\gamma$ vertically with all zeroes to obtain $\mathbf{P}_\gamma^+$. This is sufficient for functionality. However, writing down the pre-condition for evasive LWE, we need to account for additional cross-term interactions of the form:

$$\mathbf{S}_\beta^+ \cdot \mathbf{P}_\gamma^+ \quad \text{for } \beta \neq \gamma',$$

which is of the form $\mathbf{S}_{\beta,\gamma}\mathbf{P}_\gamma$ where $\mathbf{S}_{\beta,\gamma}$ comes from the randomness used to pad $\mathbf{S}_\beta$. We should be able to prove pseudorandomness of these additional terms similar to the proof of pseudorandomness for $\mathbf{S}_\gamma\mathbf{P}_\gamma$.

Remark 2 (justification for the assumption). The assumption we use is the same as the evasive LWE variant used in [HLL23] (cf. Assumption 2 and remarks therein) and refers to the public-coin setting wherein the adversaries $\mathcal{A}_0$ and $\mathcal{A}_1$ receive the coin tosses coins used by Samp (cf. [Wee22, Remark 1], [BÜW24, Fig 2]) We refer to [Wee22] for discussion and justification regarding the evasive LWE assumption, and [HLL23] for the circular small-secret variant. We stress none of the obfuscation-based counter-examples and recent attacks in [BÜW24] are applicable to the public-coin setting.

Remark 3 (additional comment about the assumption in light of later works). After the appearance of this work, several papers revisited variants of the evasive LWE assumption and gave counterexamples to some of these formulations; see, for instance, [AMYY25, DJM+25, HJL25]. Among these, the work most directly relevant to our setting is [AMYY25], since it gives a counterexample to the circular evasive assumption used in the construction of Hsieh, Lin, and Luo [HLL23]. Because our work relies on the same assumption as [HLL23], the conceptual implications of that counterexample apply to our setting as well. At the same time, as with the implications for [HLL23], this should be understood as evidence against the underlying assumption rather than as a direct attack on the resulting scheme. In particular, to the best of our knowledge, it does not currently translate into an explicit break of the concrete construction proved secure from that assumption.

3.3 Attribute-based encryption

Syntax. A key policy attribute-based encryption (KP-ABE) scheme for some class $\mathcal{F}$ consists of four algorithms:

- $\mathsf{Setup}(1^\lambda, \mathcal{F}) \to (\mathsf{mpk}, \mathsf{msk})$. The setup algorithm gets as input the security parameter 1^λ and class description $\mathcal{F}$. It outputs the master public key mpk and the master secret key msk.
- $\mathsf{Enc}(\mathsf{mpk}, \mathbf{x}, \boldsymbol{\mu}) \to \mathsf{ct}_\mathbf{x}$. The encryption algorithm gets as input mpk, an input $\mathbf{x}$ and a message $\boldsymbol{\mu} \in \{0,1\}^\lambda$. It outputs a ciphertext $\mathsf{ct}_\mathbf{x}$. Note that $\mathbf{x}$ is public given $\mathsf{ct}_\mathbf{x}$.
- $\mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, f) \to \mathsf{sk}_f$. The key generation algorithm gets as input $\mathsf{mpk}, \mathsf{msk}$ and $f \in \mathcal{F}$. It outputs a secret key sk_f. Note that f is public given sk_f.
- $\mathsf{Dec}(\mathsf{mpk}, \mathsf{sk}_f, f, \mathsf{ct}_\mathbf{x}, \mathbf{x}) \to \boldsymbol{\mu}$. The decryption algorithm gets as input sk_f and $\mathsf{ct}_\mathbf{x}$ along with mpk. It outputs a message $\boldsymbol{\mu}$.

Correctness. For all inputs $\mathbf{x}$ and f with $f(\mathbf{x}) = 0$ and all $\boldsymbol{\mu} \in \{0,1\}^\lambda$, we require

$$\Pr\left[\mathsf{Dec}(\mathsf{mpk}, \mathsf{sk}_f, \mathsf{ct}_\mathbf{x}) = \boldsymbol{\mu} : \begin{array}{c}(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda, \mathcal{F}) \\ \mathsf{sk}_f \leftarrow \mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, f) \\ \mathsf{ct}_\mathbf{x} \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathbf{x}, \boldsymbol{\mu})\end{array}\right] = 1 - \mathsf{negl}(\lambda).$$

Security. For a stateful adversary $\mathcal{A}$, we define the advantage function

$$\mathsf{Adv}^{\mathrm{ABE}}_{\mathcal{A}}(\lambda) \coloneqq \Pr\left[b = b' : \begin{array}{l} (\mathbf{x}^*, f_1, \ldots, f_Q) \leftarrow \mathcal{A}(1^\lambda) \\ (\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda, \mathcal{F}) \\ \mathsf{sk}_{f_i} \leftarrow \mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, f_i), i = 1, \ldots, Q \\ (\boldsymbol{\mu}_0, \boldsymbol{\mu}_1) \leftarrow \mathcal{A}(\mathsf{mpk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_Q) \\ b \leftarrow \{0,1\}; \mathsf{ct}_{\mathbf{x}^*} \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathbf{x}^*, \boldsymbol{\mu}_b) \\ b' \leftarrow \mathcal{A}(\mathsf{ct}_{\mathbf{x}^*}) \end{array}\right] - \frac{1}{2},$$

with the restriction that all functions f_i satisfy $f_i(\mathbf{x}^*) = 1$. An ABE scheme is *very selectively secure* if for all PPT adversaries $\mathcal{A}$, the advantage $\mathsf{Adv}^{\mathrm{ABE}}_{\mathcal{A}}(\lambda)$ is a negligible function in λ.

4 Recoding

In this section we present the recoding algorithms that will be used in the main construction. The proofs of their properties are deferred to Appendix C. Fix $\mathbf{B}, \mathbf{W} \in \mathbb{Z}_q^{n\times m}, \mathbf{W}' \in \mathbb{Z}_q^{Ln\times m}$, $\mathbf{s} \in \mathbb{Z}_q^n, \mathbf{S}, \mathbf{S}' \in \mathbb{Z}_q^{L\times n}, \mathbf{A}_0, \mathbf{A}_0' \in \mathbb{Z}_q^{n\times m}, \mathbf{A} \in \mathbb{Z}_q^{Ln\times m}$.

We define:

1. dual encoding to encoding

$$\begin{aligned} [\![\mathbf{S} \mapsto \mathbf{s}]\!]_{\mathbf{B},\mathbf{W}} &\coloneqq \begin{cases} \mathbf{C} \coloneqq \mathbf{SB} \\ \mathbf{C}' \coloneqq \mathbf{SW} + \mathbf{1}_L^\top \cdot \mathbf{sG} \end{cases} \\ [\![\mathbf{A}_0 \mapsto \mathbf{A}]\!]_{\mathbf{W}} &\coloneqq \mathbf{1}_L^\top \otimes \mathbf{A}_0\mathbf{G}^{-1}(\mathbf{W}) - (\mathbf{I}_L \otimes \mathbf{W})(\mathbf{I}_L \otimes \mathbf{G})^{-1}(\mathbf{A}) \\ \mathsf{rDec}_0\big(\mathbf{B}, \mathbf{W}, \mathbf{A}, \mathbf{C}_{\mathsf{in}}, \mathbf{C}', \mathbf{z}\big) &\coloneqq \mathsf{diag}_L(\mathbf{C}')(\mathbf{I}_L \otimes \mathbf{G})^{-1}(\mathbf{A}) - \mathsf{diag}(\mathbf{z})\mathbf{C}' - \mathbf{C}_{\mathsf{in}}\mathbf{G}^{-1}(\mathbf{W}) \\ \mathsf{rDec}\big(\mathbf{B}, \mathbf{W}, \mathbf{A}, \mathbf{K}, \mathbf{C}_{\mathsf{in}}, (\mathbf{C}, \mathbf{C}'), \mathbf{z}\big) &\coloneqq \mathsf{diag}_L(\mathbf{C}) \cdot \mathbf{K} + \mathsf{rDec}_0\big(\mathbf{B}, \mathbf{W}, \mathbf{A}, \mathbf{C}_{\mathsf{in}}, \mathbf{C}', \mathbf{z}\big) \end{aligned}$$

2. dual encoding to dual encoding

$$\begin{aligned} [\![\mathbf{S} \mapsto \mathbf{S}']\!]_{\mathbf{B},\mathbf{W}} &\coloneqq \begin{cases} \mathbf{C} \coloneqq \mathbf{SB} \\ \mathbf{C}' \coloneqq \mathbf{SW} + \mathbf{S}'\mathbf{G} \end{cases} \\ [\![\mathbf{A}_0 \mapsto \mathbf{A}_0']\!]_{\mathbf{W}} &\coloneqq \mathbf{A}_0\mathbf{G}^{-1}(\mathbf{W}) - \mathbf{W}\mathbf{G}^{-1}(\mathbf{A}_0') \\ \mathsf{rDec}_0\big(\mathbf{B}, \mathbf{W}, \mathbf{A}_0', \mathbf{C}_{\mathsf{in}}, \mathbf{C}', \mathbf{z}\big) &\coloneqq \mathbf{C}'\mathbf{G}^{-1}(\mathbf{A}_0') - \mathsf{diag}(\mathbf{z})\mathbf{C}' - \mathbf{C}_{\mathsf{in}}\mathbf{G}^{-1}(\mathbf{W}) \\ \mathsf{rDec}\big(\mathbf{B}, \mathbf{W}, \mathbf{A}_0', \mathbf{K}, \mathbf{C}_{\mathsf{in}}, (\mathbf{C}, \mathbf{C}'), \mathbf{z}\big) &\coloneqq \mathbf{C} \cdot \mathbf{K} + \mathsf{rDec}_0\big(\mathbf{B}, \mathbf{W}, \mathbf{A}_0', \mathbf{C}_{\mathsf{in}}, \mathbf{C}', \mathbf{z}\big) \end{aligned}$$

3. encoding to dual encoding

$$\begin{aligned} [\![\mathbf{s} \mapsto \mathbf{S}]\!]_{\mathbf{B},\mathbf{W}'} &\coloneqq \begin{cases} \mathbf{C} \coloneqq \mathbf{sB} \\ \mathbf{C}' \coloneqq (\mathbf{I}_L \otimes \mathbf{s})\mathbf{W}' + \mathbf{SG} \end{cases} \\ [\![\mathbf{A} \mapsto \mathbf{A}_0]\!]_{\mathbf{W}'} &\coloneqq \mathsf{diag}_L(\mathbf{A})(\mathbf{I}_L \otimes \mathbf{G})^{-1}(\mathbf{W}') - \mathbf{W}'\mathbf{G}^{-1}(\mathbf{A}_0) \\ \mathsf{rDec}_0\big(\mathbf{B}, \mathbf{W}', \mathbf{A}_0, \mathbf{C}_{\mathsf{in}}, \mathbf{C}', \mathbf{z}\big) &\coloneqq \mathbf{C}'\mathbf{G}^{-1}(\mathbf{A}_0) - \mathsf{diag}(\mathbf{z})\mathbf{C}' - \mathsf{diag}_L(\mathbf{C}_{\mathsf{in}})(\mathbf{I}_L \otimes \mathbf{G})^{-1}(\mathbf{W}') \\ \mathsf{rDec}\big(\mathbf{B}, \mathbf{W}', \mathbf{A}_0, \mathbf{K}, \mathbf{C}_{\mathsf{in}}, (\mathbf{C}, \mathbf{C}'), \mathbf{z}\big) &\coloneqq (\mathbf{I}_L \otimes \mathbf{C}) \cdot \mathbf{K} + \mathsf{rDec}_0\big(\mathbf{B}, \mathbf{W}', \mathbf{A}_0, \mathbf{C}_{\mathsf{in}}, \mathbf{C}', \mathbf{z}\big) \end{aligned}$$

Lemma 3 (recoding dual-to-encoding correctness). *Suppose*

$$\begin{aligned} \mathbf{C}_{\mathsf{in}} &= [\![\mathbf{S}, \mathbf{A}_0, \mathbf{z}]\!] \\ (\mathbf{C}, \mathbf{C}') &= [\![\mathbf{S} \mapsto \mathbf{s}]\!]_{\mathbf{B},\mathbf{W}} \\ (\mathbf{I}_L \otimes \mathbf{B}) \cdot \mathbf{K} &= [\![\mathbf{A}_0 \mapsto \mathbf{A}]\!]_{\mathbf{W}}. \end{aligned}$$

Then,

$$[\![\mathbf{s}, \mathbf{A}, \mathbf{z}]\!] = \mathsf{rDec}(\mathbf{B}, \mathbf{W}, \mathbf{A}, \mathbf{K}, \mathbf{C}_{\mathsf{in}} + \mathbf{E}_{\mathsf{in}}, (\mathbf{C} + \mathbf{E}, \mathbf{C}' + \mathbf{E}'), \mathbf{z}) + \mathbf{E}_{\mathsf{out}},$$

where $\|\mathbf{E}_{\mathsf{out}}\| \le m \cdot (2\|\mathbf{E}'\| + \|\mathbf{E}_{\mathsf{in}}\| + \|\mathbf{E}\|\|\mathbf{K}\|)$.

Proof. We have

$$\begin{aligned}\mathsf{rDec}_0\big(\mathbf{B},\mathbf{W},\mathbf{A},\mathbf{C}_{\mathsf{in}},\mathbf{C}',\mathbf{z}\big) &= \mathsf{diag}_L(\mathbf{C}')(\mathbf{I}_L\otimes\mathbf{G})^{-1}(\mathbf{A}) - \mathsf{diag}(\mathbf{z})\mathbf{C}' - \mathbf{C}_{\mathsf{in}}\mathbf{G}^{-1}(\mathbf{W})\\ &= \mathsf{diag}_L(\mathbf{S})(\mathbf{I}_L\otimes\mathbf{W})(\mathbf{I}_L\otimes\mathbf{G})^{-1}(\mathbf{A}) + (\mathbf{I}_L\otimes\mathbf{s})\mathbf{A}\\ &\quad - \mathsf{diag}(\mathbf{z})\mathbf{S}\mathbf{W} - \mathbf{z}^\top\cdot\mathbf{s}\mathbf{G}\\ &\quad - \mathbf{S}\mathbf{A}_0\mathbf{G}^{-1}(\mathbf{W}) + \mathsf{diag}(\mathbf{z})\mathbf{S}\mathbf{W}\\ &= [\![\mathbf{s},\mathbf{A},\mathbf{z}]\!] - \mathsf{diag}_L(\mathbf{S})([\![\mathbf{A}_0\mapsto\mathbf{A}]\!]_{\mathbf{W}}).\end{aligned}$$

The claim follows after noticing that $\mathsf{diag}_L(\mathbf{C})\cdot\mathbf{K} = \mathsf{diag}_L(\mathbf{S})([\![\mathbf{A}_0\mapsto\mathbf{A}]\!]_{\mathbf{W}})$ and that

$$\mathbf{E}_{\mathsf{out}} = \mathsf{diag}_L(\mathbf{E}')(\mathbf{I}_L\otimes\mathbf{G})^{-1}(\mathbf{A}) - \mathsf{diag}(\mathbf{z})\mathbf{E}' - \mathbf{E}_{\mathsf{in}}\mathbf{G}^{-1}(\mathbf{W}) + \mathsf{diag}_L(\mathbf{E})\mathbf{K}.$$

Lemma 4 (recoding dual-to-dual correctness). *Suppose*

$$\begin{aligned}\mathbf{C}_{\mathsf{in}} &= [\![\mathbf{S},\mathbf{A}_0,\mathbf{z}]\!]\\ (\mathbf{C},\mathbf{C}') &= [\![\mathbf{S}\mapsto\mathbf{S}']\!]_{\mathbf{B},\mathbf{W}}\\ \mathbf{B}\cdot\mathbf{K} &= [\![\mathbf{A}_0\mapsto\mathbf{A}_0']\!]_{\mathbf{W}}\end{aligned}$$

Then,

$$[\![\mathbf{S}',\mathbf{A}_0',\mathbf{z}]\!] = \mathsf{rDec}\big(\mathbf{B},\mathbf{W},\mathbf{A}_0',\mathbf{K},\mathbf{C}_{\mathsf{in}}+\mathbf{E}_{\mathsf{in}},(\mathbf{C}+\mathbf{E},\mathbf{C}'+\mathbf{E}'),\mathbf{z}\big) + \mathbf{E}_{\mathsf{out}}.$$

where $\|\mathbf{E}_{\mathsf{out}}\| \le m\cdot(2\|\mathbf{E}'\| + \|\mathbf{E}_{\mathsf{in}}\| + \|\mathbf{E}\|\|\mathbf{K}\|)$.

Proof. We have

$$\begin{aligned}\mathsf{rDec}_0\big(\mathbf{B},\mathbf{W},\mathbf{A},\mathbf{C}_{\mathsf{in}},\mathbf{C}',\mathbf{z}\big) &= \mathbf{C}'\mathbf{G}^{-1}(\mathbf{A}_0') - \mathsf{diag}(\mathbf{z})\mathbf{C}' - \mathbf{C}_{\mathsf{in}}\mathbf{G}^{-1}(\mathbf{W})\\ &= \mathbf{S}\mathbf{W}\mathbf{G}^{-1}(\mathbf{A}_0') + \mathbf{S}'\mathbf{A}_0'\\ &\quad - \mathsf{diag}(\mathbf{z})\mathbf{S}\mathbf{W} - \mathsf{diag}(\mathbf{z})\mathbf{S}'\mathbf{G}\\ &\quad - \mathbf{S}\mathbf{A}_0\mathbf{G}^{-1}(\mathbf{W}) + \mathsf{diag}(\mathbf{z})\mathbf{S}\mathbf{W}\\ &= [\![\mathbf{S}',\mathbf{A}_0',\mathbf{z}]\!] - \mathbf{S}([\![\mathbf{A}_0\mapsto\mathbf{A}_0']\!]_{\mathbf{W}})\end{aligned}$$

The claim follows after noticing that $\mathbf{C}\mathbf{K} = \mathbf{S}([\![\mathbf{A}_0\mapsto\mathbf{A}_0']\!]_{\mathbf{W}})$ and that

$$\mathbf{E}_{\mathsf{out}} = \mathbf{E}'\mathbf{G}^{-1}(\mathbf{A}_0') - \mathsf{diag}(\mathbf{z})\mathbf{E}' - \mathbf{E}_{\mathsf{in}}\mathbf{G}^{-1}(\mathbf{W}) + \mathbf{E}\mathbf{K}.$$

Lemma 5 (recoding encoding-to-dual correctness). *Suppose*

$$\begin{aligned}\mathbf{C}_{\mathsf{in}} &= [\![\mathbf{s},\mathbf{A},\mathbf{z}]\!]\\ (\mathbf{C},\mathbf{C}') &= [\![\mathbf{s}\mapsto\mathbf{S}]\!]_{\mathbf{B},\mathbf{W}'}\\ (\mathbf{I}_L\otimes\mathbf{B})\ \cdot\mathbf{K} &= [\![\mathbf{A}\mapsto\mathbf{A}_0]\!]_{\mathbf{W}'}\end{aligned}$$

Then,

$$[\![\mathbf{S},\mathbf{A}_0,\mathbf{z}]\!] = \mathsf{rDec}(\mathbf{B},\mathbf{W},\mathbf{A}_0,\mathbf{K},\mathbf{C}_{\mathsf{in}}+\mathbf{E}_{\mathsf{in}},(\mathbf{C}+\mathbf{E},\mathbf{C}'+\mathbf{E}'),\mathbf{z}) + \mathbf{E}_{\mathsf{out}},$$

where $\|\mathbf{E}_{\mathsf{out}}\| \le m\cdot(2\|\mathbf{E}'\| + \|\mathbf{E}_{\mathsf{in}}\| + \|\mathbf{E}\|\|\mathbf{K}\|)$.

Proof. We have

$$\begin{aligned}\mathsf{rDec}_0\big(\mathbf{B},\mathbf{W},\mathbf{A},\mathbf{C}_{\mathsf{in}},\mathbf{C}',\mathbf{z}\big) &= \mathbf{C}'\mathbf{G}^{-1}(\mathbf{A}_0) - \mathsf{diag}(\mathbf{z})\mathbf{C}' - \mathsf{diag}_L(\mathbf{C}_{\mathsf{in}})(\mathbf{I}_L\otimes\mathbf{G})^{-1}(\mathbf{W}')\\ &= (\mathbf{I}_L\otimes\mathbf{s})\mathbf{W}'\mathbf{G}^{-1}(\mathbf{A}_0) + \mathbf{S}\mathbf{A}_0\\ &\quad - \mathsf{diag}(\mathbf{z})(\mathbf{I}_L\otimes\mathbf{s})\mathbf{W}' - \mathsf{diag}(\mathbf{z})\mathbf{S}\mathbf{G}\\ &\quad - (\mathbf{I}_L\otimes\mathbf{s})\mathsf{diag}_L(\mathbf{A})(\mathbf{I}_L\otimes\mathbf{G})^{-1}(\mathbf{W}') + \mathsf{diag}(\mathbf{z})(\mathbf{I}_L\otimes\mathbf{s})\mathbf{W}'\\ &= [\![\mathbf{S},\mathbf{A}_0,\mathbf{z}]\!] - (\mathbf{I}_L\otimes\mathbf{s})([\![\mathbf{A}\mapsto\mathbf{A}_0]\!]_{\mathbf{W}'}).\end{aligned}$$

The claim follows after noticing that $(\mathbf{I}_L\otimes\mathbf{C})\cdot\mathbf{K} = (\mathbf{I}_L\otimes\mathbf{s})([\![\mathbf{A}\mapsto\mathbf{A}_0]\!]_{\mathbf{W}'})$ and that

$$\mathbf{E}_{\mathsf{out}} = \mathbf{E}'\mathbf{G}^{-1}(\mathbf{A}_0) - \mathsf{diag}(\mathbf{z})\mathbf{E}' - \mathsf{diag}_L(\mathbf{E}_{\mathsf{in}})(\mathbf{I}_L\otimes\mathbf{G})^{-1}(\mathbf{W}') + (\mathbf{I}_L\otimes\mathbf{E})\mathbf{K}.$$

5 ABE for Iterated Computation

We present our ABE for iterated computation (cf. Section 2.1). For simplicity, we assume the functions $\mathsf{read}, \mathsf{write}$ are fixed in advance, which is already sufficient for oblivious Turing machines.

Construction. Let the ABE $\Pi = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{KeyGen}, \mathsf{Dec})$ for iterated computation with locality L be defined as follows:

- $\mathsf{Setup}(1^\lambda, 1^L)$: Sample

$$
\begin{aligned}
&(\mathbf{B}_0, \mathbf{T}_{\mathbf{B}_0}), (\mathbf{B}_1, \mathbf{T}_{\mathbf{B}_1}), (\mathbf{B}_2, \mathbf{T}_{\mathbf{B}_2}), (\mathbf{B}_3, \mathbf{T}_{\mathbf{B}_3}), (\mathbf{B}_4, \mathbf{T}_{\mathbf{B}_4}) \leftarrow \mathsf{TrapGen}(1^n, 1^m, q),\\
&\mathbf{A}_{st}, \mathbf{W}_1, \mathbf{W}_2, \leftarrow \mathbb{Z}_q^{n\times m},\\
&\mathbf{W}_3 \leftarrow \mathbb{Z}_q^{Ln\times m},\\
&\mathbf{D}_{\mathsf{circ}} \leftarrow \mathbb{Z}_q^{n\times (n^2 m \lceil \log q\rceil^2 + 1)m}\\
&\mathbf{P} \leftarrow \mathbb{Z}_q^{n\times m}.
\end{aligned}
$$

  Set $\mathsf{mpk} = \big(\mathbf{B}_0, \mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3, \mathbf{B}_4, \mathbf{A}_{st}, \mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3, \mathbf{D}_{\mathsf{circ}}, \mathbf{P}\big)$, and $\mathsf{msk} = \big(\mathbf{T}_{\mathbf{B}_0}, \mathbf{T}_{\mathbf{B}_1}, \mathbf{T}_{\mathbf{B}_2}, \mathbf{T}_{\mathbf{B}_3}, \mathbf{T}_{\mathbf{B}_4}\big)$. Return $\big(\mathsf{mpk}, \mathsf{msk}\big)$.

