EECS 290S: Network Information Flow	**Fall 2008**
Lecture 24 — November 20	
Lecturer: Anant Sahai and David Tse	*Scribe: Amin Aminzadeh Gohari*

24.1 Review

Recall that in the previous lecture, we considered a layered Gaussian relay network (such as the one shown in Figure 24.1) and suggested the following coding scheme: Intuitively speaking, the communication begins by the source which is located in the first layer of the network graph. The relays located in the i^{th} layer listen to the signals transmitted by the relays in the $i-1^{th}$ layer. After receiving a noisy version of a superposition of the codewords of the relays in the $i-1^{th}$ layer, the relays in the i^{th} layer simultaneously turn on and send codewords to relays of the $i+1^{th}$ layer. The decoder, located at the last layer, decodes the input message. More specifically, the communication scheme can be briefly described as follows:

1. The source transmits T-length $\overrightarrow{X_s}(m)$ from a random codebook, where T is a natural number, m is the message and $\overrightarrow{X_s}(m)$ is the codeword transmitted by the source.
2. Each relay does the following after receiving the signals coming from relays of their previous layer, (see Figure 24.2):
   - The relay node quantizes the received signal (in a coordinate by coordinate basis) to precision q. The quantization can be done by finding the nearest quantization points to the received signals (i.e. mapping each signal to a quantization bin).
   - The relay node then transmits random codewords of length T based on the bin indices from the quantization.
3. The destination decodes the message.

Having explained the coding strategy, we then compared the performance of this scheme with the cut-set bound[1]. This led to the identification of three type of losses in this network:

1. Beamforming loss
2. Discretization loss
3. "Header accumulation" loss

[1]The cut-set bound is a general outer bound that is applicable to any multiterminal communication network.

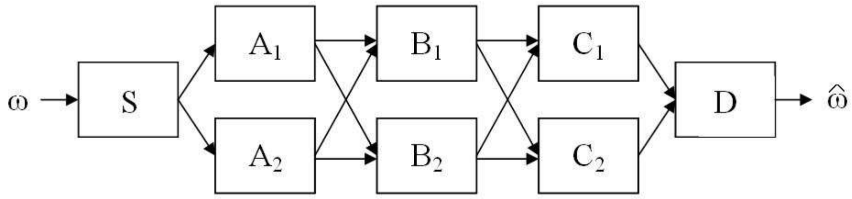

Figure 24.1. A general fading relay network.

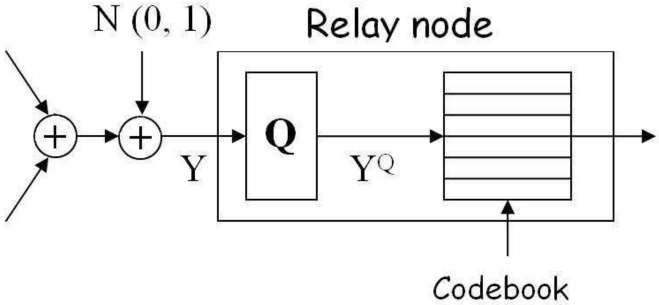

Figure 24.2. Coding scheme at a relay

As discussed in the previous lecture, the beamforming loss comes from the fact that the random codewords chosen at the relay nodes, are independent of each other whereas the cut-set bound allows the relay nodes to synchronize. We showed that the beamforming loss is proportional to the number of nodes.

The goal of this lecture is to analyze the second and the third loss. By adding all the losses, we will conclude that the scheme can get close to the cut-set bound within a constant number of bits (that depends only on the structure of the relay network).

