stitute as an invariant. That is, if $\mathbf{u}_1, \ldots, \mathbf{u}_i$ in L_U becomes linearly dependent, then this number i also becomes an invariant which can be utilised. We do not attempt to deal with such cases because they rarely happen in experiments.

4.6 Experimental results for the algorithm

We implemented the algorithm in Section 4.4 in Magma [16]. We tested our implementation on a server (AMD EPYC 7532 CPU at 2.40GHz) to solve some instances of the MCE problem. The results are given in Table 3. Our experiments demonstrate that when running ten instances, two to four of them successfully discover collisions and recover the secret matrices (A, B, C).

Because we conduct $q^{(n-2)/2}$ samplings, we cannot set q to be too large for a practical running. Therefore, we set q to be 61 or 31. As a result, the fraction of effective points is not as large as for $q = 1021$ as in Table 2. For example, in MCE-instance-1, we conducted 3721 samplings and obtained 2702 effective points. Therefore, when q is large, the success rate should increase with the number of effective points.

Parameter set	n	q	Number of effective points	Number of sampling times	Time (seconds)
MCE-instance-1	6	61	2702	3721	420
MCE-instance-2	7	61	20053	29062	5638
MCE-instance-3	8	61	149149	226981	100900
MCE-instance-4	9	31	64202	165870	137715

Table 3. Solving MCE instances

Remark 1. Following [12], a possible improvement on the sampling step (Step (a) of the algorithm in Section 4.4) is as follows.

Recall that in Step (a) of the algorithm in Section 4.4, a corank-1 point is obtained by sampling a random vector in $\mathbb{F}_q^n$ for q times. However, note that starting from a corank-1 vector $\hat{\mathbf{u}}$, the vectors in the vector tuple L_U, if successfully built, are all corank-1. So these vectors can be utilised, instead of starting from a fresh random corank-1 vector. In general, we can walk along the path in the tripartite graph starting from a corank-1 vector until we hit a vector of corank larger than 1. This has the potential of reducing the complexity of the algorithm from $O(q^{(n-2)/2} \cdot (q \cdot n^3 + n^4) \cdot (\log(q))^2)$ to $O(q^{(n-2)/2} \cdot n^4 \cdot (\log(q))^2)$, as we would only need to sample a fresh corank-1 vector very few times during the execution of the algorithm.

One question for this approach is whether it results in a distribution close to the uniform one. To test this, we implemented the above approach. In the case of MCE-instance-1, our preliminary experimental results show that when running 6 instances, one of them successfully finds a collision and recovers the

5.2 An algorithm for ATFE based on a new isomorphism invariant

The main innovation of our algorithm for ATFE is to associate distinguishing isomorphism invariants to low-rank points.

Let $\phi : \mathbb{F}_q^n \times \mathbb{F}_q^n \times \mathbb{F}_q^n \to \mathbb{F}_q$. Suppose by Theorem 2, it is expected that there are roughly q^k many $\hat{\mathbf{u}} \in \mathbb{P}(\mathbb{F}_q^n)$, such that $\mathrm{rk}_\phi(\hat{\mathbf{u}}) = r$. Let us *assume* that there is an easy-to-compute, distinguishing, isomorphism invariant[7] for those rank-r $\hat{\mathbf{u}}$.

Then the algorithm goes as follows: first sample $O(q^{k/2})$-many rank-r points for ϕ, and $O(q^{k/2})$-many rank-r points for ψ. For each point, compute this isomorphism invariant. Then by the birthday paradox, there exist one point $\hat{\mathbf{u}}$ from the list of ϕ, and one point $\hat{\mathbf{v}}$ from the list of ψ, such that their isomorphism invariants are the same. Finally, use Gröbner basis with partial information for $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$ to recover the desired isomorphism.

Following Equation 1, the running time of the above algorithm can then be estimated as

$$O(q^{k/2} \cdot (\mathsf{samp\text{-}cost} + \mathsf{inv\text{-}cost}) + \mathsf{gb\text{-}cost}),$$

where samp-cost denotes the sampling cost, the inv-cost denotes the invariant computing cost, and gb-cost denotes the Gröbner basis with partial information cost.

The sampling step can be achieved by either the min-rank method (Appendix A) or Beullens' graph-walking method [12]. For the min-rank method, the cost of sampling a low-rank matrix can be estimated for concrete values of n, k, and r by e.g. [6,35,44]. For the graph-walking method, the sampling cost can be estimated based on certain statistics of graphs associated with alternating trilinear forms by Beullens [12, Theorem 1].

