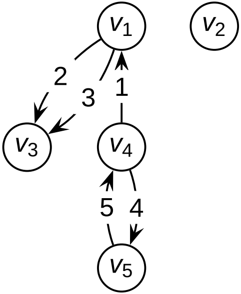
Aspects of network coding - Part I

O. Geil, Aalborg University

S^3cm: Soria Summer School on Computational Mathematics,
Universidad de Valladolid, Soria
July 12-16 2010

Terminology



Definition: Given a finite set V and a map $\epsilon : \{1, \dots, n\} \to V \times V$. Let $E = \{(1, \epsilon(1)), \dots, (n, \epsilon(n))\}$. Then $G = (V, E)$ is called a directed graph.

Elements in V are vertices and elements in E are edges.

When the edge map ϵ is known we write i instead of $(i, (u, v))$.
That is $E = \{1, \ldots, n\}$.
When it does not lead to confusion we may also write (u, v) instead of $(i, (u, v))$

Definition: Given $v \in V$ define

- $\mathrm{in}(v) = \{i \in E \mid \epsilon(i) = (w, v) \text{ for some } w\}$
- $\mathrm{out}(v) = \{i \in E \mid \epsilon(i) = (v, w) \text{ for some } w\}$

Given $j \in E$ write $\epsilon(j) = (u, v)$ and define

- $\mathrm{in}(j) = \mathrm{in}(u)$
- $\mathrm{tail}(j) = u$
- $\mathrm{out}(j) = \mathrm{out}(v)$
- $\mathrm{head}(j) = v$

Path

Definition: A path in $G = (V, E)$ is a sequence of edges $\mathcal{P} = (i_1, \ldots, i_k)$ such that $\text{head}(i_s) = \text{tail}(i_{s+1})$ for $s = 1, \ldots, k-1$.

When the graph has not multiple edges we can write this as $\mathcal{P} = \big((u_0, u_1), (u_1, u_2), \ldots, (u_{n-1}, u_n)\big)$.

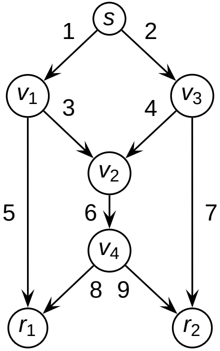
In this mini course we shall always assume that our (directed) graph are CYCLE FREE.

This by definition means that there does not exist a path $\mathcal{P}$ in which a vertex u appears more than once.

In particular we do not allow loops.

First communication problem

Sender s wants to send two messages $a, b \in \mathbf{F}_2$ to both receivers r_1 and r_2 simultaneously.



Concentrating on r_1

Flow of size 2 to r_1: $F_1 = \{(1,5),(2,4,6,8)\}$

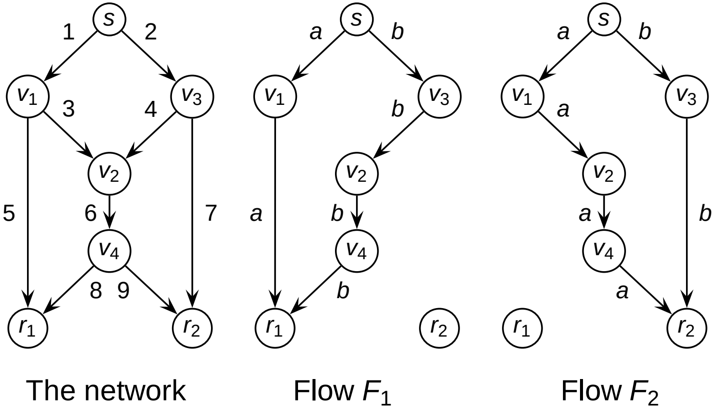
Send a along edge 1 and b along edge 2 and let them propagate.

Concentrating on r_2

Flow of size 2 to r_2: $F_2 = \{(1,3,6,9),(2,7)\}$

Send a along edge 1 and b along edge 2 and let them propagate.

