Probabilistic Methods in Cryptology: Entropy, Tail Bounds, and Operational Randomness Testing

Mohamed Humaid Saleem
Harvard College '26

We study three probabilistic tools that underpin modern cryptology: (i) entropy (and min entropy) to quantify unpredictability and compressibility, (ii) Chernoff style concentration to detect non random leakage, and (iii) empirical batteries that operationalize indistinguishability. Rather than re proving standard inequalities, we state them precisely and apply them to a concrete detection problem, then situate the analysis with a short historical case study of Andrew Gleason in WWII.

Introduction

How does probability intersect cryptology?

The probabilistic techniques in this paper are crucial to many cryptographic systems, and we shall illustrate a few such applications. Consider entropy, as discussed in section 2. One of the most impressive applications of entropy within cryptography is its role in generating keys for the Advanced Encryption Standard (AES) which is the symmetric-key block cipher used by the U.S. National Institute of Standards and Technology's (NIST). AES operates on 128 bit blocks and it allows for key lengths of 128,192, and 256 bits. This allows for seeds with relatively high levels of entropy, that allow them to resist brute force attacks in protocols that govern internet communication such as TLS (Transport Layer Security). For example, 256 bit keys with $H(K) = \log(2^{256}) \approx 177.4$ nats, are ideal. Now consider Chernoff bounds, as discussed in section 4. Chernoff bounds allow for offensive cryptographic ability. For example, they are used in side channel attacks. Such attacks use Chernoff bounds to detect non random power consumption patterns in people's smartcards, and subsequently reveal key bits with high levels of confidence. They do this by measuring deviations in power consumption or the timing required to infer key bits, thus allowing negligible failure probabilities in cryptographic protocols. By giving us methods and techniques to measure randomness, probability theory has massively furthered the field of cryptology.

Historical overview

Shannon's work during the world wars, positioned data secrecy on a probabilistic basis, by proving a system achieves perfect secrecy precisely when its key has at minimum the entropy of the message (Shannon, 1949). The time period of public keys then attempted to align security with computational feasibility, Diffie–Hellman and RSA defined efficient primitives (Diffie and Hellman, 1976; Rivest et al., 1978), while the Miller–Rabin test replaced deterministic primality proofs by probabilistic verification with tunable one sided error 4^t after t bases (Miller and Rabin, 1980). Probabilistic encryption (Goldwasser–Micali) formalized how semantic security was simply making sure there a negligible probability of adverseries deciphering your messages, by using randomness (Bressoud and Wagon, 2000; Shannon, 1948), and the historical precendent was created when case work such as the Gleason CORAL story illustrated how very marginal statistical biases can be decisive in encryption battles (Conway et al., 2008).

As to who first used probability theory in cryptology: while statistical counting predates modern probability, the first formal, explicitly probabilistic foundation is widely credited to Shannon's 1949 treatment of secrecy via entropy and independence (Shannon, 1949), with wartime traffic analysis (e.g., Gleason's CORAL work) providing earlier operational use of probabilistic tallies (Conway et al., 2008). Beyond secrecy theory, probability drives primality testing. Pseudo random number generators guide Pollard's ρ random walks (1975) and Lenstra's elliptic curve method on random order groups (1987) (Pollard, 1975; Lenstra, 1987). The birthday paradox explains why collisions emerge after $\Theta(\sqrt{N})$ samples and is routinely invoked to justify collision based heuristics and hash security margins (Bellare and Rogaway, 2005; Riesel, 1994; Bressoud and Wagon, 2000).

Remark 1.1. There will be cryptological jargon scattered through this paper, for ease of reading, some of them are summarized here.

Cryptanalysis is the science of breaking codes and deciphering encrypted messages without knowing the secret key.
Encryption is the process by which a readable message is converted to an unreadable form to prevent unauthorized parties from reading it. Decryption is the process of converting an encrypted message back to its original (readable) format.
Cryptographic keys are classified into two main types: symmetric and asymmetric. Symmetric keys use a single key for both encryption and decryption, while asymmetric keys use two mathematically related keys: a public key for encryption and a private key for decryption.
Ciphertext, is the result of applying a cipher (an encryption algorithm) to plaintext to make it unreadable without the proper decryption method. It's the scrambled, unreadable form of the original message.

