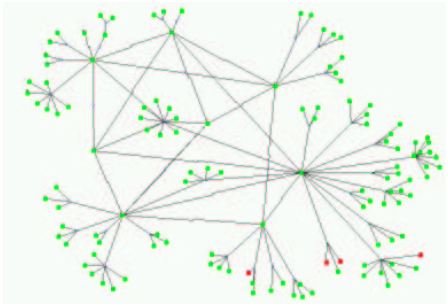
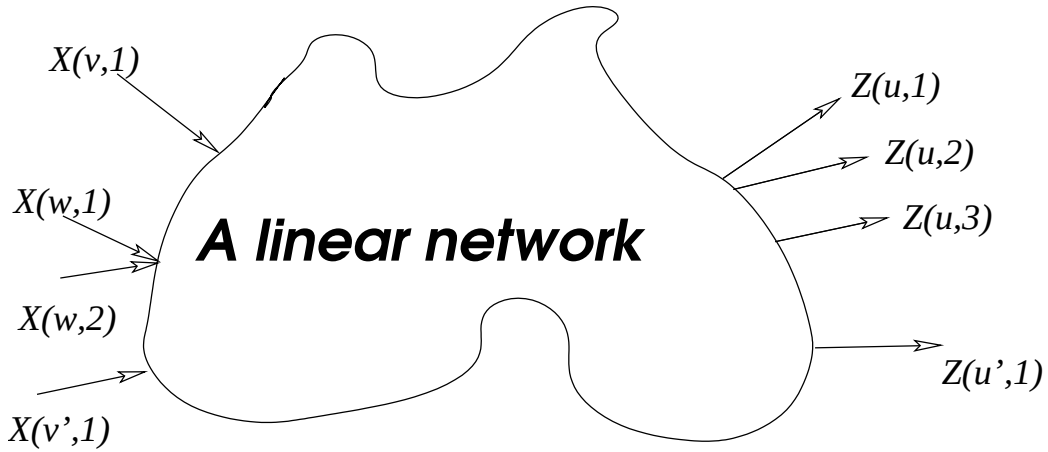
VI — The non multicast case



Questions

- What do we know about the non-multicast case?
- Vector solutions versus instantaneous solutions.
- The issue of linearity!
- Is the non-multicast case interesting?

The algebraic setup



Input vector: $\underline{x}^T = (X(v,1), X(v,2), \ldots, X(v', \mu(v')))$

Output vector: $\underline{z}^T = (Z(u,1), Z(u,2), \ldots, Z(u', \nu(u')))$

Transfer matrix: M, $\underline{z} = M\underline{x} = B \cdot G \cdot A\ \underline{x}$

$\underline{\xi} = (\xi_1, \xi_2, \ldots,) = (\ldots, \alpha_{e,l}, \ldots, \beta_{e',e}, \ldots, \varepsilon_{e',j}, \ldots)$

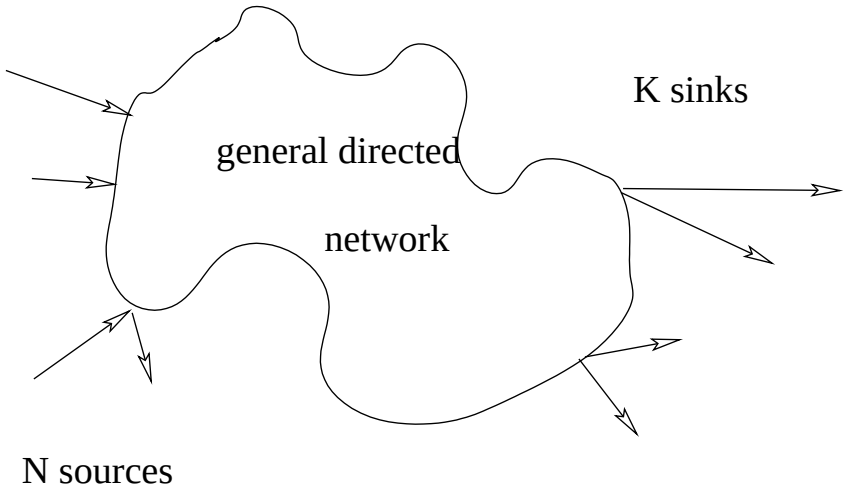
$$\underline{z} = M\underline{x} = B \cdot \underbrace{(I - F^T)^{-1}}_{G^T} \cdot A\ \underline{x}$$

$$\underline{\xi} = (\xi_1, \xi_2, \ldots,) = (\ldots, \alpha_{e,l}, \ldots, \beta_{e',e}, \ldots, \varepsilon_{e',j}, \ldots)$$

For acyclic networks the elements of G (and hence M) are polynomial functions in variables $\underline{\xi} = (\xi_1, \xi_2, \ldots,)$

$\Rightarrow$ an algebraic characterization of flows....

General Problems $(\mathcal{G}, \mathcal{C})$



$$\mathcal{C} = \{(v_i, u_j, \mathcal{X}(v_i, u_j))\}$$

$$M = \begin{pmatrix} M_{1,1} & M_{1,2} & \dots & M_{1,K} \\ M_{2,1} & M_{2,2} & & M_{2,K} \\ \vdots & & & \vdots \\ M_{N,1} & M_{N,2} & \dots & M_{N,K} \end{pmatrix}$$

$M_{i,j}$ corresponds to $c_{i,j} = (v_i, u_j, \mathcal{X}(v_i, u_j))$.

Theorem [Generalized Min-Cut Max-Flow Condition] Let an acyclic, delay-free scalar linear network problem $(\mathcal{G}, \mathscr{C})$ be given and let $M = \{M_{i,j}\}$ be the corresponding transfer matrix relating the set of input nodes to the set of output nodes. The network problem is solvable if and only if there exists an assignment of numbers to $\underline{\xi}$ such that

1. $M_{i,j} = 0$ for all pairs (v_i, v_j) of vertices such that $(v_i, v_j, \mathscr{X}(v_i, v_j)) \notin \mathscr{C}$.

