Useful Mathematical Tools for Capacity Approaching Codes Design

Fulvio Babich, *Senior Member, IEEE,* Alessandro Soranzo, and Francesca Vatta, *Member, IEEE*

***Abstract*— Focus of this letter is the oldest class of codes that can approach the Shannon limit quite closely, i.e., low-density parity-check (LDPC) codes, and two mathematical tools that can make their design an easier job under appropriate assumptions. In particular, we present a simple algorithmic method to estimate the threshold for regular and irregular LDPC codes on memoryless binary-input continuous-output additive white Gaussian noise (AWGN) channels with sum-product decoding, and, to determine how close are the obtained thresholds to the theoretical maximum, i.e., to the Shannon limit, we give a simple and invertible expression of the AWGN channel capacity in the binary input-soft output case. For these codes, the thresholds are defined as the maximum noise level, such that an arbitrarily small bit-error probability can be achieved as the block length tends to infinity. We assume a Gaussian approximation for message densities under density evolution, a widely used simplification of the decoding algorithm.**

***Index Terms*— LDPC codes, threshold, Gaussian approximation, density evolution, sum-product algorithm, capacity approximation.**

I. Introduction

In HIS introduction to regular LDPC codes [1], Gallager first noticed that these codes exhibit the so called "threshold phenomenon" on binary symmetric channels (BSC), as explained also in [2]: "as the block length tends to infinity, an arbitrarily small bit-error probability can be achieved if the noise level is smaller than a certain threshold". Later Luby [3] showed that *irregular* LDPC codes, performing better than regular ones, exhibit the threshold phenomenon, too.

In [2] Chung et al. presented a method to estimate the threshold for regular and irregular LDPC codes on memoryless binary-input continuous-output AWGN (additive white Gaussian noise) channels with sum-product decoding. This algorithm is based on approximating message densities as Gaussians for regular LDPC codes or Gaussian mixtures for irregular LDPC codes. The thresholds are calculated as the last value such that a recurrent sequence converges, or, equivalently, as the first value such that the sequence diverges, but no mathematical methods are provided to determine these values, which can be found through an empirical trial and error procedure.

This letter presents a mathematical method to allow the noise thresholds evaluation, exploiting a moderately complex technique. This algorithm replaces the empirical procedure suggested in [2] for regular codes, extending its application to irregular codes, too. Our algorithm is based on the idea that the problem of determining the last value such that a recurrent sequence converges (or, equivalently, the first value such that the sequence diverges), can be seen as a "static" problem that can be assigned to a standard software, leading to a significant computational simplification. This algorithm has been derived using the quadratic degeneracy theory, thus transforming a recurrence relation convergence problem in a problem of mathematical analysis.

This result is important because in this way it is very simple to determine which features should have an LDPC code to satisfy the design constraints, or verify whether an LDPC code has performances close to the Shannon limits, or if there is room for improvement. Moreover, it allows the design of capacity approaching codes based on noise threshold maximization.

The letter also provides a simple and invertible expression of the soft-output binary-input capacity. This is provided in order to easily compare the performances of LDPC codes with the Shannon limits. However, this is not the only possible application of the expression we found. In fact, it may be applied in all system evaluations that do not assume a generic modulation scheme but a specific one, such as BPSK or QPSK.

The letter is organized as follows. In the next section, the message passing decoding algorithm is described and the concept of threshold is explained. Moreover, the Gaussian approximation of message densities is recalled for regular and irregular LDPC codes, respectively. In Sections III-A, III-B, and III-C the proposed method to determine the thresholds is described in terms of the algorithm, software, and numeric results, respectively. In Section IV a simple and invertible expression of the soft-output binary-input capacity is provided. In Section V the obtained thresholds are described in terms of their distance from the Shannon limit. Finally, Section VI concludes the letter and gives some hints about a possible follow up of this work.

II. Gaussian Approximation for LDPC Codes

Let us consider a regular (d_v, d_c)-LDPC code, where d_v and d_c are, respectively, the column and row degrees of the code parity matrix. Under belief-propagation decoding, variable nodes and check nodes exchange "messages" between each other iteratively. A check node gets messages from the variable nodes it is connected to ("neighbours"), processes the messages and sends the result back to its neighbouring variable nodes. Similarly, a variable node receives messages from its check nodes and returns the processed message back to them. This *two-step* procedure is repeated many times. After l iterations, the variable node decodes its associated bit based on all information obtained from its depth-l subgraph of neighbours.

