Communications Network Design

lecture 17

Matthew Roughan
<matthew.roughan@adelaide.edu.au>

Discipline of Applied Mathematics
School of Mathematical Sciences
University of Adelaide

March 2, 2009

Advanced tree-like network design

Tree-like networks, and some more advanced algorithms. Starting with **cutsets** *we get* **Gomory-Hu** and **Gusfield's** methods.

Tree-like networks

The problems can be bit more complicated

- in cable TV network, no congestion cost, as content is replicated
- in Ethernet, congestion is arbitrarily delt with using weights that depend on bandwidth
- in some networks we may have to deal with load based costs

Costs

Take a general linear cost model $C(\mathbf{f}) = \sum_{e \in L} (\alpha_e f_e + \beta_e)$

- last lecture we considered the **minimum weight spanning tree** (MWST) which has $\alpha_e = 0$, so

$$C(\mathbf{f}) = \sum_{e \in T} \beta_e$$

- today, we consider the case $\beta_e = 0$, so

$$C(\mathbf{f}) = \sum_{e \in T} \alpha_e f_e$$

- unfortunately, this is NP-complete

Methods of attack

- enumeration impractical (too many trees)
- use standard trick from before

$$C(\mathbf{f}) = \sum_{e \in T} \alpha_e f_e = \sum_{[p,q] \in K} l_{pq}(T) t_{pq}$$

- use a new idea, based on **cutsets**

Cutsets

Take a graph $G(N,E)$, then X, $\bar{X}$ is a partition of the nodes N, if

$$\bar{X} = N \backslash X$$

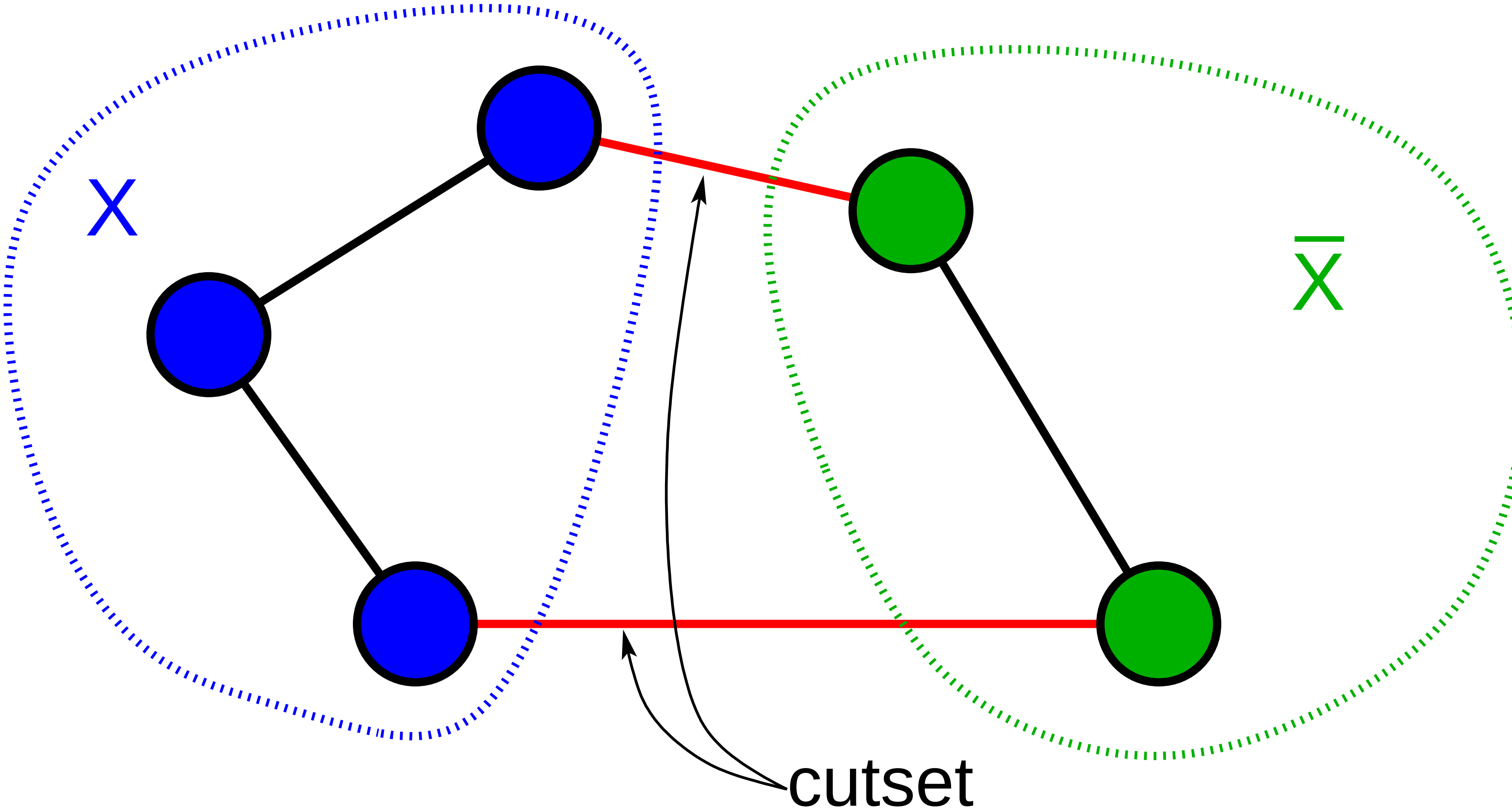
that is

$$\begin{aligned} X \cup \bar{X} &= N \\ X \cap \bar{X} &= \phi \end{aligned}$$

Definition: A **cutset** $(X,\bar{X})$ of $G(N,E)$ is the set of links

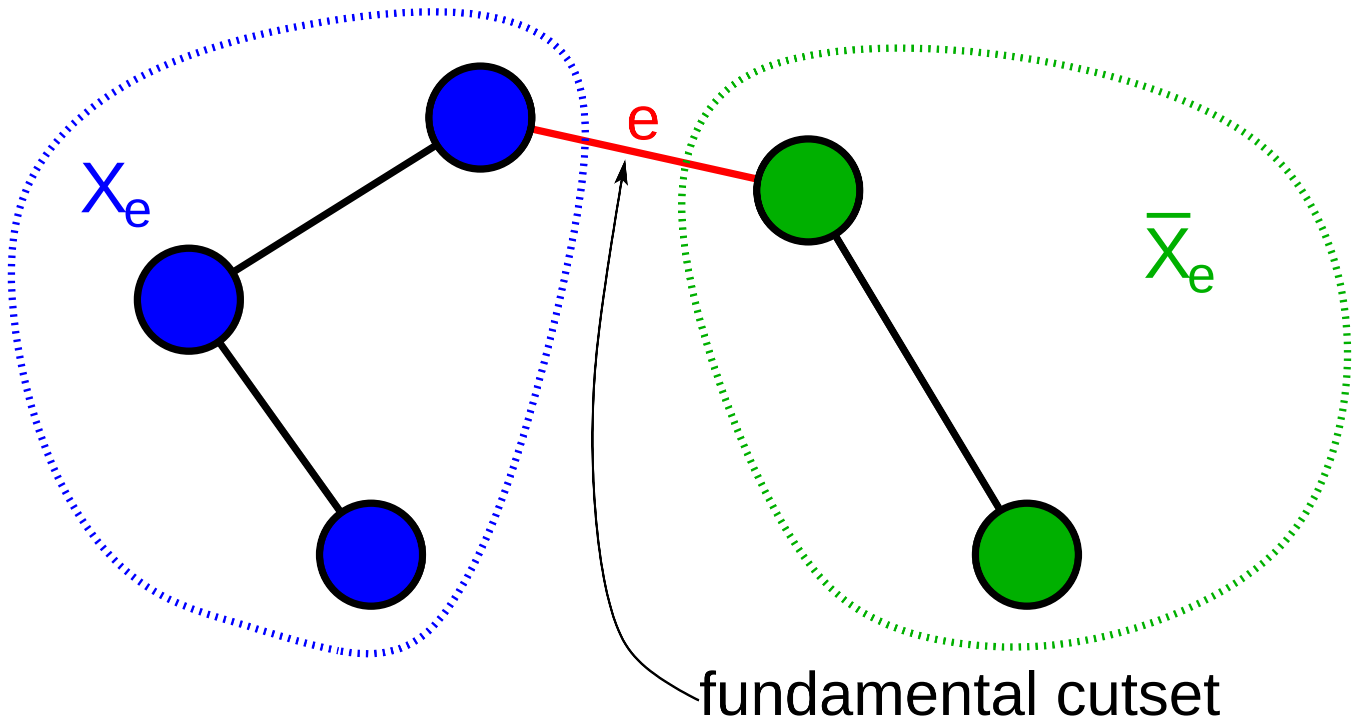
$$(X,\bar{X}) = \{(i,j) \mid i \in X, j \in \bar{X}\}$$