2. If $\mathscr{C}$ contains the connections $(v_{i_1}, v_j, \mathscr{X}(v_{i_1}, v_j)), (v_{i_2}, v_j, \mathscr{X}(v_{i_2}, v_j)), \ldots, (v_{i_\ell}, v_j, \mathscr{X}(v_{i_\ell}, v_j))$ the determinant of $\left[M^T_{i_1,j} M^T_{i_2,j}, \ldots, M^T_{i_\ell,j}\right]$ is nonzero.

The ideal of $(\mathcal{G}, \mathscr{C})$

Entries in $M_{i,j}$ that have to evaluate to zero: $f_1(\underline{\xi}), f_2(\underline{\xi}), \ldots, f_L(\underline{\xi})$

Determinants of submatrices that have to evaluate to nonzero values: $g_1(\underline{\xi}), g_2(\underline{\xi}), \ldots, g_{L'}(\underline{\xi})$

$$\mathbf{Ideal}((\mathcal{G}, \mathscr{C}))$$

$$= \langle f_1(\underline{\xi}), f_2(\underline{\xi}), \ldots, f_L(\underline{\xi}), 1 - \xi_0 \prod_{i=1}^{L'} g_i(\underline{\xi}) \rangle$$

$$\mathbf{Var}((\mathcal{G}, \mathscr{C})) = \{(a_1, a_2, \ldots, a_n) \in \bar{\mathbb{F}}^n :$$

$$f(a_1, a_2, \ldots, a_n) = 0 \; \forall \; f \in \mathbf{Ideal}((\mathcal{G}, \mathscr{C}))\}.$$

The central Theorem

Theorem Let a scalar linear network problem $(\mathcal{G}, \mathscr{C})$ be given. The network problem is solvable if and only if Var$((\mathcal{G}, \mathscr{C})$ is nonempty or equivalently , the ideal Ideal$((\mathcal{G}, \mathscr{C}))$ is a proper ideal of $\bar{\mathbb{F}}[\xi_0, \underline{\xi}]$, i.e Ideal$((\mathcal{G}, \mathscr{C})) \subsetneq \mathbb{F}_2[\xi_0, \underline{\xi}]$.

So why is the gerenal case so much harder?

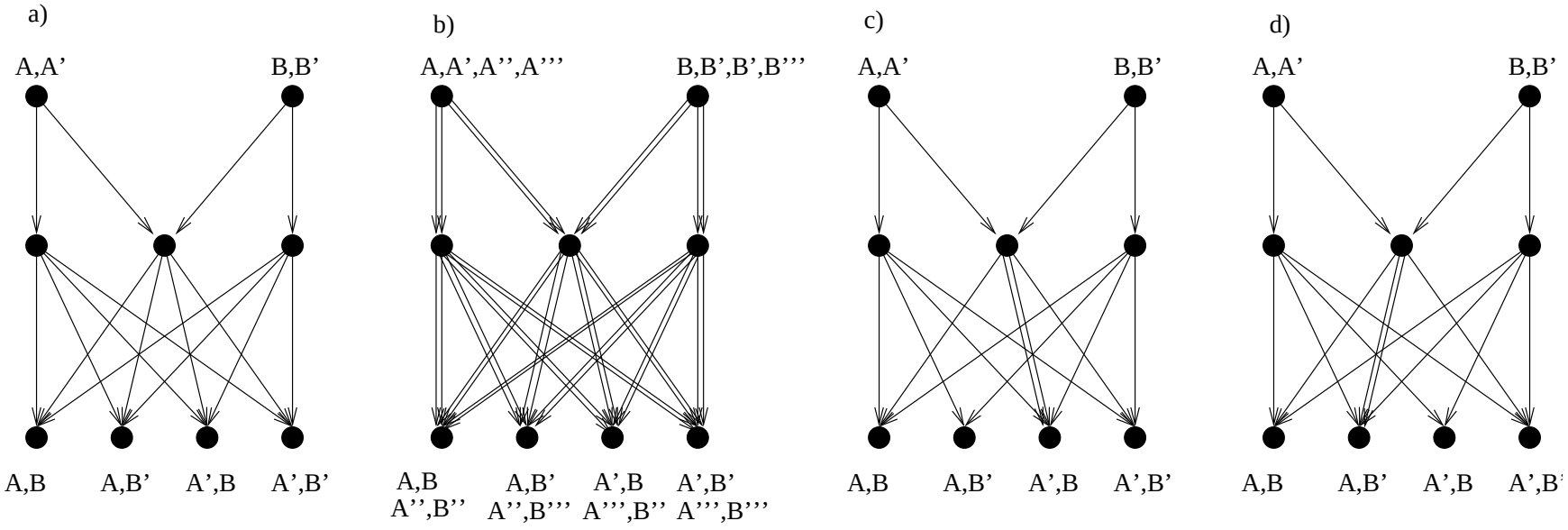
For the general case we need to find solutions to some system of polynomial equations!

For the multicast case we need to find non solutions to some system of polynomial equations!

Another way to phrase this is: In a multicast setup everybody wants everything so the issue of interference is moot!

For the general case we may have carefully balanced solutions where some unwanted information cancels out in clever ways.....