24.2 Discretization loss

A remarkable feature of our scheme is that each relay throws away some information by quantizing the received signal, immediately after its reception. This quantization is coarse because we want to reject the noise. In order to get an intuitive feeling of the effect of the quantization loss, we consider the following toy example: suppose a gaussian random variable X gets corrupted with some gaussian noise N. This gives rise to $Y = X + N$. Y is then quantized to precision q. Let Y^Q denote the quantized version of Y. We would like to measure $I(X;Y) - I(X;Y^Q)$. The difference in the mutual information quantifies the loss we incur by quantizing a random variable. Since the Markov chain $X - Y - Y^Q$ holds, this quantity is always non-negative. In theorem 24.1 we will show that $I(X;Y) - I(X;Y^Q)$ can be bounded from above by $\frac{1}{2}\log(1+q+\frac{q^2}{3})$, where q is the quantization precision. This bound does not depends on the value of $I(X;Y)$. This observation suggests that the quantization loss in each relay is no more than a constant times T, the block length as the average loss at each relay is constant. Intuitively speaking, since the total average loss is equal to a constant time the number of nodes in the network, the discretization loss will be proportional to the number of nodes.

Theorem 24.1. *Assume that X, Y and Y^Q are defined as above. $I(X;Y) - I(X;Y^Q)$ is bounded from above by $\frac{1}{2}\log(1+q+\frac{q^2}{3})$, where q is the quantization precision.*

Proof: Let $\widetilde{Y^Q} = Y^Q + U^Q$ where U^Q is a uniform random variable over $[0,q)$. Clearly X, Y, Y^Q and $\widetilde{Y^Q}$ form the Markov chain $X - Y - Y^Q - \widetilde{Y^Q}$. Therefore $I(X;Y) \geq I(X;Y^Q) \geq I(X;\widetilde{Y^Q})$. This implies that

$$I(X;Y) - I(X;\widetilde{Y^Q}) = h(Y) - h(\widetilde{Y^Q}) + h(\widetilde{Y^Q}|X) - h(Y|X).$$

Note that $h(Y|X) = \frac{1}{2}\log 2\pi e$.

Since $I(Y;\widetilde{Y^Q}) = h(Y) - h(Y|\widetilde{Y^Q}) = h(\widetilde{Y^Q}) - h(\widetilde{Y^Q}|Y)$, we get $h(Y) - h(\widetilde{Y^Q}) = h(Y|\widetilde{Y^Q}) - h(\widetilde{Y^Q}|Y)$. Therefore

$$I(X;Y) - I(X;\widetilde{Y^Q}) = h(Y|\widetilde{Y^Q}) - h(\widetilde{Y^Q}|Y) + h(\widetilde{Y^Q}|X) - \frac{1}{2}\log 2\pi e.$$

Note that $h(\widetilde{Y^Q}|Y) = h(U^Q) = \log q$, since U^Q is a uniform random variable over $[0,q)$.[2] Also, note that $h(Y|\widetilde{Y^Q}) = h(Y|Y^Q) = h(Y - Y^Q|Y^Q) \leq \log q$ since the entropy of any random variable bounded by q is at most $\log q$. Next we note that $h(\widetilde{Y^Q}|X)$ can be bounded from above by the entropy of a gaussian random variable with the same variance. Calculations show that $Var[\widetilde{Y^Q}|X] = Var[N + \delta_q(X,N) + U^Q] \leq 1 + q + \frac{q^2}{3}$. Therefore $h(\widetilde{Y^Q}|X) \leq \frac{1}{2}\log 2\pi e(1+q+\frac{q^2}{3})$.

These computations imply that

$$I(X;Y) - I(X;\widetilde{Y^Q}) = h(Y|\widetilde{Y^Q}) - h(\widetilde{Y^Q}|Y) + h(\widetilde{Y^Q}|X) - \frac{1}{2}\log 2\pi e \leq$$

$$\leq \frac{1}{2}\log(1+q+\frac{q^2}{3}).$$

$\square$

24.3 Header accumulation loss

The decoder is interested in decoding the message transmitted by the source. As long as the decoder can decode the source message, it has no interest in learning the messages

[2] $h(U^Q) = -\int_0^q \frac{1}{q}\log\frac{1}{q}dt = \log q$.

