# ML-KEMに対するハミング重みとビット値の相互活用による深層学習サイドチャネル攻撃

大西 健斗[1,a)]　中井 綱人[1]

**概要**：近年，量子計算機技術の進展に伴い，アメリカ国立標準技術研究所 (NIST) による標準化など，耐量子計算機暗号の必要性が高まっている．これらの暗号方式は，量子計算機による解読が困難だが，サイドチャネル攻撃に関する研究が多数報告されている．本研究では，NIST によって標準化された鍵交換方式である ML-KEM (CRYSTALS-KYBER) について，サイドチャネル攻撃に対する安全性を評価する．Ravi らによって最先端の選択平文攻撃ベースのサイドチャネル攻撃手法が提案されているが，ノイズ付きの平文のハミング重みに関するサイドチャネル情報のみを利用するため，多くの情報が利用されていない．ハミング重みにおいて多くの情報が利用されない原因は，得られる情報の少ないハミング重みが多いからである．本研究では，新たにビット値を情報として取得し，ハミング重みとの相互活用を行う手法を提案することで，秘密鍵復元に必要な波形数が，最先端の Revi らの手法では 5440 波形であるのに対し，提案手法では 4506 波形と，約 1000 波形程度削減できることを示す．

**キーワード**：深層学習サイドチャネル攻撃，耐量子暗号，ML-KEM，CRYSTALS-KYBER

# Deep Learning Side-Channel Attacks on ML-KEM through the Interutilization of Hamming Weights and Bit Values

Kento Oonishi[1,a)]　Tsunato Nakai[1]

***Abstract:*** In recent years, the advancement of quantum computing technology has heightened the necessity for post-quantum cryptography, particularly emphasized by the standardization efforts of the National Institute of Standards and Technology (NIST). While these cryptographic schemes are designed to be resilient against quantum computers, numerous studies have reported on side-channel attacks targeting them. This study evaluates the security of ML-KEM (CRYSTALS-KYBER), a key exchange mechanism standardized by NIST, against side-channel attacks. An advanced chosen plaintext attack-based side-channel attack methodology has been proposed by Ravi et al.; however, it relies solely on the side-channel information related to the Hamming weight of noisy plaintexts, resulting in a significant amount of unused information. The limited utilization of information concerning Hamming weights arises from the prevalence of Hamming weights that yield minimal information. In this study, we propose a novel method that acquires bit values as information and facilitates interutilization with Hamming weights, demonstrating that the number of waveforms required for secret key recovery can be reduced from 5,440, as per the cutting-edge method by Ravi et al., to 4,506 with our proposed approach, achieving a reduction of approximately 1,000 waveforms.

***Keywords:*** Deep Learning Side-Channel Attacks, Post-Quantum Cryptography, ML-KEM, CRYSTALS-KYBER

## 1. はじめに

### 1.1 研究背景

近年，量子計算機，特に，大規模な耐故障性量子計算機

[1] 三菱電機株式会社情報技術総合研究所
Mitsubishi Electric Corporation
[a)] Onishi.Kento@ap.MitsubishiElectric.co.jp

の実現に向けた研究が始まりつつある．上記の量子計算機が実現すると，現在利用されている RSA 暗号方式 [18] や楕円曲線暗号方式 [6], [10] が危殆化する．現状，RSA 暗号方式や楕円曲線暗号方式が危殆化する量子計算機は登場していないものの，アメリカ国立標準技術研究所 (NIST) によって，量子計算機に対する安全性を有する耐量子暗号方式の標準化が進められている [11]．この標準化では，現在，鍵交換方式 1 件と電子署名方式 2 件についての標準化文章の作成が完了しており，追加の電子署名方式の選定も進められている．　本稿では，鍵交換方式である ML-KEM (CRYSTALS-KYBER) [12] に着目して議論を行う．

ML-KEM は，Module Learning with Errors (MLWE) 問題 [9] の計算量的困難性を安全性の根拠とした暗号方式である．MLWE 問題は，格子 (Lattice) 上で定義される数学的問題である LWE 問題 [17] を，多項式環を利用した Module-lattice に拡張した問題である．MLWE 問題について，現状，古典計算機及び量子計算機の双方で，効率的に解くアルゴリズムは提案されていない．

上記の議論の通り，ML-KEM の安全性は理論的に担保されているが，実際の実装環境におけるサイドチャネル攻撃 [8] の研究報告が多数行われている [13], [14], [16]．サイドチャネル攻撃は，暗号方式を構成する暗号化や復号といった各処理について，実行時間 [8] や消費電力 [7] といった物理的な情報を測定することで，秘密鍵など，本来第三者が入手不可能な秘密情報を抽出する攻撃手法である．サイドチャネル攻撃は，実際の実装環境に依存する点から，理論的な安全性評価の枠組みとは別に，その脅威を評価することが極めて重要である．既存手法では，選択平文攻撃 [15]，既知暗号文攻撃 [13]，及び選択暗号文攻撃 [14], [16] による秘密鍵の抽出が主な脅威となっている．従来，秘密鍵の抽出は，機器で処理される情報と出力される物理量の相関を踏まえて行われてきた [13] が，近年，深層学習を利用した手法も主流になりつつある [1], [2], [4], [19]．深層学習を利用する手法では，マスキング対策やシャッフリング対策といった，サイドチャネル攻撃への基本的な対策が容易に無効化されており，安全性への影響を精密に評価する必要がある．本研究では，深層学習を利用したサイドチャネル攻撃，特に，選択平文攻撃に基づく ML-KEM の秘密鍵復元アルゴリズムを提案することで，ML-KEM の安全性評価を行う．

