Algebraic Network Coding Approach to Deterministic Wireless Relay Networks: Kim and Médard

**Open problem:** Capacity & code construction for wireless relay networks
- Channel noise
- Interference

[Avestimehr et al. ‘07] Deterministic model (ADT model)
- Interference
- Does not take into account channel noise
- In essence, high SNR regime

**ADT Network:**
- Min-cut Max-flow for unicast/multicast
- Matroidal

Describe ADT with Algebraic Network Coding [Koetter et al. ‘03]
- Use hyperedges: broadcast dependencies
- System matrix captures min-cut and other properties of the ADT
- Extend to non-multicast connections
- Network with erasures and cycles

**Statement:**
- System matrix $M = A(I - F)^{-1}B^T$
  - Encoding at source: $A$
  - Decoding at destination: $B$
  - Network code: $(I - F)^{-1}$

**Theorem:** Min-cut of ADT networks

\[
\text{mincut}(S,T) = \min_{\Omega} \text{rank}(G_{\Omega}) = \max_{\alpha(e,c),\beta(c'),c(e,c)} \text{rank}(M) \]

1. Show capacity characterization for unicast, multicast, multiple multicast, disjoint multicast, two-level multicast
2. Random linear network coding is capacity achieving
3. Extend to networks with random erasures/failures
4. Extend to networks with cycles

**ASSUMPTIONS AND LIMITATIONS:**
- Requires larger field size
- Does not prove/disprove ADT network’s ability to approximate the wireless networks

Algebraic network coding to understand capacity and code construction for deterministic wireless relay model
Wireless Network

- Open problem: capacity & code construction for wireless relay networks
  - Channel noise
  - Interference

- [Avestimehr et al. ‘07] “Deterministic model” (ADT model)
  - Interference
  - Does not take into account channel noise
  - In essence, high SNR regime

- High SNR
  - Noise → 0
  - Large gain
  - Large transmit power

\[
Y(e_3) = \beta_1 Y(e_1) + \beta_2 Y(e_2)
\]
• Min-cut: minimal rank of an incidence matrix of a certain cut between the source and destination [Avestimehr et al. ‘07]
  – Requires optimization over a large set of matrices
• Min-cut Max-flow Theorem holds for unicast/multicast sessions [Avestimehr et al. ‘07]
• Matroidal [Goemans et al. ‘09]
• Code construction algorithms [Amaudruz et al. ‘09][Erez et al. ‘10]
Our Contributions

• Connection to Algebraic Network Coding [Koetter et al. ‘03]:
  – Use of higher field size
  – Model broadcast constraint with hyper-edges
  – Capture ADT network problem with a single system matrix $M$
    • Prove that min-cut of ADT networks = rank($M$)
    • Prove Min-cut Max-flow for unicast/multicast holds
    • Extend optimality of linear operations to non-multicast sessions
    • Incorporate failures and erasures
    • Incorporate cycles
  – Show that random linear network coding achieves capacity
  – Do not prove/disprove ADT network model’s ability to approximate
    the wireless networks; but show that ADT network problems can be
    captured by the algebraic network coding framework
• [Jaggi et al. ‘06] “permute-and-add”: Show that network codes in higher field size $F_q$ can be converted to binary-vector code without loss in performance.

• Operations in $F_2$ is not sufficient to achieve capacity for multicast connections: need to operate in higher field size and/or binary-vector (See [Erez et al. ‘10] for an illustrative example).

• Therefore, we focus on higher field size $F_q$. 
ADT Network Model

- Original ADT model:
  - Broadcast: multiple edges (bit pipes) from the same node
  - Interference: additive MAC over binary field

- Algebraic model:

Possible "codes" at $e_{12}$, which represents the MAC constraint

<table>
<thead>
<tr>
<th>c</th>
<th>f</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Use hyperedges to model broadcast constraint

0/1 decision: To transmit or not
• Linear operations

  - Encoding at the source $S$: $\alpha(i, e_j)$

    $$A = \begin{pmatrix}
    \alpha_{1,e_1} & \alpha_{1,e_2} & 0 & \cdots & 0 \\
    \alpha_{2,e_1} & \alpha_{2,e_2} & 0 & \cdots & 0 
    \end{pmatrix}$$

  - Decoding at the destination $T$: $\varepsilon(e_j, (T, i))$

    $$B = \begin{pmatrix}
    0 & \cdots & 0 & \varepsilon(e_{11},(T,1)) & \varepsilon(e_{12},(T,1)) \\
    0 & \cdots & 0 & \varepsilon(e_{11},(T,2)) & \varepsilon(e_{12},(T,2)) 
    \end{pmatrix}$$
### Algebraic Framework

