Adversarial Network Coding

Alberto Ravagnani

University College Dublin

Paris 8, December 2018

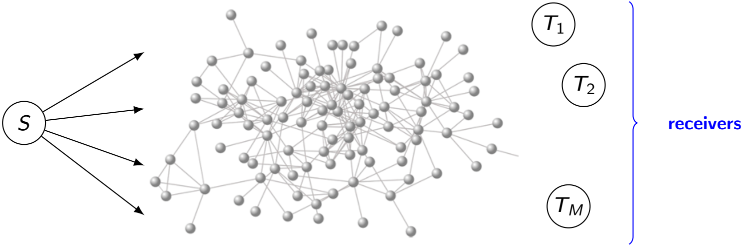
joint work with Frank R. Kschischang (UofT)

What is network coding about?

Network coding: data transmission over networks.

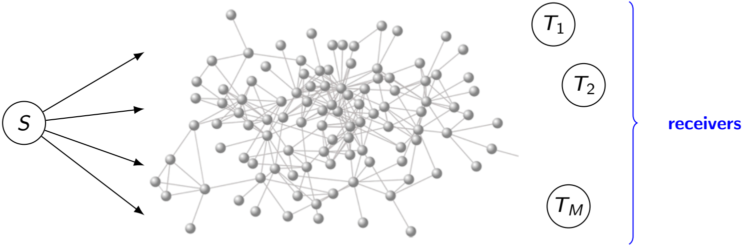
What is network coding about?

Network coding: data transmission over networks.



What is network coding about?

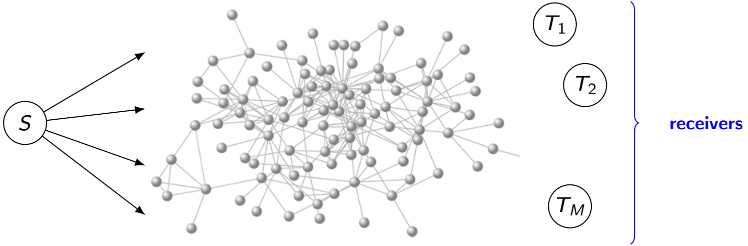
Network coding: data transmission over networks.



- One source S attempts to sends messages $m_1, \ldots, m_k \in \mathbb{F}_q^n$.
- The sinks demand **all** the messages (multicast).
- What about the intermediate nodes?

What is network coding about?

Network coding: data transmission over networks.



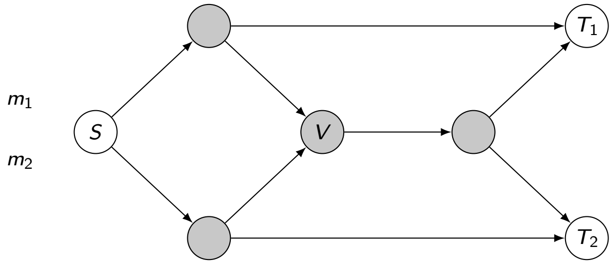
- One source S attempts to sends messages $m_1, \ldots, m_k \in \mathbb{F}_q^n$.
- The sinks demand **all** the messages (multicast).
- What about the intermediate nodes?

Goal

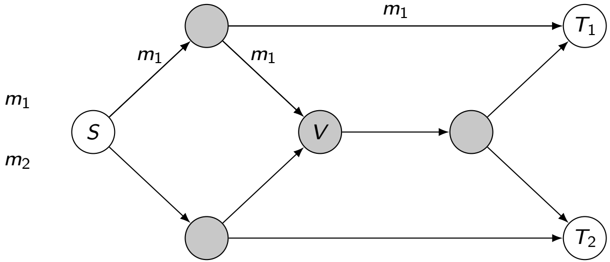
Maximize the number of messages that are transmitted to **all** sinks (**rate**).

Key idea: allow the nodes to perform operations on the received inputs.

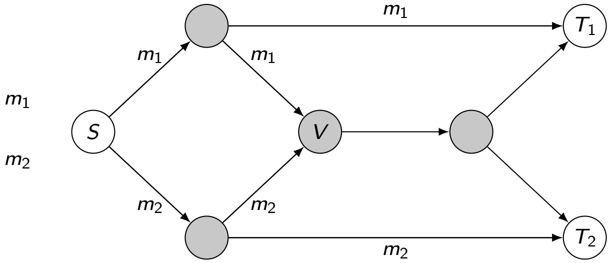
The "Butterfly" network



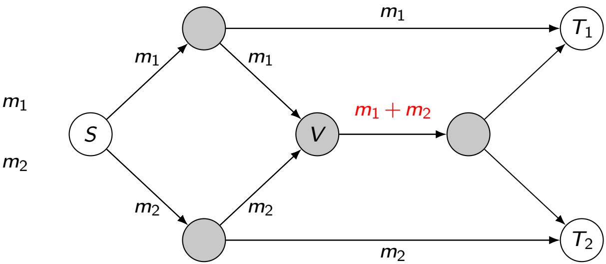
The "Butterfly" network



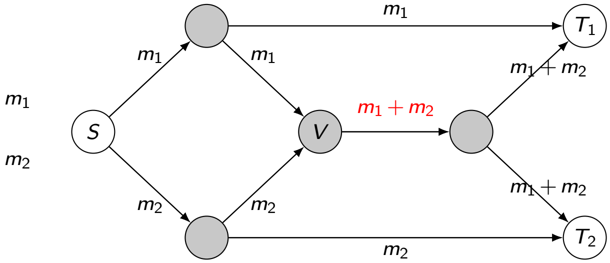
The "Butterfly" network



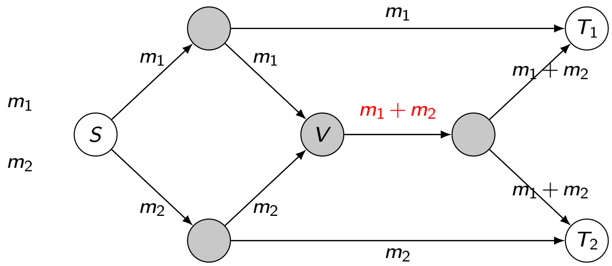
The "Butterfly" network



The “Butterfly” network



The "Butterfly" network



This strategy is optimal: there is no better strategy!

A mathematical framework for Adversarial Network Coding

Scenario

multiple sources (not just one) $+$ one or multiple **adversaries**.

A mathematical framework for Adversarial Network Coding

Scenario

multiple sources (not just one) $+$ one or multiple **adversaries**.

What we expect from the math model:

1. Give **mathematical** definitions for:
   - network capacity (maximum rate),
   - communication scheme,
   - network adversary,
   - ...

A mathematical framework for Adversarial Network Coding

Scenario

multiple sources (not just one) $+$ one or multiple **adversaries**.

What we expect from the math model:

1. Give **mathematical** definitions for:
   - network capacity (maximum rate),
   - communication scheme,
   - network adversary,
   - ...

   rate = the max. # of msg that can be transmitted to all sinks per channel use

Scenario

multiple sources (not just one) $+$ one or multiple **adversaries**.

What we expect from the math model:

1. Give **mathematical** definitions for:
   - network capacity (maximum rate),
   - communication scheme,
   - network adversary,
   - ...

✗ *rate = the max. # of msg that can be transmitted to all sinks per channel use* ✗

we are not happy with this

A mathematical framework for Adversarial Network Coding

Scenario

multiple sources (not just one) $+$ one or multiple **adversaries**.

What we expect from the math model:

1. Give **mathematical** definitions for:
   - network capacity (maximum rate),
   - communication scheme,
   - network adversary,
   - ...