Entropy, Compressibility, and Information Theoretic Security

We work with discrete random variables on finite sets, which suffices for the cryptographic settings considered here (finite messages, keys, and ciphertexts). Let X be discrete with probability mass function p. The

(natural log) Shannon entropy is

$$H(X) = \sum_x p(x) \log p(x).$$

Compression viewpoint. On finite alphabets, source coding implies that the average optimal lossless code length of X equals $H(X)$ (in the same units). Thus, higher entropy coincides with lower compressibility, which is a practical heuristic for "randomness" in empirical checks (Shannon, 1948, 1949).

Min entropy. For key material, we also track

$$H_\infty(X) = \log \max_x \mathbb{P}[X = x],$$

which measures the probability of the most likely outcome. Intuitively: if the best adversary always guesses the most likely value, then her success probability is $\max_x \mathbb{P}[X = x]$, so the "bits of unpredictability" available in a single draw is H_∞. One always has $H_\infty(X) \le H(X)$, with equality iff X is uniform.

Example 2.1 (Uniform vs. biased key). Let K be uniform on $\{0, 1\}^4$, so $H(K) = 4 \log 2$ (four bits). If instead $\mathbb{P}b[K = 0000] = ½$ and the other 15 strings share the remaining ½ uniformly, then

$$H(K) = ½ \log 2 + ½ \log 30 \approx 1.214 \text{ nats } (\approx 1.75 \text{ bits})$$

and $H_\infty(K) = \log 2$ (one bit). This key is dramatically more compressible and easier to guess than uniform.

Perfect secrecy (Shannon). An encryption scheme with message M, key K, and ciphertext C has perfect secrecy if M and C are independent: $\mathbb{P}[M = m \mid C = c] = \mathbb{P}[M = m]$ for all m, c. Shannon showed that any correct scheme with perfect secrecy must satisfy $H(M \mid C) = H(M)$ and therefore $H(K) \ge H(M)$; the one time pad with $C = M \oplus K$ and $|K| = |M|$ meets this bound with equality (Shannon, 1949).

Remark 2.2 (Operational implication). Perfect secrecy is brittle to key reuse: even with $|K| = |M|$, using the same K twice ($C_1 = M_1 \oplus K$, $C_2 = M_2 \oplus K$) yields $C_1 \oplus C_2 = M_1 \oplus M_2$, exposing structure between plaintexts (e.g., repetitions), which is classically exploited in traffic analysis (Shannon, 1949, Conway et al., 2008).

Example 2.3 Let M, K, C be independent random variables over $\{0, 1\}^n$ with K uniform and $C = M \oplus K$. For any fixed m, $c \in \{0, 1\}^n$ there is exactly one key $k = c \oplus m$ such that $M \oplus K = c$. Thus

$$\mathbb{P}[M = m \mid C = c] = \frac{\mathbb{P}[M=m, K=c\oplus m]}{\sum_{m'} \mathbb{P}[M=m', K=c\oplus m']} = \frac{\mathbb{P}[M=m]\cdot 2^n}{\sum_{m'} \mathbb{P}[M=m']\cdot 2^n} = \mathbb{P}[M = m],$$

so $M \perp C$: learning C changes nothing about the distribution of M (Shannon, 1949).

Pseudorandom Generators: Computational and Operational Notions

Definition 3.1 (A function $\text{negl} : \mathbb{N} \to \mathbb{R}_{\ge 0}$ is called negligible if for every polynomial p there exists N such that for all $n \ge N$, $\text{negl}(n) < 1/p(n)$. This formalizes "too small to matter" in asymptotic cryptography.

Definition 3.2 (Secure PRNG; indistinguishability). A generator $G : \{0, 1\}^s \to \{0, 1\}^n$ is secure if for every probabilistic polynomial time (PPT) distinguisher D,

$$\left|\mathbb{P}[D(G(U_s)) = 1] \mathbb{P}[D(U_n)) = 1]\right| \le \text{negl}(s),$$

where U_t is uniform on $\{0, 1\}^t$. Equivalently, in the left–right game, D receives either $G(U_s)$ or U_n and must guess which; security means its advantage over random guessing is negligible. See (Goldwasser and Micali, 1984, Bellare and Rogaway, 2005) for this indistinguishability viewpoint.

