Network Coding (NC)

CITHN2002 – Summer 2024

Prof. Dr.-Ing. Stephan Günther

Chair of Distributed Systems and Security
School of Computation, Information and Technology
Technical University of Munich

Chapter 5: Models

Networks as graphs

Flow problems

Minimum cost flow problem

Maximum s-t flow problem

Min-cut and its capacity

Minimum cost maximum s–t flow problem

Multicommodity flow problems

Multicast in networks

Store-forward multicast

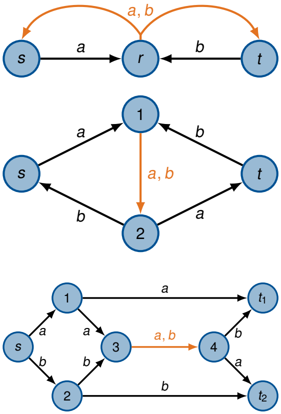
Multicast tree-based forwarding

Multicast with network coding

Wireless Packet Networks

Networks as graphs

Networks as graphs



- (Wired) Networks can be modeled as abstract graphs.
- Information flow in networks with routing and forwarding can be modeled as (multi-)commodity flow problem.
- Gives nice problems (flow optimization problems) and algorithms (Dijkstra, Bellman-Ford, etc.).
- Special properties of “Information” (arbitrarily reproducible, coded representation, etc.) are not taken into account in the standard commodity model.

Networks as graphs

- The set of nodes is given by $N = \{1, \dots, n\}$
- The set of arc indices is given by $A = \{1, 2, \dots, m\}$
  - Each arc index $j \in A$ represents an ordered pair of nodes
  - We therefore write $j \equiv (a, b)$
- The set of arcs is given by $\mathcal{A} = \{(a, b) \mid \exists$ link from $a \in N$ to $b \in N\}$

Important structures

- path (directed, undirected)
- tree (directed, undirected)
- cycle (directed, undirected)

Note: We assume $\mathcal{G}$ is connected, i. e., there exists an undirected path between any pair of nodes.

Networks as graphs

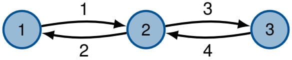
Examples

- Enumeration of arcs is arbitrary but must be fixed for a given network.
- Convention: use lexicographic order, i. e., $(2, 1) \prec (2, 3) \prec (3, 2)$.

Networks as graphs

Examples

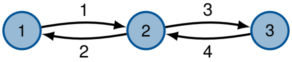
- Enumeration of arcs is arbitrary but must be fixed for a given network.
- Convention: use lexicographic order, i. e., $(2, 1) \prec (2, 3) \prec (3, 2)$.



Networks as graphs

Examples

- Enumeration of arcs is arbitrary but must be fixed for a given network.
- Convention: use lexicographic order, i. e., $(2, 1) \prec (2, 3) \prec (3, 2)$.

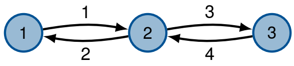


$$N = \{1, 2, 3\}$$
$$\mathcal{A} = \{(1, 2), (2, 1), (2, 3), (3, 2)\}$$

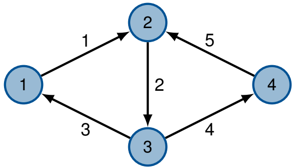
Networks as graphs

Examples

- Enumeration of arcs is arbitrary but must be fixed for a given network.
- Convention: use lexicographic order, i. e., $(2,1) \prec (2,3) \prec (3,2)$.



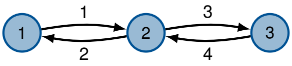
$$N = \{1,2,3\}$$
$$\mathcal{A} = \{(1,2),(2,1),(2,3),(3,2)\}$$



Networks as graphs

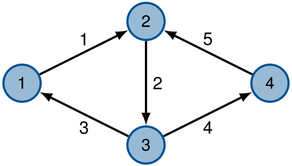
Examples

- Enumeration of arcs is arbitrary but must be fixed for a given network.
- Convention: use lexicographic order, i. e., $(2,1) \prec (2,3) \prec (3,2)$.



$N = \{1, 2, 3\}$

$\mathcal{A} = \{(1,2), (2,1), (2,3), (3,2)\}$



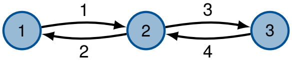
$N = \{1, 2, 3, 4\}$

$\mathcal{A} = \{(1,2), (2,3), (3,1), (3,4), (4,2)\}$

Networks as graphs

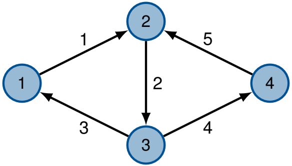
Examples

- Enumeration of arcs is arbitrary but must be fixed for a given network.
- Convention: use lexicographic order, i. e., $(2,1) \prec (2,3) \prec (3,2)$.



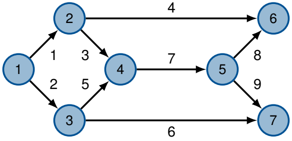
$N = \{1, 2, 3\}$

$\mathcal{A} = \{(1,2), (2,1), (2,3), (3,2)\}$



$N = \{1, 2, 3, 4\}$

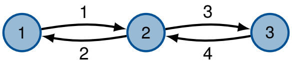
$\mathcal{A} = \{(1,2), (2,3), (3,1), (3,4), (4,2)\}$



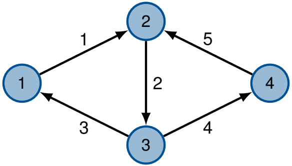
Networks as graphs

Examples

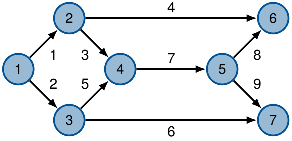
- Enumeration of arcs is arbitrary but must be fixed for a given network.
- Convention: use lexicographic order, i. e., $(2,1) \prec (2,3) \prec (3,2)$.



$$N = \{1, 2, 3\}$$
$$\mathcal{A} = \{(1,2), (2,1), (2,3), (3,2)\}$$



$$N = \{1, 2, 3, 4\}$$
$$\mathcal{A} = \{(1,2), (2,3), (3,1), (3,4), (4,2)\}$$



$$N = \{1, 2, 3, 4, 5, 6, 7\}$$
$$\mathcal{A} = \{(1,2), (1,3), (2,4), (2,6), (3,4), (3,7), (4,5), (5,6), (5,7)\}$$

Networks as graphs

Definition: incidence matrix $\boldsymbol{M}$

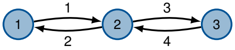
Given $\mathcal{G} = (N, A)$, we define the incidence matrix $\boldsymbol{M} = (m_{ij}) \in \{-1, 0, 1\}^{|N| \times |A|}$ where $\forall i \in N$ and $j \in A$

$$m_{ij} = \begin{cases} 1 & \text{arc } j \text{ leaves node } i, \\ -1 & \text{arc } j \text{ enters node } i, \\ 0 & \text{otherwise.} \end{cases}$$

Definition: incidence matrix $\boldsymbol{M}$

Given $\mathcal{G} = (N, A)$, we define the incidence matrix $\boldsymbol{M} = (m_{ij}) \in \{-1, 0, 1\}^{|N| \times |A|}$ where $\forall i \in N$ and $j \in A$

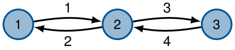
$$m_{ij} = \begin{cases} 1 & \text{arc } j \text{ leaves node } i, \\ -1 & \text{arc } j \text{ enters node } i, \\ 0 & \text{otherwise.} \end{cases}$$



Definition: incidence matrix $\boldsymbol{M}$

Given $\mathcal{G} = (N, A)$, we define the incidence matrix $\boldsymbol{M} = (m_{ij}) \in \{-1, 0, 1\}^{|N| \times |A|}$ where $\forall i \in N$ and $j \in A$

$$m_{ij} = \begin{cases} 1 & \text{arc } j \text{ leaves node } i, \\ -1 & \text{arc } j \text{ enters node } i, \\ 0 & \text{otherwise.} \end{cases}$$



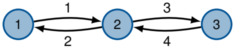
$$\boldsymbol{M} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

Networks as graphs

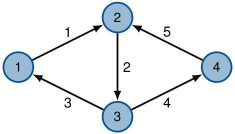
Definition: incidence matrix $\boldsymbol{M}$

Given $\mathcal{G} = (N, A)$, we define the incidence matrix $\boldsymbol{M} = (m_{ij}) \in \{-1, 0, 1\}^{|N| \times |A|}$ where $\forall i \in N$ and $j \in A$

$$m_{ij} = \begin{cases} 1 & \text{arc } j \text{ leaves node } i, \\ -1 & \text{arc } j \text{ enters node } i, \\ 0 & \text{otherwise.} \end{cases}$$



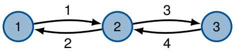
$$\boldsymbol{M} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$



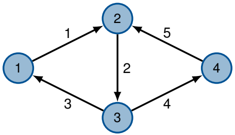
Definition: incidence matrix $\boldsymbol{M}$

Given $\mathcal{G} = (N, A)$, we define the incidence matrix $\boldsymbol{M} = (m_{ij}) \in \{-1, 0, 1\}^{|N| \times |A|}$ where $\forall i \in N$ and $j \in A$

$$m_{ij} = \begin{cases} 1 & \text{arc } j \text{ leaves node } i, \\ -1 & \text{arc } j \text{ enters node } i, \\ 0 & \text{otherwise.} \end{cases}$$



$$\boldsymbol{M} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

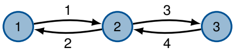


$$\boldsymbol{M} = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 & -1 \\ 0 & -1 & 1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix}$$

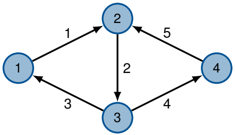
Definition: incidence matrix *M*

Given $\mathcal{G} = (N, A)$, we define the incidence matrix $\boldsymbol{M} = (m_{ij}) \in \{-1, 0, 1\}^{|N| \times |A|}$ where $\forall i \in N$ and $j \in A$

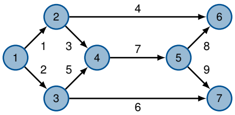
$$m_{ij} = \begin{cases} 1 & \text{arc } j \text{ leaves node } i, \\ -1 & \text{arc } j \text{ enters node } i, \\ 0 & \text{otherwise.} \end{cases}$$



$$\boldsymbol{M} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$



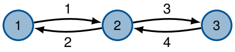
$$\boldsymbol{M} = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 & -1 \\ 0 & -1 & 1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix}$$



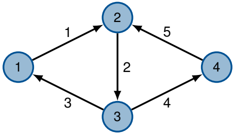
Definition: incidence matrix $\boldsymbol{M}$

Given $\mathcal{G} = (N, A)$, we define the incidence matrix $\boldsymbol{M} = (m_{ij}) \in \{-1, 0, 1\}^{|N| \times |A|}$ where $\forall i \in N$ and $j \in A$

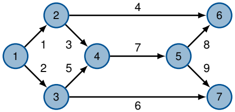
$$m_{ij} = \begin{cases} 1 & \text{arc } j \text{ leaves node } i, \\ -1 & \text{arc } j \text{ enters node } i, \\ 0 & \text{otherwise.} \end{cases}$$



$$\boldsymbol{M} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$



$$\boldsymbol{M} = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 & -1 \\ 0 & -1 & 1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix}$$



$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 \end{bmatrix}$$

Networks as graphs

Definition: undirected cycles in connected graphs

An undirected cycle $C \subset A$ is defined as vector $\boldsymbol{c} \in \{-1, 0, 1\}^{|A|}$ where

$$c_j = \begin{cases} 1 & \text{if } j \text{ is traversed in forward direction,} \\ -1 & \text{if } j \text{ is traversed in backward direction,} \\ 0 & \text{otherwise.} \end{cases}$$

The set of all cycles is denoted by $\mathcal{C}$.

1 $\mathbf{1} = [1, 1, \dots, 1]^T$

2 Proof via undirected tree in G, adding any further arc creates a cycle

3 Number of linearly independent undirected cycles

Definition: undirected cycles in connected graphs

An undirected cycle $C \subset A$ is defined as vector $\boldsymbol{c} \in \{-1, 0, 1\}^{|A|}$ where

$$c_j = \begin{cases} 1 & \text{if } j \text{ is traversed in forward direction,} \\ -1 & \text{if } j \text{ is traversed in backward direction,} \\ 0 & \text{otherwise.} \end{cases}$$

The set of all cycles is denoted by $\mathcal{C}$.

Definition: fundamental subspaces of $\boldsymbol{M}$ ($\mathcal{G}$ is connected)

$$\text{null}\, \boldsymbol{M}^\mathsf{T} = \text{span}\{\mathbf{1}\}^1$$

$$\text{null}\, \boldsymbol{M} = \text{span}\{\boldsymbol{c} : C \in \mathcal{C}\}$$

[1] $\mathbf{1} = [1, 1, \dots, 1]^T$

[2] Proof via undirected tree in G, adding any further arc creates a cycle

[3] Number of linearly independent undirected cycles

Networks as graphs

Definition: undirected cycles in connected graphs

An undirected cycle $C \subset A$ is defined as vector $\boldsymbol{c} \in \{-1, 0, 1\}^{|A|}$ where

$$c_j = \begin{cases} 1 & \text{if } j \text{ is traversed in forward direction,} \\ -1 & \text{if } j \text{ is traversed in backward direction,} \\ 0 & \text{otherwise.} \end{cases}$$

The set of all cycles is denoted by $\mathcal{C}$.