The gb-cost can be estimated as $O(n^6)$ as in [12]. This is based on the hybrid Gröbner basis method with the first row known in the variable matrix. The effectiveness of this hybrid Gröbner basis method was first discovered in [27] and then utilised in [17,42]. Beullens further improved this method by noting that (1) knowing the first row up to scalar suffices, and (2) for low-rank points, the kernel information can be incorporated [12, Section 4].

The main innovation of the above algorithm is a new isomorphism invariant which we describe next.

5.3 The isomorphism invariant step

Suppose $\hat{\mathbf{u}} \in \mathbb{P}(\mathbb{F}_q^n)$ satisfies that $\mathrm{rk}_\phi(\hat{\mathbf{u}}) = r$. Then $K := \ker(\phi_{\hat{\mathbf{u}}}) \leq \mathbb{F}_q^n$ is a dimension-$(n-r)$ space, also preserved by any isomorphism. This allows us to consider the trilinear form $\tilde{\phi}_{\hat{\mathbf{u}}} : K \times \mathbb{F}_q^n \times \mathbb{F}_q^n \to \mathbb{F}_q$, and it can be verified easily that the *isomorphism type* of $\tilde{\phi}_{\hat{\mathbf{u}}}$ under $\mathrm{GL}(K) \times \mathrm{GL}(n, q)$ is an isomorphism invariant.

[7] That is a function f from low-rank points to some set S, such that $f(\hat{\mathbf{u}}) \neq f(\hat{\mathbf{v}})$ for $\hat{\mathbf{u}} \neq \hat{\mathbf{v}}$, and f is unchanged by basis changes.

To use the isomorphism type of $\tilde{\phi}_{\hat{\mathbf{u}}}$ in the algorithm, we need the isomorphism types are (1) easy to compute, and (2) distinguishing; that is, for different $\hat{\mathbf{u}}, \hat{\mathbf{v}} \in \mathbb{P}(\mathbb{F}_q^n)$, $\tilde{\phi}_{\hat{\mathbf{u}}}$ and $\tilde{\phi}_{\hat{\mathbf{v}}}$ are different.

To verify these, we perform the following experiment in Magma [16].

1. Sample a random $\phi \in \mathrm{ATF}(n, q)$.
2. Sample a random rank-r point $\hat{\mathbf{u}} \in \mathbb{P}(\mathbb{F}_q^n)$.
3. Sample t random rank-r points $\hat{\mathbf{v}} \in \mathbb{P}(\mathbb{F}_q^n)$. For each such point, do:
   (a) Use the Gröbner basis with partial information to decide whether $\tilde{\phi}_{\hat{\mathbf{u}}}$ and $\tilde{\phi}_{\hat{\mathbf{v}}}$ are isomorphic.

Our experiments give the following.

- For $n = 9$, $r = 4$, and $p = 3$, 10 experiments (i.e. for 10 $\hat{\mathbf{u}}$ from 10 random alternating trilinear forms) with $t = 100$ comparisons (i.e. for 100 different $\hat{\mathbf{v}}$ to compare with $\hat{\mathbf{u}}$). On average, 75 out of 100 $\tilde{\phi}_{\hat{\mathbf{v}}}$ are not isomorphic with $\tilde{\phi}_{\hat{\mathbf{u}}}$.
- For $n = 10$, $r = 6$, and $p = 3$, 10 experiments (i.e. for 10 $\hat{\mathbf{u}}$ from 10 random alternating trilinear forms) with $t = 100$ comparisons (i.e. for 100 different $\hat{\mathbf{v}}$ to compare with $\hat{\mathbf{u}}$). On average, 95 out of 100 $\tilde{\phi}_{\hat{\mathbf{v}}}$ are not isomorphic with $\tilde{\phi}_{\hat{\mathbf{u}}}$.

For $n = 11$, our code does not work for $n = 11$ on a laptop, due to the Gröbner basis step.

From these experiments we see that (1) the Gröbner basis with partial information algorithm is effective in practice to test isomorphism between $\tilde{\phi}_{\hat{\mathbf{u}}}$ and $\tilde{\phi}_{\hat{\mathbf{v}}}$, and (2) as n goes from 9 to 10, the isomorphism type of $\tilde{\phi}_{\hat{\mathbf{u}}}$ becomes more distinguishing. These give some preliminary support that the isomorphism types of $\tilde{\phi}_{\hat{\mathbf{u}}}$ do serve as a easy-to-compute, distinguishing, isomorphism invariant.

Note that testing isomorphism here is not enough, and canonical forms are required to serve as an isomorphism invariant. Even though to transform an isomorphism invariant algorithm to a canonical form one may not be an easy process, it is generally regarded as doable, at least from the experience from graph isomorphism [3].

