

Lecture 18: The Road to Capacity

ENGR 76 lecture notes — May 28, 2026

David Tse, Stanford University

In the last lecture, we used the Hamming bound to understand the best possible tradeoff between code rate and error correction capability for binary codes of fixed block length. For block length $n = 7$, we were able to completely determine the Pareto-optimal tradeoff. The main idea was geometric: codewords are points in the Hamming cube $\{0, 1\}^n$, and to correct t errors the Hamming balls of radius t around the codewords must be disjoint.

Today we continue the same story, but with a major shift in viewpoint. Hamming's formulation is a worst-case formulation of noise: the decoder must succeed no matter which t coded bits are flipped. This formulation gives a clean geometric picture in terms of packing Hamming balls, but it also makes the search for optimal codes extremely hard. For $n = 7$ and $n = 8$ we can determine the Pareto-optimal tradeoff, but already at $n = 16$ there are gaps between the best known lower and upper bounds.

Shannon came to the rescue by asking a different question. Instead of requiring the code to correct every possible error pattern, he modeled the channel noise probabilistically and allowed a small target probability of decoding error. This changes the problem from exact finite-dimensional sphere packing to an asymptotic probabilistic question. The remarkable result is that, in this probabilistic model, one can achieve rates approaching the Hamming bound for large block lengths n . Moreover, asymptotically the Hamming bound has a simple expression:

$$1 - H(p),$$

where H is the binary entropy function. This quantity is the capacity of the binary symmetric channel.



Figure 1: Hamming's viewpoint treats noise as worst-case; Shannon's viewpoint treats noise as probabilistic.

1 Recap: the finite-blocklength problem

Recall the notation from the last lecture. For a binary code of length n , let

- M be the number of codewords,
- $r = \frac{\log_2 M}{n}$ be the code rate,
- $d_{\min}$ be the minimum Hamming distance,
- $t = \left\lfloor \frac{d_{\min}-1}{2} \right\rfloor$ be the number of worst-case bit flips the code can correct.

We plot the pair

$$\left(\frac{t}{n}, r \right),$$

i.e., (error tolerance rate, code rate). Since t is an integer, the tradeoff is naturally a step function rather than a smooth curve.

For $n = 7$, the Pareto-optimal points are achieved by the uncoded scheme, the Hamming (7, 4) code, and the repetition code. The step function in Figure 2 uses the convention that once we ask for a positive error tolerance rate, the rate immediately drops from the uncoded value.

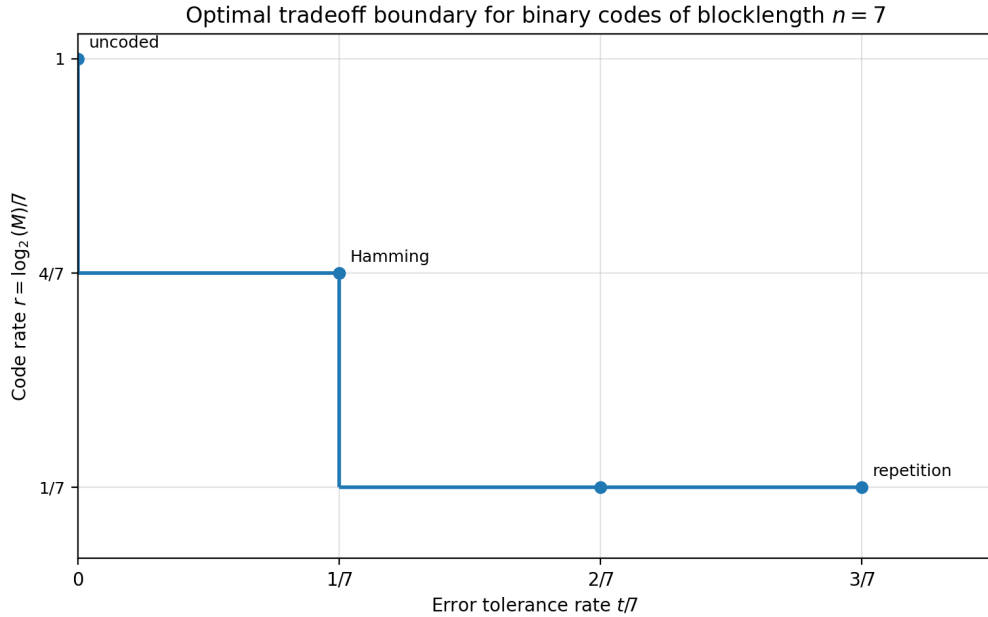


Figure 2: The optimal tradeoff boundary for binary codes of block length $n = 7$.

When we add the Hamming bound, we see that the Hamming code and the repetition code both meet the bound exactly at their respective correction levels. In this sense, both are perfect codes.

For $n = 8$, the Pareto-optimal points are also known, but already the Hamming bound is no longer always tight. The best one-error-correcting code has $M = 20$ codewords, giving rate $\log_2(20)/8 \approx 0.540$, while the Hamming bound allows the larger value $\log_2(256/9)/8 \approx 0.604$.

At $n = 16$, knowledge gaps first appear. The one-error and two-error cases are not fully resolved by the best known lower and upper bounds. Thus the exact finite-blocklength packing problem becomes hard very quickly.