- $\mathsf{Enc}(\mathsf{mpk}, (\mathbf{x}, 1^T), \boldsymbol{\mu} \in \{0,1\}^m)$: Set $\mathbf{z}_0 := [\mathbf{x} \| 0^{S-\ell}]$. Sample $\mathbf{S}_0 \leftarrow \mathcal{D}_{\mathbb{Z}^{S\times n}, \chi}$.[7] Then, for $t = 1, \ldots, T$, do the following
  - $L_t := |\mathsf{write}(t) \cup \mathsf{read}(t)|$.[8]
  - $\tilde{\mathbf{S}}_t \leftarrow \mathcal{D}_{\mathbb{Z}^{L_t\times n}, \chi}$,
  - $\mathbf{S}_t := \begin{cases} \mathbf{S}_t[\overline{\mathsf{write}(t) \cup \mathsf{read}(t)}] = \mathbf{S}_{t-1}[\overline{\mathsf{write}(t) \cup \mathsf{read}(t)}] \\ \mathbf{S}_t[\mathsf{write}(t) \cup \mathsf{read}(t)] = \tilde{\mathbf{S}}_t \end{cases} \in \mathbb{Z}^{S\times n}$

  Next, sample

$$
\begin{aligned}
&\mathbf{s}'_{t-1} \leftarrow \mathcal{D}_{\mathbb{Z}^n, \chi},\\
&\mathbf{E}_0 \leftarrow \mathcal{D}_{\mathbb{Z}^{S\times m}, \chi'''}, && \mathbf{E}'_0 \leftarrow \mathcal{D}_{\mathbb{Z}^{S\times m}, \chi'''},\\
&\mathbf{E}_{t,1} \leftarrow \mathcal{D}_{\mathbb{Z}^{L\times m}, \chi'''}, && \mathbf{E}'_{t,1} \leftarrow \mathcal{D}_{\mathbb{Z}^{L\times m}, \chi'''},\\
&\mathbf{E}_{t,2} \leftarrow \mathcal{D}_{\mathbb{Z}^{(L_t-L)\times m}, \chi'''}, && \mathbf{E}'_{t,2} \leftarrow \mathcal{D}_{\mathbb{Z}^{(L_t-L)\times m}, \chi'''},\\
&\mathbf{E}_{t,3} \leftarrow \mathcal{D}_{\mathbb{Z}^m, \chi'''}, && \mathbf{E}'_{t,3} \leftarrow \mathcal{D}_{\mathbb{Z}^{L\times m}, \chi'''},\\
&\mathbf{E}_{T,4} \leftarrow \mathcal{D}_{\mathbb{Z}^m, \chi'''}, && \mathbf{E}'_{T,4} \leftarrow \mathcal{D}_{\mathbb{Z}^m, \chi'''},
\end{aligned}
$$

  and compute

$$
\begin{aligned}
&\mathbf{C}_0 := \mathbf{S}_0\mathbf{B}_0 + \mathbf{E}_0,\\
&\mathbf{C}'_0 := [\![\mathbf{S}_0, \mathbf{A}_{st}, \mathbf{z}_0]\!] + \mathbf{E}'_0\\
&\left.\begin{aligned}
&(\mathbf{C}_{t,1}, \mathbf{C}'_{t,1}) := [\![\mathbf{S}_{t-1}[\mathsf{read}(t)] \mapsto \mathbf{s}'_{t-1}]\!]_{\mathbf{B}_1, \mathbf{W}_1} + (\mathbf{E}_{t,1}, \mathbf{E}'_{t,1})\\
&(\mathbf{C}_{t,2}, \mathbf{C}'_{t,2}) := [\![\mathbf{S}_{t-1}[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)] \mapsto \mathbf{S}_t[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)]]\!]_{\mathbf{B}_2, \mathbf{W}_2} + (\mathbf{E}_{t,2}, \mathbf{E}'_{t,2})\\
&(\mathbf{C}_{t,3}, \mathbf{C}'_{t,3}) := [\![\mathbf{s}'_{t-1} \mapsto \mathbf{S}_t[\mathsf{write}(t)]]\!]_{\mathbf{B}_3, \mathbf{W}_3} + (\mathbf{E}_{t,3}, \mathbf{E}'_{t,3})\\
&(\hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t) \leftarrow \odot[\![\mathbf{s}'_{t-1}, \mathbf{D}_{\mathsf{circ}}]\!]\\
&(\mathring{\mathbf{C}}_t, \mathring{\mathbf{C}}'_t) \leftarrow \odot[\![\mathbf{S}_{t-1}[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)], \mathbf{D}_{\mathsf{circ}}]\!]
\end{aligned}\right\} t \in [T]\\
&\mathbf{C}_{T,4} := \mathbf{S}_T[1]\mathbf{B}_4 + \mathbf{E}_{T,4}\\
&\mathbf{C}'_{T,4} := \mathbf{S}_T[1]\mathbf{P} + \boldsymbol{\mu} \cdot \lfloor q/2 \rfloor + \mathbf{E}'_{T,4}
\end{aligned}
$$

  Output $\mathsf{ct}_{\mathbf{x}} := \big(\mathbf{C}_0, \mathbf{C}'_0, \{\mathbf{C}_{t,1}, \mathbf{C}'_{t,1}, \mathbf{C}_{t,2}, \mathbf{C}'_{t,2}, \mathbf{C}_{t,3}, \mathbf{C}'_{t,3}, \hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t, \mathring{\mathbf{C}}_t, \mathring{\mathbf{C}}'_t\}_{t=1,\ldots,T}, \mathbf{C}_{T,4}, \mathbf{C}'_{T,4}\big)$.

[7] The last coordinate of each row is fixed to -1. The same applies whenever a new secret is sampled.
[8] Since $|\mathsf{write}(t)| = |\mathsf{read}(t)| = L$, one has $L_t \le 2L$.

– $\mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, f)$: Sample $\mathbf{A}_0 \leftarrow \mathbb{Z}_q^{n\times m}$, $\mathbf{D} \leftarrow \mathbb{Z}_q^{Ln\times m}$, $\mathbf{R} \leftarrow \mathcal{D}_{\mathbb{Z}^{m\times m},\tau}$. Compute $\mathbf{D}_f = \mathsf{UEvalF}(\mathbf{D}, \mathbf{D}_{\mathsf{circ}}, f)$, $\mathbf{A}_{0,\mathsf{id}} = \mathsf{UEvalF}(\mathbf{A}_0, \mathbf{D}_{\mathsf{circ}}, \mathsf{id})$, where $\mathsf{id} : \{0,1\} \to \{0,1\}$ denotes the identity map. Then, sample

$$\begin{aligned}
&\mathbf{K}_0 \leftarrow \mathsf{SamplePre}\big(\mathbf{B}_0, \mathbf{T}_{\mathbf{B}_0}, \mathbf{A}_0 - \mathbf{A}_{st}, \tau\big),\\
&\mathbf{K}_1 \leftarrow \mathsf{SamplePre}\big(\mathbf{I}_L \otimes \mathbf{B}_1, \mathbf{I}_L \otimes \mathbf{T}_{\mathbf{B}_1}, [\![\mathbf{A}_0 \mapsto \mathbf{D}]\!]_{\mathbf{W}_1}, \tau\big),\\
&\mathbf{K}_2 \leftarrow \mathsf{SamplePre}\big(\mathbf{B}_2, \mathbf{T}_{\mathbf{B}_2}, [\![\mathbf{A}_{0,\mathsf{id}} \mapsto \mathbf{A}_0]\!]_{\mathbf{W}_2}, \tau\big),\\
&\mathbf{K}_3 \leftarrow \mathsf{SamplePre}\big(\mathbf{I}_L \otimes \mathbf{B}_3, \mathbf{I}_L \otimes \mathbf{T}_{\mathbf{B}_3}, [\![\mathbf{D}_f \mapsto \mathbf{A}_0]\!]_{\mathbf{W}_3}, \tau\big),\\
&\mathbf{K}_4 \leftarrow \mathsf{SamplePre}\big(\mathbf{B}_4, \mathbf{T}_{\mathbf{B}_4}, \mathbf{P} + \mathbf{A}_0\mathbf{R}, \tau\big).
\end{aligned}$$

Output $\mathsf{sk}_f = \big(\mathbf{A}_0, \mathbf{D}, \mathbf{K}_0, \mathbf{K}_1, \mathbf{K}_2, \mathbf{K}_3, \mathbf{K}_4, \mathbf{R}\big)$.

– $\mathsf{Dec}(\mathsf{mpk}, \mathsf{sk}_f, f, \mathsf{ct}_{\mathbf{x}}, (\mathbf{x}, 1^T))$: Parse

$$\begin{aligned}
\mathsf{mpk} &= \big(\mathbf{B}_0, \mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3, \mathbf{B}_4, \mathbf{A}_{st}, \mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3, \mathbf{D}_{\mathsf{circ}}, \mathbf{P}\big)\\
\mathsf{ct}_{\mathbf{x}} &= \big(\mathbf{C}_0, \mathbf{C}'_0, \{\mathbf{C}_{t,1}, \mathbf{C}'_{t,1}, \mathbf{C}_{t,2}, \mathbf{C}'_{t,2}, \mathbf{C}_{t,3}, \mathbf{C}'_{t,3}, \hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t, \mathring{\mathbf{C}}_t, \mathring{\mathbf{C}}'_t\}_{t=1,\ldots,T}, \mathbf{C}_{T,4}, \mathbf{C}'_{T,4}\big)\\
\mathsf{sk}_f &= \big(\mathbf{A}_0, \mathbf{D}, \mathbf{K}_0, \mathbf{K}_1, \mathbf{K}_2, \mathbf{K}_3, \mathbf{K}_4, \mathbf{R}\big)
\end{aligned}$$

and set $\mathbf{z}_0 := [\mathbf{x} \| 0^{S-\ell}]$. Let

$$\tilde{\mathbf{C}}_0 := \mathbf{C}_0 \cdot \mathbf{K}_0 + \mathbf{C}'_0$$

For each $t \in [T]$, do the following

- Obtain $\tilde{\mathbf{C}}_{t-1}[\mathsf{read}(t)], \mathbf{z}_{t-1}[\mathsf{read}(t)]$ from $\tilde{\mathbf{C}}_{t-1}, \mathbf{z}_{t-1}$ respectively.
- Compute

$$\begin{aligned}
\tilde{\mathbf{C}}_t[\overline{\mathsf{write}(t) \cup \mathsf{read}(t)}] &:= \tilde{\mathbf{C}}_{t-1}[\overline{\mathsf{write}(t) \cup \mathsf{read}(t)}],^{9}\\
\tilde{\mathbf{C}}_{t,1} &:= \mathsf{rDec}\big(\mathbf{B}_1, \mathbf{W}_1, \mathbf{D}, \mathbf{K}_1, \tilde{\mathbf{C}}_{t-1}[\mathsf{read}(t)], (\mathbf{C}_{t,1}, \mathbf{C}'_{t,1}), \mathbf{z}_{t-1}[\mathsf{read}(t)]\big),\\
\tilde{\mathbf{C}}_{t,2} &:= \mathsf{UEvalFX}\big(\mathbf{A}_0, \mathbf{D}_{\mathsf{circ}}, \tilde{\mathbf{C}}_{t-1}[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)], (\mathring{\mathbf{C}}_t, \mathring{\mathbf{C}}'_t), \mathsf{id}, \mathbf{z}_{t-1}[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)]\big),\\
\tilde{\mathbf{C}}_t[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)] &:= \mathsf{rDec}\big(\mathbf{B}_2, \mathbf{W}_2, \mathbf{A}_0, \mathbf{K}_2, \tilde{\mathbf{C}}_{t,2}, (\mathbf{C}_{t,2}, \mathbf{C}'_{t,2}), \mathbf{z}_{t-1}[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)]\big),\\
\tilde{\mathbf{C}}_{t,3} &:= \mathsf{UEvalFX}\big(\mathbf{D}, \mathbf{D}_{\mathsf{circ}}, \tilde{\mathbf{C}}_{t,1}, (\hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t), f, \mathbf{z}_{t-1}[\mathsf{read}(t)]\big),\\
\tilde{\mathbf{C}}_t[\mathsf{write}(t)] &:= \mathsf{rDec}\big(\mathbf{B}_3, \mathbf{W}_3, \mathbf{A}_0, \mathbf{K}_3, \tilde{\mathbf{C}}_{t,3}, (\mathbf{C}_{t,3}, \mathbf{C}'_{t,3}), f(\mathbf{z}_{t-1}[\mathsf{read}(t)])\big),\\
\mathbf{z}_t[\overline{\mathsf{write}(t)}] &:= \mathbf{z}_{t-1}[\overline{\mathsf{write}(t)}],^{10}\\
\mathbf{z}_t[\mathsf{write}(t)] &:= f(\mathbf{z}_{t-1}[\mathsf{read}(t)]).
\end{aligned}$$

Given $\tilde{\mathbf{C}}_T$, output

$$\Big\lfloor \tfrac{2}{q} \cdot \big(\mathbf{C}'_{T,4} - \mathbf{C}_{T,4} \cdot \mathbf{K}_4 + \tilde{\mathbf{C}}_T[1] \cdot \mathbf{R}\big) \Big\rceil \in \{0,1\}^m.$$

5.1 Setting parameters

We have 5 gaussian parameters:

$$\underbrace{\overset{\approx \|\mathbf{s}'_t\|, \|\mathbf{S}_t\|}{\hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t, \mathring{\mathbf{C}}_t, \mathring{\mathbf{C}}'_t}}_{\chi} \le \underbrace{\overset{\approx (\mathbf{C}_{t,i}, \mathbf{C}'_{t,i})_{i\in\{0,1,2,3,4\}}}{\text{security}}}_{\chi_{\mathsf{pre}}} \le \underbrace{\overset{\approx \mathbf{Y}_0, \mathbf{Y}_{t,2}, \mathbf{Y}_{t,3}}{\chi'}}_{} \le \underbrace{\overset{\approx \mathbf{Y}_{t,1}, \mathbf{Y}_{T,4},}{\chi''}}_{} \le \underbrace{\overset{\approx (\mathbf{C}_{t,i}, \mathbf{C}'_{t,i})_{i\in\{0,1,2,3,4\}}}{\text{scheme}}}_{\chi'''}.$$

[9] We are only updating notation. The decryptor does not actually do any computation for indexes in $\overline{\mathsf{write}(t) \cup \mathsf{read}(t)}$.
[10] Similarly, we are only updating notation.

where $\mathbf{Y}_0, \mathbf{Y}_{t,1}, \mathbf{Y}_{t,2}, \mathbf{Y}_{t,3}, \mathbf{Y}_{T,4}$ and $\chi_{\mathsf{pre}}, \chi', \chi''$ are introduced in the security proof. The parameter requirements can be compactly specified as:

$$
\begin{aligned}
m &\geq O(n \log q) && \text{trapdoor generation (Lemma 11)} \\
2^{n^\delta} &\geq q/\chi_{\mathsf{pre}}, q/\chi, \qquad \chi_{\mathsf{pre}}, \chi \geq O(n+\lambda) && \mathsf{cssLWE}_{n,q,\chi,\chi_{\mathsf{pre}}} \text{ hardness } (\mathsf{H}'_{t-1} \approx_c \mathsf{H}_t, \mathsf{H}_t \approx_c \mathsf{H}'_t) \\
\chi'' &\geq \lambda^{\omega(1)} \cdot m(\chi'\tau + \chi_{\mathsf{pre}}) && \text{noise flooding } (\mathsf{H}_{\mathsf{pre}} \iff \mathsf{H}_{\mathsf{corr}}, \text{Eq. (16)}) \\
\chi' &\geq \lambda^{\omega(1)} \cdot m(B_{\mathsf{HLL}} 2^{-\lambda} + \chi_{\mathsf{pre}}) && \text{noise flooding } (\mathsf{H}_{\mathsf{pre}} \iff \mathsf{H}_{\mathsf{corr}}, \text{Eq. (17)}) \\
\chi''' &\geq \lambda^{\omega(1)} \cdot \max\{\chi', \chi''\} && \text{evasive LWE} \\
\tau &\geq O(\sqrt{n} \log q \log n) && \text{trapdoor generation (Lemma 11)} \\
B_{\mathsf{HLL}} &\geq \max\{\chi''', \ 3m^2\chi'''\tau, \ \chi 2^\lambda\} && \text{correctness (Lemma 2)} \\
q &\geq 12m^2\chi'''\tau && \text{correctness}
\end{aligned}
$$

Bounding $\mathsf{poly}(m, \lambda)$ and $\lambda^{\omega(1)}$ terms by $2^{\lambda/8}$, we can set

$$
\begin{aligned}
&m = O(n \log q), \\
&\chi = 2^{\lambda/8}, && \chi''' = 2^\lambda, \\
&\chi_{\mathsf{pre}} = 2^{2\lambda/8}, && \chi' = 2^{4\lambda/8} && \chi'' = 2^{7\lambda/8}, \\
&\tau = 2^{\lambda/8}, && q = 2^{11\lambda/8} && B_{\mathsf{HLL}} = 2^{10\lambda/8}, \\
&n = O(\log q)^{1/\delta} = O(\lambda)^{1/\delta}.
\end{aligned}
\tag{2}
$$

5.2 Efficiency

Our ABE scheme achieves

$$|\mathsf{mpk}| = O(L), \quad |\mathsf{ct}| = O(S + T \cdot L), \quad |\mathsf{sk}| = O(L).$$

and

$$\mathsf{Time}(\mathsf{Enc}) = O(S + T \cdot L), \quad \mathsf{Time}(\mathsf{KeyGen}) = O(L + \mathsf{size}(f)), \quad \mathsf{Time}(\mathsf{Dec}) = O(S + T \cdot \mathsf{size}(f)).$$

where $O(\cdot)$ hides factors polynomial in λ. In more detail, we have

$$
\begin{aligned}
\mathsf{mpk}: &\begin{cases} |\mathbf{A}_{st}|, |\mathbf{B}_0|, |\mathbf{B}_1|, |\mathbf{B}_2|, |\mathbf{B}_3|, |\mathbf{B}_4|, |\mathbf{W}_1|, |\mathbf{W}_2|, |\mathbf{D}_{\mathsf{circ}}|, |\mathbf{P}| = O(1) \\ |\mathbf{W}_3| = O(L), \end{cases} \\
\mathsf{ct}: &\begin{cases} |\mathbf{C}_0|, |\mathbf{C}'_0| = O(S), \\ |\mathbf{C}_{t,1}|, |\mathbf{C}'_{t,1}|, |\mathbf{C}_{t,2}|, |\mathbf{C}'_{t,2}|, |\mathbf{C}'_{t,3}|, |\mathring{\mathbf{C}}_t|, |\mathring{\mathbf{C}}'_t| = O(L), & \text{for } t \in [T] \\ |\mathbf{C}_{t,3}|, |\hat{\mathbf{C}}_t|, |\hat{\mathbf{C}}'_t| = O(1), & \text{for } t \in [T] \\ |\mathbf{C}_{T,4}|, |\mathbf{C}'_{T,4}| = O(1) \end{cases} \\
\mathsf{sk}: &\begin{cases} |\mathbf{K}_0|, |\mathbf{K}_2|, |\mathbf{K}_4| = O(1), \\ |\mathbf{K}_1|, |\mathbf{K}_3| = O(L). \end{cases}
\end{aligned}
$$

Next, we elaborate on the running times:

- Enc:
  - sampling $\mathbf{S}_0$ and computing $\mathbf{C}_0, \mathbf{C}'_0$ takes $O(S)$ time;
  - for each $t \in [T]$: sampling $\tilde{S}_t$ and computing $(\mathbf{C}_{t,1}, \mathbf{C}'_{t,1}), (\mathbf{C}_{t,2}, \mathbf{C}'_{t,2}), (\mathbf{C}_{t,3}, \mathbf{C}'_{t,3}), (\mathring{\mathbf{C}}_t, \mathring{\mathbf{C}}'_t)$ takes $O(L)$ time, while sampling $\mathbf{s}'_{t-1}$ and computing $(\hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t)$ takes $O(1)$ time;
  - computing $\mathbf{C}_{T,4}, \mathbf{C}'_{T,4}$ takes $O(1)$ time.

– Dec:
- initializes a matrix $\tilde{\mathbf{C}}$ to be $\tilde{\mathbf{C}}_0$, which takes $O(S)$ time;
- for each $t \in [T]$: updates $\tilde{\mathbf{C}}$ from $\tilde{\mathbf{C}}_{t-1}$ to $\tilde{\mathbf{C}}_t$. Computing $\tilde{\mathbf{C}}_{t,1}, \tilde{\mathbf{C}}_{t,2}, \tilde{\mathbf{C}}_t[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)], \tilde{\mathbf{C}}_{t,3}, \tilde{\mathbf{C}}_t[\mathsf{write}(t)], \mathbf{z}_t[\mathsf{write}(t)]$ takes $O(L)$ time (there is no need to update the entries corresponding to $\overline{\mathsf{write}(t) \cup \mathsf{read}(t)}$);
- the final computation involving $\tilde{\mathbf{C}}_T$ takes $O(1)$ time.

– KeyGen:
- computing $\mathbf{D}_f$ takes $\mathsf{size}(f)$ time, computing $\mathbf{A}_{0,\mathsf{id}}$ takes $O(1)$ time;
- sampling $\mathbf{K}_0, \mathbf{K}_2, \mathbf{K}_4$ takes $O(1)$ time;
- sampling $\mathbf{K}_1, \mathbf{K}_3$ takes $O(L)$ time. While matrices and trapdoor input to the sampling algorithm have size $O(L^2)$, their representation only requires $O(1)$ space and the sampling procedure can be implemented in $O(L)$ time by letting

$$\mathbf{K}_1 = (\mathbf{K}_{1,i})_{i\in[L]} \quad \text{where} \quad \mathbf{K}_{1,i} \leftarrow \mathsf{SamplePre}\big(\mathbf{B}_1, \mathbf{T}_{\mathbf{B}_1}, [\![\mathbf{A}_0 \mapsto \mathbf{D}_i]\!]_{\mathbf{W}_1}, \tau\big)$$
$$\mathbf{K}_3 = (\mathbf{K}_{3,i})_{i\in[L]} \quad \text{where} \quad \mathbf{K}_{3,i} \leftarrow \mathsf{SamplePre}\big(\mathbf{B}_1, \mathbf{T}_{\mathbf{B}_1}, [\![\mathbf{D}_{f,i} \mapsto \mathbf{A}_0]\!]_{\mathbf{W}_{3,i}}, \tau\big)$$

where $\mathbf{D}_i, \mathbf{W}_{3,i}, \mathbf{D}_{f,i} \in \mathbb{Z}_q^{n\times m}$ denote the i-th matrix block of $\mathbf{D}, \mathbf{W}_3, \mathbf{D}_f$ respectively.