5.4 Concrete estimations of this algorithm for ALTEQ parameters

We show the improvement of our algorithm over Beullens' algorithm for a set of ALTEQ parameters. In [15], $n = 13$ and $q = 2^{32} - 1$ are used for the 128-bit security. In this case, Beullens' algorithm runs in time $O(q^{(n-5)/2} \cdot n^{11} + q^{n-7} \cdot n^6)$. As the major factor comes from q^{n-7}, the bit complexity is above $32 \cdot 6 = 192$. For our algorithm, using rank-$(n-5)$ points, the time complexity is estimated as $O(q^{(n-7)/2} \cdot (\mathsf{samp\text{-}cost} + \mathsf{inv\text{-}cost}) + \mathsf{gb\text{-}cost})$. The sampling cost can be estimated as in Appendix A based on [6], which is 32-bit complexity. The inv-cost and gb-cost are lower than the sampling cost. So the total bit complexity of our algorithm is $32 \cdot 3 + 32 = 128$.

6 Quantum attacks

We lower the run time exponent of our classical algorithms for MCE and ATFE on a quantum computer by a factor of 2/3. This speed up results from deploying Szegedy type quantum random walks to find collisions, but comes at the cost of exponential quantum space requirement. Therefore, there is reason to only consider the classical algorithms to tune the parameters of the cryptosystems. We describe the quantum algorithms for ATFE in greater detail. The MCE case is analogous but a little easier, since there is no need for Gröbner basis computations.

6.1 Collision detection through quantum random walks

The first collision detection quantum algorithms were due to Brassard, Høyer, and Tapp [18] and special to two-to-one functions, building on Grover's search [33]. Ambanis removed these restrictions and devised improved collision detection algorithms through quantum random walks, that match lower bounds [2]. Szegedy further improved these algorithms and brought them under a unified framework of quantum random walks with memory [41]. We will use Szegedy's version of quantum random walks for the quantum speedups of classical algorithms to the decision version ATFE.

We first paraphrase theorem 3 in [41], specialized to the oracle function being the identity. Let X be a finite set and $R \subset X \times X$ a binary relation with a membership tester. For a positive real number α and a uniformly random subset $H \subset X$ of size $|X|^\alpha$, let p_α denote the probability that $R \cap (H \times H)$ is non empty. There is a quantum algorithm to differentiate between the cases $p_\alpha = 0$ and $p_\alpha \geq \epsilon$ in time $O(|X|^\alpha + 1000\sqrt{|X|^\alpha/\epsilon})$.

Extensions of Szegedy's algorithm by Magniez, Nayak, Richter, Roland, and Santha [38,37] may be deployed to tackle the search version ATFE within the same running time. Another extension of Szegedy's algorithm is to claw finding, by Tani [43]. The claw finding formalism is convenient to phrase ATFE in and infer polynomial speed ups. Let $f : X \to Z$ and $g : Y \to Z$ be two functions between finite sets. Given oracle access to f and g, the claw finding problem is to find an $(x, y) \in X \times Y$ such that $f(x) = g(y)$, if one exists. The functions may be presented either as standard oracles or as comparison oracles. We describe the later in the quantum setting, as they suffice. A comparison oracle maps quantum states

$$|x, y, b, w\rangle \longmapsto |x, y, b \oplus [f(x) >^? g(y)], w\rangle.$$

Here, b is a bit; x and y respectively index quantum states corresponding to elements in X and Y. Fixing an ordering on Z, $[f(x) >^? g(y)]$ is a bit that is one if and only if $f(x) > g(y)$. The last register indexed by w is an ancilla for work space. For instances with X and Y of roughly the same size, Tani's algorithm finds claws on a quantum computer in time $O((|X||Y|)^{1/3})$.

In applying these quantum random walk algorithms, we will invoke generic algorithms applicable to functions on finite sets presented as an oracle. For clarity

of exposition, we focus on speedups to the main exponential term and suppress incremental polynomial factors.

6.2 Solving ATFE through quantum random walks.

As a warm up, we first describe quantum algorithms for ATFE that do not exploit our new invariants. Then, we build on these algorithms by incorporating the invariants to achieve the aforementioned run time exponent.