Cutset example



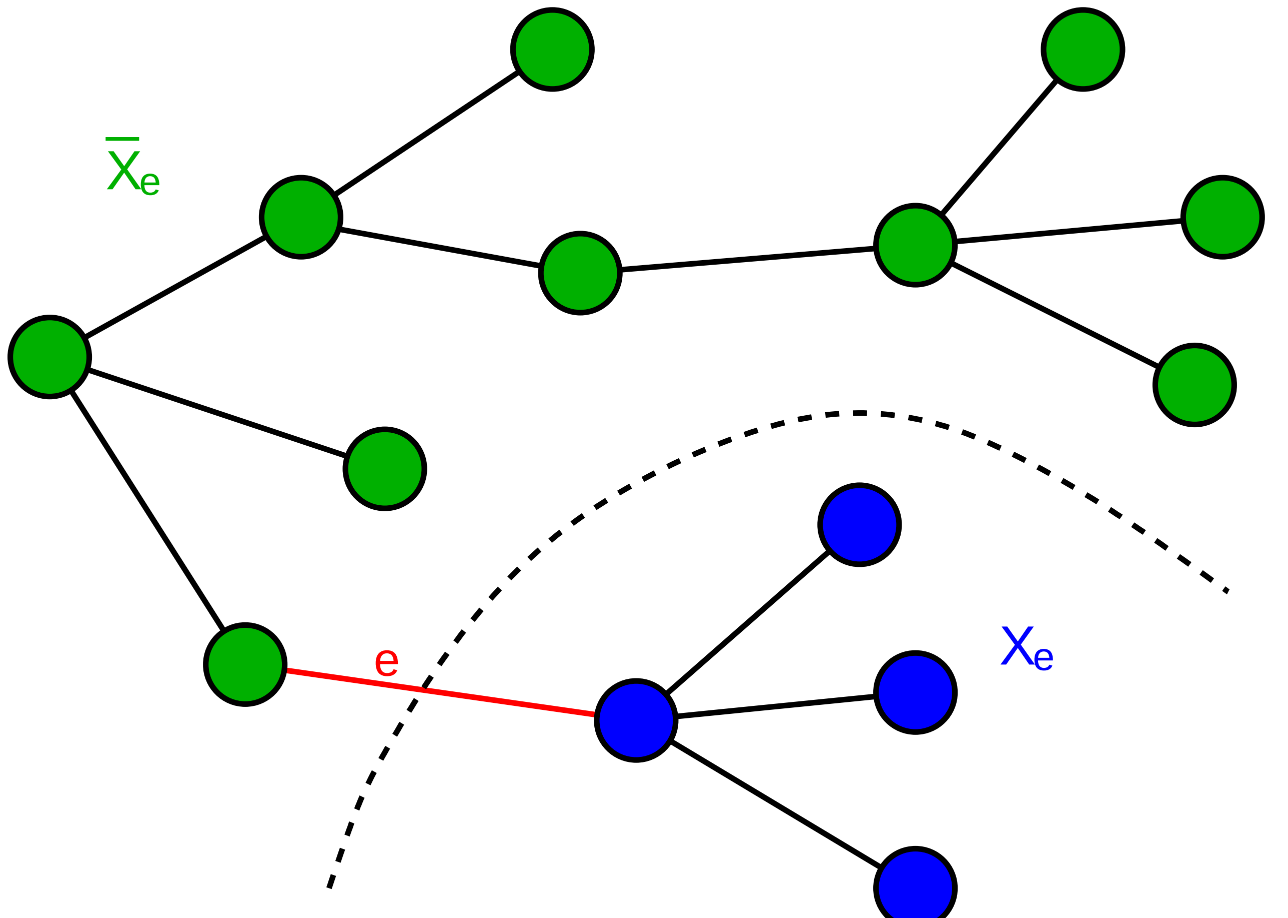
Fundamental Cutset

- Suppose a cutset contains a single link $e \in E$
- if the link e is deleted from T, then T will be disconnected into two subtrees X_e and $\bar{X}_e$
- the cutset $(X_e, \bar{X}_e)$ is called a **fundamental cutset**



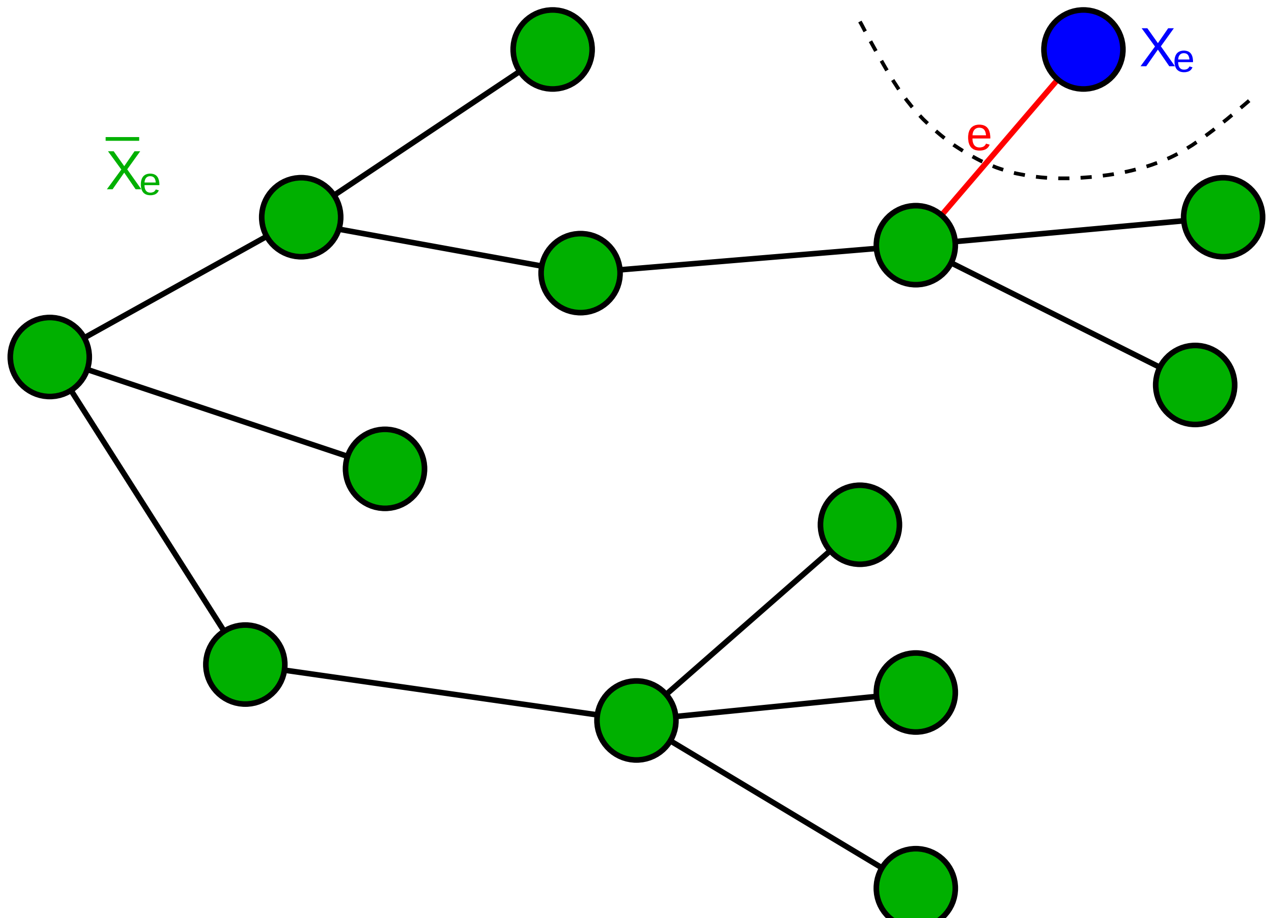
Fundamental Cutset

- for a tree T with $n-1$ links, there are $n-1$ fundamental cutsets
  - cutting any link makes network disconnected



Fundamental Cutset

- for a tree T with $n-1$ links, there are $n-1$ fundamental cutsets
  - cutting any link makes network disconnected



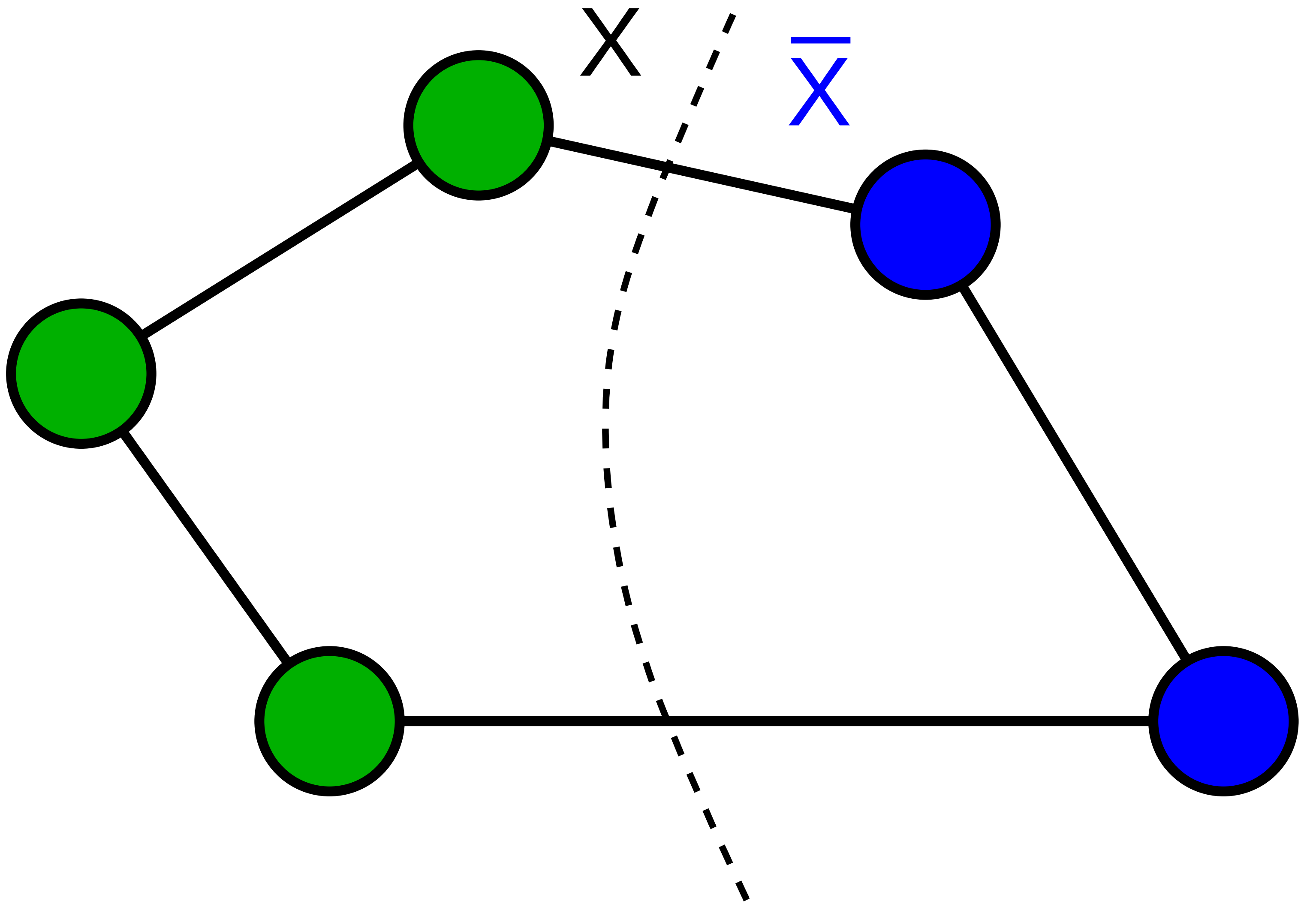
Non-crossing cutsets

Definition: Cutsets $(X, \bar{X})$ and $(Y, \bar{Y})$ are said to be **crossing** if

