Analysis of Information Set Decoding for a Sub-linear Error Weight

Rodolfo Canto Torres, Nicolas Sendrier

▶ **To cite this version:**

Rodolfo Canto Torres, Nicolas Sendrier. Analysis of Information Set Decoding for a Sub-linear Error Weight. Post-Quantum Cryptography - PQCrypto 2016, Feb 2016, Fukuoka, Japan. hal-01244886v2

HAL Id: hal-01244886
https://hal.inria.fr/hal-01244886v2

Submitted on 16 Mar 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Analysis of Information Set Decoding for a Sub-linear Error Weight

Rodolfo Canto Torres* and Nicolas Sendrier*

Inria
{rodolfo.canto-torres,nicolas.sendrier}@inria.fr

Abstract. The security of code-based cryptography is strongly related to the hardness of generic decoding of linear codes. The best known generic decoding algorithms all derive from the Information Set Decoding algorithm proposed by Prange in 1962. The ISD algorithm was later improved by Stern in 1989 (and Dumer in 1991). Those last few years, some significant improvements have occurred. First by May, Meurer, and Thomae at Asiacrypt 2011, then by Becker, Joux, May, and Meurer at Eurocrypt 2012, and finally by May and Ozerov at Eurocrypt 2015.
With those methods, correcting w errors in a binary linear code of length n and dimension k has a cost $2^{cw(1+\mathbf{o}(1))}$ when the length n grows, where c is a constant, depending of the code rate k/n and of the error rate w/n. The above ISD variants have all improved that constant c when they appeared.
When the number of errors w is sub-linear, $w = \mathbf{o}(n)$, the cost of all ISD variants still has the form $2^{cw(1+\mathbf{o}(1))}$. We prove here that the constant c only depends of the code rate k/n and is the same for all the known ISD variants mentioned above, including the fifty years old Prange algorithm.
The most promising variants of McEliece encryption scheme use either Goppa codes, with $w = \mathbf{O}(n/\log(n))$, or MDPC codes, with $w = \mathbf{O}(\sqrt{n})$. Our result means that, in those cases, when we scale up the system parameters, the improvement of the latest variants of ISD become less and less significant. This fact has been observed already, we give here a formal proof of it. Moreover, our proof seems to indicate that any foreseeable variant of ISD should have the same asymptotic behavior.

1 Introduction

Code-based cryptography is among the most promising solutions for designing cryptosystems safe against a quantum computer. In particular the McEliece public-key encryption scheme [1], based on binary Goppa codes, has so far successfully resisted to all cryptanalysis effort. Let us also mention a recent compact key variant [2] based on quasi-cyclic moderate density parity check codes. The effective security of those schemes is based on the hardness of decoding in a binary linear code. Thus, the improvement and the understanding of the best

* This work was supported by he Commission of the European Communities through the Horizon 2020 program under project number 645622 PQCRYPTO

generic decoding technique is of great interest to select secure parameters for code-based cryptosystems. Typically, when the amount of error to correct w is proportional to the code length n, the last variant of generic decoding, proposed by May and Ozerov [3] improves the asymptotic exponent (*i.e.* decreases the number of security bits) by about 20% to 30% compared with the elementary Prange algorithm [4]. This gain decreases relatively for a smaller amount of errors. Here we prove that when the error rate w/n tends to zero, the relative gain collapses completely.

The (Computational) Syndrome Decoding Problem $\mathrm{CSD}_{n,k,w}$ consists in correcting w errors (bit flips) that have occured on a binary word belonging to a binary linear $[n,k]$ code (*i.e.* a k-dimensional subspace of $\mathbf{F}_2^n$). This problem is hard [5, 6] and is central to assess the security of code-based cryptosystems.

Information Set Decoding (ISD) was introduced by Prange in 1962 [4]. It is a generic decoding algorithm: it solves CSD taking only as inputs a basis of the code and a noisy codeword. We refer to this algorithm as Pra-ISD. There has been numerous works improving and analyzing ISD [7–14, 3]. The variants which have improved the asymptotic behavior are chronologically due to: Stern [8] and Dumer [9][1], referred to as SD-ISD; May, Meurer, and Thomae [13], referred to as MMT-ISD; Becker, Joux, May, and Meurer [14], referred to as BJMM-ISD; May and Ozerov [3], referred to as Nearest Neighbors or NN-ISD. If $\mathcal{A}$ is one of the above algorithms, we denote $\mathrm{WF}_{\mathcal{A}}(n,k,w)$ its workfactor, that is its average algorithmic cost, when addressing a (solvable) instance of $\mathrm{CSD}_{n,k,w}$.

Asymptotic Analysis of Information Set Decoding. The usual setting for the asymptotic analysis of ISD variants is to consider, for growing n, a family of problems $\mathrm{CSD}_{n,Rn,\tau n}$, with two positive constants: R, $0 < R < 1$, the code rate and τ, $0 < \tau \leq h^{-1}(1-R)$, the error rate[2]. Any known variant $\mathcal{A}$ of ISD solves this family of problems for a cost

$$\mathrm{WF}_{\mathcal{A}}(n, Rn, w = \tau n) = 2^{c'n(1+\mathbf{o}(1))} = 2^{cw(1+\mathbf{o}(1))}$$

when n grows, where the constants c' and $c = c'/\tau$ depend of R, τ, and of the variant. The various improvements of ISD have gradually improved the constant c. For instance, in Figure 1 we give the value of c for $R = 0.5$ and τ varying from 0 to $h^{-1}(0.5) \approx 0.11$. We remark in the figure that the constant c does not vary very much with the error rate τ, moreover, when this rate tends to zero, all algorithms seem to have the same value for c.

[1] The results have been obtained independently, Dumer's variant is slightly better than Stern's, though by a very small amount

[2] $h^{-1}(1-R)$ is the asymptotic Gilbert-Varshamov bound, $h(x) = -x\log_2 x - (1-x)\log_2(1-x)$ is the binary entropy.

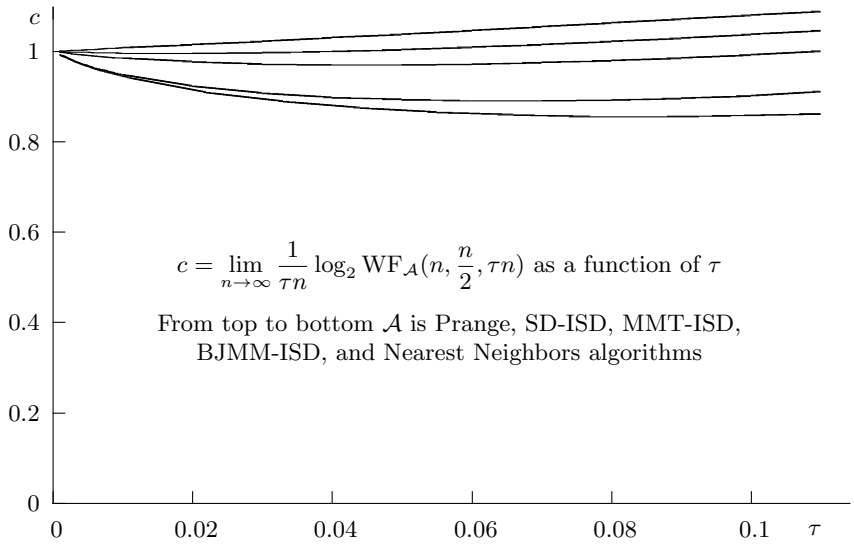


Fig. 1. Asymptotic exponent of ISD variants for binary codes of rate 1/2

Our Contribution. We prove that if we consider a family of problems $\mathrm{CSD}_{n,Rn,w}$ with $\lim_{n\to\infty} w/n = 0$, we still have