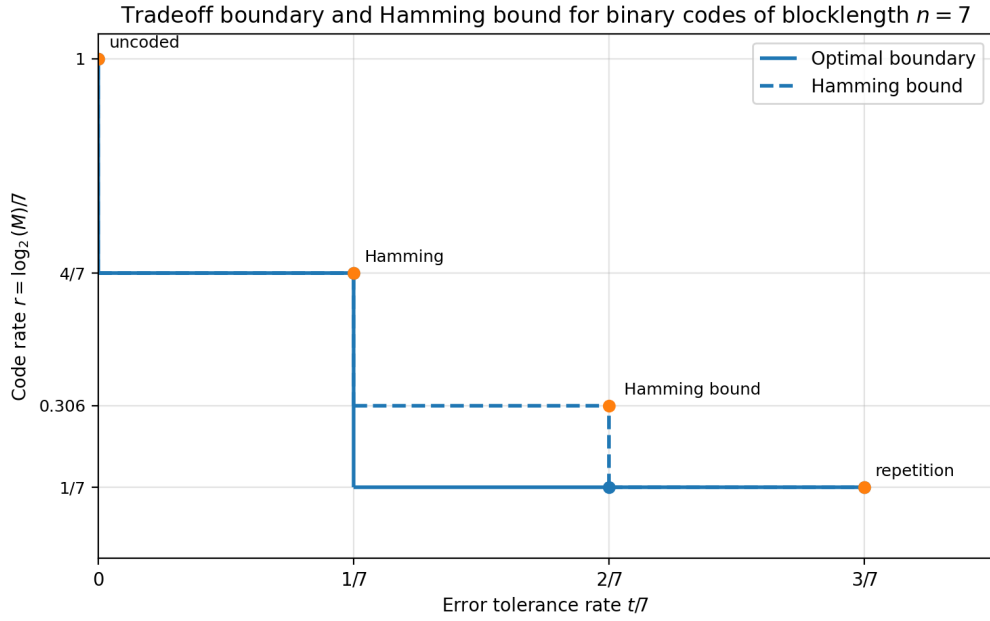


Figure 3: The $n = 7$ tradeoff boundary compared with the Hamming bound. All Pareto-optimal points are known for $n = 7$.

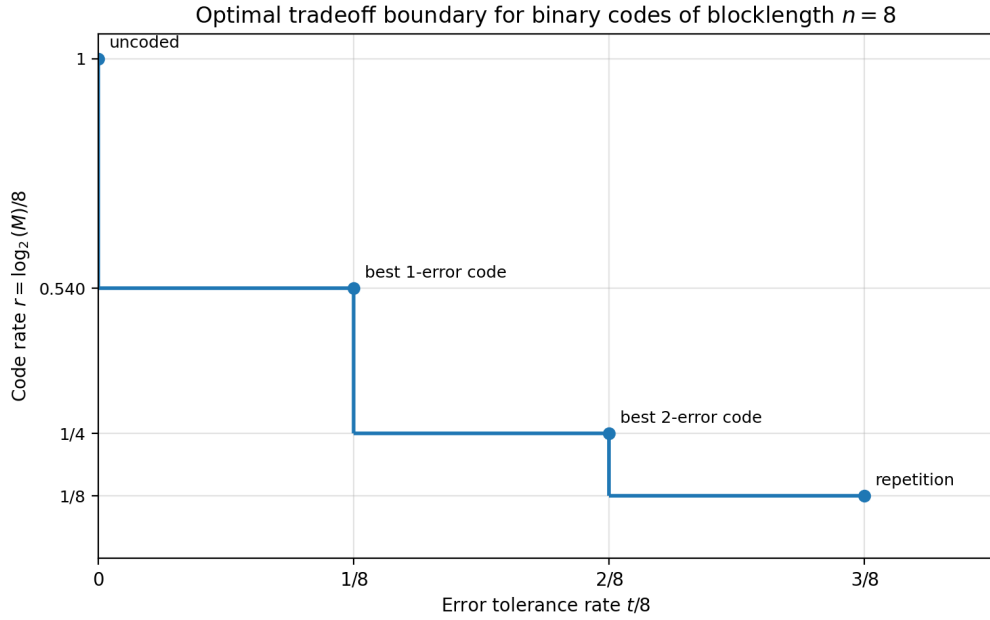


Figure 4: The optimal tradeoff boundary for binary codes of block length $n = 8$.

2 The Hamming bound and entropy

Let

$$B_n(t) = \{x \in \{0,1\}^n : d(x, 0^n) \leq t\}$$

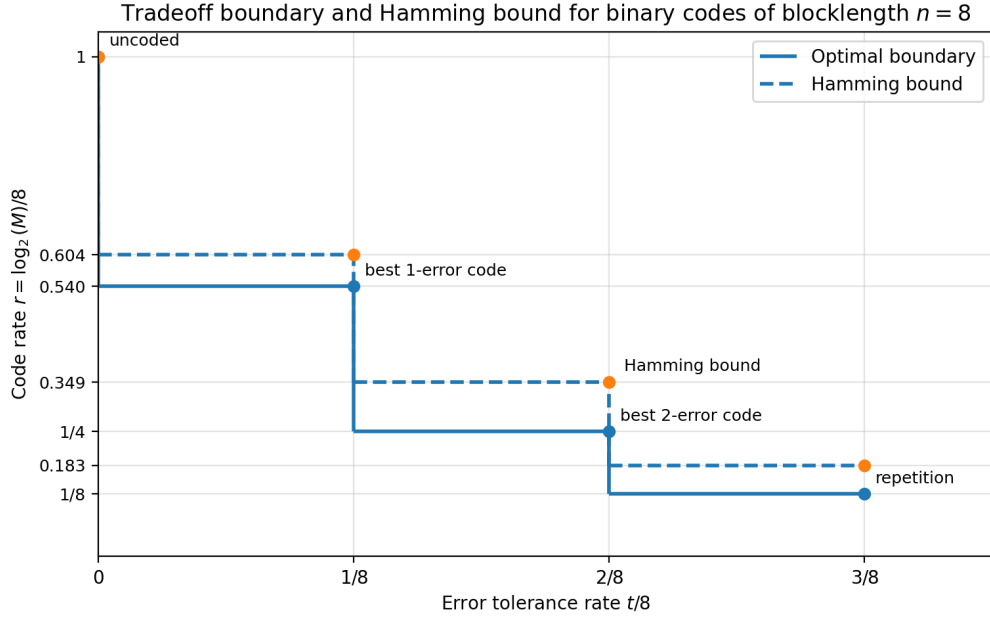


Figure 5: The $n = 8$ tradeoff boundary compared with the Hamming bound.

