Lecture 14

Expectation Analysis of BP for LDPC Codes on the BEC

In the previous lecture we have seen the belief propagation (BP) decoding algorithm for LDPC codes. In this lecture we will start to analyze the performance of this algorithm, when used over the Binary Erasure Channel (BEC).

We will briefly give an overview of the characteristics of the BP algorithm that are important for the analysis performed in this lecture. Having received the variable bits with erasures we will remove from the graph all successfully received variable nodes and the edges connected to them. After this initialization step, the algorithm continues by repeating the following steps that are illustrated in Figure 14.1

1. Find an edge that is connected to a check node with degree one.
2. Remove from the graph, this edge, the variable node it is connected to, and all edges connected to this variable node.

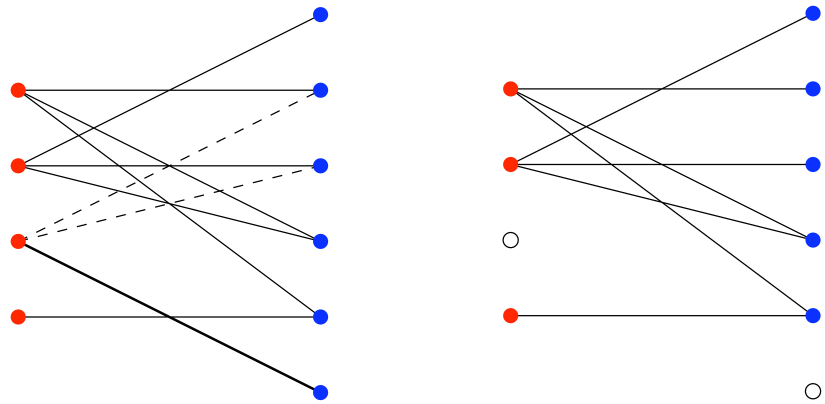
If, in the remaining graph, the algorithm continues to find edges that are connected to check nodes with degree one, the algorithm will eventually recover the values of all variable nodes. Note, that therefore we are not interested in the actual values of the variable and check nodes, but only in the induced graph after removing edges. In this lecture we will show that under certain conditions the algorithm will perform well in the sense that it will recover all variable nodes.

In general it is very difficult to analyze the performance of LDPC codes and the BP algorithm. We will simplify the analysis by not considering one specific code, but by defining a whole bunch of codes and analyzing the performance if we pick one of these codes at random. The question that we will ask ourselves is the following: *If we pick a code at random and use it to communicate over a BEC, is the BP algorithm likely to recover all erasures?* This problem formulation will be made more precise in Sections 14.1. and 14.2.. In Section 14.3. we will state our main result and in Section 14.4. and the next lecture we will give a proof.

14.1. Random Graph Model

We will consider codes/graphs that are picked uniformly at random from all graphs with specified degree distributions. More precisely fix E, the number of edges in the bipartite graph and specify $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_d)$, where λ_i is the fraction of edges that are connected to variable nodes with degree i. Similarly, let $\rho = (\rho_1, \rho_2, \ldots, \rho_\mu)$, be the degree distribution for the check nodes. In the preceding definitions we have implicitly defined, the maximum degree of the variable nodes, d, and the maximum degree of the check nodes, μ.

Lecture by Amin Shokrollahi
Scribes by Jasper Goseling



(a) Pick an edge that is connected to a check node with degree one.

(b) Remove the selected edge, the variable node it is connected to and the other edges connected to the variable node.

Figure 14.1: One step in the belief propagation algorithm.

Now pick the graph uniformly at random from all bipartite graphs with E that satisfy the degree distributions λ and ρ.

One can think of our graph as being constructed as in Figure 14.2. We create stubs of edges at our variable (check) nodes according to the degree distribution λ (resp. ρ). The graph is constructed by connecting the stubs of variable nodes to the stubs of check nodes according to a random permutation.

From the above description it is clear that by considering a random graph and removing a certain number of edges, the remaining edges form themselves a random graph (with possibly different variable and check degree distributions.)

14.2. Problem Statement

We consider the case that a random code with degree distributions λ and ρ is used over binary erasure channel with erasure probability p. Our interest is in the highest number of erasures we can tolerate such that the belief propagation algorithm still recovers all erasures. More precisely we are interested in the threshold p^* defined as

$$p^* \triangleq \sup \{p \quad | \quad \text{BP recovers all erasures } a.s.\} .$$

14.3. Main result

Let $\lambda(x) \triangleq \sum_{i=1}^{d} \lambda_i x^{i-1}$ and $\rho(x) \triangleq \sum_{i=1}^{\mu} \rho_i x^{i-1}$ be generating functions of the degree distributions.

