# Error Control for Linear and Affine Network Coding

Maximilien Gadouleau
Joint work with Alban Goupil

CReSTIC
Université de Reims Champagne-Ardenne

March 4, 2010

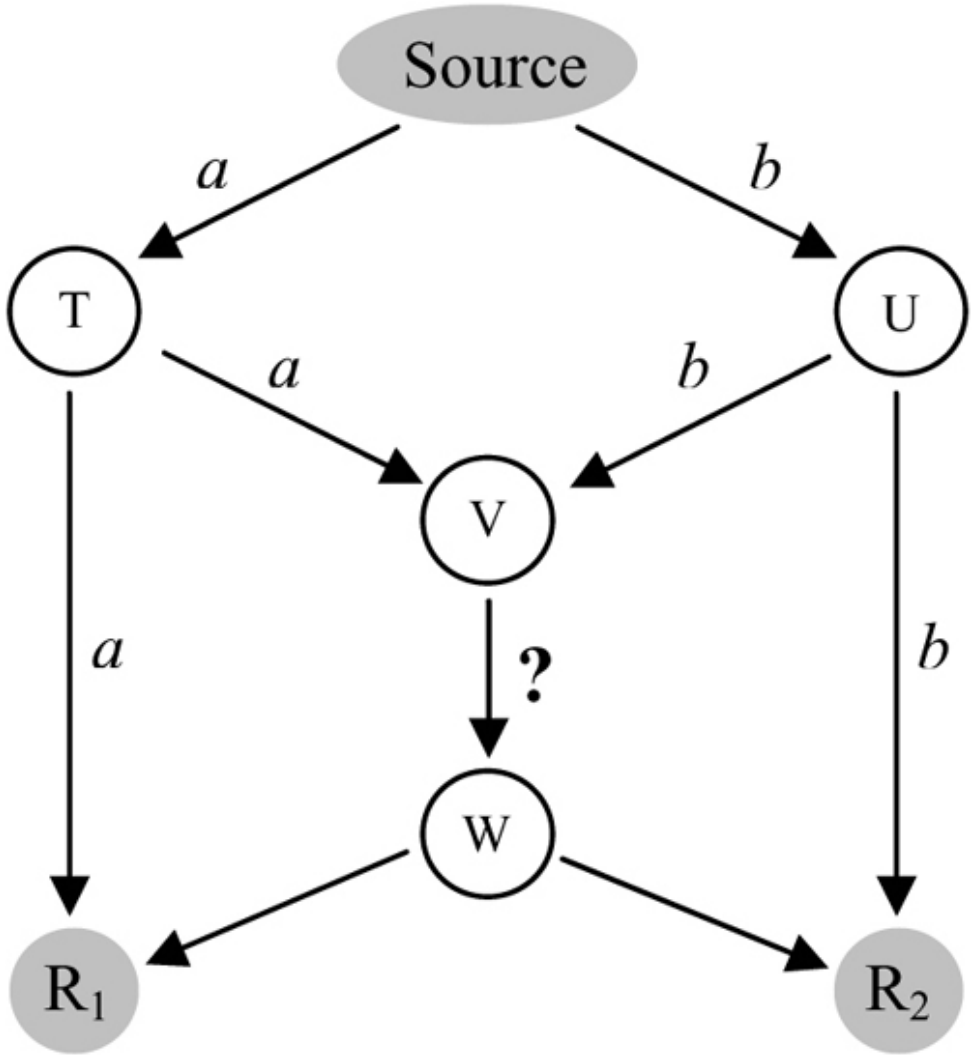
# Outline

# Outline

# Network coding
# (Yeung and Zhang 99, Ahlswede *et al.* 00)

- Routing does not reach max throughput for mulitcast
- Network coding: intermediate nodes combine packets
- Examples
  - Packet selection: routing
  - Linear combinations: Linear network coding
- Advantages
  - Higher throughput for multicast
  - Greater adaptability to topology changes
  - Robustness to packet losses

# Network coding example



# Random linear network coding

- Fixed linear combinations: too complex, too rigid
  $\Rightarrow$ Solution: choose the linear combinations randomly
- Success probability tends to 1 with field size (Kötter and Médard 02)
- Header to make LI and record combinations ('lifting')

