Improved Estimation of Collision Entropy in High and Low-Entropy Regimes and Applications to Anomaly Detection

Maciej Skorski

IST Austria

Abstract. We revisit the problem of estimating Renyi Entropy from samples, focusing on the important case of collision entropy.
With n samples we approximate the collision entropy of X within an additive factor of $O\left(2^{2\Delta}\log^{\frac{1}{2}}(1/\epsilon)\right)$ with probability $1-\epsilon$, where Δ is a known (a priori) upper bound on the difference between Renyi entropies of X of order 2 and 3 respectively.
In particular, we simplify and improve the previous result on estimating collision entropy due to Acharya et al. (SODA'15) up to a factor exponential in the entropy gap.
We also discuss applications of our bound in anomaly analysis, namely (a) detection of attacks against stateless sources in TRNGs, and (b) detection of DDoS attacks at network firewalls.

Keywords: Entropy Estimators, Collision Entropy, Anomaly Detection

1 Introduction

1.1 Collision Entropy and its Applications

Renyi Entropy is a useful measure of randomness which appears in many applications where the use of Shannon Entropy is insufficient. Particularly important is the case of Renyi entropy of order two, called *collision entropy* because it bounds the collision probability of the distribution. Applications include measuring the quality of random number generators [OW99,Knu98], determining the number of unbiased bits extractable from physical sources [BBCM95,IZ89], testing closeness of discrete distributions [BFR+13,CDVV14], testing graph expansion [GR11], complexity of reconstructing DNA sequences [MBT12] and anomaly detections in context of network security [XLZ11] or financial markets [JKS12].

1.2 Estimating Collision Entropy

Because of these applications, it is important to have a method of efficiently estimating collision entropy from available data samples. The problem reads informally as follows

> **Efficient entropy estimating:** Find an algorithm which provides an approximation to the collision entropy of unknown discrete distribution X, based on independent samples $X_1, \ldots, X_n$, with possibly small sample size, space and number of public coins.

Techniques for estimating collision entropy A naive way (which can be applied to any entropy notion) is to approximate the probability mass function based on empirical frequencies which yields the complexity of $O(|\mathcal{X}|)$ samples where $\mathcal{X}$ is the set of outcomes of X (referred to as the alphabet), with memory usage $O(|\mathcal{X}|)$. For collision entropy we can do slightly better, namely estimate the collision probability by counting collisions between consecutive samples. This method, referred to as *collision counting*, needs also $O(|\mathcal{X}|)$ samples but use constant memory [Skó16]. However for applications involving real-time data analysis optimizing running time is more important.

The state of the art regarding fast Renyi entropy estimators has been recently established by the work [AOST15] where it is shown that in particular the complexity of estimating collision entropy is actually only $O(|\mathcal{X}|^{\frac{1}{2}})$ with $O(|\mathcal{X}|)$ memory[1] Interestingly, in [CKOS15] it is further shown that even allowing queries to the probability mass function doesn't help to decrease this complexity.

For completeness, we present the algorithm in pseudocode (see Algorithm 1). Note that the sample size in the algorithm has to be sufficiently big (bounds discussed below) in order to obtain a desired approximation.

The algorithm explanation The algorithm splits the input sequence into blocks, and for each block the collision probability of the source distribution is estimated as in Line 1.7. Next, the median of block estimates is computed in Line 1.8, in order to amplify previous estimates. The result entropy estimate is computed as the negative logarithm of the estimated collision probability in Line 1.9.

[1] The hidden constant grows with approximation accuracy, for the sake of simplicity we omit the exact formula here.