$$\mathrm{WF}_{\mathcal{A}}(n, Rn, w) = 2^{cw(1+\mathbf{o}(1))}$$

when n grows, with a constant $c = -\log_2(1-R)$ regardless of the variant. There are many situations where $w = \mathbf{o}(n)$. The two most promising variants of McEliece encryption scheme for applications are based on binary Goppa codes [1] and on binary MDPC (Moderate Density Parity Check) codes [2]. Those codes correct respectively $w = \mathbf{O}(n/\log(n))$ and $w = \mathbf{O}(\sqrt{n})$ and thus they fall into the category we are considering here.

The paper is organized as follows. We first present a framework in which the known variants of ISD all fit. This framework allows us to give bounds on the algorithmic complexity. In the next section, we use those bounds to prove that asymptotically, when the error rate tends to zero, the complexity exponent is the same for all those variants. Finally we confront this asymptotic result to what we observe when computing the non asymptotic workfactors of decoding problems corresponding to the main McEliece-like code-based encryption schemes.

2 Generic Decoding

The (computational) syndrome decoding problem is stated as follows

Problem 1 (Computational Syndrome Decoding - CSD*).*

input: $H \in \mathbf{F}_2^{(n-k)\times n}$, $s \in \mathbf{F}_2^{n-k}$, and an integer $w > 0$.
problem: Find $e \in \mathbf{F}_2^n$ of Hamming weight $\leq w$ such that $eH^T = s$.

It was proven NP-complete [5] and is conjectured hard on average [6, 15]. It is equivalent to the decoding of w errors in a binary $[n, k]$ code of parity check matrix H. Solving this problem is often the best known attack against code-based cryptosystem, thus being able to accurately analyze the cost of the best CSD solvers is of great importance to select secure parameters and to understand how to scale up the security.

Our purpose is to solve CSD(H_0, s_0, w) for some $H_0 \in \mathbf{F}_2^{(n-k)\times n}$ and $s_0 \in \mathbf{F}_2^{n-k}$. We will restrict the instance as follows.

Assumption 1 (on the instance (H_0, s_0, w) of CSD)

1. *H_0 is chosen uniformly at random in $\mathbf{F}_2^{(n-k)\times n}$*
2. *s_0 is chosen uniformly at random in $\{eH_0^T \mid \mathrm{wt}(e) = w\}$*
3. *w is smaller than the Gilbert-Varshamov distance,* i.e. $\binom{n}{w} < 2^{n-k}$.

When this holds CSD(H_0, s_0, w) has exactly one solution with high probability.

2.1 Information Set Decoding and some Variants

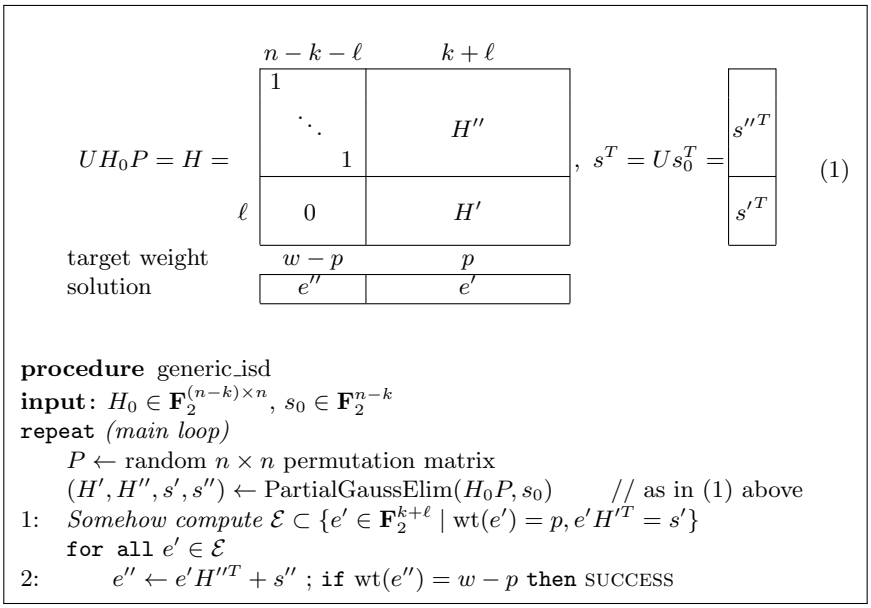
We give in Figure 2 a framework for many variants of ISD (all but the last one NN-ISD). This framework includes two additional integer parameters, p and ℓ, which will be chosen to minimize the cost of the algorithm. The optimal values of p and ℓ will depend on how instruction "1:" is implemented. The Prange algorithm corresponds to the degenerate case $p = \ell = 0$.

Proposition 1. *Within Assumption 1 on the input, we run the* generic_isd *procedure until the* SUCCESS *condition is met. The following holds on average up to a small constant factor:*

- *the instruction "1:" is executed at least* $\frac{\binom{n}{w}}{\binom{n-k-\ell}{w-p}\binom{k+\ell}{p}}$ *times,*
- *the instruction "2:" is executed at least* $\frac{\binom{n}{w}}{\binom{n-k-\ell}{w-p}2^\ell}$ *times.*

Proof is given in appendix.

Short Description of ISD Variants. We do not mean to be exhaustive nor self-contained here. We just give indications to estimate the cost of the algorithms. We refer the reader to the corresponding papers for a more detailed description. More specifically, we are interested in finding a lower bound on the cost L of one execution of instruction "1:" in Figure 2. We will use the notation of that Figure.



procedure generic_isd
input: $H_0 \in \mathbf{F}_2^{(n-k)\times n}$, $s_0 \in \mathbf{F}_2^{n-k}$
`repeat` *(main loop)*
  $P \leftarrow$ random $n \times n$ permutation matrix
  $(H', H'', s', s'') \leftarrow \text{PartialGaussElim}(H_0P, s_0)$ // as in (1) above
1: *Somehow compute* $\mathcal{E} \subset \{e' \in \mathbf{F}_2^{k+\ell} \mid \text{wt}(e') = p, e'H'^T = s'\}$
  `for all` $e' \in \mathcal{E}$
2: $e'' \leftarrow e'H''^T + s''$; `if` $\text{wt}(e'') = w - p$ `then` SUCCESS

Fig. 2. A generic framework for most ISD variants

The Stern-Dumer Variant: The instruction "1:" is performed by a birthday paradox. In that case, two lists of size $\binom{(k+\ell)/2}{p/2}$ are built then joined. We have $L \geq \binom{(k+\ell)/2}{p/2}$.

