Error correcting codes

In our daily activity we routinely send and receive, i.e. exchange, information. Our reliability requirements can vary but often it becomes especially important to keep the transmitted information uncorrupted, prone to errors. A long time ago people got the idea of redundancy: reduce the probability of error by transmitting more information than actually needed. For example, if one wants to spell the word "code" via phone he/she may say "Charlie, Oscar, Delta, Echo", so it is fairly easy to guess the 1st informative letters even if the connection is not perfect.

In 1948 Claude Shannon put the ancient idea on the solid mathematical footing proving that error-free communication in the noisy environment is possible in principle. In digital world the information is usually sent as a sequence of bits — 0s and 1s. Instead of transmitting L information bits one first preconverts them into N coded bits with $N-L$ redundant bits helping to correct transmission errors at the other end of communication line. The simplest example is given by the famous Shannon code: 0 is encoded as 000, 1 is encoded as 111. The receiver decodes the message by voting between 0s and 1s. If 0 (coded as 000) is sent, then the error is detected only if the message corrupted in the result of transmission becomes 011, 101, 110, or 111, i.e. if more then two bits are flipped. If the communication channel is good, one flip is rare and the simultaneous flip of two bits is even less probable.

Optimal decoding corresponds to funding preimage (code word) that is most probable given the detected (i.e. corrupted by noise) signal. This Maximal Likelihood method is optimal (one cannot do better then that) but expensive as it requires subsequent comparison of 2^L code words, i.e. the computational complexity grows exponentially with the code length.

In 1961 Robert Gallager invented Low Density Parity Check (LDPC) codes, that are codes with the so-called parity check matrix being sparse. For LDPC decoding Gallager suggested to use approximate (thus suboptimal) but computationally efficient method of iterative decoding. These remarkable codes and iterative decoding scheme were soon forgotten for nearly 30 years, to become reinvented in mid-90's. Now, LDPC codes are believed to be one of the best performing long codes ever invented (see, e.g., `http://www.flarion.com/products/vector.asp`).

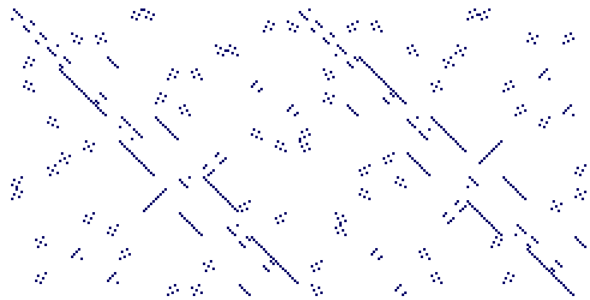

Figure 1: Parity check 240×120 matrix of Margulis code with $p = 5$.

The idea of the heuristic decoding proposed by Gallager is simple: For any detected bit check how many parity checks connected to the bit are satisfied. If the number of "unhappy" checks exceeds the number of "happy" ones, flip the bit. The process continues till a code word is recovered (no error) or the algorithm is stuck (unrecoverable error). The iterative procedure just explained allows useful generalization to the "soft" case, when detected signal is a real number, rather then 0 or 1 "hard" integer. Soft decoding is known to perform better.

Bit-Error-Rate (BER) measuring relative number of errors per transmitted sequence is the major characteristics of an error-correction scheme (e.g. iterative decoding of an LDPC code) performance. At low values of the Signal-to-Noise-Ratio (SNR) BER of an LDPC code can be analyzed numerically via Monte Carlo simulations. It can also be shown that in the low SNR range the performance of the iterative decoding is close to the Maximum Likelihood optimum.

The domain of higher SNR, lower BER is much less explored/understood. However, it is well known that performance of the iterative decoding becomes seriously polluted with the SNR increase. The transition from good to bad regimes (in the iterative decoding performance) is often called the waterfall to error-floor transition in the coding theory literature.

The main difficulty in the error floor analysis stems from the fact that to access the challenging domain of low errors one actually needs to analyze very rare a-typical events. Actually, this difficulty is not uncommon. In many problems of disordered and/or nonequilibrium statistical mechanics the challenge may also be in accessing rare a-typical events. Moreover, there exist a special method, called optimum fluctuation, instanton method, that is designed specifically to attack the difficult task of rare event analysis.