PRNGs as deterministic dynamics. Pseudo random number generators are deterministic dynamical systems on a finite state space: each new state is $x_{i+1} = T(x_i)$ for some map T, often inspired by a chaotic transform and then reduced modulo an integer (e.g., $T(x) = x^2 + c \bmod n$). In finite state every trajectory is ultimately periodic, so "good" means, in practice, long periods, nearly uniform k tuple statistics, and crucially no efficiently exploitable prediction rules; what counts as "good" is therefore state of the art and must be re validated as attacks improve (Rukhin et al., 2010; Brown et al., 2006). Classic linear generators were used for decades before their linear predictability was systematically exploited, and weak generators have undermined RSA key generation in the wild hence the dual reliance on empirical test batteries (Rukhin et al., 2010; Brown et al., 2006) and number theoretic constructions such as BBS, which offer provable hardness guarantees for certain output bits under factoring assumptions (Bressoud and Wagon, 2000; Riesel, 1994).

Example 3.3. Consider the quadratic congruential iteration $x_{i+1} = x_i^2 + c \bmod n$ and let $b_i = x_i \bmod 2$ be the least significant bit (LSB) we might naively output. Because $x_i^2 \equiv x_i \pmod 2$, the parity evolves by $b_{i+1} \equiv b_i + (c \bmod 2) \pmod 2$. If c is odd, the LSB deterministically alternates 0, 1, 0, 1,...; if c is even, the LSB is constant. Either pattern immediately fails basic frequency/runs tests and is trivially distinguishable from uniform (Rukhin et al., 2010). This illustrates that discretized "chaotic" maps can harbor simple artifacts unless parameters and outputs are chosen with care.

Chernoff Bounds and a Detection Case Study

Let $X = \sum_{i=1}^n X_i$ where $X_i \in \{0, 1\}$ are independent with $\mathbb{E}[X] = \mu$.

Theorem 4.1 (Multiplicative Chernoff). For $\delta > 0$,

$$\mathbb{P}\left[X \ge (1+\delta)\mu\right] \le \exp\left(\frac{\delta^2}{2+\delta}\mu\right),$$
$$\mathbb{P}\left[X \le (1-\delta)\mu\right] \le \exp\left(\frac{\delta^2}{2}\mu\right) \ (0 < \delta < 1).$$

Idea of Proof: Apply Markov's inequality to $e^{\lambda X}$, optimize over λ, and use independence to factor $\mathbb{E}[e^{\lambda X}] = \prod_i \mathbb{E}[e^{\lambda X_i}]$; see standard notes such as (Goemans, 2006). The key point is that deviations of size $\delta\mu$ have probability that decays exponentially in μ.

Bias test Let $B_i \in \{0, 1\}$ be independent but not necessarily identically distributed, and write $p_i = \mathbb{E}[B_i]$. Set $S = \sum_{i=1}^n B_i$. We test $H_0 : \frac{1}{n}\sum_{i=1}^n p_i = \frac{1}{2}$ against $H1 : \frac{1}{n}\sum_{i=1}^n p_i = \frac{1}{2} + \varepsilon$ for some $\varepsilon > 0$. By Hoeffding's inequality for bounded, independent summands,

$$\mathbb{P}_{H_0}\left(\left|S - \tfrac{n}{2}\right| \ge t\right) \le 2\exp\left(-\frac{2t^2}{n}\right).$$

Choosing $t_\alpha = \sqrt{\frac{n}{2}\ln\left(\frac{2}{\alpha}\right)}$ gives a two-sided level-α test: reject H_0 if $\left|S - \frac{n}{2}\right| \ge t_\alpha$. Under H_1 with average bias $\varepsilon = \frac{1}{n}\sum_i (p_i - \frac{1}{2})$, the error obeys $\mathbb{P}_{H_1}(\text{miss}) \le \exp(-2\varepsilon^2 n)$, so $n = \Theta(\varepsilon - 2 \log(1/\alpha))$ samples suffice to make both errors small.