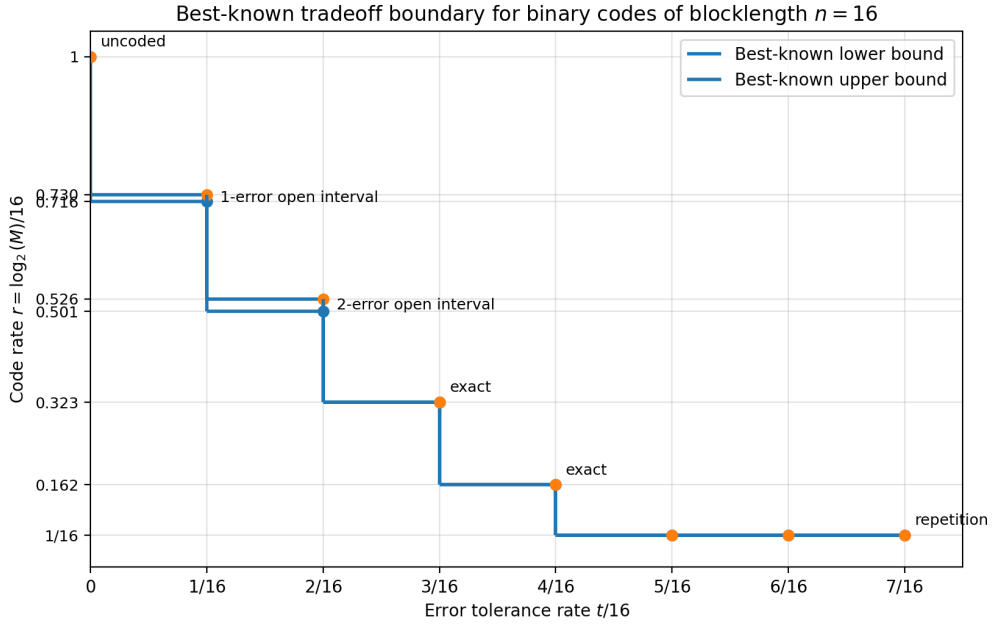


Figure 6: Best-known tradeoff boundary for binary codes of block length $n = 16$. The one-error and two-error cases remain open.

be the Hamming ball of radius t around the all-zero vector. Its size is

$$|B_n(t)| = \sum_{i=0}^t \binom{n}{i}.$$

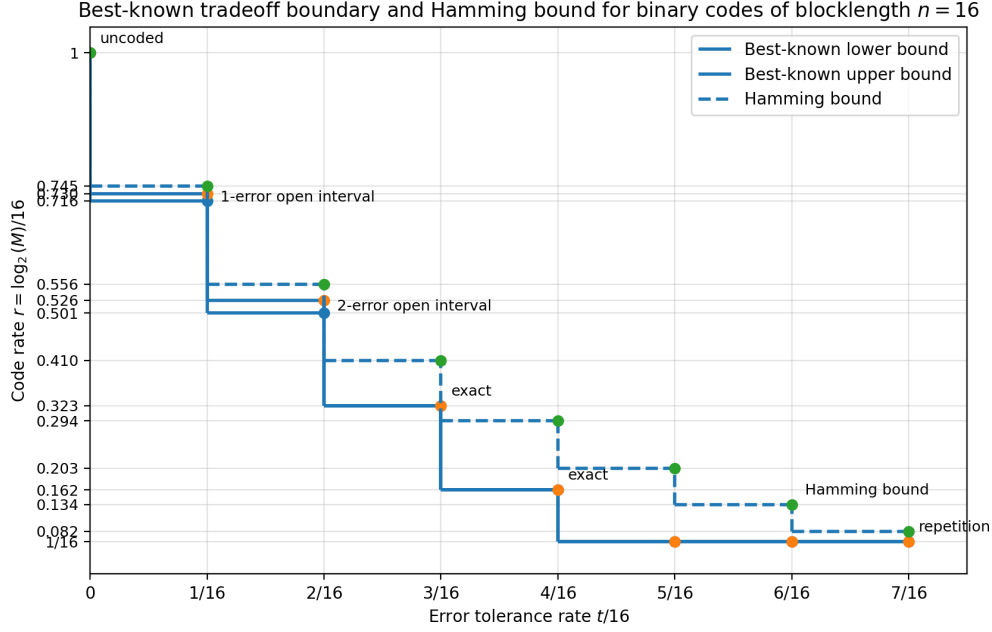


Figure 7: Best-known $n = 16$ boundary compared with the Hamming bound.

If a code corrects t worst-case errors, the Hamming balls of radius t around distinct codewords must be disjoint. Hence

$$M|B_n(t)| \leq 2^n.$$

Since $M = 2^{nr}$, we obtain

$$r \leq 1 - \frac{1}{n} \log_2 |B_n(t)| = 1 - \frac{1}{n} \log_2 \left(\sum_{i=0}^t \binom{n}{i} \right). \quad (1)$$

This is the Hamming bound.

Now set

$$p = \frac{t}{n}, \quad 0 \leq p \leq \frac{1}{2}.$$

The quantity p is the error tolerance rate. A fundamental asymptotic fact is

$$\frac{1}{n} \log_2 \sum_{i=0}^{pn} \binom{n}{i} \rightarrow H(p) \quad \text{as } n \rightarrow \infty, \quad (2)$$

where

$$H(p) = p \log_2 \frac{1}{p} + (1-p) \log_2 \frac{1}{1-p} \quad (3)$$

is the binary entropy function. Therefore, asymptotically,

$$r \leq 1 - H(p).$$

Entropy re-emerges from a purely geometric counting problem: the Hamming ball of relative radius p contains about $2^{nH(p)}$ binary sequences.

The following plots show the same comparison for larger block lengths. The staircase shown in these figures is not claimed to be the exact optimal boundary. It is an achievable curve obtained from standard structured code families, including extended BCH codes, Reed–Muller codes, and repetition codes. The dashed curve is the Hamming bound.