Theorem 1 (Correctness). *Let Π be the ABE for iterated computation with locality L construction described in Section 5, with parameters as in Eq. (2). Then, Π is correct.*

Proof. Recall that from the decryption algorithm, we have that

$$\begin{aligned}
\tilde{\mathbf{C}}_t[\overline{\mathsf{write}(t) \cup \mathsf{read}(t)}] &:= \tilde{\mathbf{C}}_{t-1}[\overline{\mathsf{write}(t) \cup \mathsf{read}(t)}],\\
\tilde{\mathbf{C}}_{t,1} &:= \mathsf{rDec}\big(\mathbf{B}_1, \mathbf{W}_1, \mathbf{D}, \mathbf{K}_1, \tilde{\mathbf{C}}_{t-1}[\mathsf{read}(t)], (\mathbf{C}_{t,1}, \mathbf{C}'_{t,1}), \mathbf{z}_{t-1}[\mathsf{read}(t)]\big),\\
\tilde{\mathbf{C}}_{t,2} &:= \mathsf{UEvalFX}\big(\mathbf{A}, \mathbf{D}_{\mathsf{circ}}, \tilde{\mathbf{C}}_{t-1}[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)], (\mathring{\mathbf{C}}_t, \mathring{\mathbf{C}}'_t), \mathsf{id}, \mathbf{z}_{t-1}[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)]\big),\\
\tilde{\mathbf{C}}_t[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)] &:= \mathsf{rDec}\big(\mathbf{B}_2, \mathbf{W}_2, \mathbf{A}_0, \mathbf{K}_2, \tilde{\mathbf{C}}_{t,2}, (\mathbf{C}_{t,2}, \mathbf{C}'_{t,2}), \mathbf{z}_{t-1}[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)]\big),\\
\tilde{\mathbf{C}}_{t,3} &:= \mathsf{UEvalFX}\big(\mathbf{D}, \mathbf{D}_{\mathsf{circ}}, \tilde{\mathbf{C}}_{t,1}, (\hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t), f, \mathbf{z}_{t-1}[\mathsf{read}(t)]\big),\\
\tilde{\mathbf{C}}_t[\mathsf{write}(t)] &:= \mathsf{rDec}\big(\mathbf{B}_3, \mathbf{W}_3, \mathbf{A}_0, \mathbf{K}_3, \tilde{\mathbf{C}}_{t,3}, (\mathbf{C}_{t,3}, \mathbf{C}'_{t,3}), f(\mathbf{z}_{t-1}[\mathsf{read}(t)])\big)
\end{aligned}$$

and

$$\boldsymbol{\mu}' := \left\lfloor \tfrac{2}{q} \cdot \big(\mathbf{C}'_{T,4} - \mathbf{C}_{T,4} \cdot \mathbf{K}_4 + \tilde{\mathbf{C}}_T[1] \cdot \mathbf{R}\big) \right\rceil.$$

By definition, for any accepting attribute $\mathbf{x}$, one has that $\mathbf{z}_T[1] = 0$, and therefore $\tilde{\mathbf{C}}_T[1] \approx \mathbf{S}_T[1](\mathbf{A}_0 - 0 \cdot \mathbf{G})$. Then, the statement follows from the fact that

$$\begin{aligned}
\mathbf{C}'_{T,4} - \mathbf{C}_{T,4} \cdot \mathbf{K}_4 + \tilde{\mathbf{C}}_T[1] \cdot \mathbf{R} \approx\ & \mathbf{S}_T[1]\mathbf{P} + \boldsymbol{\mu} \cdot \lfloor q/2 \rfloor\\
& - \mathbf{S}_T[1](\mathbf{P} + \mathbf{A}_0\mathbf{R})\\
& + \mathbf{S}_T[1](\mathbf{A}_0\mathbf{R} - 0 \cdot \mathbf{G}\mathbf{R})\\
=\ & \boldsymbol{\mu} \cdot \lfloor q/2 \rfloor
\end{aligned}$$

and that

$$q \geq 12m^2\chi'''\tau^2 \geq 4\left(\|\mathbf{E}'_{T,4}\| + \|\mathbf{E}_{T,4}\mathbf{K}_4\| + \|\tilde{\mathbf{E}}_T[1]\mathbf{R}\|\right),$$

where $\tilde{\mathbf{E}}_T[1]$ denotes the error term of $\tilde{\mathbf{C}}_T[1]$, and we have bounded its norm by $m\chi'''\tau$ thanks to the following claim.

Claim. The quantities computed during decryption satisfy:

$$\tilde{\mathbf{C}}_0 = [\![\mathbf{S}_0, \mathbf{A}_0, \mathbf{z}_0]\!] + \tilde{\mathbf{E}}_0, \tag{3}$$
$$\tilde{\mathbf{C}}_{t,1} = [\![\mathbf{s}'_{t-1}, \mathbf{D}, \mathbf{z}_{t-1}[\mathsf{read}(t)]]\!] + \tilde{\mathbf{E}}_{t,1} \tag{4}$$
$$\tilde{\mathbf{C}}_{t,2} = [\![\mathbf{S}_{t-1}[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)], \mathbf{A}_{0,\mathsf{id}}, \mathbf{z}_{t-1}[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)]]\!] + \tilde{\mathbf{E}}_{t,2} \tag{5}$$
$$\tilde{\mathbf{C}}_{t,3} = [\![\mathbf{s}'_{t-1}, \mathbf{D}_f, f(\mathbf{z}_{t-1}[\mathsf{read}(t)])]\!] + \tilde{\mathbf{E}}_{t,3} \tag{6}$$
$$\tilde{\mathbf{C}}_t = [\![\mathbf{S}_t, \mathbf{A}_0, \mathbf{z}_t]\!] + \tilde{\mathbf{E}}_t \tag{7}$$

where, setting $B_t \coloneqq m\chi'''\tau$, the error terms have bounds

$$\begin{aligned}
\|\tilde{\mathbf{E}}_0\| &\le m\chi'''\tau,\\
\|\tilde{\mathbf{E}}_{t,1}\| &\le m(\chi''' + B_{t-1} + \chi'''\tau) \le 3m^2\chi\tau,\\
\|\tilde{\mathbf{E}}_{t,2}\| &\le 3m^2\chi'''\tau \cdot 2^{-\lambda},\\
\|\tilde{\mathbf{E}}_{t,3}\| &\le 3m^2\chi'''\tau \cdot 2^{-\lambda},\\
\|\tilde{\mathbf{E}}_t\| &\le m\chi'''\tau = B_t
\end{aligned}$$

We have

$$[\![\mathbf{S}_0, \mathbf{A}_0, \mathbf{z}_0]\!] + \mathbf{E}_0\mathbf{K}_0 + \mathbf{E}'_0 = \underbrace{(\mathbf{S}_0\mathbf{B}_0 + \mathbf{E}_0)}_{\mathbf{C}_0} \cdot \underbrace{\mathbf{B}_0^{-1}(\mathbf{A}_0 - \mathbf{A}_{st})}_{\mathbf{K}_0} + \underbrace{[\![\mathbf{S}_0, \mathbf{A}_{st}, \mathbf{z}_0]\!] + \mathbf{E}'_0}_{\mathbf{C}'_0}$$

Thus, Eq. (3) follows with $\|\tilde{\mathbf{E}}_0\| = \|\mathbf{E}_0\mathbf{K}_0 + \mathbf{E}'_0\| \le O(m\chi'''\tau)$ as required.

The proof proceeds then by induction on t:

- from $\tilde{\mathbf{C}}_{t-1}[\mathsf{read}(t)]$ to $\tilde{\mathbf{C}}_{t,1}$: starting from Eq. (7) for $t-1$, invoking Lemma 3, we get Eq. (4) for t with error bound

$$\|\tilde{\mathbf{E}}_{t,1}\| \le m(\chi''' + B_{t-1} + \chi'''\tau) \le m(\chi''' + m\chi'''\tau + \chi'''\tau) \le 3m^2\chi'''\tau.$$

- from $\tilde{\mathbf{C}}_{t-1}[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)]$ to $\tilde{\mathbf{C}}_{t,2}$: starting from Eq. (7) for $t-1$, invoking Lemma 2 with $B_{\mathsf{HLL}} = 3m^2\chi'''\tau$ (which requires $\|\mathbf{S}_{t-1}[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)]\|, \|\mathring{\mathbf{E}}_t\|2^\lambda, \|\mathring{\mathbf{E}}'_t\|2\lambda \le B_{\mathsf{HLL}}$, where $\mathring{\mathbf{E}}_t, \mathring{\mathbf{E}}'_t$ denote the error terms in $\mathring{\mathbf{C}}_t, \mathring{\mathbf{C}}'_t$), we get Eqs. (5) and (6) for t with error bound

$$\|\tilde{\mathbf{E}}_{t,2}\| \le B_{\mathsf{HLL}}2^{-\lambda} = 3m^2\chi'''\tau \cdot 2^{-\lambda}.$$

- from $\tilde{\mathbf{C}}_{t,1}$ to $\tilde{\mathbf{C}}_{t,3}$: starting from Eq. (4) for t, invoking Lemma 2 with $B_{\mathsf{HLL}} = 3m^2\chi'''\tau$ (which requires $\|\mathbf{s}'_{t-1}\|$, $\|\hat{\mathbf{E}}_t\|2^\lambda, \|\hat{\mathbf{E}}'_t\|2^\lambda \le B_{\mathsf{HLL}}$, where $\hat{\mathbf{E}}_t, \hat{\mathbf{E}}'_t$ denote the error terms in $\hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t$), we get Eq. (6) for t with error bound

$$\|\tilde{\mathbf{E}}_{t,3}\| \le B_{\mathsf{HLL}}2^{-\lambda} = 3m^2\chi'''\tau \cdot 2^{-\lambda}.$$

- from $\tilde{\mathbf{C}}_{t-1}[\overline{\mathsf{write}(t) \cup \mathsf{read}(t)}]$, $\tilde{\mathbf{C}}_{t,2}, \tilde{\mathbf{C}}_{t,3}$ to $\tilde{\mathbf{C}}_t$: starting from Eq. (7) for $t-1$ and Eqs. (5) and (6) for t, invoking Lemmas 4 and 5, we obtain Eq. (7) for t with error bound

$$\begin{aligned}
\|\tilde{\mathbf{E}}_t\| &\le \max\{\tilde{\mathbf{E}}_t[\overline{\mathsf{write}(t) \cup \mathsf{read}(t)}], \tilde{\mathbf{E}}_t[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)], \tilde{\mathbf{E}}_t[\mathsf{write}(t)]\}\\
&\le \max\{m\chi'''\tau, 3m^3\chi'''\tau \cdot 2^{-\lambda}, 3m^3\chi'''\tau \cdot 2^{-\lambda}\}\\
&\le m\chi'''\tau.
\end{aligned}$$

where in the last inequality we have used that $3m^3 \le 2^\lambda$. This concludes the proof. □

6 Security Proof

Using evasive LWE. Recall that, we use $\cup$ to denote "vertical concatenation" when applied to vectors/matrices of width n.

Suppose the ABE adversary $\mathcal{A}$ with randomness $\mathsf{coins}_{\mathcal{A}}$ queries $(\mathbf{x}, 1^T)$ and $f_1, \ldots, f_Q$ such that $f_1(\mathbf{x}) = \cdots = f_Q(\mathbf{x}) = 1$.

We invoke the public-coin circular small-secret evasive LWE hardness assumption with sampler Samp outputting:

$$\begin{gathered}\mathbf{A}' \in \mathbb{Z}_q^{Jn \times m'}, (\mathbf{P}_\gamma)_{\gamma=0,\ldots,4} \in \mathbb{Z}_q^{n \times t_0 + \ldots + t_4}, (\Gamma_\gamma)_{\gamma=0,\ldots 4} \subseteq [J], \mathsf{aux} \in \{0,1\}^* \\ \chi, \chi_{\mathsf{pre}}, \chi_{\mathsf{post}}, (\chi_\gamma)_{\gamma=0,\ldots,4}\end{gathered} \tag{8}$$

where

$$\mathsf{aux} = (\mathsf{coins}_{\mathcal{A}}, f_1, \ldots, f_Q, \mathbf{x}, \mathbf{A}_{st}, \mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3, \mathbf{D}_{\mathsf{circ}}, \mathbf{P}, (\mathbf{A}_0^{(j)}, \mathbf{D}^{(j)}, \mathbf{R}^{(j)})_{j \in [Q]}), \tag{9}$$

the sets $(\Gamma_\gamma)_{\gamma=0,\ldots 4} \subseteq [J]$, for $J = S + \sum_{t=1}^{T} L_t + T$, and the matrix $\mathbf{A}' := (\mathbf{A}'_0 \mid \mathbf{A}'_1 \mid \mathbf{A}'_2 \mid \mathbf{A}'_3 \mid \mathbf{A}'_4)$ are defined via the following constraints

$$(\Gamma_i)_{i=0,\ldots,4} = \begin{cases} \mathbf{S}^\dagger[\Gamma_0] = \mathbf{S}_0 \\ \mathbf{S}^\dagger[\Gamma_1] = (\mathbf{S}_0[\mathsf{read}(1)] \cup \ldots \cup \mathbf{S}_{T-1}[\mathsf{read}(T)]) \\ \mathbf{S}^\dagger[\Gamma_2] = (\mathbf{S}_0[\overline{\mathsf{write}(1)} \cap \mathsf{read}(1)] \cup \ldots \cup \mathbf{S}_{T-1}[\overline{\mathsf{write}(T)} \cap \mathsf{read}(T)]) \\ \mathbf{S}^\dagger[\Gamma_3] = (\mathbf{s}'_0 \cup \ldots \cup \mathbf{s}'_{T-1}) \\ \mathbf{S}^\dagger[\Gamma_4] = (\mathbf{S}_T[1]) \end{cases}$$

and

$$\begin{aligned} \mathsf{flat}(\mathbf{S}^\dagger)\mathbf{A}'_0 &= \mathsf{flat}(\mathbf{S}_0\mathbf{A}_{st} + \mathsf{diag}(\mathbf{z}_0)\mathbf{S}_0\mathbf{G}), \\ \mathsf{flat}(\mathbf{S}^\dagger)\mathbf{A}'_1 &= \big(\mathsf{flat}(\mathbf{S}_{t-1}[\mathsf{read}(t)]\mathbf{W}_1 + \mathbf{1}_L^\top \cdot \mathbf{s}'_{t-1}\mathbf{G})\big)_{t \in [T]}, \\ \mathsf{flat}(\mathbf{S}^\dagger)\mathbf{A}'_2 &= \big(\mathsf{flat}(\mathbf{S}_{t-1}[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)]\mathbf{W}_2 + \mathbf{S}_t[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)]\mathbf{G})\big)_{t \in [T]}, \\ \mathsf{flat}(\mathbf{S}^\dagger)\mathbf{A}'_3 &= \big(\mathsf{flat}((\mathbf{I}_L \otimes \mathbf{s}'_{t-1})\mathbf{W}_3 + \mathbf{S}_t[\mathsf{write}(t)]\mathbf{G}))\big)_{t \in [T]}, \\ \mathsf{flat}(\mathbf{S}^\dagger)\mathbf{A}'_4 &= \mathbf{S}_T[1]\mathbf{P}, \end{aligned}$$

for any $\mathbf{S}^\dagger = (\mathbf{S}_0 \cup \tilde{\mathbf{S}}_1 \cup \cdots \cup \tilde{\mathbf{S}}_T \cup \mathbf{s}'_0 \cup \cdots \cup \mathbf{s}'_{T-1}) \in \mathbb{Z}^{J \times n}$. Moreover

$$\begin{aligned} \mathbf{P}_0 &= \big((\mathbf{A}_0^{(j)} - \mathbf{A}_{st})\big)_{j \in [Q]} & \mathbf{P}_1 &= \big(\mathsf{flat}_L([\![\mathbf{A}_0^{(j)} \mapsto \mathbf{D}^{(j)}]\!]_{\mathbf{W}_1})\big)_{j \in [Q]} \\ \mathbf{P}_2 &= \big([\![\mathbf{A}_{0,\mathsf{id}} \mapsto \mathbf{A}_0]\!]_{\mathbf{W}_2}\big)_{j \in [Q]} & \mathbf{P}_3 &= \big(\mathsf{flat}_L([\![\mathbf{D}_f^{(j)} \mapsto \mathbf{A}_0^{(j)}]\!]_{\mathbf{W}_3})\big)_{j \in [Q]} \\ \mathbf{P}_4 &= \big(\mathbf{P} + \mathbf{A}_0^{(j)}\mathbf{R}^{(j)}\big)_{j \in [Q]} \end{aligned}$$

and gaussian parameters

$$\begin{aligned} \chi, \qquad \chi_{\mathsf{pre}}, \qquad \chi_{\mathsf{post}} &= \chi''', \\ \chi_0, \chi_2, \chi_3 &= \chi', \qquad \chi_1, \chi_4 = \chi'', \end{aligned}$$

are as defined in Eq. (2). First, notice that our sampler Samp, whose output is described in Eqs. (8) and (9), is public-coin: its random coins are given by

$$\mathsf{coins} := (\mathsf{coins}_{\mathcal{A}}, f_1, \ldots, f_Q, \mathbf{x}, \mathbf{A}_{st}, \mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3, \mathbf{D}_{\mathsf{circ}}, \mathbf{P}, (\mathbf{A}_0^{(j)}, \mathbf{D}^{(j)}, \mathbf{R}^{(j)})_{j \in [Q]})$$

where $\mathsf{coins}_{\mathcal{A}}$ are random coins for the adversary and $f_1, \ldots, f_Q, \mathbf{x}$ the queried functions and challenge input output by the adversary respectively. It follows that aux as defined in Eq. (9) contains all random coin tosses of Samp. Here,

we use the standard fact that we can sample random matrices over $\mathbb{Z}_q$ (e.g., $\mathbf{A}_{st}$) and random gaussians (e.g. $\mathbf{R}^j$) using public-coin algorithms.

Second, notice that

$$\begin{aligned}
\mathsf{flat}(\mathbf{S}^\dagger)\mathbf{A}' + \mathbf{e}' &= \mathsf{flat}\Big(\mathbf{C}'_0 \cup (\cup_{t\in[T]}\mathbf{C}'_{t,1}) \cup (\cup_{t\in[T]}\mathbf{C}'_{t,2}) \cup (\cup_{t\in[T]}\mathbf{C}'_{t,3}) \cup \mathbf{C}'_{T,4}\Big)\\
\mathbf{S}^\dagger[\Gamma_0]\mathbf{B}_0 + \mathbf{E}_0 &= \mathbf{C}_0,\\
\mathbf{S}^\dagger[\Gamma_\gamma]\mathbf{B}_\gamma + \mathbf{E}_\gamma &= (\cup_{t\in[T]}\mathbf{C}_{t,\gamma}), \quad \text{for } \gamma = 1,2,3\\
\mathbf{S}^\dagger[\Gamma_4]\mathbf{B}_4 + \mathbf{E}_4 &= \mathbf{C}_{T,4}\\
\odot[\![\mathbf{S}^\dagger, \mathbf{D}_{\mathsf{circ}}]\!][\Gamma_2] &= \cup_{t\in[T]}(\mathring{\mathbf{C}}_t, \mathring{\mathbf{C}}'_t)\\
\odot[\![\mathbf{S}^\dagger, \mathbf{D}_{\mathsf{circ}}]\!][\Gamma_3] &= \cup_{t\in[T]}(\hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t),
\end{aligned}$$

and that

$$\mathbf{B}_\gamma^{-1}(\mathbf{P}_\gamma, \tau) = \big(\mathsf{flat}_L(\mathbf{K}_\gamma^{(j)})\big)_{j\in[Q]}, \quad \text{for } \gamma = 0,\dots,4.$$

In other words, the post-condition distribution of the circular small-secret evasive LWE assumption is identical to (up to rearranging rows)

$$\begin{aligned}
&\mathbf{A}_{st}, \mathbf{B}_0, \mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3, \mathbf{B}_4, \mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3, \mathbf{D}_{\mathsf{circ}}, \mathbf{P}\\
&\mathbf{C}_0, \mathbf{C}'_0, \{\, \mathbf{C}_{t,1}, \mathbf{C}'_{t,1}, \mathbf{C}_{t,2}, \mathbf{C}'_{t,2}, \mathbf{C}_{t,3}, \mathbf{C}'_{t,3}, \hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t, \mathring{\mathbf{C}}_t, \mathring{\mathbf{C}}'_t \,\}_{t\in[T]}, \mathbf{C}_{T,4}, \mathbf{C}'_{T,4}\\
&\mathbf{A}_0^{(j)}, \mathbf{D}^{(j)}, \mathbf{R}^{(j)}\\
&\mathbf{K}_0^{(j)}, \mathbf{K}_1^{(j)}, \mathbf{K}_2^{(j)}, \mathbf{K}_3^{(j)}, \mathbf{K}_4^{(j)}
\end{aligned} \tag{10}$$

which exactly corresponds to the adversary's view in the very selective ABE security experiment. Therefore, pseudorandomness of Eq. (10) directly implies ABE security. By the (circular small-secret) evasive LWE assumption, in order to prove pseudorandomness of the post-condition, it suffices to prove pseudorandomness of the pre-condition instead.

With this in mind, define

$$\begin{aligned}
\mathbf{Y}_0^{(j)} &\coloneqq \mathbf{S}_0(\mathbf{A}_0^{(j)} - \mathbf{A}_{st}) + \mathbf{E}_0''^{(j)}\\
\mathbf{Y}_{t,1}^{(j)} &\coloneqq \mathbf{S}_{t-1}[\mathsf{read}(t)](\llbracket \mathbf{A}_0^{(j)} \mapsto \mathbf{D}^{(j)} \rrbracket_{\mathbf{W}_1}) + \mathbf{E}_{t,1}''^{(j)}\\
\mathbf{Y}_{t,2}^{(j)} &\coloneqq \mathbf{S}_{t-1}[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)](\llbracket \mathbf{A}_{0,\mathsf{id}}^{(j)} \mapsto \mathbf{A}_0^{(j)} \rrbracket_{\mathbf{W}_2}) + \mathbf{E}_{t,2}''^{(j)}\\
\mathbf{Y}_{t,3}^{(j)} &\coloneqq (\mathbf{I}_L \otimes \mathbf{s}'_{t-1})(\llbracket \mathbf{D}_f^{(j)} \mapsto \mathbf{A}_0^{(j)} \rrbracket_{\mathbf{W}_3}) + \mathbf{E}_{t,3}''^{(j)}\\
\mathbf{Y}_{T,4}^{(j)} &\coloneqq \mathbf{S}_T[1](\mathbf{P} + \mathbf{A}_0^{(j)}\mathbf{R}^{(j)}) + \mathbf{E}_{T,4}''^{(j)}.
\end{aligned} \tag{11}$$

The matrices $\mathbf{Y}$'s are defined so as to capture the key-dependent linear parts of the corresponding recoding outputs. Concretely, by the definition of the recoding procedures, the correlated-view terms are obtained by adding the appropriate terms $\mathsf{rDec}_0(\cdots)$ to these quantities. The terms $\mathsf{diag}_L(\mathbf{C}_{t,1})\mathbf{K}_1^j$, $\mathbf{C}_{t,2}\mathbf{K}_2^j$, $(\mathbf{I}_L \otimes \mathbf{C}_{t,3})\mathbf{K}_3^j$, and $\mathbf{C}_{T,4}\mathbf{K}_4^j$ are the corresponding linear key-dependent components.

