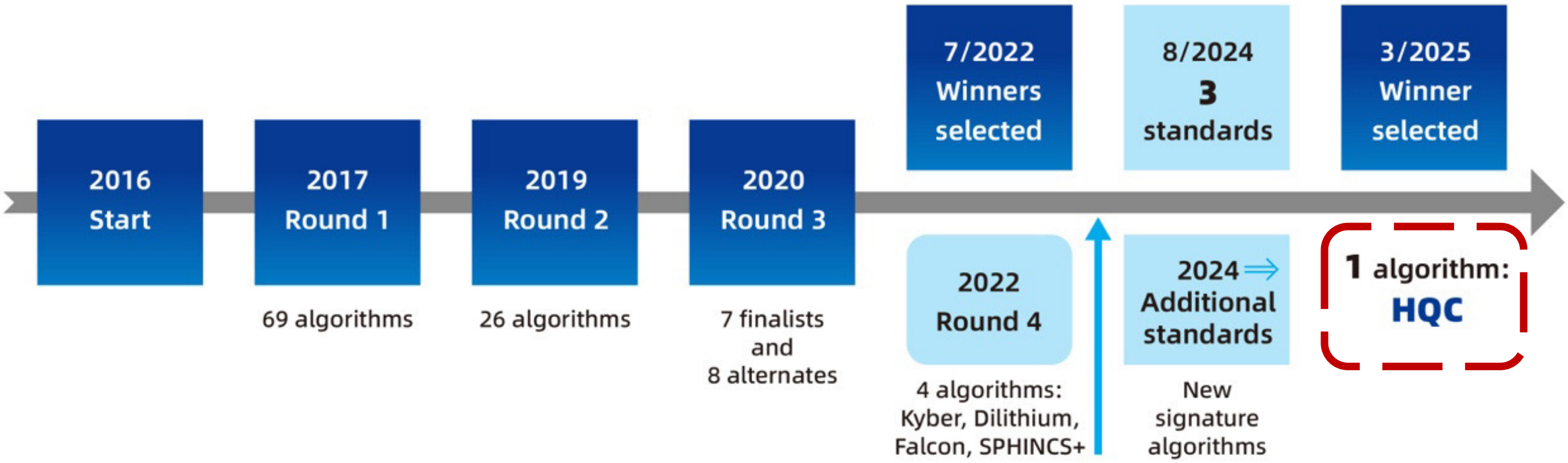
코드기반암호 HQC에 대한 양자 공격 비용 분석

한성대학교 장경배

NIST PQC 공모전 Round 4

- Round 4: ~~SIKE~~를 제외한 코드기반암호 3종 후보 Classic McEliece, BIKE, HQC
- 2025년 3월: **HQC가 최종 알고리즘**으로 선정
  - Kyber를 대체하는 것이 아닌 보완적 솔루션으로 활용 예정



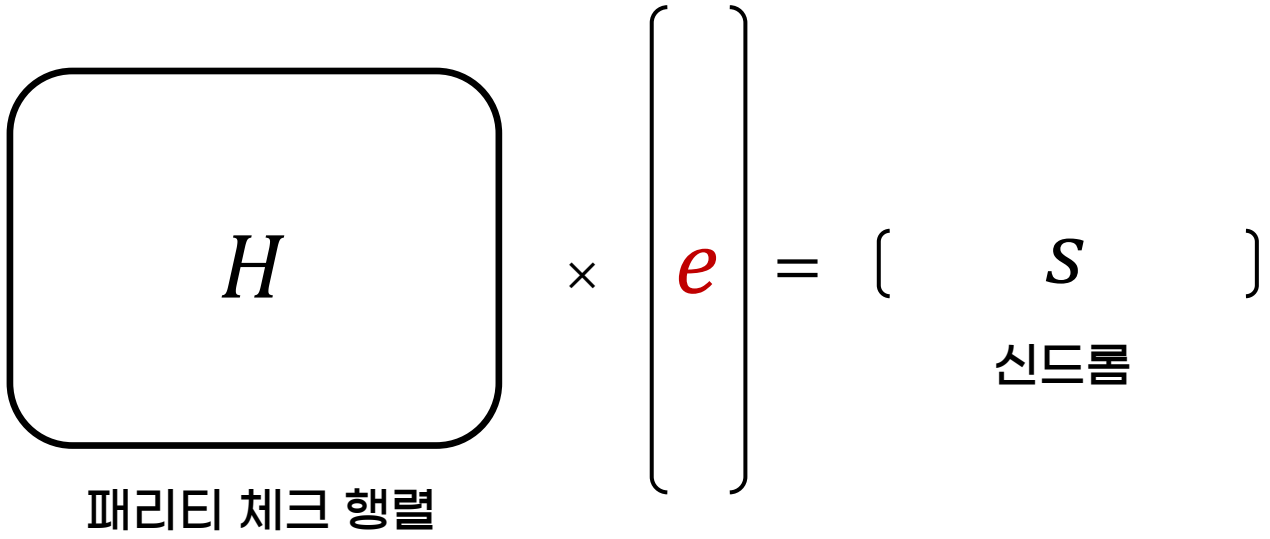
HQC가 선정된 이유

- IND-CCA2 보안을 달성하기 위해선 **충분히 낮은 복호화 실패율 (DFR)**을 보장해야함
  - HQC가 이 측면에서 BIKE 보다 더 우수하며, BIKE의 DFR 분석은 여전히 미해결 문제

- Classic McEliece는 키 크기로 인한 **실제 배포의 실용성 문제**가 존재
  - 하지만 ISO 표준화 검토 중
  - 이중 표준화의 호환성 문제가 존재, 따라서 ISO 표준화가 된다면 NIST는 채택을 고려 중