Definition: fundamental subspaces of $\boldsymbol{M}$ ($\mathcal{G}$ is connected)

$$\begin{aligned} \text{null } \boldsymbol{M}^\mathsf{T} &= \text{span}\{\mathbf{1}\}^1 \\ \text{null } \boldsymbol{M} &= \text{span}\{\boldsymbol{c} : C \in \mathcal{C}\} \end{aligned}$$

Dimensions:

- rank $\boldsymbol{M} = n - 1$ [2]
- dim null $\boldsymbol{M}^\mathsf{T} = 1$
- dim null $\boldsymbol{M} = m - n + 1$ [3]

---

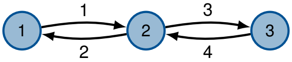
1 $\mathbf{1} = [1, 1, \dots, 1]^T$

2 Proof via undirected tree in G, adding any further arc creates a cycle

3 Number of linearly independent undirected cycles

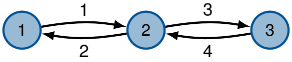
Networks as graphs

Examples: nullspace and cycles



Networks as graphs

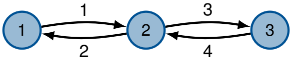
Examples: nullspace and cycles



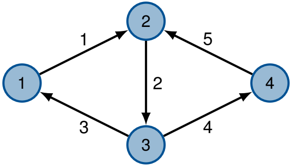
$$\text{null}\,\boldsymbol{M} = \text{span}\left\{ \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} \right\}$$

Networks as graphs

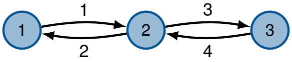
Examples: nullspace and cycles



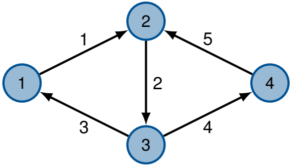
$$\operatorname{null} \boldsymbol{M} = \operatorname{span}\left\{ \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} \right\}$$



Examples: nullspace and cycles



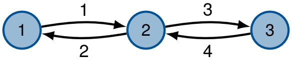
$$\operatorname{null} \boldsymbol{M} = \operatorname{span}\left\{ \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} \right\}$$



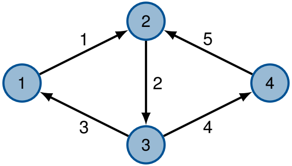
$$\operatorname{null} \boldsymbol{M} = \operatorname{span}\left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} \right\}$$

Networks as graphs

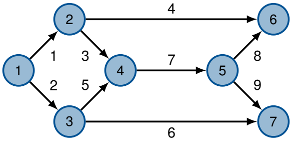
Examples: nullspace and cycles



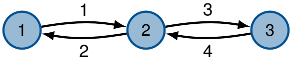
$$\operatorname{null} \boldsymbol{M} = \operatorname{span}\left\{ \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} \right\}$$



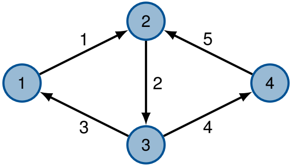
$$\operatorname{null} \boldsymbol{M} = \operatorname{span}\left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} \right\}$$



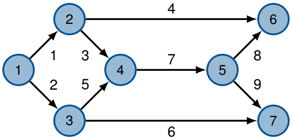
Examples: nullspace and cycles



$$\text{null}\,\boldsymbol{M} = \text{span}\left\{ \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} \right\}$$



$$\text{null}\,\boldsymbol{M} = \text{span}\left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} \right\}$$



$$\text{null}\,\boldsymbol{M} = \text{span}\left\{ \begin{bmatrix} 1 \\ -1 \\ 1 \\ 0 \\ -1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ -1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ -1 \\ 1 \\ 0 \\ 1 \end{bmatrix} \right\}$$

Networks as graphs

Flow problems

- Minimum cost flow problem
- Maximum s-t flow problem
- Min-cut and its capacity
- Minimum cost maximum s–t flow problem
- Multicommodity flow problems

Multicast in networks

Wireless Packet Networks

Flow problems

- The flow vector $\boldsymbol{x} = [x_1, \dots, x_m]^\mathsf{T}$ represents the amount of commodity (information) on each arc.
- The source vector $\boldsymbol{d} = [d_1, \dots, d_n]^\mathsf{T}$ represents the amount of commodity (information) that any node injects or consumes.
- Multiple information flows can be handled as a single commodity for routing / forwarding if they are
  - destined for a single common destination and
  - originate from a single common source.

Flow problems

- The flow vector $\boldsymbol{x} = [x_1, \dots, x_m]^\mathsf{T}$ represents the amount of commodity (information) on each arc.
- The source vector $\boldsymbol{d} = [d_1, \dots, d_n]^\mathsf{T}$ represents the amount of commodity (information) that any node injects or consumes.
- Multiple information flows can be handled as a single commodity for routing / forwarding if they are
  - destined for a single common destination and
  - originate from a single common source.

1. Nonnegativity of flows

$$\boldsymbol{x} \geq \boldsymbol{0} \qquad \Leftrightarrow \qquad x_j \geq 0 \quad \forall j \in A$$

Flow problems

- The flow vector $\boldsymbol{x} = [x_1, \dots, x_m]^{\mathsf{T}}$ represents the amount of commodity (information) on each arc.
- The source vector $\boldsymbol{d} = [d_1, \dots, d_n]^{\mathsf{T}}$ represents the amount of commodity (information) that any node injects or consumes.
- Multiple information flows can be handled as a single commodity for routing / forwarding if they are
  - destined for a single common destination and
  - originate from a single common source.

1. Nonnegativity of flows

$$\boldsymbol{x} \geq \boldsymbol{0} \qquad \Leftrightarrow \qquad x_j \geq 0 \quad \forall j \in A$$

2. Flow conservation law (Kirchhoff current law)

$$\boldsymbol{M}\boldsymbol{x} = \boldsymbol{d} \qquad \Leftrightarrow \qquad \sum_{(i,j)\in\mathcal{A}} x_{ij} - \sum_{(j,i)\in\mathcal{A}} x_{ji} = d_i \quad \forall i \in N$$

Flow problems

- The flow vector $\boldsymbol{x} = [x_1, \dots, x_m]^\mathsf{T}$ represents the amount of commodity (information) on each arc.
- The source vector $\boldsymbol{d} = [d_1, \dots, d_n]^\mathsf{T}$ represents the amount of commodity (information) that any node injects or consumes.
- Multiple information flows can be handled as a single commodity for routing / forwarding if they are
  - destined for a single common destination and
  - originate from a single common source.

1. Nonnegativity of flows

$$\boldsymbol{x} \geq \boldsymbol{0} \quad \Leftrightarrow \quad x_j \geq 0 \quad \forall j \in A$$

2. Flow conservation law (Kirchhoff current law)

$$\boldsymbol{Mx} = \boldsymbol{d} \quad \Leftrightarrow \quad \sum_{(i,j)\in\mathcal{A}} x_{ij} - \sum_{(j,i)\in\mathcal{A}} x_{ji} = d_i \quad \forall i \in N$$

- FCL cannot be satisfied if $\mathbf{1}^\mathsf{T}\boldsymbol{d} \neq 0$ since $\mathbf{1}^\mathsf{T}\boldsymbol{M} = \mathbf{0}$.

Flow problems

- The flow vector $\boldsymbol{x} = [x_1, \ldots, x_m]^{\mathrm{T}}$ represents the amount of commodity (information) on each arc.
- The source vector $\boldsymbol{d} = [d_1, \ldots, d_n]^{\mathrm{T}}$ represents the amount of commodity (information) that any node injects or consumes.
- Multiple information flows can be handled as a single commodity for routing / forwarding if they are
  - destined for a single common destination and
  - originate from a single common source.

1. Nonnegativity of flows

$$\boldsymbol{x} \geq \mathbf{0} \qquad \Leftrightarrow \qquad x_j \geq 0 \quad \forall j \in A$$

2. Flow conservation law (Kirchhoff current law)

$$\boldsymbol{Mx} = \boldsymbol{d} \qquad \Leftrightarrow \qquad \sum_{(i,j)\in\mathcal{A}} x_{ij} - \sum_{(j,i)\in\mathcal{A}} x_{ji} = d_i \quad \forall i \in N$$

- FCL cannot be satisfied if $\mathbf{1}^{\mathrm{T}}\boldsymbol{d} \neq 0$ since $\mathbf{1}^{\mathrm{T}}\boldsymbol{M} = \mathbf{0}$.
- FCL contains exactly one redundant constraint since rank $\boldsymbol{M} = n - 1$ (if graph is connected).

Flow problems

- The flow vector $\boldsymbol{x} = [x_1, \dots, x_m]^\mathsf{T}$ represents the amount of commodity (information) on each arc.
- The source vector $\boldsymbol{d} = [d_1, \dots, d_n]^\mathsf{T}$ represents the amount of commodity (information) that any node injects or consumes.
- Multiple information flows can be handled as a single commodity for routing / forwarding if they are
  - destined for a single common destination and
  - originate from a single common source.

1. Nonnegativity of flows

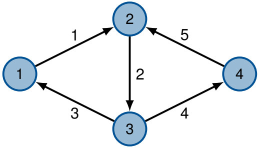
$$\boldsymbol{x} \geq \boldsymbol{0} \quad \Leftrightarrow \quad x_j \geq 0 \quad \forall j \in A$$

2. Flow conservation law (Kirchhoff current law)

$$\boldsymbol{Mx} = \boldsymbol{d} \quad \Leftrightarrow \quad \sum_{(i,j)\in\mathcal{A}} x_{ij} - \sum_{(j,i)\in\mathcal{A}} x_{ji} = d_i \quad \forall i \in N$$

- FCL cannot be satisfied if $\boldsymbol{1}^\mathsf{T}\boldsymbol{d} \neq 0$ since $\boldsymbol{1}^\mathsf{T}\boldsymbol{M} = \boldsymbol{0}$.
- FCL contains exactly one redundant constraint since rank $\boldsymbol{M} = n - 1$ (if graph is connected).
- Flows along directed cycles are independent of $\boldsymbol{d}$, i. e., flows that satisfy $\boldsymbol{Mx} = \boldsymbol{0}$, $\boldsymbol{x} \geq \boldsymbol{0}$.

Example 1: Diamond network from $s = 1$ to $t = 4$



- Incidence matrix and source vector:

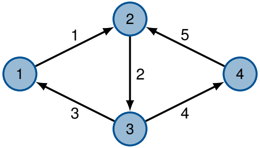
$$\boldsymbol{M} = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 & -1 \\ 0 & -1 & 1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \qquad \boldsymbol{d} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ -1 \end{bmatrix}$$

- Feasible flows for $\boldsymbol{M}, \boldsymbol{d}$:

$$\mathcal{F}(\boldsymbol{M}, \boldsymbol{d}) = \{\boldsymbol{x} : \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d}, \boldsymbol{x} \geq 0\}$$

- Flow solution(s) (Unique? How many solutions?)

Example 1: Diamond network from $s = 1$ to $t = 4$



- Incidence matrix and source vector:

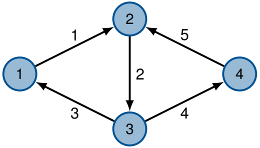
$$\boldsymbol{M} = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 & -1 \\ 0 & -1 & 1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \qquad \boldsymbol{d} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ -1 \end{bmatrix}$$

- Feasible flows for $\boldsymbol{M}, \boldsymbol{d}$:

$$\mathcal{F}(\boldsymbol{M}, \boldsymbol{d}) = \{\boldsymbol{x} : \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d}, \boldsymbol{x} \geq 0\}$$

- Flow solution(s) (Unique? How many solutions?)
  - $\boldsymbol{x}^{\mathsf{T}} = [\,1\ 1\ 0\ 1\ 0\,]$

Example 1: Diamond network from $s = 1$ to $t = 4$



- Incidence matrix and source vector:

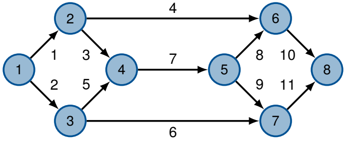
$$M = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 & -1 \\ 0 & -1 & 1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \qquad d = \begin{bmatrix} 1 \\ 0 \\ 0 \\ -1 \end{bmatrix}$$

- Feasible flows for $\boldsymbol{M}$, $\boldsymbol{d}$:

$$\mathcal{F}(\boldsymbol{M}, \boldsymbol{d}) = \{\boldsymbol{x} : \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d}, \boldsymbol{x} \geq 0\}$$

- Flow solution(s) (Unique? How many solutions?)
  - $\boldsymbol{x}^\top = [1\ 1\ 0\ 1\ 0]$
  - $\boldsymbol{x}^\top = [1\ 1\ 0\ 1\ 0] + \alpha\,[1\ 1\ 1\ 0\ 0] + \beta\,[0\ 1\ 0\ 1\ 1],\ \alpha, \beta \geq 0$

Example 2: Extended butterfly from $s = 1$ to $t = 8$



- Incidence matrix and source vector:

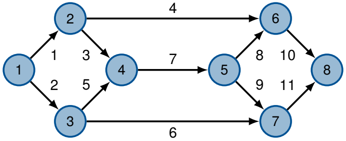
$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \end{bmatrix} \qquad \boldsymbol{d} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \end{bmatrix}$$

- Feasible flows for $\boldsymbol{M}$, $\boldsymbol{d}$:

$$\mathcal{F}(\boldsymbol{M}, \boldsymbol{d}) = \{\boldsymbol{x} : \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d}, \boldsymbol{x} \geq 0\}$$

- Flow solution(s) (Unique? How many?)

Example 2: Extended butterfly from $s = 1$ to $t = 8$



- Incidence matrix and source vector:

$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \end{bmatrix} \qquad \boldsymbol{d} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \end{bmatrix}$$

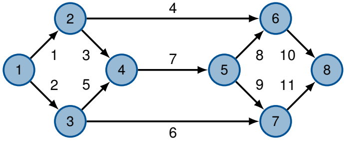
- Feasible flows for $\boldsymbol{M}$, $\boldsymbol{d}$:

$$\mathcal{F}(\boldsymbol{M}, \boldsymbol{d}) = \{\boldsymbol{x} : \boldsymbol{Mx} = \boldsymbol{d}, \boldsymbol{x} \geq 0\}$$

- Flow solution(s) (Unique? How many?)
  - $\boldsymbol{x}^{\mathrm{T}} = [\,1\;0\;0\;1\;0\;0\;0\;0\;0\;1\;0\,]$

Flow problems

Example 2: Extended butterfly from $s = 1$ to $t = 8$



- Incidence matrix and source vector:

$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \end{bmatrix} \qquad \boldsymbol{d} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \end{bmatrix}$$

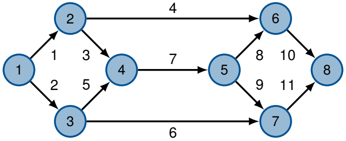
- Feasible flows for $\boldsymbol{M}, \boldsymbol{d}$:

$$\mathcal{F}(\boldsymbol{M}, \boldsymbol{d}) = \{\boldsymbol{x} : \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d}, \boldsymbol{x} \geq 0\}$$

- Flow solution(s) (Unique? How many?)
  - $\boldsymbol{x}^T = [1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0]$
  - $\boldsymbol{x}^T = [0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1]$

Flow problems

Example 2: Extended butterfly from $s = 1$ to $t = 8$



- Incidence matrix and source vector:

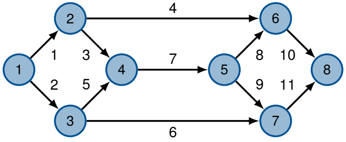
$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \end{bmatrix} \qquad \boldsymbol{d} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \end{bmatrix}$$

- Feasible flows for $\boldsymbol{M}, \boldsymbol{d}$:

$$\mathcal{F}(\boldsymbol{M}, \boldsymbol{d}) = \{\boldsymbol{x} : \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d}, \boldsymbol{x} \geq 0\}$$

- Flow solution(s) (Unique? How many?)
  - $\boldsymbol{x}^\mathsf{T} = [\,1\;0\;0\;1\;0\;0\;0\;0\;0\;1\;0\,]$
  - $\boldsymbol{x}^\mathsf{T} = [\,0\;1\;0\;0\;1\;0\;1\;0\;1\;0\;1\,]$
  - $\boldsymbol{x}^\mathsf{T} = \lambda\,[\,1\;0\;0\;1\;0\;0\;0\;0\;0\;1\;0\,] + (1-\lambda)\,[\,0\;1\;0\;0\;1\;0\;1\;0\;1\;0\;1\,],\ \lambda \in [0,1]$

Example 2: Extended butterfly from $s = 1$ to $t = 8$



- Incidence matrix and source vector:

$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \end{bmatrix} \qquad \boldsymbol{d} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \end{bmatrix}$$

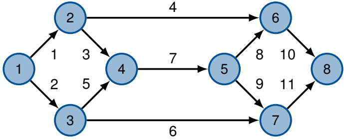
- Feasible flows for $\boldsymbol{M}, \boldsymbol{d}$:

$$\mathcal{F}(\boldsymbol{M}, \boldsymbol{d}) = \{\boldsymbol{x} : \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d}, \boldsymbol{x} \geq 0\}$$

- Flow solution(s) (Unique? How many?)
  - $\boldsymbol{x}^\mathsf{T} = [\,1\;0\;0\;1\;0\;0\;0\;0\;0\;1\;0\,]$
  - $\boldsymbol{x}^\mathsf{T} = [\,0\;1\;0\;0\;1\;0\;1\;0\;1\;0\;1\,]$
  - $\boldsymbol{x}^\mathsf{T} = \lambda\,[\,1\;0\;0\;1\;0\;0\;0\;0\;0\;1\;0\,] + (1-\lambda)\,[\,0\;1\;0\;0\;1\;0\;1\;0\;1\;0\;1\,],\ \lambda \in [0,1]$
  - ...

