Attacking and defending post-quantum cryptography candidates

Thales Paiva[1], **Routo Terada**[1]

[1]IME – USP
São Paulo – SP – Brasil

thalespaiva@larc.usp.br

rt@ime.usp.br

Abstract. *This dissertation, which is written as a collection of papers, presents original contributions to the security and implementation of three post-quantum cryptography candidates: HQC, PKP and BIKE. Both HQC and BIKE are code-based key encapsulation mechanisms that were selected as alternate candidates in NIST's post-quantum standardization process. The Permuted Kernel Problem (PKP) is an NP-hard combinatorial problem that can be used to instantiate post-quantum digital signature schemes. The first contribution is a timing attack against HQC that allows an attacker to recover the secret key after recording the decryption time of around 400 million ciphertexts, for 128 bits of security. The second contribution consists of the first attack targeting a generalization of PKP for small fields. For 80-bit security parameters, the attack is able to recover a fraction* 2^{-40} *of the keys using only* 2^{48} *operations, and about 7.2% of the keys using* 2^{62} *operations. The third and last contribution consists of a new decryption algorithm for BIKE. Our constant-time implementation of this algorithm achieves speedups of 1.18, 1.29 and 1.47, with respect to state-of-the-art decryption algorithms, for security levels 128, 192 and 256, respectively.*

1. Introduction

The focus of this dissertation[1] is on evaluating the security of post-quantum cryptography candidates. More specifically, we are interested in side-channel attacks, classical cryptanalysis and secure implementation of the candidate schemes. The three schemes for which we identified research opportunities and managed to improve the state of the art are HQC [Melchor et al. 2018], BIKE [Aragon et al. 2021], and signatures based on the Permuted Kernel Problem when defined over binary fields [Lampe and Patarin 2011]. This dissertation is based on the following three papers containing independent contributions related to these schemes.

1. Thales Bandiera Paiva and Routo Terada. A timing attack on the HQC encryption scheme. In Kenneth G. Paterson and Douglas Stebila, editors, *Selected Areas in Cryptography – SAC 2019*, pages 551–573, Cham, 2020. Springer International Publishing.
2. Thales Bandiera Paiva and Routo Terada. Cryptanalysis of the binary permuted kernel problem. In *International Conference on Applied Cryptography and Network Security – ACNS 2021*, pages 396–423. Springer, 2021.

[1]https://www.teses.usp.br/teses/disponiveis/45/45134/tde-30012023-200916/pt-br.php

3. Thales Bandiera Paiva and Routo Terada. Faster constant-time decoder for MDPC codes and applications to BIKE KEM. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2022.

During the period when this dissertation was written, we also worked on the two papers below, but these lie outside the dissertation's scope.

1. Thales Bandiera Paiva, Javier Navaridas, and Routo Terada. Robust covert channels based on DRAM power consumption. In *International Conference on Information Security*, pages 319–338. Springer, 2019.
2. Thales Bandiera Paiva, Yaissa Siqueira, Daniel Macêdo Batista, Roberto Hirata, and Routo Terada. BGP anomalies classification using features based on AS relationship graphs. In *2021 IEEE Latin-American Conference on Communications (LATINCOM)*, pages 1–6. IEEE, 2021.

In the next sections, we discuss the main techniques and results related to each of the three cryptography papers on which this dissertation is based.

2. A timing attack on HQC

Hamming Quasi-Cyclic (HQC) [Melchor et al. 2018] is a code-based public-key encryption scheme. It is based on the hardness of the quasi-cyclic syndrome decoding problem, a conjectured hard problem from Coding Theory. It offers reasonably good parameters, with better key sizes than the classical McEliece scheme [McEliece 1978, Bernstein et al. 2019, Albrecht et al. 2018], but without relying on codes with a secret sparse structure, such as QC-MDPC [Misoczki et al. 2013] and QC-LDPC [Baldi 2014].

