Explicit bounds for generic decoding algorithms for code-based cryptography

Bernstein, D.J.; Lange, T.; Peters, C.P.; van Tilborg, H.C.A.

Published in:
International Workshop on Coding and Cryptography (WCC 2009, Ullensvang, Norway, May 10-15, 2009. Pre-proceedings)

Published: 01/01/2009

Citation for published version (APA):
Bernstein, D. J., Lange, T., Peters, C. P., & Tilborg, van, H. C. A. (2009). Explicit bounds for generic decoding algorithms for code-based cryptography. In International Workshop on Coding and Cryptography (WCC 2009, Ullensvang, Norway, May 10-15, 2009. Pre-proceedings). (pp. 168-180). Bergen: Selmer Center, University of Bergen.

Explicit Bounds for Generic Decoding Algorithms for Code-Based Cryptography

Joint work with

Daniel J. Bernstein, Tanja Lange, and Henk van Tilborg

Christiane Peters

Technische Universiteit Eindhoven

EIPSI Seminar

April 1, 2009

1. Introduction

2. Attacks on the McEliece PKC

3. Explicit Bounds for Generic Decoding Algorithms

1. Introduction

2. Attacks on the McEliece PKC

3. Explicit Bounds for Generic Decoding Algorithms

Linear codes

A binary $[n, k]$ code is a binary linear code of length n and dimension k, i.e., a k-dimensional subspace of $\mathbf{F}_2^n$.

A generator matrix of an $[n, k]$ code C is a $k \times n$ matrix G such that $C = \{\mathbf{x}\, G : \mathbf{x} \in \mathbf{F}_2^k\}$.

The matrix G corresponds to a map $\mathbf{F}_2^k \to \mathbf{F}_2^n$ sending a message $\mathbf{x}$ of length k to an n-bit string.

Example: An [8,4] code C given by its generator matrix

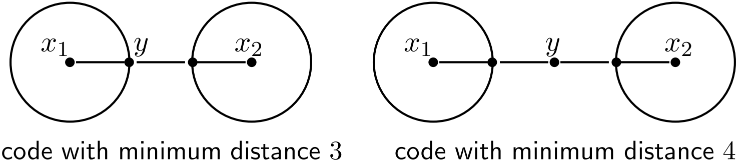
$$G = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}.$$

Example of a codeword: $\mathbf{c} = (0110)G = (11111011)$.

Hamming distance

The Hamming distance between two words in $\mathbf{F}_2^n$ is the number of coordinates where they differ. The Hamming weight of a word is the number of non-zero coordinates.

The minimum distance of a linear code C is the smallest Hamming weight of a nonzero codeword in C.



code with minimum distance 3

code with minimum distance 4

Decoding problem

Consider binary linear codes with no obvious structure.

Classical decoding problem: find the closest codeword $\mathbf{x} \in C$ to a given $\mathbf{y} \in \mathbf{F}_2^n$, assuming that there is a unique closest codeword.

Berlekamp, McEliece, van Tilborg (1978) showed that the general decoding problem for linear codes is NP-complete.

McEliece PKC from an attacker's point of view

Given a $k \times n$ generator matrix G of a public code, and an error weight w.

To encrypt a message $\mathbf{m} \in \mathbf{F}_2^k$, the sender computes $\mathbf{m}G$, adds a random weight-w error vector $\mathbf{e}$, and sends $\mathbf{y} = \mathbf{m}G + \mathbf{e}$.

Not knowing the secret code and its decoding algorithm the attacker is faced with the problem of decoding $\mathbf{y}$ in a random-looking code.

McEliece proposed choosing random degree-t classical binary Goppa codes. The standard parameter choices are $k = n - t\lceil \lg n \rceil$ and $w = t$, typically with n a power of 2.

McEliece's original suggestion: $n = 1024$, $k = 524$, and $w = 50$.

1. Introduction

2. Attacks on the McEliece PKC

3. Explicit Bounds for Generic Decoding Algorithms

Attacks on the McEliece PKC

Most effective attack against the McEliece cryptosystem is information-set decoding.

Many variants: McEliece (1978), Leon (1988), Lee and Brickell (1988), Stern (1989), van Tilburg (1990), Canteaut and Chabanne (1994), Canteaut and Chabaud (1998), and Canteaut and Sendrier (1998).

Bernstein, Lange, P. (PQCrypto 2008): improved Stern attack

Note: some of the algorithms are used for decoding; some are minimum-weight-word-finding algorithms.
For comparison we rephrase all algorithms in terms of "fixed-distance decoding".