Vector solutions may help

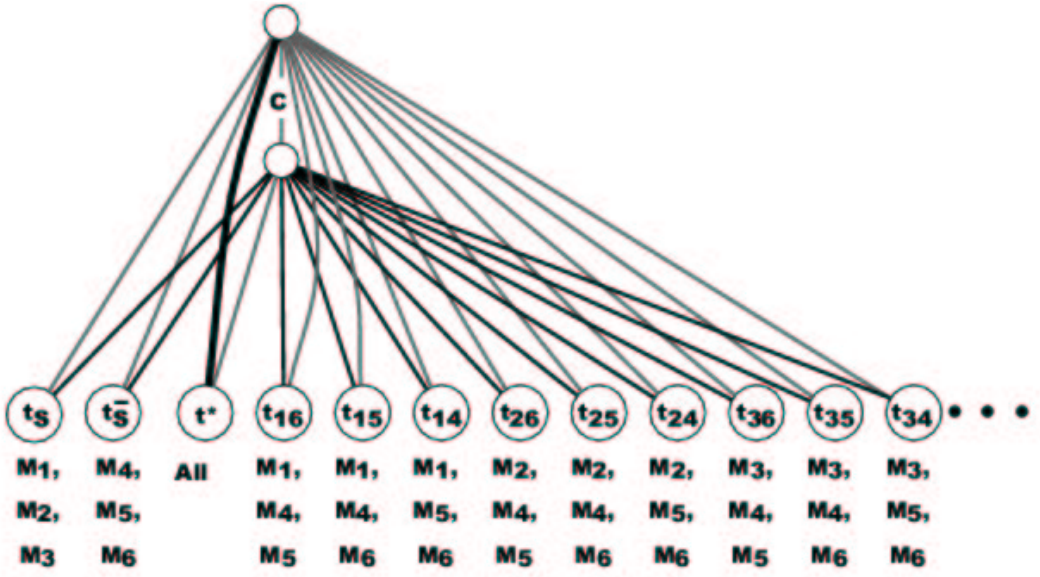


In the general problem a time sharing combination of several solutions which themselves both violate the constraints may be necessary. (This cannot happen in the multicast case!)

Doubling the bandwidth more than doubles capacity! Tripling the bandwidth does not work!

Vector solutions may help

From: A. Rasala-Lehman and E. Lehman, "Vector-Linear Network Codes: Is the Model Broken", preprint, March 2004



Examples of networks that need vector length that are multiples of k for any k.

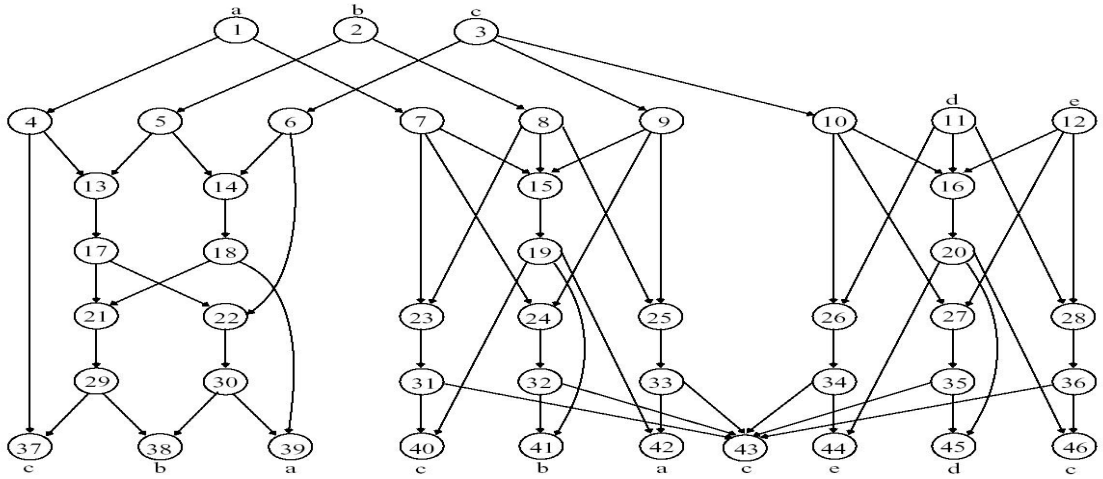
From: A. Rasala-Lehman and E. Lehman, "Vector-Linear Network Codes: Is the Model Broken", preprint, March 2004

By combining networks requiring vector length that are multiples of primes the following bound is derived:

Theorem There exist directed networks with O(n) nodes such that a solution to the network coding problem requires at least an alphabet size of $2^{(e^{\sqrt{n^{1/3}}})}$

More strange news...

R. Dougherty, C. Freiling, and K. Zeger, "Insufficiency of Linear Coding in Network Information Flow", preprint, February 2004



This network is not solvable over any Galois field, including vector versions thereof!

(still the network has a distinctly linear feel to it....)

Non-multicast connections -use of cost criterion

- We propose a linear optimization problem whose minimum cost is no greater than the minimum cost of any routing solution
- Moreover, feasible solutions correspond to network codes that perform linear operations on vectors created from the source processes
- Main idea: create a set partition of $\{1, \ldots, M\}$ that represents the sources that can be mixed (combined linearly) on links going into i.
- Code construction steps through the nodes in topological order, examining the outgoing links and defining global coding vectors on them.

Non-multicast connections -use of cost criterion

- For any node i, let $T(i)$ denote the sinks that are accessible from i
- Let $\mathcal{C}(i)$ be a set partition of $\{1, \ldots, M\}$ that represents the sources that can be mixed (combined linearly) on links going into i. For a given $C \in \mathcal{C}(i)$, the sinks that receive a source process in C by way of link (j, i) in A (set of arcs) either receive all the source processes in C or none at all.

Non-multicast connections -use of cost criterion

$$
\begin{aligned}
\text{minimize} \quad & \textstyle\sum_{(i,j)\in A} a_{ij} z_{ij} \\
\text{subject to} \quad & c_{ij} \geq z_{ij} = \textstyle\sum_{C \in \mathcal{C}(j)} y_{ij}^{(C)}, && \forall\, (i,j) \in A, \\
& y_{ij}^{(C)} \geq \textstyle\sum_{m \in C} x_{ij}^{(t,m)}, && \forall\, (i,j) \in A,\, t \in T,\, C \in \mathcal{C}(j), \\
& x_{ij}^{(t,m)} \geq 0, && \forall\, (i,j) \in A,\, t \in T,\, m = 1, \ldots, M,
\end{aligned}
$$

$$
\sum_{\{j|(i,j)\in A\}} x_{ij}^{(t,m)} - \sum_{\{j|(j,i)\in A\}} x_{ji}^{(t,m)} = \begin{cases} R_m & \text{if } v = s_m \text{ and } m \in D(t), \\ -R_m & \text{if } m \in D(i), \\ 0 & \text{otherwise,} \end{cases}
$$

$$
\forall\, i \in A,\, t \in T,\, m = 1, \ldots, M, \quad (1)
$$

where we define $D(i) := \emptyset$ for i in $N \setminus T$. Again, the optimization problem can be easily modified to accommodate convex cost functions.

