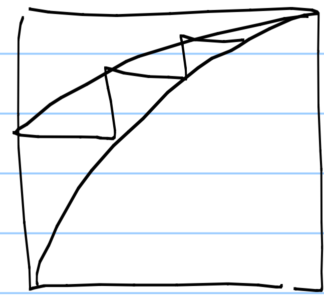
Lecture 22.

Note Title 4/2/2012

* EXIT Chart: ① Trace the mutual information
  ② Check whether there exists an open tunnel



Implication: ① decodability
② # of iterations
③ It is easier to compute
④ The area theorem

the deficiency

$$1 - R - \varepsilon^*_{DE} = \left(\sum_k \frac{\lambda_k}{k}\right)^{-1} \left(\text{area of the open tunnel}\right)$$

⑤ Curve-fitting by linear programming
when we fix $\rho(x)$ & optimize $\lambda(x)$ to fit $\rho(x)$

Var

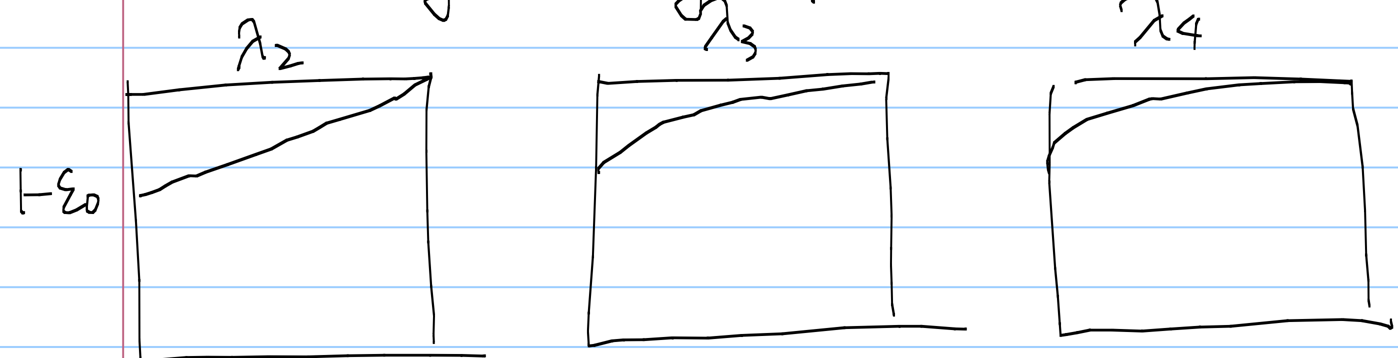
$$I_{v \to c} = 1 - (1 - I^{(0)}) \cdot \lambda(1 - I_{c \to v})$$

Chk

$$I_{c \to v} = \rho(I_{v \to c})$$

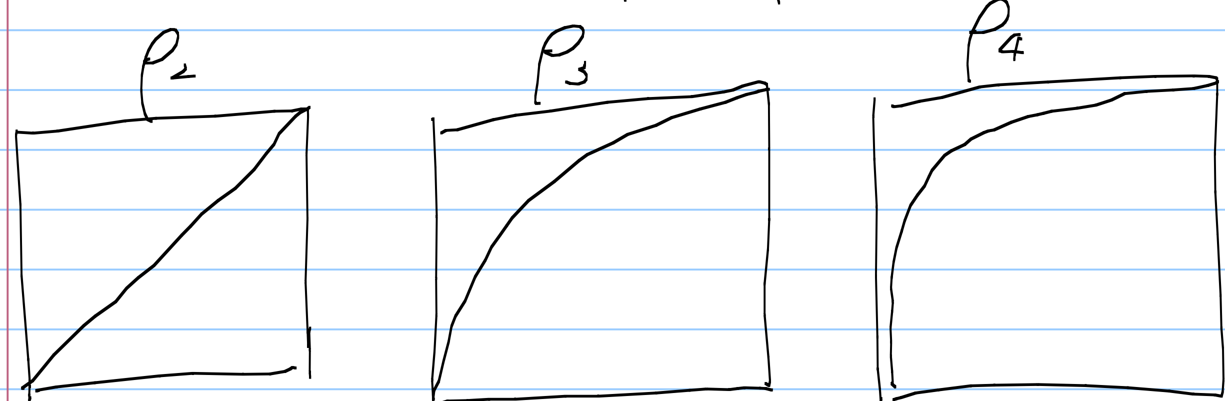
⇒ Designing an optimal code thus becomes a curve-fitting problem that would like to minimize the the area of the open tunnel.

