Distributed Reed-Solomon Codes

Farzad Parvaresh

f.parvaresh@eng.ui.ac.ir

University of Isfahan

Institute for Network Coding
CUHK, Hong Kong

August 2016

Research interests

- List-decoding of algebraic codes
  - Construction of efficient list-decodable codes over GF(2)
  - Efficient List-decoding of RS codes beyond the Johnson bound

Research interests

- List-decoding of algebraic codes
  - Construction of efficient list-decodable codes over GF(2)
  - Efficient List-decoding of RS codes beyond the Johnson bound
- Applications of compressed sensing
  - Phase retrieval problem

Research interests

- List-decoding of algebraic codes
  - Construction of efficient list-decodable codes over GF(2)
  - Efficient List-decoding of RS codes beyond the Johnson bound
- Applications of compressed sensing
  - Phase retrieval problem
- Power optimization over relay network
  - Computing the cut-set bound of a relay network efficiently
  - Computing diversity multiplexing tradeoff of generalized half-duplex relay networks

Reed-Solomon codes

Encoding of *RS(n,k,d)*:

Information symbols: $(u_1, u_2, \ldots, u_k) \in \mathbb{F}_q^k$

$$\Downarrow$$

$$f(X) = u_1 + u_2 X + \cdots + u_k X^{k-1} \in \mathbb{F}_q[X]$$

$$\Downarrow$$

Polynomial evaluation: $f(\alpha_1), f(\alpha_2), \ldots, f(\alpha_n) \quad (\alpha_1, \ldots, \alpha_n) \in \mathbb{F}_q^n$

$$\Downarrow$$

Codeword: $(c_1, c_2, \ldots, c_n) \in \mathbb{F}_q^n$

Reed-Solomon codes

Encoding of *RS(n,k,d)*:

Information symbols: $(u_1, u_2, \dots, u_k) \in \mathbb{F}_q^k$

$$\Downarrow$$

$$f(X) = u_1 + u_2 X + \cdots + u_k X^{k-1} \in \mathbb{F}_q[X]$$

$$\Downarrow$$

Polynomial evaluation: $f(\alpha_1), f(\alpha_2), \dots, f(\alpha_n)$ $\quad (\alpha_1, \dots, \alpha_n) \in \mathbb{F}_q^n$

$$\Downarrow$$

Codeword: $(c_1, c_2, \dots, c_n) \in \mathbb{F}_q^n$

Equivalently:

$$(u_1, u_2, \dots, u_k) \begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ \alpha_1^{k-1} & \cdots & \alpha_n^{k-1} \end{bmatrix} = (c_1, c_2, \dots, c_n)$$

Reed-Solomon codes

Encoding of *RS(n,k,d)*:

Information symbols: $(u_1, u_2, \dots, u_k) \in \mathbb{F}_q^k$

$$\Downarrow$$

$$f(X) = u_1 + u_2 X + \cdots + u_k X^{k-1} \in \mathbb{F}_q[X]$$

$$\Downarrow$$

Polynomial evaluation: $f(\alpha_1), f(\alpha_2), \dots, f(\alpha_n)$, $(\alpha_1, \dots, \alpha_n) \in \mathbb{F}_q^n$

$$\Downarrow$$

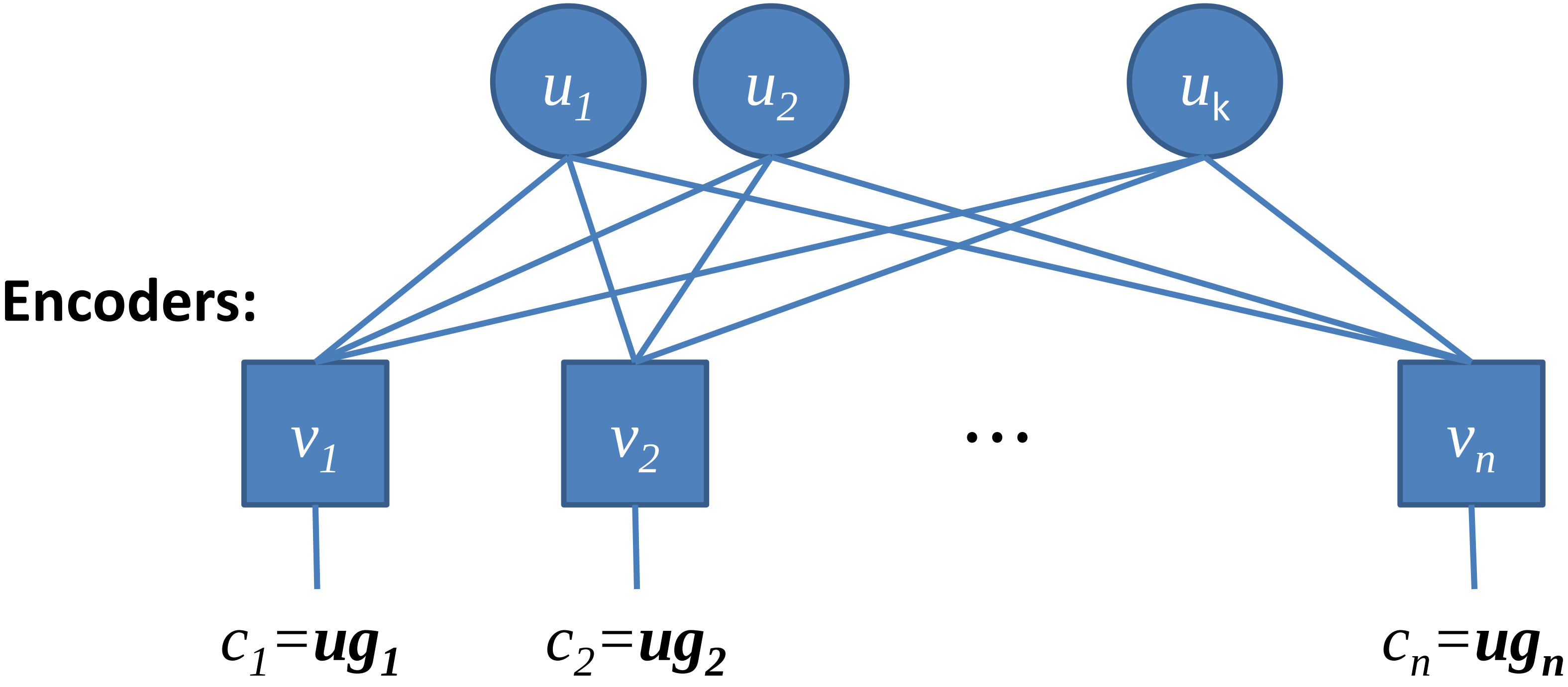
Codeword: $(c_1, c_2, \dots, c_n$

Generator matrix, G_{RS}, of RS code.

Equivalently:

$$(u_1, u_2, \dots, u_k) \boxed{\begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ \alpha_1^{k-1} & \cdots & \alpha_n^{k-1} \end{bmatrix}} = (c_1, c_2, \dots, c_n)$$

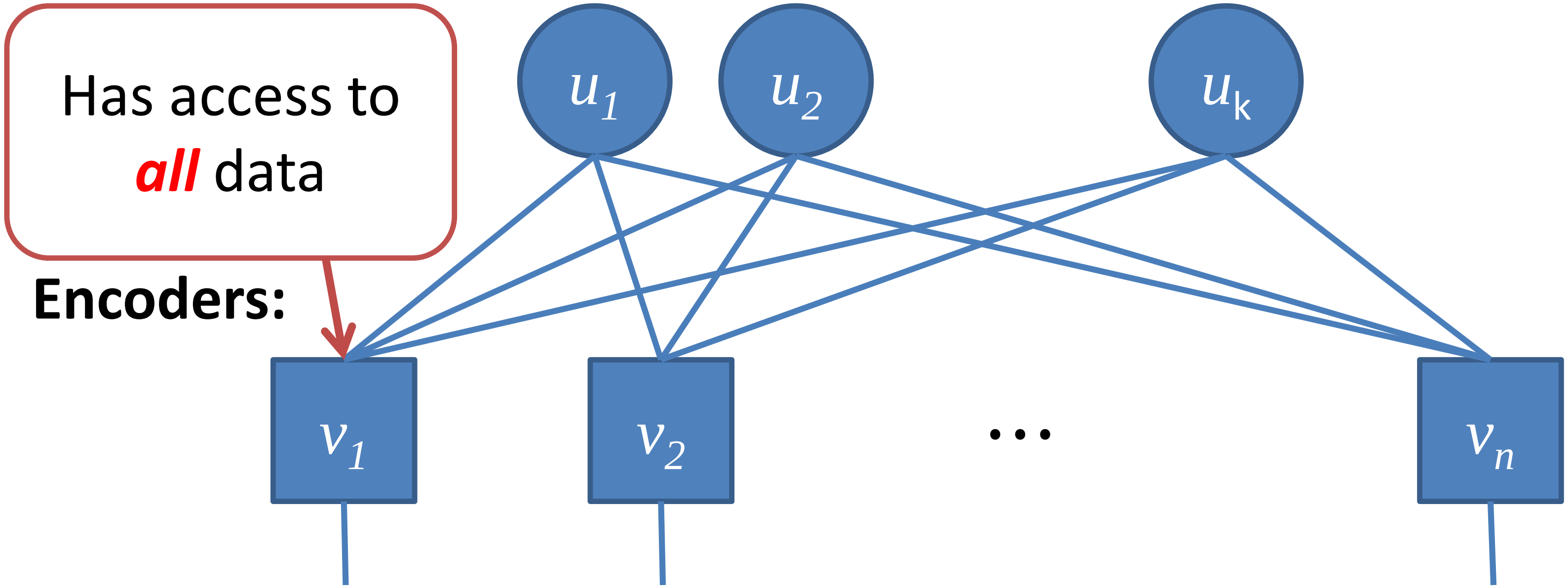
Reed-Solomon Encoding



$$G_{RS} = \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ g_1 & g_2 & \cdots & g_n \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}$$

G_{RS} is usually a **Vandermonde** matrix

Reed-Solomon Encoding



Has access to ***all*** data

Encoders:

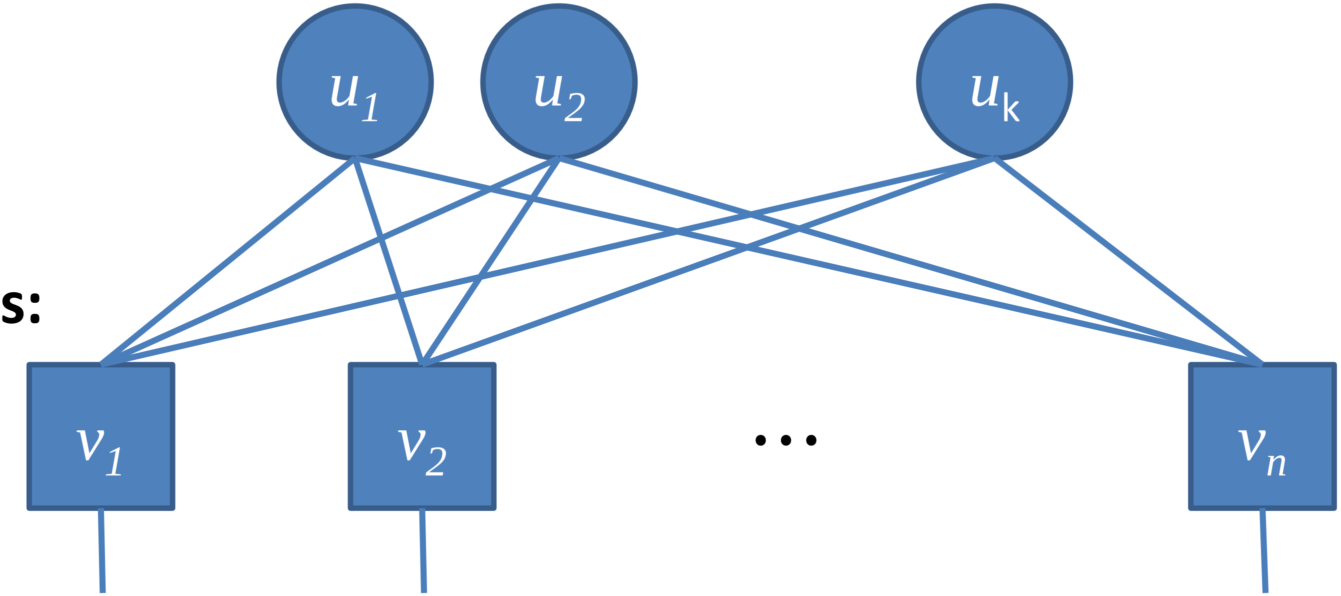
$c_1 = \boldsymbol{ug_1}$ $c_2 = \boldsymbol{ug_2}$ $c_n = \boldsymbol{ug_n}$

$$G_{RS} = \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ g_1 & g_2 & \cdots & g_n \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}$$

G_{RS} is usually a **Vandermonde** matrix

Distributed RS Encoding

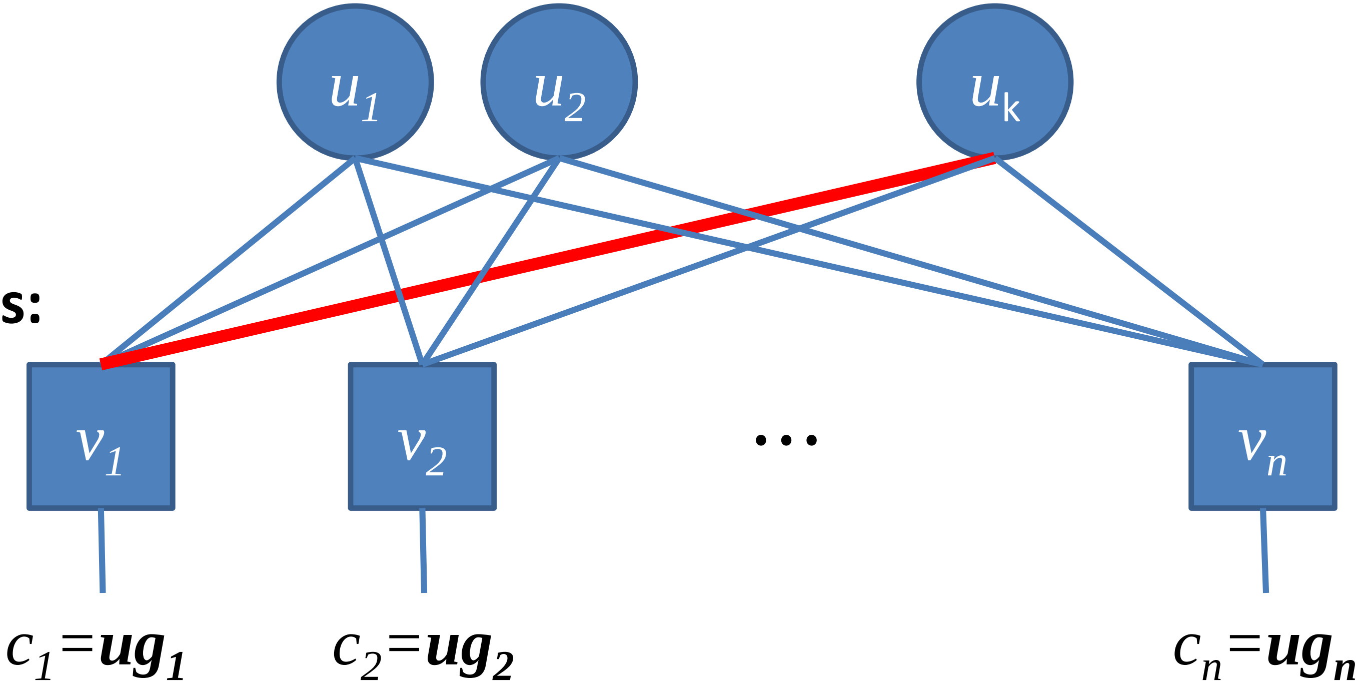
Encoders:



$c_1 = \boldsymbol{u}\boldsymbol{g}_1$ $c_2 = \boldsymbol{u}\boldsymbol{g}_2$ $c_n = \boldsymbol{u}\boldsymbol{g}_n$

$$G_{RS} = \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ g_1 & g_2 & \cdots & g_n \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}$$

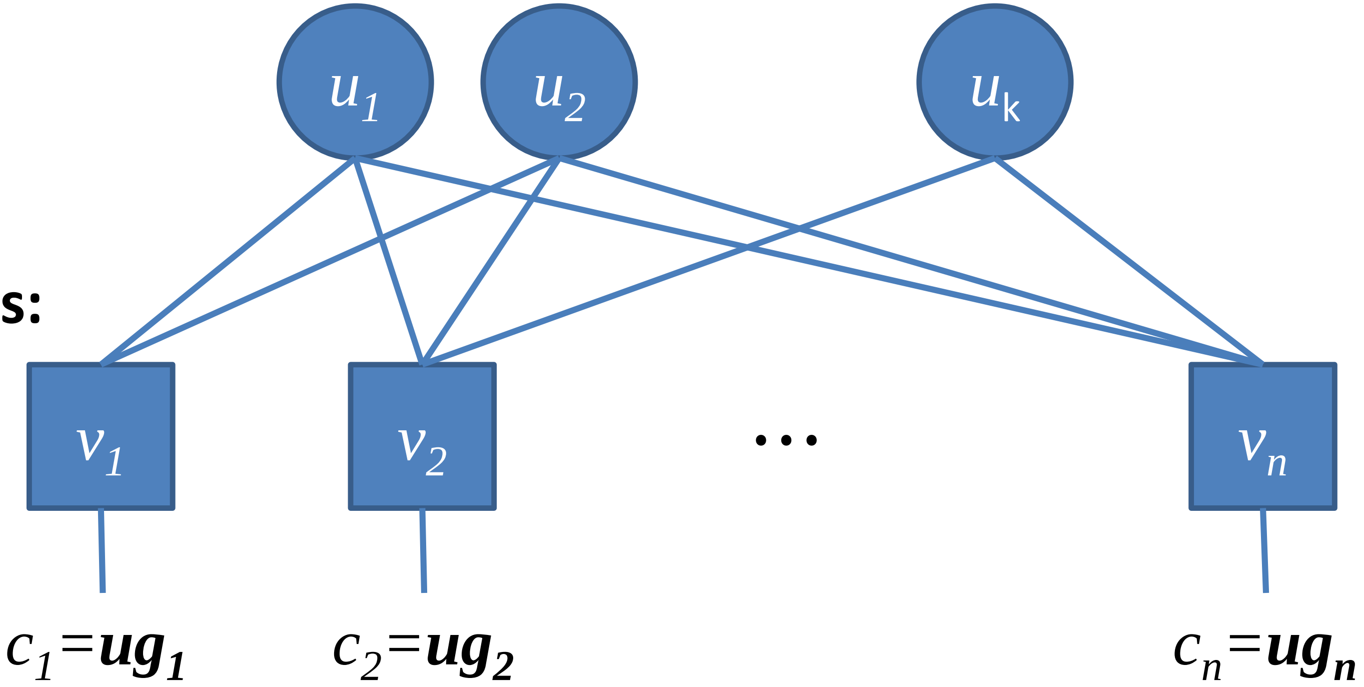
Distributed RS Encoding


u_1
u_2
u_k
Encoders:
v_1
v_2
…
v_n
$c_1 = \boldsymbol{ug_1}$
$c_2 = \boldsymbol{ug_2}$
$c_n = \boldsymbol{ug_n}$

$$G_{RS} = \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ g_1 & g_2 & \cdots & g_n \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}$$

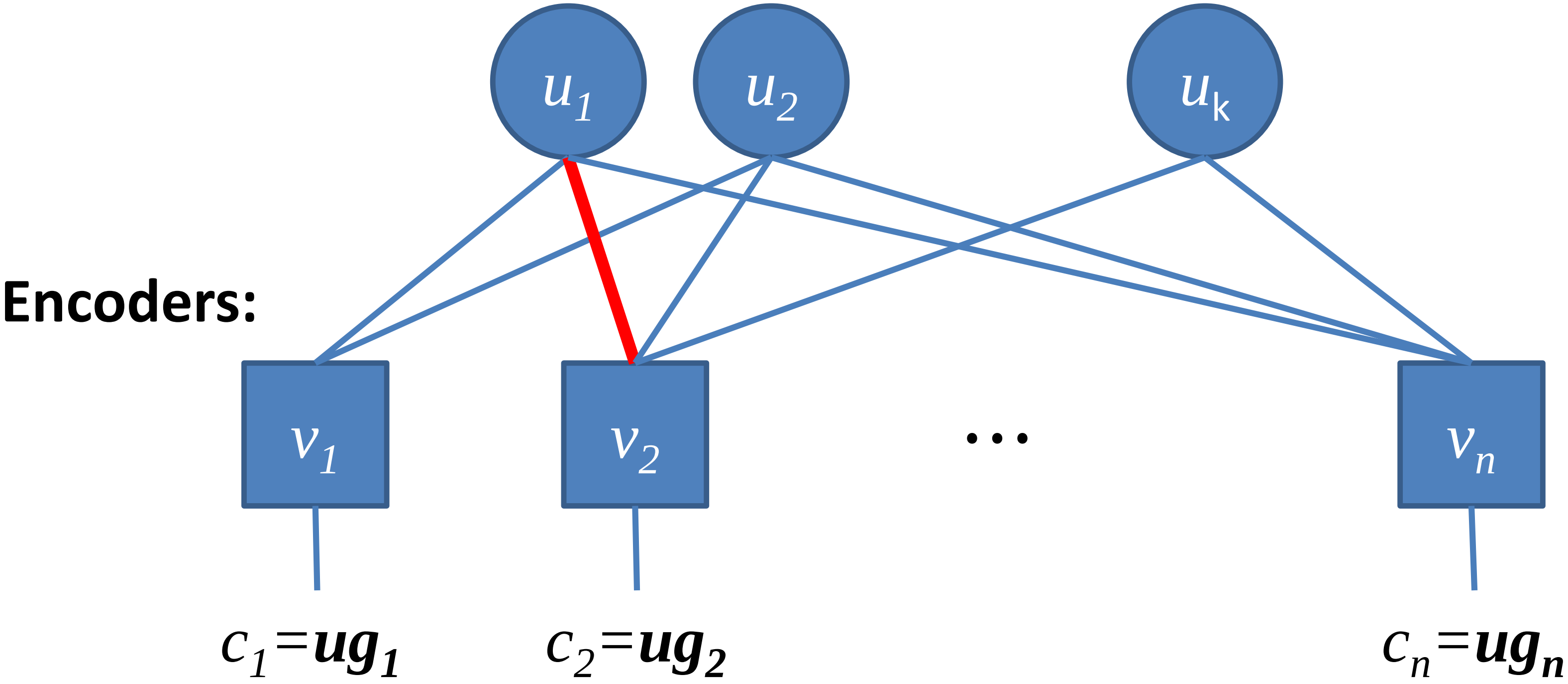
Distributed RS Encoding

Encoders:


u_1
u_2
u_k
v_1
v_2
…
v_n
$c_1 = \boldsymbol{ug_1}$
$c_2 = \boldsymbol{ug_2}$
$c_n = \boldsymbol{ug_n}$