Indeed, by the definition of the recoding procedures, the corresponding correlated-view terms are obtained from the quantities above by adding the appropriate rDec_0 terms:

$$\begin{aligned}
\tilde{\mathbf{C}}_0^{(j)} &\approx_s \mathbf{Y}_0^{(j)} + \mathbf{C}'_0\\
\tilde{\mathbf{C}}_{t,1}^{(j)} &\approx_s \mathbf{Y}_{t,1}^{(j)} + \mathsf{rDec}_0\big(\mathbf{B}_1, \mathbf{W}_1, \mathbf{D}^{(j)}, \tilde{\mathbf{C}}_{t-1}^{(j)}[\mathsf{read}(t)], \mathbf{C}'_{t,1}, \mathbf{z}_{t-1}^{(j)}[\mathsf{read}(t)]\big)\\
\tilde{\mathbf{C}}_t^{(j)}[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)] &\approx_s \mathbf{Y}_{t,2}^{(j)} + \mathsf{rDec}_0\big(\mathbf{B}_2, \mathbf{W}_2, \mathbf{A}_0^{(j)}, \tilde{\mathbf{C}}_{t,2}^{(j)}, \mathbf{C}'_{t,2}, \mathbf{z}_{t-1}^{(j)}[\overline{\mathsf{write}(t)} \cap \mathsf{read}(t)]\big)\\
\tilde{\mathbf{C}}_t^{(j)}[\mathsf{write}(t)] &\approx_s \mathbf{Y}_{t,3}^{(j)} + \mathsf{rDec}_0\big(\mathbf{B}_3, \mathbf{W}_3, \mathbf{A}_0^{(j)}, \tilde{\mathbf{C}}_{t,3}^{(j)}, \mathbf{C}'_{t,3}, f(\mathbf{z}_{t-1}^{(j)}[\mathsf{read}(t)])\big)\\
\tilde{\mathbf{C}}_{T,4}^{(j)} &\approx_s \mathbf{Y}_{T,4}^{(j)} - \mathbf{C}'_{T,4} - \tilde{\mathbf{C}}_T^{(j)}[1]\mathbf{R}^{(j)}.
\end{aligned}$$

Then, up to rearranging rows, the pre-condition distribution of the (circular small-secret) evasive LWE is identical to

$$\begin{array}{l}\mathbf{A}_{st}, \mathbf{B}_0, \mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3, \mathbf{B}_4, \mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3, \mathbf{D}_{\mathsf{circ}}, \mathbf{P} \\ \mathbf{C}_0, \mathbf{C}'_0, \{\, \mathbf{C}_{t,1}, \mathbf{C}'_{t,1}, \mathbf{C}_{t,2}, \mathbf{C}'_{t,2}, \mathbf{C}_{t,3}, \mathbf{C}'_{t,3}, \hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t, \mathring{\mathbf{C}}_t, \mathring{\mathbf{C}}'_t \,\}_{t\in[T]}, \mathbf{C}_{T,4}, \mathbf{C}'_{T,4} \\ \mathbf{A}_0^{(j)}, \mathbf{D}^{(j)}, \mathbf{R}^{(j)} \\ \mathbf{Y}_0^{(j)}, \{\, \mathbf{Y}_{t,1}^{(j)}, \mathbf{Y}_{t,2}^{(j)}, \mathbf{Y}_{t,3}^{(j)} \,\}_{t\in[T]}, \mathbf{Y}_{T,4}^{(j)}\end{array} \tag{12}$$

In other words, via the evasive LWE assumption, in order to prove security of our ABE construction, it suffices to prove pseudorandomness of Eq. (12). This is what we do next.

Proving the pre-condition. We define the following additional quantities:

$$\begin{aligned}\tilde{\mathbf{C}}_0^{(j)} &= [\![\mathbf{S}_0, \mathbf{A}_0^{(j)}, \mathbf{z}_0^{(j)}]\!] + \tilde{\mathbf{E}}_0^{(j)} \\ \tilde{\mathbf{C}}_{t,1}^{(j)} &= [\![\mathbf{s}'_{t-1}, \mathbf{D}^{(j)}, \mathbf{z}_{t-1}^{(j)}[\mathsf{read}(t)]]\!] + \tilde{\mathbf{E}}_{t,1}^{(j)} \\ \tilde{\mathbf{C}}_t^{(j)}[\mathsf{write}(t)\cup\mathsf{read}(t)] &= [\![\mathbf{S}_t[\mathsf{write}(t)\cup\mathsf{read}(t)], \mathbf{A}_0^{(j)}, \mathbf{z}_t^{(j)}[\mathsf{write}(t)\cup\mathsf{read}(t)]]\!] + \tilde{\mathbf{E}}_t^{(j)} \\ \tilde{\mathbf{C}}_{T,4}^{(j)} &= \mathbf{z}_T^{(j)}[1]\cdot\mathbf{S}_T[1]\mathbf{G}\mathbf{R}^{(j)} + \tilde{\mathbf{E}}_{T,4}^{(j)}\end{aligned} \tag{13}$$

where the error terms in the above expressions have the following gaussian parameters

$$\begin{aligned}\chi' :&\quad (\mathbf{E}_0''^{(j)}, \tilde{\mathbf{E}}_0^{(j)}), (\mathbf{E}_{t,2}''^{(j)}, \mathbf{E}_{t,3}''^{(j)}, \tilde{\mathbf{E}}_t^{(j)}) \\ \chi'' :&\quad (\mathbf{E}_{t,1}''^{(j)}, \tilde{\mathbf{E}}_{t,1}^{(j)}), (\mathbf{E}_{T,4}''^{(j)}, \tilde{\mathbf{E}}_{T,4}^{(j)})\end{aligned}$$

In addition, we define (following Dec):

$$\begin{aligned}\tilde{\mathbf{C}}_t^{(j)}[\overline{\mathsf{write}(t)\cup\mathsf{read}(t)}] &\coloneqq \tilde{\mathbf{C}}_{t-1}^{(j)}[\overline{\mathsf{write}(t)\cup\mathsf{read}(t)}], \\ \tilde{\mathbf{C}}_{t,2}^{(j)} &\coloneqq \mathsf{UEvalFX}\big(\mathbf{D}^{(j)}, \mathbf{D}_{\mathsf{circ}}, \tilde{\mathbf{C}}_{t-1}^{(j)}[\overline{\mathsf{write}(t)}\cap\mathsf{read}(t)], (\mathring{\mathbf{C}}_t, \mathring{\mathbf{C}}'_t), \mathsf{id}, \mathbf{z}_{t-1}[\overline{\mathsf{write}(t)}\cap\mathsf{read}(t)]\big), \\ \tilde{\mathbf{C}}_{t,3}^{(j)} &\coloneqq \mathsf{UEvalFX}\big(\mathbf{D}^{(j)}, \mathbf{D}_{\mathsf{circ}}, \tilde{\mathbf{C}}_{t,1}^{(j)}, (\hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t), f, \mathbf{z}_{t-1}[\mathsf{read}(t)]\big)\end{aligned} \tag{14}$$

Next, we define the following hybrid distributions:

- $\mathsf{H}_{\mathsf{pre}}$: the pre-condition, i.e., the distribution

$$\begin{array}{l}\mathbf{A}_{st}, \mathbf{B}_0, \mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3, \mathbf{B}_4, \mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3, \mathbf{D}_{\mathsf{circ}}, \mathbf{P} \\ \mathbf{C}_0, \mathbf{C}'_0, \{\, \mathbf{C}_{t,1}, \mathbf{C}'_{t,1}, \mathbf{C}_{t,2}, \mathbf{C}'_{t,2}, \mathbf{C}_{t,3}, \mathbf{C}'_{t,3}, \hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t, \mathring{\mathbf{C}}_t, \mathring{\mathbf{C}}'_t \,\}_{t\in[T]}, \mathbf{C}_{T,4}, \mathbf{C}'_{T,4} \\ \mathbf{A}_0^{(j)}, \mathbf{D}^{(j)}, \mathbf{R}^{(j)} \\ \mathbf{Y}_0^{(j)}, \{\, \mathbf{Y}_{t,1}^{(j)}, \mathbf{Y}_{t,2}^{(j)}, \mathbf{Y}_{t,3}^{(j)} \,\}_{t\in[T]}, \mathbf{Y}_{T,4}^{(j)}\end{array}$$

  where the ciphertext components, except for the circular encodings for which gaussian parameter is again χ, have gaussian parameter χ_{pre}.
- $\mathsf{H}_{\mathsf{corr}}$: the distribution obtained via the correctness of iterated computation, i.e., the distribution

$$\begin{array}{l}\mathbf{A}_{st}, \mathbf{B}_0, \mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3, \mathbf{B}_4, \mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3, \mathbf{D}_{\mathsf{circ}}, \mathbf{P} \\ \mathbf{C}_0, \mathbf{C}'_0, \{\, \mathbf{C}_{t,1}, \mathbf{C}'_{t,1}, \mathbf{C}_{t,2}, \mathbf{C}'_{t,2}, \mathbf{C}_{t,3}, \mathbf{C}'_{t,3}, \hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t, \mathring{\mathbf{C}}_t, \mathring{\mathbf{C}}'_t \,\}_{t\in[T]}, \mathbf{C}_{T,4}, \mathbf{C}'_{T,4} \\ \mathbf{A}_0^{(j)}, \mathbf{D}^{(j)}, \mathbf{R}^{(j)} \\ \tilde{\mathbf{C}}_0^{(j)}, \{\tilde{\mathbf{C}}_{t,1}^{(j)}, \tilde{\mathbf{C}}_t^{(j)}[\mathsf{write}(t)\cup\mathsf{read}(t)]\}_{t=1,\ldots,T}, \tilde{\mathbf{C}}_{T,4}^{(j)}\end{array}$$

- H_0: same as $\mathsf{H}_{\mathsf{corr}}$, except

$$\begin{array}{ll}\mathbf{C}_0, \mathbf{C}'_0, \tilde{\mathbf{C}}_0^{(j)} \leftarrow \mathbb{Z}_q^{S\times m}, & \mathbf{C}_{1,1}, \mathbf{C}'_{1,1} \leftarrow \mathbb{Z}_q^{L\times m}, \\ \mathbf{C}_{1,2}, \mathbf{C}'_{1,2} \leftarrow \mathbb{Z}_q^{(L-L_1)\times m}, & \mathring{\mathbf{C}}_1, \mathring{\mathbf{C}}'_1 \leftarrow (\mathbb{Z}_q^{n\times nm\lceil\log q\rceil} \times \mathbb{Z}_q^{(n^2m\lceil\log q\rceil^2+1)m})^{(L-L_1)}.\end{array}$$

– H'_t for $t = 1, \ldots, T$: same as H_{t-1}, except

$$\tilde{\mathbf{C}}^{(j)}_{t,1} \leftarrow \mathbb{Z}_q^{L\times m}, \qquad \hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t \leftarrow \mathbb{Z}_q^{n\times nm\lceil \log q\rceil} \times \mathbb{Z}_q^{(n^2m\lceil \log q\rceil^2+1)m},$$
$$\mathbf{C}_{t,3}, \mathbf{C}'_{t,3} \leftarrow \mathbb{Z}_q^{L\times m}.$$

– H_t, for $t = 1, \ldots, T-1$: same as H'_t, except

$$\tilde{\mathbf{C}}^{(j)}_t[\mathsf{write}(t) \cup \mathsf{read}(t)] \leftarrow \mathbb{Z}_q^{L_t\times m}, \qquad \mathbf{C}_{t+1,1}, \mathbf{C}'_{t+1,1} \leftarrow \mathbb{Z}_q^{L\times m}$$
$$\mathbf{C}_{t+1,2}, \mathbf{C}'_{t+1,2} \leftarrow \mathbb{Z}_q^{(L-L_{t+1})\times m}, \qquad \mathring{\mathbf{C}}_{t+1}, \mathring{\mathbf{C}}'_{t+1} \leftarrow (\mathbb{Z}_q^{n\times nm\lceil \log q\rceil} \times \mathbb{Z}_q^{(n^2m\lceil \log q\rceil^2+1)m})^{(L-L_{t+1})}.$$

– H_T: same as H'_T, except

$$\tilde{\mathbf{C}}^{(j)}_T[\mathsf{write}(T) \cup \mathsf{read}(T)] \leftarrow \mathbb{Z}_q^{L_T\times m}, \qquad \tilde{\mathbf{C}}^{(j)}_{T,4}, \mathbf{C}_{T,4}, \mathbf{C}'_{T,4} \leftarrow \mathbb{Z}_q^m$$

The bulk of the proof lies in proving that $\mathsf{H}_t \approx_c \mathsf{H}'_t$. This is proven in Lemma 9. The computational indistinguishability of the other pairs of consecutive hybrids are stated in Lemmas 6 to 8 and 10 and their proofs can be found in Appendix D .

Lemma 6. *If* $\mathsf{H}_{\mathsf{corr}} \approx_c \$$, *then* $\mathsf{H}_{\mathsf{pre}} \approx_c \$$.

Lemma 7. $\mathsf{H}_{\mathsf{corr}} \approx_c \mathsf{H}_0$.

Proof. We want to relate the following quantities corresponding to the differences between the two games:

$$\mathsf{H}_{\mathsf{corr}} : \left\{ \begin{array}{c} \mathbf{C}_0, \mathbf{C}'_0, \tilde{\mathbf{C}}^{(j)}_0, \\ \mathbf{C}_{1,1}, \mathbf{C}'_{1,1}, \\ \mathbf{C}_{1,2}, \mathbf{C}'_{1,2}, \\ \mathring{\mathbf{C}}_1, \mathring{\mathbf{C}}'_1 \end{array} \right\} = \left\{ \begin{array}{c} \mathbf{S}_0\mathbf{B}_0 + \mathbf{E}_0, [\![\mathbf{S}_0, \mathbf{A}_{st}, \mathbf{z}_0]\!] + \mathbf{E}'_0, [\![\mathbf{S}_0, \mathbf{A}^{(j)}_0, \mathbf{z}^{(j)}_0]\!] + \tilde{\mathbf{E}}^{(j)}_0, \\ [\![\mathbf{S}_0[\mathsf{read}(1)] \mapsto \mathbf{s}'_0]\!]_{\mathbf{B}_1,\mathbf{W}_1} + (\mathbf{E}_{1,1}, \mathbf{E}'_{1,1}), \\ [\![\mathbf{S}_0[\overline{\mathsf{write}(1)} \cap \mathsf{read}(1)] \mapsto \mathbf{S}_1[\overline{\mathsf{write}(1)} \cap \mathsf{read}(1)]]\!]_{\mathbf{B}_2,\mathbf{W}_2} + (\mathbf{E}_{1,2}, \mathbf{E}'_{1,2}), \\ \odot[\![\mathbf{S}_0[\overline{\mathsf{write}(1)} \cap \mathsf{read}(1)], \mathbf{D}_{\mathsf{circ}}]\!] \end{array} \right\}$$

$$\mathsf{H}_0 : \left\{ \begin{array}{c} \mathbf{C}_0, \mathbf{C}'_0, \tilde{\mathbf{C}}^{(j)}_0, \\ \mathbf{C}_{1,1}, \mathbf{C}'_{1,1}, \\ \mathbf{C}_{1,2}, \mathbf{C}'_{1,2}, \\ \mathring{\mathbf{C}}_1, \mathring{\mathbf{C}}'_1 \end{array} \right\} \leftarrow \left\{ \begin{array}{c} \mathbb{Z}_q^{S\times m}, \mathbb{Z}_q^{S\times m}, \mathbb{Z}_q^{S\times m}, \\ \mathbb{Z}_q^{L\times m}, \mathbb{Z}_q^{L\times m}, \\ \mathbb{Z}_q^{(L-L_1)\times m}, \mathbb{Z}_q^{(L-L_1)\times m}, \\ (\mathbb{Z}_q^{n\times nm\lceil \log q\rceil}, \mathbb{Z}_q^{(n^2m\lceil \log q\rceil^2+1)m})^{(L-L_1)} \end{array} \right\}$$

We proceed as in the proof of Lemma 9 (see below) as if $t = 0$ and $\mathsf{write}(0) = \mathsf{read}(0) = [S]$, except with the additional $\mathbf{C}_0, \mathbf{C}'_0$ terms, i.e., the LWE matrix additionally includes $\mathbf{B}_0$ and $\mathbf{A}_{st}$. Similarly to Lemma 9, we need to rely on Lemma 12 due to the presence of the terms $\mathbf{z}_0, \mathbf{z}^{(j)}_0$. □

Lemma 8. *For* $t = 1, \ldots, T$, *we have* $\mathsf{H}_{t-1} \approx_c \mathsf{H}'_t$.

Proof. We want to relate the following quantities corresponding to the differences between the two games:

$$\mathsf{H}_{t-1} : \left\{ \begin{array}{c} \tilde{\mathbf{C}}^{(j)}_{t,1}, \\ \mathbf{C}_{t,3}, \mathbf{C}'_{t,3}, \\ \hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t \end{array} \right\} = \left\{ \begin{array}{c} [\![\mathbf{s}'_{t-1}, \mathbf{D}^{(j)}, \mathbf{z}^{(j)}_{t-1}[\mathsf{read}(t)]]\!] + \tilde{\mathbf{E}}^{(j)}_{t,1}, \\ [\![\mathbf{s}'_{t-1} \mapsto \mathbf{S}_t[\mathsf{write}(t)]]\!]_{\mathbf{B}_3,\mathbf{W}_3} + (\mathbf{E}_{t,3}, \mathbf{E}'_{t,3}), \\ \odot[\![\mathbf{s}'_{t-1}, \mathbf{D}_{\mathsf{circ}}]\!] \end{array} \right\}$$

$$\mathsf{H}'_t : \left\{ \begin{array}{c} \tilde{\mathbf{C}}^{(j)}_{t,1}, \\ \mathbf{C}_{t,3}, \mathbf{C}'_{t,3}, \\ \hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t \end{array} \right\} \leftarrow \left\{ \begin{array}{c} \mathbb{Z}_q^{L\times m} \\ \mathbb{Z}_q^m, \mathbb{Z}_q^{L\times m}, \\ \mathbb{Z}_q^{n\times nm\lceil \log q\rceil}, \mathbb{Z}_q^{(n^2m\lceil \log q\rceil^2+1)m} \end{array} \right\}$$

It suffices to show that:

$$\left\{ \begin{array}{c} [\![\mathbf{s}'_{t-1}, \mathbf{D}^{(j)}, \mathbf{z}^{(j)}_{t-1}[\mathsf{read}(t)]]\!] + \tilde{\mathbf{E}}^{(j)}_{t,1}, \\ [\![\mathbf{s}'_{t-1} \mapsto \mathbf{S}_t[\mathsf{write}(t)]]\!]_{\mathbf{B}_3,\mathbf{W}_3} + (\mathbf{E}_{t,3}, \mathbf{E}'_{t,3}), \\ \odot[\![\mathbf{s}'_{t-1}, \mathbf{D}_{\mathsf{circ}}]\!] \end{array} \right\} \approx_c \left\{ \begin{array}{c} \mathbb{Z}_q^{L\times m} \\ \mathbb{Z}_q^m, \mathbb{Z}_q^{L\times m}, \\ \mathbb{Z}_q^{n\times nm\lceil \log q\rceil}, \mathbb{Z}_q^{(n^2m\lceil \log q\rceil^2+1)m} \end{array} \right\}$$

where the distinguisher also gets

$$\mathsf{aux}_t := (\mathbf{B}_3, \mathbf{W}_3, \mathbf{D}_{\mathsf{circ}}, \mathbf{D}^{(j)}, \mathbf{S}_t[\mathsf{write}(t)])$$

To show that the computational indistinguishability of the above two distributions, we rely on $\mathsf{cssLWE}_{n,q,\chi,\chi_{\mathsf{pre}}}$. The reduction works as follows:

- it parses

$$\mathbf{A} = (\mathsf{flat}_L(\tilde{\mathbf{D}}^{(j)}) \mid \mathbf{D}_{\mathsf{circ}} \mid \mathbf{B}_3 \mid \mathsf{flat}(\mathbf{W}_3)) \in \mathbb{Z}_q^{n \times (QLm + (n^2 m \lceil \log q \rceil^2 + 1)m + m + Lm)},$$
$$\mathbf{u} = (\mathsf{flat}(\mathbf{U}^{(j)}) \mid \mathbf{u}_{\mathsf{circ}} \mid \mathbf{u}_3 \mid \mathsf{flat}(\mathbf{U}'_3)) \in \mathbb{Z}_q^{QLm + (n^2 m \lceil \log q \rceil^2 + 1)m + m + Lm},$$

  and $\hat{\mathbf{U}} \in \mathbb{Z}_q^{n \times nm\lceil \log q \rceil}$ obtained from the $\mathsf{cssLWE}_{n,q,\chi,\chi_{\mathsf{pre}}}$ instance,
- it sets $\mathbf{D}^{(j)} = \tilde{\mathbf{D}}^{(j)} + \mathbf{z}_{t-1}^{(j)}[\mathsf{read}(t)]^\top \otimes \mathbf{G}$ while using $\mathbf{D}_{\mathsf{circ}}, \mathbf{B}_3, \mathbf{W}_3$ as obtained from the cssLWE instance,
- it samples $\mathbf{S}_t[\mathsf{write}(t)] \leftarrow \mathbb{Z}_q^{L \times m}$, and outputs

$$\left\{ \begin{array}{c} \mathbf{U}^{(j)}, \\ (\mathbf{u}_3, \mathbf{U}'_3) + (\mathbf{0}, \mathbf{S}_t[\mathsf{write}(t)]\mathbf{G}), \\ (\hat{\mathbf{U}}, \mathbf{u}_{\mathsf{circ}}) \end{array} \right\}$$
$$\mathsf{aux}_t = (\mathbf{B}_3, \mathbf{W}_3, \mathbf{D}_{\mathsf{circ}}, \mathbf{D}^{(j)}, \mathbf{S}_t[\mathsf{write}(t)])$$

Observe that

- if $(\mathbf{A}, \mathbf{U}, \mathbf{u})$ is a structured $\mathsf{cssLWE}_{n,q,\chi,\chi_{\mathsf{pre}}}$ instance, the view of the distinguisher is identical to H'_{t-1}; (where we implicitly use that $\chi_{\mathsf{pre}} \ll \chi'$, and that, therefore, the error term of $\tilde{\mathbf{C}}_{t,1}^{(j)}$ noise floods the LWE error terms)
- if $(\mathbf{A}, \mathbf{U}, \mathbf{u})$ is a uniform random instance, the view of the distinguisher is identical to H_t.

We conclude that $\mathsf{H}_{t-1} \approx_c \mathsf{H}'_t$ as required, as the rest of the distribution can be produced by the reduction, using the auxiliary information aux_t, as follows:

1. the matrices

$$\mathbf{A}_{st}, \mathbf{B}_0, \mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_4, \mathbf{W}_1, \mathbf{W}_2, \mathbf{P}$$
$$\mathbf{C}_0, \mathbf{C}'_0, \{\mathbf{C}_{t',1}, \mathbf{C}'_{t',1}, \mathbf{C}_{t',2}, \mathbf{C}'_{t',2}, \mathring{\mathbf{C}}_{t'}, \mathring{\mathbf{C}}'_t\}_{t' \in [t]}, \{\mathbf{C}_{t,3}, \mathbf{C}'_{t,3}, \hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t\}_{t' \in [t-1]}$$
$$\mathbf{A}_0^{(j)}, \mathbf{R}^{(j)}$$
$$\tilde{\mathbf{C}}_0^{(j)}, \{\tilde{\mathbf{C}}_{t',1}^{(j)}\}_{t' \in [t-1]}, \{\tilde{\mathbf{C}}_{t'}^{(j)}[\mathsf{write}(t) \cup \mathsf{read}(t)]\}_{t' \in [t-1]}$$

   are samples as in H_{t-1}.
2. The auxiliary information $\mathsf{aux}_t = (\mathbf{B}_3, \mathbf{W}_3, \mathbf{D}_{\mathsf{circ}}, \mathbf{D}^{(j)}, \mathbf{S}_t[\mathsf{write}(t)])$ fixes the entry $\mathbf{S}_t[\mathsf{write}(t)]$ needed at time t.
3. The reduction samples the remaining entries of $\mathbf{S}_t$ exactly as in Enc. It then samples $\mathbf{S}_{t+1}, \ldots, \mathbf{S}_T$ and $\mathbf{s}'_t, \mathbf{s}'_{t+1}, \ldots, \mathbf{s}'_{T-1}$ according to the original encryption procedure.
4. Using these samples, it produces $(\mathbf{C}_{t',1}, \mathbf{C}'_{t',1}), (\mathbf{C}_{t',2}, \mathbf{C}'_{t',2}), (\mathbf{C}_{t',3}, \mathbf{C}'_{t',3}), (\hat{\mathbf{C}}_{t'}, \hat{\mathbf{C}}'_{t'}), (\mathring{\mathbf{C}}_{t'}, \mathring{\mathbf{C}}'_{t'})$ for $t' = t+1, \ldots, T$, and $(\mathbf{C}_{T,4}, \mathbf{C}'_{T,4})$ as in Enc.
5. Finally, it computes $\tilde{\mathbf{C}}_{t'}^{(j)}[\mathsf{write}(t) \cup \mathsf{read}(t)]$ for $t' = t, \ldots, T$, and $\tilde{\mathbf{C}}_{t',1}^{(j)}$ for $t' = t+1, \ldots, T$, as in Eq. (13).

