ECE 695C, Homework #7, due date: 4/05/2012

https://engineering.purdue.edu/∼chihw/12ECE695C/12ECE695C.html

Question 1: This question is regarding a sub-optimal message passing algorithm: "Gallager Decoding Algorithm A."

The "Gallager Decoding Algorithm A" works in a very similarly way as that of the belief propagation algorithm. Basically, the output message maps for the initialization, variable and check node message maps, and the final decision are

$$m_i^{(0)} = f_{\text{ini}}(y_i) \tag{1}$$
$$m_{X_i,C_j}^{(1)} = m_i^{(0)} \tag{2}$$
$$m_{X_i,C_j}^{(t+1)} = f_{\text{var}}(m_i^{(0)}, m_{C_l,X_i}^{(t)} : l \in \partial i \backslash j) \tag{3}$$
$$m_{C_j,X_i}^{(t)} = f_{\text{chk}}(m_{X_k,C_j}^{(t)} : k \in \partial j \backslash i) \tag{4}$$
$$m_{i,\text{final}}^{(t_0)} = f_{\text{final}}(m_i^{(0)}, m_{C_j,X_i}^{(t_0)} : j \in \partial i), \tag{5}$$

where t_0 is the number of total iterations and t is the index of current iteration, and f_{ini}, f_{var}, f_{chk}, and f_{final} are specially designed functions.

The only difference between Gallager Decoding Algorithm A and Belief Propagation is that instead of using the summation and the hyperbolic tangent functions, the message maps have a simpler form, which are described as follows. Each message takes value in one of the two values $\{-1, 1\}$. For the initialization map:

$$\begin{aligned} m_i^{(0)} &= f_{\text{ini}}(y_i) \\ &\triangleq \begin{cases} 1 & \text{if } P_{Y_i|X_i}(y_i|0) > P_{Y_i|X_i}(y_i|1) \\ -1 & \text{if } P_{Y_i|X_i}(y_i|0) < P_{Y_i|X_i}(y_i|1) \end{cases}. \end{aligned} \tag{6}$$

For the variable node map, in which we use m_0 to denote $m_i^{(0)}$ and m_1 to m_{d_v-1} to denote $\{m_{C_l,X_i}^{(t)} : l \in \partial i \backslash j\}$:

$$\begin{aligned} m_{X_i,C_j}^{(t+1)} &= f_{\text{var}}(m_0, m_1, \cdots, m_{d_v-1}) \\ &\triangleq \begin{cases} -m_0 & \text{if } m_1 = m_2 = \cdots = m_{d_v-1} = -m_0 \\ m_0 & \text{otherwise} \end{cases}. \end{aligned} \tag{7}$$

For the check node map, in which we use m_1 to m_{d_c-1} to denote $\{m_{X_k,C_j}^{(t)} : k \in \partial j \backslash i\}$:

$$\begin{aligned} m_{C_j,X_i}^{(t)} &= f_{\text{chk}}(m_1, \cdots, m_{d_c-1}) \\ &\triangleq \prod_{k=1}^{d_c-1} m_k. \end{aligned} \tag{8}$$

For the final decision rule, in which we use m_0 to denote $m_i^{(0)}$ and m_1 to m_{d_v} to denote $\{m_{C_j,X_i}^{(t)} : j \in \partial i\}$:

$$m_{i,\text{final}}^{(t_0)} = f_{\text{final}}(m_0, m_1, \cdots, m_{d_v}) \triangleq \begin{cases} -m_0 & \text{if } m_1 = m_2 = \cdots = m_{d_v} = -m_0 \\ m_0 & \text{otherwise} \end{cases}. \tag{9}$$

And if the final message is $m_{i,\text{final}}^{(t_0)} = 1$, we declare $\hat{X}_i = 0$. If $m_{i,\text{final}}^{(t_0)} = -1$, we declare $\hat{X}_i = 1$.

Considering the regular (3,6) LDPC code ensemble, please answer the following questions:

1. Consider binary symmetric channels with crossover probability p, write down the density evolution formula for Gallager Decoding Algorithm A.

2. Use the binary search plus density evolution to determine the largest threshold p^* that a Gallager Decoding Algorithm A can decode.

   (For your information, the belief propagation decoder can decode $p^* = 0.084$. The Gallager Decoding Algorithm A (GDAA) is a suboptimal decoder that has inferior performance but much lower complexity. Many people also consider it as a quantized version of the BP algorithm. Please think about why the GDAA might be considered as a quantization of the BP algorithm.)