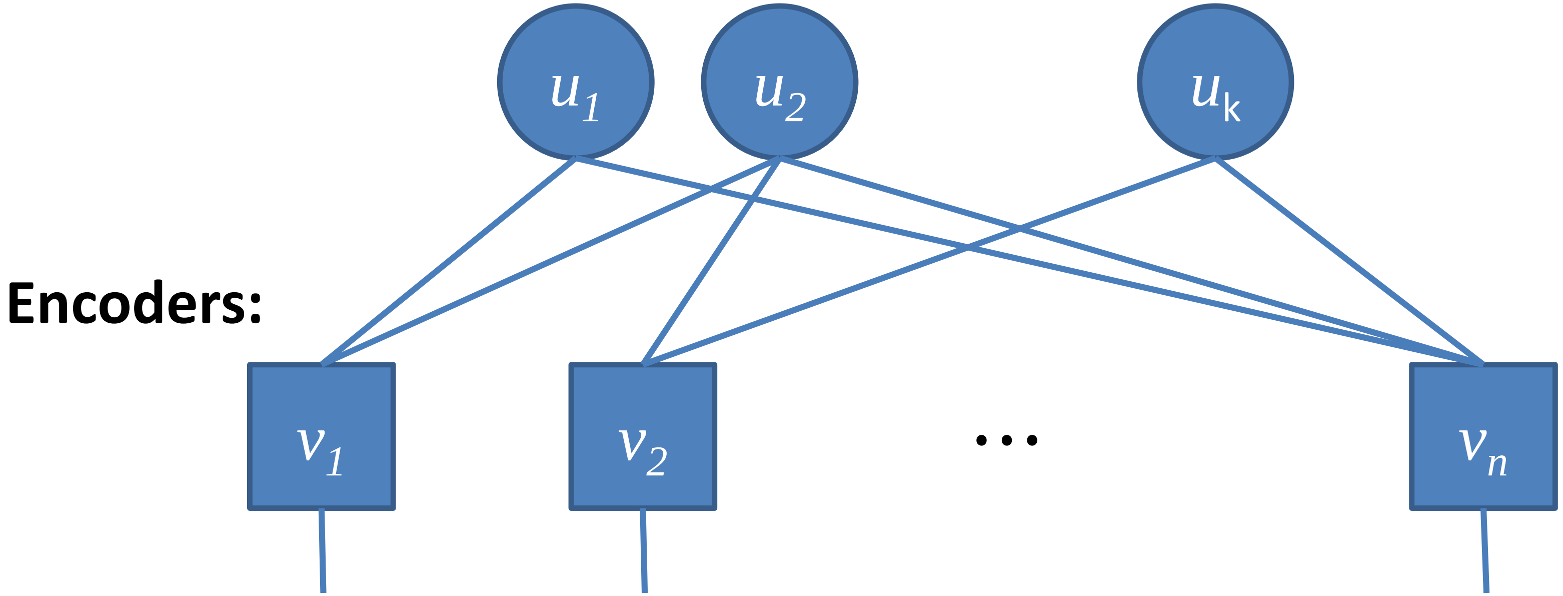
$$G_{RS} = \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ g_1 & g_2 & \cdots & g_n \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}$$

Distributed RS Encoding


u_1
u_2
u_k
Encoders:
v_1
v_2
…
v_n
$c_1 = \boldsymbol{ug_1}$
$c_2 = \boldsymbol{ug_2}$
$c_n = \boldsymbol{ug_n}$

$$G_{RS} = \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ g_1 & g_2 & \cdots & g_n \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}$$

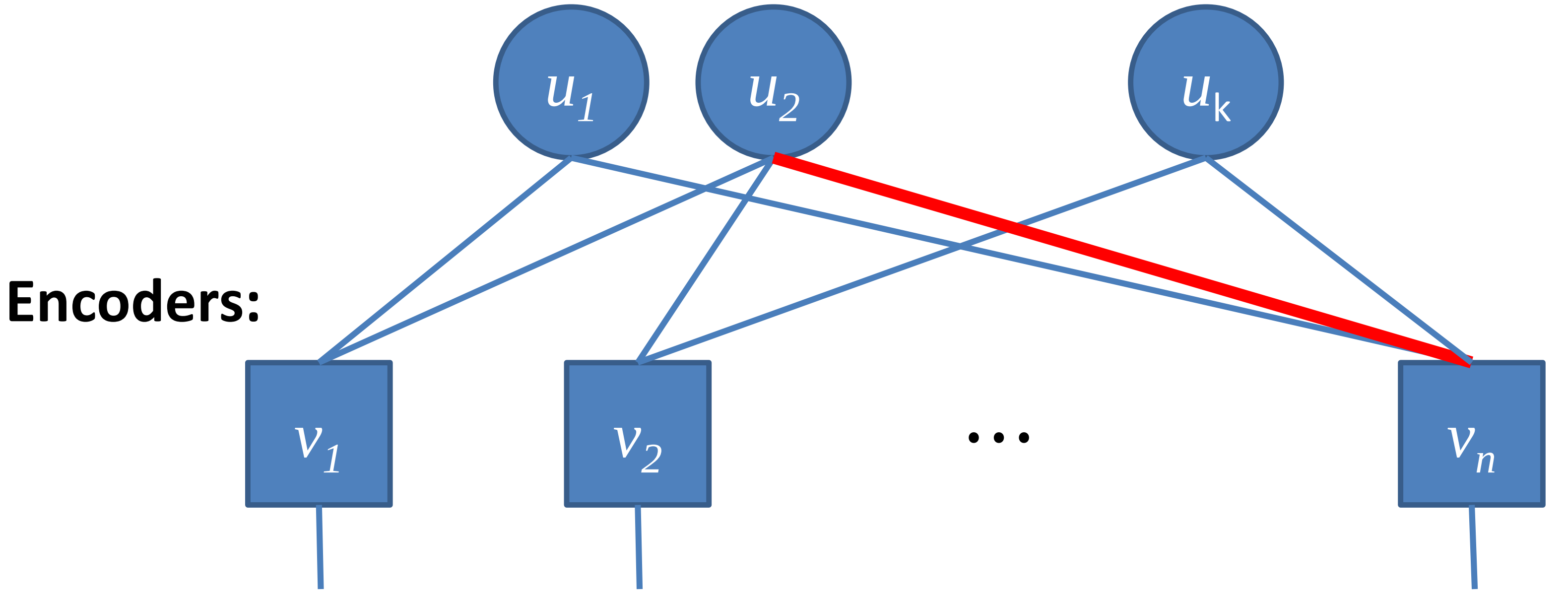
Distributed RS Encoding



$c_1 = \boldsymbol{ug_1}$ $c_2 = \boldsymbol{ug_2}$ $c_n = \boldsymbol{ug_n}$

$$G_{RS} = \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ g_1 & g_2 & \cdots & g_n \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}$$

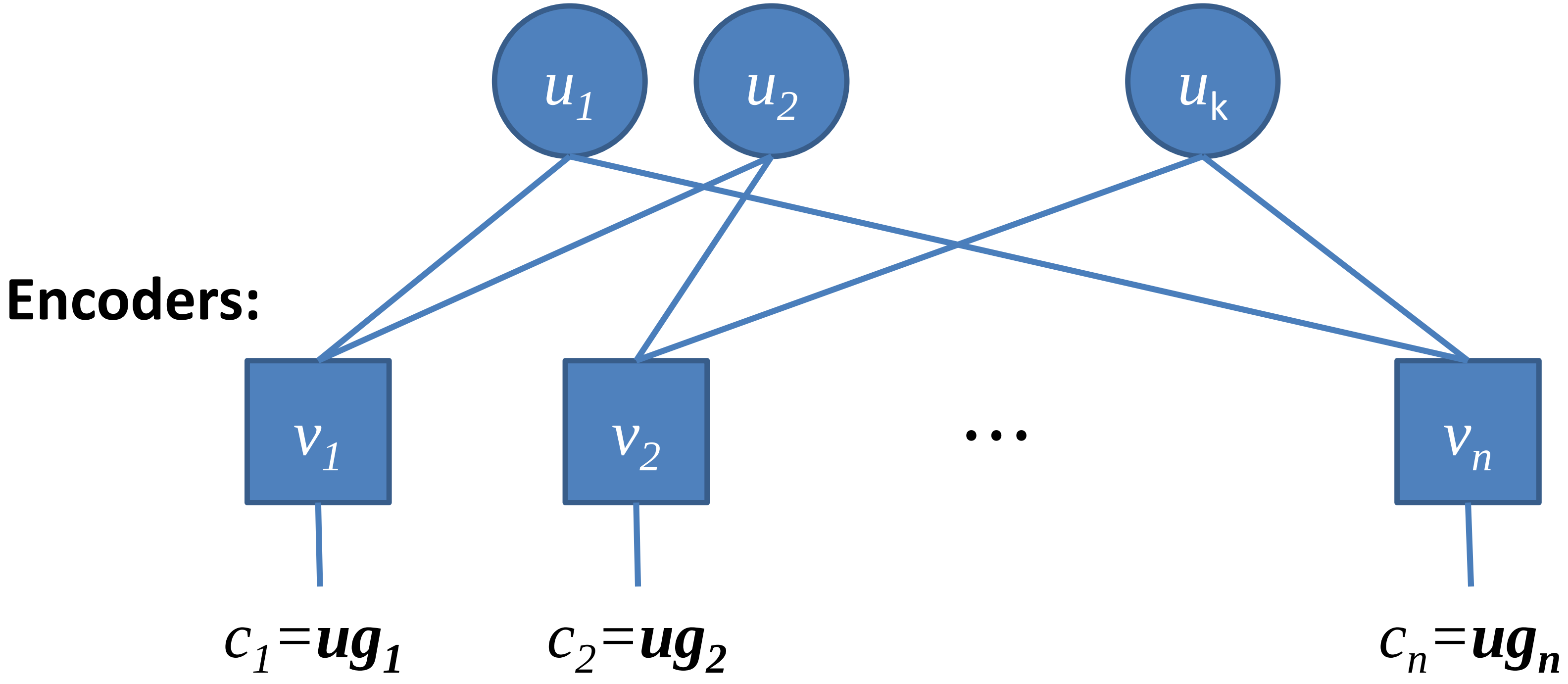
Distributed RS Encoding



$c_1 = \boldsymbol{ug_1}$ $c_2 = \boldsymbol{ug_2}$ $c_n = \boldsymbol{ug_n}$

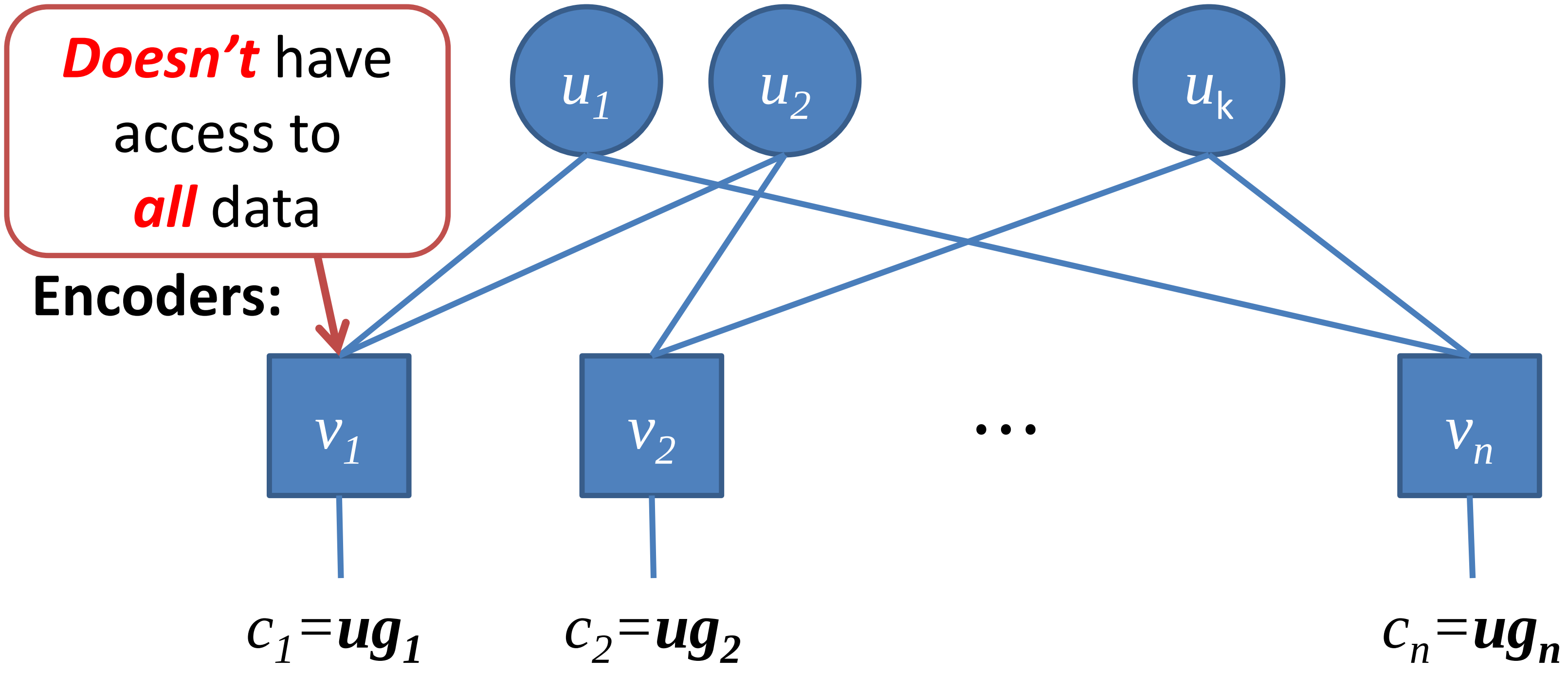
$$G_{RS} = \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ g_1 & g_2 & \cdots & g_n \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}$$

Distributed RS Encoding


u_1
u_2
u_k
Encoders:
v_1
v_2
…
v_n
$c_1 = \boldsymbol{ug_1}$
$c_2 = \boldsymbol{ug_2}$
$c_n = \boldsymbol{ug_n}$

$$G_{RS} = \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ g_1 & g_2 & \cdots & g_n \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}$$

Distributed RS Encoding



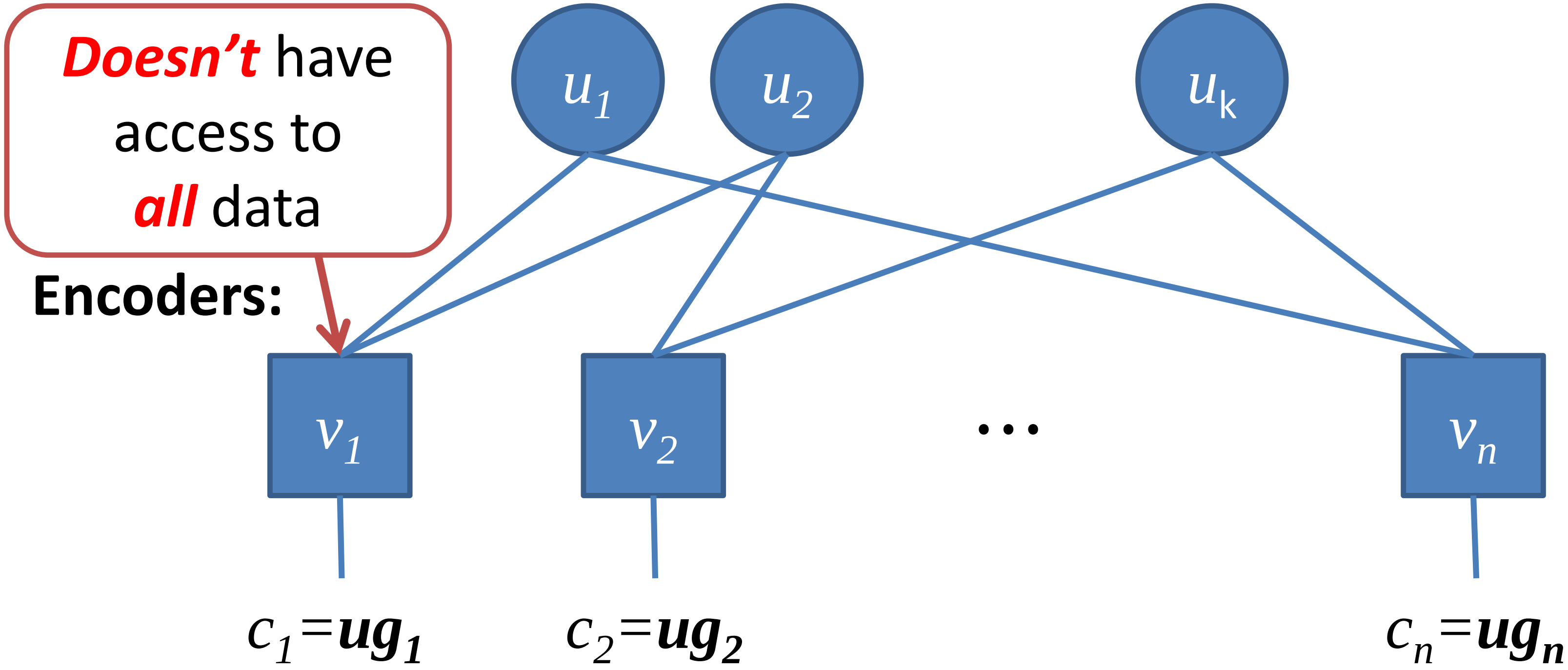
Doesn't have access to ***all*** data

Encoders:

$c_1 = \boldsymbol{u}\boldsymbol{g}_1$ $c_2 = \boldsymbol{u}\boldsymbol{g}_2$ $c_n = \boldsymbol{u}\boldsymbol{g}_n$

$$G_{RS} = \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ g_1 & g_2 & \cdots & g_n \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}$$

Distributed RS Encoding



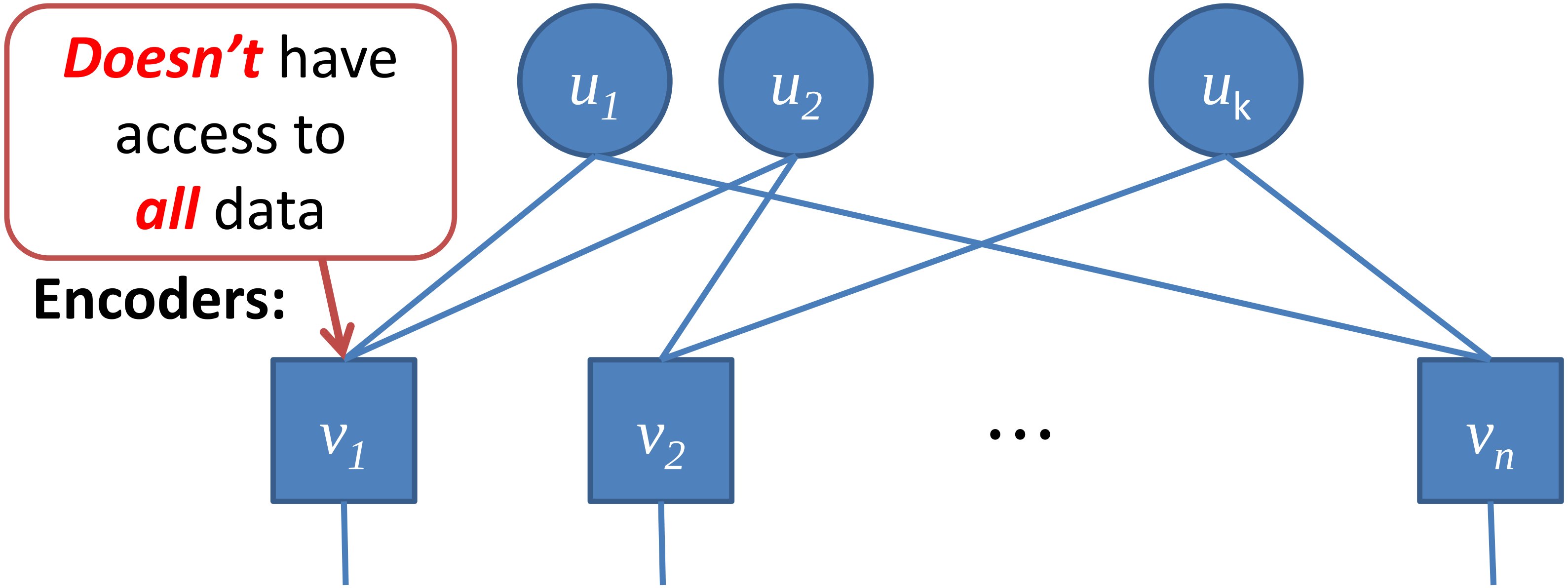
Doesn't have access to ***all*** data

Encoders:

$c_1 = \boldsymbol{u}\boldsymbol{g}_1$ $c_2 = \boldsymbol{u}\boldsymbol{g}_2$ $c_n = \boldsymbol{u}\boldsymbol{g}_n$

$$G_{RS} = \begin{bmatrix} \vdots & 0 & & \vdots \\ g_1 & \vdots & & 0 \\ \vdots & g_2 & \cdots & \vdots \\ 0 & \vdots & & g_n \end{bmatrix}$$

Distributed RS Encoding



Doesn't have access to ***all*** data

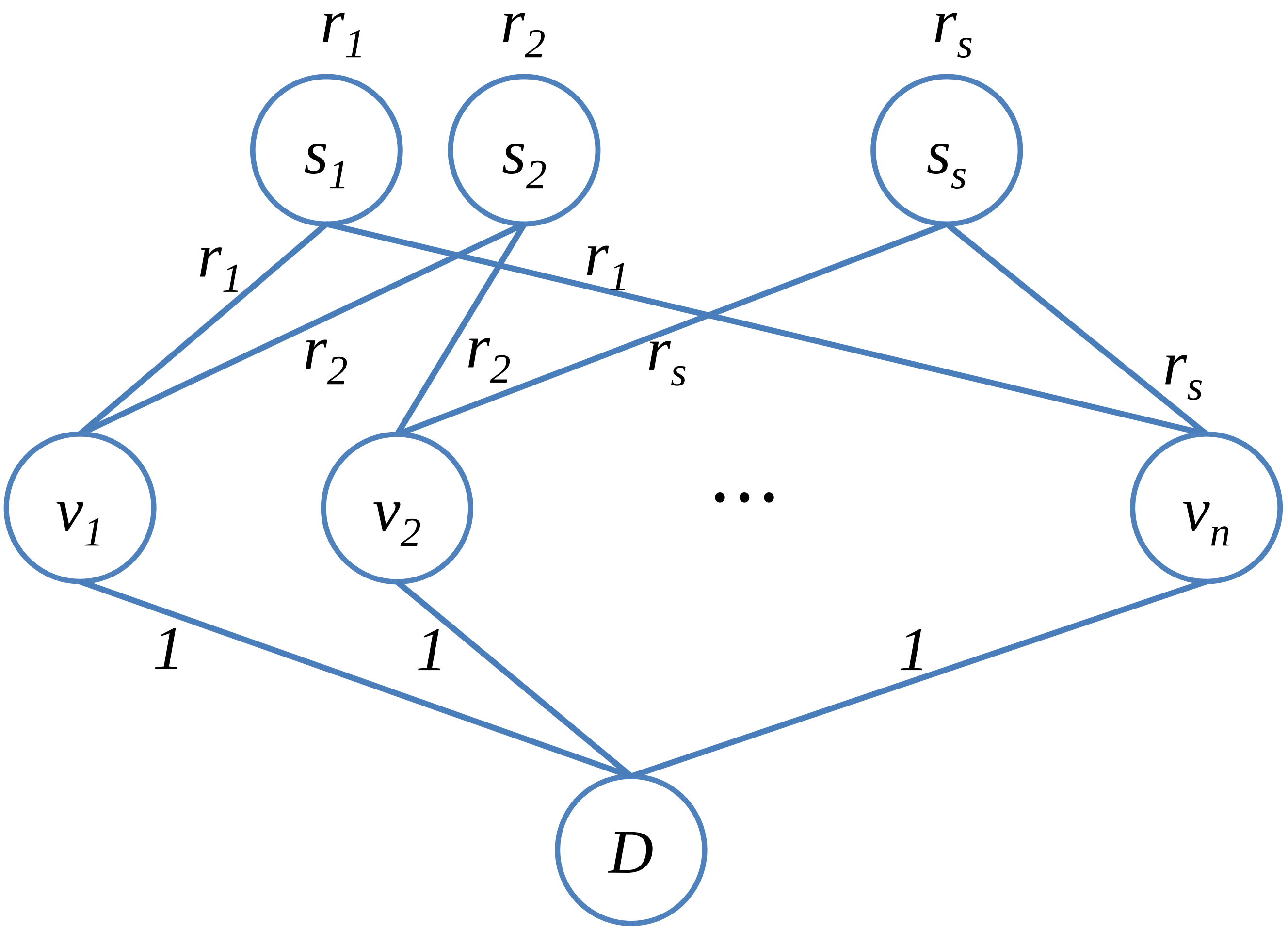
Encoders:

$c_1 = \mathbf{ug_1}$ $\qquad$ $c_2 = \mathbf{ug_2}$ $\qquad$ $c_n = \mathbf{ug_n}$

$$G_{RS} = \begin{bmatrix} \vdots & 0 & & \vdots \\ g_1 & \vdots & & 0 \\ \vdots & g_2 & \cdots & \vdots \\ 0 & \vdots & & g_n \end{bmatrix}$$

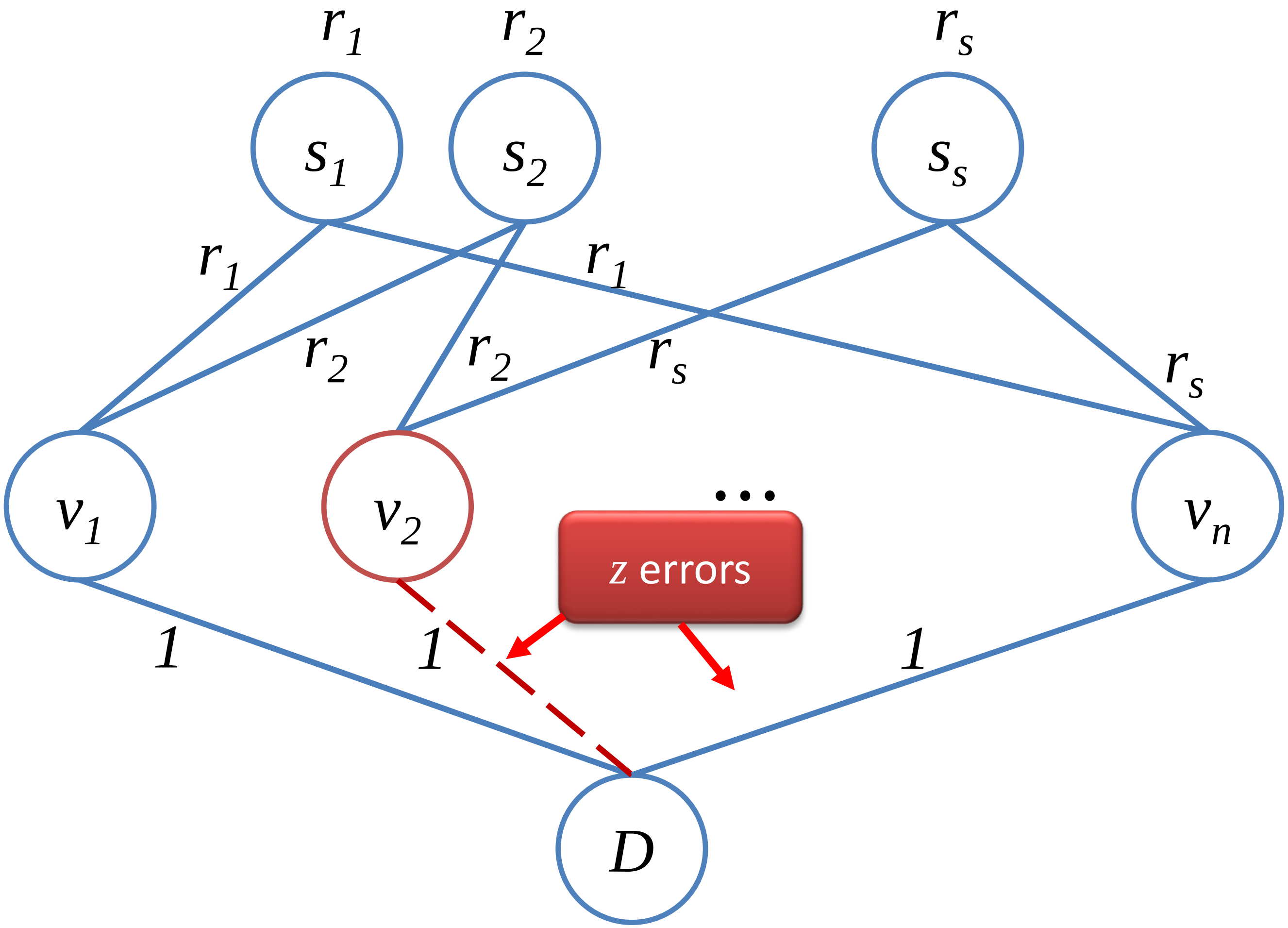
Is there any generator matrix for the RS code with the ***given constraints*** ?

Simple multiple access network



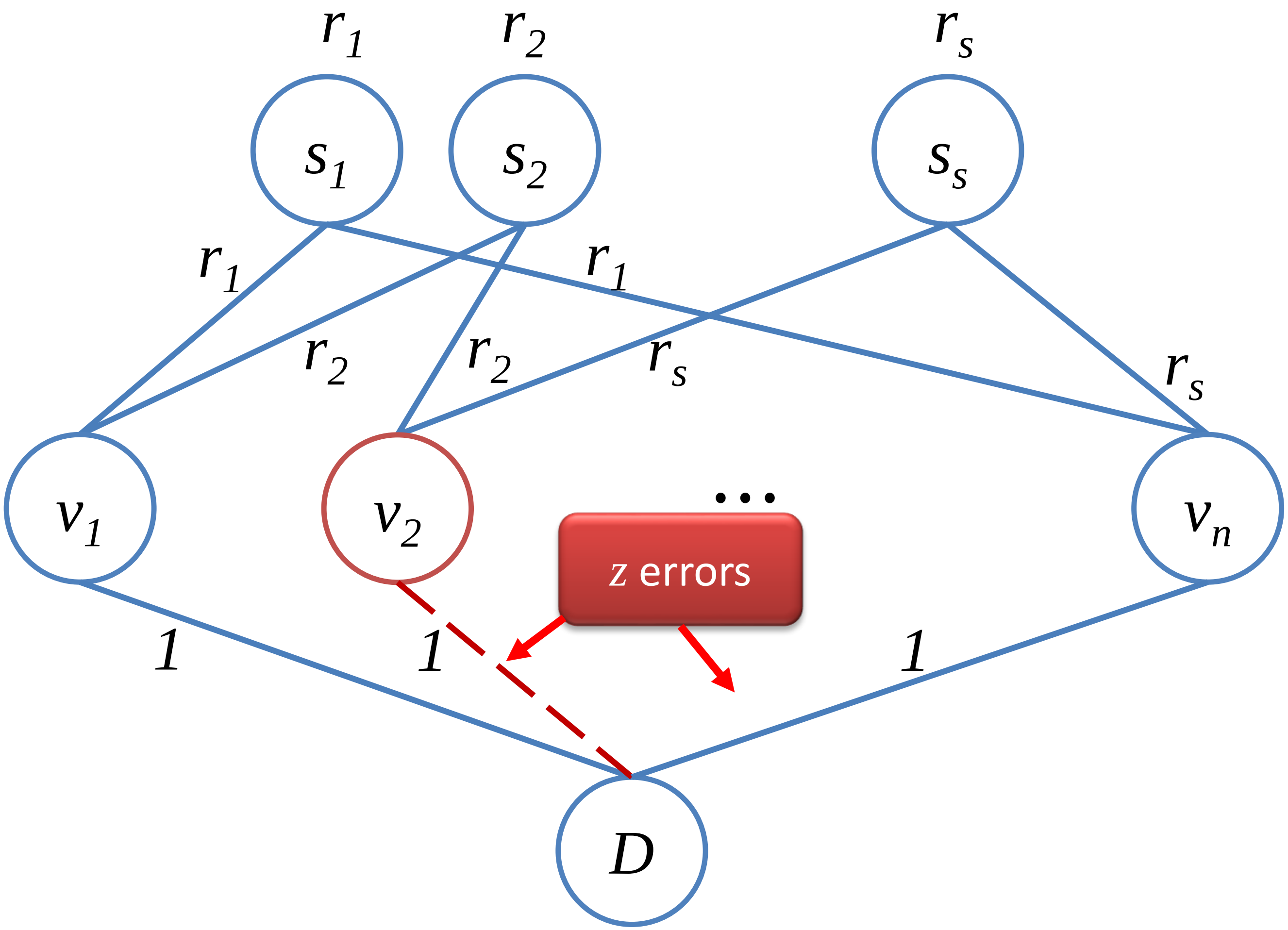
W. Halbawi, T. Ho, H. Yao and I. Duursma, “Distributed codes for simple multiple access network,” arXiv:1310.5187v1, 2013.

Simple multiple access network



At most z relays are compromised by adversaries.

Simple multiple access network



At most z relays are compromised by adversaries.

If relay nodes encode a RS[n, k, $d=2z+1$] code, then destination can recover the data.

Constrained MDS generator matrices

MDS matrix completion problem:
Assume M is a binary $n \times k$ matrix that satisfy *no rectangle condition* (it has no all-zero submatrix of total dimension exceeding *k*). Is there exist an MDS completion for *M*, i.e. replacing 1's in *M* by elements of $\mathbb{F}_q$ such that the constructed matrix generates an MDS code?

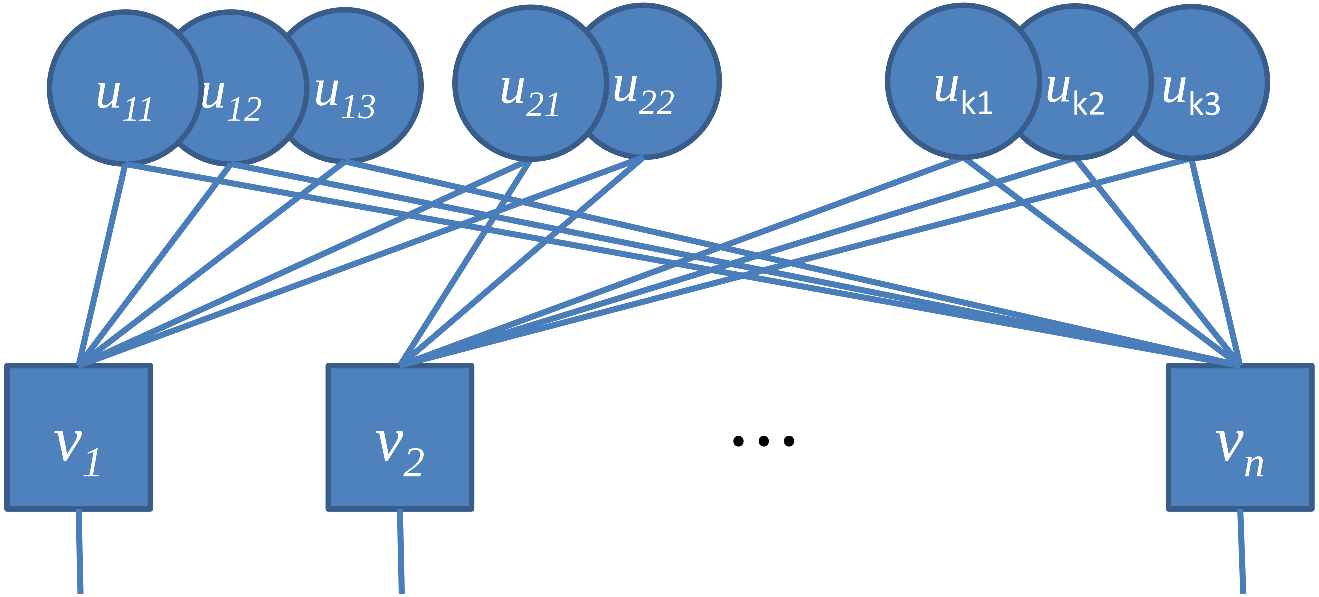
Balanced sparsest generator matrix for MDS codes:
Constructing a generator matrix, *M*, for an MDS code such that each row of *M* has weight *n-k+1* and column weights of *M* differ from each other by at most one.

Weakly secure cooperative data exchange problem
A group of wireless clients have access to different subsets of *n* packets and the like to exchange the packets over a lossless broadcast channel secuirly.

Related works

- [Yao, Ho, Nita-Rotaru ’11] Key agreement for wireless network in the presence of active adversaries
- [Halbawi, Ho, Yao, Duursma’13] Distributed Reed-Solomon codes for simple multiple access networks
- [Dau, Song, Dong, Yuen ’13] Balanced sparsest generator matrices for MDS codes
- [El Rouayheb, Sprintson, Sadeghi ’10] On coding for cooperative data exchange
- [Dau, Song, Sprintson, Yuen ’15] Constructions of MDS codes via random Vandermonde and Cauchy matrices over small fields
- [Dau, Song, Yuen ’14]On the existence of MDS codes over small fields with constrained generator matrices
- [Yan, Sprintson ’13] Algorithms for weakly secure data exchange
- [W. Halbawi, M. Thill and B. Hassibi] Coding with constraints: Minimum distance bounds and systematic constructions
- [W. Halbawi, Z. Liu and B. Hassibi] Balanced Reed-Solomon codes

Distributed RS Encoding

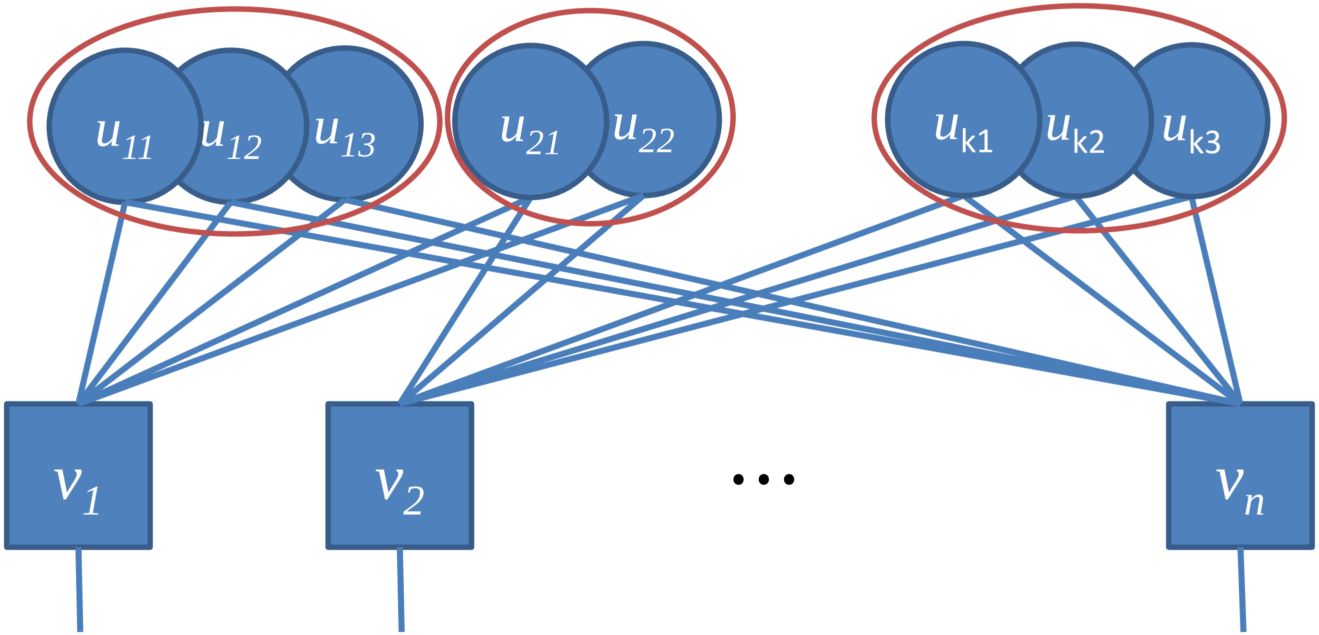


$c_1 = \boldsymbol{ug_1}$ $c_2 = \boldsymbol{ug_2}$ $c_n = \boldsymbol{ug_n}$

$$G_{RS} = \begin{bmatrix} \vdots & 0 & & \vdots \\ g_1 & \vdots & & 0 \\ \vdots & g_2 & \cdots & \vdots \\ 0 & \vdots & & g_n \end{bmatrix}$$

Is there any generator matrix for the RS code with the ***given constraints*** ?

Distributed RS Encoding

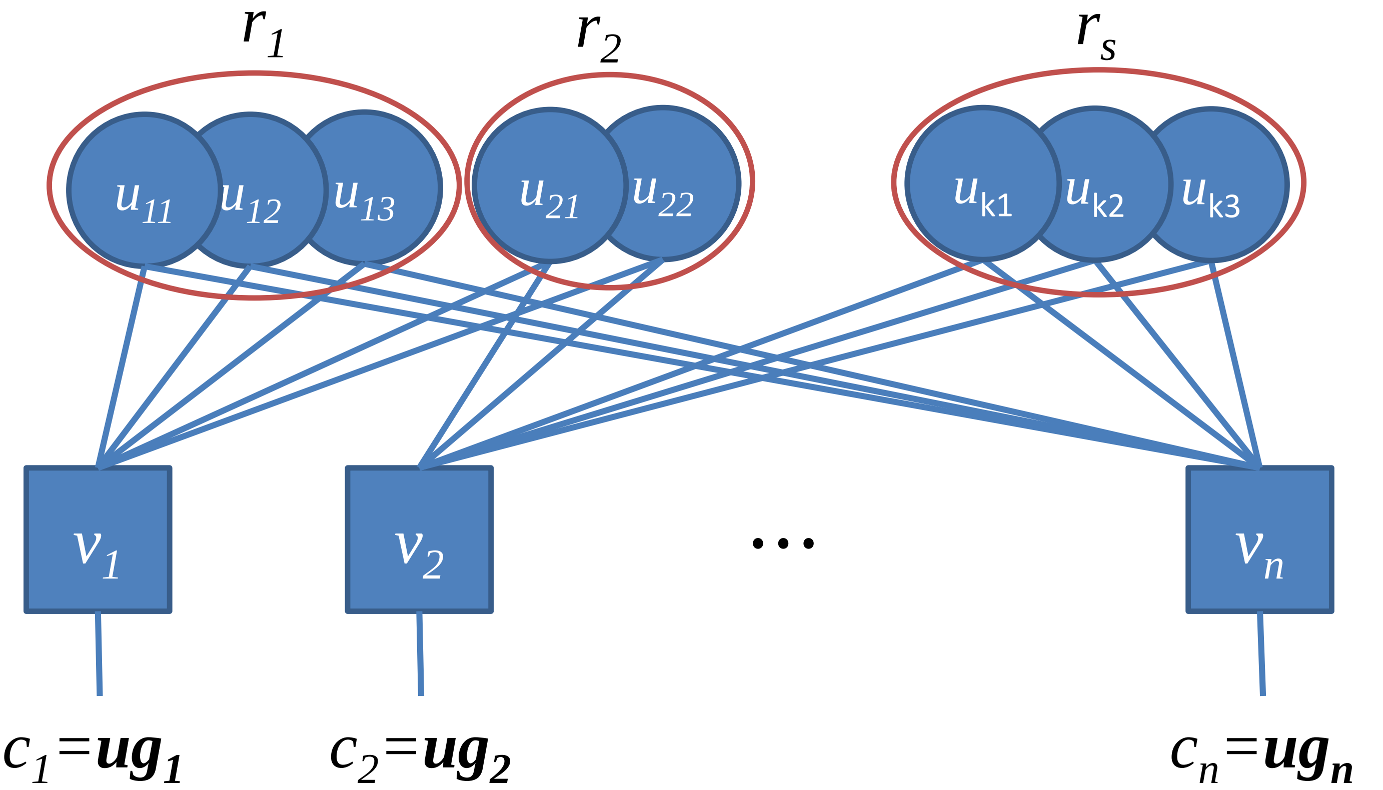


$c_1 = \mathbf{ug_1}$ $c_2 = \mathbf{ug_2}$ $c_n = \mathbf{ug_n}$

$$G_{RS} = \begin{bmatrix} \vdots & 0 & & \vdots \\ g_1 & \vdots & & 0 \\ \vdots & g_2 & \cdots & \vdots \\ 0 & \vdots & & g_n \end{bmatrix}$$

Is there any generator matrix for the RS code with the ***given constraints*** ?

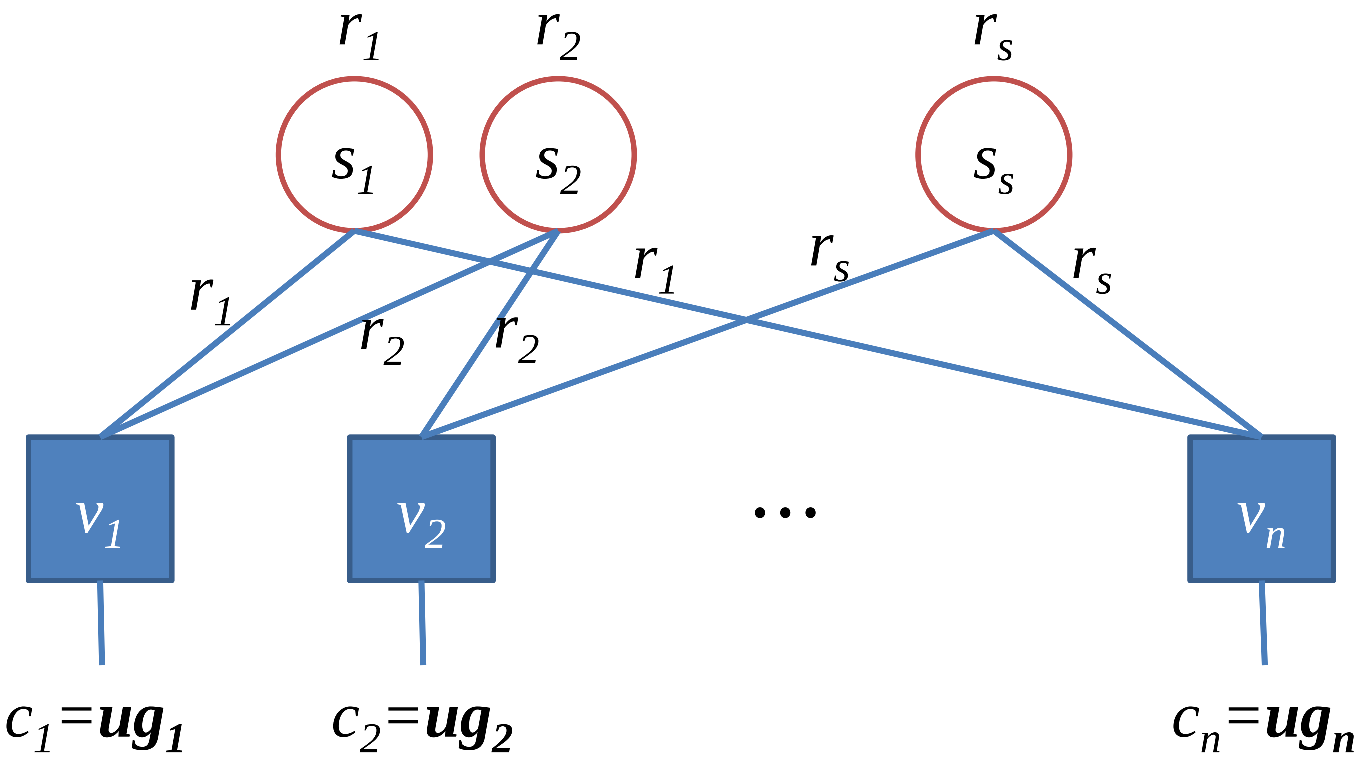
Distributed RS Encoding


r_1
r_2
r_s
u_{11}
u_{12}
u_{13}
u_{21}
u_{22}
u_{k1}
u_{k2}
u_{k3}
v_1
v_2
…
v_n
$c_1 = \boldsymbol{u}\boldsymbol{g}_1$
$c_2 = \boldsymbol{u}\boldsymbol{g}_2$
$c_n = \boldsymbol{u}\boldsymbol{g}_n$

$$G_{RS} = \begin{bmatrix} \vdots & 0 & & \vdots \\ g_1 & \vdots & & 0 \\ \vdots & g_2 & \cdots & \vdots \\ 0 & \vdots & & g_n \end{bmatrix}$$

Is there any generator matrix for the RS code with the ***given constraints*** ?