Is the non-multicast case interesting?

Summary:

The non multicast scenario exhibits far more subtleties than the multicast setup. This is due to the fact that cancellations now need to be carefully arranged.

There are some generalizations to vector solutions which can be incorporated into the algebraic framework.

Not even the principle problem of linearity vs. nonlinear operation is entirely clear.

From a practical point of view a non interacting arrangement of multicast is most interesting and robust.

Network coding for multicast relation to compression and generalization of Slepian-Wolf

Overview

Review of Slepian-Wolf

Distributed network compression

Error exponents Source-channel separation issues

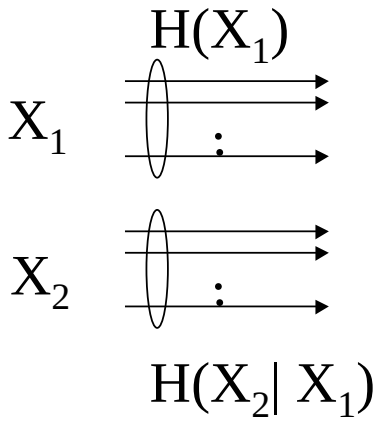
Code construction for finite field multiple access networks

Distributed data compression

Consider two correlated sources $(X, Y) \sim p(x, y)$ that must be separately encoded for a user who wants to reconstruct both

What information transmission rates from each source allow decoding with arbitrarily small probability of error?

E.g.



Distributed source code

A $((2^{nR_1}, 2^{nR_2}), n)$ distributed source code for joint source (X, Y) consists of encoder maps

$$\begin{aligned} f_1 : \mathcal{X}^n &\rightarrow \{1, 2, \ldots, 2^{nR_1}\} \\ f_2 : \mathcal{Y}^n &\rightarrow \{1, 2, \ldots, 2^{nR_2}\} \end{aligned}$$

and a decoder map

$$g : \{1, 2, \ldots, 2^{nR_1}\} \times \{1, 2, \ldots, 2^{nR_2}\} \rightarrow \mathcal{X}^n \times \mathcal{Y}^n$$

- X^n is mapped to $f_1(X^n)$
- Y^n is mapped to $f_2(Y^n)$
- (R_1, R_2) is the rate pair of the code

Probability of error

$$P_e^{(n)} = \Pr\{g(f_1(X^n), f_2(Y^n)) \neq (X^n, Y^n)\}$$

Slepian-Wolf

Definitions:

A rate pair (R_1, R_2) is *achievable* if there exists a sequence of $((2^{nR_1}, 2^{nR_2}), n)$ distributed source codes with probability of error $P_e^{(n)} \to 0$ as $n \to \infty$

achievable rate region - closure of the set of achievable rates

Slepian-Wolf Theorem:

For the distributed source coding problem for source (X, Y) drawn i.i.d. $\sim p(x, y)$, the achievable rate region is

$$\begin{aligned} R_1 &\qquad H(X|Y) \\ R_2 &\qquad H(Y|X) \\ R_1 + R_2 &\qquad H(X, Y) \end{aligned}$$

Proof of achievability

Main idea: show that if the rate pair is in the Slepian-Wolf region, we can use a random binning encoding scheme with typical set decoding to obtain a probability of error that tends to zero

Coding scheme:

Source X assigns every sourceword $x \in \mathcal{X}^n$ randomly among 2^{nR_1} bins, and source Y independently assigns every $y \in \mathcal{Y}^n$ randomly among 2^{nR_2} bins

Each sends the bin index corresponding to the message

the receiver decodes correctly if there is exactly one jointly typical sourceword pair corresponding to the received bin indexes, otherwise it declares an error

Random binning for single source compression

An encoder that knows the typical set can compress a source X to $H(X)+\epsilon$ without loss, by employing separate codes for typical and atypical sequences

Random binning is a way to compress a source X to $H(X)+\epsilon$ with asymptotically small probability of error without the encoder knowing the typical set, as well as the decoder knows the typical set

the encoder maps each source sequence X^n uniformly at random into one of 2^{nR} bins

the bin index, which is R bits long, forms the code

the receiver decodes correctly if there is exactly one typical sequence corresponding to the received bin index

Error analysis

An error occurs if:

a) the transmitted sourceword is not typical, i.e. event

$$E_0 = \{\mathbf{X} \notin A_\epsilon^{(n)}\}$$

b) there exists another typical sourceword in the same bin, i.e.event

$$E_1 = \{\exists \mathbf{x}' \neq \mathbf{X} : f(\mathbf{x}') = f(\mathbf{X}), \mathbf{x}' \in A_\epsilon^{(n)}\}$$

Use union of events bound:

$$\begin{aligned} P_e^{(n)} &= \Pr(E_0 \cup E_1) \\ &\leq \Pr(E_0) + \Pr(E_1) \end{aligned}$$

Error analysis continued

$\Pr(E_0) \to 0$ by the Asymptotic Equipartition Property (AEP)

$$\begin{aligned}
\Pr(E_1) &= \sum_{\mathbf{x}} \Pr\{\exists \mathbf{x}' \neq \mathbf{x} : f(\mathbf{x}') = f(\mathbf{x}), \\
&\qquad\qquad \mathbf{x}' \in A_\epsilon^{(n)}\} \\
&\leq \sum_{\mathbf{x}} \sum_{\substack{\mathbf{x}' \neq \mathbf{x} \\ \mathbf{x}' \in A_\epsilon^{(n)}}} \Pr(f(\mathbf{x}') = f(\mathbf{x})) \\
&= \sum_{\mathbf{x}} |A_\epsilon^{(n)}|\, 2^{-nR} \\
&\leq 2^{-nR}\, 2^{n(H(X)+\epsilon)} \\
&\to 0 \text{ if } R > H(X)
\end{aligned}$$