Manuscript received April 21, 2017; accepted May 25, 2017. Date of publication June 12, 2017; date of current version September 8, 2017. This work is partly supported by the Italian Ministry of University and Research (MIUR) within the project FRA 2015 (University of Trieste, Italy), entitled "Peer-to-peer millimeter-wave communications in 5G networks: Theoretical modeling and algorithms for massive MIMO systems". The associate editor coordinating the review of this letter and approving it for publication was Z. Ma. *(Corresponding author: Francesca Vatta.)*

F. Babich and F. Vatta are with DIA, University of Trieste, 34127 Trieste, Italy (e-mail: babich@units.it; vatta@units.it).

A. Soranzo is with DMG, University of Trieste, 34127 Trieste, Italy (e-mail: soranzo@units.it).

In soft-decision belief-propagation decoding, the messages are the log-likelihood ratios (LLRs) of received bits. Denote with v the output message of a variable node and with u the output message of a check node. The two-step procedure described above, called *density evolution*, is used to calculate the thresholds of belief-propagation decoding, defined as the maximum noise level such that the probability of error tends to zero as the number of iterations tends to infinity [2], assuming that the all-0 codeword was sent and fixing the channel parameters, namely, the noise power.

Since in [2] it is shown that, without great sacrifice in accuracy, a one-dimensional quantity, namely the mean of Gaussian densities, can act as faithful surrogate for the message densities themselves, which is an infinite-dimensional vector, we assume that regular LDPC codes message distributions for AWGN channels are approximately Gaussian and denote the means of u and v by $m_u^{(l)}$ and $m_v^{(l)}$ at the l-th iteration, respectively. Moreover, the LLR message u_0 from the channel is assumed to be Gaussian with mean $m_{u_0} = 2/\sigma_n^2$ and variance $4/\sigma_n^2$, where σ_n^2 is the variance of the channel noise.

Density evolution can be generalized to irregular LDPC codes [4]. Consider an *ensemble* of random codes with edge-perspective variable-node and check-node degree distributions $\lambda(x)$ and $\rho(x)$, respectively [2]. The polynomials $\lambda(x)$ and $\rho(x)$ have degree $d_l - 1$ and $d_r - 1$, respectively. In an irregular LDPC code, the number of edges connecting to each node is not constant: hence, the messages a node receives from its neighbours are a "mixture" of densities from neighbour nodes of different degrees. We assume, as in [2], that irregular LDPC codes message distributions for AWGN channels are approximately Gaussian, based on the empirical evidence.

III. The Proposed Method

A. Algorithm

Following the analysis conducted in [2], we derive a simple operative algorithm to calculate the thresholds by means of the Gaussian approximation.

Instead of searching the last value of the parameter s granting the convergence of the sequence

$$t_l = g_s(t_{l-1}) \tag{1}$$

where s corresponds to m_{u_0} and t_l to $m_u^{(l)}$, i.e., instead of looking for the threshold s^*, defined in [2] as *the infimum of all s in $\mathbb{R}^+$ such that $t_l(s)$ converges to ∞ as $l \to \infty$* - which is subtly difficult - we solve a problem of *quadratic degeneracy* [5] we could see as "static", which can be assigned to a standard software.

Being $\phi(x)$ given in *Definition 1* in [2], where also an exponential approximation is given (see Eq. (8)) to easily calculate ϕ^{-1}, we define $g_s(t)$ as:

$$g_s(t) = \phi^{-1}\big(1 - \big[1 - \phi(s + (d_v - 1)t)\big]^{d_c - 1}\big) \tag{2}$$

for regular LDPC codes and

$$g_s(t) = \sum_{j=2}^{d_r} \rho_j \phi^{-1}\big(1 - \big[1 - \sum_{i=2}^{d_l} \lambda_i \phi(s + (i-1)t)\big]^{j-1}\big) \tag{3}$$

for irregular LDPC codes.