   ✗ *rate = the max. # of msg that can be transmitted to all sinks per channel use* ✗

   we are not happy with this

2. Provide formal tools to derive new upper bounds for the capacity of a network.

3. Cover various communication scenarios.

A mathematical framework for Adversarial Network Coding

Scenario

multiple sources (not just one) $+$ one or multiple **adversaries**.

What we expect from the math model:

1. Give **mathematical** definitions for:
   - network capacity (maximum rate),
   - communication scheme,
   - network adversary,
   - ...

✗ *rate = the max. # of msg that can be transmitted to all sinks per channel use* ✗

we are not happy with this

2. Provide formal tools to derive new upper bounds for the capacity of a network.
3. Cover various communication scenarios.

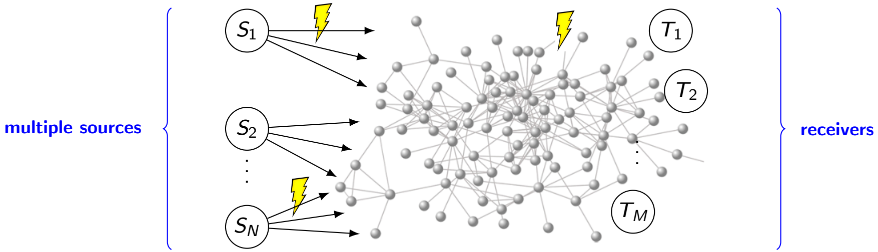
Remark

We do this in part by mathematizing and extending ideas of:

... Shannon, Cai, Li, Yeung, Yang, Zhang, Jaggi, Langberg, Katti, Ho, Katabi, Médard, Effros, Nutman, Wang, Silva, Kschischang, Kœtter, Siavoshani, Diggavi, Fragouli, Kœrner, Orlitsky, ...

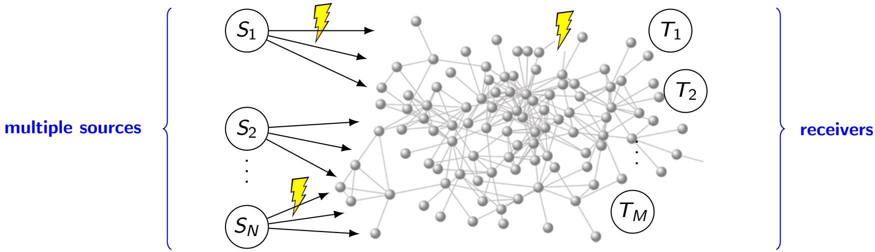
Mathematical model for Adversarial Network Coding

Edge-specific adversaries:



Mathematical model for Adversarial Network Coding

Edge-specific adversaries:



Our approach/program:

1. Adversarial point-to-point channels (no networks).
2. Operations with channels (product, concatenation, union).
3. Hamming-type adversarial channels over cartesian product alphabets.
4. Adversarial networks: network codes, error-correcting codes, capacity regions.
5. Porting bounds for Hamming-type channels to networks (general method).
6. Applications: new upper and lower bounds for some adversarial model.
7. New communication schemes for some scenarios.

Adversarial channels

Noisy channels: theory of "probability" **vs** Adversarial channels: theory of "possibility"

Definition

An (**adversarial**) **channel** is a map $\Omega : \mathscr{X} \to 2^{\mathscr{Y}} \setminus \{\emptyset\}$, where $\mathscr{X}$ and $\mathscr{Y}$ are finite non-empty sets called **input** and **output alphabet**, respectively.

Notation: $\Omega : \mathscr{X} \dashrightarrow \mathscr{Y}$.

Noisy channels: theory of "probability" **vs** Adversarial channels: theory of "possibility"

Definition

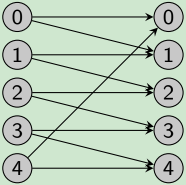
An (**adversarial**) **channel** is a map $\Omega : \mathscr{X} \to 2^{\mathscr{Y}} \setminus \{\emptyset\}$, where $\mathscr{X}$ and $\mathscr{Y}$ are finite non-empty sets called **input** and **output alphabet**, respectively.

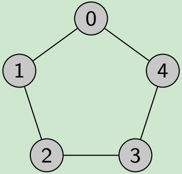
Notation: $\Omega : \mathscr{X} \dashrightarrow \mathscr{Y}$.

Example

Let $\mathscr{X} = \mathscr{Y} := \{0,1,2,3,4\}$, and let $\Omega : \mathscr{X} \dashrightarrow \mathscr{Y}$ be the channel defined by

$$\Omega(0) := \{0,1\}, \quad \Omega(1) := \{1,2\}, \quad \Omega(2) := \{2,3\}, \quad \Omega(3) := \{3,4\}, \quad \Omega(4) := \{4,0\}.$$





The graph on the right is called the *confusability graph*.

Adversarial channels

Definition

An **(adversarial) channel** is a map $\Omega : \mathscr{X} \to 2^{\mathscr{Y}} \setminus \{\emptyset\}$, where $\mathscr{X}$ and $\mathscr{Y}$ are finite non-empty sets called **input** and **output alphabet**, respectively.

Notation: $\Omega : \mathscr{X} \dashrightarrow \mathscr{Y}$.

Example

Let $\mathscr{X} = \mathscr{Y} = \mathscr{A}^4$, where $\mathscr{A}$ is a finite set.

Consider an adversary **A** able to corrupt at most one of the components indexed by $\{1,3,4\}$ of a 4-tuple

$$(x_1, x_2, x_3, x_4) \in \mathscr{A}^4.$$

Adversarial channels

Definition

An (**adversarial**) **channel** is a map $\Omega : \mathscr{X} \to 2^{\mathscr{Y}} \setminus \{\emptyset\}$, where $\mathscr{X}$ and $\mathscr{Y}$ are finite non-empty sets called **input** and **output alphabet**, respectively.

Notation: $\Omega : \mathscr{X} \dashrightarrow \mathscr{Y}$.

Example

Let $\mathscr{X} = \mathscr{Y} = \mathscr{A}^4$, where $\mathscr{A}$ is a finite set.

Consider an adversary **A** able to corrupt at most one of the components indexed by $\{1,3,4\}$ of a 4-tuple

$$(x_1, x_2, x_3, x_4) \in \mathscr{A}^4.$$

The corresponding channel $\Omega : \mathscr{A}^4 \dashrightarrow \mathscr{A}^4$ is given by

$$\Omega(x) = \{y \in \mathscr{A}^4 \mid y_2 = x_2 \text{ and } \mathrm{d_H}(x,y) \leq 1\} \qquad \text{for all } \; x \in \mathscr{A}^4,$$

where $\mathrm{d_H}$ is the Hamming distance.

(One-shot) capacity

Definition

Let $\Omega : \mathscr{X} \dashrightarrow \mathscr{Y}$ be a channel. A (**one-shot**) **code** for Ω is a non-empty subset $\mathscr{C} \subseteq \mathscr{X}$. We say that $\mathscr{C}$ is **good** for Ω when $\Omega(x) \cap \Omega(x') = \emptyset$ for all $x, x' \in \mathscr{C}$ with $x \neq x'$.

The (**one-shot**) **capacity** of $\Omega : \mathscr{X} \dashrightarrow \mathscr{Y}$ is

$$\mathrm{C}_1(\Omega) := \max\{\log_2 |\mathscr{C}| : \mathscr{C} \subseteq \mathscr{X} \text{ is good for } \Omega\}.$$

(One-shot) capacity

Definition

Let $\Omega : \mathscr{X} \dashrightarrow \mathscr{Y}$ be a channel. A **(one-shot) code** for Ω is a non-empty subset $\mathscr{C} \subseteq \mathscr{X}$. We say that $\mathscr{C}$ is **good** for Ω when $\Omega(x) \cap \Omega(x') = \emptyset$ for all $x, x' \in \mathscr{C}$ with $x \neq x'$.