For sufficiently large n,

$$\begin{aligned} & \Pr(E_0), \Pr(E_1) < \epsilon \\ \Rightarrow \quad & P_\epsilon^{(n)} < 2\epsilon \end{aligned}$$

Jointly typical sequences

The set $A_\epsilon^{(n)}$ of jointly typical sequences is the set of sequences $(\mathbf{x}, \mathbf{y}) \in \mathcal{X}^n \times \mathcal{Y}^n$ with probability:

$$2^{-n(H(X)+\epsilon)} \leq p_{\mathbf{X}}(\mathbf{x}) \leq 2^{-n(H(X)-\epsilon)}$$

$$2^{-n(H(Y)+\epsilon)} \leq p_{\mathbf{Y}}(\mathbf{y}) \leq 2^{-n(H(Y)-\epsilon)}$$

$$2^{-n(H(X,Y)+\epsilon)} \leq p_{\mathbf{X},\mathbf{Y}}(\mathbf{x}, \mathbf{y}) \leq 2^{-n(H(X,Y)-\epsilon)}$$

for $(\mathbf{X}, \mathbf{Y})$ sequences of length n IID according to $p_{\mathbf{X},\mathbf{Y}}(\mathbf{x}, \mathbf{y}) = \prod_{i=1}^{n} p_{X,Y}(x_i, y_i)$

Size of typical set:

$$|A_\epsilon^{(n)}| \leq 2^{n(H(X,Y)+\epsilon)}$$

Proof:

$$
\begin{aligned}
1 &= \sum p(\mathbf{x},\mathbf{y}) \\
&\quad \sum_{A_\epsilon^{(n)}} p(\mathbf{x},\mathbf{y}) \\
&\quad |A_\epsilon^{(n)}| 2^{-n(H(X,Y)+\epsilon)}
\end{aligned}
$$

Conditionally typical sequences

The conditionally typical set $A_\epsilon^{(n)}(X|\mathbf{y})$ for a given typical $\mathbf{y}$ sequence is the set of $\mathbf{x}$ sequences that are jointly typical with the given $\mathbf{y}$ sequence.

Size of conditionally typical set:

$$|A_\epsilon^{(n)}(X|\mathbf{y})| \leq 2^{n(H(X|Y)+\epsilon)}$$

Proof:

For $(\mathbf{x}, \mathbf{y}) \in A_\epsilon^{(n)}(X, Y)$,

$$\begin{aligned} p(\mathbf{y}) &\doteq 2^{-n(H(Y)\pm\epsilon)} \\ p(\mathbf{x}, \mathbf{y}) &\doteq 2^{-n(H(X,Y)\pm\epsilon)} \\ \Rightarrow p(\mathbf{x}|\mathbf{y}) &= \frac{p(\mathbf{x}, \mathbf{y})}{p(\mathbf{y})} \\ &\doteq 2^{-n(H(X|Y)\pm 2\epsilon)} \end{aligned}$$

Hence

$$\begin{aligned} 1 &\qquad \sum_{\mathbf{x}\in A_\epsilon^{(n)}(X|\mathbf{y})} p(\mathbf{x}|\mathbf{y}) \\ &\qquad |A_\epsilon^{(n)}| 2^{-n(H(X|Y)+2\epsilon)} \end{aligned}$$

Proof of achievability – error analysis

Errors occur if:

a) the transmitted sourcewords are not jointly typical, i.e. event

$$E_0 = \{(X, Y) \notin A_\epsilon^{(n)}\}$$

b) there exists another pair of jointly typical sourcewords in the same pair of bins, i.e. one or more of the following events

$$\begin{aligned} E_1 &= \{\exists \mathbf{x}' \neq \mathbf{X} : f_1(\mathbf{x}') = f_1(\mathbf{X}), (\mathbf{x}', \mathbf{Y}) \in A_\epsilon^{(n)}\} \\ E_2 &= \{\exists \mathbf{y}' \neq \mathbf{Y} : f_2(\mathbf{y}') = f_2(\mathbf{Y}), (\mathbf{X}, \mathbf{y}') \in A_\epsilon^{(n)}\} \\ E_{12} &= \{\exists (\mathbf{x}', \mathbf{y}') : \mathbf{x}' \neq \mathbf{X}, \mathbf{y}' \neq \mathbf{Y}, f_1(\mathbf{x}') = f_1(\mathbf{X}), \\ &\quad f_2(\mathbf{y}') = f_2(\mathbf{Y}), (\mathbf{x}', \mathbf{y}') \in A_\epsilon^{(n)}\} \end{aligned}$$

Use union of events bound:

$$\begin{aligned} P_e^{(n)} &= \Pr(E_0 \cup E_1 \cup E_2 \cup E_{12}) \\ &\leq \Pr(E_0) + \Pr(E_1) + \Pr(E_2) + \Pr(E_{12}) \end{aligned}$$

Error analysis continued

$\Pr(E_0) \to 0$ by the AEP

$$
\begin{aligned}
\Pr(E_1) &= \sum_{(\mathbf{x},\mathbf{y})} \Pr\{\exists \mathbf{x}' \neq \mathbf{x} : f_1(\mathbf{x}') = f_1(\mathbf{x}), \\
&\qquad\qquad (\mathbf{x}', \mathbf{y}) \in A_\epsilon^{(n)}\} \\
&\leq \sum_{(\mathbf{x},\mathbf{y})} \sum_{\substack{\mathbf{x}' \neq \mathbf{x} \\ (\mathbf{x}', \mathbf{y}) \in A_\epsilon^{(n)}}} \Pr(f_1(\mathbf{x}') = f_1(\mathbf{x})) \\
&= \sum_{(\mathbf{x},\mathbf{y})} |A_\epsilon^{(n)}(X|\mathbf{y})|\, 2^{-nR_1} \\
&\leq 2^{-nR_1}\, 2^{n(H(X|Y)+2\epsilon)} \\
&\to 0 \text{ if } R_1 > H(X|Y)
\end{aligned}
$$