Example 4.2. We can proceed to work out an example to illustrate how Chernoff bounds detect deviations from expected random behavior. Suppose we have a cipher with the standard 26 letter

alphabet and we want to send a random ciphertext of length $n = 1000$. Then each letter should probabilistically have $1000/26 \approx 38.46$ occurences due to randomness. Let $X_i = 1$ if the i-th letter is 'A', else $X_i = 0$, with $p_i = 1/26$ for all i. Then $X = \sum X_i$ counts 'A's, with $\mu = \sum p_i = 1000/26 \approx 38.46$. Suppose we want to know whether the occurence of 60 or more 'A's is a significant deviation from the expected randomness of the ciphertext. We can test this using the upper tail bound. $t\ (1 + \delta)\mu = 60$, so $\delta = (60/38.46)1 \approx 0.56$. Then the probability of at least 60 'A's is,

$$\mathrm{P}[\mathrm{X} \geq 60] \leq \mathrm{e}^{\left(\frac{0{\cdot}56^2{\cdot}38{\cdot}46}{2+0{\cdot}56}\right)} \approx \mathrm{e}^{\left(\frac{0{\cdot}3136{\cdot}38{\cdot}46}{2{\cdot}56}\right)} \approx \mathrm{e}^{(4.71)} \approx 0.009.$$

This low probability indicates that observing 60 'A's is highly unlikely under random ciphertext distribution, suggesting the cipher produces biased outputs. Such non randomness reveals exploitable patterns in the cipher's structure, enabling an adversary to use cryptanalysis via frequency analysis (Section 6) or other statistical methods to decipher our message.

Case study (Gleason and CORAL): tail bounds as a triage tool. Consider a teleprinter stream over an alphabet Σ of size q (e.g., $q = 32$), where an additive stream cipher mixes a periodic key with period P into plaintext. For a candidate lag d, define the repetition statistic

$$\mathrm{R_d} \;=\; \sum_{\mathrm{i=1}}^{\mathrm{n-d}} \mathrm{Ind}(\mathrm{C_i} = \mathrm{C_{i+d}}),$$

where '*Ind*' in the summation above represents the indicator function, where each term contributes 1 when the pair matches and 0 when it doesn't. Summing those 0/1 values gives a count of matches at lag d.

Under the null model H_0 (ciphertext symbols independent and uniform on Σ), the indicators are i.i.d. Bernoulli($1/q$), hence $\mu_d = \mathbb{E}[R_d] = (n - d)/q$ and $R_d = \sum_{i=1}^{n-d} Y_i$ with $Y_i \sim \mathrm{Bern}(1/q)$. By the multiplicative Chernoff bound (Thm. 4.1), for any $\delta > 0$,

$$\mathbb{P}\Big[\mathrm{R_d} \geq (1+\delta)\mu_\mathrm{d}\Big] \;\leq\; \exp\Big(\tfrac{\delta^2}{2+\delta}\,\mu_\mathrm{d}\Big).$$

If d equals (or shares a factor with) the key period P, then ciphertext symbols line up with the same key symbols more often, pushing R_d above its null mean. Gleason's practical move was to scan many lags and treat unusually large R_d as evidence of periodic structure rather than chance (Conway et al., 2008).

Suppose we scan $d = 1,..., L$ lags and want a global false alarm rate α. By Bonferroni, it suffices to set a per lag tail bound α/L. For small/medium deviations one may use the simpler upper tail $\mathbb{P}[\mathrm{R_d} \geq (1+\delta)\mu_\mathrm{d}] \lesssim \exp(\tfrac{\delta^2}{2}\mu_\mathrm{d})$, which yields the closed form design rule

$$\delta_\mathrm{d} \;\approx\; \sqrt{\frac{2\,\ln(\mathrm{L}/\alpha)}{\mu_\mathrm{d}}} \text{ and } \tau_d = (1 + \delta_d)\,\mu_d.$$

A flag at lag d occurs when $R_d \geq \tau_d$. Numerical example: with $q = 32$, $n = 105$, $L = 200$, and $\alpha = 106$, we have $\mu_d \approx 3125$ and thus $\delta_\mathrm{d} \approx \sqrt{\frac{2\ln(2\times10^8)}{3125}} \approx 0.067$, so $\tau_d \approx 3125 \cdot 1.067 \approx 3335$. Any R_d above 3335 is exponentially unlikely under H_0 but plausible if d divides P.