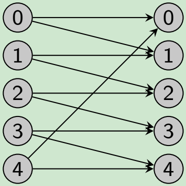
The **(one-shot) capacity** of $\Omega : \mathscr{X} \dashrightarrow \mathscr{Y}$ is

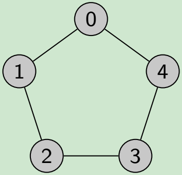
$$C_1(\Omega) := \max\{\log_2 |\mathscr{C}| : \mathscr{C} \subseteq \mathscr{X} \text{ is good for } \Omega\}.$$

Example

Let $\mathscr{X} = \mathscr{Y} := \{0,1,2,3,4\}$, and let $\Omega : \mathscr{X} \dashrightarrow \mathscr{Y}$ be the channel defined by

$$\Omega(0) := \{0,1\}, \quad \Omega(1) := \{1,2\}, \quad \Omega(2) := \{2,3\}, \quad \Omega(3) := \{3,4\}, \quad \Omega(4) := \{4,0\}.$$





(One-shot) capacity

Definition

Let $\Omega : \mathscr{X} \dashrightarrow \mathscr{Y}$ be a channel. A (**one-shot**) **code** for Ω is a non-empty subset $\mathscr{C} \subseteq \mathscr{X}$. We say that $\mathscr{C}$ is **good** for Ω when $\Omega(x) \cap \Omega(x') = \emptyset$ for all $x, x' \in \mathscr{C}$ with $x \neq x'$.

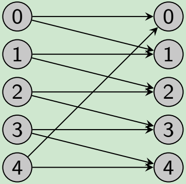
The (**one-shot**) **capacity** of $\Omega : \mathscr{X} \dashrightarrow \mathscr{Y}$ is

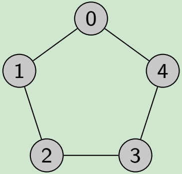
$$C_1(\Omega) := \max\{\log_2 |\mathscr{C}| : \mathscr{C} \subseteq \mathscr{X} \text{ is good for } \Omega\}.$$

Example

Let $\mathscr{X} = \mathscr{Y} := \{0,1,2,3,4\}$, and let $\Omega : \mathscr{X} \dashrightarrow \mathscr{Y}$ be the channel defined by

$$\Omega(0) := \{0,1\}, \quad \Omega(1) := \{1,2\}, \quad \Omega(2) := \{2,3\}, \quad \Omega(3) := \{3,4\}, \quad \Omega(4) := \{4,0\}.$$





We have $C_1(\Omega) = \log_2(2) = 1$.

Capacities

We study various notions of capacity of an adversarial channel:

- (**One-shot**) **capacity**, modeling one use of the channel;
- **Zero-error capacity**, modeling multiple uses of channels;
- **Compound zero-error capacity**, modeling adversaries with certain restrictions.

Concatenation of channels

Definition

Let $\Omega_1 : \mathscr{X}_1 \dashrightarrow \mathscr{Y}_1$ and $\Omega_2 : \mathscr{X}_2 \dashrightarrow \mathscr{Y}_2$ be channels, with $\mathscr{Y}_1 \subseteq \mathscr{X}_2$.

The **concatenation** of Ω_1 and Ω_2 is the channel $\Omega_1 \blacktriangleright \Omega_2 : \mathscr{X}_1 \dashrightarrow \mathscr{Y}_2$ defined by

$$(\Omega_1 \blacktriangleright \Omega_2)(x) := \bigcup_{y \in \Omega_1(x)} \Omega_2(y) \quad \text{for all } x \in \mathscr{X}_1.$$

Diagram: $\mathscr{X}_1 \overset{\Omega_1}{\dashrightarrow} \mathscr{Y}_1 \subseteq \mathscr{X}_2 \overset{\Omega_2}{\dashrightarrow} \mathscr{Y}_2.$

Concatenation of channels

Definition

Let $\Omega_1 : \mathscr{X}_1 \dashrightarrow \mathscr{Y}_1$ and $\Omega_2 : \mathscr{X}_2 \dashrightarrow \mathscr{Y}_2$ be channels, with $\mathscr{Y}_1 \subseteq \mathscr{X}_2$.

The **concatenation** of Ω_1 and Ω_2 is the channel $\Omega_1 \blacktriangleright \Omega_2 : \mathscr{X}_1 \dashrightarrow \mathscr{Y}_2$ defined by

$$(\Omega_1 \blacktriangleright \Omega_2)(x) := \bigcup_{y \in \Omega_1(x)} \Omega_2(y) \quad \text{for all } x \in \mathscr{X}_1.$$

Diagram: $\mathscr{X}_1 \overset{\Omega_1}{\dashrightarrow} \mathscr{Y}_1 \subseteq \mathscr{X}_2 \overset{\Omega_2}{\dashrightarrow} \mathscr{Y}_2.$

ACHTUNG! The confusability graph of $\Omega_1 \blacktriangleright \Omega_2$ is not determined by the confusability graphs of the two channels Ω_1 and Ω_2.

Operations

We study various channels operations:

- **product**, modeling combined channels uses;
- **power**, modeling multiple uses of a channel (zero-error capacity);
- **concatenation**, modeling channels used one after the other;
- **union**, modeling some restricted adversaries (compound zero-error capacity).

Channels can be combined with each other using these operations in an “algebraic fashion”.

What is a communication network?

What is a communication network?

Definition

A (**combinational**) **network** is a 4-tuple $\mathscr{N} = (\mathscr{V}, \mathscr{E}, \mathbf{S}, \mathbf{T})$ where:

1. $(\mathscr{V}, \mathscr{E})$ is a finite directed acyclic multigraph,
2. $\mathbf{S} \subseteq \mathscr{V}$ is the set of **sources**,
3. $\mathbf{T} \subseteq \mathscr{V}$ is the set of **terminals** or **sinks**.

What is a communication network?

Definition

A **(combinational) network** is a 4-tuple $\mathscr{N} = (\mathscr{V}, \mathscr{E}, \mathbf{S}, \mathbf{T})$ where:

1. $(\mathscr{V}, \mathscr{E})$ is a finite directed acyclic multigraph,
2. $\mathbf{S} \subseteq \mathscr{V}$ is the set of **sources**,
3. $\mathbf{T} \subseteq \mathscr{V}$ is the set of **terminals** or **sinks**.

(We allow multiple parallel directed edges). We also assume that the following hold.

4. $|\mathbf{S}| \geq 1$, $|\mathbf{T}| \geq 1$, $\mathbf{S} \cap \mathbf{T} = \emptyset$.
5. For any $S \in \mathbf{S}$ and $T \in \mathbf{T}$ there exists a directed path from S to T.
6. Sources do not have incoming edges, and terminals do not have outgoing edges.
7. For every vertex $V \in \mathscr{V} \setminus (\mathbf{S} \cup \mathbf{T})$ there exists a directed path from S to V for some $S \in \mathbf{S}$, and a directed path from V to T for some $T \in \mathbf{T}$.

What is a communication network?

Definition

A **(combinational) network** is a 4-tuple $\mathscr{N} = (\mathscr{V}, \mathscr{E}, \mathbf{S}, \mathbf{T})$ where:

1. $(\mathscr{V}, \mathscr{E})$ is a finite directed acyclic multigraph,
2. $\mathbf{S} \subseteq \mathscr{V}$ is the set of **sources**,
3. $\mathbf{T} \subseteq \mathscr{V}$ is the set of **terminals** or **sinks**.

(We allow multiple parallel directed edges). We also assume that the following hold.

