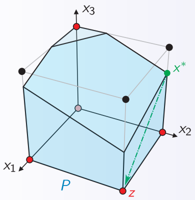
Some Recent Applications of Algorithmic Matroid Theory

Rico Zenklusen

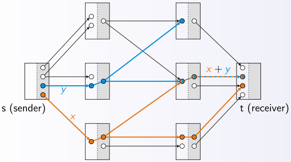
ETH Zurich

Part I



Rounding algorithms with applications to multi-objective optimization

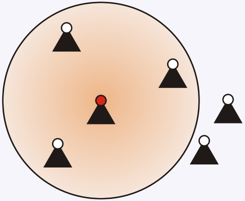
Part II



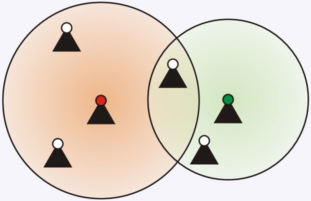
From algorithmic matroid theory to wireless network flows.

Features of wireless information flows

Broadcasting



Superposition

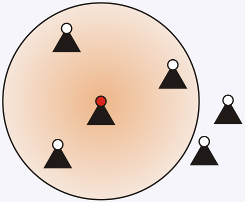


$\Rightarrow$ Complex signal interactions.

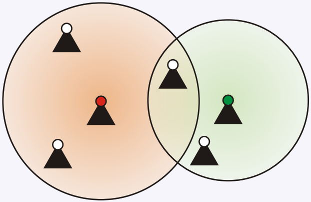
Wireless information flows

Features of wireless information flows

Broadcasting



Superposition



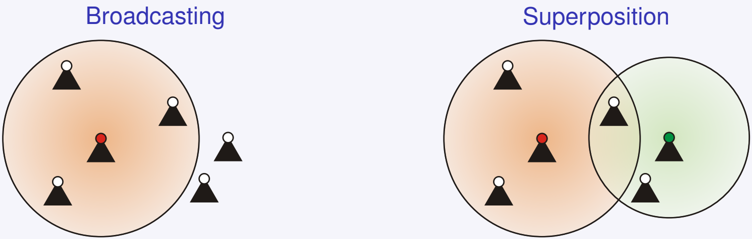
$\Rightarrow$ Complex signal interactions.

Classical model: Multiuser Gaussian Channel

- Unknown how to determine capacity of network.

Wireless information flows

Features of wireless information flows



⇒ Complex signal interactions.

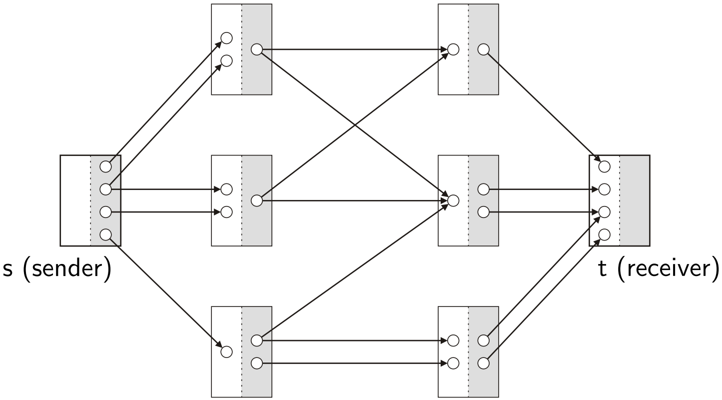
Classical model: Multiuser Gaussian Channel

- Unknown how to determine capacity of network.

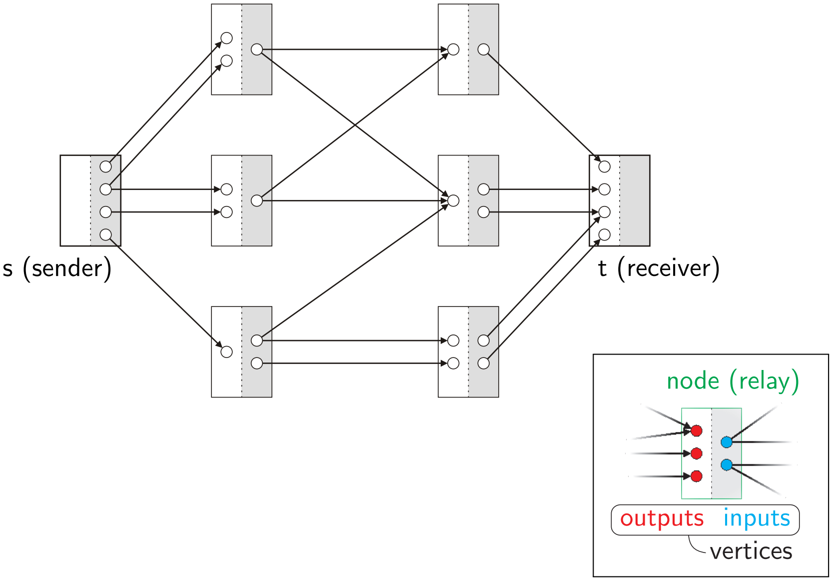
The ADT model (Avestimehr, Diggavi, Tse [2007,2011])

- A deterministic model to approximate Multiuser Gaussian Channels.