選択平文攻撃による深層学習サイドチャネル攻撃としては，Ravi らの攻撃手法 [15] 及びその攻撃の改良を試みた大西と中井の攻撃手法 [21] が提案されている．Ravi らは，鍵生成の encap 及び decap 処理で利用する平文に着目して，その平文を抽出する手法を提案した．まず，decap 処理の際にノイズ付きで計算される平文のハミング重みを抽出する．次に，ハミング重みを，ノイズ付き平文の存在区間 (HW 区間) に置き換えて，秘密情報に関する連立不等式を立てる．なお，この秘密情報には，鍵生成に利用する秘密鍵を含む，最後に，この連立不等式を反復法によって解いて秘密鍵を算出する．しかし，Ravi らの手法では，ハミング重みを HW 区間に置き換える際，HW 区間が広くなるハミング重みの情報を廃棄しているという課題がある．特に，本来利用可能な情報のうち，Ravi らが実際に利用している情報はたった約 23%となっているため，残りの情報の有効活用法を考えることは，攻撃に必要な波形数を減らし，より精密な安全性評価を行う点で極めて重要である．

以上を踏まえ，大西と中井は，ノイズ付きの平文の存在区間 (DL 区間) 自体の推定を行い，HW 区間と組み合わせる手法を提案した．彼らの手法では連立不等式の境界の推定を直接的に行っているため，区間幅を考慮することで，任意に波形を利用することが可能である．彼らの手法では，HW 区間と DL 区間を独立に推定し，単純な加算平均を行っているが，単純な加算平均にとどまらない相互活用を行うことで，よりサイドチャネル情報の質を高めることが可能であると予想される．以上を踏まえ，本研究では，HW 区間と DL 区間の情報を相互活用することで，ノイズ付きの平文の存在区間 (統合区間) をなるべく精度良く推定する手法を議論する．そのうえで，秘密鍵復元に必要な波形数が Ravi らの攻撃手法 [15] と比較してどの程度減少するかを実機実験により検証することで，深層学習サイドチャネル攻撃に対する ML-KEM の安全性評価を行う．

### 1.2　研究成果

本研究では，ML-KEM において，Ravi らの攻撃手法 [15] と比較して，秘密鍵復元に必要な波形数を減らすことを目的として研究を行う．

はじめに，3 節において，新たに提案するノイズ付き平文の絞り込み手法について議論する．3 節では，ノイズ付きの平文について，最上位ビットから 1 ビットずつ逐次的に学習及び推定を行い，DL 区間を推定する手法を提案する．本提案手法では，各ビットの訓練状況を確認しながら，次の下位ビットを導出するかを動的に決める．さらに，あるビット値を求める際，そのビットよりも上位のビットに着目し，それらの上位ビットの値ごとに異なる学習モデルを生成する．推定を行う際も，上位ビットの推定結果，つまり，それ以前の推定結果に応じて，対応する学習モデルを利用して推定を行う．以上により，他の上位ビットの値によるノイズを可能な限り減らしたうえで，それぞれ，学習及び推定を行う．さらに，本研究では，0 付近の値の存在区間が高精度で推定できる場合に，HW 区間の取りうる値が既存研究よりも狭まることを利用して，統合区間のさらなる絞り込みを行う．

その上で，4 節において，3 節で導出した統合区間をサイドチャネル情報として利用した際，ML-KEM の秘密鍵復元に必要な波形数を実機を用いて評価する．以上の実験で

は，ML-KEM の秘密鍵復元に必要な波形数が，最先端の Revi らの手法では 5440 波形であるのに対し，提案手法では 4506 波形と，約 1000 波形程度削減できることを示す．

以上を通して，本研究では，ML-KEM の秘密鍵をより効率的に算出する深層学習サイドチャネル攻撃及びその脅威の評価を行った．

## 2. 準備

本節では，まず，2.1 節において，ML-KEM (CRYSTALS-KYBER) [12] を説明する．さらに，Ravi らのサイドチャネル攻撃 [15] の攻撃手法を 2.2 節で説明した後，大西と中井によって提案された改良手法 [21] を 2.3 節で説明する．

### 2.1 ML-KEM [12]

ML-KEM [12] は，IND-CCA 安全性を持つ鍵交換方式である．まず，ML-KEM では，IND-CPA 安全な公開鍵暗号方式を構築する．そのうえで，Fujisaki-Okamoto 変換 [3] により，IND-CPA 安全な公開鍵暗号方式を IND-CCA 安全な鍵交換方式に変換する．以下，ML-KEM，特に，本稿で対象とする Kyber768 を説明する．

はじめに，本稿で導入する記号について説明する．まず，$q$ を素数とする．その上で，$R_q$ を $\mathbb{Z}_q[X]/(X^n+1)$ なる多項式環とする．さらに，$R_q$ を $k$ 個並べたベクトル及び $k \times k$ の行列を利用し，それぞれ，$R_q^k, R_q^{k\times k}$ と表す．ここで，ML-KEM で利用する乱数の記法について述べる．$a \leftarrow \mathcal{U}(S)$ は，集合 $S$ から $a$ を一様ランダムに選択する．また，$\boldsymbol{a} \leftarrow \chi(R_q^k)$ は，ベクトル $\boldsymbol{a}$ を構成する $k$ 個の多項式の各係数を，確率分布 $\chi$ に従って選択する．ここで，$\chi$ は，$[-\eta, \eta]$ の範囲を持つ centered binomial distribution (CBD) である．また，関数 $G$，$H$，及び $J$ はハッシュ関数を示す．なお，以上で登場した変数について，Kyber768 では，$q = 3329, n = 256, k = 3, \eta = 2$ が利用される．