Our recent paper [1] adopts the generic instanton method for calculation of BER of LDPC codes detected iteratively. Brief description of the method idea is as follows. Typical noise configuration is small and the error probability falls down exponentially with the noise amplitude (level) increase. The instanton method is about finding such a special (optimal) configuration of the noise that makes the major contribution into BER. The optimal configuration has a very specific code-dependent shape. BER is sharply picked at the optimum configuration: minor distortion of the optimum configuration costs an essential fall off in the error probability. Therefore, at large SNR only optimal configuration and its minor vicinity contribute BER.

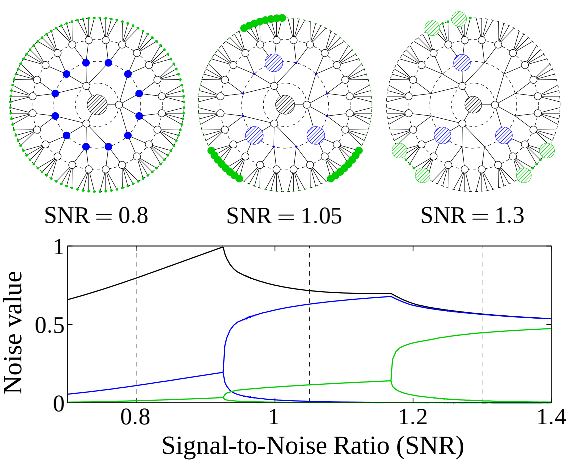

Figure 2: Optimal noise configuration as a function of SNR. The code is represented by a tree-like Tanner graph.

We found that for low SNR the optimal noise configuration is localized in some finite vicinity of the tested bit (that is the bit where BER is measured). With the SNR increase bits start to communicate with each other through adjusted parity checks, leading to essential delocalization of the optimal configuration in the bit-space. The process of the noise delocalization may go through some number of steps/transitions. Eventually (at large SNR) the delocalized configuration becomes sensitive to such global features of the LDPC code as loops on the Tanner graph characterizing the code.

Furthermore, we discovered that in the case of iterative decoding BER at the largest values of SNR

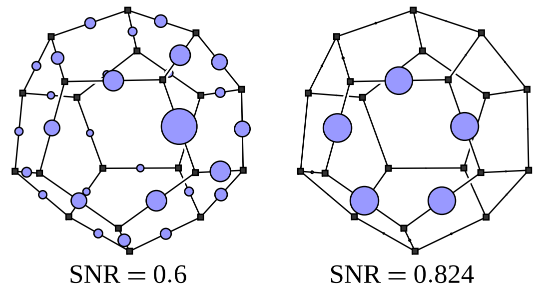

Figure 3: Optimal noise configuration: tree-like phase *vs.* loopy phase. The number of iterations in iterative decoding is 8.

is fully explained by the code-specific instanton configuration sensitive to some global characteristics of the code [2]. This analysis offers comprehensive (i.e. both qualitative and quantitative) explanation of the error-floor phenomenon.

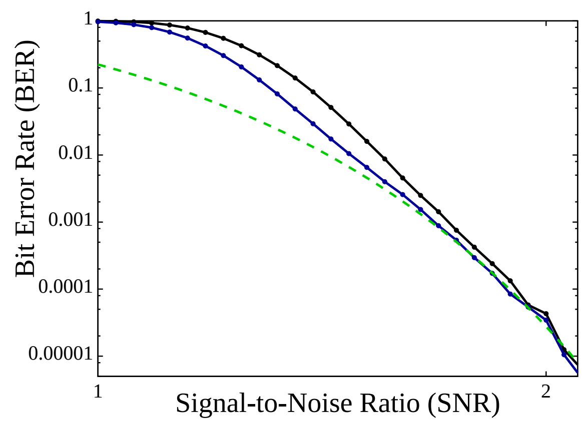

Figure 4: Probability of error as function of SNR for Margulis code with $p = 5$. Monte Carlo simulation for 8 iterations (black line) and 32 iterations (blue line) of iterative decoding, and optimal fluctuation method prediction (green line).

References

[1] V. Chernyak, M. Chertkov, M.G. Stepanov, B. Vasic, *Error correction on a tree: an instanton approach*, Phys. Rev. Lett. **93** (19) 198702 (2004).

[2] V. Chernyak, M. Chertkov, M.G. Stepanov, B. Vasic, *Instanton detection of the error-floor*, in preparation.

Contact Information: Misha Stepanov – Center for Nonlinear Studies and Complex Systems Group, Theoretical Division, Los Alamos National Laboratory, MS-B213, Los Alamos, NM, 87545, Phone: (505) 667-6840, E-mail: stepanov@cnls.lanl.gov.