EECS 290S: Network Information Flow	**Fall 2008**

Lecture 15 — October 16

Lecturer: David Tse *Scribe: Igor Ganichev*

Before we consentrated on the case of single source multiple destinations. We first looked at the wireline model then at linear deterministic model. Then, we generalized to nonlinear deterministic model and finally showed some results for the Gausian network. Today we will start looking into several extensions beyond the single source multiple destinations case.

15.1 Natural Extensions of the Single Source Multiple Destinations Case

1. **Many-to-One:** Multiple senders sending independent information to the same destination t. The destination t is interested in all of the informations. Common practical scenarios with many-to-one communication pattern include cases when a central node is gathering information from multiple nodes in the network. Note that the solution in this case is categorized not by a single rate but by a set of achievable rate tuples. For each tuple $(r_1, r_2, \dots r_q)$ in the set, r_i is the rate of sender s_i. This extension is the relatively easy one.

2. **One-to-Many:** Single source is sending independent as well as common information to multiple destinations. The destinction from the case that we studied is that not all destinations are interested in all of the messages. For example, let s be the source node and t_1, t_2 be the destination nodes. Assume that s has three messages m_1, m_2, m_3 to send and that t_1 is interested in messages m_1 and m_2, while t_2 is interested in messages m_1 and m_3. In this case, m_1 is the common information in which all destinations are interested and m_2, m_3 is the independent information. In this case, the solution is also a collection of achievable tuples of rates, but rates are the rates of independent information that the sender can send to each of the destinations and one rate for the common information. This case is a somewhat strange case, because there is a solution for the case of two destinations but no general solution is known.

3. **Many-to-Many:** Multiple sources are sending independent information to multiple destinations. Each source is sending to a single destination. This problem is also known as multiple unicast problem. The solution would also take the form of achievable tuples, but no proof is known even for the case of two sources and two destinations.

15.2 Many-to-One

We first consider the wire-line model. For the ease of explanation we will consider a special case with 2 senders s_1 and s_2. The proof should be easily extensible to the general case.

Let r_i be the rate of sender s_i. We can immediately write a set of constraints on the rates using the minimum cut upper bound. For any cut separating a set of senders from the destination, the cumulative rate of all these senders cannot exceed the capacity of the cut. For the case of two senders, we have

$$\begin{aligned} r_1 &\leq mincut(s_1, t) \\ r_2 &\leq mincut(s_2, t) \\ r_1 + r_2 &\leq mincut\left(\{s_1, s_2\}, t\right) \end{aligned}$$

In general, there will be $2^q - 1$ constraints of the form

$$\sum_{i \in A} r_i \leq mincut\left(\{s_i : i \in A\}, t\right)$$

for all $A \subset \{1, 2, \ldots, q\}$, where q is the number of senders.

Each constraint restricts the set of possible solutions to lie below the hyperplane defined by the constraint in N^q. Thus, a q-tuple of rates that satisfies all of these constraints will lie in a convex region that is the intersection of all half-spaces defined by the hyperplanes.

We next show that all tuples that satisfy all of the constraits are in fact achievable. For brievity, let us define a shorthand notation for the value of the mincut

$$c(u; v) \stackrel{\text{def}}{=} \underset{\substack{S\ cut \\ u \in S, v \in S^c}}{mincut}(S, S^c)$$

Given a rate pair (R_1, R_2) satisfying the three constraints, we create a super source s and connect it to all sources s_i with links of capacity R_i. We call the resulting network *augmented* and the given one the *original* network. Next, we show that $c(s; t) = R_1 + R_2$.