以上を踏まえ，まず，ML-KEM における IND-CPA 安全な公開鍵暗号方式について説明する．公開鍵暗号方式は Algorithm 1 の通りであり，$T$ は行列やベクトルの転置を示す．平文 $m$ は，メッセージのビットから構築された多項式であり，各々の係数がビット値に対応する．復号で計算される $m'$ は，$m\lfloor q/2 \rceil$ に対して，ノイズ項 $\boldsymbol{e}$，$\boldsymbol{e}_1$，及び $e_2$ 及び暗号文の Compress，Decompress 処理による微小ノイズが付加される．しかし，これらのノイズを統合しても，ほとんどの場合 $q/4$ より小さいため，平文 $m$ を復号することが可能である．特に，Ravi ら [15] によると，$m'$ の各係数の標準偏差は 79 である．

次に，ML-KEM の IND-CCA 安全な鍵交換方式について説明する．鍵交換方式は，Algorithm 2 の通りである．Algorithm 2 において，|| は文字列の結合を示す．本方式では，PKE.Encrypt で $m$ と紐づく乱数のシード $r$ が利用され，再暗号化における整合性確認を可能としている．

**Algorithm 1** ML-KEM の公開鍵暗号方式

**PKE.Keygen:**
**Input:** なし
**Output:** 公開鍵 pk，秘密鍵 sk
$A \leftarrow \mathcal{U}\left(R_q^{k\times k}\right)$, $(\boldsymbol{s}, \boldsymbol{e}) \leftarrow \chi\left(R_q^k\right) \times \chi\left(R_q^k\right)$
$\boldsymbol{b} = A\boldsymbol{s} + \boldsymbol{e}$
**return** $\mathrm{pk} = (A, \boldsymbol{b}), \mathrm{sk} = \boldsymbol{s}$
**PKE.Encrypt:**
**Input:** 公開鍵 pk，平文 $m$，シード $r \leftarrow \mathcal{U}\left(\{0,1\}^{256}\right)$
**Output:** 暗号文 $c$
$(\boldsymbol{r}, \boldsymbol{e}_1, e_2) \leftarrow \chi\left(R_q^k\right) \times \chi\left(R_q^k\right) \times \chi\left(R_q\right)$
$\boldsymbol{u} = A^T\boldsymbol{r} + \boldsymbol{e}_1$, $v = \boldsymbol{b}^T\boldsymbol{r} + e_2 + m\lfloor q/2 \rceil$
$\boldsymbol{c}_1 = \mathrm{Compress}(\boldsymbol{u})$, $c_2 = \mathrm{Compress}(v)$
**return** $c = (\boldsymbol{c}_1, c_2)$
**PKE.Decrypt:**
**Input:** 秘密鍵 sk，暗号文 $c$
**Output:** 平文 $m$
$\boldsymbol{u}' = \mathrm{Decompress}(\boldsymbol{c}_1)$, $v' = \mathrm{Decompress}(c_2)$
$m' = v' - \boldsymbol{s}^T\boldsymbol{u}'$
$m = \mathrm{Compress}(m')$
**return** $m$

**Algorithm 2** ML-KEM の鍵交換方式

**KEM.Keygen:**
**Input:** なし
**Output:** 公開鍵 pk，秘密鍵 sk
$z \leftarrow \mathcal{U}\left(\{0,1\}^{256}\right)$
$\mathrm{pk}, \mathrm{sk}' = \mathrm{PKE.Keygen}()$
$\mathrm{sk} = \mathrm{sk}'||pk||H(\mathrm{pk})||z$
**return** pk, sk
**KEM.Encap:**
**Input:** 公開鍵 pk
**Output:** 共有鍵 $K$，暗号文 $c$
$m \leftarrow \mathcal{U}\left(\{0,1\}^{256}\right)$
$m = H(m)$
$K||r = G(m||H(\mathrm{pk}))$
$c = \mathrm{PKE.Encrypt}(pk, m, r)$
**return** $K$, $c$
**KEM.Decap:**
**Input:** 暗号文 $c$，秘密鍵 sk
**Output:** 共有鍵 $K$
pk, $\mathrm{sk}'$ を sk から抽出
$m' = \mathrm{PKE.Decrypt}(\mathrm{sk}', c)$
$K'||r' = G(m'||H(\mathrm{pk}))$
$c' = \mathrm{PKE.Encrypt}(pk, m', r')$
**if** $c \neq c'$: **then**
  $K = J(z||c)$
**else**
  $K = K'$
**end if**
**return** $K$

### 2.2 Ravi らによる既存のサイドチャネル攻撃 [15]

Ravi らの攻撃手法は，平文の加工によるサイドチャネル情報の取得と秘密鍵復元アルゴリズムの二つのステップに分類される．

はじめに，平文の加工によるサイドチャネル情報の取得について説明する．Ravi らは，KEM.Encap の $m = H(m)$ によって生成される $m$ を改ざんし，さらに，PKE.Decrypt

で計算される $m'$ をサイドチャネル情報として抽出した．ここで，抽出される情報は，$m' = v' - \boldsymbol{s}^T\boldsymbol{u}'$ の各係数であり，mod $q$ を計算する前の値が 16 ビット数として取得される．特に，$m$ の係数が 0 の場合，対応する $m'$ の係数の取りうる値の絶対値が小さく，そのハミング重みをサイドチャネル情報として利用することで，秘密情報の絞り込みが可能となる．以上を踏まえ，Ravi らの既存研究では，リファレンス実装とアセンブリ実装の二種類についてサイドチャネル情報の取得が行われている．リファレンス実装では，$m$ をランダム値に改ざんし，そのビット値 0 に対応する $m'$ の係数についてハミング重みの取得を行っており，各波形について 128 個の情報の取得が可能である．アセンブリ実装では，処理が並列実行されることから，$m$ の下位 130 ビットを 1111111110 を 13 回繰り返したパターンとし，残りをランダム生成したビットとすることで，ビット値 0 の情報以外の情報が平均して打ち消されるようにしており，各波形について 13 個の情報の取得が可能である．本研究では，後者のアセンブリ実装に着目するが，ビット値 0 に対応する $m'$ の係数のハミング重みの取得方法は同様である．ここで，Ravi らの攻撃手法では，ハミング重みに応じて，$m'$ の係数の存在区間 (HW 区間) を抽出する．ただし，表 1 より，大多数のハミング重みについて HW 区間が広くなり，後述する秘密鍵抽出アルゴリズムに無意味な情報を与えることとなる．そこで，Ravi らの論文においては，HW 区間の上限を 317 に設定し，ハミング重みが 0, 1, 12, 13, 14, 15, 及び 16 の情報を利用した．

