A status update on NIST's post-quantum standardization effort

University of Ottawa
28/08/2020

Nina Bindel
University of Waterloo | IQC Institute for Quantum Computing

Outline

NIST standardization effort

- 3rd round candidates and timeline
- Security estimations (of LWE)

Decryption failures of PKEs/KEMs

- Definition
- Attack

NIST PQ Standardization Effort - Overview

Functionality

- Signature schemes
  - EUF-CMA
- PKEs/KEMs
  - IND-CPA
  - IND-CCA

Evaluation criteria

- Security
- Performance
- Other

Most information taken from Dustin Moody‘s talk during PQCrypto 2019 in Chongqing, China

NIST PQ Standardization Effort - Overview

Functionality

Signature schemes
- EUF-CMA

PKEs/KEMs
- IND-CPA
- IND-CCA

Evaluation criteria

Security

Performance

Other

Level	As hard as …
I	AES128 (exhaustive key search)
II	SHA256 (collision resistance)
III	AES192 (exhaustive key search)
IV	SHA384 (collision resistance)
V	AES256 (exhaustive key search)
	… against classical and quantum algorithms

- perfect forward secrecy
- resistance to side-channel attacks
- multi-key attacks
- resistance to misuse

Most information taken from Dustin Moody's talk during PQCrypto 2019 in Chongqing, China

NIST PQ Standardization Effort - Overview

Functionality

Signature schemes
- EUF-CMA

PKEs/KEMs
- IND-CPA
- IND-CCA

Evaluation criteria

Security

Performance

Other

- “measured on various classical platforms”
- Asked for reference implemetation
- Recommended to provided AVX optimized implementation

NIST PQ Standardization Effort - Overview

Functionality

Signature schemes
- EUF-CMA

PKEs/KEMs
- IND-CPA
- IND-CCA

Evaluation criteria

Security

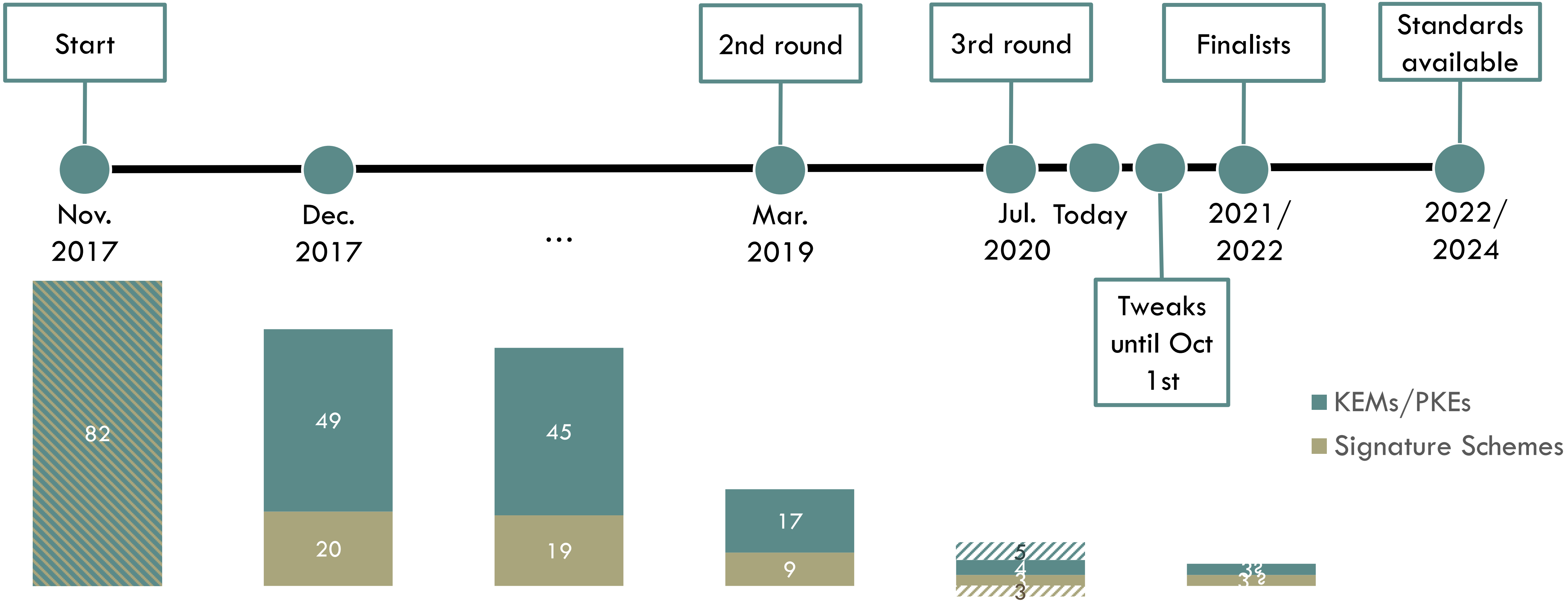
Performance

Other

- Suitability for hybrid solutions
- Simplicity
- Flexibility, e.g. usage in TLS, IPSec, certificates
- Security against physical attacks
- Intellectual property
- …

Most information taken from Dustin Moody's talk during PQCrypto 2019 in Chongqing, China

NIST PQ Standardization Effort - Timeline



“NIST does not feel the need to choose these standards all at once but will rather prioritize those schemes which seem closest to being ready for standardization and wide adoption. NIST feels this strategy best serves to balance the desire for diversity with the need for all standards to be thoroughly vetted before they are released.”

Finalists vs Alternate Candidates

Finalists are […] the most promising to fit the majority of use cases and most likely to be ready for standardization soon after the end of the third round.