Algorithm 1: Collision Entropy Estimator

Input : source alphabet $\mathcal{X}$
number of independent samples n,
samples $x_1, \ldots, x_n$ from an unknown distribution X on $\mathcal{X}$,
accuracy parameter δ
maximal statistical error ϵ

Internal: an array $[n(x)]_{x \in \mathcal{X}}$ for storing frequencies

Output : A number $\widehat{H}$ which approximates the collision entropy $H_2(X)$

1.1 $N \leftarrow \lceil c \cdot 2^{\frac{1}{2}M} \delta^{-2} \rceil$ `/* Set the block length */`
1.2 $\ell \leftarrow \lfloor n/N \rfloor$ `/* Set the number of blocks */`
1.3 **for** $j = 1, \ldots, n/N$ **do**
1.4 $\quad I \leftarrow [(j-1)N + 1 \ldots (j-1)N]$ `/* get the next block */`
1.5 $\quad$ **foreach** $i \in I$ **do**
1.6 $\quad\quad n(x_{i+1}) = n(x_i) + 1$ `/* update frequencies */`
1.7 $\quad \hat{q}_j \leftarrow \frac{1}{m(m-1)} \left(\sum_{x \in \mathcal{X}} n(x)^2 - m \right)$ `/* block estimate */`
1.8 $\hat{q} \leftarrow$ `FindMedian`$(\hat{q}_1, \ldots, \hat{q}_\ell)$ `// amplify block estimates`
1.9 **return** $-\log \hat{q}$

1.3 Our Contribution

In this paper we provide a simpler analysis of the original algorithm from [AOST15], obtaining better bounds for settings where a non-trivial bound on the entropy is known. Thus, our improvement is twofold:

(a) We present a more elementary analysis of the algorithm, using only the Chebyszev and Chernoff-Hoeffding Inequalities. In particular we get rid of the reduction to Poisson Sampling, used in the original work [AOST15]
(b) We obtain a quantitative improvement for the necessary sample size. This applies to settings where entropy is a-priori bounded (in turn the work [AOST15] is focused on completely unknown distributions and couldn't optimize its result further). Our improvement is up to a factor exponential in the alphabet size.

In order to formulate our result, we need a notion of (δ, ϵ) approximation, which with respect to an approximation algorithm states that it provides an estimation within an additive error δ and fails with probability at most ϵ (see Section 2 for a formal definition).

Theorem 1. *There exists a universal constant c such that Algorithm 1 provides a (δ, ϵ)-approximation to collision entropy with the necessary*

number of samples

$$n = O\left(2^{2(H_2(X)-H_3(X))}\delta^{-2}\log(1/\epsilon)\right)$$

and memory $O(|\mathcal{X}|)$*, where* H_2 *and* H_3 *are Renyi entropies of* X *of order* 2 *and* 3*, respectively.*

This statement is actually much stronger than we need in our applications, which exploit the following corollary

Corollary 1 (Better Complexity with Bounded Entropy). *Suppose that* $H_2(X) \leqslant M$*, then the necessary number of samples is* $n = O\left(2^{\frac{1}{2}M}\delta^{-2}\log(1/\epsilon)\right)$.

Remark 1 (Concluding the previous bound). To see that our result improves [AOST15] note that $H_2(X) \leqslant \log|\mathcal{X}|$ and thus $2^{\frac{1}{2}H_2(X)} \leqslant |\mathcal{X}|^{\frac{1}{2}}$.

A comprehensive comparison of our result with respect to other techniques is given in Table 1.

authors	number of samples	memory	technique
folklore (see [VSH11])	$O(\|\mathcal{X}\|)$	$O(\|\mathcal{X}\|)$	naive plug-in estimator
folklore (see [Skó16])	$O(\|\mathcal{X}\|)$	$O(1)$	collisions counting
[AOST15]	$O(\|\mathcal{X}\|^{\frac{1}{2}})$	$O(\|\mathcal{X}\|)$	Algorithm 1
this work	$O\left(2^{\frac{1}{2}M}\right)$, $H_2(X) \leqslant M$	$O(\|\mathcal{X}\|)$	Algorithm 1

Table 1: Different techniques of estimating collision entropy, for a distribution X over an alphabet $\mathcal{X}$. For simplicity, constant approximation and probability errors are assumed.

