POLITECNICO DI MILANO

Dipartimento di
Elettronica, Informazione
e Bioingegneria

Bit-flipping Decoder Failure Rate Estimation for (v,w)-regular Codes

Alessandro Annechini, Alessandro Barenghi, Gerardo Pelosi

Fifth PQC Standardization Conference - 12th April 2024

Outline

Context: Code-based KEMs with iterative decoding

- Current 4th round candidate BIKE is built on sparse QC random codes (QC-MDPC)
  - QC-MDPCs are decoded with an iterative, fixed point procedure
  - Achieved DFR depends on both the code and the decoder choice
- Decoding failures reveal information on the private key, breaking IND-CCA2
  - Estimating DFR in closed-form has proven to be challenging
  - [WWW23]: estimates of DFR for BIKE w/ BGF decoder were optimistic

Contributions

1. Closed form estimate of avg. DFR for (v, w)-regular codes w/ 2-iteration BF decoder
2. Analyze the code parameters for a IND-CCA2 QC-MDPC scheme

- Accepted at IEEE International Symposium on Information Theory (ISIT 2024)

(v, w)-regular and QC-MDPC binary codes

(v, w)-regular codes

- Binary block codes with length n, dimension k and redundancy $n - k = r$
- Each column $h_{:,j}$ of the parity check matrix H has Hamming weight $\mathtt{wt}(h_{:,j}) = v$
- Each row of $h_{i,:}$ the parity check matrix H has Hamming weight $\mathtt{wt}(h_{i,:}) = w = \frac{n}{r}v$

QC-MDPC codes

- Subset of (v, w)-regular codes with H defined tiling $p \times p$ circulant matrices, $v \approx \sqrt{n}$
- Both BIKE and LEDAcrypt use $n = n_0 p, r = p$ codes, $w = n_0 v$, p prime, $\mathrm{ord}_p(2) = p - 1$
  - BIKE uses $n_0 = 2$
  - LEDAcrypt uses $n_0 \in \{2, 3, 4\}$ codes

Iterative syndrome decoding: find e, given H and $s = He^T$

Toy example: $n = 10, r = 5, v = 2, w = 4, \mathtt{wt}(e) = 2$

p.c. matrix H

1	0	1	0	0	0	1	0	0	1
0	1	0	1	0	1	0	1	0	0
0	0	1	0	1	0	1	0	1	0
1	0	0	1	0	0	0	1	0	1
0	1	0	0	1	1	0	0	1	0

s

1
1
1
1
0

error e

0	0	1	0	0	0	0	1	0	0

Iterative BF decoder: initialization

$\mathtt{iter} = 0$ completed iter.s; invariant $s_{(\mathtt{iter})} = H(e \oplus \bar{e}_{(\mathtt{iter})})^T$

p.c. matrix H

1	0	1	0	0	0	1	0	0	1
0	1	0	1	0	1	0	1	0	0
0	0	1	0	1	0	1	0	1	0
1	0	0	1	0	0	0	1	0	1
0	1	0	0	1	1	0	0	1	0

$s_{(\mathtt{iter})}$

1
1
1
1
0

error e

?	?	?	?	?	?	?	?	?	?

error est. $\bar{e}_{(\mathtt{iter})}$

0	0	0	0	0	0	0	0	0	0

UPC computation

$\texttt{iter} = 0$ completed iter.s; invariant $s_{(\texttt{iter})} = H(e \oplus \bar{e}_{(\texttt{iter})})^T$

p.c. matrix H	1	0	1	0	0	0	1	0	0	1	1	$s_{(\texttt{iter})}$
	0	1	0	1	0	1	0	1	0	0	1	
	0	0	1	0	1	0	1	0	1	0	1	
	1	0	0	1	0	0	0	1	0	1	1	
	0	1	0	0	1	1	0	0	1	0	0	
error est. $\bar{e}_{(\texttt{iter})}$	0	0	0	0	0	0	0	0	0	0		
$\texttt{upc}$	2	1	2	2	1	1	2	1	1	2		

Flip $\bar{e}_{(\mathtt{iter}),j}$ if $\mathtt{upc}_j \geq \mathtt{th}$

$\mathtt{iter} = 0$ completed iter.s; invariant $s_{(\mathtt{iter})} = H(e \oplus \bar{e}_{(\mathtt{iter})})^T$

p.c. matrix H

1	0	1	0	0	0	1	0	0	1
0	1	0	1	0	1	0	1	0	0
0	0	1	0	1	0	1	0	1	0
1	0	0	1	0	0	0	1	0	1
0	1	0	0	1	1	0	0	1	0