Flow problems

Example 3: Flows from multiple sources to a single destination



- Incidence matrix and source vector:

$$
\boldsymbol{M} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \end{bmatrix} \qquad \boldsymbol{d} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ -4 \end{bmatrix}
$$

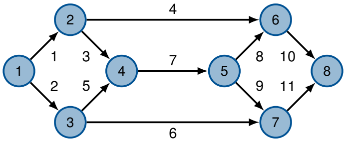
- Feasible flows for $\boldsymbol{M}$, $\boldsymbol{d}$

$$
\mathcal{F}(\boldsymbol{M}, \boldsymbol{d}) = \{\boldsymbol{x} : \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d}, \boldsymbol{x} \geq 0\}
$$

- Flow solution(s) (Unique? How many?)

Flow problems

Example 3: Flows from multiple sources to a single destination



- Incidence matrix and source vector:

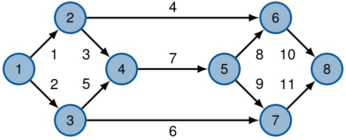
$$
\boldsymbol{M} = \begin{bmatrix}
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1
\end{bmatrix}
\qquad
\boldsymbol{d} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ -4 \end{bmatrix}
$$

- Feasible flows for $\boldsymbol{M}$, $\boldsymbol{d}$

$$
\mathcal{F}(\boldsymbol{M}, \boldsymbol{d}) = \{\boldsymbol{x} : \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d}, \boldsymbol{x} \geq 0\}
$$

- Flow solution(s) (Unique? How many?)
  - $\boldsymbol{x} = [\,0\;1\;0\;1\;1\;1\;2\;1\;1\;2\;2\,]$

Example 3: Flows from multiple sources to a single destination



- Incidence matrix and source vector:

$$
\boldsymbol{M} = \begin{bmatrix}
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1
\end{bmatrix}
\qquad
\boldsymbol{d} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ -4 \end{bmatrix}
$$

- Feasible flows for $\boldsymbol{M}$, $\boldsymbol{d}$

$$
\mathcal{F}(\boldsymbol{M}, \boldsymbol{d}) = \{\boldsymbol{x} : \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d}, \boldsymbol{x} \geq 0\}
$$

- Flow solution(s) (Unique? How many?)
  - $\boldsymbol{x} = [\,0\ 1\ 0\ 1\ 1\ 1\ 2\ 1\ 1\ 2\ 2\,]$
  - ...

Definition: feasible flow region

Given the incidence matrix $\boldsymbol{M}$ of a connected graph $\mathcal{G} = (N, A)$ and a source vector $\boldsymbol{d} \geq \mathbf{0}$, the feasible flow region is given by

$$\mathcal{F}(\boldsymbol{M}, \boldsymbol{d}) = \{\boldsymbol{x} : \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d}, \boldsymbol{x} \geq 0\},$$

which is

- a closed[1] polyhedral[2] convex[3] set,
- nonempty if $\mathbf{1}^\top \boldsymbol{d} = 0$ (and G is connected),
- bounded[4] if G is acyclic (contains no directed cycles), i. e., $\mathcal{F}(\boldsymbol{M}, \mathbf{0}) = \{\mathbf{0}\}$,
- and, in general, contains infinitely many solutions.

1 A set $\mathcal{X}$ is closed if it contains all its limit points.
2 A set $\mathcal{X}$ is a polyhedron if it is defined by a finite number of affine (in)equalities, i.e., $\mathcal{X} = \{\boldsymbol{x} : A\boldsymbol{x} \geq \boldsymbol{b}\}$.
3 A set $\mathcal{X}$ is convex if for any two points $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$ and any real scalar $\lambda \in [0, 1]$, $\lambda \boldsymbol{x} + (1 - \lambda)\boldsymbol{y} \in \mathcal{X}$.
4 A set $\mathcal{X}$ is bounded if it is contained in some ball around the origin, i.e., $\mathcal{X} \subset B_r(\mathbf{0})$ for some $r > 0$.

Minimum cost flow problem

Uncapacitated minimum cost flow problem

Cost per unit flow on arcs: $\boldsymbol{c} = [c_1, \dots, c_m]^\mathsf{T}$

$$\begin{aligned} \min \quad & \boldsymbol{c}^\mathsf{T} \boldsymbol{x} \\ \text{s. t.} \quad & \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d} \\ & \boldsymbol{x} \geq \boldsymbol{0} \end{aligned}$$

Capacitated minimum cost flow problem

$\boldsymbol{z} = [z_1, \dots, z_m]^\mathsf{T}$ maximum flow on each arc

$$\begin{aligned} \min \quad & \boldsymbol{c}^\mathsf{T} \boldsymbol{x} \\ \text{s. t.} \quad & \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d} \\ & \boldsymbol{x} \leq \boldsymbol{z} \\ & \boldsymbol{x} \geq \boldsymbol{0} \end{aligned}$$

[5] Not all flow solutions to these two problems describe shortest paths, but at least one does.

Minimum cost flow problem

Uncapacitated minimum cost flow problem

Cost per unit flow on arcs: $\boldsymbol{c} = [c_1, \dots, c_m]^\mathsf{T}$

$$\begin{aligned} \min \quad & \boldsymbol{c}^\mathsf{T}\boldsymbol{x} \\ \text{s. t.} \quad & \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d} \\ & \boldsymbol{x} \geq \boldsymbol{0} \end{aligned}$$

Capacitated minimum cost flow problem

$\boldsymbol{z} = [z_1, \dots, z_m]^\mathsf{T}$ maximum flow on each arc

$$\begin{aligned} \min \quad & \boldsymbol{c}^\mathsf{T}\boldsymbol{x} \\ \text{s. t.} \quad & \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d} \\ & \boldsymbol{x} \leq \boldsymbol{z} \\ & \boldsymbol{x} \geq \boldsymbol{0} \end{aligned}$$

Example: Shortest path[5]

- $\boldsymbol{c}$ "length" of each arc, e.g., $\boldsymbol{c} = \boldsymbol{1}$ (number of hops metric)
- Shortest path from s to t: $d_s = 1$, $d_t = -1$, $d_i = 0 \ \forall\ i \neq s, t$
- Simultaneous shortest paths to t: $d_t = -n + 1$, $d_i = 1 \ \forall\ i \neq t$

[5] Not all flow solutions to these two problems describe shortest paths, but at least one does.

Minimum cost flow problem

Uncapacitated minimum cost flow problem

Cost per unit flow on arcs: $\boldsymbol{c} = [c_1, \ldots, c_m]^{\mathsf{T}}$

$$\begin{aligned} \min \quad & \boldsymbol{c}^{\mathsf{T}} \boldsymbol{x} \\ \text{s.t.} \quad & \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d} \\ & \boldsymbol{x} \geq \boldsymbol{0} \end{aligned}$$

Solution approaches

- General purpose linear programming solver (Simplex, Interior point, etc.)
- Specialized algorithms (Dijkstra, Bellman-Ford, network simplex, etc.) exploiting graph structure and recursive structure of the optimal solution (if available)

Maximum s-t flow problem

Capacitated minimum cost flow problem

$\boldsymbol{z} = [z_1, \dots, z_m]^\mathsf{T}$ maximum flow on each arc with source vector $d_s = 1$, $d_t = -1$, $d_i = 0 \; \forall \; i \neq s, t$

$$\begin{aligned} \max \quad & r \\ \text{s. t.} \quad & \boldsymbol{Mx} = r\boldsymbol{d} \\ & \boldsymbol{x} \leq \boldsymbol{z} \\ & \boldsymbol{x} \geq \boldsymbol{0} \end{aligned}$$

Solution approaches

- General purpose linear programming solver (Simplex, Interior point, etc.)
- Lagrangian duality approaches (selectively relax one constraint)
- Specialized algorithms (Ford-Fulkerson) exploiting graph structure and relation to min-cut

Min-cut and its capacity

- An s–t cut is a subset of nodes $S \subset N$ such that $s \in S$ and $t \notin S$.
- An arc $(i, j) \in \mathcal{A}$ crosses S if $i \in S$ and $j \notin S$.
- $\mathcal{A}(S)$ denotes all crossing arcs.
- The value of an s–t cut given the capacity vector $\boldsymbol{z}$ is defined as

$$v(S) = \sum_{(i,j) \in \mathcal{A}(S)} z_{ij}.$$

- The value of any s–t cut upper bounds the maximum s–t flow.

Min-cut and its capacity

- An s–t cut is a subset of nodes $S \subset N$ such that $s \in S$ and $t \notin S$.
- An arc $(i, j) \in \mathcal{A}$ crosses S if $i \in S$ and $j \notin S$.
- $\mathcal{A}(S)$ denotes all crossing arcs.
- The value of an s–t cut given the capacity vector $\boldsymbol{z}$ is defined as

$$v(S) = \sum_{(i,j)\in\mathcal{A}(S)} z_{ij}.$$

- The value of any s–t cut upper bounds the maximum s–t flow.

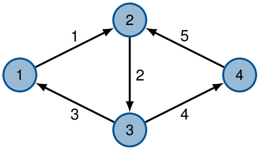
Max-flow min-cut theorem

The value of the minimum s–t cut equals the value of the maximum s–t flow, i.e.,

$$\max\{r : \boldsymbol{Mx} = r\boldsymbol{d},\ \boldsymbol{0} \le \boldsymbol{x} \le \boldsymbol{z}\} = \min\{v(S) : S \text{ is } s\text{–}t \text{ cut}\}.$$

Min-cut and its capacity

Example 1: Diamond network from $s = 1$ to $t = 4$

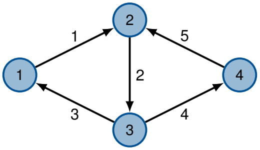


- Incidence matrix, source vector, capacity vector:

$$\boldsymbol{M} = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 & -1 \\ 0 & -1 & 1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \qquad \boldsymbol{d} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ -1 \end{bmatrix} \qquad \boldsymbol{z} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Min-cut and its capacity

Example 1: Diamond network from $s = 1$ to $t = 4$



- Incidence matrix, source vector, capacity vector:

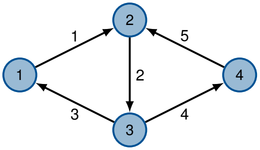
$$\boldsymbol{M} = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 & -1 \\ 0 & -1 & 1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \qquad \boldsymbol{d} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ -1 \end{bmatrix} \qquad \boldsymbol{z} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

- Max-flow:

$$\max\{r : \boldsymbol{M}\boldsymbol{x} = r\boldsymbol{d},\ \boldsymbol{0} \leq \boldsymbol{x} \leq \boldsymbol{z}\} = 1$$

Min-cut and its capacity

Example 1: Diamond network from $s = 1$ to $t = 4$



- Incidence matrix, source vector, capacity vector:

$$M = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 & -1 \\ 0 & -1 & 1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \qquad d = \begin{bmatrix} 1 \\ 0 \\ 0 \\ -1 \end{bmatrix} \qquad z = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

- Max-flow:

$$\max\{r : \boldsymbol{Mx} = r\boldsymbol{d},\ \boldsymbol{0} \leq \boldsymbol{x} \leq \boldsymbol{z}\} = 1$$

- Min-cut:

$$\min\{v(S) : S \text{ is } s\text{-}t \text{ cut}\} = 1$$

Minimum cost maximum s–t flow problem

Generalizes (uncapacitated) minimum cost and (capacitated) maximum flow s–t problem:

- Source and flow vector: $\boldsymbol{d}$, $\boldsymbol{x}$
- Capacity and cost vector: $\boldsymbol{z}$, $\boldsymbol{c}$

$$\begin{aligned} \min \quad & \boldsymbol{c}^{\mathsf{T}}\boldsymbol{x} \\ \text{s. t.} \quad & \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d} \\ & \boldsymbol{x} \geq \boldsymbol{0} \\ & \boldsymbol{x} \leq \boldsymbol{z} \end{aligned}$$

Minimum cost maximum s–t flow problem

Generalizes (uncapacitated) minimum cost and (capacitated) maximum flow s–t problem:

- Source and flow vector: $\boldsymbol{d}$, $\boldsymbol{x}$
- Capacity and cost vector: $\boldsymbol{z}$, $\boldsymbol{c}$

$$\begin{aligned} \min \quad & \boldsymbol{c}^\top \boldsymbol{x} \\ \text{s. t.} \quad & \boldsymbol{M}\boldsymbol{x} = \boldsymbol{d} \\ & \boldsymbol{x} \geq \boldsymbol{0} \\ & \boldsymbol{x} \leq \boldsymbol{z} \end{aligned}$$

Special cases

- Maximum s–t flow (see tutorial)
- Minimum cost flow (capacitated $\boldsymbol{z} < \infty$, uncapacitated $\boldsymbol{z} = \infty$)

Multicommodity flow problems

In contrast to single-commodity flow problems we now have multiple commodities, e. g. flows, that compete with each other:

- Commodities $C = \{1, \dots, c\}$,
- Source, flow, and cost vector of commodity k: $\boldsymbol{d}_k$, $\boldsymbol{x}_k$, $\boldsymbol{c}_k$
- Capacity shared across all commodities: $\boldsymbol{z}$

The min-cost max-flow problem then reads as:

$$\begin{aligned}
\min \quad & \sum_{k \in C} \boldsymbol{c}_k^T \boldsymbol{x}_k \\
\text{s. t.} \quad & \boldsymbol{M}\boldsymbol{x}_k = \boldsymbol{d}_k \quad \forall k \in C \\
& \boldsymbol{x}_k \geq \boldsymbol{0} \quad \forall k \in C \\
& \sum_{k \in C} \boldsymbol{x}_k \leq \boldsymbol{z}
\end{aligned}$$

Multicommodity flow problems

In contrast to single-commodity flow problems we now have multiple commodities, e. g. flows, that compete with each other:

- Commodities $C = \{1, \dots, c\}$,
- Source, flow, and cost vector of commodity k: $\boldsymbol{d}_k$, $\boldsymbol{x}_k$, $\boldsymbol{c}_k$
- Capacity shared across all commodities: $\boldsymbol{z}$

The min-cost max-flow problem then reads as:

$$
\begin{aligned}
\min \quad & \sum_{k \in C} \boldsymbol{c}_k^T \boldsymbol{x}_k \\
\text{s. t.} \quad & \boldsymbol{M}\boldsymbol{x}_k = \boldsymbol{d}_k \quad \forall k \in C \\
& \boldsymbol{x}_k \geq \boldsymbol{0} \quad \forall k \in C \\
& \sum_{k \in C} \boldsymbol{x}_k \leq \boldsymbol{z}
\end{aligned}
$$

Properties

- Flow conservation applies to all commodities individually
- Capacity is shared among all commodities

Optimality of a solution now even more depends on what is considered “optimal”:

- The previous definition is a joint optimization of the weighted sum rate $\sum_k \boldsymbol{c}_k^\mathsf{T} \boldsymbol{x}_k$.
- This allows that commodities (flows) are assigned few or no resources at all.
- Fairness?