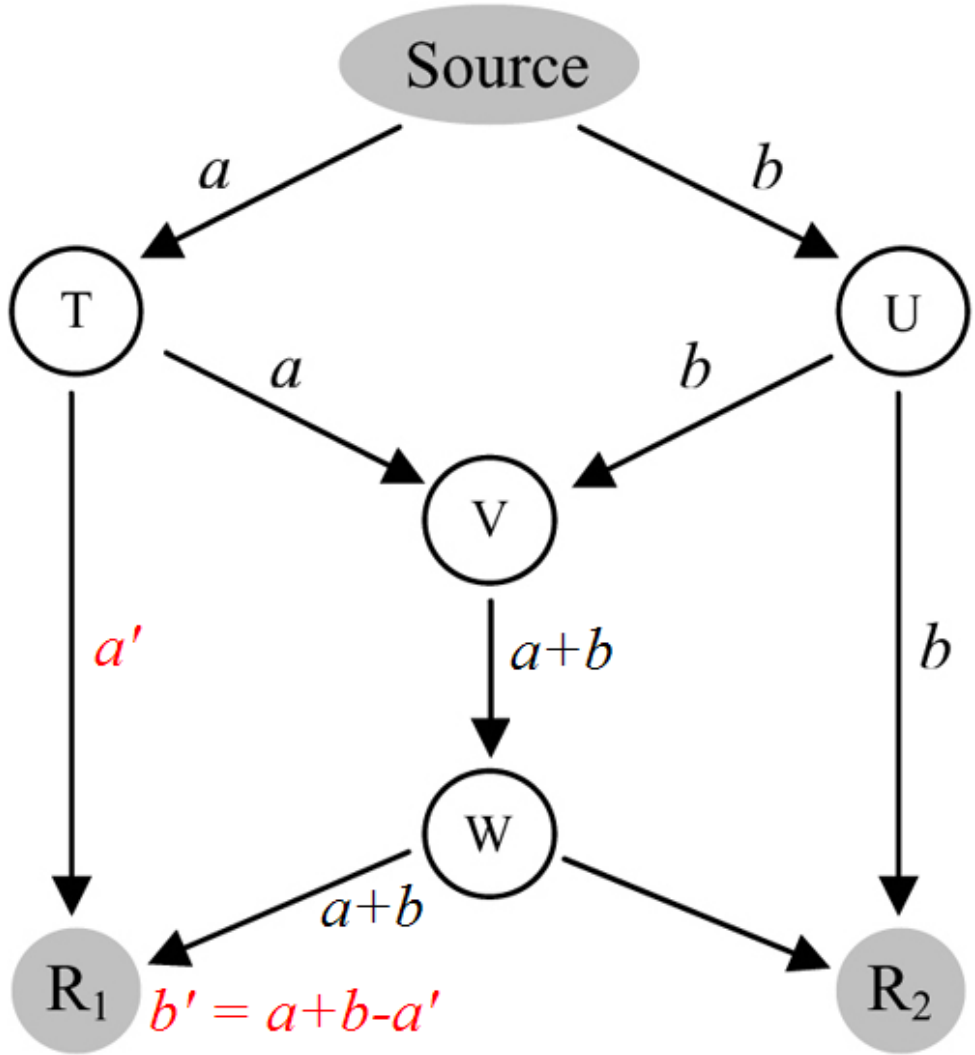
$$a, b, c, d \rightarrow \left(\begin{array}{cccc|c} 1 & 0 & 0 & 0 & a \\ 0 & 1 & 0 & 0 & b \\ 0 & 0 & 1 & 0 & c \\ 0 & 0 & 0 & 1 & d \end{array}\right) = (\mathbf{I}_4|\mathbf{M})$$

- Easy decoding: we receive $(\mathbf{L}|\mathbf{LM})$, we compute $\mathbf{L}^{-1}(\mathbf{LM})$

# Error control for RLNC

- RLNC sensitive to errors for two reasons
- 1. Different types of errors
  - Faulty links
  - Insufficient field size
  - Faulty or malevolent nodes
  - Adversary on the network
- 2. Error propagation: a packet in error can corrupt all packets after linear combination
- Hamming metric codes unadapted to a noncoherent approach

# Error in our example



# Operator channel (Kötter and Kschischang 08)

- RLNC preserves row space of the transmitted matrix
  $\Rightarrow$ Modeled as the transmission of a linear subspace
- Two ways to modify a subspace
  - Erasure: removal of an existing dimension
  - Disclosure: addition of a new dimension
- Operator channel: send $U$, receive

$$V = E(U) \oplus D$$

  $E$: erasures, $D$: disclosures
- Error control in RLNC is a coding problem, where codewords are subspaces

# Subspace codes
# (Delsarte 76, Schwartz and Etzion 02, K. and K. 08)

- Constant-dimension code (CDC): set of linear subspaces of $\mathrm{GF}(q)^n$ with equal dimension $k$
- Advantages
  - Non-ambiguous decoding
  - Simplified decoding procedure
- Metric: subspace distance

$$\begin{aligned} d_S(U,V) &= \sharp\text{erasures} + \sharp\text{disclosures} \\ &= 2\dim(U+V) - \dim(U) - \dim(V) \end{aligned}$$

- Nearly optimal codes based on liftings of rank metric codes

# Rank metric codes
# (Delsarte 78, Gabidulin 85, Roth 91)

- Rank metric code: set of matrices in $\mathrm{GF}(q)^{m\times n}$
  Rank distance
  $$d_{\mathrm{R}}(\mathbf{M},\mathbf{N}) = \mathrm{rank}(\mathbf{M}-\mathbf{N})$$
- Applications
  - Data storage
  - Cryptography
  - Space-time coding
  - ISI channels, etc
- Singleton bound: $|\mathcal{C}| \leq q^{m(n-d_{\mathrm{R}}+1)}$
  Optimal codes reach the Singleton bound

