Finite-Length Analysis of a Capacity-Achieving Ensemble for the Binary Erasure Channel

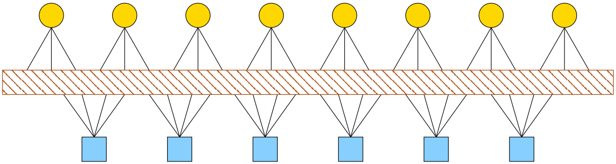
H. D. Pfister
Swiss Federal Institute of Technology, Lausanne (EPFL)

Information Theory Workshop
Rotorua, New Zealand
September 1st, 2005

Outline

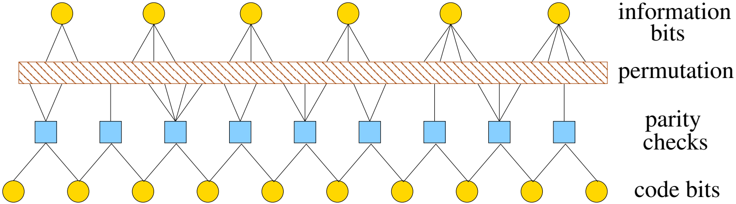
Outline

Low Density Parity Check (LDPC) Codes



- Linear codes with sparse parity-check matrix H
  - Regular (j, k): H has j ones per column and k ones per row
  - Irregular (λ, ρ): uses degree distributions for ones in H
- Bipartite Graph
  - An edge connects check node i to bit node j if $H_{ij} = 1$
  - Used for *message passing iterative* (MPI) decoding

Irregular Repeat-Accumulate (IRA) Codes



- Can be viewed either as a Turbo or LDPC variation
  - LDPC: Simply add zig-zag structured degree 2 bits
  - Turbo: Repeat info bits, parity-check, and accumulate
- Repeat-parity given by sparse generator matrix G
  - Information bit j included in parity check i if $G_{ij} = 1$
  - Regular (j, k): G has j ones per column and k ones per row
  - Irregular (λ, ρ): uses degree distributions for ones in G

Degree Distributions and Density Evolution

- Definition: *degree distribution* (d.d) function

$$\lambda(x) = \sum_{i\geq 1} \lambda_i x^{i-1} \qquad \rho(x) = \sum_{i\geq 1} \rho_i x^{i-1}$$

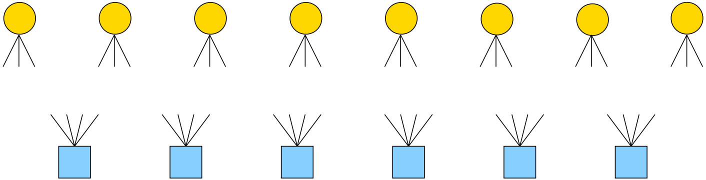
  - λ_i = Fraction of edges attached to bits of degree i
  - ρ_i = Fraction of edges attached to checks of degree i

- Density evolution (DE)
  - Tracks distribution of messages during iterative decoding
  - Long codes decode almost surely if DE converges
  - For BEC, let x_i = erasure rate of bit output messages

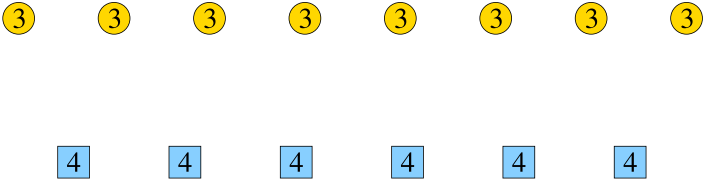
$$x_{i+1} = p\lambda\left(1 - \rho(1 - x_i)\right)$$

Outline

Peeling Style Analysis of LDPC Codes (Luby et al.)



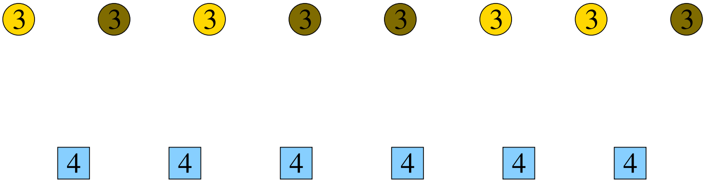
Peeling Style Analysis of LDPC Codes (Luby et al.)



$$\{0, 0, 0, 24\}$$

Number of Edges by Degree

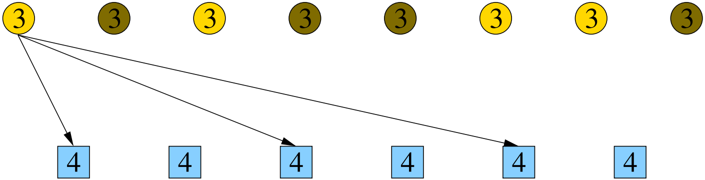
Peeling Style Analysis of LDPC Codes (Luby et al.)