transmitted by the intermediate relays. It turns out however that our decoding scheme allows the decoder to learn the messages transmitted by the intermediate relays as well. The output signals of a relay are computed using the quantized version of its input signals (see Figure 24.2). The input signals of a relay (at different time steps) are equal to a weighted superposition of the output signals of the preceding layer, and a gaussian noise. Although the quantization is done to reject the noise, but a gaussian distribution has an infinitely long tail. The quantized input signal will therefore inevitably convey some information about the noise realizations at the input terminal of the relay. The decoder therefore gets to learn some information about the noise realizations at the input terminals of the relays. This unintended information that "comes around for the ride" lowers the overall performance of the system. We will make this effect more precise in the next section. We call this loss the "Header accumulation" loss.

In order to get an intuitive feeling about the header accumulation loss, consider the following example: suppose gaussian random variables $X_1,\ X_2,\ ...,\ X_T$ get corrupted with independent gaussian noises $N_1,\ N_2,\ ...,\ N_T$. This gives rise to $Y_i = X_i + N_i$ $(i = 1, 2, ..., T)$. Y_i is then quantized to precision q. Let Y_i^Q $(i = 1, 2, ..., T)$ denote the quantized version of Y_i. It is a well known fact in information theory that given a typical sequence $(X_1, X_2, ..., X_T)$, the number of typical $(Y_1^Q, Y_2^Q, ..., Y_T^Q)$ sequences that are jointly typical with $(X_1, X_2, ..., X_T)$ is equal to $2^{T(H(Y^Q|X)+\epsilon)}$. In other words, the number of quantization bins that $(Y_1, Y_2, ..., Y_T)$ typically maps to is equal to $2^{T(H(Y^Q|X)+\epsilon)}$.

The value of $H(Y^Q|X) = \int_x H(Y^Q|X = x) f_X(x) dx$ can be bounded from above by $\max_{x\in\mathbb{R}}(H(Y^Q|X = x)) = \max_{x\in[-\frac{q}{2},\frac{q}{2}]}(H(Y^Q|X = x))$. We can further bound the latter expression from above by $\sum_{i\in\mathbb{Z}} \overline{p_q(i)} \log \frac{1}{\underline{p_q(i)}}$ where

$$\overline{p_q(i)} = \max_{x\in[-\frac{q}{2},\frac{q}{2}]} P(Y^Q = i|X = x);$$

$$\underline{p_q(i)} = \min_{x\in[-\frac{q}{2},\frac{q}{2}]} P(Y^Q = i|X = x).$$

Each of these terms can be explicitly computed using the complementary error function defined as $\mathrm{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2}\, dt$.

$$\overline{p_q(i)} = \frac{1}{2}[\mathrm{erfc}(|i|q - q) - \mathrm{erfc}(|i|q)] \quad \forall i \neq 0$$

$$\overline{p_q(0)} = 1 - \mathrm{erfc}(\frac{q}{2})$$

$$\underline{p_q(i)} = \frac{1}{2}[\mathrm{erfc}(|i|q) - \mathrm{erfc}(|i|q + q)] \quad \forall i \neq 0$$

$$\underline{p_q(0)} = \frac{1}{2}[1 - \mathrm{erfc}(q)]$$

Figure 24.3 shows the upper bound on $H(Y^Q|X)$ as a function of q.

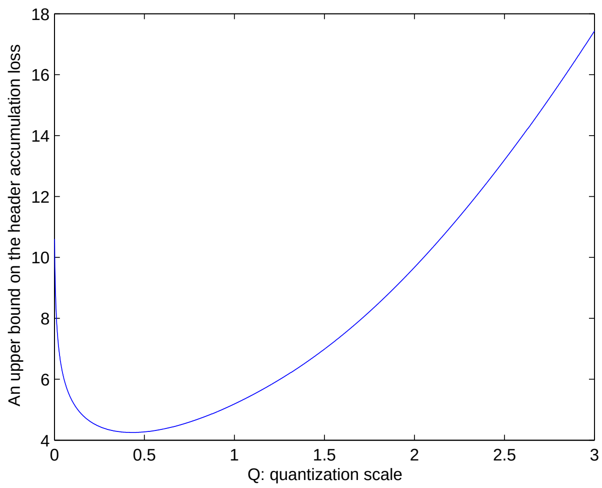

Figure 24.3. Plot of an upper bound on the header accumulation loss as a function of Q, the quantization scale.