1.4 Related works

Acharya at al. [AOST15] The current state of the art with respect to complexity of entropy estimators is summarized in the work [AOST15], where the authors construct estimators for Renyi entropy of arbitrary order and show that they are generally optimal in terms of complexity. Basically, they show that $O\left(|\mathcal{X}|^{1-\frac{1}{\alpha}}\right)$ samples are sufficient for fixed accuracy and error parameters[2]. Their proof involves Poisson Sampling techniques, used to calculate higher moments of the estimator.

[2] Where the hidden constant equals $\delta^{-2}\log(1/\epsilon)$

Online Shannon Entropy Estimators [LPR11] Lauradoux at al., having cryptographic applications in mind, propose an estimator for Shannon Entropy, which operates in constant memory for a given accuracy parameter. Unfortunately Shannon Entropy is much weaker than other Renyi Entropies, and in particular is insufficient for cryptographic applications like evaluating random number generators or extracting randomness.

Relations to streaming algorithms. Since Renyi entropy is defined in terms of moments of a probability distribution (high collision probabilities), the problem of estimating entropy from samples looks related to a well-studied problem of estimating frequency moments in a stream. Unfortunately streaming algorithms estimate the frequency moments of input data whereas in our case input is only a sample of the actual data, which adds an extra *bias*. Moreover, without biased-correction techniques the resulting entropy estimate is negatively-biased [3], which is a serious issue for security applications, where entropy cannot be underestimated.

1.5 Applications

We describe two applications, both in settings when collision entropy needs to be measured in real time on the basis of data samples, and every substantial decrease in entropy needs to be quickly detected and reported. The one is the so called testable design of random number generators, and the second one is discovering DDoS attacks by packet filtering. Details are discussed in Section 4.

1.6 Organization

Necessary notions and auxiliary results are explained and stated in Section 2. The proof of Theorem 1 appears in Section 3. Applications to random number generators and DDoS detection systems are discussed in Section 4. We end with conclusions in Section 5.

2 Preliminaries

In this section we explain necessary notations and conventions. All logarithms are taken at base 2. For any two integers k_1, k_2 by $[k_1, k_2]$ we

[3] If we calculate *exactly* frequencies $\hat{p}(x) = \frac{1}{n}\sum_i \mathbf{1}_{\{X_i=x\}}$, and plug them into the entropy $p_\alpha = \sum_x p(x)^\alpha$ for $\alpha > 1$, we obtain a biased estimator, more precisely $\mathbb{E}\hat{p}_\alpha \geqslant \sum_x p(x)^\alpha$ by the Jensen inequality. For the Renyi entropy of order α defined as $H_\alpha = -\log p_\alpha$ the inequality is reversed, yielding a negative bias.

understand the sett of all integers between k_1 and k_2, endpoints included. Less standard notions are defined below

Definition 1 (Moments). *The alpha-th moment of a non-negative discrete function f is defined as $f_\alpha = \sum_x f(x)^\alpha$.*

Below we define the notion of Renyi entropy for any order $\alpha \geqslant 1$.

Definition 2 (Renyi Entropy). *The Renyi entropy of order α of a discrete probability distribution $p(\cdot)$ is defined as*

$$H_\alpha(p) = \sum_x p(x)^\alpha.$$

For a discrete random variable X we define $H_\alpha(X) = H_\alpha(p_X)$ where p_X is the distribution of X.

We also refer to p_2 as the collision probability of p, and $H_2(p) = -\log p_2$ as the collision entropy of p.

Definition 3 (Entropy Estimator). *Let $\mathcal{X}$ be a fixed finite alphabet. We say that an algorithm A, which receives n symbols on input and outputs a real number, provides a (δ, ϵ)-approximation to collision entropy, if for any random variable X we have*

$$\Pr_{x_1,\ldots,x_n \leftarrow X}[\mathsf{A}(x_1, \ldots, x_n) \geqslant H_2(X) - \delta] \geqslant 1 - \epsilon$$

where samples $x_1, \ldots, x_n$ are drawn from X uniformly and independently.