4. $|\mathbf{S}| \geq 1$, $|\mathbf{T}| \geq 1$, $\mathbf{S} \cap \mathbf{T} = \emptyset$.
5. For any $S \in \mathbf{S}$ and $T \in \mathbf{T}$ there exists a directed path from S to T.
6. Sources do not have incoming edges, and terminals do not have outgoing edges.
7. For every vertex $V \in \mathscr{V} \setminus (\mathbf{S} \cup \mathbf{T})$ there exists a directed path from S to V for some $S \in \mathbf{S}$, and a directed path from V to T for some $T \in \mathbf{T}$.

The elements of $\mathscr{V}$ are called **vertices**. The elements of $\mathscr{V} \setminus (\mathbf{S} \cup \mathbf{T})$ are the **intermediate** vertices. We denote the set of incoming and outgoing edges of a $V \in \mathscr{V}$ by $\mathrm{in}(V)$ and $\mathrm{out}(V)$, respectively.

Nodes operations and network codes

The edges of a network $\mathscr{N}$ can carry precisely one symbol from a finite set $\mathscr{A}$, the **alphabet**.

Definition

A **network code** $\mathscr{F}$ for $\mathscr{N}$ is a family of functions $\{\mathscr{F}_V : V \in \mathscr{V} \setminus (\mathbf{S} \cup \mathbf{T})\}$, where

$$\mathscr{F}_V : \mathscr{A}^{|\mathrm{in}(V)|} \to \mathscr{A}^{|\mathrm{out}(V)|} \quad \text{for all } V \in \mathscr{V} \setminus (\mathbf{S} \cup \mathbf{T}).$$

Nodes operations and network codes

The edges of a network $\mathscr{N}$ can carry precisely one symbol from a finite set $\mathscr{A}$, the **alphabet**.

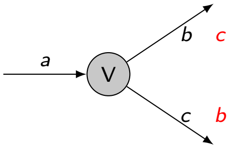
Definition

A **network code** $\mathscr{F}$ for $\mathscr{N}$ is a family of functions $\{\mathscr{F}_V : V \in \mathscr{V} \setminus (\mathbf{S} \cup \mathbf{T})\}$, where

$$\mathscr{F}_V : \mathscr{A}^{|\mathrm{in}(V)|} \to \mathscr{A}^{|\mathrm{out}(V)|} \qquad \text{for all } V \in \mathscr{V} \setminus (\mathbf{S} \cup \mathbf{T}).$$

ACHTUNG! This definition is not good (yet).

Let $a \in \mathscr{A}$ and $\mathscr{F}_V(a) = (b, c) \in \mathscr{A}^2$



Nodes operations and network codes

The edges of a network $\mathscr{N}$ can carry precisely one symbol from a finite set $\mathscr{A}$, the **alphabet**.

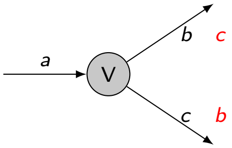
Definition

A **network code** $\mathscr{F}$ for $\mathscr{N}$ is a family of functions $\{\mathscr{F}_V : V \in \mathscr{V} \setminus (\mathbf{S} \cup \mathbf{T})\}$, where

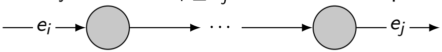
$$\mathscr{F}_V : \mathscr{A}^{|\mathrm{in}(V)|} \to \mathscr{A}^{|\mathrm{out}(V)|} \quad \text{for all } V \in \mathscr{V} \setminus (\mathbf{S} \cup \mathbf{T}).$$

ACHTUNG! This definition is not good (yet).

Let $a \in \mathscr{A}$ and $\mathscr{F}_V(a) = (b, c) \in \mathscr{A}^2$



The edges of $\mathscr{N}$ can be partially ordered: $e_i \preceq e_j$ is there exists a path in $\mathscr{N}$ of the form



Nodes operations and network codes

The edges of a network $\mathscr{N}$ can carry precisely one symbol from a finite set $\mathscr{A}$, the **alphabet**.

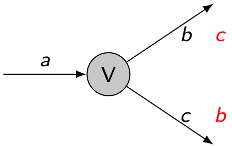
Definition

A **network code** $\mathscr{F}$ for $\mathscr{N}$ is a family of functions $\{\mathscr{F}_V : V \in \mathscr{V} \setminus (\mathbf{S} \cup \mathbf{T})\}$, where

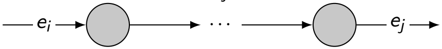
$$\mathscr{F}_V : \mathscr{A}^{|\mathrm{in}(V)|} \to \mathscr{A}^{|\mathrm{out}(V)|} \qquad \text{for all } V \in \mathscr{V} \setminus (\mathbf{S} \cup \mathbf{T}).$$

ACHTUNG! This definition is not good (yet).

Let $a \in \mathscr{A}$ and $\mathscr{F}_V(a) = (b, c) \in \mathscr{A}^2$



The edges of $\mathscr{N}$ can be partially ordered: $e_i \preceq e_j$ is there exists a path in $\mathscr{N}$ of the form



Theorem

The order $\preceq$ can be extended to a total order.

We fix such a total oder and denote it by $\leq$. This resolves the ambiguity.

Network channels

Network channels

Let $(\mathscr{N}, \mathbf{A})$ be a network with an adversary. Let $\mathscr{A}$ be the network alphabet.

- $\mathbf{S} = \{S_1, ..., S_N\}$ is the set of network sources.
- $J \subseteq \{1, ..., N\}$ is a set of source indices, $\mathbf{S}_J = \{S_i \mid i \in J\}$.
- $\mathscr{F}$ is a network code.
- The sources $\{S_i \mid i \notin J\}$ transmit fixed messages $\overline{x} \in \prod_{i \notin J} \mathscr{A}^{|\mathrm{out}(S_i)|}$.
- $\mathscr{E}' \subseteq \mathscr{E}$ is a non-empty set of edges.

Network channels

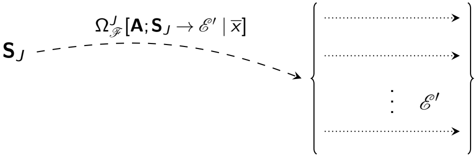
Let $(\mathscr{N}, \mathbf{A})$ be a network with an adversary. Let $\mathscr{A}$ be the network alphabet.

- $\mathbf{S} = \{S_1, ..., S_N\}$ is the set of network sources.
- $J \subseteq \{1, ..., N\}$ is a set of source indices, $\mathbf{S}_J = \{S_i \mid i \in J\}$.
- $\mathscr{F}$ is a network code.
- The sources $\{S_i \mid i \notin J\}$ transmit fixed messages $\overline{x} \in \prod_{i \notin J} \mathscr{A}^{|\mathrm{out}(S_i)|}$.
- $\mathscr{E}' \subseteq \mathscr{E}$ is a non-empty set of edges.

The channel

$$\Omega^J_{\mathscr{F}}[\mathbf{A}; \mathbf{S}_J \to \mathscr{E}' \mid \overline{x}] \;:\; \prod_{i \in J} \mathscr{A}^{|\mathrm{out}(S_i)|} \dashrightarrow \mathscr{A}^{|\mathscr{E}'|}$$

describes the transfer



Network channels

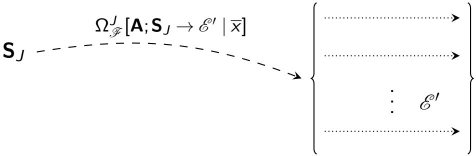
Let $(\mathscr{N}, \mathbf{A})$ be a network with an adversary. Let $\mathscr{A}$ be the network alphabet.

- $\mathbf{S} = \{S_1, ..., S_N\}$ is the set of network sources.
- $J \subseteq \{1, ..., N\}$ is a set of source indices, $\mathbf{S}_J = \{S_i \mid i \in J\}$.
- $\mathscr{F}$ is a network code.
- The sources $\{S_i \mid i \notin J\}$ transmit fixed messages $\overline{x} \in \prod_{i \notin J} \mathscr{A}^{|\mathrm{out}(S_i)|}$.
- $\mathscr{E}' \subseteq \mathscr{E}$ is a non-empty set of edges.