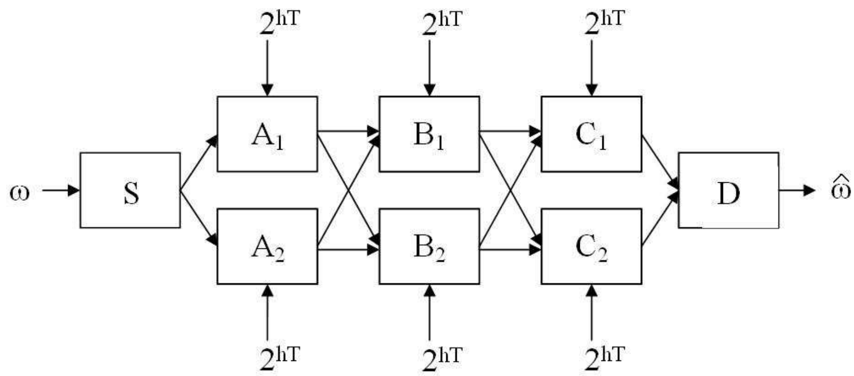

Figure 24.4. Adding an extra noise input in the deterministic model in order to capture the header accumulation loss.

24.4 Analyzing the performance of the decoder

In order to gain insights into the performance of the decoder, let us consider the decoding scheme in the deterministic model. This analysis can be carried over to the original gaussian model.

In order to capture the header accumulation loss in the deterministic model, we add a new input to each relay (see Figure 24.4). This new input is of size 2^{hT} bits, where h is equal to $H(Y^Q|X)$ discussed in the previous section.

Assign a label to each codeword of the relays as follows: label every codeword with the list of all *jointly typical* codewords of the relays of the preceding layers. In the other words, for each codeword of a given relay, we form a set each element of which consists of one codeword for each relay in the preceding layers; these codewords of the preceding layers must be "likely" to have been responsible for the given codeword. In forming the set, for instance, we may use our apriori knowledge that some combination of codewords are physically impossible since they belong to essentially different source messages.

At the decoder, we use the received message and the labels associated to the previous nodes, to list the set of all possible codewords of the previous stages that are "likely" to have been responsible for the received codeword. We declare an error in two cases: one, when the real set of codewords is not in our list of candidate codewords (a *false message* error event); two, when a false set of codewords sneaks into our list of candidate codewords of the previous layers (a *confusion* error event). Clearly if the error does not occur, we can safely decode all the codewords of previous stages, in particular the codeword of the transmitter. The probability of a *false message* error event is very small by the law of large numbers. The analysis of the probability of the *confusion* error event is similar to the one covered in previous lectures with a minor difference: the number of effective false messages to worry about is $2^{RT} \times 2^{|V|hT}$ (rather than 2^{RT}) where $|V|$ is the total number of relays plus two (one for the transmitter and one for the receiver). A cut-set type argument (like the one we had before) demonstrates that the chance that an effective false message is equal to the true message on one side of the cut Ω, and is different on the other side of the cut is at least $2^{-nI(X_\Omega;Y_\Omega|X_{\Omega^c})}$. Probability of error therefore converges to zero as long as $2^{(R+|V|h)T} \times 2^{-nI(X_\Omega;Y_\Omega|X_{\Omega^c})} \to 0$ as $n \to \infty$. This shows that we only need to give up $|V|h$ bits (which is constant) on the rate R in order to guarantee a vanishing probability of error.