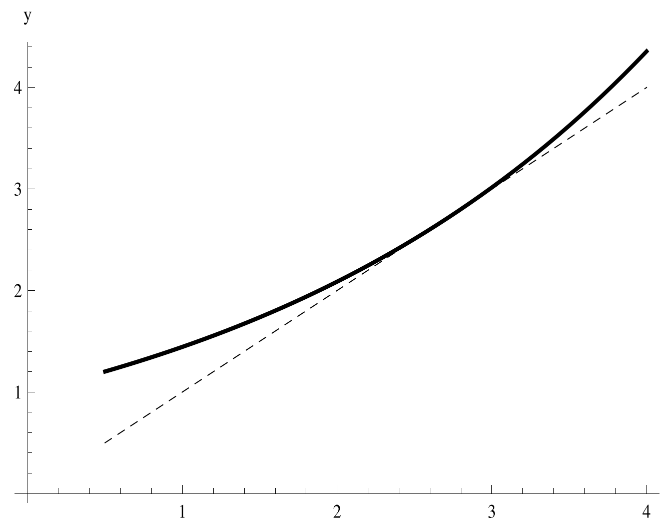

Fig. 1. Graphical representation of (4). Solid curve: $y = g_s(t)$; dashed curve: $y = t$.

For $g_s''(t) \neq 0$ the "static" equivalent task is the solution of the system of equations

$$\boxed{\begin{cases} g_s(t) = t \\ g_s'(t) = 1 \end{cases}} \tag{4}$$

The first equation expresses the fact that t is a fixed point of g_s, the second one the fact that the graph of $g_s(t)$ is tangent to the line $y = t$, bisector of I quadrant (Fig. 1).

B. Software

To implement the algorithm of previous section, the Mathematica$^{(R)}$ software has been used. We report here the instruction set we produced for a regular binary (j, k)-LDPC code, where j denotes the number of neighbors of a variable node and k the number of neighbors of a check node:

```
σ[j_,k_]:=(
α = -0.4527;
β = 0.0218;
γ = 0.86;
φ[x_]:=Exp[α x^γ + β];
iφ[y_]:=((Log[y] - β)/α)^(1/γ);
g[t_,s_]:=iφ[1-(1-(φ[s+(j-1)t]))^(k-1)];
s0 = 2.6;
s∞ = s/.FindRoot[{D[g[t,s],t]-1,
g[t,s]-t},{{s,s0},{t,1}}];
p = Sqrt[2/s∞];
Return[p] )
```

Let us explain the Mathematica$^{(R)}$ script line by line.

The function returns the value $\sigma(j, k) = p = \sqrt{\frac{2}{s^*}}$, the approximate threshold σ_{GAnew}, reported in Table I for some values of j and k, each with computational time of 2.6 ms.[1]

[1] In this research we have also produced a Mathematica$^{(R)}$ script implementing the algorithm sketched in [2], obtaining computational times at least one order of magnitude higher.

TABLE I

APPROXIMATE THRESHOLD VALUES FOR VARIOUS (j,k)-REGULAR LDPC CODES

j	k	rate	$\sigma_{\rm GA}$	$\sigma_{\rm exact}$	error[dB]	$\sigma_{\rm GAnew}$
3	6	0.5	0.8747	0.8809	0.06	0.8748
4	8	0.5	0.8323	0.8376	0.06	0.8324
5	10	0.5	0.7910	0.7936	0.03	0.7911
3	5	0.4	1.0003	1.0093	0.08	1.0003
4	6	1/3	1.0035	1.0109	0.06	1.0036
3	4	0.25	1.2517	1.2667	0.10	1.2518
4	10	0.6	0.7440	0.7481	0.05	0.7440
3	9	2/3	0.7051	0.7082	0.04	0.7051
3	12	0.75	0.6297	0.6320	0.03	0.6298

The coefficients α, β and γ define the approximation $\phi(x)$, called `ϕ[x_]` in the Mathematica$^{(R)}$ script, given in Eq. (8) of [2], and its inverse $\phi^{-1}(y)$, called `iϕ[y_]`:

$$\phi(x) := e^{\alpha x^{\gamma}+\beta} \quad \alpha=-0.4527 \; \beta=0.0218 \; \gamma=0.86 \quad (5)$$

$$\phi^{-1}(y) = \Big(\frac{\log y-\beta}{\alpha}\Big)^{1/\gamma} \quad (6)$$

The informatic definition of $g_s(t)$, called `g[t_,s_]` in the script, follows (2).