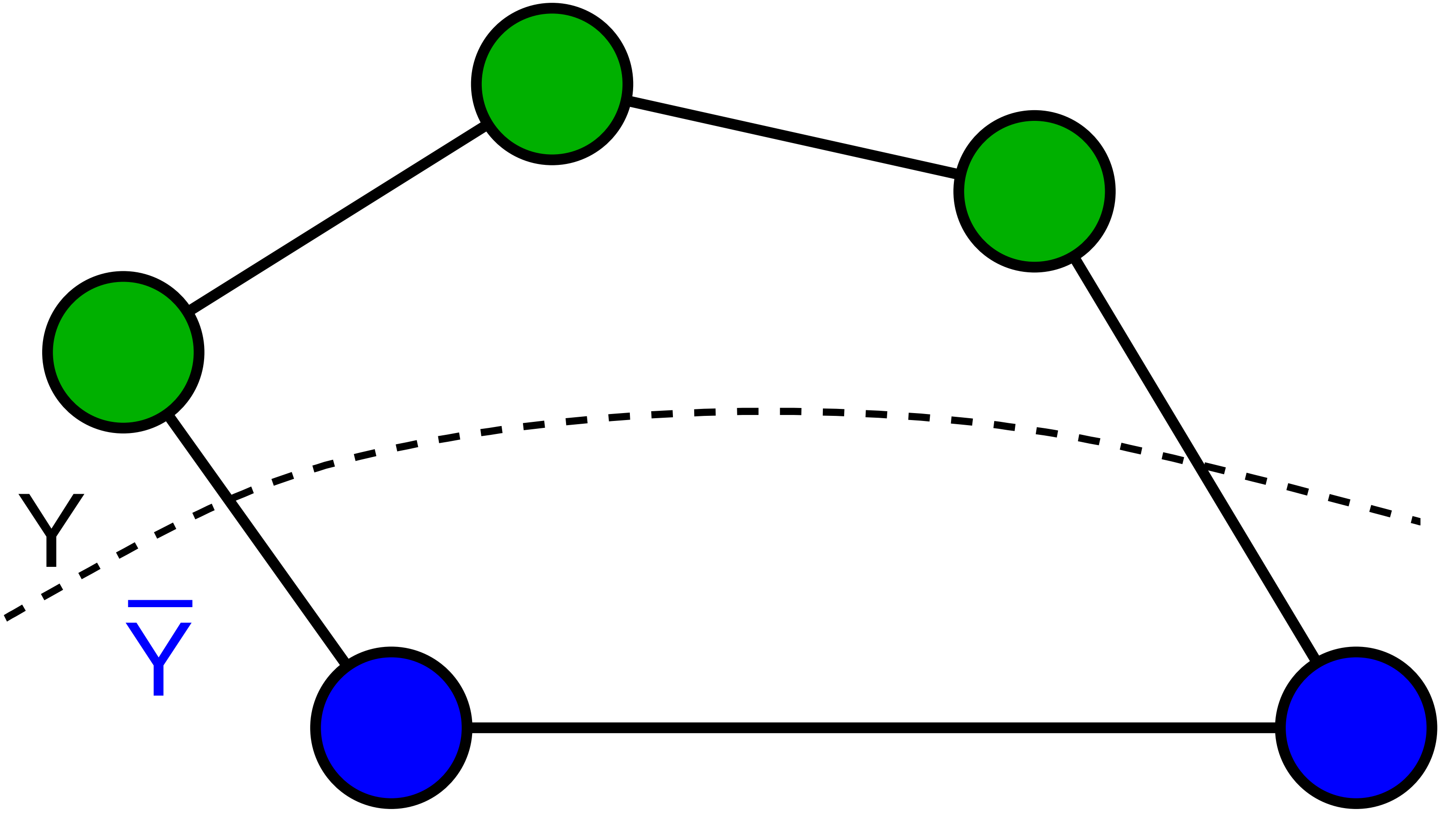
$X \cap Y \neq \emptyset, \qquad X \cap \bar{Y} \neq \emptyset, \qquad \bar{X} \cap Y \neq \emptyset, \qquad$ and $\qquad \bar{X} \cap \bar{Y} \neq \emptyset$

Definition: Cutsets $(X, \bar{X})$ and $(Y, \bar{Y})$ are said to be **non-crossing** if at least one of the above intersections **is** empty.