That is the var curve $I_{v\to c} = 1 - \varepsilon_0 \lambda(1 - I_{c\to v})$ is the weighted average of



$1-\varepsilon_0(1-x)$ $\quad$ $1-\varepsilon_0(1-x)^2$ $\quad$ $1-\varepsilon_0(1-x)^3$

the check nod map $\rho(I_{v\to c}) = I_{c\to v}$



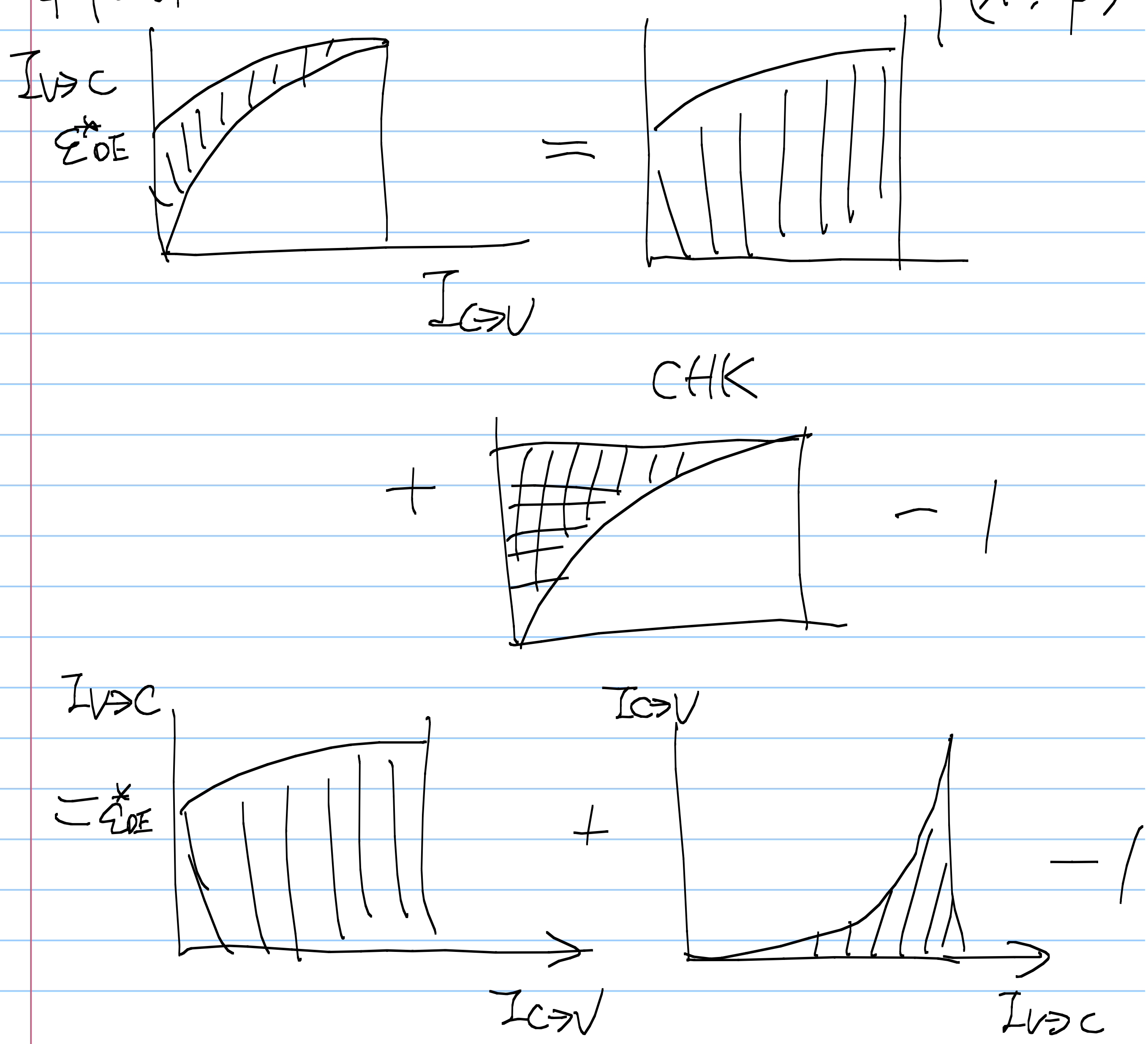
satisfying $1 - \frac{\sum \rho_k}{\sum \lambda_k} = R$ is a fixed value