$$\{0, 0, 0, 24\}$$

Number of Edges by Degree

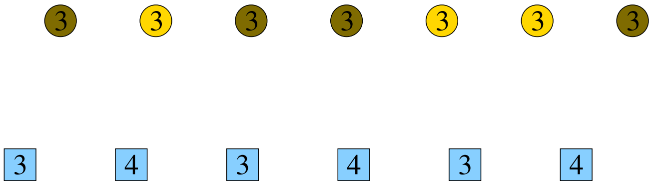
Peeling Style Analysis of LDPC Codes (Luby et al.)



$$\{0, 0, 0, 24\}$$

Number of Edges by Degree

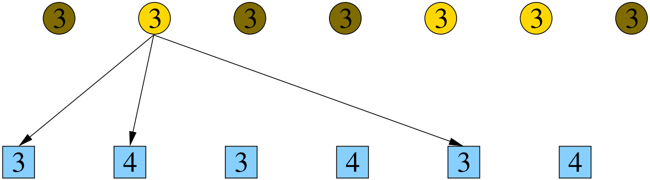
Peeling Style Analysis of LDPC Codes (Luby et al.)



$$\{0, 0, 9, 12\}$$

Number of Edges by Degree

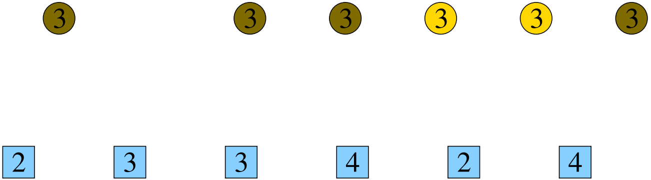
Peeling Style Analysis of LDPC Codes (Luby et al.)



$$\{0, 0, 9, 12\}$$

Number of Edges by Degree

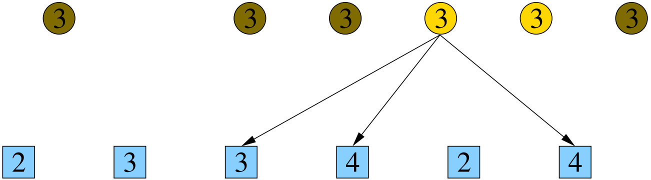
Peeling Style Analysis of LDPC Codes (Luby et al.)



$$\{0, 4, 6, 8\}$$

Number of Edges by Degree

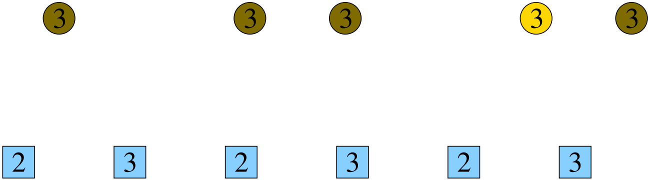
Peeling Style Analysis of LDPC Codes (Luby et al.)



$$\{0, 4, 6, 8\}$$

Number of Edges by Degree

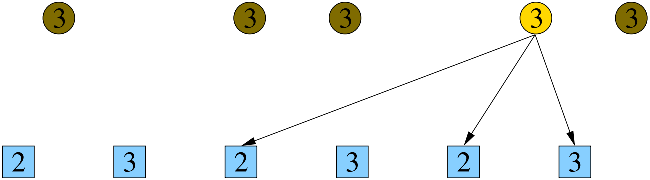
Peeling Style Analysis of LDPC Codes (Luby et al.)



$$\{0, 6, 9, 0\}$$

Number of Edges by Degree

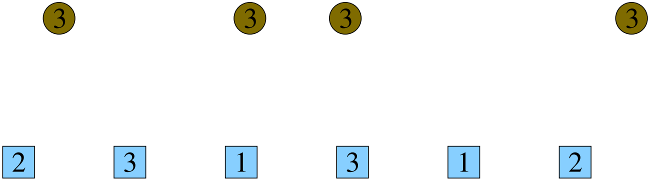
Peeling Style Analysis of LDPC Codes (Luby et al.)



$$\{0, 6, 9, 0\}$$

Number of Edges by Degree

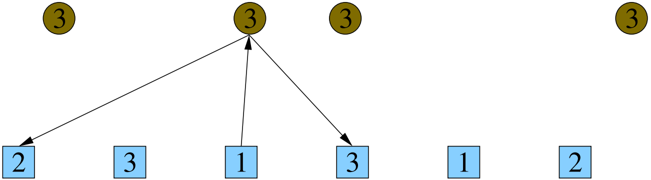
Peeling Style Analysis of LDPC Codes (Luby et al.)



$$\{2, 4, 6, 0\}$$

Number of Edges by Degree

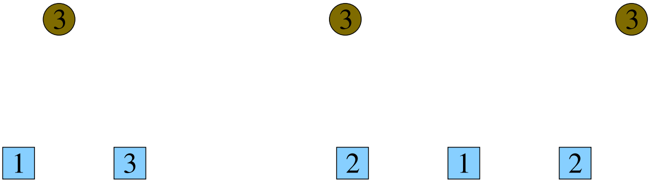
Peeling Style Analysis of LDPC Codes (Luby et al.)