Fixed-distance decoding

A fixed-distance-decoding algorithm searches for a codeword at a fixed distance from a received vector.

Input:	the received vector $\mathbf{y}$ and a generator matrix G for the code.
Output:	a sequence of weight-w elements $\mathbf{e} \in \mathbf{y} - \mathbf{F}_2^k G$.

Note that the output consists of error vectors $\mathbf{e}$, rather than codewords $\mathbf{y} - \mathbf{e}$.

In the important special case $\mathbf{y} = 0$, a fixed-distance-decoding algorithm searches for codewords of weight w.

Information sets

Given a generator matrix G of an $[n, k]$ code.

An information set is a size-k subset $I \subseteq \{1, 2, \ldots, n\}$ such that the I-indexed columns of G are invertible.

Denote the matrix formed by the I-indexed columns of G by G_I. The I-indexed columns of $G_I^{-1}G$ are the $k \times k$ identity matrix.

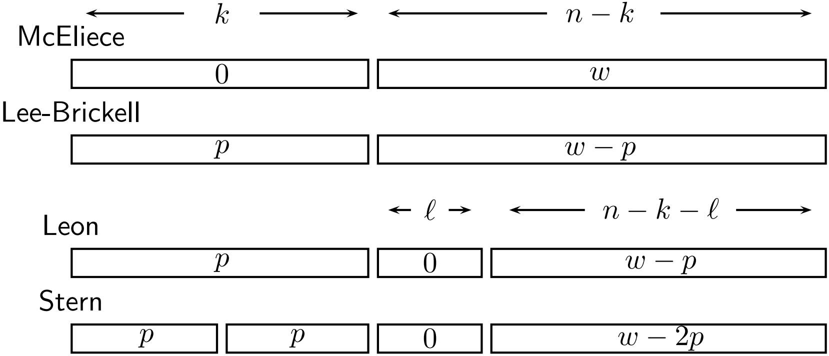
Let $\mathbf{y} \in \mathbf{F}_2^n$ have distance w to a codeword in $\mathbf{F}_2^k G$, i.e., $\mathbf{y} = \mathbf{c} + \mathbf{e}$ for a codeword $\mathbf{c} \in \mathbf{F}_2^k G$ and a vector $\mathbf{e}$ of weight w.

Denote the I-indexed positions of $\mathbf{y}$ by $\mathbf{y}_I$.

If $\mathbf{y}_I$ is error-free, $\mathbf{y}_I G_I^{-1}$ is the original message and $\mathbf{c} = (\mathbf{y}_I G_I^{-1})G$. Thus, $\mathbf{e} = \mathbf{y} - (\mathbf{y}_I G_I^{-1})G$.

Information-set decoding algorithms

Error distribution among the columns of G.



$\longleftarrow k \longrightarrow$ $\longleftarrow n-k \longrightarrow$

McEliece

0 | w

Lee-Brickell

p | $w-p$

$\leftarrow \ell \rightarrow$ $\longleftarrow n-k-\ell \longrightarrow$

Leon

p | 0 | $w-p$

Stern

p | p | 0 | $w-2p$

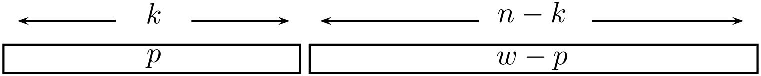
Figure from Overbeck and Sendrier: *Code-based Cryptography*, in *Post-Quantum Cryptography* (eds.: Bernstein, Buchmann, and Dahmen)

1. Introduction

2. Attacks on the McEliece PKC

3. Explicit Bounds for Generic Decoding Algorithms

Model of the number of iterations (Lee–Brickell)



If $\mathbf{e}$ is a uniform random weight-w element of $\mathbf{F}_2^n$, and I is a size-k subset of $\{1,\ldots,n\}$, then $\mathbf{e}$ has probability exactly

$$\mathrm{LBPr}(n,k,w,p) = \frac{\binom{n-k}{w-p}\binom{k}{p}}{\binom{n}{w}}$$

of having weight exactly p on I.

Consequently the Lee–Brickell algorithm, given $\mathbf{c}+\mathbf{e}$ as input for some codeword $\mathbf{c}$, has probability exactly $\mathrm{LBPr}(n,k,w,p)$ of printing $\mathbf{e}$ in the first iteration.

Note on probablities

These probabilities are averages over $\mathbf{e}$!

Extreme example Take $n = 5$, $k = 1$, $w = 1$, and $p = 0$, and consider the code $C = \{(0,0,0,0,0),(1,1,0,0,0)\}$.