A classical oracle from the Gröbner basis attack with partial information. First, consider the decision version of ATFE. That is, given two alternating trilinear forms ϕ and ψ, the existence of an $A \in GL(n,q)$ such that $\psi = \phi \circ A$ is in question. Central to all our methods is a polynomial time classical algorithm to test membership in the relation set

$$R_{\phi,\psi} := \{(\hat{\mathbf{u}}, \hat{\mathbf{v}}) \in \mathbb{P}(\mathbb{F}_q^n)^2 \mid \exists A \in GL(n,q) \text{ such that } \psi = \phi \circ A \text{ and } A^{-1}\hat{\mathbf{u}} = \hat{\mathbf{v}}\}.$$

If ϕ and ψ are not isomorphic, $R_{\phi,\psi}$ is empty. A pair $(\hat{\mathbf{u}}, \hat{\mathbf{v}}) \in \mathbb{P}(\mathbb{F}_q^n)^2$ satisfying $A^{-1}\hat{\mathbf{u}} = \hat{\mathbf{v}}$ enforces n $\mathbb{F}_q$-linear constraints on A. The Gröbner basis attack with partial information in [27], augmented with these linear constraints can tell in heuristic polynomial time if the pair $(\hat{\mathbf{u}}, \hat{\mathbf{v}})$ is in $R_{\phi,\psi}$. We henceforth make the same assumptions. This polynomial time classical algorithm to test membership can be converted to a polynomial sized quantum circuit that can test membership in superposition. Further, incorporate a time out clause into the membership algorithm to make the Gröbner basis methods stop searching and declare non existence.

Invoke Szegedy's algorithm with X as $\mathbb{P}(\mathbb{F}_q^n)$, R as $R_{\phi,\psi}$, α as $1/3$ and uniformly sampling an $H \subset \mathbb{P}(\mathbb{F}_q^n)$ of size $\Theta\left(q^{n/3}\right)$. We claim that the probability gap may be taken to be $\epsilon = \Omega\left(q^{-n/3}\right)$. To prove the claim, consider two isomorphic ϕ and ψ. That is, there exists at least one $A_{\phi,\psi} \in \mathrm{GL}(n,q)$ such that $\psi = \phi \circ A_{\phi,\psi}$. Therefore,

$$\Pr_H\left((R_{\phi,\psi} \cap (H \times H)) \neq \emptyset\right) \geq \Pr_H\left((H \cap A_{\phi,\psi}(H)) \neq \emptyset\right) \geq \Omega\left(q^{-n/3}\right),$$

proving the claim. In summary, we can tell if ϕ and ψ are isomorphic in time $q^{n/3}\mathrm{poly}(n, \log q)$ on a quantum computer. This strategy also tackles the promise search version ATFE within the same running time, thanks to extensions of Szegedy's algorithm by Magniez, Nayak, Richter, Roland, and Santha [38,37]. An alternative is to solve ATFE by claw finding. To phrase ATFE as claw finding, independently draw uniformly random sets $X \subset \mathbb{P}(\mathbb{F}_q^n)$ and $Y \subset \mathbb{P}(\mathbb{F}_q^n)$, each of size $q^{n/2}$. Take $f : X \to \mathbb{P}(\mathbb{F}_q^n)$ as the multiplication by A^{-1} map $\mathbf{u} \longmapsto A^{-1}\mathbf{u}$ and $g : Y \to \mathbb{F}_q^n$ as the identity. The birthday paradox ensures for isomorphic ϕ and ψ that there is a solution to claw finding with constant positive probability. The algorithm for testing membership in $R_{\phi,\psi}$ from the previous subsection yields a comparison oracle. Tani's algorithm for claw finding solves ATFE in time $q^{n/3}\mathrm{poly}(n, \log q)$.

6.3 Low-rank birthday attacks on ATFE via quantum random walks

We next describe how our invariant functions can be incorporated into the quantum algorithms. For $\phi \in \mathrm{ATF}(\mathbb{F}_q^n)$ and $\hat{\mathbf{v}} \in \mathbb{P}(\mathbb{F}_q^n)$, let $\phi/\hat{\mathbf{v}}$ denote the isomorphism class of the restriction of ϕ to $\mathsf{ker}(\phi_{\hat{\mathbf{v}}}) \times \mathbb{F}_q^n \times \mathbb{F}_q^n$ under the $\mathrm{GL}(\mathsf{ker}(\phi_{\hat{\mathbf{v}}})) \times \mathrm{GL}(n,q)$ action. For a positive number R, let

$$S_R := \left\{(\phi, \hat{\mathbf{v}}) \in \mathrm{ATF}(\mathbb{F}_q^n) \times \mathbb{P}(\mathbb{F}_q^n) \mid \mathrm{rk}(\phi_{\hat{\mathbf{v}}}) = R \right. \quad .$$

The invariant function from section 5 then takes the form

$$(\phi, \hat{\mathbf{v}}) \xmapsto{F_1} \phi/\hat{\mathbf{v}}.$$

Fix the choice of rank R and let k be the exponent such that $\|\mathbb{P}_{\phi,R}\| = q^k$. Assume that F restricted to S_R is distinguishing.