The channel

$$\Omega^J_{\mathscr{F}}[\mathbf{A}; \mathbf{S}_J \to \mathscr{E}' \mid \overline{x}] \ : \ \prod_{i \in J} \mathscr{A}^{|\mathrm{out}(S_i)|} \dashrightarrow \mathscr{A}^{|\mathscr{E}'|}$$

describes the transfer

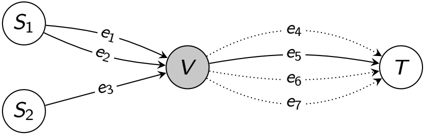


Special case: $\mathscr{E}' = \mathrm{in}(T)$, where $T \in \mathbf{T}$ is a terminal.

Example

Consider the following network $\mathscr{N}$ with alphabet $\mathscr{A}$.

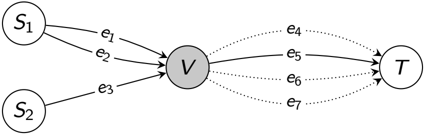
An adversary **A** is able to corrupt at most one of the values of the dotted edges of $\mathscr{N}$.



Example

Consider the following network $\mathscr{N}$ with alphabet $\mathscr{A}$.

An adversary **A** is able to corrupt at most one of the values of the dotted edges of $\mathscr{N}$.

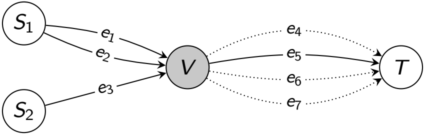


A network code $\mathscr{F}$ for $\mathscr{N}$ is the assignment of a function $\mathscr{F}_V : \mathscr{A}^3 \to \mathscr{A}^4$.

Example

Consider the following network $\mathscr{N}$ with alphabet $\mathscr{A}$.

An adversary **A** is able to corrupt at most one of the values of the dotted edges of $\mathscr{N}$.



A network code $\mathscr{F}$ for $\mathscr{N}$ is the assignment of a function $\mathscr{F}_V : \mathscr{A}^3 \to \mathscr{A}^4$.

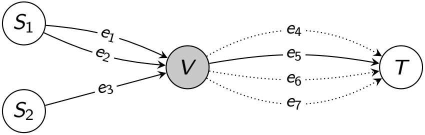
Let $J := \{1\}$, and assume that S_2 emits a fixed element $\overline{x} \in \mathscr{A}$. Let us describe

$$\Omega^J_{\mathscr{F}}[\mathbf{A}; \mathbf{S}_J \to \mathrm{in}(T) \mid \overline{x}].$$

Example

Consider the following network $\mathscr{N}$ with alphabet $\mathscr{A}$.

An adversary **A** is able to corrupt at most one of the values of the dotted edges of $\mathscr{N}$.



A network code $\mathscr{F}$ for $\mathscr{N}$ is the assignment of a function $\mathscr{F}_V : \mathscr{A}^3 \to \mathscr{A}^4$.

Let $J := \{1\}$, and assume that S_2 emits a fixed element $\overline{x} \in \mathscr{A}$. Let us describe

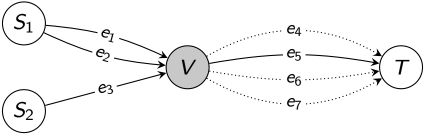
$$\Omega^J_{\mathscr{F}}[\mathbf{A}; \mathbf{S}_J \to \mathrm{in}(T) \mid \overline{x}].$$

Remark: we have to say what $\Omega^J_{\mathscr{F}}[\mathbf{A}; \mathbf{S} \to \mathrm{in}(T)](x_1, x_2) \subseteq \mathscr{A}^4$ is for $(x_1, x_2) \in \mathscr{A}^2$.

Example

Consider the following network $\mathscr{N}$ with alphabet $\mathscr{A}$.

An adversary **A** is able to corrupt at most one of the values of the dotted edges of $\mathscr{N}$.



A network code $\mathscr{F}$ for $\mathscr{N}$ is the assignment of a function $\mathscr{F}_V : \mathscr{A}^3 \to \mathscr{A}^4$.

Let $J := \{1\}$, and assume that S_2 emits a fixed element $\overline{x} \in \mathscr{A}$. Let us describe

$$\Omega^J_{\mathscr{F}}[\mathbf{A}; \mathbf{S}_J \to \mathrm{in}(T) \mid \overline{x}].$$

Remark: we have to say what $\Omega^J_{\mathscr{F}}[\mathbf{A}; \mathbf{S} \to \mathrm{in}(T)](x_1, x_2) \subseteq \mathscr{A}^4$ is for $(x_1, x_2) \in \mathscr{A}^2$.

If $(x_1, x_2) \in \mathscr{A}^2$, and $\overline{z} := \mathscr{F}_V(x_1, x_2, \overline{x}) \in \mathscr{A}^4$, then

$$\Omega^J_{\mathscr{F}}[\mathbf{A}; \mathbf{S} \to \mathrm{in}(T)](x_1, x_2) = \{y \in \mathscr{A}^4 \mid y_2 = \overline{z}_2 \ \text{ and } \ \mathrm{d_H}(y, \overline{z}) \leq 1\}.$$

Capacity region

Definition

- $\mathscr{N}$ a network with N sources $\mathbf{S} = \{S_1, \ldots, S_N\}$.
- $\mathbf{T}$ is the set of terminals.
- $\mathbf{A}$ is an adversary.

Definition

- $\mathscr{N}$ a network with N sources $\mathbf{S} = \{S_1, \dots, S_N\}$.
- $\mathbf{T}$ is the set of terminals.
- $\mathbf{A}$ is an adversary.

The (**one shot**) **capacity region** of $(\mathscr{N}, \mathbf{A})$ is the set

$$\mathscr{R}(\mathscr{N}, \mathbf{A}) \subseteq \mathbb{R}^N_{\geq 0}$$

of all the N-tuples $(\alpha_1, \dots, \alpha_N)$ for which there exist:

- a network code $\mathscr{F}$ for $\mathscr{N}$
- non-empty sets $\mathscr{C}_i \subseteq \mathscr{A}^{|\mathrm{out}(S_i)|}$, for $1 \leq i \leq N$

with the following properties:

1. $\log_{|\mathscr{A}|} |\mathscr{C}_i| = \alpha_i$,
2. $\mathscr{C} = \mathscr{C}_1 \times \cdots \times \mathscr{C}_N$ is a good code for each channel $\Omega_{\mathscr{F}}[\mathbf{A}; \mathbf{S} \to \mathrm{in}(T)]$, $T \in \mathbf{T}$.

We say that such a pair $(\mathscr{F}, \mathscr{C})$ **achieves** the **rate** $(\alpha_1, \dots, \alpha_N)$ in one shot.

Definition

- $\mathscr{N}$ a network with N sources $\mathbf{S} = \{S_1, \dots, S_N\}$.
- $\mathbf{T}$ is the set of terminals.
- $\mathbf{A}$ is an adversary.

The (**one shot**) **capacity region** of $(\mathscr{N}, \mathbf{A})$ is the set

$$\mathscr{R}(\mathscr{N}, \mathbf{A}) \subseteq \mathbb{R}^N_{\geq 0}$$

of all the N-tuples $(\alpha_1, \dots, \alpha_N)$ for which there exist:

- a network code $\mathscr{F}$ for $\mathscr{N}$
- non-empty sets $\mathscr{C}_i \subseteq \mathscr{A}^{|\mathrm{out}(S_i)|}$, for $1 \leq i \leq N$

with the following properties:

1. $\log_{|\mathscr{A}|} |\mathscr{C}_i| = \alpha_i$,
2. $\mathscr{C} = \mathscr{C}_1 \times \cdots \times \mathscr{C}_N$ is a good code for each channel $\Omega_{\mathscr{F}}[\mathbf{A}; \mathbf{S} \to \mathrm{in}(T)]$, $T \in \mathbf{T}$.