The main result that we will prove in this lecture and the next one is given by the following theorem.

Theorem 14.1.

$$p^* = \sup \{p \quad | \quad \rho(1 - p\lambda(1-x)) > x, x \in (0,1)\} .$$

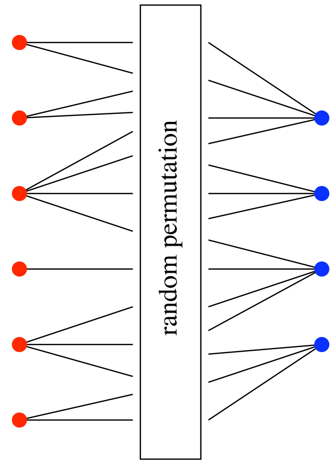


Figure 14.2: We can think of our random graph as having stubs of edges at the variable and check nodes, where the stubs are connected according to a random permutation.

14.4. Analysis

14.4.1. Notation

We denote the average degree of variable nodes by a, i.e. $a = \sum_{i=1}^{E} i\lambda_i$. Let the number of variable nodes be N. We then have the relation $N = E/a$.

Since we have an erasure of our variables with probability p, we have an expected number of erasures equal to pN. We therefore have an expected number of pN steps in our algorithm. The steps are denoted by time $t = 0, 1, 2, \ldots, pN = pE/a$. By definition, at time 0 we have removed all variable nodes corresponding to successfully received variables and the edges connected to these nodes.

In our analysis we will make use of a change of variables from time t to $\tau \triangleq t/E$. The scaled time τ runs from 0 to p/a.

Moreover, we introduce the following:

$a(t)$	average degree of variable nodes at time t
$e(t)$	fraction of edges remaining at time t
$R_t^{(i)}$	number of edges connected to a check node of degree i at time t
$r_t^{(i)}$	$\triangleq \frac{R_t^{(i)}}{E}$
$L_t^{(i)}$	number of edges connected to a variable node of degree i at time t
$l_t^{(i)}$	$\triangleq \frac{L_t^{(i)}}{E}$.

All these quantities are random variables, but we are only interested in their expected value. With a slight abuse of notation we will not explicitly take the expected value in our derivations, e.g. $e(t)$ means the expected number of edges remaining at time t.

14.4.2. Outline of the Analysis

In order for the algorithm to recover all erasures, we need a check node with degree one at all times. We will derive an expression for $r_t^{(1)}$, the fraction of edges at time t that are connected to a check node with degree one. Requiring this fraction to be positive gives the bound stated in Theorem 14.1.

We will proceed by first deriving an expression for $l_t^{(i)}$ in Section 14.4.3.. In Section 14.4.4. we will use this result to find an expression for $r_t^{(i)}$.

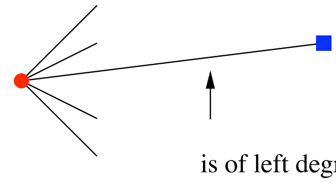


Figure 14.3: Change in number of edges connected to variable node of degree i.

14.4.3. Variable node degree

We wil start with deriving an expression for, $l_t^{(i)}$, the fraction of edges that have left degree i, $i = 1, \ldots, d$, at each time instant.

First consider $L_{t+1}^{(i)} - L_t^{(i)}$, the expected change in the number of edges with left degree i at time t. We pick an edge that is connected to a check node with degree one, remove that edge and remove all edges that are connected to it's adjacent variable node. The number of edges connected to a variable node with degree i is only affected if the edge we pick is connected to a variable node with degree 1. As explained in Figure 14.2, picking an edge connected to a check node with degree 1 does not give any information about the degree of it's variable node. The probability that the picked edge is connected to a variable node of degree i is $l_t^{(i)}/e(t)$, and in this case we lose i edges of left degree i, see Figure 14.3. We therefore have in expectation that

$$L_{t+1}^{(i)} - L_t^{(i)} = -\frac{i l_t^{(i)}}{e(t)}. \tag{14.1}$$

If we divide the numerator en the denominator on the lhs with E we get

$$\frac{\frac{L_{t+1}^{(i)}}{E} - \frac{L_t^{(i)}}{E}}{\frac{1}{E}} = -\frac{i l_t^{(i)}}{e(t)}.$$

Taking the limit for large E and making a change of variables to the scaled time $\tau = t/E$ we get the following differential equations

$$\frac{d l_\tau^{(i)}}{d\tau} = -i \frac{l_\tau^{(i)}}{e(\tau)}, \tag{14.2}$$

for $i = 1, 2, \ldots, d$.