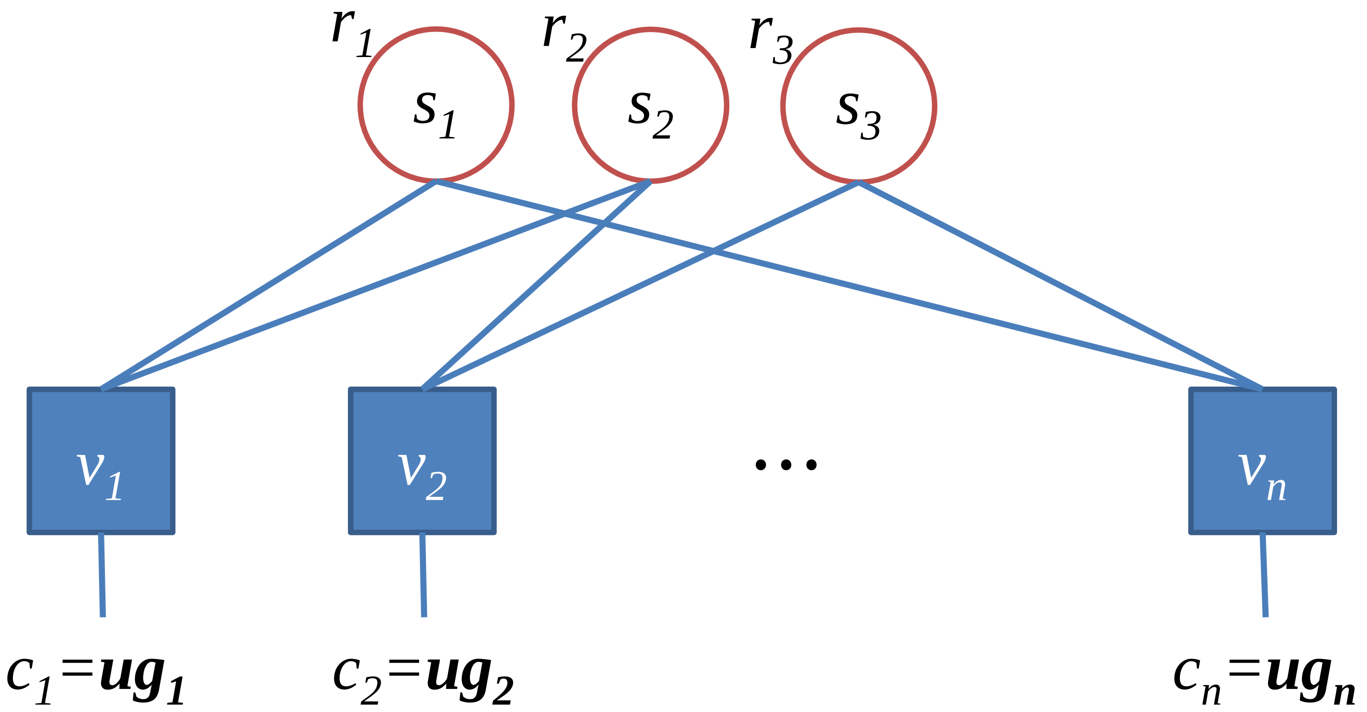
Distributed RS Encoding



$$G_{RS}=\begin{bmatrix} \vdots & 0 & & \vdots \\ g_1 & \vdots & & 0 \\ \vdots & g_2 & \cdots & \vdots \\ 0 & \vdots & & g_n \end{bmatrix}$$

Is there any generator matrix for the RS code with the ***given constraints*** ?

Example: Three sources



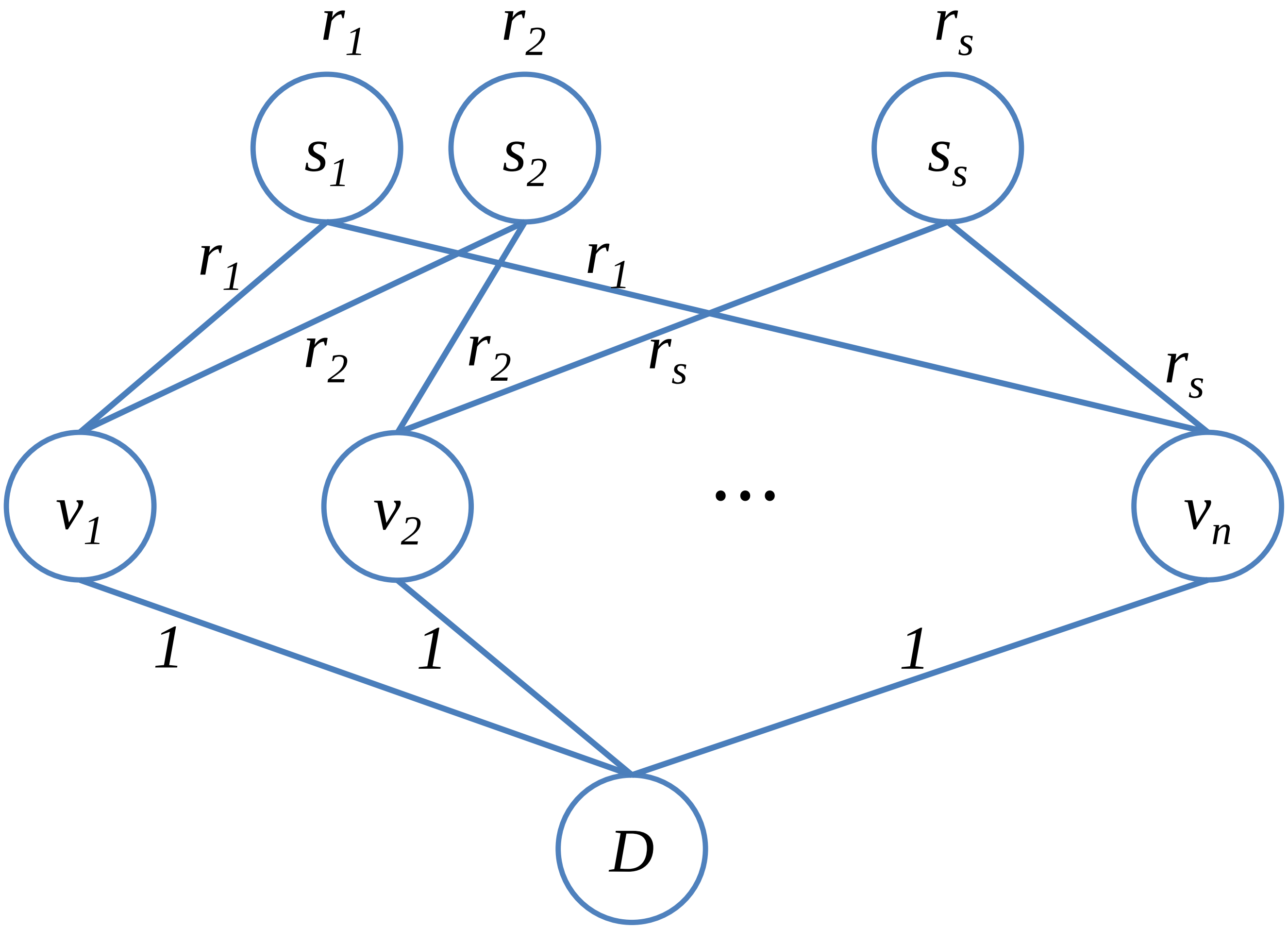
$$G_{RS}=\begin{array}{c|ccccccc|} & n_1 & n_2 & n_3 & n_{12} & n_{13} & n_{23} & n_{123} \\ r_1 & \times & 0 & 0 & \times & \times & 0 & \times \\ r_2 & 0 & \times & 0 & \times & 0 & \times & \times \\ r_3 & 0 & 0 & \times & 0 & \times & \times & \times \end{array}$$

$\mathcal{Z}_1$={positions of zeros of the first group}

$\mathcal{Z}_2$={positions of zeros of the second group}

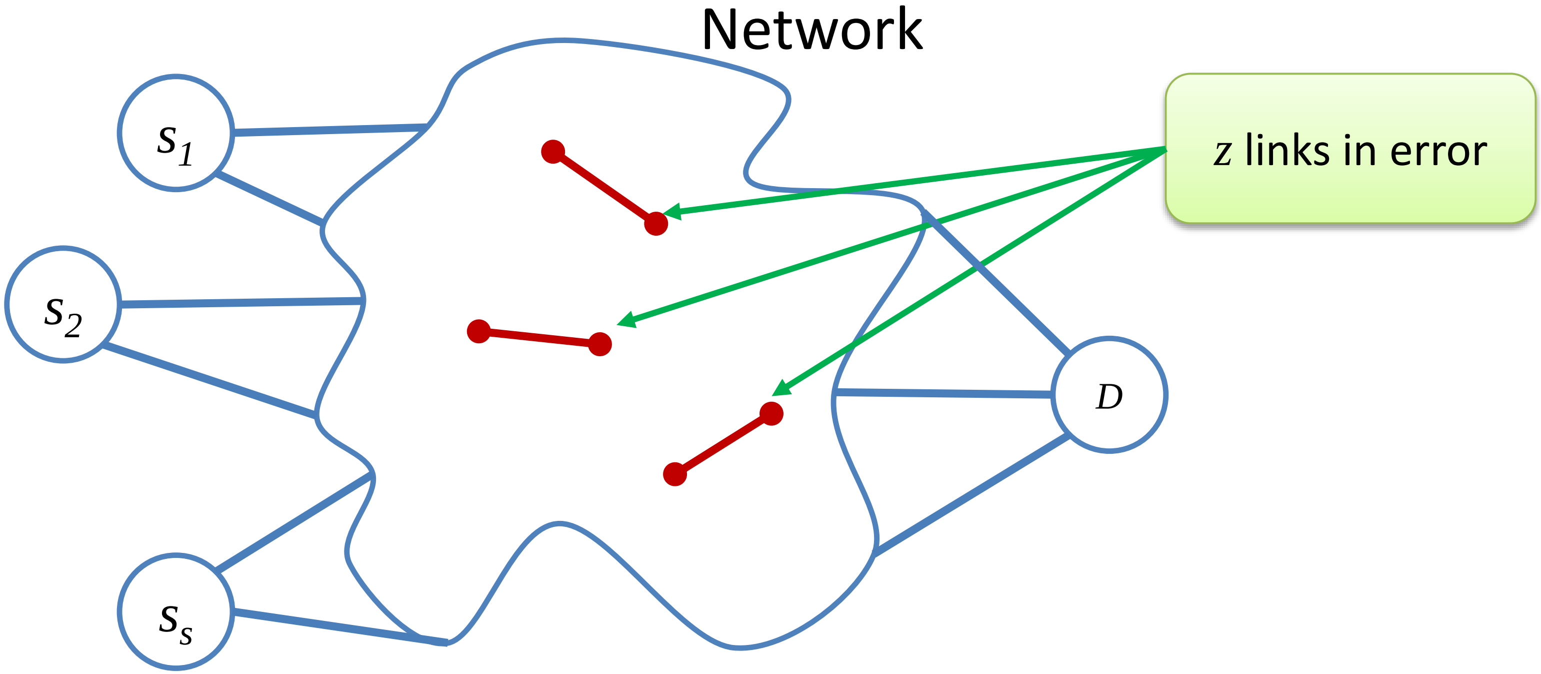
$\mathcal{Z}_3$={positions of zeros of the third group}

Simple multiple access network



W. Halbawi, T. Ho, H. Yao and I. Duursma, “Distributed codes for simple multiple access network,” arXiv:1310.5187v1, 2013.

Necessary condition (network coding)

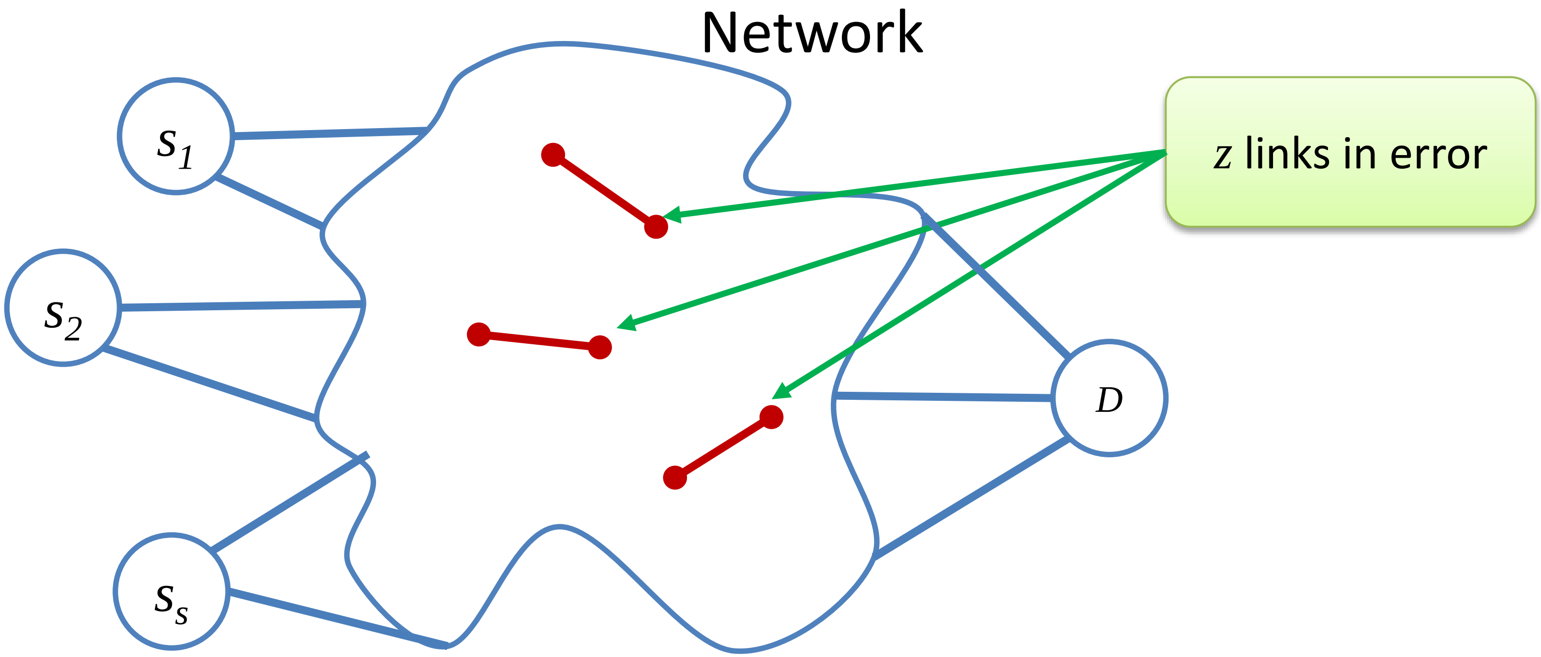


We can only hope to find a distributed RS code for rates $(r_1, r_2, \ldots, r_s)$ in the *capacity region* of the network.

Network error correction

- N. Cai and R. W. Yeung (2006)
  - Single source multicast networks
- D. Silva, F. R. Kschischang, and R. Koetter (2008)
  - Rank-metric codes
- S. Mohajer, M. Jafari, S. Diggavi, and C. Fragouli (2009)
  - Two source multicast networks
- T. Dikaliotis, T. Ho, S. Jaggi, S. Vyetrenko, H. Yao, M. Effros, J. Kliewer, and E. Erez (2011)
  - Multisource multicast networks
- X. Guang and Z. Zhang (2014)
  - Linear network error correction coding

Network error correction

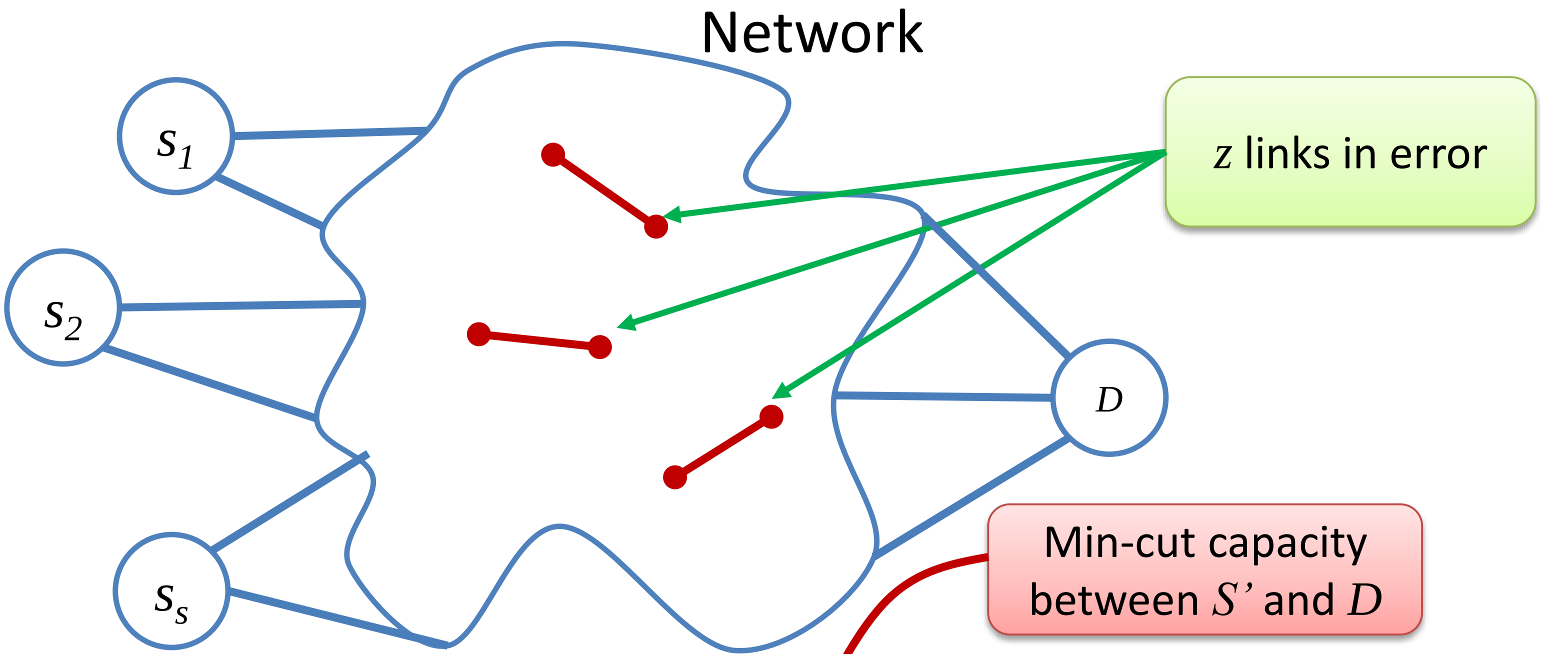


z links in error

Capacity region: $\sum_{i \in I(S')} r_i \leq C_{S'} - 2z, \forall S' \subseteq S$

T. Dikaliotis, T. Ho, S. Jaggi, S. Vyetrenko, H. Yao, M. Effros, J. Kliewer, and E. Erez, "Multiple access network information-flow and correction codes", *IEEE IT*, 57(2), 2011.

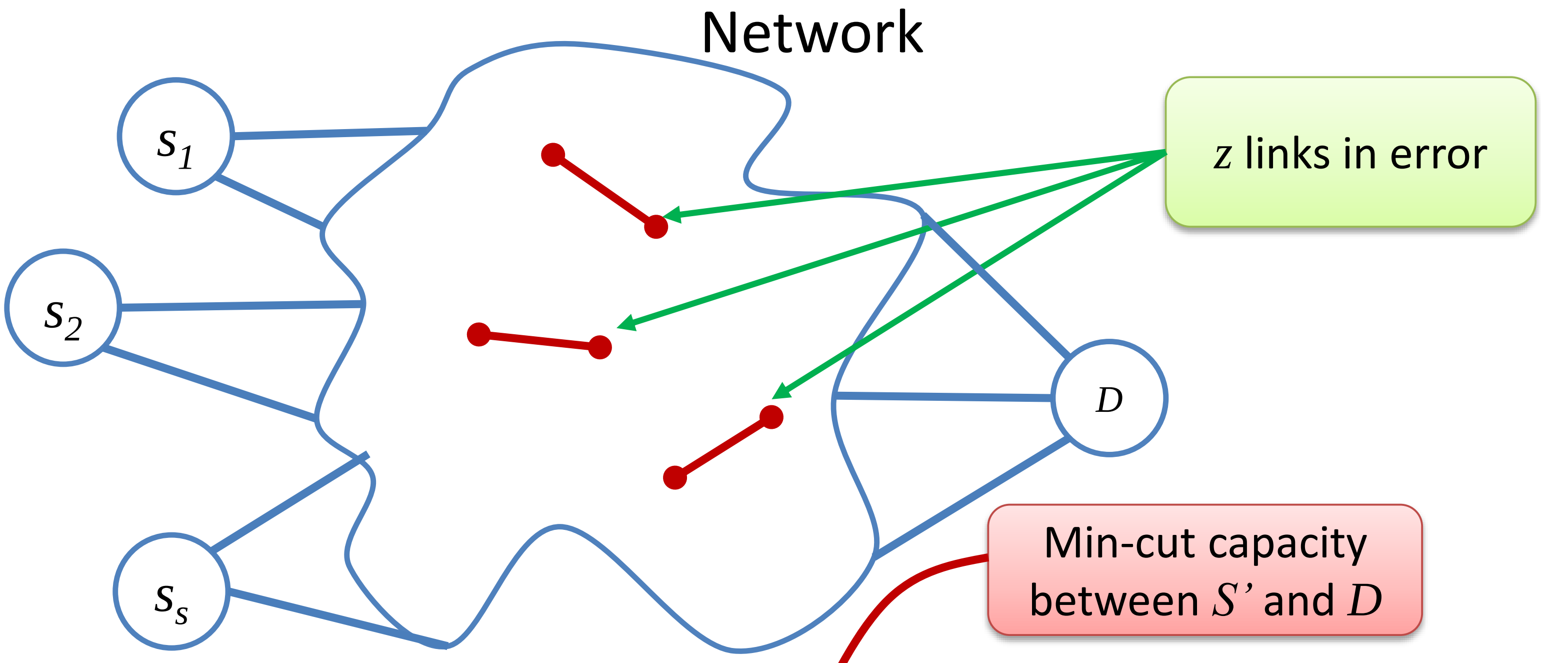
Network error correction



Capacity region: $$\sum_{i \in I(S')} r_i \le C_{S'} - 2z, \forall S' \subseteq S$$

T. Dikaliotis, T. Ho, S. Jaggi, S. Vyetrenko, H. Yao, M. Effros, J. Kliewer, and E. Erez, "Multiple access network information-flow and correction codes", *IEEE IT*, 57(2), 2011.