The MMT Variant: Four lists are joined in a two level tree structure. Four initial lists are joined pairwise to obtain two lists which are joined to produce $\mathcal{E}$. The four initial lists have size $\binom{(k+\ell)/2}{p/4}$, therefore $L \geq \binom{(k+\ell)/2}{p/4}$.

The BJMM Variant: The tree structure to join the lists has three levels and we initially build 8 lists of size $\binom{(k+\ell)/2}{p_2/2}$ with $p_2 = p/8+\varepsilon_1/4+\varepsilon_2/2$ where ε_1 and ε_2 are positive additional parameters. It follows that $L \geq \binom{(k+\ell)/2}{p_2/2} \geq \binom{(k+\ell)/2}{p/8}$. We remark in addition that we should also have $\binom{(k+\ell)/2}{p_2/2} \leq \binom{k+\ell}{p}$ else the algorithm would not perform better than a mere enumeration, in particular not better than SD-ISD, which cannot happen since SD-ISD is a particular case of BJMM-ISD in which some optimization parameters are restricted. It follows that the optimal value of p_2 is proportional to p with a ratio somwhere between 0.25 and 2. Because $p_2 = p/8+\varepsilon_1/4+\varepsilon_2/2$, it also follows that $\varepsilon_1 = \mathbf{O}(p)$ and $\varepsilon_2 = \mathbf{O}(p)$.

The Nearest Neighbors Variant: This most recent variant does not fit exactly into the framework of Figure 2. Still, we have the two parameters p and ℓ and the same "*main loop*" starting with same partial Gaussian elimination. Next, it starts as the BJMM variants by building 8 lists of size $\binom{(k+\ell)/2}{p_2/2}$ (with $p_2 =$

$p/8 + \varepsilon_1/4 + \varepsilon_2/2$ as in BJMM-ISD). The tree structure to join the lists is the same as BJMM except for the last join which is replaced by a "nearest neighbors" search. We do not need the analyze further to find a lower bound.

The algorithm has the same *main loop* which succeeds with probability at most $\frac{\binom{n-k-\ell}{w-p}\binom{k+\ell}{p}}{\binom{n}{w}}$ and because it starts as BJMM, the total cost of an execution of the algorithm is at least

$$\mathrm{WF}_{\mathrm{NN-ISD}}(n,k,w) \geq \min_{p,\ell} \frac{\binom{n}{w}\binom{(k+\ell)/2}{p/8}}{\binom{n-k-\ell}{w-p}\binom{k+\ell}{p}} \geq \min_{p,\ell} \frac{\binom{n}{w}}{\binom{n-k-\ell}{w-p}\binom{k+\ell}{\frac{7}{8}p}} \tag{2}$$

for large enough n, k (see proof of Corollary 1 in appendix for a proof of the rightmost inequality above). Moreover, from the algorithm description [3] the optimal value of ℓ verifies

$$2^\ell = \binom{p}{p/2}\binom{k+\ell-p}{\varepsilon_1}.$$

Finally note that, as in BJMM-ISD, we must have p_2 proportional to p, else the algorithm is outperformed by simpler variants. In particular, this means that $\varepsilon_1 = \mathbf{O}(p)$.

Lower Bound for ISD Variants. The cost L of one execution of instruction "1:" depends of the variants, from above we easily obtain the following bounds:

- for the SD-ISD variant we have $L \geq \binom{(k+\ell)/2}{p/2}$,
- for the MMT-ISD variant we have $L \geq \binom{(k+\ell)/2}{p/4}$,
- for the BJMM-ISD variant we have $L \geq \binom{(k+\ell)/2}{p/8}$.

Except for SD-ISD, the above bounds are loose. Nevertheless they are sufficient to serve our purpose, that is to prove the following statement.

Corollary 1. *For sufficiently large values of n, k, we have*

$$\mathrm{WF}_{\mathcal{A}}(n,k,w) \geq \min_{p,\ell} \frac{\binom{n}{w}}{\binom{n-k-\ell}{w-p}} \left(\frac{1}{\binom{k+\ell}{ap}} + \frac{1}{2^\ell} \right)$$

where a equals to $1/2$, $3/4$, and $7/8$ when $\mathcal{A}$ is respectively SD-ISD, MMT-ISD, and BJMM-ISD.

Proof is given in appendix.

Lower Bound for the Nearest Neighbors Variant. A lower bound is given above in equation (2). If we add to this bound that for an optimal choice of parameters we have $2^\ell = \binom{p}{p/2}\binom{k+\ell-p}{\varepsilon_1}$ and $\varepsilon_1 = \mathbf{O}(p)$, we have enough for our analysis in the next section.

3 Asymptotic Analysis

Our key result comes next and states that if the error weight w is negligible compared with n, then if we write the workfactors in the form 2^{cw}, then, when n grows, c tends to a constant which only depends of the code rate k/n.

Proposition 2. *Let k and w be two functions of n such that $\lim_{n\to\infty} k/n = R$, $0 < R < 1$, and $\lim_{n\to\infty} w/n = 0$. For any algorithm $\mathcal{A}$ among Pra-ISD, SD-ISD, MMT-ISD, BJMM-ISD, and NN-ISD, we have*

$$\mathrm{WF}_{\mathcal{A}}(n,k,w) = 2^{cw(1+\mathbf{o}(1))}, c = \log_2 \frac{1}{1-R}$$

when n tends to infinity.

The rest of this section is devoted to a proof of the above statement.

3.1 Main Theorem

We will divide the proof of the last proposition into two of cases: when $\mathcal{A}$ is among Pra-ISD, SD-ISD, MMT-ISD or BJMM-ISD and, finally, when $\mathcal{A}$ is NN-ISD. The first case is solved by the next theorem. We will prove the second case differently but with similar techniques. The proofs of the theorem and of the lemmas can be found in Appendix B.

Theorem 1. *Let k and w be two functions of n such that $\lim_{n\to\infty} k/n = R$, $0 < R < 1$, and $\lim_{n\to\infty} w/n = 0$. For any real number a, $0 \le a < 1$, we have*

$$\lim_{n\to\infty} c_a(n,k,w) = \log_2 \frac{1}{1-R}$$

where

$$c_a(n,k,w) = \min_{p,\ell} \frac{1}{w} \log_2 \left(\frac{\binom{n}{w}}{\binom{n-k-\ell}{w-p}} \left(\frac{1}{\binom{k+\ell}{ap}} + \frac{1}{2^\ell} \right) \right) \tag{3}$$

We will first show a series of properties about the following expression, related to the workfactors of the various algorithms,

$$B_a(\ell,p) = \frac{\binom{n}{w}}{\binom{n-k-\ell}{w-p}} \left(\frac{1}{\binom{k+\ell}{ap}} + \frac{1}{2^\ell} \right).$$

The next lemma describes useful properties of the optimal arguments of B_a.

Lemma 1. *Let $\mathcal{D}$ the domain of definition of B_a and $a \in]0,1[$. If $w < \frac{n-k}{2}$ then*

$$\min_{(\ell,p)\in\mathcal{D}} B_a(\ell,p) = \min_{(\ell,p)\in\mathcal{V}} B_a(\ell,p) \quad \textit{where} \quad \mathcal{V} = \left\{ (\ell,p) \in \mathcal{D}\,,\, 2^\ell = \binom{k+\ell}{ap} \right\}.$$

Now, we will use this lemma to analyze the asymptotic behavior of parameter ℓ with respect n when we know asymptotic behavior of w and k with respect n.

