EE388 Modern Coding Theory

Erasure decoding

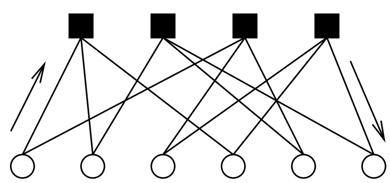
Andrea Montanari *Lecture 5-6 - 1/22/2008*

This homework is about decoding regular LDPC codes over the erasure channel.

Message passing

Consider a code with factor graph G. Each node (either check or variable) is a little processing unit (I usually imagine a little genie inside the node.) Nodes exchange messages along edges, with the objective of decoding the transmitted codeword. 'Messages' are elements of an alphabet $\mathcal{M}$, that you are free to choose at your convenience.

At any given time, a node can only use the messages it received so far, and (if it is the variable node) the channel output at its position. For the sake of simplicity, we'll assume that nodes are perfectly synchronized: at each iteration all variable nodes send out messages, and then all check nodes, and so on.

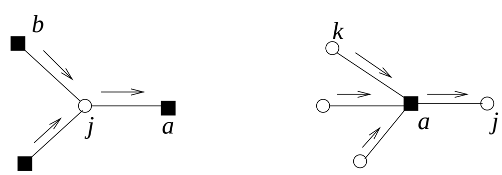


Consider now decoding a message transmitted over the BEC(ϵ):

∗ Assume the degree of G is bounded by d, and the alphabet size $|\mathcal{M}|$ to be finite. How many operations does it take to run t iterations?

∗ Is there a way to implement the *peeling* decoder in the above model?

∗ Consider now a further restriction. The message sent at any time from node i to node a only depends on messages reaching i at the previous iteration from other nodes. Is it still possible to implement the peeling decoder?



∗ In the previous, what would you use as a stopping criterion? After how many iterations will the algorithm stop under this criterion?

∗ Suppose you had to implement a message passing procedure on your computer. Which data structure would you use to store the graph G?

Density evolution

Let $x_t \in [0,1]$ be the probability that a uniformly random message from variable to check nodes is wrong after t iterations. In the limit $n \to \infty$, this satisfies the recursive equation

$$x_{t+1} = \epsilon(1-(1-x_t)^{k-1})^{l-1}\,, \tag{1}$$

with initial condition $x_0 = \epsilon$. What can you say about this recursion?

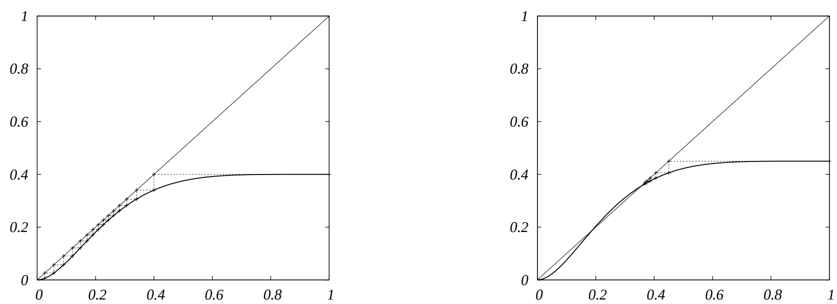


Figure 1: Density evolution for the $(3,6)$ ensemble over the erasure channel. Left $\epsilon = 0.4$, right $\epsilon = 0.45$.

Let $\lambda(z) \equiv z^{l-1}$ and $\rho(z) = z^{k-1}$. Then the recursion can be written as

$$x_{t+1} = \epsilon\lambda(1-\rho(1-x_t))\,, \tag{2}$$

or

$$\lambda^{-1}(x_{t+1}/\epsilon) = 1-\rho(1-x_t)\,. \tag{3}$$

Represent this last form of the recursion graphically.

Notice that

$$\int_0^1 \epsilon\lambda(x)\,\mathrm{d}x = \frac{\epsilon}{l}\,,\qquad \int_0^1 \rho(x)\,\mathrm{d}x = \frac{1}{k}\,. \tag{4}$$

Interpret these identities graphically. What do they imply?

WORK !!!!!

Generate random codes from the $(3,6)$ ensemble with blocklength $n = 100$, 1000, 10000. Get rid of double edges in your favourite way. Implement the message passing version of the peeling algorithm. Evaluate the bit error probability curves for communication over the BEC(ϵ). Compare them with the density evolution threshold ϵ_*.

I expect to receive

1. A print-out of the program you used.
2. A plot of error probability curves versus ϵ for the three blocklengths.
3. A three lines description of the following features of your simulation: How many channel realizations did you use? How many values of ϵ did you simulate? How much CPU time did the simulation take for each value of ϵ, and for each size?