The scheme uses an error correction code C as a public parameter. The secret key is a sparse vector, while the public key is its syndrome with respect to a systematic quasi-cyclic matrix chosen at random, together with the description of this matrix. To encrypt a message, the sender first encodes it with respect to the public code C, then adds to it a binary error vector which appears to be random for anyone who is not the intended receiver. The receiver, using the sparseness of her secret key, is able transform the ciphertext in such a way to significantly reduce the weight of the error vector. Then, the receiver can use the efficient decoding procedure for C to correct the remaining errors of the transformed ciphertext to recover the message. The code C proposed by Aguilar-Melchor et al. [Melchor et al. 2018] is a tensor code between a BCH code and a repetition code. One problem of the HQC implementation submitted to NIST is that the decoder [Joiner and Komo 1995] for the BCH code is not constant-time, and depends on the weight of the error it corrects.

We present the first timing attack on HQC using only valid ciphertexts. It is based on the observation that the secret spectrum, that is, the set of cyclic distances between non-null entries of the secret key, can be learned from the total decryption time. The relation is illustrated in Figure 1. We analyze in detail the reason behind the observed patterns and provide an explanation of the behavior based on a mathematical analysis of HQC's error vectors. Furthermore, we show a simple algorithm that allows us to classify distances inside the secret spectrum based on the neighboring points.

We then show how an efficient randomized variant of Guo's et al. algorithm for key reconstruction can be used to recover the key when partial information on the secret

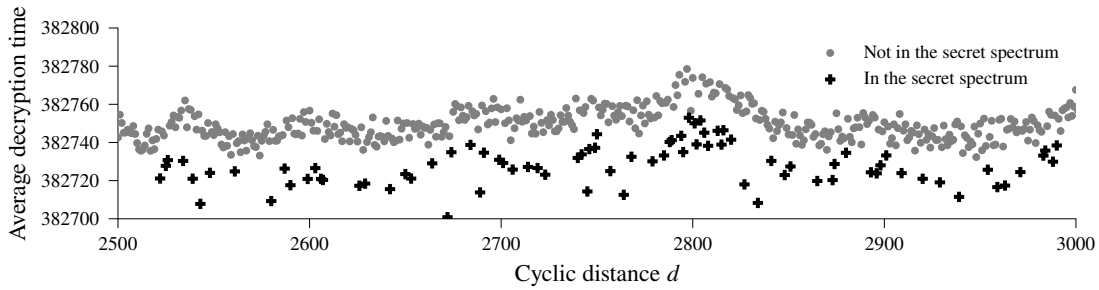

Figure 1. The average decryption time for distances that are inside or outside the secret spectrum.

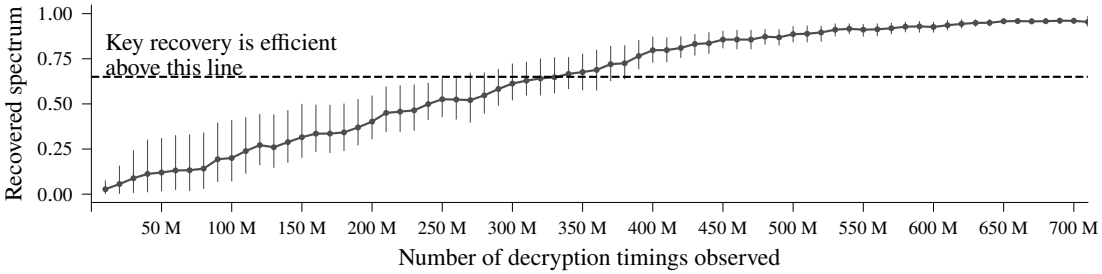

Figure 2. The number of ciphertexts needed for the attack.

key's spectrum is available. Figure 2 shows that an attacker who observes the decryption time for around 400 million valid ciphertexts can successfully recover the key. The source code used for the attack can be found in `https://www.ime.usp.br/~tpaiva/sources/attack-hqc/`.