The value $\mathrm{LBPr}(5,1,1,0) = \frac{\binom{4}{1}\binom{1}{0}}{\binom{5}{1}} = 4/5$ is the average of these probabilities over all choices of $\mathbf{e}$.

- If $\mathbf{e} = (0,0,1,0,0)$ or $\mathbf{e} = (0,0,0,1,0)$ or $\mathbf{e} = (0,0,0,0,1)$ then the iteration has chance 1 of printing $\mathbf{e}$;
- if $\mathbf{e} = (1,0,0,0,0)$ or $\mathbf{e} = (0,1,0,0,0)$ then the iteration has chance only $1/2$ of printing $\mathbf{e}$.

Thus, the average number of iterations is 1 for three choices of $\mathbf{e}$, and 2 for two choices of $\mathbf{e}$. The overall average, if $\mathbf{e}$ is uniformly distributed, is $7/5$, while $1/\mathrm{LBPr}(5,1,1,0) = 5/4$.

Model of the total cost (Lee–Brickell)

The function LBCost defined as

$$\mathrm{LBCost}(n,k,w,p) = \frac{\frac{1}{2}(n-k)^2(n+k) + \binom{k}{p}p(n-k)}{\mathrm{LBPr}(n,k,w,p)}.$$

is a model of the average time used by the Lee–Brickell algorithm.

- The term $\frac{1}{2}(n-k)^2(n+k)$ is a model of row-reduction time;
- $\binom{k}{p}$ is the number of size-p subsets A of $\{1,2,\ldots,k\}$;
- and $p(n-k)$ is a model of the cost of computing $y - \sum_{a\in A} G_a$.

Asymptotic analysis

Let R be the code rate and S the error fraction S; i.e., $k = Rn$ and $w = Sn$.

Goal: Measure the scalability of the information-set algorithm.

The simplest form of information-set decoding takes time $2^{(\alpha(R,S)+o(1))n}$ to find Sn errors in a dimension-Rn length-n binary code if R and S are fixed while $n \to \infty$; here

$$\alpha(R,S) = (1-R-S)\lg(1-R-S) - (1-R)\lg(1-R) - (1-S)\lg(1-S)$$

and $\lg$ means the logarithm base 2.

Stirling revisited

We assume that the code rate $R = k/n$ and error fraction $S = w/n$ satisfy $0 < S < 1 - R < 1$.

We put bounds on binomial coefficients as follows. Define $\epsilon(m)$ for each integer $m \geq 1$ by the formula

$$m! = \sqrt{2\pi}\ m^{m+1/2}\ e^{-m+\epsilon(m)}.$$

The classic Stirling approximation is $\epsilon(m) \approx 0$. Robbins showed that

$$\frac{1}{12m+1} < \epsilon(m) < \frac{1}{12m}. \tag{1}$$

Define $\mathrm{LBErr}(n, k, w, p)$ as

$$\frac{k!}{(k-p)!k^p}\,\frac{w!}{(w-p)!w^p}\,\frac{(n-k-w)!(n-k-w)^p}{(n-k-w+p)!}\,\frac{e^{\epsilon(n-k)+\epsilon(n-w)}}{e^{\epsilon(n-k-w)+\epsilon(n)}}.$$

Putting upper and lower bounds on $\mathrm{LBPr}(n,k,w,p)$

Define $\beta(R,S) = \sqrt{(1-R-S)/((1-R)(1-S))}$.

Lemma

$\mathrm{LBPr}(n,k,w,p)$ *equals*

$$2^{-\alpha(R,S)n}\,\frac{1}{p!}\,\left(\frac{RSn}{1-R-S}\right)^{p}\,\frac{1}{\beta(R,S)}\;\mathrm{LBErr}(n,k,w,p).$$

Furthermore

$$\frac{(1-\frac{p}{k})^p(1-\frac{p}{w})^p}{(1+\frac{p}{n-k-w})^p}e^{-\frac{1}{12n}(1+\frac{1}{1-R-S})} < \mathrm{LBErr}(n,k,w,p) < e^{\frac{1}{12n}(\frac{1}{1-R}+\frac{1}{1-S})}.$$

Note that for fixed rate R, fixed error fraction S, and fixed p the error factor $\mathrm{LBErr}(n,nR,nS,p)$ is close to 1 as n tends to infinity.

Comparing Lee-Brickell for various p

Corollary

$\mathrm{LBCost}(n, Rn, Sn, 0) = (c_0 + O(1/n))2^{\alpha(R,S)n}n^3$ *as* $n \to \infty$
where $c_0 = (1/2)(1 - R)(1 - R^2)\beta(R, S)$.