Multicommodity flow problems

Optimality of a solution now even more depends on what is considered “optimal”:

- The previous definition is a joint optimization of the weighted sum rate $\sum_k \boldsymbol{c}_k^{\mathsf{T}} \boldsymbol{x}_k$.
- This allows that commodities (flows) are assigned few or no resources at all.
- Fairness?

Additional constraints, e. g. assigning a minimum amount of resources for individual commodities, may amend the cost function.

Optimality of a solution now even more depends on what is considered “optimal”:

- The previous definition is a joint optimization of the weighted sum rate $\sum_k \boldsymbol{c}_k^{\mathrm{T}} \boldsymbol{x}_k$.
- This allows that commodities (flows) are assigned few or no resources at all.
- Fairness?

Additional constraints, e. g. assigning a minimum amount of resources for individual commodities, may amend the cost function.

Solution approaches

- General purpose linear programming solver
- Lagrangian duality approaches (selectively relax one constraint, mostly the capacity constraint which couples all flows)

Networks as graphs

Flow problems

Multicast in networks

- Store-forward multicast
- Multicast tree-based forwarding
- Multicast with network coding

Wireless Packet Networks

Multicast in networks

Multicast in networks as flow problems

- Multicast communication is identified by its terminal set $T \subset N$.
- We can consider one or multiple sources (there is no big difference from a theoretical perspective).
- Special cases:

Multicast in networks

Multicast in networks as flow problems

- Multicast communication is identified by its terminal set $T \subset N$.
- We can consider one or multiple sources (there is no big difference from a theoretical perspective).
- Special cases:
  - unicast (one source, one terminal)
  - bidirectional communication (two nodes that are sources and terminals)
  - broadcast (all nodes other than the source are terminals)

Multicast in networks

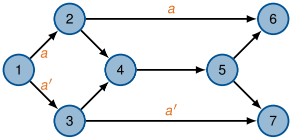
Multicast in networks as flow problems

- Multicast communication is identified by its terminal set $T \subset N$.
- We can consider one or multiple sources (there is no big difference from a theoretical perspective).
- Special cases:
  - unicast (one source, one terminal)
  - bidirectional communication (two nodes that are sources and terminals)
  - broadcast (all nodes other than the source are terminals)

How is multicast treated in networks?

- Convert to unicasts
  $\rightarrow$ replicate packets at source and store-forward at all other nodes
- Allow replication at all nodes
  $\rightarrow$ multicast tree / Steiner tree based forwarding
- Allow coding at all nodes
  $\rightarrow$ network coding

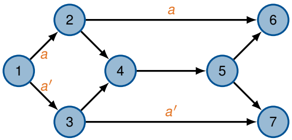
Store-forward multicast



- The flows to the terminals are independent of each other.
- Capacity needs to be split among all flows.

max s–T flow problem:

Store-forward multicast



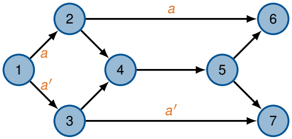
- The flows to the terminals are independent of each other.
- Capacity needs to be split among all flows.

max s–T flow problem:

- One commodity for each terminal $t \in T$
- Source vector $\boldsymbol{d}_{st}$ such that $d_{st,s} = 1$, $d_{st,t} = -1$, and $d_{st,i} = 0$ otherwise
- Capacity vector $\boldsymbol{z}$ split among commodities

$$\max\ r \quad \text{s.t.} \quad \begin{aligned} \boldsymbol{M}\boldsymbol{x}_t &= r\boldsymbol{d}_{st} && \forall t \in T \\ \boldsymbol{x}_t &\geq \boldsymbol{0} && \forall t \in T \\ \sum_{t \in T} \boldsymbol{x}_t &\leq \boldsymbol{z} \end{aligned}$$

Store-forward multicast



- The flows to the terminals are independent of each other.
- Capacity needs to be split among all flows.

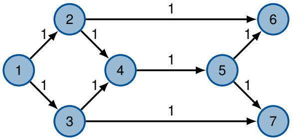
max s–T flow problem:

- One commodity for each terminal $t \in T$
- Source vector $\boldsymbol{d}_{st}$ such that $d_{st,s} = 1$, $d_{st,t} = -1$, and $d_{st,i} = 0$ otherwise
- Capacity vector $\boldsymbol{z}$ split among commodities

$$\begin{aligned} \max\ r \quad \text{s. t.} \quad \boldsymbol{M}\boldsymbol{x}_t &= r\boldsymbol{d}_{st} & \forall t \in T \\ \boldsymbol{x}_t &\geq \boldsymbol{0} & \forall t \in T \\ \sum_{t \in T} \boldsymbol{x}_t &\leq \boldsymbol{z} & \end{aligned}$$

$\Rightarrow$ That is a multicommodity flow problem!

Store-forward multicast

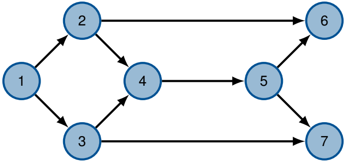


- Optimal flow solutions
  - $\boldsymbol{x}_6 = [1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0]^T$
  - $\boldsymbol{x}_7 = [0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0]^T$
- Total flow which is capacity relevant
  - $\boldsymbol{x}_6 + \boldsymbol{x}_7 = [1\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 0]^T$

maximum multicast s–T flow = 1

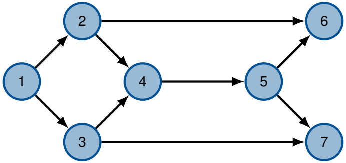
Multicast tree-based forwarding

- s–T multicast tree: a tree rooted at s such that there exists a directed path to each $t \in T$ (arcs belong to at least one path).
- Unit flow on multicast tree delivers one unit (the same unit) of information to each terminal.



Multicast tree-based forwarding

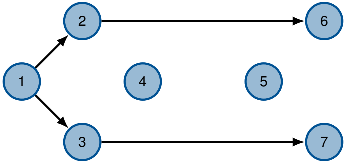
- s–T multicast tree: a tree rooted at s such that there exists a directed path to each $t \in T$ (arcs belong to at least one path).
- Unit flow on multicast tree delivers one unit (the same unit) of information to each terminal.



- Example 1–$\{6, 7\}$ multicast trees:

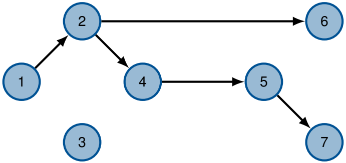
Multicast tree-based forwarding

- s–T multicast tree: a tree rooted at s such that there exists a directed path to each $t \in T$ (arcs belong to at least one path).
- Unit flow on multicast tree delivers one unit (the same unit) of information to each terminal.



- Example 1–$\{6, 7\}$ multicast trees:
  - (1, 2), (1, 3), (2, 6), (3, 7)

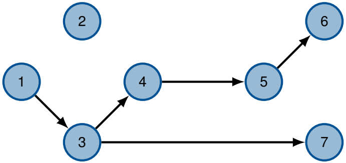
- s–T multicast tree: a tree rooted at s such that there exists a directed path to each $t \in T$ (arcs belong to at least one path).
- Unit flow on multicast tree delivers one unit (the same unit) of information to each terminal.



- Example 1–$\{6, 7\}$ multicast trees:
  - $(1, 2), (1, 3), (2, 6), (3, 7)$
  - $(1, 2), (2, 4), (2, 6), (4, 5), (5, 7)$

Multicast tree-based forwarding

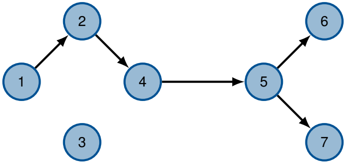
- s–T multicast tree: a tree rooted at s such that there exists a directed path to each $t \in T$ (arcs belong to at least one path).
- Unit flow on multicast tree delivers one unit (the same unit) of information to each terminal.



- Example 1–$\{6, 7\}$ multicast trees:
  - (1, 2), (1, 3), (2, 6), (3, 7)
  - (1, 2), (2, 4), (2, 6), (4, 5), (5, 7)
  - (1, 3), (3, 4), (3, 7), (4, 5), (5, 6)

Multicast tree-based forwarding

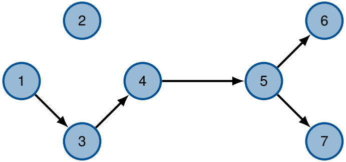
- s–T multicast tree: a tree rooted at s such that there exists a directed path to each $t \in T$ (arcs belong to at least one path).
- Unit flow on multicast tree delivers one unit (the same unit) of information to each terminal.



- Example 1–$\{6, 7\}$ multicast trees:
  - (1, 2), (1, 3), (2, 6), (3, 7)
  - (1, 2), (2, 4), (2, 6), (4, 5), (5, 7)
  - (1, 3), (3, 4), (3, 7), (4, 5), (5, 6)
  - (1, 2), (2, 4), (4, 5), (5, 6), (5, 7)

Multicast tree-based forwarding

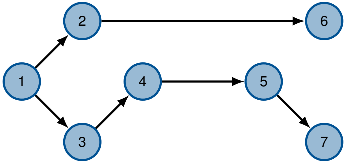
- s–T multicast tree: a tree rooted at s such that there exists a directed path to each $t \in T$ (arcs belong to at least one path).
- Unit flow on multicast tree delivers one unit (the same unit) of information to each terminal.



- Example 1–$\{6, 7\}$ multicast trees:
  - (1, 2), (1, 3), (2, 6), (3, 7)
  - (1, 2), (2, 4), (2, 6), (4, 5), (5, 7)
  - (1, 3), (3, 4), (3, 7), (4, 5), (5, 6)
  - (1, 2), (2, 4), (4, 5), (5, 6), (5, 7)
  - (1, 3), (3, 4), (4, 5), (5, 6), (5, 7)

Multicast tree-based forwarding

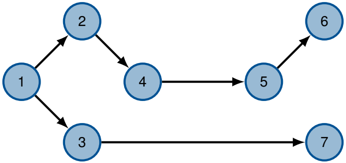
- s–T multicast tree: a tree rooted at s such that there exists a directed path to each $t \in T$ (arcs belong to at least one path).
- Unit flow on multicast tree delivers one unit (the same unit) of information to each terminal.



- Example 1–$\{6, 7\}$ multicast trees:
  - (1, 2), (1, 3), (2, 6), (3, 7)
  - (1, 2), (2, 4), (2, 6), (4, 5), (5, 7)
  - (1, 3), (3, 4), (3, 7), (4, 5), (5, 6)
  - (1, 2), (2, 4), (4, 5), (5, 6), (5, 7)
  - (1, 3), (3, 4), (4, 5), (5, 6), (5, 7)
  - (1, 2), (1, 3), (2, 6), (3, 4), (4, 5), (5, 7)

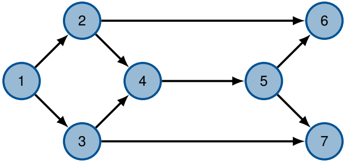
Multicast tree-based forwarding

- s–T multicast tree: a tree rooted at s such that there exists a directed path to each $t \in T$ (arcs belong to at least one path).
- Unit flow on multicast tree delivers one unit (the same unit) of information to each terminal.



- Example 1–$\{6, 7\}$ multicast trees:
  - $(1, 2), (1, 3), (2, 6), (3, 7)$
  - $(1, 2), (2, 4), (2, 6), (4, 5), (5, 7)$
  - $(1, 3), (3, 4), (3, 7), (4, 5), (5, 6)$
  - $(1, 2), (2, 4), (4, 5), (5, 6), (5, 7)$
  - $(1, 3), (3, 4), (4, 5), (5, 6), (5, 7)$
  - $(1, 2), (1, 3), (2, 6), (3, 4), (4, 5), (5, 7)$
  - $(1, 2), (1, 3), (2, 4), (3, 7), (4, 5), (5, 6)$

- s–T multicast tree: a tree rooted at s such that there exists a directed path to each $t \in T$ (arcs belong to at least one path).
- Unit flow on multicast tree delivers one unit (the same unit) of information to each terminal.



- Example 1–$\{6, 7\}$ multicast trees:
  - (1, 2), (1, 3), (2, 6), (3, 7)
  - (1, 2), (2, 4), (2, 6), (4, 5), (5, 7)
  - (1, 3), (3, 4), (3, 7), (4, 5), (5, 6)
  - (1, 2), (2, 4), (4, 5), (5, 6), (5, 7)
  - (1, 3), (3, 4), (4, 5), (5, 6), (5, 7)
  - (1, 2), (1, 3), (2, 6), (3, 4), (4, 5), (5, 7)
  - (1, 2), (1, 3), (2, 4), (3, 7), (4, 5), (5, 6)
- Optimal solution is a superposition of those multicast trees
  - subject to the capacity constraints but
  - without explicit flow conservation law.