$s_{(\mathtt{iter})}$

1
1
1
1
0

error est. $\bar{e}_{(\mathtt{iter})}$	1	0	1	1	0	0	1	0	0	1
$\mathtt{upc}$	2	1	2	2	1	1	2	1	1	2

Update s as $s \oplus h_{:,j}$ if $\bar{e}_{(\mathtt{iter}),j}$ was flipped

$\mathtt{iter} = 0$ completed iter.s; invariant $s_{(\mathtt{iter})} = H(e \oplus \bar{e}_{(\mathtt{iter})})^T$

p.c. matrix H

1	0	1	0	0	0	1	0	0	1
0	1	0	1	0	1	0	1	0	0
0	0	1	0	1	0	1	0	1	0
1	0	0	1	0	0	0	1	0	1
0	1	0	0	1	1	0	0	1	0

$s_{(\mathtt{iter})}$

1
0
1
0
0

error est. $\bar{e}_{(\mathtt{iter})}$

1	0	1	1	0	0	1	0	0	1

Increment `iter`, if $s_{(\mathtt{iter})} = 0 \Rightarrow e \oplus \bar{e}_{(\mathtt{iter})} = 0$ return $\bar{e}_{(\mathtt{iter})} = 0$

$\mathtt{iter} = 0$ completed iter.s; invariant $s_{(\mathtt{iter})} = H(e \oplus \bar{e}_{(\mathtt{iter})})^T$

p.c. matrix H

1	0	1	0	0	0	1	0	0	1
0	1	0	1	0	1	0	1	0	0
0	0	1	0	1	0	1	0	1	0
1	0	0	1	0	0	0	1	0	1
0	1	0	0	1	1	0	0	1	0

$s_{(\mathtt{iter})}$

1
0
1
0
0

error est. $\bar{e}_{(\mathtt{iter})}$

1	0	1	1	0	0	1	0	0	1

Average DFR estimation technique

Outline of the method

1. Derive the distribution of the syndrome weight $\mathtt{wt}(s)$, $\Pr(\mathcal{W}_t = y)$
2. Derive the probability distribution of number of discrepancies between the error e and its estimate $\bar{e}_{(1)}$ added (d_+) and removed (d_-) by the first iteration
3. Partition error estimate $\bar{e}_{(1)}$ bits after first iteration in classes, derive $\Pr(\mathcal{E}_{(2)} = d)$, and the DFR as $1 - \Pr(\mathcal{E}_{(2)} = 0)$

Bonus from code-specific knowledge (if available)

- [Til18, BBC+23]: Given a specific H compute $\tau(H)$ s.t. for all $0 \leq x \leq \tau(H)$ $\Pr(\mathcal{E}_{(2)} = 0 | \mathcal{E}_{(1)} = x) = 1$, i.e., if $\mathtt{wt}(e \oplus \bar{e}_{(1)}) \leq \tau(H)$ the 2nd iteration converges to $s = 0$

Syndrome weight distribution estimation

Method - Step 1

- Compute distribution of the r.v. $\mathcal{W}_t$ modeling $\mathtt{wt}(s) = \mathtt{wt}(He^T)$, i.e., the syndrome weight of a weight-t error e through a (v, w)-regular p.c. matrix H

Working assumption

- Rows of H are independently and uniformly random drawn from the set of binary vectors with length n and w asserted bits

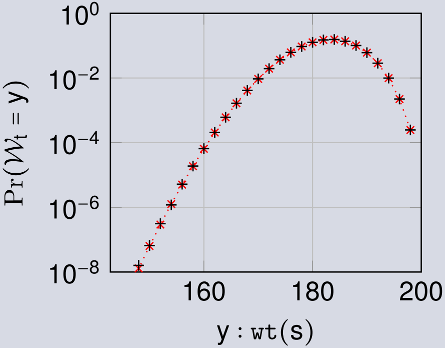
Strategy

- $\mathcal{W}_t$ derived as the result of t steps on a non-homogeneous Markov Chain (MC):
  - MC steps model the effect of adding an asserted bit to e $\Rightarrow$ column of H to s
  - MC transition probabilities derived counting the number of flips taking place in s
  - Initial distribution, i.e., $\mathcal{W}_0$ is simply $\Pr(\mathcal{W}_0 = 0) = 1$