3. Cryptanalysis of the binary variant of the permuted kernel problem

One interesting problem for building post-quantum signatures is the permuted kernel problem (PKP), which was recently used to instantiate PKP-DSS [Beullens et al. 2019]. This signature scheme is obtained by applying the Fiat-Shamir [Fiat and Shamir 1986] transform on Shamir's PKP-based identification scheme[Shamir 1989], which dates back from 1989. Given a matrix $\mathbf{A}$ and a vector $\mathbf{v}$ with elements in a finite field, PKP asks to find a permutation of the entries of $\mathbf{v}$ that is in the kernel of $\mathbf{A}$. PKP is NP-hard and there is no known quantum algorithm which have a significant advantage over classical algorithms when solving the problem.

In 2010, Lampe and Patarin proposed a generalized version of PKP, in which vector $\mathbf{v}$ is substituted by a matrix $\mathbf{V}$. This enabled them to instantiate PKP in the binary field, without an apparent security loss. At the time, this binary variant presented some interesting advantages such as a reduction in the number of operations. To estimate the security of binary PKP, the authors considered the best attacks against the original PKP, with minor adjustments to make they work against the binary variant. They noted that none of the available attacks was significantly faster against binary PKP.

However, the use of binary coefficients for matrix $\mathbf{A}$ comes with a security risk. We observed that that low weight binary words occur with non-negligible probability in

two public spaces: one is generated by the matrix $\mathbf{A}$ while the other is generated by the kernel of $\mathbf{V}$. It is then possible to devise an attack against binary PKP by matching these low weight codewords using subgraph isomorphism algorithms, and recovering the secret permutation from these matchings.

We present the first attack that specifically targets the binary PKP. Unlike previous attacks, which need a very large amount of memory to run efficiently, our attack uses only a negligible amount of memory. This allows us to provide a concrete implementation of the attack. We provide a detailed analysis of the attack, and then compare these results with the attack's performance in practice. As an example of the power of the attack: for binary PKP parameters originally targeting 80 bits of security, it uses about 2^{63} CPU cycles to fully recover the key, while the best previously known attack [Koussa et al. 2019] needs about 2^{76} matrix-vector multiplications and 2^{50} bytes of memory.

The attack proceeds as follows. By the definition of the binary PKP, we are given the public matrices $\mathbf{A}$ and $\mathbf{V}$ and we want to find the secret permutation π such that $\mathbf{AV}_\pi = \mathbf{0}$. Let $C_\mathbf{A}$ and $C_\mathbf{K}$ be the binary codes generated by $\mathbf{A}$ and $\mathbf{K}$, respectively, where $\mathbf{K}$ is the left kernel matrix of $\mathbf{V}$. Fix an integer w small enough so that we can build the sets $\mathcal{L}^w_\mathbf{A}$ and $\mathcal{L}^w_\mathbf{K}$ consisting of all the codewords of weight w in $C_\mathbf{A}$ and $C_\mathbf{K}$, correspondingly. Notice that, since $\mathbf{AV}_\pi = \mathbf{0}$, then $\mathcal{L}^w_\mathbf{A} \subset \mathcal{L}^w_{\pi(\mathbf{K})} = \left\{\mathbf{u}_\pi : \mathbf{u} \in \mathcal{L}^w_\mathbf{K}\right\}$.

This idea gives the following simple algorithm to find the secret permutation π. First find a subset S of $\mathcal{L}^w_\mathbf{K}$, such that, for some permutation τ, $\mathcal{L}^w_\mathbf{A} = \{\mathbf{u}_\tau : \mathbf{u} \in S\}$. Then, test if the corresponding column permutation τ is valid, that is, if $\mathbf{AV}_\tau = \mathbf{0}$. If τ is valid, return it as π. Otherwise, restart the search. Figure 3 can be useful for visualizing the relationship between the two sets of codewords, which is the core of the attack.