# Gabidulin codes

- $\mathbf{M} \in \mathrm{GF}(q)^{m \times n} \rightarrow \mathbf{m} \in \mathrm{GF}(q^m)^n$
- $(n, k, d_{\mathrm{R}} = n - k + 1)$ Gabidulin codes on $\mathrm{GF}(q^m)$: optimal codes similar to Reed-Solomon codes
- Generator matrix: $g_0, g_1, \ldots, g_{n-1} \in \mathrm{GF}(q^m)$ LI

$$\mathbf{G} = \begin{pmatrix} g_0^{q^0} & g_1^{q^0} & \cdots & g_{n-1}^{q^0} \\ g_0^{q^1} & g_1^{q^1} & \cdots & g_{n-1}^{q^1} \\ \vdots & \vdots & \vdots & \vdots \\ g_0^{q^{k-1}} & g_1^{q^{k-1}} & \cdots & g_{n-1}^{q^{k-1}} \end{pmatrix}$$

- Evaluation of linearized polynomials
  Decoding algorithms: EEA, BMA, PGZ, Welch-Berlekamp

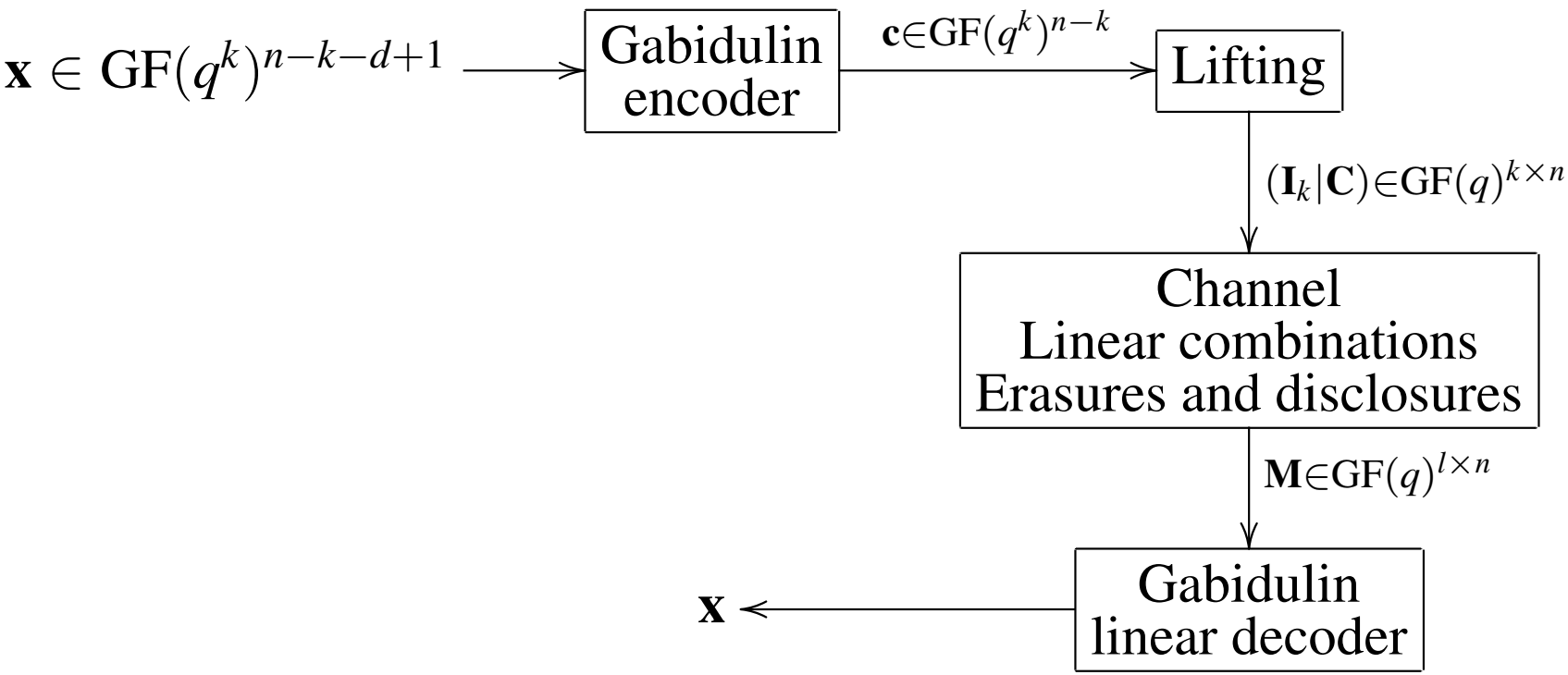
# Liftings of rank metric codes

- Lifting. $\mathbf{M} \in \mathrm{GF}(q)^{k\times(n-k)}$, $I(\mathbf{M})$: row space of $(\mathbf{I}_k|\mathbf{M})$
- Lifting preserves distance

$$d_{\mathrm{S}}(I(\mathbf{M}), I(\mathbf{N})) = 2d_{\mathrm{R}}(\mathbf{M}, \mathbf{N})$$

  $\Rightarrow I(\mathcal{C})$ CDC, $d_{\mathrm{S}}(I(\mathcal{C})) = 2d_{\mathrm{R}}(\mathcal{C})$
- Error control for RLNC with liftings is a rank metric problem
- Lifting of a Gabidulin code
  - Nearly optimal
  - Decoding algorithm: Sudan 1-list