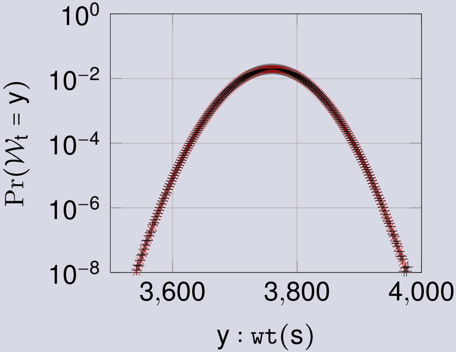
Numerical validation of the distribution of $\mathcal{W}_t$

$(v, n_0 v)$ regular codes with $n = n_0 r$, $\mathtt{wt}(e) = t$, 10^9 samples per pt. (sim $+$, model $\times$)

$n_0 = 2, r = 2200, v = 11, t = 18$



$n_0 = 4, r = 13397, v = 83, t = 66$



First iteration discrepancy distribution estimation

Method - Step 2

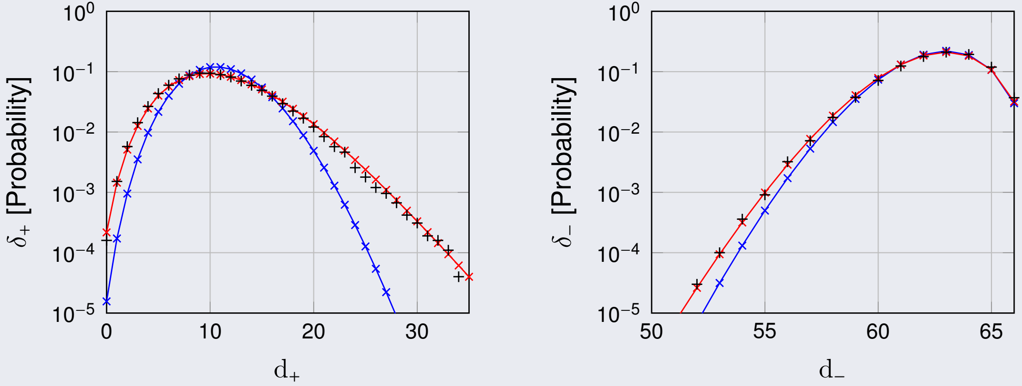
- Model #discrepancies between e and $\bar{\mathsf{e}}_{(1)}$, split into added (d_+) and removed (d_-), as random variables: $\delta_+(\mathrm{d}_+) = \Pr(\mathrm{d}_+$ discrepancies added) and $\delta_-(\mathrm{d}_-)$

Strategy

- Knowing $\mathcal{W}_t$, compute $\Pr(_+$ discrepancies added$|\mathcal{W}_t = w)$ for all $w \in \{0, \dots, n-k\}$ through counting arguments
- Compute probability $p_{\mathtt{unsat}|b}$ that a p.c. equation is unsatisfied, given that a bit involved in it e_j is equal to $b \in \{0, 1\}$
- Compute probability distribution of $\mathtt{upc}_j$ given that e_j is equal to $b \in \{0, 1\}$
- For any 1st iteration threshold $\mathtt{th}_{(1)}$ of choice, compute $\delta_+(\mathrm{d}_+)$ and $\delta_-(\mathrm{d}_-)$
  - Note: The number of discrepancies after the 1st it. is: $\mathcal{E}_{(1)} = t - \mathrm{d}_- + \mathrm{d}_+$

Numerical validation of $\delta_+(\mathrm{d}_+)$ and $\delta_-(\mathrm{d}_-)$

$n_0 = 4$, $p = 13397$, $n = n_0 p$, $k = (n_0 - 1)p$, $v = 83$, $t = 66$, 10^5 samples per point



(sim +, model ×, technique from [BBC+23] ×)