In order to solve (14.2) we first simplify notation by making a change of variables from τ to x, by defining x so that $dx/d\tau = -x/e(\tau)$. This gives

$$\frac{d l_\tau^{(i)}}{dx} \frac{dx}{d\tau} = -i \frac{l_\tau^{(i)}}{e(\tau)}$$

and hence

$$l_i'(x) = i \frac{l_i(x)}{x}, \tag{14.3}$$

where $l_i(x) \triangleq l_x^{(i)}$. The solution of (14.3) is easily verified to be of the form

$$l_i(x) = c_i x^i$$

with c_i a constant that is to be determined from an initial condition of the problem.

Note that by using a change of variables defined through $dx/d\tau = -x/e(\tau)$, we have

$$x \triangleq \exp\left(-\int_0^\tau \frac{1}{e(s)} ds\right).$$

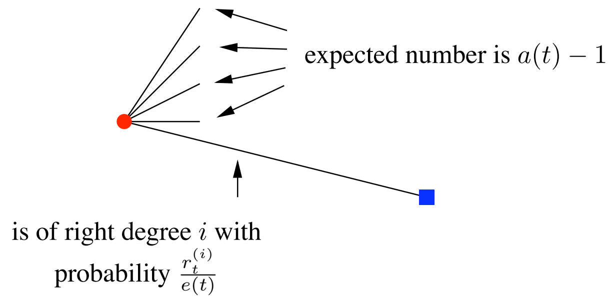


Figure 14.4: rhs

Initially, at $t = \tau = 0$ we have, therefore, $x = 1$. Since we have pE variable nodes initially,

$$l_i(1) = p\lambda_i.$$

Hence

$$l_i(x) = p\lambda_i x^i.$$

14.4.4. Check node degree

We will again take the limit for large E and make use of the change of variable from τ to x defined by $dx/d\tau = -x/e(\tau)$. For each $x \in (0, 1)$ we denote fraction of edges that are connected to check nodes of degree i, by $r_i(x)$.

Lemma 14.2. *The $r_i(x)$, $i = 2, \dots, \mu$, satisfy the following system of differential equations*

$$r_i'(x) = i(r_i(x) - r_{i+1}(x))\frac{\lambda'(x)}{\lambda(x)}.$$

Proof. We start by finding a difference equation for number of edges connected to check nodes of degree i. Figure 14.4 illustrates the algorithm. The edge with right-degree that is being removed has an expected left degree of a_t (by definition of a_t). Hence the expected number of other edges that is being removed is $a_t - 1$. These edges can affect $R_{t+1}^{(i)}$ in two ways:

- If the right degree of the edge is $i + 1$, removing this edge will create a check node with degree i. This increases the number of edges with right degree i by i.
- If the right degree of the edge is i, removing this edge will remove a right edge with degree i. We loose i edges with right degree i.

This gives the following difference equations for $i > 1$

$$\begin{aligned} R_{t+1}^{(i)} - R_t^{(i)} &= (a(t) - 1)\left(i\frac{r_{t+1}^{(i)}}{e(t)} - i\frac{r_t^{(i)}}{e(t)}\right) \\ &= i\left(r_t^{(i+1)} - r_t^{(i)}\right)\frac{a(t) - 1}{e(t)}. \end{aligned} \tag{14.4}$$

As in Section 14.4.3. we rewrite (14.4) in terms of $r_t^{(i)}$, take the large E limit and make a change of variables to τ, gives us the following set of differential equations

$$\frac{dr_\tau^{(i)}}{d\tau} = i(r_\tau^{(i+1)} - r_\tau^{(i)})\frac{a(\tau) - 1}{e(\tau)}, \tag{14.5}$$

$i = 2, \ldots, \mu$. Again making the change of variables to x, defined by $dx/d\tau = -x/e(\tau)$ gives

$$r_i'(x) = i(r_i(x) - r_{i+1}(x))\frac{a(x) - 1}{e(x)}. \tag{14.6}$$

It remains to show that $a(x) = 1 + x\frac{\lambda'(x)}{\lambda(x)}$, where $\lambda(x) \triangleq \sum_i \lambda_i x^{i-1}$ as defined in Section 14.4.1..