Network error correction



z links in error

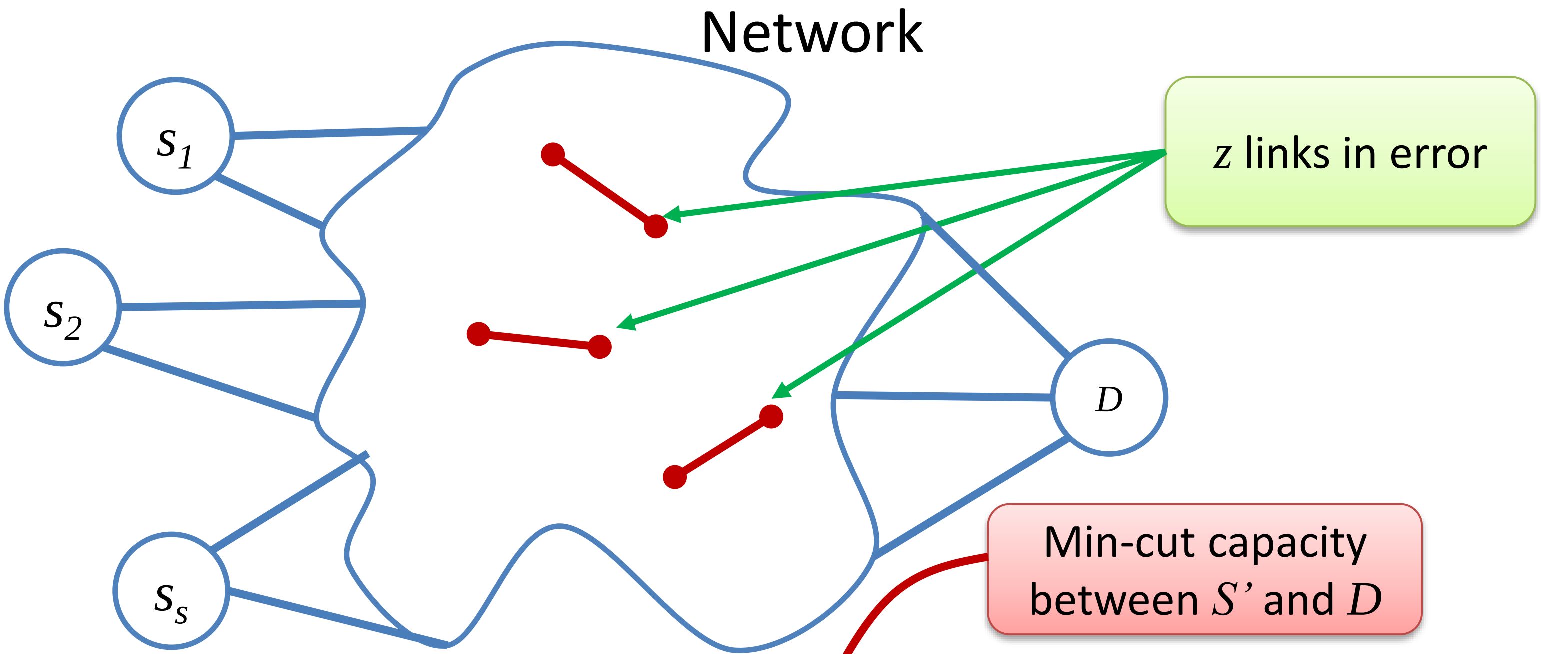
Min-cut capacity between S' and D

Capacity region: $$\sum_{i\in I(S')} r_i \le C_{S'} - 2z, \forall S' \subseteq S$$

Computationally efficient linear network codes with decoding success probability of at least 1-$|s||E|/p$ and complexity $O(l\ m^{|S|})$

T. Dikaliotis, T. Ho, S. Jaggi, S. Vyetrenko, H. Yao, M. Effros, J. Kliewer, and E. Erez, "Multiple access network information-flow and correction codes", *IEEE IT*, 57(2), 2011.

Network error correction



z links in error

Min-cut capacity between S' and D

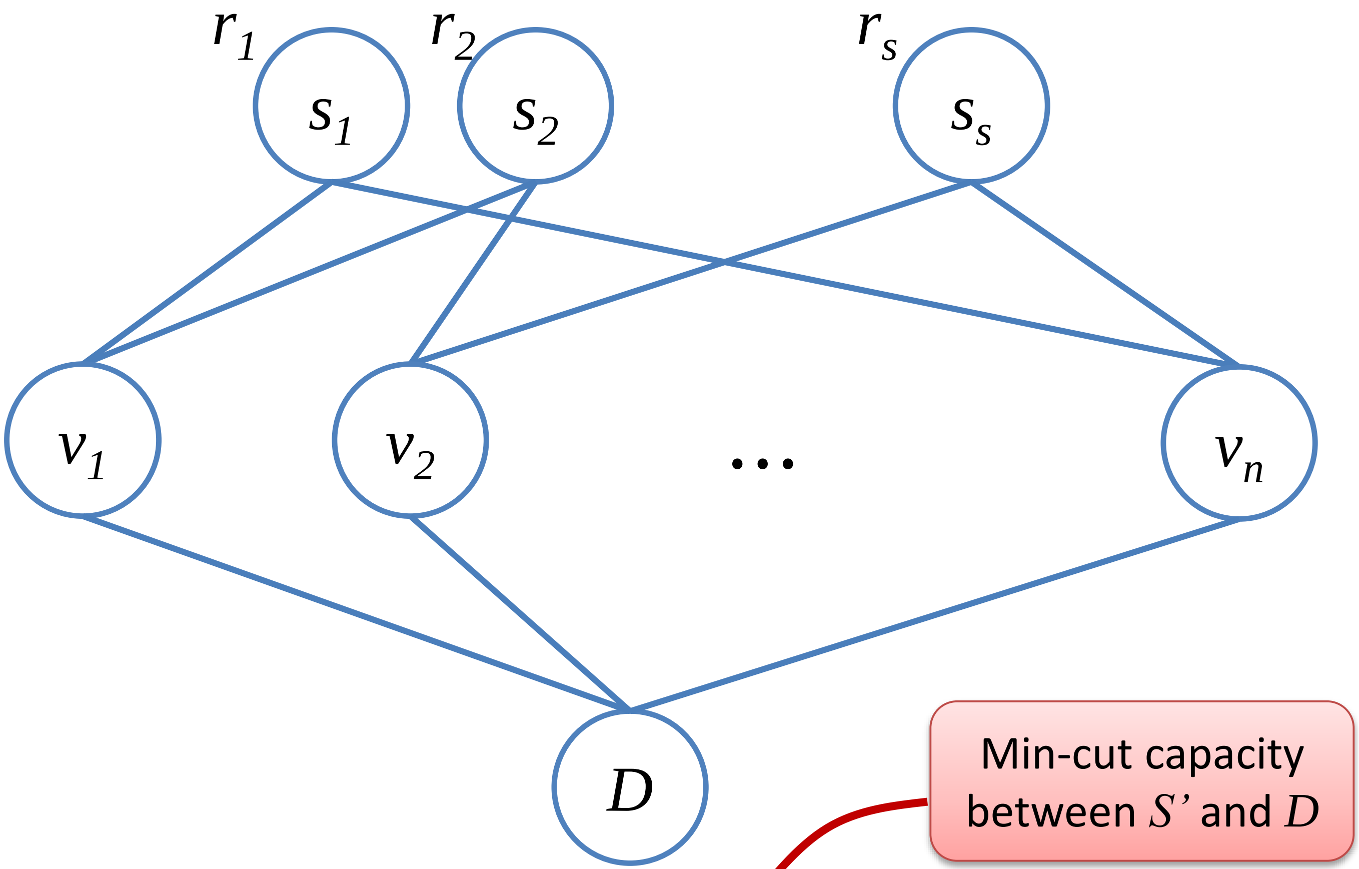
Capacity region: $$\sum_{i\in I(S')} r_i \leq C_{S'} - 2z, \forall S' \subseteq S$$

Exponential in $|S|$

Computationally efficient linear network codes with decoding success probability of at least 1-$|s||E|/p$ and complexity $O(l\ m^{|S|})$

T. Dikaliotis, T. Ho, S. Jaggi, S. Vyetrenko, H. Yao, M. Effros, J. Kliewer, and E. Erez, "Multiple access network information-flow and correction codes", *IEEE IT*, 57(2), 2011.

Necessary condition (network coding)



Min-cut capacity between S' and D

Necessary condition: $\sum_{i \in I(S')} r_i \leq C_{S'} - 2z, \forall S' \subseteq S$

T. Dikaliotis, T. Ho, S. Jaggi, S. Vyetrenko, H. Yao, M. Effros, J. Kliewer, and E. Erez, "Multiple access network information-flow and correction codes", *IEEE IT*, 57(2), 2011.

Necessary condition (three sources)

$$G_{RS} = \begin{array}{c|ccccccc|} & n_1 & n_2 & n_3 & n_{12} & n_{13} & n_{23} & n_{123} \\ r_1 & \times & 0 & 0 & \times & \times & 0 & \times \\ r_2 & 0 & \times & 0 & \times & 0 & \times & \times \\ r_3 & 0 & 0 & \times & 0 & \times & \times & \times \end{array}$$

$\mathcal{Z}_1$ ={positions of zeros of the first group}

$\mathcal{Z}_2$ ={positions of zeros of the second group}

$\mathcal{Z}_3$ ={positions of zeros of the third group}

Capacity region:

$$r_i \leq k - |\mathcal{Z}_i|, i \in \{1,2,3\}$$

$$r_i + r_j \leq k - \left|\mathcal{Z}_i \cap \mathcal{Z}_j\right|, i, j \in \{1,2,3\}$$

$$r_1 + r_2 + r_3 \leq k$$

Necessary condition (three sources)

$$G_{RS} = \begin{array}{c|ccccccc|} & n_1 & n_2 & n_3 & n_{12} & n_{13} & n_{23} & n_{123} \\ \hline r_1 & \times & 0 & 0 & \times & \times & 0 & \times \\ r_2 & 0 & \times & 0 & \times & 0 & \times & \times \\ r_3 & 0 & 0 & \times & 0 & \times & \times & \times \end{array}$$

$\mathcal{Z}_1$ ={positions of zeros of the first group}

$\mathcal{Z}_2$ ={positions of zeros of the second group}

$\mathcal{Z}_3$ ={positions of zeros of the third group}

Capacity region:

$$r_i \leq k - |\mathcal{Z}_i|, i \in \{1,2,3\}$$

$$r_i + r_j \leq k - \left|\mathcal{Z}_i \cap \mathcal{Z}_j\right|, i, j \in \{1,2,3\}$$

$$r_1 + r_2 + r_3 \leq k$$

Necessary condition (three sources)

$$G_{RS} = \begin{array}{c|ccccccc|} & n_1 & n_2 & n_3 & n_{12} & n_{13} & n_{23} & n_{123} \\ r_1 & \times & 0 & 0 & \times & \times & 0 & \times \\ r_2 & 0 & \times & 0 & \times & 0 & \times & \times \\ r_3 & 0 & 0 & \times & 0 & \times & \times & \times \end{array}$$

$\mathcal{Z}_1$ ={positions of zeros of the first group}

$\mathcal{Z}_2$ ={positions of zeros of the second group}

$\mathcal{Z}_3$ ={positions of zeros of the third group}

Capacity region:

$$r_i \leq k - |\mathcal{Z}_i|, i \in \{1,2,3\}$$

$$r_i + r_j \leq k - \left|\mathcal{Z}_i \cap \mathcal{Z}_j\right|, i, j \in \{1,2,3\}$$

$$r_1 + r_2 + r_3 \leq k$$

Necessary conditions (three sources)

$$G_{RS} = \begin{array}{c} \\ r_1 \\ r_2 \\ r_3 \end{array} \begin{array}{c} \begin{array}{ccccccc} n_1 & n_2 & n_3 & n_{12} & n_{13} & n_{23} & n_{123} \end{array} \\ \begin{bmatrix} \times & 0 & 0 & \times & \times & 0 & \times \\ 0 & \times & 0 & \times & 0 & \times & \times \\ 0 & 0 & \times & 0 & \times & \times & \times \end{bmatrix} \end{array}$$

$\mathcal{Z}_1$ ={positions of zeros of the first group}

$\mathcal{Z}_2$ ={positions of zeros of the second group}

$\mathcal{Z}_3$ ={positions of zeros of the third group}

Capacity region:

$$r_i \leq k - |\mathcal{Z}_i|, i \in \{1,2,3\}$$
$$r_i + r_j \leq k - \left|\mathcal{Z}_i \cap \mathcal{Z}_j\right|, i, j \in \{1,2,3\}$$
$$r_1 + r_2 + r_3 \leq k$$

If rates are inside the capacity region, Halbawi *et al.* (2014) showed that for up to ***three sources*** one can always find G_{RS} .

Necessary conditions (three sources)

n_1 n_2 n_3 n_{12} n_{13} n_{23} n_{123}

up}

oup}

p}

We are going to show that for

any number of sources

if rates are inside the capacity region of SMAN, one can always construct the generator matrix G_{RS} for the distributed RS code over a finite field of size at least n.

If rates are inside the capacity region, Halbawi *et al.* (2014) showed that for up to ***three sources*** one can always find G_{RS} .

Proof (by induction on # sources)

- The result holds for the case of two sources
- We assume that the result holds for the case of having less than s sources. We show that it holds for the case of s sources

Constraints for the case of s sources:

$$r_1 \leq k - |\mathcal{Z}_1|$$

$$r_2 \leq k - |\mathcal{Z}_2|$$

$$\vdots$$

$$r_s \leq k - |\mathcal{Z}_s|$$

$$\vdots$$

$$r_{i_1} + r_{i_2} + \cdots + r_{i_l} \leq k - |\mathcal{Z}_{i_1} \cap \mathcal{Z}_{i2} \cap \cdots \cap \mathcal{Z}_{il}|$$

$$\vdots$$

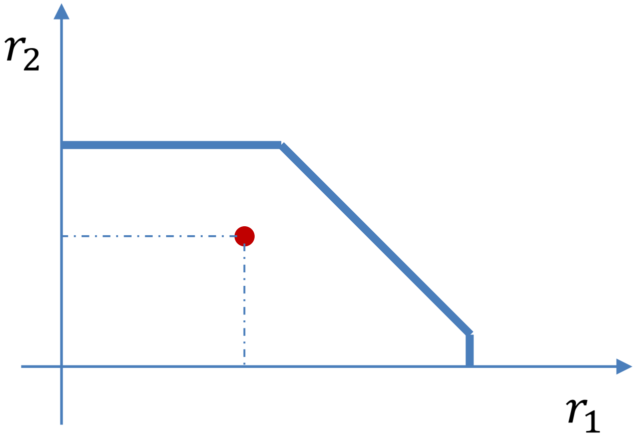
$$r_1 + r_2 + \cdots + r_s \leq k$$

When rates are inside the boundaries

All the constraints ***are not tight***.

Constraints for the case of s sources:

$$
\begin{aligned}
r_1 &< k - |\mathcal{Z}_1| \\
r_2 &< k - |\mathcal{Z}_2| \\
&\;\vdots \\
r_s &< k - |\mathcal{Z}_s| \\
&\;\vdots \\
r_{i_1} + r_{i_2} + \cdots + r_{i_l} &< k - |\mathcal{Z}_{i_1} \cap \mathcal{Z}_{i_2} \cap \cdots \cap \mathcal{Z}_{i_l}| \\
&\;\vdots \\
r_1 + r_2 + \cdots + r_s &< k
\end{aligned}
$$

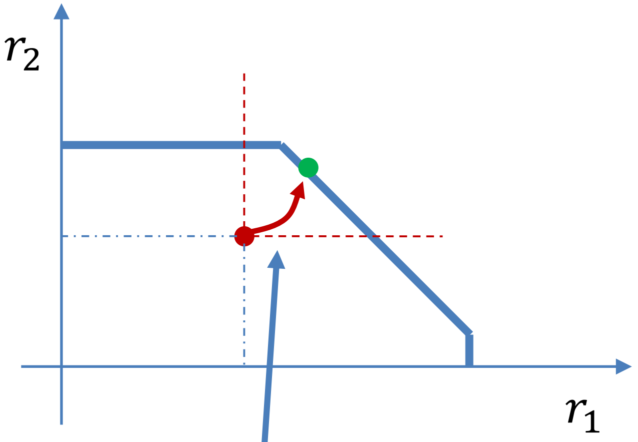


Rates inside the boundaries

All the constraints ***are not tight***.

Constraints for the case of s sources:

$$
\begin{aligned}
r_1 &< k - |\mathcal{Z}_1| \\
r_2 &< k - |\mathcal{Z}_2| \\
&\vdots \\
r_s &< k - |\mathcal{Z}_s| \\
&\vdots \\
r_{i_1} + r_{i_2} + \cdots + r_{i_l} &< k - |\mathcal{Z}_{i_1} \cap \mathcal{Z}_{i_2} \cap \cdots \cap \mathcal{Z}_{i_l}| \\
&\vdots \\
r_1 + r_2 + \cdots + r_s &\leq k
\end{aligned}
$$



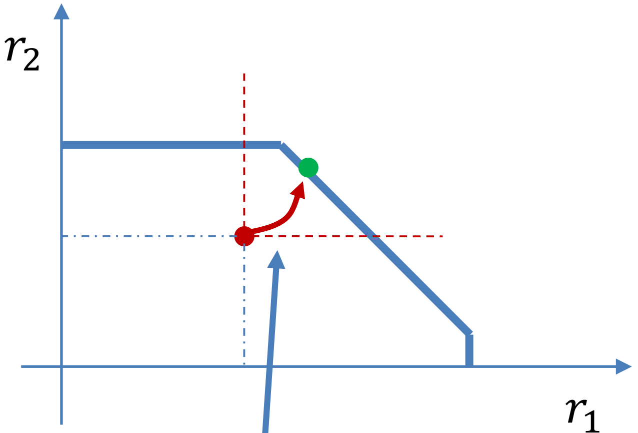
Increase r_i's till we hit the boundaries

Rates inside the boundaries

All the constraints ***are not tight***.

Constraints for the case of s sources:

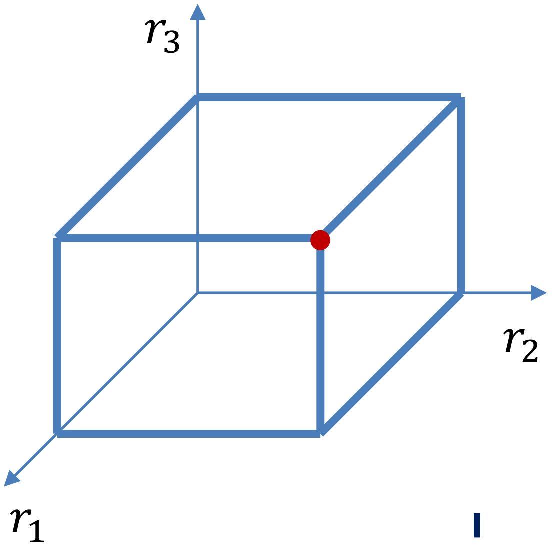
$$
\begin{aligned}
r_1 &< k - |\mathcal{Z}_1| \\
r_2 &< k - |\mathcal{Z}_2| \\
&\ \vdots \\
r_s &< k - |\mathcal{Z}_s| \\
&\ \vdots \\
r_{i_1} + r_{i_2} + \cdots + r_{i_l} &< k - |\mathcal{Z}_{i_1} \cap \mathcal{Z}_{i_2} \cap \cdots \cap \mathcal{Z}_{i_l}| \\
&\ \vdots \\
r_1 + r_2 + \cdots + r_s &\leq k
\end{aligned}
$$



Increase r_i's till we hit the boundaries

Notice: If we construct G_{RS} for the **new rates**, then by removing some rows of G_{RS}, we can construct G_{RS} for the **original rates**.

Rates on the boundary



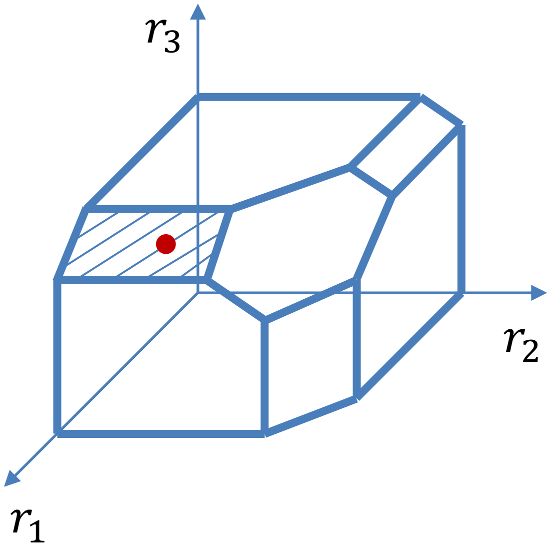
$r_1 = k - |\mathcal{Z}_1|$

$r_2 = k - |\mathcal{Z}_2|$

$\vdots$

$r_s = k - |\mathcal{Z}_s|$

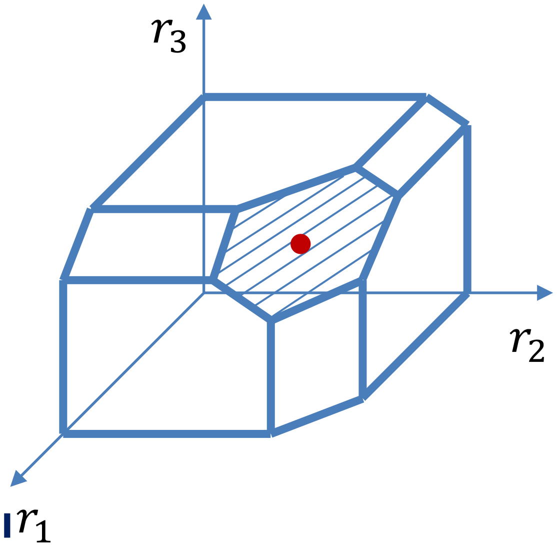
Case I



$\exists l : 2 \leq l < s$

$r_{i_1} + \cdots + r_{i_l} = k - |\mathcal{Z}_{i_1} \cap \cdots \cap \mathcal{Z}_{i_l}|$

Case II



$r_1 + r_2 + \cdots + r_s = k$

$\exists i : r_i < k - |\mathcal{Z}_i|$

$r_{i_1} + \cdots + r_{i_l} <$
$k - |\mathcal{Z}_{i_1} \cap \cdots \cap \mathcal{Z}_{i_l}|$

Case III

Rates on the boundary (Case III)

