MODERN CODING THEORY – FIGURES

Modern Coding Theory – Figures

BY

T. RICHARDSON AND R. URBANKE

Cambridge University Press

Modern Coding Theory – Figures

Library of Congress Catalog Card Number: 00–00000

ISBN 0-000-00000-0

LIST OF FIGURES

Figure 0.1: Basic point-to-point communications problem.

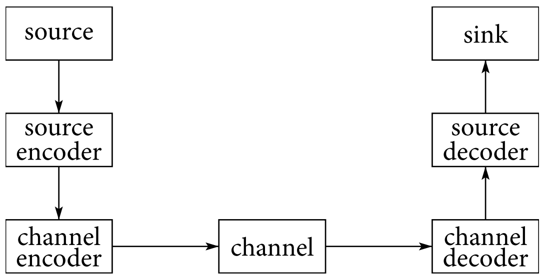


Figure 0.2: Basic point-to-point communications problem in view of the source-channel separation theorem.

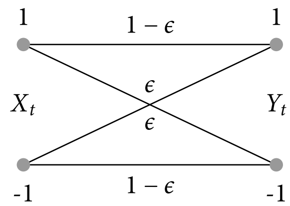


Figure 0.3: BSC(ϵ).

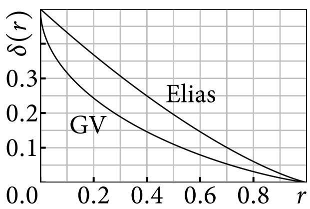


Figure 0.4: Upper and lower bound on $\delta^*(r)$.

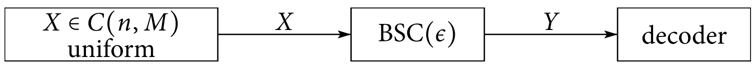


Figure 0.5: Transmission over the BSC(ϵ).

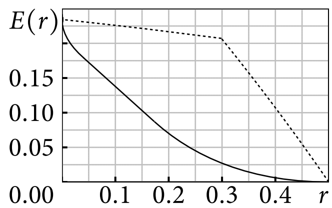


Figure 0.6: Error exponent of block codes (solid line) and of convolutional codes (dashed line) for the BSC($\epsilon \approx 0.11$).

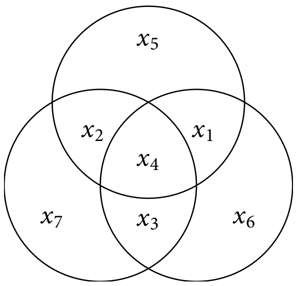


Figure 0.7: Venn diagram representation of C.

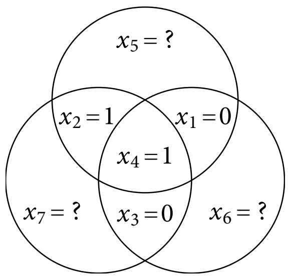


Figure 0.8: Encoding corresponding to $(x_1, x_2, x_3, x_4) = (0, 1, 0, 1)$. The number of ones contained in each circle must be even. By applying one such constraint at a time the initially unknown components x_5, x_6, and x_7 can be determined. The resulting codeword is $(x_1, x_2, x_3, x_4, x_5, x_6, x_7) = (0, 1, 0, 1, 0, 1, 0)$.

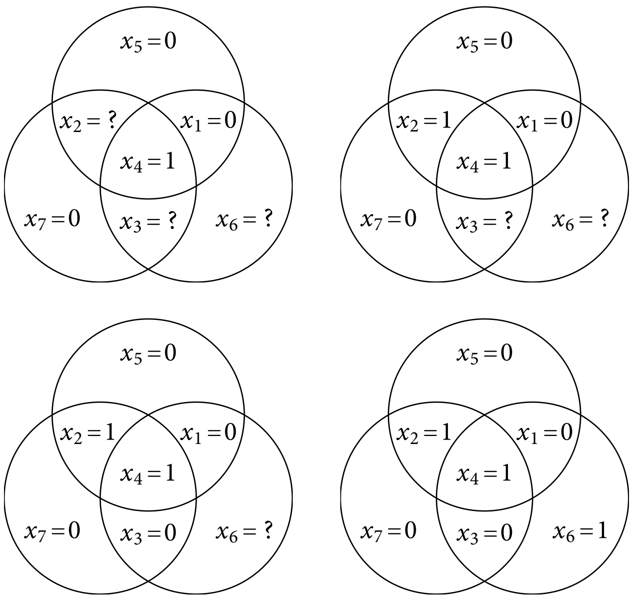


Figure 0.9: Decoding corresponding to the received message $(0, ?, ?, 1, 0, ?, 0)$. First we recover $x_2 = 1$ using the constraint implied by the top circle. Next we determine $x_3 = 0$ by resolving the constraint given by the left circle. Finally, using the last constraint, we recover $x_6 = 1$.

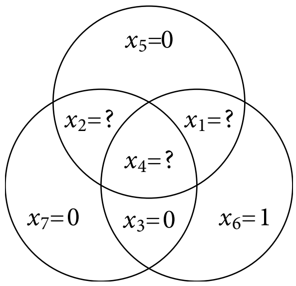


Figure 0.10: Decoding corresponding to the received message $(?, ?, 0, ?, 0, 1, 0)$. The local decoding fails since none of the three parity-check equations by themselves can resolve any ambiguity.

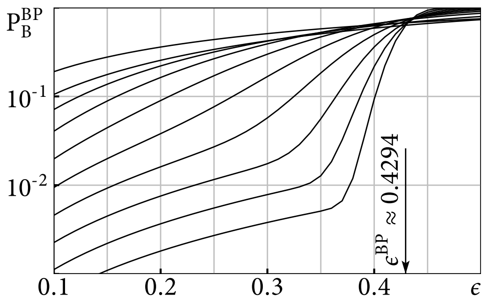


Figure 0.11: $\mathbb{E}_{\mathrm{LDPC}\left(nx^3, \frac{n}{2}x^6\right)}\left[\mathrm{P}_{\mathrm{B}}^{\mathrm{BP}}(\mathtt{G}, \epsilon)\right]$ as a function of ϵ for $n = 2^i$, $i \in [10]$.

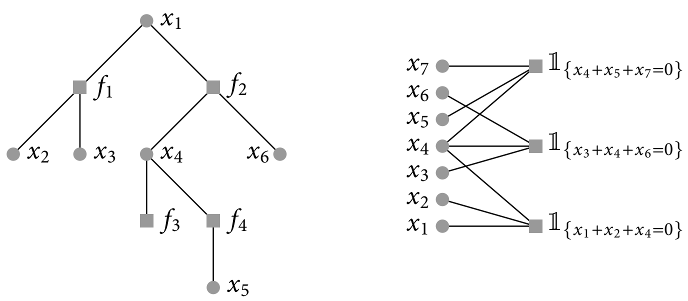


Figure 0.12: Left: Factor graph of f given in Example 2.2. Right: Factor graph for the code membership function defined in Example 2.5.

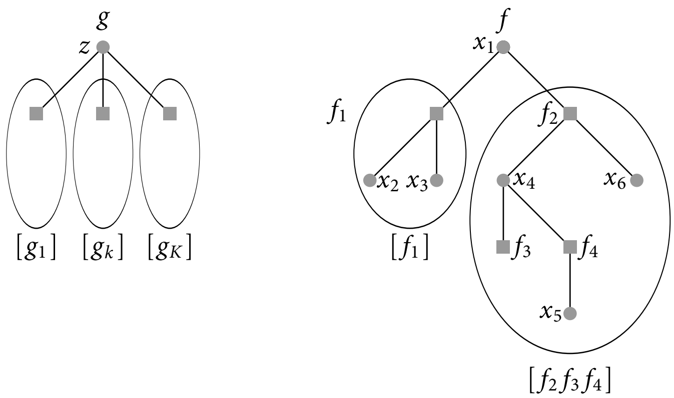


Figure 0.13: Generic factorization and the particular instance.

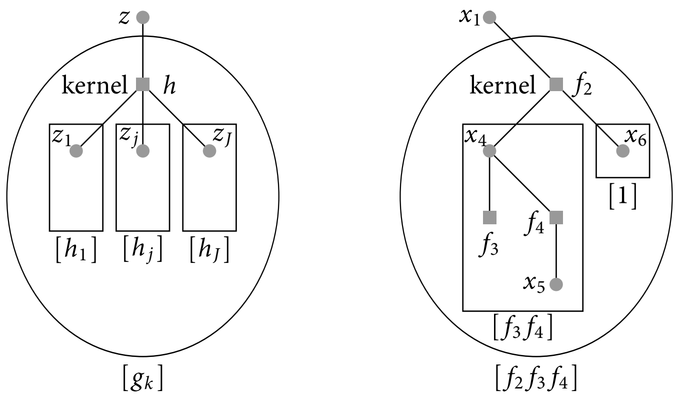


Figure 0.14: Generic factorization of g_k and the particular instance.

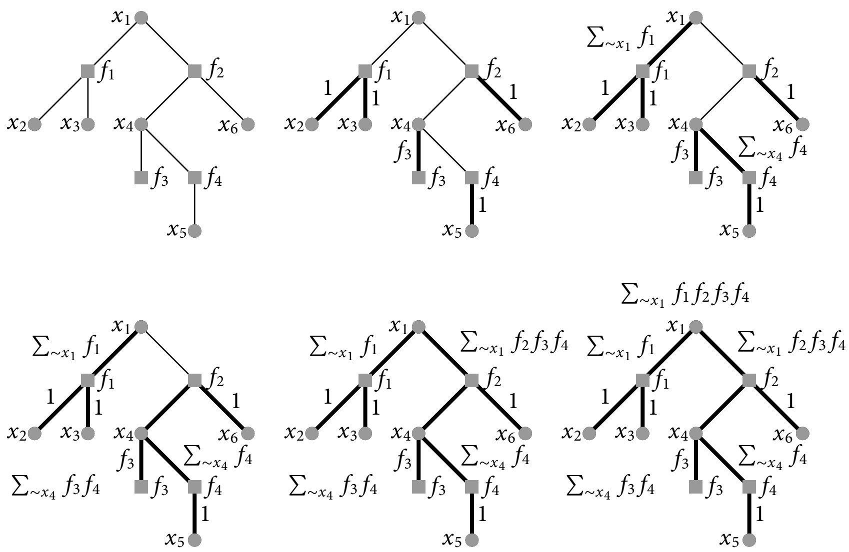


Figure 0.15: Marginalization of function f from Example 2.2 via message passing. Message passing starts at the leaf nodes. A node that has received messages from all its children processes the messages and forwards the result to its parent node. Bold edges indicate edges along which messages have already been sent.

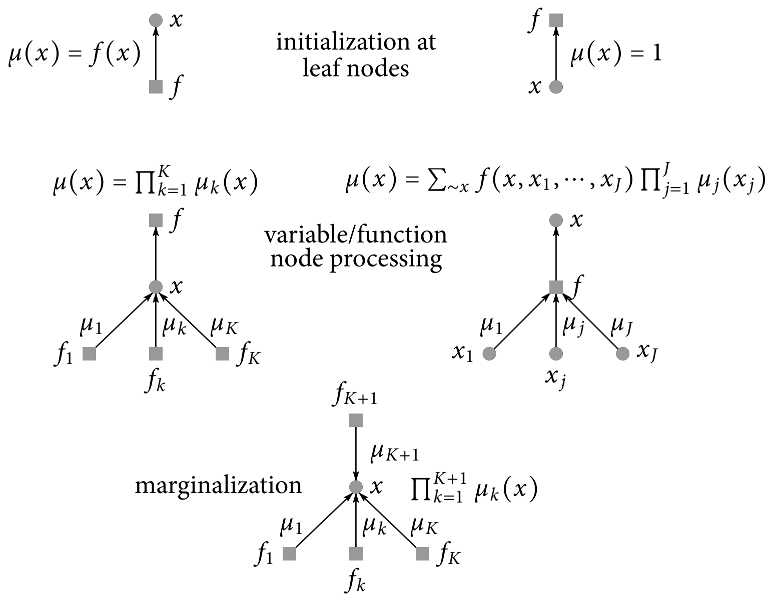


Figure 0.16: Message-passing rules. The top row shows the initialization of the messages at the leaf nodes. The middle row corresponds to the processing rules at the variable and function nodes, respectively. The bottom row explains the final marginalization step.

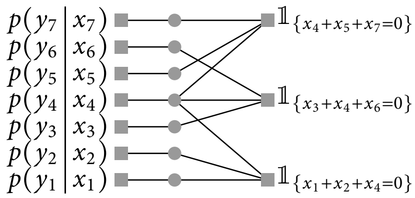


Figure 0.17: Factor graph for the MAP decoding of our running example.

Figure 0.18: Left: Standard FG in which each variable node has degree at most 2. Right: Equivalent FSFG. The variables in the FSFG are associated with the edges in the FG.

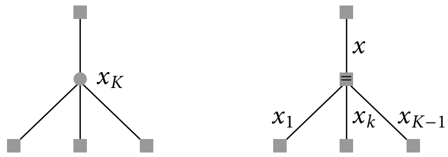


Figure 0.19: Representation of a variable node of degree K as an FG (left) and the equivalent representation as an FSFG (right).

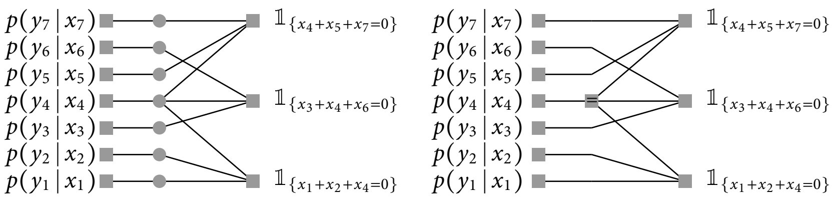


Figure 0.20: Standard FG and the corresponding FSFG for the MAP decoding problem of our running example.

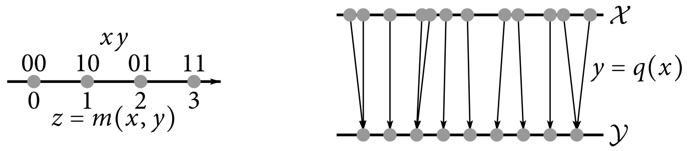


Figure 0.21: Left: Mapping $z = m(x, y)$. Right: Quantizer $y = q(x)$.

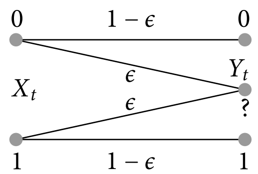


Figure 0.22: Binary erasure channel with parameter ϵ.

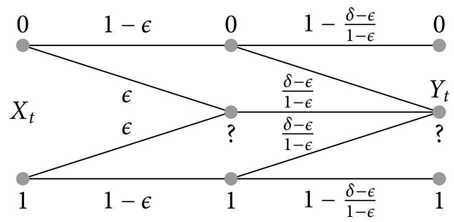


Figure 0.23: For $\epsilon \leq \delta$, the BEC(δ) is degraded with respect to the BEC(ϵ).

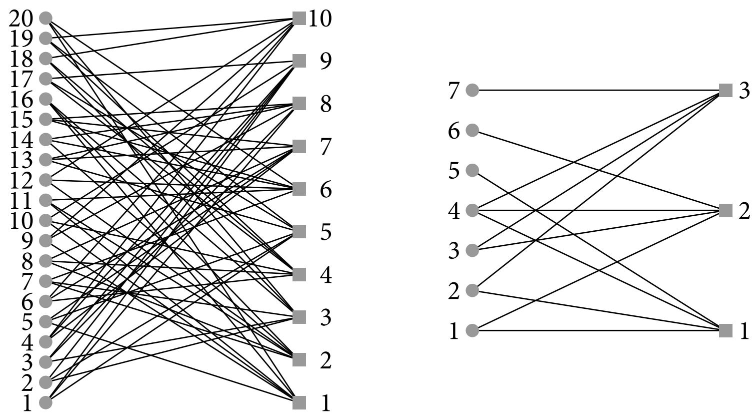


Figure 0.24: Left: Tanner graph of H given in (3.9). Right: Tanner graph of $[7, 4, 3]$ Hamming code corresponding to the parity-check matrix on page 15. This graph is discussed in Example 3.11.

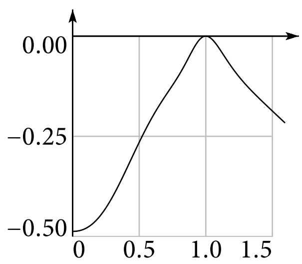


Figure 0.25: Function $\Psi(y)$.