- **$X(S, i)$**: source process $i$
- **$Y(e)$**: process at port $e$
- **$Z(T, i)$**: destination process $i$

#### Linear operations
- at the source $S$: $\alpha(i, e_j)$
- at the nodes $V$: $\beta(e_j, e_{j'})$
- at the destination $T$: $\varepsilon(e_j, (T, i))$

#### Example Equations

- $Y(e_1) = \alpha_{(1,e_1)}X(S, 1) + \alpha_{(2,e_1)}X(S, 2)$
- $Y(e_2) = \alpha_{(1,e_2)}X(S, 1) + \alpha_{(2,e_2)}X(S, 2)$
- $Y(e_3) = Y(e_6) = Y(e_1)$
- $Y(e_4) = Y(e_2)$
- $Y(e_5) = Y(e_8) = 0$
- $Y(e_7) = \beta_{(e_3,e_7)}Y(e_3) + \beta_{(e_4,e_7)}Y(e_4)$
- $Y(e_9) = Y(e_{11}) = \beta_{(e_6,e_9)}Y(e_6)$
- $Y(e_{10}) = \beta_{(e_6,e_{10})}Y(e_6)$
- $Y(e_{12}) = Y(e_7) + Y(e_{10})$
- $Z(T, 1) = \varepsilon_{(e_{11},(T,1))}Y(e_{11}) + \varepsilon_{(e_{12},(T,1))}Y(e_{12})$
- $Z(T, 2) = \varepsilon_{(e_{11},(T,2))}Y(e_{11}) + \varepsilon_{(e_{12},(T,2))}Y(e_{12})$
- Broadcast constraint: Use hyperedges
- MAC constraint: possible “codes”
- Use higher field sizes: combine binary edges (binary-vector)
  - Interaction between β and α, γ:
  - (α+β) alone does not constrain the choice of α
  - (α+γ) alone is fine as well
  - Larger field to ensure both are satisfied

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>c</th>
<th>⊕</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>γ</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>α</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>α+γ</td>
<td></td>
</tr>
</tbody>
</table>
System Matrix $M = A(I - F)^{-1}B^T$

- Linear operations
  - Coding at the nodes $V$: $\beta(e_j, e_{j'})$
    - $F$ represents physical structure of the ADT network
    - $F^k$: non-zero entry = path of length $k$ between nodes exists
    - $(I-F)^{-1} = I + F + F^2 + F^3 + \ldots$ : connectivity of the network (impulse response of the network)

\[
F = \begin{bmatrix}
0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \beta(e_3, e_7) & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \beta(e_4, e_7) & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\]

Broadcast constraint (hyperedge)
MAC constraint (addition)
Internal operations (network code)
System Matrix \( M = A(I - F)^{-1}B^T \)

- Input-output relationship of the network

\[
Z = X(S) \cdot M
\]

Captures rate of the network

\[
= |\mathcal{X}(S)| \cdot A
\]

\[
\begin{bmatrix}
|O(S)| & |\mathcal{E}|
\end{bmatrix}
\]

\[
|\mathcal{E}|
\]

\[
(I - F)^{-1}
\]

Captures network code, topology (Field size as well)
Theorem: Min-cut of ADT Networks

\[ \mincut(S, T) = \min_{\Omega} \text{rank}(G_{\Omega}) \]
\[ = \max_{\alpha(i, e), \beta(e', e), \epsilon(e', i)} \text{rank}(M) \]

- From the original paper by Avestimehr et al.
  - Requires optimizing over ALL cuts between \( S \) and \( T \)
  - Not constructive: assumes infinite block length, internal node operations not considered

- Show that the rank of \( M \) is equivalent to optimizing over all cuts
  - System matrix captures the structure of the network
  - Constructive: the assignment of variables gives a network code
Min-cut Max-flow Theorem

- For a unicast/multicast connection from source $S$ to destination $T$, the following are equivalent:
  1. A unicast/multicast connection of rate $R$ is feasible.
  2. $\text{mincut}(S, T_i) \geq R$ for all destinations $T_i$.
  3. There exists an assignment of variables such that $M$ is invertible.

- Proof idea:
  1. & 2. equivalent by previous work.
  3.→1. If $M$ is invertible, then connection has been established.
  1.→3. If connection established, $M = I$. Therefore, $M$ is invertible.