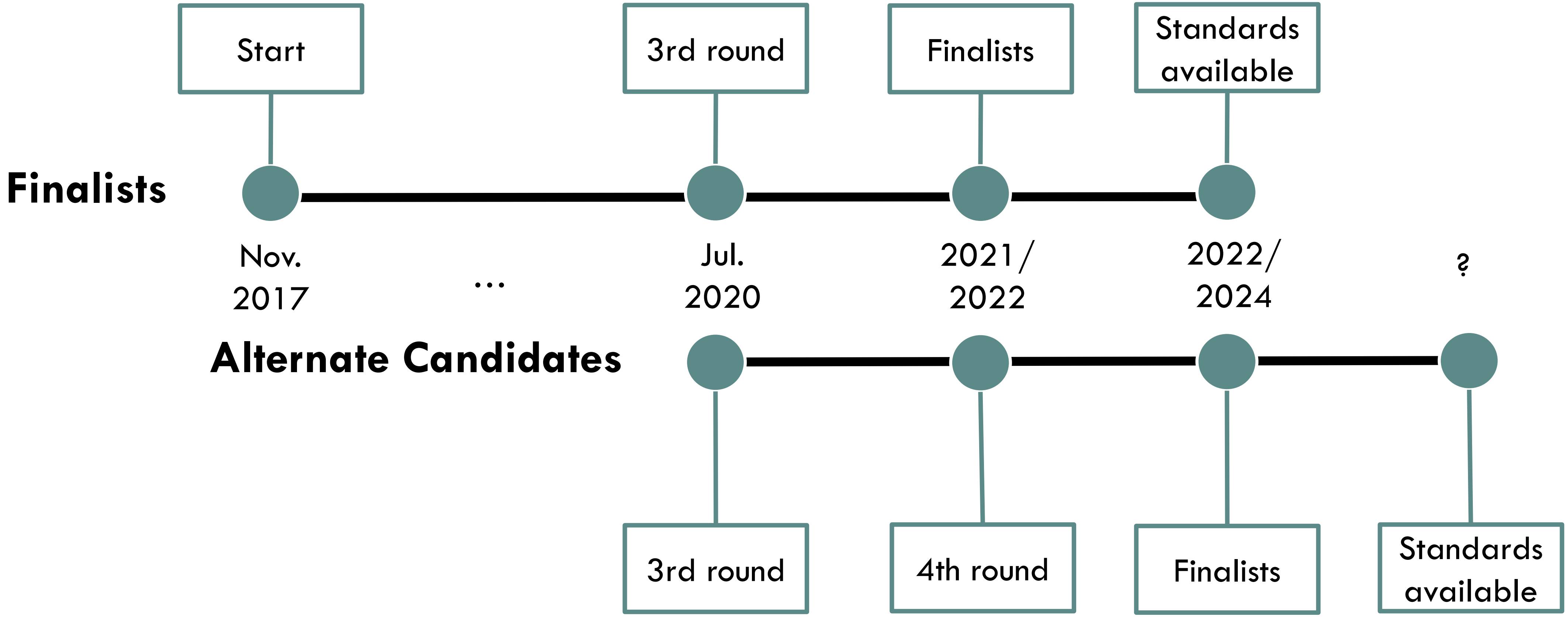
Alternate candidates are […] candidates for future standardization, most likely after another round of evaluation.

- Low performance but high confidence in their security
- Acceptable performance but not sufficient confidence in their security
- Desire for diversity
- Potential for further improvement.

3rd Round Candidates

	Code-based	Lattice-based	Multivariate	Isogeny-based	Symmetric-based
5	BIKE HQC	FrodoKEM NTRU Prime		SIKE	
4	Classic McEliece	CRYSTALS-KYBER NTRU SABER			
3		CRYSTALS-DILITHIUM FALCON	Rainbow		
3			GeMSS		Picnic SPHINCS+

NIST PQ Standardization Effort – Timeline revisited



Selection of 3rd Round Candidates

Security

Attack exploiting LAC's error correction

"Although LAC has been modified to resist those attacks, NIST believes that further study is needed before it can be confident that there are no remaining vulnerabilities in the LAC design. Thus, […], LAC was not selected to move on to the third round."

Similarity (Performance)