A cut separating S and t can be one of four types depending on which side of the cut lie the sources s_1 and s_2. The mincut is the minimum over all these four types of cuts. Thus,

$$\begin{aligned} c(s; t) = \min(\quad & c(s, s_1, s_2; t),\ c(s; s_1, s_2, t), \\ & c(s, s_1; s_2, t),\ c(s, s_2; s_1, t) \quad) \end{aligned}$$

We look at each of the four types of cuts separately.

$$c(s, s_1, s_2; t) = c(s_1, s_2; t) \geq R_1 + R_2$$

because a cut separating s, s_1 and s_2 from t in the augmented network separates s_1 and s_2 from t in the original network and vice versa.

$$c(s; s_1, s_2, t) = R_1 + R_2$$

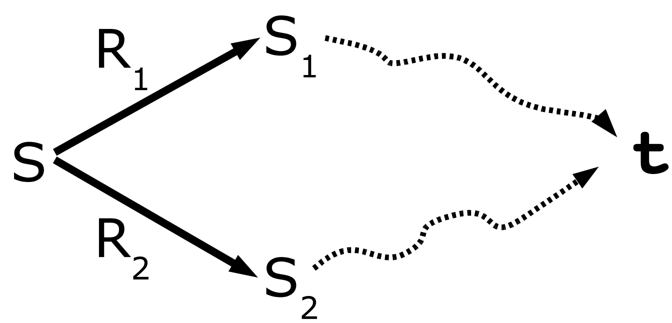

Figure 15.1. Many-to-One network with an added super source.

because the cut separating s from s_1, s_2 and t has to cut at least edges (s, s_1) and (s, s_2) and cutting this edges gives one such cut.

$$\begin{aligned} c(s, s_1; s_2, t) &= R_2 + c(s_1; s_2, t) && (15.1) \\ &\geq R_2 + c(s_1; t) && (15.2) \\ &\geq R_2 + R_1 && (15.3) \end{aligned}$$

Equation 15.1 is true because (1) any cut that separates s and s_1 from s_2 and t has to include edge (s, s_1) and (2) any cut separating s and s_1 from s_2 and t has to separate s_1 from s_2 and t in the original network and (3) vice versa any cut separating s_1 from s_2 and t in the original network plus the edge (s, s_2) is a cut separating s and s_1 from s_2 and t in the augmented network. Equation 15.2 is true because dropping a constraint on the structure of the cut increases the number of possibilities over which the minimum is taken, i.e. the value can only become smaller. Finally, equation 15.3 is true by assumption that the rates satisfy the constraints.

$$c(s, s_2; s_1, t) \geq R_2 + R_1$$

by a symmetrical argument.

Thus, $c(s; t) = R_1 + R_2$, which implies that there is actually a routing solution achieving a rate of $R_1 + R_2$. Because the capacities of edges (s, s_1) and (s, s_2) are R_1 and R_2, the routing solution on the augmented network has to send R_1 symbols through s_1 and R_2 symbols through s_2. Thus, if we just ignore the super source s, we immediately get a routing solution for the original problem. One can verify that the argument extends to arbitrary number of sources.

A natural extension to this problem would be to consider multiple destinations, each of which is interested in all of the information. In this case, the constraints would have similar structure but mincut will be replaces by the min mincut over all destinations.

To show achievability of the region defined by the constraints, one first adds a super source s. Then, given a pair (or a tuple in the general case of many sources) of rates (R_1, R_2) satisfying the constraints, one shows that $c(s; t_i) \geq R_1 + R_2$ for all destinations t_i

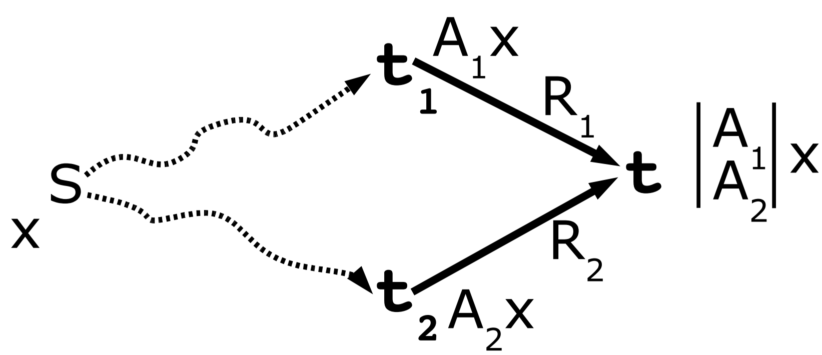

Figure 15.2. One-to-Many network with an added super destination.

using analogous argument to the one we did above. This inequalities imply that the min mincut between s and the destinations in the augmented network is $R_1 + R_2$. Thus, these is a linear network coding that achieves this rate. The last step is to convert the linear solution to the augmented network to a linear solution to the original problem, which can be done as follows.