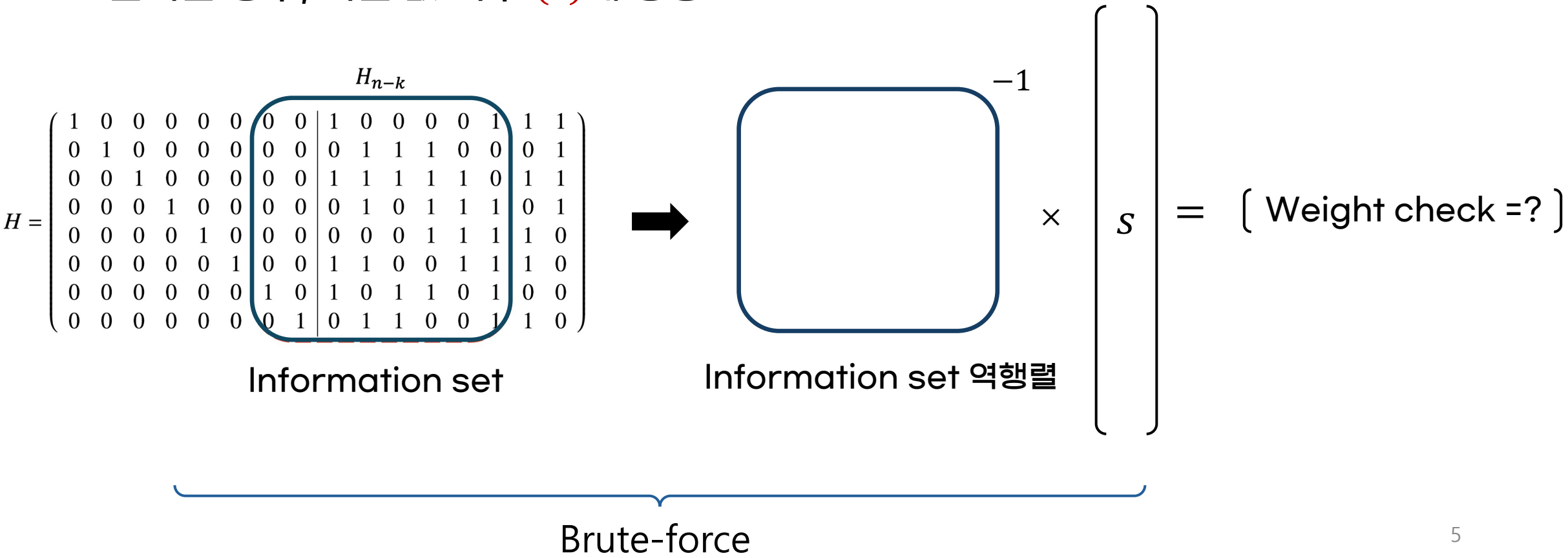
HQC (Hamming Quasi-Cyclic)

- **안전성 가정:** 순환 구조를 가지는 Quasi-Cyclic 부호 디코딩 문제와 랜덤 선형 부호 디코딩 문제의 난이도가 비슷할 것
  - (H, s)가 주어졌을 때 $He^T = s^T$ 라는 신드롬 계산 식, 낮은 해밍 무게를 만족하는 벡터 e 를 찾아내기 매우 어려움



분석 알고리즘: Information Set Decoding (ISD)

- ISD 알고리즘은 코드기반암호에 대한 가장 강력한 공격 기법
  - 공개키 내 행렬 H_{n-k} (Information set, ▢)을 선택해가며 전수 조사
    - Information set의 역행렬 × 신드롬 (암호문, s)의 결과 벡터가 조건 해밍 무게를 만족할 경우, 비밀 값 복구 (e)에 성공



Syndrome Decoding 챌린지

Syndrome Decoding in the Quasi-cyclic Setting

This page is dedicated to the syndrome decoding problem for random quasi-cyclic binary linear codes, in the range of parameters similar to the BIKE cryptosystem.

Quasi-cyclic Syndrome Decoding problem. Given integers n, k, w such that $k \leq n$ and $w \leq n$, an instance of the problem QCSD(n, k, w) consists of a quasi-cyclic parity-check matrix $\mathbf{H} \in \mathbb{F}_2^{(n-k)\times n}$ and a vector $\mathbf{s} \in \mathbb{F}_2^{n-k}$ (called the *syndrome*). A solution to the problem is a vector $\mathbf{e} \in \mathbb{F}_2^n$ of Hamming weight $\leq w$ such that $\mathbf{He}^\top = \mathbf{s}^\top$.

The challenge. Here, we focus on instances with code rate $R = 0.5$, that is $n = 2k$. First, an even weight w is set. Then, the length $n = w^2 + 2$ and the dimension $k = n/2$ are derived, so that $w \simeq \sqrt{n}$. Notice that the block length k is an odd integer, preventing the following attack. With these choices of parameters, we expect the problem to be as close as possible to the problem (2,1)-QCSD on which rely the BIKE, HQC and LEDAcrypt cryptosystems proposed for the NIST standardization process.

Under these conditions, instances with cryptographic size are assumed to be out of reach, so we propose **instances with increasing size** to see how hard this problem is in practice.

Submit your solution

Hall of fame

Download instances

Format of instances

A list of instances (indexed by error weight w) generated by:

0 - Inria Paris

Tooltips give an indication of

< https://decodingchallenge.org/q-c >

Syndrome Decoding 챌린지

Best solutions

Length	Weight	Authors	Details
3846	62	Shintaro Narisada, Hiroki Okada, Shusaku Uemura, Yusuke Aikawa, Kazuhide Fukushima and Shinsaku Kiyomoto	See details
3602	60	Shintaro Narisada, Hiroki Okada, Shusaku Uemura, Yusuke Aikawa, Kazuhide Fukushima and Shinsaku Kiyomoto	See details
3366	58	Shintaro Narisada, Hiroki Okada, Shusaku Uemura, Yusuke Aikawa, Kazuhide Fukushima, and Shinsaku Kiyomoto	See details

< 베스트 솔루션 >

Instance	Security	n_1	n_2	n	k	ω	$\omega_r = \omega_e$	DFR
HQC-1	NIST-1	46	384	17 669	128	66	75	$< 2^{-128}$
HQC-3	NIST-3	56	640	35 851	192	100	114	$< 2^{-192}$
HQC-5	NIST-5	90	640	57 637	256	131	149	$< 2^{-256}$

< HQC 파라미터 >

Quantum Information Set Decoding (QISD)

- ISD 알고리즘의 양자 구현
  - Grover 알고리즘 + 기본 ISD 알고리즘 (Prange)*

- 다양한 개선된 ISD 알고리즘들이 존재
  - 기본 ISD에서 검색 범위 및 해밍 무게 조건을 변형하는 방식

- **최신 ISD의 양자화?**
  - 변형 버전들은 더 섬세한 조건 및 연산을 요구함
    - 즉, 검사의 반복 수를 줄일 순 있지만, 검사 한 번에 대한 **양자 구현 비용이 높아짐**

> The improvement introduced by Stern in [23] increases the number of roots by a larger factor. However, it also increases the cost of each iteration by more than the square root of the same factor. I do not see how Stern's collision-searching idea can save time in quantum information-set decoding.