If we fix $\rho(x) = \rho_{d_c} x^{d_c-1}$, the check node regular code, then the curve-fitting can be expressed as a linear programming problem with the rate-based linear constraint & the curve-fitting based objective function (or vice versa.) Specifically: the rate condition is linear
All 50pts of the curves correspond to 50 linear inequalities.

Then given ϵ_0, we can use an LP solver to search for the existence of ϵ_0-achieving (λ, ρ)

pf: of the area theorem

VAR



$$= \int_{x=0}^{1} 1 - \varepsilon_{DE}^{*} \sum_{k=2} \lambda_k (1-x)^{k-1} dx$$

$$+ \int_{x=0}^{1} \sum_{k=2} \rho_k x^{k-1} dx \quad - 1$$

$$= -\varepsilon_{DE}^{*} \sum_{k=2} \frac{\lambda_k}{k} + \sum_{k=2} \frac{\rho_k}{k}$$

Comparison

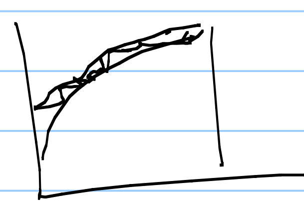
$$1 - \varepsilon_{DE}^{*} - R = 1 - \varepsilon_{DE}^{*} - \left(1 - \frac{\sum \rho_k / k}{\sum \lambda_k / k}\right)$$

$$= \frac{-\varepsilon_{DE}^{*} \sum_{k=2} \frac{\lambda_k}{k} + \sum_{k=2} \frac{\rho_k}{k}}{\sum_{k=2} \frac{\lambda_k}{k}}$$

* Perhaps the most important implication of the area theorem is that for a sequence of capacity-approaching codes with

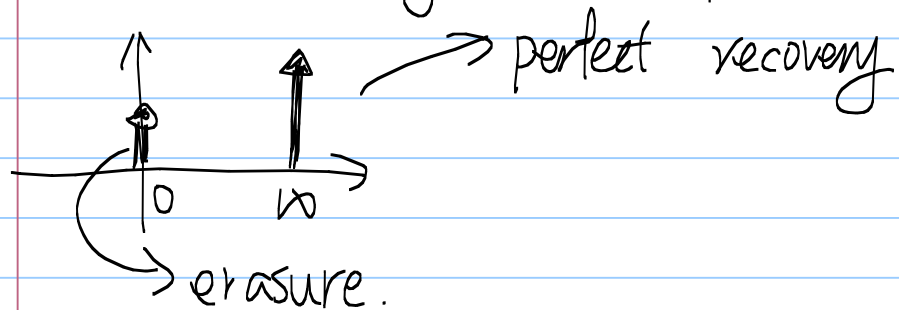
$$\varepsilon_{DE}^{*} \longrightarrow \varepsilon_{cap} = 1 - R,$$

then the # of required iterations to also tends to ∞.