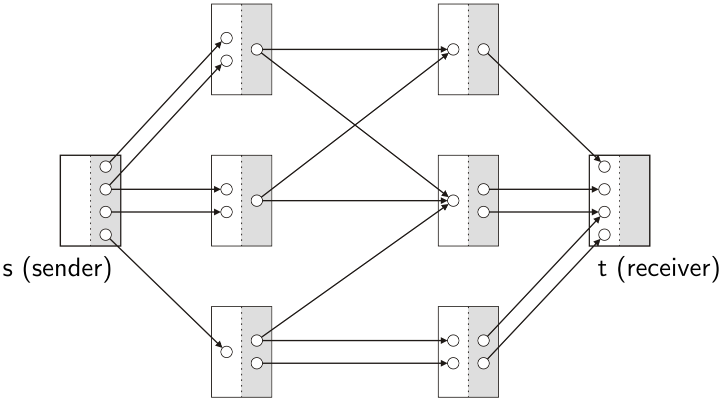
The ADT information flow model



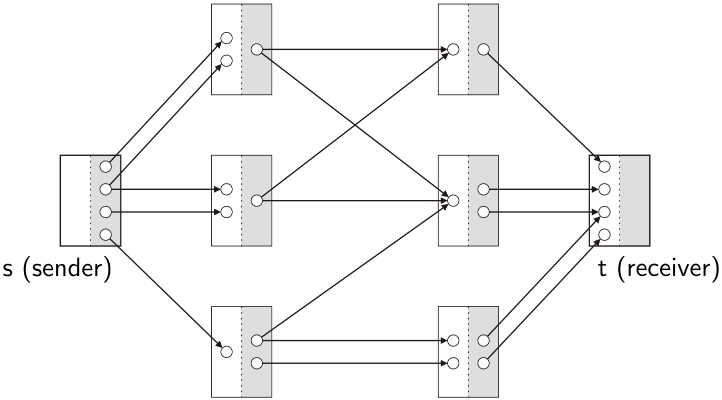
The ADT information flow model



The ADT information flow model

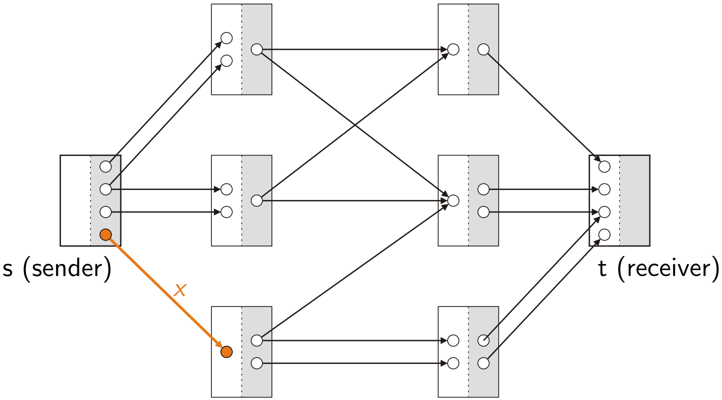


The ADT information flow model

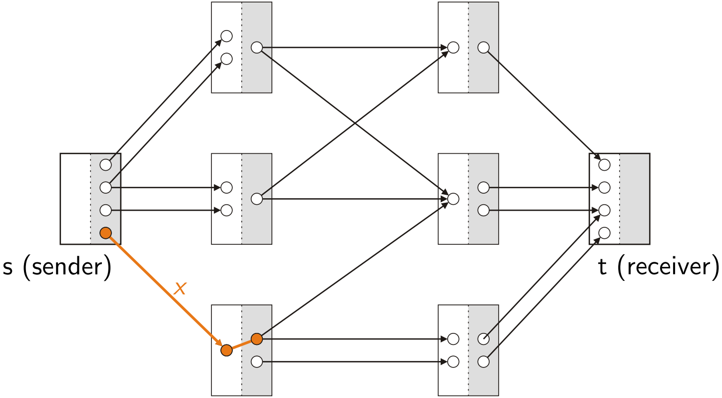


- Task: Send maximum number of signals from s to t.
- A signal is an element of $\mathbb{F}_2$.