< Bernstein, Daniel J. "Grover vs. McEliece", International Workshop on Post-Quantum Cryptography, 2010. >

[*] Eugene Prange. 1962. The use of information sets in decoding cyclic codes. IRE Trans. Inf. Theory 8, 5 (1962)

Overview: 대칭키 암호 vs 코드기반 암호

- 대칭키 암호와 코드기반암호에 대한 **양자 분석 비교** (Grover 알고리즘)

	대칭키 암호 키 복구	vs	코드기반암호 양자 ISD
Input	알려진 평문 P		패리티 체크 행렬 $H_{(n-k)\times n}$
Grover Oracle	암호화 알고리즘: $Enc_k(P)$ (e.g., AES, ARIA) ✓ $Enc_k(P) = C?$		Information set 추출: H_{n-k} Information set 역행렬 계산: ${H_{n-k}}^{-1}$ ✓ Weight $\left({H_{n-k}}^{-1}s\right) = t?$
Diffusion Operator	비밀키 K		Information set H_{n-k} 추출에 사용 된 벡터

Overview: 대칭키 암호 vs 코드기반 암호

- 코드기반암호의 경우 **Input 단계부터 많은 수의 큐빗**이 필요
  - HQC: 패리티 체크 행렬 $H \in \mathbb{F}_2^{(n-k)\times n}$ (HQC-1: $n = 17699,\ k = n/2$)
  - AES: 128 큐빗 평문

	대칭키 암호 키 복구	vs	코드기반암호 양자 ISD
Input	알려진 평문 P		패리티 체크 행렬 H
Grover Oracle	암호화 알고리즘: $Enc_k(P)$ (e.g., AES, ARIA) ✓ $Enc_k(P) = C$?		Information set 추출: H_{n-k} Information set 역행렬 계산: ${H_{n-k}}^{-1}$ ✓ Weight $\left({H_{n-k}}^{-1}s\right) = t$?
Diffusion Operator	비밀키 K		Information set H_{n-k} 추출에 사용 된 벡터

Overview: 대칭키 암호 vs 코드기반 암호

- Oracle에 들어가는 **로직이 더욱 복잡**하며, **대규모 행렬**을 대상으로 함
  - HQC: Information set 추출, 역행렬 계산 (가우스 소거), 해밍 무게 계산 (덧셈 트리)
  - AES: 암호화 연산 (AddRoundkey, SubBytes, MixColumns, KeySchedule)

	대칭키 암호 키 복구	vs	코드기반암호 양자 ISD
Input	알려진 평문 P		패리티 체크 행렬 H
Grover Oracle	암호화 알고리즘: $Enc_K(P)$ (e.g., AES, ARIA) ✓ $Enc_K(P) = C$?		Information set 추출: H_{n-k} Information set 역행렬 계산: ${H_{n-k}}^{-1}$ ✓ Weight $\left({H_{n-k}}^{-1}s\right) = t$?
Diffusion Operator	비밀키 K		Information set H_{n-k} 추출에 사용 된 벡터

Overview: 대칭키 암호 vs 코드기반 암호

- Diffusion Operator의 **Multi-Controlled 게이트**
  - HQC: n-qubit 벡터를 대상으로 동작 (HQC-1: $n = 17699$)
  - AES: 비밀키 K를 대상으로 동작

	대칭키 암호 키 복구	vs	코드기반암호 양자 ISD
Input	알려진 평문 P		패리티 체크 행렬 H
Grover Oracle	암호화 알고리즘: $Enc_K(P)$ (e.g., AES, ARIA) ✓ $Enc_K(P) = C$?		Information set 추출: H_{n-k} Information set 역행렬 계산: ${H_{n-k}}^{-1}$ ✓ Weight $\left({H_{n-k}}^{-1}s\right) = t$?
Diffusion Operator	비밀키 K		Information set H_{n-k} 추출에 사용 된 벡터

코드기반암호에 대한 양자 ISD 회로 구현

Input Setting (1/2)

- 큐빗 패리티 체크 행렬 $H_{(n-k)\times n}$로부터 Information set H_{n-k} 을 준비
  - **Dicke state 기법**[*]: n개의 큐빗 중 $(n-k)$개의 큐빗만을 1로 변경 → $|D_{n-k}^{n}\rangle$
    - 큐빗이 1인 열 만을 선택하여 Information set을 추출

$|D_8^{16}\rangle$: 0 0 0 0 0 0 1 1 1 1 1 1 1 1 0 0

H_{n-k}

$$H = \left(\begin{array}{cccccc|cc|cccccc} 1&0&0&0&0&0&0&0&1&0&0&0&0&1&1&1\\ 0&1&0&0&0&0&0&0&0&1&1&1&0&0&0&1\\ 0&0&1&0&0&0&0&0&1&1&1&1&1&0&1&1\\ 0&0&0&1&0&0&0&0&0&1&0&1&1&1&0&1\\ 0&0&0&0&1&0&0&0&0&0&0&1&1&1&1&0\\ 0&0&0&0&0&1&0&0&1&1&0&0&1&1&1&0\\ 0&0&0&0&0&0&1&0&1&0&1&1&0&1&0&0\\ 0&0&0&0&0&0&0&1&0&1&1&0&0&1&1&0 \end{array}\right)$$

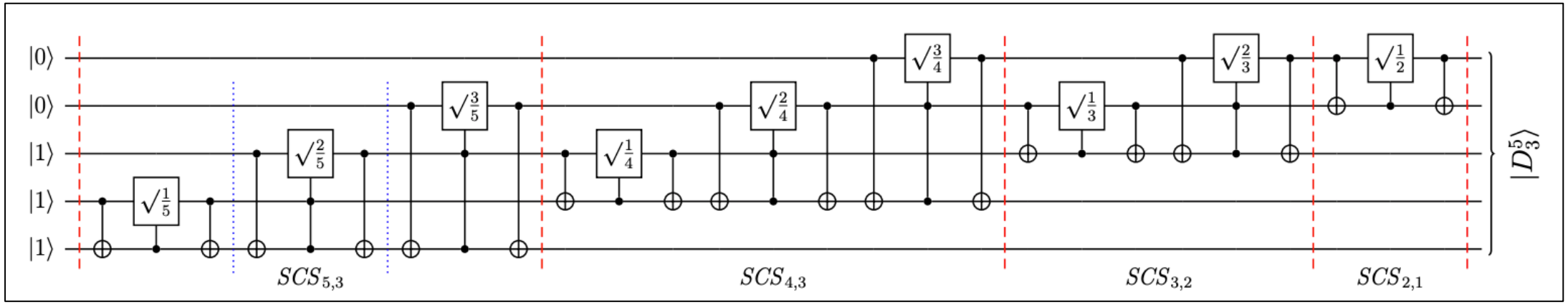
Information set

[*] C. Mukherjee et al. “Preparing Dicke States on a Quantum Computer”, IEEE Transactions on Quantum Engineering, 2020.