Two partial solutions

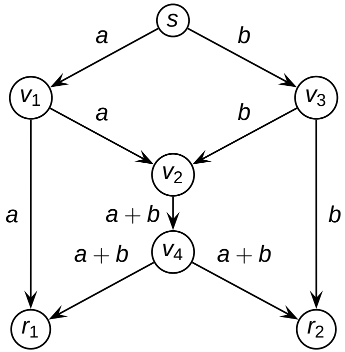


The flow system is $\mathcal{F} = \{F_1, F_2\}$

$F_1 = \{(1,5), (2,4,6,8)\}, F_2 = \{(1,3,6,9), (2,7)\}$

A solution

Routing is insufficient, but problem is solvable



Receiver r_1 can reconstruct b as $a + (a + b)$
Receiver r_2 can reconstruct a as $(a + b) + b$

Ancestral orderings

The assumption that G is cycle free implies that we can order E by an ancestral ordering.

An ancestral ordering on E is a total ordering such that $i < j$ implies there is not path with i visited before j.

Similarly, ancestral orderings on V.

The general problem

$G = (V, E)$
$S = \{s_1, \ldots, s_{|S|}\} \subseteq V$ called senders
$R = \{r_1, \ldots, r_{|R|}\} \subseteq V$ called receivers

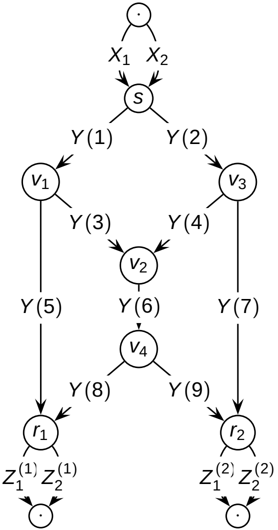
Message vector $\vec{X} = (X_1, \ldots, X_h)$. The messages X_i takes on values in $\mathcal{A}$ (an abelian group)

$K : \{X_1, \ldots, X_h\} \rightarrow S$ a surjective map

If $K(X_i) = s_j$ then we say that message X_i is generated at s_j.

$D(r_l) = (X_{i_1}, \ldots, X_{i_{|D(r_l)|}})$ which is called demand.

General set-up



<u>Encoding functions;</u>

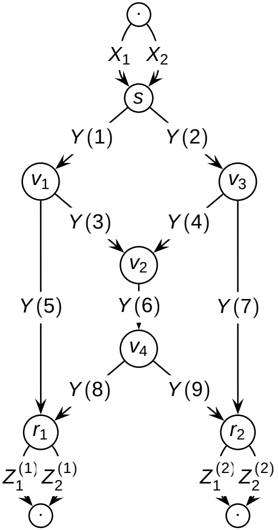
For every edge j we define a variable $Y(j)$ that takes on values in $\mathcal{A}$.

Visiting the edges by an ancestral ordering we define relations

$$Y(j) = f_j\Big(\big(Y(i) \mid i \in \text{in}(j)\big), \big(X_k \mid X_k \text{ is generated at tail}(j)\big)\Big)$$

If argument empty $Y(j)$ always takes on the value 0.

General set-up cont.



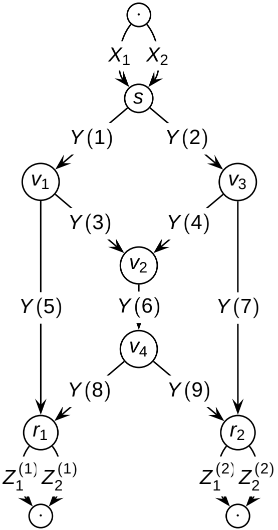
Decoding functions:

For every receiver r_l we define variables $Z_1^{(r_l)}, \dots, Z_{|D(r_l)|}^{(r_l)}$.

$$Z_j^{(r_l)} = d_j^{(r_l)}\Big(\big(Y(i) \mid i \in \mathrm{in}(r_l)\big), \big(X_k \mid X_k \text{ is generated at } r_l\big) \Big)$$

Network coding problem: Choose if possible f_j, d_j such that $(Z_1^{(r_l)}, \dots, Z_{|D(r_l)|}^{(r_l)}) = D(r_l)$.