- **Corollary:**
  Random linear network coding achieves capacity for a unicast/multicast connection.
Extensions to Non-multicast Connections

- [Multiple multicast] Multiple sources $S_1 S_2 \ldots S_k$ wants to transmit to all destinations $T_1 T_2 \ldots T_N$
  - Connection feasible if and only if $\text{mincut}\{S_1 S_2 \ldots S_k\}, T_j \geq \text{sum of rate from sources } S_1 S_2 \ldots S_k$

- Proof idea:
  Introduce a super-source $S$, and apply the multicast min-cut max-flow theorem.
• [Disjoint Multicast]

\[ \mathcal{X}(S, T_i) \cap \mathcal{X}(S, T_j) = \emptyset \text{ for all } i \neq j \]

The connection is feasible if and only if:

\[ \text{mincut}(S, \mathcal{T}') \geq \sum_{T_i \in \mathcal{T}'} |\mathcal{X}(S, T_i)| \text{ for any } \mathcal{T}' \subseteq \mathcal{T} \]

• Proof idea:
Introduce a super-destination T, and apply the multicast min-cut max-flow theorem.
• [Two-level Multicast] A set of destinations, $T_m$, participate in a multicast connection; rest of the destinations, $T_d$, in a disjoint multicast. The connection is feasible if and only if:
  - $T_m$: Satisfy single multicast connection requirement.
  - $T_d$: Satisfy disjoint multicast connection requirement.

![System matrix $M$](image)
• Multiple multicast:
  – Random linear network coding achieves capacity

• Disjoint/Two-level multicast:
  – Random linear network coding at *intermediate nodes*; but a carefully chosen *encoding matrix at source* achieves capacity
Generalized Min-cut Max-flow Theorem

- For any general connection set
- Sufficiency condition (not necessary)

Given an acyclic network $G$ with a connection set $\mathcal{C}$, let $M = \{M_{i,j}\}$ where $M_{i,j}$ is the system matrix for source processes $\mathcal{X}(S_i)$ to destination processes $\mathcal{Z}(T_j)$. Then, $(G, \mathcal{C})$ is solvable if there exists an assignment of $\alpha(i,e_j)$, $\epsilon(e_i,(T_j,k))$, and $\beta(e_i,e_j)$ such that

1) $M_{i,j} = 0$ for all $(S_i, T_j, \mathcal{X}(S_i, T_j)) \notin \mathcal{C}$,
2) Let $(S_{\sigma(i)}, T_j, \mathcal{X}(S_{\sigma(i)}, T_j)) \in \mathcal{C}$ for $i \in [1, K(j)]$.

Thus, this is the set of connections with $T_j$ as a receiver. Then, $[M^{T}_{\sigma(1),j}, M^{T}_{\sigma(2),j}, \ldots, M^{T}_{\sigma(K_j),j}]$ is a $|\mathcal{Z}(T_j)| \times |\mathcal{Z}(T_j)|$ is a nonsingular system matrix.
Incorporating erasures in ADT network

• Wireless networks: stochastic in nature
  – Random erasures occur

• ADT network
  – Models wireless deterministically with parallel bit-pipes
  – Min-cut as well as previous code construction algorithm needs to be recomputed every time the network changes

• Algebraic framework:
  – Robust against some set of link failures (network code will remain successful regardless of these failures)
  – The time average of $\text{rank}(M)$ gives the true min-cut of the network
  – Applies to the connections described previously
Incorporating cycles in ADT network

- Wireless networks intrinsically have bi-directional links; therefore, cycles exist.

- ADT network mode
  - A directed network without cycles: links from the source to the destinations

- Algebraic framework
  - To incorporate cycles, need a notion of time (causal); therefore, introduce delay on links $D$
  - Express network processes in power series in $D$
  - The same theorems as for delay-less network apply
Conclusions

- ADT network can be expressed with Algebraic Network Coding Formulation [Koetter et al. ‘03].
  - Use of higher field size
  - Model broadcast constraint with hyper-edge
  - Capture ADT network problem with a single system matrix $M$

- Prove an algebraic definition of \textbf{min-cut} = rank($M$)

- Prove Min-cut Max-flow for unicast/multicast holds

- Extend optimality of linear operations to \textbf{non-multicast} sessions
  - Disjoint multicast, Two-level multicast, multiple source multicast, generalized min-cut max-flow theorem

- Show that \textbf{random linear network coding} achieves capacity

- Incorporate \textbf{delay} and \textbf{failures} (allows cycles within the network)