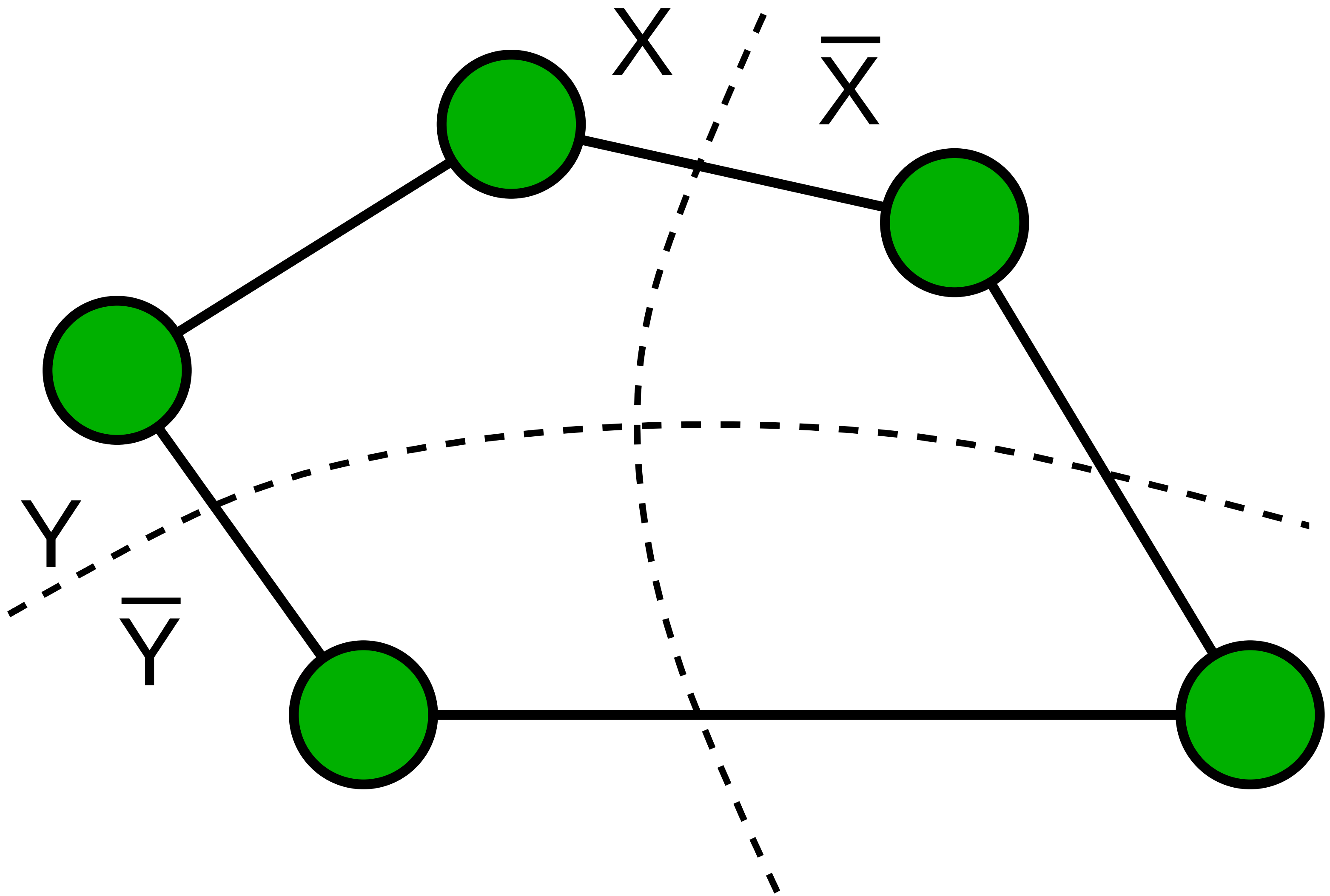
Crossing cutsets examples



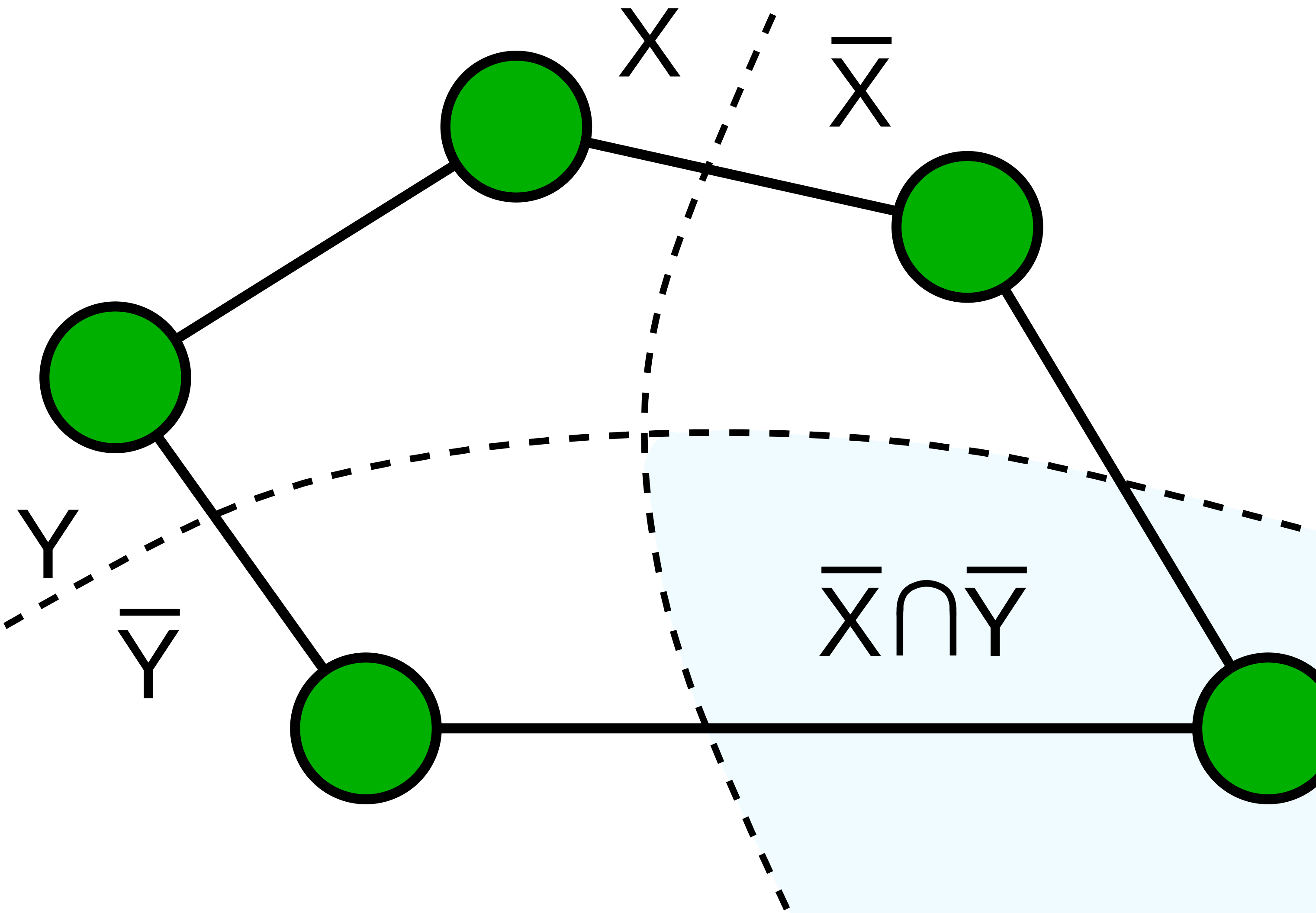
Crossing cutsets examples



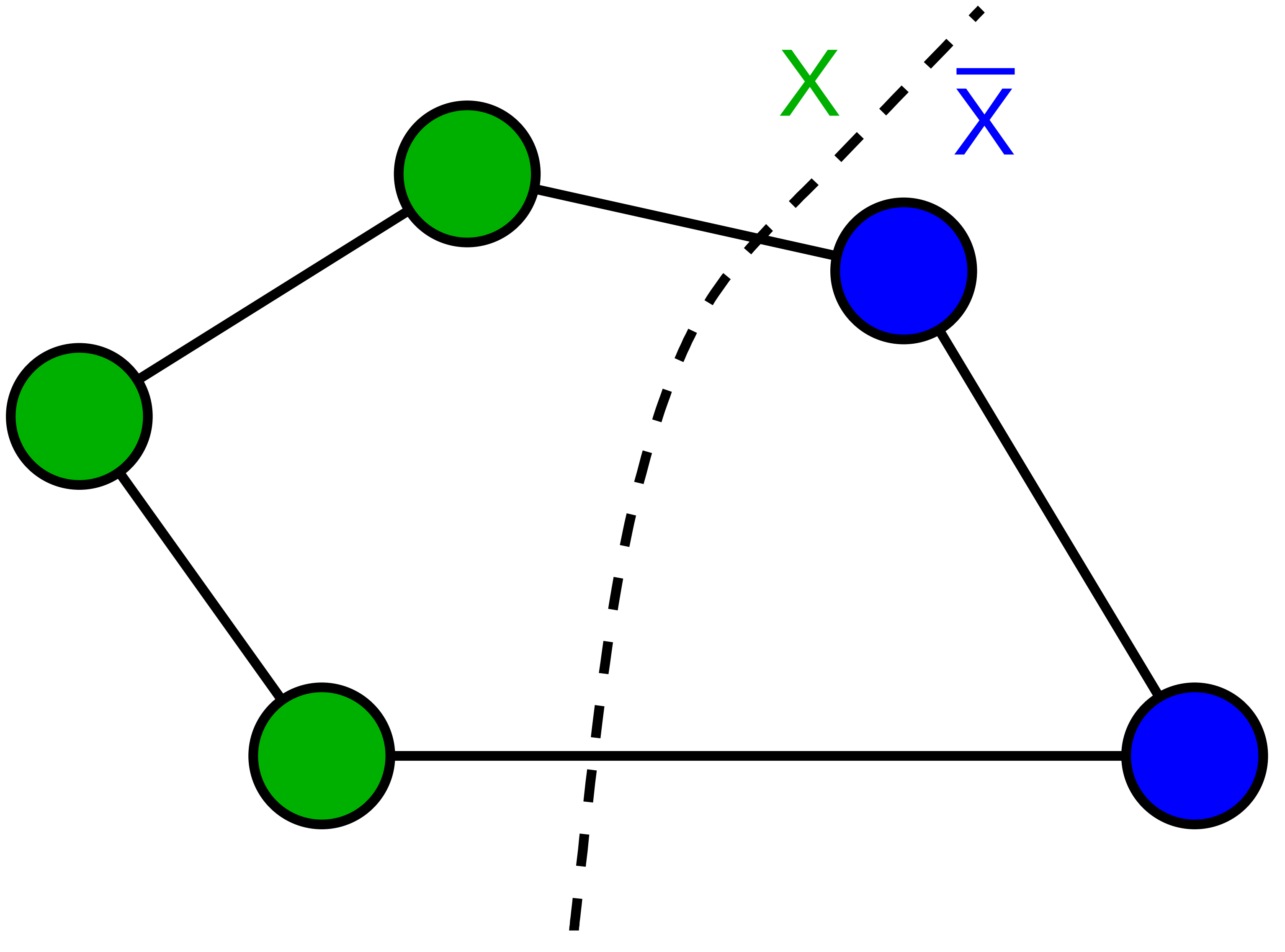
Crossing cutsets examples



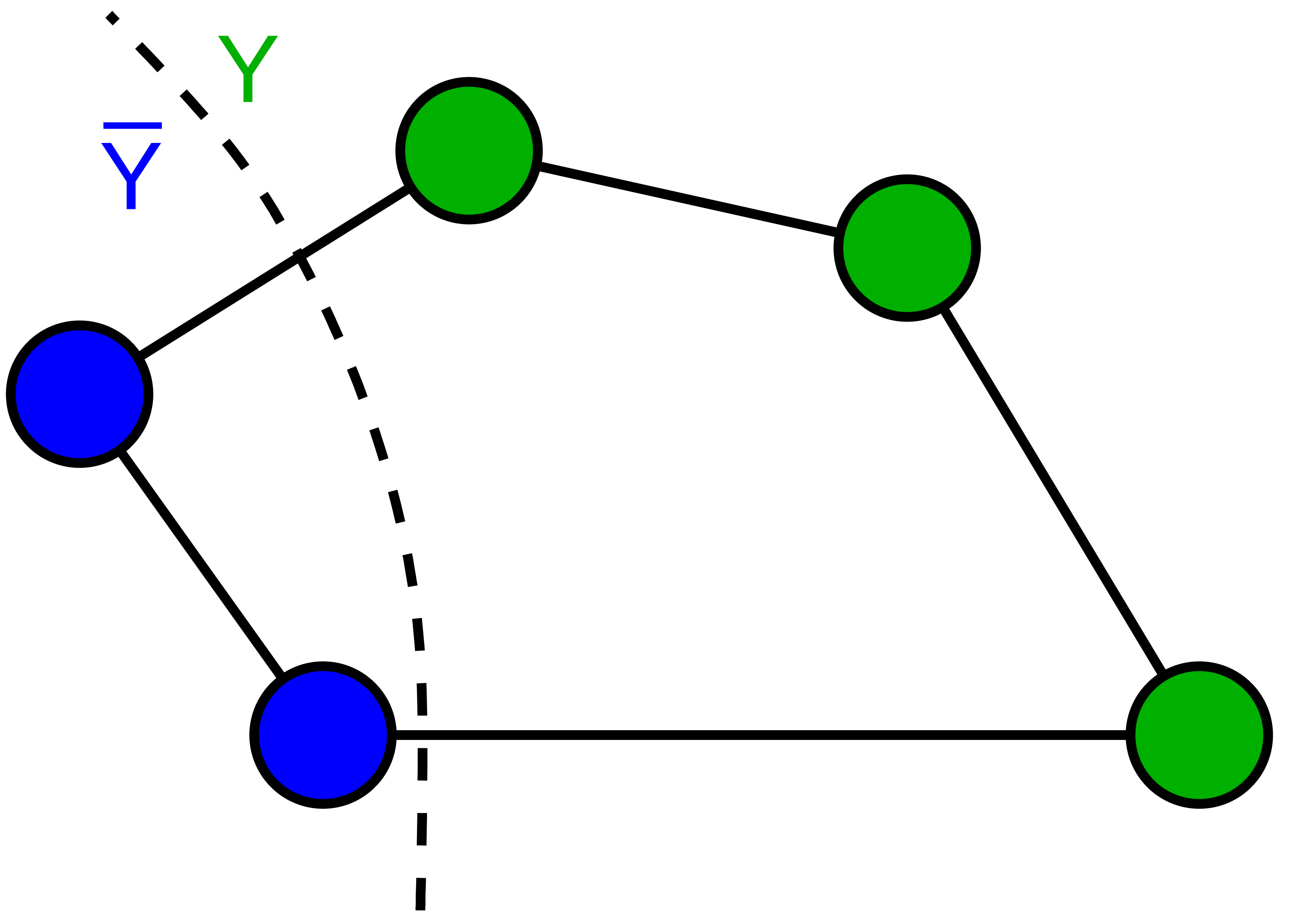
Crossing cutsets examples



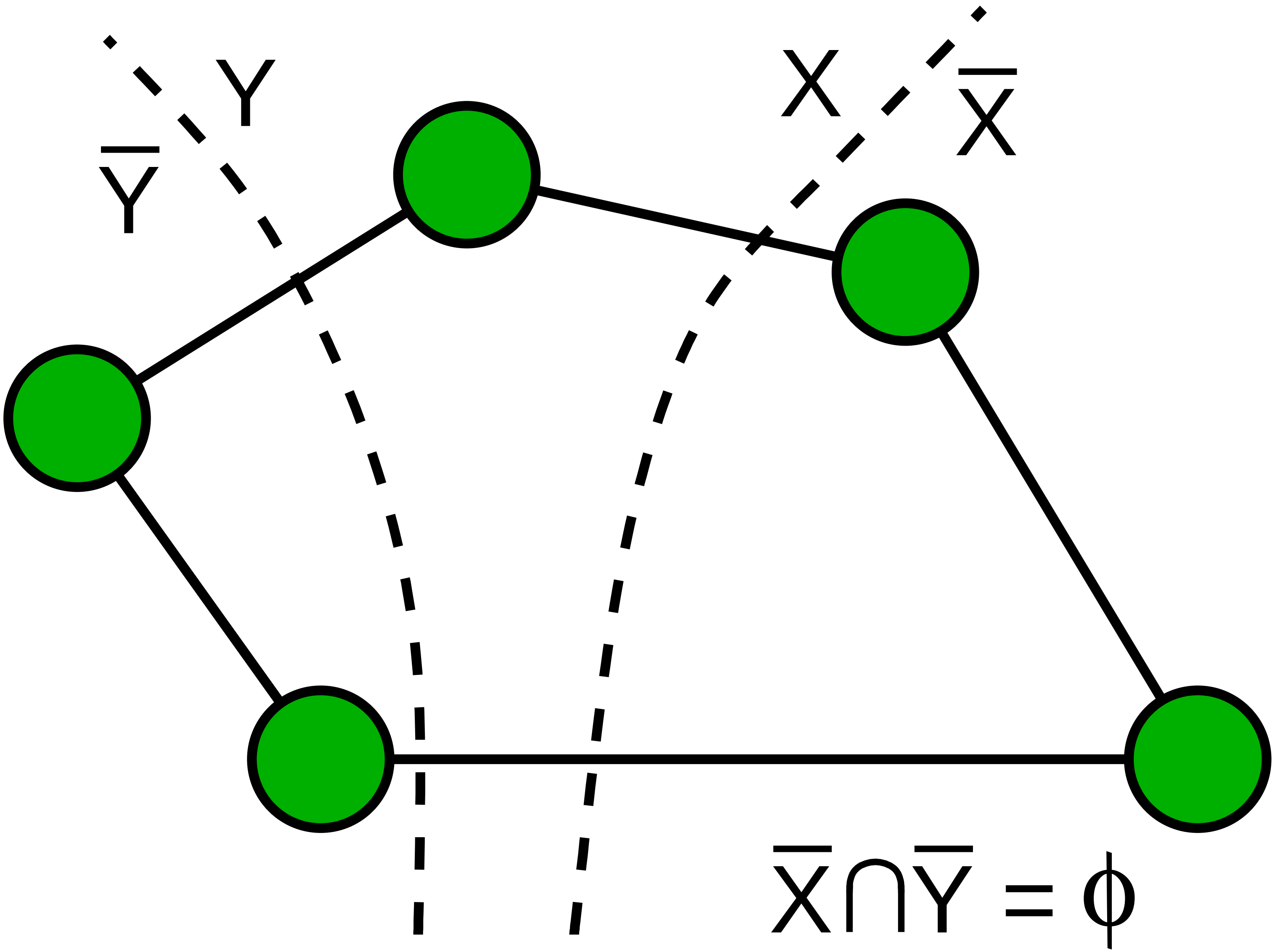
Non-crossing cutsets examples



Non-crossing cutsets examples



Non-crossing cutsets examples



$\overline{X} \cap \overline{Y} = \phi$

Non-crossing cutsets and trees

- Fundamental cutsets are non-crossing!
  - so a tree has at least $n-1$ non-crossing cutsets
- also, suppose $(X_e, \bar{X}_e)$ is a fundamental cutset
  - if the O-D pair has $p \in X_e$ and $q \in \bar{X}_e$
  - all traffic t_{pq} must pass through e
  - $(X_e, \bar{X}_e)$ is said to **separate** p and q
  - the traffic on link e will be

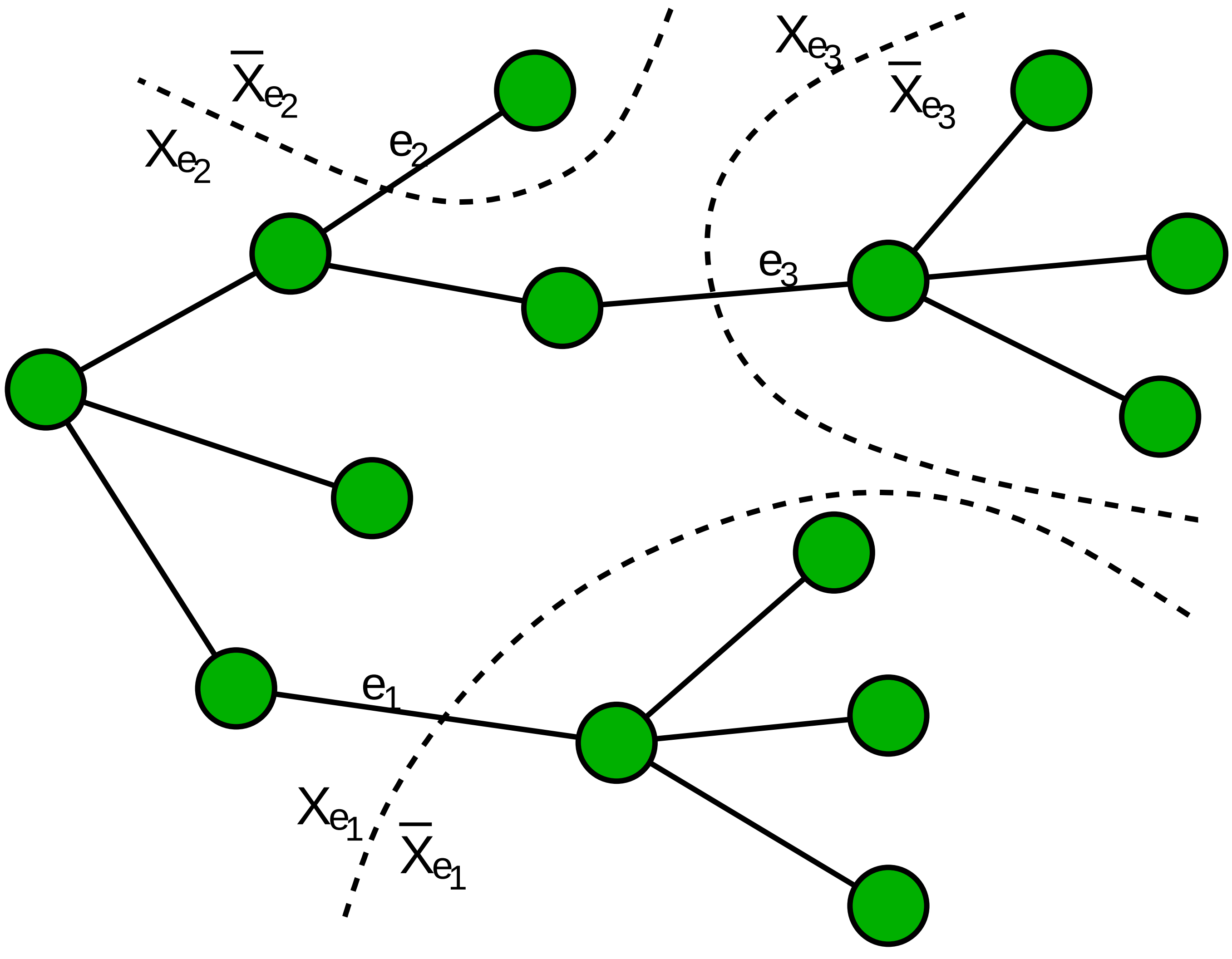
$$f_e = \sum_{p \in X_e} \sum_{q \in \bar{X}_e} t_{pq} := t(X_e, \bar{X}_e)$$

i.e., the traffic between sets X_e and $\bar{X}_e$ is $t(X_e, \bar{X}_e)$

- network cost will be

$$C(\mathbf{f}) = \sum_{e \in T} \alpha_e f_e = \sum_{e \in T} \alpha_e t(X_e, \bar{X}_e)$$

Cutsets and trees example



