Ball-collision decoding

Christiane Peters

Technische Universiteit Eindhoven

joint work with Daniel J. Bernstein and Tanja Lange

Oberseminar Cryptography and Computer Algebra
TU Darmstadt

November 18, 2010

Problem

- Given a parity-check matrix $H \in \mathbf{F}_2^{(n-k)\times n}$ of a binary linear code, a syndrome $s \in \mathbf{F}_2^{n-k}$, and a weight $w \in \{0, 1, 2, \ldots\}$.
- Assume that $w \leq$ half the minimum distance of the code with parity check matrix H.
- Attacker needs to find a vector $e \in \mathbf{F}_2^n$ of weight w such that $s = He^t$.
- Assume that the attacker does not know the structure of the underlying code.

1. The ball-collision decoding algorithm

2. Relation to previous algorithms

3. Complexity analysis

4. Concrete parameter examples

5. Asymptotic complexity

6. Choosing parameters

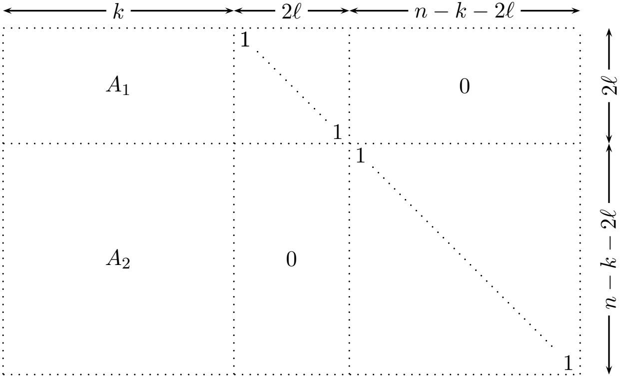
Ball-collision decoding

For simplicity assume $s = 0$. Goal: find w columns of the parity check matrix H adding up to zero.

$$\begin{array}{cccc} & \xleftrightarrow{\quad k \quad} & \xleftrightarrow{\qquad n-k \qquad} & \\ H = & \begin{array}{|c|} \hline \\ \quad A \quad \\ \\ \hline \end{array} & \begin{array}{|ccc|} \hline 1 & & \\ & \ddots & \\ & & 1 \\ \hline \end{array} & \Big\updownarrow n-k \end{array}$$

Ball-collision decoding

For simplicity assume $s = 0$. Goal: find w columns of the parity check matrix H adding up to zero.



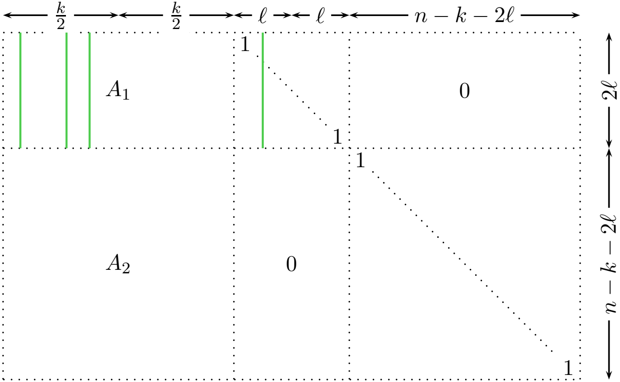
Ball-collision decoding

For simplicity assume $s = 0$. Goal: find w columns of the parity check matrix H adding up to zero.

$$\begin{array}{cc}
\begin{array}{cccc} \overset{\frac{k}{2}}{\longleftrightarrow} \; \overset{\frac{k}{2}}{\longleftrightarrow} & \overset{\ell}{\leftrightarrow} \; \overset{\ell}{\leftrightarrow} & \overset{n-k-2\ell}{\longleftrightarrow} \end{array} & \\
\left[\begin{array}{c|c|c}
A_1 & \begin{matrix} 1 & & \\ & \ddots & \\ & & 1 \end{matrix} & 0 \\
\hline
A_2 & 0 & \begin{matrix} 1 & & \\ & \ddots & \\ & & 1 \end{matrix}
\end{array}\right] & \begin{array}{c} \updownarrow 2\ell \\ \updownarrow n-k-2\ell \end{array}
\end{array}$$