Let ϕ and ψ denote the two input trilinear forms with the existence of an $A \in \mathrm{GL}(n, \mathbb{F}_q)$ such that $\psi = \phi \circ A$ in question. Consider the relation set

$$R_{\phi,\psi}^{F_1} := \{(\mathbf{u}, \mathbf{v}) \in \mathbb{P}_{\phi,R}^2 \mid F_1(\phi, \hat{\mathbf{u}}) = F_1(\psi, \hat{\mathbf{v}})\}.$$

If ϕ and ψ are not isomorphic, then neither are their restrictions to $\mathsf{ker}(\phi_{\hat{\mathbf{v}}}) \times \mathbb{F}_q^n \times \mathbb{F}_q^n$, implying $R_{\phi,\psi}^{F_1}$ is empty. If ϕ and ψ are isomorphic, by the distinguishing property of F_1, with high probability, $F_1(\phi, \hat{\mathbf{u}}) = F_1(\psi, \hat{\mathbf{v}})$ if and only if $\exists A \in \mathrm{GL}(n,q)$ such that $\psi = \phi \circ A$ and $A^{-1}\hat{\mathbf{u}} = \hat{\mathbf{v}}$.

The invariance and the distinguishing property of F_1 together ensure that with high probability, a random pair $(\hat{\mathbf{u}}, \hat{\mathbf{v}}) \in R_{\phi,\psi}^{F_1}$ is a witness to the isomorphism of ϕ and ψ restricted to $\mathsf{ker}(\phi_{\hat{\mathbf{u}}}) \times \mathbb{F}_q^n \times \mathbb{F}_q^n$. That is, there exists an $A \in \mathrm{GL}(n,q)$ such that $\hat{\mathbf{v}} = A^{-1}\hat{\mathbf{u}}$ and A moves the restriction of ϕ to the restriction of ψ. In particular, A restricted to $\mathsf{ker}(\phi_{\hat{\mathbf{u}}})$ acts in the first dimension. Therefore, with $(\hat{\mathbf{u}}, \hat{\mathbf{v}})$ as the partial information, the Gröbner basis algorithm of [20,42] becomes a heuristic polynomial time test of membership in $R_{\phi,\psi}^{F_1}$.

Invoke Szegedy's algorithm with X as $\mathbb{P}_{\phi,R}$, R as $R_{\phi,\psi}^{F_1}$, α as $1/3$ and uniformly sampling an $H \subset \mathbb{P}_{\phi,R}$ of size $\Theta\left(q^{k/3}\right)$. For isomorphic ϕ and ψ, there exists at least one $A_{\phi,\psi} \in \mathrm{GL}(n,q)$ such that $\psi = \phi \circ A_{\phi,\psi}$. Therefore, by the invariance and the distinguishing nature of F_1,

$$\Pr\left((R_{\phi,\psi}^{F_1} \cap (H \times H)) \neq \emptyset\right) \geq \Pr\left((H \cap A_{\phi,\psi}(H)) \neq \emptyset\right) \geq \Omega\left(q^{-k/3}\right),$$

proving that the probability gap may be taken to be $\epsilon = \Omega\left(q^{-k/3}\right)$. Therefore, for a rank parameter such that the sampling cost $\mathsf{samp\text{-}cost}$ is in polynomial time, the decision version of ATFE can be solved in $q^{k/3}\mathsf{poly}(n, \log q)$ time on a quantum computer. To tackle the promise search version ATFE within the same running time, applying the extensions of Szegedy's algorithm by Magniez, Nayak, Richter, Roland, and Santha [38,37], the search version ATFE can also be solved in

$$q^{k/3} \cdot \mathsf{poly}(n, \log q)$$

time on a quantum computer. Curiously, it is not obvious if the claw finding formalism in Tani's algorithm can be adapted to the low-rank birthday attacks. If we can efficiently derive canonical forms in addition to testing the isomorphism class of the restriction, then Tani's algorithm apply immediately. The reason being that we can order the canonical form representatives and obtain a comparison oracle.

6.4 Low-rank birthday attacks on MCE via quantum random walks

Recall the notation from section 4. We next phrase MCE as claw finding. Let ϕ, ψ be the two input isomorphic trilinear forms. Take X and Y as uniformly random subsets of co-rank 1 projective points, each of size $q^{n/2}$. Take f as the $\hat{\mathbf{u}} \longmapsto \bar{\phi}[\hat{\mathbf{u}}]$ map and g as the $\hat{\mathbf{u}} \longmapsto \bar{\psi}[\hat{\mathbf{u}}]$ map. The birthday paradox ensures that there is a solution to claw finding with constant positive probability. Invoking Tani's algorithm solves MCE in $q^{n/3}\mathsf{poly}(n, \log q)$ time.