What happens after a flag. A spike in R_d at several harmonically related lags suggests a candidate period $\hat{\mathrm{P}}$ (the greatest common divisor of flagged lags). Analysts then destripe the traffic modulo $\hat{\mathrm{P}}$ examining each residue class i mod $\hat{\mathrm{P}}$ separately to look for symbol biases/periodicities that persist when the (unknown) key symbol is constant. These steps do not "break" the cipher alone, but they convert a vague suspicion into a quantified improbability (exponentially small under H_0), focusing the subsequent search on key wheel lengths and alignments (Conway et al., 2008; Goemans, 2006).

Analysts for the US Navy during WWII, were tasked with sifting through daily teleprinter traffic by running fast counts such as R_d across multiple lags to identify where there might be structure. They utilized exponentially small Chernoff tails to ensure the shop wide false alarm rate was under control when traversing thousands of d values (Conway et al., 2008; Goemans, 2006). Once a few lags were fired (oftentimes at harmonics), they came to the hypothesis that candidate wheel lengths for a periodic key, destriped the traffic modulo, and re counted symbol frequencies within each residue class. They found the persistent biases there suggested that the same key element was mixing with plaintext in that class.

Andrew Gleason utilized a variant of the Chernoff bound (a similar tail-bound, albeit pre-Chernoff), in order to break a highly sophisticated cipher known as the Japanese Coral cipher. He famously dubbed it the "Gleason Crutch". The Gleason Crutch analyzed ciphertext patterns to show that observed repetition within the Coral cipher was unlikely under randomness. This lead to the revelation of the specific mechanics of the Coral cipher and thus led to its decryption in 1944. In the modern age, Chernoff bounds are applied in side-channel attacks, by analyzing deviations in power consumption or timing to infer key bits, ensuring negligible failure probabilities in cryptographic protocols.

Statistical Tests for Randomness

Statistical tests validate randomness in PRNGs or ciphers, detecting biases that could enable attacks. We shall define 2 such statisitcal tests. Note that within this section, we shall not discuss in detail, the specific applications of such tests. This is because their primary function is to test for randomness, thus it suffices to prove that these tests provide a sufficient metric to evaluate randomness.

Definition 5.1 (Frequency Test). For a binary sequence $S = s_1,..., s_n$ where $s_i \in \{0, 1\}$, let $X = \sum_{i=1} s_i$ be the number of 1s. Under the null hypothesis of randomness, each s_i is 0 or 1 with probability 1/2, so $X \sim \mathrm{Binomial}(n, 1/2)$. Define O_k to be the observed frequency of some outcome $k \in \{0, 1\}$, with $O_1 = X$ (number of 1s) and $O_0 = nX$ (number of 0s). The expected frequency E_k is $E_0 = E_1 = n/2$, as 0s and 1s are equally likely. The chi square statistic measures deviation from expected randomness:

$$\chi^2 = \sum_{\mathrm{k=0,1}} \frac{(\mathrm{O_k E_k})^2}{\mathrm{E_k}}.$$

By construction, the statistic above follows a chi square distribution that has a single degree of freedom. The p value, which can be defined as the probability of observing a χ^2 value that is at least as extreme under the null hypothesis, is then calculated using the chi square distribution's cumulative distribution function. A p value that is below an agreed upon threshold, usually $p = 0.01$ means that we can reject the null hypothesis and conclude that there is non randomness.

Definition 5.2 (Autocorrelation Test). For a binary sequence $S = s_1,..., s_n$ where $s_i \in \{0, 1\}$, define a shift d (where $1 \leq d < n$) to be an operation that offsets the sequence by d positions, thus allowing us to compare s_i with s_{i+d}. The autocorrelation coefficient for shift d is defined to be:

$$\mathrm{A(d)} = \sum_{\mathrm{i=1}}^{\mathrm{n-d}} (\mathrm{s_i} \oplus \mathrm{s_{i+d}}),$$

where $\oplus$ is defined as the XOR operation. Then $s_i \oplus s_{i+d} = 1$ if $s_i \neq s_{i+d}$ and 0 otherwise. From this we infer that $A(d)$ allows us to calculate the number of positions in which the original binary sequence and the shifted sequence differ. Under our null hypothesis (of randomness), each computation that yields $s_i \oplus s_{i+d} = 1$ has a probability of 1/2, thus we can say that $A(d) \sim \text{Binomial}(n - d, 1/2)$, with expected value $\frac{(n-d)}{2}$ and variance $\frac{(n-d)}{4}$. We then define the test statistic ζ to be as follows:

$$\zeta = \frac{|\mathrm{A(d)} - (\mathrm{n} - \mathrm{d})/2|}{\sqrt{(\mathrm{n} - \mathrm{d})/4}}.$$