Input Setting (1/2)

- Dicke State 구현 비용 및 양자 회로

State	X	Qubits	CNOT	Ry	Depth
$\|D_k^n\rangle$	k	n	$5nr - 5r^2 - 2n$	$4nr - 4r^2 - 2n + 1$	$\frac{27nk - 12k - 27k^2 + 3}{k - 2}$

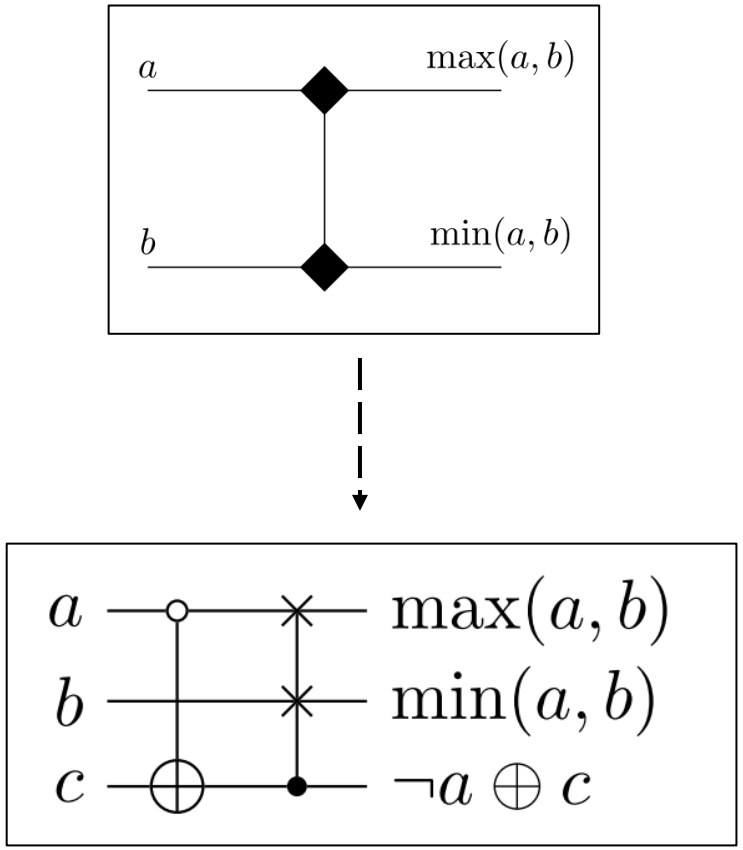


< $|D_3^5\rangle$ 양자 회로 >

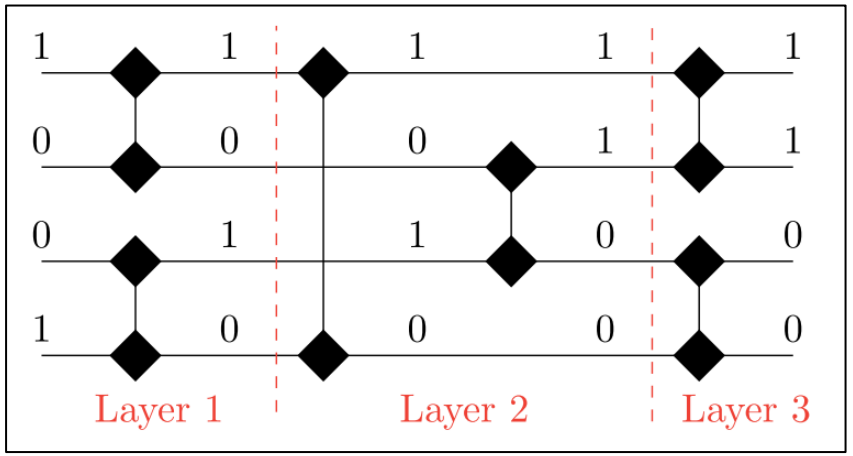
[*] C. Mukherjee et al. “Preparing Dicke States on a Quantum Computer”, IEEE Transactions on Quantum Engineering, 2020.

Input Setting (2/2)

- 양자 분류 네트워크: Dicke State로부터 값이 1인 열 만을 모아 **Information set을 추출**
  - 1은 좌측, 0은 우측으로 구분 → 좌측의 행렬을 Information set으로 사용



