Finite-Length Scaling of Convolutional LDPC Codes

Markus Stinner, Pablo Olmos

markus.stinner@tum.de, olmos@tsc.uc3m.es

Motivation and Example

- Capacity-reaching LDPC codes exist
- The optimal parameters are known for long block lengths

Question:
What is the best performance for finite block lengths?

LDPC Codes [1]

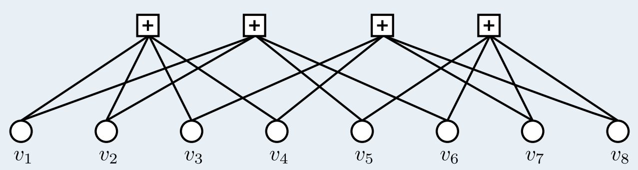
Definition: Low-density Parity-Check Code (Gallager,1962)
Low-density parity-check codes are codes specified by a matrix containing mostly 0's and only a small number of 1's.

$$\mathbf{H} = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Example: $(2,4)$ regular code

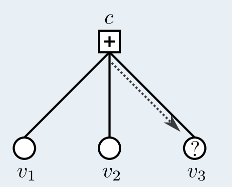
- **Regular** (l,r) codes:
  l ones in every column, respectively r ones in every row
- **Irregular** codes:
  Edge degree distributions described by polynomials
  $\lambda(x) = \sum_{i=1}^{d_v} \lambda_i x^{i-1}$, $\rho(x) = \sum_{i=1}^{d_c} \rho_i x^{i-1}$

Graphical Representation as Tanner Graph (Tanner,1981):

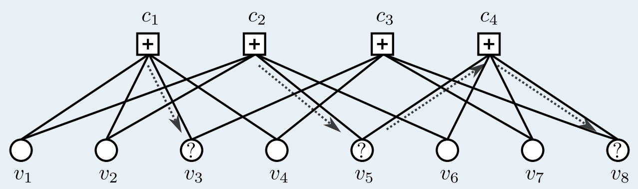


Properties
- **LDPC codes can reach capacity**
- **The decoding complexity stays linear**

Decoding

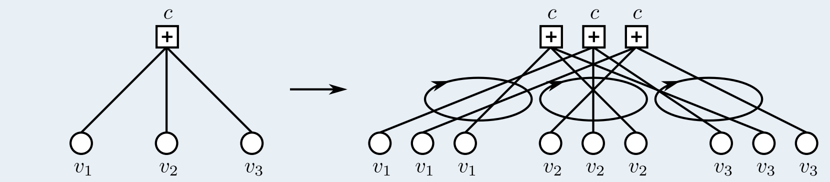


Check equation for check node c:
$$\sum_{k \in N(c)} v_k = 0 \mod 2$$



If variable nodes are erased due to the transmission over a binary erasure channels (BEC), they can be iteratively restored with the help of the knowledge of the rest of the graph of the code.

Protograph Based Construction [2]

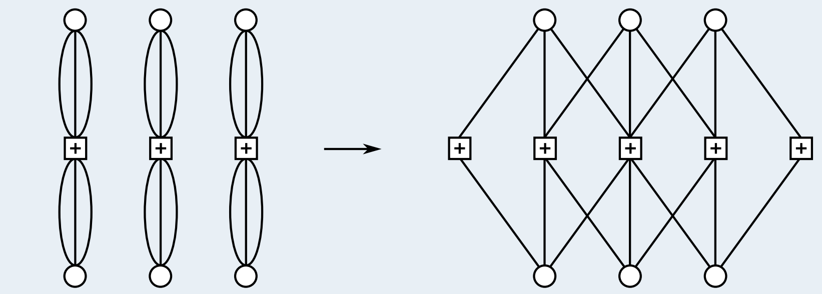


- Small Tanner graphs are used as a "blue print" of the structure
- This structure gets copied several times
- Similar connections are randomly permuted to obtain larger girths which avoids dependencies during the iterative decoding