□

Lemma 9. *For $t = 1, \ldots, T-1$, we have $\mathsf{H}'_t \approx_c \mathsf{H}_t$.*

Lemma 10. $\mathsf{H}'_T \approx_c \mathsf{H}_T$.

Proof. We want to relate the following quantities corresponding to the differences between the two games:

$$\mathsf{H}_T : \left\{\begin{array}{c} \tilde{\mathbf{C}}_T^{(j)}[\mathsf{write}(T) \cup \mathsf{read}(T)], \\ \tilde{\mathbf{C}}_{T,4}^{(j)}, \\ \mathbf{C}_{T,4}, \\ \mathbf{C}'_{T,4} \end{array}\right\} = \left\{\begin{array}{c} [\![\mathbf{S}_T[\mathsf{write}(T) \cup \mathsf{read}(T)], \mathbf{A}_0^{(j)}, \mathbf{z}_T^{(j)}[\mathsf{write}(T) \cup \mathsf{read}(T)]]\!] + \tilde{\mathbf{E}}_T^{(j)}, \\ \mathbf{z}_T^{(j)}[1]\mathbf{S}_T[1]\mathbf{G}\mathbf{R}^{(j)} + \tilde{\mathbf{E}}_{T,4}^{(j)}, \\ \mathbf{S}_T[1]\mathbf{B}_4 + \mathbf{E}_{T,4}, \\ \mathbf{S}_T[1]\mathbf{P} + \boldsymbol{\mu} \cdot \lfloor q/2 \rfloor + \mathbf{E}'_{T,4} \end{array}\right\}$$

$$\mathsf{H}'_T : \left\{\begin{array}{c} \tilde{\mathbf{C}}_T^{(j)}[\mathsf{write}(T) \cup \mathsf{read}(T)], \\ \tilde{\mathbf{C}}_{T,4}^{(j)}, \\ \mathbf{C}_{T,4}, \\ \mathbf{C}'_{T,4} \end{array}\right\} \leftarrow \left\{\begin{array}{c} \mathbb{Z}_q^{L_t \times m}, \\ \mathbb{Z}_q^m, \\ \mathbb{Z}_q^m, \\ \mathbb{Z}_q^m \end{array}\right\}$$

where the distinguisher also gets

$$\mathsf{aux}_t := (\mathbf{A}_0^{(j)}, \mathbf{R}^{(j)}, \mathbf{B}_4, \mathbf{P})$$

Notice that since the attribute $\mathbf{z}_0$ does not satisfy any of the function $f^{(j)}$, this means that $\mathbf{z}_T^{(j)}[1] = 1$ for all j. In what follows we ignore the terms $\mathbf{z}_T^{(j)}$ in $\tilde{\mathbf{C}}_T^{(j)}[\mathsf{write}(T) \cup \mathsf{read}(T)]$: these can be dealt with as in Lemma 12, via a hybrid argument over the rows of $\mathbf{S}_T$. To show that the computational indistinguishability of the above two distributions, we rely on $\mathsf{LWE}_{n,q,\chi_{\mathsf{pre}}}$. The reduction works as follows:

- it parses

$$\mathbf{A} = (\tilde{\mathbf{A}}_0^{(j)} \mid \mathbf{V}^{(j)} \mid \mathbf{B}_4 \mid \mathbf{P}) \in \mathbb{Z}_q^{n \times (Qm+Qm+m+m)},$$
$$\mathbf{u} = (\mathbf{u}^{(j)} \mid \mathbf{v}^{(j)} \mid \mathbf{u}_4 \mid \mathbf{v}) \in \mathbb{Z}_q^{Qm+Qm+m+m},$$

  obtained from the $\mathsf{LWE}_{n,q,\chi_{\mathsf{pre}}}$ instance,
- it sets $\mathbf{R}^{(j)} \leftarrow \mathsf{SamplePre}(\mathbf{G}, \mathbf{I}, \mathbf{V}^{(j)}, \tau)$, i.e., $\mathbf{V}^{(j)} = \mathbf{G}\mathbf{R}^{(j)}$, and uses $\mathbf{A}_0^{(j)}, \mathbf{B}_4, \mathbf{P}$ as obtained from the LWE instance,
- it outputs

$$\left\{\begin{array}{c} \mathbf{u}^{(j)}, \\ \mathbf{v}^{(j)}, \\ \mathbf{u}_4, \\ \mathbf{v} \end{array}\right\}$$
$$\mathsf{aux}_t = (\mathbf{A}_0^{(j)}, \mathbf{R}^{(j)}, \mathbf{B}_4, \mathbf{P},)$$

Observe that

- if $(\mathbf{A}, \mathbf{u})$ is a structured $\mathsf{LWE}_{n,q,\chi_{\mathsf{pre}}}$ instance, the view of the distinguisher is identical to H_T ;
- if $(\mathbf{A}, \mathbf{u})$ is a uniform random instance, the view of the distinguisher is identical to H'_T.

The rest of the distribution will be produced by the reduction, using the auxiliary information aux_t, by sampling

$$\mathbf{A}_{st}, \mathbf{B}_0, \mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3, \mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3, \mathbf{D}_{\mathsf{circ}}$$
$$\mathbf{C}_0, \mathbf{C}'_0, \{\mathbf{C}_{t',1}, \mathbf{C}'_{t',1}, \mathbf{C}_{t',2}, \mathbf{C}'_{t',2}, \mathring{\mathbf{C}}_{t'}, \mathring{\mathbf{C}}'_t\}_{t' \in [T]}, \{\mathbf{C}_{t,3}, \mathbf{C}'_{t,3}, \hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t\}_{t' \in [T]}$$
$$\mathbf{D}_0^{(j)}$$
$$\tilde{\mathbf{C}}_0^{(j)}, \{\tilde{\mathbf{C}}_{t',1}^{(j)}\}_{t' \in [T]}, \{\tilde{\mathbf{C}}_{t'}^{(j)}[\mathsf{write}(t) \cup \mathsf{read}(t)]\}_{t' \in [T-1]}$$

as in H'_T. We conclude that $\mathsf{H}'_T \approx_c \mathsf{H}_T$ as required. □

We now prove the main lemma.

Proof (Lemma 9). The two games differ in how the quantities

$$\left\{\begin{array}{c}\tilde{\mathbf{C}}_t^{(j)}[\mathsf{write}(t)\cup\mathsf{read}(t)],\\ \mathbf{C}_{t+1,1},\mathbf{C}'_{t+1,1},\\ \mathbf{C}_{t+1,2},\mathbf{C}'_{t+1,2},\\ \mathring{\mathbf{C}}_{t+1},\mathring{\mathbf{C}}'_{t+1}\end{array}\right\}$$

are computed. In particular, we have

$$\mathsf{H}'_t:\left\{\begin{array}{c}[\![\mathbf{S}_t[\mathsf{write}(t)\cup\mathsf{read}(t)],\mathbf{A}_0^{(j)},\mathbf{z}_t^{(j)}[\mathsf{write}(t)\cup\mathsf{read}(t)]]\!]+\tilde{\mathbf{E}}_t^{(j)},\\ [\![\mathbf{S}_t[\mathsf{read}(t+1)]\mapsto\mathbf{s}'_t]\!]_{\mathbf{B}_1,\mathbf{W}_1}+(\mathbf{E}_{t+1,1},\mathbf{E}'_{t+1,1}),\\ [\![\mathbf{S}_t[\overline{\mathsf{write}(t+1)}\cap\mathsf{read}(t+1)]\mapsto\mathbf{S}_{t+1}[\overline{\mathsf{write}(t+1)}\cap\mathsf{read}(t+1)]]\!]_{\mathbf{B}_2,\mathbf{W}_2}+(\mathbf{E}_{t+1,2},\mathbf{E}'_{t+1,2}),\\ \odot[\![\mathbf{S}_t[\overline{\mathsf{write}(t+1)}\cap\mathsf{read}(t+1)],\mathbf{D}_{\mathsf{circ}}]\!]\end{array}\right\}$$

$$\mathsf{H}_t:\qquad\left\{\begin{array}{c}\mathbb{Z}_q^{L_t\times m},\\ \mathbb{Z}_q^{L\times m},\ \mathbb{Z}_q^{L\times m},\\ \mathbb{Z}_q^{(L-L_t)\times m},\ \mathbb{Z}_q^{(L-L_t)\times m},\\ (\mathbb{Z}_q^{n\times nm\lceil\log q\rceil},\ \mathbb{Z}_q^{(n^2m\lceil\log q\rceil^2+1)m})^{(L-L_t)}\end{array}\right\}$$

Therefore, it suffices to show that:

$$\left\{\begin{array}{c}[\![\mathbf{S}_t[\mathsf{write}(t)\cup\mathsf{read}(t)],\mathbf{A}_0^{(j)},\mathbf{z}_t^{(j)}[\mathsf{write}(t)\cup\mathsf{read}(t)]]\!]+\tilde{\mathbf{E}}_t^{(j)},\\ [\![\mathbf{S}_t[\mathsf{read}(t+1)]\mapsto\mathbf{s}'_t]\!]_{\mathbf{B}_1,\mathbf{W}_1}+(\mathbf{E}_{t+1,1},\mathbf{E}'_{t+1,1}),\\ [\![\mathbf{S}_t[\overline{\mathsf{write}(t+1)}\cap\mathsf{read}(t+1)]\mapsto\mathbf{S}_{t+1}[\overline{\mathsf{write}(t+1)}\cap\mathsf{read}(t+1)]]\!]_{\mathbf{B}_2,\mathbf{W}_2}+(\mathbf{E}_{t+1,2},\mathbf{E}'_{t+1,2}),\\ \odot[\![\mathbf{S}_t[\overline{\mathsf{write}(t+1)}\cap\mathsf{read}(t+1)],\mathbf{D}_{\mathsf{circ}}]\!]\end{array}\right\}$$
$$\approx_c\left\{\begin{array}{c}\mathbb{Z}_q^{L_t\times m},\\ \mathbb{Z}_q^{L\times m},\ \mathbb{Z}_q^{L\times m},\\ \mathbb{Z}_q^{(L-L_t)\times m},\ \mathbb{Z}_q^{(L-L_t)\times m},\\ (\mathbb{Z}_q^{n\times nm\lceil\log q\rceil},\ \mathbb{Z}_q^{(n^2m\lceil\log q\rceil^2+1)m})^{(L-L_t)}\end{array}\right\}$$

where the distinguisher also gets

$$\begin{aligned}\mathsf{aux}_t:=(&\mathbf{A}_0^{(j)},\mathbf{B}_1,\mathbf{W}_1,\mathbf{B}_2,\mathbf{W}_2,\mathbf{s}'_t,\mathbf{S}_{t+1}[\mathsf{read}(t+1)\cup\mathsf{write}(t+1)]\\ &\mathbf{S}_t[\overline{\mathsf{read}(t+1)}]\cdot(\mathbf{B}_1\mid\mathbf{W}_1)+(\mathbf{E}\mid\mathbf{E}')\\ &\mathbf{S}_t[\overline{\mathsf{read}(t+1)}]\cdot(\mathbf{B}_2\mid\mathbf{W}_2)+(\mathbf{F}\mid\mathbf{F}')\\ &\odot[\![\mathbf{S}_t[\overline{\mathsf{read}(t+1)}],\mathbf{D}_{\mathsf{circ}}]\!]\end{aligned}$$

with $\mathbf{E},\mathbf{E}',\mathbf{F},\mathbf{F}'$ gaussian of parameter χ_{pre}. In fact, we will prove a stronger statement, namely:

$$\left\{\begin{array}{c}[\![\mathbf{S}_t,\mathbf{A}_0^{(j)},\mathbf{z}_t^{(j)}]\!]+\tilde{\mathbf{E}}_t^{(j)},\\ [\![\mathbf{S}_t[\mathsf{read}(t+1)]\mapsto\mathbf{s}'_t]\!]_{\mathbf{B}_1,\mathbf{W}_1}+(\mathbf{E}_{t+1,1},\mathbf{E}'_{t+1,1}),\\ [\![\mathbf{S}_t[\overline{\mathsf{write}(t+1)}\cap\mathsf{read}(t+1)]\mapsto\mathbf{S}_{t+1}[\overline{\mathsf{write}(t+1)}\cap\mathsf{read}(t+1)]]\!]_{\mathbf{B}_2,\mathbf{W}_2}+(\mathbf{E}_{t+1,2},\mathbf{E}'_{t+1,2}),\\ \odot[\![\mathbf{S}_t[\overline{\mathsf{write}(t+1)}\cap\mathsf{read}(t+1)],\mathbf{D}_{\mathsf{circ}}]\!]\end{array}\right\}$$
$$\approx_c\left\{\begin{array}{c}\mathbb{Z}_q^{S\times m},\\ \mathbb{Z}_q^{L\times m},\ \mathbb{Z}_q^{L\times m},\\ \mathbb{Z}_q^{(L-L_t)\times m},\ \mathbb{Z}_q^{(L-L_t)\times m},\\ (\mathbb{Z}_q^{n\times nm\lceil\log q\rceil},\ \mathbb{Z}_q^{(n^2m\lceil\log q\rceil^2+1)m})^{(L-L_t)}\end{array}\right\}$$

where, with an abuse of notation, we denote the error term of $[\![\mathbf{S}_t,\mathbf{A}_0^{(j)},\mathbf{z}_t^{(j)}]\!]$ as that of $[\![\mathbf{S}_t[\mathsf{write}(t)\cup\mathsf{read}(t)],\mathbf{A}_0^{(j)},\mathbf{z}_t^{(j)}[\mathsf{write}(t)\cup\mathsf{read}(t)]]\!]$, i.e., $\tilde{\mathbf{E}}_t^{(j)}$ as before: it has the same gaussian parameter, namely χ', but different dimensions ($S\times m$ instead of $L_t\times m$). As before, the distinguisher also gets aux_t defined above.

To show that the computational indistinguishability of the above two distributions, we rely on $\mathsf{cssLWE}_{n,q,\chi,\chi_{\mathsf{pre}}}$. In what follows we ignore the terms $\mathbf{z}_t^{(j)}$: these can be dealt with as in Lemma 12, via a hybrid argument over the rows of $\mathbf{S}_t$. The reduction works as follows:

- it parses

$$\begin{aligned}\mathbf{A} &= (\mathbf{A}_0^{(j)} \mid \mathbf{D}_{\mathsf{circ}} \mid \mathbf{B}_1 \mid \mathbf{W}_1 \mid \mathbf{B}_2 \mid \mathbf{W}_2) \in \mathbb{Z}_q^{n\times(Qm+(n^2m\lceil\log q\rceil^2+1)m+m+m+m+m)},\\ \mathbf{U} &= (\mathbf{U}^{(j)} \mid \mathbf{U}_{\mathsf{circ}} \mid \mathbf{U}_1 \mid \mathbf{U}_1' \mid \mathbf{U}_2 \mid \mathbf{U}_2') \in \mathbb{Z}_q^{S\times(Qm+(n^2m\lceil\log q\rceil^2+1)m+m+m+m+m)},\\ &\text{where } (\mathbf{U}_1,\mathbf{U}_1')[\overline{\mathsf{read}(t+1)}] = \mathbf{S}_t[\overline{\mathsf{read}(t+1)}]\cdot(\mathbf{B}_1 \mid \mathbf{W}_1) + (\mathbf{E} \mid \mathbf{E}')\\ &\text{and } (\mathbf{U}_2,\mathbf{U}_2')[\overline{\mathsf{read}(t+1)}] = \mathbf{S}_t[\overline{\mathsf{read}(t+1)}]\cdot(\mathbf{B}_2 \mid \mathbf{W}_2) + (\mathbf{F} \mid \mathbf{F}')\end{aligned}$$

  and $\mathring{\mathbf{U}} = (\mathring{\mathbf{U}}_i)_{i\in[S]} \in (\mathbb{Z}_q^{n\times nm\lceil\log q\rceil})^S$, where

$$((\mathring{\mathbf{U}}_i)_{i\in\overline{[\mathsf{read}(t+1)]}}, \mathbf{U}_{\mathsf{circ}}[\overline{\mathsf{read}(t+1)}]) = \odot[\![\mathbf{S}_t[\overline{\mathsf{read}(t+1)}], \mathbf{D}_{\mathsf{circ}}]\!],$$

  obtained from the $\mathsf{cssLWE}_{n,q,\chi,\chi_{\mathsf{pre}}}$ instance, (this formulation can be shown to be equivalent to the standard definition, again by a hybrid argument),
- it uses $\mathbf{A}_0^{(j)}, \mathbf{D}_{\mathsf{circ}}, \mathbf{B}_1, \mathbf{W}_1, \mathbf{B}_2, \mathbf{W}_2$ as obtained from the LWE instance,
- it samples $\mathbf{s}_t' \leftarrow \mathcal{D}_{\mathbb{Z}^m,\chi^*}$, $\mathbf{S}_{t+1}[\mathsf{read}(t+1)\cup\mathsf{write}(t+1)] \leftarrow \mathcal{D}_{\mathbb{Z}^{L_t\times m},\chi^*}$, and outputs

$$\left\{\begin{array}{c}\mathbf{U}^{(j)},\\ (\mathbf{U}_1[\mathsf{read}(t+1)], \mathbf{U}_1'[\mathsf{read}(t+1)]) + (\mathbf{0}, (\mathbf{I}_L\otimes\mathbf{s}_t')\mathbf{G}),\\ (\mathbf{U}_2[\overline{\mathsf{write}(t+1)}\cap\mathsf{read}(t+1)], \mathbf{U}_2'[\overline{\mathsf{write}(t+1)}\cap\mathsf{read}(t+1)]) + (\mathbf{0}, \mathbf{S}_{t+1}[\overline{\mathsf{write}(t+1)}\cap\mathsf{read}(t+1)]\mathbf{G}),\\ ((\mathring{\mathbf{U}}_i)_{i\in\overline{[\mathsf{write}(t+1)\cap\mathsf{read}(t+1)]}}, \mathbf{U}_{\mathsf{circ}}[\overline{\mathsf{write}(t+1)}\cap\mathsf{read}(t+1)])\end{array}\right\}$$

  and

$$\begin{aligned}\mathsf{aux}_t = (&\mathbf{B}_1, \mathbf{W}_1, \mathbf{B}_2, \mathbf{W}_2, \mathbf{A}_0^{(j)}, \mathbf{s}_t', \mathbf{S}_{t+1}[\mathsf{read}(t+1)\cup\mathsf{write}(t+1)]\\ &(\mathbf{U}_1,\mathbf{U}_1')[\overline{\mathsf{read}(t+1)}] = \mathbf{S}_t[\overline{\mathsf{read}(t+1)}]\cdot(\mathbf{B}_1 \mid \mathbf{W}_1) + (\mathbf{E} \mid \mathbf{E}')\\ &(\mathbf{U}_2,\mathbf{U}_2')[\overline{\mathsf{read}(t+1)}] = \mathbf{S}_t[\overline{\mathsf{read}(t+1)}]\cdot(\mathbf{B}_2 \mid \mathbf{W}_2) + (\mathbf{F} \mid \mathbf{F}')\\ &((\mathring{\mathbf{U}}_i)_{i\in\overline{[\mathsf{read}(t+1)]}}, \mathbf{U}_{\mathsf{circ}}[\overline{\mathsf{read}(t+1)}])\end{aligned}$$

Observe that

- if $(\mathbf{A}, \mathbf{U}, \mathring{\mathbf{U}})$ is a structured $\mathsf{cssLWE}_{n,q,\chi,\chi_{\mathsf{pre}}}$ instance, the view of the distinguisher is identical to H_t' (where we implicitly use that $\chi_{\mathsf{pre}} \ll \chi'$, and that, therefore, the error term of $\tilde{\mathbf{C}}_t^{(j)}$ can be used to noise flood (Lemma 1) the LWE error terms);
- if $(\mathbf{A}, \mathbf{U}, \mathring{\mathbf{U}})$ is a uniform random instance, the view of the distinguisher is identical to H_t.