Observe that ζ measures the deviation of $A(d)$ from its expected value (normalized by the standard deviation). For sufficiently large nd, this approximates to a standard normal distribution. Then if ζ exceeds a threshold under such a distribution, we know that there exists a significant correlation between bits separated by d, and this indicates non randomness.

Such tests are necessary to test for randomness alongside techniques such as the Chernoff bounds. They are quite handy when deciding if a given ciphertext is likely to leak patterns in plaintext. It is precisely such tests that allows us to determine to what degree an encryption scheme is deterministic (Goemans, 2006; Miller and Rabin, 1980).

Conclusion

Probability is the essence of cryptology as it has paved the way for the quantification of unpredictability, the formalization of indistinguishability of permutations of elements in a data set, and it morphs leakage into testable hypotheses via concentration. This paper does not begin to cover the range of cryptological probabilistic techniques at our disposal, but our examples: entropy/min-entropy tradeoffs, Chernoff-style bias detection, and simple empirical batteries, show how sample complexity governs when non-random effects become statistically undeniable. In the future, post-quantum primitives with explicit failure bounds, modern randomness generation that composes measured entropy with DRBGs, and public randomness beacons are predicted to rapidly advance cryptological defenses as well as offensive capabilities rapidly within the next decade. Although much remains uncertain, we can be sure of the fact that probability will always be the cornerstone of cryptographic encryption.

Acknowledgements

Sir Sarath Nanayakkara was my high-school mathematics mentor, and if I were to list all that he has done for me, this paper would be infinitely longer. Thank you for everything Sir, rest in peace.

References

Bellare, M. and Rogaway, P. (2005). Introduction to modern cryptography. Lecture notes. Includes the birthday bound and collision analysis.

Bressoud, D. and Wagon, S. (2000). *A Course in Computational Number Theory*. Key College Publishing, Emeryville, CA.

Brown, R. G., Eddelbuettel, D., and Bauer, D. (2006). *Dieharder: A Random Number Test Suite*. Duke University Department of Physics. Software package for empirical randomness testing; successor to Marsaglia's Diehard suite.

Conway, J. H., Curtis, C. W., Norton, S. P., Parker, R. A., and Wilson, R. A. (2008). The secret life of andrew gleason. *Notices of the American Mathematical Society*, 55(2):230–245.

Diffie, W. and Hellman, M. E. (1976). New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654.

Goemans, M. X. (2006). Chernoff bounds. Lecture notes. Standard reference for multiplicative Chernoff bounds.

Goldwasser, S. and Micali, S. (1984). Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270–299.

Lenstra, H. W. (1987). Factoring integers with elliptic curves. *Annals of Mathematics*, 126(3):649–673.

Miller, G. L. and Rabin, M. O. (1980). Primality by randomized algorithms. Combined reference: Miller (1976) and Rabin (1980). Miller: J. Comput. Syst. Sci. 13(3): 300–317 (1976).

Pollard, J. M. (1975). A monte carlo method for factorization. *BIT Numerical Mathematics*, 15(3):331–334.

Riesel, H. (1994). *Prime Numbers and Computer Methods for Factorization*. Birkhäuser, Boston, 2 edition.

Rivest, R. L., Shamir, A., and Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126.

Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, A., Heckert, N., Dray, J., and Vo, S. (2010). A statistical test suite for random and pseudorandom number generators for cryptographic applications. Technical Report NIST Special Publication 800-22 Rev. 1a, National Institute of Standards and Technology, Gaithersburg, MD.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27(3):379–423. Part I; Part II appears in vol. 27(4): 623–656.

Shannon, C. E. (1949). Communication theory of secrecy systems. *Bell System Technical Journal*, 28(4):656–715.