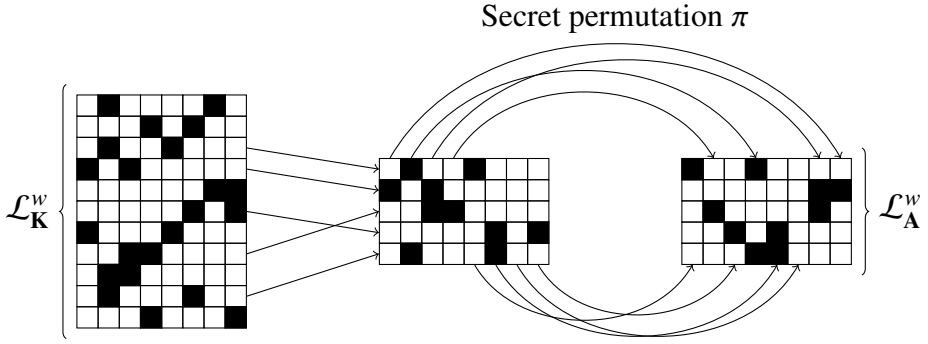

Figure 3. Illustration of the relationship between $\mathcal{L}^w_\mathbf{A}$ and $\mathcal{L}^w_\mathbf{K}$ with respect to the secret column permutation π for codewords of weight $w = 2$. White and black squares represent null and non-null entries, respectively.

Even though it has a rather simple description, we need to carefully deal with the following two problems. The first one is that matching vectors in $\mathcal{L}^w_\mathbf{A}$ and a subset of $\mathcal{L}^w_\mathbf{K}$ is closely related to the subgraph isomorphism problem, which is NP-hard [Garey and Johnson 1979]. The second problem is that, since we are dealing with sparse codewords, there may be a large number of repeated columns in $\mathcal{L}^w_\mathbf{A}$. This could potentially make it infeasible to find the secret permutation π because of the combinatorial explosion on the number of possible permutations between columns.

w	$\left\lvert\mathcal{L}_{\mathbf{A}}^{w}\right\rvert$	Fraction of vulnerable keys	Theoretical estimate on the work factor (matrix-vector products)	Empirical estimate (clock cycles)
5	14	0	$2^{39.46}$	$2^{34.39}$
6	11	$2^{-43.32}$	$2^{49.75}$	$2^{47.58}$
7	10	$2^{-17.86}$	$2^{55.84}$	$2^{48.62}$
8	9	$2^{-2.88}$	$2^{62.28}$	$2^{60.54}$
9	9	$2^{-0.00}$	$2^{64.16}$	$2^{62.31}$

Table 1. Estimates on the attack complexity together with the fraction of keys that are subject to the attack.

We solve these problems by combining simulations and algorithmic analysis. This allows us to provide both accurate concrete estimates on the number of cycles needed for the attack and also a careful asymptotic analysis, which show that our attack performs better than the previous best attack on PKP by Koussa et al. [Koussa et al. 2019]. Table 1 shows the complexity of the attack together with the fraction of vulnerable keys when different attack parameters $\left(w, \left\lvert\mathcal{L}_{\mathbf{A}}^{w}\right\rvert\right)$ are used. Notice that, when we increase parameter w, we need to compute a smaller set $\left\lvert\mathcal{L}_{\mathbf{A}}^{w}\right\rvert$ of vectors of weight w to be able to perform the attack, and, while the attack complexity is larger, it can attack a larger fraction of keys. We also remark that the cost of identifying weak keys is very low. The source code, together with a demo of the attack, are available at `https://github.com/thalespaiva/attack-on-binary-pkp`.