Acknowledgements. We thank the anonymous reviewers for their careful reading and helpful suggestions. Youming Qiao was partly supported by ARC DP200100950 and LP220100332. Gang Tang was partly supported by ARC LP220100332, Sydney Quantum Academy, and EPSRC grant EP/V011324/1.

References

1. Alamati, N., Feo, L.D., Montgomery, H., Patranabis, S.: Cryptographic group actions and applications. In: Moriai, S., Wang, H. (eds.) Advances in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7-11, 2020, Proceedings, Part II. Lecture Notes in Computer Science, vol. 12492, pp. 411–439. Springer (2020). https://doi.org/10.1007/978-3-030-64834-3_14, https://doi.org/10.1007/978-3-030-64834-3_14
2. Ambainis, A.: Quantum walk algorithm for element distinctness. SIAM Journal on Computing **37**(1), 210–239 (2007). https://doi.org/10.1137/S0097539705447311, https://doi.org/10.1137/S0097539705447311
3. Babai, L.: Graph isomorphism in quasipolynomial time [extended abstract]. In: Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016. pp. 684–697 (2016)
4. Babai, L.: Canonical form for graphs in quasipolynomial time: preliminary report. In: Charikar, M., Cohen, E. (eds.) Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019. pp. 1237–1246. ACM (2019). https://doi.org/10.1145/3313276.3316356, https://doi.org/10.1145/3313276.3316356
5. Baldi, M., Barenghi, A., Beckwith, L., Biasse, J.F., Esser, A., Gaj, K., Mohajerani, K., Pelosi, G., Persichetti, E., Saarinen, M.J., Santini, P., Wallace, R.: Less: Linear equivalence signature scheme (2023), https://www.less-project.com/LESS-2023-08-18.pdf

6. Bardet, M., Bros, M., Cabarcas, D., Gaborit, P., Perlner, R.A., Smith-Tone, D., Tillich, J., Verbel, J.A.: Improvements of algebraic attacks for solving the rank decoding and minrank problems. In: Moriai, S., Wang, H. (eds.) Advances in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7-11, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12491, pp. 507–536. Springer (2020). https://doi.org/10.1007/978-3-030-64837-4_17, https://doi.org/10.1007/978-3-030-64837-4_17
7. Bardet, M., Otmani, A., Saeed-Taha, M.: Permutation code equivalence is not harder than graph isomorphism when hulls are trivial. In: 2019 IEEE International Symposium on Information Theory (ISIT). pp. 2464–2468. IEEE (2019). https://doi.org/10.1109/ISIT.2019.8849855
8. Barenghi, A., Biasse, J.F., Ngo, T., Persichetti, E., Santini, P.: Advanced signature functionalities from the code equivalence problem. International Journal of Computer Mathematics: Computer Systems Theory **7**(2), 112–128 (2022)
9. Battagliola, M., Borin, G., Meneghetti, A., Persichetti, E.: Cutting the grass: Threshold group action signature schemes. Cryptology ePrint Archive (2023)
10. Belsley, E.: Rates of convergence of Markov chains related to association schemes, Harvard University Ph. D. Ph.D. thesis, thesis (1993)
11. Beullens, W.: Not enough less: An improved algorithm for solving code equivalence problems over f q. In: International Conference on Selected Areas in Cryptography. pp. 387–403. Springer (2020)
12. Beullens, W.: Graph-theoretic algorithms for the alternating trilinear form equivalence problem. In: Handschuh, H., Lysyanskaya, A. (eds.) Advances in Cryptology - CRYPTO 2023 - 43rd Annual International Cryptology Conference, CRYPTO 2023, Santa Barbara, CA, USA, August 20-24, 2023, Proceedings, Part III. Lecture Notes in Computer Science, vol. 14083, pp. 101–126. Springer (2023). https://doi.org/10.1007/978-3-031-38548-3_4, https://doi.org/10.1007/978-3-031-38548-3_4
13. Biasse, J.F., Micheli, G., Persichetti, E., Santini, P.: Less is more: code-based signatures without syndromes. In: Progress in Cryptology-AFRICACRYPT 2020: 12th International Conference on Cryptology in Africa, Cairo, Egypt, July 20–22, 2020, Proceedings 12. pp. 45–65. Springer (2020)
14. Bläser, M., Chen, Z., Duong, D.H., Joux, A., Nguyen, N.T., Plantard, T., Qiao, Y., Susilo, W., Tang, G.: On digital signatures based on isomorphism problems: Qrom security, ring signatures, and applications. Cryptology ePrint Archive, Paper 2022/1184 (2022), https://eprint.iacr.org/2022/1184
15. Bläser, M., Duong, D.H., Narayanan, A.K., Plantard, T., Qiao, Y., Sipasseuth, A., Tang, G.: The alteq signature scheme: Algorithm specifications and supporting documentation (2023), https://pqcalteq.github.io/ALTEQ_spec_2023.09.18.pdf
16. Bosma, W., Cannon, J., Playoust, C.: The Magma algebra system. I. The user language. J. Symbolic Comput. **24**(3-4), 235–265 (1997). https://doi.org/10.1006/jsco.1996.0125, http://dx.doi.org/10.1006/jsco.1996.0125, computational algebra and number theory (London, 1993)
17. Bouillaguet, C., Fouque, P., Véber, A.: Graph-theoretic algorithms for the "isomorphism of polynomials" problem. In: Advances in Cryptology - EUROCRYPT 2013. pp. 211–227 (2013)
18. Brassard, G., Hoyer, P., Tapp, A.: Quantum cryptanalysis of hash and claw-free functions. In: Lucchesi, C.L., Moura, A.V. (eds.) LATIN'98: Theoretical Informatics. pp. 163–169. Springer Berlin Heidelberg, Berlin, Heidelberg (1998)