Similarly,

$$\begin{aligned}
\Pr(E_2) &\leq 2^{-nR_2}\, 2^{n(H(Y|X)+2\epsilon)} \\
&\rightarrow 0 \text{ if } R_2 > H(Y|X) \\
\Pr(E_{12}) &\leq 2^{-n(R_1+R_2)}\, 2^{n(H(X,Y)+\epsilon)} \\
&\rightarrow 0 \text{ if } R_1 + R_2 > H(X,Y)
\end{aligned}$$

Error analysis continued

Thus, if we are in the Slepian-Wolf rate region, for sufficiently large n,

$$\begin{aligned} & \Pr(E_0), \Pr(E_1), \Pr(E_2), \Pr(E_{12}) < \epsilon \\ \Rightarrow \quad & P_\epsilon^{(n)} < 4\epsilon \end{aligned}$$

Since the average probability of error is less than 4ϵ, there exist at least one code (f_1^*, f_2^*, g^*) with probability of error $< 4\epsilon$.

Thus, there exists a sequence of codes with $P_\epsilon^{(n)} \to 0$.

Model for distributed network compression

arbitrary directed graph with integer capacity links

discrete memoryless source processes with integer bit rates

randomized linear network coding over vectors of bits in $\mathbb{F}_2$

coefficients of overall combination transmitted to receivers

receivers perform minimum entropy or maximum a posteriori probability decoding

Distributed compression problem

Consider

two sources of bit rates r_1, r_2, whose output values in each unit time period are drawn i.i.d. from the same joint distribution Q

linear network coding in $\mathbb{F}_2$ over vectors of nr_1 and nr_2 bits from each source respectively

Define

m_1 and m_2 the minimum cut capacities between the receiver and each source respectively

m_3 the minimum cut capacity between the receiver and both sources

L the maximum source-receiver path length

Theorem 1 *The error probability at each receiver using minimum entropy or maximum a posteriori probability decoding is at most* $\sum_{i=1}^{3} p_e^i$, *where*

$$
\begin{aligned}
p_e^1 &\leq \exp\left\{ -n \min_{X_1,X_2} \left(D(P_{X_1X_2}||Q) \right.\right. \\
&\qquad \left.\left. + \left| m_1(1 - \frac{1}{n}\log L) - H(X_1|X_2) \right|^+ \right) + 2^{2r_1+r_2}\log(n+1) \right\} \\
p_e^2 &\leq \exp\left\{ -n \min_{X_1,X_2} \left(D(P_{X_1X_2}||Q) \right.\right. \\
&\qquad \left.\left. + \left| m_2(1 - \frac{1}{n}\log L) - H(X_2|X_1) \right|^+ \right) + 2^{r_1+2r_2}\log(n+1) \right\} \\
p_e^3 &\leq \exp\left\{ -n \min_{X_1,X_2} \left(D(P_{X_1X_2}||Q) \right.\right. \\
&\qquad \left.\left. + \left| m_3(1 - \frac{1}{n}\log L) - H(X_1X_2) \right|^+ \right) + 2^{2r_1+2r_2}\log(n+1) \right\}
\end{aligned}
$$

Distributed compression

Redundancy is removed or added in different parts of the network depending on available capacity

Achieved without knowledge of source entropy rates at interior network nodes

For the special case of a Slepian-Wolf source network consisting of a link from each source to the receiver, the network coding error exponents reduce to known error exponents for linear Slepian-Wolf coding [Csi82]

Proof outline

Error probability $\leq \sum_{i=1}^{3} p_e^i$, where

- p_e^1 is the probability of correctly decoding X_2 but not X_1,
- p_e^2 is the probability of correctly decoding X_1 but not X_2
- p_e^3 is the probability of wrongly decoding X_1, X_2

Proof approach using method of types similar to that in [Csi82]

Types $P_{\mathbf{x}_i}$, joint types $P_{\mathbf{xy}}$ are the empirical distributions of elements in vectors $\mathbf{x}_i$

Proof outline (cont'd)

Bound error probabilities by summing over

sets of joint types

$$\mathcal{P}_n^i = \begin{cases} \{P_{X_1\tilde{X}_1X_2\tilde{X}_2} \mid \tilde{X}_1 \neq X_1, \tilde{X}_2 = X_2\} & i = 1 \\ \{P_{X_1\tilde{X}_1X_2\tilde{X}_2} \mid \tilde{X}_1 = X_1, \tilde{X}_2 \neq X_2\} & i = 2 \\ \{P_{X_1\tilde{X}_1X_2\tilde{X}_2} \mid \tilde{X}_1 \neq X_1, \tilde{X}_2 \neq X_2\} & i = 3 \end{cases}$$

where $X_i, \tilde{X}_i \in \mathbb{F}_2^{nr_i}$

sequences of each type

$$\mathcal{T}_{X_1X_2} = \left\{ [\ \mathbf{x}\ \ \mathbf{y}\ ] \in \mathbb{F}_2^{n(r_1+r_2)} \;\middle|\; P_{\mathbf{xy}} = P_{X_1X_2} \right\}$$

$$\mathcal{T}_{\tilde{X}_1\tilde{X}_2|X_1X_2}(\mathbf{xy}) = \left\{ [\ \tilde{\mathbf{x}}\ \ \tilde{\mathbf{y}}\ ] \in \mathbb{F}_2^{n(r_1+r_2)} \;\middle|\; P_{\tilde{\mathbf{x}}\tilde{\mathbf{y}}\mathbf{xy}} = P_{\tilde{X}_1\tilde{X}_2X_1X_2} \right\}$$

Proof outline (cont'd)

Define

- $P_i, i = 1, 2$, the probability that distinct $(\mathbf{x}, \mathbf{y}), (\tilde{\mathbf{x}}, \mathbf{y})$, where $\mathbf{x} \neq \tilde{\mathbf{x}}$, at the receiver