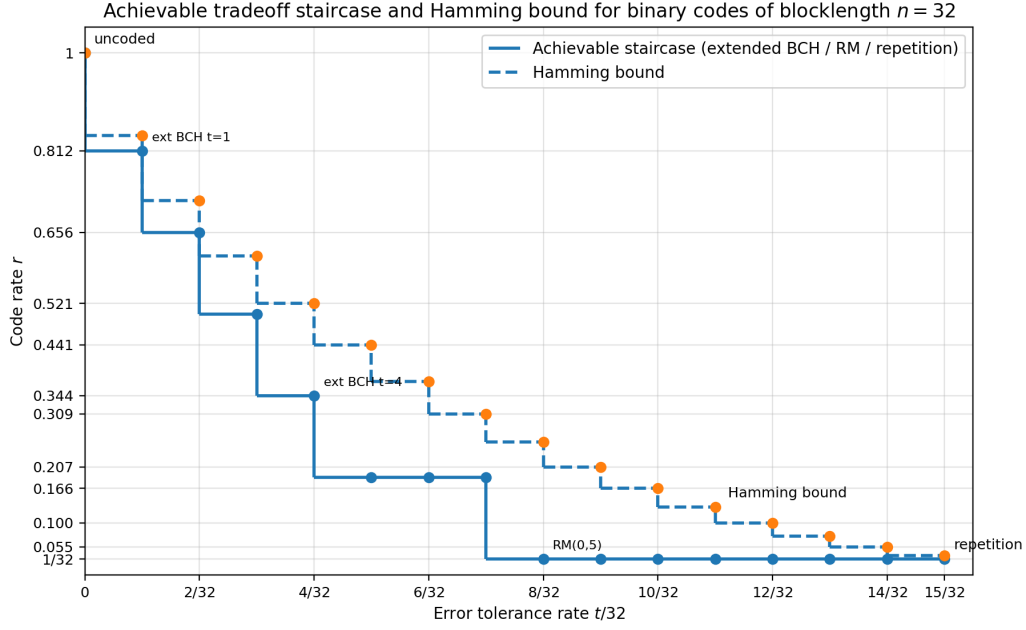


Figure 8: An achievable staircase and the Hamming bound for $n = 32$. Gaps between the Hamming bound and this particular family of constructed codes remain.

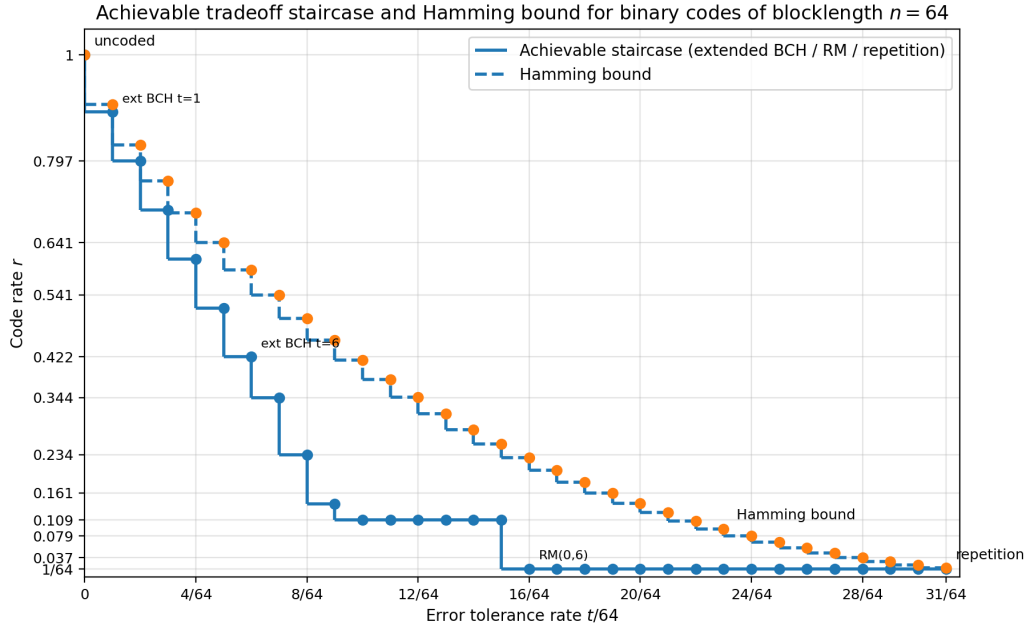


Figure 9: An achievable staircase and the Hamming bound for $n = 64$.

3 Worst-case and probabilistic noise models

Let the codebook be $\mathcal{C} \subseteq \{0, 1\}^n$. A decoder is a map

$$D : \{0, 1\}^n \rightarrow \mathcal{C},$$

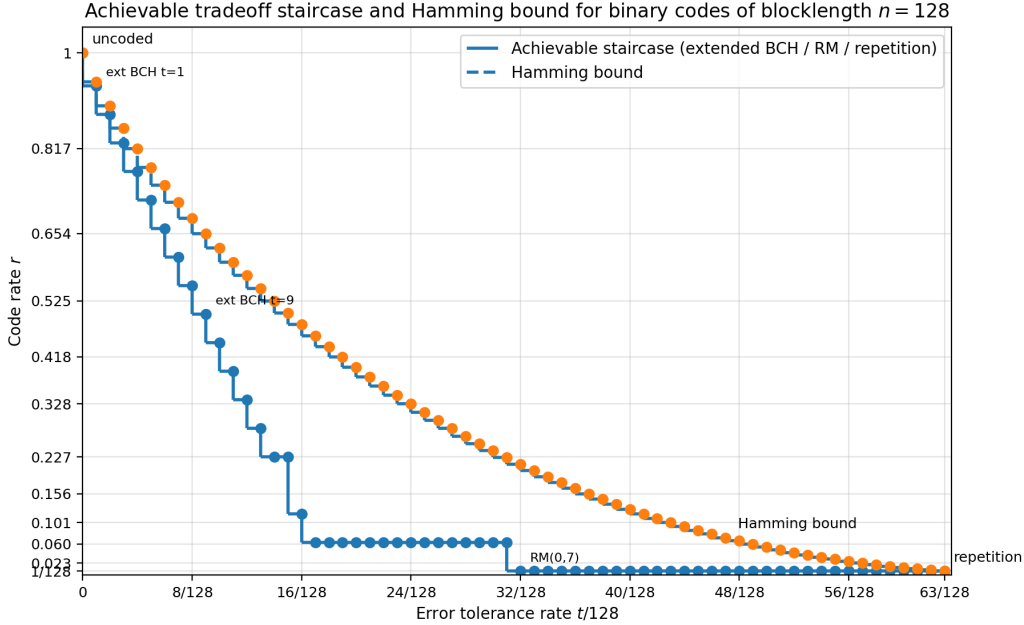


Figure 10: An achievable staircase and the Hamming bound for $n = 128$. The Hamming bound is approaching the curve $1 - H(p)$.