次に，秘密鍵抽出アルゴリズムについて説明する．前述のとおり，サイドチャネル攻撃により，攻撃者は，HW 区間の情報を取得している．ここで，Compress，Decompress 処理により，$\boldsymbol{u}, v$ に生じる微小変化をそれぞれ $\boldsymbol{\Delta u}, \Delta v$ とすると，

$$
\begin{aligned}
m' &= v' - \boldsymbol{s}^T\boldsymbol{u}' && (1)\\
&= (v + \Delta v) - \boldsymbol{s}^T(\boldsymbol{u} + \boldsymbol{\Delta u}) && (2)\\
&= m\lfloor q/2 \rceil - \left(\boldsymbol{\Delta u} + \boldsymbol{e}_1^T\right)\boldsymbol{s} + \boldsymbol{r}^T\boldsymbol{e} + e_2 + \Delta v && (3)
\end{aligned}
$$

となり，攻撃者にとって，$\boldsymbol{s}, \boldsymbol{e}$ を除いた残りのすべての値が既知である．したがって，攻撃者は，先程計算された HW 区間を利用して，$\boldsymbol{s}, \boldsymbol{e}$ に関する連立不等式を立てることができる．Ravi らは，この連立不等式に対して反復法を適用することで，秘密鍵を含む秘密情報の算出を行った．この反復法の一反復では，生成した連立不等式に関して，秘密情報 $\boldsymbol{s}, \boldsymbol{e}$ を変化させた場合の合致度を計算し，その合致度が高くなるように秘密情報 $\boldsymbol{s}, \boldsymbol{e}$ を更新する，という貪欲アルゴリズムを，反復の上限回数 (既存研究では 1000 回) を指定して行っている．ただし，本アルゴリズムでは，$\boldsymbol{s}, \boldsymbol{e}$ の係数の一部を変えるだけで，式 (3) の値が離散的に大きく変化するため，必ず秘密鍵が復元できるとは限らない．

### 2.3 大西と中井による既存のサイドチャネル攻撃 [21]

2.2 節において，Ravi らの攻撃手法について説明したが，その攻撃手法の問題点は，大多数のハミング重みの情報が切り捨てられることである．そこで，大西と中井は，DL 区間，つまり，$m'$ の係数を直接的に推定し，HW 区間の情報と加算平均により統合することで，秘密鍵の復元を行う攻撃手法を提案した．彼らの手法では，$m'$ の係数のほとんどの値が $q = 3329$ を法として $[-256, 255]$ に含まれることに着目し，この区間を等分したクラスで深層学習を行うことで，DL 区間の推定を行った．彼らの手法では，HW 区間と DL 区間の推定を独立に行っている．特に，彼らの手法では，本研究では，$m'$ の係数 $x$ について，$q$ に関する法を取ったうえで，$-256 \le x < -128, -128 \le x < 0, 0 \le x < 128, 128 \le x < 256$，及びそれ以外，といった 5 つのクラスを用意して，モデルの学習を行った．しかし，上記のクラス分けでは，ハミング重みとの整合性を考慮することが出来ず，加算平均より詳細な相互活用を行うことは困難である．ハミング重みとの整合性を考慮するためには，$q$ で法をとる前の DL 区間を精度よく算出する必要がある．さらに，DL 区間の精度は，各区間によって大幅に異なることが予想され，それを踏まえた統合方法を構築することが重要である．以上を踏まえ，本研究では，区間の値をなるべく精度良く推定する手法を議論し，そのうえで，秘密鍵復元に必要な波形数を減少させる手法について議論する．

## 3. 研究成果 1: 新たなサイドチャネル情報の取得方法の提案

本節では，まず，3.1 節において，攻撃者モデルを定義する．そのうえで，3.2 節において，ビット値を繰り返し推定することによる，$m'$ の係数の存在区間の抽出方法について説明する．最後に，3.3 節において，3.2 節で算出された情報とハミング重みの情報の統合方法について説明する．なお，本節以降において，既存研究 [21] と同様，ハミング重みから算出される存在区間を HW 区間，3.2 節で算出される存在区間を DL 区間，統合された区間を統合区間と呼ぶ．

### 3.1 攻撃者モデル

本節では，本研究の攻撃者モデルを定義する．攻撃者は，Ravi らの既存研究 [15] と同様に，選択平文攻撃を行う．具体的に，攻撃者は，

- KEM.Encap の $m = H(m)$ によって生成される $m$ の改ざん
- PKE.Decrypt で計算される $m'$ に関する電力波形の取得

を行う．そのうえで，得られた電力波形に対してプロファイリング攻撃を行い，電力波形の解析を行う．なお，本研究では，大西と中井の既存研究 [21] と同様，一回取得した電力波形を，様々な条件下で学習したモデルによって解析