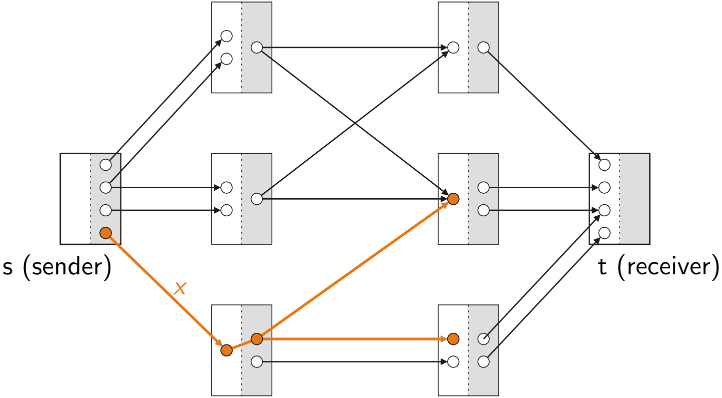
The ADT information flow model



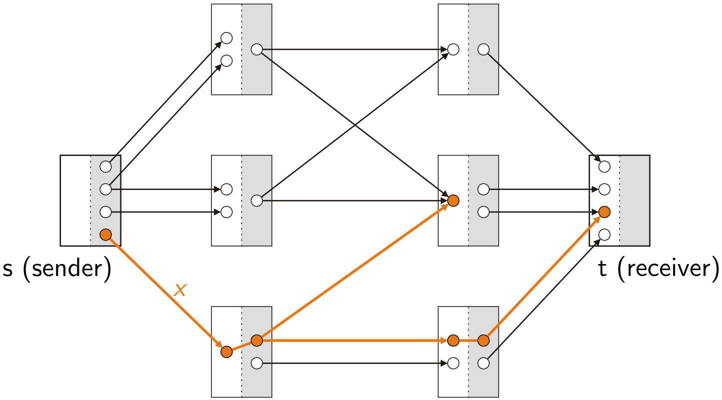
The ADT information flow model



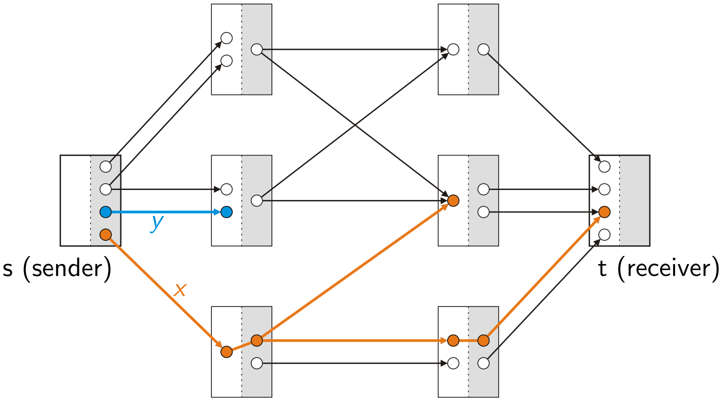
The ADT information flow model

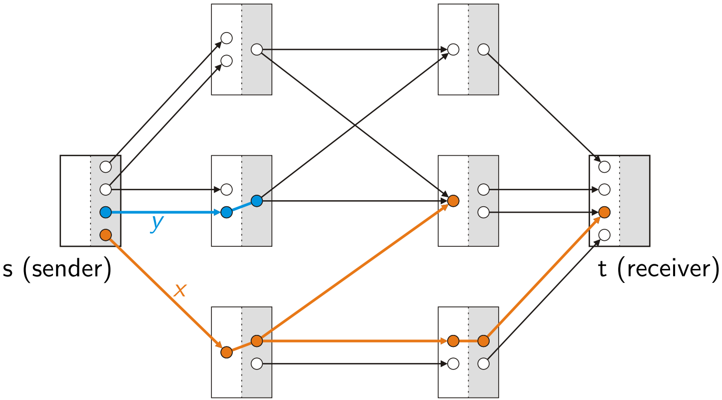


The ADT information flow model

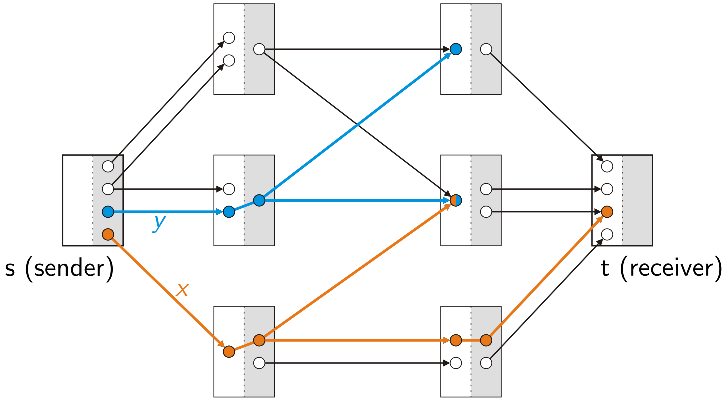


The ADT information flow model



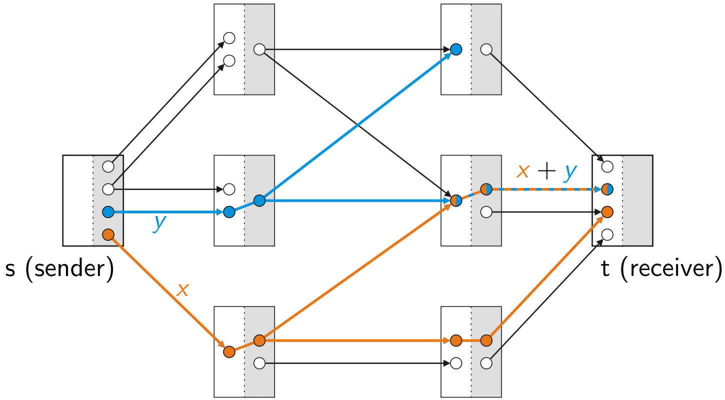

y
x
s (sender)
t (receiver)

The ADT information flow model



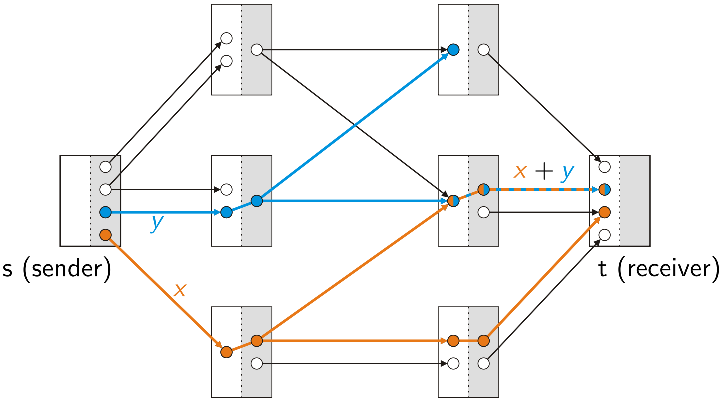
- $\rightarrow$ Interference between the two signals!
- Interference is modelled as XOR.

The ADT information flow model



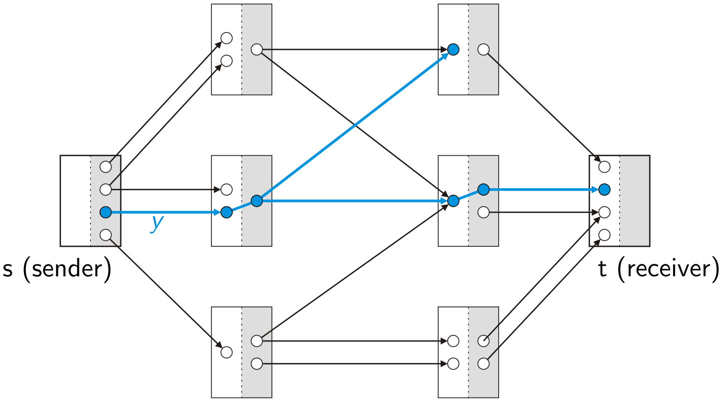
- → Interference between the two signals!
- Interference is modelled as XOR.