which takes a received word and outputs an estimate of the transmitted codeword. If the transmitted codeword is $x \in \mathcal{C}$ and the channel adds a binary noise vector $e \in \{0, 1\}^n$, then the receiver observes

$$y = x \oplus e.$$

In Hamming's formulation, the noise vector is not random. The requirement is

$$D(x \oplus e) = x \quad \text{for every codeword } x \in \mathcal{C} \text{ and every } e \text{ with } w(e) \leq t.$$

This is a worst-case model: the channel may choose whichever error vector of weight at most t is most difficult for the decoder. This formulation is strong and geometrically clean, since it leads directly to packing Hamming balls. But it also makes the optimal-code problem very hard. Even for moderate block lengths, we quickly run into gaps between the best known constructions and the best known upper bounds.

The reappearance of entropy suggests a different formulation. Instead of insisting on success against every possible low-weight error pattern, Shannon modeled the noise probabilistically and allowed a small but nonzero decoding error probability. In this setting, E is a random noise vector and the goal is to make

$$\mathbb{P}\{D(X \oplus E) \neq X\} \leq p_e,$$

where p_e is a target probability of decoding error.

There are several closely related probabilistic noise models one can consider:

- (1) The noise vector E is chosen uniformly among all bit-flip patterns of weight at most $t = np$.
- (2) The noise vector E is chosen uniformly among all bit-flip patterns of exactly weight $t = np$.
- (3) Each coded bit is flipped independently with probability p . This is the binary symmetric channel, denoted $\text{BSC}(p)$.

The first two models are useful for connecting directly to the Hamming-ball picture. The third model is Shannon’s standard channel model. It captures a simple memoryless noisy channel: each bit is transmitted correctly with probability $1 - p$ and flipped with probability p , independently of the other bits.

In the following figures, we fix the target decoding error probability at

$$p_e = 10^{-3}$$

and plot the rate achieved by Shannon coding for several block lengths. The plots also show the Hamming bound and either the exact or best-known finite-blocklength boundary for small n , or an achievable staircase built from standard structured code families for larger n .

For small block lengths, Shannon coding performs poorly. The finite-blocklength penalty is very large, and the curve is barely visible near the bottom of the plot for $n = 16$.

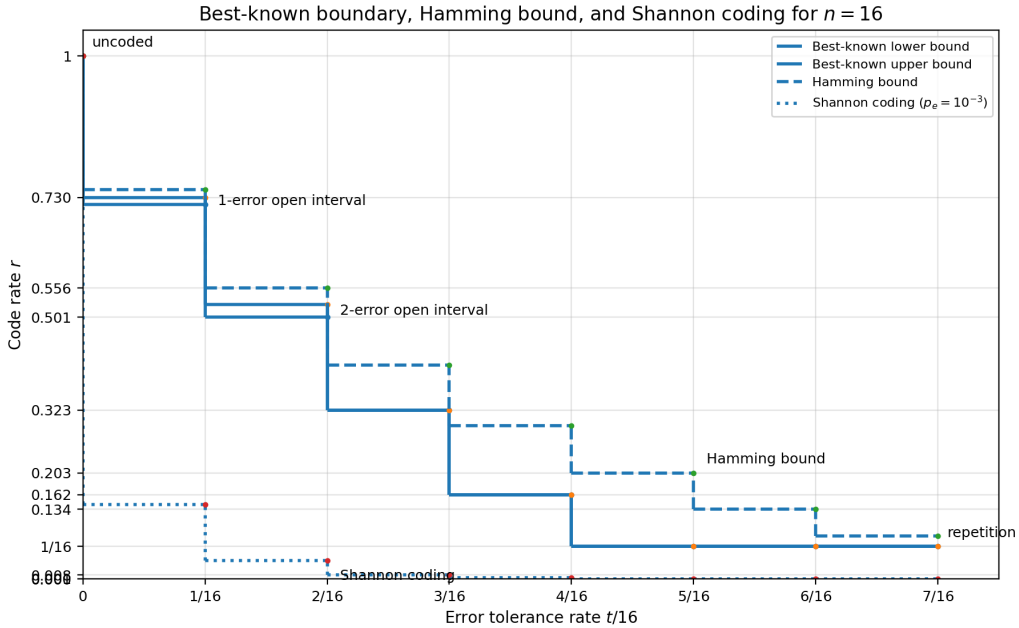


Figure 11: Best-known boundary, Hamming bound, and Shannon coding for $n = 16$ with $p_e = 10^{-3}$. For small n , Shannon coding is far from the Hamming bound.

As n grows, Shannon coding gets closer to the Hamming bound and eventually beats the particular structured codes shown here. This does not mean that the structured codes are the best possible structured codes; it means that Shannon coding is proving the existence of much better codes in the probabilistic model.