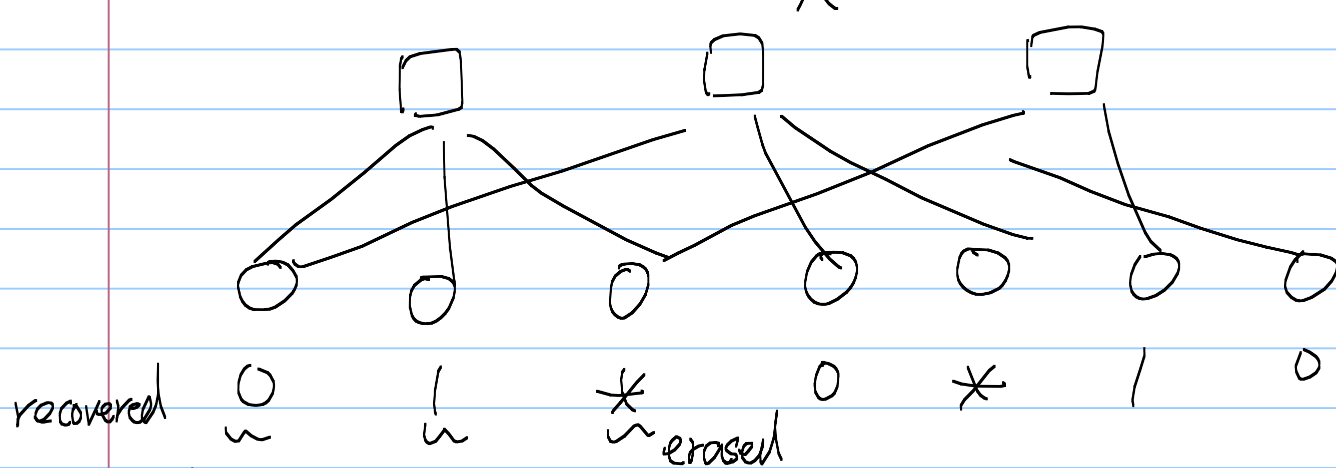
* The area theorem is so elegant, What is the intuition behind it?
* A closer look at the BEC decoding & some intuition of the area theorem.

* BEC decoding is simple. We either have



During implementation, we have the following alternative description of the BEC decoding

① For each variable x_{i_0} that is still



"erased". check its check node neighbors. If any of them is connected to all "recovered" nodes, except for x_{i_0}, then we can recover the value of x_{i_0}.

continue this process until no erased node can be recovered. This is an efficient suboptimal decoder.

$\boxed{\text{A detour}}$

Comparison: The optimal BEC decoder

Suppose the transmitted codeword is $\vec{x}$ & assume that the subvectors $\vec{x}_{recovered}$ & $\vec{x}_{erased}$ correspond to the recovered & erased positions.

$$\vec{x} = \begin{pmatrix} \vec{x}_{recovered} \\ \vec{x}_{erased} \end{pmatrix}$$

$$\Rightarrow \quad H\vec{x} = (H_0, H_1)\begin{pmatrix} \vec{x}_{recovered} \\ \vec{x}_{erased} \end{pmatrix} = 0$$

Since the receiver knows the values of $\vec{x}_{recovered}$

$$\Rightarrow \underbrace{H_1}_{\text{known}} \underbrace{\vec{x}_{erased}}_{\text{unknown}} = \underbrace{-H_0 \vec{x}_{recovered}}_{\text{known}}$$

where $H = (H_0 \; H_1)$

The optimal BEC decoder is thus to solve the matrix equation.