Multicast tree-based forwarding

max s–T flow problem

- Multicast trees $\mathrm{MT}_{sT} = \{1, \dots, K\}$
- Multicast tree incidence vector $\boldsymbol{x}_k$ such that $x_{k,j} = 1$ if arc j is in the k-th multicast tree, otherwise $x_{k,j} = 0$
- One commodity for the multicast, no flow conservation constraint
- Capacity vector $\boldsymbol{z}$ split among all trees

$$\begin{aligned}
\max \quad & \sum_{k \in \mathrm{MT}_{sT}} r_k \\
\text{s. t.} \quad & \sum_{k \in \mathrm{MT}_{sT}} r_k \boldsymbol{x}_k \leq \boldsymbol{z} \\
& r_k \geq 0 \quad \forall k \in \mathrm{MT}_{sT}
\end{aligned}$$

Multicast tree-based forwarding

max s–T flow problem

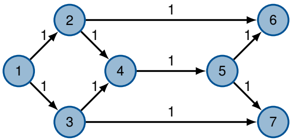
- Multicast trees $\mathrm{MT}_{sT} = \{1, \dots, K\}$
- Multicast tree incidence vector $\boldsymbol{x}_k$ such that $x_{k,j} = 1$ if arc j is in the k-th multicast tree, otherwise $x_{k,j} = 0$
- One commodity for the multicast, no flow conservation constraint
- Capacity vector $\boldsymbol{z}$ split among all trees

$$
\begin{aligned}
\max \quad & \sum_{k \in \mathrm{MT}_{sT}} r_k \\
\text{s. t.} \quad & \sum_{k \in \mathrm{MT}_{sT}} r_k \boldsymbol{x}_k \le \boldsymbol{z} \\
& r_k \ge 0 \quad \forall k \in \mathrm{MT}_{sT}
\end{aligned}
$$

Notes:

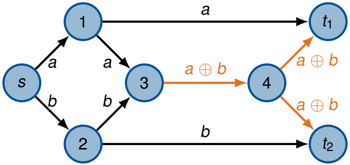
- Finding all multicast trees is a hard problem.
- In practice, heuristics that approximate optimal solutions are being used.

Multicast tree-based forwarding



- Trees in optimal solution
  - (1, 2), (1, 3), (2, 6), (3, 7) $\boldsymbol{x}_1 = [\,1\;1\;0\;1\;0\;1\;0\;0\;0\,]^{\mathrm{T}}$
  - (1, 2), (2, 4), (2, 6), (4, 5), (5, 7) $\boldsymbol{x}_2 = [\,1\;0\;1\;1\;0\;0\;1\;0\;1\,]^{\mathrm{T}}$
  - (1, 3), (3, 4), (3, 7), (4, 5), (5, 6) $\boldsymbol{x}_3 = [\,0\;1\;0\;0\;1\;1\;1\;1\;0\,]^{\mathrm{T}}$
- Each tree carries rate 0.5.
- Total flow which is capacity relevant
  - $0.5(\boldsymbol{x}_1 + \boldsymbol{x}_2 + \boldsymbol{x}_3) = [\,1\;1\;0.5\;1\;0.5\;1\;1\;0.5\;0.5\,]^{\mathrm{T}}$

Maximum Multicast s–T Flow = 1.5



- A single packet (coded unit of information) may serve multiple terminals simultaneously.
- Consider flow to each terminal separately.
- But capacity is shared among all flows, i. e., each flow can use the full capacity on each arc.
- Example $(4, 5)$: flow s–t_1 and s–t_2 transmit unit of information over this arc, but only one coded packet is transmitted.

Multicast with network coding

Network coding: Max s–T flow problem

- One commodity flow $\boldsymbol{x}_t$ for each terminal $t \in T$
- Source vector $\boldsymbol{d}_{st}$ for each terminal $t \in T$
- Capacity vector $\boldsymbol{z}$ is shared for all flows, i. e., capacity on each arc can be fully exploited by each commodity flow.

$$\begin{aligned} \max\ r \quad \text{s. t.} \quad \boldsymbol{M}\boldsymbol{x}_t &= r\boldsymbol{d}_{st} && \forall t \in T \\ \boldsymbol{x}_t &\geq \boldsymbol{0} && \forall t \in T \\ \boldsymbol{x}_t &\leq \boldsymbol{z} && \forall t \in T \end{aligned}$$

Multicast with network coding

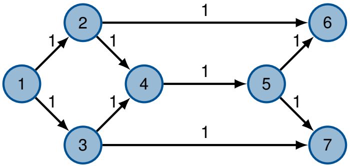
Network coding: Max s–T flow problem

- One commodity flow $\boldsymbol{x}_t$ for each terminal $t \in T$
- Source vector $\boldsymbol{d}_{st}$ for each terminal $t \in T$
- Capacity vector $\boldsymbol{z}$ is shared for all flows, i. e., capacity on each arc can be fully exploited by each commodity flow.

$$\max\ r \quad \text{s. t.} \quad \begin{aligned} \boldsymbol{M}\boldsymbol{x}_t &= r\boldsymbol{d}_{st} && \forall t \in T \\ \boldsymbol{x}_t &\geq \boldsymbol{0} && \forall t \in T \\ \boldsymbol{x}_t &\leq \boldsymbol{z} && \forall t \in T \end{aligned}$$

Note the difference:

- Capacity constraint must be fulfilled for individual flows only.
- There is no joint capacity constraint any more!

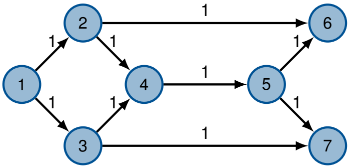


- Optimal flow solutions
  - $\boldsymbol{x}_6 = [1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 1]^T$
  - $\boldsymbol{x}_7 = [1\ 1\ 1\ 0\ 0\ 1\ 1\ 1\ 0]^T$
- Total flow which is capacity relevant
  - $\max(\boldsymbol{x}_6, \boldsymbol{x}_7) = [1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]^T$

Maximum Multicast s–T Flow = 2

Multicast with network coding

Comparison for Butterfly

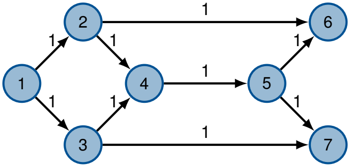


mode	achievable capacity
store-forward	1.0
multicast tree	1.5
network coding	2.0

- Can we do even better?

Multicast with network coding

Comparison for Butterfly

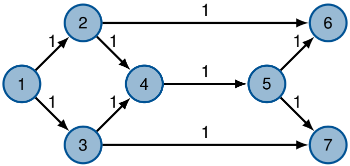


mode	achievable capacity
store-forward	1.0
multicast tree	1.5
network coding	2.0

- Can we do even better? No!

Multicast with network coding

Comparison for Butterfly

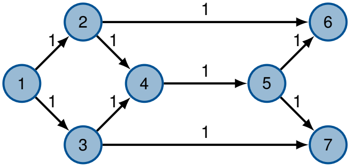


mode	achievable capacity
store-forward	1.0
multicast tree	1.5
network coding	2.0

- Can we do even better? No! Why?

Multicast with network coding

Comparison for Butterfly



mode	achievable capacity
store-forward	1.0
multicast tree	1.5
network coding	2.0

- Can we do even better? No! Why?

Min-cut upper bound on multicast rate

- Find all s–T cuts S and their values $v(S)$.
- The cut with $v(S)$ minimal limits the maximum flow.

Max-flow min-cut theorem (reformulated)

The value of the minimum s–t cut for all terminals $t \in T$ equals the value of the maximum s–T flow with network coding, i. e.,

$$\max\{r : \boldsymbol{M}\boldsymbol{x}_t = r\boldsymbol{d}_{st},\ \boldsymbol{0} \leq \boldsymbol{x}_t \leq \boldsymbol{z},\ \forall t \in T\}$$

$$=$$

$$\min_{t \in T} \min\{v(S) : S \text{ is } s\text{–}t \text{ cut}\}$$

Wired vs. Wireless — Typical Properties

Wired Networks

- Most wired networks are composed from individual point-to-point links, which do not interact and share no resources on the physical layer.
- Physical links are almost lossless and error-free.
- Wired networks can be modeled as abstract graphs with perfect capacitated links for throughput calculation.

Wireless Packet Networks

Wired vs. Wireless — Typical Properties

Wired Networks

- Most wired networks are composed from individual point-to-point links, which do not interact and share no resources on the physical layer.
- Physical links are almost lossless and error-free.
- Wired networks can be modeled as abstract graphs with perfect capacitated links for throughput calculation.

Wireless Networks

- Wireless networks share a common transmission medium.
- The medium is shared and omnidirectional, which turns it into a broadcast medium and causes interference.
- Wireless transmissions are prone to errors leading to packet errors or packet loss.
- How can we model wireless networks? Graphs?

Note: There are wired networks that use broadcast media, such as good old Ethernet without switches. Are there other such networks in use today?

Wireless Packet Networks

Packet Networks

- Information is encoded into packets, which are protected by an
  - error correcting code on the physical layer (channel code) for removing inevitable transmission errors and an
  - error detecting code (e. g. CRC) to detect any residual errors or decoding failures of the channel code, and
- have individual addressing information attached in order to route packets indepedently from source to destination.

Wireless Packet Networks

Packet Networks

- Information is encoded into packets, which are protected by an
  - error correcting code on the physical layer (channel code) for removing inevitable transmission errors and an
  - error detecting code (e. g. CRC) to detect any residual errors or decoding failures of the channel code, and
- have individual addressing information attached in order to route packets indepedently from source to destination.

Note: on point-to-point links there may be no need for addressing information, e. g. Serial Line Internet Protocol (SLIP).

Wireless Packet Networks

- Due the broadcast nature of wireless transmissions, elaborated schemes for medium access are needed:
  - Simultaneous transmissions may cause interference.
  - Without simultaneous transmissions resources may be wasted.
  - Some kind of fairness should be provided.

  ⇒ Medium access needs to be organized (centrally or distributed).

Wireless Packet Networks

- Due the broadcast nature of wireless transmissions, elaborated schemes for medium access are needed:
  - Simultaneous transmissions may cause interference.
  - Without simultaneous transmissions resources may be wasted.
  - Some kind of fairness should be provided.

  $\Rightarrow$ Medium access needs to be organized (centrally or distributed).

- Transmitted packets are randomly lost, i. e., not decodable at the physical layer:
  - Loss may be due to imperfections of wireless communication (channel fading, mobility, etc.).
  - Loss may be also due to interference (packet collisions).

Wireless Packet Networks

- Due the broadcast nature of wireless transmissions, elaborated schemes for medium access are needed:
  - Simultaneous transmissions may cause interference.
  - Without simultaneous transmissions resources may be wasted.
  - Some kind of fairness should be provided.

  $\Rightarrow$ Medium access needs to be organized (centrally or distributed).

- Transmitted packets are randomly lost, i. e., not decodable at the physical layer:
  - Loss may be due to imperfections of wireless communication (channel fading, mobility, etc.).
  - Loss may be also due to interference (packet collisions).

- Transmitted packets are not only received by one (intended) node but by multiple nodes (known as wireless broadcast advantage).
  - Need to model selective overhearing of individual packets. Who gets which packet?

Model 1: simple graph model with orthogonal medium access

- Ignore broadcast advantage, i. e., transmissions are ignored by all but the intended receiver.
- Modify arc capacities to consider
  - medium access and interference, and
  - packet losses.

Model 1: simple graph model with orthogonal medium access

- Ignore broadcast advantage, i. e., transmissions are ignored by all but the intended receiver.
- Modify arc capacities to consider
  - medium access and interference, and
  - packet losses.

Wireless network model

- Graph (N, A)
- Arc capacity vector $\boldsymbol{z}$
- Region of admissible capacity vectors $\mathcal{Z}$:
  - Each $\boldsymbol{z} \in \mathcal{Z}$ corresponds to a different trade-off between all arcs.
  - Trade-off is necessary due to shared resources and interference.

Note: Compare to wired networks, where each arc capacity depends only on the properties of the underlying link.

Model 1: simple graph model with orthogonal medium access

Assumptions

- Same code rate for all packets
- Equal and arbitrarily fine splitting of resources
- No simultaneous transmissions (orthogonal medium access)
- No interference
- Shared transmission time / frequency resources:
  resource share τ_j of arc j s. t. total resource shares add up to 1
- Packet loss (due to fading / noise / mobility / . . .):
  packet loss probability $\varepsilon_j \in [0, 1]$ on arc j ($\varepsilon_j = 0$ for all j means no packet loss)

Model 1: simple graph model with orthogonal medium access

Assumptions

- Same code rate for all packets
- Equal and arbitrarily fine splitting of resources
- No simultaneous transmissions (orthogonal medium access)
- No interference
- Shared transmission time / frequency resources:
  resource share τ_j of arc j s. t. total resource shares add up to 1
- Packet loss (due to fading / noise / mobility / . . .):
  packet loss probability $\varepsilon_j \in [0, 1]$ on arc j ($\varepsilon_j = 0$ for all j means no packet loss)

Arc Capacity Region (NC or ACK/NACK)

$$\mathcal{Z} = \bigcup_{\substack{\boldsymbol{\tau} \geq \mathbf{0}: \\ \mathbf{1}^{\mathsf{T}} \boldsymbol{\tau} \leq 1}} \{\boldsymbol{z} : z_j = \tau_j(1 - \varepsilon_j)\}$$

Model 1: simple graph model with orthogonal medium access

Maximum s-t Flow

- Source vector $\boldsymbol{d}_{st}$
- Incidence matrix $\boldsymbol{M}$
- Arc capacity region $\mathcal{Z}$

$$\max \; r \quad \text{s.t.} \quad \boldsymbol{M}\boldsymbol{x} = r\boldsymbol{d}_{st}$$
$$\boldsymbol{x} \geq \boldsymbol{0}$$
$$\boldsymbol{x} \leq \boldsymbol{z}$$
$$\boldsymbol{z} \in \mathcal{Z}$$

Model 1: simple graph model with orthogonal medium access

Maximum s-t Flow

- Source vector $\boldsymbol{d}_{st}$
- Incidence matrix $\boldsymbol{M}$
- Arc capacity region $\mathcal{Z}$

$$\max\ r \quad \text{s.t.} \quad \begin{aligned} \boldsymbol{Mx} &= r\boldsymbol{d}_{st} \\ \boldsymbol{x} &\geq \boldsymbol{0} \\ \boldsymbol{x} &\leq \boldsymbol{z} \\ \boldsymbol{z} &\in \mathcal{Z} \end{aligned}$$

Maximum s-T multicast flow

- Source vector $\boldsymbol{d}_{st}$ for all $t \in T$
- Incidence matrix $\boldsymbol{M}$
- Arc capacity region $\mathcal{Z}$

$$\max\ r \quad \text{s.t.} \quad \begin{aligned} \boldsymbol{Mx}_t &= r\boldsymbol{d}_{st} && \forall t \in T \\ \boldsymbol{x}_t &\geq \boldsymbol{0} && \forall t \in T \\ \boldsymbol{x}_t &\leq \boldsymbol{z} && \forall t \in T \\ \boldsymbol{z} &\in \mathcal{Z} \end{aligned}$$

Multicast max-flow min-cut theorem (model 1)

The value of the minimum s-T cut for all terminals $t \in T$ equals the value of the maximum s-T flow with network coding, i. e., ,

$$\max\left\{ r : \boldsymbol{M}\boldsymbol{x}_t = r\boldsymbol{d}_{st},\ \boldsymbol{0} \leq \boldsymbol{x}_t \leq \boldsymbol{z},\ \forall t \in T, \boldsymbol{z} \in \mathcal{Z} \right\}$$

$$=$$

$$\max_{\boldsymbol{z}\in\mathcal{Z}} \min_{t \in T} \min \left\{ v(S) = \sum_{j \in A(S)} z_j : S \text{ is } s\text{-}T \text{ cut} \right\}$$

Hypergraphs

Directed hypergraphs

A directed hypergraph $G = (N, H)$ consists of a

- set of nodes $N = \{1, \dots, n\}$ and
- set of hyperarcs $H = \{1, \dots, m\}$ where
  - each hyperarc $j \in H$ represents an ordered pair (a, B) of
  - of a source node $a \in N$ and
  - a subset of nodes $B \subset N$ with $a \notin B$.

