Weight Distribution for Non-binary Cluster LDPC Code Ensemble

Takayuki Nozaki[1], Masaki Maehara[2], Kenta Kasai[2], Kohichi Sakaniwa[2]

[1]Kanagawa University

[2]Tokyo Institute of Technology

July 12th, 2013

Introduction (1)

Non-binary cluster LDPC codes [SD2011]

- One of a class of non-binary LDPC code
- A generalization of non-binary LDPC code defined over general linear group
- Good decoding performance in low rate codes

Contribution of this research

For irregular non-binary cluster LDPC code ensemble,

- We derive the symbol and bit weight distributions
- We show the growth rate of symbol and bit weight distributions
- We give a condition that the relative minimum distance linearly grows with the code length

[SD2011] V.Savin and D.Declercq, "Linear growing minimum distance of ultra-sparse non-binary cluster-LDPC codes," ISIT2011, pp.523–527, July-Aug. 2011

Introduction (2)

Weight distribution for non-binary LDPC code ensembles

Gallager	Gallager code ensemble over $\mathbb{Z}/q\mathbb{Z}$	symbol-WD
Kasai et al.	Irregular ensemble over $\mathbb{F}_{2^m}$	symbol and bit-WD
Andriyanova et al.	Regular ensemble over $\mathrm{GL}(m, \mathbb{F}_2)$	bit-WD
This research	Irregular cluster ensemble	symbol and bit-WD

Related research for minimum distance of non-binary cluster LDPC codes

- [SD2011] shows that there exist $(2, d_\mathrm{c})$-regular expurgated ensemble whose minimum distance linearly grows with the code length
- This research shows a condition that the relative minimum distance linear grows with the code length for random irregular non-binary cluster LDPC code ensemble.

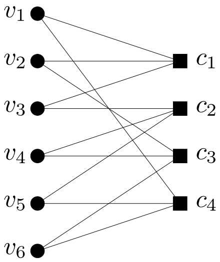
Outline

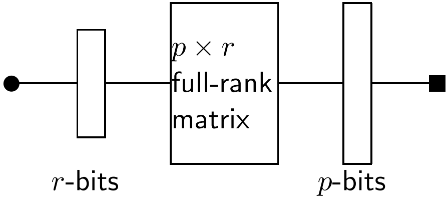
Non-binary Cluster LDPC Code

[Definition] Non-binary cluster LDPC code

- Defined by a Tanner graph G
- Each edge in Tanner graphs is labeled by *cluster*, which is $p \times r$ full-rank matrix
- A symbol (r-bits) is assigned to each variable node

$$C(\mathsf{G}) := \left\{ (\boldsymbol{x}_1, \dots, \boldsymbol{x}_N) \in (\mathbb{F}_2^r)^N \mid \sum_{i \in \mathcal{N}_c(j)} h_{j,i} \boldsymbol{x}_i^T = \boldsymbol{0}^T \ \forall j \in [1, M] \right\}$$





Irregular Non-binary Cluster LDPC Code Ensemble

[Definition] Irregular non-binary cluster LDPC code ensemble $\mathcal{G}(N, p, r, \lambda, \rho)$

- N : the number of variable nodes (symbol code length)
- p, r : size of cluster
- λ, ρ : degree distribution pair (edge perspective)

- Bit code length $n := rN$
- Degree distribution pair (node perspective) (L, R)

$$L_i := \lambda_i / \left(i \int_0^1 \lambda(x)dx\right), \quad R_j := \rho_j / \left(j \int_0^1 \rho(x)dx\right)$$

- Total number of edges $E := N / \int_0^1 \lambda(x)dx$
- Fraction of check nodes κ

$$\kappa := \frac{\#(\text{check nodes})}{\#(\text{variable nodes})} = \frac{\int_0^1 \rho(x)dx}{\int_0^1 \lambda(x)dx}$$

- Design rate

$$1 - \kappa p / r$$

Outline

Weight of Codeword

[Definition] Symbol and bit weight of a codeword

- Symbol weight of a codeword $\boldsymbol{x}$

$$w(\boldsymbol{x}) := |\{i \in [1, N] \mid \boldsymbol{x}_i \neq \mathbf{0}\}|$$