19. Brassard, G., Yung, M.: One-way group actions. In: Advances in Cryptology - CRYPTO 1990. pp. 94–107 (1990)
20. Bürgisser, P., Franks, C., Garg, A., de Oliveira, R.M., Walter, M., Wigderson, A.: Efficient algorithms for tensor scaling, quantum marginals, and moment polytopes. In: 59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018. pp. 883–897 (2018). https://doi.org/10.1109/FOCS.2018.00088
21. Chou, T., Niederhagen, R., Persichetti, E., Ran, L., Randrianarisoa, T.H., Reijnders, K., Samardjiska, S., Trimoska, M.: Matrix code equivalence digital signature (2023), https://www.meds-pqc.org/spec/MEDS-2023-07-26.pdf
22. Chou, T., Niederhagen, R., Persichetti, E., Randrianarisoa, T.H., Reijnders, K., Samardjiska, S., Trimoska, M.: Take your meds: Digital signatures from matrix code equivalence. In: International Conference on Cryptology in Africa. pp. 28–52. Springer (2023)
23. Couvreur, A., Debris-Alazard, T., Gaborit, P.: On the hardness of code equivalence problems in rank metric. arXiv preprint arXiv:2011.04611 (2020)
24. D'Alconzo, G., Gangemi, A.: Trifors: Linkable trilinear forms ring signature. Cryptology ePrint Archive (2022)
25. Ducas, L., Postlethwaite, E.W., Pulles, L.N., Woerden, W.v.: Hawk: Module lip makes lattice signatures fast, compact and simple. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 65–94. Springer (2022)
26. Ducas, L., van Woerden, W.: On the lattice isomorphism problem, quadratic forms, remarkable lattices, and cryptography. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 643–673. Springer (2022)
27. Faugère, J., Perret, L.: Polynomial equivalence problems: Algorithmic and theoretical aspects. In: Advances in Cryptology - EUROCRYPT 2006. pp. 30–47 (2006)
28. Fiat, A., Shamir, A.: How to prove yourself: Practical solutions to identification and signature problems. In: Advances in Cryptology – CRYPTO 1986. pp. 186–194 (1986)
29. Fulman, J., Goldstein, L.: Stein's method and the rank distribution of random matrices over finite fields. The Annals of Probability **43**(3) (may 2015). https://doi.org/10.1214/13-aop889, https://doi.org/10.1214%2F13-aop889
30. Goldreich, O., Micali, S., Wigderson, A.: Proofs that yield nothing but their validity for all languages in NP have zero-knowledge proof systems. J. ACM **38**(3), 691–729 (1991). https://doi.org/10.1145/116825.116852
31. Grochow, J.A., Qiao, Y.: On the complexity of isomorphism problems for tensors, groups, and polynomials I: tensor isomorphism-completeness. SIAM J. Comput. **52**(2), 568–617 (2023). https://doi.org/10.1137/21m1441110, https://doi.org/10.1137/21m1441110
32. Grochow, J.A., Qiao, Y., Tang, G.: Average-case algorithms for testing isomorphism of polynomials, algebras, and multilinear forms. journal of Groups, complexity, cryptology **14** (2022)
33. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: Proceedings of the twenty-eighth annual ACM symposium on Theory of computing. pp. 212–219 (1996)
34. Ji, Z., Qiao, Y., Song, F., Yun, A.: General linear group action on tensors: A candidate for post-quantum cryptography. In: Hofheinz, D., Rosen, A. (eds.) Theory of Cryptography - 17th International Conference, TCC 2019. vol. 11891, pp. 251–281. Springer (2019)