The ADT information flow model

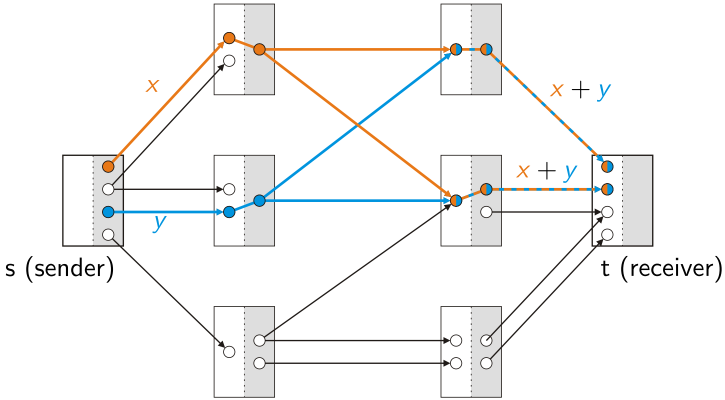


- Receiver gets signals $(x, x+y)$.
- Due to linear independence, received signals can be decoded to get (x, y).

The ADT information flow model



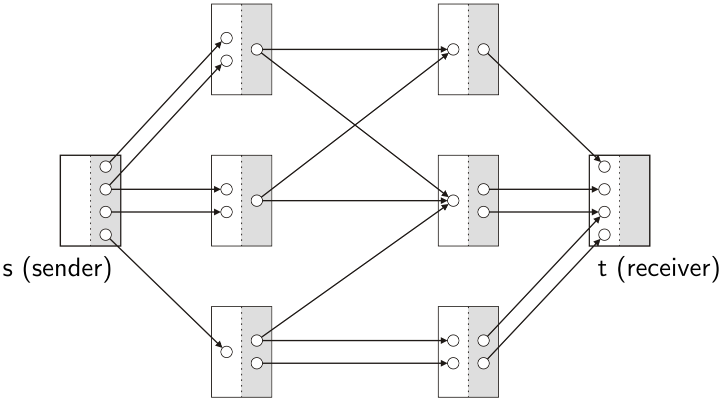
The ADT information flow model



- Received signals are linearly dependent.

  ➞ Receiver cannot decode.

The ADT information flow model



Goal

Route maximum number of decodable (i.e., linearly indep.) signals from *s* to *t*.

First results on ADT model

Theorem **(Avestimehr, Diggavi, Tse [2007,2011])**

There is a notion of ADT cut (think of s-t cut) such that

$$\text{Max ADT flow (coding)} = \text{Min ADT cut.}$$

However, they do not show how to find a max flow or a min cut.

Theorem **(Amadruz, Fragouli [2009] and Ebrahimi, Fragouli [2012])**

A maximum flow and a minimum cut can be found in polynomial time.

Algorithm is computationally expensive and works only for binary signals.

What we will discuss today

Connecting ADT fows to matroids **(Goemans, Iwata, Z. [2012])**

We present a strong link between the ADT model and matroids.

➙ Results/techniques from matroid theory translate into ADT setting.

What we will discuss today

Connecting ADT fows to matroids **(Goemans, Iwata, Z. [2012])**

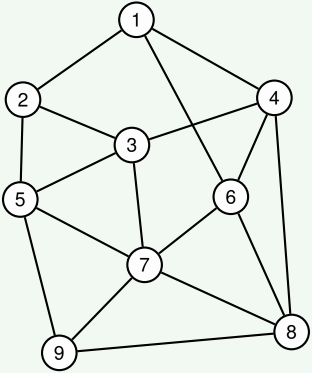
We present a strong link between the ADT model and matroids.

➙ Results/techniques from matroid theory translate into ADT setting.

- Classical matroid algorithms can be used to find optimal coding.
  - ➙ Yields currently fastest algorithm to optimize ADT networks.
  - ➙ Interesting extensions (e.g., min-cost flows, non-binary signals).
- Polyhedral descriptions of all feasible solutions.
- Generalized max-flow min-cut theorem.

Beyond greedy: matroid union

Finding disjoint spanning trees



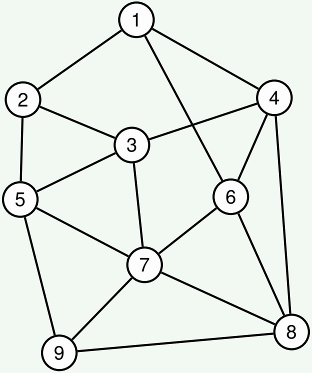
Beyond greedy: matroid union

Definition: union of matroids

Union of matroids $M_i = (E, \mathcal{I}_i)\ \forall i \in [k]$ is $\bigvee_{i=1}^{k} M_i = (E, \bigvee_{i=1}^{n} \mathcal{I}_i)$, where

$$\bigvee_{i=1}^{k} \mathcal{I}_i = \{\cup_{i=1}^{k} I_i \mid I_i \in \mathcal{I}_i\ \forall i \in [k]\}.$$

Finding disjoint spanning trees



Beyond greedy: matroid union

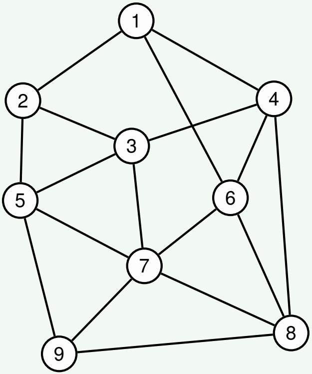
Definition: union of matroids

Union of matroids $M_i = (E, \mathcal{I}_i)\ \forall i \in [k]$ is $\bigvee_{i=1}^{k} M_i = (E, \bigvee_{i=1}^{n} \mathcal{I}_i)$, where

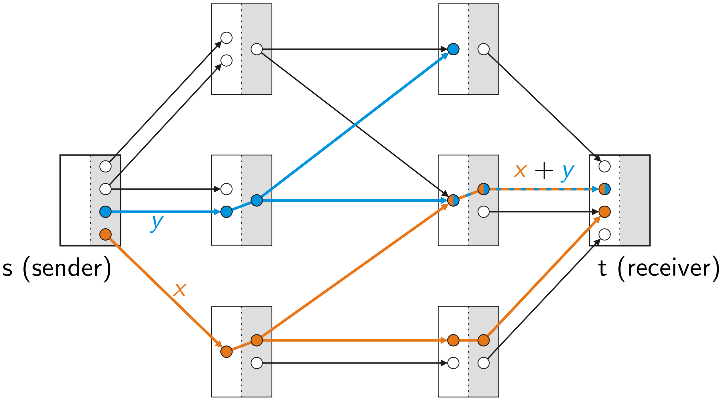
$$\bigvee_{i=1}^{k} \mathcal{I}_i = \{\cup_{i=1}^{k} I_i \mid I_i \in \mathcal{I}_i\ \forall i \in [k]\}.$$

- $M = \bigvee_{i=1}^{k} M_i$ is a matroid.
- Independence oracle for M can be constructed out of the ones for M_i.
- Partition for indep. set in M can be found efficiently.