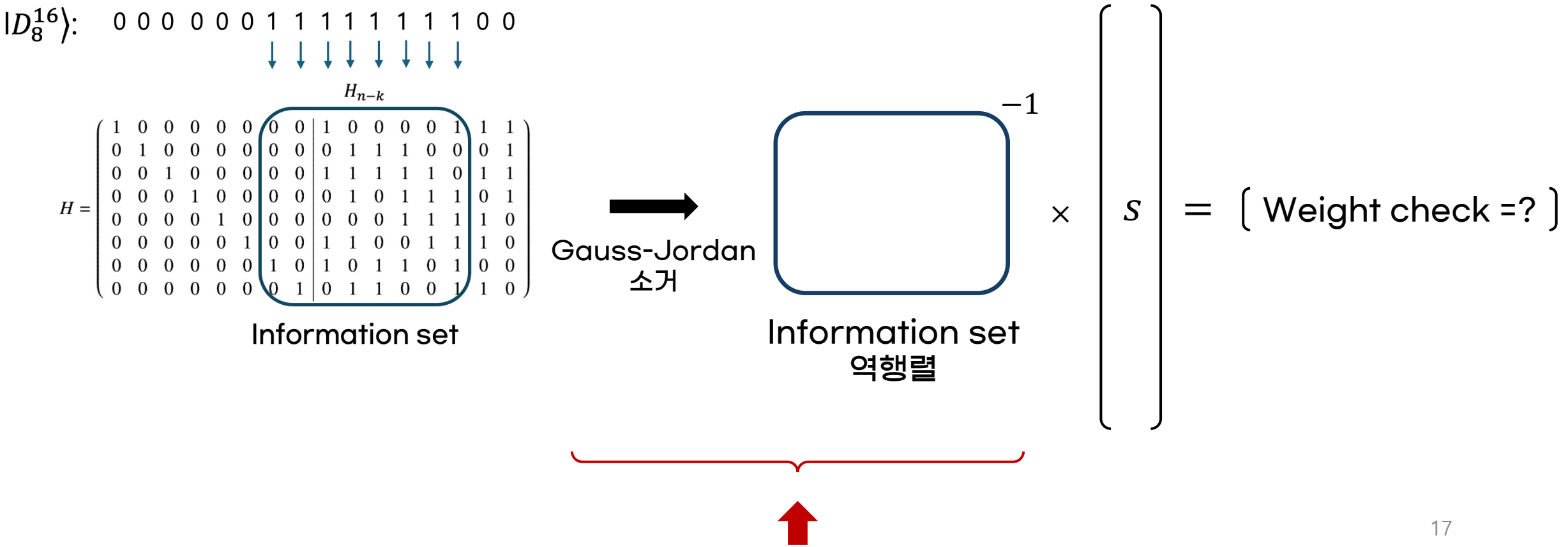
< 분류기 >



< 분류 네트워크 >

Oracle: Gauss-Jordan Elimination

- 추출한 Information set ( )에 **Gauss-Jordan 소거**를 통해 Inverse 행렬 계산
  - **가장 많은 양자 비용**이 드는 연산이며, **양자 ISD 최적화 핵심**



Oracle: Gauss-Jordan Elimination

- Gauss-Jordan 소거 목표: Information set 행렬을 Identity 행렬로 만듦
  - 해당 연산을 신드롬 s에도 동일하게 수행한다면, Information set의 역행렬을 신드롬에 곱한 값을 동일하게 얻을 수 있음 → $(H_{n-k})^{-1}s$

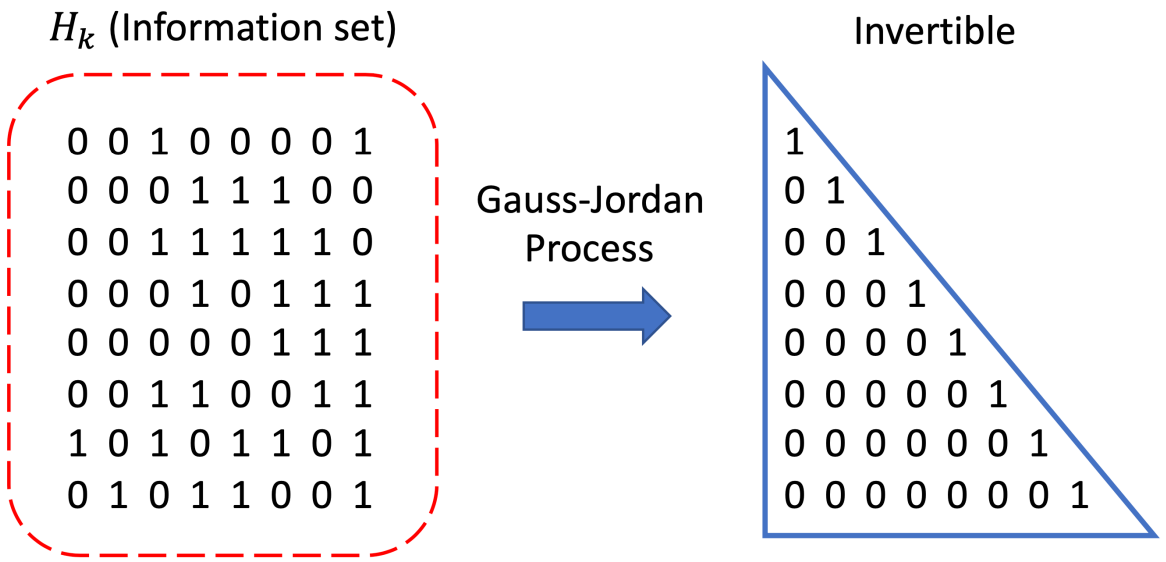
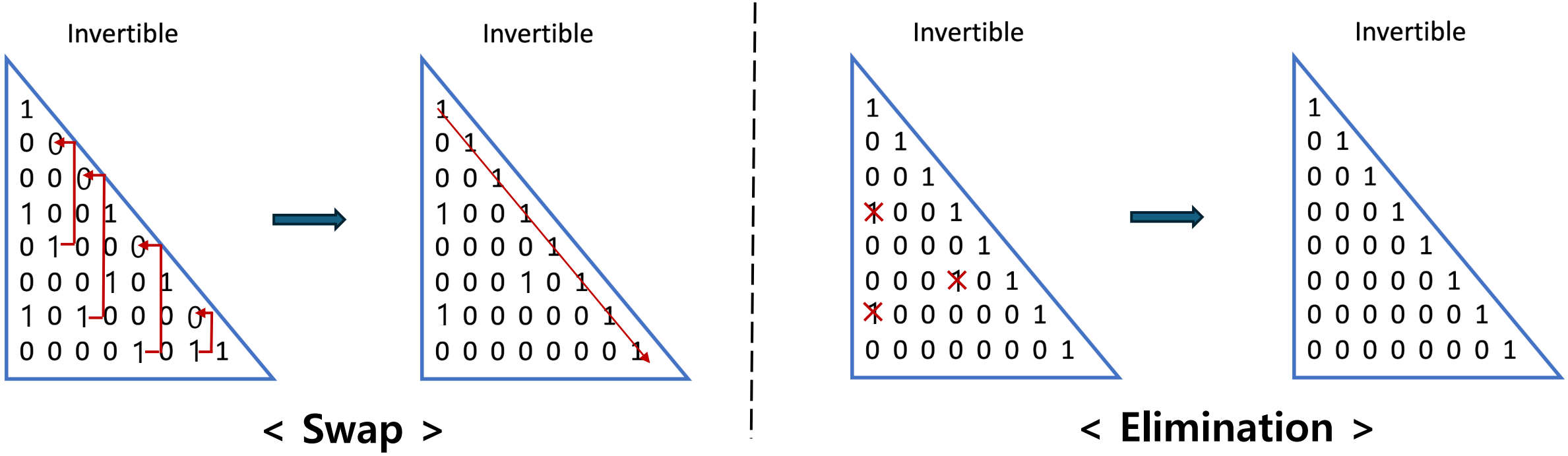


Fig. 4. Primary goal of Gauss-Jordan elimination.

