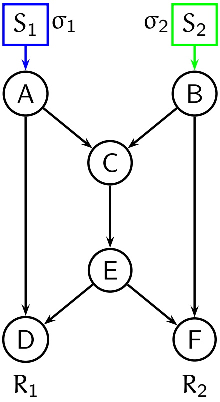
NETWORK MULTICAST:
A theoretical Minimum and an Open Problem

Emina Soljanin

Rutgers

Aalto U., August 2016

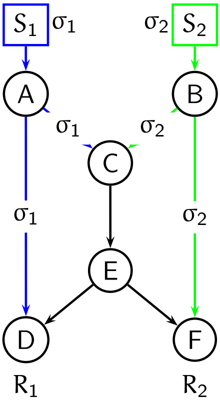
NETWORK MULTICAST – The Butterfly



- Sources S_1 and S_2 produce bits σ_1 and σ_2.
- Each receiver needs bits from both sources.
- The edges have unit capacity.

Can both sources simulaneosly transmit to both reseivers?

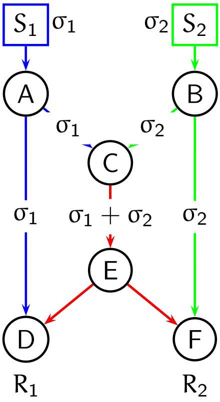
NETWORK MULTICAST – The Butterfly



- Sources S_1 and S_2 produce bits σ_1 and σ_2.
- Each receiver needs bits from both sources.
- The edges have unit capacity.

Can both sources simulaneosly transmit to both reseivers?

NETWORK MULTICAST – The Butterfly

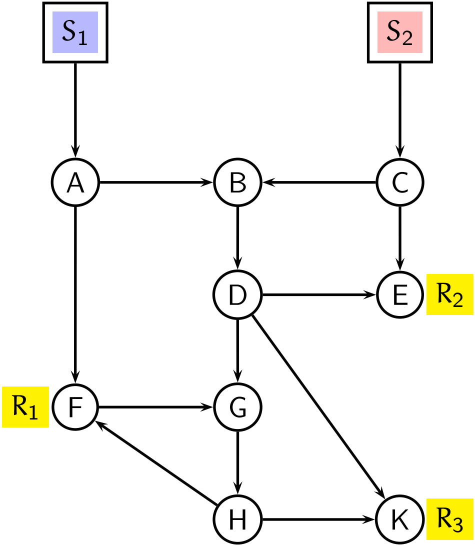


- Sources S_1 and S_2 produce bits σ_1 and σ_2.
- Each receiver needs bits from both sources.
- The edges have unit capacity.

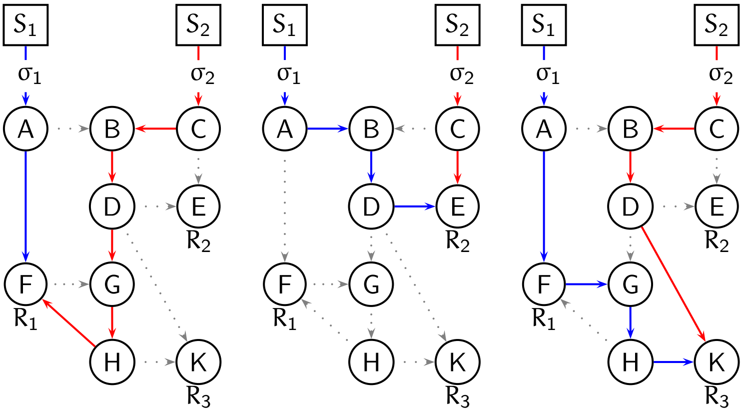
Can both sources simulaneosly transmit to both reseivers?

Yes if nodes can XOR bits.

A Network for Multicast



Three Unicasts in a Multicast Network



Network Multicast Theorem

Conditions:

- Network is represented as a directed, acyclic graph.
- Edges have unit-capacity and parallel edges are allowed.
- There are h unit-rate information sources $S_1, \ldots, S_h$.
- There are N receivers $R_1, \ldots, R_N$ located at N distinct nodes.
- Between the sources and each receiver node,
  - the number of edges in the min-cut is h (or equivalently)
  - there are h edge-disjoint paths (S_i, R_j) for $1 \leqslant i \leqslant h$.

Network Multicast Theorem

Conditions:

- Network is represented as a directed, acyclic graph.
- Edges have unit-capacity and parallel edges are allowed.
- There are h unit-rate information sources $S_1, \ldots, S_h$.
- There are N receivers $R_1, \ldots, R_N$ located at N distinct nodes.
- Between the sources and each receiver node,
  - the number of edges in the min-cut is h (or equivalently)
  - there are h edge-disjoint paths (S_i, R_j) for $1 \leqslant i \leqslant h$.

Claim: There exists a multicast transmission scheme of rate h.

Moreover, multicast at rate h

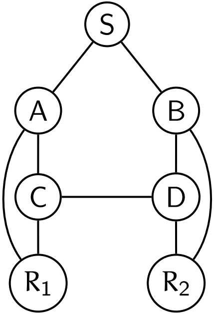
- cannot always be achieved by routing, but
- can be achieved by allowing the nodes to linearly combine their inputs over a sufficiently large finite field.

UNDIRECTED GRAPHS

- The main theorem does not hold.
- Coding can at most double the throughput.

UNDIRECTED GRAPHS

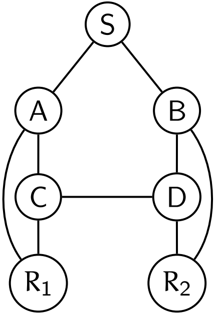
- The main theorem does not hold.
- Coding can at most double the throughput.



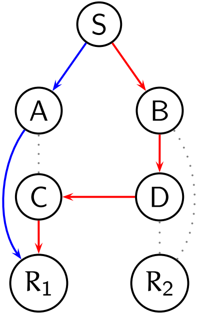
Original Graph

UNDIRECTED GRAPHS

- The main theorem does not hold.
- Coding can at most double the throughput.