Lemma 2. *If* $\lim_{n\to\infty} \frac{w}{n} = 0$, $\lim_{n\to\infty} \frac{k}{n} = R$ *and* $2^\ell = \binom{k+\ell}{ap}$, *then* $\lim_{n\to\infty} \frac{\ell}{n} = 0$.

The above lemma will allow us to "remove" ℓ from ours formulae as stated in the following lemma.

Lemma 3. *If* $\lim_{n\to\infty} \frac{w}{n} = 0$, $\lim_{n\to\infty} \frac{k}{n} = R$ *and* $\ell = \mathbf{o}(n)$, *then*

$$\frac{\binom{n}{w}}{\binom{n-k-\ell}{w-p}\binom{k+\ell}{ap}} \geq 2^{\mathbf{o}(w)} b_a(p) \quad \textit{where} \quad b_a(p) = \frac{\binom{n}{w}}{\binom{n-k}{w-p}\binom{k}{ap}}.$$

Finally, this new bound allows us to predict the asymptotic behavior of p with respect w.

Lemma 4. *If* $w = \mathbf{o}(n)$ *and* $\hat{p} = \operatorname{argmin}_p b_a(p)$ *then we have* $\frac{\hat{p}}{w} = \mathbf{O}\Big(\big(\frac{w}{n}\big)^{1-a}\Big)$.

Those lemmas tell us that if $w = \mathbf{o}(n)$ then the optimal values of the parameters ℓ and p will be such that $\ell = \mathbf{o}(n)$ and $p = \mathbf{o}(w)$. This will allow us the prove the main theorem (in Appendix) and the corollaries of the next section.

3.2 Asymptotic Behaviour of the Workfactors

Now, we have all the elements to show the first case of proposition 2.

Corollary 2. *For all* $\mathcal{A}$ *among SD-ISD, MMT-ISD, and BJMM-ISD, and for any code rate* R, $0 < R < 1$, *if* w *is a function of* n *such that* $w(n) = \mathbf{o}(n)$, *then, when* n *grows, we have*

$$\mathrm{WF}_{\mathcal{A}}(n, Rn, w) = 2^{cw(1+\mathbf{o}(1))} \ \textit{where}\ c = \log_2 \frac{1}{1-R}$$

Proof. First recall that we have

$$\mathrm{WF}_{\mathrm{Pra-ISD}}(n, Rn, w) \geq \mathrm{WF}_{\mathcal{A}}(n, Rn, w),$$

for all $\mathcal{A}$ among SD-ISD, MMT-ISD, and BJMM-ISD. The workfactor of Prange is equal to $\binom{n}{w}/\binom{n}{w-k} = B_a(0,0)$ (for any $a \in]0,1[$). So, when $w(n) = \mathbf{o}(n)$, we have $\mathrm{WF}_{\mathcal{A}}(n, Rn, w) \leq 2^{cw(1+\mathbf{o}(1))}$. The other inequality derives from Theorem 1 and Corollary 1. □

Now, we want to show the same result for the case of NN-ISD. For that purpose it is enough to show that

$$\mathrm{WF}_{\mathrm{NN-ISD}} \geq 2^{\mathbf{o}(w)} \min_p b_a(p),$$

for some $a \in]0,1[$, and then proceed as in the proof of Theorem 1 and its Corollary 2. We use the inequality given in the previous section

$$\mathrm{WF}_{\mathrm{NN-ISD}}(n, k, w) \geq \min_{p,\ell} \frac{\binom{n}{w}}{\binom{n-k-\ell}{w-p}\binom{k+\ell}{\frac{7}{8}p}},$$

where ℓ verifies $2^\ell = \binom{p}{p/2}\binom{k+\ell-p}{\varepsilon_1}$ with $\varepsilon_1 = \mathbf{O}(p)$. So, when $w = \mathbf{o}(n)$, ε_1 is also $\mathbf{o}(n)$ and

$$2^\ell \leq 2^p \binom{k+\ell}{\varepsilon_1}.$$

This is similar to lemma 2 and we can also deduce $\ell = \mathbf{o}(n)$. So, we can apply the lemma 3 and obtain

$$\mathrm{WF}_{\mathrm{NN-ISD}} \geq 2^{\mathbf{o}(w)} \min_p b_{\frac{7}{8}}(p),$$

which proves the following corollary.

Corollary 3. *For any code rate R, $0 < R < 1$, if w is a function of n such that $w(n) = \mathbf{o}(n)$, then, when n grows, we have*

$$\mathrm{WF}_{\mathrm{NN-ISD}}(n, Rn, w) = 2^{cw(1+\mathbf{o}(1))} \text{ where } c = \log_2 \frac{1}{1-R}.$$

This resolves the last case of Proposition 2.

4 Comparing With Observations

We confront here our result to estimates of ISD complexity. First in an asymptotic context, then for specific code parameters arising from variants of the McEliece encryption scheme.

4.1 Asymptotic Complexity of ISD Variants

Using ad-hoc optimization techniques, we have computed the asymptotic exponent of all variants of ISD for a code rate $R \in \{0.5, 0.75, 0.875\}$ and various error rates from 0 to the Gilbert-Varshamov bound. We observe in Figure 3 that the hierarchy is respected throughout the range. This was known up to BJMM-ISD, and expected for NN-ISD. We also observe that the asymptotic exponent $\frac{1}{\tau n}\mathrm{WF}(n, Rn, \tau n)$ obviously tends to $-\log_2(1-R)$ when $\tau \to 0$.

4.2 Non Asymptotic Complexity of ISD Variants

We examine two case, the QC-MDPC-McEliece scheme [2], and the original McEliece scheme using binary Goppa codes [1].

We compute estimates of the workfactor for various algorithms. Non asymptotic estimates for NN-ISD are not available at this moment, moreover there is a huge polynomial overhead which probably makes the algorithm unpractical at this moment for cryptographic sizes. All our numbers here are given in (log of) number of "vector operations".

In Figure 4 and Figure 5 we give security of some parameter sets respectively for QC-MDPC-McEliece and Goppa-McEliece. For the same code rate,

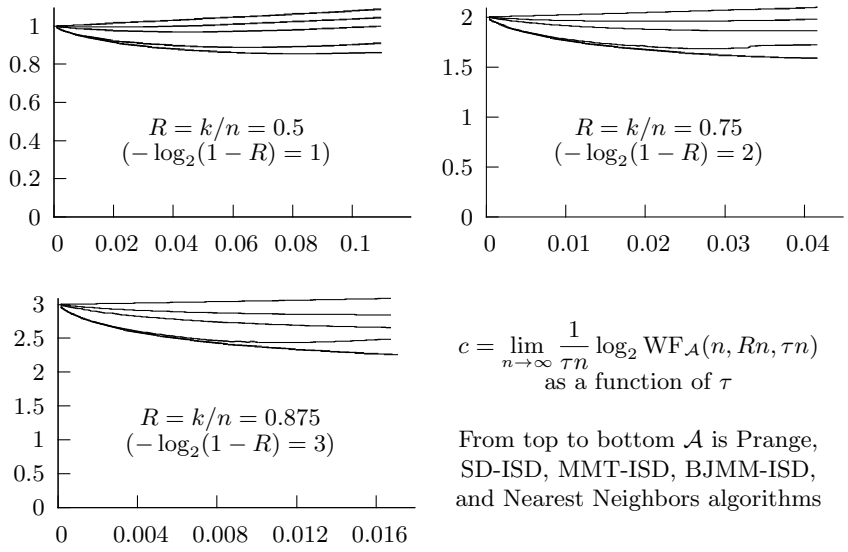


Fig. 3. Asymptotic Exponents for Variants of ISD and Various Code Rates

	security bits		
(n, k, w)	SD-ISD	MMT-ISD	BJMM-ISD
$(9602, 4801, 84)$	88.7	87.7	85.8
$(19714, 9857, 134)$	138.1	137.1	134.9
$(65536, 32768, 264)$	267.5	266.4	263.3