We say that such a pair $(\mathscr{F}, \mathscr{C})$ **achieves** the **rate** $(\alpha_1, \dots, \alpha_N)$ in one shot.

These conditions guarantee that the sources can transmit in one shot to each of the sinks $\alpha_1 + \cdots + \alpha_N$ alphabet symbols, α_i of which are emitted by S_i, for $1 \leq i \leq N$.

Other capacities

We study various notions of capacity region:

- (**one shot**) **capacity region**, modeling one network use;
- **zero-error capacity region**, modeling multiple uses of the network;
- **compound zero-error capacity region**, modeling certain restrictions on the adversaries.

Decomposition idea

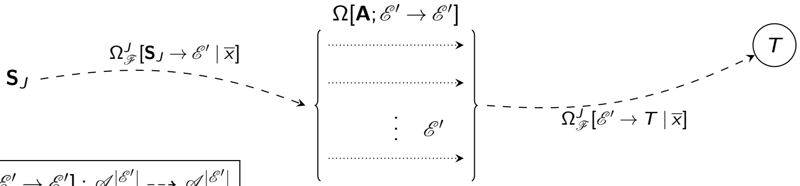
Let $(\mathscr{N}, \mathbf{A})$ be a network with an adversary. Let $\mathscr{A}$ be the network alphabet.

- $\mathbf{S} = \{S_1, ..., S_N\}$ the sources, $J \subseteq \{1, ..., N\}$ and $\mathbf{S}_J = \{S_i \mid i \in J\}$.
- $\mathscr{F}$ is a network code.
- The sources $\{S_i \mid i \notin J\}$ transmit fixed messages $\overline{x} \in \prod_{i \notin J} \mathscr{A}^{|\mathrm{out}(S_i)|}$.
- $\mathscr{E}' \subseteq \mathscr{E}$ is an edge-cut that separates $\mathbf{S}_J$ from $T \in \mathbf{T}$.

Decomposition idea

Let $(\mathscr{N}, \mathbf{A})$ be a network with an adversary. Let $\mathscr{A}$ be the network alphabet.

- $\mathbf{S} = \{S_1, ..., S_N\}$ the sources, $J \subseteq \{1, ..., N\}$ and $\mathbf{S}_J = \{S_i \mid i \in J\}$.
- $\mathscr{F}$ is a network code.
- The sources $\{S_i \mid i \notin J\}$ transmit fixed messages $\overline{x} \in \prod_{i \notin J} \mathscr{A}^{|\mathrm{out}(S_i)|}$.
- $\mathscr{E}' \subseteq \mathscr{E}$ is an edge-cut that separates $\mathbf{S}_J$ from $T \in \mathbf{T}$.

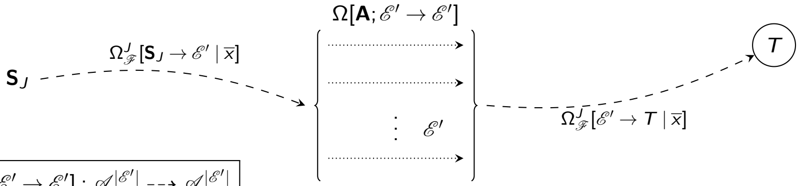


$\boxed{\Omega[\mathbf{A}; \mathscr{E}' \to \mathscr{E}'] : \mathscr{A}^{|\mathscr{E}'|} \dashrightarrow \mathscr{A}^{|\mathscr{E}'|}}$

Decomposition idea

Let $(\mathscr{N}, \mathbf{A})$ be a network with an adversary. Let $\mathscr{A}$ be the network alphabet.

- $\mathbf{S} = \{S_1, ..., S_N\}$ the sources, $J \subseteq \{1, ..., N\}$ and $\mathbf{S}_J = \{S_i \mid i \in J\}$.
- $\mathscr{F}$ is a network code.
- The sources $\{S_i \mid i \notin J\}$ transmit fixed messages $\overline{x} \in \prod_{i \notin J} \mathscr{A}^{|\mathrm{out}(S_i)|}$.
- $\mathscr{E}' \subseteq \mathscr{E}$ is an edge-cut that separates $\mathbf{S}_J$ from $T \in \mathbf{T}$.



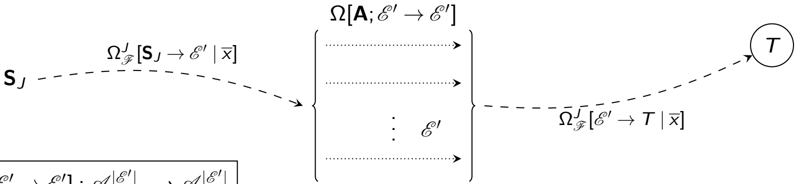
$$\boxed{\Omega[\mathbf{A}; \mathscr{E}' \to \mathscr{E}'] : \mathscr{A}^{|\mathscr{E}'|} \dashrightarrow \mathscr{A}^{|\mathscr{E}'|}}$$

$$\mathrm{C}_1 \left(\Omega^J_{\mathscr{F}}[\mathbf{A}; \mathbf{S}_J \to T \mid \overline{x}] \right)$$

Decomposition idea

Let $(\mathscr{N}, \mathbf{A})$ be a network with an adversary. Let $\mathscr{A}$ be the network alphabet.

- $\mathbf{S} = \{S_1, ..., S_N\}$ the sources, $J \subseteq \{1, ..., N\}$ and $\mathbf{S}_J = \{S_i \mid i \in J\}$.
- $\mathscr{F}$ is a network code.
- The sources $\{S_i \mid i \notin J\}$ transmit fixed messages $\overline{x} \in \prod_{i \notin J} \mathscr{A}^{|\mathrm{out}(S_i)|}$.
- $\mathscr{E}' \subseteq \mathscr{E}$ is an edge-cut that separates $\mathbf{S}_J$ from $T \in \mathbf{T}$.



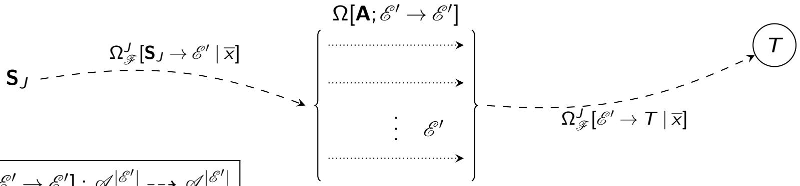
$$\boxed{\Omega[\mathbf{A}; \mathscr{E}' \to \mathscr{E}'] : \mathscr{A}^{|\mathscr{E}'|} \dashrightarrow \mathscr{A}^{|\mathscr{E}'|}}$$

$$\mathrm{C}_1\left(\Omega^J_{\mathscr{F}}[\mathbf{A}; \mathbf{S}_J \to T \mid \overline{x}]\right) \leq \mathrm{C}_1\left(\Omega^J_{\mathscr{F}}[\mathbf{S}_J \to \mathscr{E}' \mid \overline{x}] \blacktriangleright \Omega[\mathbf{A}; \mathscr{E}' \to \mathscr{E}'] \blacktriangleright \Omega^J_{\mathscr{F}}[\mathscr{E}' \to T \mid \overline{x}]\right)$$

Decomposition idea

Let $(\mathscr{N},\mathbf{A})$ be a network with an adversary. Let $\mathscr{A}$ be the network alphabet.