# Implementation scheme for RLNC



# Outline

# Matroid (Whitney 35)

- $\mathcal{M} = (E, \mathcal{I})$, $\mathcal{I} \subseteq P(E)$: independent elements
  - $\mathcal{I} \neq \emptyset$
  - If $A \in \mathcal{I}$ and $B \subset A$, then $B \in \mathcal{I}$
  - If $A, B \in \mathcal{I}$ and $|B| > |A|$, then $\exists e \in B : |A| \cup \{e\} \in \mathcal{I}$
- $\mathcal{I}$: hereditary, basis of same cardinality, any independent family can be completed
- Examples
  - Vector. $E$: columns of a matrix, $\mathcal{I}$: LI
  - Graphic. $E$: edges of a graph, $\mathcal{I}$: no cycles
  - Free. $E$: a set, $\mathcal{I} = P(E)$: distinct

  Free $\subseteq$ Graphic $\subseteq$ Vector

# Flats and other concepts

| Linear algebra | Matroid |
|---|---|
| LI vectors | Independent elements |
| Basis | Basis |
| Span | Closure |
| Linear subspace | Flat |
| Dimension | Rank |
| All-zero vector | Loop |
| Collinear vectors | Parallel elements |

Flats ordered by inclusion form a lattice

# Error-free communication model

1. At the source
   - Encodes data into a flat of a matroid
   - Sends a basis of this flat
2. In the network
   - Intermediate nodes choose and transmit elements of the closure of the received elements
3. At the destination
   - Waits until a basis of the flat is received
   - Determines the flat and decodes the message

Comments

- Matroid = Protocol
- Error-free reconstruction always possible
- In practice, all transmitted flats have equal rank $k$

# Example: routing (G. and G. 10)

- Combinations are selections: flats are all the subsets
  $\Rightarrow$ Free matroid, all elements are independent
- Independent = Distinct
  $\Rightarrow$ Header to make packets distinct

$$a, b, c, d \rightarrow \left(\begin{array}{c|c} 0 & a \\ 1 & b \\ 2 & c \\ 3 & d \end{array}\right)$$

- Subsets in bijection with binary vectors: binary codes for error control in routing

# Rate

- Assumption: one element per packet in $\{0, 1, \ldots, q-1\}^n$
- Matroid rate: Efficiency of embedding a flat into a message

$$
\begin{aligned}
R(\mathcal{M}, k) &= \frac{\log_q \sharp\text{Flats with rank } k}{kn} \\
R(\text{routing}) &= \frac{\log_q \binom{q^n}{k}}{kn} \sim 1 - \frac{\log_q k}{n} \\
R(\text{linear}) &= \frac{\log_q \begin{bmatrix} n \\ k \end{bmatrix}}{kn} \sim 1 - \frac{k}{n}
\end{aligned}
$$

- $\begin{bmatrix} n \\ k \end{bmatrix}$: Gaussian binomial, $\begin{bmatrix} n \\ k \end{bmatrix} \sim q^{k(n-k)}$

# Delay

- Delay: average number of packets received to obtain $k$ independent
- Assumption: packets received i.i.d. and uniformly distributed
- $C_k$: cardinality of a flat with rank $k$ = number of combinations

$$
\begin{aligned}
D(\mathcal{M}, k) &= \sum_{i=0}^{k-1} \frac{C_k}{C_k - C_i} \\
D(\text{routing}) &= k \sum_{i=1}^{k} \frac{1}{i} \sim k \ln k \\
D(\text{linear}) &= \sum_{i=0}^{k-1} \frac{q^k - 1}{q^k - q^i} \sim k + \frac{1}{q-1}
\end{aligned}
$$