4. A novel decoder for BIKE

BIKE [Aragon et al. 2021] is a code-based key encapsulation mechanism (KEM) selected as an alternate candidate for the NIST post quantum standardization process. The scheme consists of a variant of the Niederreiter [Niederreiter 1986] scheme using quasi-cyclic moderate-density parity-check (QC-MDPC) codes instead of Goppa codes. As such, BIKE can be seen as a refinement of Misoczki's et al. QC-MDPC McEliece [Misoczki et al. 2013].

The use of QC-MDCP [Misoczki et al. 2013] codes yields two advantages. The first one is that the public key is much smaller, since one needs only one row to represent a quasi-cyclic matrix in systematic form. The second is that matrix multiplication, and thus encoding, is much faster for quasi-cyclic matrices. However, QC-MDPC codes comes with an important disadvantage: their decoding algorithms have a non-zero probability of failure. This fact was exploited in the famous GJS [Guo et al. 2016] key-recovery reaction attack, that provided the ground for side-channel attacks against QC-MDPC [Rossi et al. 2017] and further attacks against other code-based encryption schemes [Samardjiska et al. 2019, Fabšič et al. 2017].

To deal with this problem, BIKE's original proposal [Aragon et al. 2017] used ephemeral keys. However, recent approaches on obtaining negligible decryption failure rate (DFR) [Tillich 2018, Sendrier and Vasseur 2020, Vasseur 2021], together with Hofheinz et al. [Hofheinz et al. 2017] CCA security conversions that accounts for decryption errors, motivated BIKE proponents to consider key-reuse. In particular, Sendrier

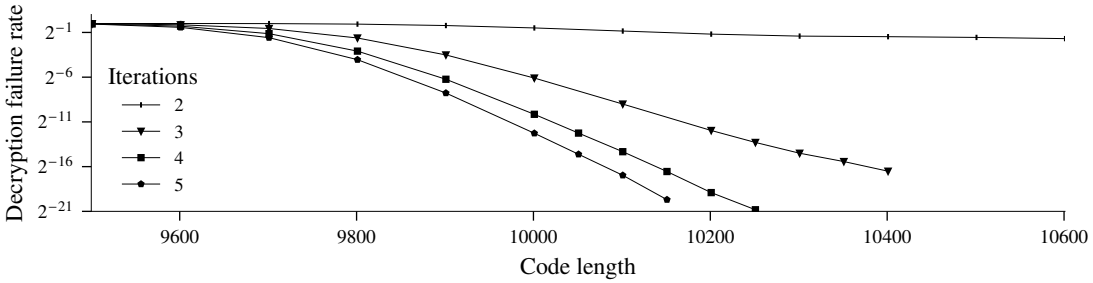

Figure 4. The non-concavity of the log DFR curve of the BGF decoder when less than 5 iterations are used, considering parameter set BIKE Level 1.

and Vasseur [Sendrier and Vasseur 2020, Vasseur 2021] propose a framework that, under reasonable assumptions, allows them to find parameters where the DFR should be negligible using experiments and statistical analysis. This framework was used in BIKE's last revision [Aragon et al. 2021], which uses the state-of-the-art BGF decoder [Drucker et al. 2020, Drucker et al. 2019] with parameters that supposedly achieve negligible DFR.

While trying to improve BGF's performance, we noticed two limitations. The first one is that its performance cannot be improved by considering a lower number of iterations, otherwise it breaks the main hypothesis for using Vasseur's extrapolation framework [Vasseur 2021]: the log of the decoder's DFR should be concave with respect to the code length used. Figure 4 shows the non-concavity of BGF's DFR curves when 2 to 4 iterations are used The second limitation is that some of its iterations can be made more efficient by merging them into one iteration. After analyzing BGF's strengths and weaknesses, we were able to derive not only a more efficient decoder, but also seems to provide slightly stronger security arguments.