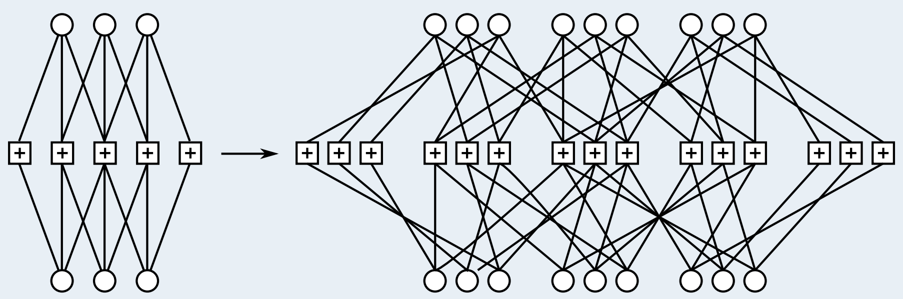
⇒ "copy-and-permute"

Advantages
- The protograph representation can be used for analysis

$(l,r,L)_{\mathcal{P}}$ Codes Based on Coupled Protographs [3]



1. Choose a simple (l,r) protograph
2. Couple L protographs to a spatially coupled protograph



3. Lift the coupled protograph with the "copy-and-permute" operation

The convolutional-like band matrix $\mathbf{H}$ consists of submatrices $\mathbf{H}_{i,j}$ which are permutation matrices for edge permutations:

$$\mathbf{H} = \begin{pmatrix} \mathbf{H}_{0,0} & \mathbf{H}_{0,1} & & & & & \\ \mathbf{H}_{1,0} & \mathbf{H}_{1,1} & \mathbf{H}_{0,0} & \mathbf{H}_{0,1} & & & \\ \mathbf{H}_{2,0} & \mathbf{H}_{2,1} & \mathbf{H}_{1,0} & \mathbf{H}_{1,1} & \mathbf{H}_{0,0} & \mathbf{H}_{0,1} & \\ & & \mathbf{H}_{2,0} & \mathbf{H}_{2,1} & \mathbf{H}_{1,0} & \mathbf{H}_{1,1} & \\ & & & & \mathbf{H}_{2,0} & \mathbf{H}_{2,1} & \ddots \end{pmatrix}$$

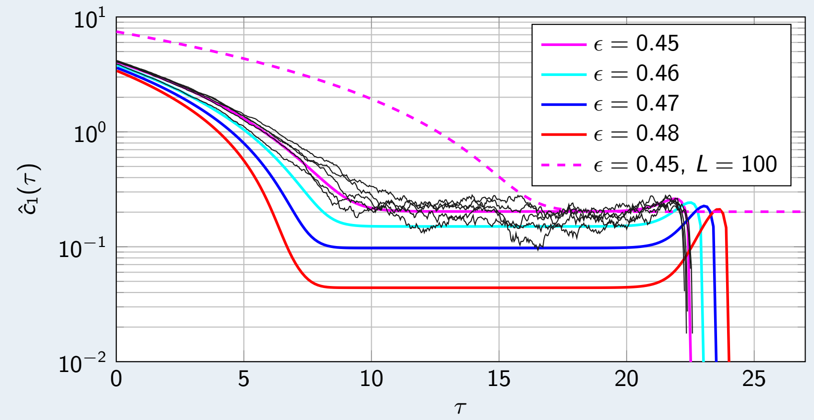
Advantages
- Systematic encoding is possible
- **The MAP threshold can be reached with iterative belief propagation (BP) decoding [4]**