# Throughput

- Throughput: ratio of useful info received by the destination

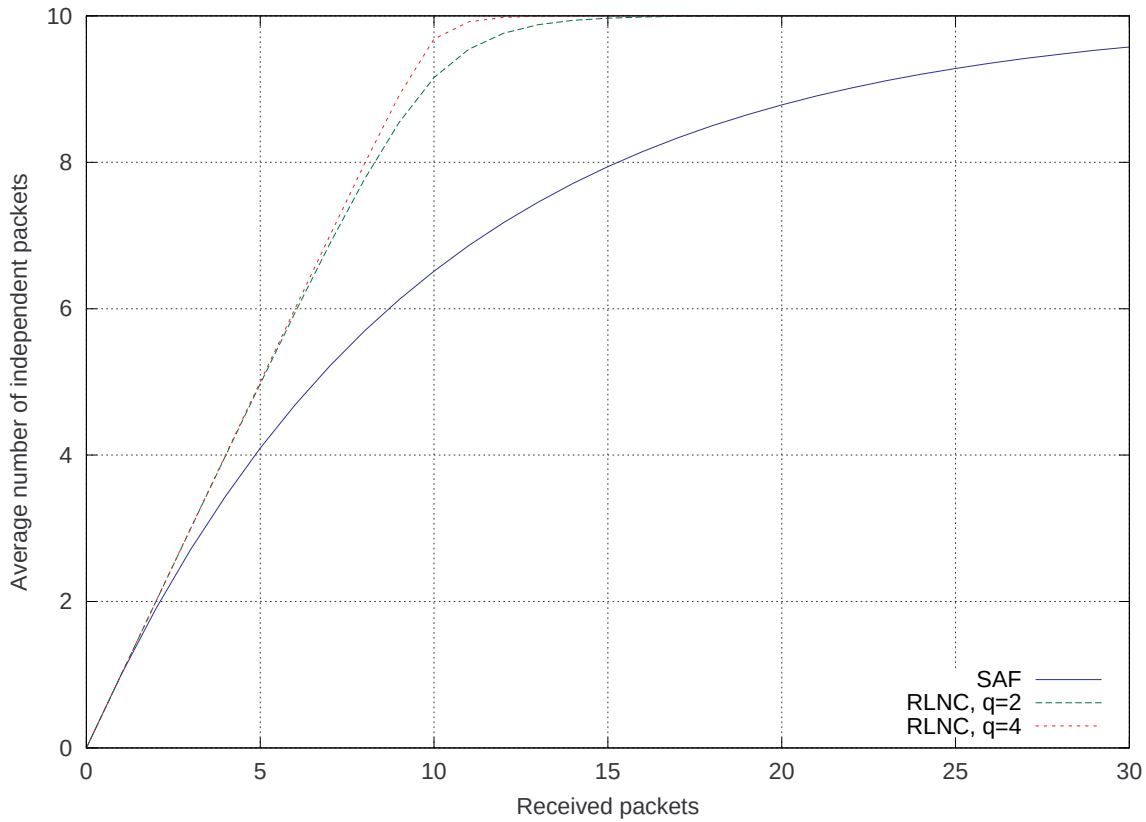
$$
\begin{aligned}
T(\mathcal{M},k) &= \frac{\log_q \sharp\text{Flats with rank } k}{nD(\mathcal{M},k)} = k\frac{R(\mathcal{M},k)}{D(\mathcal{M},k)} \\
T(\text{routing}) &\sim \frac{1}{\ln k} - \frac{1}{n\ln q} \\
T(\text{linear}) &\sim 1 - \frac{k}{n}
\end{aligned}
$$

- RLNC has a higher throughput, but is limited in the number of packets

# Advanced parameters: independence

$E_I(r)$: Average number of independent packets among $r$

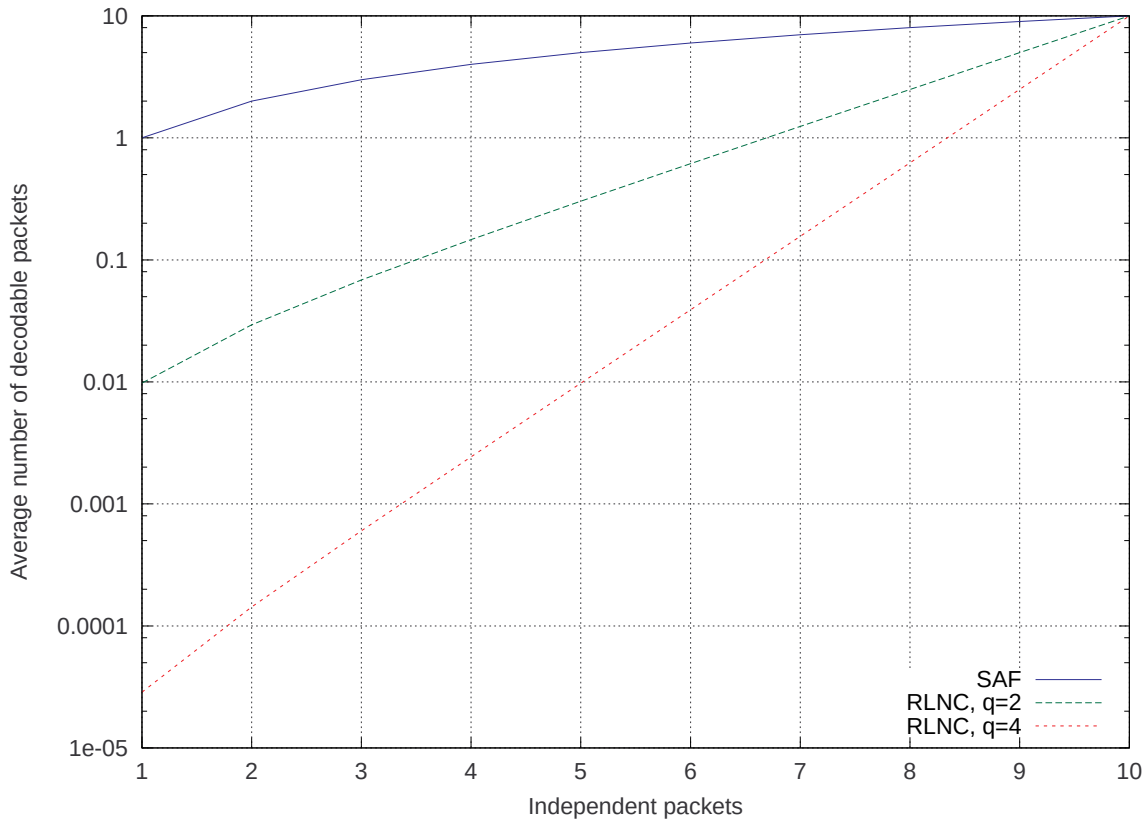
$$E_I(\text{routing}) \sim k(1 - e^{-\frac{r}{k}}), \qquad E_I(\text{linear}) \sim \min\{r, k\}$$