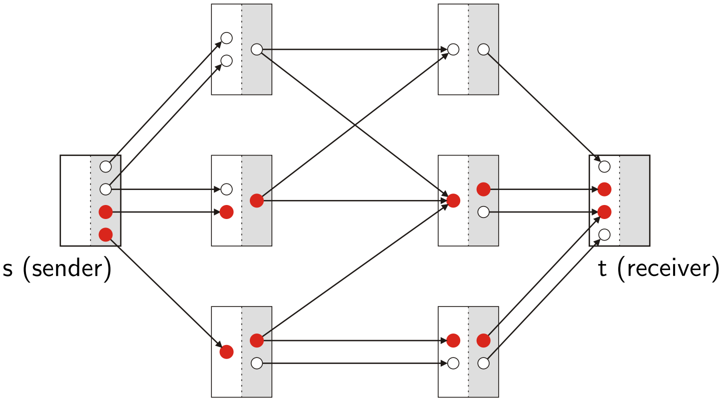
Finding disjoint spanning trees



Representing ADT flows as a set system



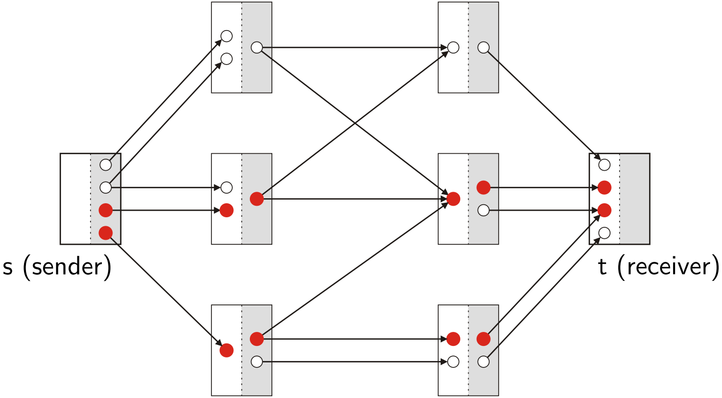
Representing ADT flows as a set system



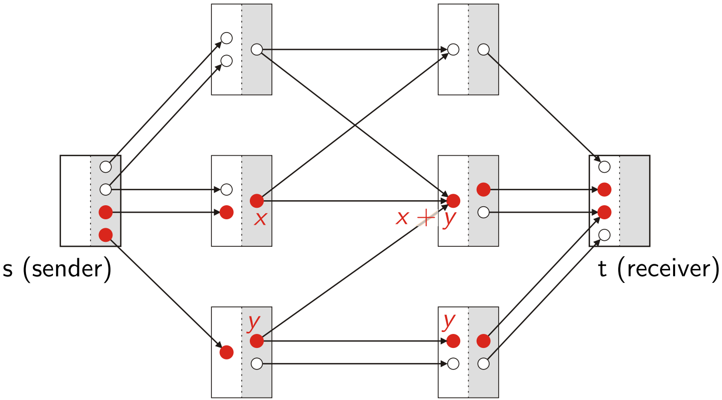
An ADT flow can be represented by set of "used" vertices.

- Information about exact wiring is lost.
  $\rightarrow$ Does not matter since wiring always preserves linear independence.

Representing ADT flows as a set system

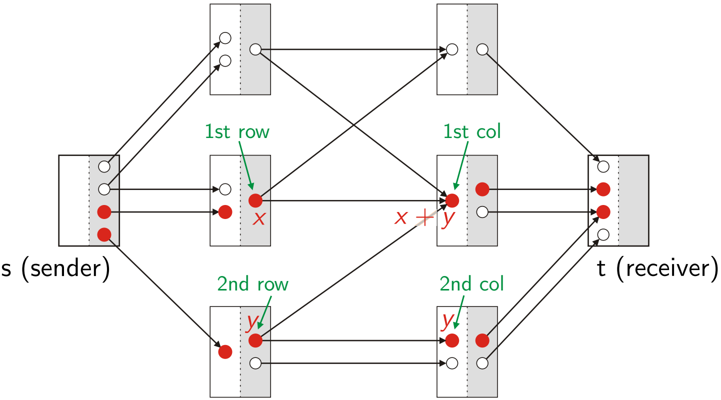


Feasibility for red vertex sets



Propagation of signals from second to third layer:

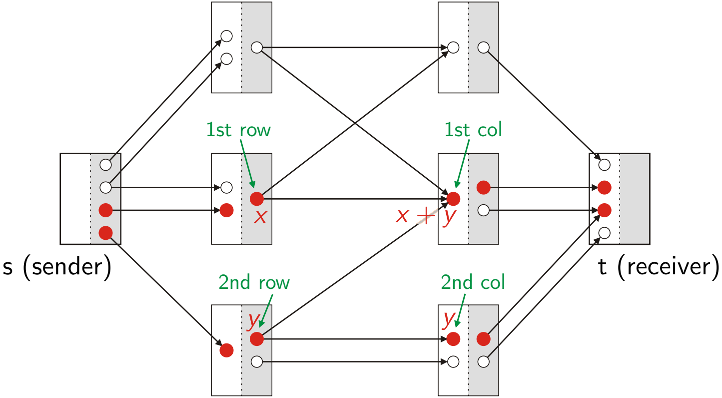
$$(x, y) \cdot \underbrace{\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}}_{\text{Induced adjacency matrix}} = (x + y, y).$$