e.g. $\bar{X}_{e_1} \cap \bar{X}_{e_2} = \bar{X}_{e_2} \cap \bar{X}_{e_3} = \bar{X}_{e_3} \cap \bar{X}_{e_1} = \phi$

Min-hop tree

- we will simplify to the case where

$$\alpha_e = 1, \quad \forall e \in E$$

$$C(\mathbf{f}) = \sum_{e \in T} f_e = \sum_{[p,q] \in K} \hat{l}_{pq}(T) t_{pq} = \sum_{e \in T} t(X_e, \bar{X}_e)$$

- equivalent to minimizing hop count $\hat{l}_\mu(T) = \sum_{e:e\in\mu} 1$
  - implicitly assumes **processing** time for a packet at a node dominates performance.
- result is called a **min hop tree**
  - also called a Gomory-Hu tree (we see why below)
- can be found in $O(|N|^2|E|)$ time, which is polynomial

Gomory-Hu Method

Objective: given a graph $G(N,E)$, and predicted traffic t_{pq}, find a min hop tree.

Principle: find a set of $n-1$ non-crossing cutsets that minimize $t(X_e, \overline{X}_e)$ at each step.

- another greedy algorithm
  - choose the best cutset at each stage
  - however, it does reach the optimum
- $n-1$ non-crossing cutsets define our tree, e.g.
  - **Lemma:** A spanning tree with $n-1$ links corresponds uniquely to a set of $n-1$ non-crossing cutsets.
  - the links occuring in exactly one cutset form a spanning tree T.