**表 1** ハミング重みに対応する $m'$ の係数の存在区間 (HW 区間) [15]
**Table 1** The existence interval of the coefficients of $m'$ corresponding to the Hamming weight [15]

| ハミング重み | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 最大値 | 0 | 256 | 320 | 328 | 338 | 351 | 324 | 319 | 316 | 251 | 253 | 254 | −16 | −8 | −4 | −2 | −1 |
| 最小値 | 0 | 1 | −257 | −256 | −254 | −250 | −352 | −357 | −314 | −338 | −319 | −332 | −324 | −305 | −321 | −257 | −1 |
| 区間幅 | 0 | 255 | 577 | 584 | 592 | 601 | 676 | 676 | 630 | 589 | 572 | 586 | 308 | 297 | 317 | 255 | 0 |

し，それぞれの出力結果を統合して，秘密鍵の復元を目指す．以上が，本研究における攻撃者モデルである．

### 3.2 ビット値を繰り返し推定することによる DL 区間の抽出方法

はじめに，$m'$ の係数について，攻撃者が既知の情報を整理する．まず，サイドチャネル攻撃によって得られる $m'$ の係数は 16 ビットである．さらに，$m'$ の各係数の標準偏差は 79 であり，そのほとんどの値が $q = 3329$ を法として $[-256, 255]$ に含まれる．ここで，$m'$ の各係数について，その分布は釣り鐘型 [15] であり，上記区間の境界に近づくにつれ，取りうる確率は小さくなるため，多少の区間の違いは無視することができる．このとき，微小な個数の係数を無視して，ほとんどの係数は，

(1) −256 以上 −1 以下: 上位 8 ビットが 1
(2) 0 以上 255 以下: 上位 8 ビットが 0
(3) $(3329 - 257)$ 以上 $(3329 + 254)$ 以下: 上位 7 ビットが 0000110

に含まれるとみなせる．つまり，図 1 のように，

- 最上位ビットが 1 の場合，パターン (1)，つまり，上位の 8 ビットが 1 であることが確定する．最上位ビットが 0 の場合，パターン (2) 又は (3) である．
- 最上位ビットが 0 の場合，上位から 5 ビット目が 0 の場合はパターン (2)，1 の場合はパターン (3) である．
- それ以降のビットは，0 と 1 の双方の可能性がある．ただし，0 と 1 の分布には偏りがある．

といった判別方法をとることが可能である．本稿では，DL 区間における精度を確認し，利用する情報を選別しつつ秘密鍵復元を行うため，1 ビットずつの復元を行う．それぞれのビット値の分類のため，図 1 のように，対象とするビットより上位のビットの値について，値が同一，つまり，他の上位ビットの値を持つ訓練データや推定データを含めない下で，学習モデルの構築及び推定を行う．以上により，他の上位ビットの値によるノイズを可能な限り減らしたうえで，DL 区間の学習及び推定を行う．

### 3.3 HW 区間と DL 区間の統合による統合区間の抽出方法

本節では，HW 区間と DL 区間の統合方法について説明する．本研究では，ハミング重みを，統合区間を狭めるために利用する．例えば，DL 区間が $[-256, -1]$ であり，か

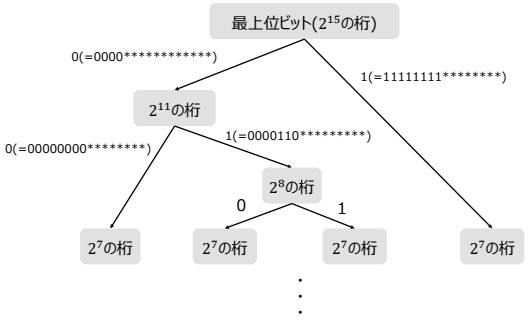


**図 1** ノイズ付き平文 $m'$ の係数の推定順序
**Fig. 1** Estimation order of coefficients of the noisy plaintext $m'$

つ，ハミング重みが 9 の場合，表 1 とは異なり，最大値は $1111111110000000_2 = -128$ となる．以上のような絞り込みは，DL 区間が $[-256, 255]$ に含まれる場合に可能となる．上記の絞り込みは，$m'$ の各係数の分布が釣り鐘型であることより，発生確率の低い統合区間に着目する．したがって，不等式を満たす $\boldsymbol{s}, \boldsymbol{e}$ の候補が少なくなるので，より効率的に秘密鍵が復元可能になる．

しかし，取得されたハミング重みの正答率は，既存研究でも本研究でも約 30%と低い．そこで，本研究では，ハミング重みの重みづけ平均を利用する．本研究では，学習モデルで算出されたハミング重みの信頼度スコアを利用して，HWsmall 及び HWlarge を計算する．具体的には，以下の通り計算を行う．

- **HWsmall**: ハミング重みが 8 以下の場合，8 以下のハミング重みに着目し，信頼度スコアの総和が 1 となるように正規化する．その上で，(信頼度スコア)×(ハミング重み) の総和を，HWsmall とする．
- **HWlarge**: ハミング重みが 8 以上の場合，8 以上のハミング重みに着目し，信頼度スコアの総和が 1 となるように正規化する．その上で，(信頼度スコア)×(16−ハミング重み) の総和を，HWlarge とする．

また，同様に，すべてのハミング重みの信頼度スコアの総和が 1 となるように正規化したうえで，(信頼度スコア)×(ハミング重み) の総和を計算し，四捨五入して，平均ハミング重み HWave とする．

以上のハミング重みの重みづけ平均を利用する，HW 区間と 3.2 節で算出された DL 区間の統合方法は以下の通りである．ただし，以下の統合方法は学習モデルの精度が高