Constraints:

$$r_1 \leq k - |\mathcal{Z}_1|$$

$$\vdots$$

$$\boxed{\exists i: r_i < k \ - |\mathcal{Z}_i|}$$

$$\vdots$$

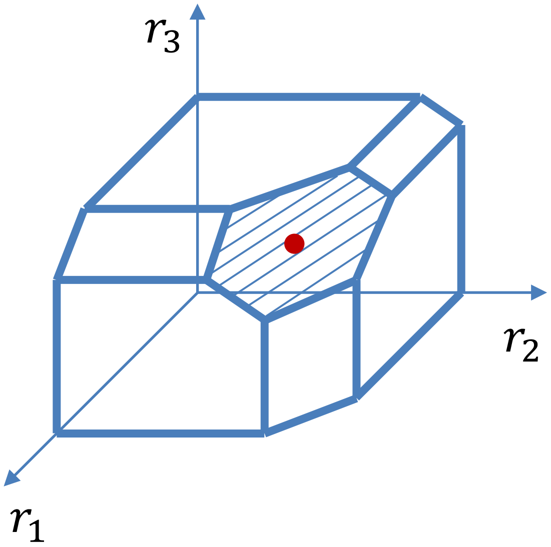
$$r_s \leq k - |\mathcal{Z}_s|$$

$$\vdots$$

$$\boxed{r_{i_1} + r_{i_2} + \cdots + r_{i_l} < k - |\mathcal{Z}_{i_1} \cap \mathcal{Z}_{i_2} \cap \cdots \cap \mathcal{Z}_{i_l}|}$$

$$\vdots$$

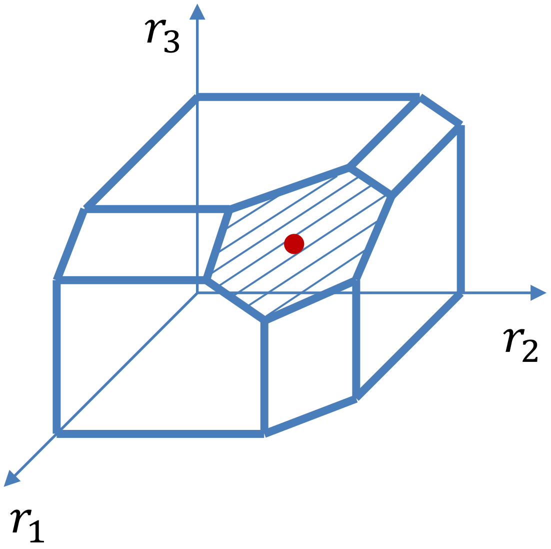
$$\boxed{r_1 + r_2 + \cdots + r_s = k}$$



Rates on the boundary (Case III)

Constraints:

$$\begin{gathered}
r_1 \le k - |\mathcal{Z}_1| \\
\vdots \\
\exists i: r_i {\color{red}<} k - |\mathcal{Z}_i| \\
\vdots \\
r_s \le k - |\mathcal{Z}_s| \\
\vdots \\
r_{i_1} + r_{i_2} + \cdots + r_{i_l} {\color{red}<} k - |\mathcal{Z}_{i_1} \cap \mathcal{Z}_{i_2} \cap \cdots \cap \mathcal{Z}_{i_l}| \\
\vdots \\
r_1 + r_2 + \cdots + r_s {\color{red}=} k
\end{gathered}$$



$$G_{RS} = \begin{array}{c} r_1 \\ r_2 \\ r_3 \end{array} \begin{bmatrix} \times & 0 & 0 & \times & \times & \times & 0 & \times \\ 0 & \times & 0 & \times & \times & 0 & \times & \times \\ 0 & 0 & \times & 0 & 0 & \times & \times & \times \end{bmatrix}$$

Rates on the boundary (Case III)

Constraints:

$$r_1 \le k - |\mathcal{Z}_1|$$

$$\vdots$$

$$\exists i: r_i < k - |\mathcal{Z}_i|$$

$$\vdots$$

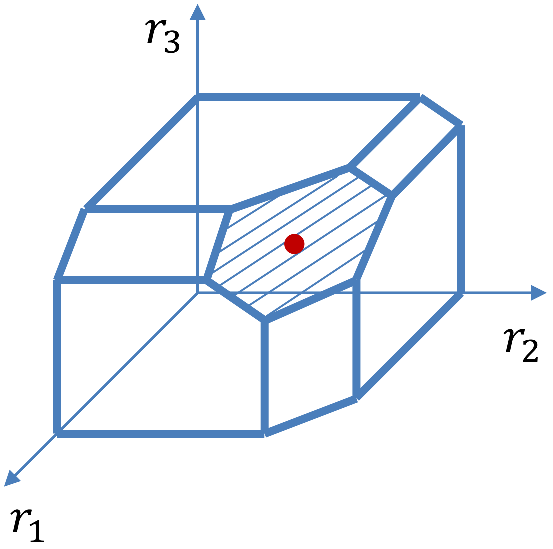
$$r_s \le k - |\mathcal{Z}_s|$$

$$\vdots$$

$$r_{i_1} + r_{i_2} + \cdots + r_{i_l} < k - |\mathcal{Z}_{i_1} \cap \mathcal{Z}_{i_2} \cap \cdots \cap \mathcal{Z}_{i_l}|$$

$$\vdots$$

$$r_1 + r_2 + \cdots + r_s = k$$



We can always ***add a column of all zeros*** to the i-th group of G_{RS} without violating the constraints.

$$G_{RS} = \begin{array}{c} r_1 \\ r_2 \\ r_3 \end{array}\begin{bmatrix} \times & 0 & 0 & \times & \times & \times & 0 & \times \\ 0 & \times & 0 & \times & \times & 0 & \times & \times \\ 0 & 0 & \times & 0 & 0 & \times & \times & \times \end{bmatrix}$$

Rates on the boundary (Case III)

Constraints:

$$r_1 \leq k - |\mathcal{Z}_1|$$

$$\vdots$$

$$\exists i: r_i < k - |\mathcal{Z}_i|$$

$$\vdots$$

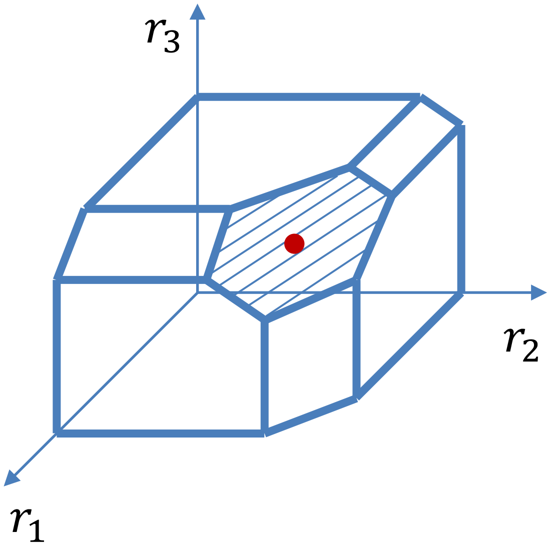
$$r_s \leq k - |\mathcal{Z}_s|$$

$$\vdots$$

$$r_{i_1} + r_{i_2} + \cdots + r_{i_l} < k - |\mathcal{Z}_{i_1} \cap \mathcal{Z}_{i_2} \cap \cdots \cap \mathcal{Z}_{i_l}|$$

$$\vdots$$

$$r_1 + r_2 + \cdots + r_s = k$$



We can always ***add a column of all zeros*** to the i-th group of G_{RS} without violating the constraints.

$$G_{RS} = \begin{array}{c} r_1 \\ r_2 \\ r_3 \end{array} \begin{bmatrix} \times & 0 & 0 & \times & \times & \times & 0 & \times \\ 0 & \times & 0 & \times & 0 & 0 & \times & \times \\ 0 & 0 & \times & 0 & 0 & \times & \times & \times \end{bmatrix}$$

Rates on the boundary (Case III)

Constraints:

$$r_1 \le k - |\mathcal{Z}_1|$$

$$\vdots$$

$$r_i \le k - (|\mathcal{Z}_i| + 1)$$

$$\vdots$$

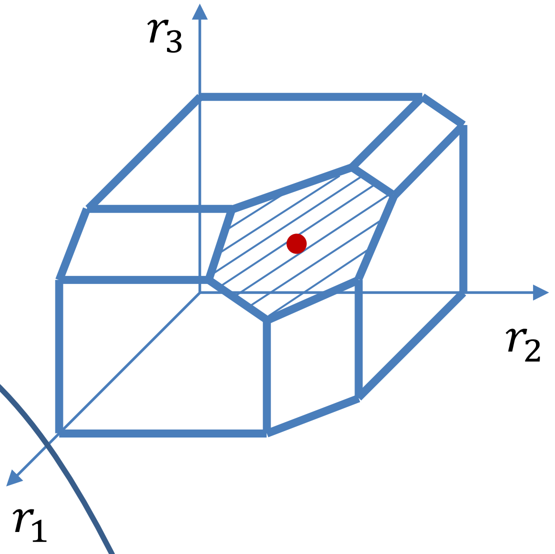
$$r_s \le k - |\mathcal{Z}_s|$$

$$\vdots$$

$$r_{i_1} + r_{i_2} + \cdots + r_{i_l} \le k - (|\mathcal{Z}_{i_1} \cap \mathcal{Z}_{i_2} \cap \cdots \cap \mathcal{Z}_{i_l}| + 1)$$

$$\vdots$$

$$r_1 + r_2 + \cdots + r_s = k$$



$$G_{RS} = \begin{array}{c} r_1 \\ r_2 \\ r_3 \end{array} \begin{bmatrix} \times & 0 & 0 & \times & \times & \times & 0 & \times \\ 0 & \times & 0 & \times & 0 & 0 & \times & \times \\ 0 & 0 & \times & 0 & 0 & \times & \times & \times \end{bmatrix}$$

We can always ***add a column of all zeros*** to the i-th group of G_{RS} without violating the constraints.

Rates on the boundary (Case III)

Constraints:

$$r_1 \le k - |\mathcal{Z}_1|$$

$$\vdots$$

$$r_i \le k - (|\mathcal{Z}_i| + 1)$$

$$\vdots$$

$$r_s \le k - |\mathcal{Z}_s|$$

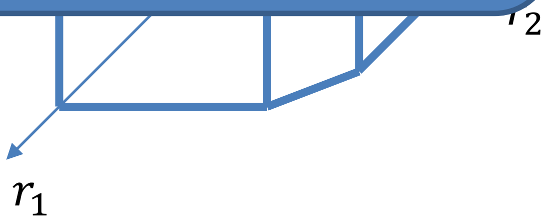
$$\vdots$$

$$r_{i_1} + r_{i_2} + \cdots + r_{i_l} \le k - \left(\left|\mathcal{Z}_{i_1} \cap \mathcal{Z}_{i_2} \cap \cdots \cap \mathcal{Z}_{i_l}\right| + 1\right)$$

$$\vdots$$

$$r_1 + r_2 + \cdots + r_s = k$$

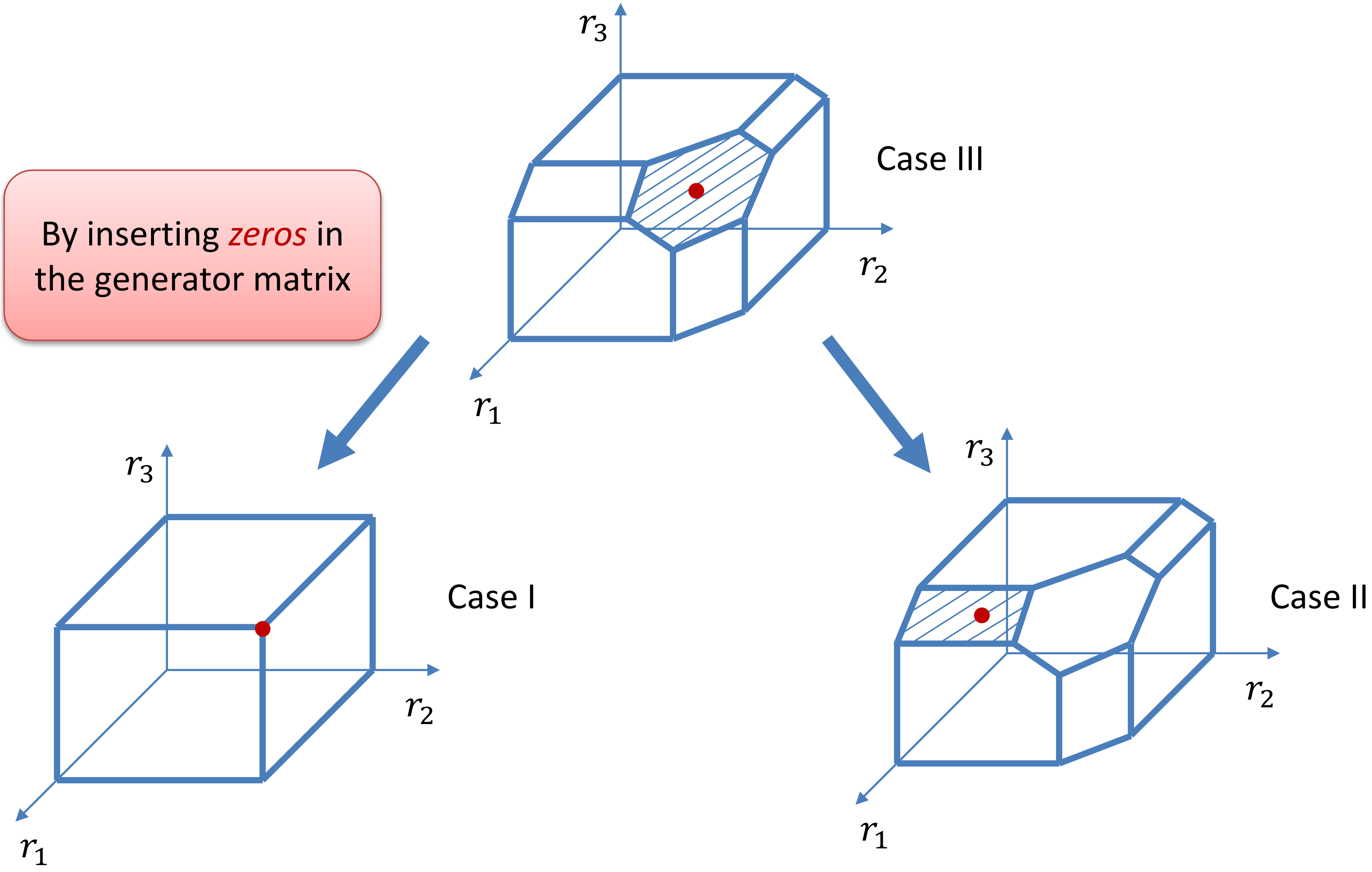
Keep adding zero columns until a set of inequalities becomes tight.



We can always ***add a column of all zeros*** to the i-th group of G_{RS} without violating the constraints.

$$G_{RS} = \begin{array}{c} r_1 \\ r_2 \\ r_3 \end{array} \begin{bmatrix} \times & 0 & 0 & \times & \times & \times & 0 & \times \\ 0 & \times & 0 & \times & 0 & 0 & \times & \times \\ 0 & 0 & \times & 0 & 0 & \times & \times & \times \end{bmatrix}$$

Rates on the boundary (Case III)



By inserting *zeros* in the generator matrix

Rates on the boundary (Case II)

Constraints:

$$r_1 \leq k - |\mathcal{Z}_1|$$

$$\vdots$$

$$r_i \leq k - |\mathcal{Z}_i|$$

$$\vdots$$

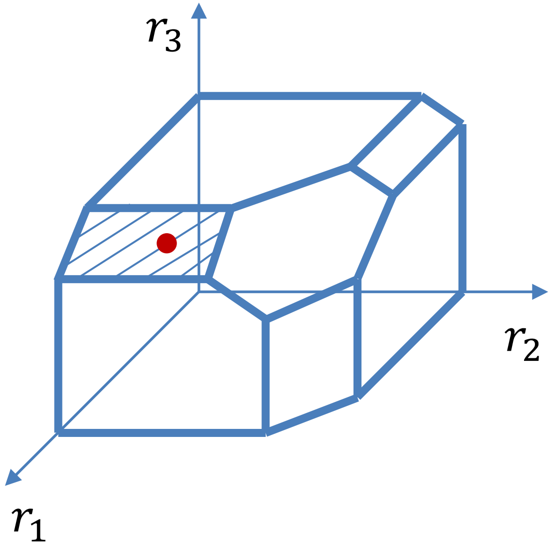
$$r_s \leq k - |\mathcal{Z}_s|$$

$$\vdots$$

$$\boxed{r_{i_1} + r_{i_2} + \cdots + r_{i_l} = k - |\mathcal{Z}_{i_1} \cap \mathcal{Z}_{i_2} \cap \cdots \cap \mathcal{Z}_{i_l}|}$$

$$\vdots$$

$$r_1 + r_2 + \cdots + r_s \leq k$$



$$G_{RS} = \begin{array}{c} r_1 \\ r_2 \\ r_3 \end{array} \begin{bmatrix} \times & 0 & 0 & \times & \times & 0 & \times \\ 0 & \times & 0 & \times & 0 & \times & \times \\ 0 & 0 & \times & 0 & \times & \times & \times \end{bmatrix}$$

Rates on the boundary (Case II)

Constraints:

$$r_1 \leq k - |\mathcal{Z}_1|$$

$$\vdots$$

$$r_i \leq k - |\mathcal{Z}_i|$$

$$\vdots$$

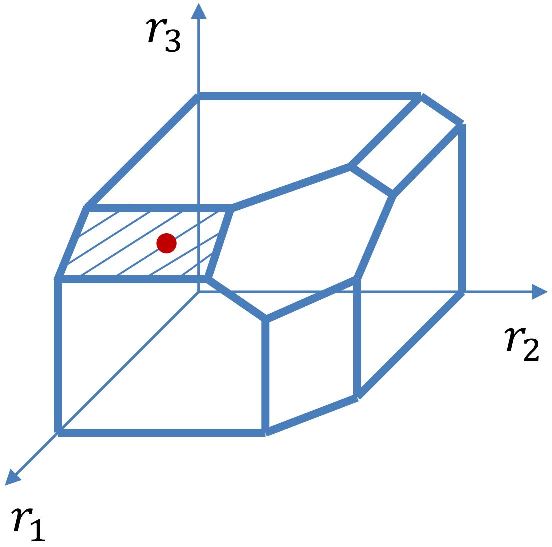
$$r_s \leq k - |\mathcal{Z}_s|$$

$$\vdots$$

$$\boxed{r_{i_1} + r_{i_2} + \cdots + r_{i_l} = k - |\mathcal{Z}_{i_1} \cap \mathcal{Z}_{i_2} \cap \cdots \cap \mathcal{Z}_{i_l}|}$$

$$\vdots$$

$$r_1 + r_2 + \cdots + r_s \leq k$$



$$G_{RS} = \begin{array}{c} r_1 \\ r_2 \\ r_3 \end{array} \left[\begin{array}{ccccccc} \times & 0 & 0 & \times & \times & 0 & \times \\ 0 & \times & 0 & \times & 0 & \times & \times \\ 0 & 0 & \times & 0 & \times & \times & \times \end{array} \right]$$

Rates on the boundary (Case II)

$$G_{RS} = \begin{array}{c} r_1 \\ r_2 \\ r_3 \end{array} \begin{bmatrix} \times & 0 & 0 & \times & \times & 0 & \times \\ 0 & \times & 0 & \times & 0 & \times & \times \\ 0 & 0 & \times & 0 & \times & \times & \times \end{bmatrix}$$

$$r_1 + r_2 = k - |\mathcal{Z}_1 \cap \mathcal{Z}_2|$$

Rates on the boundary (Case II)

$$G_{RS} = \begin{array}{c|ccccccc|} r_1 & \times & 0 & 0 & \times & \times & 0 & \times \\ r_2 & 0 & \times & 0 & \times & 0 & \times & \times \\ r_3 & 0 & 0 & \times & 0 & \times & \times & \times \end{array}$$

$$\underset{\underset{r_{12}}{\shortparallel}}{r_1 + r_2} = k - |\underset{\underset{\mathcal{Z}_{12}}{\shortparallel}}{\mathcal{Z}_1 \cap \mathcal{Z}_2}|$$

Create *two* new problems

Problem 1:

$$G_{RS}^{(1)} = \begin{array}{c|ccccccc|} r_1 & \times & 0 & 0 & \times & \times & 0 & \times \\ r_2 & 0 & \times & 0 & \times & 0 & \times & \times \\ & & & & & & & \end{array}$$

$$r_1 \leq k - |\mathcal{Z}_1|$$

$$r_2 \leq k - |\mathcal{Z}_2|$$

$$r_1 + r_2 = k - |\mathcal{Z}_1 \cap \mathcal{Z}_2|$$

Problem 2:

$$G_{RS}^{(2)} = \begin{array}{c|ccccccc|} r_{12} & \times & \times & 0 & \times & \times & \times & \times \\ r_3 & 0 & 0 & \times & 0 & \times & \times & \times \end{array}$$

$$r_{12} = k - |\mathcal{Z}_{12}|$$

$$r_3 \leq k - |\mathcal{Z}_3|$$

$$r_{12} + r_3 \leq k - |\mathcal{Z}_{12} \cap \mathcal{Z}_3|$$

Rates on the boundary (Case II)

$$G_{RS} = \begin{array}{c} r_1 \\ r_2 \\ r_3 \end{array} \begin{bmatrix} \times & 0 & 0 & \times & \times & 0 & \times \\ 0 & \times & 0 & \times & 0 & \times & \times \\ 0 & 0 & \times & 0 & \times & \times & \times \end{bmatrix}$$

$$\underset{\underset{r_{12}}{\shortparallel}}{r_1 + r_2} = k - |\underset{\underset{\mathcal{Z}_{12}}{\shortparallel}}{\mathcal{Z}_1 \cap \mathcal{Z}_2}|$$

Notice: These subspaces are ***identical***.

Problem 1:

$$G_{RS}^{(1)} = \begin{array}{c} r_1 \\ r_2 \\ \end{array} \begin{bmatrix} \times & 0 & 0 & \times & \times & 0 & \times \\ 0 & \times & 0 & \times & 0 & \times & \times \\ & & & & & & \end{bmatrix}$$

$$r_1 \leq k - |\mathcal{Z}_1|$$
$$r_2 \leq k - |\mathcal{Z}_2|$$
$$r_1 + r_2 = k - |\mathcal{Z}_1 \cap \mathcal{Z}_2|$$

Problem 2:

$$G_{RS}^{(2)} = \begin{array}{c} \\ r_{12} \\ r_3 \end{array} \begin{bmatrix} & & & & & & \\ \times & \times & 0 & \times & \times & \times & \times \\ 0 & 0 & \times & 0 & \times & \times & \times \end{bmatrix}$$

$$r_{12} = k - |\mathcal{Z}_{12}|$$
$$r_3 \leq k - |\mathcal{Z}_3|$$
$$r_{12} + r_3 \leq k - |\mathcal{Z}_{12} \cap \mathcal{Z}_3|$$

Rates on the boundary (Case II)

$$G_{RS} = \begin{array}{c|ccccccc|} r_1 & \times & 0 & 0 & \times & \times & 0 & \times \\ r_2 & 0 & \times & 0 & \times & 0 & \times & \times \\ r_3 & 0 & 0 & \times & 0 & \times & \times & \times \end{array}$$

$$\underset{\underset{r_{12}}{\shortparallel}}{r_1 + r_2} = k - |\underset{\underset{\mathcal{Z}_{12}}{\shortparallel}}{\mathcal{Z}_1 \cap \mathcal{Z}_2}|$$

Notice: These subspaces are ***identical***.