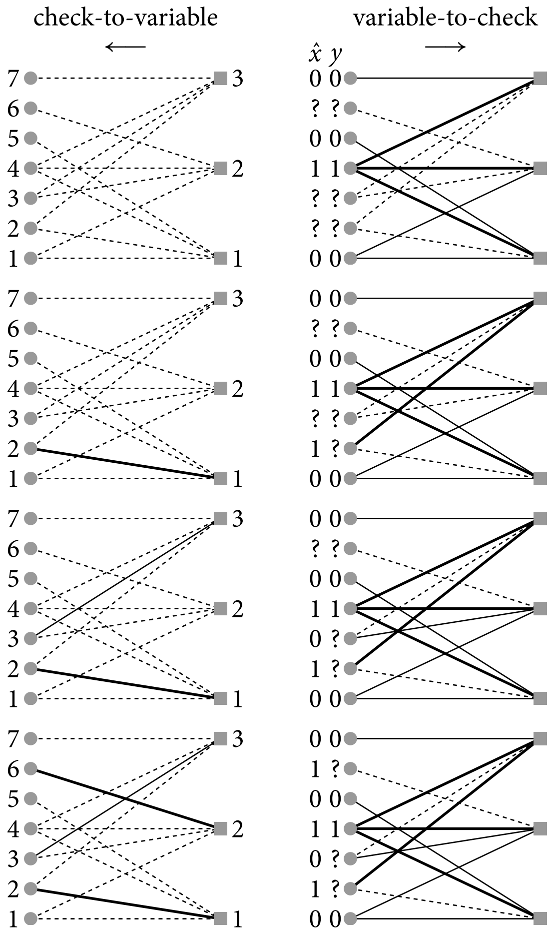


Figure 0.26: Message-passing decoding of the $[7, 4, 3]$ Hamming code with the received word $y = (0, ?, ?, 1, 0, ?, 0)$. The vector $\hat{x}$ denotes the current estimate of the transmitted word x. A 0 message is indicated as thin line, a 1 message is indicated as thick line, and a ? message is drawn as dashed line. The four rows correspond to iterations 0 to 3. After the first iteration we recover $x_2 = 1$, after the second $x_3 = 0$, and after the third we know that $x_6 = 1$. The recovered codeword is $x = (0, 1, 0, 1, 0, 1, 0)$.

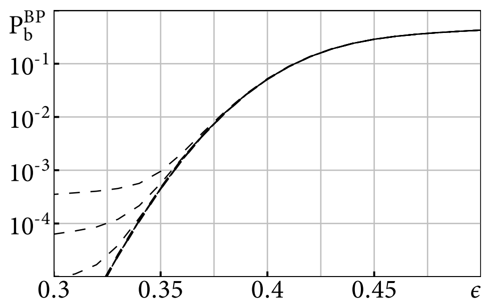


Figure 0.27: Bit erasure probability of 10 random samples from $\text{LDPC}\left(512x^3, 256x^6\right) \mathrel{\hat{=}} \text{LDPC}\left(512, x^2, x^5\right)$.

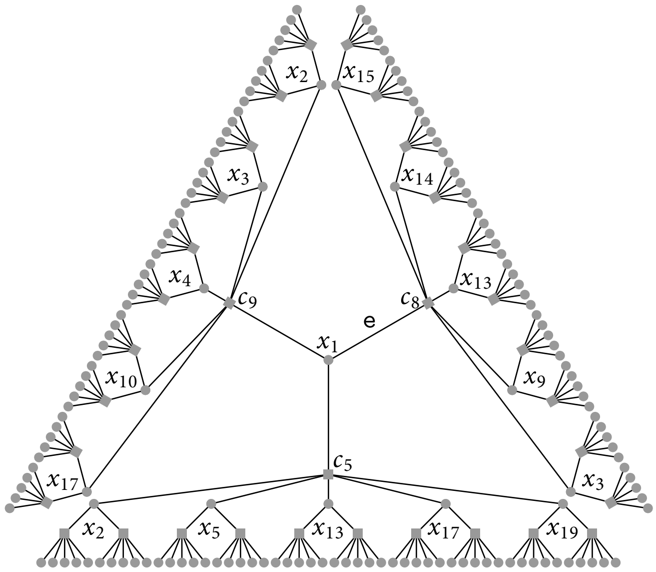


Figure 0.28: Computation graph of height 2 (two iterations) for bit x_1 and the code $C(H)$ for H given in (3.9). The computation graph of height 2 (two iterations) for edge e is the subtree consisting of edge e, variable node x_1, and the two subtrees rooted in check nodes c_5 and c_9.

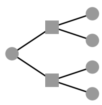
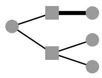
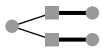
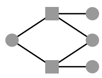
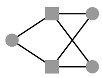
T	$\mathbb{P}\{\mathtt{T} \in \mathring{\mathcal{C}}_1(n, x, x^2)\}$	$\mathrm{P}_{\mathrm{b}}^{\mathrm{BP}}(\mathtt{T}, \epsilon)$
	$\frac{(2n-6)(2n-8)}{(2n-1)(2n-5)}$	$\epsilon(1-(1-\epsilon)^2)^2$
	$\frac{2(2n-6)}{(2n-1)(2n-5)}$	$\epsilon^2(1-(1-\epsilon)^2)$
	$\frac{1}{(2n-1)(2n-5)}$	ϵ^3
	$\frac{4(2n-6)}{(2n-1)(2n-5)}$	$\epsilon^2+\epsilon^3(1-\epsilon)$
	$\frac{2}{(2n-1)(2n-5)}$	$\epsilon(1-(1-\epsilon)^2)$
	$\frac{2}{2n-1}$	ϵ^2

Figure 0.29: Elements of $\mathring{\mathcal{C}}_1(n, \lambda(x) = x, \rho(x) = x^2)$ together with their probabilities $\mathbb{P}\{\mathtt{T} \in \mathring{\mathcal{C}}_1(n, x, x^2)\}$ and the conditional probability of error, $\mathrm{P}_{\mathrm{b}}^{\mathrm{BP}}(\mathtt{T}, \epsilon)$. Thick lines indicate double edges.

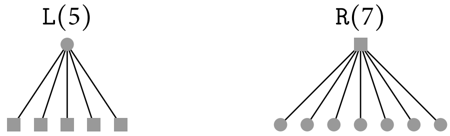


Figure 0.30: Examples of basic trees, L(5) and R(7).

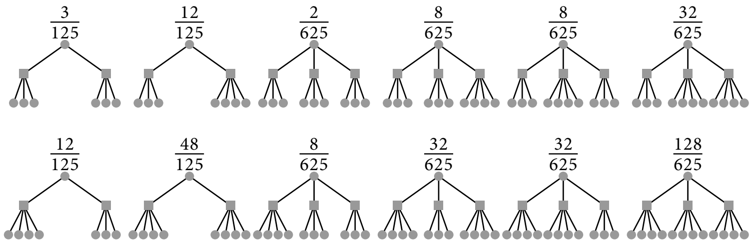


Figure 0.31: Twelve elements of $\mathring{\mathcal{T}}_1(\lambda, \rho)$ together with their probabilities.

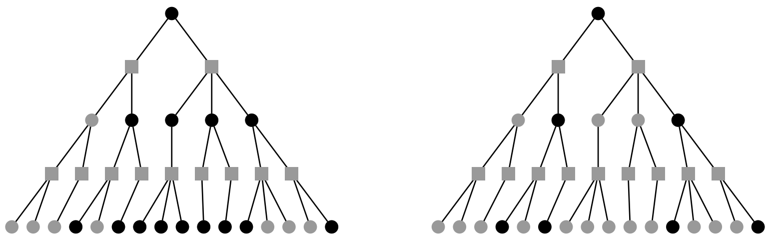


Figure 0.32: Left: Element of $C^1(\mathtt{T})$ for a given tree $\mathtt{T} \in \mathcal{T}_2$. Right: Minimal such element, i.e., an element of $C^1_{\min}(\mathtt{T})$. Every check node has either no connected variables of value 1 or exactly two such neighbors. Black and gray circles indicate variables with associated values of 1 or 0, respectively.

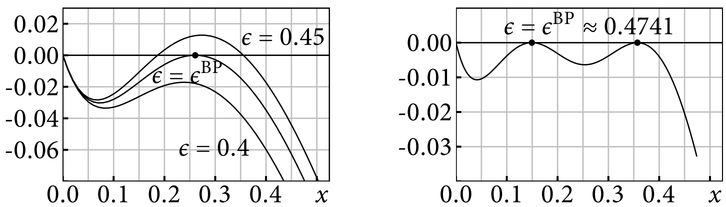


Figure 0.33: Left: Graphical determination of the threshold for $(\lambda, \rho) = (x^2, x^5)$. There is one critical point, $x^{BP} \approx 0.2606$ (black dot). Right: Graphical determination of the threshold for optimized degree distribution described in Example 3.63. There are two critical points, $x^{BP}_{1,2} \approx 0.1493, 0.3571$ (two black dots).

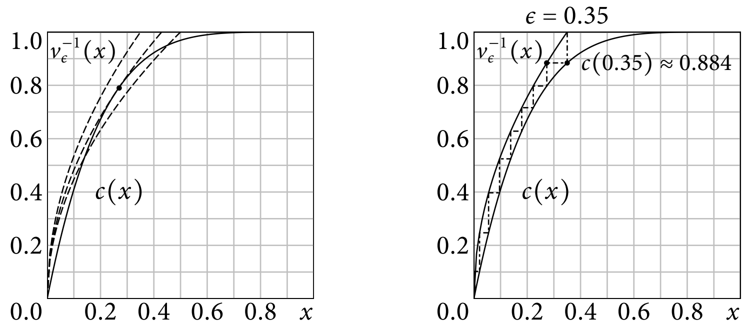


Figure 0.34: Left: Graphical determination of the threshold for $(\lambda(x) = x^2, \rho(x) = x^5)$. The function $v_\epsilon^{-1}(x) = (x/\epsilon)^{1/2}$ is shown as a dashed line for $\epsilon = 0.35$, $\epsilon = \epsilon^{\mathrm{BP}} \approx 0.42944$, and $\epsilon = 0.5$. The function $c(x) = 1 - (1 - x)^5$ is shown as a solid line. Right: Evolution of the decoding process for $\epsilon = 0.35$. The initial fraction of erasure messages emitted by the variable nodes is $x = 0.35$. After half an iteration (at the output of the check nodes) this fraction has evolved to $c(x = 0.35) \approx 0.88397$. After one full iteration, i.e., at the output of the variable nodes, we see an erasure fraction of $x = v_\epsilon(0.88397)$, i.e., x is the solution to the equation $0.883971 = v_\epsilon^{-1}(x)$. This process continues in the same fashion for each subsequent iteration, corresponding graphically to a *staircase* function which is bounded below by $c(x)$ and bounded above by $v_\epsilon^{-1}(x)$.

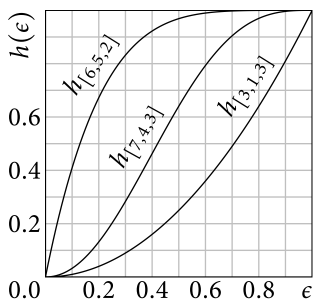


Figure 0.35: EXIT function of the $[3, 1, 3]$ repetition code, the $[6, 5, 2]$ parity-check code, and the $[7, 4, 3]$ Hamming code.

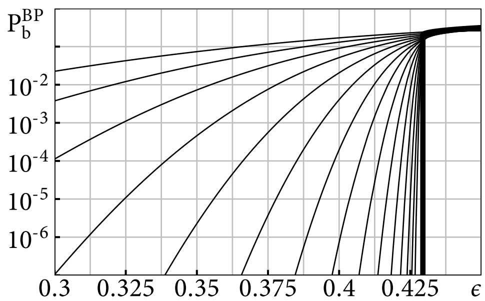


Figure 0.36: $\mathbb{E}_{\text{LDPC}(n,x^2,x^5)}\left[\text{P}_\text{b}^{\text{BP}}(\texttt{G},\epsilon,\ell=\infty)\right]$ as a function of ϵ for $n = 2^i$, $i = 6,\dots,20$. Also shown is the limit $\mathbb{E}_{\text{LDPC}(\infty,x^2,x^5)}\left[\text{P}_\text{b}^{\text{BP}}(\texttt{G},\epsilon,\ell\to\infty)\right]$, which is discussed in Problem 3.17 (thick curve).

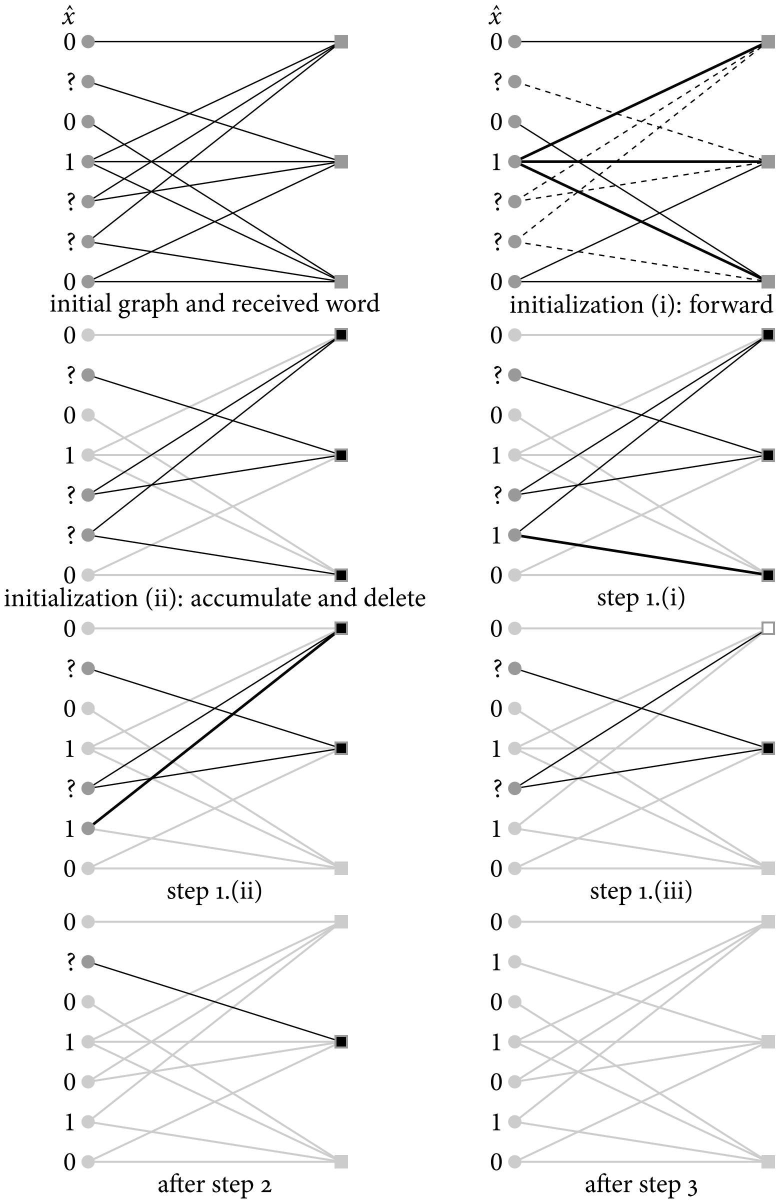


Figure 0.37: Peeling decoder applied to the $[7, 4, 3]$ Hamming code with the received word $y = (0, ?, ?, 1, 0, ?, 0)$. The vector $\hat{x}$ indicates the current estimate of the decoder of the transmitted codeword x. After three decoding steps the peeling decoder has successfully recovered the codeword.

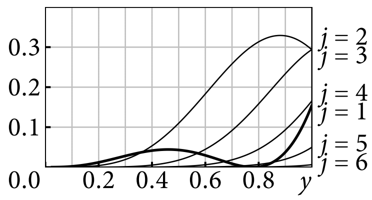


Figure 0.38: Evolution of the residual degrees $R_j(y)$, $j = 0, \ldots, 6$, as a function of the parameter y for the $(3,6)$-regular degree distribution. The channel parameter is $\epsilon = \epsilon^{\text{BP}} \approx 0.4294$. The curve corresponding to nodes of degree 1 is shown as a thick line.

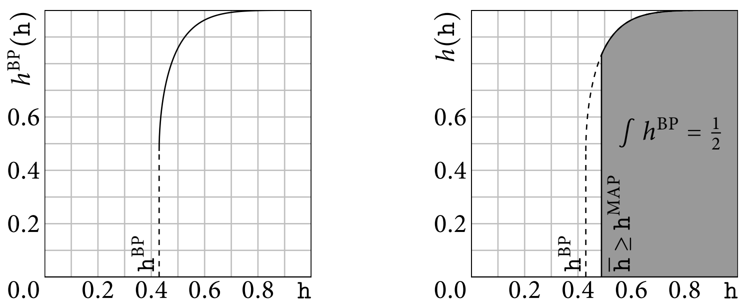


Figure 0.39: Left: BP EXIT function $h^{BP}(\epsilon)$; Right: Corresponding EXIT function $h(\epsilon)$ constructed according to Theorem 3.120.