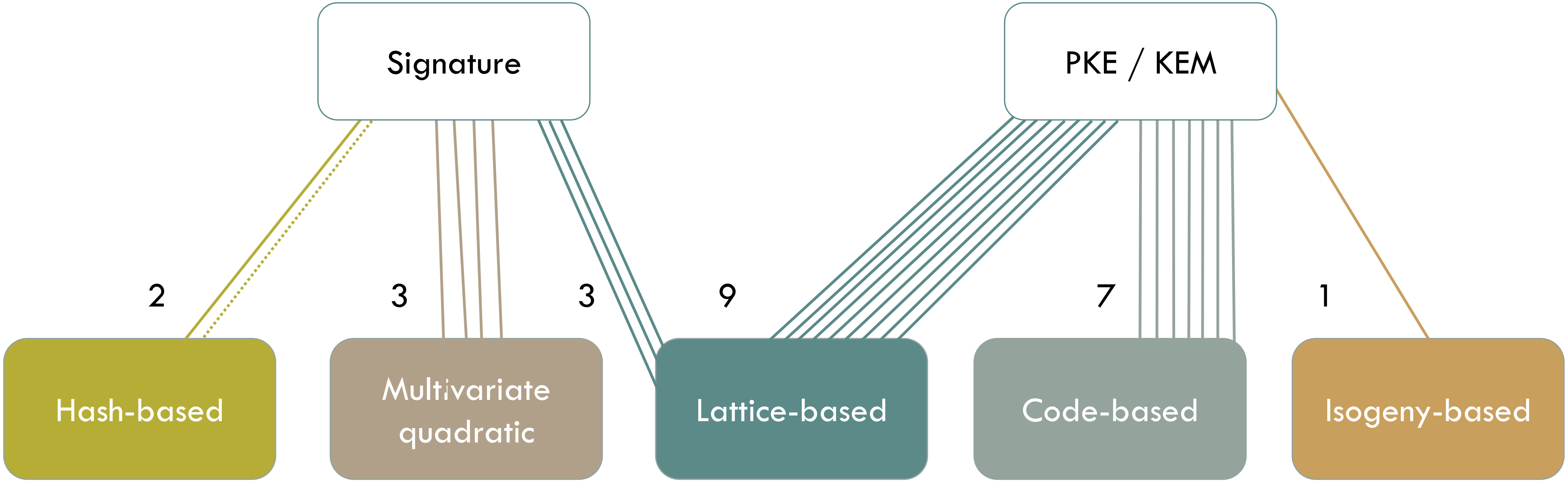
NewHope vs Kyber

- Similar design, except
  - Kyber over modular-LWE
  - NewHope over ring-LWE

qTESLA vs Dilithium

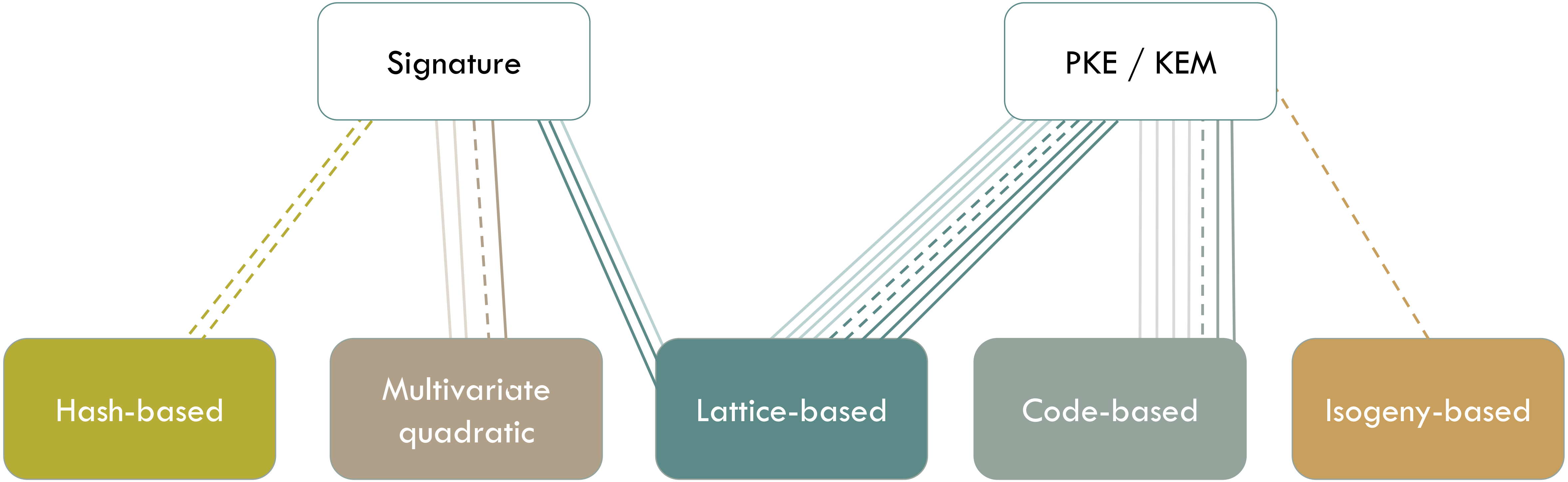
- Similar design (ring/modular-LWE)
- Dilithium < qTESLA

NIST candidates Round 2



With courtesy of Denis Butin and Johannes Buchmann

NIST candidates Round 3



With courtesy of Denis Butin and Johannes Buchmann

Gazing into the crystal ball – 2021/2022 Finalists

Code-based	Lattice-based	Multivariate
Classic McEliece	**One of** { CRYSTALS-KYBER NTRU SABER	
	One of { CRYSTALS-DILITHIUM FALCON	**Rainbow**

“ *NIST also sees* ***diversity of computational hardness assumptions*** *as an important long-term security goal for its standards. NIST hopes to standardize practically efficient* ***schemes from different families of cryptosystems*** *to reduce the risk that a single breakthrough in cryptanalysis will leave the world without a viable standard for either key-establishment or digital signatures.* ”

Computation Hardness Assumptions

Lattice-based

- **L**earning **W**ith **E**rrors
- Module LWE
- Module LWR
- Sort Integer Solution
- SelfTargetMSIS
- NTRU problem
- NTRU-SIS

Hash-based

- PQ-DM-SPR
- PQ-ITSR

Isogeny-based

- Supersingular Isogeny DH

Multivariate

- MQ
- MinRank
- IP

Code-based

- **Q**uasi-**c**yclic codeword finding
- QC syndrome decoding
- QC syndrome decoding with parity
- Goppa code distinguishing

With courtesy of Denis Butin and Johannes Buchmann

Learning with errors problem

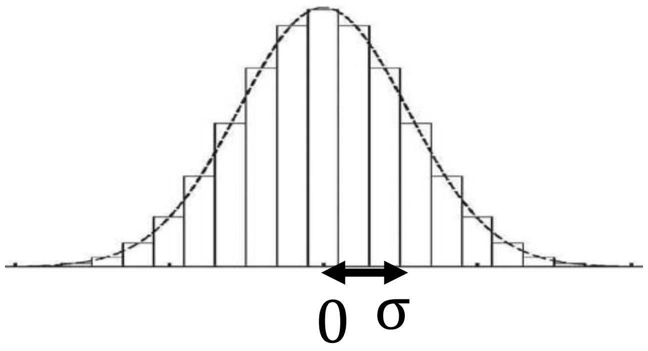
Given: (A,b) with

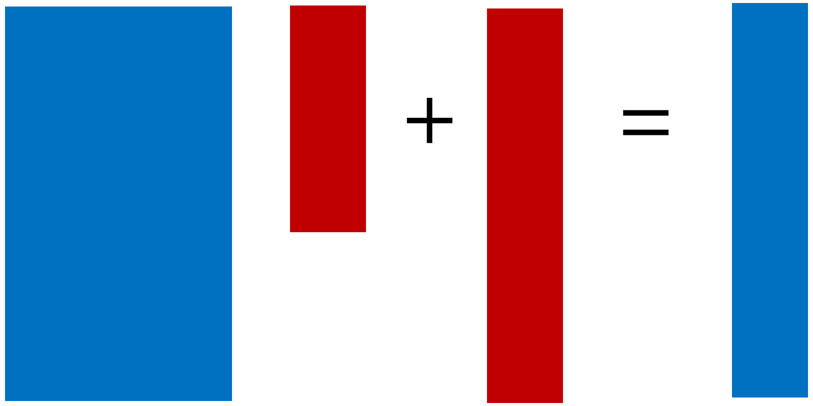
$A \leftarrow_{\$} \mathbb{Z}^{m\times n}$

