Error-Correcting Codes That Nearly Saturate Shannon’s Bound

Ido Kanter\(^1\) and David Saad\(^2\)

\(^1\)Department of Physics, Bar-Ilan University, Ramat-Gan 52900, Israel.
\(^2\)The Neural Computing Research Group, Aston University, Birmingham B4 7ET, UK.

Gallager-type error-correcting codes that nearly saturate Shannon’s bound are constructed using
insight gained from mapping the problem onto that of an Ising spin system. The performance of
the suggested codes is evaluated for different code rates in both finite and infinite message length.

Efficient information transmission plays a central role
in modern society, taking a variety of forms, from tele-
phone and satellite communication to storing and retrieving
information on disk-drives. Error-correcting codes
are commonly used in most methods of information
transmission to compensate for noise corrupting the data
during transmission; they require the use of additional
information transmitted together with the data itself. The
percentage of informative transmitted bits, determines
the coding efficiency and subsequently the speed of
communication channels and the effective storage space
on hard-disks. In his seminal paper of 1948, Shannon \([1]\) derived the channel capacity, providing bounds on the
maximal code rate for which codes, capable of achieving perfect
tagreement, may be more realistic in some sce-
narios. The receiver can correct the flipped bits only if
be considered, and may be more realistic in some sce-
narios. The receiver can correct the flipped bits only if
the source transmits \(M(f) > N\) bits; the ratio between
the original number of bits and those of the transmitted
message \(R = N/M\) constitutes the code-rate for unbi-
ased messages. Shannon \([1]\) derived the channel capacity
and provided bounds on the maximal code rate \(R_c\), for
a given flip rate \(f\) and code bit error probability \(p_b\), for
which codes, capable of achieving perfect retrieval, exist.
The maximal code rate equals the channel capacity and
is given explicitly \([3]\) by

\[
R_c = \frac{(1 - H_2(f))}{(1 - H_2(p_b))},
\]

where \(H_2(x) = x \log_2(x) + (1 - x) \log_2(1 - x)\).

Shannon’s theory is unconstructive, and the many
good algorithms that have been introduced over the years
(e.g., BCH, Reed-Muller and Reed-Solomon codes, for a
review see \([4]\)) fall short of saturating Shannon’s bounds,
although they may provide close-to-optimal performance
in specific scenarios. Even the most advanced code
to date, the Turbo code \([5]\) is somewhat below Shannon’s
bound.

One error-correcting code which recently became pop-
ular is the Gallager code \([2,6–8]\), which was abandoned
shortly after its introduction due to the limited computa-
tional abilities of the time. In this method, represent-
ing a special case of parity-check codes, the transmitted
message comprises the original message itself and addi-
tional bits, each of which is derived from the parity of
a sum of certain message-vector bits. The choice of the
message-vector elements used for generating single codeword
bits is carried out according to a predetermined
random set-up and may be represented by a product of a
randomly generated sparse matrix and the message-
vector in a manner explained below. Decoding the re-
ceived message relies on iterative probabilistic methods
like belief propagation \([6]\).

It has been shown that by using Gallager-type methods
and specific choices of the encoding/decoding matrix it
is possible to improve the maximal practically achievable
code-rate \([6]\) although results are still somewhat below
Shannon’s capacity. The root of the problem is the inevi-
table tradeoff between improving the code’s corrective
capabilities and the need for a practical and reliable iter-
ative decoding process, guaranteed to converge from any
initial condition (i.e., that will not require additional,
typically unavailable, information about the message it-
self). This goal is achieved by understanding the physical
characteristics of the problem and devising a new method
based on this insight. As Gallager-type methods form the
basis of our proposal we will now explain explicitly the
version we employ - the MN code \([6]\).

In the MN code one constructs two sparse matrices
\(A\) and \(B\) of dimensionalities \(M \times N\) and \(M \times M\) respec-
tively. The matrix \(A\) has \(K\) non-zero (unit) elements
per row and \(C(=KM/N)\) per column while \(B\) has \(L\) per
row/column. The matrix \(B^{-1}A\) is then used for encoding
the message.
The received message comprises the transmitted vector corrupted by the noise vector $n$: $r = t + n$ (mod 2). Decoding is carried out by employing the matrix $B$ to obtain: $z = B (t + n) = As + Bn$, and requires solving the equation
\[
[A, B] \begin{bmatrix} s' \\ n' \end{bmatrix} = z,
\]
where $s'$ and $n'$ are the unknowns. This may be carried out using methods of belief network decoding \cite{6}, where pseudo-posterior probabilities, for the decoded message bits being 0 or 1, are calculated by solving iteratively a set of equations for the conditional probabilities of the codeword bits given the decoded message and vice versa. For exact details of the method used and the equation themselves see \cite{6}.