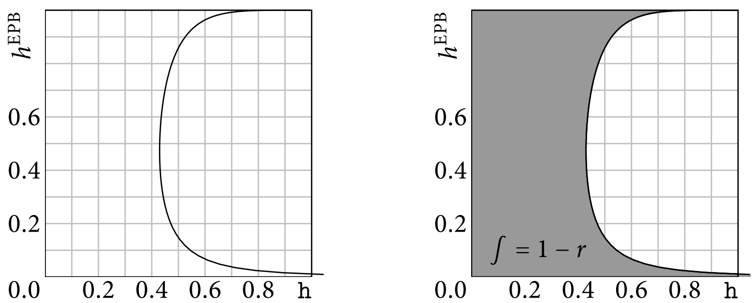


Figure 0.40: Left: EBP EXIT curve of the $(\mathtt{l} = 3, \mathtt{r} = 6)$-regular ensemble. Note that the curve goes "outside the box" and tends to infinity. Right: According to Lemma 3.128 the gray area is equal to $1 - r(\mathtt{l}, \mathtt{r}) = \frac{\mathtt{l}}{\mathtt{r}} = \frac{1}{2}$.

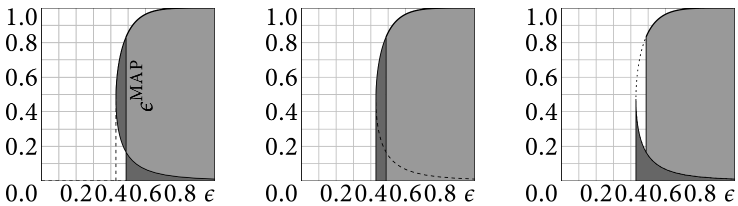
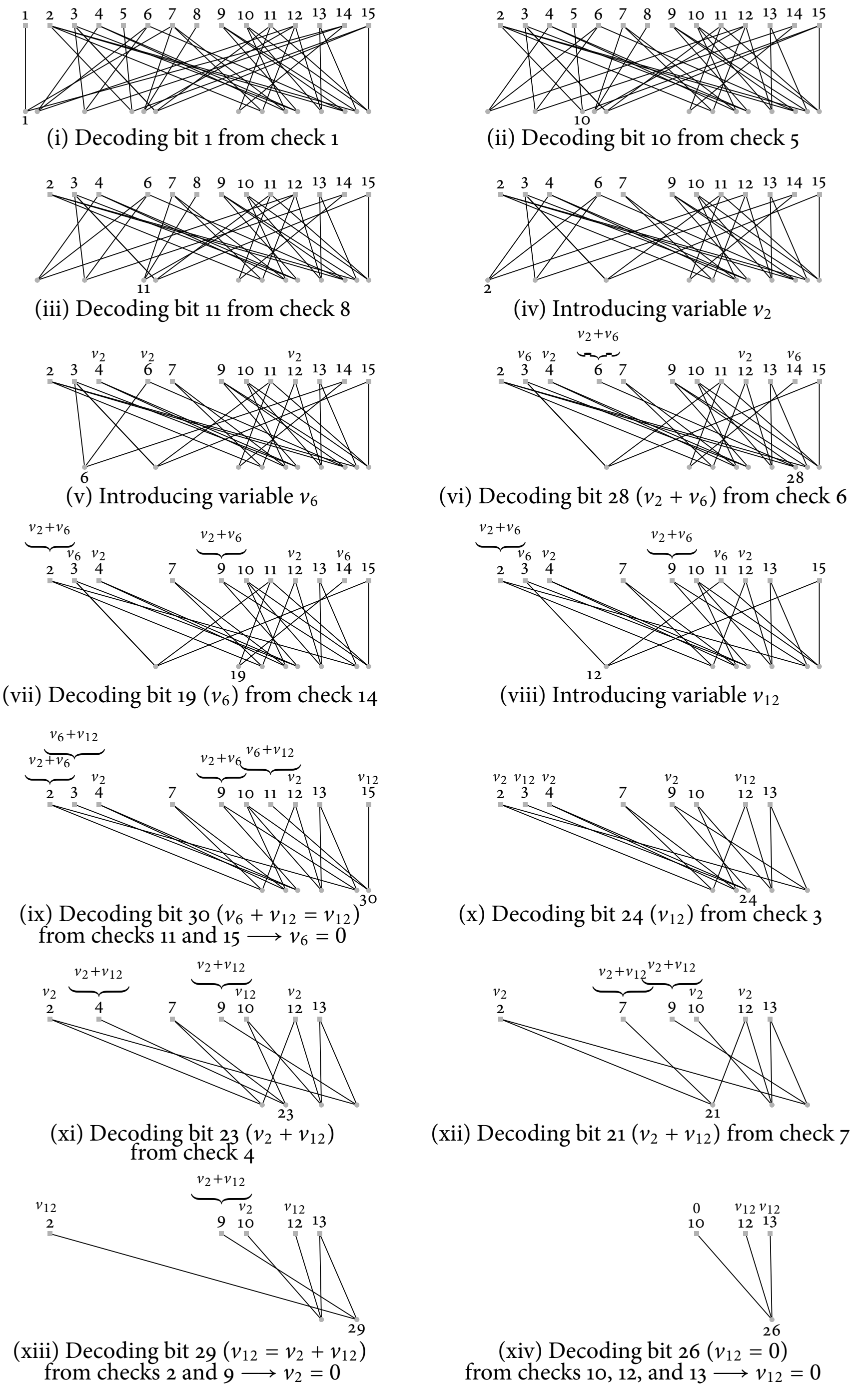


Figure 0.41: Left: Because of Theorem 3.120 and Lemma 3.128, at the MAP threshold ϵ^{MAP} the two dark gray areas are in balance. Middle: The dark gray area is proportional to the total number of variables which the M decoder introduces. Right: The dark gray area is proportional to the total number of equations which are produced during the decoding process and which are used to resolve variables.



(i) Decoding bit 1 from check 1

(ii) Decoding bit 10 from check 5

(iii) Decoding bit 11 from check 8

(iv) Introducing variable v_2

(v) Introducing variable v_6

(vi) Decoding bit 28 ($v_2 + v_6$) from check 6

(vii) Decoding bit 19 (v_6) from check 14

(viii) Introducing variable v_{12}

(ix) Decoding bit 30 ($v_6 + v_{12} = v_{12}$) from checks 11 and 15 $\longrightarrow v_6 = 0$

(x) Decoding bit 24 (v_{12}) from check 3

(xi) Decoding bit 23 ($v_2 + v_{12}$) from check 4

(xii) Decoding bit 21 ($v_2 + v_{12}$) from check 7

(xiii) Decoding bit 29 ($v_{12} = v_2 + v_{12}$) from checks 2 and 9 $\longrightarrow v_2 = 0$

(xiv) Decoding bit 26 ($v_{12} = 0$) from checks 10, 12, and 13 $\longrightarrow v_{12} = 0$

Figure 0.42: M decoder applied to a $(\mathtt{l} = 3, \mathtt{r} = 6)$-regular code of length $n = 30$.

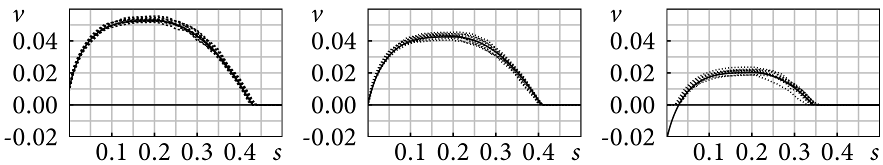


Figure 0.43: Comparison of the number of unresolved variables for the Maxwell decoder applied to the LDPC $\left(n, x^{\mathtt{l}-1}, x^{\mathtt{r}-1}\right)$ ensembles as predicted by Lemma 3.134 with samples for $n = 10,000$. The asymptotic curves are shown as solid lines, whereas the sample values are printed as dashed lines. The parameters are $\epsilon = 0.50$ (left), $\epsilon = \epsilon^{\mathrm{MAP}} \approx 0.48815$ (middle), and $\epsilon = 0.46$ (right). The parameter $\epsilon = 0.46$ is not covered by Lemma 3.134. Nevertheless, up to the point where the predicted curve dips below zero the experimental data agrees well.

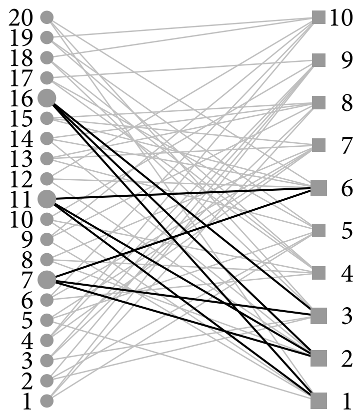


Figure 0.44: The subset of variable nodes $\mathcal{S} = \{7, 11, 16\}$ is a stopping set.

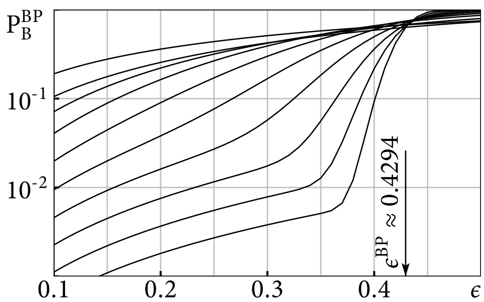


Figure 0.45: $\mathbb{E}_{\mathrm{LDPC}(nx^3, \frac{n}{2}x^6)}\left[\mathrm{P}_{\mathrm{B}}^{\mathrm{BP}}(\mathtt{G}, \epsilon)\right]$ as a function of ϵ for $n = 2^i$, $i \in [10]$.

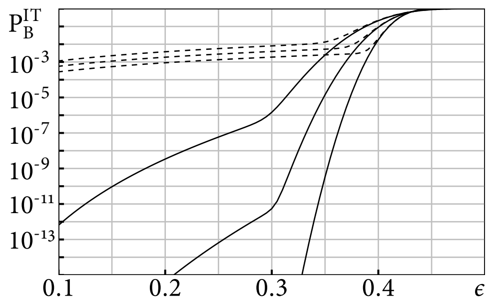


Figure 0.46: $P(\chi = 0, s_{\min}, \epsilon)$ for the ensemble LDPC$\left(nx^3, \frac{n}{2}x^6\right)$, where $n = 500, 1000, 2000$. The dashed curves correspond to the case $s_{\min} = 1$, whereas the solid curves correspond to the case where $s_{\min}$ was chosen to be 12, 22, and 40, respectively. In each of these cases the expected number of ss of size smaller than $s_{\min}$ is less than 1.

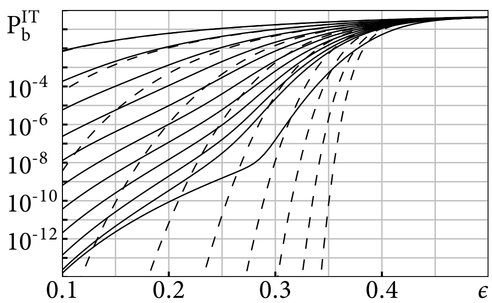


Figure 0.47: $P(\chi = 1, s_{\min} = 12, \ell, \epsilon)$ for the ensemble LDPC $\left(500x^3, 250x^6\right)$ and the first 10 iterations (solid curves). Also shown are the corresponding curves of the asymptotic density evolution for the first 10 iterations (dashed curves).

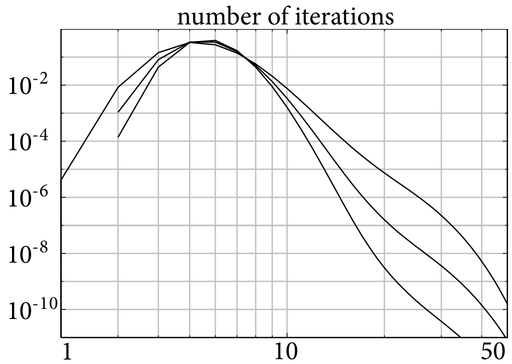


Figure 0.48: Probability distribution of the iteration number for the LDPC $\left(nx^3, \frac{n}{2}x^6\right)$ ensemble, lengths $n = 400$ (top curve), 600 (middle curve), and 800 (bottom curve) and $\epsilon = 0.3$. The typical number of iterations is around 5, but, e.g., for $n = 400$, 50 iterations are required with a probability of roughly 10^{-10}.

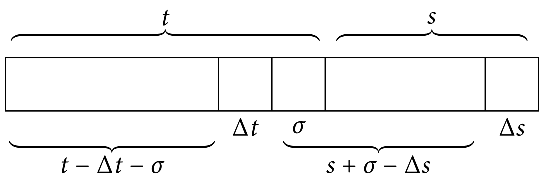


Figure 0.49: Derivation of the recursion for $A(v,t,s)$ for LDPC $\left(\Lambda(x) = nx^{\mathtt{l}}, P(x) = n\frac{\mathtt{l}}{\mathtt{r}}x^{\mathtt{r}}\right)$ and an unbounded number of iterations.

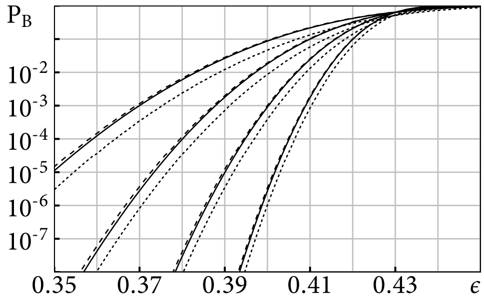


Figure 0.50: Scaling of $\mathbb{E}_{\mathrm{LDPC}(n,x^2,x^5)}[\mathrm{P_B}(\mathtt{G},\epsilon)]$ for transmission over the $\mathrm{BEC}(\epsilon)$ and BP decoding. The threshold for this combination is $\epsilon^{\mathrm{BP}} \approx 0.4294$. The blocklengths/expurgation parameters are $n/s = 1024/24$, 2048/43, 4096/82, and 8192/147, respectively. The solid curves represent the exact ensemble averages. The dotted curves are computed according to the basic scaling law stated in Theorem 3.151. The dashed curves are computed according to the refined scaling law stated in Conjecture 3.152. The scaling parameters are $\alpha = 0.56036$ and $\beta/\Omega = 0.6169$; see Table 3.154.

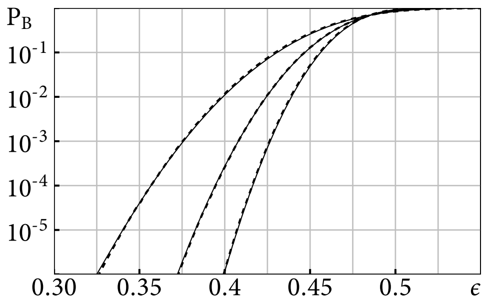


Figure 0.51: Scaling of $\mathbb{E}_{\text{LDPC}(n,\lambda=\frac{1}{6}x+\frac{5}{6}x^3,\rho=x^5)}[P_B(\mathtt{G},\epsilon)]$ for transmission over BEC(ϵ) and BP decoding. The threshold for this combination is $\epsilon^{\text{BP}} \approx 0.482803$. The blocklengths/expurgation parameters are $n/s = 350/14$, $700/23$, and $1225/35$. The solid curves represent the simulated ensemble averages. The dashed curves are computed according to the refined scaling law stated in Conjecture 3.152 with scaling parameters $\alpha = 0.5791$ and $\beta/\Omega = 0.6887$. The two curves are almost on top of each other and are hard to distinguish.

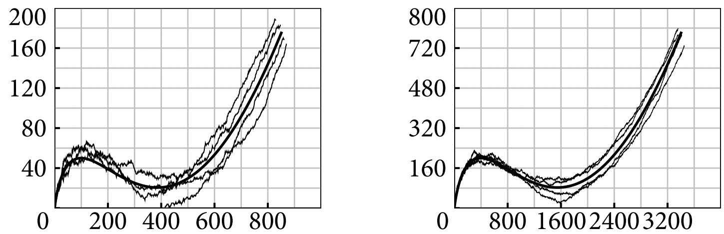


Figure 0.52: Evolution of $n(1-r)R_1$ as a function of the size of the residual graph for several instances for the ensemble LDPC $\left(n, \lambda(x) = x^2, \rho(x) = x^5\right)$ for $n = 2048$ (left) and $n = 8192$ (right). The transmission is over the BEC($\epsilon = 0.415$).

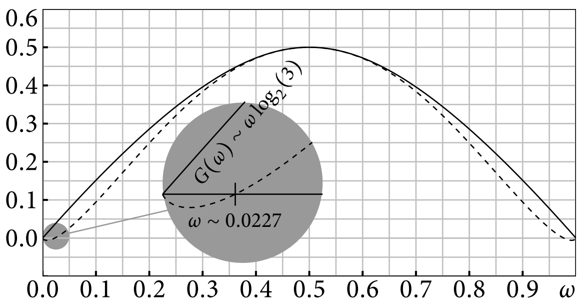


Figure 0.53: Growth rate $G(\omega)$ for the $(3,6)$ (dashed line) as well as the $(2,4)$ (solid line) ensemble.