$s \leftarrow_{\sigma} \mathbb{Z}^{n}$, $e \leftarrow_{\sigma} \mathbb{Z}^{n}$

$b = As + e \bmod q$

Find: s



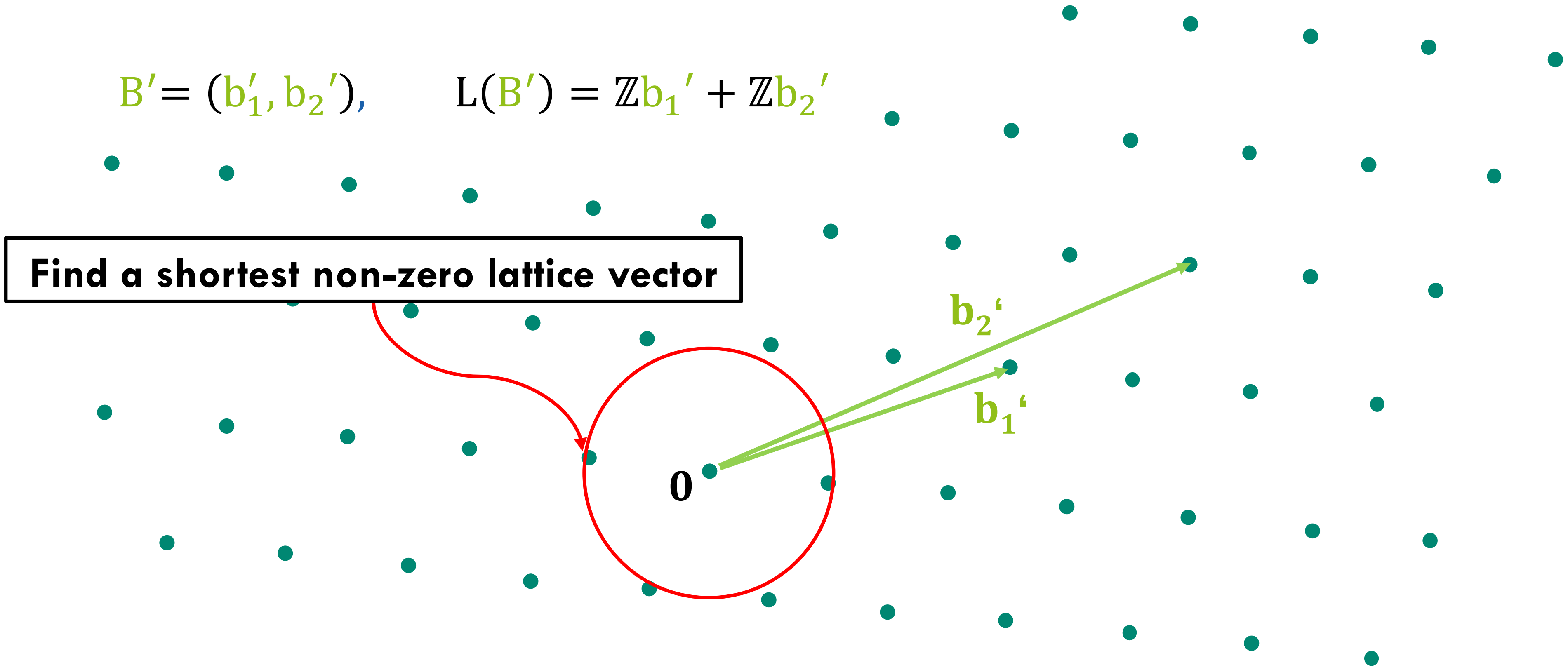


mod q

To solve LWE, solve SVP

Shortest Vector Problem (SVP)

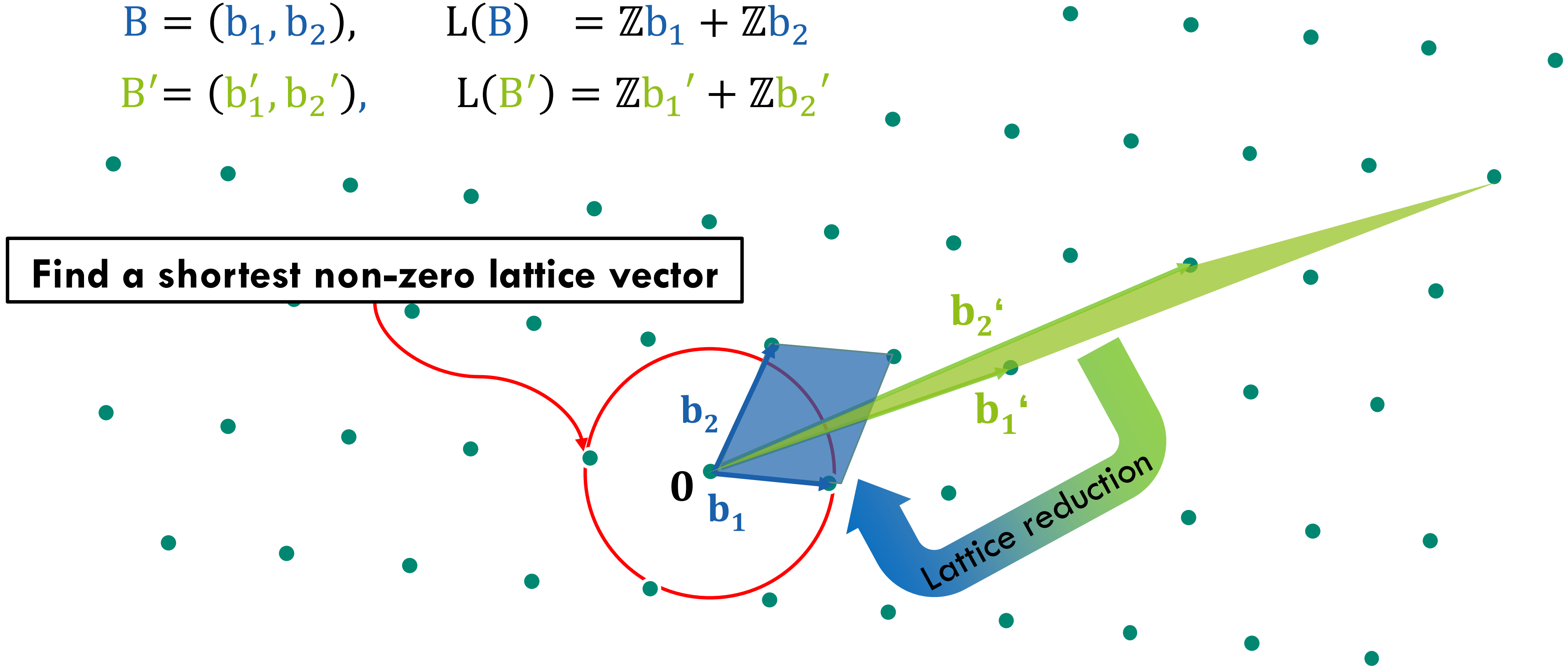
$B' = (b_1', b_2')$, $\quad L(B') = \mathbb{Z}b_1' + \mathbb{Z}b_2'$



Solving the SVP

$B = (b_1, b_2), \qquad L(B) = \mathbb{Z}b_1 + \mathbb{Z}b_2$

$B' = (b_1', b_2'), \qquad L(B') = \mathbb{Z}b_1' + \mathbb{Z}b_2'$



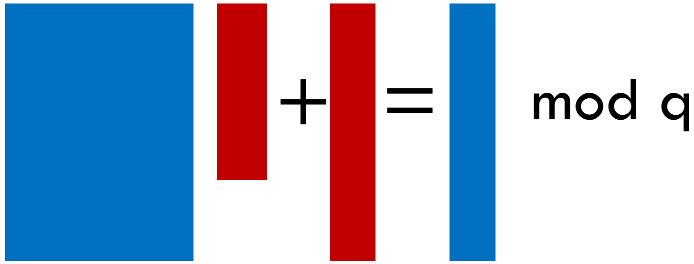
Lattice reduction – LLL Algorithm

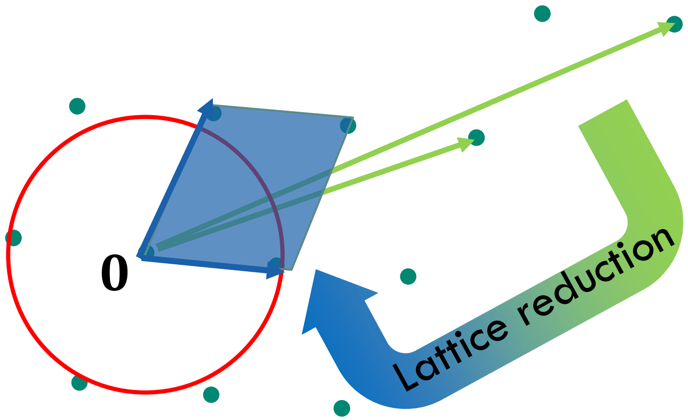
+ Polynomial runtime (in dimension)
- Basis quality (shortness/orthogonality) is poor