QC-MDPC codes parameters

	security bits		
(n, k, w)	SD-ISD	MMT-ISD	BJMM-ISD
$(780, 390, 86)$	92.7	90.4	85.5
$(1260, 630, 139)$	148.3	143.8	134.8
$(2520, 1260, 278)$	293.9	283.3	263.2

Same code rates as above, error rate at Gilbert-Varshamov

Fig. 4. Estimates for ISD Complexity Exponent for QC-MDPC Codes

	security bits		
(n, k, w)	SD-ISD	MMT-ISD	BJMM-ISD
$(2048, 1608, 40)$	89.5	87.3	81.1
$(4096, 3424, 56)$	144.3	139.5	127.6
$(8192, 6528, 128)$	290.0	280.2	256.2

Goppa codes parameters

	security bits		
(n, k, w)	SD-ISD	MMT-ISD	BJMM-ISD
$(1200, 942, 41)$	92.2	88.5	81.2
$(2400, 2006, 58)$	149.2	141.8	127.6
$(4150, 3307, 132)$	300.1	284.2	255.4

Same code rates as above, error rate at Gilbert-Varshamov

Fig. 5. Estimates for ISD Complexity Exponent for Goppa Codes

we give parameters providing the same security with the same code rate when the amount of error is close to the Gilbert-Varshamov bound.

It appears clearly in Figure 4 that the security of QC-MDPC-McEliece is not reduced by a big amount when using the most elaborate variants of ISD. In fact, because the newest variants are slightly more difficult to implement and require more memory, it is likely that the best attack in practice do not perform better than SD-ISD. This was expected form our result, since for MDPC codes the amount of error is $w = \mathbf{O}(\sqrt{n})$ and is very small compared to the length.

The situation is different for Goppa code, here we have $w = \mathbf{O}(n/\log n)$ and though w is eventually negligible compared to the code length, there is still a huge advantage in using the newest variants for codes of cryptographic size.

5 Conclusion

We have given in this paper a comprehensive way to measure the performance of the various ISD variants by writing the workfactor in the form 2^{cw} were w is the amount errors to be corrected.

The constant c does not vary very much when for the different variants of ISD. Moreover, we have proven that this constant is relatively close to $-\log_2(1-k/n)$ (where n is the code length and k the code dimension) with equality when $w \ll n$.

References

1. McEliece, R.: A public-key cryptosystem based on algebraic coding theory. DSN Prog. Rep., Jet Prop. Lab., California Inst. Technol., Pasadena, CA (January 1978) 114–116
2. Misoczki, R., Tillich, J.P., Sendrier, N., Barreto, P.S.L.M.: MDPC-McEliece: New McEliece variants from moderate density parity-check codes. In: IEEE Conference, ISIT 2013, Instanbul, Turkey (July 2013) 2069–2073

3. May, A., Ozerov, I.: On computing nearest neighbors with applications to decoding of binary linear codes. In Oswald, E., Fischlin, M., eds.: Advances in Cryptology – EUROCRYPT 2015, Part I. Volume 9056 of LNCS., Springer (2015) 203–228
4. Prange, E.: The use of information sets in decoding cyclic codes. IRE Transactions **IT-8** (1962) S5–S9
5. Berlekamp, E., McEliece, R., van Tilborg, H.: On the inherent intractability of certain coding problems. IEEE Transactions on Information Theory **24**(3) (May 1978)
6. Alekhnovich, M.: More on average case vs approximation complexity. In: FOCS 2003, IEEE (2003) 298–307
7. Lee, P., Brickell, E.: An observation on the security of McEliece's public-key cryptosystem. In Günther, C., ed.: Advances in Cryptology - EUROCRYPT '88. Volume 330 of LNCS., Springer (1988) 275–280
8. Stern, J.: A method for finding codewords of small weight. In Cohen, G., Wolfmann, J., eds.: Coding theory and applications. Volume 388 of LNCS., Springer (1989) 106–113
9. Dumer, I.: On minimum distance decoding of linear codes. In: Proc. 5th Joint Soviet-Swedish Int. Workshop Inform. Theory, Moscow (1991) 50–52
10. Canteaut, A., Chabaud, F.: A new algorithm for finding minimum-weight words in a linear code: Application to McEliece's cryptosystem and to narrow-sense BCH codes of length 511. IEEE Transactions on Information Theory **44**(1) (January 1998) 367–378
11. Finiasz, M., Sendrier, N.: Security bounds for the design of code-based cryptosystems. In Matsui, M., ed.: Advances in Cryptology - ASIACRYPT 2009. Volume 5912 of LNCS., Springer (2009) 88–105
12. Bernstein, D., Lange, T., Peters, C.: Smaller decoding exponents: Ball-collision decoding. In Rogaway, P., ed.: Advances in Cryptology - CRYPTO 2011. Volume 6841 of LNCS., Springer (2011) 743–760
13. May, A., Meurer, A., Thomae, E.: Decoding random linear codes in $\tilde{O}(2^{0.054n})$. In Lee, D., Wang, X., eds.: Advances in Cryptology - ASIACRYPT 2011. Volume 7073 of LNCS., Springer (2011) 107–124
14. Becker, A., Joux, A., May, A., Meurer, A.: Decoding random binary linear codes in $2^{n/20}$: How 1+1=0 improves information set decoding. In Pointcheval, D., Johansson, T., eds.: Advances in Cryptology - EUROCRYPT 2012. Volume 7237 of LNCS., Springer (2012) 520–536
15. Alekhnovich, M.: More on average case vs approximation complexity. Computational Complexity **20**(4) (2011) 755–786

A Proof of Proposition 1

Proof (of Proposition 1). We consider the execution of generic_isd (Fig. 2) and use the corresponding notations. If the input (H_0, s_0, w) verify Assumption 1 then so does (H, s, w) inside any particular execution of the main loop.

1. From the assumption, as long as we wish to estimate the cost up to a constant factor, we may assume that there is a unique solution to our problem. In one particular loop, we can only find an error pattern (e'', e') such that its first $n-k-\ell$ bits have weight $w-p$ and its last $k+\ell$ have weight p. This happens

with probability at most $P = \frac{\binom{n-k-\ell}{w-p}\binom{k+\ell}{p}}{\binom{n}{w}}$. Thus we expect to execute the main loop, and thus instruction "1:", at least $1/P$ times.