- Bit weight of a codeword $\boldsymbol{x}$

$$w_{\mathrm{b}}(\boldsymbol{x}) := |\{(i, j) \in [1, N] \times [1, r] \mid x_{i,j} \neq 0\}|$$

[Definition] Average number of codeword of weight ℓ

For an ensemble $\mathcal{G} = \mathcal{G}(N, p, r, \lambda, \rho)$,

- average number of codeword of symbol weight ℓ

$$A(\ell) = |\mathcal{G}|^{-1} \textstyle\sum_{\mathtt{G} \in \mathcal{G}} |\{\boldsymbol{x} \in C(\mathtt{G}) \mid w(\boldsymbol{x}) = \ell\}|$$

- average number of codeword of bit weight ℓ

$$A_{\mathrm{b}}(\ell) = |\mathcal{G}|^{-1} \textstyle\sum_{\mathtt{G} \in \mathcal{G}} |\{\boldsymbol{x} \in C(\mathtt{G}) \mid w_{\mathrm{b}}(\boldsymbol{x}) = \ell\}|$$

Symbol Weight Distribution

[Theorem 1] Symbol weight distribution

For the irregular non-binary cluster LDPC code ensemble $\mathcal{G}(N, p, r, \lambda, \rho)$, the average number of codewords of symbol weight ℓ is

$$A(\ell) = \sum_{k=0}^{E} A(\ell, k)$$

$$= \sum_{k=0}^{E} \frac{(2^r - 1)^\ell \mathrm{coef}\left(P(s,t)^N, s^\ell t^k\right) \mathrm{coef}\left(Q(u)^N, u^k\right)}{\binom{E}{k}(2^p - 1)^k},$$

$$P(s,t) := \prod\nolimits_{i \in \mathcal{L}} \left(1 + st^i\right)^{L_i}, \quad Q(u) := \prod\nolimits_{j \in \mathcal{R}} f_j(u)^{\kappa R_j},$$

$$f_j(u) := 2^{-p}\left[\{1 + (2^p - 1)u\}^j + (2^p - 1)(1 - u)^j\right],$$

where $\mathrm{coef}(g(s,t), s^i t^j)$ is the coefficient of the term $s^i t^j$ of a polynomial $g(s,t)$.

Bit Weight Distribution

[Theorem 2] Bit weight distribution

For the irregular non-binary cluster LDPC code ensemble $\mathcal{G}(N, p, r, \lambda, \rho)$, the average number of codewords of bit weight ℓ is

$$A_{\mathrm{b}}(\ell) = \sum_{k=0}^{E} \frac{\mathrm{coef}\left(P_{\mathrm{b}}(s,t)^n, s^{\ell} t^k\right) \mathrm{coef}\left(Q_{\mathrm{b}}(u)^n, u^k\right)}{\binom{E}{k}(2^p-1)^k},$$

$$P_{\mathrm{b}}(s,t) := \prod_{i \in \mathcal{L}} [1 + \{(1+s)^r - 1\} t^i]^{L_i/r},$$

$$Q_{\mathrm{b}}(u) := \prod_{j \in \mathcal{R}} f_j(u)^{\kappa R_j/r},$$

$$f_j(u) := 2^{-p}\left[\{1 + (2^p-1)u\}^j + (2^p-1)(1-u)^j\right].$$

Outline

Growth Rate of Weight Distribution

[Definition] Growth rate

- Growth rate of symbol weight distribution

$$\gamma(\omega) := \lim_{N\to\infty} N^{-1} \log_{2^r} A(\omega N) = \lim_{N\to\infty} (rN)^{-1} \log A(\omega N)$$

- Growth rate of bit weight distribution

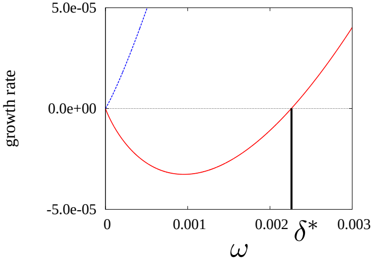
$$\gamma_{\mathrm{b}}(\omega_b) := \lim_{n\to\infty} n^{-1} \log A_{\mathrm{b}}(\omega_{\mathrm{b}} n)$$

Properties

The following approximations hold

$$A(\omega N) \approx 2^{\gamma(\omega) r N}, \quad A_{\mathrm{b}}(\omega_{\mathrm{b}} n) \approx 2^{\gamma_{\mathrm{b}}(\omega_{\mathrm{b}}) n}.$$

If $\gamma(\omega) < 0$ for $\omega \in (0, \delta^*)$, there are exponentially few codewords of symbol weight ωN.