If we notice that $\qquad\qquad n \cdot \varepsilon$

H_1 is of dimension $n(1-R)$

where ε is the erasure prob

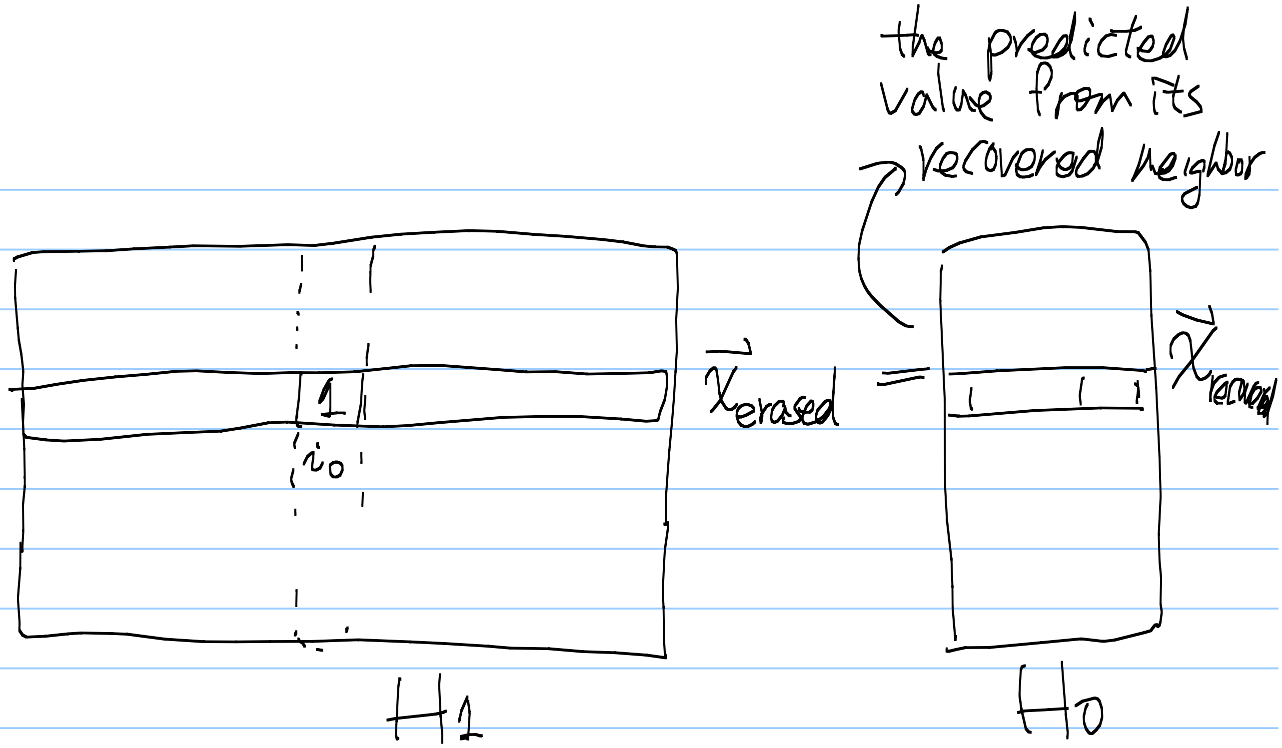
$\Rightarrow$ decodable only when $1-R \geq \varepsilon$

$$\Leftrightarrow \underbrace{1-\varepsilon}_{\text{cap}} \geq \underbrace{R}_{\text{rate}}$$

* Compare the optimal BEC decoder & the message passing (MP) decoder

$\Rightarrow$ The MP decoder tries to solve the same matrix equation by a laid-back manner:

Find a variable x_{i_0} in $\vec{x}_{erased}$ such that it participates in a parity check equation



Once x_{i_0} is recovered, move the i_0-th column of H_1 to H_0, the RHS of the equation.

$\Rightarrow$ MP decoder is a sequential solver of the matrix equation

Note that the MP decoder can operate in <u>batch</u>. The initial batch of x_{i_0} moved from H_1 to H_0 contains $n \times (1-\varepsilon)$ non-erased bits of the channel

H x

$H_1 \vec{X}_{erased} = \boxed{\text{new batch}} \vec{X}_{recovered}$ initial not-erased bits

$p^{(1)}$

$\vec{X}_{erased} = \boxed{\text{new}} \vec{X}_{recovered}$ all x_{i_0} participating in an deg-1 row.

$p^{(2)}$

new

$p^{(3)}$

How large is each batch? $p^{(1)} = p^{(0)} = \varepsilon$

Ans: Predicted by DE.

The first batch has $(1-p^{(1)}_{node})\cdot n$

The second batch has $(1-p^{(2)}_{node}) - (1-p^{(1)}_{node})$

$$= p^{(1)}_{node} - p^{(2)}_{node}$$

where $p^{(2)}_{edge} = p^{(0)}\lambda(1-\rho(1-p^{(1)}_{edge}))$

... third ...... $= p^{(3)}_{node} - p^{(2)}_{node}$

By EXIT

* This batch-based interpretation tells us many things

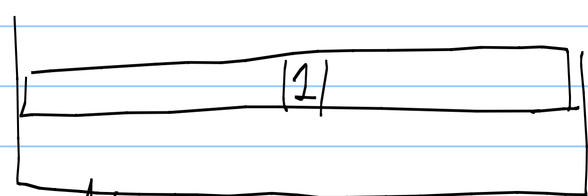
①

LDPC codes are close to optimal, but still suboptimal.