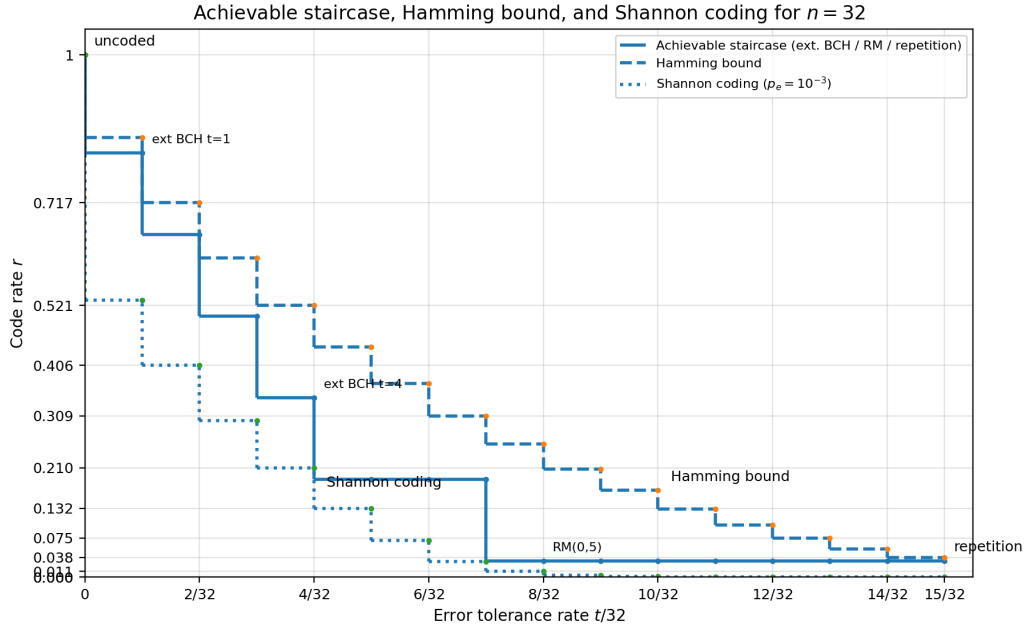


Figure 12: Achievable staircase, Hamming bound, and Shannon coding for $n = 32$.

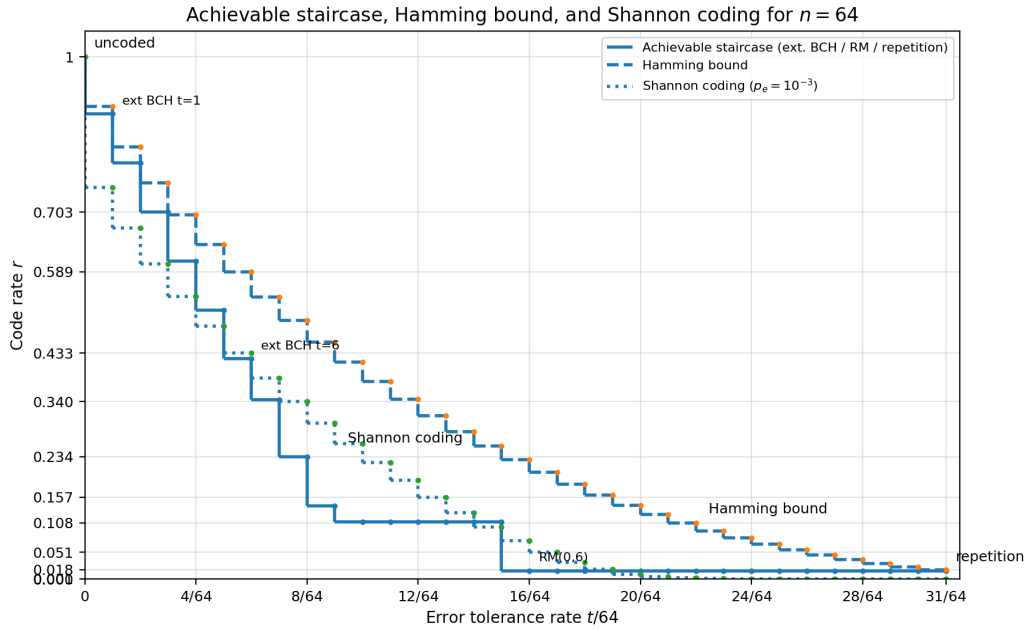


Figure 13: Achievable staircase, Hamming bound, and Shannon coding for $n = 64$.

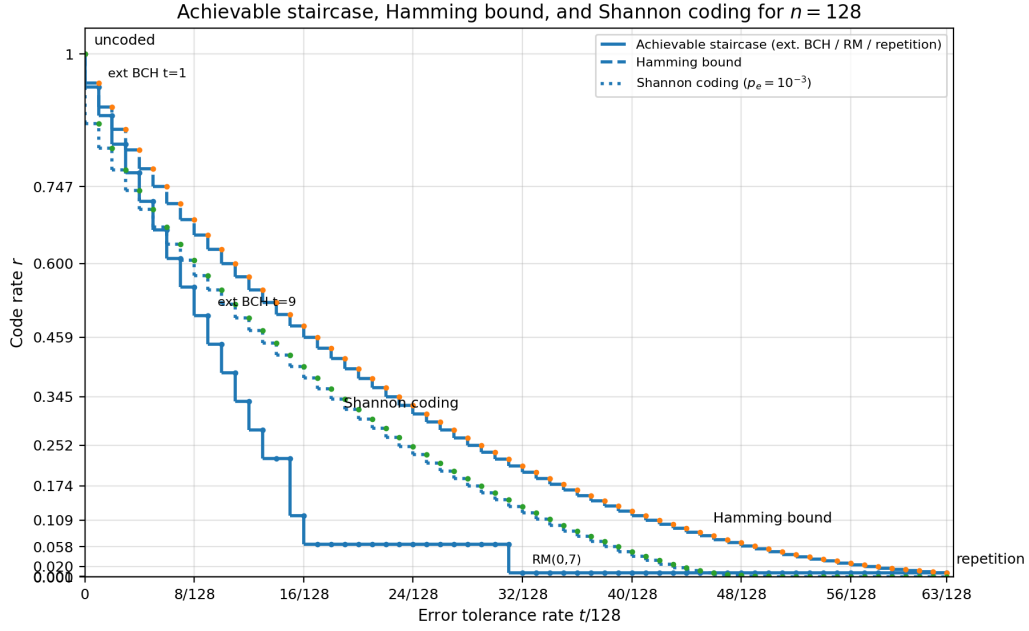


Figure 14: Achievable staircase, Hamming bound, and Shannon coding for $n = 128$. Shannon coding is now close enough to outperform the displayed worst-case constructions over a substantial range.