We write $j \equiv (a, B) \in \mathcal{H}$, similar to the notation of ordinary arcs and their indices.

Directed hypergraphs

A directed hypergraph $G = (N, H)$ consists of a

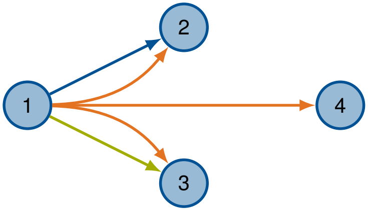
- set of nodes $N = \{1, \dots, n\}$ and
- set of hyperarcs $H = \{1, \dots, m\}$ where
  - each hyperarc $j \in H$ represents an ordered pair (a, B) of
  - of a source node $a \in N$ and
  - a subset of nodes $B \subset N$ with $a \notin B$.

We write $j \equiv (a, B) \in \mathcal{H}$, similar to the notation of ordinary arcs and their indices.

Example:

All hyperarcs $(a, B) \in \mathcal{H}$ with $a = 1$:

- $(1, \{2\})$, $(1, \{3\})$, $(1, \{4\})$
- $(1, \{2, 3\})$, $(1, \{2, 4\})$, $(1, \{3, 4\})$
- $(1, \{2, 3, 4\})$



Directed hypergraphs

A directed hypergraph $G = (N, H)$ consists of a

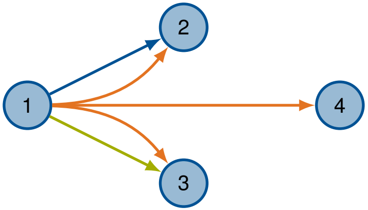
- set of nodes $N = \{1, \dots, n\}$ and
- set of hyperarcs $H = \{1, \dots, m\}$ where
  - each hyperarc $j \in H$ represents an ordered pair (a, B) of
  - of a source node $a \in N$ and
  - a subset of nodes $B \subset N$ with $a \notin B$.

We write $j \equiv (a, B) \in \mathcal{H}$, similar to the notation of ordinary arcs and their indices.

Example:

All hyperarcs $(a, B) \in \mathcal{H}$ with $a = 1$:

- $(1, \{2\})$, $(1, \{3\})$, $(1, \{4\})$
- $(1, \{2, 3\})$, $(1, \{2, 4\})$, $(1, \{3, 4\})$
- $(1, \{2, 3, 4\})$



If a packet is sent over hyperarc $j \equiv (a, B)$, then

- all nodes $b \in B$ overhear an (identical) copy of that packet and
- no other node $i \notin B$ overhears that packet.

Hypergraphs

The directed graph (N, A) induced by hypergraph (N, H) consists of

- all arcs $k \equiv (a, b)$ such that
- there exists $j \equiv (a, B) \in \mathcal{H}$ with $b \in B$.

Hypergraphs

The directed graph (N, A) induced by hypergraph (N, H) consists of

- all arcs $k \equiv (a, b)$ such that
- there exists $j \equiv (a, B) \in \mathcal{H}$ with $b \in B$.

For the induced graph we

- denote the set of arcs that is generated by hyperarc j by A_j and
- define the hyperarc-arc incidence matrix as

$$\boldsymbol{N} = (n)_{jk} = \begin{cases} 1 & \text{if } k \in A_j, \\ 0 & \text{otherwise.} \end{cases}$$

Hypergraphs

The directed graph (N, A) induced by hypergraph (N, H) consists of

- all arcs $k \equiv (a, b)$ such that
- there exists $j \equiv (a, B) \in \mathcal{H}$ with $b \in B$.

For the induced graph we

- denote the set of arcs that is generated by hyperarc j by A_j and
- define the hyperarc-arc incidence matrix as

$$\boldsymbol{N} = (n)_{jk} = \begin{cases} 1 & \text{if } k \in A_j, \\ 0 & \text{otherwise.} \end{cases}$$

Example:

$j \in H$	$(a, B) \in \mathcal{H}$	A_j	$k \in A$
1	$(1, \{2\})$	$\{(1, 2)\}$	1
2	$(1, \{3\})$	$\{(1, 3)\}$	2
3	$(1, \{4\})$	$\{(1, 4)\}$	3
4	$(1, \{2, 3\})$	$\{(1, 2), (1, 3)\}$	1,2
5	$(1, \{2, 4\})$	$\{(1, 2), (1, 4)\}$	1,3
6	$(1, \{3, 4\})$	$\{(1, 3), (1, 4)\}$	2,3
7	$(1, \{2, 3, 4\})$	$\{(1, 2), (1, 3), (1, 4)\}$	1,2,3

Model 2: lossless hypergraph model with orthogonal medium access

- Hypergraph (N, H)
- One hyperarc per node (simplification), enumerated according to the node the hyperarc is originating at
- Inherits MAC properties from model 1 (orthogonal medium access)
- Each node gets a resource share $\tau_i \geq 0$ such that $\sum_{i \in N} \tau_i \leq 1$
- Packets transmitted on a hyperarc $j \equiv (a, B)$ are received by all nodes $b \in B$
- No packets are lost

Model 2: lossless hypergraph model with orthogonal medium access

Information flow in lossless hypergraphs (model 2)

- Information flow vector $\boldsymbol{x}$ on induced graph (N, A)
- Demand vector $\boldsymbol{d}$ and incidence matrix $\boldsymbol{M}$
- Flow must be conserved on induced graph, i. e.,

$$\boldsymbol{M}\boldsymbol{x} = \boldsymbol{d}$$

Model 2: lossless hypergraph model with orthogonal medium access

Information flow in lossless hypergraphs (model 2)

- Information flow vector $\boldsymbol{x}$ on induced graph (N, A)
- Demand vector $\boldsymbol{d}$ and incidence matrix $\boldsymbol{M}$
- Flow must be conserved on induced graph, i. e.,

$$\boldsymbol{Mx} = \boldsymbol{d}$$

- Hyperarc-arc incidence matrix $\boldsymbol{N}$
- Receivers of a hyperarc get identical packets over this hyperarc
- Each piece of information can only be used once (by one node)

$$\boldsymbol{Nx} \leq \boldsymbol{z} \quad \Leftrightarrow \quad \sum_{k \in A_j} x_k \leq z_j \quad \forall j \in H$$

Model 2: lossless hypergraph model with orthogonal medium access

Information flow in lossless hypergraphs (model 2)

- Information flow vector $\boldsymbol{x}$ on induced graph (N, A)
- Demand vector $\boldsymbol{d}$ and incidence matrix $\boldsymbol{M}$
- Flow must be conserved on induced graph, i. e.,

$$\boldsymbol{M}\boldsymbol{x} = \boldsymbol{d}$$

- Hyperarc-arc incidence matrix $\boldsymbol{N}$
- Receivers of a hyperarc get identical packets over this hyperarc
- Each piece of information can only be used once (by one node)

$$\boldsymbol{N}\boldsymbol{x} \leq \boldsymbol{z} \qquad \Leftrightarrow \qquad \sum_{k \in A_j} x_k \leq z_j \quad \forall j \in H$$

- Lossless Hyperarc Capacity Region (NC)

$$\mathcal{Z} = \bigcup_{\substack{\boldsymbol{\tau} \geq \boldsymbol{0}: \\ \boldsymbol{1}^\mathrm{T} \boldsymbol{\tau} \leq 1}} \{\boldsymbol{z} : z_j = \tau_j\}$$

Model 2: lossless hypergraph model with orthogonal medium access

Hyperarc maximum s-t flow (routing/network coding)

- Source vector $\boldsymbol{d}_{st}$
- Incidence matrix $\boldsymbol{M}$
- Hyperarc-arc incidence matrix $\boldsymbol{N}$
- Hyperarc capacity region $\mathscr{Z}$

$$\begin{aligned} \max\ r \quad \text{s.t.} \quad \boldsymbol{M}\boldsymbol{x} &= r\boldsymbol{d}_{st} \\ \boldsymbol{x} &\geq \boldsymbol{0} \\ \boldsymbol{N}\boldsymbol{x} &\leq \boldsymbol{z} \\ \boldsymbol{z} &\in \mathscr{Z} \end{aligned}$$

Model 2: lossless hypergraph model with orthogonal medium access

Hyperarc maximum *s*-*t* flow (routing/network coding)

- Source vector $\boldsymbol{d}_{st}$
- Incidence matrix $\boldsymbol{M}$
- Hyperarc-arc incidence matrix $\boldsymbol{N}$
- Hyperarc capacity region $\mathcal{Z}$

$$\begin{aligned} \max\ r \quad \text{s.t.} \quad \boldsymbol{Mx} &= r\boldsymbol{d}_{st} \\ \boldsymbol{x} &\geq \boldsymbol{0} \\ \boldsymbol{Nx} &\leq \boldsymbol{z} \\ \boldsymbol{z} &\in \mathcal{Z} \end{aligned}$$

Hyperarc maximum *s*-*T* multicast flow (network coding)

- Source vector $\boldsymbol{d}_{st}$
- Incidence matrix $\boldsymbol{M}$
- Hyperarc-arc incidence matrix $\boldsymbol{N}$
- Hyperarc capacity region $\mathcal{Z}$

$$\begin{aligned} \max\ r \quad \text{s.t.} \quad \boldsymbol{Mx}_t &= r\boldsymbol{d}_{st} & \forall t \in T \\ \boldsymbol{x}_t &\geq \boldsymbol{0} & \forall t \in T \\ \boldsymbol{Nx}_t &\leq \boldsymbol{z} & \forall t \in T \\ \boldsymbol{z} &\in \mathcal{Z} \end{aligned}$$

Note

- We can use each hyperarc (packet) only once for each terminal.
- But we can use each hyperarc differently for each terminal.

Model 2: lossless hypergraph model with orthogonal medium access

Hyperarc min-cut model

- An s-t cut is a subset of nodes $S \subset N$ such that $s \in S$ and $t \notin S$.
- A hyperarc $j \equiv (a, B) \in \mathcal{H}$ crosses S if $a \in S$ and $B \not\subset S$.
  $\mathcal{H}(S)$ denotes all crossing arcs, and $H(S)$ their indices.
- The value of any s-t cut upper bounds the maximum s-t flow.
- The value of an s-t cut given the capacity vector $\boldsymbol{z}$ is defined as

$$v(S) = \sum_{j \in H(S)} z_j = \sum_{(a,B) \in \mathcal{H}(S)} z_{aB}$$

- Model 2 (only one hyperarc per node, $z_{aB} = \tau_a$):

$$v(S) = \sum_{(a,B) \in \mathcal{H}(S)} \tau_a .$$

Model 2: lossless hypergraph model with orthogonal medium access

Multicast max-flow min-cut theorem (model 2)

The value of the minimum s-T cut for all terminals $t \in T$ equals the value of the maximum s-T flow with network coding, i. e.,

$$\max\left\{ r : \boldsymbol{M}\boldsymbol{x}_t = r\boldsymbol{d}_{st},\ \boldsymbol{0} \leq \boldsymbol{x}_t,\ \boldsymbol{N}\boldsymbol{x}_t \leq \boldsymbol{z},\ \forall t \in T, \boldsymbol{z} \in \mathcal{Z} \right\}$$

$$=$$

$$\max_{\boldsymbol{z}\in\mathcal{Z}} \min_{t\in T} \min \left\{ v(S) = \sum_{j\in H(S)} z_j : S \text{ is } s\text{-}T \text{ cut} \right\}$$

Model 3: lossy hypergraph model with orthogonal medium access

Idea: Use model 2 but

- account for packet erasures and
- allow different erasure probabilities to each neighbor.

$\Rightarrow$ one hyperarc per node is insufficient

Model 3: lossy hypergraph model with orthogonal medium access

Idea: Use model 2 but

- account for packet erasures and
- allow different erasure probabilities to each neighbor.

$\Rightarrow$ one hyperarc per node is insufficient

Changes made to model 2:

- Hypergraph (N, H) with induced graph (N, A).
- Let N_a denote the set of neighbors of node $a \in N$.
- For each node $a \in N$ consider all possible hyperarcs $j \equiv (a, B)$ for any $B \subset N_a$.
- Packet loss is independent across all receivers (simplification).
- Packets from a to b are lost with probability ε_k where $k \equiv (a, b)$, i. e., $k \in A$ is an arc index of the induced graph.
- A packet transmitted by $a \in N$ is transmitted on hyperarc $j \equiv (a, B)$, i. e., it is received precisely by $B \subset N_a$ and lost by all other nodes $N_a \setminus B$, with probability

$$\Pr[\text{“no loss on } j \equiv (a, B)\text{”} \mid \text{“}a \text{ transmits”}] = \prod_{b \in B} (1 - \varepsilon_{ab}) \prod_{b \notin B} \varepsilon_{ab}.$$

Model 3: lossy hypergraph model with orthogonal medium access

Information flow in lossy hypergraphs (model 3)

- Information flow vector $\boldsymbol{x}$ on induced graph (N, A).
- Flow must be conserved on induced graph, i. e., $\boldsymbol{Mx} = \boldsymbol{d}$.
- Receivers of a hyperarc get identical packets over this hyperarc provided they have not lost the packets.

Given a transmitter a, we are interested in an upper bound for the flow from a to a set of receivers $B \subset N_a$ (where N_a denotes the neighborhood of node a).

- Each piece of information can only be used by one successful receiver.
- The total flow from a to B must not exceed the total amount of different received packets of this set of nodes.

This is equivalent to the probability that

- node a is transmitting at all and
- at least one node $b \in B$ overhears the transmission.

It does not matter whether or not more than one or which specific node in B overhears the transmission.

Model 3: lossy hypergraph model with orthogonal medium access

For a 1-receiver set $B = \{b\}$ and induced arc $k \equiv (a, b)$:

- Which hyperarcs may transport packets from a to b?