Lemma 1. *We have $H_3(X) \geqslant \frac{3}{4} H_2(X)$ for any X.*

Proof. By the well-known inequality for l_p-norms we have

$$\left(\sum_x p(x)^2\right)^{\frac{1}{2}} \geqslant \left(\sum_x p(x)^3\right)^{\frac{1}{3}}$$

which rewritten in terms of entropies is precisely what is claimed.

3 Main Result

We first given an overview of the proof and state key lemmas. The proof of lemmas below are given at the end of this section.

We start by showing that the estimates Line 1.7 approximate collision probability p_2 with a relative error δ and the error probability at most $\frac{1}{3}$ with $O\left(p_3 p_2^{-2} \delta^{-2}\right)$ samples.

Lemma 2 (Block Estimator Accuracy). *There exists a constant c_1 such that for $\ell = c_1 \cdot p_3 p_2^{-2} \delta^{-2}$ (defined as in Algorithm 1) The numbers $\hat{q}_j$ in Line 1.7 of Algorithm 1 satisfy*

$$|\hat{q}_j - p_2| \leqslant \delta \cdot p_2$$

with probability $\frac{2}{3}$ over the choice of samples $x_1, \ldots, x_n$.

The previous lemma guarantees that we have good estimates (the error being smaller than δ) for each j-th block where $j = 1, \ldots, N/n$, each time with probability at least $\frac{2}{3}$. Now we show that the median of these estimates amplifies the error probability.

Lemma 3 (Median Amplifies Block Estimators). *There exists a constant c_2 such that if $n/N < c_2 \log(1/\epsilon)$ then the number $\hat{q}$ in Line 1.8 of Algorithm 1 satisfies*

$$|\hat{q} - p_2| \leqslant \delta \cdot p_2$$

with probability $1 - \epsilon$ over the choice of samples $x_1, \ldots, x_n$.

Proof (Proof of Lemma 3). We repeat the argument given in [AOST15]. By the Chernoff Bound, it follows that for $O(\log(1/\epsilon))$ blocks more that one half of the numbers are good with probability $1 - \epsilon$. If so, so their median. Note that the median can be computed in linear time, so it doesn't affect the final complexity.

Having proved these lemmas, we easily conclude the result claimed in Theorem 1. Namely, it follows that the number $\hat{q}$ is an approximation to p_2 with a relative error δ provided that we have $n = c \cdot p_3 p_2^{-2} \delta^{-2} \log(1/\epsilon)$. It remains to observe that a relative error δ in approximating p_2 translates to an additive error $O(\delta)$ in the collision entropy estimate, because the (true) entropy equals $-\log \hat{q} = -\log\left(p(1 + O(\delta))\right) = -\log p + O(\delta)$. Here we use the Taylor formula $\log(1 + O(u)) = O(\delta)$ valid for all sufficiently small δ (for other values of δ it suffices to decrease δ at most by a constant factor (which translates to a constant factor in n and prove the result for a smaller value of δ). This argument gives the bound $n = O\left(\cdot p_3 p_2^{-2} \delta^{-2} \log(1/\epsilon)\right)$ which is the claimed bound if we express moments in terms of entropies $p_3 p_2^{-2} = 2^{2H_2(X) - 2H_3(x)}$ (see Definition 2).

Proof (Proof of Lemma 2). Let $X_1, \ldots, X_n$ be iid over an alphabet $\mathcal{X}$, with the probability mass function $p(x)$. Let $n(x) = \sum_{i=1}^{n} \mathbf{1}_{\{X_i = x\}}$ be the empirical frequency of the element x (according to samples $X_1, \ldots, X_n$),

and let $p_k = \sum_x (p(x))^k$ be the k-th moment of the probability function $p(x)$. Consider the estimator

$$\hat{p} = \frac{\sum_x n(x)^2 - n}{n(n-1)} \tag{1}$$

which corresponds to Line 1.7 in Algorithm 1. The factor $n(n-1)$ in the denominator is just for scaling, whereas the purpose of subtracting the term n is to obtained an *unbiased* estimator. To apply the second moment technique we will need the following fact