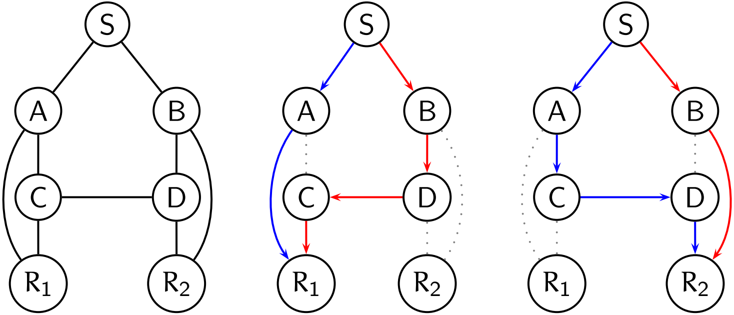
Original Graph



Paths to R_1

UNDIRECTED GRAPHS

- The main theorem does not hold.
- Coding can at most double the throughput.



Original Graph | Paths to R_1 | Paths to R_2

Network Multicast – Linear Combining

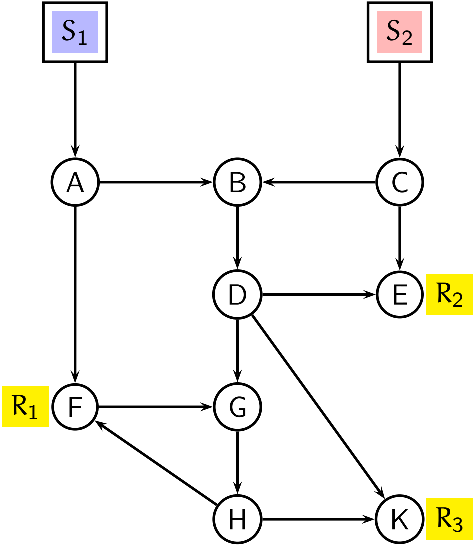
- Source S_i emits σ_i which is an element of some finite field.
- Edges carry linear combinations of their parent node inputs.
- Consequently,
  edges carry linear combinations of source symbols σ_i.

Network Multicast – Linear Combining

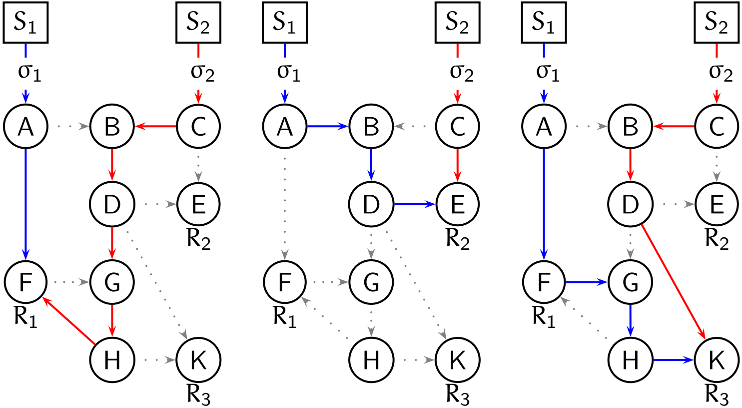
- Source S_i emits σ_i which is an element of some finite field.
- Edges carry linear combinations of their parent node inputs.
- Consequently,
  edges carry linear combinations of source symbols σ_i.

Network Coding Multicast Problem:
How should nodes combine their inputs to ensure that any h edges observed by a receiver carry independent combinations of σ_i-s?