$\Rightarrow$ Flow bound:

$$x_k = x_{ab} \leq$$

Model 3: lossy hypergraph model with orthogonal medium access

For a 1-receiver set $B = \{b\}$ and induced arc $k \equiv (a, b)$:

- Which hyperarcs may transport packets from a to b?

  Any hyperarc $j' \equiv (a, B')$ with $b \in B'$, and precisely these hyperarcs induce arc $k \equiv (a, b)$.

  $\Rightarrow$ Consider all $(a, B') \equiv j' \in H : (a, b) \equiv k \in A_{j'}$.

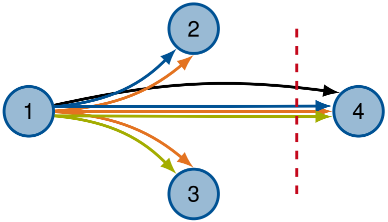


Figure 1: Example for $a = 1$ and $b = 4$, only hyperarcs $j' \in H : k \in A_{j'}$ are shown

$\Rightarrow$ Flow bound:

$$x_k = x_{ab} \leq \sum_{\substack{(a,B')\equiv j':\\ (a,b)\in A_{j'}}}$$

For a 1-receiver set $B = \{b\}$ and induced arc $k \equiv (a, b)$:

- Which hyperarcs may transport packets from a to b?

  Any hyperarc $j' \equiv (a, B')$ with $b \in B'$, and precisely these hyperarcs induce arc $k \equiv (a, b)$.

  $\Rightarrow$ Consider all $(a, B') \equiv j' \in H : (a, b) \equiv k \in A_{j'}$.

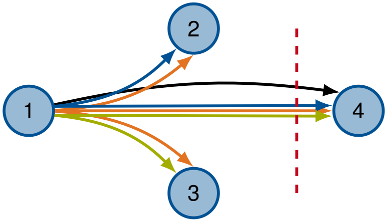


Figure 1: Example for $a = 1$ and $b = 4$, only hyperarcs $j' \in H : k \in A_{j'}$ are shown

- Packets may be transferred over any of these hyperarcs $(a, B') \equiv j'$, but only if node a is transmitting at all.

$\Rightarrow$ Flow bound:

$$x_k = x_{ab} \leq \sum_{\substack{(a,B')\equiv j' : \\ (a,b)\in A_{j'}}} \tau_a$$

Model 3: lossy hypergraph model with orthogonal medium access

For a 1-receiver set $B = \{b\}$ and induced arc $k \equiv (a, b)$:

- Which hyperarcs may transport packets from a to b?

  Any hyperarc $j' \equiv (a, B')$ with $b \in B'$, and precisely these hyperarcs induce arc $k \equiv (a, b)$.

  $\Rightarrow$ Consider all $(a, B') \equiv j' \in H : (a, b) \equiv k \in A_{j'}$.

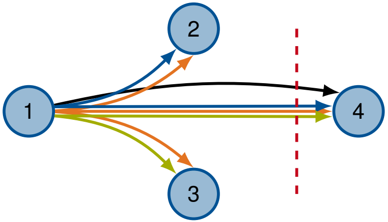


Figure 1: Example for $a = 1$ and $b = 4$, only hyperarcs $j' \in H : k \in A_{j'}$ are shown

- Packets may be transferred over any of these hyperarcs $(a, B') \equiv j'$, but only if node a is transmitting at all.
- What is the probability that a packet transmitted over $j' \equiv (a, B')$ is successfully received by precisely $B' \subset N$?

$\Rightarrow$ Flow bound:

$$x_k = x_{ab} \leq \sum_{\substack{(a,B')\equiv j': \\ (a,b)\in A_{j'}}} \tau_a$$

Model 3: lossy hypergraph model with orthogonal medium access

For a 1-receiver set $B = \{b\}$ and induced arc $k \equiv (a, b)$:

- Which hyperarcs may transport packets from a to b?

  Any hyperarc $j' \equiv (a, B')$ with $b \in B'$, and precisely these hyperarcs induce arc $k \equiv (a, b)$.

  $\Rightarrow$ Consider all $(a, B') \equiv j' \in H : (a, b) \equiv k \in A_{j'}$.

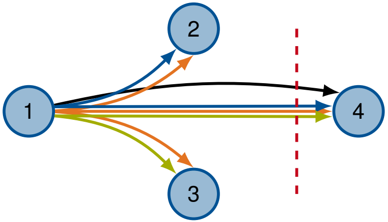


Figure 1: Example for $a = 1$ and $b = 4$, only hyperarcs $j' \in H : k \in A_{j'}$ are shown

- Packets may be transferred over any of these hyperarcs $(a, B') \equiv j'$, but only if node a is transmitting at all.

- What is the probability that a packet transmitted over $j' \equiv (a, B')$ is successfully received by precisely $B' \subset N$?
  - Since all $b' \in B'$ must receive the packet, none of the arcs $(a, b') \in A_{j'}$ induced by hyperarc $j' \equiv (a, B')$ must fail, and
  - no other induced arc originating at a must succeed.

$\Rightarrow$ Flow bound:

$$x_k = x_{ab} \leq \sum_{\substack{(a,B') \equiv j' : \\ (a,b) \in A_{j'}}} \tau_a \prod_{b' \in B'} (1 - \varepsilon_{ab'}) \prod_{b' \notin B'} \varepsilon_{ab'} = y_{aB} = y_j$$

Model 3: lossy hypergraph model with orthogonal medium access

For a 2-receiver set $B = \{b_1, b_2\}$ and induced arcs $k_1 \equiv (a, b_1)$ and $k_2 \equiv (a, b_2)$:

- Which $j' \in H$ may transport packets to either b_1 or b_2?

$\Rightarrow$ Flow bound:

$$x_{k_1} + x_{k_2} = x_{ab_1} + x_{ab_2} \leq$$

Model 3: lossy hypergraph model with orthogonal medium access

For a 2-receiver set $B = \{b_1, b_2\}$ and induced arcs $k_1 \equiv (a, b_1)$ and $k_2 \equiv (a, b_2)$:

- Which $j' \in H$ may transport packets to either b_1 or b_2?

  Any hyperarc $j' \equiv (a, B')$ with $B' \cap B \neq \emptyset$, and precisely these hyperarcs induce either k_1, k_2, or both.

  $\Rightarrow$ Consider all $(a, B') \equiv j' \in H : \{k_1, k_2\} \cap A_{j'} \neq \emptyset$.

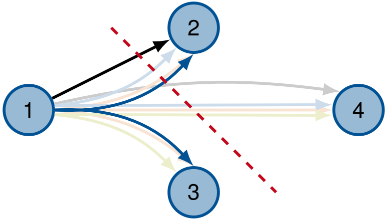


Figure 2: Example for $a = 1$ and $B = \{2, 4\}$; shaded hyperarcs are for $B' = \{4\}$, solid hyperarcs are the additions for $B = B' \cup \{2\}$

$\Rightarrow$ Flow bound:

$$x_{k_1} + x_{k_2} = x_{ab_1} + x_{ab_2} \leq \sum_{\substack{(a,B')\equiv j': \\ \{k_1,k_2\}\cap A_{j'}\neq\emptyset}}$$

Model 3: lossy hypergraph model with orthogonal medium access

For a 2-receiver set $B = \{b_1, b_2\}$ and induced arcs $k_1 \equiv (a, b_1)$ and $k_2 \equiv (a, b_2)$:

- Which $j' \in H$ may transport packets to either b_1 or b_2?

  Any hyperarc $j' \equiv (a, B')$ with $B' \cap B \neq \emptyset$, and precisely these hyperarcs induce either k_1, k_2, or both.

  $\Rightarrow$ Consider all $(a, B') \equiv j' \in H : \{k_1, k_2\} \cap A_{j'} \neq \emptyset$.

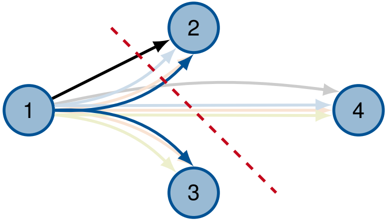


Figure 2: Example for $a = 1$ and $B = \{2, 4\}$; shaded hyperarcs are for $B' = \{4\}$, solid hyperarcs are the additions for $B = B' \cup \{2\}$

- Packets may be transferred over any of these hyperarcs $(a, B') \equiv j'$, but only if a is transmitting at all.

$\Rightarrow$ Flow bound:

$$x_{k_1} + x_{k_2} = x_{ab_1} + x_{ab_2} \leq \sum_{\substack{(a,B') \equiv j' : \\ \{k_1, k_2\} \cap A_{j'} \neq \emptyset}} \tau_a$$

Model 3: lossy hypergraph model with orthogonal medium access

For a 2-receiver set $B = \{b_1, b_2\}$ and induced arcs $k_1 \equiv (a, b_1)$ and $k_2 \equiv (a, b_2)$:

- Which $j' \in H$ may transport packets to either b_1 or b_2?

  Any hyperarc $j' \equiv (a, B')$ with $B' \cap B \neq \emptyset$, and precisely these hyperarcs induce either k_1, k_2, or both.

  $\Rightarrow$ Consider all $(a, B') \equiv j' \in H : \{k_1, k_2\} \cap A_{j'} \neq \emptyset$.

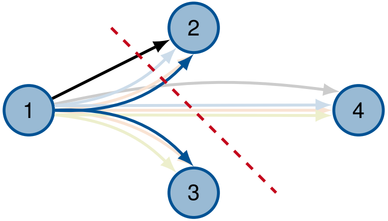


Figure 2: Example for $a = 1$ and $B = \{2, 4\}$; shaded hyperarcs are for $B' = \{4\}$, solid hyperarcs are the additions for $B = B' \cup \{2\}$

- Packets may be transferred over any of these hyperarcs $(a, B') \equiv j'$, but only if a is transmitting at all.

- What is the probability that a packet transmitted over $j' \equiv (a, B')$ is successfully received by precisely $B' \subset N$?

$\Rightarrow$ Flow bound:

$$x_{k_1} + x_{k_2} = x_{ab_1} + x_{ab_2} \leq \sum_{\substack{(a,B') \equiv j' : \\ \{k_1, k_2\} \cap A_{j'} \neq \emptyset}} \tau_a$$

Model 3: lossy hypergraph model with orthogonal medium access

For a 2-receiver set $B = \{b_1, b_2\}$ and induced arcs $k_1 \equiv (a, b_1)$ and $k_2 \equiv (a, b_2)$:

- Which $j' \in H$ may transport packets to either b_1 or b_2?

  Any hyperarc $j' \equiv (a, B')$ with $B' \cap B \neq \emptyset$, and precisely these hyperarcs induce either k_1, k_2, or both.

  $\Rightarrow$ Consider all $(a, B') \equiv j' \in H : \{k_1, k_2\} \cap A_{j'} \neq \emptyset$.

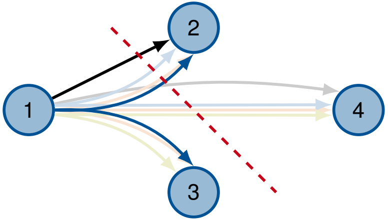


Figure 2: Example for $a = 1$ and $B = \{2, 4\}$; shaded hyperarcs are for $B' = \{4\}$, solid hyperarcs are the additions for $B = B' \cup \{2\}$

- Packets may be transferred over any of these hyperarcs $(a, B') \equiv j'$, but only if a is transmitting at all.
- What is the probability that a packet transmitted over $j' \equiv (a, B')$ is successfully received by precisely $B' \subset N$?
  - Since all $b' \in B'$ must receive the packet, none of the arcs $(a, b') \in A_{j'}$ induced by hyperarc $j' \equiv (a, B')$ must fail, and
  - no other induced arc originating at a must succeed.

$\Rightarrow$ Flow bound:

$$x_{k_1} + x_{k_2} = x_{ab_1} + x_{ab_2} \leq \sum_{\substack{(a,B') \equiv j' : \\ \{k_1, k_2\} \cap A_{j'} \neq \emptyset}} \tau_a \prod_{b' \in B'} (1 - \varepsilon_{ab'}) \prod_{b' \notin B'} \varepsilon_{ab'} = y_{aB} = y_j$$

Model 3: lossy hypergraph model with orthogonal medium access

Generalization to multiple receiver sets:

- Each pair (a, B) corresponds to some hyperarc j, i. e., $j \equiv (a, B)$.
- That hyperarc induces the set A_j of arcs.
- The flow bound is determined by all hyperarcs $j' \in H : A_j \cap A_{j'} \neq \emptyset$:

$$\sum_{k \in A_j} x_k = \sum_{(a,b) \in \mathcal{A}_j} x_{ab} \leq \sum_{\substack{(a,B') \equiv j': \\ A_j \cap A_{j'} \neq \emptyset}} \tau_a \prod_{b' \in B'} (1 - \varepsilon_{ab'}) \prod_{b' \notin B'} \varepsilon_{ab'} = y_{aB} = y_j$$

Model 3: lossy hypergraph model with orthogonal medium access

Generalization to multiple receiver sets:

- Each pair (a, B) corresponds to some hyperarc j, i. e., $j \equiv (a, B)$.
- That hyperarc induces the set A_j of arcs.
- The flow bound is determined by all hyperarcs $j' \in H : A_j \cap A_{j'} \neq \emptyset$:

$$\sum_{k \in A_j} x_k = \sum_{(a,b) \in \mathcal{A}_j} x_{ab} \leq \sum_{\substack{(a,B') \equiv j': \\ A_j \cap A_{j'} \neq \emptyset}} \tau_a \prod_{b' \in B'} (1 - \varepsilon_{ab'}) \prod_{b' \notin B'} \varepsilon_{ab'} = y_{aB} = y_j$$

- Hyperarc capacity:

$$z_j = z_{aB} = \tau_a \prod_{b \in B} (1 - \varepsilon_{ab}) \prod_{b \notin B} \varepsilon_{ab}$$

Model 3: lossy hypergraph model with orthogonal medium access

Generalization to multiple receiver sets:

- Each pair (a, B) corresponds to some hyperarc j, i. e., $j \equiv (a, B)$.
- That hyperarc induces the set A_j of arcs.
- The flow bound is determined by all hyperarcs $j' \in H : A_j \cap A_{j'} \neq \emptyset$:

$$\sum_{k \in A_j} x_k = \sum_{(a,b) \in \mathcal{A}_j} x_{ab} \leq \sum_{\substack{(a,B') \equiv j': \\ A_j \cap A_{j'} \neq \emptyset}} \tau_a \prod_{b' \in B'} (1 - \varepsilon_{ab'}) \prod_{b' \notin B'} \varepsilon_{ab'} = y_{aB} = y_j$$

- Hyperarc capacity:

$$z_j = z_{aB} = \tau_a \prod_{b \in B} (1 - \varepsilon_{ab}) \prod_{b \notin B} \varepsilon_{ab}$$