Claim (Frequency moments and mixed moments). For every x and every $y \neq x$, we have the following identities

$$\begin{aligned}
\mathbf{E}\left[n(x)^2\right] &= n(n-1)p(x)^2 + np(x) && (2)\\
\mathbf{E}\left[n(x)n(y)\right] &= n(n-1)p(x)p(y) && (3)\\
\mathbf{E}\left[n(x)^4\right] &= n(n-1)(n-2)(n-3)p(x)^4 + \\
&\quad + O(n^3)p(x)^3 + O(n^2)p(x)^2 + np(x) && (4)\\
\mathbf{E}\left[n(x)^2 n(y^2\right] &= n(n-1)(n-2)(n-3)p(x)^2p(y)^2 + n^2(n-1)p(x)^2p(y) + \\
&\quad + n^2(n-1)p(x)p(y)^2 && (5)
\end{aligned}$$

The derivation is by elementary algebraic calculations and is omitted. In particular, the claim implies

$$\begin{aligned}
\mathbb{E}\left(\sum_x n(x)^2\right)^2 &= \sum_x \mathbb{E}n(x)^4 + \sum_{x\neq y} \mathbb{E}n(x)^2 n(y)^2 = \\
&= n(n-1)(n-2)(n-3)p_4 + O(n^3p_3) + O(n^2p_2) + n \\
&\quad + n(n-1)(n-2)(n-3)\sum_x p(x)^2(p_2 - p(x)^2) + \\
&\quad + 2n^2(n-1)(p_2 - p_3) \\
&= n(n-1)(n-2)(n-3)p_2^2 + (2n^3 + O(n^2))p_2 + O(n^3)p_3
\end{aligned}$$

(where both terms with p_4 cancel) and therefore

$$\begin{aligned}
\mathrm{Var}\left(\sum_x n(x)^2\right) &= \sum_{x\neq y} \mathbf{E}\left[n(x)^2 n(y)^2\right] + \sum_x \mathbf{E}\left[n(x)^4\right] - (n(n-1)p_2 + n)^2 \\
&= (n(n-1)p_2)^2 + n^2 - n^2p_2 + O(n^3p_3) + O(n^2p_2) - (n(n-1)p_2 + n)^2 \\
&= O(n^3p_3) && (6)
\end{aligned}$$

From the Chebyszev inequality, we conclude that

$$\Pr\left[|\hat{p} - p_2| > \delta p_2\right] = O\left(n^{-1} p_3 p_2^{-2} \delta^{-2}\right)$$

Note that the relative error δ for p_2 corresponds to an additive error $O(\delta)$ for the entropy. This can be amplified by independent repetitions.

Remark 2. Note that we have $p_3 \leqslant |\mathcal{X}|^{\frac{1}{4}}(p_4)^{\frac{3}{4}} < |\mathcal{X}|^{\frac{1}{4}}(p_2)^{\frac{3}{2}} \leqslant |\mathcal{X}|^{\frac{1}{2}}(p_2)^2$ and thus

4 Applications to Low Entropy Regimes

4.1 Low Entropy Detection for True Random Number Generators

Bucci and Luzzi in [BL05] introduced the concept of *testable random bit generators*, where the entropy source X is assumed to be stateless (that is, consecutive outputs $X_1, X_2, \ldots$ are independent and identically distributed) and is coupled with an *online entropy estimator* which is used to detect changes in the entropy rate. Estimating entropy in the source in real time allows for adjusting the postprocessing algorithm (which needs more raw data in case of a decrease in randomness quality) or take other actions for significant entropy decreases (consider this an adversarial attack and rise an alarm [BL05]). Note that there are real-world sources that are stateless or can be modified to be stateless [BL05, BL07].

Define the entropy rate as $r = \frac{H_2(X)}{\log|\mathcal{X}|}$ where $\mathcal{X}$ is the set of possible outcomes of the source X. The initial value for this number can be evaluated during laboratory tests (as recommended by standards [TBK+]).