∵ Consider a regular (d_v, d_c) code, & consider the scenario after the first batch is moved to the RHS.

Since each check node equation (each row) has d_c participating var nodes

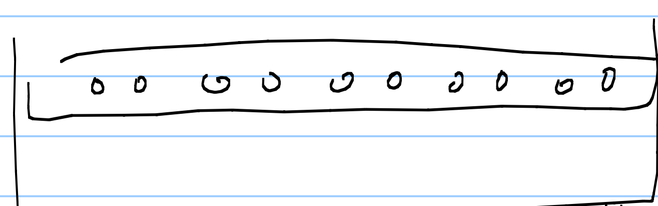
$\Rightarrow$ We will have $\binom{d_c}{1}\varepsilon \cdot (1-\varepsilon)^{d_c-1}$ of them that will have the right form



H_1

that is good for decoding

Unfortunately, $(1-\varepsilon)^{d_c}$ of them will have a bad form



H_2

$\Rightarrow$ After the first batch

$$\begin{array}{cc} & \varepsilon n \\ n(1-R) & \boxed{\quad H_1 \quad} \end{array} \vec{x}_{erased} = \boxed{\quad H_0 \quad}\, \vec{x}_{recovered}$$

but only $n(1-R)\times(1-(1-\varepsilon)^{d_c})$ of them have non-zero entries.

$\Rightarrow$ the effective matrix equation becomes

$$\begin{array}{cc} & \varepsilon n \\ n(1-R)(1-(1-\varepsilon)^{d_c}) & \boxed{\qquad\qquad} \\ & \boxed{\text{all } 0} \end{array} \vec{x}_{erased} = \boxed{\quad H_0 \quad}\, \vec{x}_{recovered}$$

$\Rightarrow$ the decodability (even by an optimal BEC solver) requires

$$(1-R)(1-(1-\varepsilon)^{d_c}) \geq \varepsilon$$

* Theorem: For any irregular LDPC code ensemble with rate R & finite maximum check node degree $\max d_c$, then the DE decodable thresholds are strictly bounded away from the capacity

$$\varepsilon_{cap} = 1-R > \underbrace{(1-R)}_{\text{the \# of rows}}\underbrace{\left(1-(1-\varepsilon^*_{DE})^{\max d_c}\right)}_{\text{effective rows after 1st batch}} \geq \varepsilon^*_{DE}$$

Even when using MAP

$$\varepsilon_{cap} = 1-R > (1-R)\left(1-(1-\varepsilon_{MAP})^{\max d_c}\right) \geq \varepsilon^*_{MAP}$$

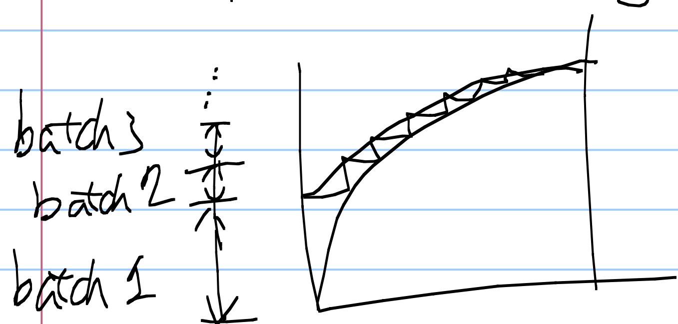
* Implication: LDPC code is suboptimal for finite $\max d_c$. To approach the capacity, d_c needs to be $\to \infty$.

However, the larger the d_c, the higher the complexity of each decoding iteration. (when $d_c = \frac{n}{2}$, LDPC codes become random linear codes.