Oracle: Gauss-Jordan Elimination

- 크게 2 가지 단계로 구분 됨 → **Swap 과 Elimination**
  - Classical 구현과 달리 (branch), 양자 구현 시 (branch X) 고려 사항이 다수 존재함



Oracle: Gauss-Jordan Elimination

- Gaussian-Jordan 소거를 대상 행렬 (Information set)의 첫번째 Pivot부터 마지막 Pivot까지 수행

Algorithm 2: Quantum Implementation of Gauss-Jordan Elimination

Input: A matrix H of size $n \times n$, a vector c of size n (the n-th column of H)
Output: Updated vector c

1: **for** $i = 0$ to $(n-2)$ **do**
  //Swap stage
2: **for** $j = 0$ to $(n-2-i)$ **do**
3: *Copy* $H_{i,i} \oplus 1$ to ancillas using exponential copy //Size of ancillas is $(n+1-i)$
4: Toffoli (ancillas, $(i+1+j)$-th row of H, i-th row of H)
5: *Initialize* ancillas using reuse technique //$(n-i)$ ancillas can be initialized
  //Elimination stage
6: **for** $j = 0$ to $(n-1-i)$ **do**
7: *Copy* i-th column (except for $H_{i,i}$) of H to ancillas$_0$ using exponential copy
8: **for** $j = 0$ to $(n-2)$ **do**
9: *Copy* i-th row (except for $H_{i,i}$) of H to ancillas$_1$ using exponential copy
10: **for** $j = 0$ to $(n-2-i)$ **do**
11: Toffoli (i-th column of H, i-th row of H, $H_{(i+j+1),(i+j+1)}$)
12: Toffoli (ancillas$_0$ + i-th column of H, ancillas$_1$ + i-th row of H, $H_{((i+1)\sim n),0\sim((n-1)\neq i)}$)
13: *Initialize* ancillas using reuse technique // ancillas$_{0,1}$ are initialized
14: **return** the n-th column of H (i.e., c)

Swap

Elimination

[*] Jang et al. "Depth Optimized Quantum Circuit of Gauss-Jordan Elimination", Applied sciences, 2024.

Oracle: Gauss-Jordan Elimination

- Gauss-Jordan elimination 양자 구현 비용

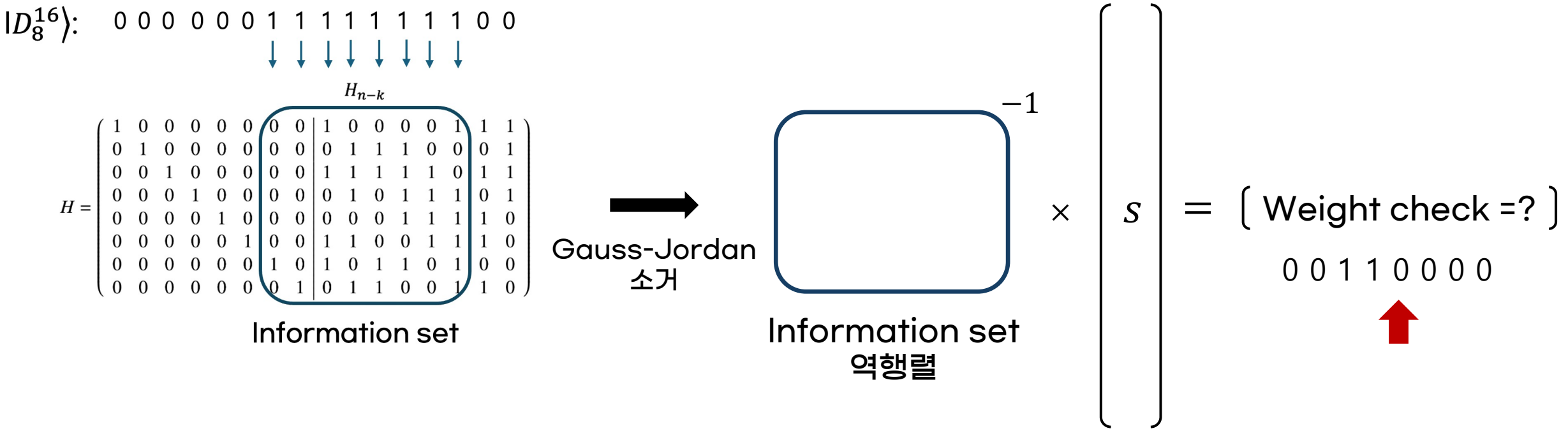
Matrix size	Qubits	# T	T-depth
n	$\frac{7n^2 - 11n + 6}{2}$	$\frac{7n(5n^2 + 3n - 8)}{6}$	$2n^2 + 2n - 4$

Matrix size	Qubits	# CNOT	# 1qCliff	# T	T-depth	Full depth
8 x 8	183	3861	492	3,136	140	112
16 x 16	811	31329	3,352	24,640	540	407
32 x 32	3,411	251321	24,368	194,432	2,108	1,880
48 x 48	7,803	848401	79,432	652,736	4,700	12,260

[*] Jang et al. "Depth Optimized Quantum Circuit of Gauss-Jordan Elimination", Applied sciences, 2024.