Lemma proof

Proof: $(\Rightarrow)$ Given T, removing any link $e \in T$ disconnects the network into T_e and $\overline{T_e}$, and so corresponds to a fundamental cutset $(T_e, \overline{T_e})$. Now we can do the same with T_e, or $\overline{T_e}$. Imagine we partition T_e with cutset $(T_g, \overline{T_g})$, then $T_g \subset T_e$, and so $T_g \cap \overline{T_e} = \phi$, and so these are non-crossing cutsets. Repeat recursively, until, after removing $n-1$ links, we will have $n-1$ non-crossing cutsets.

Lemma proof (continued)

Proof: $(\Leftarrow)$

Suppose we have a set of $(n-1)$ non-crossing cutsets, $\{F_1, F_2, \ldots, F_{n-1}\}$. Construct a spanning tree T as follows. Consider the cut $F_1 = (X_1, \bar{X}_1)$. Draw two supernodes, one corresponding to the set of nodes in X_1, and the other to those in $\bar{X}_1$; connect by a link. This creates a link in the spanning tree. Now consider the next cut, $F_2 = (X_2, \bar{X}_2)$. Since F_2 does not cross F_1, we have $X_2 \subset X_1$ and $\bar{X}_1 \subset \bar{X}_2$, (or we have $X_1 \subset X_2$ and $\bar{X}_2 \subset \bar{X}_1$). Then we can create a tree with three supernodes, X_2, $X_1 - X_2$, and $\bar{X}_1$, and two links in a spanning tree. Continue in this manner for all $n-1$ cutsets F_i, to get the $(n-1)$ links in T.

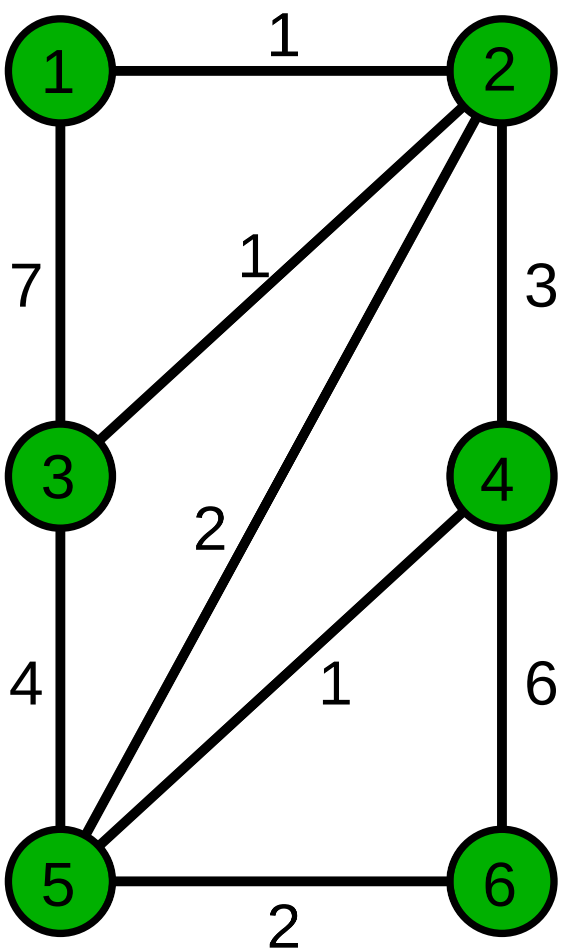
□

Gomory-Hu Algorithm

- **Initialize:** $\mathcal{F} = \phi$ is a list of non-crossing cutsets.
- **While:** at least one pair of nodes p and q are not yet separated by a cutset in $\mathcal{F}$.
  1. select a pair of nodes $p, q \in N$ not yet separated by a cutset in $\mathcal{F}$
  2. find a cutset $(X_{pq}, \bar{X}_{pq})$ that
     - minimizes $t(X, \bar{X})$ subject to
     - $(X, \bar{X})$ separates p and q
     - $(X, \bar{X})$ does not cross any cutset in $\mathcal{F}$
  3. put $\mathcal{F} \leftarrow \mathcal{F} \cup \{(X_{pq}, \bar{X}_{pq})\}$, and record $t(X_{pq}, \bar{X}_{pq})$
- **Terminate:** Determine the set of links contained in exactly one cutset — these links form T.

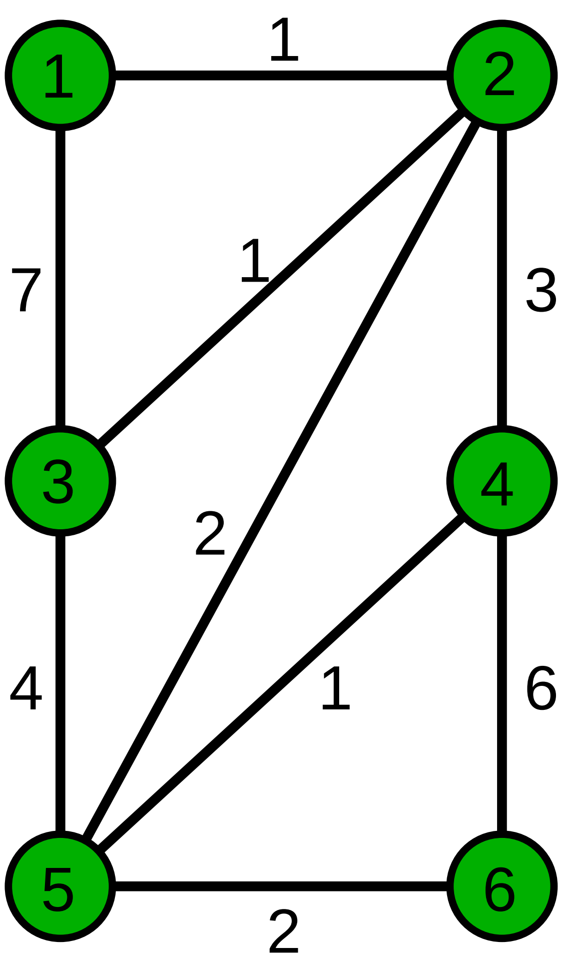
Gomory-Hu Example

The traffic t_{pq}
(zero entries not shown)

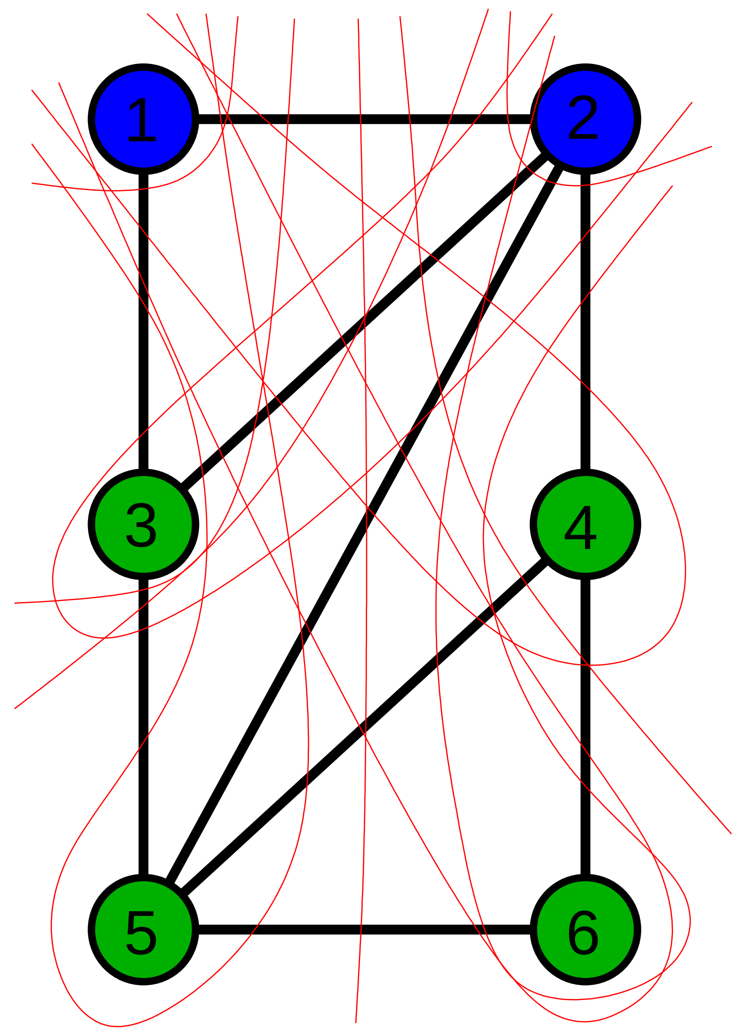


Gomory-Hu Example

The traffic t_{pq}
(zero entries not shown)



The possible cutsets $(X_{12}, \bar{X}_{12})$



Gomory-Hu Example

A list of the possible cutsets separating nodes 1 and 2

$$
\begin{aligned}
X_{12} \quad &= \{1\}\ \{1,3\}\ \{1,4\}\ \{1,5\}\ \{1,6\}\ \{1,3,4\}\ \{1,3,5\}\ \{1,3,6\} \\
&\quad \{1,4,5\}\ \{1,4,6\}\ \{1,5,6\}\ \{1,3,4,5\}\ \{1,3,4,6\} \\
&\quad \{1,3,5,6\}\ \{1,4,5,6\}\ \{1,3,4,5,6\}.
\end{aligned}
$$