We propose a new decoding algorithm for QC-MDPC codes, called PickyFix. This decoder uses two auxiliary iterations that are significantly different from previous approaches: the FixFlip iteration, which flips a fixed number of bits, and the PickyFlip iteration, which uses different thresholds to flip ones and zeros. These iterations allow PickyFix to work with a lower number of iterations than BGF, because they yield concave log DFR curves with respect to the code length, as illustrated in Figure 5. Our constant-

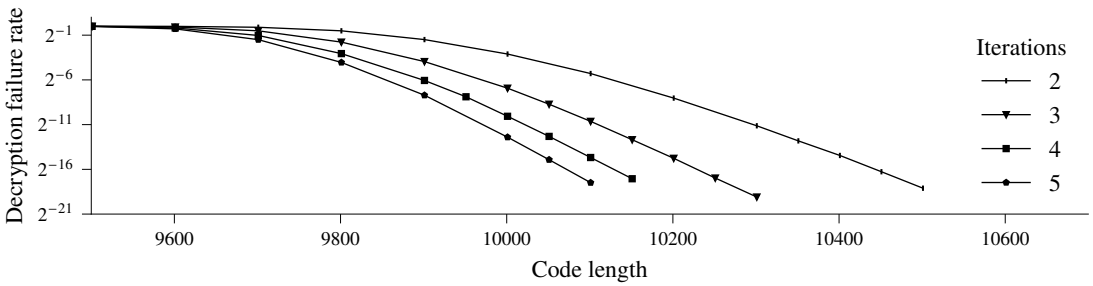

Figure 5. The apparent concavity of the log DFR curve of the PickyFix considering parameter set BIKE Level 1, for multiple number of iterations.

time implementation, makes PickyFix achieve speedups of 1.18, 1.29, and 1.47 for the security levels 128, 192 and 256, respectively. The code and data are publicly available at `https://github.com/thalespaiva/pickyfix`.

Acknowledgments

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. This research is part of the INCT of the Future Internet for Smart Cities funded by CNPq proc. 465446/2014-0, Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001, FAPESP proc. 14/50937-1, proc. 15/24485-9, and CAPES 23038.000776/2017-54.

References

Albrecht, M., Cid, C., Paterson, K. G., Tjhai, C. J., and Tomlinson, M. (2018). NTS-KEM.

Aragon, N., Barreto, P. S. L. M., Bettaieb, S., Bidoux, L., Blazy, O., Deneuville, J.-C., Gaborit, P., Ghosh, S., Gueron, S., Güneysu, T., Aguilar-Melchor, C., Misoczki, R., Persichetti, E., Richter-Brockmann, J., Sendrier, N., Tillich, J.-P., Vasseur, V., and Zémor, G. (2021). BIKE: Bit flipping key encapsulation. `https://bikesuite.org/files/v4.2/BIKE_Spec.2021.09.29.1.pdf`.

Aragon, N., Barreto, P. S. L. M., Bettaieb, S., Bidoux, L., Blazy, O., Deneuville, J.-C., Gaborit, P., Gueron, S., Guneysu, T., Melchor, C. A., Misoczki, R., Persichetti, E., Sendrier, N., Tillich, J.-P., and Zémor, G. (2017). BIKE: Bit flipping key encapsulation. `https://bikesuite.org/files/BIKE.2017.11.30.pdf`.

Baldi, M. (2014). *QC-LDPC Code-Based Cryptosystems*, pages 91–117. Springer International Publishing, Cham.

Bernstein, D. J., Chou, T., Lange, T., Misoczki, R., Niederhagen, R., Persichetti, E., Schwabe, P., Szefer, J., and Wang, W. (2019). Classic McEliece: conservative code-based cryptography.

Beullens, W., Faugère, J.-C., Koussa, E., Macario-Rat, G., Patarin, J., and Perret, L. (2019). PKP-based signature scheme. In *International Conference on Cryptology in India*, pages 3–22. Springer.

Drucker, N., Gueron, S., and Kostic, D. (2019). On constant-time QC-MDPC decoding with negligible failure rate. *IACR Cryptol. ePrint Arch.*, 2019:1289.