The rest of the distribution will be produced by the reduction, using the auxiliary information aux_t, as follows:

1. the matrices

$$\begin{aligned}&\mathbf{A}_{st}, \mathbf{B}_0, \mathbf{B}_3, \mathbf{B}_4, \mathbf{W}_3, \mathbf{P}\\ &\mathbf{C}_0, \mathbf{C}_0', \{\mathbf{C}_{t',1}, \mathbf{C}_{t',1}', \mathbf{C}_{t',2}, \mathbf{C}_{t',2}', \mathring{\mathbf{C}}_{t'}, \mathring{\mathbf{C}}_t'\}_{t'\in[t]}, \{\mathbf{C}_{t,3}, \mathbf{C}_{t,3}', \hat{\mathbf{C}}_t, \hat{\mathbf{C}}_t'\}_{t'\in[t]}\\ &\mathbf{D}_0^{(j)}, \mathbf{R}^{(j)}\\ &\tilde{\mathbf{C}}_0^{(j)}, \{\tilde{\mathbf{C}}_{t',1}^{(j)}\}_{t'\in[t]}, \{\tilde{\mathbf{C}}_{t'}^{(j)}[\mathsf{write}(t)\cup\mathsf{read}(t)]\}_{t'\in[t-1]}\end{aligned}$$

   are samples as in H_t'.
2. it samples $\mathbf{s}_t', \mathbf{s}_{t+1}', \cdots, \mathbf{s}_{T-1}'$, starting from $\mathbf{s}_t'$ given in aux_t exactly as in Enc,

3. it samples $\mathbf{S}_{t'}[\mathsf{read}(t') \cup \mathsf{write}(t')] \leftarrow \mathbb{Z}_q^{L_{t'} \times m}$, for $t' = t+1, \ldots, T$,
4. it obtains

$$\left.\begin{array}{l} \mathbf{S}_{t'}(\mathbf{B}_1 \mid \mathbf{W}_1) + (\mathbf{E}_{t'} \mid \mathbf{E}'_{t'}), \\ \mathbf{S}_{t'}(\mathbf{B}_2 \mid \mathbf{W}_2) + (\mathbf{F}_{t'} \mid \mathbf{F}'_{t'}), \\ \odot[\![\mathbf{S}_{t'}, \mathbf{D}_{\mathsf{circ}}]\!] \end{array}\right\} \quad \text{for } t' = t+1, \ldots, T$$

where $(\mathbf{E}_{t'} \mid \mathbf{E}'_{t'}), (\mathbf{F}_{t'} \mid \mathbf{F}'_{t'}) \sim \mathcal{D}_{\mathbb{Z}^{S \times 2m}, \chi_{\mathsf{pre}}}$, from

$$\left.\begin{array}{l} \mathbf{S}_t[\overline{\mathsf{read}(t+1)}] \cdot (\mathbf{B}_1 \mid \mathbf{W}_1) + (\mathbf{E} \mid \mathbf{E}'), \\ \mathbf{S}_t[\overline{\mathsf{read}(t+1)}] \cdot (\mathbf{B}_2 \mid \mathbf{W}_2) + (\mathbf{F} \mid \mathbf{F}'), \\ \odot[\![\mathbf{S}_t[\mathsf{read}(t+1)], \mathbf{D}_{\mathsf{circ}}]\!] \end{array}\right\} \quad \text{given in } \mathsf{aux}_t, \text{ and} \tag{15}$$
$$\mathbf{S}_{t+1}[\mathsf{read}(t+1) \cup \mathsf{write}(t+1)], \ldots, \mathbf{S}_T[\mathsf{read}(T) \cup \mathsf{write}(T)],$$

respectively. This follows from the following two observation (we are going to state them for $(\mathbf{B}_1 \mid \mathbf{W}_1)$ only, but they apply identically to $(\mathbf{B}_2 \mid \mathbf{W}_2)$ and $\odot[\![\mathbf{S}_{t'}, \mathbf{D}_{\mathsf{circ}}]\!]$ too):

i) given

$$\mathbf{S}_t[\overline{\mathsf{read}(t+1)}] \cdot (\mathbf{B}_1 \mid \mathbf{W}_1) + (\mathbf{E} \mid \mathbf{E}') \quad \text{and} \quad \mathbf{S}_{t+1}[\mathsf{read}(t+1) \cup \mathsf{write}(t+1)],$$

where $(\mathbf{E} \mid \mathbf{E}') \sim \mathcal{D}_{\mathbb{Z}^{L_{t+1} \times 2m}, \chi_{\mathsf{pre}}}$, one can derive

$$\mathbf{S}_{t+1}(\mathbf{B}_1 \mid \mathbf{W}_1) + (\mathbf{E}_{t+1}, \mathbf{E}'_{t+1}),$$

with $(\mathbf{E}_{t+1}, \mathbf{E}'_{t+1}) \sim \mathcal{D}_{\mathbb{Z}^{S \times 2m}, \chi_{\mathsf{pre}}}$. This follows from the fact that

$$\mathbf{S}_{t+1}[\overline{\mathsf{write}(t+1) \cup \mathsf{read}(t+1)}] = \mathbf{S}_t[\overline{\mathsf{write}(t+1) \cup \mathsf{read}(t+1)}]$$

Specifically, the reduction samples $(\mathbf{E}'' \mid \mathbf{E}''') \leftarrow \mathcal{D}_{\mathbb{Z}^{(S - L_{t+1}) \times 2m}, \chi_{\mathsf{pre}}}$ and defines $\mathbf{S}_{t+1}(\mathbf{B}_1 \mid \mathbf{W}_1) + (\mathbf{E}_{t+1}, \mathbf{E}'_{t+1})$ via

$$\begin{cases} \big(\mathbf{S}_t[\overline{\mathsf{read}(t+1)}] \cdot (\mathbf{B}_1 \mid \mathbf{W}_1) + (\mathbf{E} \mid \mathbf{E}')\big)[j] & \text{for } j \in [\overline{\mathsf{read}(t+1) \cup \mathsf{write}(t+1)}] \\ \big(\mathbf{S}_{t+1}[\mathsf{read}(t+1) \cup \mathsf{write}(t+1)](\mathbf{B}_1 \mid \mathbf{W}_1) + (\mathbf{E}'' \mid \mathbf{E}''')\big)[j] & \text{for } j \in [\mathsf{read}(t+1) \cup \mathsf{write}(t+1)] \end{cases}$$

ii) for any t', given

$$\mathbf{S}_{t'}(\mathbf{B}_1 \mid \mathbf{W}_1) + (\mathbf{E}_{t'} \mid \mathbf{E}'_{t'}) \quad \text{and} \quad \mathbf{S}_{t'+1}[\mathsf{read}(t'+1) \cup \mathsf{write}(t'+1)]$$

where $(\mathbf{E}_{t'} \mid \mathbf{E}'_{t'}) \sim \mathcal{D}_{\mathbb{Z}^{S \times 2m}, \chi_{\mathsf{pre}}}$, one can derive

$$\mathbf{S}_{t'+1}(\mathbf{B}_1 \mid \mathbf{W}_1) + (\mathbf{E}_{t'+1} \mid \mathbf{E}'_{t'+1}),$$

where $(\mathbf{E}_{t'+1} \mid \mathbf{E}'_{t'+1}) \sim \mathcal{D}_{\mathbb{Z}^{S \times 2m}, \chi_{\mathsf{pre}}}$. This follows from the same fact from before, i.e., that by definition

$$\mathbf{S}_{t'+1}[\overline{\mathsf{write}(t'+1) \cup \mathsf{read}(t'+1)}] = \mathbf{S}_{t'}[\overline{\mathsf{write}(t'+1) \cup \mathsf{read}(t'+1)}]$$

Specifically, the reduction samples $(\mathbf{E}'' \mid \mathbf{E}''') \leftarrow \mathcal{D}_{\mathbb{Z}^{(S - L_{t'+1}) \times 2m}, \chi_{\mathsf{pre}}}$ and defines $\mathbf{S}_{t+1}(\mathbf{B}_1 \mid \mathbf{W}_1) + (\mathbf{E}_{t+1}, \mathbf{E}'_{t+1})$ via

$$\begin{cases} \big(\mathbf{S}_{t'}(\mathbf{B}_1 \mid \mathbf{W}_1) + (\mathbf{E}_{t'} \mid \mathbf{E}'_{t'})\big)[j] & \text{for } j \in [\overline{\mathsf{read}(t'+1) \cup \mathsf{write}(t'+1)}] \\ \big(\mathbf{S}_{t'+1}[\mathsf{read}(t'+1) \cup \mathsf{write}(t'+1)](\mathbf{B}_1 \mid \mathbf{W}_1) + (\mathbf{E}'' \mid \mathbf{E}''')\big)[j] & \text{for } j \in [\mathsf{read}(t'+1) \cup \mathsf{write}(t'+1)] \end{cases}$$

In other words, given the elements in Eq. (15), the reduction can produce $\mathbf{S}_{t'}(\mathbf{B}_1 \mid \mathbf{W}_1) + (\mathbf{E}_{t'} \mid \mathbf{E}'_{t'})$, $\mathbf{S}_{t'}(\mathbf{B}_2 \mid \mathbf{W}_2) + (\mathbf{F}_{t'} \mid \mathbf{F}'_{t'})$, and $\odot[\![\mathbf{S}_{t'}, \mathbf{D}_{\mathsf{circ}}]\!]$, where $(\mathbf{E}_{t'} \mid \mathbf{E}'_{t'}), (\mathbf{F}_{t'} \mid \mathbf{F}'_{t'}) \sim \mathcal{D}_{\mathbb{Z}^{S \times 2m}, \chi_{\mathsf{pre}}}$, for each $t' \geq t+1$,

5. it produces $(\mathbf{C}_{t',1}, \mathbf{C}'_{t',1})$ for $t' = t+2, \ldots, T$, using $\mathbf{S}_{t'}(\mathbf{B}_1 \mid \mathbf{W}_1) + (\mathbf{E}_{t'} \mid \mathbf{E}'_{t'})$, for $t' \geq t+1$,
6. it produces $(\mathbf{C}_{t',2}, \mathbf{C}'_{t',2})$ for $t' = t+2, \ldots, T$, using $\mathbf{S}_{t'}(\mathbf{B}_2 \mid \mathbf{W}_2) + (\mathbf{F}_{t'} \mid \mathbf{F}'_{t'})$, for $t' \geq t+1$,
7. it produces $(\mathring{\mathbf{C}}_{t'}, \mathring{\mathbf{C}}'_{t'})$ for $t' = t+2, \ldots, T$, using $\odot[\![\mathbf{S}_t[\overline{\mathsf{read}(t+1)}], \mathbf{D}_{\mathsf{circ}}]\!]$, for $t' \geq t+1$,
8. it produces $(\mathbf{C}_{t',3}, \mathbf{C}'_{t',3}), (\hat{\mathbf{C}}_{t'}, \hat{\mathbf{C}}'_{t'})$ for $t' = t+1, \ldots, T$, and $(\mathbf{C}_{T,4}, \mathbf{C}'_{T,4})$ as in Enc,
9. it computes $\tilde{\mathbf{C}}^{(j)}_{t',1}, \tilde{\mathbf{C}}^{(j)}_{t'}[\mathsf{write}(t) \cup \mathsf{read}(t)]$, for $t' = t+1, \ldots, T$, as in Eq. (13).

We conclude that $\mathsf{H}'_t \approx_c \mathsf{H}_t$ as required. □

Putting Lemmas 6 to 10 together yields the claimed result after noticing that the distribution in H_T is independent of security experiment bit b. □

Acknowledgments. We thank Junqing Gong for numerous helpful and insightful discussions back in 2021, Ji Luo for helpful clarifications regarding [HLL23], as well as the Eurocrypt reviewers for their feedback on the write-up.

References

ACPS09. Benny Applebaum, David Cash, Chris Peikert, and Amit Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 595–618. Springer, Berlin, Heidelberg, August 2009. 33

Ajt96. Miklós Ajtai. Generating hard instances of lattice problems (extended abstract). In *28th ACM STOC*, pages 99–108. ACM Press, May 1996. 33

AKY24. Shweta Agrawal, Simran Kumari, and Shota Yamada. Attribute based encryption for turing machines from lattices. In Leonid Reyzin and Douglas Stebila, editors, *CRYPTO 2024, Part III*, volume 14922 of *LNCS*, pages 352–386. Springer, Cham, August 2024. 1, 2, 31

AMVY21. Shweta Agrawal, Monosij Maitra, Narasimha Sai Vempati, and Shota Yamada. Functional encryption for Turing machines with dynamic bounded collusion from LWE. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part IV*, volume 12828 of *LNCS*, pages 239–269, Virtual Event, August 2021. Springer, Cham. 2, 31, 32

AMY19a. Shweta Agrawal, Monosij Maitra, and Shota Yamada. Attribute based encryption (and more) for nondeterministic finite automata from LWE. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 765–797. Springer, Cham, August 2019. 1, 2, 31, 32

AMY19b. Shweta Agrawal, Monosij Maitra, and Shota Yamada. Attribute based encryption for deterministic finite automata from DLIN. In Dennis Hofheinz and Alon Rosen, editors, *TCC 2019, Part II*, volume 11892 of *LNCS*, pages 91–117. Springer, Cham, December 2019. 1

AMYY25. Shweta Agrawal, Anuja Modi, Anshu Yadav, and Shota Yamada. Zeroizing attacks against evasive and circular evasive LWE. In Benny Applebaum and Huijia (Rachel) Lin, editors, *TCC 2025, Part II*, volume 16269 of *LNCS*, pages 259–290. Springer, Cham, December 2025. 13

AS17. Shweta Agrawal and Ishaan Preet Singh. Reusable garbled deterministic finite automata from learning with errors. In Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl, editors, *ICALP 2017*, volume 80 of *LIPIcs*, pages 36:1–36:13. Schloss Dagstuhl, July 2017. 1

BGG+14. Dan Boneh, Craig Gentry, Sergey Gorbunov, Shai Halevi, Valeria Nikolaenko, Gil Segev, Vinod Vaikuntanathan, and Dhinakaran Vinayagamurthy. Fully key-homomorphic encryption, arithmetic circuit ABE and compact garbled circuits. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 533–556. Springer, Berlin, Heidelberg, May 2014. 1, 2, 11

BÜW24. Chris Brzuska, Akin Ünal, and Ivy K. Y. Woo. Evasive LWE assumptions: Definitions, classes, and counterexamples. In Kai-Min Chung and Yu Sasaki, editors, *ASIACRYPT 2024, Part IV*, volume 15487 of *LNCS*, pages 418–449. Springer, Singapore, December 2024. 13

DHM+24. Fangqi Dong, Zihan Hao, Ethan Mook, Hoeteck Wee, and Daniel Wichs. Laconic function evaluation and ABE for RAMs from (ring-)LWE. In Leonid Reyzin and Douglas Stebila, editors, *CRYPTO 2024, Part III*, volume 14922 of *LNCS*, pages 107–142. Springer, Cham, August 2024. 1

DJM+25. Nico Döttling, Abhishek Jain, Giulio Malavolta, Surya Mathialagan, and Vinod Vaikuntanathan. Simple and general counterexamples for private-coin evasive LWE. In Yael Tauman Kalai and Seny F. Kamara, editors, *CRYPTO 2025, Part VII*, volume 16006 of *LNCS*, pages 73–92. Springer, Cham, August 2025. 13

GKP+13. Shafi Goldwasser, Yael Tauman Kalai, Raluca A. Popa, Vinod Vaikuntanathan, and Nickolai Zeldovich. How to run Turing machines on encrypted data. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 536–553. Springer, Berlin, Heidelberg, August 2013. 1

GKPV10. Shafi Goldwasser, Yael Tauman Kalai, Chris Peikert, and Vinod Vaikuntanathan. Robustness of the learning with errors assumption. In Andrew Chi-Chih Yao, editor, *ICS 2010*, pages 230–240. Tsinghua University Press, January 2010. 11

GKW16. Rishab Goyal, Venkata Koppula, and Brent Waters. Semi-adaptive security and bundling functionalities made generic and easy. In Martin Hirt and Adam D. Smith, editors, *TCC 2016-B, Part II*, volume 9986 of *LNCS*, pages 361–388. Springer, Berlin, Heidelberg, October / November 2016. 31

GPSW06. Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. In Ari Juels, Rebecca N. Wright, and Sabrina De Capitani di Vimercati, editors, *ACM CCS 2006*, pages 89–98. ACM Press, October / November 2006. Available as Cryptology ePrint Archive Report 2006/309. 1

GPV08. Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In Richard E. Ladner and Cynthia Dwork, editors, *40th ACM STOC*, pages 197–206. ACM Press, May 2008. 33

GVW13. Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee. Attribute-based encryption for circuits. In Dan Boneh, Tim Roughgarden, and Joan Feigenbaum, editors, *45th ACM STOC*, pages 545–554. ACM Press, June 2013. 1

GWW19. Junqing Gong, Brent Waters, and Hoeteck Wee. ABE for DFA from k-Lin. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 732–764. Springer, Cham, August 2019. 1

HJL25. Yao-Ching Hsieh, Aayush Jain, and Huijia Lin. Lattice-based post-quantum iO from circular security with random opening assumption. In Yael Tauman Kalai and Seny F. Kamara, editors, *CRYPTO 2025, Part VII*, volume 16006 of *LNCS*, pages 3–38. Springer, Cham, August 2025. 13

HLL23. Yao-Ching Hsieh, Huijia Lin, and Ji Luo. Attribute-based encryption for circuits of unbounded depth from lattices: Garbled circuits of optimal size, laconic functional evaluation, and more. In *FOCS*, 2023. 1, 2, 8, 11, 13, 30, 32

HLL24. Yao-Ching Hsieh, Huijia Lin, and Ji Luo. A general framework for lattice-based ABE using evasive inner-product functional encryption. In Marc Joye and Gregor Leander, editors, *EUROCRYPT 2024, Part II*, volume 14652 of *LNCS*, pages 433–464. Springer, Cham, May 2024. 1, 2, 8, 31, 32

LL20. Huijia Lin and Ji Luo. Compact adaptively secure ABE from k-Lin: Beyond NC^1 and towards NL. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part III*, volume 12107 of *LNCS*, pages 247–277. Springer, Cham, May 2020. 1

MP12. Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 700–718. Springer, Berlin, Heidelberg, April 2012. 33

PF79. Nicholas Pippenger and Michael J. Fischer. Relations among complexity measures. *J. ACM*, 26(2):361–381, 1979. 4

SW05. Amit Sahai and Brent R. Waters. Fuzzy identity-based encryption. In Ronald Cramer, editor, *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 457–473. Springer, Berlin, Heidelberg, May 2005. 1

Tsa19. Rotem Tsabary. Fully secure attribute-based encryption for t-CNF from LWE. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part I*, volume 11692 of *LNCS*, pages 62–85. Springer, Cham, August 2019. 8

Tsa22. Rotem Tsabary. Candidate witness encryption from lattice techniques. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part I*, volume 13507 of *LNCS*, pages 535–559. Springer, Cham, August 2022. 1

Wat12. Brent Waters. Functional encryption for regular languages. In Reihaneh Safavi-Naini and Ran Canetti, editors, *CRYPTO 2012*, volume 7417 of *LNCS*, pages 218–235. Springer, Berlin, Heidelberg, August 2012. 1

Wee21. Hoeteck Wee. ABE for DFA from LWE against bounded collusions, revisited. In Kobbi Nissim and Brent Waters, editors, *TCC 2021, Part II*, volume 13043 of *LNCS*, pages 288–309. Springer, Cham, November 2021. 1

Wee22. Hoeteck Wee. Optimal broadcast encryption and CP-ABE from evasive lattice assumptions. In Orr Dunkelman and Stefan Dziembowski, editors, *EUROCRYPT 2022, Part II*, volume 13276 of *LNCS*, pages 217–241. Springer, Cham, May / June 2022. 1, 8, 12, 13, 32

Wee24. Hoeteck Wee. Circuit ABE with $\mathsf{poly}(\mathsf{depth}, \lambda)$-sized ciphertexts and keys from lattices. In Leonid Reyzin and Douglas Stebila, editors, *CRYPTO 2024, Part III*, volume 14922 of *LNCS*, pages 178–209. Springer, Cham, August 2024. 1

A Prior ABE for TM

While the prior lattice-based ABE for TMs in AKY24 [AKY24] as well as that for log-space TM (L) in HLL24 [HLL24] are conceptually simple, modular and elegant, the final schemes themselves are actually quite complex due to the composition of multiple highly technical building blocks. Notably, the 81-page full version of HLL24 only spells out the ABE scheme for DFA (amongst other results), along with a single sentence "starting from the linear secret sharing scheme [LL20a] for L, we also obtain ABE for L via the same construction", even though ABE for L is mentioned in the abstract.[11] Similarly, the camera-ready version of [AKY24] only has room for the underlying CP-ABE for circuits. Here, we provide a brief exposition of these schemes to facilitate a comparison with ours.

AKY24 scheme for TM [AKY24]. The AKY24 ABE for TM follows the template laid out in [AMVY21, Section 6] (and [AMY19a]), and uses a KP-ABE for bounded-depth circuits and a CP-ABE for unbounded-depth circuits as building blocks, along with the "bundling" compiler from [GKW16].

The first idea is that the encryptor would "unroll" the entire TM computation of time T on attribute x into a circuit $U_{x,T}$ that on input a TM M, outputs 1 if M accepts x within T steps. Note that the circuit $U_{x,T}$ has size

[11] The paper does not contain a description of the noisy linear garbling for L (a building block for ABE for L), nor its analysis. Admittedly, we do not present a self-contained description of ABE for TM either, but we do so for iterated computation, which is a stronger model. DFA is weaker than L.

$\tilde{O}(T \cdot |M|)$ and depth T. The encryptor will then encrypt $U_{x,T}$ and the message μ using a ciphertext policy ABE (CP-ABE). The key technical novelty in AKY24 lies on building a new CP-ABE for unbounded-depth circuits, which relies on circular evasive LWE and circular tensor LWE. Applying the ensuing CP-ABE to the circuit $U_{x,T}$ yields encryption time $\tilde{O}(T \cdot |M|)$ and key size $O(|M|)$.

So far, encryption requires knowing $|M|$. Instead, following [AMVY21, AMY19a], the encryptor would generate a fresh CP-ABE ciphertexts for all possible choices of $|M|$ from 1 to T; the case where $|M| > T$ is handled using a separate KP-ABE scheme. This blows up the encryption time to $O(T(1 + \cdots + T)) = O(T^3)$.

In more detail,

- Encryption on input $(x, 1^T)$ computes CP-ABE encryptions for $O(T)$ circuits given by

$$U_{i,x,T} : \{0,1\}^i \to \{0,1\}, \quad i \in [|(x, 1^T)| - 1]$$

  where $U_{i,x,T}$ takes as input a TM $M \in \{0,1\}^i$, and accepts if M accepts input x within T steps. The circuit $U_{i,x,T}$ has size $O(i \cdot T)$. In addition, encryption also computes a KP-ABE encryption of $(x, 1^T)$.
- Key generation on input M, generates a CP-ABE key for M (as a bit-string) as well as KP-ABE keys for $O(|M|)$ circuits given by

$$U'_{i,M} : \{0,1\}^i \to \{0,1\}, \quad i \in [|M|]$$

  where $U'_{i,M}$ takes as input $(x, 1^T)$ and accepts if M accepts input x within T steps. The circuit $U'_{i,M}$ has size $O(i \cdot |M|)$.
- To decrypt, we run CP-ABE decryption when $|(x, 1^T)| > |M|$ and KP-ABE decryption $|(x, 1^T)| \leq |M|$.

In particular,

- encryption takes time $\sum_{i=1}^{T} O(i \cdot T) = O(T^3)$ and produces a ciphertext of size $O(T^2)$;
- key generation takes time $\sum_{i=1}^{|M|} O(i \cdot |M|) = O(|M|^3)$ and produces a key of size $O(|M|^2)$;
- decryption takes time $O(T \cdot |M|)$.

The scheme relies on circular evasive LWE as well as tensor LWE; these assumptions are inherited from the prior KP-ABE and CP-ABE schemes in [HLL23, Wee22]. We clarify that the main goal in AKY24 was feasibility, and the authors did not optimize on the concrete complexity of the scheme.

- For instance, it seems quite plausible that we can reduce encryption time to $O(T^2 \log T)$ and key generation time to $O(|M|^2 \log |M|)$ by only enumerating over i that are powers of 2 in $[|(x, 1^T)| - 1]$ and in $[|M|]$ respectively.
- By modifying the case analysis, we could hope to reduce the encryption time to $\tilde{O}(T^{1+\delta})$ while increasing key generation time to $\tilde{O}(|M|^{1+1/\delta})$ for any $\delta > 0$.

Either way, the technique seems to inherently incur super-linear running times for both encryption and key generation, in order to compute the circuits $U_{T^\delta,x,T}$ and $U'_{|M|^{1/\delta},M}$ respectively.

HLL24 scheme for log-space TM [HLL24]. The decryption invariant in the HLL24 ABE for log-space TM with Q states is as follows: at step $t = 0, 1, \ldots, T$, if the machine M is at state $u \in [Q]$ and the contents of the work tape is $\mathbf{w} \in \{0,1\}^S$, then the decryptor computes

$$\mathbf{s}_{t,\mathbf{w}} \mathbf{d}_u$$

In particular, the encryptor samples $T \cdot 2^S$ LWE secrets, one for each time step and every possible configuration of the work tape, and key generation samples Q vectors $\mathbf{d}_u$, one each state. In our construction, we compute a dual attribute encoding where we treat $(\mathbf{w}, u)$ as an attribute of length at most $S + \lambda$ bits.