The constant `s0` is an initial value for the internal algorithm of Mathematica$^{(R)}$ to solve the system (4).

The system (4) has one solution (t^*, s^*) and `s∞` gives s^*. The threshold, in terms of noise power, is equal to $\frac{2}{s^*}$. The instruction `FindRoot[{D[g[t,s],t]-1,g[t,s]-t}` searches for the solutions (zeros) of the system

$$\begin{cases} g'_s(t)-1=0 \\ g_s(t)-t=0 \end{cases} \quad (7)$$

equivalent to (4). Of course, `D[g[t,s]t]` is the derivative $g'_s(t)$.

For irregular LDPC codes, defined by the polynomials $\lambda(x)$ and $\rho(x)$, the informatic definition of $g_s(t)$ follows (3)

```
g[t_,s_] := Sum[ρ[[j]]iϕ[1-(1-(Sum[λ[[i]]ϕ]
            [s+(i-1)t],{i,2,dl}]))^(j-1)],
            {j,2,dr}];
```

and the software returns the threshold $\sigma_{\rm GAnew}$ reported in Table II for some good rate-1/2 codes listed in [6].

C. Numeric Results

With regard to regular LDPC codes, Table I reports in the first 6 columns the values of Table I in [2] and in the 7-th the corresponding values $\sigma_{\rm GAnew}=\sigma(j,k)$ which were computed with the software given in the previous section. In the first 6 columns of Table I are reported the regular LDPC rates $r=1-j/k$, the thresholds $\sigma_{\rm GA}$ evaluated in [2] using the Gaussian approximation, the exact thresholds $\sigma_{\rm exact}$ evaluated in [2] using the algorithms developed in [6], and the errors in dB between the approximated and exact thresholds.

With regard to irregular LDPC codes, for the irregular rate-1/2 LDPC (17) of [2], after defining the arrays with the chosen values of λ_i and ρ_j, reported in Table II with $d_l=20$ and $d_r=9$, the software of previous section gives $\sigma_{\rm GAnew}=\sqrt{\frac{2}{s^*}}=0.947304$, in good accordance with the result 0.9473 of [2]. To validate the proposed method, we report in Table II the $\sigma_{\rm GAnew}$ values also for some good rate-1/2 codes listed in [6].

TABLE II

PARAMETERS OF GOOD RATE-1/2 CODES LISTED IN [6] WITH $d_l=20$, 30, AND 40

d_l	20		30		40	
	i	λ_i	i	λ_i	i	λ_i
	2	0.234029	2	0.209626	2	0.200642
	3	0.212425	3	0.212527	3	0.189227
	6	0.146898	6	0.013831	6	0.106155
	7	0.102840	7	0.079586	7	0.091511
	20	0.303808	8	0.195912	13	0.157193
			30	0.288518	40	0.255272
d_r	9		9		10	
	j	ρ_j	j	ρ_j	j	ρ_j
	8	0.71875	8	0.04370	9	0.56860
	9	0.28125	9	0.95630	10	0.43140
$\sigma_{\rm exact}$	0.96693		0.97068		0.97265	
$\sigma_{\rm GAnew}$	0.9473		0.9502		0.9512	
$\mathrm{SNR_{gap}}$	0.2754		0.2488		0.2397	
$\mathrm{SNR_{gap-spline}}$	0.2832		0.2563		0.2472	

IV. BINARY-INPUT SOFT-OUTPUT CAPACITY APPROXIMATION

Considering an AWGN channel with binary input and soft output, the channel capacity is defined as [7]

$$C(\gamma) := (\log_2 e) \times\Big(2\gamma-\frac{1}{\sqrt{\pi}}\int_{-\infty}^{+\infty} e^{-(y-\sqrt{\gamma})^2}\ln\cosh(2y\sqrt{\gamma})\mathrm{d}y\Big) \quad (8)$$

where γ is the symbol signal-to-noise ratio.