Drucker, N., Gueron, S., and Kostic, D. (2020). QC-MDPC decoders with several shades of gray. In *International Conference on Post-Quantum Cryptography*, pages 35–50. Springer.

Fabšič, T., Hromada, V., Stankovski, P., Zajac, P., Guo, Q., and Johansson, T. (2017). A reaction attack on the QC-LDPC McEliece cryptosystem. In *International Workshop on Post-Quantum Cryptography*, pages 51–68. Springer.

Fiat, A. and Shamir, A. (1986). How to prove yourself: Practical solutions to identification and signature problems. In *Conference on the Theory and Application of Cryptographic Techniques*, pages 186–194. Springer.

Garey, M. R. and Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York.

Guo, Q., Johansson, T., and Stankovski, P. (2016). A key recovery attack on MDPC with CCA security using decoding errors. In *22nd Annual International Conference on the Theory and Applications of Cryptology and Information Security (ASIACRYPT)*.

Hofheinz, D., Hövelmanns, K., and Kiltz, E. (2017). A modular analysis of the Fujisaki-Okamoto transformation. In *Theory of Cryptography Conference*, pages 341–371. Springer.

Joiner, L. L. and Komo, J. J. (1995). Decoding binary BCH codes. In *Proceedings IEEE Southeastcon'95. Visualize the Future*, pages 67–73. IEEE.

Koussa, E., Macario-Rat, G., and Patarin, J. (2019). On the complexity of the Permuted Kernel Problem. *IACR Cryptology ePrint Archive*, 2019:412.

Lampe, R. and Patarin, J. (2011). Analysis of some natural variants of the PKP algorithm. *IACR Cryptology ePrint Archive*, 2011:686.

McEliece, R. J. (1978). A public-key cryptosystem based on algebraic coding theory. *Deep Space Network Progress Report*, 44:114–116.

Melchor, C. A., Aragon, N., Bettaieb, S., Bidoux, L., Blazy, O., Deneuville, J.-C., Gaborit, P., Persichetti, E., Zémor, G., and Bourges, I.-C. (2018). Hamming quasi-cyclic (HQC). Technical report, Technical report, National Institute of Standards and Technology.

Misoczki, R., Tillich, J.-P., Sendrier, N., and Barreto, P. S. L. M. (2013). MDPC-McEliece: New McEliece variants from moderate density parity-check codes. In *Information Theory Proceedings (ISIT), 2013 IEEE International Symposium on*, pages 2069–2073. IEEE.

Niederreiter, H. (1986). Knapsack-type cryptosystems and algebraic coding theory. *Problems of Control and Information Theory*, 15(2):159–166.

Rossi, M., Hamburg, M., Hutter, M., and Marson, M. E. (2017). A side-channel assisted cryptanalytic attack against QcBits. In *International Conference on Cryptographic Hardware and Embedded Systems*, pages 3–23. Springer.

Samardjiska, S., Santini, P., Persichetti, E., and Banegas, G. (2019). A reaction attack against cryptosystems based on LRPC codes. In *International Conference on Cryptology and Information Security in Latin America*, pages 197–216. Springer.

Sendrier, N. and Vasseur, V. (2020). About low DFR for QC-MDPC decoding. In *PQCrypto 2020-Post-Quantum Cryptography 11th International Conference*, volume 12100, pages 20–34. Springer.

Shamir, A. (1989). An efficient identification scheme based on permuted kernels. In *Conference on the Theory and Application of Cryptology*, pages 606–609. Springer.

Tillich, J.-P. (2018). The decoding failure probability of MDPC codes. In *2018 IEEE International Symposium on Information Theory (ISIT)*, pages 941–945. IEEE.

Vasseur, V. (2021). QC-MDPC codes DFR and the IND-CCA security of BIKE. Cryptology ePrint Archive, Report 2021/1458. `https://ia.cr/2021/1458`.