Oracle: 해밍 무게 확인

- 마지막으로 Gauss-Jordan 소거 후 결과인 $(H_{n-k})^{-1}s$ 값의 **해밍 무게**가 특정 무게 조건 t를 만족하는지 확인



Oracle: 해밍 무게 확인

- 해밍 무게 확인에는 **양자 덧셈기가 트리 형식으로 사용 됨**, 길이 n-qubit인 벡터에 대해
  - $(n-1)$ 번의 양자 덧셈
  - $\log_2(n)$의 레이어 뎁스

$$(H_{n-k})^{-1}s = \left[\begin{matrix}1 & 1 & 1 & 0 & 1 & 1\end{matrix}\right]\begin{matrix}1 & 1\end{matrix}$$

1-qubit 덧셈

10 01 10 10

2-qubit 덧셈

11 100

3-qubit 덧셈

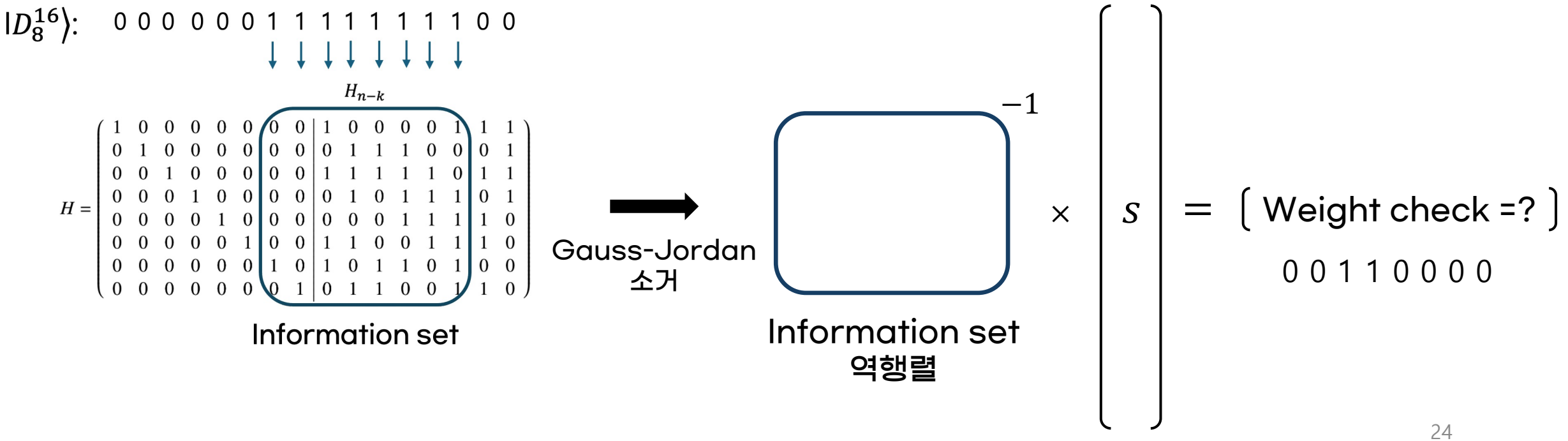
111 (weight = 7)

- 최종적으로, 해밍 무게를 만족한다면, **Oracle은 솔루션 큐빗을 체크**

Diffusion Operator & Post-processing (Classical)

- Diffusion operator는 Dicke State를 대상으로 동작
  - 최종 반환되는 Dicke State는 공격 성공에 유효한 Information set을 알 수 있음

공격 성공 조건: 선택한 Information set의 역행렬 × 신드롬 (암호문, s)의 결과 벡터가 해밍 무게를 만족할 경우, 비밀 값 e 복구에 성공 → $He^T = s^T$



Summary

- 양자 ISD 양자 회로 비용 정리

QISD	Module	Qubits	# T	T-depth
Input Setting	Public Key: $H_{(n-k)\times n}$	$n \times (n/2)$	–	–
	Dicke State: $\|D_k^n\rangle$	n	$149(n-1)^2$	$\frac{149(3n^2-48n+12)}{2(n-4)}$
Oracle	Information set H_{n-k}	–	$7(n-1)\log_2(n)(\log_2(n)-1)$	$2\log_2(n)(\log_2(n)+1)$
	Gauss-Jordan Elimination: $(H_{n-k})^{-1}s$	$\frac{7n^2-11n+6}{2}$	$\frac{7n(5n^2+3n-8)}{6}$	$2n^2+2n-4$
	Weight $(H_{n-k})^{-1}s$	–	$4\lceil\log_2(n/2)\rceil^2$	$10.5n-7\lceil\log_2(n/2)\rceil^2$
Diffusion Operator	Amplify Dicke state $\|D_k^n\rangle$	–	n –controlled gate	

$\#\ k = n/2$

HQC에 대한 양자 암호 분석

HQC 양자 ISD 공격 비용 추정

- HQC의 3가지 파라미터에 대해 **Grover 알고리즘**을 사용한 **양자 ISD 공격 비용 추정**
  - HQC-1: $n = 17669$, Grover 반복 횟수: 2^{59}
  - HQC-3: $n = 35851$, Grover 반복 횟수: 2^{93}
  - HQC-5: $n = 57637$, Grover 반복 횟수: 2^{123}

Instance	Security	n_1	n_2	n	k	ω	$\omega_r = \omega_e$	DFR
HQC-1	NIST-1	46	384	17 669	128	66	75	$< 2^{-128}$
HQC-3	NIST-3	56	640	35 851	192	100	114	$< 2^{-192}$
HQC-5	NIST-5	90	640	57 637	256	131	149	$< 2^{-256}$

< HQC 파라미터 >

HQC 양자 ISD 공격 비용 추정

- HQC에 대한 **양자 ISD 회로 비용**

HQC	Information set H_{n-k}		Gauss-Jordan Elimination: $(H_{n-k})^{-1}s$		Weight $(H_{n-k})^{-1}s$	
	T gates	T-depth	T gates	T-depth	T gates	T-depth
-1	2^{37}	2^{26}	2^{44}	2^{29}	2^{18}	2^{9}
-3	2^{39}	2^{27}	2^{47}	2^{31}	2^{19}	2^{10}
-5	2^{40}	2^{27}	2^{49}	2^{32}	2^{20}	2^{10}

- HQC에 대한 **Grover 양자 ISD 공격 비용** (+ AES와 비교*)

HQC	Qubits	T gates	T-depth
-1	2^{30}	2^{104}	2^{89}
-3	2^{32}	2^{141}	2^{125}
-5	2^{34}	2^{173}	2^{156}

>

AES	Qubits	T gates	T-depth
-128	2^{11}	2^{81}	2^{71}
-192	2^{11}	2^{113}	2^{104}
-256	2^{12}	2^{145}	2^{136}

[*] Jang, K., Baksi, A., Kim, H., Song, G., Seo, H., & Chattopadhyay, A. Quantum Analysis of AES. IACR Communications in Cryptology, 2025.