Propagation of signals from second to third layer:

$$(x, y) \cdot \underbrace{\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}}_{\text{Induced adjacency matrix}} = (x + y, y).$$

Feasibility for red vertex sets

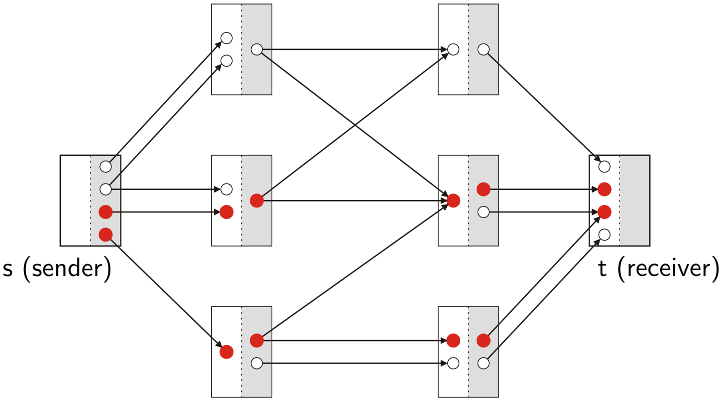


Propagation of signals from second to third layer:

$$(x, y) \cdot \underbrace{\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}}_{\text{Induced adjacency matrix}} = (x + y, y).$$

This matrix has to be full rank.

Feasibility for red vertex sets

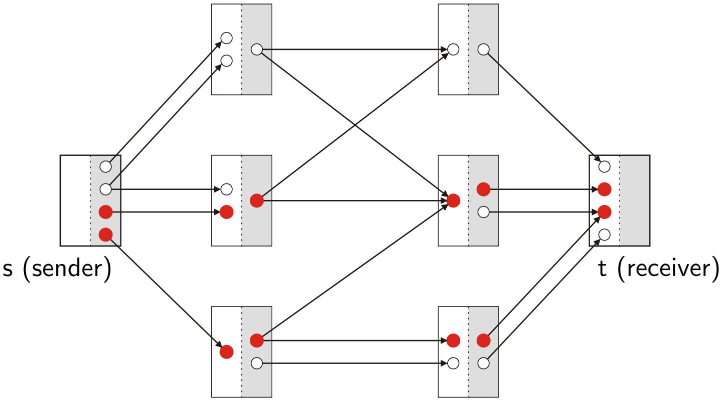


Set of ● is feasible if and only if:

i) Each node has same number of red inputs and outputs.

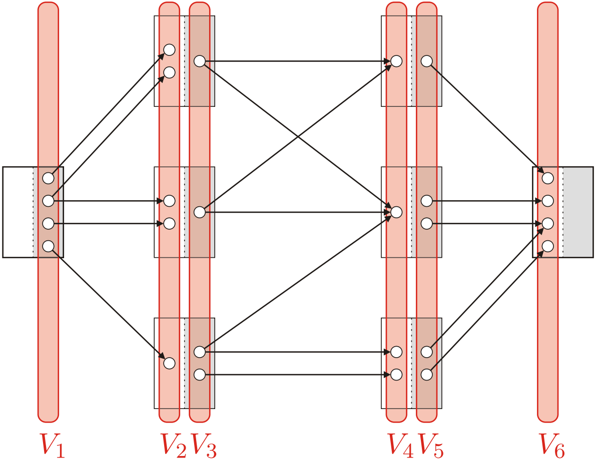
ii) Induced adjacency matrices are full rank.

Feasibility for red vertex sets



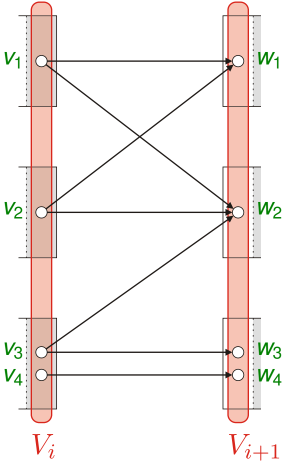
Feasible ADT flow $\leftrightarrow$ feasible set of red vertices.

Describing layer interactions by matroids



We distinguish between relationships $V_i \leftrightarrow V_{i+1}$ → i odd / → i even

Full rank submatrices via matroids (Schrijver [1978])

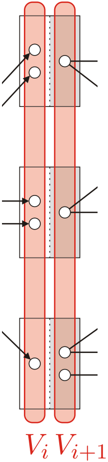


$$A = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} w_1 & w_2 & w_3 & w_4 \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{array}$$

The following defines a matroid $M = (E, \mathcal{I})$:

- $E = V_i \cup V_{i+1}$, and
- $B \subseteq E$ is basis of $M \Leftrightarrow A[B \cap V_i, V_{i+1} \setminus B]$ is (square and) full rank.

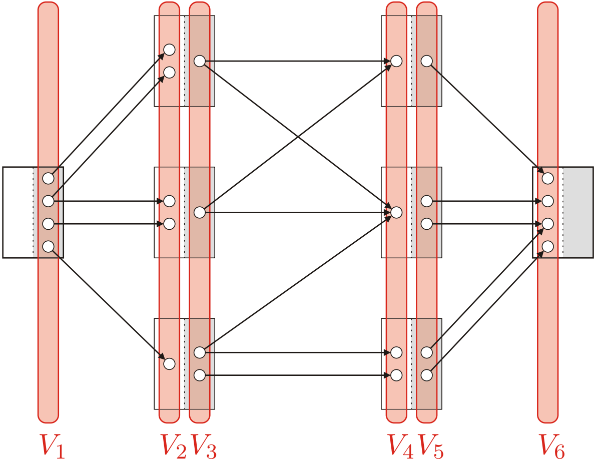
Wiring via matroids



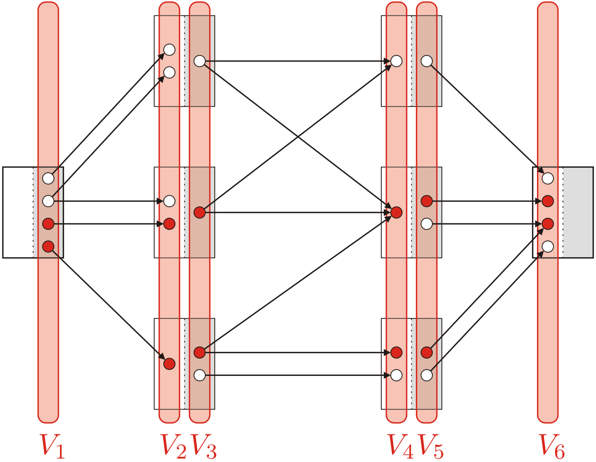
The following defines a matroid $M = (E, \mathcal{I})$:

- $E = V_i \cup V_{i+1}$, and
- $B \subseteq E$ is basis of $M \Leftrightarrow B \cap V_i$ can be wired with $V_{i+1} \setminus B$.

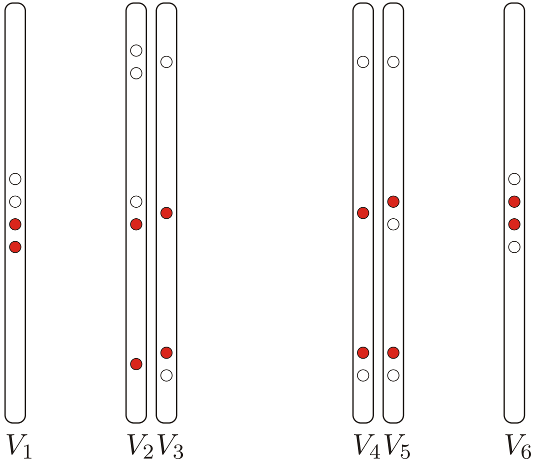
Putting things together



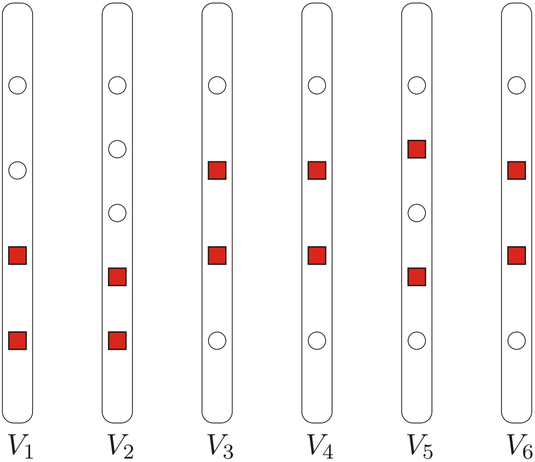
Putting things together

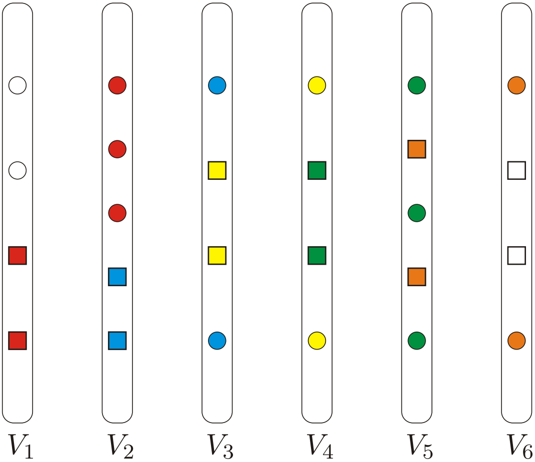


Putting things together

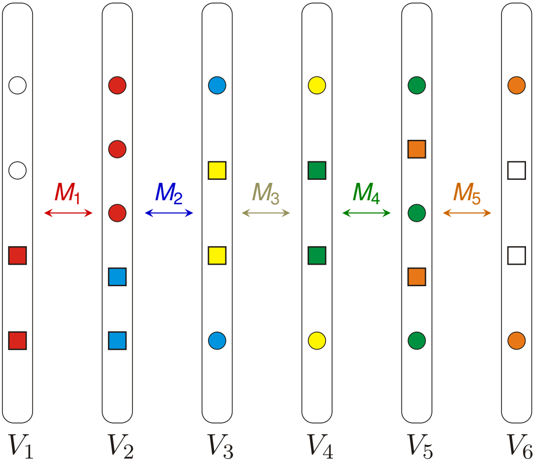


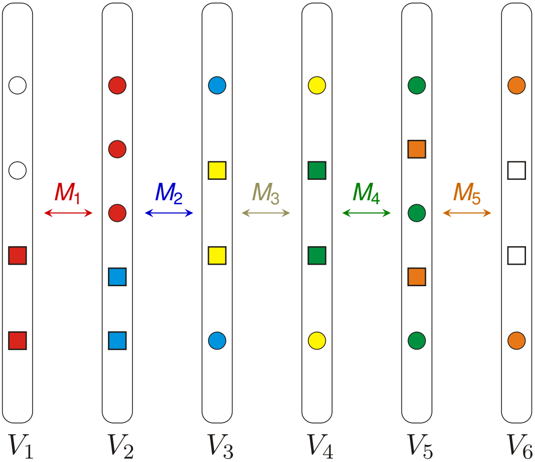
Putting things together




V_1
V_2
V_3
V_4
V_5
V_6

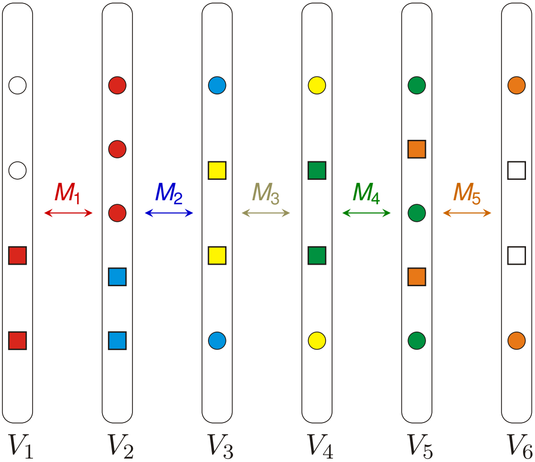
Putting things together





- Each color is a basis in the matroid M_i describing the relation $V_i \leftrightarrow V_{i+1}$.