When a device operates in a production environment, the estimator is being run in background and applied to consecutive source outputs $X_1, X_2, \ldots X_n$ to see if the entropy per sample doesn't fall below r.

Our bound implies that the time for discovering that entropy decreases by a constant amount equals

$$T_{\text{detect}} = O\left(|\mathcal{X}|^r\right)$$

whereas previously known bounds imply $O\left(|\mathcal{X}|^{\frac{1}{2}}\right)$. Thus our bound is better whenever $r < \frac{1}{2}$. In fact, entropy rates for real-world sources are often below $\frac{1}{2}$ [LPR11, VSH11]. For the sake of the completeness we also mention that the use of collision entropy here is justified by the fact that postprocessing functions used in practice are based on universal hashing [BST03, VSH11] and the right measure of randomness in that case is precisely collision entropy.

4.2 Faster and More Reliable DDoS Detection

In [LZYD09, XLZ11] it was shown that the collision entropy (and more generally: Renyi entropy) of the distribution of source IP addresses can be used to detect a DDoS attack at a packet filter (more specifically: IP spoofing attacks). In this scenario an attacker manages a pool of spoof IP addresses and sends a flow of TCP SYN packets against a victim host. There is a "pick" in the distribution during an attack, due to heavy packet traffic originated from a fraction of all addresses owned by the adversary; this pick can be seen as the difference between the entropy of addresses during the flooding attack[4] and the "normal" flows.

Note that while the attack-free traffic is observed over a long time period and well-approximated (typically computed off-line), an attack traffic needs to be discovered quickly in real time. With our bound we can detect[5] a δ-decrease in the entropy at confidence $1-\epsilon$ in time

$$T \approx 2^{\frac{1}{2}M}\delta^{-2}\log(1/\epsilon),$$

where M is the entropy of the normal traffic. The previous bound implies only

$$T \approx 2^{\frac{1}{2}M_{\max}}\delta^{-2}\log(1/\epsilon),$$

where $M_{\max}$ is the entropy assuming uniform distribution of all IP addresses, which is substantially bigger than M.

For sample empirical data used in [LZYD09, XLZ11] and [XLZ11] we save a factor of 4 comparing to the bound [AOST15], which means that we can detect DDoS attacks 4 times faster. Alternatively, within the same sample size we get accuracy better by a factor of $4^2 = 16$ or an error probability better by a factor $2^4 = 16$ which means a more reliable detection scheme (less false alarms).

5 Conclusion

In this section we improve the complexity of estimating collision entropy, provided that an upper bound is known a priori. We show same examples of real world applications where this assumption is fulfilled and our bound can be used to obtain quantitative improvements.

[4] In automatic DDoS detection systems the entropy is being updated periodically in a sliding time-window, so that every sufficiently long attack can be caught.

[5] An assumption is made that the entropy only decreases from the default bound, which is realistic for DDoS scenarios.

References

AOST15. Jayadev Acharya, Alon Orlitsky, Ananda Theertha Suresh, and Himanshu Tyagi, *The complexity of estimating Rényi entropy*, Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms (Philadelphia, PA, USA), SODA '15, Society for Industrial and Applied Mathematics, 2015, pp. 1855–1869.

BBCM95. C. H. Bennett, G. Brassard, C. Crepeau, and U. M. Maurer, *Generalized privacy amplification*, IEEE Transactions on Information Theory **41** (1995), no. 6, 1915–1923.

BFR+13. Tuğkan Batu, Lance Fortnow, Ronitt Rubinfeld, Warren D. Smith, and Patrick White, *Testing closeness of discrete distributions*, J. ACM **60** (2013), no. 1, 4:1–4:25.

BL05. Marco Bucci and Raimondo Luzzi, *Design of testable random bit generators*, Cryptographic Hardware and Embedded Systems - CHES 2005, 7th International Workshop, Edinburgh, UK, August 29 - September 1, 2005, Proceedings, 2005, pp. 147–156.