Second iteration failure rate estimation

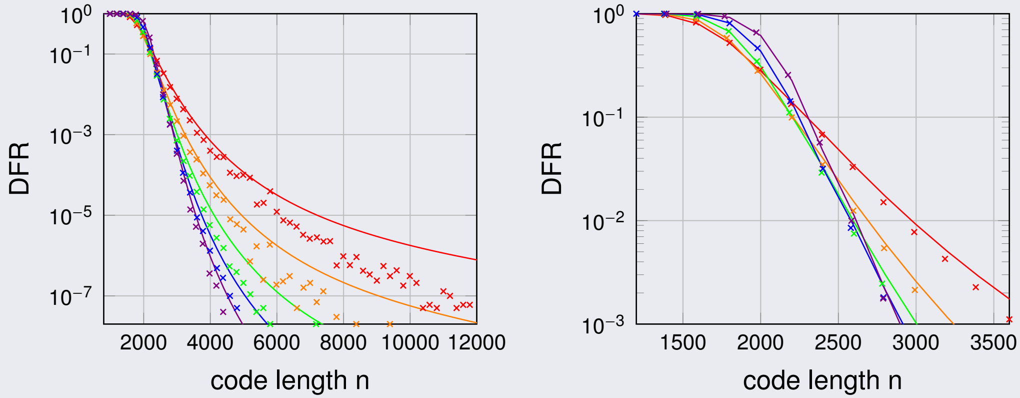
Method - Step 3

- Obtain the second iteration DFR as $1 - \Pr(\mathcal{E}_{(2)} = 0)$

Strategy

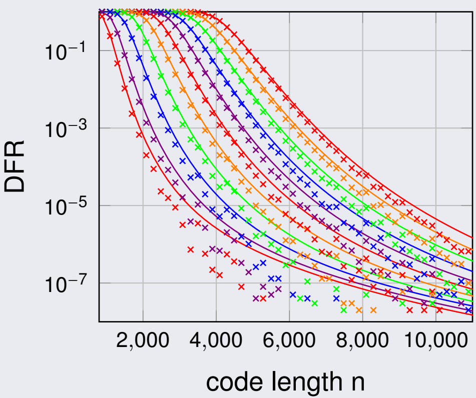
- Partition positions of $\bar{e}_{(1)}$ into $\mathbf{J}_{a,b}$, $a, b \in \{0, 1\}$, $a = e_j$, $b = e_j \oplus \bar{e}_{(1),j}$; for each $\mathbf{J}_{a,b}$:
- Derive the probability that a p.c. equation involving $\bar{e}_{(1),j}, j \in \mathbf{J}_{a,b}$ becomes/stays unsat after the first iteration
- Derive the UPC value distribution in the second iteration for $\bar{e}_{(1),j}, j \in \mathbf{J}_{a,b}$
- Combine the above with the distributions of $|\mathbf{J}_{a,b}|$ (obtained from the ones of d_+ and d_-) to obtain $\Pr(\mathcal{E}_{(2)} = d)$

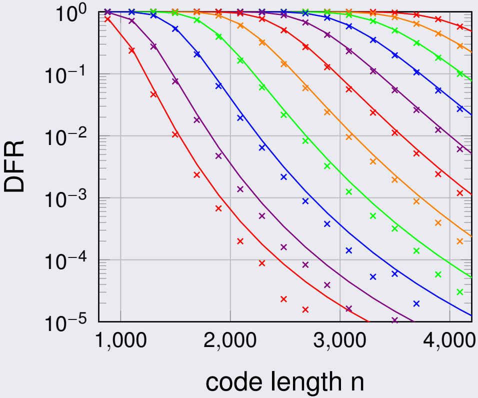
DFR estimate numerical validation - code density



(v, 2v)-regular LDPC codes,$v \in \{9, 11, 13, 15, 17\}$, $\frac{k}{n} = \frac{1}{2}$, $t = 18$, parallel decoder w/ thresholds, $\texttt{th1} = \texttt{th2} = \lceil \frac{v+1}{2} \rceil$. 10^8 decodes or 100 decoding failures per point

DFR estimate numerical validation - error weight





$(v, 2v)$-regular LDPC codes,$t \in \{10, \ldots, 39\}$, $\frac{k}{n} = \frac{1}{2}$, $v = 11$, parallel decoder w/ thresholds, $\texttt{th1} = \texttt{th2} = \lceil \frac{v+1}{2} \rceil$. 10^8 decodes or 100 decoding failures per point

Impact on code based-cryptosystem design

Comparison with previous non-extrapolation estimates on 2 iterations decoder

n_0	p	v	t	min τ(H)	LEDAcrypt	This work
2	23371	71	130	10	2^{-64}	2^{-147}
3	16067	79	83	9	2^{-64}	2^{-139}
4	13397	83	66	8	2^{-64}	2^{-134}
2	28277	69	129	11	2^{-128}	2^{-203}
3	19709	79	82	10	2^{-128}	2^{-198}
4	16229	83	65	9	2^{-128}	2^{-189}

- Computations above consider that for all $0 \leq x \leq \tau(H)$ $\Pr(\mathcal{E}_{(2)} = 0 | \mathcal{E}_{(1)} = x) = 1$
- Computations above done with syndrome independent thresholds
  - Syndrome weight dependent thresholds can also be modeled
  - Employing them yields a more effective decoder, lowering DFR further

Considerations on weak keys

Effects of weak keys

- Weak keys [DGK20, Vas21, ABH+22, WWW23] are p.c. matrices defining codes with poor correction capabilities; they are detrimental to the average DFR
- This work provides a technique to estimate the average DFR over all the possible codes (keypairs), employing a 2-iteration BF decoder
  - This matches the IND-CCA2 requirement [HHK17]