- $\mathbf{S}=\{S_1,...,S_N\}$ the sources, $J\subseteq\{1,...,N\}$ and $\mathbf{S}_J=\{S_i \mid i\in J\}$.
- $\mathscr{F}$ is a network code.
- The sources $\{S_i \mid i\notin J\}$ transmit fixed messages $\overline{x}\in\prod_{i\notin J}\mathscr{A}^{|\mathrm{out}(S_i)|}$.
- $\mathscr{E}'\subseteq\mathscr{E}$ is an edge-cut that separates $\mathbf{S}_J$ from $T\in\mathbf{T}$.



$$\boxed{\Omega[\mathbf{A};\mathscr{E}'\to\mathscr{E}'] : \mathscr{A}^{|\mathscr{E}'|} \dashrightarrow \mathscr{A}^{|\mathscr{E}'|}}$$

$$\mathrm{C}_1\left(\Omega^J_{\mathscr{F}}[\mathbf{A};\mathbf{S}_J\to T\mid\overline{x}]\right) \;\le\; \mathrm{C}_1\left(\Omega^J_{\mathscr{F}}[\mathbf{S}_J\to\mathscr{E}'\mid\overline{x}] \blacktriangleright \Omega[\mathbf{A};\mathscr{E}'\to\mathscr{E}'] \blacktriangleright \Omega^J_{\mathscr{F}}[\mathscr{E}'\to T\mid\overline{x}]\right)$$

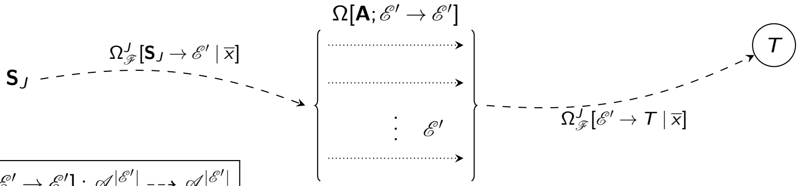
Proposition (R., Kschischang)

$\mathrm{C}_1(\Omega_1\blacktriangleright\Omega_2\blacktriangleright\Omega_3)\le\min_{i=1}^3\mathrm{C}_1(\Omega_i)$.

Decomposition idea

Let $(\mathscr{N},\mathbf{A})$ be a network with an adversary. Let $\mathscr{A}$ be the network alphabet.

- $\mathbf{S}=\{S_1,\dots,S_N\}$ the sources, $J\subseteq\{1,\dots,N\}$ and $\mathbf{S}_J=\{S_i \mid i\in J\}$.
- $\mathscr{F}$ is a network code.
- The sources $\{S_i \mid i\notin J\}$ transmit fixed messages $\overline{x}\in\prod_{i\notin J}\mathscr{A}^{|\mathrm{out}(S_i)|}$.
- $\mathscr{E}'\subseteq\mathscr{E}$ is an edge-cut that separates $\mathbf{S}_J$ from $T\in\mathbf{T}$.



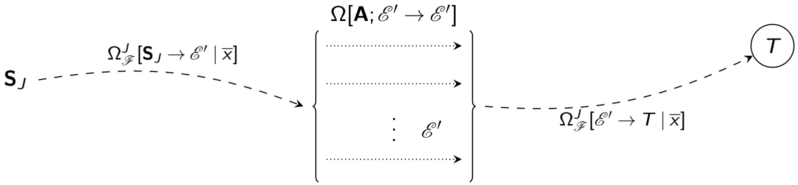
$$\boxed{\Omega[\mathbf{A};\mathscr{E}'\to\mathscr{E}'] : \mathscr{A}^{|\mathscr{E}'|} \dashrightarrow \mathscr{A}^{|\mathscr{E}'|}}$$

$$\mathrm{C}_1\left(\Omega_{\mathscr{F}}^J[\mathbf{A};\mathbf{S}_J\to T \mid \overline{x}]\right) \;\le\; \mathrm{C}_1\left(\Omega_{\mathscr{F}}^J[\mathbf{S}_J\to\mathscr{E}' \mid \overline{x}] \blacktriangleright \Omega[\mathbf{A};\mathscr{E}'\to\mathscr{E}'] \blacktriangleright \Omega_{\mathscr{F}}^J[\mathscr{E}'\to T \mid \overline{x}]\right)$$

Proposition (R., Kschischang)

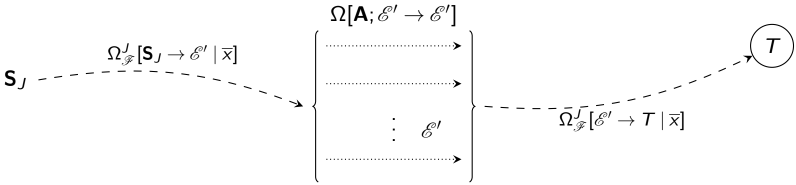
$\mathrm{C}_1(\Omega_1\blacktriangleright\Omega_2\blacktriangleright\Omega_3)\le\min_{i=1}^3\mathrm{C}_1(\Omega_i)$. Therefore $\mathrm{C}_1\left(\Omega_{\mathscr{F}}^J[\mathbf{A};\mathbf{S}_J\to T \mid \overline{x}]\right) \le \mathrm{C}_1\left(\Omega[\mathbf{A};\mathscr{E}'\to\mathscr{E}']\right)$.

Remarks



$\Omega[\mathbf{A}; \mathscr{E}' \to \mathscr{E}']$
$\mathbf{S}_J$
$\Omega^J_{\mathscr{F}}[\mathbf{S}_J \to \mathscr{E}' \mid \overline{x}]$
$\mathscr{E}'$
T
$\Omega^J_{\mathscr{F}}[\mathscr{E}' \to T \mid \overline{x}]$

Remarks



- This can be made rigorous.

- Using channel operations, this decomposition idea can be extended to:
  - zero-error capacity,
  - compound zero-error capacity.

- This allows to port bounds for channels $\Omega : \mathscr{A}^n \dashrightarrow \mathscr{A}^n$ to networks in a systematic way.

- This applies to single source and multiple sources networks.

- We study also erasure adversaries (alphabet extensions).

Theorem (R., Kschischang)

Let $\mathscr{N}$ be a network with N sources $\mathbf{S} = \{S_1, ..., S_N\}$ and set of terminals $\mathbf{T}$. Set $I := \{1, ..., N\}$.

Denote by $\mathbf{A}$ an aversary:

- having access to all the network edges $\mathscr{E}$,
- able to corrupt at most t of them, and erase up to e of them.

Bounds

Theorem (R., Kschischang)

Let $\mathscr{N}$ be a network with N sources $\mathbf{S} = \{S_1, \dots, S_N\}$ and set of terminals $\mathbf{T}$. Set $I := \{1, \dots, N\}$.

Denote by $\mathbf{A}$ an aversary:

- having access to all the network edges $\mathscr{E}$,
- able to corrupt at most t of them, and erase up to e of them.

For all $(\alpha_1, \dots, \alpha_N) \in \mathscr{R}(\mathscr{N}, \mathbf{A})$ and for all non-empty $J \subseteq I$ we have

$$\sum_{i \in J} \alpha_i \leq \min_{T \in \mathbf{T}} \max\{0, \text{min-cut}(\mathbf{S}_J, T) - 2t - e\}$$

Bounds

Theorem (R., Kschischang)

Let $\mathscr{N}$ be a network with N sources $\mathbf{S} = \{S_1, ..., S_N\}$ and set of terminals $\mathbf{T}$. Set $I := \{1, ..., N\}$.

Denote by $\mathbf{A}$ an aversary:

- having access to all the network edges $\mathscr{E}$,
- able to corrupt at most t of them, and erase up to e of them.