Corollary

$\mathrm{LBCost}(n, Rn, Sn, 1) = (c_1 + O(1/n))2^{\alpha(R,S)n}n^2$ *as* $n \to \infty$
where $c_1 = (1/2)(1 - R)(1 - R^2)(1 - R - S)(1/RS)\beta(R, S)$.

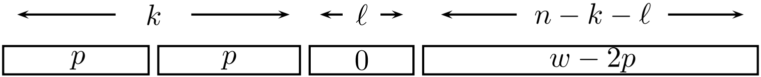
Corollary

$\mathrm{LBCost}(n, Rn, Sn, 2) = (c_2 + O(1/n))2^{\alpha(R,S)n}n$ *as* $n \to \infty$
where $c_2 = (1 - R)(1 + R^2)(1 - R - S)^2(1/RS)^2\beta(R, S)$.

Corollary

$\mathrm{LBCost}(n, Rn, Sn, 3) = (c_3 + O(1/n))2^{\alpha(R,S)n}n$ *as* $n \to \infty$
where $c_3 = 3(1 - R)(1 - R - S)^3(1/S)^3\beta(R, S)$.

Complexity of Stern's attack



k ℓ $n-k-\ell$

p p 0 $w-2p$

Model of the number of iterations Define

$$\mathrm{STPr}(n,k,w,\ell,p) = \binom{k/2}{p}^2 \binom{n-k-\ell}{w-2p} \Big/ \binom{n}{w}.$$

Define the function STCost: a model of the average time used by Stern's algorithm.

$$\mathrm{STCost}(n,k,w,\ell,p) = \frac{\left(\frac{1}{2}(n-k)^2(n+k)+2\binom{k/2}{p}p\ell+\binom{k/2}{p}^2 p(n-k)/2^\ell\right)}{\mathrm{STPr}(n,k,w,\ell,p)}.$$

Bounds on $\mathrm{STPr}(n, k, w, \ell, p)$

Define error term as $\mathrm{STErr}(n, k, w, \ell, p) = \frac{e^{\epsilon(n-k)+\epsilon(n-w)}}{e^{\epsilon(n-k-w)+\epsilon(n)}}$

$$\cdot \left(\frac{(k/2)!}{(k/2-p)!(k/2)^p}\right)^2 \frac{w!}{(w-2p)!w^{2p}} \frac{(n-k-\ell)!(n-k)^\ell}{(n-k)!} \frac{(n-k-w)!}{(n-k-\ell-w+2p)!(n-k-w)^{\ell-2p}}.$$

Lemma

$\mathrm{STPr}(n, k, w, \ell, p)$ *equals*

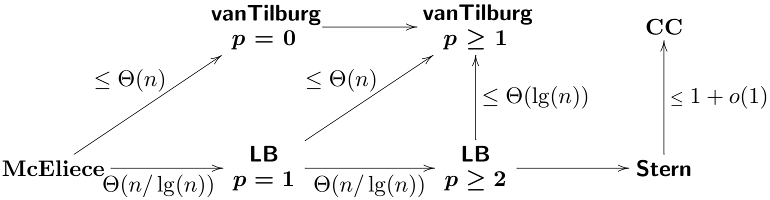
$$2^{-\alpha(R,S)n} \frac{1}{(p!)^2} \left(\frac{RSV}{2(1-R-S)}\right)^{2p} \left(\frac{1-R-S}{1-R}\right)^\ell \frac{1}{\beta(R,S)} \mathrm{STErr}(n, k, w, \ell, p).$$

Furthermore

$$(1 - \tfrac{2p}{k})^{2p}(1 - \tfrac{2p}{w})^{2p}(1 - \tfrac{n-k-\ell-w+2p}{n-k-w})^p e^{-\frac{1}{12n}\left(1 + \frac{1}{1-R-S}\right)} <$$

$$\mathrm{STErr}(n, k, w, \ell, p) < (1 + \tfrac{\ell-1}{n-k-\ell-1})^p e^{\frac{1}{12n}\left(\frac{1}{1-R} + \frac{1}{1-S}\right)}.$$

Decoding complexity comparison



- There are several variants of information-set decoding designed to reduce the cost of row reduction, sometimes at the expense of success probability.
- These variants save a non-constant factor for Lee–Brickell but save at most a factor $1 + o(1)$ for Stern. The critical point is that row reduction takes negligible time inside Stern's algorithm, since p is large.

Thank you for your attention!