BL07. ———, *A testable random bit generator based on a high resolution phase noise detection*, Proceedings of the 10th IEEE Workshop on Design & Diagnostics of Electronic Circuits & Systems (DDECS 2007), Kraków, Poland, April 11-13, 2007, 2007, pp. 23–28.

BST03. Boaz Barak, Ronen Shaltiel, and Eran Tromer, *True random number generators secure in a changing environment*, In Workshop on Cryptographic Hardware and Embedded Systems (CHES, Springer-Verlag, 2003, pp. 166–180.

CDVV14. Siu-On Chan, Ilias Diakonikolas, Gregory Valiant, and Paul Valiant, *Optimal algorithms for testing closeness of discrete distributions*, Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms (Philadelphia, PA, USA), SODA '14, Society for Industrial and Applied Mathematics, 2014, pp. 1193–1203.

CKOS15. Cafer Caferov, Baris Kaya, Ryan O'Donnell, and A. C. Cem Say, *Optimal bounds for estimating entropy with PMF queries*, Mathematical Foundations of Computer Science 2015 - 40th International Symposium, MFCS 2015, Milan, Italy, August 24-28, 2015, Proceedings, Part II, 2015, pp. 187–198.

GR11. Oded Goldreich and Dana Ron, *Studies in complexity and cryptography*, Springer-Verlag, Berlin, Heidelberg, 2011, pp. 68–75.

IZ89. R. Impagliazzo and D. Zuckerman, *How to recycle random bits*, Proceedings of the 30th Annual Symposium on Foundations of Computer Science (Washington, DC, USA), SFCS '89, IEEE Computer Society, 1989, pp. 248–253.

JKS12. Petr Jizba, Hagen Kleinert, and Mohammad Shefaat, *Rényi's information transfer between financial time series*, Physica A: Statistical Mechanics and its Applications **391** (2012), no. 10, 2971 – 2989.

Knu98. Donald E. Knuth, *The art of computer programming, volume 3: (2nd ed.) sorting and searching*, Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA, 1998.

LPR11. Cédric Lauradoux, Julien Ponge, and Andrea Roeck, *Online Entropy Estimation for Non-Binary Sources and Applications on iPhone*, Research Report RR-7663, INRIA, June 2011.

LZYD09. Ke Li, Wanlei Zhou, Shui Yu, and Bo Dai, *Effective ddos attacks detection using generalized entropy metric*, Proceedings of the 9th International Conference on Algorithms and Architectures for Parallel Processing (Berlin, Heidelberg), ICA3PP '09, Springer-Verlag, 2009, pp. 266–280.

MBT12. Abolfazl S. Motahari, Guy Bresler, and David Tse, *Information theory of DNA sequencing*, CoRR **abs/1203.6233** (2012).

OW99. Paul C. Oorschot and Michael J. Wiener, *Parallel collision search with cryptanalytic applications*, J. Cryptol. **12** (1999), no. 1, 1–28.

Skó16. Maciej Skórski, *Evaluating entropy sources for true random number generators by collision counting*, Applications and Techniques in Information Security - 6th International Conference, ATIS 2016, Cairns, QLD, Australia, October 26-28, 2016, Proceedings, 2016, pp. 69–80.

TBK$^+$. Meltem Sönmez Turan, Elaine Barker, John Kelsey, Kerry A. McKay, Baish Mary L., and Mike Boyle.

VSH11. Jonathan Voris, Nitesh Saxena, and Tzipora Halevi, *Accelerometers and randomness: Perfect together*, Proceedings of the Fourth ACM Conference on Wireless Network Security (New York, NY, USA), WiSec '11, ACM, 2011, pp. 115–126.

XLZ11. Y. Xiang, K. Li, and W. Zhou, *Low-rate ddos attacks detection and traceback by using new information metrics*, IEEE Transactions on Information Forensics and Security **6** (2011), no. 2, 426–437.