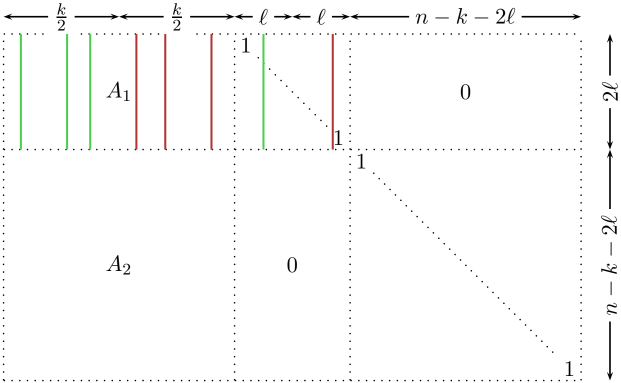
Ball-collision decoding

For simplicity assume $s = 0$. Goal: find w columns of the parity check matrix H adding up to zero.



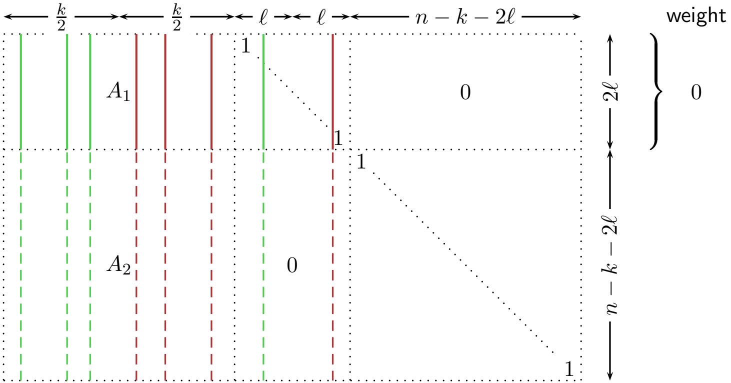
Ball-collision decoding

For simplicity assume $s = 0$. Goal: find w columns of the parity check matrix H adding up to zero.



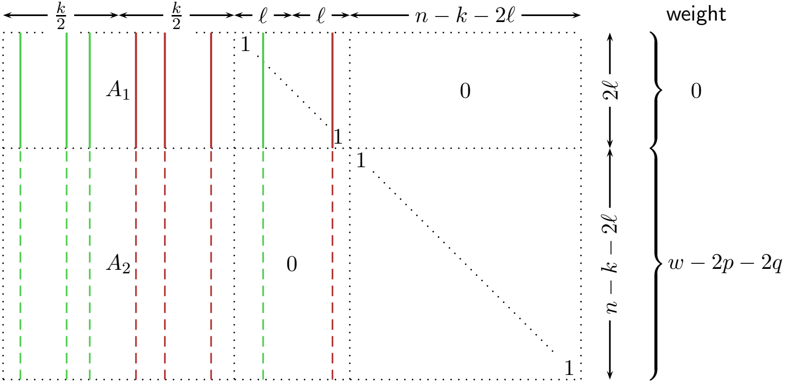
Ball-collision decoding

For simplicity assume $s = 0$. Goal: find w columns of the parity check matrix H adding up to zero.



Ball-collision decoding

For simplicity assume $s = 0$. Goal: find w columns of the parity check matrix H adding up to zero.



If the sum has weight $w-2p-2q$ add the corresponding $w-2p-2q$ columns in the $(n-k-2\ell)\times(n-k-2\ell)$ submatrix.

Else make a new column selection.

1. The ball-collision decoding algorithm
2. Relation to previous algorithms
3. Complexity analysis
4. Concrete parameter examples
5. Asymptotic complexity
6. Choosing parameters

Collision decoding

- Stern's algorithm is, aside from trivial details, exactly the special case $q = 0$.

- Dumer independently introduced the core idea, although in a more limited form, and achieved an algorithm similar to Stern's.

- Collision decoding searches for collisions in $\mathbf{F}_2^{2\ell}$ between points $A_1 x_0$ (sum of p cols on ℓ positions) and points $A_1 y_0 + s_1$ (sum of p cols + syndrome on ℓ) positions.

Supercode decoding

Ball-collision decoding is inspired by one of the steps in supercode decoding.

- [BKvT99] Barg, Krouk, van Tilborg. On the complexity of minimum distance decoding of long linear codes. IEEE Transactions on Information Theory, 45(5):1392–1405, 1999.