Most studies of Gallager-type codes have been carried out via methods of information theory (e.g., \cite{6}). The first link between a special case of Gallager’s method, where $B = I$ the identity matrix, and the realm of physical spin-systems was established by Sourlas \cite{10} by mapping the problem onto that of a Hamiltonian system, replacing the original Boolean variables by binary ones which are analogous to spins in Ising-type systems with Multi-Spin Interactions (MSI). For this simple case the system is described by the Hamiltonian
\[
H = - \sum_{\{i_1, i_2, \ldots, i_K\}} J_{i_1, i_2, \ldots, i_K} s'_1 s'_2 \cdots s'_K \tag{2}
\]
where $\{s'_i\}$ are the binary dynamical variables ($\pm 1$), used in the decoding process. The tensor $J_{i_1, i_2, \ldots, i_K} = \pm s_1 s_2 \cdots s_K$ with probabilities $1 - f$ and $f$ correspondingly, represents the received codeword corrupted by noise during transmission, $\hat{s}$ being the binary representation of the original Boolean message vector $s$: the choice of indices $i_1, i_2, \ldots, i_K$ corresponds to the non-zero row elements of the matrix $A$. Under a gauge transformation this model is mapped onto an Ising spin system with ferromagnetic bias; finding the ground state of the Hamiltonian is closely related to finding the Bayes optimal posterior under a certain noise level \cite{10}. This mapping onto Hamiltonian spin-systems, suggested by Sourlas for highly connected systems, was recently extended to particular forms of sparse matrices $A$ (where $B = I$) as well as to certain $B$ matrices \cite{11}. In this extended framework, $K$ and $L$ represent the number of MSI among the signal and noise components respectively.

Our method uses the same structure as the MN codes and builds on insight gained from the study of physical systems with symmetric and asymmetric \cite{11} MSI and from examining a special case of Gallager’s method \cite{6,5}. These theoretical studies indicate that one may obtain superior capabilities, in terms of the achievable code rate, by choosing high $K$ and $L$ values; however, they come at the expense of poor decoding performance as the corresponding basins of attraction shrink rapidly with the increasing $K$ and $L$ values, making it essential to have high initial overlap between the original message and the dynamical variables for the iterative decoding process to converge successfully. Such information is clearly unavailable in practical scenarios. One should emphasise that the basin of attraction shrinks dramatically. In the system suggested by Sourlas, for instance, the initial overlap (magnetisation in the physical system) $m = 1/N \sum_{i=1}^{N} (2s_i - 1)/(2s'_i - 1)$ required in the case of $K = 6$ should be higher than 0.99 for a successful convergence; this has been shown by numerical simulations as well as by a mean-field calculation to be presented elsewhere. On the other hand, highly robust iterative decoding is obtained for low $K$ and $L$ values at the expense of sub-optimal capabilities (i.e., low end overlap).

The method presented here is based on constructing the matrices $A$ and $B$ in a manner that correspond to the gradual introduction of higher connectivity sparse sub-matrices, exploiting the excellent convergence properties of codes based on low $K$ and $L$ values with the superior performance of high-$K$ codes. More specifically, one aims at starting with low MSI values, in this case $K + L \leq 3$, so as to bring the system to high overlap values from practically any initial condition; higher values of $K$ and $L$, e.g., $3 < K + L \leq 5$, may then be used for bringing the system to a perfect overlap between the decoded and the original word.

The practical implementation of the encoding is similar to that of the MN code except that the composed matrix used, $[A|B]$, comprises randomly chosen sparse sub-matrices of different connectivities. The generated codeword, constructed by taking the parity of sums of message vector bits selected according to the specific choice of $A$ and $B$, is then transmitted through the noisy channel. Decoding the corrupted codeword is carried out using an iterative process identical to that of Ref. \cite{6} and can take two forms: a) A gradual introduction of higher connectivity sub-matrix components in the Hamiltonian system used for decoding following the above description, where end result at each stage serves as an initial condition for the next. This is equivalent, from a physical point of view, to changing the Hamiltonian as a function of time by gradually summing over more message bits in Eq.\cite{6}. b) Using the composed matrices, including a variety of sub-matrices with different connectivities, right from the start. The latter, which simply correspond to a particular construction of the matrices $A$ and $B$ in the MN code, has been used in most of our experiments due to its simplicity, although the former has shown faster convergence at high noise levels. In both cases the explicit choice of sites for generating a specific code-word bit is carried out at random, in a similar fashion to most
Gallager-type codes.