# Advanced parameters: partial decoding

$E_D(l)$: Average number of decodable packets among $l$ independent

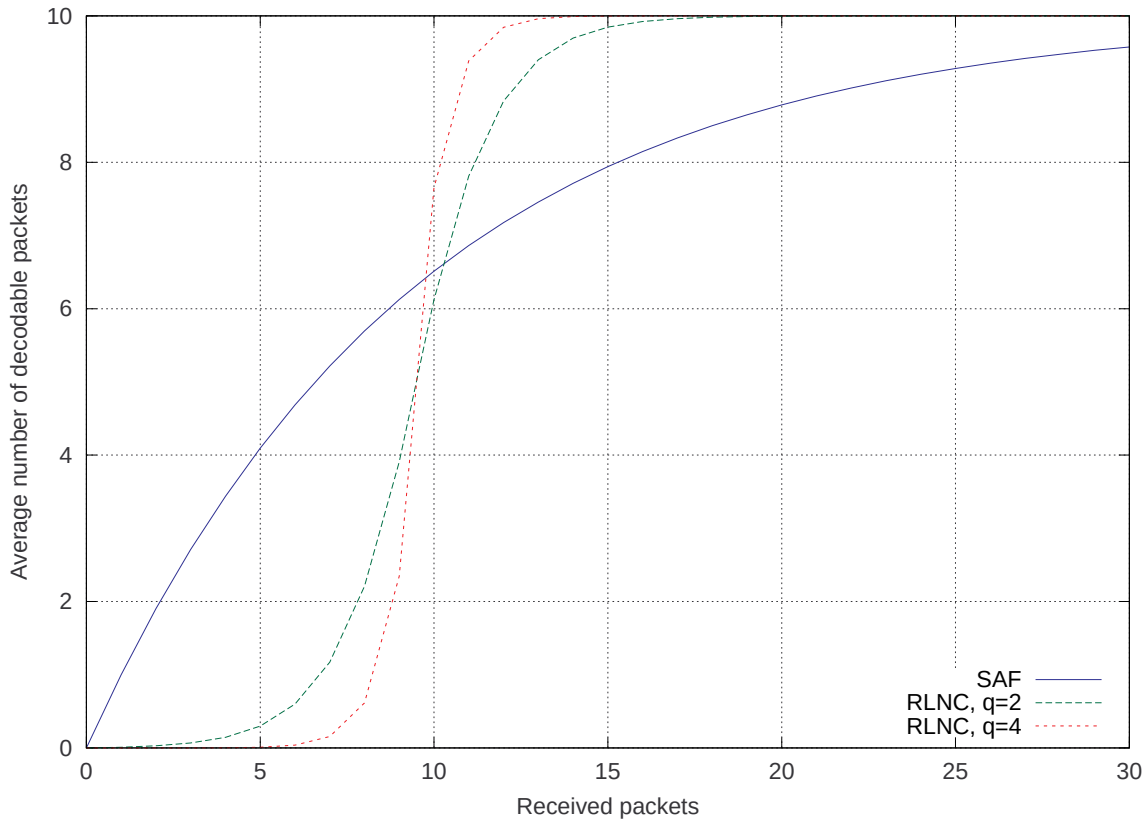
$$E_D(\text{routing}) = l, \qquad E_D(\text{linear}) \sim kq^{l-k}$$



# Advanced parameters: partial decoding

$E_T(r)$: Average number of decodable packets among $r$

$$E_T(\text{routing}) \sim k(1 - e^{-\frac{r}{k}}), \qquad E_T(\text{linear}) : \text{zero-one}$$



# Outline

# Simple matroid

- Loop: never independent (all-zero vector)
  $\Rightarrow$ Belongs to all flats: no information
- Parallel elements: elements pairwise dependent (collinear vectors)
  $\Rightarrow$ Belong to the same flats: same information
- Loop and parallel elements useless to the destination
  $\Rightarrow$ Simple matroid: neither loops nor parallel elements
- Any matroid can be simplified by removing loops and parallel elements
  $\Rightarrow$ Lattice of flats is preserved

# Simple matroid for RLNC

- 1 loop and $q-1$ parallel elements to remove

$$\frac{q^n-1}{q-1} \sim q^{n-1} \text{elements}$$

  $\Rightarrow$ Rate loss

- In a flat of rank $k$,

$$\frac{q^k-1}{q-1} \sim q^{k-1} \text{combinations}$$

  $\Rightarrow$ Combination power loss

- Simple matroid: projective geometry with rank $n$

# Affine network coding

- Affine combination of points

$$\sum_{i=1}^{k} \lambda_i a_i \qquad \text{where} \qquad \sum_{i=1}^{k} \lambda_i = 1$$