Linear network coding



Alphabet now is $\mathbf{F}_q$ and coefficients below belong to $\mathbf{F}_q$.

$$Y(j) = \sum_{i \in \text{in}(j)} f_{i,j} Y(i) + \sum_{\kappa(X_i)=\text{tail}(j)} a_{i,j} X_i$$

$$Z_j^{(r_l)} = \sum_{i \in \text{in}(r_l)} b_{i,j}^{(r_l)} Y(i) + \sum_{\kappa(X_i)=r_l} \tilde{b}_{i,j}^{(r_l)} X_i$$

We often assume $R \cap S = \emptyset$

Scenarios

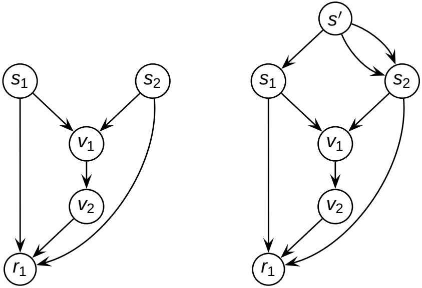
- Unicast. $R = \{r_1\}$. $D(r_1) = (X_1, \ldots, X_h)$. (Classical theory)
- Multicast. More receivers. Every receiver demands everything. (Recent theory)
- General situation. More receivers, different demands. (Rather open)

We concentrate mainly on multicast.

Definition: Given a multicast problem a flow (of size h) to receiver r is a set of h edge disjoint paths from $S = \{s_1, \ldots, s_{|S|}\}$ to r such that the number of paths starting in s_i equals the number of messages generated in s_i

Note, a flow is NOT a sub graph.

Unicast:
Existence of a flow is necessary and a sufficient condition for solvability.



X_1 is generated at s_1 and X_2, X_3 are generated at s_2.

Add s' and edges $e_1, \ldots, e_h$ from s' to $S = \{s_1, \ldots, s_{|S|}\}$. Number of edges from s' to s_i is number of messages generated at s_i.

Consider all possible V_1, V_2, $V_1 \cap V_2 = \emptyset$, $V_1 \cup V_2 = V \cup \{s'\}$, $s' \in V_1$, $r \in V_2$.

cut(V_1, V_2) is the edges FROM V_1 TO V_2.

min cut $=$ max flow

Flow necessary and sufficient (use routing) condition

Multicast

corresponds to set of unicast problems.

Flow system (of size h)

$\mathcal{F} = (F_1, \ldots, F_{|R|})$

F_l is a flow (of size h) from S to r_l.

Existence of a flow system i necessary.

Surprisingly also sufficient (Ahlswede, Cai, Li and Yeung, 2000).

Actually when solvable; linear network coding is enough!

Matrices

A is $h \times |E|$
$A_{i,j} = a_{i,j}$ if $K(X_i) = \text{tail}(j)$
$A_{i,j} = 0$ else

F is $|E| \times |E|$
$F_{i,j} = f_{i,j}$ if $i \in \text{in}(j)$
$F_{i,j} = 0$ else

For $l = 1, \ldots, |R|$

$B^{(r_l)}$ is $|E| \times h$
$B^{(r_l)}_{i,j} = b^{(r_l)}_{i,j}$ if $i \in \text{in}(r_l)$
$B^{(r_l)}_{i,j} = 0$ else

The $F_{i,j}$ "holds" information on all paths of length 2 starting in edge i and ending in edge j.