The main question that should be addressed is the optimal choice of sub-matrix connectivities. There are many possibilities for choosing \( K \) and \( L \) values for the different stages and one should examine various possibilities before arriving at the optimal configuration. However, there are a few guidelines one should follow: 1) Initial stages are characterised by low \( K \) and \( L \) values; \( K \) values are chosen gradually higher, so as to support the correction of faulty bits. 2) One should choose the number of non-zero column elements as uniformly as possible, as the number of connections per bit (spin) defines the corrective input it receives (this is somewhat in contrast to the approach adopted for irregular Gallager codes in which column/row connectivity is taken from some distribution \( \mathcal{D} \)). 3) As in most of these systems both solutions, with \( m = \pm 1 \), are equally attractive one should break the inversion symmetry. This may be achieved by adding some odd-MSI (i.e., an odd value for \( K + L \)) to the mainly even \( K + L \) value used initially; this assists in breaking the symmetry from any initialisation of the iterative equations with practically no effect on the basin of attraction. 4) To guarantee the inversion of the matrix \( B \), and since noise bits have no explicit correlation, we use a patterned structure, \( B_{i,k} = \delta_i \pm \delta_{i,k-5} \), for the \( B \)-submatrices with \( L = 2 \) and \( B_{i,k} = \delta_i \) for \( L = 1 \). Other practical points as well as a more detailed explanation of the physical insight leading to the optimal choice of MSI connectivity and the relation to Sourlas’s code will be presented elsewhere.

| \( R \) | \( N \) | \( A \) | \( K \) | \( B \) | \( L \) | \( f_c^N \) | \( f_c^\infty \) |
|---|---|---|---|---|---|---|---|
| 1/3 | 10000 | \( N \times N \) | 1 | \( N \times 3N \) | 2 | 0.159 | 0.169 | 0.174 |
| | | \( 3/4 N \times N \) | 3 | \( 3/4 N \times 3N \) | 2 | -0.170 |
| | | \( 5/4 N \times N \) | 3 | \( 5/4 N \times 3N \) | 1 | |
| 1/4 | 30000 | \( 3/2 N \times N \) | 1 | \( 3/2 N \times 4N \) | 2 | 0.204 | 0.210 | 0.214 |
| | | \( N/2 \times N \) | 3 | \( N/2 \times 4N \) | 2 | -0.211 |
| | | \( 2N \times N \) | 3 | \( 2N \times 4N \) | 1 | |
| 1/5 | 36000 | \( 3N \times N \) | 1 | \( 3N \times 5N \) | 2 | 0.235 | 0.239 | 0.243 |
| | | \( 2N \times N \) | 3 | \( 2N \times 5N \) | 1 | -0.240 |

**TABLE I.** The critical flip rates \( f_c^N \) and \( f_c^\infty \) obtained by employing our method for various code rates in comparison to the maximal flip rate \( f_c \) provided by Shannon’s bound. Details of the specific architectures used and their row/column connectivities are also provided.

We conclude this presentation with a demonstration of the method’s capabilities for three different code-rates \( R = 1/3, 1/4 \) and \( 1/5 \). In each of the cases we divided the composed matrix \( [ A | B ] \) to six sub-matrices characterised by specific \( K \) and \( L \) values as explained in table 1; the dimensionality of the full \( A \) and \( B \) matrices are \( M \times N \) and \( M \times M \) respectively. Sub-matrix elements were chosen at random according to the guidelines mentioned above. Encoding was carried out straightforwardly by using the matrix \( B^{-1} A \) and the corrupted messages were decoded using the set of recursive equations of Ref. [6], using random initial conditions. In each case, \( T \) blocks of \( N \)-bit unbiased messages (where exactly 1/2 of the bits are 1) were sent through a noisy channel of flip rate \( f \) (i.e., an exact fraction \( f \) of the codeword bits were flipped); both bit and block error-rates, denoted \( p_b \) and \( p_a \) respectively, were monitored. We performed at least \( T = 10000 \) trials runs for the smaller systems \( (N = 10000, 12000) \) and \( T = 1000–2000 \) runs for the larger ones \( (N = 30000, 36000) \) for each flip-rate value, starting from different initial conditions. These were averaged to obtain the mean bit error-rate and the corresponding variance. In most of our experiments we observed convergence after less than 100 iterations, except very close to the critical flip rate. The main halting criterion we adopted relies on the stationarity of the first \( N \) bits (i.e., the decoded message) over a certain number of iterations. The decoding algorithm’s complexity is of \( O(N) \) as all matrices are sparse. The inversion of the matrix \( B \) is carried out only once and requires \( O(N^2 \log N) \) operations.