Growth Rate of Symbol Weight Distribution

[Theorem 3] Growth rate of symbol weight distribution

Define $\omega = \ell/N$ and $\epsilon := E/N$.
For $\mathcal{G}(N, p, r, \lambda, \rho)$ with sufficiently large N and for $0 < \omega < 1$, the growth rate $\gamma(\omega)$ of symbol weight distribution is given by

$$\gamma(\omega) = \sup_{\beta>0} \inf_{s>0,t>0,u>0} r^{-1}\Big[\log P(s,t) + \log Q(u) - \epsilon h(\beta/\epsilon) \\ - \beta \log[tu(2^p - 1)] - \omega \log[s/(2^r - 1)]\Big],$$

where $h(x) := -x\log x - (1-x)\log(1-x)$ for $0 < x < 1$.

A point (s, t, u) achieving the infimum is a solution of

$$\omega = \frac{s}{P}\frac{\partial P}{\partial s}, \quad \beta = \frac{t}{P}\frac{\partial P}{\partial t}, \quad \beta = \frac{u}{Q}\frac{\partial Q}{\partial u}.$$

The value β giving the supremum needs to satisfy the stationary condition

$$\beta = (2^p - 1)tu(\epsilon - \beta).$$

Growth Rate of Bit Weight Distribution

[Theorem 4] Growth rate of bit weight distribution

Define $\omega_{\mathrm{b}} = \ell/n$ and $\epsilon_{\mathrm{b}} := E/n$.
For $\mathcal{G}(N, p, r, \lambda, \rho)$ with sufficiently large n and for $0 < \omega_{\mathrm{b}} < 1$, the growth rate $\gamma_{\mathrm{b}}(\omega_{\mathrm{b}})$ of the bit weight distribution is given by

$$\gamma_{\mathrm{b}}(\omega_{\mathrm{b}}) = \sup_{\beta_{\mathrm{b}}>0} \inf_{s>0,t>0,u>0} \Big[\log P_{\mathrm{b}}(s,t) + \log Q_{\mathrm{b}}(u) - \epsilon_{\mathrm{b}} h(\beta_{\mathrm{b}}/\epsilon_{\mathrm{b}}) - \beta_{\mathrm{b}} \log(tu(2^p-1)) - \omega_{\mathrm{b}} \log s\Big].$$

A point (s, t, u) achieving the infimum is given in a solution of

$$\omega_{\mathrm{b}} = \frac{s}{P_{\mathrm{b}}}\frac{\partial P_{\mathrm{b}}}{\partial s}, \quad \beta_{\mathrm{b}} = \frac{t}{P_{\mathrm{b}}}\frac{\partial P_{\mathrm{b}}}{\partial t}, \quad \beta_{\mathrm{b}} = \frac{u}{Q_{\mathrm{b}}}\frac{\partial Q_{\mathrm{b}}}{\partial u}.$$

The value β_{b} giving the supremum needs to satisfy the stationary condition

$$\beta_{\mathrm{b}} = (2^p-1)tu(\epsilon_{\mathrm{b}} - \beta_{\mathrm{b}}).$$

Derivations of Growth Rates of Weight Distributions

[Lemma 1] Derivation of growth rate of symbol weight distribution

The derivation of growth rate $\gamma(\omega)$ is

$$\frac{d\gamma}{d\omega}(\omega) = -\frac{1}{r}\log\frac{s(\omega)}{2^r - 1}.$$

[Lemma 2] Derivation of growth rate of bit weight distribution

The derivation of growth rate $\gamma_{\mathrm{b}}(\omega_{\mathrm{b}})$ is

$$\frac{d\gamma_{\mathrm{b}}}{d\omega_{\mathrm{b}}}(\omega_{\mathrm{b}}) = -\log s(\omega_{\mathrm{b}}).$$