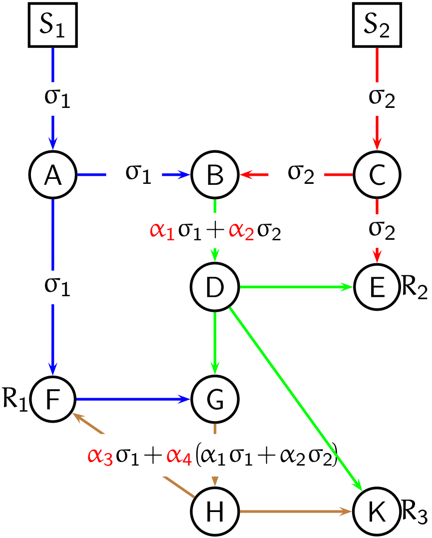
Network Multicast – Example



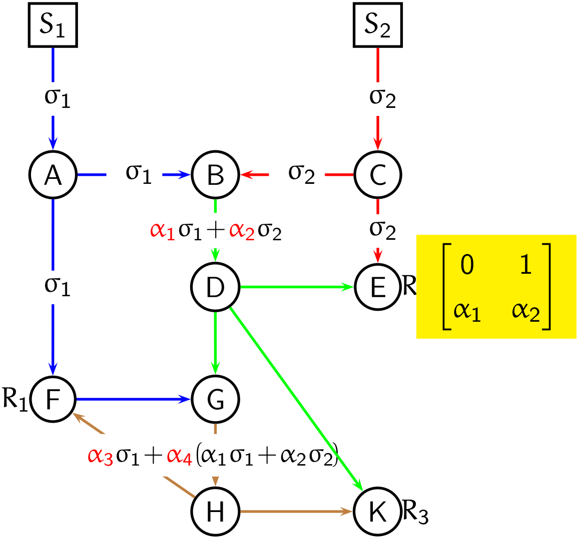
Network Multicast – Example



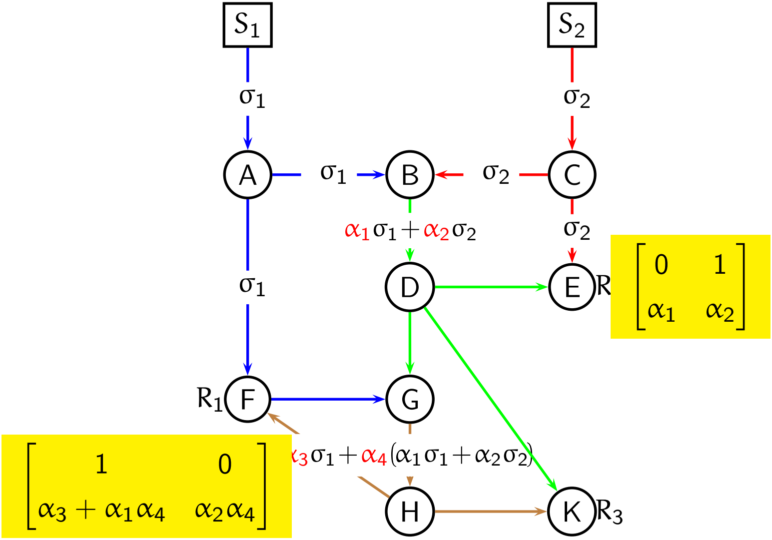
Network Multicas – Example



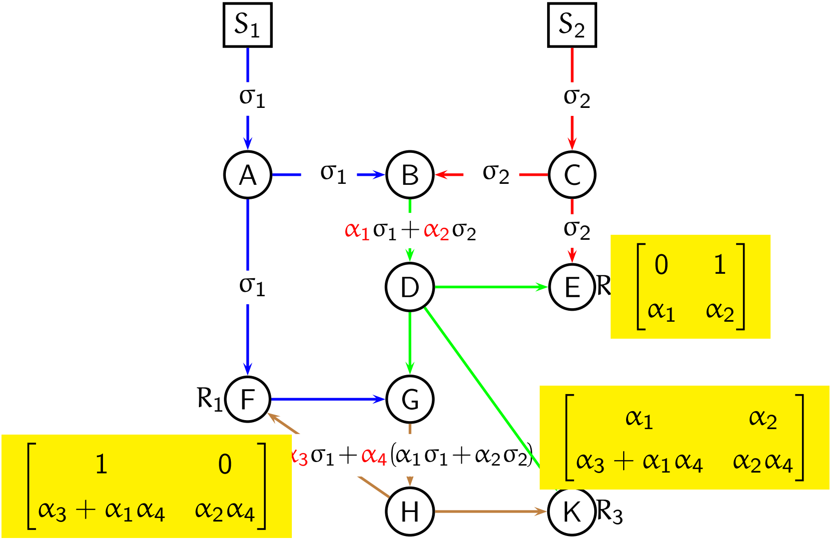
Network Multicas – Example



Network Multicas – Example



Network Multicas – Example



Network Multicast – Code Design

- Edges carry linear combinations of their parent node inputs; $\{\alpha_k\}$ are the coefficients used in these linear combinations.

Network Multicast – Code Design

- Edges carry linear combinations of their parent node inputs; $\{\alpha_k\}$ are the coefficients used in these linear combinations.
- ρ_i^j is the symbol on the last edge of the path (S_i, R_j) $\Rightarrow$ Receiver j has to solve the following system of equations:

$$\begin{bmatrix} \rho_1^j \\ \vdots \\ \rho_h^j \end{bmatrix} = \mathbf{C}_j \begin{bmatrix} \sigma_1 \\ \vdots \\ \sigma_h \end{bmatrix}$$

where the elements of matrix $\mathbf{C}_j$ are polynomials in $\{\alpha_k\}$.

Network Multicast – Code Design

- Edges carry linear combinations of their parent node inputs; $\{\alpha_k\}$ are the coefficients used in these linear combinations.
- ρ_i^j is the symbol on the last edge of the path $(S_i, R_j) \Rightarrow$ Receiver j has to solve the following system of equations:

$$\begin{bmatrix} \rho_1^j \\ \vdots \\ \rho_h^j \end{bmatrix} = \mathbf{C}_j \begin{bmatrix} \sigma_1 \\ \vdots \\ \sigma_h \end{bmatrix}$$

where the elements of matrix $\mathbf{C}_j$ are polynomials in $\{\alpha_k\}$.

The Code Design Problem:

Select $\{\alpha_k\}$ so that all matrices $\mathbf{C}_1 \ldots \mathbf{C}_N$ are full rank.

Network Multicast – Code Existence

- The goal is to select $\{\alpha_k\}$ so that $\mathbf{C}_1 \ldots \mathbf{C}_N$ are full rank.
- Equivalently, the goal is to select $\{\alpha_k\}$ so that

$$f(\{\alpha_k\}) \triangleq \det(\mathbf{C}_1) \cdots \det(\mathbf{C}_N) \neq 0.$$

Can such $\{\alpha_k\}$ be found?

Network Multicast – Code Existence

- The goal is to select $\{\alpha_k\}$ so that $\mathbf{C}_1 \ldots \mathbf{C}_N$ are full rank.
- Equivalently, the goal is to select $\{\alpha_k\}$ so that

$$f(\{\alpha_k\}) \triangleq \det(\mathbf{C}_1) \cdots \det(\mathbf{C}_N) \neq 0.$$

Can such $\{\alpha_k\}$ be found?

RLNC [Ho et al.]

Yes, by selecting $\{\alpha_k\}$ uniformly at random from a "large filed", we will have the polynomial $f(\{\alpha_k\}) \neq 0$ with "high probability".

Network Multicast – Code Existence

- The goal is to select $\{\alpha_k\}$ so that $\mathbf{C}_1 \ldots \mathbf{C}_N$ are full rank.
- Equivalently, the goal is to select $\{\alpha_k\}$ so that

$$f(\{\alpha_k\}) \triangleq \det(\mathbf{C}_1) \cdots \det(\mathbf{C}_N) \neq 0.$$

Can such $\{\alpha_k\}$ be found?

RLNC [Ho et al.]

Yes, by selecting $\{\alpha_k\}$ uniformly at random from a "large filed", we will have the polynomial $f(\{\alpha_k\}) \neq 0$ with "high probability".

LIF [Jaggi et al.]

Yes, $\{\alpha_k\}$ can be selected form $\mathbb{F}_q$ where $q > N$.

Network Multicast – Code Existence

- The goal is to select $\{\alpha_k\}$ so that $\mathbf{C}_1 \ldots \mathbf{C}_N$ are full rank.
- Equivalently, the goal is to select $\{\alpha_k\}$ so that

$$f(\{\alpha_k\}) \triangleq \det(\mathbf{C}_1) \cdots \det(\mathbf{C}_N) \neq 0.$$

Can such $\{\alpha_k\}$ be found?

RLNC [Ho et al.]

Yes, by selecting $\{\alpha_k\}$ uniformly at random from a "large filed", we will have the polynomial $f(\{\alpha_k\}) \neq 0$ with "high probability".