**FIG. 1.** Bit-error rate \( p_b \) as a function of the flip rate for given code-rates \( R = 1/3, 1/4 \) and \( 1/5 \). Results for each code-rate appear as symbols adjacent to a line representing Shannon’s theoretical bound; triangles and squares, represent mean values obtained for small and large network sizes respectively, corresponding to \( N = 10000 \) and \( 30000 \) for \( R = 1/3, 1/4 \), \( N = 12000 \) and \( 36000 \) for \( R = 1/5 \). Predicted code-rate values in the \( N \rightarrow \infty \) limit appear as arrows on the \( x \) axis. In table 1 we present the typical architectures used as well as the maximal flip rate \( f_c^N \) for which not more than a single error-bit per block have been observed on average for a particular message length \( N \), the predicted maximal flip rate \( f_c^\infty \) once finite size effects have been considered (discussed below) and Shannon’s maximal flip rate \( f_c \) defined in Eq. (6). In all these cases one obtains, on average, perfect retrieval for noise rates that almost saturate Shannon’s bound for the critical flip rate. Just for comparison, the corresponding results reported in Ref. [6] for regular and irregular Gallager codes \( (R = 1/4) \), based on 10000 trials and \( N = 16000 \) report a critical value around \( f = 0.160 \) in comparison to \( f_c^N = 0.204 \) and \( f_c^\infty = 0.210–0.211 \) reported here.
Figure 1 shows results obtained for code-rates $R = 1/3, 1/4$ and $1/5$ and various flip rates; results for each one of the code-rates appear as symbols adjacent to a line representing Shannon’s theoretical bound for the given code-rate and noise level. Triangles and squares, represent mean values obtained for small and large network sizes respectively, corresponding to $N = 10000$ and $30000$ for $R = 1/3, 1/4$ and $N = 12000$ and $36000$ for $R = 1/5$; variances are smaller than the symbol size. One notes the existence of finite size effects, manifested in the difference between the results obtained for different system sizes. Predicted code-rate values in the $N \to \infty$ limit, derived below, are represented as arrows on the $x$ axis. The results clearly show that in all the code-rates examined our method comes very close to saturating Shannon’s bound.

**FIG. 2.** The block magnetisations profile for $R = 1/5$, $f = 0.236, 0.237$ (solid and dashed lines respectively) and $N = 12000, 36000$, showing the sample magnetisation $m$ vs. the fraction of the complete set of trials. A total of 1000-10000 trials (for larger and smaller systems respectively) were rearranged in a descending order according to their magnetisation values (directly related to the overlap between the decoded and the original message). The fraction of perfectly retrieved blocks increases with system size (thick lines). Inset - log-log plots of the mean convergence times $\tau$ for $R = 1/3, 1/4, 1/5$ and different $N$ values, carried out on perfectly retrieved blocks with less than 2 error bits. The optimal fitting of expressions of the form $\tau \propto 1/(f_c^\infty - f)$ through the data provides another indication for the $f_c^\infty$ values, which are consistent with those obtained by the first method. To conclude, we have shown that through a successive change in MSI and connectivity, while keeping the connectivity low ($\leq 5$), one can boost the performance of matrix based error-correcting codes, getting ever closer to saturating the theoretical bounds set by Shannon. It is quite plausible that the performance reported here may be improved upon by fine tuning the choice of architecture, which is currently under way. Moreover, it is highly likely that several architectures will provide similar performance in the thermodynamic limit; it would be worthwhile to examine their finite size behaviour above and below saturation which is of great practical significance.

1. C.E. Shannon, *Bell Sys. Tech. J.*, 27, 379 and 623 (1948).
2. R.G. Gallager, *Low density parity check codes* Research monograph series 21 (MIT press), 1963.
3. T.M. Cover and J.A. Thomas, *Elements of Information Theory* (Wiley), 1991.
4. A.M. Michelson and A.H. Levesque, *Error-Control Techniques for Digital Communications* (Wiley), 1985.
5. C. Berrou and A. Glavieux *IEEE Trans. Comm.*, 44, 1261 (1996).
6. D.J.C. MacKay, *IEEE Trans. IT*, 45, 399 (1999).
7. M.C. Davey and D.J.C. MacKay, *IEEE Comm. Lett.*, in press (1999).
8. M. Luby, M. Mitzenmacher, A. Shokrollahi and D. Spielman, in *IEEE proceedings of the International Symposium on Information Theory ISIT98* (1998).
9. B.J. Frey, *Graphical Models for Machine Learning and Digital Communication* (MIT Press), 1998.
10. N. Sourlas, *Nature*, 339 693 (1989).
11. Y. Kabashima and D. Saad, *Euro. Phys. Lett.*, 45 97 (1999); Y. Kabashima, T. Murayama and D. Saad, unpublished (1999).
12. I. Kanter *Phys. Rev. A*, 38, 5972 (1988), and I. Kanter and H. Sompolinsky, *Phys. Rev. Lett.* 58, 164 (1987).