- P_3, the probability that $(\mathbf{x}, \mathbf{y}), (\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$, where $\mathbf{x} \neq \tilde{\mathbf{x}}$,$\mathbf{y} \neq \tilde{\mathbf{y}}$, are mapped to the same output at the receiver

These probabilities can be calculated for a given network, or bounded in terms of block length n and network parameters

Proof outline (cont'd)

A link with 1 nonzero incoming signal carries the zero signal with probability $\frac{1}{2^{nc}}$, where c is the link capacity

this is equal to the probability that a pair of distinct input values are mapped to the same output on the link

We can show by induction on the minimum cut capacities m_i that

$$\begin{aligned} P_i &\leq \left(1 \quad (1 \quad \frac{1}{2^n})^L\right)^{m_i} \\ &\leq \left(\frac{L}{2^n}\right)^{m_i} \end{aligned}$$

Proof outline (cont'd)

We substitute in

cardinality bounds

$$\begin{aligned} |\mathcal{P}_n^1| &< (n+1)^{2^{2r_1+r_2}} \\ |\mathcal{P}_n^2| &< (n+1)^{2^{r_1+2r_2}} \\ |\mathcal{P}_n^3| &< (n+1)^{2^{2r_1+2r_2}} \\ |\mathcal{T}_{X_1X_2}| &\leq \exp\{nH(X_1X_2)\} \\ |\mathcal{T}_{\tilde{X}_1\tilde{X}_2|X_1X_2}(\mathbf{xy})| &\leq \exp\{nH(\tilde{X}_1\tilde{X}_2|X_1X_2)\} \end{aligned}$$

probability of source vector of type $(\mathbf{x}, \mathbf{y}) \in \mathcal{T}_{X_1X_2}$

$$Q^n(\mathbf{xy}) = \exp\{ \ n(D(P_{X_1X_2}||Q) + H(X_1X_2))\}$$

Proof outline (cont'd)

and the decoding conditions

minimum entropy decoder:

$$H(\tilde{X}_1\tilde{X}_2) \leq H(X_1X_2)$$

maximum a posteriori probability decoder:

$$D(P_{\tilde{X}_1\tilde{X}_2}||Q) + H(\tilde{X}_1\tilde{X}_2) \leq D(P_{X_1X_2}||Q) + H(X_1X_2)$$

to obtain the result

Conclusions

Distributed randomized network coding can achieve distributed compression of correlated sources

Error exponents generalize results for linear Slepian Wolf coding

Further work: investigation of non-uniform code distributions, other types of codes, and other decoding schemes

Network coding for security and robustness

Outline

Network coding for detecting attacks

Network management requirements for robustness

Centralized versus distributed network management

Byzantine security

Robustness against faulty/malicious components with arbitrary behavior, e.g.

- dropping packets
- misdirecting packets
- sending spurious information

Abstraction as Byzantine generals problem [LSP82]

Byzantine robustness in networking [P88,MR97,KMM98,CL99]

Byzantine detection with network coding [HLKMEK04]

Distributed randomized network coding can be extended to detect Byzantine behavior

Small computational and communication overhead

– small number of hash bits included with each packet, calculated as simple polynomial function of data

Require only that a Byzantine attacker does not design and supply modified packets with complete knowledge of other nodes' packets

Byzantine modification detection scheme

Suppose each packet contains θ data symbols $x_1, \ldots, x_\theta$ and $\phi \leq \theta$ hash symbols $y_1, \ldots, y_\phi$

Consider the function $\pi(x_1, \ldots, x_k) = x_1^2 + \quad + x_k^{k+1}$

Set

$$\begin{aligned} y_i &= \pi(x_{(i-1)k+1}, \ldots, x_{ik}) \quad \text{for} \quad i = 1, \ldots, \phi \quad 1 \\ y_\phi &= \pi(x_{(\phi-1)k+1}, \ldots, x_\theta) \end{aligned}$$

where $k = \left\lceil \frac{\theta}{\phi} \right\rceil$ is a design parameter trading off overhead against detection probability

Detection probability

[HLKMEK04] If the receiver gets s genuine packets, then the detection probability is at least 1 $\left(\frac{k+1}{q}\right)^s$.

E.g. With 2% overhead ($k = 50$), code length=7, $s = 5$, the detection probability is 98.9%.

with 1% overhead ($k = 100$), code length=8, $s = 5$, the detection probability is 99.0%.

Analysis

Let M be the matrix whose i^{th} row $\underline{m_i}$ represents the concatenation of the data and corresponding hash value for packet i

Suppose the receiver tries to decode using

- s unmodified packets, represented as $C_a\,[M|I]$, where the i^{th} row of the coefficient matrix C_a is the vector of code coefficients of the i^{th} packet

- $r \quad s$ modified packets, represented by $[C_b M + V|C_b]$, where V is an arbitrary matrix

Analysis (cont'd)

Let $C = \left[\frac{C_a}{C_b}\right]$

Decoding is equivalent to pre-multiplying the matrix

$$\left[\begin{array}{c|c} C_aM & C_a \\ \hline C_bM + V & C_b \end{array}\right]$$

with C^{-1}, which gives

$$\left[\begin{array}{c|c} M + C^{-1}\left[\frac{0}{V}\right] & I \end{array}\right]$$

For any C_b and V, since receiver decodes only with a full rank set of packets, possible values of C_a are s.t. C is non-singular

Analysis (cont'd)

We can show that

for each of s packets, the attacker knows only that the decoded value will be one of $q^{\text{rank}(V)}$ possibilities

$$\left\{ \underline{m}_i + \sum_{j=1}^{\text{rank}(V)} \gamma_{i,j}\underline{v}_j \;\middle|\; \gamma_{i,j} \in \mathbb{F}_q \right\}$$

at most $k+1$ out of the q vectors in a set $\{\underline{u} + \gamma\underline{v} | \gamma \in \mathbb{F}_q\}$, where $\underline{u} = (u_1, \ldots, u_{k+1})$ is a fixed length-$(k+1)$ vector and $\underline{v} = (v_1, \ldots, v_{k+1})$ a fixed nonzero length-$(k+1)$ vector, can satisfy the property that the last element of the vector equals the hash of the first k elements.