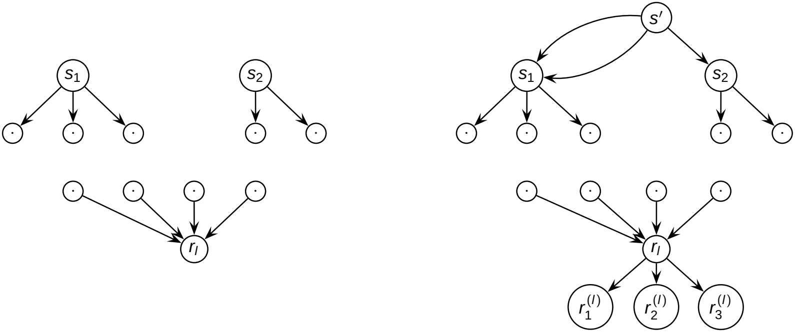
The (i,j)th entry of F^n "holds" information on all paths of length $n+1$ starting in edge i and ending in edge j.

$$\left(F^n\right)_{i,j} = \sum_{\substack{(i=j_0, j_1, \ldots, j_n = j \\ \text{a path} \\ \text{in } G}} f_{i=j_0,j_1} f_{j_1,j_2} \cdots f_{j_{n-1},j_n=j}$$

This in particular holds for F^0.

G being cycle free $F^N = 0$ for some big enough N.

$I + F + \cdots + F^{N-1}$
holds information on all paths of any length.



Modification of network. In original network two sources at s_1 and one source at s_2.

In modified network the $a_{i,j}$'s and the $b_{i,j}^{(r_l)}$'s from the original network plays the same role as the $f_{i,j}$'s

Lemma:
$M^{(r_l)} = A(I + F + \cdots + F^{N-1})B^{(r_l)}$
holds information on all paths from s' to $\{r_1^{(l)}, \ldots, r_h^{(l)}\}$

From this we derive: **Theorem:**
$(X_1, \ldots, X_h)M^{(r_l)} = (Z_1^{(r_l)}, \ldots Z_h^{(r_l)})$

$M^{(r_l)}$ is called the transfer matrix for r_l

Note $(I + F + \cdots + F^{N-1})(I - F) = I$

$M^{(r_l)} = A(I - F)^{-1}B^{(r_l)}$

For successful encoding/decoding we require
$M^{(r_1)} = \cdots = M^{(r_{|R|})} = I$

Relaxed requirement:
$\det(M^{(r_l)}) \neq 0$ for $l = 1, \ldots, |R|$.

Success iff
$\prod_{l=1,\ldots,|R|} \det(M^{(r_l)}) \neq 0$

Considered as a polynomial in the $a_{i,j}$'s, $f_{i,j}$'s and $b_{i,j}^{(r_l)}$'s this product is called the transfer polynomial.

Proposition:
$|\det(M^{(r_l)})| = |\det(E^{(r_l)})|$ where
$E^{(r_l)}) = \begin{bmatrix} A & 0 \\ I - F & B^{(r_l)} \end{bmatrix}$

Theorem: The permanent per($M^{(r_l)}$) is the sum of all monomial expressions in the $a_{i,j}$'s, $f_{i,j}$'s and $b_{i,j}^{(r_l)}$'s which correspond to a flow of size h from s' to $\{r_1^{(l)}, \ldots, r_h^{(l)}\}$ in the modified graph.

Proof: Apply the lemma carefully.

As a consequence det($M^{(r_l)}$) is a linear combination of the expressions corresponding to flows. The coefficients being 1 or -1.

In the transfer polynomial $\prod_{l=1,\ldots,|R|} \det(M^{(r_l)})$ every monomial corresponds to a flow system.

Coefficients are integers
which in $\mathbf{F}_q$ becomes elements in $\mathbf{F}_p$, p being the characteristic.

Indeed terms may cancel out when taking the product of the $\det(M^{(r_l)})$'s (See Example 1.1)

However, clearly if all $\det(M^{(r_l)})$'s are different from 0 so is the transfer polynomial.