いことを仮定しており，4 節で述べる通り，実際には，精度の低い情報を除外しながら統合区間を算出する．

(1) ハミング重みが 0 又は 16 のとき，表 1 の HW 区間を統合区間に設定して，終了する．そうでない場合は (2) に進む．

(2) DL 区間が $[-256,-1]$ に含まれ，かつ，HWave が 8 以上の場合，統合区間の最大値を $-2^{\mathrm{HWlarge}}$ とする．$2^{\mathrm{HWlarge}}$ の部分は，整数への切り捨てを行う．ただし，DL 区間の最小値より $-2^{\mathrm{HWlarge}}$ の方が小さくなった場合には，統合区間の最大値，最小値をともに DL 区間の最小値とする．以上を満たさない場合は (3) に進む．

(3) DL 区間が $[0,255]$ に含まれ，かつ，HWave が 8 以下の場合，統合区間の最小値を $2^{\mathrm{HWsmall}}$ とする．$2^{\mathrm{HWsmall}}$ の部分は，整数への切り捨てを行う．ただし，DL 区間の最大値より $2^{\mathrm{HWsmall}}$ の方が大きくなった場合には，統合区間の最大値，最小値をともに DL 区間の最大値とする．以上を満たさない場合は (4) に進む．

(4) HWave が 1 以下又は 12 以上のとき，HWave に基づき，表 1 の HW 区間を統合区間に設定して，終了する．そうでない場合は (5) に進む．

(5) DL 区間を統合区間に設定して，終了する．

4 節では，以上の統合区間の算出方法をベースに，秘密鍵復元に必要な波形数がどの程度減少するかを検証する．

## 4. 研究成果 2: 実波形に対する秘密鍵復元アルゴリズムの適用による安全性評価

本節では，ML-KEM の実波形に対して，3 節に基づいたサイドチャネル情報の取得を行うとともに，秘密鍵復元アルゴリズムを適用することで，実機を用いた秘密鍵の復元実験を行う．まず，4.1 節において，本研究で利用する実験環境について述べる．次に，4.2 節において，ベンチマークとして，ハミング重みに基づく秘密鍵復元状況について述べる．最後に，4.3 節において，本研究の提案手法による秘密鍵復元結果を述べる．なお，既存研究 [15] の秘密鍵復元アルゴリズムは反復法であり，たとえ十分な情報が揃っていても，復元対象である $\boldsymbol{s},\boldsymbol{e}$ の 1536 個の係数が完全に復元できるとは限らない．そこで，本研究では，1000 回の反復のうち 1 回でも，1535 個の係数の復元に成功している状況があれば，秘密鍵の復元に成功したとみなす．1535 個の係数の復元に成功している場合には，各反復終了時に，

(1) $\boldsymbol{s},\boldsymbol{e}$ のいずれか一つの係数を総当たりで操作する．(操作しないパターンを含む)

(2) それぞれの $\boldsymbol{s},\boldsymbol{e}$ について，$\boldsymbol{b}=A\boldsymbol{s}+\boldsymbol{e}$ なる方程式を満たすかどうか確認する．

なる $1536\times 5=7680$ 回の総当たりを行うことで，秘密鍵の復元が可能である．なお，本研究では，総当たり可能な範囲を想定して成功条件を設定したが，その成功条件の設定に関する考察は今後の課題である．

**表 2** 本稿で利用する深層学習モデル
**Table 2** The deep learning models utilized in this paper

| 層の種類 | ノード数 | 活性化関数 |
|---|---|---|
| 畳み込み層 | (Input)×1ch→(Input)×4ch | SeLU |
| バッチ正規化層 | (Input)×4ch→(Input)×4ch | – |
| プーリング層 | (Input)×4ch→(Input)×2ch | – |
| 一次元化 | (Input)×2ch→(2×Input)×1ch | – |
| 全結合層 | (2×Input)→10 | SeLU |
| 全結合層 | 10→10 | SeLU |
| 全結合層 | 10→(Output) | Softmax |

### 4.1 実験環境

本研究では，pqm4 ライブラリ [5] に格納されている Kyber768 の m4fspeed に関して，波形の取得及び $m'$ の係数の取得を試みた．本研究では，16GHz で動作する実機の評価ボードを利用して実験を行い，CW308 UFO (NAE-CW308-04) で制御された STM32F4 UFO Target (NAE-CW308T-STM32F4HWC) 上で ML-KEM を動作させ，波形の取得を行った．ここで，平文や暗号文は ChipWhisperer-Lite (NPCB-CWLITECAP-01 CW1173) を通して入出力した．そのうえで，Tektronix DSA71604 のオシロスコープを利用して，125MS/s でサンプリングを行い，各波形について，50000 点を取得した．以上の条件下で，波形の取得を行った．本研究では，10000 波形を取得したが，特徴が表れていない波形 6 波形を除外した上で，訓練用の波形と検証用の波形に分割した．具体的には，訓練用の波形を 999 波形，検証用の波形を残りの 8995 波形とした．

ここで，Kyber768 の m4fspeed は，該当箇所がアセンブリで実装されている．そのため，既存研究 [15] と同様の波形に，$m$ の下位 130 ビットを 1111111110 を 13 回繰り返したパターンとし，残りをランダム生成した．そのうえで，訓練用の波形 999 波形を利用した t 検定を行い，リークポイントを抽出した．具体的には，$m$ の下位 130 ビットについて，0 となる部分 (13 箇所) に対応する $m'$ の係数のハミング重みが 8 未満か 8 以上かの二値分類を行い，t 値が 5 以上になるリークポイントを抽出した．

以上で得られるリークポイントからなる点の集合について，本研究では，表 2 で与えられる，Wouters ら [20] の深層学習モデルを利用した解析を行った．深層学習モデルは，$m$ のビット位置及び判定条件ごとに生成する．表 2 において，(Input) は $m$ のビット位置ごとに異なる入力点数を表し，(Output) は，各判定条件のクラス数を表す．以上の深層学習モデルそれぞれについて，学習率 $5\times 10^{-3}$，バッチサイズ 10 で，100 epoch 分の学習を行った．