Peeling Decoding

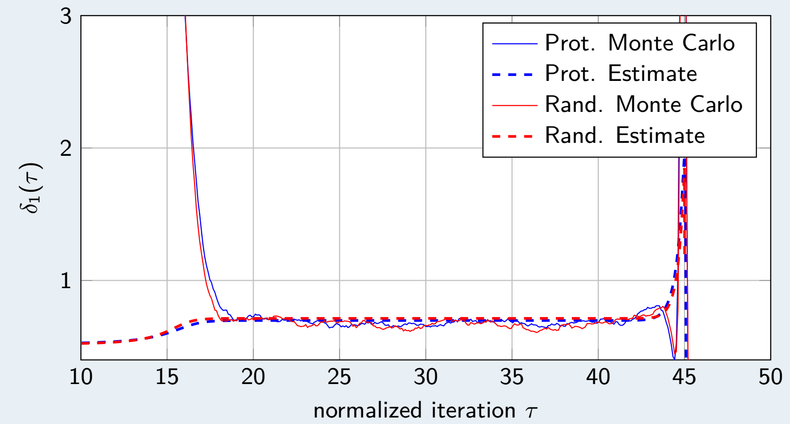
The decoding can only proceed if check nodes with only 1 unknown edge remain in the residual graph which is used as stability criterion.

- τ: Decoding iterations normalized by M
- $\hat{c}_1(\tau)$: Sum of mean of deg-1 check nodes normalized by M
  $$\hat{c}_1(\tau) \doteq \frac{1}{M}\sum_{i=1}^{m} \hat{R}(\mathbf{0}_{\sim i},\tau)$$
- $\delta_1(\tau)$: Variance of deg-1 check nodes of all processes
  $$\text{Var}[c_1(\tau)] = \frac{1}{M}\delta_1(\tau) = \frac{1}{M}\sum_{i=1}^{m}\sum_{b=1}^{m}\delta_{\mathbf{0}_{\sim i},\mathbf{0}_{\sim b}}$$
- $\phi_1(\tau,\zeta)$: process covariance with time
  $$\phi_1(\tau,\zeta) \doteq \mathbb{E}[c_1(\tau)c_1(\zeta)] - \hat{c}_1(\tau)\hat{c}_1(\zeta)$$

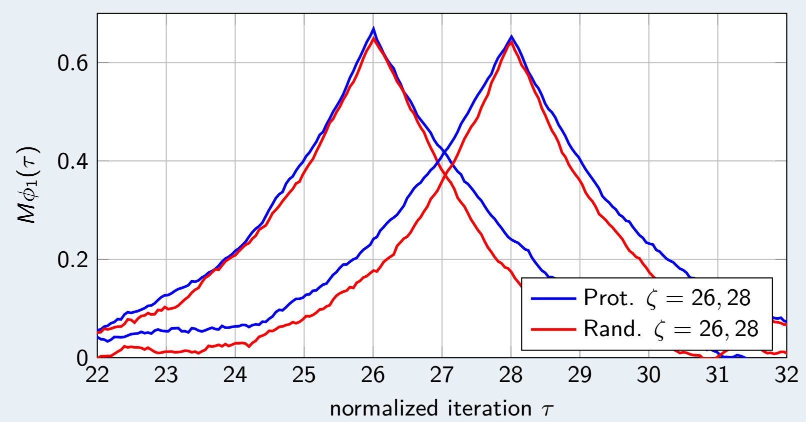
Mean Evolution of Deg-1 Check Nodes



Calculated $\hat{c}_1(\tau)$ for the $(l,r,L)_{\mathcal{P}} = (3,6,50)_{\mathcal{P}}$ ensemble for a varying ϵ. For $\epsilon = 0.45$, the subplot includes actual decoding trajectories.



Monte Carlo and the proposed estimates to $\delta_1(\tau)$ for the $(3,6,100)_{\mathcal{P}}$ and $(3,6,100)$ ensembles with $M = 2000$.



Process covariance estimation at 2 time instants for the same ensembles. All results are computed for $\epsilon = 0.45$.