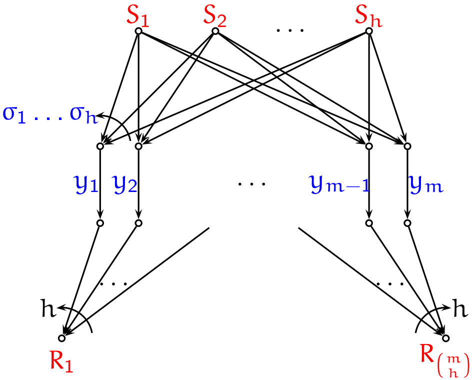
LIF [Jaggi et al.]

Yes, $\{\alpha_k\}$ can be selected form $\mathbb{F}_q$ where $q > N$.

But, we don't know of any networks for which $q > \mathcal{O}(\sqrt{N})$ is required.

Combination Network $B(h, m)$

A Popular Network With a Small-Alphabt Code



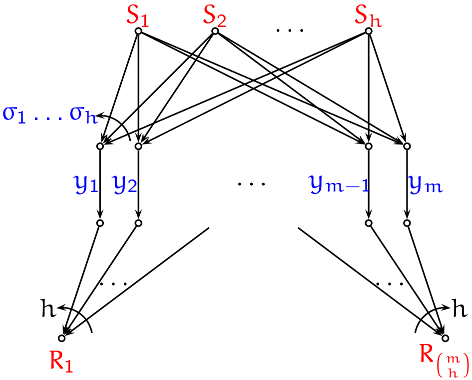
$B(h, m)$ has

- h information sources,
- $\binom{m}{h}$ receivers, and
- m bottlenecks.

Design a rate-h multicast!

Combination Network $B(h, m)$

A Popular Network With a Small-Alphabt Code



$B(h, m)$ has

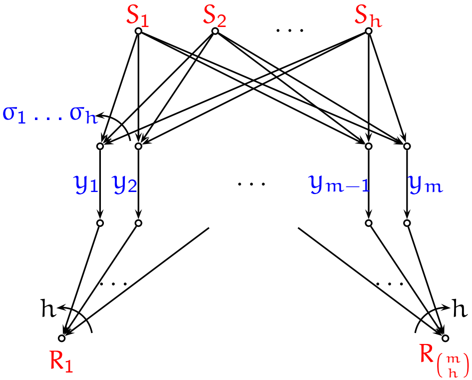
- h information sources,
- $\binom{m}{h}$ receivers, and
- m bottlenecks.

Design a rate-h multicast!

Map $\{\sigma_j\}$ to $\{y_k\}$ by an $[m, h]$ Reed-Solomon code.

Combination Network $B(h, m)$

A Popular Network With a Small-Alphabt Code



$B(h, m)$ has

- h information sources,
- $\binom{m}{h}$ receivers, and
- m bottlenecks.

Design a rate-h multicast!

Map $\{\sigma_j\}$ to $\{y_k\}$ by an $[m, h]$ Reed-Solomon code.

But, what if fewer than h sources are available at the bottlenecks?

Coding Points

The multicast condition:

Between the sources and each receiver node,

- the number of edges in the min-cut is h (or equivalently)
- there are h edge-disjoint paths (S_i, R_j) for $1 \leqslant i \leqslant h$.

Coding points are edges where paths from different sources merge.

Local and Global Coding Vectors

- Edges carry linear combinations of their parent node inputs.
- $\{\alpha_k\}$ are the local coding coefficients.
- Each edge e carries a linear combination of source symbols:

$$c_1(e)\sigma_1 + \cdots + c_h(e)\sigma_h = \left[c_1(e) \ldots c_h(e)\right] \begin{bmatrix} \sigma_1 \\ \vdots \\ \sigma_h \end{bmatrix}$$

- $[c_1(e) \ldots c_h(e)] \in \mathbb{F}_q^h$ is the global coding vector of edge e.

Decoding for Receiver j

- ρ_i^j is the symbol on the last edge on the path (S_i, R_j).
- $\mathbf{c}_i^j$ is the coding vector of the last edge on the path (S_i, R_j).
- $\mathbf{C}_j$ is the matrix whose i-th row is $\mathbf{c}_i^j$.
- Receiver j has to solve the following system of equations:

$$\begin{bmatrix} \rho_1^j \\ \vdots \\ \rho_h^j \end{bmatrix} = C_j \begin{bmatrix} \sigma_1 \\ \vdots \\ \sigma_h \end{bmatrix}.$$

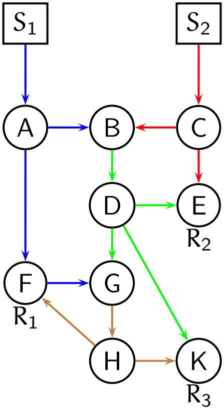
Network Multicast – Code Design

Select a coding vector for each edge e of the network so that

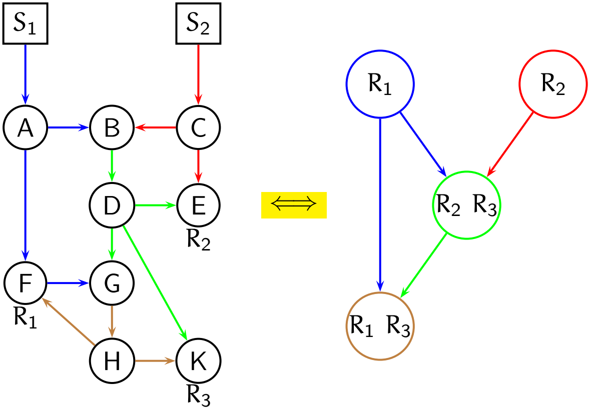
1. the matrices $\mathbf{C_1} \ldots \mathbf{C_N}$ are full rank.
2. the coding vector of e is in the linear span of the coding vectors of the input edges to the parent node of e.

The only edges of interest are coding points.

