

Lecture 11

In this lecture, we shall construct examples of networks where network coding gives a large gain compared to routing. We shall also look at the design of network codes based on error control coding techniques.

1 Routing vs Network coding

As an example, consider the following network

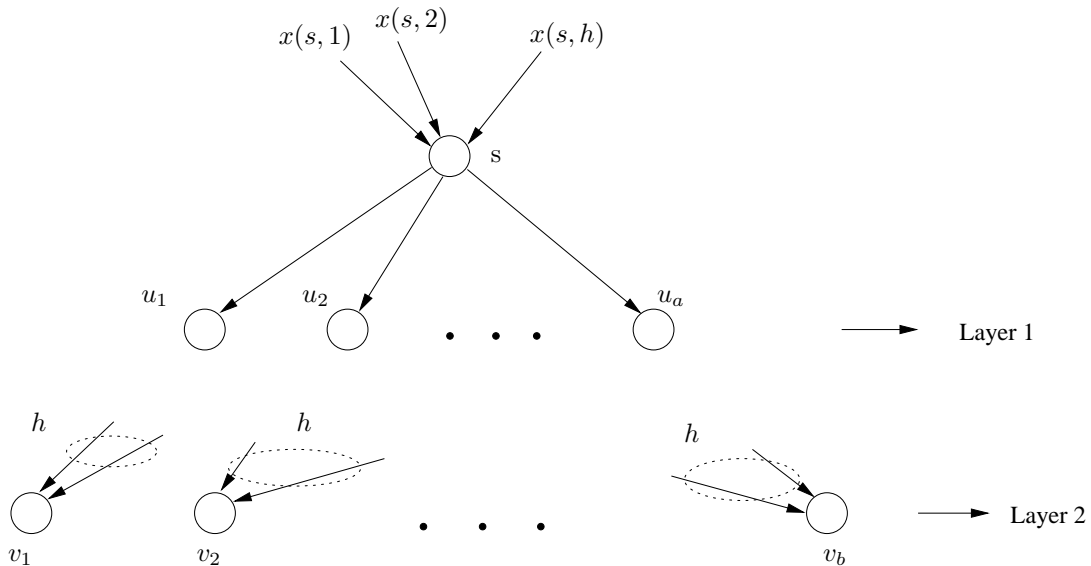


Figure 1: A network with a source node and two sets of nodes in layer-1 and layer-2

Let $G_{a,b}^h$ be the graph shown in Fig. 1, where h is the number of sources connected to the source node, a is the number of nodes in layer-1 and b is the number of nodes in layer-2. Every node, u_i (for $i = 1, \dots, a$), in layer-1 is connected to the source node. Each node, v_i (for $i = 1, \dots, b$), is connected to a h -sized subset of $\{u_1, u_2, \dots, u_a\}$. Thus, there can be a total of $\binom{a}{h}$ terminals that connect to each possible subset of $\{u_1, u_2, \dots, u_a\}$. All edges in the graph have unit-capacity.

We can see that the multicast capacity under network coding for the above graph is h , because h independent paths exist from the source node to each terminal node. Now,

we shall calculate the multicast capacity of the network when coding is not allowed. As a specific case, consider $G_{(2h, \binom{2h}{h})}^h$ with h sources, where the number of nodes in layer-1 and layer-2 are $2h$ and $\binom{2h}{h}$, respectively.

Claim 1. *Under routing, the capacity < 2 .*

Proof. Suppose that we can achieve a rate of 2 under routing. Let the sources be denoted as $x(s, 1) = a$ and $x(s, 2) = b$. Since routing is employed, we can label each edge $s \rightarrow u_i$ with either a or b . Suppose that x such edges ($x \leq h$) are labeled a .

It implies that there exists a subset of edges $\{s \rightarrow u_1, s \rightarrow u_2, \dots, s \rightarrow u_{2h}\}$ of size $2h - x$ that are labeled b . i.e. there exists a subset of $\{u_1, u_2, \dots, u_{2h}\}$, denoted \mathcal{U}^{bad} of size at least h that receives only b .

By construction, there exists a terminal that connects to a h -sized subset of \mathcal{U}^{bad} . It is clear that this subset receives only the source b which means that it is not possible to transmit two sources simultaneously to all the terminals under routing. Thus we arrive at a contradiction which means that we cannot transmit two sources simultaneously to all the terminals. This also rules out the possibility that we can transmit at a rate higher than two.

From the multicast theorem of network coding, we know that a network code exists for $G_{(2h, \binom{2h}{h})}^h$ in a field of size greater than $\binom{2h}{h}$ such that each terminal in layer-2 can recover all h sources. This shows that there are multicast instances when the gap between network coding and routing is arbitrarily large. But in practice, it is not clear whether the capacity gains are that high.

In the next part of the lecture, we shall look at the construction of a network code for $G_{a,b}^h$ using error control coding techniques.

2 Error Control Coding

Error control coding is a methodology used to correct or detect, errors and erasures in a set of received symbols. Suppose a vector, $\vec{M}_{1 \times k}$ ($M_i \in \text{GF}(q) \forall 1 \leq i \leq k$), has to be transmitted. To protect against errors, we add redundancy to the transmitting vector. This process is called *Encoding*.

$$\text{Enc} : \vec{M}_{1 \times k} \rightarrow \vec{C}_{1 \times n} \quad (n \geq k)$$

The encoding process can be linear or non-linear. Once the vector is encoded, it can be transmitted on the channel. At the receiver, a decoding process is implemented on the

received vector. If only a small number of symbols from $\vec{C}_{1 \times n}$ are either dropped or in error, the encoding and decoding processes can help us get back the k symbols error-free. The encoding operation at the transmitter and the decoding operation at the receiver constitute the process of *error control coding* for the system.