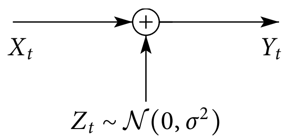


Figure 0.54: BAWGNC(σ).

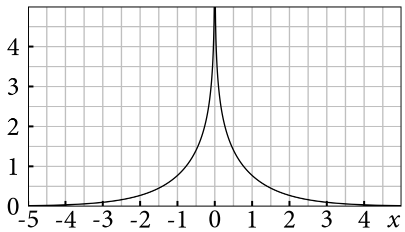


Figure 0.55: $\ln \coth \frac{|x|}{2}$.

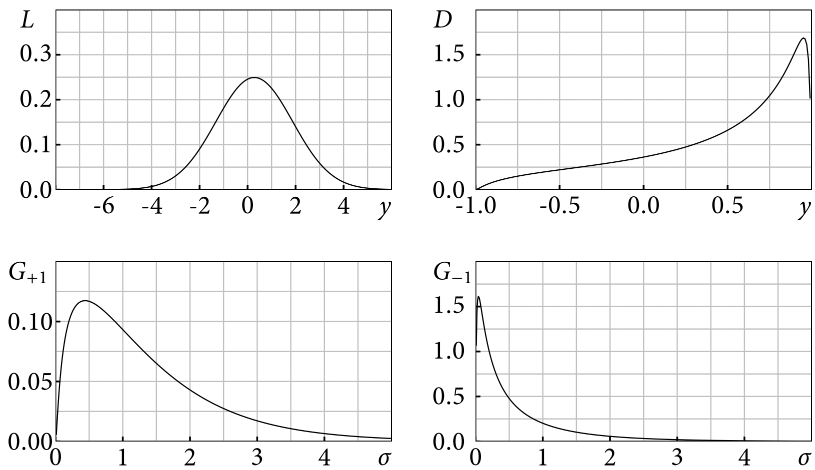


Figure 0.56: The L-density $\mathsf{a}_{\text{BAWGNC}(\sigma)}(y)$, the D-density $\mathfrak{a}_{\text{BAWGNC}(\sigma)}(y)$, as well as the corresponding G-density $a_{\text{BAWGNC}(\sigma)}(\pm 1, y)$ for $\sigma = 5/4$.

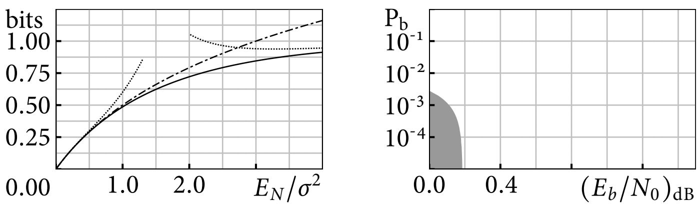


Figure 0.57: Left: Capacity of the BAWGNC (solid line) and the AWGNC (dashed line) in bits per channel use as a function of E_N/σ^2. Also shown are the asymptotic expansions (dotted) for large and small values of $\frac{E_N}{\sigma^2}$ discussed in Problem 4.12. Right: The achievable (white) region for the BAWGNC and $r = \frac{1}{2}$ as a function of $(E_b/N_0)_{\text{dB}}$.

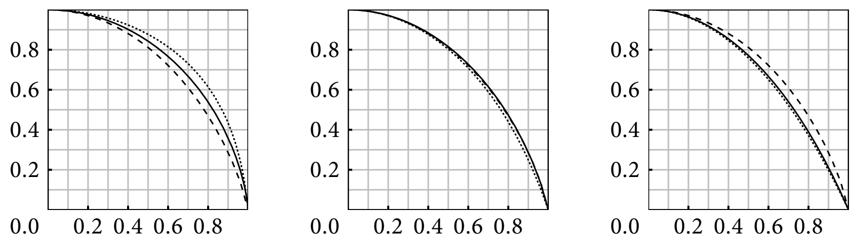


Figure 0.58: Comparison of the kernels $|d|^{\mathsf{a}_{\mathrm{BEC}(\mathtt{h})}}(\cdot)$ (dashed line) with $|d|^{\mathsf{a}_{\mathrm{BSC}(\mathtt{h})}}(\cdot)$ (dotted line) and $|d|^{\mathsf{a}_{\mathrm{BAWGNC}(\mathtt{h})}}(\cdot)$ (solid line) at channel entropy $\mathtt{h} = 0.1$ (left), $\mathtt{h} = 0.5$ (middle), and $\mathtt{h} = 0.9$ (right).

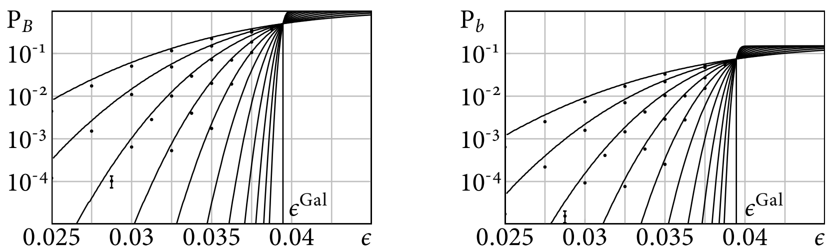


Figure 0.59: Performance of Gallager's algorithm A for the $(3, 6)$-regular ensemble when transmission takes place over the BSC. The blocklengths are $n = 2^i$, $i = 10, \ldots, 20$. The left-hand graph shows the block error probability, whereas the right-hand graph concerns the bit error probability. The dots correspond to simulations. For most simulation points the 95% confidence intervals (see Problem 4.37) are smaller than the dot size. The lines correspond to the analytic approximation of the waterfall curves based on scaling laws (see Section 4.13).

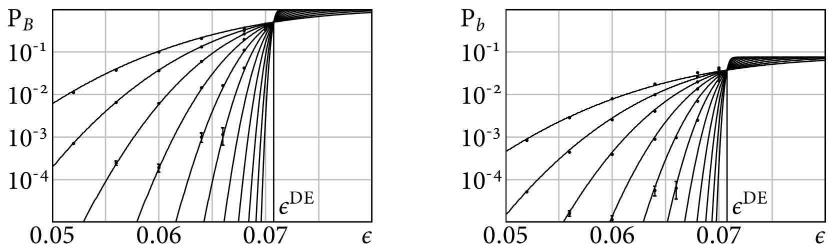


Figure 0.60: Performance of the decoder with erasures for the $(3, 6)$-regular ensemble when transmission takes place over the BSC. The blocklengths are $n = 2^i$, $i = 10, \ldots, 20$. The left-hand graph shows the block error probability, whereas the right-hand graph concerns the bit error probability. The dots correspond to simulations. The lines correspond to the analytic approximation of the waterfall curves based on scaling laws (see Section 4.13).

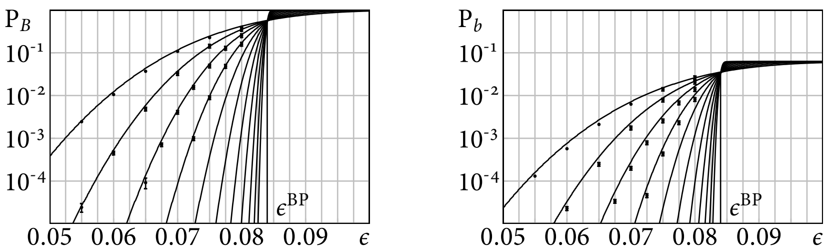


Figure 0.61: Performance of the BP decoder for the $(3,6)$-regular ensemble when transmission takes place over the BSC. The blocklengths are $n = 2^i$, $i = 10, \ldots, 20$. The left-hand graph shows the block error probability, whereas the right-hand graph concerns the bit error probability. The dots correspond to simulations. The lines correspond to the analytic approximation of the waterfall curves based on scaling laws.

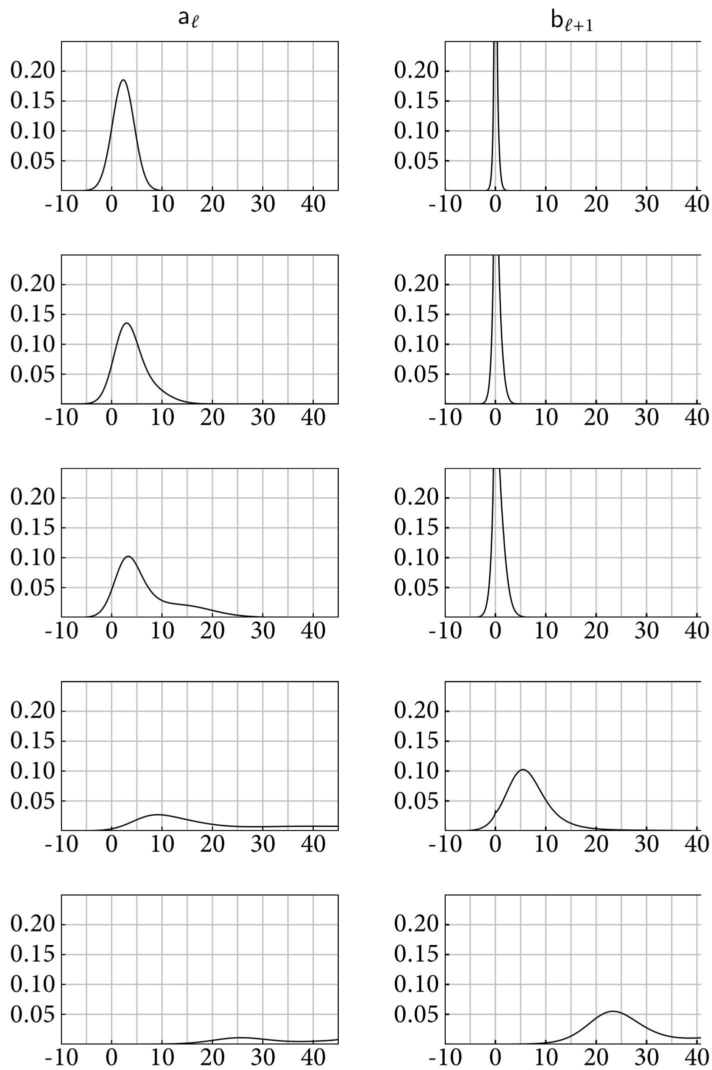


Figure 0.62: Evolution of a_ℓ (densities of messages emitted by variable nodes) and $b_{\ell+1}$ (densities of messages emitted from check nodes) for $\ell = 0, 5, 10, 50$, and 140 for the BAWGNC($\sigma = 0.93$) and the code given in Example 4.100. The densities "move to the right," indicating that the error probability decreases as a function of the number of iterations.

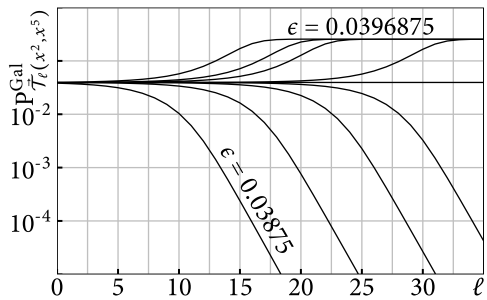


Figure 0.63: Evolution of $P^{\text{Gal}}_{\vec{\mathcal{T}}_\ell(x^2,x^5)}(\epsilon)$ as a function of ℓ for various values of ϵ. For $\epsilon = 0.03875, 0.039375, 0.0394531$, and 0.039462 the error probability converges to zero, whereas for $\epsilon = 0.039465, 0.0394922, 0.0395313$, and 0.0396875 the error probability converges to a non-zero value. For $\epsilon \approx 0.03946365$ the error probability stays constant. We conclude that $\epsilon^{\text{Gal}}(3,6) \approx 0.03946365$. Note that for $\epsilon > \epsilon^{\text{Gal}}(3,6)$, $P^{\text{Gal}}_{\vec{\mathcal{T}}_\ell(x^2,x^5)}(\epsilon)$ is an *increasing* function of ℓ, whereas below this threshold it is a *decreasing* function. In either case, $P^{\text{Gal}}_{\vec{\mathcal{T}}_\ell(x^2,x^5)}(\epsilon)$ is monotone as guaranteed by Lemma 4.104.

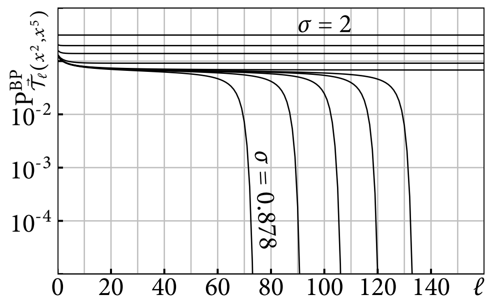


Figure 0.64: Evolution of $P^{BP}_{\vec{\mathcal{T}}_\ell(x^2,x^5)}(\sigma)$ as a function of the number of iterations ℓ for various values of σ. For $\sigma = 0.878, 0.879.0.8795, 0.8798$, and 0.88 the error probability converges to zero, whereas for $\sigma = 0.9, 1, 1.2$, and 2 the error probability converges to a non-zero value. We see that $\sigma^{BP}(3,6) \approx 0.881$. Note that, as predicted by Lemma 4.107, $P^{BP}_{\vec{\mathcal{T}}_\ell(x^2,x^5)}(\sigma)$ is a non-increasing function in ℓ.

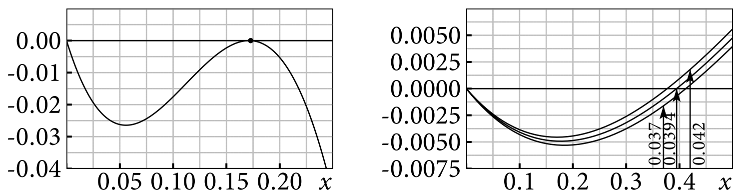


Figure 0.65: Left: $f(\epsilon, x) - x$ as a function of x for the $(3,3)$-regular ensemble and $\epsilon = \epsilon^{\text{Gal}} \approx 0.22305$. Right: $f(\epsilon, x) - x$ as a function of x for the $(3,6)$-regular ensemble and $\epsilon = 0.037$, $\epsilon = \epsilon^{\text{Gal}} \approx 0.394$, and $\epsilon = 0.042$.

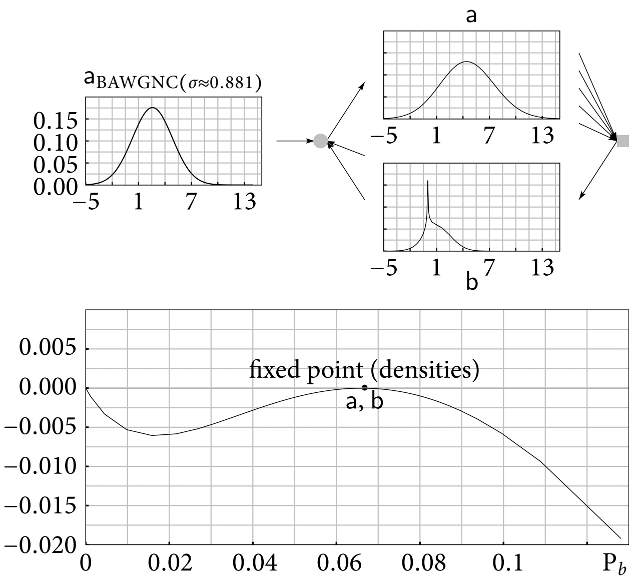


Figure 0.66: Progress per iteration (change of error probability) of density evolution for the $(3,6)$-ensemble and the BAWGNC(σ) channel with $\sigma \approx 0.881$ as a function of the bit error probability. In formulae: we plot $\mathfrak{E}(\mathsf{a}_\ell) - \mathfrak{E}(\mathsf{a}_{\ell-1})$ as a function of $\mathrm{P}_b = \mathfrak{E}(\mathsf{a}_{\text{BAWGNC}(\sigma)} \circledast L(\rho(\mathsf{a}_{\ell-1})))$, where $L(x) = x^3$ and $\rho(x) = x^5$. For cosmetic reasons this discrete set of points was interpolated to form a smooth curve. The initial error probability is equal to $Q(1/0.881) \approx 0.12817$. At the fixed point the progress is zero. The associated fixed point densities are a (emitted at the variable nodes) and b (emitted at the check nodes).

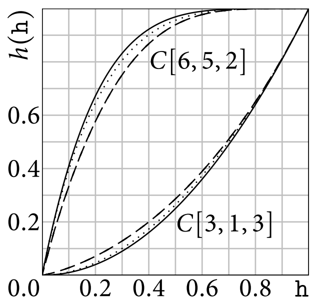


Figure 0.67: EXIT function of the $[3, 1, 3]$ repetition code and the $[6, 5, 2]$ parity-check code for the BEC (solid curve), the BSC (dashed curve), and also the BAWGNC (dotted curve).