  $\Rightarrow q^{k-1}$ combinations

- Affinely independent points: $\sum_i \mu_i b_i \neq 0$ for $\sum_i \mu_i = 0$

$$b_1, b_2, \ldots, b_k \text{ AI} \Leftrightarrow (1, b_1), (1, b_2), \ldots, (1, b_k) \text{ LI}$$

- Matroid: affine geometry with rank $n+1$
  - Submatroid very close to the projective geometry with rank $n+1$
  - Already simple: no loss of rate or combination power

# Flats of affine network coding

- Flat: affine subspace with rank $k \leq n+1$
  - Linear subspace with dimension $k-1$
  - A point belonging to a complementary linear subspace
- Examples

$$\begin{aligned}
\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} &: \text{point} = (0,1), \text{direction} = (1,0) \\
&= \text{parallel to the x-axis} \\
\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} &: \text{point} = \text{origin}, \text{direction} = (0,1) \\
&= \text{y-axis}
\end{aligned}$$

- Cardinality: $q^{k-1}$, Number of flats: $q^{n-k+1}\begin{bmatrix} n \\ k-1 \end{bmatrix} \sim q^{k(n-k+1)}$

# Parameter comparison

| Network coding | Routing | Linear | Affine |
|---|---|---|---|
| Matroid | $U_{q^n,q^n}$ | $PG(n-1,q)$ | $AG(n,q)$ |
| Combination | Selection | Linear | Affine |
| Flat | Susbet | Subspace linear | Subpace affine |
| Max rank | $q^n$ | $n$ | $n+1$ |
| ♯Flats | $\binom{q^n}{k}$ | $\sim q^{k(n-k)}$ | $\sim q^{k(n-k+1)}$ |
| ♯Combinations | $k$ | $\sim q^{k-1}$ | $q^{k-1}$ |
| Rate | $\sim 1-\frac{\log_q k}{n}$ | $\sim 1-\frac{k}{n}$ | $\sim 1-\frac{k-1}{n}$ |
| Delay | $\sim k\log k$ | $\sim k$ | $\sim k$ |
| Throughput | $\sim \frac{1}{\log_q k}-\frac{1}{n\log q}$ | $\sim 1-\frac{k}{n}$ | $\sim 1-\frac{k-1}{n}$ |
| ♯Independent | $\sim k(1-e^{-\frac{r}{k}})$ | $\sim \min\{r,k\}$ | $\sim \min\{r,k\}$ |
| ♯Partial decoding | $l$ | $\sim kq^{l-k}$ | $kq^{l-k}$ |

# Error control

- Message alterations correspond to flat modifications
  $\Rightarrow$ Erasures and disclosures
- Codes on flats with the same rank
- Lattice distance

$$\begin{aligned} d_{\mathrm{L}}(f,g) &= \sharp\min(\text{erasures} + \text{disclosures}) \\ &= 2\mathrm{rk}(f \cup g) - \mathrm{rk}(f) - \mathrm{rk}(g) \end{aligned}$$

- $d_{\mathrm{L}}(f,g)$ =length of shortest path from $f$ to $g$
  - first $\mathrm{rk}(f \cup g) - \mathrm{rk}(f)$ disclosures
  - then $\mathrm{rk}(f \cup g) - \mathrm{rk}(g)$ erasures

  $\Rightarrow$ Illustrates NC's sensitivity to errors

# Lattice distance

- Inequality

$$d_{\mathrm{L}}(f,g) = 2\mathrm{rk}(f \cup g) - \mathrm{rk}(f) - \mathrm{rk}(g) \leq \mathrm{rk}(f) + \mathrm{rk}(g) - 2\mathrm{rk}(f \cap g)$$

- Always equality for routing and linear NC
- Inequality sometimes strict for affine NC: $f, g$ parallel hyperplanes $\mathrm{rk}(f) = \mathrm{rk}(g) = n$, $\mathrm{rk}(f \cup g) = n+1$, $\mathrm{rk}(f \cap g) = 0$

$$d_{\mathrm{L}}(f,g) = 2 \qquad \text{but} \qquad \mathrm{rk}(f \cup g) - \mathrm{rk}(f \cap g) = n+1$$

- However, not a problem for the design of good codes

# Liftings for affine NC

- Lifting: $A(\mathbf{M})$ is the affine subspace generated by

$$a,b,c,d \rightarrow \left(\begin{array}{ccc|c} 0 & 0 & 0 & a \\ 1 & 0 & 0 & b \\ 0 & 1 & 0 & c \\ 0 & 0 & 1 & d \end{array}\right) = (\mathbf{I}'_4|\mathbf{M})$$

- We can use rank metric codes

$$\begin{aligned} d_{\mathrm{L}}(A(\mathbf{M}),A(\mathbf{N})) &= 2\mathrm{rank}\left(\begin{array}{c|c|c} \mathbf{1} & \mathbf{I}'_k & \mathbf{M} \\ \hline \mathbf{1} & \mathbf{I}'_k & \mathbf{N} \end{array}\right) - 2k \\ &= 2\mathrm{rank}\left(\begin{array}{c|c|c} \mathbf{1} & \mathbf{I}'_k & \mathbf{M} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{M}-\mathbf{N} \end{array}\right) - 2k \\ &= 2d_{\mathrm{R}}(\mathbf{M},\mathbf{N}) \end{aligned}$$