To construct a solution to the original problem, we keep all the local transformations of the solution to the augmented problem. Let x_i be the vector of R_i symbols that s_i is receiving from s. Then, there are matrices A_i such that $x_i = A_i x$. Since all of the destination are able to decode x, we can ask each destination to multiply x by A_i to get the data that s_i is sending. We only need to note that because A_i's must have full rank, s_i has all R_i dimensions to send its data.

All the results in this section also hold for the linear deterministic model but the mincuts in the constraints are replaced with corresponding ranks of transformation matrices.

15.3 One-to-Many

The One-to-Many problem is very similar to the Many-to-One problem we just considered. Both the results and the solution method are analogous. Recall that in this problem the source sends independent information to each destination (we also call this information private). As before, we consider the case with two destinations and leave the generalization to an arbitrary number of destinations as an exercise. We proceed by writing the constraints using the mincut bounds.

$$\begin{aligned} R_1 &\leq c(s; t_1) \\ R_2 &\leq c(s; t_2) \\ R_1 + R_2 &\leq c\,(s; t_1, t_2) \end{aligned}$$

Note that the last bound relies on the assumption that information is idependent. Given a pair of rates (R_1, R_2) satisfying all the constraints, we show that these rates are indeed achievable. Create a super destination t and connection it to t_i with an edge of capacity R_i. As before, we show that $c(s;t) = R_1 + R_2$.

$$\begin{aligned} c(s;t) = \min(\quad & c(s,t_1,t_2;t),\ c(s;t_1,t_2,t), \\ & c(s,t_1;t_2,t),\ c(s,t_2;t_1,t) \quad) \end{aligned}$$

With an analogous to the above argument, one can show that each of the four types of cuts has capacity at least $R_1 + R_2$ and that $c(s,t_1,t_2;t)$ is exactly $R_1 + R_2$.

By the max flow min cut theorem, the augmented network has a routing solution with flow of $R_1 + R_2$. One can also convert a linear coding solution to the augmented problem into a solution to the original problem in the following way.

As depicted in the figure, let A_i be the global transformation of the data from s to the edge (t_i,t). A_i has be of dimention $R_i \times (R_1 + R_2)$ because the source is sending $R_1 + R_2$ symbols and the capacity of the edge (t_i,t) is R_i. Then, the matrix

$$A = \begin{vmatrix} A_1 \\ A_2 \end{vmatrix}$$

has to be invertible because the destination is able to recover x. Then, let the inverse of A be $A^{-1} = |C_1 C_2|$. Now, we can construct the solution to the original problem. If the source wants to send x_1 to t_1 and x_2 to t_2, it first multiplies $x = \begin{vmatrix} x_1 \\ x_2 \end{vmatrix}$ by A^{-1} and treats the result, $u = A^{-1}x$, as it treated its data in the solution to the augmented network. The destination t_i would then get the data x_i as the data that it would have sent to the super source (which we discard). Indeed, t_1 and t_2 would have sent

$$A_1 u = A_1 A^{-1} x = A_1 |C_1\ C_2| \begin{vmatrix} x_1 \\ x_2 \end{vmatrix} = |A_1 C_1\ A_1 C_2| \begin{vmatrix} x_1 \\ x_2 \end{vmatrix} = |I\ 0| \begin{vmatrix} x_1 \\ x_2 \end{vmatrix} = x_1 \qquad (15.4)$$

$$A_2 u = A_2 A^{-1} x = A_2 |C_1\ C_2| \begin{vmatrix} x_1 \\ x_2 \end{vmatrix} = |A_2 C_1\ A_2 C_2| \begin{vmatrix} x_1 \\ x_2 \end{vmatrix} = |0\ I| \begin{vmatrix} x_1 \\ x_2 \end{vmatrix} = x_2 \qquad (15.5)$$

As before, this result extends to arbitrary number of destinations and to linear deterministic models.