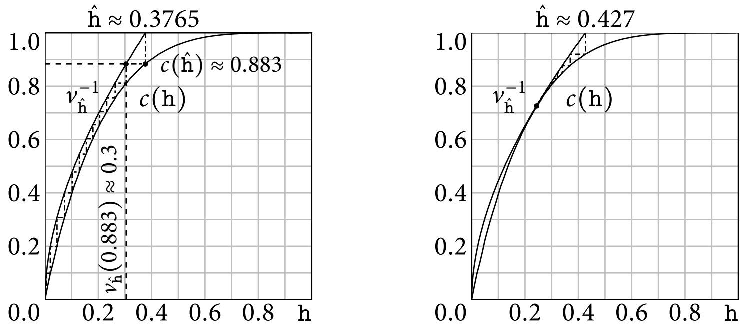


Figure 0.68: EXIT function of the $(3, 6)$-regular ensemble on the BAWGN channel. In the left-hand graph the parameter is $\hat{h} \approx 0.3765$ ($\sigma \approx 0.816$), whereas in the right-hand graph we chose $\hat{h} \approx 0.427$ ($\sigma = 0.878$).

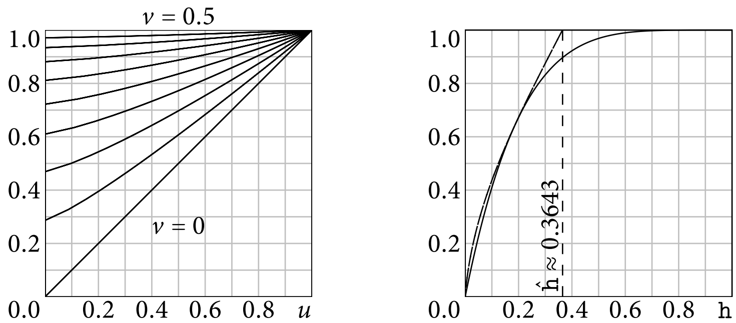


Figure 0.69: Left: For $v \in [0, \frac{1}{2}]$ the function $h_2\big(h_2^{-1}(u)(1-2v)+v\big)$ is non-decreasing and convex-∪ in u, $u \in [0,1]$. Right: Universal bound applied to the $(3,6)$-regular ensemble.

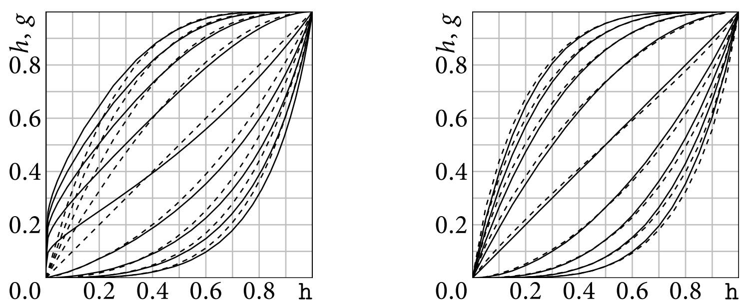


Figure 0.70: EXIT (solid) and GEXIT (dashed) function of the $[n, 1, n]$ repetition code and the $[n, n-1, 2]$ parity-check code assuming that transmission takes place over the BSC($\mathtt{h}$) (left) or the BAWGNC($\mathtt{h}$) (right), $n \in \{2, 3, 4, 5, 6\}$.

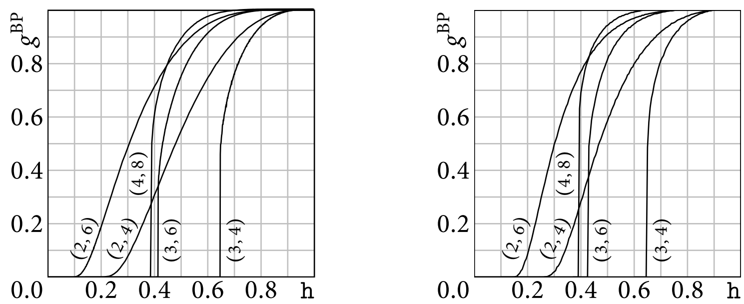


Figure 0.71: BP GEXIT curve for several regular LDPC ensembles for the BSC (left) and the BAWGNC (right).

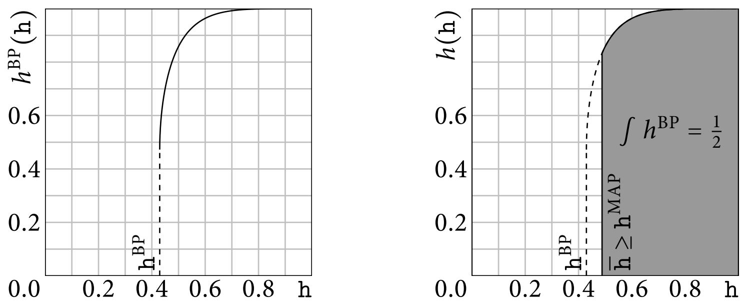


Figure 0.72: Left: BP GEXIT function $g^{\text{BP}}(\mathtt{h})$ for the $(3,6)$-regular ensemble; Right: Corresponding upper bound on GEXIT function $g(\mathtt{h})$ constructed according to Theorem 4.172.

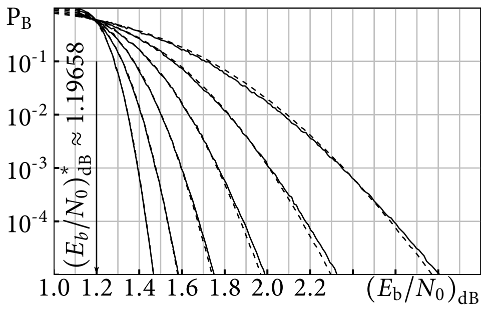


Figure 0.73: Scaling of $\mathbb{E}_{\text{LDPC}(n,x^2,x^5)}[\mathrm{P_B}(\mathtt{G},\mathtt{h})]$ for transmission over the BAWGNC($\mathtt{h}$) and a quantized version of belief propagation decoding implemented in hardware. The threshold for this combination is $(E_b/N_0)^*_{\text{dB}} \approx 1.19658$. The blocklengths n are $n = 1000$, 2000, 4000, 8000, 16,000, and 32,000, respectively. The solid curves represent the simulated ensemble averages. The dashed curves are computed according to the scaling law of Conjecture 4.176 with scaling parameters $\alpha = 0.8694$ and $\beta = 5.884$. These parameters were fitted to the empirical data.

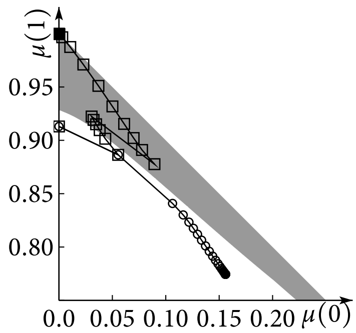


Figure 0.74: Region of convergence for the all-one weight sequence (indicated in gray).

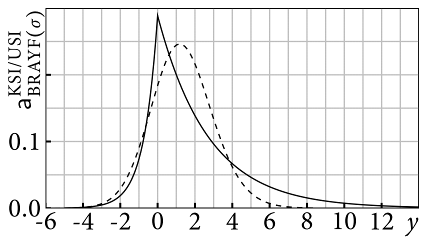


Figure 0.75: L-densities $a^{KSI}_{BRAYFC(\sigma)}$ (solid curve) and $a^{USI}_{BRAYFC(\sigma)}$ (dashed curve) for $\sigma = 1$.

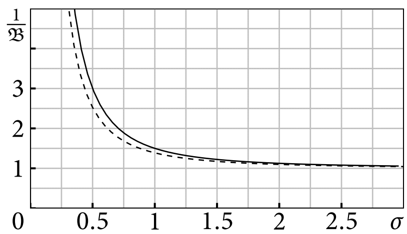


Figure 0.76: Upper bound on $\lambda'(0)\rho'(1)$, i.e., $1/\mathfrak{B}$, for the KSI case (solid curve), computed according to (5.1), and for the USI case (dashed curve).

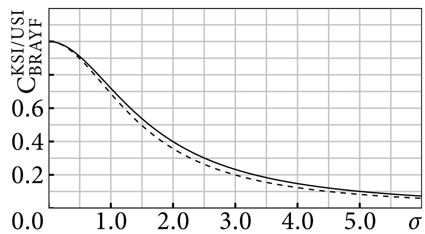


Figure 0.77: Capacities $C^{KSI}_{BRAYFC}(\sigma)$ (solid curve) and $C^{USI}_{BRAYFC}(\sigma)$ (dashed curve) as a function of σ measured in bits.

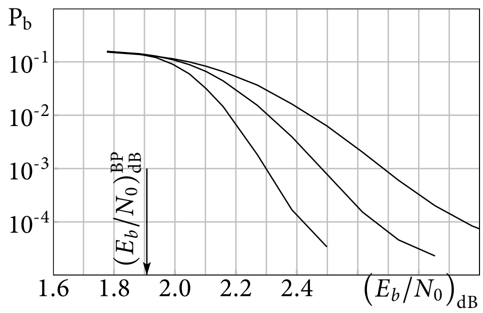


Figure 0.78: $\mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}[\text{P}_b(\mathtt{G}, E_b/N_0)]$ for the optimized ensemble stated in Example 5.6 and transmission over the BRAYF(E_b/N_0) with KSI and belief-propagation decoding. As stated in Example 5.6, the threshold for this combination is $\sigma^{\text{BP}}_{\text{KSI}} \approx 0.8028$ which corresponds to $\left(E_b/N_0\right)_{\text{dB}} \approx 1.90785$. The blocklengths/expurgation parameters n/s are $n = 8192/10$, $16384/10$, and $32768/10$, respectively.

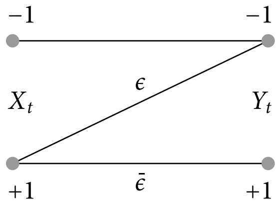


Figure 0.79: Z channel with parameter ϵ.

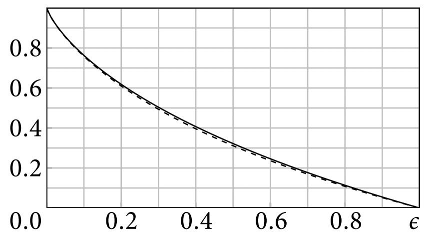


Figure 0.80: Comparison of $C_{ZC(\epsilon)}$ (solid curve) with $I_{\alpha=\frac{1}{2}}(X;Y)$ (dashed curve), both measured in bits.

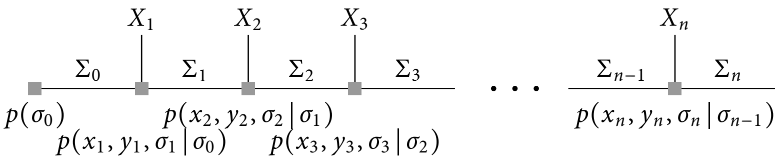


Figure 0.81: FSFG corresponding to (5.13).

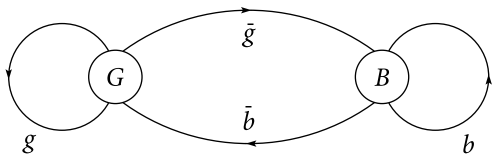


Figure 0.82: Gilbert-Elliott channel with two states.

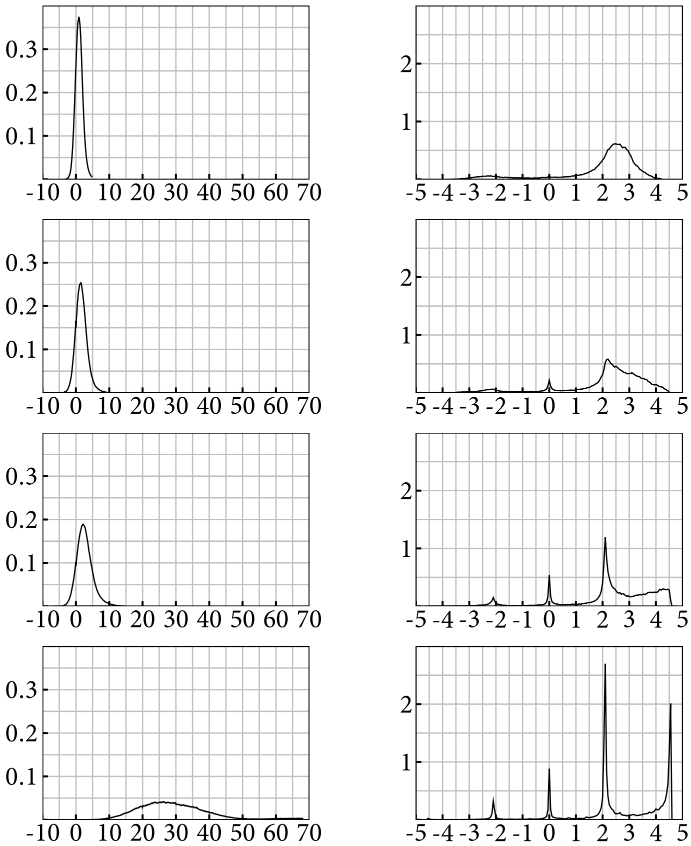


Figure 0.83: L-densities of density evolution at iteration 1, 2, 4, and 10. The left pictures show the densities of the messages which are passed from the code toward the part of the FSFG which estimates the channel state. The right-hand side shows the density of the messages which are the estimates of the channel state and which are passed to the part of the FSFG corresponding to the code.

Figure 0.84: Two specific maps ψ for the 4-PAM constellation.

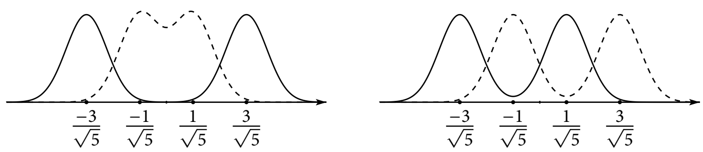


Figure 0.85: Transition probabilities $p_{Y\,|\,X^{[1]}}\big(y\,|\,x^{[1]}\big)$ for $\sigma \approx 0.342607$ as a function of $x^{[1]} = 0/1$ (solid/dashed). The two cases correspond to the two maps ψ shown in Figure 0.84.

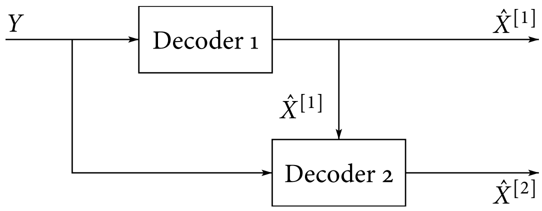


Figure 0.86: Multilevel decoding scheme. The two decoding parts correspond to the two parts of (5.25).

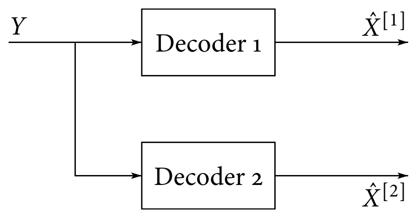


Figure 0.87: BICM decoding scheme. The two decoding parts correspond to $I\big(X^{[1]}; Y\big)$ and $I\big(X^{[2]}; Y\big)$, respectively.

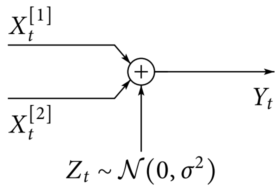


Figure 0.88: BAWGNMA channel with two users.

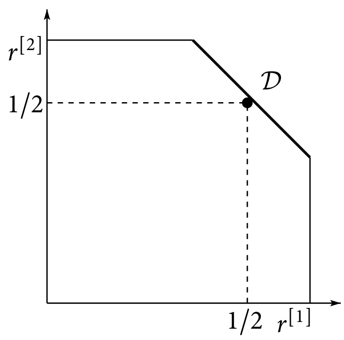


Figure 0.89: Capacity region for $\sigma \approx 0.778$. The dominant face $\mathcal{D}$ (thick diagonal line) is the set of rate tuples of the capacity region of maximal sum rate.

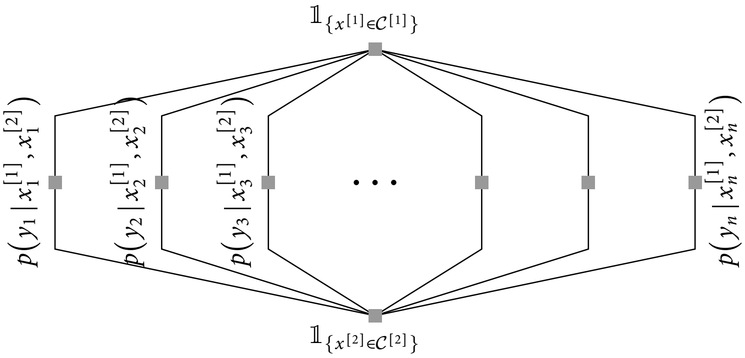


Figure 0.90: FSFG corresponding to decoding on the BAWGNMA channel.