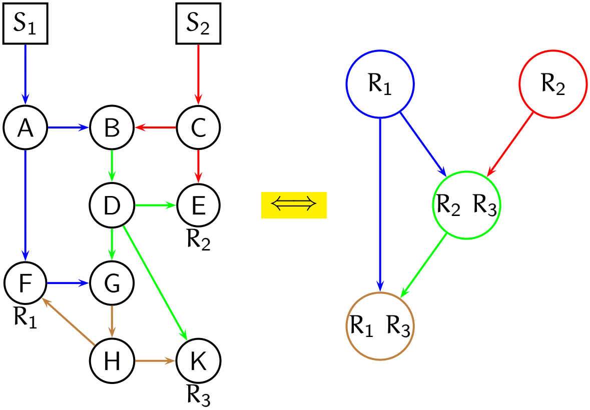
Local and Global View



Local and Global View



Local and Global View



Roughly speaking, we need to find a collection of vectors s.t.

some are in the span of others & some are linearly independent.

Minimal h-Multicast Graph $\Gamma = (G, \mathcal{S}, \mathcal{R})$

Ingredients:

1. Directed, acyclic graph G with
   - h source nodes $\mathcal{S} = S_1, \ldots, S_h$
   - nodes with in-degree d, $2 \leqslant d \leqslant h$.
2. Set of labels $\mathcal{R} = R_1, \ldots, R_N$ (receivers).

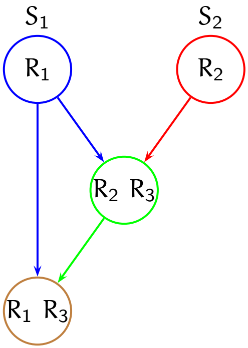
Multicast property (labeling rules):

1. Each R_i is used to label exactly h nodes. Nodes can have multiple labels.
2. Nodes labeled by R_i are connectible to the sources by h node-disjoint paths.

Minimality:

If an edge is removed, the multicast property is lost.

Example:



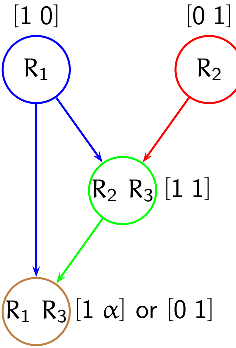
Code Design Problem for Network Multicast

Select a vector in $\mathbb{F}_q^h$ for each node in G s.t.

1. S_j is assigned e_j.
2. vectors of the h nodes sharing a receiver label are linearly independent
3. the vector assigned to a node is in the span of the vectors assigned to its parents.

We call such assignments network multicast codes.

Example:



Can such selection of vectors be made? Over how small field?

The Field Size?

Theorem [Fragouli & Soljanin '06]:

- For networks with 2 sources and N receivers,

$$q \geqslant \alpha = \lfloor \sqrt{2N - 7/4} + 1/2 \rfloor$$

  is sufficient, and, for some networks, necessary.

- For networks with h sources and N receivers,

$$q \geqslant \alpha = N$$

  is sufficient. (Proven even earlier a couple of times.)

We don't have any examples where we need $\alpha > \mathcal{O}(\sqrt{N})$.

Coding for Networks with Two Sources

- Let $\mathcal{L}$ be the following set of $(q+1)$ vectors:

  $$[0\,1],\ [1\,0],\ \text{and } [1\,\alpha^i] \text{ for } 0 \leqslant i \leqslant q-2,$$

  where α is a primitive element of $\mathbb{F}_q$.

Coding for Networks with Two Sources

- Let $\mathcal{L}$ be the following set of $(q+1)$ vectors:

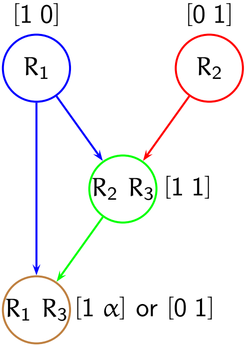
  $$[0\,1],\ [1\,0],\ \text{and } [1\,\alpha^i] \text{ for } 0 \leqslant i \leqslant q-2,$$

  where α is a primitive element of $\mathbb{F}_q$.

- Consider any two different vectors in $\mathcal{L}$:
  - they are linearly independent, and
  - any vector in $\mathcal{L}$ is in their linear span.

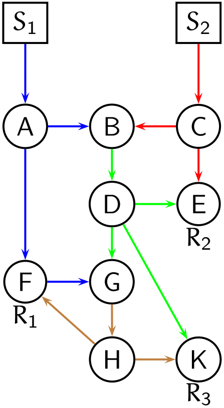
$\Longrightarrow$ Vectors in $\mathcal{L}$ can be treated as colors.

Example:

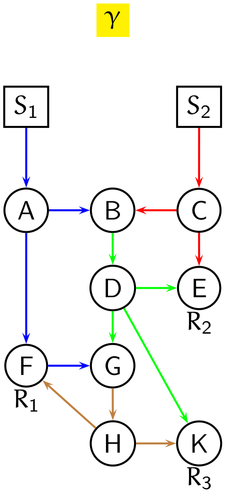


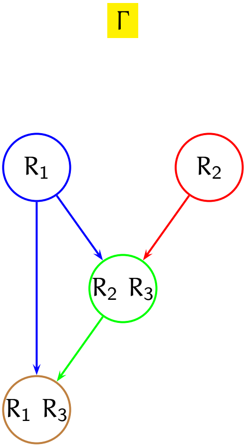
Vertex Coloring and Code Design

γ

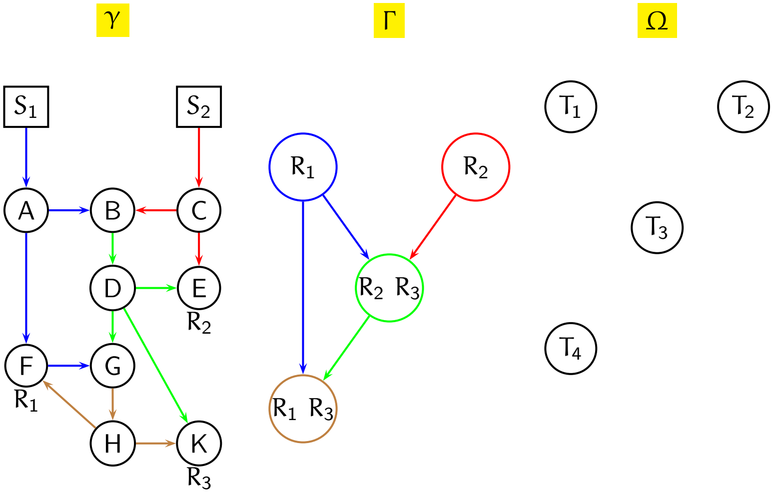


Vertex Coloring and Code Design

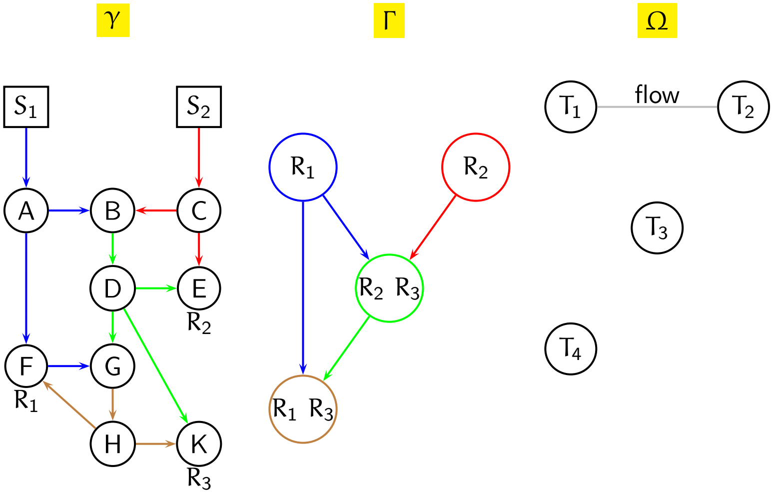




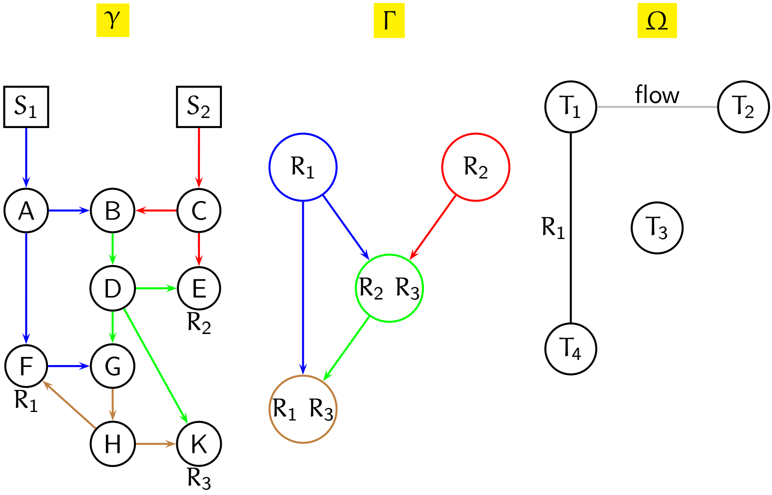
Vertex Coloring and Code Design



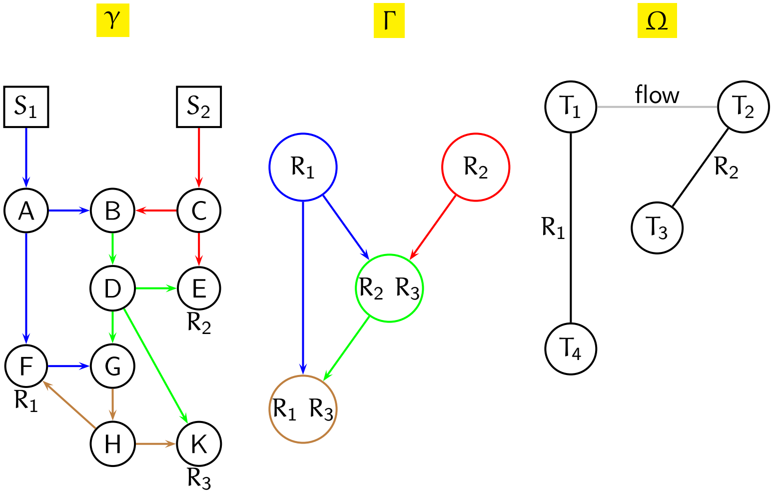
Vertex Coloring and Code Design



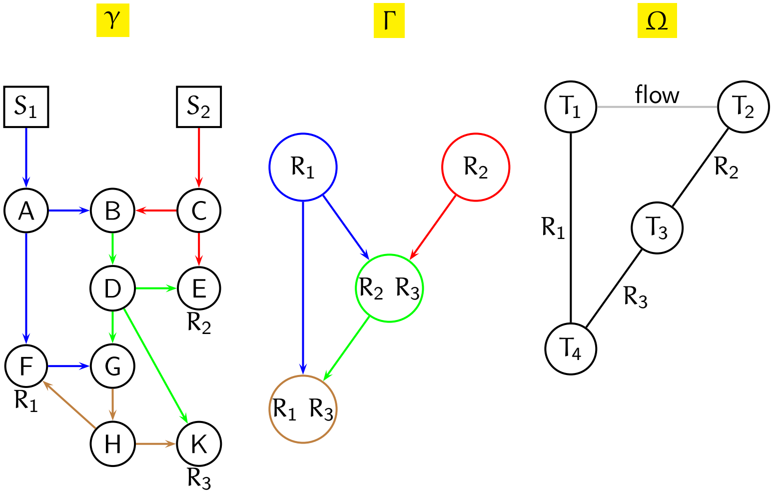
Vertex Coloring and Code Design



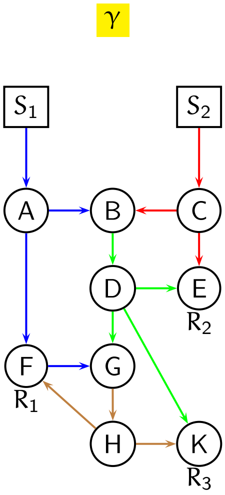
Vertex Coloring and Code Design

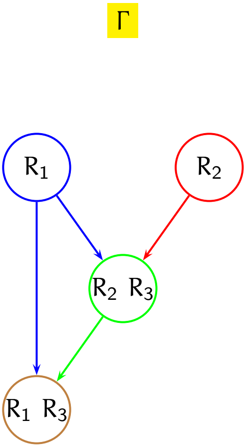


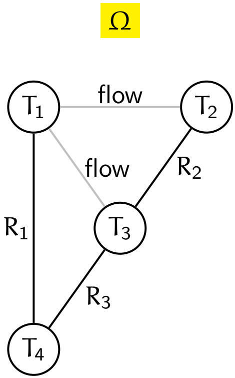
Vertex Coloring and Code Design



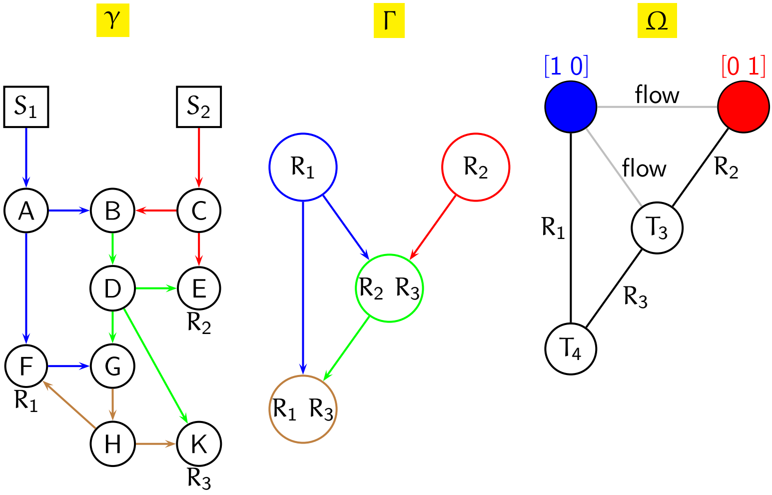
Vertex Coloring and Code Design



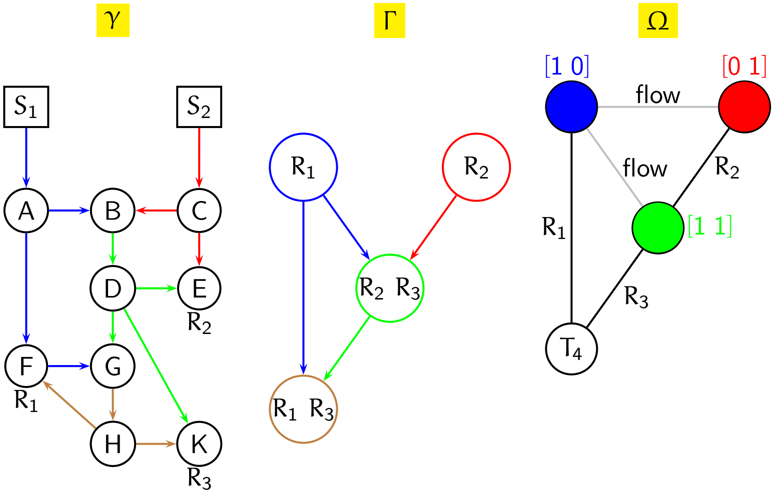




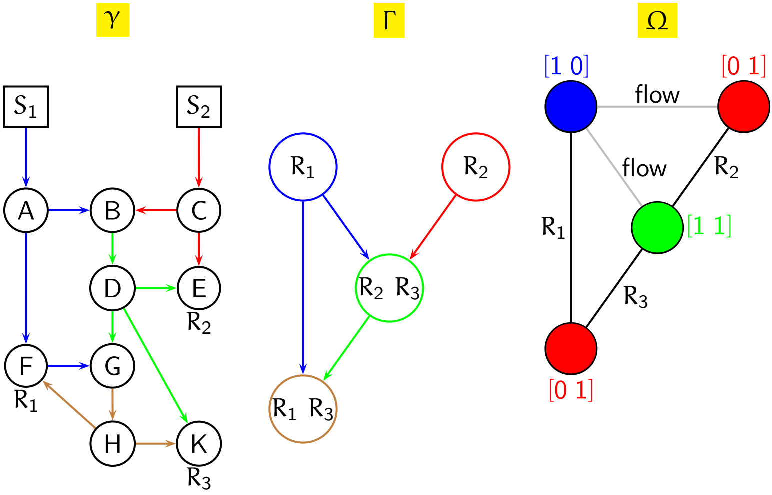
Vertex Coloring and Code Design



Vertex Coloring and Code Design



Vertex Coloring and Code Design



Field Size for Network with Two Sources

ℓ -The Chromatic Number of Ω

Claim: $\ell \leqslant \sqrt{2\mathrm{N} - 7/4} + 1/2\rfloor + 1$

Elements of the Proof:

- Lemma: Every vertex in an Ω has degree at least two.

Field Size for Network with Two Sources

ℓ -The Chromatic Number of Ω

Claim: $\ell \leqslant \sqrt{2N - 7/4} + 1/2\rfloor + 1$

Elements of the Proof:

- Lemma: Every vertex in an Ω has degree at least two.
- Lemma: Every ℓ-chromatic graph has at least ℓ vertices of degree at least $\ell - 1$.

Field Size for Network with Two Sources

ℓ -The Chromatic Number of Ω

Claim: $\ell \leqslant \sqrt{2N - 7/4} + 1/2\rfloor + 1$

Elements of the Proof:

- Lemma: Every vertex in an Ω has degree at least two.
- Lemma: Every ℓ-chromatic graph has at least ℓ vertices of degree at least $\ell - 1$.
- For an Ω with n nodes, chromatic number ℓ, and ϵ edges:
  1. $\epsilon \geqslant [\ell(\ell-1) + (n-\ell)2]/2$ ← from the lemmas
  2. $\epsilon \leqslant N + n - 2$ ← receiver and flow edges

Recall that $\mathbb{F}_q$ provides $q + 1$ colors when $h = 2$.

$h > 2$

We cannot dispose of geometry and just do combinatorics

Is there generalization of the coloring idea?

- We have used points on the projective line as colors.
- Con we use the points on arcs in $\mathbb{PG}(h-1, q)$ as colors?