15.3.1 One-to-Many with Common and Private Information

A natural extension to the one-to-many case we looked at above is the case when the source sends not only private information to each destination but also some common information to all of the destinations. This extension is similar to the original problem but the similarity breaks at the end and to this day there are no published results for the case of more than two destinations. In this section, we look into this case.

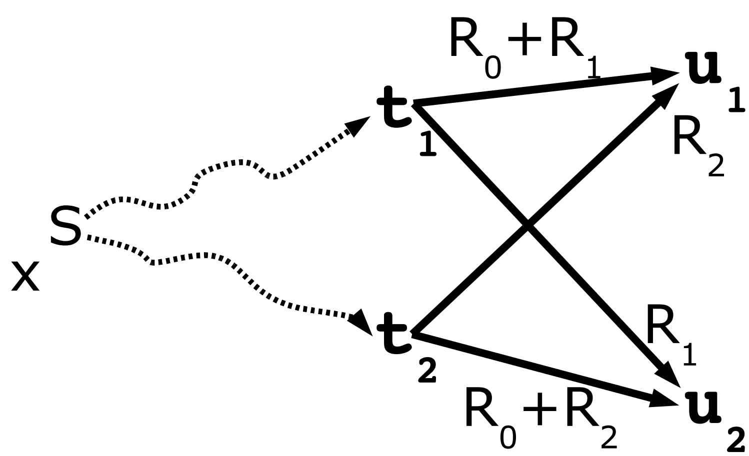

Figure 15.3. One-to-Many network with common information and added super destinations.

Because we now have common information, we need to introduce a rate R_0 for common information. Then, using then mincut bounds we get the following

$$\begin{aligned} R_0 + R_1 &\leq c(s;t_1) \\ R_0 + R_2 &\leq c(s;t_2) \\ R_0 + R_1 + R_2 &\leq c\,(s;t_1,t_2) \end{aligned}$$

It is obvious that to show the achievability of the whole region we only need to show the achievability of the extreme points when at least one of the three constraints is tight. Any other point inside the region can be obtained by time sharing because the region is convex. The part of the region where the third constraint is tight is a plane in three dimentional space bounded by two half spaces (the first two constraints) and the conditions that the rates are positive. Thus this region is the intersection of a plane and several convex subspaces. Therefore, it is a convex polygon and its corners are either points where one of the first constraints is also tight or when one of the rates is zero. First, consider the case that one of the rates is zero.

If R_0 is zero, we are back to the case with no common information which we just solved. With loss of generality, assume that $R_2 = 0$. But then,

$$R_0 + R_1 = R_0 + R_1 + R_2 = c(s;t_1,t_2) \geq c(s;t_1) \geq R_0 + R_1$$

which implies that the first constraint is tight. Thus, we have shown that at all of the corners one of the first two constraints is tight. Without loss of generality, assume that the first constraint is tight.

Following the earlier proofs, we would like to modify the network so that we can use our multicast case result. The most natural thing to try is to add some destinations and have them receive all the information. However, this has to be done in such a way that we will later be able to construct a solution for our case from the multicast case. To achieve that, our modification to the network has to be such that any solution of the multicast case has the property that the necessary information (private and common) is delivered to our destinations t_1 and t_2.

One such a modification is illustrated in the figure 15.3. We added two super destinations and edges between original destinations and super destinations with capacities as depicted in the figure. Since we want to achieve a total rate of $R_0 + R_1 + R_2$ and the multicast result guarantees achievability of rates less than or equal to min mincut over all destinations, we need to show that

$$\min(c(s; u_1), c(s, u_2)) \geq R_0 + R_1 + R_2.$$

We decompose $c(s; u_1)$ based on which side of the cut t_1 and t_2 lie.

$$c(s; u_1) = \min(\quad c(s, t_1, t_2; u_1),\ c(s; t_1, t_2, u_1), \\ c(s, t_1; t_2, u_1),\ c(s, t_2; t_1, u_1) \quad)$$

Next, we bound each of the four cut types.