Problem 1:

$$G_{RS}^{(1)} = \begin{array}{c|ccccccc|} r_1 & \times & 0 & 0 & \times & \times & 0 & \times \\ r_2 & 0 & \times & 0 & \times & 0 & \times & \times \end{array}$$

$r_1 \le k - |\mathcal{Z}_1$

$r_2 \le k - |\mathcal{Z}_2$

$r_1 + r_2 = k - |\mathcal{Z}_1 \cap \mathcal{Z}_2|$

Problem 2:

$$G_{RS}^{(2)} = \begin{array}{c|ccccccc|} r_{12} & \times & \times & 0 & \times & \times & \times & \times \\ r_3 & 0 & 0 & \times & 0 & \times & \times & \times \end{array}$$

$r_{12} = k - |\mathcal{Z}_{12}|$

$r_3 \le k - |\mathcal{Z}_3|$

$r_{12} + r_3 \le k - |\mathcal{Z}_{12} \cap \mathcal{Z}_3|$

By induction, we can solve the sub-problems ☺

Rates on the boundary (Case I)

Constraints:

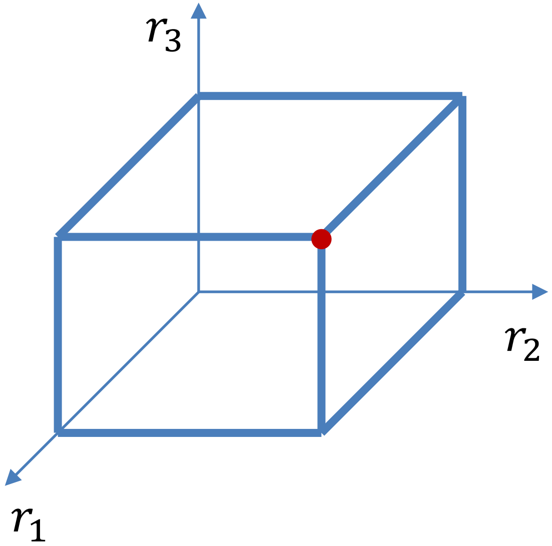
$$\boxed{\begin{array}{c} r_1 = k - |\mathcal{Z}_1| \\ \vdots \\ r_i = k - |\mathcal{Z}_i| \\ \vdots \\ r_s = k - |\mathcal{Z}_s| \end{array}}$$

$$\vdots$$

$$r_{i_1} + r_{i_2} + \cdots + r_{i_l} \leq k - |\mathcal{Z}_{i_1} \cap \mathcal{Z}_{i_2} \cap \cdots \cap \mathcal{Z}_{i_l}|$$

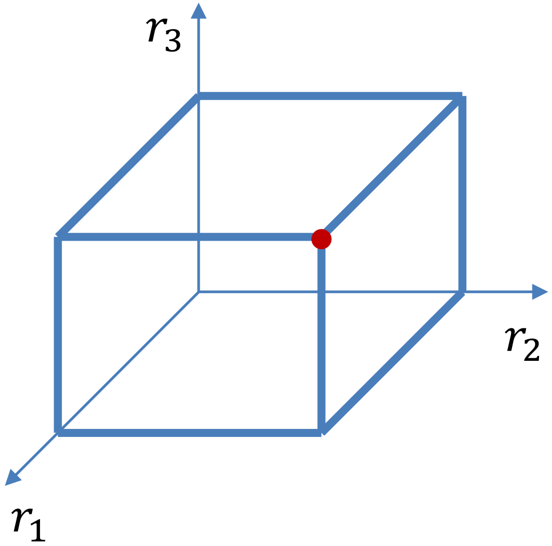
$$\vdots$$

$$r_1 + r_2 + \cdots + r_s \leq k$$



Case I: Example

Consider we are looking for a distributed RS code with length $n=6$ and dimension $k=3$ such that $r_1=r_2=r_3=1$ and the generator matrix has the following form:



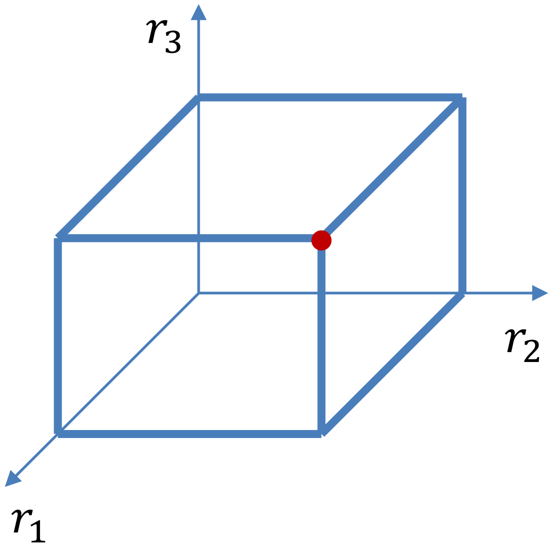
Evaluation points:

$$\begin{array}{cc} & \begin{array}{cccccc} \alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 & \alpha_5 & \alpha_6 \end{array} \\ G_{RS} = & \begin{bmatrix} \times & \times & \times & \times & 0 & 0 \\ \times & \times & 0 & 0 & \times & \times \\ 0 & 0 & \times & \times & \times & \times \end{bmatrix} \end{array}$$

So $r_i = k - |\mathcal{Z}_i| = 3 - 2 = 1, r_i + r_j \leq 3, r_1 + r_2 + r_3 \leq 3$

Case I: Example

Consider we are looking for a distributed RS code with length $n=6$ and dimension $k=3$ such that $r_1=r_2=r_3=1$ and the generator matrix has the following form:



Evaluation points:

$$\begin{array}{cc} & \begin{array}{cccccc} \alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 & \alpha_5 & \alpha_6 \end{array} \\ G_{RS} = & \begin{bmatrix} \times & \times & \times & \times & 0 & 0 \\ \times & \times & 0 & 0 & \times & \times \\ 0 & 0 & \times & \times & \times & \times \end{bmatrix} \end{array}$$

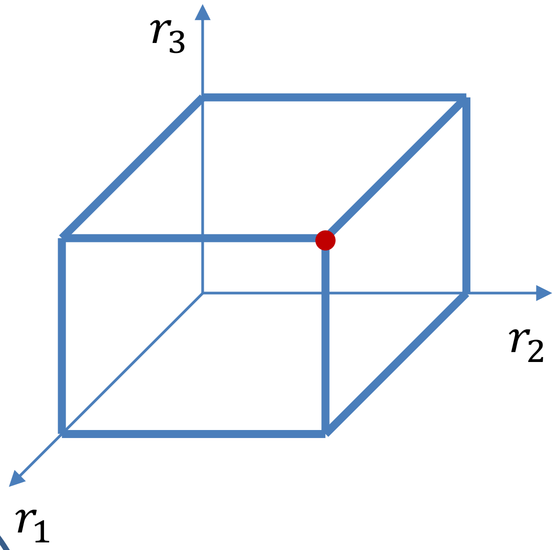
So $r_i = k - |\mathcal{Z}_i| = 3 - 2 = 1, r_i + r_j \leq 3, r_1 + r_2 + r_3 \leq 3$

General form of RS codewords from the first two rows:

$$c_{12}(x) = f_0(x-\alpha_5)(x-\alpha_6) + g_0(x-\alpha_3)(x-\alpha_4)$$

Case I: Example

Consider we are looking for a distributed RS code with length $n=6$ and dimension $k=3$ such that $r_1=r_2=r_3=1$ and the generator matrix has the following form:



$$\begin{array}{lc}
\text{Evaluation points:} & \begin{array}{cccccc} \alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 & \alpha_5 & \alpha_6 \end{array} \\
G_{RS} = & \begin{bmatrix} \times & \times & \times & \times & 0 & 0 \\ \times & \times & 0 & 0 & \times & \times \\ 0 & 0 & \times & \times & \times & \times \end{bmatrix}
\end{array}$$

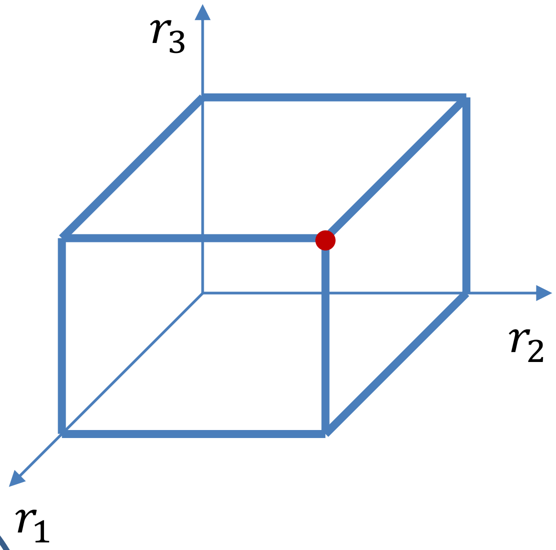
So $r_i = k - |\mathcal{Z}_i| = 3 - 2 = 1, r_i + r_j \leq 3, r_1 + r_2 + r_3 \leq 3$

General form of RS codewords from the first two rows:

$$c_{12}(x) = \boxed{f_0(x-\alpha_5)(x-\alpha_6)} + \boxed{g_0(x-\alpha_3)(x-\alpha_4)}$$

Case I: Example

Consider we are looking for a distributed RS code with length $n=6$ and dimension $k=3$ such that $r_1=r_2=r_3=1$ and the generator matrix has the following form:



$$\begin{array}{cc} \text{Evaluation points:} & \begin{array}{cccccc} \alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 & \alpha_5 & \alpha_6 \end{array} \\ G_{RS} = & \begin{bmatrix} \times & \times & \times & \times & 0 & 0 \\ \times & \times & 0 & 0 & \times & \times \\ 0 & 0 & \times & \times & \times & \times \end{bmatrix} \end{array}$$

So $r_i = k - |\mathcal{Z}_i| = 3 - 2 = 1, r_i + r_j \leq 3, r_1 + r_2 + r_3 \leq 3$

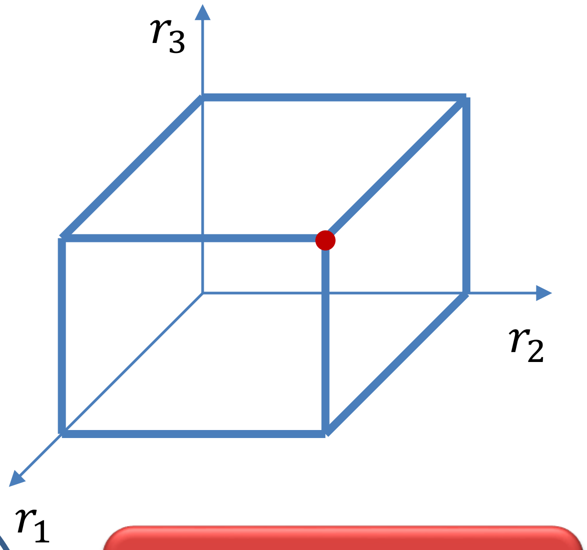
General form of RS codewords from the first two rows:

$$c_{12}(x) = \boxed{f_0(x-\alpha_5)(x-\alpha_6)} + \boxed{g_0(x-\alpha_3)(x-\alpha_4)}$$

If $\dfrac{(\alpha_1-\alpha_5)(\alpha_1-\alpha_6)}{(\alpha_2-\alpha_5)(\alpha_2-\alpha_6)} = \dfrac{(\alpha_1-\alpha_3)(\alpha_1-\alpha_4)}{(\alpha_2-\alpha_3)(\alpha_3-\alpha_4)} \Rightarrow \boxed{\exists\, f_0, g_0\colon\ c_{12}(\alpha_1) = c_{12}(\alpha_2) = 0}$

Case I: Example

Consider we are looking for a distributed RS code with length n=*6* and dimension k=*3* such that r_1=r_2=r_3=1 and the generator matrix has the following form:



Evaluation points:

$$\begin{array}{cc} & \begin{array}{cccccc} \alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 & \alpha_5 & \alpha_6 \end{array} \\ G_{RS} = & \begin{bmatrix} \times & \times & \times & \times & 0 & 0 \\ \times & \times & 0 & 0 & \times & \times \\ 0 & 0 & \times & \times & \times & \times \end{bmatrix} \end{array}$$

The generator matrix is **not full rank**! ☹

So $\quad r_i = k - |\mathcal{Z}_i| = 3 - 2 = 1, r_i + r_j \leq 3, r_1 + r_2 + r$

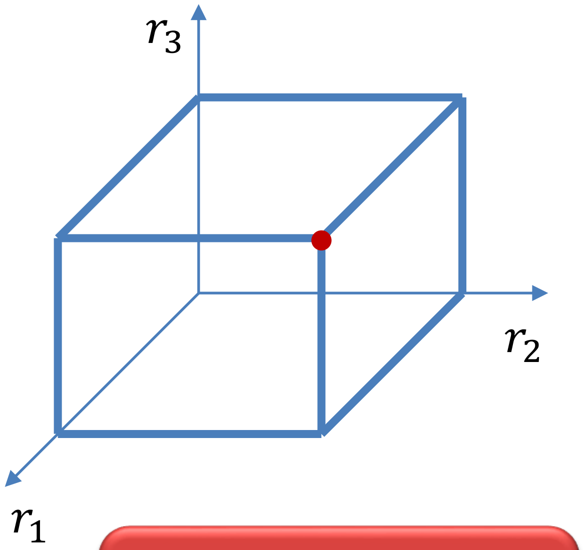
General form of RS codewords from the first two rows:

$$c_{12}(x) = \boxed{f_0(x-\alpha_5)(x-\alpha_6)} + \boxed{g_0(x-\alpha_3)(x-\alpha_4)}$$

If $\dfrac{(\alpha_1-\alpha_5)(\alpha_1-\alpha_6)}{(\alpha_2-\alpha_5)(\alpha_2-\alpha_6)} = \dfrac{(\alpha_1-\alpha_3)(\alpha_1-\alpha_4)}{(\alpha_2-\alpha_3)(\alpha_3-\alpha_4)} \Rightarrow \boxed{\exists\, f_0, g_0\colon\ c_{12}(\alpha_1) = c_{12}(\alpha_2) = 0}$

Case I: Example

Consider we are looking for a distributed RS code with length $n=6$ and dimension $k=3$ such that $r_1=r_2=r_3=1$ and the generator matrix has the following form:



$$\begin{array}{lcccccc} \text{Evaluation points:} & \alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 & \alpha_5 & \alpha_6 \\ G_{RS} = & \times & \times & \times & \times & 0 & 0 \\ & \times & \times & 0 & 0 & \times & \times \\ & 0 & 0 & \times & \times & \times & \times \end{array}$$

To have a counterexample, evaluation points should satisfy *specific* constraints!

S $r_i + r_j \leq 3, r_1 + r_2 + r$

C the first two rows:

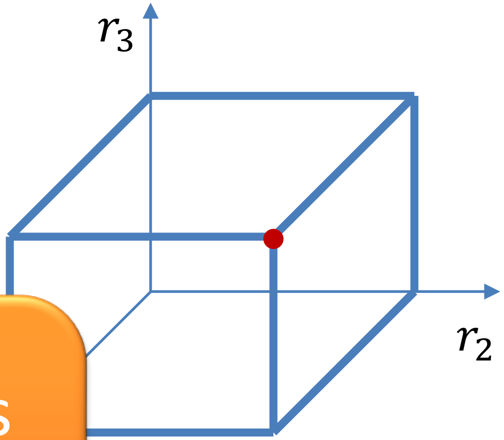
$$c_{12}(x) = f_0(x-\alpha_5)(x-\alpha_6) + g_0(x-\alpha_3)(x-\alpha_4)$$

The generator matrix is **not full rank**! ☹

If $\dfrac{(\alpha_1-\alpha_5)(\alpha_1-\alpha_6)}{(\alpha_2-\alpha_5)(\alpha_2-\alpha_6)} = \dfrac{(\alpha_1-\alpha_3)(\alpha_1-\alpha_4)}{(\alpha_2-\alpha_3)(\alpha_3-\alpha_4)} \Rightarrow \boxed{\exists\, f_0, g_0\colon\ c_{12}(\alpha_1) = c_{12}(\alpha_2) = 0}$

Case I: Example

Consider we are looking for a distributed RS code with length $n=6$ and dimension $k=3$ such that $r_1=r_2=r_3=1$ and the generator matrix has the following form:



Evaluation points: $\alpha_1 \quad \alpha_2 \quad \alpha_3 \quad \alpha_4 \quad \alpha_5 \quad \alpha_6$

$$G_{RS} = \begin{bmatrix} \times \\ \times \\ 0 \end{bmatrix}$$

Choose evaluation points of the distributed RS code ***carefully***.

To have a c

evaluation po

specific constraints!

S

C

he first two rows.

The generator matrix is **not full rank**! ☹

$$c_{12}(x) = f_0(x-\alpha_5)(x-\alpha_6) + g_0(x-\alpha_3)(x-\alpha_4)$$

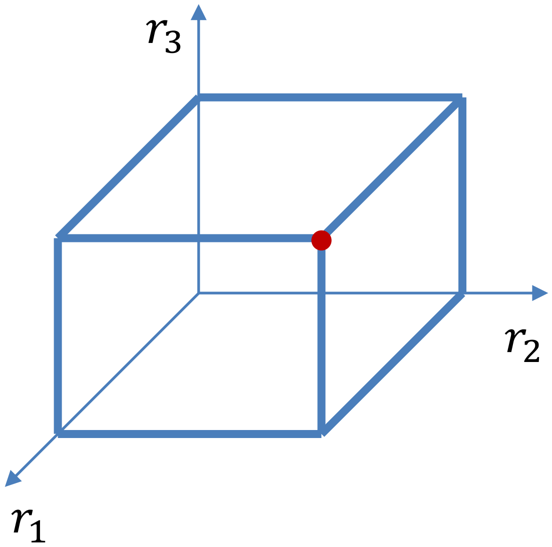
If $\boxed{\dfrac{(\alpha_1-\alpha_5)(\alpha_1-\alpha_6)}{(\alpha_2-\alpha_5)(\alpha_2-\alpha_6)} = \dfrac{(\alpha_1-\alpha_3)(\alpha_1-\alpha_4)}{(\alpha_2-\alpha_3)(\alpha_3-\alpha_4)}} \Rightarrow \boxed{\exists\, f_0, g_0\colon\ c_{12}(\alpha_1) = c_{12}(\alpha_2) = 0}$

Rates on the boundary (Case I)

Constraints:

$$\boxed{\begin{aligned} r_1 &= k - |\mathcal{Z}_1| \\ &\vdots \\ r_i &= k - |\mathcal{Z}_i| \\ &\vdots \\ r_s &= k - |\mathcal{Z}_s| \end{aligned}}$$

$$G_{RS} = \begin{array}{c} r_1 \\ r_2 \\ r_3 \end{array} \begin{bmatrix} \times & 0 & 0 & \times & \times & 0 & \times \\ 0 & \times & 0 & \times & 0 & \times & \times \\ 0 & 0 & \times & 0 & \times & \times & \times \end{bmatrix}$$



Theorem: There is always *a set of evaluation points* such that one can construct a full rank generator matrix in case I.

Rates on the boundary (Case I)

Theorem: There is always *a set of evaluation points* such that one can construct a full rank generator matrix in case I.

Proof by induction:

$$G_{RS} = \begin{array}{c} r_1 \\ r_2 \\ r_3 \end{array} \begin{bmatrix} \times & 0 & 0 & \times & \times & 0 & \times \\ 0 & \times & 0 & \times & 0 & \times & \times \\ 0 & 0 & \times & 0 & \times & \times & \times \end{bmatrix}$$

$$P_1(x) = \prod_{i \in Z_1} (x - \alpha_i)$$

$$P_2(x) = \prod_{i \in Z_2} (x - \alpha_i)$$

$c_1(x) = f(x)P_1(x)$
$\deg f(x) \leq r_1 - 1$

$c_2(x) = g(x)P_2(x)$
$\deg g(x) \leq r_2 - 1$

Rates on the boundary (Case I)

Theorem: There is always *a set of evaluation points* such that one can construct a full rank generator matrix in case I.

Proof by induction:

$$G_{RS} = \begin{array}{c} r_1 \\ r_2 \\ r_3 \end{array} \begin{bmatrix} \times & 0 & 0 & \times & \times & 0 & \times \\ 0 & \times & 0 & \times & 0 & \times & \times \\ 0 & 0 & \times & 0 & \times & \times & \times \end{bmatrix}$$

$$P_1(x) = \prod_{i \in \mathcal{Z}_1} (x - \alpha_i)$$

$$P_2(x) = \prod_{i \in \mathcal{Z}_2} (x - \alpha_i)$$

$c_1(x) = f(x)P_1(x)$
$\deg f(x) \leq r_1 - 1$

$c_2(x) = g(x)P_2(x)$
$\deg g(x) \leq r_2 - 1$

$\{\alpha_{i_1}, \dots, \alpha_{|\mathcal{Z}_3|}\} \in \mathcal{Z}_3$

$$|\mathcal{Z}_3| \overbrace{\begin{bmatrix} P_1(\alpha_{i_1}) & \cdots & \alpha_{i_1}^{r_1-1} P_1(\alpha_{i_1}) \\ \vdots & \ddots & \vdots \\ P_1(\alpha_{i_{|\mathcal{Z}_3|}}) & \cdots & \alpha_{i_{|\mathcal{Z}_3|}}^{r_1-1} P_1(\alpha_{i_{|\mathcal{Z}_3|}}) \end{bmatrix}}^{r_1} \overbrace{\begin{matrix} P_2(\alpha_{i_1}) & \cdots & \alpha_{i_1}^{r_2-1} P_2(\alpha_{i_1}) \\ \vdots & \ddots & \vdots \\ P_2(\alpha_{i_{|\mathcal{Z}_3|}}) & \cdots & \alpha_{i_{|\mathcal{Z}_3|}}^{r_2-1} P_2(\alpha_{i_{|\mathcal{Z}_3|}}) \end{matrix}}^{r_2} \begin{bmatrix} f_0 \\ \vdots \\ f_{r_1-1} \\ g_0 \\ \vdots \\ g_{r_2-1} \end{bmatrix} = 0$$

Rates on the boundary (Case I)

Theorem: There is always *a set of evaluation points* such that one can always construct a full rank generator matrix in case I.

Proof by induction:

$$\begin{array}{c|cccccc}
 & \multicolumn{3}{c}{\overset{r_1}{\longleftrightarrow}} & \multicolumn{3}{c}{\overset{r_2}{\longleftrightarrow}} \\
|\mathcal{Z}_3| \updownarrow & P_1(\alpha_{i_1}) & \cdots & \alpha_{i_1}^{r_1-1}P_1(\alpha_{i_1}) & P_2(\alpha_{i_1}) & \cdots & \alpha_{i_1}^{r_2-1}P_2(\alpha_{i_1}) \\
 & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
 & P_1(\alpha_{i_{|\mathcal{Z}_3|}}) & \cdots & \alpha_{i_{|\mathcal{Z}_3|}}^{r_1-1}P_1(\alpha_{i_{|\mathcal{Z}_3|}}) & P_2(\alpha_{i_{|\mathcal{Z}_3|}}) & \cdots & \alpha_{i_{|\mathcal{Z}_3|}}^{r_2-1}P_2(\alpha_{i_{|\mathcal{Z}_3|}})
\end{array}
\begin{bmatrix} f_0 \\ \vdots \\ f_{r_1-1} \\ g_0 \\ \vdots \\ g_{r_2-1} \end{bmatrix} = 0$$

$$r_1 + r_2 + r_3 \le k, r_3 = k - |\mathcal{Z}_3| \Rightarrow |\mathcal{Z}_3| \ge r_1 + r_2$$

Rates on the boundary (Case I)

Theorem: There is always *a set of evaluation points* such that one can construct a full rank generator matrix in case I.