* Fortunately, the gap diminishes exponentially with respect to $\max d_c$. Most practical LDPC codes use regular check node degree with $\rho(x) = \rho_k x^{k-1}$ for $k = 6 \sim 10$.

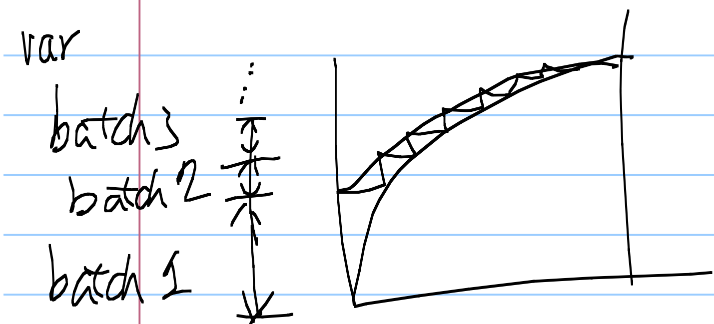
* This batch-based argument can be further strengthened.

⇒ To minimize the performance loss, or equivalently the deficiency, we would like to have each batch size being as small as possible, which is equivalent to making only small-step improvement through the EXIT chart.

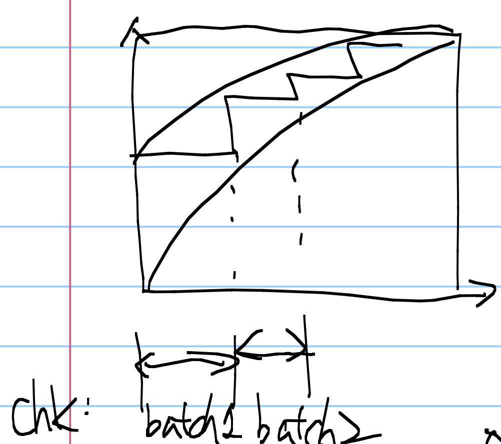


This is thus the essence of the area theorem.

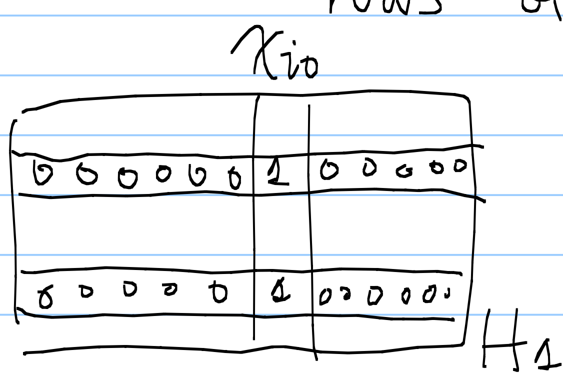
⇒ The improvement of each variable node batch needs to be small. (To avoid creating too many all-zero rows.)



A similar argument for the check node batch size can also be made.



The improvement of each check node batch needs to be small, otherwise, we may create too many x_{i_0} that participates in strictly more than one rows of deg 1.



Again the valuable check nodes are wasted.

* EXIT Chart for LDPC + Gsn channel.

For any channel other than BECs, EXIT chart is only a good approximation tool.

* EXIT Chart for LDPC codes + Gsn contains two main components.

① How to compute the mutual information from a message density traced by DE?

② Gsn approximation

For ①, Consider any BI/SO channel

$$I(X;Y) = D(P_{XY} \| P_X P_Y)$$

$$= E\left(\log_2\left(\frac{P_{XY}(X,Y)}{P_X(X)P_Y(Y)}\right)\right) \Big\downarrow \text{by symmetry}$$

$$= E\left(\log_2\left(\frac{P_{Y|X}(Y|0)}{P_Y(Y)}\right) \Big| X=0\right)$$

$$= E\left(\log_2\left(\frac{P_{Y|X}(Y|0)}{\frac{1}{2}P_{Y|X}(Y|0) + \frac{1}{2}P_{Y|X}(Y|1)}\right) \Big| X=0\right)$$

$$= E\left(\log_2 \frac{2}{1 + \frac{P_{Y|X}(Y|1)}{P_{Y|X}(Y|0)}} \,\middle|\, X=0 \right)$$

$$= E\left(\log_2 \frac{2}{1 + e^{-m}} \,\middle|\, X=0 \right)$$

$$= \int_{m=-\infty}^{\infty} \log_2 \frac{2}{1 + e^{-m}} \, d\underline{P_{m|X=0}} \text{ ——— } ①$$

↳ the unit being bits

↳ the message density traced by DE.