First note that

$$\begin{aligned}\lambda'(x) &= \sum_i (i-1)\lambda_i x^{i-2} \\ &= \frac{\sum_i i\lambda_i x^i}{x^2} - \frac{\lambda_i x^i}{x^2} \\ &= \frac{\sum_i i\lambda_i x^i}{x^2} - \frac{x\lambda(x)}{x^2},\end{aligned}$$

which gives us

$$\begin{aligned}a(x) &= \frac{p}{e(x)}\sum_i i\lambda_i x^i \\ &= \frac{p}{e(x)}\left[x^2\lambda'(x) + x\lambda(x)\right] \\ &= 1 + x\frac{\lambda'(x)}{\lambda(x)}\end{aligned}$$

where, for the last equality we used the fact that $e(x) = px\lambda(x)$.

This completes the proof of Lemma 14.2 □

Lemma 14.3. *The system of differential equations*

$$r_i'(x) = i(r_i(x) - r_{i+1}(x))\frac{\lambda'(x)}{\lambda(x)}.$$

$i = 2, \ldots, \mu$, *is satisfied by*

$$r_i(x) = \sum_{j \geq i}(-1)^{i+j}\binom{j-1}{i-1}\lambda(x)^j\gamma_j. \tag{14.7}$$

for some constants $\gamma_j, j = 2, 3, \ldots, \mu$.

Proof. First solve the homogeneous system

$$r_i'(x) = ir_i(x)\frac{\lambda'(x)}{\lambda(x)}. \tag{14.8}$$

This gives us

$$r_i(x) = \gamma_i(x)\lambda(x)^i. \tag{14.9}$$

Now to find $\gamma_i(x)$

$$r_i'(x) = \gamma'(x)\lambda^i + i\frac{\lambda'(x)}{\lambda(x)}\lambda(x)^i\gamma(x) \tag{14.10}$$

$$= \gamma'(x)\lambda^i + ir_i(x)\frac{\lambda'(x)}{\lambda(x)}, \tag{14.11}$$

which gives us

$$\gamma\lambda^i = -ir_{i+1}(x)\frac{\lambda'(x)}{\lambda(x)}. \tag{14.12}$$

Solving

$$\gamma = -i \int_0^x r_{i+1}(x)\lambda(x)^{-i} \frac{\lambda'(x)}{\lambda(x)} dx. \tag{14.13}$$

Total solution, for $i > 1$,

$$r_i(x) = \lambda(x)^i \left(-i \int_0^x r_{i+1}(x)\lambda(x)^{-i} \frac{\lambda'(x)}{\lambda(x)} dx + \gamma_i \right). \tag{14.14}$$

The claim is easily shown by induction. For $i = d$, we have

$$r_d(x) = \lambda(x)^d \left(-i \int_0^x r_{d+1}(x)\lambda(x)^{-d} \frac{\lambda'(x)}{\lambda(x)} dx + \gamma_d \right) \tag{14.15}$$

$$= \gamma_d \lambda(x)^d = \sum_{j \geq d} (-1)^{d+j} \binom{j-1}{d-1} \lambda(x)^j \gamma_j. \tag{14.16}$$

Assuming $r_{i+1}(x)$ to hold as the induction hypothesis we have

$$r_i(x) = \lambda(x)^i \left[-i \int_0^x \left(\sum_{j \geq i+1} (-1)^{i+j+1} \binom{j-1}{i} \lambda(y)^j \gamma_j \right) \lambda(y)^{-i} \frac{\lambda'(y)}{\lambda(y)} dy + \gamma_i \right] \tag{14.17}$$

$$= \sum_{j \geq i+1} \left[i(-1)^{i+j} \binom{j-1}{i} \lambda(x)^i \left(\int_0^x \lambda(y)^j \gamma_j \lambda(y)^{-i} \frac{\lambda'(y)}{\lambda(y)} dy \right) \right] + \gamma_i \lambda(x)^i \tag{14.18}$$

$$= \sum_{j \geq i+1} \left[i(-1)^{i+j} \binom{j-1}{i} \lambda(x)^i \left(\lambda(x)^{j-i} \frac{1}{j-i} \right) \right] + \gamma_i \lambda(x)^i \tag{14.19}$$

$$= \sum_{j \geq i+1} (-1)^{i+j} \binom{j-1}{i-1} \lambda(x)^j + \gamma_i \lambda(x)^i, \tag{14.20}$$

which proves Lemma 14.3. □

It remains to solve for the γ_i by using knowledge about the initial values of the $r_i(x)$.

Lemma 14.4.

$$r_i(x) = \sum_{m,j} (-1)^{i+j} \binom{j-1}{i-1} \binom{m-1}{j-1} \rho_m (p\lambda(x))^j,$$

for $i > 1$.