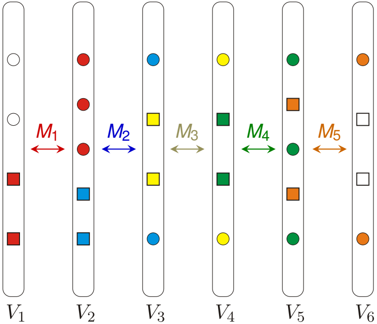
Putting things together



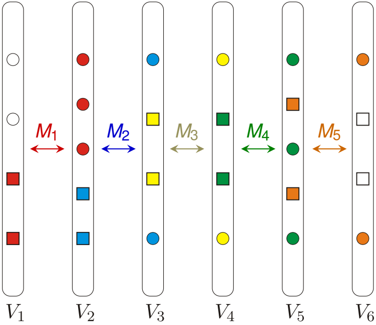
- $\{B_1 \cup \cdots \cup B_5 \mid B_i \text{ basis in } M_i\}$ are bases of another matroid

$$M = M_1 \vee M_2 \vee M_3 \vee M_4 \vee M_5.$$

ADT flows via greedy algorithm



ADT flows via greedy algorithm

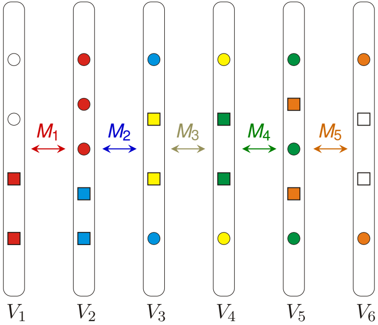


- There is a one-to-one correspondence between:

ADT flows	$\longleftrightarrow$	Bases B (and corresponding color-partition) of M covering all middle layers (V_2, V_3, V_4, V_5).

- Number of indep. signals: $\lvert B \cap V_1 \rvert$.

ADT flows via greedy algorithm



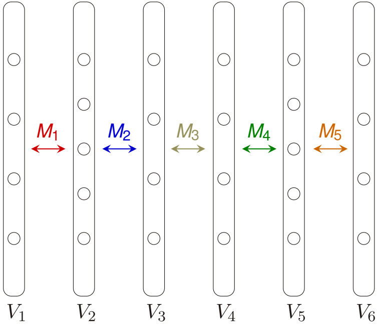
- There is a one-to-one correspondence between:

ADT flows	$\longleftrightarrow$	Bases B (and corresponding color-partition) of M covering all middle layers (V_2, V_3, V_4, V_5).

- Number of indep. signals: $|B \cap V_1|$.

Basis B corresponding to max ADT flow can be found with greedy algorithm.

Getting optimal coding via matroid union



Goal: Find basis B in M covering all middle layers and maximizing $|B \cap V_1|$.

Algorithm: Greedy algorithm to find max weight basis in M wrt weights:

first layer (V_1)	:	1,
middle layers (V_2, V_3, V_4, V_5)	:	$1 + \epsilon$,
last layer (V_6)	:	$1 - \epsilon$.

More implications through link to matroids

Signals in arbitrary finite fields (instead of $\mathbb{F}_2$)

The matroids used to describe full rank submatrices work on any finite field.
$\Rightarrow$ Results hold for any finite field without modification.

More implications through link to matroids

Signals in arbitrary finite fields (instead of $\mathbb{F}_2$)

The matroids used to describe full rank submatrices work on any finite field.
$\Rightarrow$ Results hold for any finite field without modification.

Cost version of ADT flows

One can find max-flow min-cost codings, where using a vertex incurs a cost.

More implications through link to matroids

Signals in arbitrary finite fields (instead of $\mathbb{F}_2$)

The matroids used to describe full rank submatrices work on any finite field.
$\Rightarrow$ Results hold for any finite field without modification.

Cost version of ADT flows

One can find max-flow min-cost codings, where using a vertex incurs a cost.

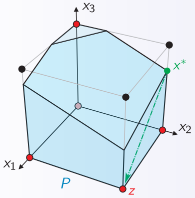
Optimality certificates: notion of s-t cuts

The rank r of a matroid union $M = \vee_{i=1}^{p} M_i$ is

$$r(S) = \min \left\{ |S \setminus A| + \sum_{i=1}^{p} r_i(A) \;\middle|\; A \subseteq S \right\} \qquad \forall S \subseteq V.$$

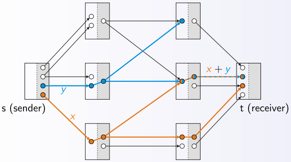
Minimizer $A \subseteq V$ for $r(V \setminus V_p)$ can be interpreted as min s-t cut.
$\rightarrow$ A can be found efficiently using standard matroid algorithms.

Part I



Rounding algorithms with applications to multi-objective optimization

Part II



From algorithmic matroid theory to wireless network flows.

Amaudruz, A. and Fragouli, C. (2009). Combinatorial algorithms for wireless information flow. In *SODA '09: Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 555–564.

Avestimehr, A. S., Diggavi, S. N., and Tse, D. (2011). Wireless network information flow: A deterministic approach. *IEEE Transactions on Information Theory*, 57(4):1872.

Avestimehr, A. S., Diggavi, S. N., and Tse, D. N. C. (2007). A deterministic approach to wireless relay networks. In *Proceedings of Allerton Conference on Communication, Control, and Computing*.

Ebrahimi, J. B. and Fragouli, C. (2012). Combinatiorial algorithms for wireless information flow. *ACM Transactions on Algorithms*, 9(1):8:1–8:33.

Goemans, M. X., Iwata, S., and Zenklusen, R. (2012). A flow model based on polylinking systems. *Mathematical Programming, Series A*, 135(1-2):1–23.