- Currently fastest lattice reduction used to break lattice problems:
  Block Korkine Zolotarev (BKZ) algorithm
- BKZ uses LLL as subroutine

Arjen Lenstra,
Hendrik Lenstra,
László Lovász

Solving LWE by solving SVP





Given $As + e = b \bmod q$

1. Construct

$$L = \left\{ v \in \mathbb{Z}^m \,|\, \exists\, x \in \mathbb{Z}^n : \begin{pmatrix} A & b \\ 0 & 1 \end{pmatrix} \cdot x = v \bmod q \right\}$$

$e \in L$:

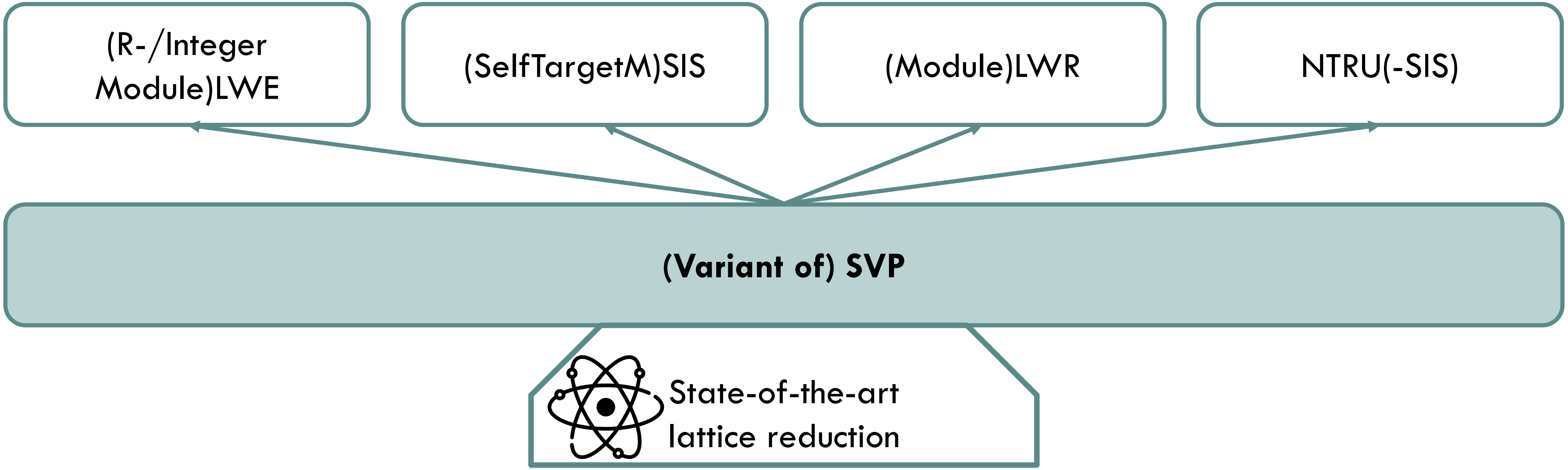
$$\begin{pmatrix} A & b \\ 0 & 1 \end{pmatrix} \begin{pmatrix} -s \\ 1 \end{pmatrix} = \begin{pmatrix} -As + b \\ 0 \cdot s + 1 \cdot 1 \end{pmatrix} = \begin{pmatrix} e \\ 1 \end{pmatrix} =: v$$

2. Solve SVP in L to find $\begin{pmatrix} e \\ 1 \end{pmatrix}$

3. Compute s from

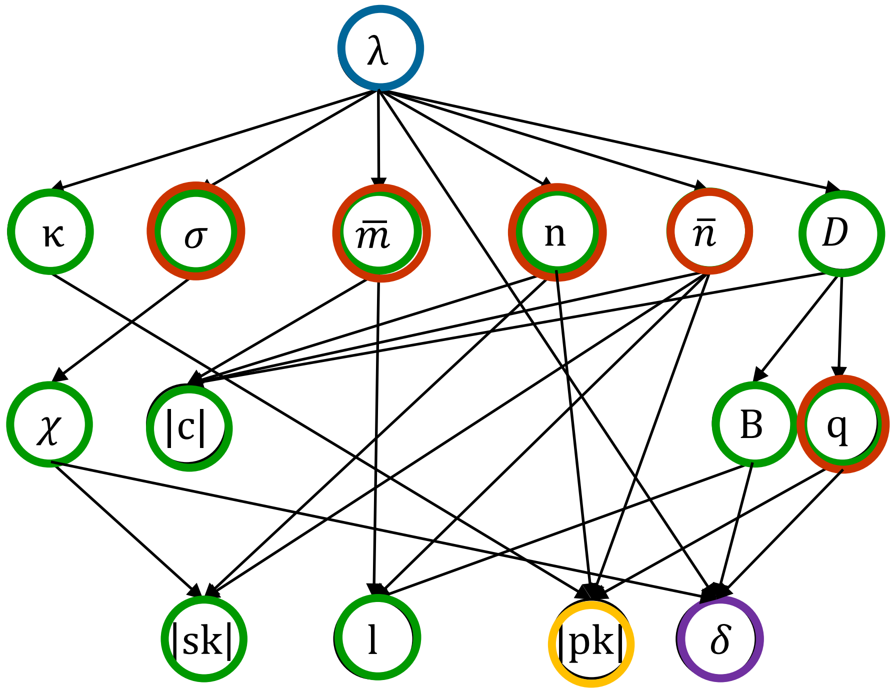
$b - e = As \bmod q$

Lattice-based problems



Bit hardness of problem = #ops to break instance by fastest algorithms

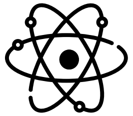
How to choose quantum secure parameters -- FrodoKEM



Choose targeted security level

Solve optimization problem

Small pk

LWE  hardness $\geq \lambda$?

Decryption failure $\delta \leq 2^{-\lambda}$?