- $c(s, t_1, t_2; u_1) = R_0 + R_1 + R_2$ because to separate u_1 from t_1 and t_2 one needs to cut edges (t_1, u_1) and (t_2, u_1) whose total capacity is $R_0 + R_1 + R_2$. Further, cutting just these edges gives us the desired cut.

- $c(s; t_1, t_2, u_1) \geq c(s; t_1, t_2) \geq R_0 + R_1 + R_2$. First inequality is because dropping a constraint on the cut can only decrease its value and the second inequality is by assumption on the rates.

- $c(s, t_1; t_2, u_1) \geq R_0 + R_1 + c(s; t_2) \geq R_0 + R_1 + R_0 + R_2 \geq R_0 + R_1 + R_2$. The first inequality is because to separate u_1 and t_2 from t_1 and s one needs to cut edge (t_1, u_1) and separate s from t_2 in the augmented network. The cut that separates nodes in the augmented network has capacity at least as big as the cut separating the same nodes in the original network. Thus, separating s from t_2 in the augmented network is at least as hard as in the original network. Therefore, we ca view $c(s; t_2)$ as a mincut in the original network, which has capacity at least $R_0 + R_2$ (the second inequality).

- $c(s, t_2; t_1, u_1) \geq R_2 + c(s; t_1) \geq R_2 + R_1 + R_0$ by an argument analogous to the one above.

Thus, we have shown that $c(s; u_1) = R_0 + R_1 + R_2$. By symmetry, we can conclude that $c(s; u_2) = R_0 + R_1 + R_2$ and hence that $\min(c(s; u_1), c(s, u_2)) = R_0 + R_1 + R_2$. So, there is a linear network code that achieves $R_0 + R_1 + R_2$ sending from s to u_1 and u_2. We just need to covert a solution to the augmented network into a solution to the original network. We do that next.

Let U_1 , U_2, V_1, V_2 be the row spaces of the global transformation from s to edges (t_1, u_1), (t_2, u_1), (t_1, u_2), (t_2, u_2), respectively. Figure 15.4 depicts these spaces. Further, let A_1 and A_2 be the global transformation from s to t_1 and t_2, respectively.

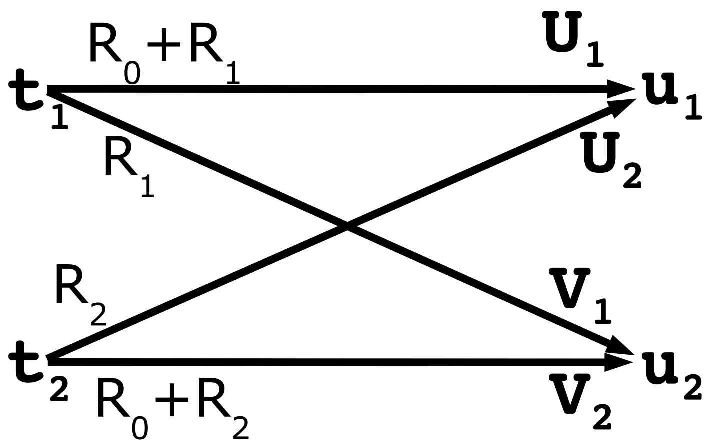

Figure 15.4. Subspaces U_1, U_2, V_1 and V_2 of the augmented one-to-many network.

Consider $S_1 = span(U_1, U_2)$, the subspace of $\Omega = F^{R_0+R_1+R_2}$ spanned by all the rows both in U_1 and U_2. If S_1 is not equal to Ω, then consider a vector $v \in \Omega$, $v \neq 0$ perpendicular to S_1 (i.e. perpendicular to all vectors in S_1). Then, if the source sends v, u_1 will get all zeros and hence won't be able to distinguish v from 0. Therefore, $S_1 = \Omega$ and $S_2 = \Omega$ by an analogous argument, where $S_2 = span(V_1, V_2)$.