$\{2, 4, 6, 0\}$

Number of Edges by Degree

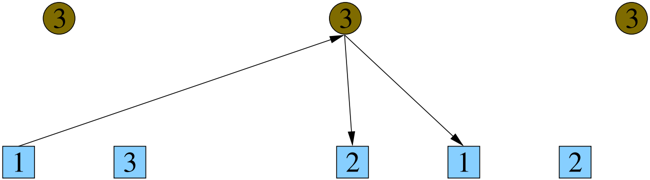
Peeling Style Analysis of LDPC Codes (Luby et al.)



$$\{2, 4, 3, 0\}$$

Number of Edges by Degree

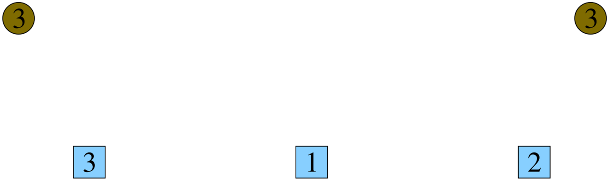
Peeling Style Analysis of LDPC Codes (Luby et al.)



$$\{2, 4, 3, 0\}$$

Number of Edges by Degree

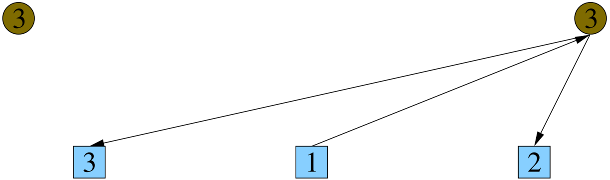
Peeling Style Analysis of LDPC Codes (Luby et al.)



$\{1, 2, 3, 0\}$

Number of Edges by Degree

Peeling Style Analysis of LDPC Codes (Luby et al.)



$$\{1, 2, 3, 0\}$$

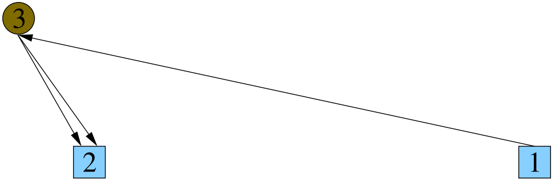
Number of Edges by Degree

Peeling Style Analysis of LDPC Codes (Luby et al.)

$$\{1, 2, 0, 0\}$$

Number of Edges by Degree

Peeling Style Analysis of LDPC Codes (Luby et al.)



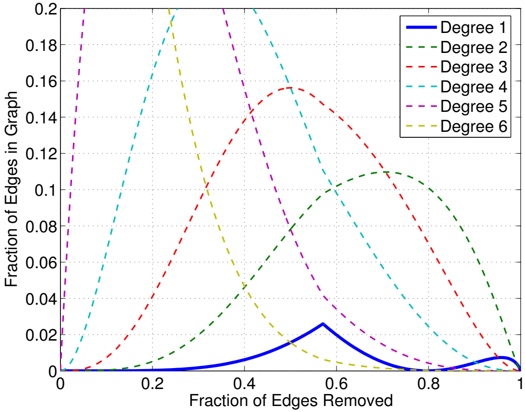
$$\{1, 2, 0, 0\}$$

Number of Edges by Degree

Peeling Style Analysis of LDPC Codes (Luby et al.)

Decoding Successful

Mean Trajectory for the (3,6) LDPC Code



- *Critical point* is where the fraction of deg. 1 edges is zero

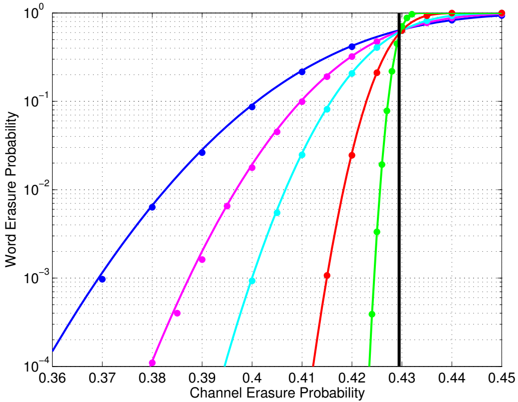
Finite Length Scaling for LDPC Codes

- Refined analysis of peeling style decoding (Amraoui et al.)
  - Number of bit and check edges asymptotically Gaussian
    - Use differential equations to track the mean and covariance
  - Probability of block error versus block length n given by

$$P_B = Q\left(\frac{\sqrt{n}(p^* - \beta n^{-2/3} - p)}{\alpha}\right) + o(1)$$

  - Exact in the limit as $n \to \infty$ with $\sqrt{n}\,(p^* - p)$ held constant
- Parameters defined in the neighborhood of the critical point
  - α related to std. dev. of number of degree 1 edges
  - β related to width of parabola at the critical point

Scaling Results for (3,6) LDPC



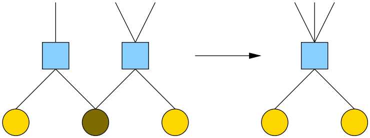
- Parameters: $p^* = 0.42944$, $\alpha = 0.56036$, and $\beta = 0.61695$
- Block length: $n = 1024, 2048, 4096, 16384, 131072$
- Outer code assumed to eliminate small stopping sets