B Additional Preliminaries

Trapdoor and preimage sampling. Let $n, q \in \mathbb{Z}$,

$$\mathbf{g} = (1, 2, 4, \ldots, 2^{\lceil \log q \rceil - 1}) \in \mathbb{Z}^{\lceil \log q \rceil}.$$

The gadget matrix $\mathbf{G}$ is defined as the diagonal concatenation of $\mathbf{g}$ n times. Formally, $\mathbf{G} = \mathbf{g} \otimes \mathbf{I}_n \in \mathbb{Z}^{n \times n \cdot \lceil \log q \rceil}$. For any $t \in \mathbb{Z}$, the function $\mathbf{G}^{-1} : \mathbb{Z}_q^{n \times t} \to \{0,1\}^{n \cdot \lceil \log q \rceil \times t}$ expands each entry $a \in \mathbb{Z}_q$ of the input matrix into a column of size $\lceil \log q \rceil$ consisting of the bit-representation of a. For any matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times t}$ it holds that $\mathbf{G} \cdot \mathbf{G}^{-1}(\mathbf{A}) = \mathbf{A} \bmod q$.

Let $n, m, q \in \mathbb{N}$ and consider a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$. For all $\mathbf{V} \in \mathbb{Z}_q^{n \times m'}$ we let $\mathbf{A}^{-1}(\mathbf{V}, \tau)$ denote the random variable whose distribution is the discrete Gaussian $\mathcal{D}_{\mathbb{Z}^{m \times m'}, \tau}$ conditioned on $\mathbf{A} \cdot \mathbf{A}^{-1}(\mathbf{V}, \tau) = \mathbf{V} \bmod q$. If $\mathbf{Y} \leftarrow \mathbf{A}^{-1}(\mathbf{V}, \tau)$ then $\|\mathbf{Y}\| \leq \lambda \cdot \tau$ with probability at least $1 - e^{-\Omega(\lambda^2)}$. When using this bound, for the sake of ease of exposition, we will omit the term λ. All bounds can be modified to account for this extra λ factor.

Lemma 11 (Trapdoor Generation and Sampling [Ajt96, GPV08, MP12]). *There exists a pair of probabilistic polynomial-time algorithms:*

- $\mathsf{TrapGen}(1^n, 1^m, q)$ *that for all* $m \geq m_0 = m_0(n, q) = O(n \log q)$*, outputs* $(\mathbf{A}, \mathbf{T}_\mathbf{A})$ *s.t.* $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ *is within statistical distance* 2^{-n} *from uniform and* $\mathbf{T}_\mathbf{A}$ *is a* τ*-trapdoor for* $\mathbf{A}$ *where* $\tau = O(\sqrt{n} \cdot \log q \cdot \log n)$*.*
- $\mathsf{SamplePre}(\mathbf{A}, \mathbf{T}, \mathbf{V}, \tau)$ *that given* $\mathbf{A}$ *and any* τ*-trapdoor* $\mathbf{T}$ *of* $\mathbf{A}$*, outputs a sample from* $\mathbf{A}^{-1}(\mathbf{V}, \tau)$*.*

Learning with Errors (LWE). Given $n, m, q, \chi_e \in \mathbb{N}$, the $\mathsf{LWE}_{n,q,\chi_e}$ assumption states that

$$(\mathbf{A}, \mathbf{s}\mathbf{A} + \mathbf{e}) \approx_c (\mathbf{A}, \mathbf{u}),$$

where

$$\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}, \mathbf{s} \leftarrow \mathbb{Z}_q^n, \mathbf{e} \leftarrow \mathcal{D}_{\mathbb{Z}^m, \chi_e}, \mathbf{u} \leftarrow \mathbb{Z}_q^m.$$

An analog assumption holds when the secret $\mathbf{s}$ itself is chosen from a Gaussian distribution [ACPS09]. When $\mathbf{s} \leftarrow \mathcal{D}_{\mathbb{Z}^n, \chi_e}$, we refer to such assumption as the $\mathsf{sLWE}_{n,q,\chi_e}$ assumption.

Circular Small-Secret Learning with Errors. Given $n, m, m', q, \chi, \chi' \in \mathbb{N}$, the $\mathsf{cssLWE}_{n,q,\chi,\chi'}$ assumption states that

$$\left(\begin{bmatrix} \mathbf{A}' \\ \mathbf{s}\mathbf{A}' + \hat{\mathbf{e}} \end{bmatrix} - \mathsf{bits}(\mathbf{s}') \otimes \mathbf{G}, \mathbf{A}, \mathbf{s}'\mathbf{A} + \mathbf{e} \right) \approx_c (\mathbf{U}, \mathbf{A}, \mathbf{u}),$$

where

$$\begin{array}{lll} \mathbf{A}' \leftarrow \mathbb{Z}_q^{(n-1) \times nm\lceil \log q \rceil}, & \mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m'}, & \mathbf{s} \leftarrow \mathcal{D}_{\mathbb{Z}^{n-1}, \chi}, \\ \hat{\mathbf{e}} \leftarrow \mathcal{D}_{\mathbb{Z}^{nm\lceil \log q \rceil}, \chi} & \mathbf{e} \leftarrow \mathcal{D}_{\mathbb{Z}^{m'}, \chi'}, & \\ \mathbf{u} \leftarrow \mathbb{Z}_q^{m'}, & \mathbf{U} \leftarrow \mathbb{Z}_q^{n \times nm\lceil \log q \rceil}, & \end{array}$$

and $\mathbf{s}' := [\mathbf{s}, -1]$.

C Recoding: Proofs

Lemma 3 (recoding dual-to-encoding correctness). *Suppose*

$$\begin{aligned} \mathbf{C}_{\mathsf{in}} &= [\![\mathbf{S}, \mathbf{A}_0, \mathbf{z}]\!] \\ (\mathbf{C}, \mathbf{C}') &= [\![\mathbf{S} \mapsto \mathbf{s}]\!]_{\mathbf{B}, \mathbf{W}} \\ (\mathbf{I}_L \otimes \mathbf{B}) \cdot \mathbf{K} &= [\![\mathbf{A}_0 \mapsto \mathbf{A}]\!]_{\mathbf{W}}. \end{aligned}$$

Then,

$$[\![\mathbf{s}, \mathbf{A}, \mathbf{z}]\!] = \mathsf{rDec}(\mathbf{B}, \mathbf{W}, \mathbf{A}, \mathbf{K}, \mathbf{C}_{\mathsf{in}} + \mathbf{E}_{\mathsf{in}}, (\mathbf{C} + \mathbf{E}, \mathbf{C}' + \mathbf{E}'), \mathbf{z}) + \mathbf{E}_{\mathsf{out}},$$

where $\|\mathbf{E}_{\mathsf{out}}\| \leq m \cdot (2\|\mathbf{E}'\| + \|\mathbf{E}_{\mathsf{in}}\| + \|\mathbf{E}\|\|\mathbf{K}\|)$.

Proof. We have

$$\begin{aligned}\mathsf{rDec}_0\big(\mathbf{B},\mathbf{W},\mathbf{A},\mathbf{C}_{\mathsf{in}},\mathbf{C}',\mathbf{z}\big) &= \mathsf{diag}_L(\mathbf{C}')(\mathbf{I}_L\otimes\mathbf{G})^{-1}(\mathbf{A}) - \mathsf{diag}(\mathbf{z})\mathbf{C}' - \mathbf{C}_{\mathsf{in}}\mathbf{G}^{-1}(\mathbf{W})\\ &= \mathsf{diag}_L(\mathbf{S})(\mathbf{I}_L\otimes\mathbf{W})(\mathbf{I}_L\otimes\mathbf{G})^{-1}(\mathbf{A}) + (\mathbf{I}_L\otimes\mathbf{s})\mathbf{A}\\ &\quad - \mathsf{diag}(\mathbf{z})\mathbf{S}\mathbf{W} - \mathbf{z}^\top\cdot\mathbf{s}\mathbf{G}\\ &\quad - \mathbf{S}\mathbf{A}_0\mathbf{G}^{-1}(\mathbf{W}) + \mathsf{diag}(\mathbf{z})\mathbf{S}\mathbf{W}\\ &= [\![\mathbf{s},\mathbf{A},\mathbf{z}]\!] - \mathsf{diag}_L(\mathbf{S})([\![\mathbf{A}_0\mapsto\mathbf{A}]\!]_{\mathbf{W}}).\end{aligned}$$

The claim follows after noticing that $\mathsf{diag}_L(\mathbf{C})\cdot\mathbf{K} = \mathsf{diag}_L(\mathbf{S})([\![\mathbf{A}_0\mapsto\mathbf{A}]\!]_{\mathbf{W}})$ and that

$$\mathbf{E}_{\mathsf{out}} = \mathsf{diag}_L(\mathbf{E}')(\mathbf{I}_L\otimes\mathbf{G})^{-1}(\mathbf{A}) - \mathsf{diag}(\mathbf{z})\mathbf{E}' - \mathbf{E}_{\mathsf{in}}\mathbf{G}^{-1}(\mathbf{W}) + \mathsf{diag}_L(\mathbf{E})\mathbf{K}.$$

Lemma 4 (recoding dual-to-dual correctness). *Suppose*

$$\begin{aligned}\mathbf{C}_{\mathsf{in}} &= [\![\mathbf{S},\mathbf{A}_0,\mathbf{z}]\!]\\ (\mathbf{C},\mathbf{C}') &= [\![\mathbf{S}\mapsto\mathbf{S}']\!]_{\mathbf{B},\mathbf{W}}\\ \mathbf{B}\cdot\mathbf{K} &= [\![\mathbf{A}_0\mapsto\mathbf{A}_0']\!]_{\mathbf{W}}\end{aligned}$$

Then,

$$[\![\mathbf{S}',\mathbf{A}_0',\mathbf{z}]\!] = \mathsf{rDec}\big(\mathbf{B},\mathbf{W},\mathbf{A}_0',\mathbf{K},\mathbf{C}_{\mathsf{in}}+\mathbf{E}_{\mathsf{in}},(\mathbf{C}+\mathbf{E},\mathbf{C}'+\mathbf{E}'),\mathbf{z}\big) + \mathbf{E}_{\mathsf{out}}.$$

where $\|\mathbf{E}_{\mathsf{out}}\| \le m\cdot(2\|\mathbf{E}'\| + \|\mathbf{E}_{\mathsf{in}}\| + \|\mathbf{E}\|\|\mathbf{K}\|)$.

Proof. We have

$$\begin{aligned}\mathsf{rDec}_0\big(\mathbf{B},\mathbf{W},\mathbf{A},\mathbf{C}_{\mathsf{in}},\mathbf{C}',\mathbf{z}\big) &= \mathbf{C}'\mathbf{G}^{-1}(\mathbf{A}_0') - \mathsf{diag}(\mathbf{z})\mathbf{C}' - \mathbf{C}_{\mathsf{in}}\mathbf{G}^{-1}(\mathbf{W})\\ &= \mathbf{S}\mathbf{W}\mathbf{G}^{-1}(\mathbf{A}_0') + \mathbf{S}'\mathbf{A}_0'\\ &\quad - \mathsf{diag}(\mathbf{z})\mathbf{S}\mathbf{W} - \mathsf{diag}(\mathbf{z})\mathbf{S}'\mathbf{G}\\ &\quad - \mathbf{S}\mathbf{A}_0\mathbf{G}^{-1}(\mathbf{W}) + \mathsf{diag}(\mathbf{z})\mathbf{S}\mathbf{W}\\ &= [\![\mathbf{S}',\mathbf{A}_0',\mathbf{z}]\!] - \mathbf{S}([\![\mathbf{A}_0\mapsto\mathbf{A}_0']\!]_{\mathbf{W}})\end{aligned}$$

The claim follows after noticing that $\mathbf{C}\mathbf{K} = \mathbf{S}([\![\mathbf{A}_0\mapsto\mathbf{A}_0']\!]_{\mathbf{W}})$ and that

$$\mathbf{E}_{\mathsf{out}} = \mathbf{E}'\mathbf{G}^{-1}(\mathbf{A}_0') - \mathsf{diag}(\mathbf{z})\mathbf{E}' - \mathbf{E}_{\mathsf{in}}\mathbf{G}^{-1}(\mathbf{W}) + \mathbf{E}\mathbf{K}.$$

Lemma 5 (recoding encoding-to-dual correctness). *Suppose*

$$\begin{aligned}\mathbf{C}_{\mathsf{in}} &= [\![\mathbf{s},\mathbf{A},\mathbf{z}]\!]\\ (\mathbf{C},\mathbf{C}') &= [\![\mathbf{s}\mapsto\mathbf{S}]\!]_{\mathbf{B},\mathbf{W}'}\\ (\mathbf{I}_L\otimes\mathbf{B})\ \cdot\mathbf{K} &= [\![\mathbf{A}\mapsto\mathbf{A}_0]\!]_{\mathbf{W}'}\end{aligned}$$

Then,

$$[\![\mathbf{S},\mathbf{A}_0,\mathbf{z}]\!] = \mathsf{rDec}(\mathbf{B},\mathbf{W},\mathbf{A}_0,\mathbf{K},\mathbf{C}_{\mathsf{in}}+\mathbf{E}_{\mathsf{in}},(\mathbf{C}+\mathbf{E},\mathbf{C}'+\mathbf{E}'),\mathbf{z}) + \mathbf{E}_{\mathsf{out}},$$

where $\|\mathbf{E}_{\mathsf{out}}\| \le m\cdot(2\|\mathbf{E}'\| + \|\mathbf{E}_{\mathsf{in}}\| + \|\mathbf{E}\|\|\mathbf{K}\|)$.

Proof. We have

$$\begin{aligned}\mathsf{rDec}_0\big(\mathbf{B},\mathbf{W},\mathbf{A},\mathbf{C}_{\mathsf{in}},\mathbf{C}',\mathbf{z}\big) &= \mathbf{C}'\mathbf{G}^{-1}(\mathbf{A}_0) - \mathsf{diag}(\mathbf{z})\mathbf{C}' - \mathsf{diag}_L(\mathbf{C}_{\mathsf{in}})(\mathbf{I}_L\otimes\mathbf{G})^{-1}(\mathbf{W}')\\ &= (\mathbf{I}_L\otimes\mathbf{s})\mathbf{W}'\mathbf{G}^{-1}(\mathbf{A}_0) + \mathbf{S}\mathbf{A}_0\\ &\quad - \mathsf{diag}(\mathbf{z})(\mathbf{I}_L\otimes\mathbf{s})\mathbf{W}' - \mathsf{diag}(\mathbf{z})\mathbf{S}\mathbf{G}\\ &\quad - (\mathbf{I}_L\otimes\mathbf{s})\mathsf{diag}_L(\mathbf{A})(\mathbf{I}_L\otimes\mathbf{G})^{-1}(\mathbf{W}') + \mathsf{diag}(\mathbf{z})(\mathbf{I}_L\otimes\mathbf{s})\mathbf{W}'\\ &= [\![\mathbf{S},\mathbf{A}_0,\mathbf{z}]\!] - (\mathbf{I}_L\otimes\mathbf{s})([\![\mathbf{A}\mapsto\mathbf{A}_0]\!]_{\mathbf{W}'}).\end{aligned}$$

The claim follows after noticing that $(\mathbf{I}_L\otimes\mathbf{C})\cdot\mathbf{K} = (\mathbf{I}_L\otimes\mathbf{s})([\![\mathbf{A}\mapsto\mathbf{A}_0]\!]_{\mathbf{W}'})$ and that

$$\mathbf{E}_{\mathsf{out}} = \mathbf{E}'\mathbf{G}^{-1}(\mathbf{A}_0) - \mathsf{diag}(\mathbf{z})\mathbf{E}' - \mathsf{diag}_L(\mathbf{E}_{\mathsf{in}})(\mathbf{I}_L\otimes\mathbf{G})^{-1}(\mathbf{W}') + (\mathbf{I}_L\otimes\mathbf{E})\mathbf{K}.$$

D Security Proofs

Lemma 6. *If* $\mathsf{H}_{\mathsf{corr}} \approx_c \$$, *then* $\mathsf{H}_{\mathsf{pre}} \approx_c \$$.

Proof. We want to relate the following quantities corresponding to the differences between $\mathsf{H}_{\mathsf{pre}}$ and $\mathsf{H}_{\mathsf{corr}}$:

$$\mathsf{H}_{\mathsf{pre}} : \big(\mathbf{Y}_0^{(j)}, \{\mathbf{Y}_{t,1}^{(j)}, \mathbf{Y}_{t,2}^{(j)}, \mathbf{Y}_{t,3}^{(j)}\}_{t\in[T]}, \mathbf{Y}_{T,4}^{(j)}\big)$$
$$\mathsf{H}_{\mathsf{corr}} : \big(\tilde{\mathbf{C}}_0^{(j)}, \{\tilde{\mathbf{C}}_{t,1}^{(j)}, \tilde{\mathbf{C}}_t^{(j)}[\mathsf{write}(t)\cup\mathsf{read}(t)]\}_{t=1,\dots,T}, \tilde{\mathbf{C}}_{T,4}^{(j)}\big)$$

It suffices to show that:

$$\begin{aligned}
\tilde{\mathbf{C}}_0^{(j)} &\approx_s \mathbf{Y}_0^{(j)} + \mathbf{C}'_0\\
\tilde{\mathbf{C}}_{t,1}^{(j)} &\approx_s \mathbf{Y}_{t,1}^{(j)} + \mathsf{rDec}_0\big(\mathbf{B}_1, \mathbf{W}_1, \mathbf{D}^{(j)}, \tilde{\mathbf{C}}_{t-1}^{(j)}[\mathsf{read}(t)], \mathbf{C}'_{t,1}, \mathbf{z}_{t-1}^{(j)}[\mathsf{read}(t)]\big)\\
\tilde{\mathbf{C}}_t^{(j)}[\overline{\mathsf{write}(t)}\cap\mathsf{read}(t)] &\approx_s \mathbf{Y}_{t,2}^{(j)} + \mathsf{rDec}_0\big(\mathbf{B}_2, \mathbf{W}_2, \mathbf{A}_0^{(j)}, \tilde{\mathbf{C}}_{t,2}^{(j)}, \mathbf{C}'_{t,2}, \mathbf{z}_{t-1}^{(j)}[\overline{\mathsf{write}(t)}\cap\mathsf{read}(t)]\big)\\
\tilde{\mathbf{C}}_t^{(j)}[\mathsf{write}(t)] &\approx_s \mathbf{Y}_{t,3}^{(j)} + \mathsf{rDec}_0\big(\mathbf{B}_3, \mathbf{W}_3, \mathbf{A}_0^{(j)}, \tilde{\mathbf{C}}_{t,3}^{(j)}, \mathbf{C}'_{t,3}, f(\mathbf{z}_{t-1}^{(j)}[\mathsf{read}(t)])\big)\\
\tilde{\mathbf{C}}_{T,4}^{(j)} &\approx_s \mathbf{Y}_{T,4}^{(j)} - \mathbf{C}'_{T,4} - \tilde{\mathbf{C}}_T^{(j)}[1]\mathbf{R}^{(j)}
\end{aligned}$$

where $\tilde{\mathbf{C}}_{t,2}^{(j)}, \tilde{\mathbf{C}}_{t,3}^{(j)}$ are computed as in the decryption algorithm, i.e., Eq. (14).

Then, it follows that if the terms on the LHS are jointly pseudorandom, then the first summands on the RHS are jointly pseudorandom.

From Lemmas 3 to 5, which bound the error term of the output of the rDec algorithms, and noise flooding (Lemma 1), it follows that the two distributions are statistically close as long as

$$\begin{aligned}
\chi'' &\geq \lambda^{\omega(1)}\cdot m(\chi'\tau + \chi_{\mathsf{pre}})\\
&= \lambda^{\omega(1)}\cdot\max\{m(\|\tilde{\mathbf{E}}_{t-1}^{(j)}[\mathsf{read}(t)]\| + \|\mathbf{E}'_{t,1}\|), \|\mathbf{E}'_{T,4}\| + \|\tilde{\mathbf{E}}_T^{(j)}[1]\mathbf{R}^{(j)}\|\}
\end{aligned} \tag{16}$$

for the $(\mathbf{Y}_{t,1}^{(j)},\tilde{\mathbf{C}}_{t,1}^{(j)})$ and $(\mathbf{Y}_{T,4}^{(j)},\tilde{\mathbf{C}}_{T,4}^{(j)})$ terms, and

$$\begin{aligned}
\chi' &\geq \lambda^{\omega(1)}\cdot m(B_{\mathsf{HLL}}2^{-\lambda} + \chi_{\mathsf{pre}})\\
&\geq \lambda^{\omega(1)}\max\Big\{\|\mathbf{E}'_0\|, m(\|\tilde{\mathbf{E}}_{t,2}^{(j)} + \mathbf{E}'_{t,2}\|), m(\|\tilde{\mathbf{E}}_{t,3}^{(j)} + \mathbf{E}'_{t,3}\|)\Big\},
\end{aligned} \tag{17}$$

for the $(\mathbf{Y}_0^{(j)}, \tilde{\mathbf{C}}_0^{(j)})$, $(\mathbf{Y}_{t,2}^{(j)}, \tilde{\mathbf{C}}^{(j)}[\overline{\mathsf{write}(t)}\cap\mathsf{read}(t)])$ and $(\mathbf{Y}_{t,3}^{(j)}, \tilde{\mathbf{C}}^{(j)}[\mathsf{write}(t)])$ terms, where $B_{\mathsf{HLL}}2^{-\lambda}$ bounds the error terms $\tilde{\mathbf{E}}_{t,2}^{(j)}, \tilde{\mathbf{E}}_{t,3}^{(j)}$ of $\tilde{\mathbf{C}}_{t,2}^{(j)}, \tilde{\mathbf{C}}_{t,3}^{(j)}$ output by UEvalF. □

Lemma 7. $\mathsf{H}_{\mathsf{corr}} \approx_c \mathsf{H}_0$.