Physical 비용 추정: Fault-Tolerant

- 실제 하드웨어의 오류 수정 비용을 고려한 **실제 Runtime 및 Physical 큐빗** 비용 분석
  - Surface 코드 기반의 오류 수정 + Reed-Muller 15-to-1 distillation*
  - Physical 파라미터는 **Gidney의 RSA 공격 논문과 동일하게** 설정*
    - Magic state injection error rate p_{in}: $\mathbf{10^{-3}}$
    - Per-gate error rate p_g : $\mathbf{10^{-4}}$ $(p_{in}/10)$
    - Surface code cycle: **1μs** (microsecond)

Stage	Remark	HQC-1	HQC-2	HQC-3
Distilleries	Code distances Factories Physical qubits	{35, 14, 7} 10923 2^{32}	{46, 18, 9} 21846 2^{34}	{56, 21, 10} 43691 2^{35}
Grover	Code distance Physical qubits	57 2^{43}	77 2^{46}	94 2^{48}
Total	Physical qubits Code cycles Runtime	2^{43} 2^{98} 9.6×10^{15} years	2^{46} 2^{134} 6.6×10^{26} years	2^{48} 2^{165} 1.35×10^{36} years

[*] Bravyi, S., Kitaev, A.: Universal quantum computation with ideal clifford gates and noisy ancillas. Physical Review A—Atomic, Molecular, and Optical Physics, 2005.
[*] Gidney, C.: How to factor 2048 bit rsa integers with less than a million noisy qubits. arXiv preprint, 2025.
[*] Gidney, C., Ekerå, M.: How to factor 2048 bit rsa integers in 8 hours using 20 million noisy qubits. Quantum 5, 2021

On the practical cost of Grover for AES key recovery

Sarah D. and Peter C.

UK National Cyber Security Centre

March 22, 2024

1 Introduction

Traditional public-key algorithms such as RSA, ECDH, and ECDSA are vulnerable to polynomial-time quantum attacks via Shor's algorithm [22]. It has been estimated that 2048-bit RSA could be broken in 8 hours on a device with 20 million physical qubits [11] and that 256-bit ECDSA could be broken in a day on a device with 13 million physical qubits [23].

On the other hand, symmetric algorithms such as AES are believed to be immune to Shor. In most cases, the best-known quantum key recovery attack uses Grover's algorithm [14] which provides a generic square-root speed-up over classical exhaustion in terms of the number of queries to the symmetric algorithm. In other words, Grover would recover the 256-bit key for AES-256 with around 2^{128} quantum queries to AES compared to around 2^{256} classical queries for exhaustion.

In theory, this means that Grover cuts the security of AES in half. However, considering only the query cost can be misleading as it neglects overheads from:

- The cost of implementing the algorithm queried by Grover as a quantum circuit;
- The cost of parallelising Grover so that a solution can be found in a reasonable amount of time; and
- The cost of quantum error correction so that Grover succeeds with high enough probability.

Previous work. The literature contains a range of estimates for the logical cost of quantum AES circuits under different optimisation targets; for example, Grassl et al. [13], Jaques et al. [16], and Jang et al. [15]. In their Call for Proposals [20], NIST provided estimates for the logical cost of Grover in terms of the total gate count when the quantum circuit was limited to a given maximum depth. Gheorghiu

D. Sarah and C. Peter, "On the Practical Cost of Grover for AES Key Recovery" 5th NIST PQC Standardization Conference, 2024.

AES 양자 공격 분석 연구: NIST 컨퍼런스

- NIST 컨퍼런스에서 발표된 **AES에 대한 양자 공격 분석 연구** (다양한 현실적 측면을 고려한)
  - 효율적인 AES 양자 회로, 양자 오류 정정 코드에 대한 기존 연구들을 기반으로 AES를 양자 컴퓨터로 해독하는 데 필요한 물리적 자원들을 추정 (Physical)

- 다음 Physical 파라미터들을 고려하였음
  - 양자 컴퓨터의 오류 수정 Cycle time: $\mathbf{200}$ **ns**
  - 양자 컴퓨터 오류율: $\mathbf{10^{-4}}$

Note: 현재 초전도 양자 컴퓨터의 오류율은 $\mathbf{10^{-3}}$, Cycle time은 **1μs** (microsecond)

Physical 비용 추정: Fault-Tolerant

- **해당 논문의 향상된 조건으로 HQC** 해킹 런타임 및 큐빗 수 재 추정
  - 오류율을 낮추고 Surface code cycle 속도 증가
    - Magic state injection error rate p_{in}: 10^{-3} → $\mathbf{10^{-4}}$
    - Per-gate error rate p_g : 10^{-4} $(p_{in}/10)$ → $\mathbf{10^{-5}}$
    - Surface code cycle: 1μs (microsecond) → **200** ns (nanoseconds)

Stage	Remark	HQC-1	HQC-2	HQC-3
Distilleries	Code distances Factories Physical qubits	{23, 9, 4} 16384 2^{31}	{31, 12, 5} 32768 2^{33}	{37, 14, 6} 65536 2^{34}
Grover	Code distance Physical qubits	29 2^{43}	40 2^{44}	49 2^{46}
Total	Physical qubits Code cycles Runtime	2^{41} 2^{97} 1.41×10^{15} years	2^{44} 2^{133} 1.3×10^{26} years	2^{46} 2^{165} 3.3×10^{35} years

요약 및 결론

- 코드기반 암호에 대한 가장 강력한 분석 기법은 **Information Set Decoding 알고리즘**
- 다양한 ISD 버전이 존재하지만, Classical 의 성능이 Quantum 에서도 동일하진 않음
  - 본 연구에서는 **기본 버전의 ISD 양자 회로 구현** (Prange ISD)
- 2025년 3월 최종 선정된 코드기반암호 **HQC를 대상으로 양자 암호 분석 수행**
  - Grover 알고리즘 + 양자 ISD 회로
  - **기초적으로 높은 큐빗과 회로 뎁스**가 요구 됨
    - AES보다 높은 양자 공격 비용 필요
  - 실제 양자 하드웨어 환경을 고려하여 Physical 비용 추정 (Runtime, Physical 큐빗)
    - **현실적으로 공격 불가** (우주나이 보다 많음)
  - 새로운 공격 알고리즘 및 기법이 나오지 않는 이상 코드기반암호는 양자 컴퓨터의 공격으로부터 매우 안전

감사합니다