Covariance Evolution for LDPC Decoding

- Bit Regular Decoding
  - Assume we start with n check edges
  - Let $X_{n,i}^{(j)}$ be the number of deg. j check edges after i steps
  - Number check edges of each deg. is a Markov process
- Phase 1: Remove $(1-p^*)n$ edges for known bits
  - Pick random edge $\sim X_{n,i}^{(j)}/(n-i)$
  - If deg. k, replace k deg. k edges with $k-1$ deg. $k-1$ edges
  - Differential eq. for mean and covariance (Amraoui et al.)
- Phase 2: Remove $(t_{crit}-1+p^*)n$ edges for decoding
  - Remove a degree 1 edge
  - Repeat $d-1$ times: Remove random edge as above
  - Differential eqns for mean and covariance (Amraoui et al.)
- Parameter α given by the variance of degree 1 edges

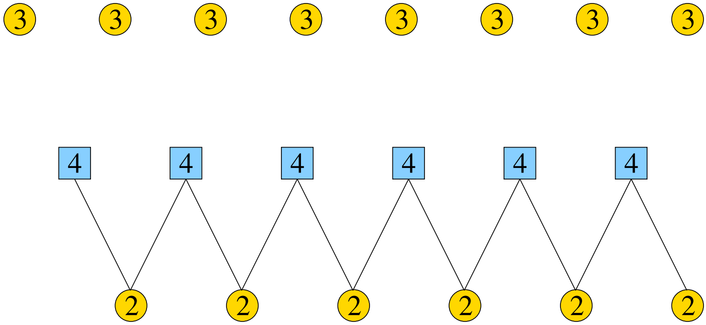
Outline

Graph Reduction For IRA Codes

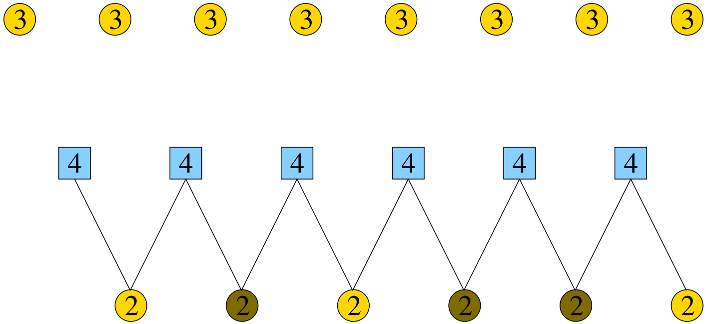


- *Graph reduction* removes all code bits from the graph
  - Peeling style decoding removes all known code bits
  - Merging check nodes removes all erased code bits
    - Equivalent to summing check equations to remove bit
- After graph reduction we have
  - A standard LDPC code with a modified check d.d.
  - Check d.d. is random and depends on erased code bits
- Straightforward generalization of scaling also possible
  - A *degree vector* for each node, but complexity increased