Theorem 1.1+1.2: A multicast problem is solvable iff the graph contains a flow system of size h. If solvable then solvable with linear network coding whenever $q \geq |R|$.

Proof: Necessity follows from unicast considerations. Assume a flow system exists. The transfer polynomial is non-zero and no indeterminate appears in power exceeding $|R|$. Therefore if $q > |R|$ then over $\mathbf{F}_q$ a non-zero solution exists (here we used the Schwarts-Zippel bound).
We shall later see that $q \geq |R|$ is enough.

Minimal field size

NP-complete problem to find smallest feasible fields size.

To check if $\mathbf{F}_q$ works reduce the transfer polynomial modulo $X^q - X$ where X run through the variables $a_{i,j}$ and $f_{i,j}$.

Easy replacement operation.

Must imply that transfer polynomial can sometimes have exponential many terms.

In linear network coding we always have
$Y(i) = c_1 X_1 + \cdots + c_h X_h$ for some $c_1, \ldots, c_h \in \mathbf{F}_q$.

We shall call $d_c(i) = (c_1, \ldots, c_h)$ the global coding vector for edge i.

A receiver that does not know how encoding was done can learn how to decode (if possible) as follows.

Senders inject into the system h message vectors
$(1, 0, \cdots, 0), (0, 1, 0 \ldots, 0), \ldots, (0, \ldots, 0, 1)$.

These generate the global coding vectors at each edge including the in edges of r_l.

If the received global coding vectors span $\mathbf{F}_q^h$ then proper $b_{i,j}^{(r_l)}$'s can be found.

Jaggi-Sanders algorithm

Jaggi-Sanders algorithm take as input a solvable multicast problem.
It add a new source s' and moves all processes to this point and add edges $e_1, \ldots, e_h$ from s' to S.
In the extended graph a flow system is found.

The algorithm for every receiver keeps a list of edges corresponding to a cut.

Also it updates along the way encoding coefficients in such a way that the global coding vectors corresponding to any of the $|R|$ cuts at any time span the whole of $\mathbf{F}_q^h$.

Edges in the flow system are visited according to an ancestral ordering.

In every update at most one edge is replaced in a given cut.

The Jaggi-Sanders algorithm cont.

Lemma 1.1: Given a basis $\{\vec{b}_1, \ldots, \vec{b}_h\}$ for $\mathbf{F}_q^h$ and $\vec{c} \in \mathbf{F}_q^h$,
there is exactly one choice of $a \in \mathbf{F}_q$ such that $\vec{c} + a\vec{b}_h \in \mathrm{span}_{\mathbf{f}_q}\{\vec{b}_1, \ldots, \vec{b}_{h-1}\}$.

Proof. Expand $\vec{c}$ over $\{\vec{b}_1, \ldots, \vec{b}_h\}$.

$q \geq |R|$ is enough

Given j let $R_{\mathcal{F}}(j)$ be the number of B_l's being updated and let $k = \text{in'}(j) \cap \mathcal{F}$.
Clearly, $k \leq R_{\mathcal{F}}(j) \leq |R|$

Out of the q^k choices of encoding coefficients in worst case when one B_{l_t} fails to span $\mathbf{F}_q^h$ all others do.

Hence at most $R_{\mathcal{F}}(j)q^{k-1}$ choices of $(f_{i_1,j}, \ldots, f_{i_k,j})$ fails.

But $f_{i_1,j} = \cdots = f_{i_k,j} = 0$ has ben counted $R_{\mathcal{F}}$ times. So at most $R_{\mathcal{F}}(j)q^{k-1} - (R_{\mathcal{F}} - 1)$ choices fails.

Probability of success in one step is at least

$$\frac{q^k - R_{\mathcal{F}}q^{k-1} + (R_{\mathcal{F}} - 1)}{q^k}$$

which is > 0 if $q \geq |R|$.