For all $(\alpha_1, ..., \alpha_N) \in \mathscr{R}(\mathscr{N}, \mathbf{A})$ and for all non-empty $J \subseteq I$ we have

$$\sum_{i \in J} \alpha_i \leq \min_{T \in \mathbf{T}} \max\{0, \text{min-cut}(\mathbf{S}_J, T) - 2t - e\}$$

and

$$\sum_{i \in J} \alpha_i \leq \min_{T \in \mathbf{T}} \max\left\{0, \text{min-cut}(\mathbf{S}_J, T) - \log_{|\mathscr{A}|}\left(\sum_{h=0}^{t'} \binom{\text{min-cut}(\mathbf{S}_J, T)}{h} (|\mathscr{A}| - 1)^h\right)\right\}, \quad t' := \lfloor t + e/2 \rfloor.$$

Bounds

Theorem (R., Kschischang)

Let $\mathscr{N}$ be a network with N sources $\mathbf{S} = \{S_1, ..., S_N\}$ and set of terminals $\mathbf{T}$. Set $I := \{1, ..., N\}$.

Denote by $\mathbf{A}$ an aversary:

- having access to all the network edges $\mathscr{E}$,
- able to corrupt at most t of them, and erase up to e of them.

For all $(\alpha_1, ..., \alpha_N) \in \mathscr{R}(\mathscr{N}, \mathbf{A})$ and for all non-empty $J \subseteq I$ we have

$$\sum_{i \in J} \alpha_i \leq \min_{T \in \mathbf{T}} \max\{0, \text{min-cut}(\mathbf{S}_J, T) - 2t - e\}$$

and

$$\sum_{i \in J} \alpha_i \leq \min_{T \in \mathbf{T}} \max \left\{ 0, \text{min-cut}(\mathbf{S}_J, T) - \log_{|\mathscr{A}|} \left(\sum_{h=0}^{t'} \binom{\text{min-cut}(\mathbf{S}_J, T)}{h} (|\mathscr{A}| - 1)^h \right) \right\}, \quad t' := \lfloor t + e/2 \rfloor.$$

These are obtained by "porting" the Singleton and the Hamming bounds, respectively.

Remark: any other bound from classical Coding Theory can be ported.

Theorem (R., Kschischang)

Let $\mathscr{N}$ be a network with N sources $\mathbf{S} = \{S_1, \ldots, S_N\}$ and set of terminals $\mathbf{T}$. Set $I := \{1, \ldots, N\}$.

Denote by $\mathbf{A}$ a set of L aversaries $\mathbf{A}_1, \ldots, \mathbf{A}_L$ such that:

- adversary ℓ has access to $\mathscr{E}_\ell \subseteq \mathscr{E}$ for all $1 \leq \ell \leq L$,
- the $\mathscr{E}_\ell$'s are pairwise disjoint,
- adversary ℓ is able to corrupt at most t_ℓ edges, and erase at most e_ℓ edges.

Theorem (R., Kschischang)

Let $\mathscr{N}$ be a network with N sources $\mathbf{S}=\{S_1,...,S_N\}$ and set of terminals $\mathbf{T}$. Set $I:=\{1,...,N\}$.

Denote by $\mathbf{A}$ a set of L aversaries $\mathbf{A}_1,...,\mathbf{A}_L$ such that:

- adversary ℓ has access to $\mathscr{E}_\ell \subseteq \mathscr{E}$ for all $1 \le \ell \le L$,
- the $\mathscr{E}_\ell$'s are pairwise disjoint,
- adversary ℓ is able to corrupt at most t_ℓ edges, and erase at most e_ℓ edges.

For all $(\alpha_1,...,\alpha_N) \in \mathscr{R}(\mathscr{N},\mathbf{A})$ and for all non-empty $J \subseteq I$ we have

$$\sum_{i \in J} \alpha_i \le \min_{T \in \mathbf{T}} \min \left\{ |\mathscr{E}'| - \sum_{\ell=1}^{L} \min\left\{2t_\ell + e_\ell, |\mathscr{E}' \cap \mathscr{E}_\ell|\right\} : \mathscr{E}' \subseteq \mathscr{E} \text{ is a cut between } \mathbf{S}_J \text{ and } T \right\}.$$

Other results

- Similar bounds can be proved for:
  - zero-error capacity region,
  - compound zero-error capacity region.
- These bounds apply to single source and multiple sources networks.
- These bounds show that when the adversary is restricted, capacity cannot be achieved in general with linear network coding.
- We give capacity-achieving schemes for some adversarial scenarios.

Recall:

Theorem (R., Kschischang)

Let $\mathscr{N}$ be a network with N sources $\mathbf{S} = \{S_1, ..., S_N\}$ and set of terminals $\mathbf{T}$. Set $I := \{1, ..., N\}$.
Denote by $\mathbf{A}$ an aversary:

- having access to all the network edges $\mathscr{E}$,
- able to corrupt at most t of them.

For all $(\alpha_1, ..., \alpha_N) \in \mathscr{R}(\mathscr{N}, \mathbf{A})$ and all $\emptyset \neq J \subseteq I$ we have $\sum_{i \in J} \alpha_i \leq \min_{T \in \mathbf{T}} \max\{0, \text{min-cut}(\mathbf{S}_J, T) - 2t\}$.

Lower bounds

Recall:

Theorem (R., Kschischang)

Let $\mathscr{N}$ be a network with N sources $\mathbf{S} = \{S_1, ..., S_N\}$ and set of terminals $\mathbf{T}$. Set $I := \{1, ..., N\}$. Denote by $\mathbf{A}$ an aversary:

- having access to all the network edges $\mathscr{E}$,
- able to corrupt at most t of them.

For all $(\alpha_1, ..., \alpha_N) \in \mathscr{R}(\mathscr{N}, \mathbf{A})$ and all $\emptyset \neq J \subseteq I$ we have $\sum_{i \in J} \alpha_i \le \min_{T \in \mathbf{T}} \max\{0, \text{min-cut}(\mathbf{S}_J, T) - 2t\}$.

Theorem (R., Kschischang)

Under the same hypotheses, we have

$$\mathscr{R}(\mathscr{N}, \mathbf{A}) \supseteq \left\{ (a_1, ..., a_N) \in \mathbb{N}^N : \sum_{i \in J} a_i \le \min_{T \in \mathbf{T}} \max\{0, \text{min-cut}(\mathbf{S}_J, T) - 2t\} \text{ for all } \emptyset \neq J \subseteq I \right\},$$

provided that $\mathscr{A} = \mathbb{F}_q^m$, and q and m are sufficiently large.

A different scheme

For $N = 2$ sources and 1 terminal, to achieve a rate (a_1, a_2) the previous scheme requires as network alphabet

$$\mathbb{F}_q^m \qquad \text{where} \qquad m = (a_1 - 2t) \cdot (a_2 - 2t).$$

A different scheme

For $N = 2$ sources and 1 terminal, to achieve a rate (a_1, a_2) the previous scheme requires as network alphabet

$$\mathbb{F}_q^m \qquad \text{where} \qquad m = (a_1 - 2t) \cdot (a_2 - 2t).$$

Theorem (R., Kschischang)

There exists a scheme (with efficient coding and decoding) for the same problem parameters that requires as network alphabet

$$\mathbb{F}_q^m \qquad \text{where} \qquad m = a_1 + a_2 - 2t.$$

A different scheme

For $N = 2$ sources and 1 terminal, to achieve a rate (a_1, a_2) the previous scheme requires as network alphabet

$$\mathbb{F}_q^m \qquad \text{where} \qquad m = (a_1 - 2t) \cdot (a_2 - 2t).$$

Theorem (R., Kschischang)

There exists a scheme (with efficient coding and decoding) for the same problem parameters that requires as network alphabet

$$\mathbb{F}_q^m \qquad \text{where} \qquad m = a_1 + a_2 - 2t.$$

Thank you very much!