Network mgt for link failure recovery [HMK02, HMK03]

Structured schemes for link failure recovery, e.g. end-to-end path protection, loopback, generalized loopback

Network coding admits any solution feasible on surviving links

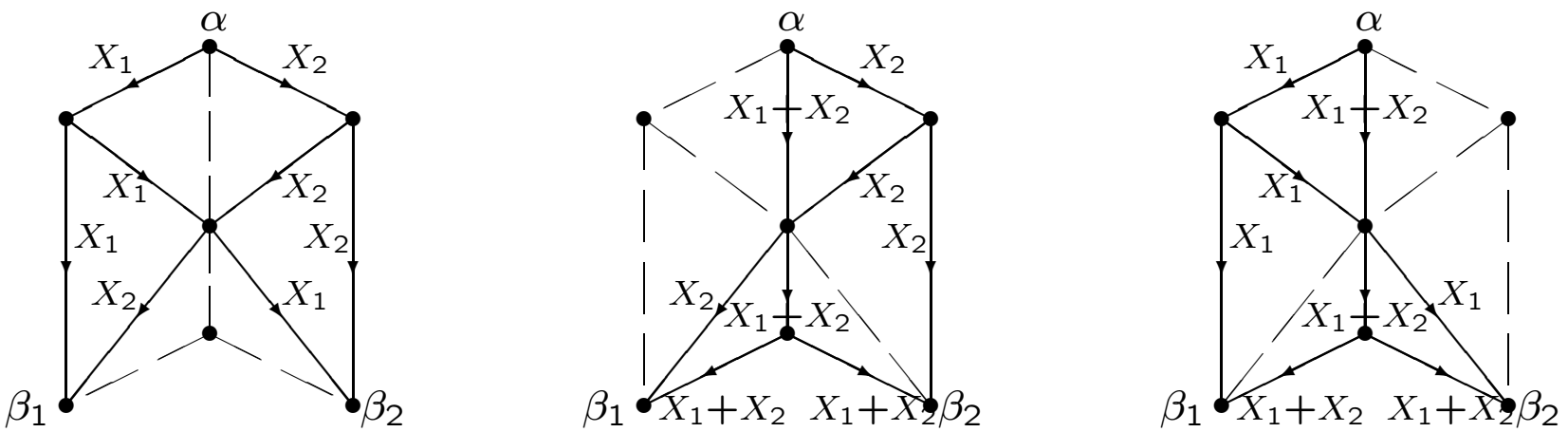
Network management information directs network's response to different link failures

Questions:
-How to quantify fundamental amount of information needed

to direct recovery?
-How do different types of recovery schemes compare in management overhead?

A theoretical framework for network management

Network management information can be quantified by the log of the number of different behaviors (codes) used [tbh]



Allowing general network coding solutions gives fundamental limits on management information required

Classes of failure recovery schemes considered

Receiver-based schemes: only receivers change behavior under different failure scenarios

Network-wide schemes: any node may change behavior, includes receiver based schemes as a special case

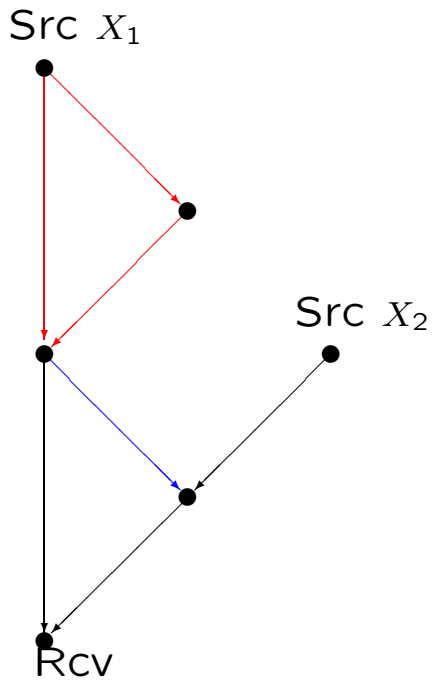
Linear schemes: linear operations at all nodes

Nonlinear receiver-based schemes: nonlinear decoding at receivers

Need for network management

A link h is called *integral* if there exists some subgraph of the network on which the set of source-receiver connections is feasible if and only if h has not failed.

For any network connection problem with at least one integral link whose failure is recoverable, no single linear code can cover the no-failure scenario and all recoverable failures


Src X_1
Src X_2
Rcv

Bounds on network management

Network management for single recoverable link, using network parameters

r, number of source processes transmitted in network;

m, the number of links in a minimum cut between the source nodes and receiver nodes;

d, the number of receiver nodes;

t_{min}, the minimum number of terminal links among all receivers.

Some bounds

Tight lower bounds on no. of linear codes for general case:

receiver-based	$\left\lceil \frac{m}{m-r} \right\rceil$
network-wide	$\left\lceil \frac{m+1}{m-r+1} \right\rceil$

- Tight upper bounds on no. of linear codes for the single-receiver:

receiver-based	$\begin{cases} r+1 & \text{for } r = 1 \text{ or } m-1 \\ r & \text{for } 2 \le r \le m-2 \end{cases}$
network-wide	$\begin{cases} r+1 & \text{for } r = 1,\ r = 2 = m-1 \\ r & \text{for } r = 2 \le m-2, \\ & \quad r = 3,\ r = m-1 \ge 3 \\ r-1 & \text{for } 4 \le r \le m-2 \end{cases}$

- Upper bound on no. of linear codes for multicast: $(r^2+2)(r+1)^{d-2}$

- Tight lower bounds for nonlinear receiver-based codes for multicast:

$$\begin{cases} r & \text{for } 1 < r = t_{\min} - 1 \\ 1 & \text{for } r = 1 \text{ or } r \leq t_{\min} - 2 \end{cases}$$