Return parameters

Outline

NIST standardization effort

- 3rd candidates and Timeline
- Security estimations (of LWE)

Decryption failures of PKEs/KEMs

- Definition
- Attacks

Key generation

$$A \cdot S + E = B \mod q$$

Encryption

A B m

$$A \cdot S + E = B \mod q$$

$$A \cdot S' + E' = C \mod q$$

$$\underbrace{B \cdot S' + E''}_{\approx V} + \lfloor q/4 \rceil m = C' \mod q$$

Decryption

A B m

$$A \cdot S + E = B \mod q$$

$$A \cdot S' + E' = C \mod q$$

$$B \cdot S' + E'' + \lfloor q/4 \rceil m = C' \mod q$$

$$\underbrace{B \cdot S' + E''} \approx V$$

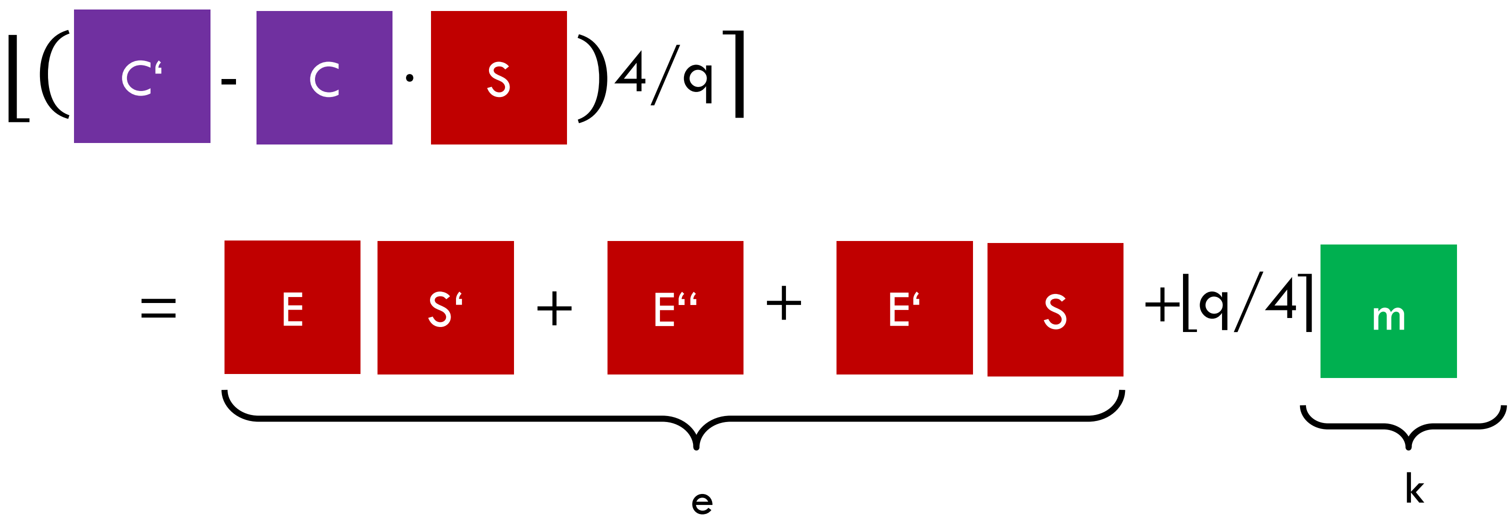
S C C'

$$\lfloor (C' - \underbrace{C \cdot S}) 4/q \rceil = m$$

Example statement:
Frodo NIST submission, Section 2.2.7

The next lemma states bounds on the size of errors that can be handled by the decoding algorithm.

Lemma 2.18. *Let* $q = 2^D$, $B \leq D$. *Then* $\mathrm{dc}(\mathrm{ec}(k) + e) = k$ *for any* $k, e \in \mathbb{Z}$ *such that* $0 \leq k < 2^B$ *and* $-q/2^{B+1} \leq e < q/2^{B+1}$.



P is **δ-correct** if

$$\Pr[\mathrm{Decrypt}(\mathrm{c}, \mathrm{sk}) \neq m : c \leftarrow Encrypt(m, pk), (pk, sk) \leftarrow Gen()] \leq \delta$$

Impact of decryption errors

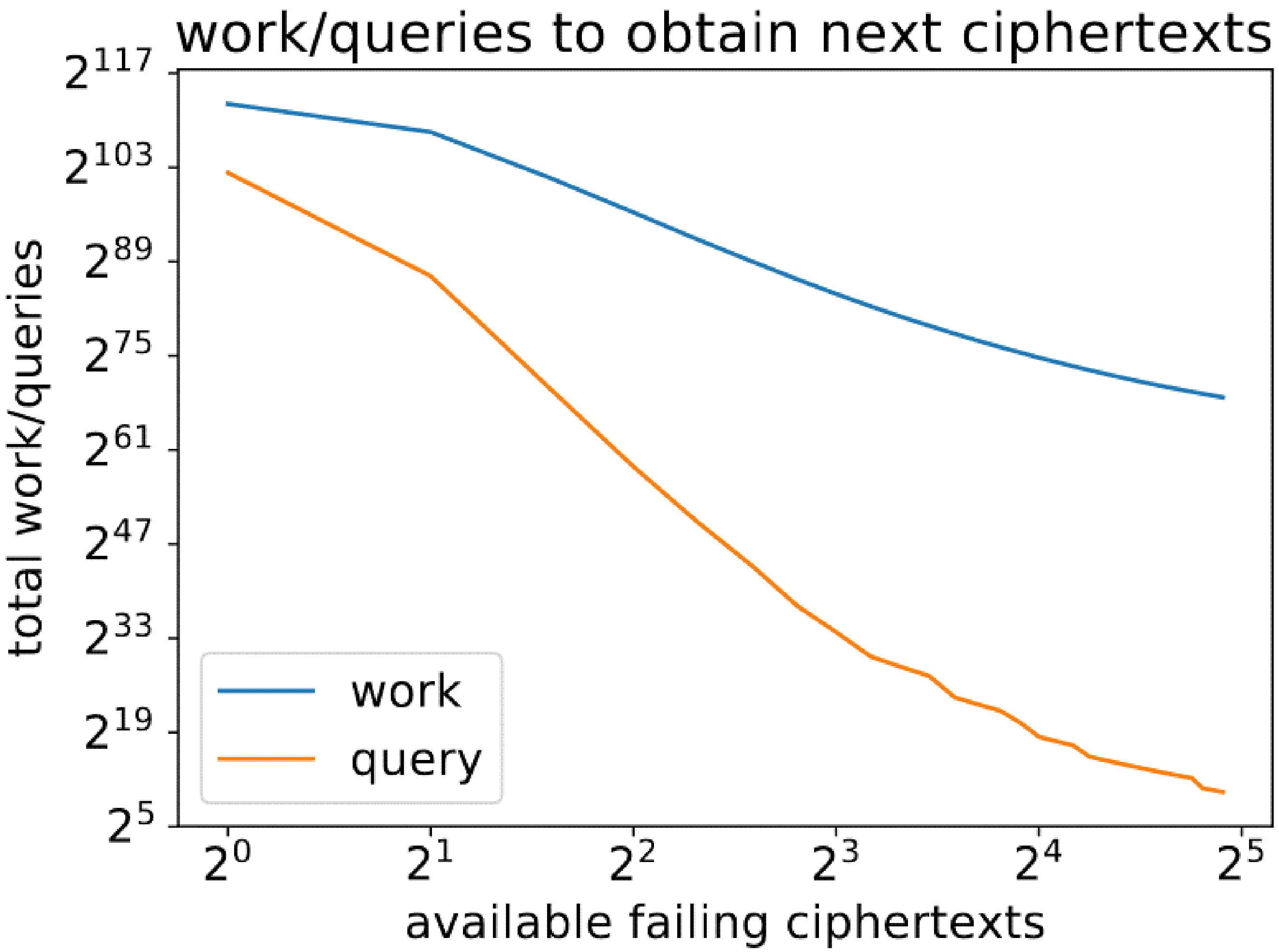
Every decryption error tells us…

E S‘ + E‘‘ + E‘ S $\geq q/2^{B+1}$

or

E S‘ + E‘‘ + E‘ S $< -q/2^{B+1}$

“One failure is not an option…“



J.P. D'Anvers, M. Rossi, F. Virdia: (One) failure is not an option: Bootstrapping the search for failures in lattice-based encryption schemes. EuroCrypt 2020, ePrint Archive, Report 2019/1399