Filtering

- Weak keys from [DGK20, Vas21] can be filtered via pattern-matching
- [BBC+20, BBC+23]: Weak keys are characterized by $\tau(\mathrm{H})$ values definitely below average and can be removed discarding codes with $\tau(\mathrm{H})$ below a chosen threshold $\bar{\tau}$
  - Bonus point: the improvement of the average DFR is automatically quantified in our approach

Concluding remarks

Take-away points

- We provide a closed-form method to estimate the average DFR of a random (v, w)-regular code decoded via 2-iterations parallel BF iterative decoding
- Adopting our approach and tuning BIKE parameters accordingly would yield an IND-CCA2 version of BIKE
- The effect of weak keys is taken into account in our estimates, considering both the case in which they are discarded and the one in which they're not

Ongoing future directions

- Extend the technique to a higher number of parallel BF decoder iterations
- Complete a performance-security optimized design for LEDAcrypt parameters, with syndrome-weight dependent thresholds

Questions?

Thank you for the attention!

Bibliography I

- Sarah Arpin, Tyler Raven Billingsley, Daniel Rayor Hast, Jun Bo Lau, Ray A. Perlner, and Angela Robinson.
  A Study of Error Floor Behavior in QC-MDPC Codes.
  In Jung Hee Cheon and Thomas Johansson, editors, Post-Quantum Cryptography - 13th International Workshop, PQCrypto 2022, Virtual Event, September 28-30, 2022, Proceedings, volume 13512 of Lecture Notes in Computer Science, pages 89–103. Springer, 2022.

- Marco Baldi, Alessandro Barenghi, Franco Chiaraluce, Gerardo Pelosi, and Paolo Santini.
  A failure rate model of bit-flipping decoders for QC-LDPC and QC-MDPC code-based cryptosystems.
  In Pierangela Samarati, Sabrina De Capitani di Vimercati, Mohammad S. Obaidat, and Jalel Ben-Othman, editors, Proceedings of the 17th International Joint Conference on e-Business and Telecommunications, ICETE 2020 - Volume 2: SECRYPT, Lieusaint, Paris, France, July 8-10, 2020, pages 238–249. ScitePress, 2020.

- Marco Baldi, Alessandro Barenghi, Franco Chiaraluce, Gerardo Pelosi, and Paolo Santini.
  LEDAcrypt - version 3.0 Specification.
  [Online] Available: `https://www.ledacrypt.org/documents/LEDAcrypt_v3.pdf`, 2023.

- Nir Drucker, Shay Gueron, and Dusan Kostic.
  On constant-time QC-MDPC decoders with negligible failure rate.
  In Marco Baldi, Edoardo Persichetti, and Paolo Santini, editors, Code-Based Cryptography - 8th International Workshop, CBCrypto 2020, Zagreb, Croatia, May 9-10, 2020, Revised Selected Papers, volume 12087 of Lecture Notes in Computer Science, pages 50–79. Springer, 2020.

- Dennis Hofheinz, Kathrin Hövelmanns, and Eike Kiltz.
  A modular analysis of the fujisaki-okamoto transformation.
  In Yael Kalai and Leonid Reyzin, editors, Theory of Cryptography - 15th International Conference, TCC 2017, Baltimore, MD, USA, November 12-15, 2017, Proceedings, Part I, volume 10677 of Lecture Notes in Computer Science, pages 341–371. Springer, 2017.

Bibliography II

- Jean-Pierre Tillich.
  The Decoding Failure Probability of MDPC Codes.
  In 2018 IEEE International Symposium on Information Theory, ISIT 2018, Vail, CO, USA, June 17-22, 2018, pages 941–945. IEEE, 2018.

- Valentin Vasseur.
  Post-quantum cryptography: a study of the decoding of QC-MDPC codes. (Cryptographie post-quantique : étude du décodage des codes QC-MDPC).
  PhD thesis, University of Paris, France, 2021.

- Tianrui Wang, Anyu Wang, and Xiaoyun Wang.
  Exploring decryption failures of BIKE: new class of weak keys and key recovery attacks.
  In Helena Handschuh and Anna Lysyanskaya, editors, Advances in Cryptology - CRYPTO 2023 - 43rd Annual International Cryptology Conference, CRYPTO 2023, Santa Barbara, CA, USA, August 20-24, 2023, Proceedings, Part III, volume 14083 of Lecture Notes in Computer Science, pages 70–100. Springer, 2023.