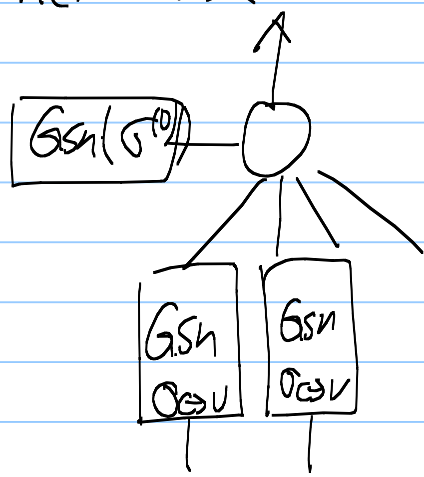
② Gsn approximation

$\boxed{\text{VAR, EXIT Curve}}$

$$I_{V\to C} = f_{var}(I^{(0)}, I_{C\to V})$$

From the given $I^{(0)}, I_{C\to V}$, construct Gsn channel with $\sigma^{(0)}, \sigma_{C\to V}$ (their message densities are $N\left(\frac{2}{(\sigma^{(0)})^2}, \frac{4}{(\sigma^{(0)})^2}\right)$, $N\left(\frac{2}{\sigma_{C\to V}^2}, \frac{4}{\sigma_{C\to V}^2}\right)$)
having mutual info
$I^{(0)}, I_{C\to V}$ as computed from ①

Then use



to compute the outgoing density. $P_{m\,out}$.

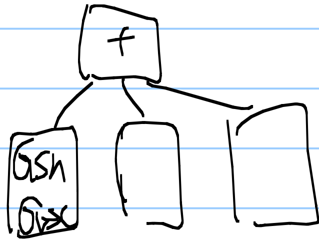
then use $P_{m\,out}$ + Φ to compute $I_{v\to c}$. Plot the curve of $I_{v\to c}$ versus $I_{c\to v}$.

CHK, EXIT curve

$$I_{c\to v} = f_{chk}(I_{v\to c})$$

Given any $I_{v\to c}$, construct a Gsn channel with $\sigma_{v\to c}$ ($dP_m \sim N(\frac{2}{\sigma^2_{v\to c}}, \frac{4}{\sigma^2_{v\to c}})$) having mutual info $I_{v\to c}$ by Φ

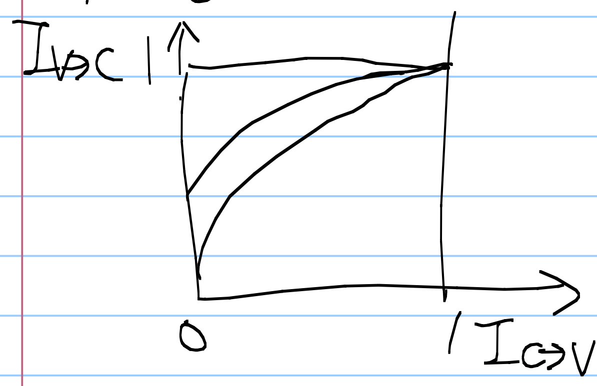
Then use



to compute the outgoing density $dP_{m,out}$.

Use $dP_{m,out}$ + Φ to compute $I_{c\to v}$, Plot $I_{v\to c}$ versus $I_{c\to v}$

We thus have the EXIT Chart for LDPC codes with general channel models.



Assuming the approximation is correct, we can again use the EXIT chart to make the same implications as we did for BEC.

* In essence, it is no different than a Gsn approximation of the density by matching the mutual information value

$$I(X;Y) = \int_{m=-\infty}^{\infty} \log \frac{2}{1+e^{-m}} \, dP_{m|X=0}$$