2. To estimate the number of times we have to compute instruction "2:", we need to estimate for any $e' \in \mathbf{F}_2^{k+\ell}$ of weight p the probability that e' leads to a success given that $e'H'^T = s'$.
   If we fix H, the sample space in which we compute the probabilities is $\Omega_H = \{eH^T \mid \mathrm{wt}(e) = w\}$ equipped with a uniform distribution (because of Assumption 1). We consider the two events
   - $\mathcal{S}_H(e') = \{s = (s'', e'H'^T) \in \Omega_H\}$,
   - $\mathrm{Succ}_H(e') = \{eH^T \mid e = (e'', e'), e'' \in \mathbf{F}_2^{n-k-\ell}, \mathrm{wt}(e'') = w - p\}$.

   The probability we are interested in is $\Pr_{\Omega_H}(\mathrm{Succ}_H(e') \mid \mathcal{S}_H(e'))$. We have $\Pr_{\Omega_H}(\mathcal{S}_H(e')) \approx 2^{-\ell}$ because we expect the set Ω_H to behave like a set of random vector (true for almost all matrix H). And the set $\mathrm{Succ}_H(e') \subset \Omega_H$ has cardinality $\binom{n-k-\ell}{w-p}$ as it contains for a fixed e', as many elements as we have vectors $e'' \in \mathbf{F}_2^{n-k-\ell}$ of weight $w - p$. Finally

$$\Pr_{\Omega_H}(\mathrm{Succ}_H(e') \mid \mathcal{S}_H(e')) = \frac{\Pr_{\Omega_H}(\mathrm{Succ}_H(e'))}{\Pr_{\Omega_H}(\mathcal{S}_H(e'))} = \frac{\binom{n-k-\ell}{w-p}2^\ell}{\binom{n}{w}}$$

   The second part of the statement follows. □

Proof (of Corollary 1). Using the fact that $\binom{n}{w}$ is proportional to $\frac{2^{nh(w/n)}}{\sqrt{w(1-w/n)}}$, where $h(x) = -x\log_2(x) - (1-x)\log_2(1-x)$ is the binary entropy function, we easily obtain that

$$\log_2 \frac{\binom{(k+\ell)/2}{(1-a)p}\binom{k+\ell}{ap}}{\binom{k+\ell}{p}} = (k+\ell)\left(\frac{h(2(1-a)x)}{2} + h(ax) - h(x)\right)(1 + \mathbf{o}(x))$$

where $x = p/(k+\ell)$. An easy study of the above function proves that it is positive for any a, $1 > a \geq 0.5$. Using Proposition 1, if $L \geq \binom{(k+\ell)/2}{(1-a)p}$ then the total contribution of instruction "1:" is at least

$$\frac{\binom{n}{w}L}{\binom{n-k-\ell}{w-p}\binom{k+\ell}{p}} \geq \frac{\binom{n}{w}\binom{(k+\ell)/2}{(1-a)p}}{\binom{n-k-\ell}{w-p}\binom{k+\ell}{p}} \geq \frac{\binom{n}{w}}{\binom{n-k-\ell}{w-p}\binom{k+\ell}{ap}}.$$

Adding to that the contribution of "2:", we obtain the lower bound of the statement. □

B Proofs of Main Theorem Section

Proof (of Lemma 1). We have

$$\log\left(B_a(\ell, p)\right) \approx \max\left\{\log\left(\frac{\binom{n}{k}}{\binom{n-k-\ell}{w-p}\binom{k+\ell}{ap}}\right), \log\left(\frac{\binom{n}{k}}{\binom{n-k-\ell}{w-p}2^\ell}\right)\right\}.$$

We divide our function in two parts

$$f(\ell,p)=\log\left(\frac{\binom{n}{k}}{\binom{n-k-\ell}{w-p}\binom{k+\ell}{ap}}\right)\quad\text{and}\quad g(\ell,p)=\log\left(\frac{\binom{n}{k}}{\binom{n-k-\ell}{w-p}2^{\ell}}\right).$$

The function B_a is defined on the domain shown in Fig. 6. Our goal is to show that there is a point $(\hat{\ell},\hat{p})\in\mathcal{D}$ such that $f(\hat{\ell},\hat{p})=g(\hat{\ell},\hat{p})$ who minimizes B_a. We will start by studying the interior of $\mathcal{D}$ and we will verify that if B_a achieve its minimum at (ℓ^*,p^*) then there is a point $(\hat{\ell},\hat{p})\in\mathcal{V}$ which B_a has the same value. Secondly, we will search all possible minimum points in the boundary $\partial\mathcal{D}$ and we will show that again the minimum is attained in a point of $\mathcal{V}$; these two cases will allow us to conclude this theorem.

We suppose $(\ell^*,p^*)\notin\partial\mathcal{D}$ minimizes B_a and it holds that $f(\ell^*,p^*)>g(\ell^*,p^*)$. So, $B_a(\ell,p)=\max\{f(\ell,p),g(\ell,p)\}$ for all (ℓ,p) in a neighborhood U of (ℓ^*,p^*) which does not intercept the boundary. Then,

$$\min_{(\ell,p)\in\mathcal{D}}B_a(\ell,p)=\min_{(\ell,p)\in U}B_a(\ell,p)=\min_{(\ell,p)\in U}f(\ell,p),$$

and in particular $\nabla f(\ell^*,p^*)=(0,0)$. Since $a\in\,]0,1\,[$, that equality has a unique solution

$$\frac{w-p^*}{n-k-\ell^*}=0\text{ or }1.$$

That means $(\ell,p)\in\partial\mathcal{D}$, so this case is impossible.

In the case where $g(\ell^*,p^*)>f(\ell^*,p^*)$, we deduce similarly $\nabla g=(0,0)$. And, we obtain

$$\begin{aligned}\frac{\partial g}{\partial\ell}&=-\log\Big(1-\frac{w-p^*}{n-k-\ell^*}\Big)-1=0.\\ \frac{\partial g}{\partial p}&=h'\Big(\frac{w-p^*}{n-k-\ell^*}\Big)=0\end{aligned}$$

Therefore,

$$\mathscr{L}^*:\quad\frac{w-p^*}{n-k-\ell^*}=\frac{1}{2},$$

this equation defines a line in the plane where $g(\ell,p)$ is constant. We use the function $p_0:[\,n-k-2w,n-k\,]\to\mathbb{R}$ defined by $p_0(\ell)=w-\frac{n-k-\ell}{2}$ to describe some points in $\mathscr{L}^*$. Now, our objetive is to show there is a point belonging to $\mathcal{V}$ in this line $\mathscr{L}^*$. By hypothesis $\ell_0=n-k-2w>0$, so $(\ell_0,p_0(\ell_0))=(\ell_0,0)\in\mathscr{L}^*\cap\mathcal{D}$ and

$$g(\ell_0,p_0(\ell_0))=nh\Big(\frac{w}{n}\Big)-(n-k-\ell_0)-\ell_0<nh\Big(\frac{w}{n}\Big)-(n-k-\ell_0)=f(\ell_0,p_0(\ell_0)).$$

Since $g(\ell^*,p_0(\ell^*))>f(\ell^*,p_0(\ell^*))$ and the segment of line between $(\ell_0,p_0(\ell_0))$ and $(\ell^*,p_0(\ell^*))$ belongs to $\mathcal{D}$, there is a $\hat{\ell}\in\,]\ell_0,\ell^*[$ such that $g(\hat{\ell},p_0(\hat{\ell}))=f(\hat{\ell},p_0(\hat{\ell}))$. Because $g(\hat{\ell},p_0(\hat{\ell}))=g(\ell^*,p(\ell^*))$, we conclude that $(\hat{\ell},p_0(\hat{\ell}))$ is a minimum point for B_a and it belongs to $\mathcal{V}$.