### 4.2 基礎実験: ハミング重みのみを利用した場合の秘密鍵復元

はじめに，既存研究 [15] との比較を行うため，ハミング重みを本研究の学習モデルを利用して取得した場合の秘密

表 3　ハミング重みにおける実験結果
**Table 3**　Experimental results from Hamming weight

| 不等式数 | 波形数 | 復元された係数の最大個数 | 復元 |
|---|---|---|---|
| 31000 | **5440** | 1536 | **成功** |
| 30000 | 5267 | 1528 | 失敗 |
| 29000 | 5094 | 1472 | 失敗 |
| 28000 | 4923 | 1339 | 失敗 |
| 27000 | 4747 | 1318 | 失敗 |
| 26000 | 4564 | 1291 | 失敗 |

表 4　提案手法における実験結果
**Table 4**　Experimental results the proposed method

| 不等式数 | 波形数 | 復元された係数の最大個数 | 復元 |
|---|---|---|---|
| 45000 | **4722** | 1536 | **成功** |
| 44000 | **4610** | 1535 | **成功** |
| 43000 | **4506** | 1536 | **成功** |
| 42000 | 4406 | 1531 | 失敗 |
| 41000 | 4301 | 1531 | 失敗 |
| 40000 | 4191 | 1415 | 失敗 |

鍵復元性能について議論する．本研究では，連立不等式における不等式数を指定して，その不等式に達するまで波形を利用する．本研究では，不等式を 1000 個ずつ減らしていき，復元される秘密鍵係数の最大個数を観測する．ハミング重みを利用した場合の実験結果は，表 3 の通りである．ここで，既存研究 [15] では約 7800 波形が秘密鍵復元に必要であったが，本研究の深層学習モデルでは，表 3 の 1 行目の通り，5440 波形が必要である．それ以降，波形数が減少するとともに，復元される秘密鍵係数の最大個数も減少する．4.3 節では，ハミング重みのみを利用した場合の 5440 波数より少ない波形数での攻撃を目指す．

### 4.3　提案手法を利用した場合の秘密鍵復元

本節では，提案手法を実際に適用した際の秘密鍵復元性能について議論する．はじめに，3 節に従って検証データに対するビット値の分類を行い，13 箇所の $m$ のビット位置ごとに (正答数)/(全数) を算出し，その平均を，平均分類精度として算出した．その結果，平均分類精度は，

- 最上位ビットの平均分類精度: 98.3%
- 上位 5 ビット目の平均分類精度 (最上位ビットが 0 の場合に利用): 75.4%
- 上位 8 ビット目の平均分類精度 ($q$ 付近の値に利用): 64.9%

となるため，これらの精度が逐次的に適用されることを踏まえると，

- $[-256,-1]$ の精度はとても高い．
- $[0,255]$ の精度はあまり高くない．
- $[3329-257, 3329+254]$ の精度は，区間が広いのに関わらず低い．

となった．ただし，この精度は，0 と 1 のクラスを一緒に評価しており，クラス間のサンプル数や精度の違いは表れていないことに注意されたい．しかし，精度の高い $[-256,-1]$ と判定された区間について，不等式数 35000(波形数 5381) で秘密鍵復元実験を行ったところ，表 3 よりも区間幅の小さい部分が多く，波形数及び不等式数が多いにも関わらず，復元された係数の最大個数は 1184 個となり，秘密鍵復元に失敗した．以上の現象は，3.3 節で議論した通り，不等式を満たす $\boldsymbol{s}, \boldsymbol{e}$ の候補の絞り込みが行えていないことが原因である．したがって，本研究では，3.3 節で議論した手法で，秘密鍵復元を行い，より少ない波形数での攻撃を目指す．

本研究では，DL 区間の区間幅を 255 に設定し，

- $[-256,-1]$ に分類された係数の情報は，DL 区間として利用する．
- $[0,255]$ に分類された係数のうち，学習モデルの訓練データにおける精度が担保されており，かつ，ハミング重みが 3 以下と判定された係数の情報を DL 区間として利用する．なお，ここで，訓練データにおける精度は 85%以上に設定した．
- それ以外の係数は廃棄する．

として，DL 区間を算出したうえで，HW 区間との統合を行った．

以上により求まった統合区間を利用した場合の実験結果は，表 4 の通りである．表 4 より，秘密鍵復元に必要な波形数は 4506 波形となり，HW 区間単体の場合と比較して，約 1000 波形程度削減することができた．さらに，復元された係数の最大個数に着目した場合，既存手法では，表 3 より，4923 波形で，すでに 200 個程度の係数の復元が出来ていないのに対して，本研究では，それ以下の波形数でも秘密鍵の復元に成功している．したがって，今後，復元されていない係数を復元する過程を考慮する際にも，本研究の手法は，より少ない波形数での復元が可能となることが予想される．

今後の展望として，学習モデルの精度が向上したときに，提案アルゴリズムの運用を変えるなどして，精度がどの程度向上するかを検証することは，今後の ML-KEM の安全性評価にとって，極めて重要な課題である．

## 5.　おわりに

本研究では，ML-KEM において，Ravi らの攻撃手法 [15] と比較して，秘密鍵復元に必要な波形数を減らすことを目的として研究を行った．

はじめに，3 節において，新たに提案するノイズ付き平文の絞り込み手法について議論した．3 節では，ノイズ付きの平文について，最上位ビットから 1 ビットずつ逐次的に学習及び推定を行い，DL 区間を推定する手法を提案した．本提案手法では，各ビットの訓練状況を確認しながら，次の下位ビットを導出するかを動的に決めた．さらに，あ