Proof by induction:

$$\begin{array}{c|cccccc|}
 & \multicolumn{3}{c}{\xleftrightarrow{\quad r_1 \quad}} & \multicolumn{3}{c}{\xleftrightarrow{\quad r_2 \quad}} \\
 & P_1(\alpha_{i_1}) & \cdots & \alpha_{i_1}^{r_1-1}P_1(\alpha_{i_1}) & P_2(\alpha_{i_1}) & \cdots & \alpha_{i_1}^{r_2-1}P_2(\alpha_{i_1}) \\
|\mathcal{Z}_3| & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
 & P_1(\alpha_{i_{|\mathcal{Z}_3|}}) & \cdots & \alpha_{i_{|\mathcal{Z}_3|}}^{r_1-1}P_1(\alpha_{i_{|\mathcal{Z}_3|}}) & P_2(\alpha_{i_{|\mathcal{Z}_3|}}) & \cdots & \alpha_{i_{|\mathcal{Z}_3|}}^{r_2-1}P_2(\alpha_{i_{|\mathcal{Z}_3|}})
\end{array}
\begin{bmatrix} f_0 \\ \vdots \\ f_{r_1-1} \\ g_0 \\ \vdots \\ g_{r_2-1} \end{bmatrix} = 0$$

$$r_1 + r_2 + r_3 \leq k, r_3 = k - |\mathcal{Z}_3| \Rightarrow |\mathcal{Z}_3| \geq r_1 + r_2$$

$$M_{(r_1+r_2)\times(r_1+r_2)} = \begin{bmatrix} P_1(y_1) & \cdots & y_1^{r_1-1}P_1(y_1) & P_2(y_1) & \cdots & y_1^{r_2-1}P_2(y_1) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ P_1(y_{r_1+r_2}) & \cdots & y_{r_1+r_2}^{r_1-1}P_1(y_{r_1+r_2}) & P_2(y_{r_1+r_2}) & \cdots & y_{r_1+r_2}^{r_2-1}P_2(y_{r_1+r_2}) \end{bmatrix}$$

Rates on the boundary (Case I)

Theorem: There is always *a set of evaluation points* such that one can construct a full rank generator matrix in case I.

Proof by induction:

$$|\mathcal{Z}_3| \left[\begin{array}{cccccc} \multicolumn{3}{c}{\overleftrightarrow{\quad r_1 \quad}} & \multicolumn{3}{c}{\overleftrightarrow{\quad r_2 \quad}} \\ P_1(\alpha_{i_1}) & \cdots & \alpha_{i_1}^{r_1-1}P_1(\alpha_{i_1}) & P_2(\alpha_{i_1}) & \cdots & \alpha_{i_1}^{r_2-1}P_2(\alpha_{i_1}) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ P_1(\alpha_{i_{|\mathcal{Z}_3|}}) & \cdots & \alpha_{i_{|\mathcal{Z}_3|}}^{r_1-1}P_1(\alpha_{i_{|\mathcal{Z}_3|}}) & P_2(\alpha_{i_{|\mathcal{Z}_3|}}) & \cdots & \alpha_{i_{|\mathcal{Z}_3|}}^{r_2-1}P_2(\alpha_{i_{|\mathcal{Z}_3|}}) \end{array}\right] \begin{bmatrix} f_0 \\ \vdots \\ f_{r_1-1} \\ g_0 \\ \vdots \\ g_{r_2-1} \end{bmatrix} = 0$$

$$r_1 + r_2 + r_3 \le k, r_3 = k - |\mathcal{Z}_3| \Rightarrow |\mathcal{Z}_3| \ge r_1 + r_2$$

$$M_{(r_1+r_2)\times(r_1+r_2)} = \begin{bmatrix} P_1(y_1) & \cdots & y_1^{r_1-1}P_1(y_1) & P_2(y_1) & \cdots & y_1^{r_2-1}P_2(y_1) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ P_1(y_{r_1+r_2}) & \cdots & y_{r_1+r_2}^{r_1-1}P_1(y_{r_1+r_2}) & P_2(y_{r_1+r_2}) & \cdots & y_{r_1+r_2}^{r_2-1}P_2(y_{r_1+r_2}) \end{bmatrix}$$

By induction $\exists \alpha_1, \dots, \alpha_{r_1+r_2}$: $\det M(\alpha_1, \dots, \alpha_{r_1+r_2}) \neq 0 \Rightarrow \det M(y_1, \dots, y_{r_1+r_2}) \neq 0$

Rates on the boundary (Case I)

Theorem: There is always *a set of evaluation points* such that one can construct a full rank generator matrix in case I.

Proof by induction:

$$\begin{array}{cc}
 & \overset{r_1}{\longleftrightarrow} \qquad\qquad\qquad\qquad \overset{r_2}{\longleftrightarrow} \\
|\mathcal{Z}_3| \updownarrow & \begin{bmatrix}
P_1(\alpha_{i_1}) & \cdots & \alpha_{i_1}^{r_1-1} P_1(\alpha_{i_1}) & P_2(\alpha_{i_1}) & \cdots & \alpha_{i_1}^{r_2-1} P_2(\alpha_{i_1}) \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
P_1(\alpha_{i_{|\mathcal{Z}_3|}}) & \cdots & \alpha_{i_{|\mathcal{Z}_3|}}^{r_1-1} P_1(\alpha_{i_{|\mathcal{Z}_3|}}) & P_2(\alpha_{i_{|\mathcal{Z}_3|}}) & \cdots & \alpha_{i_{|\mathcal{Z}_3|}}^{r_2-1} P_2(\alpha_{i_{|\mathcal{Z}_3|}})
\end{bmatrix}
\end{array}
\begin{bmatrix} f_0 \\ \vdots \\ f_{r_1-1} \\ g_0 \\ \vdots \\ g_{r_2-1} \end{bmatrix} = 0$$

$$r_1 + r_2 + r_3 \leq k, r_3 = k - |\mathcal{Z}_3| \Rightarrow |\mathcal{Z}_3| \geq r_1 + r_2$$

$$M_{(r_1+r_2)\times(r_1+r_2)} = \begin{bmatrix}
P_1(y_1) & \cdots & y_1^{r_1-1} P_1(y_1) & P_2(y_1) & \cdots & y_1^{r_2-1} P_2(y_1) \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
P_1(y_{r_1+r_2}) & \cdots & y_{r_1+r_2}^{r_1-1} P_1(y_{r_1+r_2}) & P_2(y_{r_1+r_2}) & \cdots & y_{r_1+r_2}^{r_2-1} P_2(y_{r_1+r_2})
\end{bmatrix}$$

By induction $\exists \alpha_1, \dots, \alpha_{r_1+r_2}$: $\det M\left(\alpha_1, \dots, \alpha_{r_1+r_2}\right) \neq 0 \Rightarrow \det M\left(y_1, \dots, y_{r_1+r_2}\right) \neq 0$

Choose $\alpha_{i_1}, \dots, \alpha_{i_{|\mathcal{Z}_3|}}$ such that $\det M\left(\alpha_{i_1}, \dots, \alpha_{i_{r_1+r_2}}\right) \neq 0$

Rates on the boundary (Case I)

$$\begin{array}{cc} & \xleftrightarrow{\qquad\qquad r_1 + \cdots + r_l \qquad\qquad} \\ n \; \updownarrow & \begin{bmatrix} P_1(y_1) & \cdots & y_1^{r_1-1}P_1(y_1) & P_2(y_1) & \cdots & y_1^{r_2-1}P_2(y_1) & \cdots \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \cdots \\ P_1(y_n) & \cdots & y_n^{r_1-1}P_1(y_n) & P_2(y_n) & \cdots & y_n^{r_2-1}P_2(y_n) & \cdots \end{bmatrix} \end{array} \begin{bmatrix} f_0 \\ \vdots \\ f_{r_1-1} \\ g_0 \\ \vdots \\ g_{r_2-1} \\ \vdots \end{bmatrix} = \begin{bmatrix} c_1 \\ \vdots \\ c_n \end{bmatrix}$$

Rates on the boundary (Case I)

$$
\begin{array}{cc}
 & r_1+\cdots+r_l \\
 & M \\
n & \begin{bmatrix}
P_1(y_1) & \cdots & y_1^{r_1-1}P_1(y_1) & P_2(y_1) & \cdots & y_1^{r_2-1}P_2(y_1) & \cdots \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \cdots \\
P_1(y_n) & \cdots & y_n^{r_1-1}P_1(y_n) & P_2(y_n) & \cdots & y_n^{r_2-1}P_2(y_n) & \cdots
\end{bmatrix}
\end{array}
\begin{bmatrix} f_0 \\ \vdots \\ f_{r_1-1} \\ g_0 \\ \vdots \\ g_{r_2-1} \\ \vdots \end{bmatrix}
=
\begin{bmatrix} c_1 \\ \vdots \\ c_n \end{bmatrix}
$$

We just showed that there exist a matrix M such that

$$\det M = h(y_1, \ldots, y_n) \neq 0$$

Rates on the boundary (Case I)

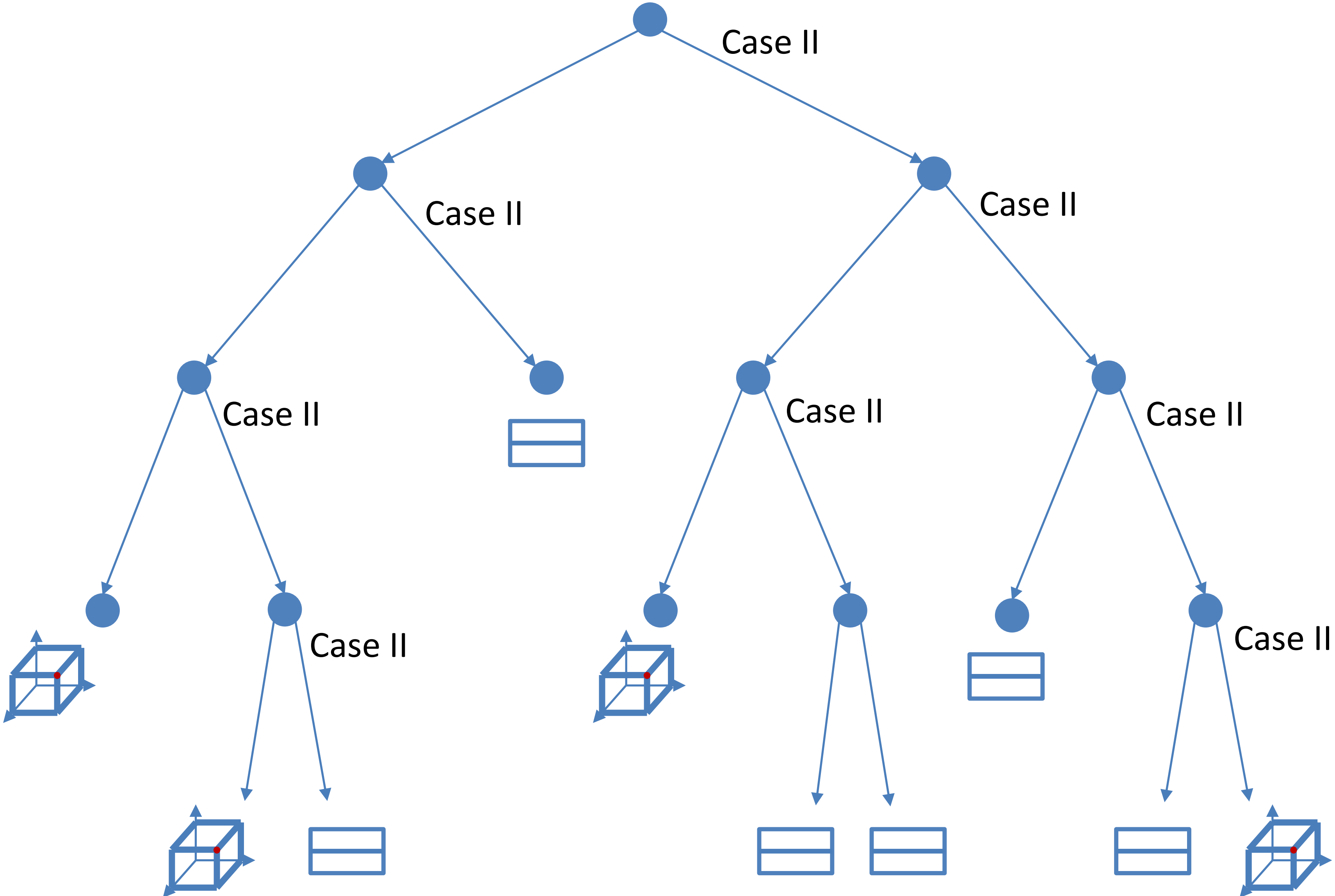
$$\begin{array}{cc} & r_1+\cdots+r_l \\ & M \\ n & \begin{bmatrix} P_1(y_1) & \cdots & y_1^{r_1-1}P_1(y_1) & P_2(y_1) & \cdots & y_1^{r_2-1}P_2(y_1) & \cdots \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \cdots \\ P_1(y_n) & \cdots & y_n^{r_1-1}P_1(y_n) & P_2(y_n) & \cdots & y_n^{r_2-1}P_2(y_n) & \cdots \end{bmatrix} \end{array} \begin{bmatrix} f_0 \\ \vdots \\ f_{r_1-1} \\ g_0 \\ \vdots \\ g_{r_2-1} \\ \vdots \end{bmatrix} = \begin{bmatrix} c_1 \\ \vdots \\ c_n \end{bmatrix}$$

We just showed that there exist a matrix M such that

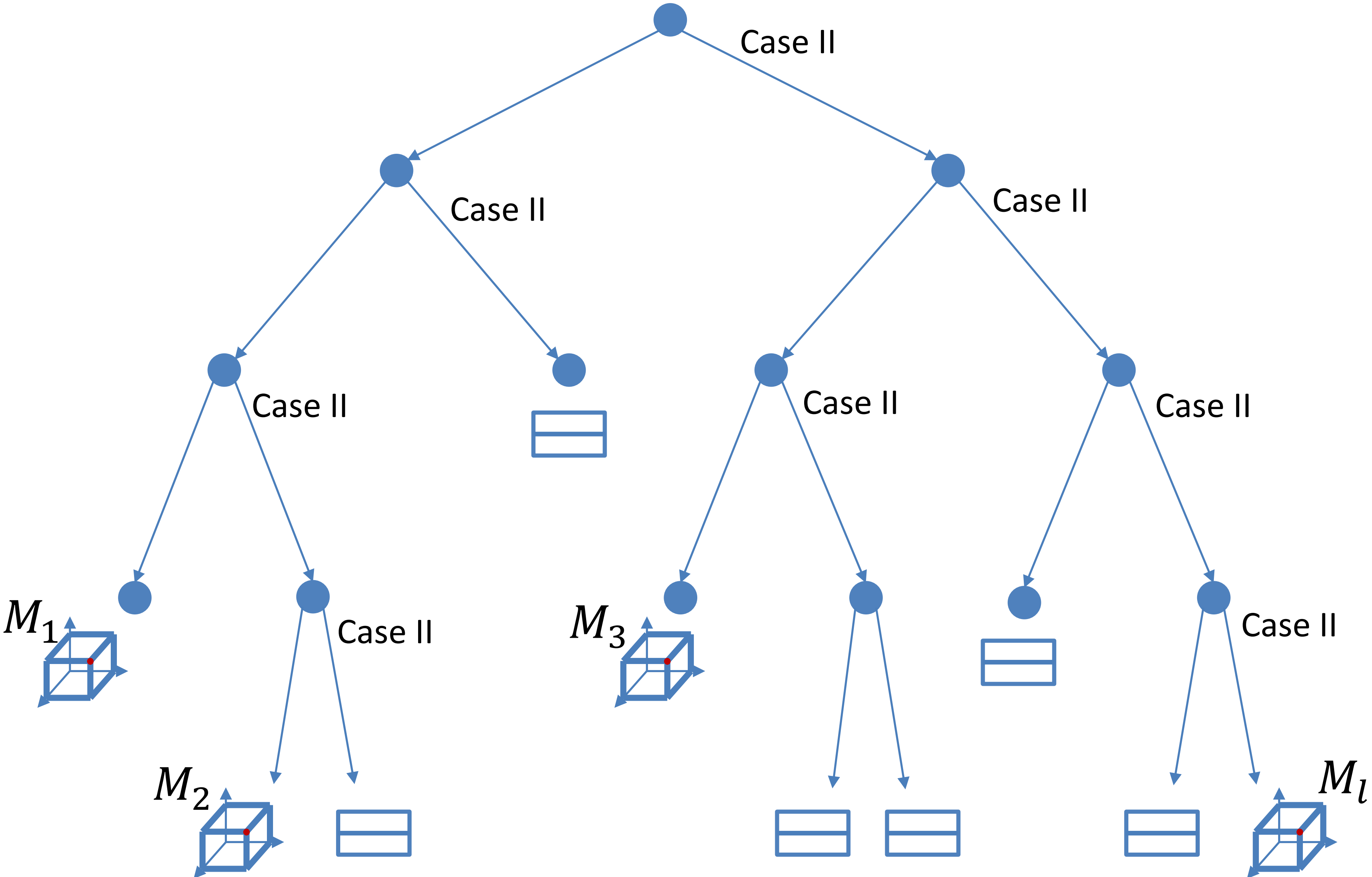
$$\det M = h(y_1, \dots, y_n) \neq 0$$

choose evaluation points $(\alpha_1, \dots, \alpha_n)$ such that $h(\alpha_1, \dots, \alpha_n) \neq 0$

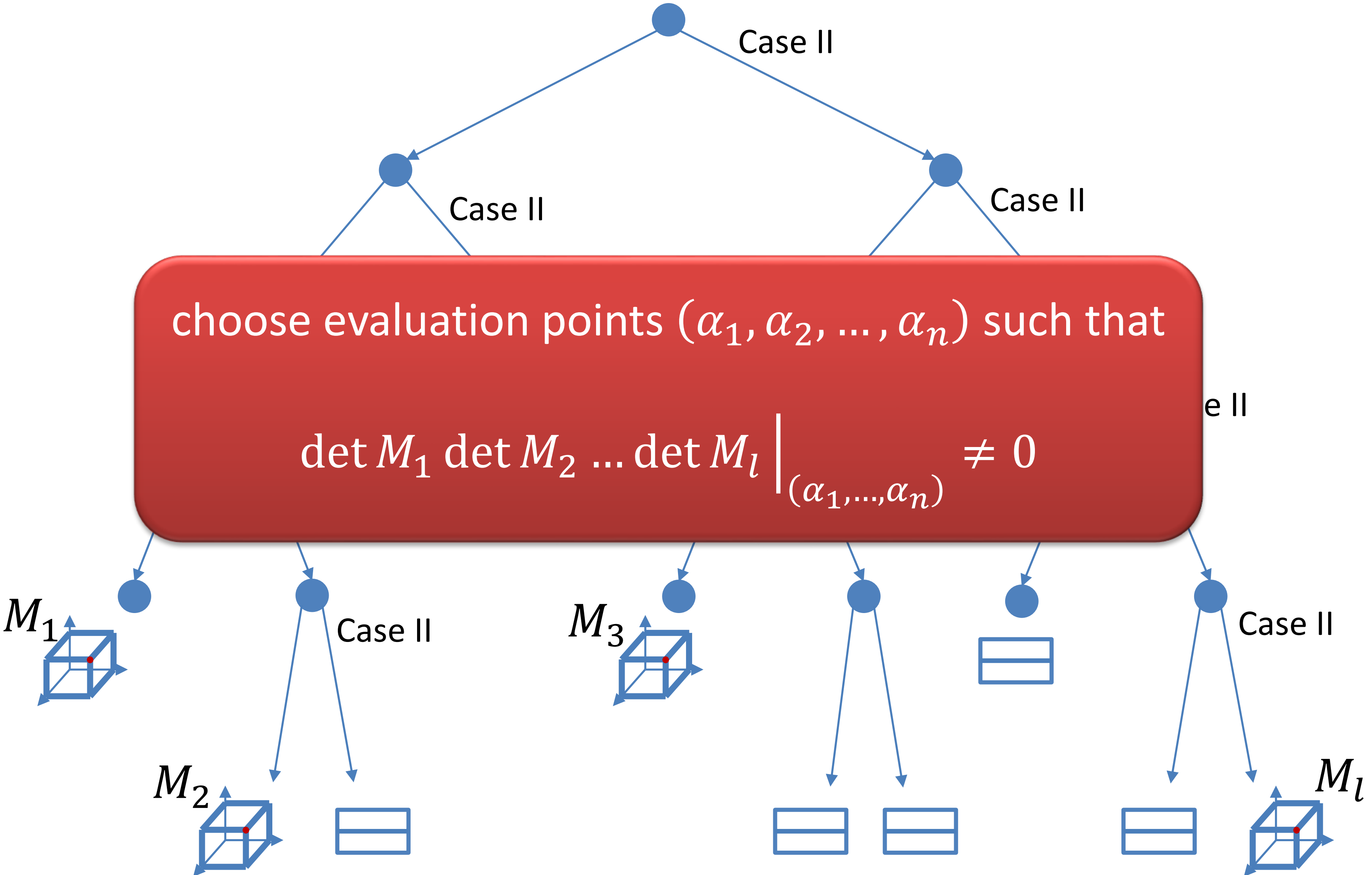
Induction



Induction


Case II
Case II
Case II
Case II
Case II
Case II
Case II
Case II
M_1
M_2
M_3
M_l

Induction



choose evaluation points $(\alpha_1, \alpha_2, \ldots, \alpha_n)$ such that

$$\det M_1 \det M_2 \ldots \det M_l \Big|_{(\alpha_1,\ldots,\alpha_n)} \neq 0$$

Size of the required finite-field

$$G_{RS} = \begin{array}{c} \overset{n}{\longleftrightarrow} \\ \left[\begin{array}{ccc} f_1(\alpha_1)P_1(\alpha_1) & \cdots & f_1(\alpha_n)P_1(\alpha_n) \\ \vdots & \ddots & \vdots \\ f_{r_1}(\alpha_1)P_1(\alpha_1) & \cdots & f_{r_1}(\alpha_n)P_1(\alpha_n) \\ g_1(\alpha_1)P_2(\alpha_1) & \cdots & g_1(\alpha_n)P_2(\alpha_n) \\ \vdots & \ddots & \vdots \\ g_{r_2}(\alpha_1)P_2(\alpha_1) & \cdots & g_{r_2}(\alpha_n)P_2(\alpha_n) \\ \vdots & \ddots & \vdots \end{array}\right] \end{array} \begin{array}{l} \\ \\ \updownarrow r_1 \\ \\ \\ \updownarrow r_2 \\ \\ \\ \end{array}$$

Size of the required finite-field

$$G_{RS} = \begin{array}{cc} \overset{n}{\longleftrightarrow} & \\ \begin{bmatrix} f_1(\alpha_1)P_1(\alpha_1) & \cdots & f_1(\alpha_n)P_1(\alpha_n) \\ \vdots & \ddots & \vdots \\ f_{r_1}(\alpha_1)P_1(\alpha_1) & \cdots & f_{r_1}(\alpha_n)P_1(\alpha_n) \\ g_1(\alpha_1)P_2(\alpha_1) & \cdots & g_1(\alpha_n)P_2(\alpha_n) \\ \vdots & \ddots & \vdots \\ g_{r_2}(\alpha_1)P_2(\alpha_1) & \cdots & g_{r_2}(\alpha_n)P_2(\alpha_n) \\ \vdots & \ddots & \vdots \end{bmatrix} & \begin{array}{c} \updownarrow\, r_1 \\ \\ \\ \updownarrow\, r_2 \\ \\ \\ \end{array} \end{array}$$

$M_{(r_1+\cdots+r_s)\times(r_1+\cdots+r_s)}$ Full rank

Choose evaluation points such that: $\det M \neq 0$

$$\deg\det M \leq k(k-1), \max_{\mathrm{i}} \deg \alpha_i \leq k-1$$

Extended Schwartz-Zippel Lemma: If size of the finite-field is larger than or equal to n, then there are sets of evaluation points that satisfy the inequality.

Future work

- We can construct a randomized algorithm that finds G_{RS}. Find an efficient determinist algorithm for the problem.
- Extend the results to general multiple-source networks (Gabidulin codes)
- Look for applications in storage systems