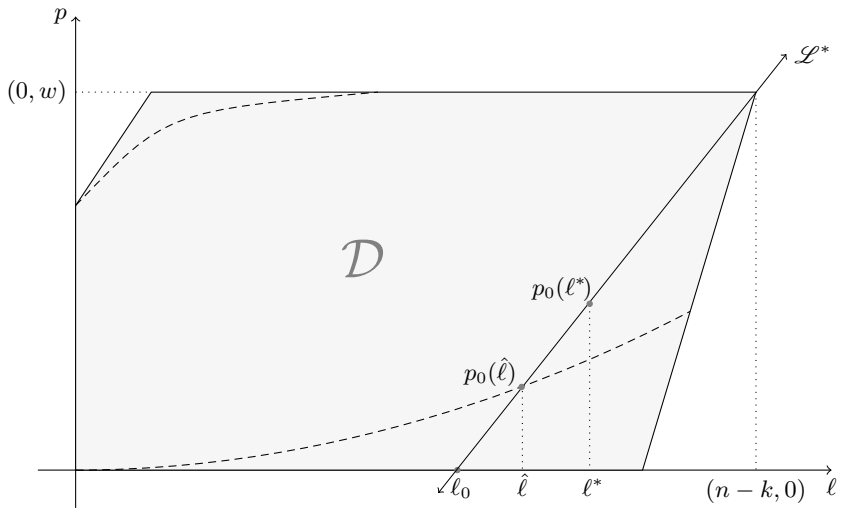


Fig. 6. Definition Domain of Function B_a

Now, we suppose that the minimum point (ℓ^*, p^*) belongs to the boundary $\partial\mathcal{D}$ and we search all possibles candidates in the boundary. We can divide the boundary into 5 segments of line and we analyze the monotony of f and g respect to ℓ or p. We will obtain that

$$\min_{\partial\mathcal{D}} f(\ell, p) = \min\{f(0,0), g(0,0), f(k/a, 0), \max\{f(n-k, w), g(n-k, w)\}\};$$

Since $f(0,0) = g(0,0) \leq g(k/a, 0) = f(k/a, 0)$, we focus our analysis on the points $(0,0)$ and $(n-k, w)$, so it is enough to study the case $(\ell^*, p^*) = (n-k, w)$. We can analyze f over the line

$$\mathscr{L}_{m_0} : \frac{w-p}{n-k-\ell} = \frac{w}{n-k} = \frac{p}{k+\ell}.$$

We can obtain a derivate function

$$\frac{\partial f}{\partial \ell} = h\Big(\frac{w}{n}\Big) - \bigg(-a\frac{w}{n}\log\Big(a\frac{w}{n}\Big) - \Big(1-a\frac{w}{n}\Big)\log\Big(1-a\frac{w}{n}\Big)\bigg) = h\Big(\frac{w}{n}\Big) - h\Big(a\frac{w}{n}\Big) > 0,$$

because $w/n < 1/2$. Therefore, B is increasing respect to ℓ into this line, and B does not achieve its local minimun at $(n-k, w)$; for that reason that minimun is not $B_a(n-k, w)$ in the case $f(n-k, w) > g(n-k, w)$.

In the case of $B_a(\ell^*, p^*) = g(n-k, w) > f(n-k, w)$, we can take any point (ℓ^{**}, p^{**}) who belongs to the interior of $\mathcal{D}$ and the line $\mathscr{L}^*$ (which we used before), and we obtain again another point $(\hat{\ell}, \hat{p}) \in \mathcal{V}$ as before. □

Proof (of Lemma 2). We take binary logarithm over the third hypothesis and we obtain

$$\begin{aligned}\frac{\ell}{n} &= \frac{p}{n}\Big(1-\log\Big(\frac{p}{k+\ell}\Big)-\mathbf{O}\Big(\frac{p}{k+\ell}\Big)\Big)+\frac{p}{n}\\ &\le \frac{w}{n}\Big(2+\mathbf{O}\Big(\frac{w}{k}\Big)\Big)-\frac{w}{n}\log\Big(\frac{w}{k}\Big)\\ &= \frac{w}{n}\Big(2-\log\Big(\frac{n}{k}\Big)+\frac{n}{k}\mathbf{O}\Big(\frac{w}{n}\Big)\Big)-\frac{w}{n}\log\Big(\frac{w}{n}\Big)\\ &= \mathbf{o}(1)-\mathbf{o}(1).\end{aligned}$$

So we conclude $\frac{\ell}{n}=\mathbf{o}(1)$. □

Proof (of Lemma 3). It is enough that we analyze the quotient of the respectives terms in these expressions:

$$\begin{aligned}\log\left(\frac{\binom{n-k-\ell}{w-p}}{\binom{n-k}{w-p}}\right) &= (n-k-\ell)h\Big(\frac{w-p}{n-k-\ell}\Big)-(n-k)h\Big(\frac{w-p}{n-k}\Big)\\ &= -(w-p)\Big(\log\Big(\frac{w-p}{n-k-\ell}\Big)+\Big(1+\mathbf{O}\big(\frac{w-p}{n-k-\ell}\big)\Big)\Big)\\ &\quad +(w-p)\Big(\log\Big(\frac{w-p}{n-k}\Big)+\Big(1+\mathbf{O}\big(\frac{w-p}{n-k}\big)\Big)\Big)\\ &\le (w-p)\Big(\log\Big(1-\frac{\ell}{n-k}\Big)+\mathbf{O}\Big(\frac{w-p}{n-k-\ell}\Big)\Big)\\ &\le w\Big(\log\Big(1-\frac{\ell}{n}\Big)+\frac{n}{n-k-\ell}\mathbf{O}\Big(\frac{w}{n}\Big)\Big)\\ &= w\mathbf{o}(1)\end{aligned}$$

In the same way, we obtain

$$\log\left(\frac{\binom{k+\ell}{ap}}{\binom{k}{ap}}\right)\le ap\Big(\log\Big(1+\frac{\ell}{k}\Big)+\frac{n}{k+\ell}\mathbf{O}\Big(\frac{w}{n}\Big)\Big)\le w\mathbf{o}(1).$$

□

Proof (of Lemma 4).

