18.413: Error-Correcting Codes Lab March 16, 2004

Lecture 12

Lecturer: Daniel A. Spielman

12.1 Representing Probabilities, Equality nodes

It turns out that probabilities are not always the best way to represent the quantities we are considering. To see why, I'll begin with an examination of the computation at equality nodes. Say that an equality node has incomming messages p_i^{int}, for $i = 1, \ldots, k$, and we want to compute

$$p_1^{ext} = \frac{\prod_{i=2}^k p_i^{int}}{\prod_{i=2}^k p_i^{int} + \prod_{i=2}^k (1 - p_i^{int})}.$$

It turns out that this computation is much easier if we use likelihood ratios:

$$lr_i^{int} = \frac{p_i^{int}}{1 - p_i^{int}}.$$

The reason is that we have

$$lr_1^{ext} = \prod_{i=2}^k lr_i^{int}.$$

To see this, note that

$$\frac{p_1^{ext}}{1 - p_1^{ext}} = \frac{\prod_{i=2}^k p_i^{int}}{\prod_{i=2}^k (1 - p_i^{int})}.$$

This observation also makes it easier to compute p_i^{ext} for all i at once: first multiply all of the p_i^{int}s together, and then just divide out the appropriate terms as needed. However, I should warn you that this solution is better in theory than in practice: in practice it is very easy to get zero divided by zero this way.

Pushing things a little further, note that it is much easier to add than multiply. So, we could take the logs of all these terms, and then just add them. This gives the log-likelihood-ratios

$$llr_i = \log(lr_i).$$

Log-likelihood-ratios are nicer than likelihood ratios is that they treat probability values near 0 and near 1 symmetrically: going to the other side just changes the sign in the llr. On the other hand, the representation of numbers in floating point creates assymetry in likelihood ratios.

12.2 Representing Probabilities, Parity nodes

It seems like we should try to do something similar for parity nodes, and we can. Let's look again at the computation that we have to do at a parity node:

$$p_1^{ext} = \frac{1 - \prod_{i=2}^{k}(1 - 2p_i^{int})}{2}.$$

Rearranging terms, we can write this as

$$2p_1^{ext} - 1 = \prod_{i=2}^{k}(1 - 2p_i^{int}) = (-1)^{k-1}\prod_{i=2}^{k}(2p_i^{int} - 1).$$

This suggests introducing the "soft-bit",

$$\chi_i = 2p_i - 1.$$

We then have

$$\chi_1^{ext} = (-1)^{k-1}\prod_{i=2}^{k}\chi_i^{int}.$$

We would again like to consider taking logs and adding. However, we can get into trouble this way because the terms we are multiplying can be negative! To resolve this, we could separate out the signs, and write

$$\chi_1^{ext} = (-1)^{k-1} = \prod_{i=2}^{k}\mathrm{sign}(\chi_i^{int}).\sum_{i=2}^{k}\log(|\chi_i^{int}|).$$

12.3 tanh!?

If you really like log-likelihood ratios, then you might want to try to keep all of your computations in terms of them. We first recall that

$$\tanh(x/2) = \frac{e^x - 1}{e^x + 1}.$$

So,

$$\tanh(llr_i/2) = \frac{\frac{p}{1-p} - 1}{\frac{p}{1-p} + 1} = 2p - 1.$$

So, we have

$$\chi_i = -\tanh(llr_i/2).$$

As tanh is an odd function, we can push the minus sign inside if we wish, to obtain

$$\chi_i = \tanh(-llr_i).$$

Plugging this into the formula derived in the previous section, we obtain

$$-\tanh(llr_1^{ext}/2) = \tanh(-llr_1^{ext}/2) = (-1)^{k-1}\prod_{i=2}^{k}(-\tanh(llr_i^{int}/2)) = \prod_{i=2}^{k}(\tanh(llr_i^{int}/2)).$$

This implies,

$$llr_1^{ext} = -2\tanh^{-1}\left(\prod_{i=2}^{k}(\tanh(llr_i^{int}/2))\right).$$