Outline

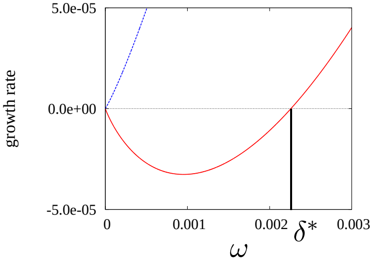
Growth Rate for Small Weight (1)

Properties of growth rate (Written again)

The following approximations hold

$$A(\omega N) \approx 2^{\gamma(\omega) r N}, \quad A_{\mathrm{b}}(\omega_{\mathrm{b}} n) \approx 2^{\gamma_{\mathrm{b}}(\omega_{\mathrm{b}}) n}.$$

If $\gamma(\omega) < 0$ for $\omega \in (0, \delta^*)$, there are exponentially few codewords of symbol weight ωN.



[Definition] Normalized typical minimum distance

Define

$$\delta^* := \inf\{\omega > 0 \mid \gamma(\omega) \geq 0\}, \quad \delta_{\mathrm{b}}^* := \inf\{\omega_{\mathrm{b}} > 0 \mid \gamma_{\mathrm{b}}(\omega_{\mathrm{b}}) \geq 0\}.$$

Properties

For $\mathcal{G}(N, p, r, \lambda, \rho)$, if the normalized typical minimum distance is strictly positive, then the typical minimum distance linearly grows with code length.

Growth Rate for Small Weight (2)

[Theorem 5] Growth rate for small symbol weight

For $\mathcal{G}(N, p, r, \lambda, \rho)$ with $\lambda_2 > 0$, the growth rate $\gamma(\omega)$ for small ω is

$$\gamma(\omega) = -\frac{\omega}{r} \log\left[\frac{2^p - 1}{(2^r - 1)\lambda'(0)\rho'(1)}\right] + o(\omega),$$

where $f(x) = o(g(x))$ means $\lim_{x \searrow 0}\left|\frac{f(x)}{g(x)}\right| = 0$ and $\lambda'(0)\rho'(1) = \lambda_2 \sum_{j \in \mathcal{R}} (j-1)\rho_j$.

[Theorem 6] Growth rate for small bit weight

For $\mathcal{G}(N, p, r, \lambda, \rho)$ with $\lambda_2 > 0$, the growth rate $\gamma_{\mathrm{b}}(\omega_{\mathrm{b}})$ for small ω_{b} is

$$\gamma_{\mathrm{b}}(\omega_{\mathrm{b}}) = -\omega_{\mathrm{b}} \log\left[\left(\frac{2^p - 1}{\lambda'(0)\rho'(1)} + 1\right)^{1/r} - 1\right] + o(\omega_{\mathrm{b}}).$$

Growth Rate for Small Weight (3)

[Corollary] Condition that typical minimum distance linearly grows with code length

For $\mathcal{G}(N, p, r, \lambda, \rho)$ with sufficiently large N, the normalized typical minimum distances δ^* and δ_{b}^* are strictly positive if

$$\lambda'(0)\rho'(1) < \frac{2^p - 1}{2^r - 1}.$$

[Remark] Comparison with other non-binary LDPC codes

- Case for non-binary LDPC code over Galois field

$$\lambda'(0)\rho'(1) < 1.$$

- Case for non-binary LDPC code defined by general linear group

$$\lambda'(0)\rho'(1) < 1.$$

Hence, for (2,d_{c})-regular non-binary LDPC code, $\delta^* = 0$ and $\delta_{\mathrm{b}}^* = 0$.