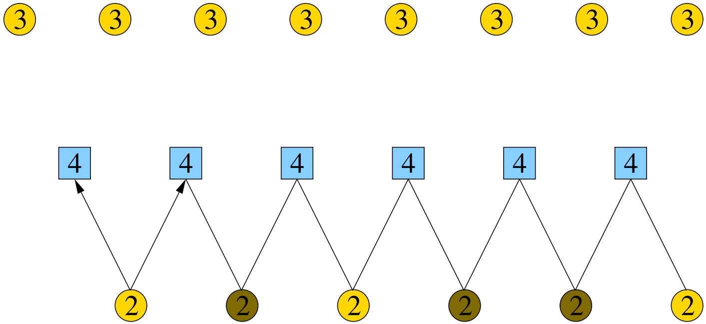
Peeling Style Analysis for IRA Codes



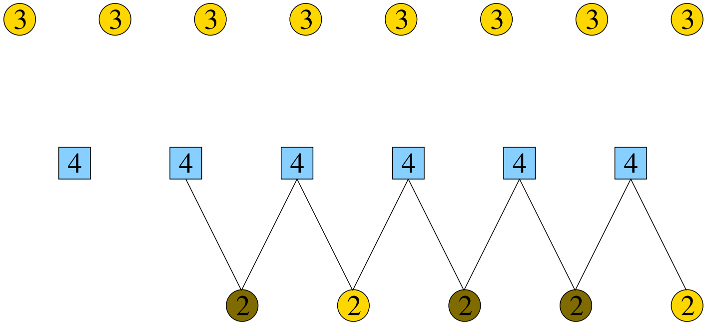
Peeling Style Analysis for IRA Codes



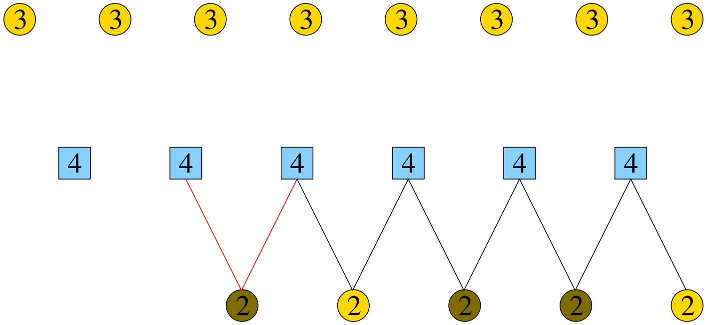
Peeling Style Analysis for IRA Codes



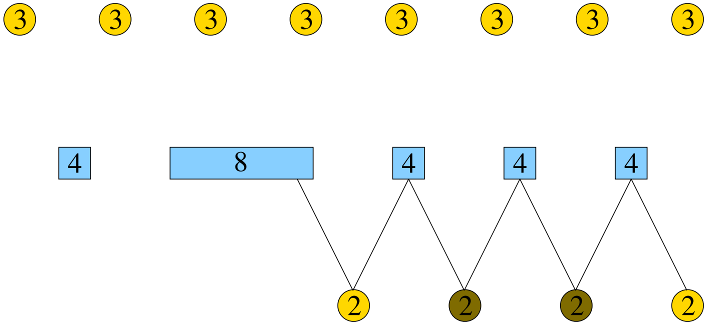
Peeling Style Analysis for IRA Codes



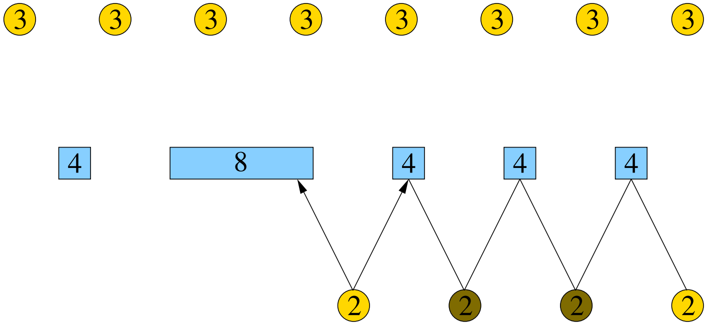
Peeling Style Analysis for IRA Codes



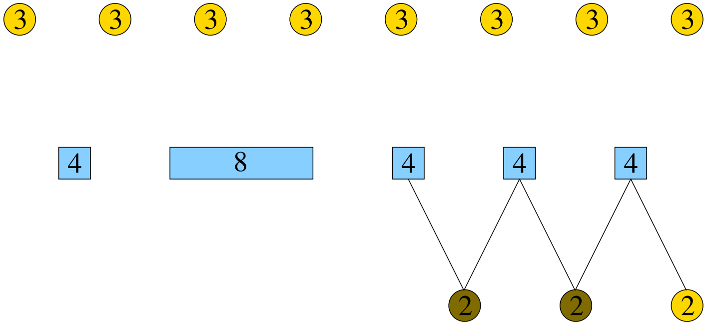
Peeling Style Analysis for IRA Codes



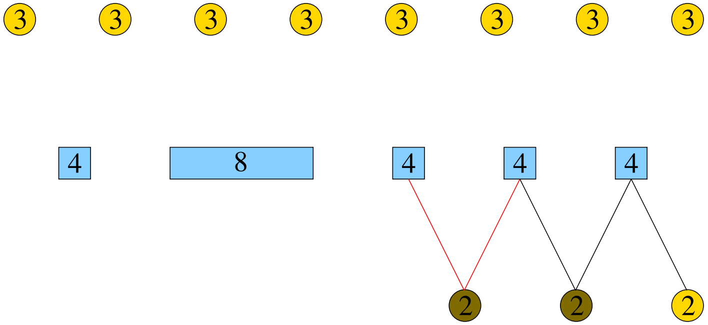
Peeling Style Analysis for IRA Codes



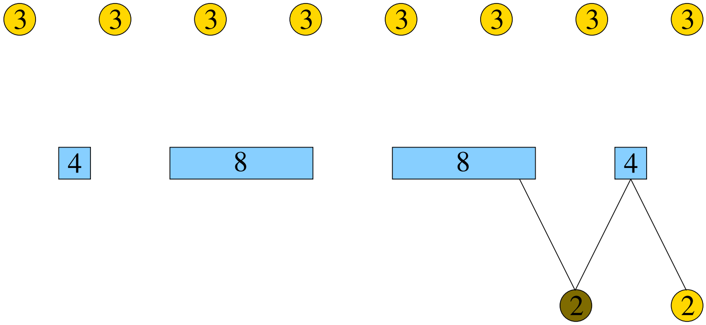
Peeling Style Analysis for IRA Codes



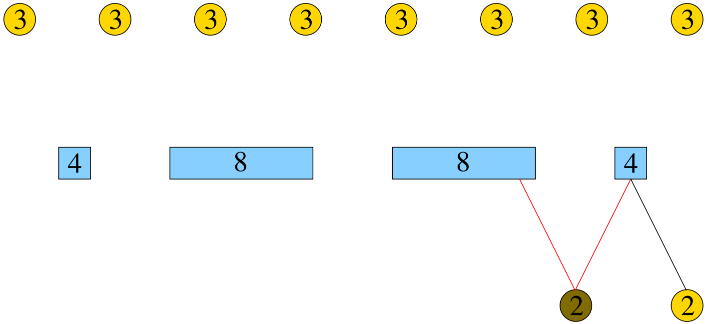
Peeling Style Analysis for IRA Codes



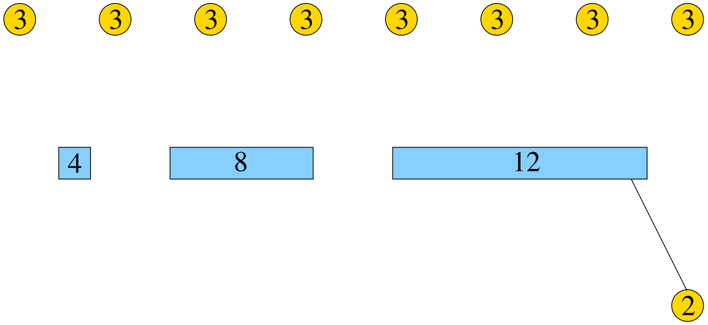
Peeling Style Analysis for IRA Codes



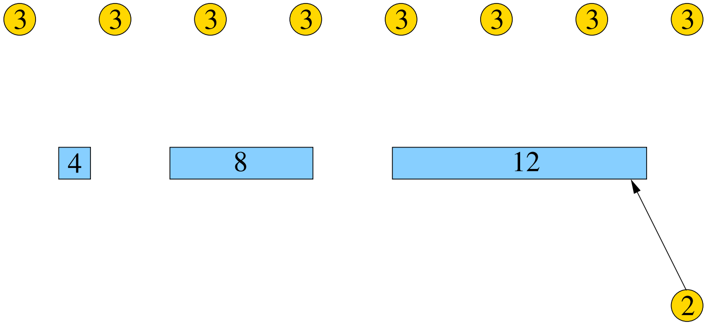
Peeling Style Analysis for IRA Codes



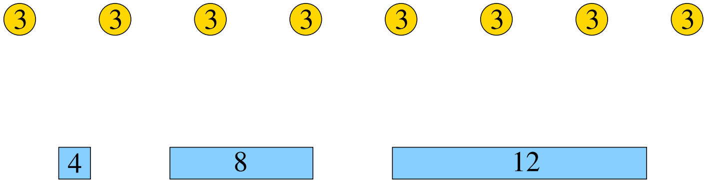
Peeling Style Analysis for IRA Codes



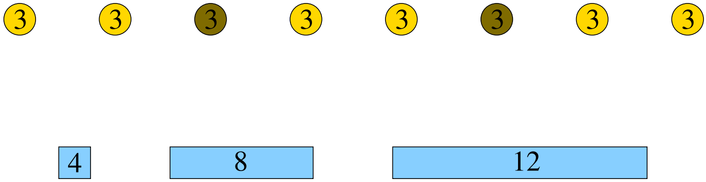
Peeling Style Analysis for IRA Codes



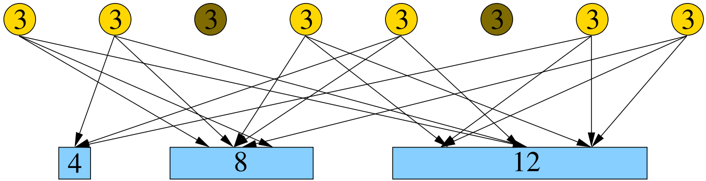
Peeling Style Analysis for IRA Codes



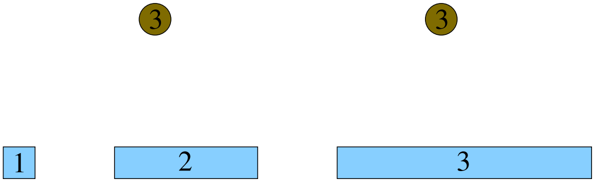
Peeling Style Analysis for IRA Codes



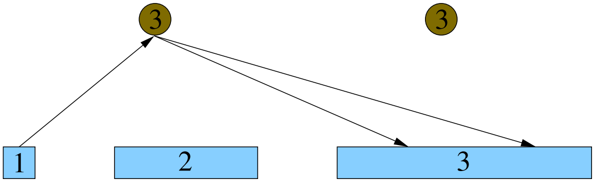
Peeling Style Analysis for IRA Codes



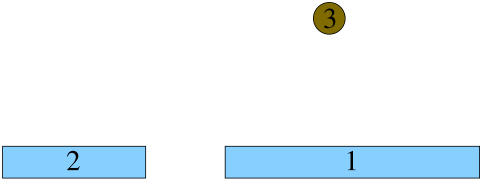
Peeling Style Analysis for IRA Codes



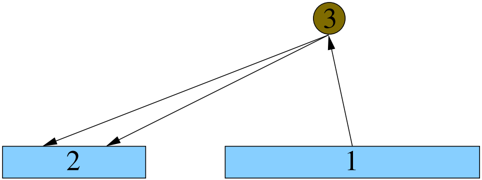
Peeling Style Analysis for IRA Codes



Peeling Style Analysis for IRA Codes



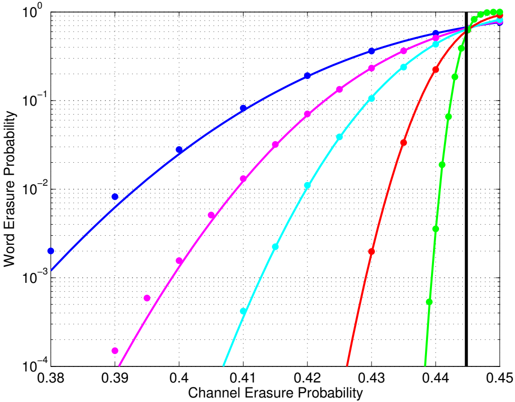
Peeling Style Analysis for IRA Codes



Peeling Style Analysis for IRA Codes

Decoding Successful

Rate 1/2 Systematic (3,3) IRA Code



- Parameters: $p^* = 0.44478$ $\alpha = 0.59588$ $\beta = 0.83874$