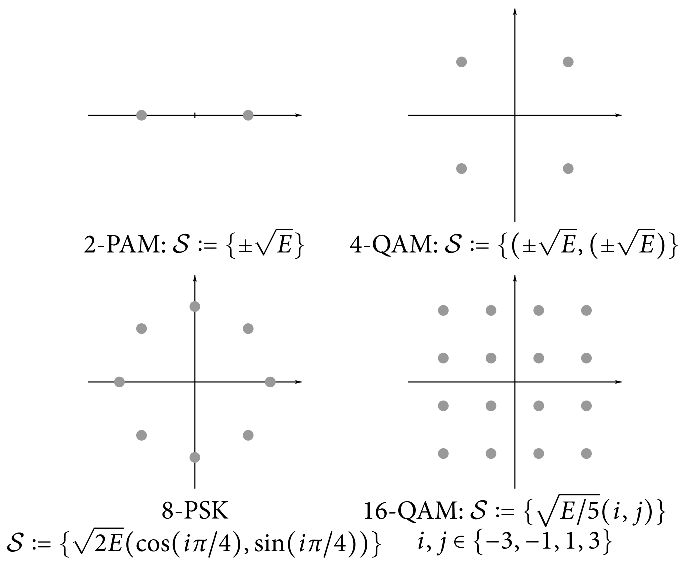


Figure 0.91: Four standard signal constellations: 2-PAM (top left), 4-QAM (top right), 8-PSK (bottom left), and 16-QAM (bottom right). In all cases it is assumed that the prior on $\mathcal{S}$ is uniform. The signal constellations are scaled so that the average energy per dimension is E.

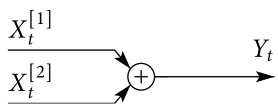


Figure 0.92: Multiple-access binary adder channel.

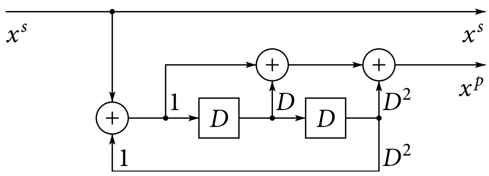


Figure 0.93: Binary systematic recursive convolutional encoder of memory $m = 2$ and rate one-half defined by $G = 7/5$. The two square boxes are delay elements. The 7 corresponds to $1 + D + D^2$. These are the coefficients of the "forward" branch (the top branch of the filter) with 1 corresponding to the leftmost coefficient. In a similar manner, 5 corresponds to $1 + D^2$, which represents the coefficients of the "feedback" branch. Again, the leftmost coefficient corresponds to 1.

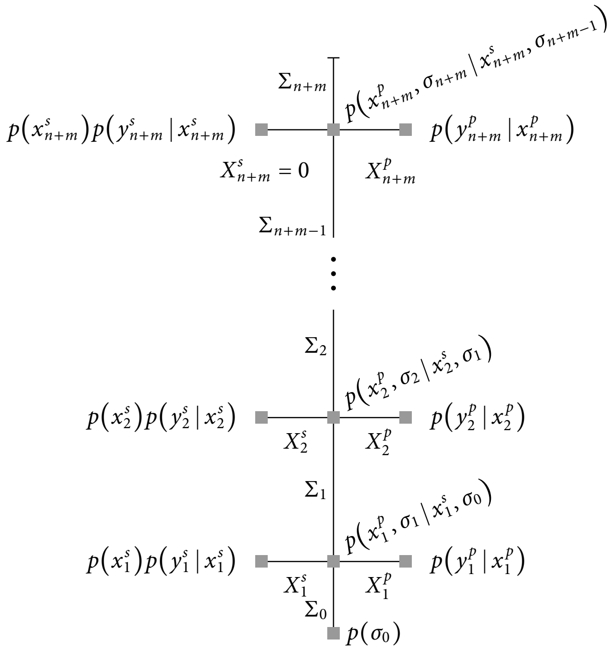


Figure 0.94: FSFG for the MAP decoding of $C(G, n)$.

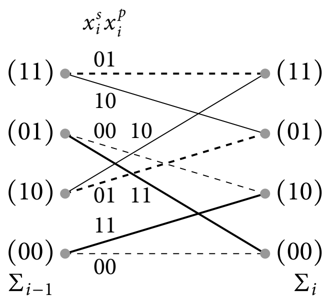


Figure 0.95: Trellis section for the case $G = 7/5$. There are four states. A dashed/solid line indicates that $x_i^s = 0/1$ and thin/thick lines indicate that $x_i^p = 0/1$.

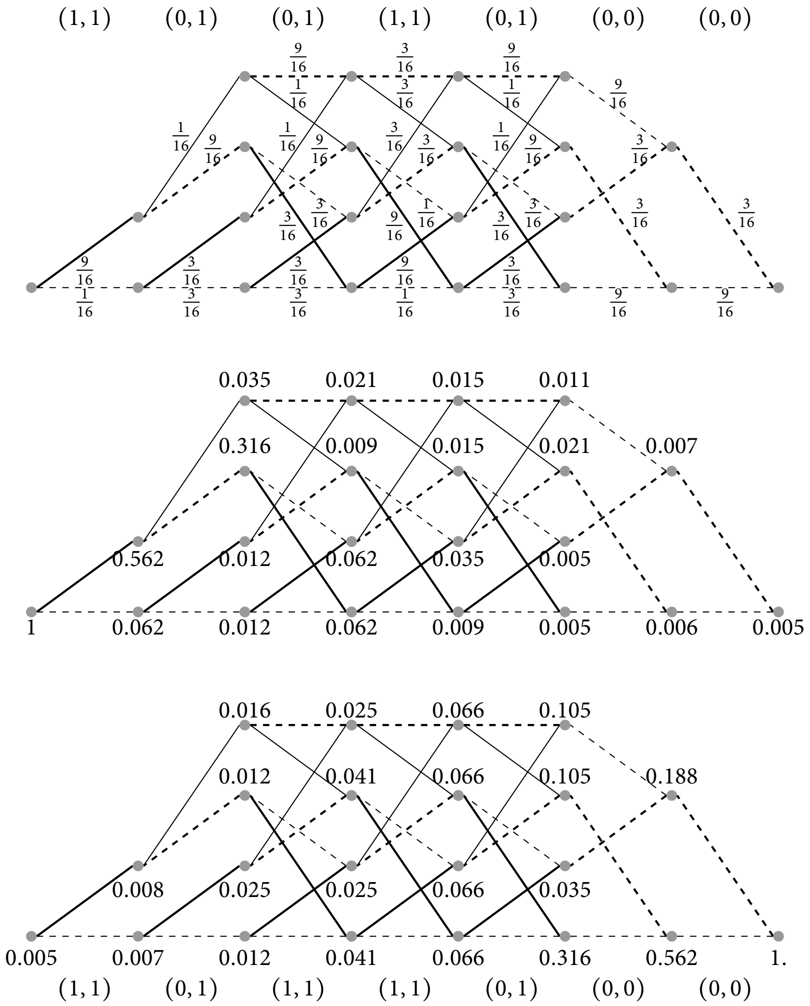


Figure 0.96: BCJR algorithm applied to the code $C(G = 7/5, n = 5)$ assuming transmission takes place over the BSC($\epsilon = 1/4$). The received word is equal to $(y^s, y^p) = (1001000, 1111100)$. The top figure shows the trellis with branch labels corresponding to the received sequence. We have *not* included the prior $p(x_i^s)$, since it is uniform. The middle and bottom figures show the α- and the β- recursion, respectively. On the very bottom, the estimated sequence is shown.

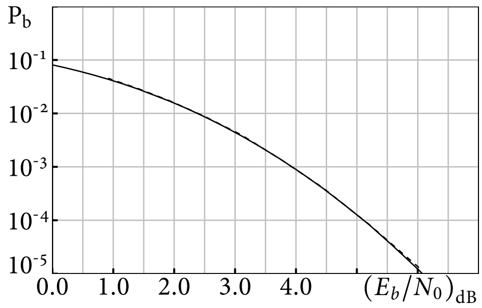


Figure 0.97: Performance of the rate one-half code $C(G = 21/37, n = 2^{16})$ over the BAWGNC under optimal bit-wise decoding (BCJR, solid line). Note that $(E_b/N_0)_{\mathrm{dB}} = 10\log_{10}\frac{1}{2r\sigma^2}$. Also shown is the performance under optimal block-wise decoding (Viterbi, dashed line). The two curves overlap almost entirely. Although the performance under the Viterbi algorithm is strictly worse the difference is negligible.

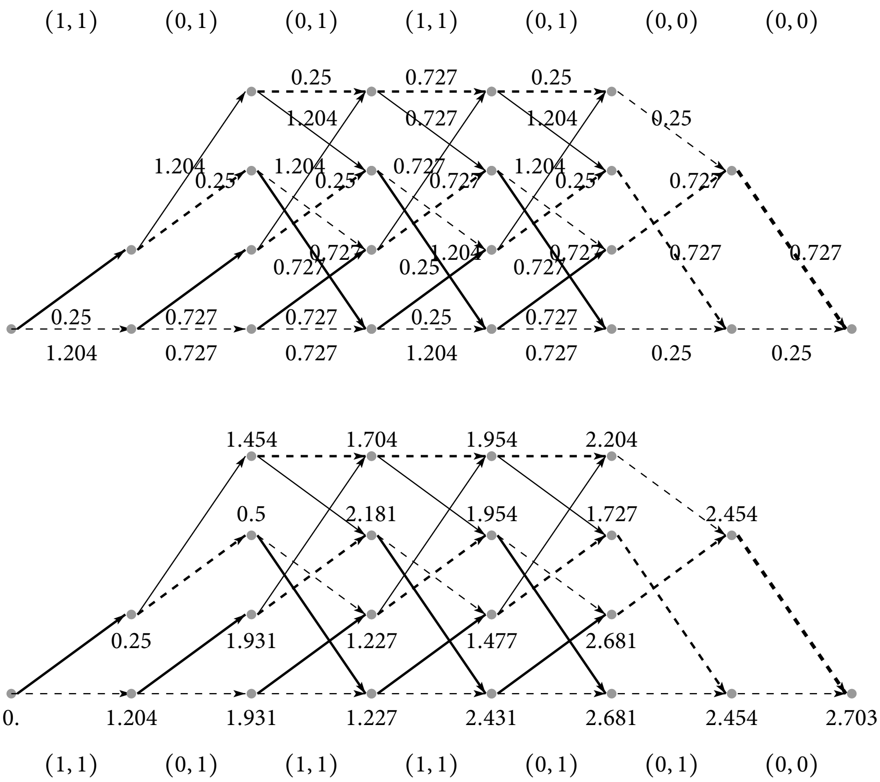


Figure 0.98: Viterbi algorithm applied to the code $C(G = 7/5, n = 5)$ assuming transmission takes place over the BSC($\epsilon = 1/4$). The received word is $(y^s, y^p) = (1001000, 1111100)$. The top figure shows the trellis with branch labels corresponding to $-\log_{10}\left(p(y_i^s \mid x_i^s)p(y_i^p \mid x_i^p)\right)$. Since we have a uniform prior we can take out the constant $p(x_i^s)$. These branch labels are easily derived from Figure 0.96 by applying the function $-\log_{10}$. The bottom figure show the workings of the Viterbi algorithm. On the very bottom the estimated sequence is shown.

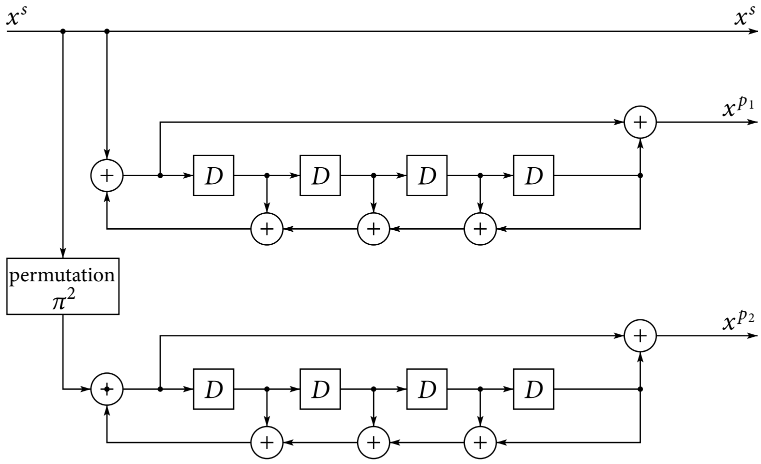


Figure 0.99: Encoder for $C(G = 21/37, n, \pi = (\pi^1, \pi^2))$, where π^1 is the identity permutation.

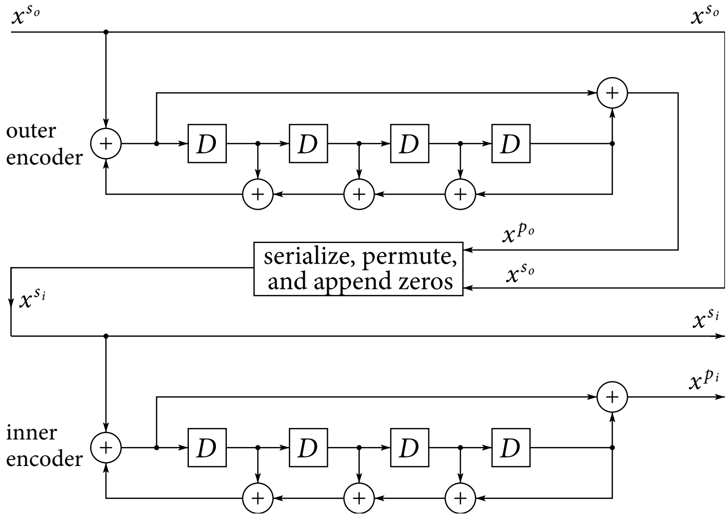


Figure 0.100: Encoder for $C(G^o = 21/37, G^i = 21/37, n, \pi)$.

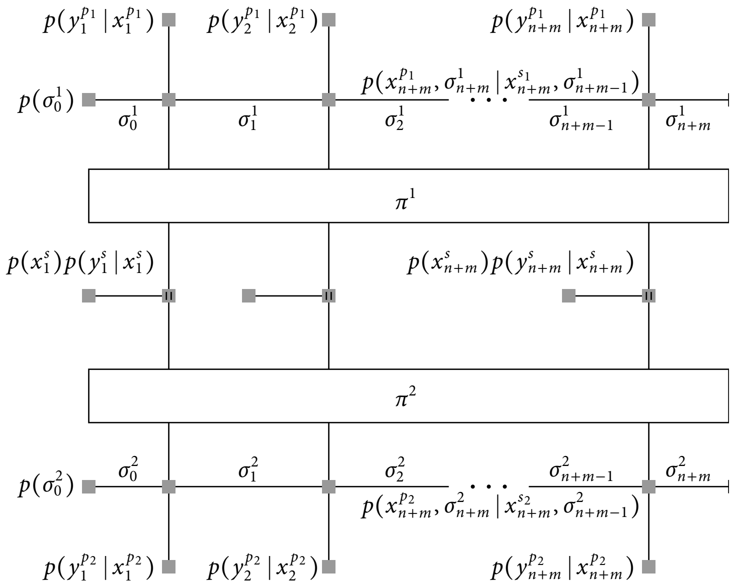


Figure 0.101: FSFG for the optimum bit-wise decoding of an element of $\mathcal{P}(G, n)$.

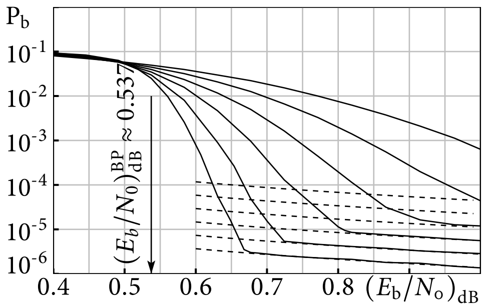


Figure 0.102: $\mathbb{E}_{\mathcal{P}(G=21/37,n,r=1/2)}[\mathrm{P_b}(C, E_b/N_0)]$ for an alternating puncturing pattern (identical on both branches), $n = 2^{11}, \dots, 2^{16}$, 50 iterations, and transmission over the BAWGNC(E_b/N_0). The arrow indicates the position of the threshold $(E_b/N_0)^{BP}_{dB} \approx 0.537$ ($\sigma^{BP} \approx 0.94$) which we compute in Section 6.5. The dashed curves are analytic approximations of the error floor discussed in Lemma 6.52.

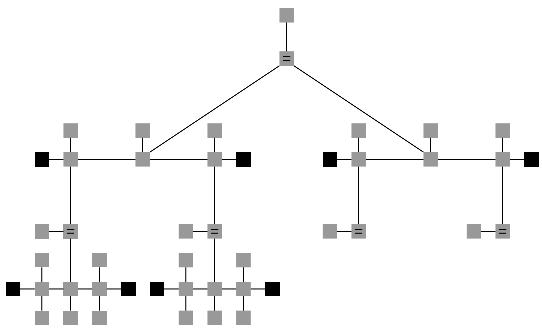


Figure 0.103: Computation graph corresponding to windowed ($w = 1$) iterative decoding of a parallel concatenated code for two iterations. The black factor nodes indicate the end of the decoding windows and represent the prior which we impose on the boundary states.