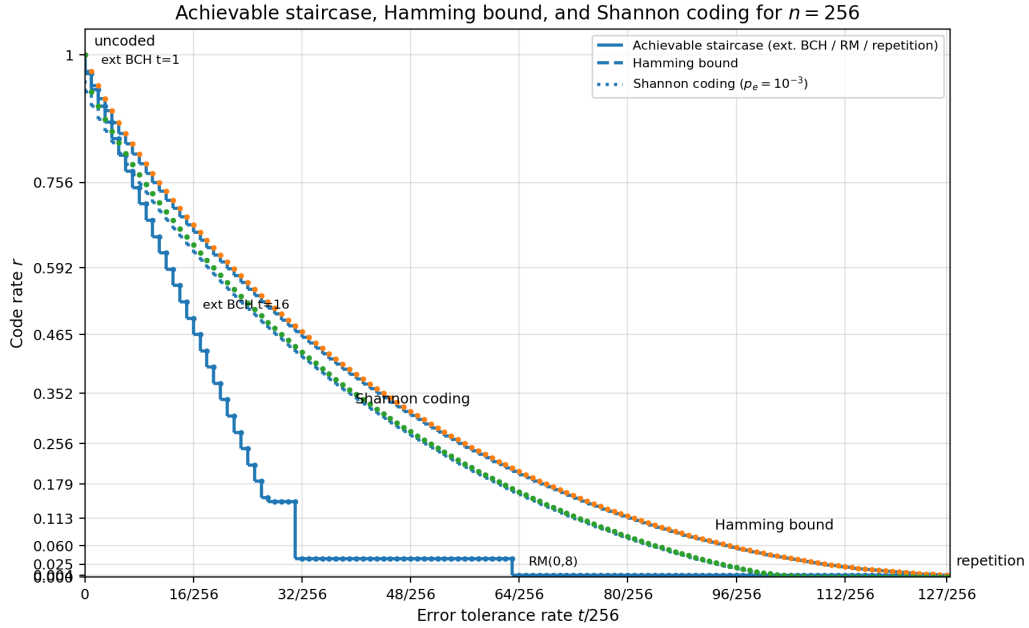


Figure 15: Achievable staircase, Hamming bound, and Shannon coding for $n = 256$.

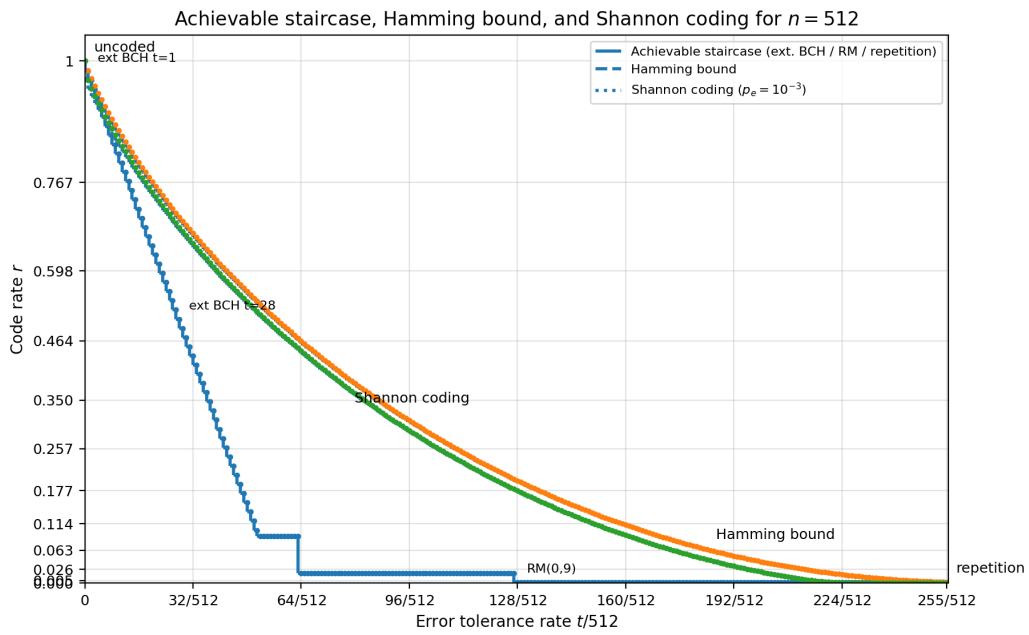
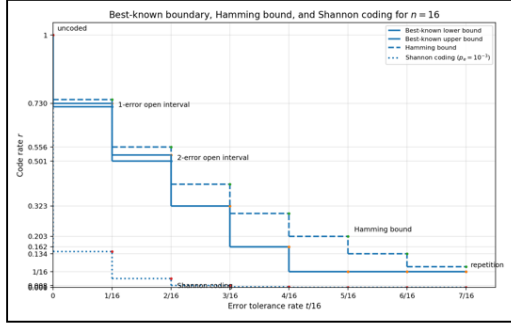
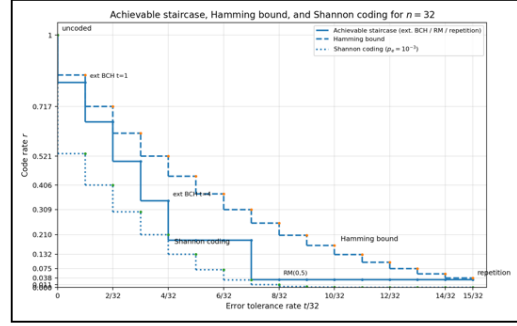


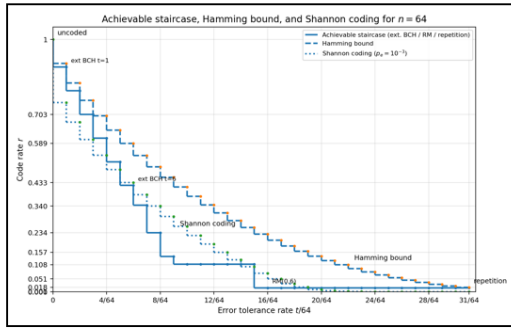
Figure 16: Achievable staircase, Hamming bound, and Shannon coding for $n = 512$. At this block length, Shannon coding is already very close to the Hamming bound for $p_e = 10^{-3}$.