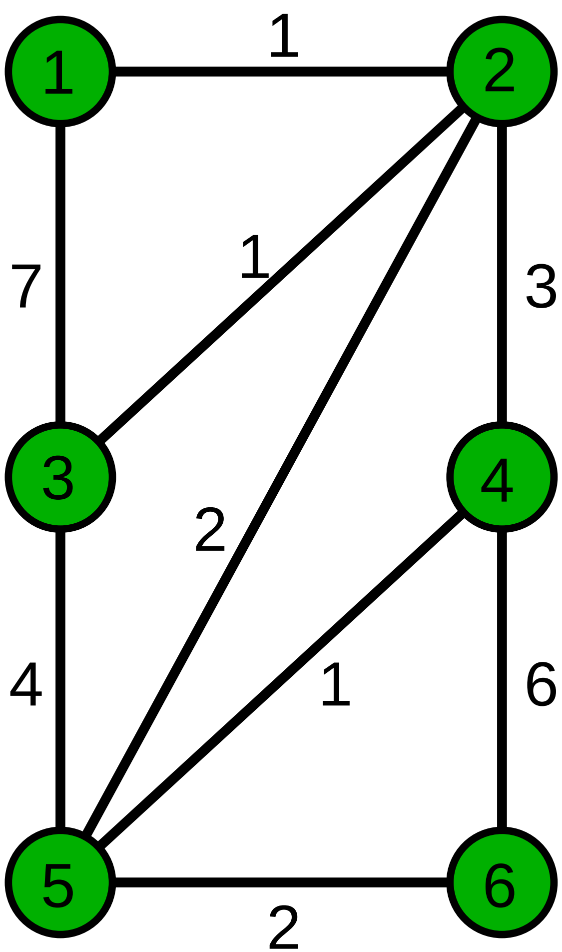
Here the one with minimum value has

$$X_{12} = \{1,3\} \quad \text{and} \quad \bar{X}_{12} = \{2,4,5,6\}$$

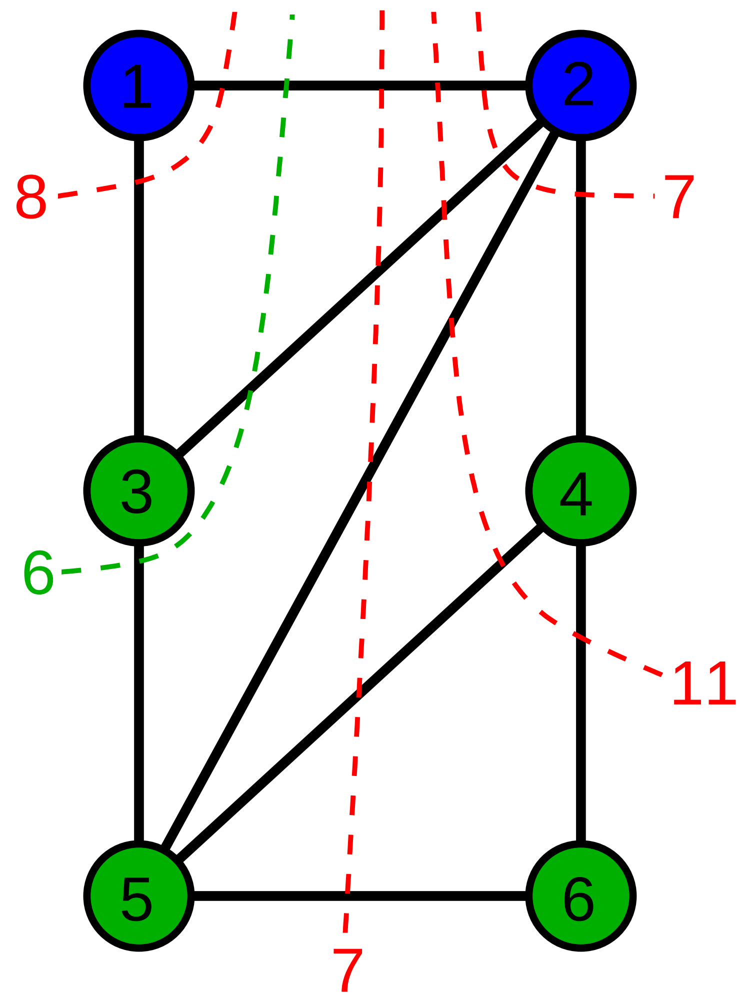
with value $4+1+1 = 6 = \nu_e$, so $\mathcal{F} = \{(X_{12}, \bar{X}_{12})\}$

Gomory-Hu Example

The traffic t_{pq} (zero entries not shown)