- Block length: $n = 1024, 2048, 4096, 16384, 131072$
- Outer code assumed to eliminate small stopping sets

Covariance Evolution for Graph Reduction

- Graph Reduction (starting with n checks)
  - Number check nodes of each deg. is a Markov process
  - State $X_{n,i}^{(j)}$ is number of checks of degree j after i steps
  - $X_{n,0}^{(j)} = n\,R_j$ where R_j is the fraction of check nodes deg. j
  - For each erasure, pick two checks and combine

$$Pr(\deg j,\ \deg k \rightarrow \deg j+k) = \frac{X_{n,i}^{(j)} X_{n,i}^{(k)}}{(n-i)(n-i)} + O\left(\frac{1}{n}\right)$$

  - This is sufficient to apply the theorem of Amraoui et al.
- Conversion to edge perspective (to continue decoding)
  - Number of edges deg. j after i steps: $Y_{n,i}^{(j)} = \frac{j\,X_{n,i}^{(j)}}{\sum_k k\,R_k}$

Outline

Approaching Capacity in Practice

- The biggest obstacle is the enormous block length required
  - Irregular LDPC codes limited by length, not complexity
  - Length 10^7 used for Chung's 0.04 dB from capacity result
- Block length vs. gap to capacity for iterative decoding?
  - First, need a capacity achieving sequence of ensembles
  - Second, need to pick a block length for each ensemble
  - Empirically: If length grows too slowly, performance is bad
- Two Approaches
  - Scaling law: Determine $\{p^*, \alpha, \beta\}$ for c.a. sequence
  - Weight enumerator: Focus on low weight codewords

Capacity-Achieving LDPC Codes for the BEC

- A seq. of codes is *capacity-achieving* (c.a.) on a channel
  - If DE converges for each code in the sequence
  - Sequence of code rates converges to channel capacity
- Complexity of iterative decoding
  - Proportional to number of edges in the graph