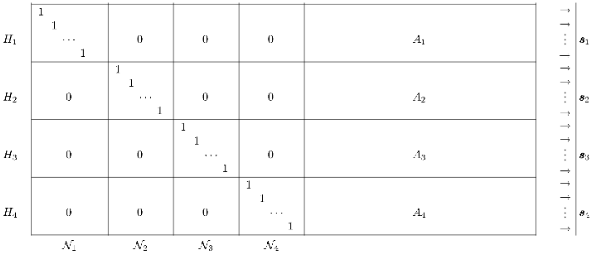
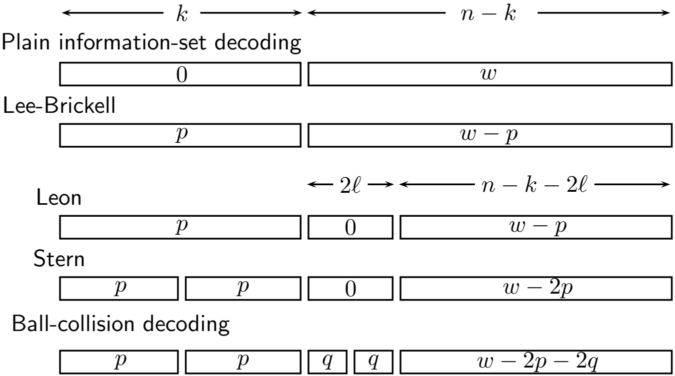


Figure in [BKvT99, Section III]

Error distribution in various ISD algorithms



Include ball-collision decoding in visual comparison by Overbeck & Sendrier. Code-based Cryptography, in *Post-Quantum Cryptography* (eds.: Bernstein, Buchmann, and Dahmen).

Ball-collision decoding

- Expand each p-sum $A_1 x_0$ into a small ball namely $\{A_1 x_0 + x_1 : x_1 \in \mathbf{F}_2^{\ell} \times \{0\}^{\ell}, \mathrm{wt}(x_1) = q\}$.

- Expand each p-sum $A_1 y_0$ into a small ball.

- Search for collisions between these balls.

Advantages of ball-collision decoding

- Disadvantage of collision decoding is that errors are required to avoid an asymptotically quite large stretch of ℓ positions.

- Requires extra work to enumerate the points in each ball, but the extra work is only about the square root of the improvement in success probability.

- The cost ratio is asymptotically superpolynomial.

1. The ball-collision decoding algorithm

2. Relation to previous algorithms

3. Complexity analysis

4. Concrete parameter examples

5. Asymptotic complexity

6. Choosing parameters

Success probability

- The chance that the algorithm succeeds after the first round is
$$\frac{\binom{k/2}{p}^2 \binom{\ell}{q}^2 \binom{n-k-2\ell}{w-2p-2q}}{\binom{n}{w}}.$$

- The expected number of iterations is very close to the reciprocal of the success probability of a single iteration.

- Ignore extremely unusual codes for which the average number of iterations is significantly different from the reciprocal of the success probability of a single iteration.

Cost of one iteration

- (Row-reduction)

$$\frac{1}{2}(n-k)^2(n+k)$$

- $+$ (Use intermediate sums to construct balls [using speedup from Bernstein-Lange-P., PQCrypto 2008]+fast handling of adding q cols)

$$2\ell\Big(2L(k/2,p)-k/2\Big)+2\min\{1,q\}\binom{k/2}{p}L(\ell,q)$$

- $+$ (Collision step: compute the whole vector and check its weight [using speedup from Bernstein-Lange-P., PQCrypto 2008])

$$2(w-2p-2q+1)(2p)\binom{k/2}{p}^2\binom{\ell}{q}^2 2^{-2\ell}$$

where $L(k,p)=\sum_{i=1}^{p}\binom{k}{i}$.

1. The ball-collision decoding algorithm
2. Relation to previous algorithms
3. Complexity analysis
4. Concrete parameter examples
5. Asymptotic complexity
6. Choosing parameters

Example #1

- Bernstein-Lange-P. (PQCrypto 2008): parameters $(6624, 5129, 117)$ achieve 256-bit security ($2^{255.87}$ bit ops)

- A lower bound on collision decoding are $2^{255.1787}$ bit operations (Finiasz–Sendrier, Asiacrypt 2009). ($1.6112985\times$ speedup compared to collision decoding)

- Ball-collision decoding with parameters $\ell = 47$, $p = 8$, and $q = 1$ needs only $2^{254.1519}$ bit operations to attack the same system.