- Hyperarc capacity region:

$$\mathcal{Z} = \bigcup_{\substack{\boldsymbol{\tau} \geq \mathbf{0} \\ \mathbf{1}^{\mathrm{T}} \boldsymbol{\tau} \leq 1}} \left\{ \mathbf{z} : z_j = z_{aB} = \tau_a \prod_{b \in B} (1 - \varepsilon_{ab}) \prod_{b \notin B} \varepsilon_{ab} \quad \forall j \equiv (a, B) \in \mathcal{H} \right\}$$

Model 3: lossy hypergraph model with orthogonal medium access

Information flow in lossy hypergraphs (model 3)

- Reformulation of the lossy flow bound (for all receiver set):

$$\sum_{k \in A_j} x_k \leq \tau_a \left(1 - \prod_{k \in A_j} \varepsilon_k \right) = y_j \quad \forall j \equiv (a, B) \in \mathcal{H}$$

Model 3: lossy hypergraph model with orthogonal medium access

Information flow in lossy hypergraphs (model 3)

- Reformulation of the lossy flow bound (for all receiver set):

$$\sum_{k \in A_j} x_k \leq \tau_a \left(1 - \prod_{k \in A_j} \varepsilon_k\right) = y_j \quad \forall j \equiv (a, B) \in \mathcal{H}$$

- Broadcast capacity region

$$\mathcal{Y} = \bigcup_{\substack{\boldsymbol{\tau} \geq \mathbf{0} \\ \mathbf{1}^\mathrm{T} \boldsymbol{\tau} \leq 1}} \left\{ \boldsymbol{y} : y_j = \tau_a \left(1 - \prod_{k \in A_j} \varepsilon_k\right) \quad \forall j \equiv (a, B) \in \mathcal{H} \right\}$$

Model 3: lossy hypergraph model with orthogonal medium access

Information flow in lossy hypergraphs (model 3)

- Hyperarc-arc incidence matrix

$$\boldsymbol{N} = (N_{jk}) = \begin{cases} 1 & \text{if } k \in A_j \\ 0 & \text{otherwise} \end{cases}$$

- Hyperarc-hyperarc incidence matrix

$$\boldsymbol{Q} = (Q_{ij}) = \begin{cases} 1 & \text{if } A_i \cap A_j \neq \emptyset \\ 0 & \text{otherwise} \end{cases}$$

- Hyperarc-to-broadcast transformation

$$\boldsymbol{y} = \boldsymbol{Q}\boldsymbol{z}$$

- Lossy hyperarc flow bound with hyperarc capacity region

$$\boldsymbol{N}\boldsymbol{x} \leq \boldsymbol{Q}\boldsymbol{z}$$

- Lossy hyperarc flow bound with broadcast capacity region

$$\boldsymbol{N}\boldsymbol{x} \leq \boldsymbol{y}$$

Model 3: lossy hypergraph model with orthogonal medium access

Lossy hyperarc maximum s-t flow (opportunistic RT/NC)

- Source vector $\boldsymbol{d}_{st}$
- Incidence matrix $\boldsymbol{M}$
- Hyperarc-arc incidence matrix $\boldsymbol{N}$
- Broadcast capacity region $\mathcal{Y}$

$$
\begin{aligned}
\max \quad r \quad \text{s.t.} \quad & \boldsymbol{M}\boldsymbol{x} = r\boldsymbol{d}_{st} \\
& \boldsymbol{x} \geq \boldsymbol{0} \\
& \boldsymbol{N}\boldsymbol{x} \leq \boldsymbol{y} \\
& \boldsymbol{y} \in \mathcal{Y}
\end{aligned}
$$

Model 3: lossy hypergraph model with orthogonal medium access

Lossy hyperarc maximum s-t flow (opportunistic RT/NC)

- Source vector $\boldsymbol{d}_{st}$
- Incidence matrix $\boldsymbol{M}$
- Hyperarc-arc incidence matrix $\boldsymbol{N}$
- Broadcast capacity region $\mathcal{Y}$

$$\max \; r \quad \text{s.t.} \quad \boldsymbol{M}\boldsymbol{x} = r\boldsymbol{d}_{st}, \quad \boldsymbol{x} \geq \boldsymbol{0}, \quad \boldsymbol{N}\boldsymbol{x} \leq \boldsymbol{y}, \quad \boldsymbol{y} \in \mathcal{Y}$$

Lossy hyperarc maximum s-T multicast flow (NC)

- Source vector $\boldsymbol{d}_{st}$
- Incidence matrix $\boldsymbol{M}$
- Hyperarc-arc incidence matrix $\boldsymbol{N}$
- Broadcast capacity region $\mathcal{Y}$

$$\begin{aligned} \max \; r \quad \text{s.t.} \quad \boldsymbol{M}\boldsymbol{x}_t &= r\boldsymbol{d}_{st} && \forall t \in T \\ \boldsymbol{x}_t &\geq \boldsymbol{0} && \forall t \in T \\ \boldsymbol{N}\boldsymbol{x}_t &\leq \boldsymbol{y} && \forall t \in T \\ \boldsymbol{y} &\in \mathcal{Y} \end{aligned}$$

Model 3: lossy hypergraph model with orthogonal medium access

Lossy hyperarc min-cut model

- An s-t cut is a subset of nodes $S \subset N$ such that $s \in S$ and $t \notin S$.
- A hyperarc $j \equiv (a, B) \in \mathcal{H}$ crosses S if $a \in S$ and $B \not\subset S$, i. e., $B \cap (N \setminus S) \neq \emptyset$.
- $H(S)$ denotes all crossing hyperarc indices, $\mathcal{H}(S)$ all crossing hyperarcs.
- The value of any s-t cut upper bounds the maximum s-t flow.
- The value of an s-t cut given the capacity vector $\boldsymbol{z}$ is defined as

$$v(S) = \sum_{j \in H(S)} z_j.$$

- Cut value of model 3:

$$v(S) = \sum_{(a,B) \in \mathcal{H}(S)} \tau_a \prod_{b \in B} (1 - \varepsilon_{ab}) \prod_{b \notin B} \varepsilon_{ab}.$$

- $\mathcal{A}_a(S)$: Set of arcs $(a, b) \in \mathcal{A} : b \in N \setminus S$
  ($A_a(S)$ denotes index set of crossing arcs)
- $\mathcal{H}_a(S)$: Set of hyperarcs $(a, B) \in \mathcal{H} : B \cap (N \setminus S) \neq \emptyset$
  ($H_a(S)$ denotes the index set of crossing hyperarcs)

Model 3: lossy hypergraph model with orthogonal medium access

Lossy hyperarc min-cut model

- Characterize $\mathcal{H}(S)$:

$$\begin{aligned}\mathcal{H}(S) &= \{(a,B) \in \mathcal{H} : a \in S, B \cap (N \setminus S) \neq \emptyset\} \\ &= \bigcup_{a \in S} \mathcal{H}_a(S) \\ &= \bigcup_{a \in S} \{(a,B) \equiv j \in H : A_j \cap A_a(S) \neq \emptyset\}\end{aligned}$$

- Cut value of Model 3 (looks very much like flow bound):

$$v(S) = \sum_{a \in S} \sum_{\substack{j \equiv (a,B): \\ A_j \cap A_a(S) \neq \emptyset}} \tau_a \prod_{b \in B} (1 - \varepsilon_{ab}) \prod_{b \notin B} \varepsilon_{ab}$$

Model 3: lossy hypergraph model with orthogonal medium access

Lossy hyperarc min-cut model

- Flow bound:

$$\sum_{k \in A_j} x_k \leq \sum_{\substack{j' \equiv (a,B'): \\ A_j \cap A_{j'} \neq \emptyset}} \tau_a \prod_{b' \in B'} (1 - \varepsilon_{ab'}) \prod_{b' \notin B'} \varepsilon_{ab'} = y_j \quad \forall j \equiv (a, B) \in \mathcal{H}$$

- Cut value of model 3:

$$v(S) = \sum_{a \in S} \sum_{\substack{j \equiv (a,B): \\ A_j \cap A_a(S) \neq \emptyset}} \tau_a \prod_{b \in B} (1 - \varepsilon_{ab}) \prod_{b \notin B} \varepsilon_{ab}$$

Lossy hyperarc min-cut model

- Flow bound:

$$\sum_{k \in A_j} x_k \leq \sum_{\substack{j' \equiv (a,B'): \\ A_j \cap A_{j'} \neq \emptyset}} \tau_a \prod_{b' \in B'} (1 - \varepsilon_{ab'}) \prod_{b' \notin B'} \varepsilon_{ab'} = y_j \quad \forall j \equiv (a, B) \in \mathcal{H}$$

$$\sum_{k \in A_j} x_k \leq \tau_a \left(1 - \prod_{k \in A_j} \varepsilon_k\right) = y_j \quad \forall j \equiv (a, B) \in \mathcal{H}$$

- Cut value of model 3:

$$v(S) = \sum_{a \in S} \sum_{\substack{j \equiv (a,B): \\ A_j \cap A_a(S) \neq \emptyset}} \tau_a \prod_{b \in B} (1 - \varepsilon_{ab}) \prod_{b \notin B} \varepsilon_{ab}$$

Model 3: lossy hypergraph model with orthogonal medium access

Lossy hyperarc min-cut model

- Flow bound:

$$\sum_{k \in A_j} x_k \leq \sum_{\substack{j' \equiv (a,B'): \\ A_j \cap A_{j'} \neq \emptyset}} \tau_a \prod_{b' \in B'} (1 - \varepsilon_{ab'}) \prod_{b' \notin B'} \varepsilon_{ab'} = y_j \quad \forall j \equiv (a, B) \in \mathcal{H}$$

$$\sum_{k \in A_j} x_k \leq \tau_a \left(1 - \prod_{k \in A_j} \varepsilon_k \right) = y_j \quad \forall j \equiv (a, B) \in \mathcal{H}$$

- Cut value of model 3:

$$v(S) = \sum_{a \in S} \sum_{\substack{j \equiv (a,B): \\ A_j \cap A_a(S) \neq \emptyset}} \tau_a \prod_{b \in B} (1 - \varepsilon_{ab}) \prod_{b \notin B} \varepsilon_{ab}$$

$$v(S) = \sum_{a \in S} \tau_a \left(1 - \prod_{k \in A_a(S)} \varepsilon_k \right) = \sum_{\substack{j \equiv (a,B) \in \mathcal{H}: \\ a \in S \wedge B = N_a \setminus S}} y_j$$

Multicast max-flow min-cut theorem (model 3)

The value of the minimum s-t cut for all terminals $t \in T$ equals the value of the maximum s-T flow with network coding, i. e.,

$$\max\left\{r : \boldsymbol{M}\boldsymbol{x}_t = r\boldsymbol{d}_{st},\ \boldsymbol{0} \leq \boldsymbol{x}_t,\ \boldsymbol{N}\boldsymbol{x}_t \leq \boldsymbol{y},\ \forall t \in T, \boldsymbol{y} \in \mathcal{Y}\right\}$$

$$=$$

$$\max_{\boldsymbol{y}\in\mathcal{Y}} \min_{t\in T} \min\left\{v(S) = \sum_{\substack{j\equiv(a,B)\in\mathcal{H}: \\ a\in S\wedge B=N_a\setminus S}} y_j : S \text{ is } s\text{-}t \text{ cut}\right\}$$

Model overview

Section 1: Model 1 – lossy non-hypergraph model

- Reflects lossy wired networks
- Cannot intuitively cope with broadcast media
- Flow bound: $\boldsymbol{x} \leq \boldsymbol{z} \quad \mathcal{Z} = \bigcup_{\boldsymbol{\tau} \geq \mathbf{0} \wedge \mathbf{1}^\mathsf{T} \boldsymbol{\tau} \leq 1} \{\boldsymbol{z} : z_k = \tau_k(1 - \varepsilon_k) \quad \forall k \in A\}$

Note: Model 1 $\subset$ Model 2 $\subset$ Model 3

Model overview

Section 1: Model 1 – lossy non-hypergraph model

- Reflects lossy wired networks
- Cannot intuitively cope with broadcast media
- Flow bound: $\boldsymbol{x} \le \boldsymbol{z} \quad \mathcal{Z} = \bigcup_{\boldsymbol{\tau} \ge \boldsymbol{0} \wedge \boldsymbol{1}^\mathsf{T}\boldsymbol{\tau} \le 1} \{\boldsymbol{z} : z_k = \tau_k(1-\varepsilon_k) \quad \forall k \in A\}$

Section 3: Model 2 – lossless hypergraph

- Considers broadcast media by hyperarcs
- No losses, i. e., single hyperarc from some $a \in N$ to all its neighbors $b \in N_a$
- Flow bound: $\boldsymbol{N}\boldsymbol{x} \le \boldsymbol{z} \quad \mathcal{Z} = \bigcup_{\boldsymbol{\tau} \ge \boldsymbol{0} \wedge \boldsymbol{1}^\mathsf{T}\boldsymbol{\tau} \le 1} \{\boldsymbol{z} : z_j = \tau_j \quad \forall j \in H\}$

Note: Model 1 $\subset$ Model 2 $\subset$ Model 3

Model overview

Section 1: Model 1 – lossy non-hypergraph model

- Reflects lossy wired networks
- Cannot intuitively cope with broadcast media
- Flow bound: $\boldsymbol{x} \leq \boldsymbol{z} \quad \mathcal{Z} = \bigcup_{\boldsymbol{\tau} \geq \boldsymbol{0} \wedge \boldsymbol{1}^{\mathrm{T}} \boldsymbol{\tau} \leq 1} \{\boldsymbol{z} : z_k = \tau_k(1 - \varepsilon_k) \quad \forall k \in A\}$

Section 3: Model 2 – lossless hypergraph

- Considers broadcast media by hyperarcs
- No losses, i. e., single hyperarc from some $a \in N$ to all its neighbors $b \in N_a$
- Flow bound: $\boldsymbol{N x} \leq \boldsymbol{z} \quad \mathcal{Z} = \bigcup_{\boldsymbol{\tau} \geq \boldsymbol{0} \wedge \boldsymbol{1}^{\mathrm{T}} \boldsymbol{\tau} \leq 1} \{\boldsymbol{z} : z_j = \tau_j \quad \forall j \in H\}$

Section 4: Model 3 – lossy hypergraph

- Considers broadcast media by hyperarcs
- Allows for losses, i. e., hyperarcs from $a \in N$ to all subsets $B \subset N_a$ of neighbors
- Flow bound: $\boldsymbol{N x} \leq \boldsymbol{Q z} = \boldsymbol{y} \quad \mathcal{Y} = \bigcup_{\boldsymbol{\tau} \geq \boldsymbol{0} \wedge \boldsymbol{1}^{\mathrm{T}} \boldsymbol{\tau} \leq 1} \left\{\boldsymbol{y} : y_j = \tau_a \left(1 - \prod_{k \in A_j} \varepsilon_k\right) \quad \forall j \equiv (a, B) \in \mathcal{H}\right\}$

Note: Model 1 $\subset$ Model 2 $\subset$ Model 3