Growth Rate for (2,d_c)-Regular LDPC Code Ensemble

$$\lambda'(0)\rho'(1) < \frac{2^p - 1}{2^r - 1}.$$

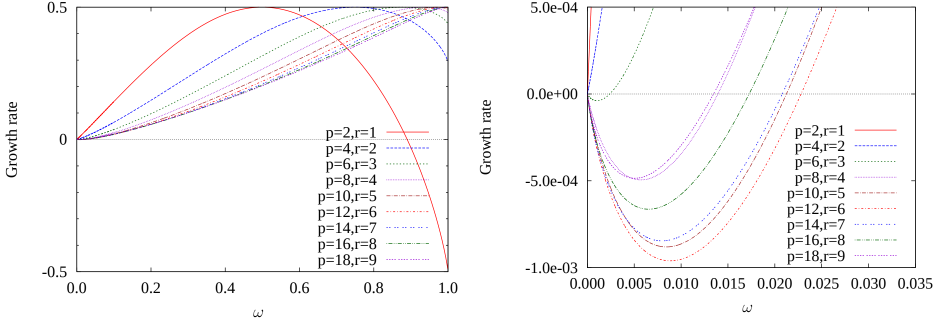
[Example] For (2,8)-regular LDPC code ensemble with rate 1/2

$\lambda'(0)\rho'(1) = 7$, $p/r = 2$.

- If $(p, r) = (2, 1)$, $\frac{2^p - 1}{2^r - 1} = 3$. Hence, $\delta^* = 0$ and $\delta_b^* = 0$
- If $(p, r) = (4, 2)$, $\frac{2^p - 1}{2^r - 1} = 5$. Hence, $\delta^* = 0$ and $\delta_b^* = 0$
- If $(p, r) = (6, 3)$, $\frac{2^p - 1}{2^r - 1} = 9$. Hence, $\delta^* > 0$ and $\delta_b^* > 0$

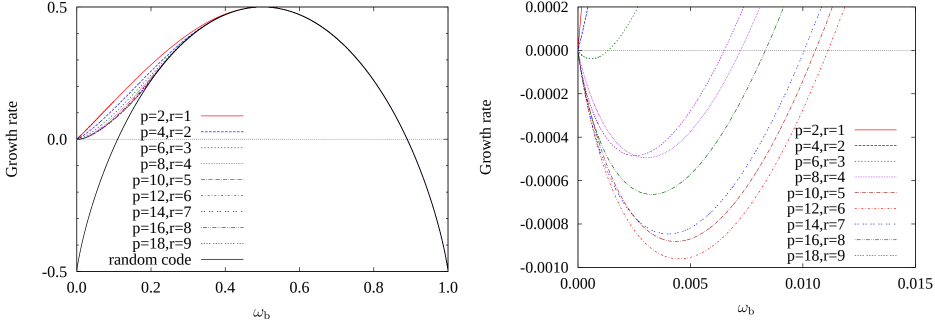
Outline

Numerical Example (1 : Symbol Weight)



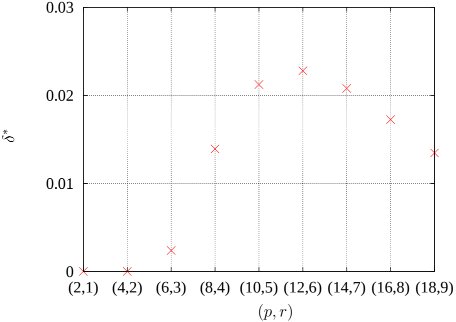
(2,8)-regular non-binary cluster LDPC code ensemble

Numerical Example (2 : Bit Weight)

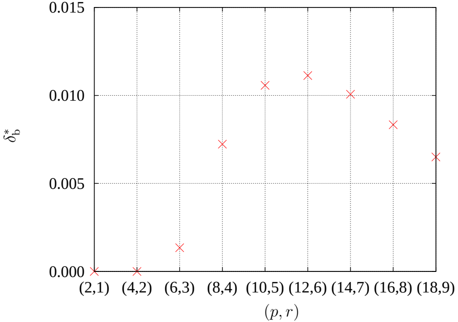


(2,8)-regular non-binary cluster LDPC code ensemble

Numerical Example (3 : Normalized Typical Minimum Distance)



Symbol case



Bit case

(2,8)-regular non-binary cluster LDPC code ensemble

Conclusion

Summary

- We derive the symbol and bit weight distributions
- We show the growth rate of symbol and bit weight distributions
- We give a condition that the relative minimum distance linearly grows with the code length

$$\lambda'(0)\rho'(1) < \frac{2^p - 1}{2^r - 1}.$$