35. Kipnis, A., Shamir, A.: Cryptanalysis of the hfe public key cryptosystem by relinearization. In: Annual International Cryptology Conference. pp. 19–30. Springer (1999)
36. Leon, J.: Computing automorphism groups of error-correcting codes. IEEE Transactions on Information Theory **28**(3), 496–511 (1982)
37. Magniez, F., Nayak, A., Richter, P.C., Santha, M.: On the hitting times of quantum versus random walks. Algorithmica **63**, 91–116 (2012)
38. Magniez, F., Nayak, A., Roland, J., Santha, M.: Search via quantum walk. In: Proceedings of the thirty-ninth annual ACM symposium on Theory of computing. pp. 575–584 (2007)
39. Reijnders, K., Samardjiska, S., Trimoska, M.: Hardness estimates of the code equivalence problem in the rank metric. Designs, Codes and Cryptography pp. 1–30 (2024)
40. Sendrier, N.: Finding the permutation between equivalent linear codes: The support splitting algorithm. IEEE Trans. Inf. Theory **46**(4), 1193–1203 (2000). https://doi.org/10.1109/18.850662, https://doi.org/10.1109/18.850662
41. Szegedy, M.: Spectra of quantized walks and a $\sqrt{\delta\epsilon}$-rule. arXiv preprint quant-ph/0401053 (2004)
42. Tang, G., Duong, D.H., Joux, A., Plantard, T., Qiao, Y., Susilo, W.: Practical post-quantum signature schemes from isomorphism problems of trilinear forms. In: Dunkelman, O., Dziembowski, S. (eds.) Advances in Cryptology - EUROCRYPT 2022 - 41st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Trondheim, Norway, May 30 - June 3, 2022, Proceedings, Part III. Lecture Notes in Computer Science, vol. 13277, pp. 582–612. Springer (2022). https://doi.org/10.1007/978-3-031-07082-2_21, https://doi.org/10.1007/978-3-031-07082-2_21
43. Tani, S.: Claw finding algorithms using quantum walk. Theoretical Computer Science **410**(50), 5285–5297 (2009)
44. Verbel, J., Baena, J., Cabarcas, D., Perlner, R., Smith-Tone, D.: On the complexity of "superdetermined" minrank instances. In: Post-Quantum Cryptography: 10th International Conference, PQCrypto 2019, Chongqing, China, May 8–10, 2019 Revised Selected Papers 10. pp. 167–186. Springer (2019)

A Low-rank point sampling via min-rank step

The sampling step can be done by either the min-rank method, or the graph-walking method. The graph-walking method involves q, so it works best for relatively small q. When q is large, the min-rank method is more effective. To use min-rank to do sampling requires a bit of twist, so we record the idea here.

Suppose we wish to sample a rank-r point $\hat{\mathbf{v}} \in \mathbb{P}(\mathbb{F}_q^n)$ for an alternating trilinear form ϕ, and suppose that there are q^k-many rank-r projective points for a random ϕ. To sample such points, we make a heuristic assumption that the first k coordinates of these rank-r points are in uniform random. Therefore, to sample one point, we can randomly choose the first k coordinates and then resort to the min-rank procedure.

More specifically, for $i \in [n]$, let A_i be the alternating matrix representing the bilinear form ϕ_{e_i}, where e_i is the ith standard basis vector. Let x_i, $i \in [n]$, be formal variables, and set $A = \sum_{i\in[n]} x_i A_i$. So for $i \in [1 \dots k]$, let $x_i = \alpha_i x_1$,

where $\alpha_i \in_R \mathbb{F}_q$. This gives us a min-rank instance with $n-k$ matrices of size $n \times n$.

To estimate the min-rank cost, we use the algorithm from [6]. Consider an (n, K, r) minrank instance, namely finding a rank-r matrix in a linear span of K $n \times n$ matrices. First, we need to compute the smallest b such that $b < r+2$ and

$$\binom{n}{r}\binom{K+b-1}{b} - 1 \leq \sum_{i=1}^{b} (-1)^{i+1} \binom{n}{r+i}\binom{n+i-1}{i}\binom{K+b-i-1}{b-i}.$$

Based on this b, the complexity is estimated as

$$O\big(K \cdot (r+1) \cdot (\binom{n}{r} \cdot \binom{K+b-1}{b})^2\big).$$

For concrete values of n, $K = n-k$ and r, the above formulas allow for the estimation of the concrete security parameters.

Note that the min-rank instance above has some structural constraints due to alternating trilinear forms. As pointed out in [12], such structures should impact the min-rank algorithm from [6] adversely. Still, we use the estimates from [6] as they should serve as a lower bound. We also compare the estimates from [6] with the analysis of the Kipnis–Shamir modelling [35] in [44], and found the ones from [6] are lower.