- Ball-collision decoding results in a $3.2830\times$ speedup compared to the upper bound given at PQCrypto 2008.

Example #2

- Attacking a system based on a code with parameters $(30332, 22968, 494)$ requires $2^{1000.9577}$ bit operations using collision decoding with the optimal parameters $\ell = 140$, $p = 27$ and $q = 0$.

- The lower bound by Finiasz and Sendrier breaks the complexity down to $2^{999.45027}$, $2.8430\times$ smaller than the cost of collision decoding.

- Ball-collision decoding takes $2^{996.21534}$ bit operations. This is $26.767\times$ smaller than the cost of collision decoding, and $9.415\times$ smaller than the Finiasz–Sendrier lower bound. (using parameters $\ell = 156$, $p = 29$ and $q = 1$).

1. The ball-collision decoding algorithm

2. Relation to previous algorithms

3. Complexity analysis

4. Concrete parameter examples

5. Asymptotic complexity

6. Choosing parameters

Finiasz–Sendrier bound

- Finiasz and Sendrier. Security bounds for the design of code-based cryptosystems. Asiacrypt 2009.

  The gain of the new version of ISD is $\approx \lambda\sqrt[4]{\pi p/2}$ *which [. . .] correspond to the improvement of the "birthday paradox" part of the algorithm.*

- Any polynomial factor in n makes *no* change in the asymptotic cost exponent. (in the FS paper changes to the ℓ-stretch depend on k (which depends on n))

- The speedup from ball-collision decoding is asymptotically much larger than the speedup from the birthday trick.

Asymptotic analysis

Input sizes

- Fix a real number W with $0 < W < 1/2$, and fix a real number R with $-W \lg W - (1 - W) \lg(1 - W) \leq 1 - R < 1$.
- Consider codes and error vectors of very large length n, where the codes have dimension $k \approx Rn$, and the error vectors have weight $w \approx Wn$.

Attack parameters

- Fix real numbers P, Q, L with $0 \leq P \leq R/2$, $0 \leq Q \leq L$, and $0 \leq W - 2P - 2Q \leq 1 - R - 2L$.
- Fix ball-collision parameters p, q, ℓ with $p/n \to P$, $q/n \to Q$, and $\ell/n \to L$.

Tools

We repeatedly invoke the standard asymptotic formula for binomial coefficients, namely

$$\frac{1}{n} \lg \binom{(\alpha + o(1))n}{(\beta + o(1))n} \to \alpha \lg \alpha - \beta \lg \beta - (\alpha - \beta) \lg(\alpha - \beta).$$

Success probability

Asymptotic exponent of the success probability of a single iteration of ball-collision decoding:

$$
\begin{aligned}
B(P,Q,L) &= \lim_{n\to\infty} \frac{1}{n} \lg\left(\binom{n}{w}^{-1} \binom{n-k-2\ell}{w-2p-2q} \binom{k/2}{p}^2 \binom{\ell}{q}^2 \right) \\
&= W \lg W + (1-W)\lg(1-W) \\
&\quad + (1-R-2L)\lg(1-R-2L) \\
&\quad - (W-2P-2Q)\lg(W-2P-2Q) \\
&\quad - (1-R-2L-(W-2P-2Q)) \\
&\qquad \cdot \lg(1-R-2L-(W-2P-2Q)) \\
&\quad + R\lg(R/2) - 2P\lg P - (R-2P)\lg(R/2-P) \\
&\quad + 2L\lg L - 2Q\lg Q - 2(L-Q)\lg(L-Q).
\end{aligned}
$$

The success probability of a single iteration is asymptotically $2^{n(B(P,Q,L)+o(1))}$.