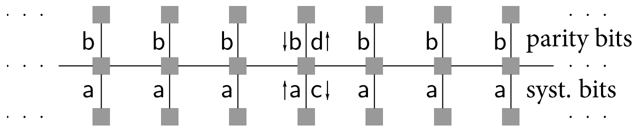


Figure 0.104: Definition of the maps $\mathsf{c} = \Gamma_G^s(\mathsf{a}, \mathsf{b})$ and $\mathsf{d} = \Gamma_G^p(\mathsf{a}, \mathsf{b})$. We are given a bi-infinite trellis defined by a rational function $G(D)$. Associated with all systematic variables are iid samples from a density a, whereas the parity bits experience the channel b. The resulting densities of the outgoing messages are denoted by c and d, respectively.

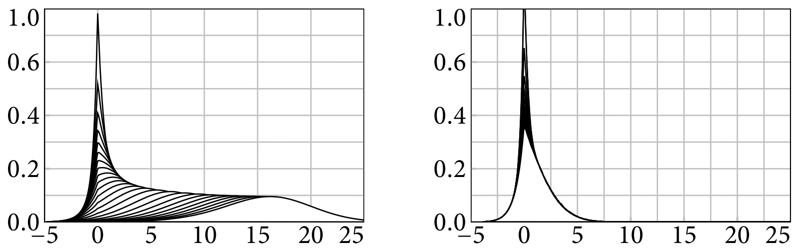


Figure 0.105: Evolution of c_ℓ for $\ell = 1, \ldots, 25$ for the ensemble $\mathcal{P}(G = 21/37, r = 1/2)$, an alternating puncturing pattern of the parity bits, and transmission over the BAWGNC(σ). In the left picture $\sigma = 0.93$ ($E_b/N_0 \approx 0.63$ dB). For this parameter the densities keep moving "to the right" toward Δ_∞. In the right picture $\sigma = 0.95$ ($E_b/N_0 \approx 0.446$ dB). For this parameter the densities converge to a fixed point density.

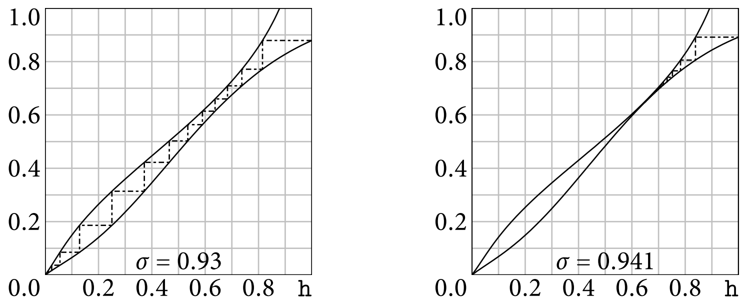


Figure 0.106: EXIT chart method for the ensemble $\mathcal{P}(G = 21/37, r = 1/2)$ with alternating puncturing on the BAWGN channel. In the left-hand picture the parameter is $\sigma = 0.93$, whereas in the right-hand picture we chose $\sigma = 0.941$.

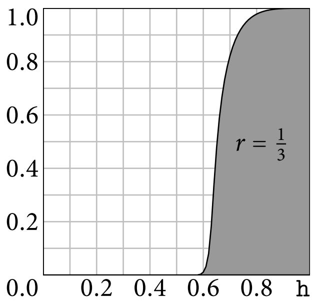


Figure 0.107: BP GEXIT curve for the ensemble $\mathcal{P}(G = 7/5, r = 1/3)$ assuming that transmission takes place over the BAWGNC($\mathtt{h}$). The BP and the MAP thresholds coincide and both thresholds are given by the stability condition. We have $\mathtt{h}^{\text{MAP/BP}} \approx 0.559$ ($\sigma^{\text{MAP/BP}} \approx 1.073$).

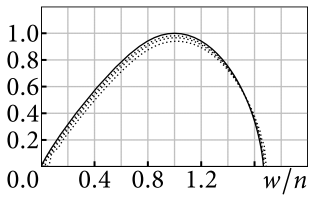


Figure 0.108: Exponent $\frac{1}{n}\log_2(a_{w,n})$ of the regular weight distribution of the code $C(G = 7/5, n)$ as a function of the normalized weight w/n for $n = 64, 128$, and 256 (dashed curves). Also shown is the asymptotic limit (solid line).

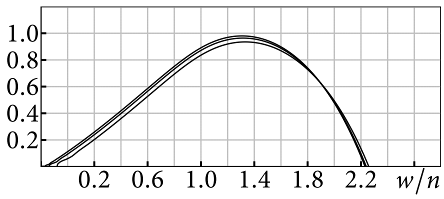


Figure 0.109: Exponent $\frac{1}{n}\log_2(p_{w,n})$ as a function of the normalized weight w/n for the ensemble $\mathcal{P}(G = 7/5, n, r = 1/3)$ and $n = 64, 128$, and 256. The normalization of the weight is with respect to n, not the blocklength.

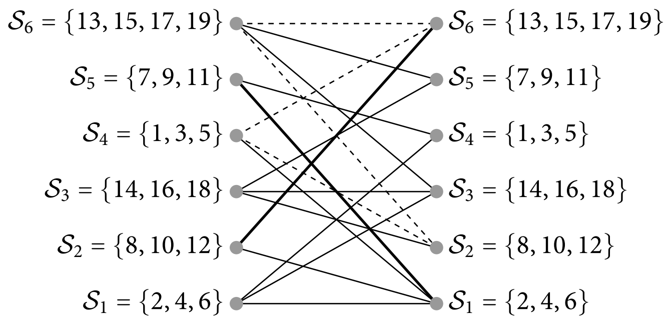


Figure 0.110: Bipartite graph corresponding to the parameters $n = 19$, $d = 2$, $\Delta = 3$, and $\pi = \{12, 1, 4, 18, 17, 4, 3, 19, 5, 13, 2, 6, 16, 7, 15, 14, 9, 8, 10\}$. Double edges are indicated by thick lines. The cycle of length 4, formed by (starting on the left) $\mathcal{S}_6 \to \mathcal{S}_6 \to \mathcal{S}_4 \to \mathcal{S}_2$, is shown as dashed lines.

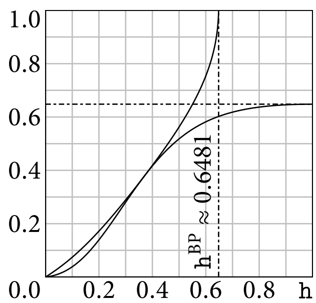


Figure 0.111: EXIT chart for transmission over the BEC($h \approx 0.6481$) for an asymmetric (big-numerator) parallel concatenated ensemble.

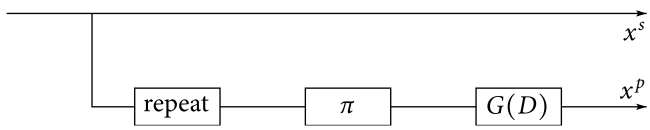


Figure 0.112: Alternative view of an encoder for a standard parallel concatenated code.

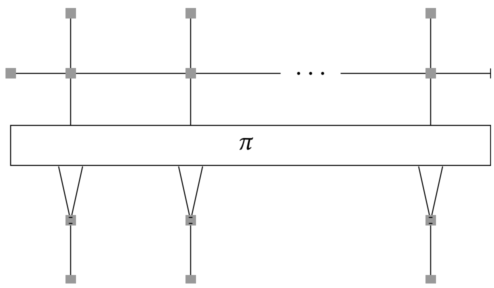


Figure 0.113: Alternative view of the FSFG of a standard parallel concatenated code.

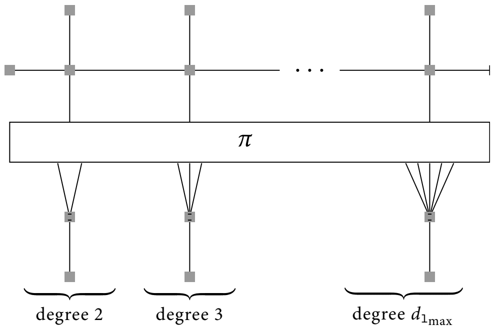


Figure 0.114: FSFG of an irregular parallel concatenated turbo code.

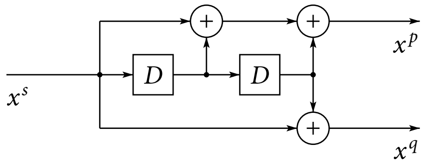


Figure 0.115: Binary feed-forward convolutional encoder of memory $m = 2$ and rate one-half defined by $(p(D) = 1 + D + D^2, q(D) = 1 + D^2)$.

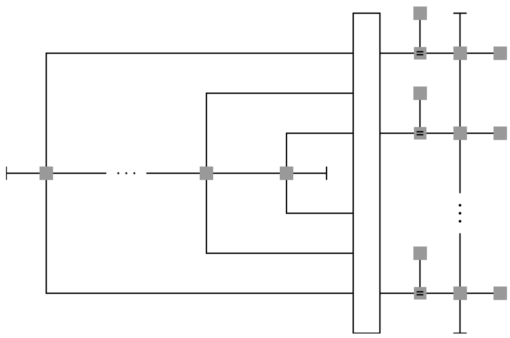


Figure 0.116: FSFG for the optimal bit-wise decoding of $\mathcal{S}(G^o, G^i, n, r)$.

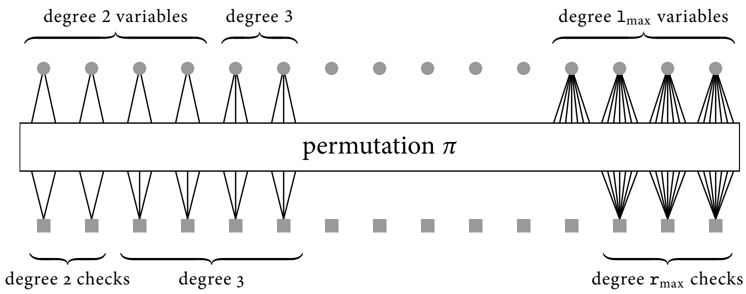


Figure 0.117: Tanner graph of a standard irregular LDPC code.

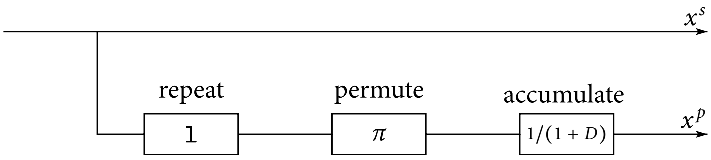


Figure 0.118: Encoder for an RA code. Each systematic bit is repeated `l` times; the resulting vector is permuted and fed into a filter with response $1/(1+D)$ (accumulate).

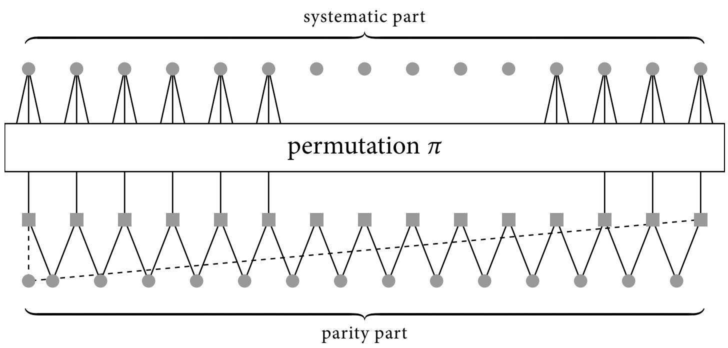


Figure 0.119: Tanner graph of an RA code with `l` = 3.

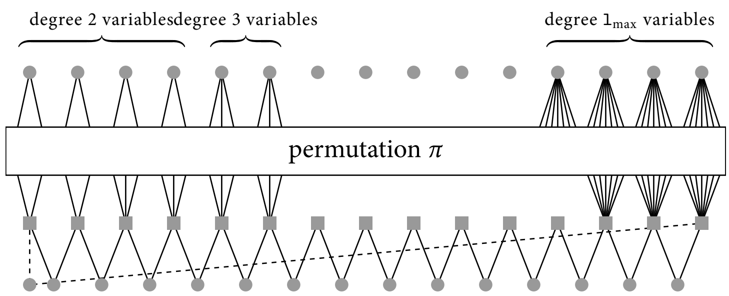


Figure 0.120: Tanner graph corresponding to an IRA code.

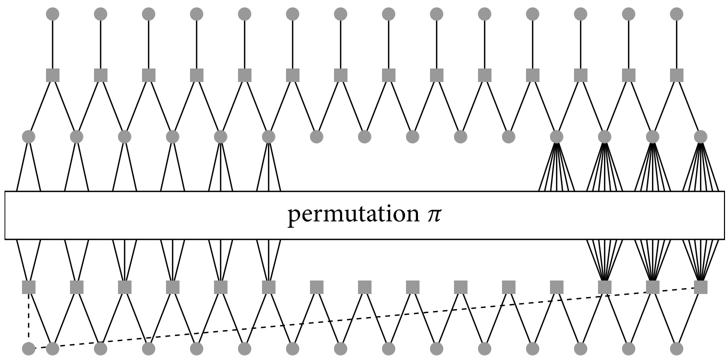


Figure 0.121: Tanner graph of an ARA code.

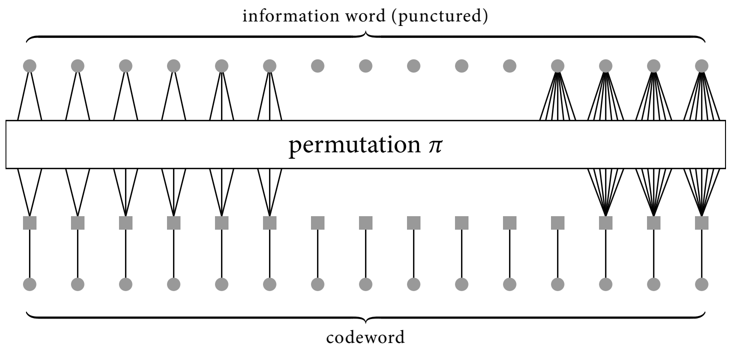


Figure 0.122: Tanner graph of an irregular LDGM code.

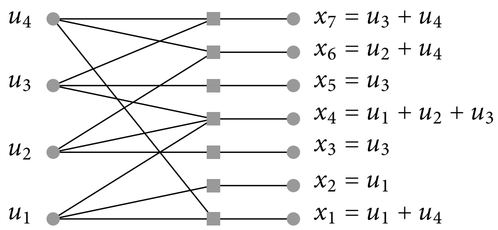


Figure 0.123: Tanner graph of a simple LDGM code.

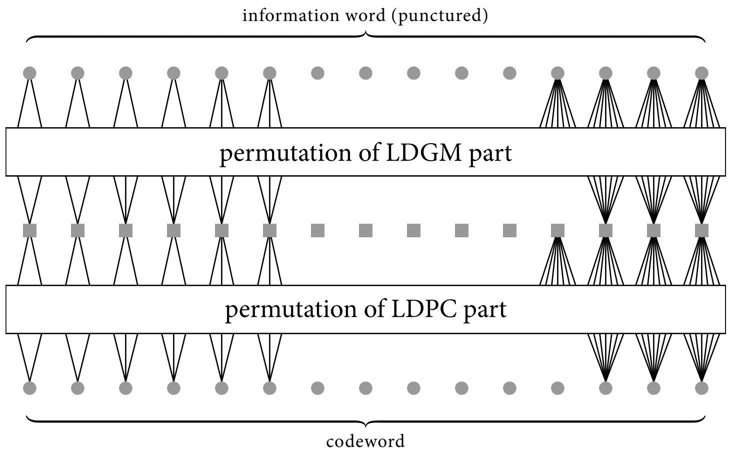


Figure 0.124: Tanner graph of an MN code.

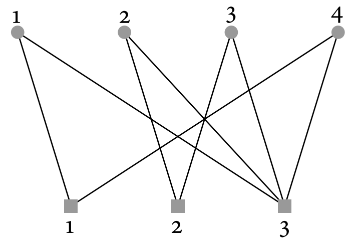


Figure 0.125: Base graph.

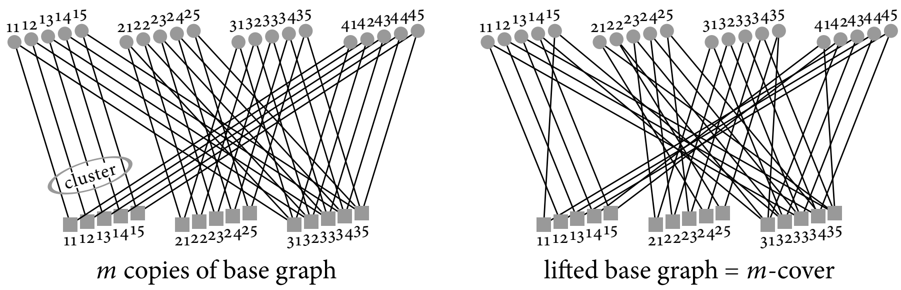


Figure 0.126: Left: m copies of base graph with $m = 5$. Right: Lifted graph resulting from applying permutations to the edge clusters.