# Liftings of Gabidulin codes

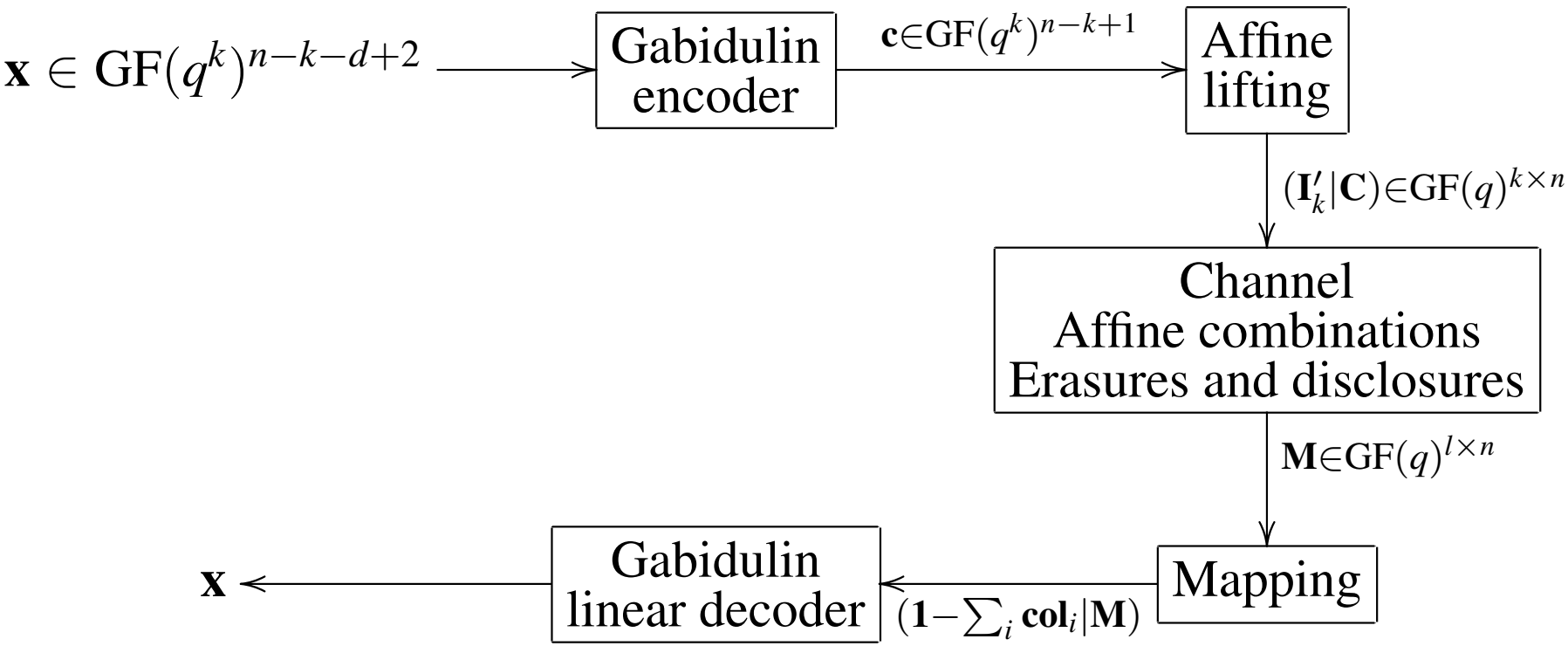
- Nearly optimal

$$|A(Gabidulin)| > K_q \max |\mathcal{C}| \qquad \text{where} \qquad K_q \sim 1 - q^{-1}$$

- Decoding in subspace metric

$$\begin{aligned}
&d_{\mathrm{L}}(M, A(\mathbf{C})) \\
&= 2\mathrm{rank}\left(\begin{array}{c|c|c} \mathbf{1} & \multicolumn{2}{c}{\mathbf{M}} \\ \hline \mathbf{1} & \mathbf{I}'_k & \mathbf{C} \end{array}\right) - \mathrm{rank}(\mathbf{1}|\mathbf{M}) - k \\
&= 2\mathrm{rank}\left(\frac{(\mathbf{1} - \sum_{i=1}^{k} \mathbf{col}_i|\mathbf{M})}{(\mathbf{I}_k|\mathbf{C})}\right) - \mathrm{rank}\left(\mathbf{1} - \sum_{i=1}^{k} \mathbf{col}_i \middle| \mathbf{M}\right) - k \\
&= d_{\mathrm{S}}(r(M), I(\mathbf{C}))
\end{aligned}$$

# Implementation scheme



# Outline

# Conclusion

- Contributions
  - Model based on matroids for the study of and noncoherent error control with NC
  - Affine NC outperforms linear NC, nearly optimal error correcting codes for affine NC
- Possibles extensions
  - Other matroids, other performance parameters
  - Incorporate characteristics of the network in the model
  - Fountain coding