The problem of finding an invertible approximation for the function $C(\gamma)$ is important since it allows to determine the estimated gap of the approximate thresholds we found from the Shannon limit. This evaluation can be useful to identify good and bad codes and get design guidelines for them.

We obtained the following couple of functions, one inverse of the other, approximating $C(\gamma)$ and $C^{-1}(r)$, with forms, respectively,

$$C(\gamma)\simeq 1-e^{u\gamma^{w}+v} \quad C^{-1}(r)\simeq\Big(\frac{\log(1-r)-v}{u}\Big)^{\frac{1}{w}} \quad (9)$$

We propose these values:

$$u=-1.286 \quad v=0.01022 \; w=0.9308 \quad (10)$$

for which we found these limitations for the absolute and relative errors $\forall\gamma\in[0.1,3.35]$ and $\forall r\in[0.13,0.98]$

$$C_1(\gamma) := 1-e^{u\gamma^{w}+v}\simeq C(\gamma)\begin{cases}|\varepsilon(\gamma)|<0.0009\\ \\ |\varepsilon_r(\gamma)|<0.0021\end{cases} \quad (11)$$

$$C_1^{-1}(r) := \Big(\frac{\log(1-r)-v}{u}\Big)^{\frac{1}{w}}\simeq C^{-1}(r)\begin{cases}|\varepsilon(r)|<0.008\\ \\ |\varepsilon_r(r)|<0.0024\end{cases} \quad (12)$$

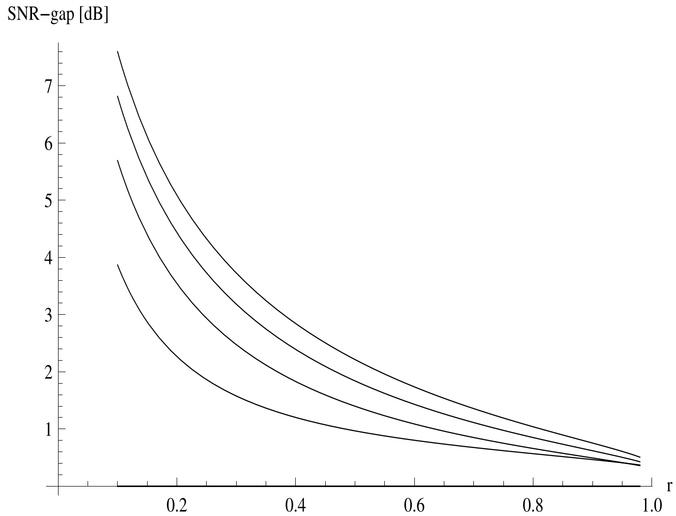

Fig. 2. $\mathrm{SNR}_{\mathrm{gap}}$ [dB] vs. the code rate $r = 1 - j/k$ for some (j,k)-regular LDPC codes: $j = 3$ (lower), 4, 5, 6 (upper).

The function $C_1(\gamma)$ approximates $C(\gamma)$ very accurately even if it is not correct for γ very close to 0, due to the positive value of the parameter υ.

V. Distance From the Shannon Limit

Fig. 2 reports $\mathrm{SNR}_{\mathrm{gap}}$ in dB, the distance from the Shannon limit in dB of the approximate thresholds we found - which, in terms of noise power, are equal to $(2\sigma^2_{\mathrm{GAnew}}(j,k))^{-1}$ - for some (j,k)-regular LDPC codes. This graph rebuilds and extends the analogous graph reported in [2].

To calculate this gap the following equation has been used:

$$\mathrm{SNR}_{\mathrm{gap}}(j,r) := \frac{(2\sigma^2_{\mathrm{GAnew}}(j,k))^{-1}}{C_1^{-1}(r)} \tag{13}$$

where r is the code rate, $r = 1 - j/k$ and $C_1^{-1}(r)$, the inverse of the channel capacity evaluated in r, is the symbol signal-to-noise ratio γ for which $C_1(\gamma) = r$, i.e., the Shannon limit.

To evaluate $C^{-1}(r)$, a spline interpolation can be used, too, derived in the Appendix.