- Check regular c.a. sequence $\{\lambda^{(k)}, \rho^{(k)}\}$ (Shokrollahi)
  - Let $\rho^{(k)}(x) = x^k$ and $\widetilde{\lambda}^{(k)}(x) = \frac{1}{p}\left(1-(1-x)^{1/k}\right)$
  - $\lambda^{(k)}(x)$ given by truncating series for $\widetilde{\lambda}^{(k)}(x)$ so $\lambda^{(k)}(1) = 1$
  - Complexity grows like $\ln \frac{1}{\varepsilon}$ for gap to capacity ε

Capacity-Achieving LDPC Codes for the BEC

- A seq. of codes is *capacity-achieving* (c.a.) on a channel
  - If DE converges for each code in the sequence
  - Sequence of code rates converges to channel capacity
- Complexity of iterative decoding
  - Proportional to number of edges in the graph
- Check regular c.a. sequence $\{\lambda^{(k)}, \rho^{(k)}\}$ (Shokrollahi)
  - Let $\rho^{(k)}(x) = x^k$ and $\widetilde{\lambda}^{(k)}(x) = \frac{1}{p}\left(1-(1-x)^{1/k}\right)$
  - $\lambda^{(k)}(x)$ given by truncating series for $\widetilde{\lambda}^{(k)}(x)$ so $\lambda^{(k)}(1) = 1$
  - Complexity grows like $\ln \frac{1}{\varepsilon}$ for gap to capacity ε

Capacity-Achieving LDPC Codes for the BEC

- A seq. of codes is *capacity-achieving* (c.a.) on a channel
  - If DE converges for each code in the sequence
  - Sequence of code rates converges to channel capacity
- Complexity of iterative decoding
  - Proportional to number of edges in the graph
- Check regular c.a. sequence $\left\{\lambda^{(k)}, \rho^{(k)}\right\}$ (Shokrollahi)
  - Let $\rho^{(k)}(x) = x^k$ and $\widetilde{\lambda}^{(k)}(x) = \frac{1}{p}\left(1 - (1-x)^{1/k}\right)$
  - $\lambda^{(k)}(x)$ given by truncating series for $\widetilde{\lambda}^{(k)}(x)$ so $\lambda^{(k)}(1) = 1$
  - Complexity grows like $\ln \frac{1}{\varepsilon}$ for gap to capacity ε