We analyze the derivate of b:

$$b'_a(p)=a\log\Big(\frac{ap}{k-ap}\Big)-\log\Big(\frac{w-p}{n-k-(w-p)}\Big).$$

We can see the function b_a is decreasing in a neighborhood of $p=0$ and increasing in a neighborhood of $p=w$. Moreover, $b''_a(p)>0$, so $b_a(p)$ is a convex function and the minimization problem has unique solution $\hat{p}$. We analyze the equation $b_a(\hat{p})=0$, we obtain

$$\frac{a^a\hat{p}}{w-\hat{p}}=\frac{(k-\hat{p})^a\hat{p}^{1-a}}{n-k-(w-\hat{p})}$$

That implies

$$a^a \frac{\hat{p}}{w} \leq \frac{k^a w^{1-a}}{n-k-w}.$$

We deduce that $\frac{\hat{p}}{w} = \mathbf{O}(\frac{w}{n}^{1-a})$. □

Now, we have all the asymptotic properties and reductions that we need to prove our principal result. So, we will use the well known Stirling's approximation for binomial coefficient

$$\binom{n}{w} \approx \frac{2^{nh(w/n)}}{\sqrt{w(1-w/n)}},$$

and we will ignore polynomial factors.

Proof (of Theorem 1).

The first estimation of c_a, when $w = \mathbf{o}(n)$, gives us

$$\begin{aligned} c_a(n,k,w) &\leq \frac{1}{w}\log\big(B_a(0,0)\big) \\ &= \frac{1}{w}\Big(\log\binom{n}{w} - \log\binom{n-k}{w}\Big) \\ &= \frac{n}{w}h\Big(\frac{w}{n}\Big) - \frac{n-k}{w}h\Big(\frac{w}{n-k}\Big) \\ &= \Big(1 - \log\Big(\frac{w}{n}\Big) + \mathbf{O}\Big(\frac{w}{n}\Big)\Big) - \Big(1 - \log\Big(\frac{w}{n-k}\Big) + \mathbf{O}\Big(\frac{w}{n-k}\Big)\Big) \\ &= \log\Big(\frac{n}{n-k} + \mathbf{O}\Big(\frac{w}{n}\Big)\Big). \end{aligned}$$

So, our objective will be show the another inequality. The lemmas 1, 2 and 3 lets us simplify the equation to that inequality

$$c_a(n,k,w) \geq \mathbf{o}(w) + \min_p \log(b_a(p)).$$

Finally, we analyze the binary logarithm of b_a evaluated in the optimal argument $\hat{p}$:

$$\log(b_a(\hat{p})) = \underbrace{nh\Big(\frac{w}{n}\Big)}_{(1)} - \underbrace{(n-k)h\Big(\frac{w-\hat{p}}{n-k}\Big)}_{(2)} - \underbrace{kh\Big(\frac{a\hat{p}}{k}\Big)}_{(3)}.$$

So, we study these three parts

$$\begin{aligned} &(1): nh\Big(\frac{w}{n}\Big) = w\Big(1 - \log\Big(\frac{w}{n}\Big) + \mathbf{O}\Big(\frac{w}{n}\Big)\Big) \\ &(2): (n-k)h\Big(\frac{w-\hat{p}}{n-k}\Big) = (w-\hat{p})\Big(1 - \log\Big(\frac{w-\hat{p}}{n-k}\Big) + \mathbf{O}(\frac{w-\hat{p}}{n-k})\Big) \\ &(3): kh\Big(\frac{a\hat{p}}{k}\Big) = a\hat{p}\Big(1 - \log\Big(\frac{a\hat{p}}{k}\Big) + \mathbf{O}\Big(\frac{a\hat{p}}{k}\Big)\Big) \end{aligned}$$

We group these terms in two sums: the sum of logarithms and the sum of negligible addends:

$$(I) = -w\log\Big(\frac{w}{n}\Big) + (w-\hat{p})\log\Big(\frac{w-\hat{p}}{n-k}\Big) + ap\log\Big(\frac{a\hat{p}}{k}\Big)$$
$$(II) = -w\Big(1+\mathbf{O}\Big(\frac{w}{n}\Big)\Big) + (w-\hat{p})\Big(1+\mathbf{O}\Big(\frac{w-\hat{p}}{n-k}\Big)\Big) + ap\Big(1+\mathbf{O}\Big(\frac{a\hat{p}}{k}\Big)\Big)$$

We continue with the easy part

$$\begin{aligned}(II) &= -w\mathbf{O}\Big(\frac{w}{n}\Big) + (a-1)\hat{p} + (w-\hat{p})\mathbf{O}\Big(\frac{w-\hat{p}}{n-k}\Big) + a\hat{p}\mathbf{O}\Big(\frac{\hat{p}}{k}\Big)\\ &= -w\mathbf{o}(1) + (a-1)\mathbf{o}(w) + (w-\hat{p})\mathbf{o}(1) + a\mathbf{o}(w)\\ &= \mathbf{o}(w).\end{aligned}$$

Finally,

$$\begin{aligned}(I) &= w\Big(\log\Big(\frac{w-\hat{p}}{n-k}\Big) - \log\Big(\frac{w}{n}\Big)\Big) + \hat{p}\Big(a\log\Big(\frac{ap}{k}\Big) - \log\Big(\frac{w-\hat{p}}{n-k}\Big)\Big)\\ &= w\log\Big(\frac{w-\hat{p}}{w}/\frac{n-k}{n}\Big) + \hat{p}\log\Big(a^a\frac{\hat{p}^a}{w-\hat{p}}\frac{n-k}{k^a}\Big)\\ &= w\log\Big(\frac{1-\hat{p}/w}{1-k/n}\Big) + a\hat{p}\log\Big(\frac{a\hat{p}}{w}\Big) + \hat{p}\log\Big(\frac{w^a}{(w-\hat{p})^a}\Big) + \hat{p}\log\Big(\frac{n-k}{k^a(w-\hat{p})^{1-a}}\Big)\\ &= w\log\Big(\frac{1-\hat{p}/w}{1-k/n}\Big) + w\Big(\mathbf{o}(1) + a\frac{\hat{p}}{w}\log\Big(\frac{w}{w-\hat{p}}\Big) + \frac{\hat{p}}{w}\log\Big(\frac{(n-k)^{1-a}}{(w-\hat{p})^{1-a}}\Big)\Big)\\ &= w\log\Big(\frac{1-\hat{p}/w}{1-k/n}\Big) + w\Big(\mathbf{o}(1) + (1-a)\frac{\hat{p}}{w}\log\Big(\frac{n-k}{w-\hat{p}}\Big)\Big)\\ &= w\log\Big(\frac{1-\hat{p}/w}{1-k/n}\Big) + w\Big(\mathbf{o}(1) - (1-a)\frac{\hat{p}}{w}\Big(\mathbf{O}(1) + \log\big(\frac{n}{w}\big)\Big)\Big)\\ &= w\log\Big(\frac{1-\hat{p}/w}{1-k/n}\Big) + w\Big(\mathbf{o}(1) + \frac{\hat{p}}{w}\log\Big(\frac{w}{n}^{1-a}\Big)\Big)\end{aligned}$$

So, the lemma 4 implies

$$(I) = w\log\Big(\frac{1-\hat{p}/w}{1-k/n}\Big) + w\Big(\mathbf{o}(1)+\mathbf{o}(1)\Big).$$

Finally, we conclude

$$c_a(n,k,w) \geq (I)+(II) = w\Big(\log\Big(\frac{1}{1-R}\Big) + \mathbf{o}(1)\Big).$$

□