Iteration cost

We similarly compute the asymptotic exponent of the cost of an iteration:

$$
\begin{aligned}
C(P,Q,L) &= \lim_{n\to\infty} \frac{1}{n} \lg \left(2\binom{k/2}{p}\binom{\ell}{q} + \binom{k/2}{p}^2 \binom{\ell}{q}^2 2^{-2\ell} \right) \\
&= \max\{(R/2)\lg(R/2) - P\lg P \\
&\qquad - (R/2-P)\lg(R/2-P) + L\lg L - Q\lg Q \\
&\qquad - (L-Q)\lg(L-Q), \\
&\qquad R\lg(R/2) - 2P\lg P - (R-2P)\lg(R/2-P) \\
&\qquad + 2L\lg L - 2Q\lg Q \\
&\qquad - 2(L-Q)\lg(L-Q) - 2L\}.
\end{aligned}
$$

The cost of a single iteration is asymptotically $2^{n(C(P,Q,L)+o(1))}$.

Comparison (interval arithmetic)

Take $W = 0.04$ and
$R = 1 + W \lg W + (1 - W) \lg(1 - W) = 0.7577078109\ldots.$

- Choose $P = 0.004203556640625$, $Q = 0.000192998046875$, and $L = 0.017429431640625$.

- Then the ball-collision decoding exponent is $D(P, Q, L) = C(P, Q, L) - B(P, Q, L) = 0.0807023942\ldots.$

- Choosing $P = 0.00415087890625$, $Q = 0$, and $L = 0.0164931640625$ achieves decoding exponent $0.0809085120\ldots.$

- We partitioned the (P, L) space into small intervals and performed interval-arithmetic calculations to show that $Q = 0$ cannot do better than 0.0809.

Proof: $Q = 0$ is always suboptimal

Theorem

For each R, W it holds that

$$\mathit{min}\{D(P,0,L) : 0 \le P \le R/2,\ 0 \le W - 2P \le 1 - R - 2L\}$$
$$> \mathit{min}\left\{ D(P,Q,L) : \begin{array}{c} 0 \le P \le R/2,\ 0 \le Q \le L, \\ 0 \le W - 2P - 2Q \le 1 - R - 2L \end{array} \right\}.$$

Sketch:

- Increase Q from 0 to δ and increase L by $-(1/2)\delta \lg \delta$, for very small δ while obeying the parameter space.
- We show that the optimal collision-decoding parameters $(P, 0, L)$ are beaten by $(P, \delta, L - (1/2)\delta \lg \delta)$ for all sufficiently small $\delta > 0$.

Asymptotics for non-constant error fractions

- Constant rates and constant error fractions are traditional in the study of coding-theory asymptotics.
- McEliece uses error fraction approximately $(1-R)/\lg n$ ($1/\lg n$ slowly decreases to 0 as $n \to \infty$).
- Asymptotics for collision-decoding cost in general appear to have the form

  $$(1-R)^{-(1-R)n/\lg n + (\text{constant}+o(1))n/(\lg n)^2}.$$

  (Bernstein, Lange, Peters, van Tilborg, WCC 2009).
- The ball-collision-decoding speedup factor is asymptotically $2^{(c+o(1))n/(\lg n)^2}$ with $c > 0$.
- This factor is asymptotically much larger than any of the recent speedups in [BLP, PQC'08] and [FS, Asiacrypt'09].

1. The ball-collision decoding algorithm

2. Relation to previous algorithms

3. Complexity analysis

4. Concrete parameter examples

5. Asymptotic complexity

6. Choosing parameters

Lower bound

- We propose a new lower bound

$$\min\left\{\frac{1}{2}\binom{n}{w}\binom{n-k}{w-p}^{-1}\binom{k}{p}^{-1/2} : p \geq 0\right\}$$

which gives security levels in the same ballpark of the cost of known attacks.

- Parameters protecting against this bound pay only about a 20% performance penalty at high security levels, compared to parameters that merely protect against known attacks.

Conservative bound

Apply conservative bound:

- For the original parameters $(n, k, w) = (1024, 524, 50)$ the bound is $2^{49.69}$
- For $(n, k, w) = (6624, 5129, 117)$ one gets $2^{236.49}$

Our suggestion:

- $(n, k, w) = (3178, 2384, 68)$ achieve 128-bit security against our bound.
- For 256-bit security $(n, k, w) = (6944, 5208, 136)$ are recommended.

Preprint

Daniel J. Bernstein, Tanja Lange, Christiane Peters.
Ball-collision decoding. 2010.
http://eprint.iacr.org/2010/585

Thank you for your attention!