Capacity-Achieving IRA Codes for the BEC

- Density Evolution (Turbo style decoding):

$$x_{i+1} = \lambda \left(1 - \left(\frac{1-p}{1-pR(1-x_i)}\right)^2 \rho(1-x_i)\right)$$

- Bit regular non-sys. IRA Ensemble $\lambda(x) = x^2$ (deg. 3)

$$\rho(x) = \sum_{i\geq 1} \rho_i x^{i-1} = \frac{1-(1-x)^{1/2}}{\left(1-p\left(1-3x+2\left(1-(1-x)^{3/2}\right)\right)\right)^2}$$

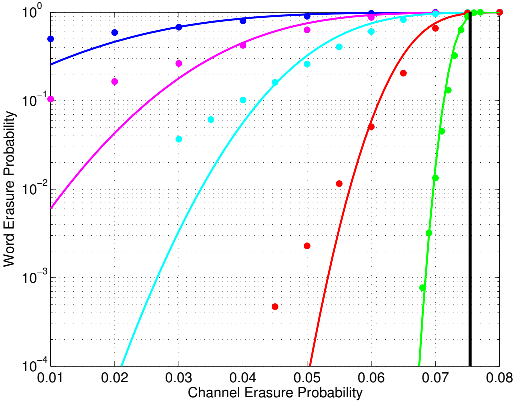
- Sequence of ensembles $\left\{\lambda, \rho^{(M)}\right\}$ by truncation of $\rho(x)$
  - where $\rho_M(x) = \sum_{i=2}^{M-1} \rho_i x^{i-1} + \sum_{i=M}^{\infty} \rho_i x^{M-1}$
  - Capacity achieving for $p \leq 1/13$
  - Complexity converges to $3 + \frac{2}{1-p}$ and $\varepsilon = O\left(M^{-1/2}\right)$

Scaling for Capacity-Achieving IRA Sequence

Code	γ	Rate	p^*	α	β
IRA M=20	.0019	.9126	.0754	.4122	2.938
IRA M=30	.0021	.9173	.0754	.4842	4.079
IRA M=40	.0020	.9194	.0754	.5684	5.462
IRA M=50	.0019	.9206	.0753	.6577	7.017
IRA M=60	.0017	.9214	.0753	.7491	8.737

- Bit regular (degree 3) c.a. non-systematic IRA codes
- Design rate = 0.925, γ = fraction of sys. bits transmitted
- Parameter α rising slowly, but β rising quickly
    - Need bounds $\overline{\alpha}_M, \overline{\beta}_M$ on α, β as a function of M
    - Then, choose n_M so $\overline{\alpha}_M n_M^{-1/2}$ and $\overline{\beta}_M n_M^{-2/3}$ are bounded

Capacity-Achieving IRA Sequence M=40



- Parameters: $p^* = 0.754$ $\alpha = 0.5684$ $\beta = 5.462$
- Block length: $n = 1024, 2048, 4096, 16384, 131072$
- Real problem: Scaling law convergence not uniform

Weight Enumerator (WE) Analysis

- An IRA encoder is the serial concatenation of a
  - Repeat code IOWE: $A^{(rep)}_{p,s} = \binom{nR'(1)/3}{p}\delta_{s,3p}$
  - Parity code IOWE: $A^{(par)}_{s,q} \leq \binom{n}{q}(R'(1))^q \frac{\left(\frac{1}{2}nR''(1)\right)^k}{k!}\delta_{s-2k,q}$
  - Accumulate code CIOWE: $A^{(acc)}_{q,\leq w} \leq \binom{n}{\lfloor q/2 \rfloor}\frac{w^{\lceil q/2 \rceil}}{\lceil q/2 \rceil !}$

$$\overline{A}^{(IRA)}_{p,\leq w} = \sum_{s,q} A^{(rep)}_{p,s} \frac{A^{(par)}_{s,q}}{\binom{nR'(1)}{s}} \frac{A^{(acc)}_{q,\leq w}}{\binom{n}{q}}$$

- Notice the $R''(1)$ in $A^{(par)}_{s,q}$
  - For this sequence, we find that $R''(1) = \Theta\left(M^{1/2}\right)$
  - For fixed n, we find $d_{min} \to 0$ as M increases
  - For fixed M, we find $d_{min} \geq n^{1/3-\varepsilon}$ as n increases
  - Fixed input wt., $n = \Omega\left(M^{3/2}\right)$ sufficient for $d_{min} \geq n^{1/3-\varepsilon}$

Conclusions

- Block length vs. gap to capacity for iterative decoding
    - The real obstacle for capacity achieving codes
- Finite length scaling law
    - Has great potential for this problem
        - Problem A: Parameters require numerical methods
        - Problem B: Non-uniform convergence
        - Can we get upper/lower bounds on n instead?
- Weight Enumerator Analysis
    - Required to prove convergence to zero erasures
        - Gives lower bounds on n
        - Needs refinement to prove $d_{min} = \Omega\left(n^{1/3-\varepsilon}\right)$