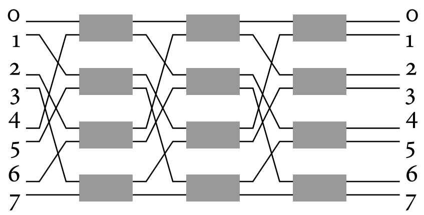


Figure 0.127: Ω-network for 8 elements. It has $\log_2(8) = 3$ stages, each consisting of a perfect shuffle.

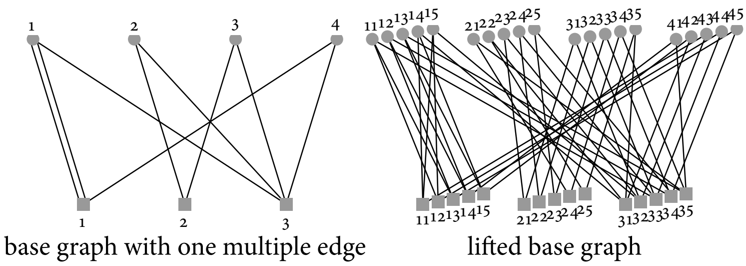
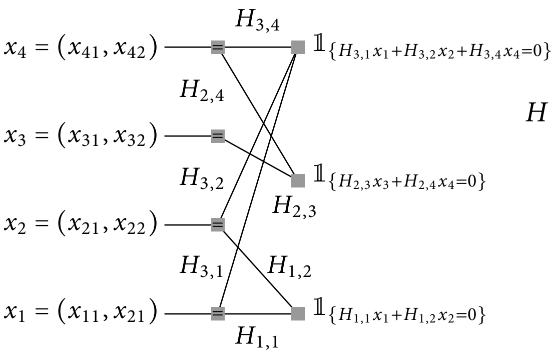


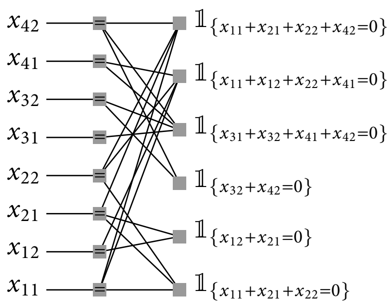
Figure 0.128: Left: Base graph with a multiple edge between variable node 1 to check node 1. Right: Lifted graph.



$$H = \begin{pmatrix} H_{1,1} & H_{1,2} & 0 & 0 \\ 0 & 0 & H_{2,3} & H_{2,4} \\ H_{3,1} & H_{3,2} & 0 & H_{3,4} \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 1+z & 0 & 0 \\ 0 & 0 & z & z \\ 1+z & z & 0 & 1 \end{pmatrix}$$

Figure 0.129: FSFG of a simple code over $\mathbb{F}_4$ and its associated parity-check matrix H. The primitive polynomial generating $\mathbb{F}_4$ is $p(z) = 1 + z + z^2$.



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

Figure 0.130: FSFG of a simple code over $\mathbb{F}_4$ and its associated parity-check matrix H. The primitive polynomial generating $\mathbb{F}_4$ is $p(z) = 1 + z + z^2$.

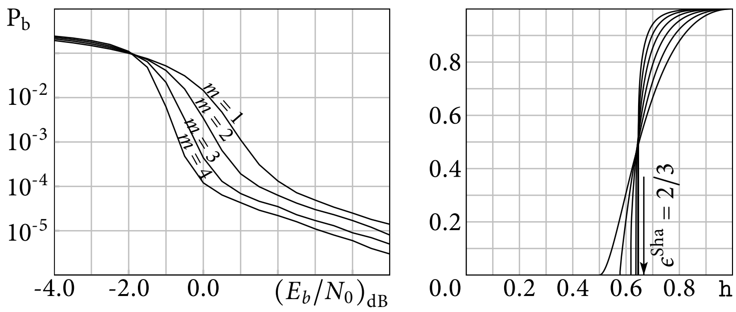


Figure 0.131: Left: Performance of the $(2,3)$-regular ensemble over $\mathbb{F}_{2^m}$, $m = 1,2,3,4$ of binary length 4320 over the BAWGNC(σ). Right: EXIT curves for the $(2,3)$-regular ensembles over $\mathbb{F}_{2^m}$ for $m = 1,2,3,4,5,6$, and transmission over the BEC(ϵ).

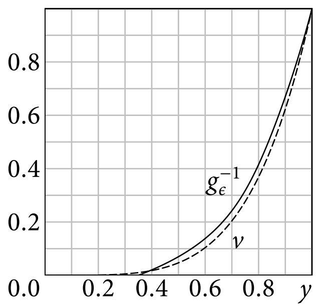


Figure 0.132: EXIT chart for the LDGM ensemble with $\lambda(x) = \frac{1}{2}x^4 + \frac{1}{2}x^5$ and $\rho(x) = \frac{2}{5}x + \frac{1}{5}x^2 + \frac{2}{5}x^8$ and transmission over the BEC($\epsilon = 0.35$).

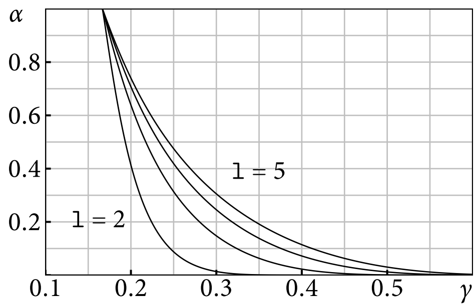


Figure 0.133: Value of α as a function of γ for $\mathtt{l}$ = 2, 3, 4, 5 and $\mathtt{r}$ = 6.

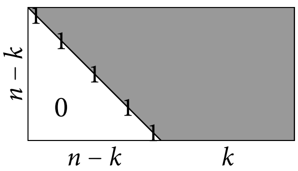


Figure 0.134: H in upper triangular form.

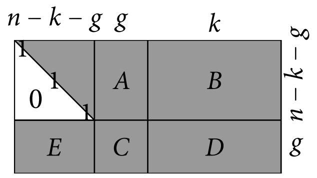


Figure 0.135: H in approximate upper triangular form.

INITIALIZE: Set $H_0 = H$ and $t = g = 0$. Go to CONTINUE.

CONTINUE: If $t = n-k-g$ then stop and output H_t. Otherwise, if the minimum residual degree is 1 go to EXTEND, else go to CHOOSE.

EXTEND: Choose uniformly at random a column $\mathtt{c}$ of residual degree 1 in H_t. Let $\mathtt{r}$ be the row (in the range $[t+1, n-k-g]$) of H_t that contains the (residual) non-zero entry in column $\mathtt{c}$. Swap column $\mathtt{c}$ with column $t+1$ and row $\mathtt{r}$ with row $t+1$. (This places the non-zero element at position $(t+1, t+1)$, extending the diagonal by 1.) Call the resulting matrix H_{t+1}. Increase t by 1 and go to CONTINUE.

CHOOSE: Choose uniformly at random a column $\mathtt{c}$ in H_t with minimum positive residual degree, call the degree d. Let $\mathtt{r}_1, \mathtt{r}_2, \ldots, \mathtt{r}_d$ denote the rows of H_t in the range $[t+1, n-k-g]$ which contain the d residual non-zero entries in column $\mathtt{c}$. Swap column $\mathtt{c}$ with column $t+1$. Swap row $\mathtt{r}_1$ with row $t+1$ and move rows $\mathtt{r}_2, \mathtt{r}_3, \ldots, \mathtt{r}_d$ to the bottom of the matrix. Call the resulting matrix H_{t+1}. Increase t by 1 and increase g by $d-1$. Go to CONTINUE.

Figure 0.136: Greedy algorithm to perform approximate upper triangulation.

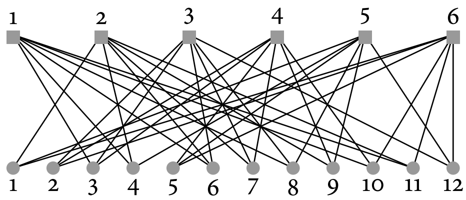


Figure 0.137: Tanner graph corresponding to H_0.

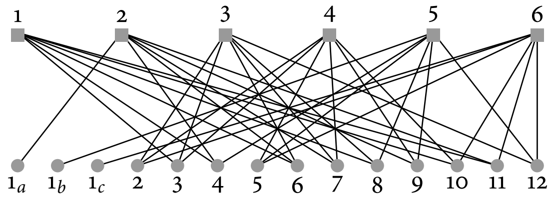


Figure 0.138: Tanner graph after splitting of node 1.

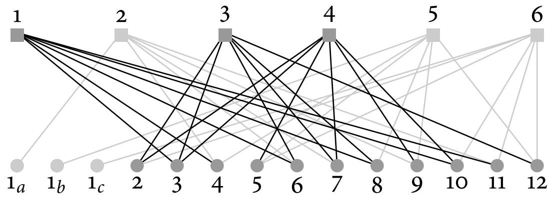


Figure 0.139: Tanner graph after one round of dual erasure decoding.

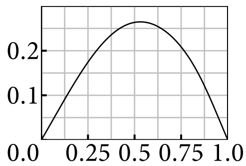


Figure 0.140: $1 - z - \rho(1 - \lambda(z))$.

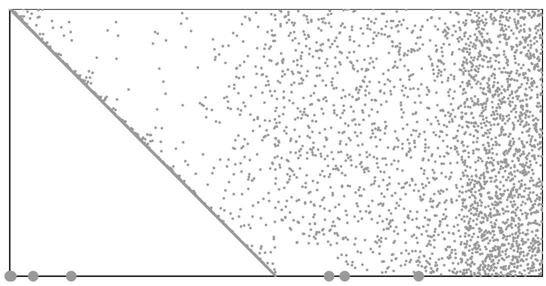


Figure 0.141: Element chosen uniformly at random from LDPC $(1024, \lambda, \rho)$, with (λ, ρ) as described in Example A.19, after the application of the greedy algorithm. For the particular experiment we get $g = 1$. The non-zero elements in the last row (in the gap) are drawn larger to make them more visible.

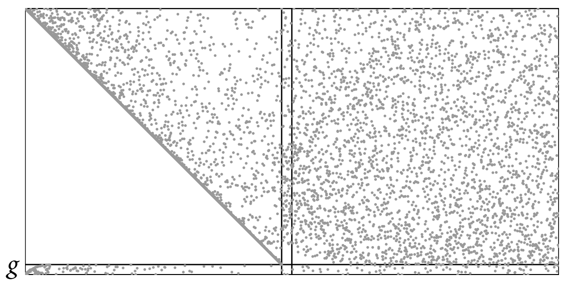


Figure 0.142: Element chosen uniformly at random from LDPC $\left(2048, x^2, x^5\right)$ after the application of the greedy algorithm. The result is $g = 39$.

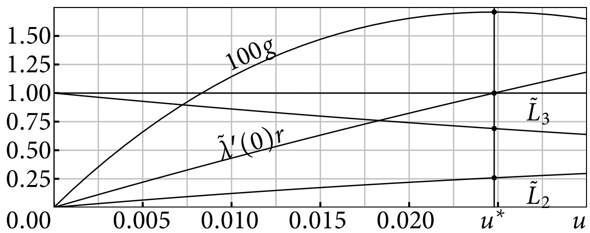


Figure 0.143: Evolution of the differential equation for the $(3,6)$-regular ensemble. For $u^* \approx 0.0247856$ we have $\tilde{\lambda}'(0)r = 1$, $\tilde{L}_2 \approx 0.2585$, $\tilde{L}_3 \approx 0.6895$, and $g \approx 0.01709$.

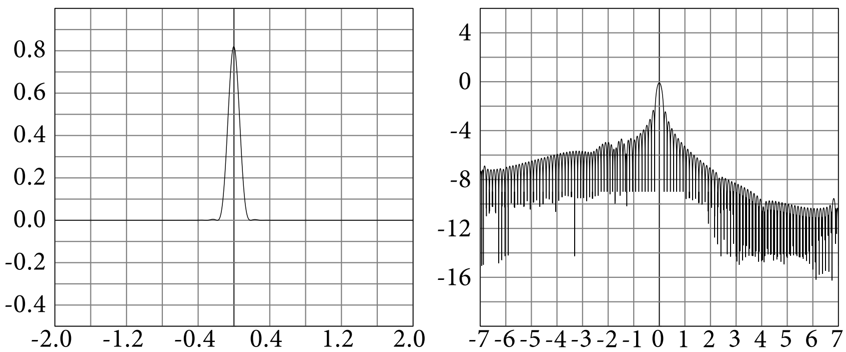


Figure 0.144: Left: Example $\mathfrak{K}(x)$, $\delta = 0.125$. Right: Logarithm (base 10) $\mathfrak{K}(x)$.

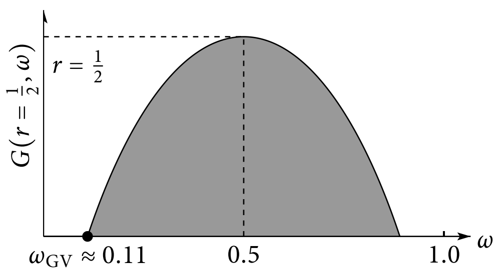


Figure 0.145: Exponent $G(r = 1/2, \omega)$ of the weight distribution of typical elements of $\mathcal{G}(n, k = n/2)$ as a function of the normalized weight ω. For $w/n \in (\delta_{GV}, 1 - \delta_{GV})$ the number of codewords of weight w in a typical element of $\mathcal{G}(n, k)$ is $2^{n(G(r,w/n)+o(1))}$.

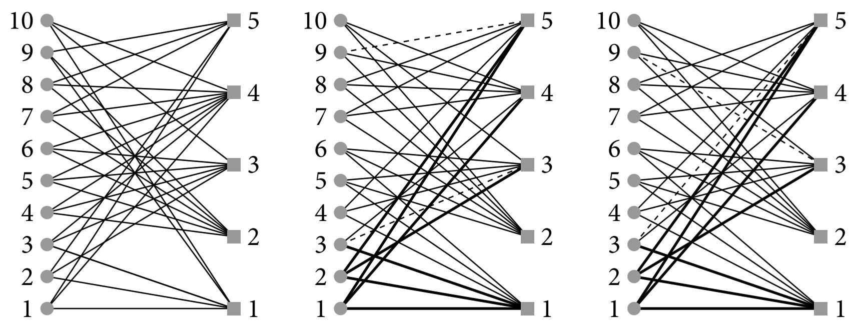


Figure 0.146: Left: Graph $\mathtt{G}$ from the ensemble LDPC $\left(10, x^2, x^5\right)$; Middle: Graph $\mathtt{H}$ from the ensemble $\mathcal{G}_7(\mathtt{G}, 7)$ (note that the labels of the sockets are not shown – these labels should be inferred from the order of the connections in the middle figure); the first 7 edges that $\mathtt{H}$ has in common with $\mathtt{G}$ are drawn in bold; Right: the associated graph $\phi_{7,30}(\mathtt{H})$. The two dashed lines correspond to the two edges whose endpoints are switched.

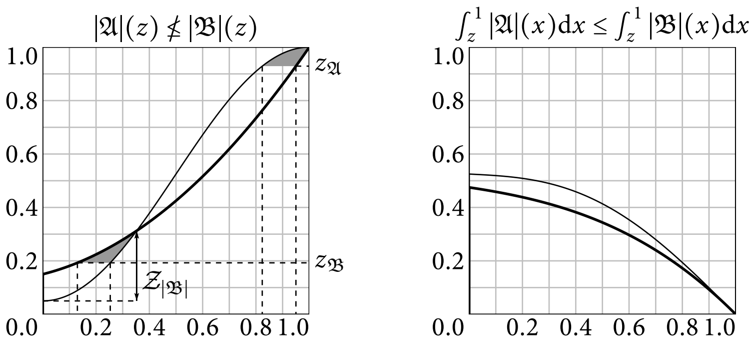


Figure 0.147: Left: Two $|D|$-distributions $|\mathfrak{A}|$ (thick line) and $|\mathfrak{B}|$ (thin line). Right: Since $\int_z^1 |\mathfrak{A}|(x)\mathrm{d}x \leq \int_z^1 |\mathfrak{B}|(x)\mathrm{d}x$ we know that $|\mathfrak{A}| \multimap |\mathfrak{B}|$.

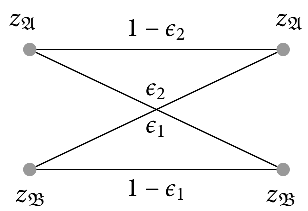


Figure 0.148: Definition of q on $(z_{\mathfrak{B}}, z_{\mathfrak{A}})$ pair.