Example: Let $M_i \in \text{GF}(2)$. The encoding function is

$$\text{Enc} : \vec{M}_{1 \times 1} \rightarrow [\text{repeat } n \text{ times}]$$

So, we have $0 \rightarrow [0 \ 0 \dots 0]_{1 \times n}$ and $1 \rightarrow [1 \ 1 \dots 1]_{1 \times n}$. Assume that $[0 \ 0 \dots 0]_{1 \times n}$ is transmitted and an error is introduced in the channel. At the receiver, we find $(n - 1)$ zeros and a single one. So, the receiver can say that the codeword would have been most probably $[0 \ 0 \dots 0]_{1 \times n}$ because the number of zeros is much larger than the number of ones. This particular code is called a *Repetition code* of length n . Observe that using this code, to transmit one bit of information we actually transmit n bits so that the rate of the code is $1/n$. If n is odd, then it is easy to see that we can correct $(n - 1)/2$ bit errors using this code if the decoder, decodes using majority logic.

2.1 Reed-Solomon Codes

Reed-Solomon (RS) codes belong to the class of non-binary error correcting codes that operate on $\text{GF}(q)$ i.e. each symbol that is transmitted belongs to $\text{GF}(q)$.

A (n, k) RS code (it encodes k message symbols into n coded symbols) on $\text{GF}(q)$ is encoded as follows. Pick n distinct non-zero elements from $\text{GF}(q)$. Let them be $\{\alpha_1, \dots, \alpha_n\}$. Note that this necessitates $q > n$. Follow the following steps to encode the message vector.

- Message vector is $\vec{M} = [M_0, M_1, \dots, M_{k-1}]$.
- Form the polynomial $M(x) = \sum_{i=0}^{k-1} M_i x^i$.
- Evaluate $M(x)$ at $x = \alpha_i$ for $i = 1, \dots, n$.
- Define the encoded vector as $\vec{C} = [M(\alpha_1), \dots, M(\alpha_n)]$.

Now, we shall calculate the minimum distance between any two codewords. i.e. the minimum number of positions where two distinct codewords have to differ. Let $M_1(x) = \sum_{i=0}^{k-1} M_{1i} x^i$

and $M_2(x) = \sum_{i=0}^{k-1} M_{2i}x^i$ be the two polynomials corresponding to the two codewords \vec{c}_1 and \vec{c}_2 , respectively. This means that $(M_1(x) - M_2(x))$ is a polynomial with degree at most $(k - 1)$. So, the polynomial $(M_1(x) - M_2(x))$ can have at most $(k - 1)$ roots. This implies that the codewords \vec{c}_1 and \vec{c}_2 are identical in at most $(k - 1)$ locations and are different in at least $(n - k + 1)$ locations. This value $(n - k + 1)$ is called the minimum distance of the RS code, also denoted as d_{\min} . Intuitively, if the minimum distance of a code is large, we expect to be able to correct more errors using it.

The encoding process for RS codes can also be expressed as a matrix multiplication as shown below.

$$\begin{aligned} \vec{C} &= [M_0 \ M_1 \ \cdots \ M_{k-1}] \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_n \\ \vdots & \vdots & & \vdots \\ \alpha_1^{k-1} & \alpha_2^{k-1} & \cdots & \alpha_n^{k-1} \end{pmatrix} \\ &= \vec{M} G \end{aligned}$$

G is called the generator matrix of the RS code. From the structure of G , we can say that it is a $k \times n$ Vandermonde matrix. So, any $k \times k$ sub-matrix of G has the Vandermonde structure which makes it non-singular¹. This implies that the message \vec{M} can be recovered from any k symbols of the corresponding codeword \vec{C} . We shall use this property of the RS codes to design a network code for the graph $G_{a,b}^h$.

3 Network code using RS codes

Consider the graph $G_{a,b}^h$ as described in the initial part of the lecture. Then pick an RS(n, k) code with the parameters $n = a$ and $k = h$. Consider the following coding procedure on the graph $G_{a,b}^h$. Note that a has to be greater than h .

- The source node s receives h symbols from the sources each time instant.
- Source node encodes the h symbols using RS(a, h) code.

¹For a square Vandermonde matrix, $V = [V_{ij}] = [a_i^{j-1}]$ ($1 \leq i, j \leq k$), the determinant is given by $\det(V) = \prod_{(1 \leq i < j \leq k)} (a_j - a_i)$. So, the determinant is non-zero as long as a_i are distinct for $1 \leq i \leq k$

- Each of the a symbols of the codeword is passed on to one node in layer-1.

Since each terminal receives a h -sized subset of the codeword, by the property of the (a, h) RS code, the original h symbols created at the source can be reconstructed at every terminal. So, the multicast capacity for the network is achieved using the network code designed based on RS code.

Note that the (a, h) RS code can operate on $\text{GF}(a + 1)$. So, the designed network code can use a field size much smaller than $|T|$ (the number of terminals can be as large as $\binom{2h}{h}$). It is interesting to note in this case that we construct network codes from error correcting codes even though there are no errors on the links.