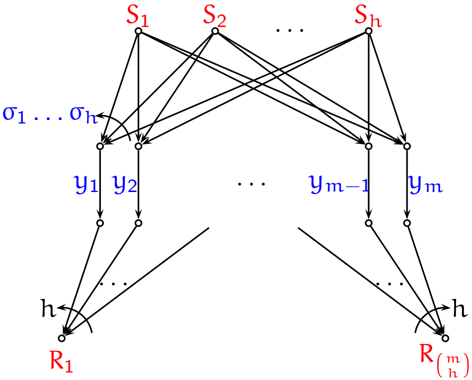
Yes, if each non-source node has h inputs.

Roughly speaking, we need to find a collection of vectors s.t.
some are in the span of others & some are linearly independent.

Are there counterparts to the "coloring graph" Ω?
E.g., matroids, finite geometry relations?

Combination Network $B(h, m)$

A Popular Network With a Small-Alphabt Code



$B(h, m)$ has

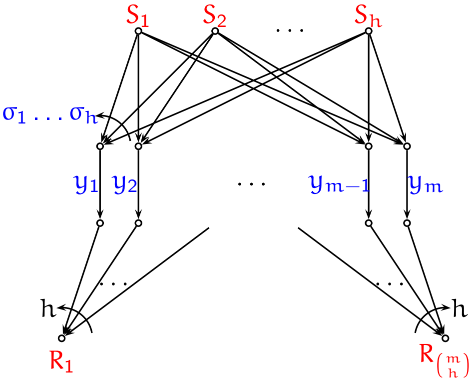
- h information sources,
- $\binom{m}{h}$ receivers, and
- m bottlenecks.

Design a rate-h multicast!

Map $\{\sigma_j\}$ to $\{y_k\}$ by an $[m, h]$ Reed-Solomon code.

Combination Network $B(h, m)$

A Popular Network With a Small-Alphabt Code



$B(h, m)$ has

- h information sources,
- $\binom{m}{h}$ receivers, and
- m bottlenecks.

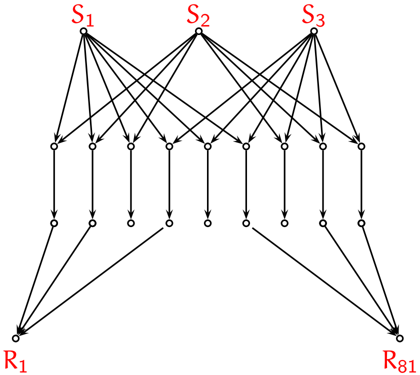
Design a rate-h multicast!

Map $\{\sigma_j\}$ to $\{y_k\}$ by an $[m, h]$ Reed-Solomon code.

But, what if fewer than h sources are available at the bottlenecks?

A Distributed Combination Network

Fewer than h sources are available at the bottlenecks



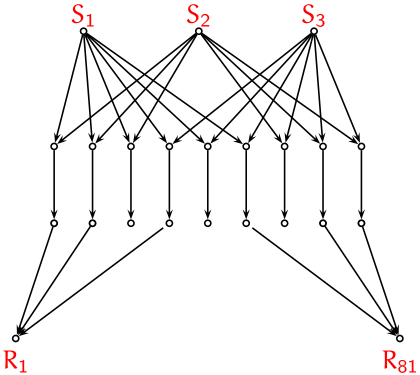
There are

- 3 information sources,
- 9 bottlenecks, and
- $\binom{9}{3} - 3 = 81$ receivers.

Design a rate-3 multicast!

A Distributed Combination Network

Fewer than h sources are available at the bottlenecks



There are

- 3 information sources,
- 9 bottlenecks, and
- $\binom{9}{3} - 3 = 81$ receivers.

Design a rate-3 multicast!

Only information that is locally available can be combined.

Non-Monotonicity

There may be a solution over $\mathbb{F}_{q_0}$ but not over $\mathbb{F}_q$ for some $q > 0$

Coding vectors for our example network:

$$\left[\begin{array}{ccc|ccc|ccc} a_1 & a_2 & a_3 & b_1 & b_2 & b_3 & 0 & 0 & 0 \\ c_1 & c_2 & c_3 & 0 & 0 & 0 & d_1 & d_2 & d_3 \\ \underbrace{0 \quad 0 \quad 0}_{v_1} & & & \underbrace{e_1 \quad e_2 \quad e_3}_{v_2} & & & \underbrace{f_1 \quad f_2 \quad f_3}_{v_3} & & \end{array}\right]$$

All 3×3 sub-matrices, except v_1, v_2, v_3, should be non-singular.

Non-Monotonicity

There may be a solution over $\mathbb{F}_{q_0}$ but not over $\mathbb{F}_q$ for some $q > 0$

Coding vectors for our example network:

$$\left[\begin{array}{ccc|ccc|ccc} a_1 & a_2 & a_3 & b_1 & b_2 & b_3 & 0 & 0 & 0 \\ c_1 & c_2 & c_3 & 0 & 0 & 0 & d_1 & d_2 & d_3 \\ \underbrace{0 \quad 0 \quad 0}_{v_1} & & & \underbrace{e_1 \quad e_2 \quad e_3}_{v_2} & & & \underbrace{f_1 \quad f_2 \quad f_3}_{v_3} & & \end{array}\right]$$

All 3×3 sub-matrices, except v_1, v_2, v_3, should be non-singular.

In which fields $\mathbb{F}_q$ does a solution exist?

- **No** solution exists when $q < 7$.
- A solution exists for all $q \geqslant 9$.
- A solution exists for $q = 7$
- **No** solution exists for $q = 8$.

What Would We Like To Do?

... short of solving the problem ...

Find relations (equivalences) with other problems, e.g.,

What Would We Like To Do?

... short of solving the problem ...

Find relations (equivalences) with other problems, e.g.,

Something old :

Three problems of Segre in $\mathbb{PG}(h-1, q)$

1. What is the size $g(h, q)$ of the maximal arc, and which arcs have $g(h, q)$ points?
2. For which q and $h < q$ are all arcs with $q + 1$ points equivalent?
3. What are the sizes of the complete arcs, and what is the size of the second largest complete arc?

Something new :

constrained MDS codes, codes with locality constraints, minimal multicast graph topologies vs. geometry of arcs.

Who are We?

From left to right: Fragouli, Valdez, Manganiello, Halbawi, Soljanin, Anderson, Walker, Kaplan