As far as good rate-1/2 irregular codes, listed in Table II, are concerned, the Shannon limit we obtain using our approximation $C_1^{-1}(1/2)$ is $\sigma_{\mathrm{Shannon1}} = 0.977813$, in good accordance with [6]. Using the spline interpolation reported in the Appendix we obtain $\sigma_{\mathrm{Shannon-spline}} = 0.978696$. In terms of noise power, the Shannon limit is given by $C_1^{-1}(1/2) = (2\sigma^2_{\mathrm{Shannon1}})^{-1} = 0.522948$ and $C^{-1}_{\mathrm{spline}}(1/2) = (2\sigma^2_{\mathrm{Shannon-spline}})^{-1} = 0.522005$. In the last two rows of Table II are reported also the $\mathrm{SNR}_{\mathrm{gap}}$ and $\mathrm{SNR}_{\mathrm{gap-spline}}$ values in dB, given by

$$\mathrm{SNR}_{\mathrm{gap}}(d_l) := \frac{(2\sigma^2_{\mathrm{GAnew}}(d_l))^{-1}}{C_1^{-1}(1/2)} \tag{14}$$

and

$$\mathrm{SNR}_{\mathrm{gap-spline}}(d_l) := \frac{(2\sigma^2_{\mathrm{GAnew}}(d_l))^{-1}}{C^{-1}_{\mathrm{spline}}(1/2)} \tag{15}$$

where the σ_{GAnew} values are those listed in Table II.

VI. Conclusions

In this letter, we provided an operative, though simple, algorithm to determine the decoding thresholds of regular and irregular LDPC codes over memoryless binary-input continuous-output AWGN channels with sum-product decoding. This algorithm was derived using the quadratic degeneracy theory, thus transforming a recurrence relation convergence problem in a calculus problem we could see as "static". This method can be used as a design tool which can be employed towards finding degree distributions that approach the Shannon capacity limit. This is a part of our ongoing research.

In addition, we provided some approximations for the calculation of the channel capacity, leading to approximate decoding thresholds distances from the Shannon capacity limit. The approximations provide reasonable estimates of these distances, with less complexity with respect to the exact calculation of the channel capacity.

Appendix
Approximation of $C(\gamma)$ and $C^{-1}(r)$

The function $C(\gamma)$, defined in (8) by an integral from $-\infty$ to $+\infty$, is approximated in Mathematica$^{(R)}$ by a numeric integration (NIntegrate) from $-200 + \sqrt{\gamma}$ to $200 + \sqrt{\gamma}$:

```
c[γ_] := Log[2,E](2γ-(1/Sqrt[Pi])NIntegrate
        [Exp[-(y-Sqrt[γ])^2]Log
        [Cosh[2ySqrt[Gamma]]],
        {y,-200+Sqrt[γ],200+Sqrt[γ]}])
```

To implement an approximation of the inverse $C^{-1}(r)$ we used a spline interpolation $C^{-1}_{\mathrm{spline}}(r)$ based on 66 points $(\gamma_i, C(\gamma_i)) = (C^{-1}(r_i), r_i)$ with $\gamma_i = 0.1, ..., 3.35$ with steps 0.05, corresponding to $0.1314 \le r_i \le 0.9806$.

References

[1] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inf. Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962.

[2] S.-Y. Chung, T. J. Richardson, and R. L. Urbanke, "Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 657–670, Feb. 2001.

[3] M. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. Spielman, "Analysis of low density codes and improved designs using irregular graphs," in *Proc. 30th Annu. ACM Symp. Theory Comput.*, May 1998, pp. 249–258.

[4] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 619–637, Feb. 2001.

[5] J. K. Hale and H. Koçak, *Dynamics and Bifurcations*. New York, NY, USA: Springer-Verlag, 1991.

[6] S.-Y. Chung, "On the construction of some capacity-approaching coding schemes," Ph.D. dissertation, Dept. Elect. Eng., Massachusetts Inst. Technol., Cambridge, MA, USA, 2000. [Online]. Available: http://dspace.mit.edu/handle/1721.1/8981

[7] F. Babich, "A sphere-packing exponent approximation," in *Proc. VTC-Spring*, Dresden, Germany, Jun. 2013, pp. 2–5.