Proof. We want to relate the following quantities corresponding to the differences between the two games:

$$\mathsf{H}_{\mathsf{corr}} : \left\{\begin{matrix}\mathbf{C}_0, \mathbf{C}'_0, \tilde{\mathbf{C}}_0^{(j)},\\ \mathbf{C}_{1,1}, \mathbf{C}'_{1,1},\\ \mathbf{C}_{1,2}, \mathbf{C}'_{1,2},\\ \mathring{\mathbf{C}}_1, \mathring{\mathbf{C}}'_1\end{matrix}\right\} = \left\{\begin{matrix}\mathbf{S}_0\mathbf{B}_0 + \mathbf{E}_0, [\![\mathbf{S}_0, \mathbf{A}_{st}, \mathbf{z}_0]\!] + \mathbf{E}'_0, [\![\mathbf{S}_0, \mathbf{A}_0^{(j)}, \mathbf{z}_0^{(j)}]\!] + \tilde{\mathbf{E}}_0^{(j)},\\ [\![\mathbf{S}_0[\mathsf{read}(1)] \mapsto \mathbf{s}'_0]\!]_{\mathbf{B}_1,\mathbf{W}_1} + (\mathbf{E}_{1,1}, \mathbf{E}'_{1,1}),\\ [\![\mathbf{S}_0[\overline{\mathsf{write}(1)}\cap\mathsf{read}(1)] \mapsto \mathbf{S}_1[\overline{\mathsf{write}(1)}\cap\mathsf{read}(1)]]\!]_{\mathbf{B}_2,\mathbf{W}_2} + (\mathbf{E}_{1,2}, \mathbf{E}'_{1,2}),\\ \odot[\![\mathbf{S}_0[\overline{\mathsf{write}(1)}\cap\mathsf{read}(1)], \mathbf{D}_{\mathsf{circ}}]\!]\end{matrix}\right\}$$

$$\mathsf{H}_0 : \left\{\begin{matrix}\mathbf{C}_0, \mathbf{C}'_0, \tilde{\mathbf{C}}_0^{(j)},\\ \mathbf{C}_{1,1}, \mathbf{C}'_{1,1},\\ \mathbf{C}_{1,2}, \mathbf{C}'_{1,2},\\ \mathring{\mathbf{C}}_1, \mathring{\mathbf{C}}'_1\end{matrix}\right\} \leftarrow \left\{\begin{matrix}\mathbb{Z}_q^{S\times m}, \mathbb{Z}_q^{S\times m}, \mathbb{Z}_q^{S\times m},\\ \mathbb{Z}_q^{L\times m}, \mathbb{Z}_q^{L\times m},\\ \mathbb{Z}_q^{(L-L_1)\times m}, \mathbb{Z}_q^{(L-L_1)\times m},\\ (\mathbb{Z}_q^{n\times nm\lceil\log q\rceil}, \mathbb{Z}_q^{(n^2m\lceil\log q\rceil^2+1)m})^{(L-L_1)}\end{matrix}\right\}$$

We proceed as in the proof of Lemma 9 (see below) as if $t = 0$ and $\mathsf{write}(0) = \mathsf{read}(0) = [S]$, except with the additional $\mathbf{C}_0, \mathbf{C}'_0$ terms, i.e., the LWE matrix additionally includes $\mathbf{B}_0$ and $\mathbf{A}_{st}$. Similarly to Lemma 9, we need to rely on Lemma 12 due to the presence of the terms $\mathbf{z}_0, \mathbf{z}_0^{(j)}$. □

Lemma 8. *For $t = 1, \ldots, T$, we have $\mathsf{H}_{t-1} \approx_c \mathsf{H}'_t$.*

Proof. We want to relate the following quantities corresponding to the differences between the two games:

$$\mathsf{H}_{t-1} : \left\{ \begin{array}{c} \tilde{\mathbf{C}}^{(j)}_{t,1}, \\ \mathbf{C}_{t,3}, \mathbf{C}'_{t,3}, \\ \hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t \end{array} \right\} = \left\{ \begin{array}{c} [\![\mathbf{s}'_{t-1}, \mathbf{D}^{(j)}, \mathbf{z}^{(j)}_{t-1}[\mathsf{read}(t)]]\!] + \tilde{\mathbf{E}}^{(j)}_{t,1}, \\ [\![\mathbf{s}'_{t-1} \mapsto \mathbf{S}_t[\mathsf{write}(t)]]\!]_{\mathbf{B}_3, \mathbf{W}_3} + (\mathbf{E}_{t,3}, \mathbf{E}'_{t,3}), \\ \odot [\![\mathbf{s}'_{t-1}, \mathbf{D}_{\mathsf{circ}}]\!] \end{array} \right\}$$

$$\mathsf{H}'_t : \left\{ \begin{array}{c} \tilde{\mathbf{C}}^{(j)}_{t,1}, \\ \mathbf{C}_{t,3}, \mathbf{C}'_{t,3}, \\ \hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t \end{array} \right\} \leftarrow \left\{ \begin{array}{c} \mathbb{Z}_q^{L \times m} \\ \mathbb{Z}_q^m, \ \mathbb{Z}_q^{L \times m}, \\ \mathbb{Z}_q^{n \times nm\lceil \log q \rceil}, \ \mathbb{Z}_q^{(n^2 m \lceil \log q \rceil^2 + 1)m} \end{array} \right\}$$

It suffices to show that:

$$\left\{ \begin{array}{c} [\![\mathbf{s}'_{t-1}, \mathbf{D}^{(j)}, \mathbf{z}^{(j)}_{t-1}[\mathsf{read}(t)]]\!] + \tilde{\mathbf{E}}^{(j)}_{t,1}, \\ [\![\mathbf{s}'_{t-1} \mapsto \mathbf{S}_t[\mathsf{write}(t)]]\!]_{\mathbf{B}_3, \mathbf{W}_3} + (\mathbf{E}_{t,3}, \mathbf{E}'_{t,3}), \\ \odot [\![\mathbf{s}'_{t-1}, \mathbf{D}_{\mathsf{circ}}]\!] \end{array} \right\} \approx_c \left\{ \begin{array}{c} \mathbb{Z}_q^{L \times m} \\ \mathbb{Z}_q^m, \ \mathbb{Z}_q^{L \times m}, \\ \mathbb{Z}_q^{n \times nm\lceil \log q \rceil}, \ \mathbb{Z}_q^{(n^2 m \lceil \log q \rceil^2 + 1)m} \end{array} \right\}$$

where the distinguisher also gets

$$\mathsf{aux}_t := (\mathbf{B}_3, \mathbf{W}_3, \mathbf{D}_{\mathsf{circ}}, \mathbf{D}^{(j)}, \mathbf{S}_t[\mathsf{write}(t)])$$

To show that the computational indistinguishability of the above two distributions, we rely on $\mathsf{cssLWE}_{n,q,\chi,\chi_{\mathsf{pre}}}$. The reduction works as follows:

- it parses

$$\mathbf{A} = (\mathsf{flat}_L(\tilde{\mathbf{D}}^{(j)}) \mid \mathbf{D}_{\mathsf{circ}} \mid \mathbf{B}_3 \mid \mathsf{flat}(\mathbf{W}_3)) \in \mathbb{Z}_q^{n \times (QLm + (n^2 m \lceil \log q \rceil^2 + 1)m + m + Lm)},$$
$$\mathbf{u} = (\mathsf{flat}(\mathbf{U}^{(j)}) \mid \mathbf{u}_{\mathsf{circ}} \mid \mathbf{u}_3 \mid \mathsf{flat}(\mathbf{U}'_3)) \in \mathbb{Z}_q^{QLm + (n^2 m \lceil \log q \rceil^2 + 1)m + m + Lm},$$

  and $\hat{\mathbf{U}} \in \mathbb{Z}_q^{n \times nm\lceil \log q \rceil}$ obtained from the $\mathsf{cssLWE}_{n,q,\chi,\chi_{\mathsf{pre}}}$ instance,
- it sets $\mathbf{D}^{(j)} = \tilde{\mathbf{D}}^{(j)} + \mathbf{z}^{(j)}_{t-1}[\mathsf{read}(t)]^\top \otimes \mathbf{G}$ while using $\mathbf{D}_{\mathsf{circ}}, \mathbf{B}_3, \mathbf{W}_3$ as obtained from the cssLWE instance,
- it samples $\mathbf{S}_t[\mathsf{write}(t)] \leftarrow \mathbb{Z}_q^{L \times m}$, and outputs

$$\left\{ \begin{array}{c} \mathbf{U}^{(j)}, \\ (\mathbf{u}_3, \mathbf{U}'_3) + (\mathbf{0}, \mathbf{S}_t[\mathsf{write}(t)]\mathbf{G}), \\ (\hat{\mathbf{U}}, \mathbf{u}_{\mathsf{circ}}) \end{array} \right\}$$
$$\mathsf{aux}_t = (\mathbf{B}_3, \mathbf{W}_3, \mathbf{D}_{\mathsf{circ}}, \mathbf{D}^{(j)}, \mathbf{S}_t[\mathsf{write}(t)])$$

Observe that

- if $(\mathbf{A}, \mathbf{U}, \mathbf{u})$ is a structured $\mathsf{csLWE}_{n,q,\chi,\chi_{\mathsf{pre}}}$ instance, the view of the distinguisher is identical to H'_{t-1}; (where we implicitly use that $\chi_{\mathsf{pre}} \ll \chi'$, and that, therefore, to error term of $\tilde{\mathbf{C}}^{(j)}_{t,1}$ noise floods the LWE error terms)
- if $(\mathbf{A}, \mathbf{U}, \mathbf{u})$ is a uniform random instance, the view of the distinguisher is identical to H_t.

We conclude that $\mathsf{H}_{t-1} \approx_c \mathsf{H}'_t$ as required, as the rest of the distribution can be produced by the reduction, using the auxiliary information aux_t, as follows:

1. the matrices

$$\mathbf{A}_{st}, \mathbf{B}_0, \mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_4, \mathbf{W}_1, \mathbf{W}_2, \mathbf{P}$$
$$\mathbf{C}_0, \mathbf{C}'_0, \{\mathbf{C}_{t',1}, \mathbf{C}'_{t',1}, \mathbf{C}_{t',2}, \mathbf{C}'_{t',2}, \mathring{\mathbf{C}}_{t'}, \mathring{\mathbf{C}}'_t\}_{t' \in [t]}, \{\mathbf{C}_{t,3}, \mathbf{C}'_{t,3}, \hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t\}_{t' \in [t-1]}$$
$$\mathbf{A}^{(j)}_0, \mathbf{R}^{(j)}$$
$$\tilde{\mathbf{C}}^{(j)}_0, \{\tilde{\mathbf{C}}^{(j)}_{t',1}\}_{t' \in [t-1]}, \{\tilde{\mathbf{C}}^{(j)}_{t'}[\mathsf{write}(t) \cup \mathsf{read}(t)]\}_{t' \in [t-1]}$$

   are samples as in H_{t-1}.

2. it samples $\mathbf{s}'_t, \mathbf{s}'_{t+1}, \cdots, \mathbf{s}'_{T-1}$ exactly as in Enc,
3. it computes $\mathbf{S}_t$ by using $\mathbf{S}_t[\mathsf{write}(t)]$ given in aux_t and sampling $\mathbf{S}_t[\overline{\mathsf{write}(t)}]$, and then samples $\mathbf{S}_{t+1}, \ldots, \mathbf{S}_T$ exactly as in Enc,[12]
4. it produces $(\mathbf{C}_{t',1}, \mathbf{C}'_{t',1}), (\mathbf{C}_{t',2}, \mathbf{C}'_{t',2}), (\mathbf{C}_{t',3}, \mathbf{C}'_{t',3}), (\hat{\mathbf{C}}_{t'}, \hat{\mathbf{C}}'_{t'}), (\mathring{\mathbf{C}}_{t'}, \mathring{\mathbf{C}}'_{t'})$, for $t' = t{+}1, \ldots, T$, and $(\mathbf{C}_{T,4}, \mathbf{C}'_{T,4})$ as in Enc.
5. it computes $\tilde{\mathbf{C}}^{(j)}_{t'}[\mathsf{write}(t) \cup \mathsf{read}(t)]$, for $t' = t, \ldots, T$, and $\tilde{\mathbf{C}}^{(j)}_{t',1}$, for $t' = t+1, \ldots, T$, as in Eq. (13).

□

Lemma 10. $\mathsf{H}'_T \approx_c \mathsf{H}_T$.

Proof. We want to relate the following quantities corresponding to the differences between the two games:

$$\mathsf{H}_T : \left\{ \begin{array}{c} \tilde{\mathbf{C}}^{(j)}_T[\mathsf{write}(T) \cup \mathsf{read}(T)], \\ \tilde{\mathbf{C}}^{(j)}_{T,4}, \\ \mathbf{C}_{T,4}, \\ \mathbf{C}'_{T,4} \end{array} \right\} = \left\{ \begin{array}{c} [\![\mathbf{S}_T[\mathsf{write}(T) \cup \mathsf{read}(T)], \mathbf{A}^{(j)}_0, \mathbf{z}^{(j)}_T[\mathsf{write}(T) \cup \mathsf{read}(T)]]\!] + \tilde{\mathbf{E}}^{(j)}_T, \\ \mathbf{z}^{(j)}_T[1]\mathbf{S}_T[1]\mathbf{G}\mathbf{R}^{(j)} + \tilde{\mathbf{E}}^{(j)}_{T,4}, \\ \mathbf{S}_T[1]\mathbf{B}_4 + \mathbf{E}_{T,4}, \\ \mathbf{S}_T[1]\mathbf{P} + \boldsymbol{\mu} \cdot \lfloor q/2 \rfloor + \mathbf{E}'_{T,4} \end{array} \right\}$$

$$\mathsf{H}'_T : \left\{ \begin{array}{c} \tilde{\mathbf{C}}^{(j)}_T[\mathsf{write}(T) \cup \mathsf{read}(T)], \\ \tilde{\mathbf{C}}^{(j)}_{T,4}, \\ \mathbf{C}_{T,4}, \\ \mathbf{C}'_{T,4} \end{array} \right\} \leftarrow \left\{ \begin{array}{c} \mathbb{Z}_q^{L_t \times m}, \\ \mathbb{Z}_q^m, \\ \mathbb{Z}_q^m, \\ \mathbb{Z}_q^m \end{array} \right\}$$

where the distinguisher also gets

$$\mathsf{aux}_t := (\mathbf{A}^{(j)}_0, \mathbf{R}^{(j)}, \mathbf{B}_4, \mathbf{P})$$

Notice that since the attribute $\mathbf{z}_0$ does not satisfy any of the function $f^{(j)}$, this means that $\mathbf{z}^{(j)}_T[1] = 1$ for all j. In what follows we ignore the terms $\mathbf{z}^{(j)}_T$ in $\tilde{\mathbf{C}}^{(j)}_T[\mathsf{write}(T) \cup \mathsf{read}(T)]$: these can be dealt with as in Lemma 12, via a hybrid argument over the rows of $\mathbf{S}_T$. To show that the computational indistinguishability of the above two distributions, we rely on $\mathsf{LWE}_{n,q,\chi_{\mathsf{pre}}}$. The reduction works as follows:

- it parses

$$\mathbf{A} = (\tilde{\mathbf{A}}^{(j)}_0 \mid \mathbf{V}^{(j)} \mid \mathbf{B}_4 \mid \mathbf{P}) \in \mathbb{Z}_q^{n \times (Qm+Qm+m+m)},$$
$$\mathbf{u} = (\mathbf{u}^{(j)} \mid \mathbf{v}^{(j)} \mid \mathbf{u}_4 \mid \mathbf{v}) \in \mathbb{Z}_q^{Qm+Qm+m+m},$$

  obtained from the $\mathsf{LWE}_{n,q,\chi_{\mathsf{pre}}}$ instance,
- it sets $\mathbf{R}^{(j)} \leftarrow \mathsf{SamplePre}(\mathbf{G}, \mathbf{I}, \mathbf{V}^{(j)}, \tau)$, i.e., $\mathbf{V}^{(j)} = \mathbf{G}\mathbf{R}^{(j)}$, and uses $\mathbf{A}^{(j)}_0, \mathbf{B}_4, \mathbf{P}$ as obtained from the LWE instance,
- it outputs

$$\mathsf{aux}_t = (\mathbf{A}^{(j)}_0, \mathbf{R}^{(j)}, \mathbf{B}_4, \mathbf{P}, \left\{ \begin{array}{c} \mathbf{u}^{(j)}, \\ \mathbf{v}^{(j)}, \\ \mathbf{u}_4, \\ \mathbf{v} \end{array} \right\})$$

Observe that

- if $(\mathbf{A}, \mathbf{u})$ is a structured $\mathsf{LWE}_{n,q,\chi_{\mathsf{pre}}}$ instance, the view of the distinguisher is identical to H_T ;

[12] Here, we are using that the marginal distribution of $\mathbf{S}_t$ is uniform random: since we have already replaces with uniform random all the components which had dependencies on $\mathbf{S}_1, \ldots \mathbf{S}_{t-1}$, one can sample $\mathbf{S}_t$ uniformly at random.

– if $(\mathbf{A}, \mathbf{u})$ is a uniform random instance, the view of the distinguisher is identical to H'_T.

The rest of the distribution will be produced by the reduction, using the auxiliary information aux_t, by sampling

$$\begin{array}{l}\mathbf{A}_{st}, \mathbf{B}_0, \mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3, \mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3, \mathbf{D}_{\mathsf{circ}} \\ \mathbf{C}_0, \mathbf{C}'_0, \{\mathbf{C}_{t',1}, \mathbf{C}'_{t',1}, \mathbf{C}_{t',2}, \mathbf{C}'_{t',2}, \mathring{\mathbf{C}}_{t'}, \mathring{\mathbf{C}}'_t\}_{t' \in [T]}, \{\mathbf{C}_{t,3}, \mathbf{C}'_{t,3}, \hat{\mathbf{C}}_t, \hat{\mathbf{C}}'_t\}_{t' \in [T]} \\ \mathbf{D}_0^{(j)} \\ \tilde{\mathbf{C}}_0^{(j)}, \{\tilde{\mathbf{C}}_{t',1}^{(j)}\}_{t' \in [T]}, \{\tilde{\mathbf{C}}_{t'}^{(j)}[\mathsf{write}(t) \cup \mathsf{read}(t)]\}_{t' \in [T-1]}\end{array}$$

as in H'_T. We conclude that $\mathsf{H}'_T \approx_c \mathsf{H}_T$ as required. □

E Additional Results

Lemma 12. *Let $\mathbf{z} \in \{0,1\}^\ell$, with $z_i \in \{0,1\}$ denoting the i-th entry of $\mathbf{z}$. The distributions*

$$\mathcal{D}_0 \coloneqq \left\{ \begin{pmatrix} \mathbf{z}, \mathbf{A}_0, \mathbf{B} \\ \mathbf{s}_1(\mathbf{A}_0 - z_1\mathbf{G}) + \mathbf{e}_1, \ldots, \mathbf{s}_\ell(\mathbf{A}_0 - z_\ell\mathbf{G}) + \mathbf{e}_\ell, \\ \mathbf{s}_1\mathbf{B} + \mathbf{f}_1, \ldots, \mathbf{s}_\ell\mathbf{B} + \mathbf{f}_\ell \end{pmatrix} : \begin{array}{c} \mathbf{s}_i \leftarrow \mathbb{Z}_q^n, \mathbf{A}_0 \leftarrow \mathbb{Z}_q^{n \times m}, \mathbf{B} \leftarrow \mathbb{Z}_q^{n \times m'}, \\ \mathbf{e}_i \leftarrow \mathcal{D}_{\mathbb{Z}^m, \chi}, \mathbf{f}_i \leftarrow \mathcal{D}_{\mathbb{Z}^{m'}, \chi} \end{array} \right\}$$

and

$$\mathcal{D}_1 \coloneqq \left\{ \begin{pmatrix} \mathbf{z}, \mathbf{A}_0, \mathbf{B} \\ \mathbf{u}_1, \ldots, \mathbf{u}_\ell, \\ \mathbf{v}_1, \ldots, \mathbf{v}_\ell \end{pmatrix} : \begin{array}{c} \mathbf{A}_0 \leftarrow \mathbb{Z}_q^{n \times m}, \mathbf{B} \leftarrow \mathbb{Z}_q^{n \times m'}, \\ \mathbf{u}_i \leftarrow \mathbb{Z}_q^m, \mathbf{v}_i \leftarrow \mathbb{Z}_q^{m'} \end{array} \right\}$$

are computationally indistinguishable under $\mathsf{LWE}_{n,q,\chi}$.

Proof. For $j = 0, 1, \ldots, \ell$, define

$$\mathsf{H}_j \coloneqq \left\{ \begin{pmatrix} \mathbf{z}, \mathbf{A}_0, \mathbf{B}, \\ \mathbf{u}_1, \ldots, \mathbf{u}_j, \mathbf{s}_{j+1}(\mathbf{A}_0 - z_{j+1}\mathbf{G}) + \mathbf{e}_{j+1}, \ldots, \mathbf{s}_\ell(\mathbf{A}_0 - z_\ell\mathbf{G}) + \mathbf{e}_\ell \\ \mathbf{v}_1, \ldots, \mathbf{v}_j, \mathbf{s}_{j+1}\mathbf{B} + \mathbf{f}_{j+1}, \ldots, \mathbf{s}_\ell\mathbf{B} + \mathbf{f}_\ell \end{pmatrix} : \begin{array}{c} \mathbf{s}_i \leftarrow \mathbb{Z}_q^n, \mathbf{A}_0 \leftarrow \mathbb{Z}_q^{n \times m}, \mathbf{B} \leftarrow \mathbb{Z}_q^{n \times m'}, \\ \mathbf{e}_i \leftarrow \mathcal{D}_{\mathbb{Z}^m, \chi}, \mathbf{f}_i \leftarrow \mathcal{D}_{\mathbb{Z}^{m'}, \chi}, \\ \mathbf{u}_i \leftarrow \mathbb{Z}_q^m, \mathbf{v}_i \leftarrow \mathbb{Z}_q^{m'} \end{array} \right\}$$

We will show that, under $\mathsf{LWE}_{n,q,\chi}$, one has $\mathsf{H}_{j-1} \approx_c \mathsf{H}_j$, for $j \in [\ell]$. This proves the claimed result after noticing that $\mathsf{H}_0 = \mathcal{D}_0$ and $\mathsf{H}_\ell = \mathcal{D}_1$. □

Claim. $\mathsf{H}_{j-1} \approx_c \mathsf{H}_j$.

Proof (of claim). To show that the computational indistinguishability of the above two distributions, we rely on $\mathsf{LWE}_{n,q,\chi}$. On input $\mathbf{z}$, the reduction works as follows:

– it parses $((\tilde{\mathbf{A}} \mid \tilde{\mathbf{B}}), (\tilde{\mathbf{u}} \mid \tilde{\mathbf{v}}))$ obtained from the $\mathsf{LWE}_{n,\chi,q}$ instance.
– it sets $\mathbf{A}_0 \coloneqq \tilde{\mathbf{A}} + z_j\mathbf{G}$ and $\mathbf{B} = \tilde{\mathbf{B}}$.
– it samples $\mathbf{u}_i \leftarrow \mathbb{Z}_q^m, \mathbf{v}_i \leftarrow \mathbb{Z}_q^{m'}$, for $i \in [j-1]$.
– it samples $\mathbf{s}_i \leftarrow \mathbb{Z}_q^n, \mathbf{e}_i \leftarrow \mathcal{D}_{\mathbb{Z}^m, \chi}, \mathbf{f}_i \leftarrow \mathcal{D}_{\mathbb{Z}^{m'}, \chi}$ for $i = j+1, \ldots, \ell$.
– it outputs

$$\left\{ \begin{pmatrix} \mathbf{z}, \mathbf{A}_0, \mathbf{B}, \\ \mathbf{u}_1, \ldots, \mathbf{u}_{j-1}, \tilde{\mathbf{u}}, \mathbf{s}_{j+1}(\mathbf{A}_0 - z_{j+1}\mathbf{G}) + \mathbf{e}_{j+1}, \ldots, \mathbf{s}_\ell(\mathbf{A}_0 - z_\ell\mathbf{G}) + \mathbf{e}_\ell, \\ \mathbf{v}_1, \ldots, \mathbf{v}_{j-1}, \tilde{\mathbf{v}}, \mathbf{s}_{j+1}\mathbf{B} + \mathbf{f}_{j+1}, \ldots, \mathbf{s}_\ell\mathbf{B} + \mathbf{f}_\ell \end{pmatrix} \right\}.$$

Observe that

– if $((\tilde{\mathbf{A}} \mid \tilde{\mathbf{B}}), (\tilde{\mathbf{u}} \mid \tilde{\mathbf{v}}))$ is a structured $\mathsf{LWE}_{n,q,\chi}$ instance, the view of the distinguisher is identical to H_{j-1} since

$$\begin{aligned} \tilde{\mathbf{u}} &= \mathbf{s}_j\tilde{\mathbf{A}} + \mathbf{e}_j = \mathbf{s}_j(\mathbf{A}_0 - z_j\mathbf{G}) + \mathbf{e}_j, \\ \tilde{\mathbf{v}} &= \mathbf{s}_j\tilde{\mathbf{B}} + \mathbf{f}_j = \mathbf{s}_j\mathbf{B} + \mathbf{f}_j. \end{aligned}$$

– if $((\tilde{\mathbf{A}} \mid \tilde{\mathbf{B}}), (\tilde{\mathbf{u}} \mid \tilde{\mathbf{v}}))$ is a uniform random instance, the view of the distinguisher is identical to H_j.

We conclude that $\mathsf{H}_{j-1} \approx_c \mathsf{H}_j$ as required. □