Impact of decryption errors

Every decryption error tells us…

$$E\,S' + E'' + E'\,S \geq q/2^{B+1}$$

or

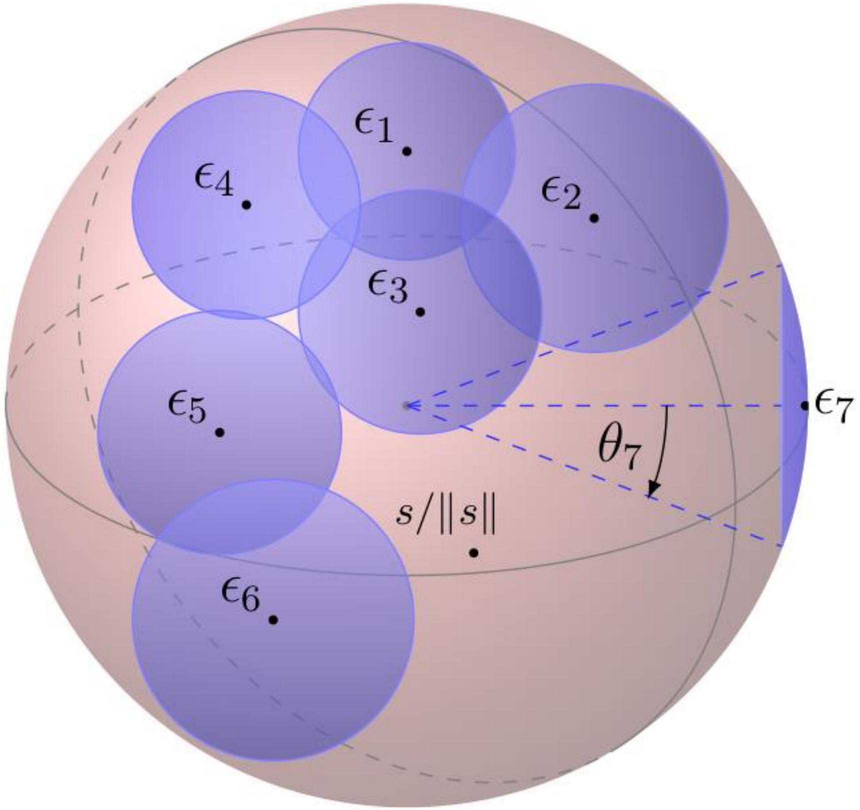
$$E\,S' + E'' + E'\,S < -q/2^{B+1}$$

Every successful decryption tells us…

$$-q/2^{B+1} \leq E\,S' + E'' + E'\,S < q/2^{B+1}$$

Even garther information from successful decryption.

Idea of our attack



Recall:

$\mathrm{sk} = \mathrm{s}, \mathrm{e}$

$\mathrm{C}_1 = \mathrm{s}'\mathrm{a} + e'\mathrm{mod}\ 16$

$\mathrm{C}_2 = \mathrm{v} + \mathrm{Encode}(\mathrm{m})$

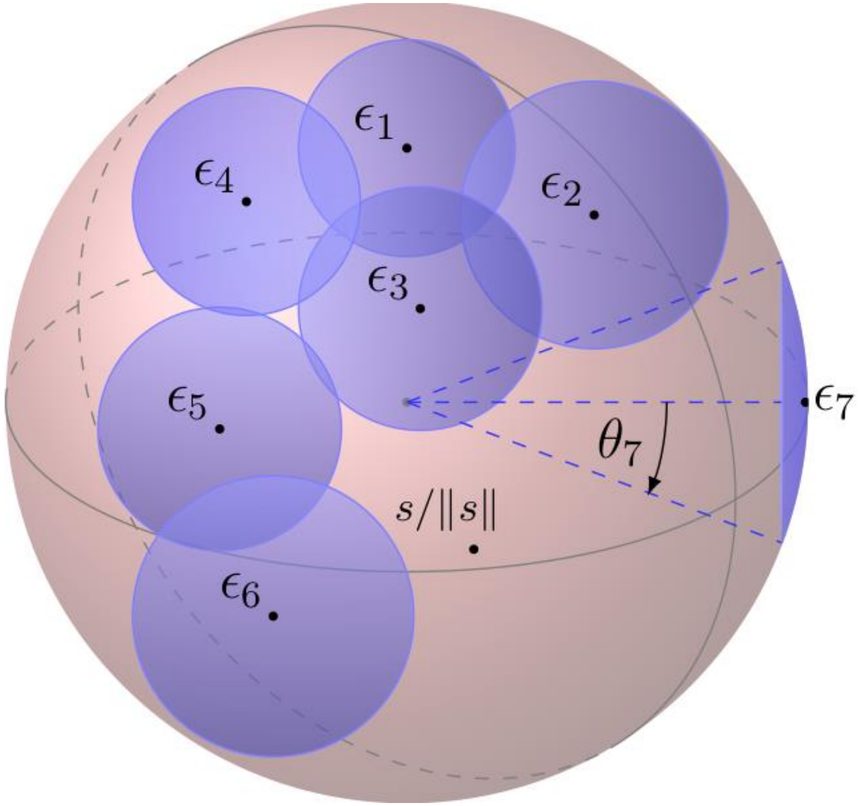
$\epsilon_i = \epsilon_i(s', e')$ randomness used in encryption queried to decryption oracle

Adversary learns from succesfull decryptions:

- s is not in blue region
- To trigger decryption error with higher probability, choose ϵ_8 in red region

N. Bindel, J.M. Schanck, Decryption failure is more likely after success, PQCrypto 2020, ePrint Archive, Report 2019/1392

Efficacy of a query set



$\mathrm{E} = \{\epsilon_1, \dots, \epsilon_7, \dots\}$

Efficacy of E = fraction of the sphere covered by caps

$= \frac{\text{blue area}}{\text{red area}}$

Intelligent adversary:

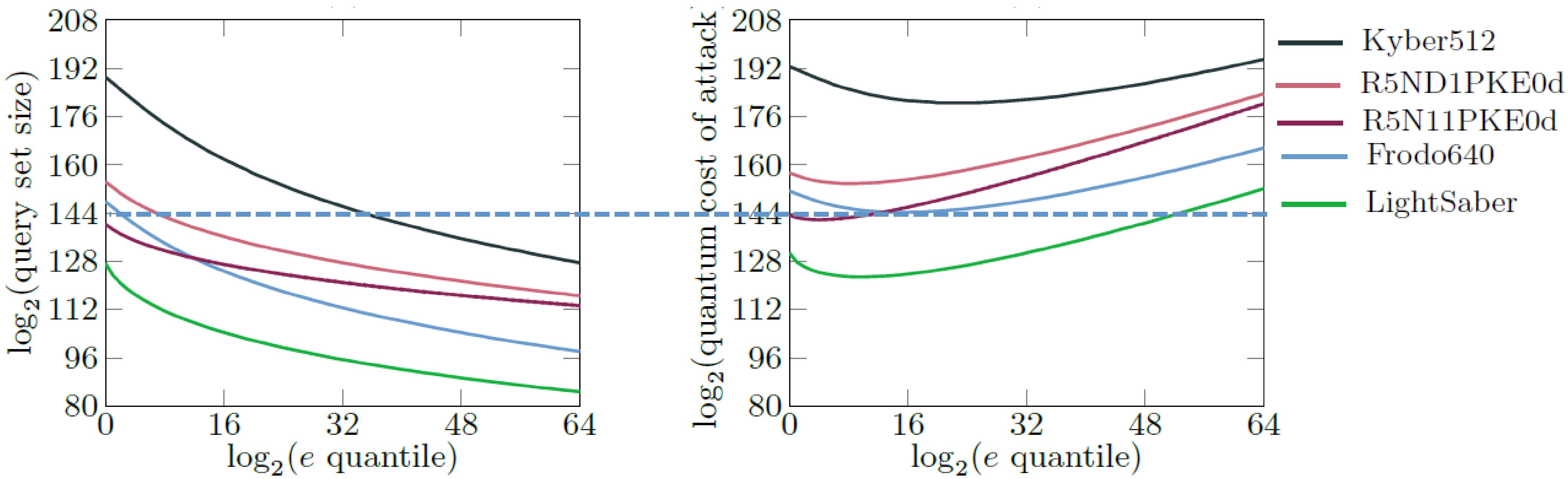
Efficacy ⇧ and #E ⇩

Cost of adversary:

- Cost of generation efficient query set
- Cost of asking queries: $\leq 2^{64}$ (NIST CfS)

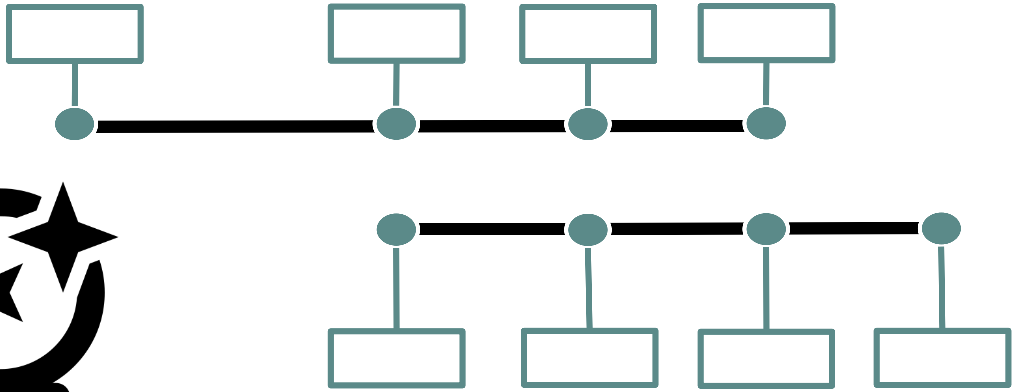
Experimental results

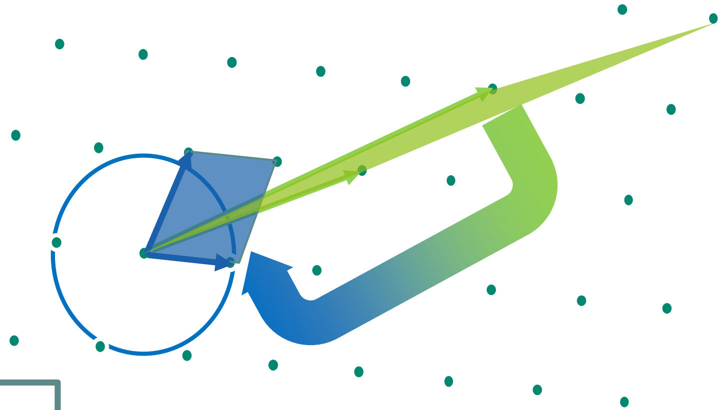
Predicted size of a query set of unit efficacy and quantum cost of producing such a query set

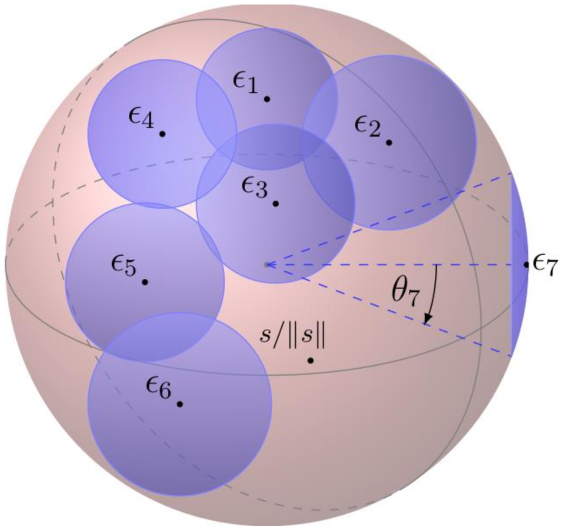


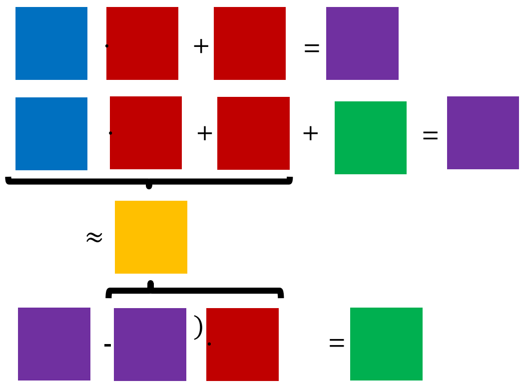
Summary

NIST









UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

THANKS