るビット値を求める際，そのビットよりも上位のビットに着目し，それらの上位ビットの値ごとに異なる学習モデルを生成した．推定を行う際も，上位ビットの推定結果，つまり，それ以前の推定結果に応じて，対応する学習モデルを利用して推定を行った．以上により，他の上位ビットの値によるノイズを可能な限り減らしたうえで，それぞれ，学習及び推定を行った．さらに，本研究では，0 付近の値の存在区間が高精度で推定できる場合に，HW 区間の取りうる値が既存研究よりも狭まることを利用して，統合区間のさらなる絞り込みを行った．

その上で，4 節において，3 節で導出した統合区間をサイドチャネル情報として利用した際，ML-KEM の秘密鍵復元に必要な波形数を評価した．以上の実験では，ML-KEM の秘密鍵復元に必要な波形数が，最先端の Revi らの手法では 5440 波形であるのに対し，提案手法では 4506 波形と，約 1000 波形程度削減できることを示した．

以上を通して，本研究では，ML-KEM の秘密鍵をより効率的に算出する深層学習サイドチャネル攻撃及びその脅威の評価を行った．

**謝辞**　本論文に示す結果の一部は内閣府が進める経済安全保障重要技術育成プログラム（K-program）「半導体・電子機器等のハードウェアにおける不正機能排除のための検証基盤の確立」（JPNP23013）(NEDO(国立研究開発法人新エネルギー・産業技術総合開発機構)) によって得られたものである．

## 参考文献

[1] Backlund, L., Ngo, K., Gärtner, J. and Dubrova, E.: Secret key recovery attack on masked and shuffled implementations of crystals-kyber and saber, *International Conference on Applied Cryptography and Network Security*, Springer, pp. 159–177 (2023).

[2] Dubrova, E., Ngo, K., Gärtner, J. and Wang, R.: Breaking a fifth-order masked implementation of crystals-kyber by copy-paste, *Proceedings of the 10th ACM Asia public-key cryptography workshop*, pp. 10–20 (2023).

[3] Fujisaki, E. and Okamoto, T.: Secure integration of asymmetric and symmetric encryption schemes, *Journal of cryptology*, Vol. 26, No. 1, pp. 80–101 (2013).

[4] Ji, Y. and Dubrova, E.: A side-channel attack on a masked hardware implementation of CRYSTALS-Kyber, *Proceedings of the 2023 Workshop on Attacks and Solutions in Hardware Security*, pp. 27–37 (2023).

[5] Kannwischer, M. J., Petri, R., Rijneveld, J., Schwabe, P. and Stoffelen, K.: PQM4: Post-quantum crypto library for the ARM Cortex-M4, `https://github.com/mupq/pqm4`.

[6] Koblitz, N.: Elliptic curve cryptosystems, *Mathematics of computation*, Vol. 48, No. 177, pp. 203–209 (1987).

[7] Kocher, P., Jaffe, J. and Jun, B.: Differential power analysis, *Annual international cryptology conference*, Springer, pp. 388–397 (1999).

[8] Kocher, P. C.: Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems, *Annual international cryptology conference*, Springer, pp. 104–113 (1996).

[9] Langlois, A. and Stehlé, D.: Worst-case to average-case reductions for module lattices, *Designs, Codes and Cryptography*, Vol. 75, No. 3, pp. 565–599 (2015).

[10] Miller, V. S.: Use of elliptic curves in cryptography, *Conference on the theory and application of cryptographic techniques*, Springer, pp. 417–426 (1985).

[11] NIST: Post-Quantum Cryptography, `https://csrc.nist.gov/projects/post-quantum-cryptography`.

[12] NIST: FIPS 203 Module-Lattice-Based Key-Encapsulation Mechanism Standard, `https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.203.pdf` (2024).

[13] Primas, R., Pessl, P. and Mangard, S.: Single-trace side-channel attacks on masked lattice-based encryption, *International Conference on Cryptographic Hardware and Embedded Systems*, Springer, pp. 513–533 (2017).

[14] Rajendran, G., Ravi, P., D'Anvers, J.-P., Bhasin, S. and Chattopadhyay, A.: Pushing the Limits of Generic Side-Channel Attacks on LWE-based KEMs-Parallel PC Oracle Attacks on Kyber KEM and Beyond, *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 418–446 (2023).

[15] Ravi, P., Paiva, T., Jap, D., D'Anvers, J.-P. and Bhasin, S.: Defeating Low-Cost Countermeasures against Side-Channel Attacks in Lattice-based Encryption: A Case Study on Crystals-Kyber, *IACR Transactions on Cryptographic Hardware and Embedded Systems*, Vol. 2024, No. 2, pp. 795–818 (2024).

[16] Ravi, P., Roy, S. S., Chattopadhyay, A. and Bhasin, S.: Generic side-channel attacks on CCA-secure lattice-based PKE and KEMs, *IACR transactions on cryptographic hardware and embedded systems*, pp. 307–335 (2020).

[17] Regev, O.: On lattices, learning with errors, random linear codes, and cryptography, *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pp. 84–93 (2005).

[18] Rivest, R. L., Shamir, A. and Adleman, L.: A method for obtaining digital signatures and public-key cryptosystems, *Communications of the ACM*, Vol. 21, No. 2, pp. 120–126 (1978).

[19] Wang, R., Brisfors, M. and Dubrova, E.: A side-channel attack on a higher-order masked crystals-kyber implementation, *International Conference on Applied Cryptography and Network Security*, Springer, pp. 301–324 (2024).

[20] Wouters, L., Arribas, V., Gierlichs, B. and Preneel, B.: Revisiting a methodology for efficient CNN architectures in profiling attacks, *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 147–168 (2020).

[21] 大西健斗, 中井綱人：耐量子計算機暗号 KYBER に対する高効率な深層学習サイドチャネル攻撃, *SCIS2025* (2025).