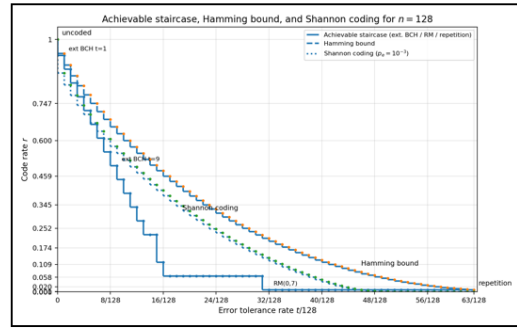
$n = 16$



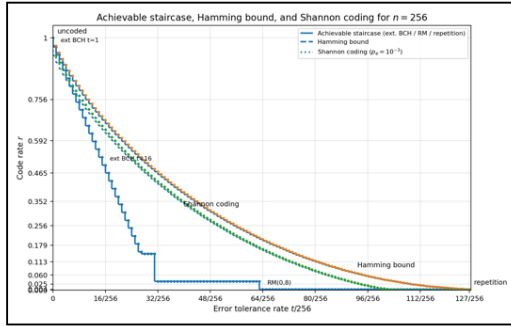
$n = 32$



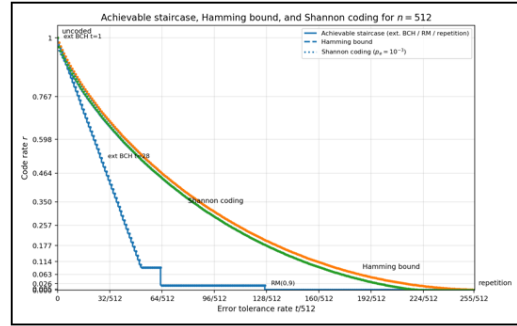
$n = 64$



$n = 128$



$n = 256$



$n = 512$

Figure 17: As n increases, the Shannon coding curve approaches the Hamming bound.

4 Shannon’s noisy channel coding theorem

The sequence of plots suggests Shannon’s noisy channel coding theorem. Shannon proved this result in his 1948 paper, “A Mathematical Theory of Communication” [1]. We state it here in the language of the probabilistic noise models above.

Theorem. *Consider any of the three probabilistic noise models above. For every code rate*

$$r < 1 - H(p)$$

and every target decoding error probability $p_e > 0$, there exists, for all sufficiently large block lengths n , a binary code of block length n and rate at least r whose probability of block decoding error is at most p_e . The number

$$C = 1 - H(p)$$

is called the capacity of this channel.

What this theorem says is very strong. No matter how close the target rate is to $1 - H(p)$, say

$$r = 1 - H(p) - 0.01,$$

and no matter how small the target error probability is, say

$$p_e = 10^{-3},$$

Shannon coding can achieve both goals, provided the block length is sufficiently large.

Rates below capacity are achievable with vanishing probability of error. Rates above capacity are not achievable.

Shannon’s proof of the theorem is itself a probabilistic construction. Like the noise model, the code construction is random: choose each of the M codewords independently and uniformly from the space of all length- n binary sequences. Shannon then shows that, when n is large enough and $r < 1 - H(p)$, this randomly chosen codebook has small decoding error probability with positive probability. Therefore at least one deterministic codebook with the desired performance must exist. In this sense, Shannon coding is a proof by random construction rather than an explicit engineering recipe.

There is a serious computational drawback. A randomly chosen codebook has no useful structure. To decode, the receiver would essentially have to compare the received word with all codewords and choose the nearest one. Since the number of codewords grows exponentially with n , this decoding rule is computationally infeasible for large block lengths. This is very different from Viterbi decoding of convolutional codes, where the trellis structure gives a linear-time algorithm for fixed constraint length.

5 Polar codes

This raises a natural engineering question: can one construct codes that approach capacity and also have efficient encoding and decoding algorithms?

A major answer came from Erdal Arıkan’s polar codes. Polar codes were introduced through the channel polarization method, which constructs code sequences achieving the symmetric capacity of binary-input memoryless channels [2]. Under successive cancellation decoding, polar codes have encoding and decoding complexity on the order of

$$O(n \log n),$$

which is vastly better than the exponential complexity of Shannon’s randomly chosen codebooks. Polar codes are also part of the 5G New Radio channel coding standard, where polar coding is specified for several control and broadcast-channel uses [3, 4].

Thus the story from Hamming to Shannon to polar codes is the following:

worst-case sphere packing $\longrightarrow$ probabilistic capacity $\longrightarrow$ practical capacity-achieving code families.

6 Takeaways

- Hamming’s model is worst-case: the code must correct every error pattern of weight at most t .
- Shannon’s model is probabilistic: the noise is random, and we allow a small decoding error probability.
- The Hamming ball of relative radius p has exponential size

$$|B_n(pn)| \approx 2^{nH(p)}.$$

- Therefore the Hamming bound approaches

$$r \leq 1 - H(p).$$

- Shannon coding shows that rates below $1 - H(p)$ are achievable for the binary symmetric channel.
- The capacity of the binary symmetric channel is

$$C = 1 - H(p).$$

- Shannon coding proves existence but has exponential decoding complexity. Polar codes give a structured, efficient route toward capacity.

References

- [1] C. E. Shannon, “A Mathematical Theory of Communication,” *Bell System Technical Journal*, vol. 27, pp. 379–423 and 623–656, 1948. Reprinted version available at <https://people.math.harvard.edu/~ctm/home/text/others/shannon/entropy/entropy.pdf>.
- [2] E. Arikan, “Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels,” *IEEE Transactions on Information Theory*, vol. 55, no. 7, pp. 3051–3073, 2009. DOI: <https://doi.org/10.1109/TIT.2009.2021379>.
- [3] ETSI / 3GPP, *5G; NR; Multiplexing and channel coding (3GPP TS 38.212)*, Technical Specification TS 138 212. Available from https://www.etsi.org/deliver/etsi_ts/138200_138299/138212/.
- [4] MathWorks, “5G New Radio Polar Coding,” documentation example describing 3GPP NR polar coding for DCI, UCI, and BCH. <https://www.mathworks.com/help/5g/gs/polar-coding.html>.