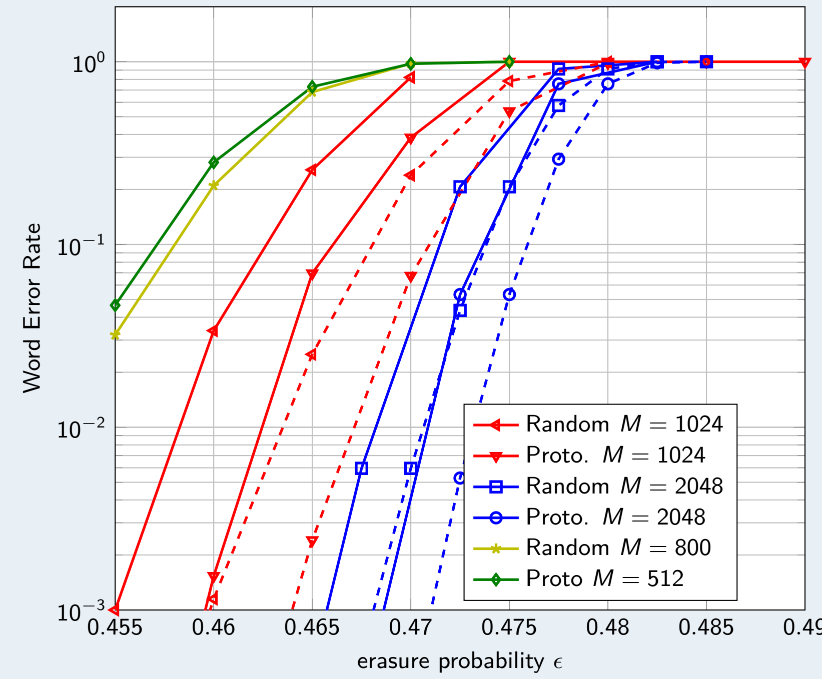
Conjecture of the Scaling Law [5]

Scaling laws stem from statistical physics where a system follows a control parameter in a very specific way around a phase transition. Around the threshold there holds a scaling law for LDPC codes using an iterative erasure decoder:

$$P^* \approx 1 - \exp\left(-\frac{(\epsilon L - \tau^*)}{\mu_0(M,\epsilon,l,r)}\right)$$

- $(\epsilon L - \tau^*)$ is the duration of the steady-state phase
- The average survival time μ_0 of $c_1(\tau)$ during the steady-state phase is a function of $\hat{c}_1(\tau)$, $\delta_1(\tau)$.

These parameters depend on the code ensemble.



Word error rate on the BEC for the $(l,r,L) = (3,6,100)$ and $(l,r,L)_{\mathcal{P}} = (3,6,100)_{\mathcal{P}}$ ensembles.

Protograph ensembles significantly improve the performance in the waterfall region. The scaling The resulting scaling law prediction matches the slope of the simulation results closely.

Outview and Future Tasks

- Can the complexity be reduced?
- Can we use this tool to design better codes?

References

[1] R. Gallager, "Low-Density Parity-Check Codes," *IEEE Trans. Inf. Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962. [Online].

[2] J. Thorpe, "Low-Density Parity-Check (LDPC) Codes Constructed from Protographs," JPL IPN, Tech. Rep., 2003. [Online].

[3] M. Lentmaier, G. P. Fettweis, K. S. Zigangirov, and D. J. Costello, "Approaching Capacity with Asymptotically Regular LDPC Codes," *Proc. Inf. Theory and Applicat. Workshop (ITA)*, pp. 173–177, 2009. [Online].

[4] S. Kudekar and T. J. Richardson, "Threshold Saturation via Spatial Coupling: Why Convolutional LDPC Ensembles Perform so well over the BEC," *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 1–29, 2010. [Online].

[5] P. Olmos, and R. L. Urbanke, "A Closed-Form Scaling Law for Convolutional LDPC Codes Over the BEC," *IEEE Inf. Theory Workshop (ITW)*, 2013. [Online].

Institute for Communications Engineering

Unterstützt von / Supported by
Alexander von Humboldt Stiftung/Foundation