Lemma 15.1. *Let A be an $m \times n$ matrix and B a $k \times m$ matrix. Then, the row space of BA is a subspace of the row space of A.*

Proof: Recall from linear algebra that the row space is the orthogonal compliment of the null space of the matrix. It is obvious that the null space of A is a subspace of the null space of AB. Therefore, the the row space of BA is a subspace of the row space of A. □

Furthermore, by our assumption that the mincut separating s and t_1 is equal to $R_0 + R_1$, the dimention of the row space of A_1 is at most $R_0 + R_1$. However, because U_1 has dimension $R_0 + R_1$ and is a row space of a product of A_1 and a local transformation matrix, we can apply the lemma 15.1 to see that the dimention of the row space of A_1 is equal to $R_0 + R_1$. In addition, the row space of A_1 is equal to U_1. Then, by lemma 15.1, $V_1 \subset U_1$.

Thus, we know that U_1 and U_2 span the whole Ω and that $V_1 \subset U_1$. Let, W_0 by a subspace of U_1 such that $W_0 \cap V_1 = 0$ and $W_0 \cup V_1 = U_1$. Now, we can identify the private and common information. The common information will be encoded in W_0, private information for t_1 will be encoded in V_1, and private information for t_2 will be encoded in U_2

Let us, choose a basis P_0 for the subspace W_0, a basis P_1 for the subspace V_1 and a basis P_2 for the subspace U_2. Further, let Q_i be the matrix of dimension $R_i \times (R_0 + R_1 + R_2)$ whose rows are basis vectors in P_i. We are now ready to specify the encoding and decoding scheme.

If the source wants to send first R_0 entries of x to both destination, next R_1 entries to the first destination, and the last R_2 entries to the second destination, it first multiplies x by the matrix Q^{-1} to get x',

$$x' = Q^{-1}x$$

where

$$Q = \begin{vmatrix} Q_0 \\ Q_1 \\ Q_2 \end{vmatrix}$$

Then, the nodes t_1 and t_2 decode the data by multiplying the data it receives by matrices B_1 and B_2, respectively, which are defined as follows. B_1 is the $(R_0 + R_1) \times (R_0 + R_1)$-dimensional matrix such that

$$\begin{vmatrix} Q_0 \\ Q_1 \end{vmatrix} = B_1 A_1$$

B_2 is the $(R_0 + R_2) \times (R_0 + R_2)$-dimensional matrix such that

$$\begin{vmatrix} Q_0 \\ Q_2 \end{vmatrix} = B_2 A_2$$

Matrices B_1 and B_2 exist because because the span of rows of A_1 is equal to the span of rows of $\begin{vmatrix} Q_0 \\ Q_1 \end{vmatrix}$ and the span of rows of A_2 includes to the span of rows of $\begin{vmatrix} Q_0 \\ Q_2 \end{vmatrix}$. The only thing that we might need to do is to discard linearly dependent rows from A_1 to make it have $R_0 + R_1$ rows, and to discard linearly dependent rows from A_2 as well as rows not in the span of $\begin{vmatrix} Q_0 \\ Q_2 \end{vmatrix}$ to make A_2 have $R_0 + R_2$ rows.

With this definitions it is straightforward to see that each destination gets the right information. Destination t_1 gets

$$B_1 A_1 x' = B_1 A_1 Q^{-1} x = \begin{vmatrix} Q_0 \\ Q_1 \end{vmatrix} Q^{-1} x = \begin{vmatrix} I & 0 & 0 \\ 0 & I & 0 \end{vmatrix} x$$

Destination t_2 gets

$$B_2 A_2 x' = B_2 A_2 Q^{-1} x = \begin{vmatrix} Q_0 \\ Q_2 \end{vmatrix} Q^{-1} x = \begin{vmatrix} I & 0 & 0 \\ 0 & 0 & I \end{vmatrix} x$$