Proof. At $x = 1$

$$r_i(1) = \sum_{j \geq i} (-1)^{i+j} \binom{j-1}{i-1} \gamma_j. \tag{14.21}$$

We will first show that

$$\gamma_i = \sum_{j \geq i} \binom{j-1}{i-1} r_j(1) \tag{14.22}$$

For $i = d$, from (14.21) we get $r_d(1) = \gamma_d$, so (14.22) is true for $i = d$. Let (14.22) be true for $i = k, ..., d$. If show that (14.22) is true for $i = k-1$, then by induction the proof is done. Using the induction hypothesis, γ_{k-1} can be written as

$$\gamma_{k-1} = \sum_{j \geq k-1} a_j r_j(1)$$

where,

$$a_j = -\sum_{l=k-1}^{j-1} \binom{l}{k-2}\binom{j-1}{l}(-1)^{l+k-2}.$$

If we show that

$$\sum_{l=k-2}^{j-1} \binom{l}{k-2}\binom{j-1}{l}(-1)^{l} = 0, \quad (14.23)$$

we get $a_j = \binom{j-1}{k-2}$ and the proof is done. Now consider $(x+y+z)^{j-1}$. The coefficient of x^{k-2} is given by,

$$\sum_{l=k-2}^{j-1} \binom{l}{k-2}\binom{j-1}{l} z^{j-1-l} y^{l-k-2}.$$

Substituting $y=-1, z=1$, we get the coefficient of x^{k-2} to be zero and we prove (14.23) and hence (14.22).

At this point we will give an initial condition for r_j. At time 0, $x=1$, we have removed all edges that are connected to successfully received variable nodes. To the check nodes, this is as if a fraction of $1-p$ edges is removed at random. Hence, an edge whose check node had degree j before removing these edges has degree i afterwards with probability $\binom{j-1}{i-1}p^i(1-p)^{j-1}$. Thus

$$r_j(1) = \sum_{m \geq j} \rho_m \binom{m-1}{j-1} p^j (1-p)^{m-j}. \quad (14.24)$$

Using (14.24) in (14.22), and using the fact that $\binom{m-1}{j-1}\binom{j-1}{i-1} = \binom{m-1}{i-1}\binom{m-j}{j-i}$, gives us

$$\gamma_i = \sum_{m \geq i} \binom{m-1}{i-1} \rho_m p^i.$$

Plugging this into the expression given in Lemma 14.3 concludes our proof. □

Lemma 14.5.

$$r_1(x) = p\lambda(x)\left[x - 1 + \rho(1 - p\lambda(x))\right], \quad (14.25)$$

$0 \leq x \leq 1$.

Proof. Let $\tilde{r}_i(x) \triangleq r_i(x)$, with $r_i(x)$ as in Lemma 14.4 but extended to $i=1$, i.e.

$$\tilde{r}_i(x) = \sum_{m,j} (-1)^{i+j} \binom{j-1}{i-1}\binom{m-1}{j-1} \rho_m (p\lambda(x))^j,$$

$i = 1, \ldots, \mu$. Note that $r_1(x) = e(x) - \sum_{i \geq 1} \tilde{r}_i(x) + \tilde{r}_i(x)$.

We have

$$\begin{aligned}\sum_{i \geq 1} \tilde{r}_i(x) &= \sum_{m \geq j} (-1)^{j-1} \binom{m-1}{j-1} \rho_m (p\lambda(x))^j \sum_{i \leq j} (-1)^{i-1} \binom{j-1}{i-1} \\ &= p\lambda(x),\end{aligned}$$

since the inner sum equals 1 if $j=1$, and is zero otherwise. This gives us

$$\begin{aligned} r_1(x) &= e(x) - \sum_{i \geq 1} \tilde{r}_i(x) + \tilde{r}_i(x) \\ &= e(x) - p\lambda(x) + p\lambda(x) \sum_m \rho_m \sum_{j \leq m} (-1)^{j-1} \binom{m-1}{j-1} (p\lambda(x))^{j-1} \\ &= xp\lambda(x) - p\lambda(x) + p\lambda(x) \sum_m \rho_m (1 - p\lambda(x))^{m-1} \\ &= p\lambda(x)[x - 1 + \rho(1 - p\lambda(x))], \end{aligned}$$

which is the desired relation. □

Lemma 14.5 immediately gives us the bound that is given in Theorem 14.1.

In this lecture we have shown that the expected number of check nodes with degree 1 is positive under the conditions given in Theorem 14.1. We'll show in the next lecture that the random variable $r_1(x)$ concentrates. This will complete the proof to Theorem 14.1.