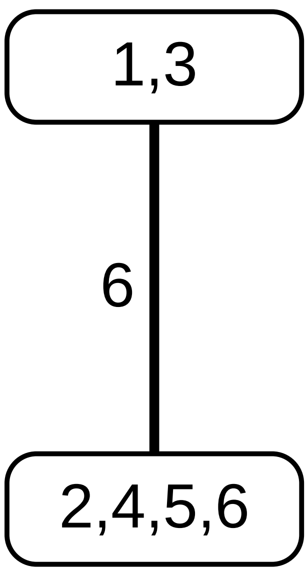


Some values $t(X_{12}, \bar{X}_{12})$ and the min for $X_{12} = \{1, 3\}$

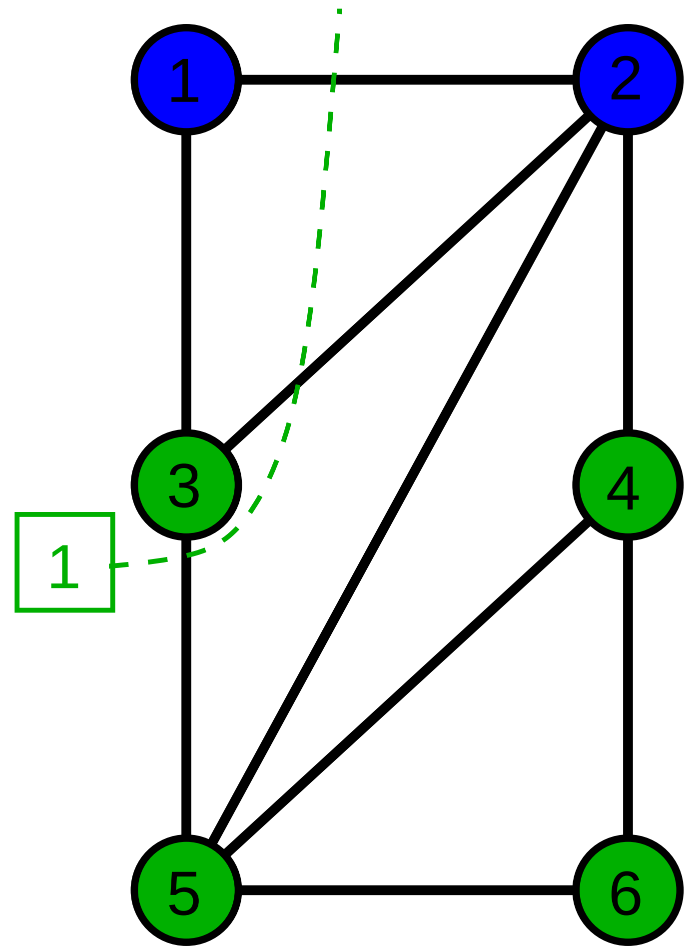


Gomory-Hu Example

Current partitioning of G along with $t(X, \bar{X})$

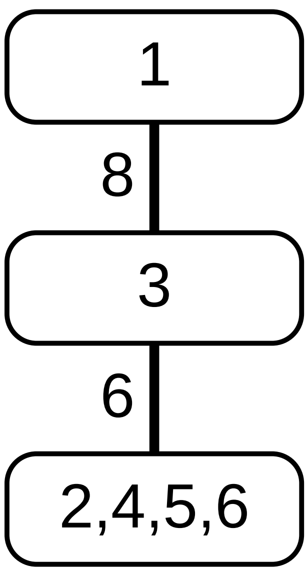


Step 1: $(p,q) = (1,2)$ and $X_{12} = \{1,3\}$

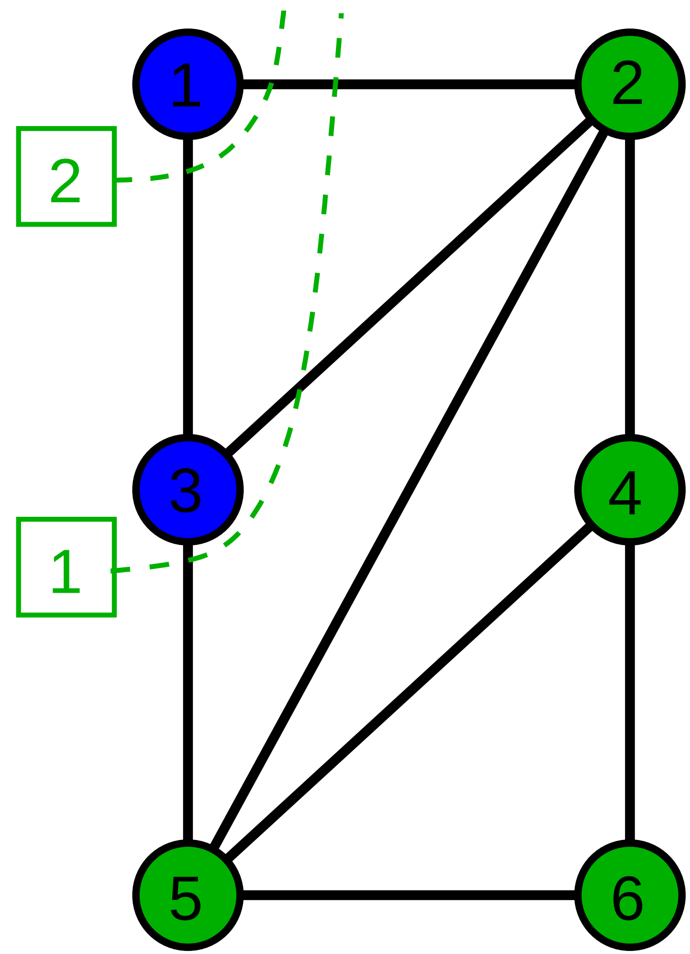


Gomory-Hu Example

Current partitioning of G along with $t(X,\bar{X})$

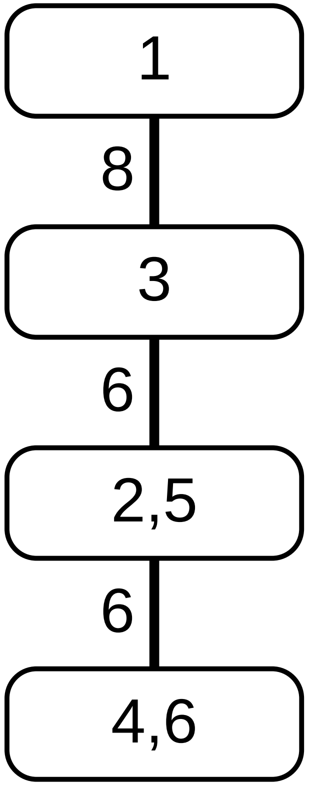


Step 2: $(p,q) = (1,3)$ and $X_{13} = \{1\}$

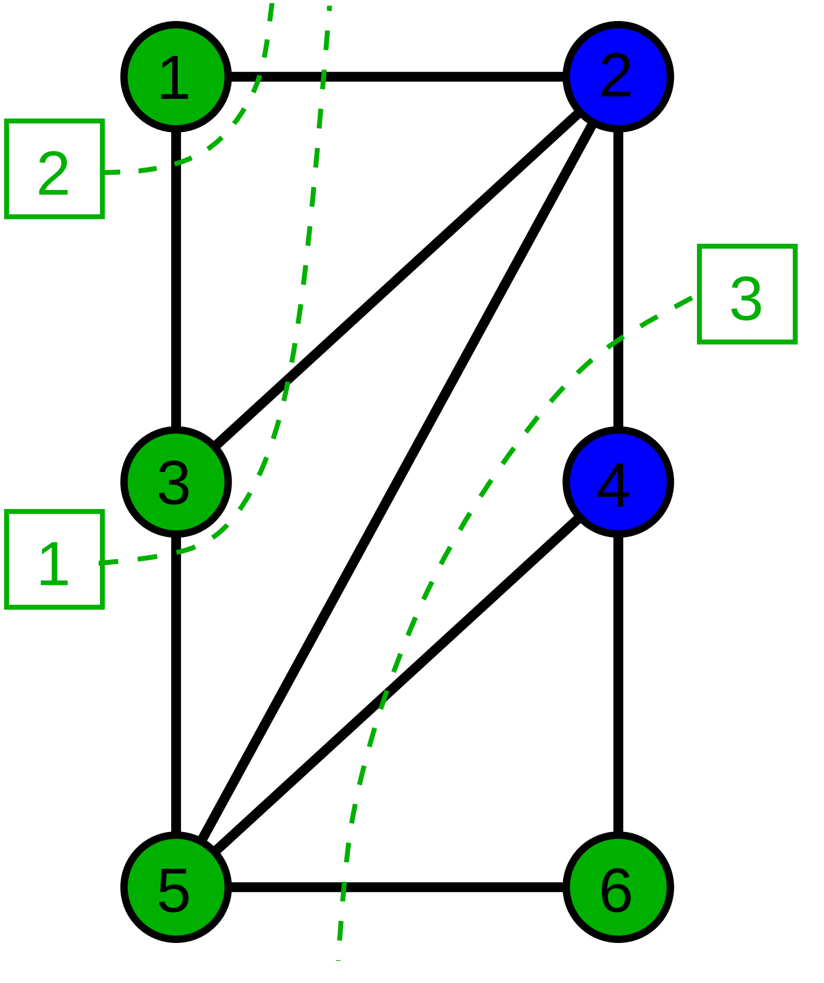


Gomory-Hu Example

Current partitioning of G along with $t(X, \bar{X})$



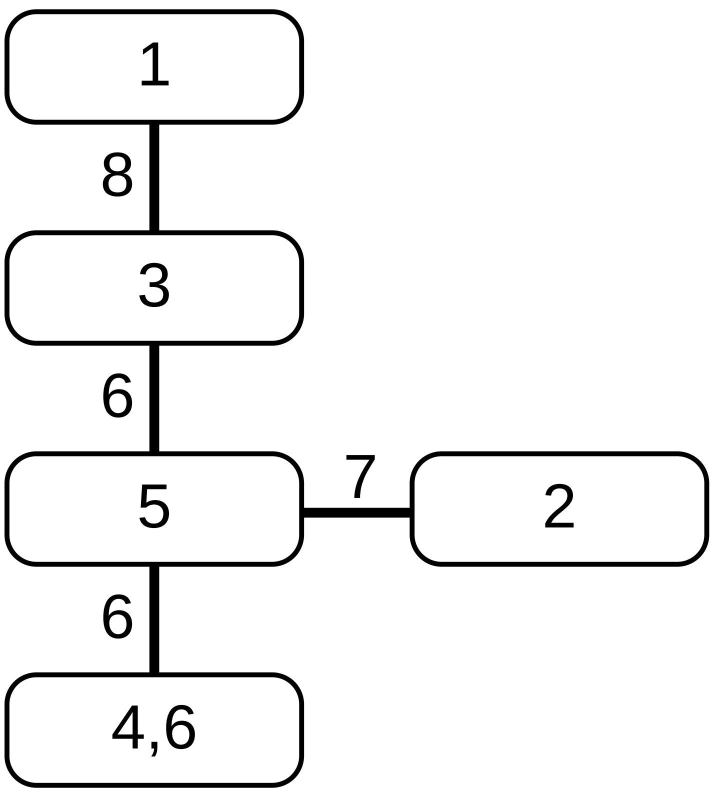
Step 3: $(p,q) = (2,4)$ and $X_{24} = \{1,2,3,5\}$

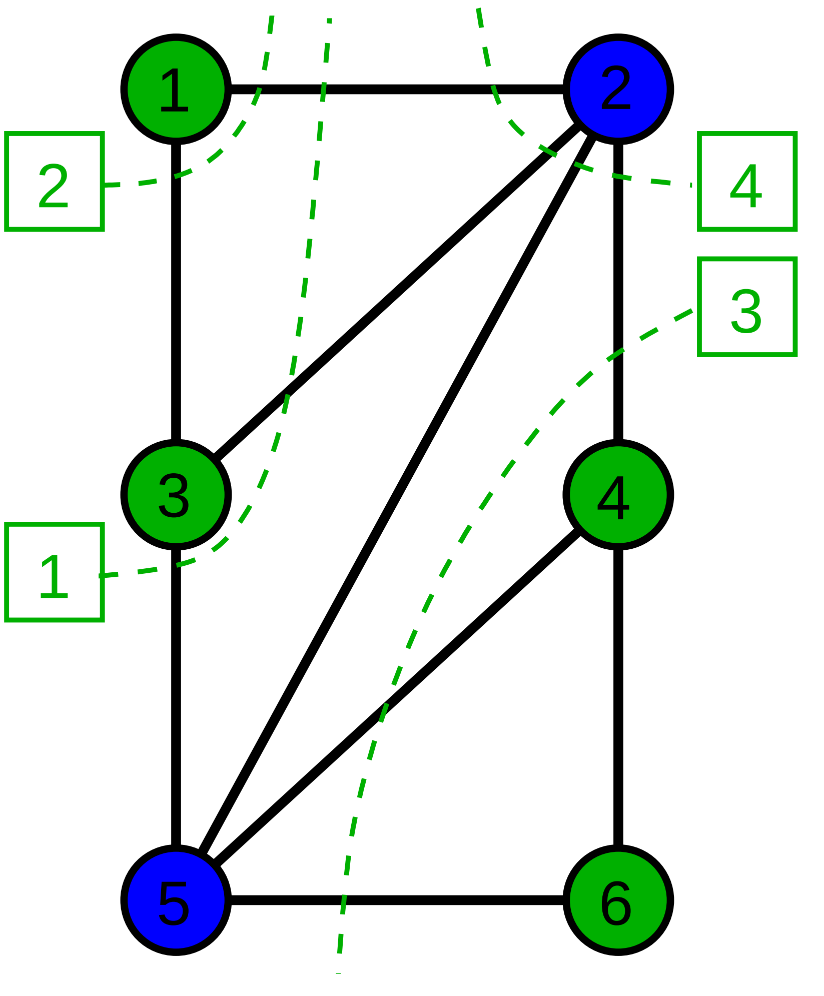


Gomory-Hu Example

Current partitioning of G along with $t(X,\bar{X})$

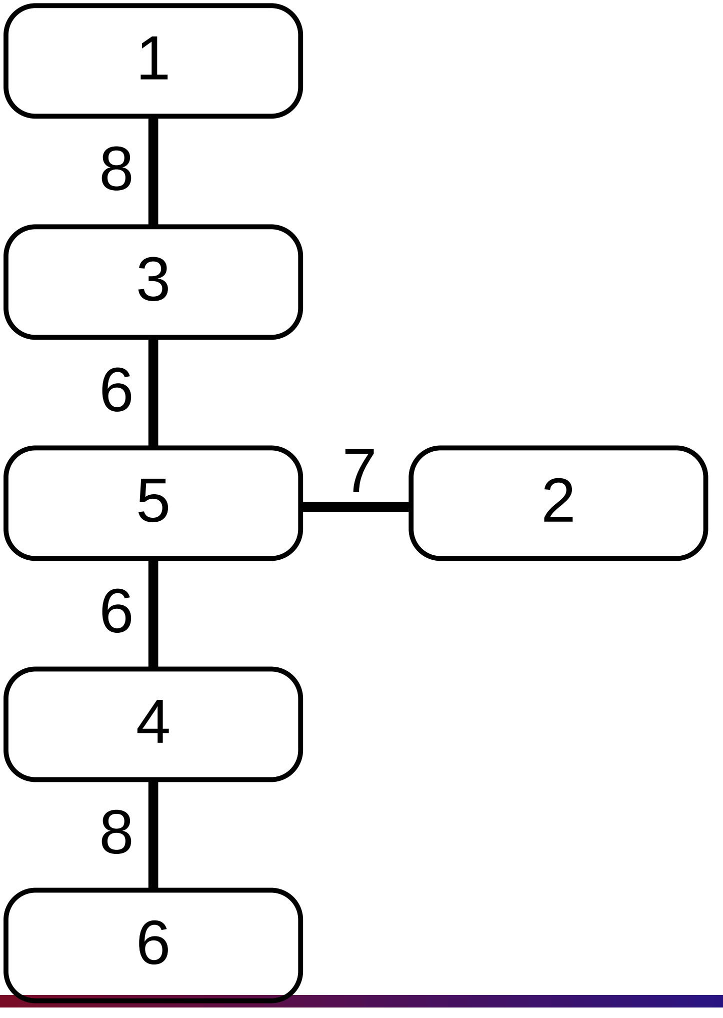
Step 4: $(p,q) = (2,5)$ and $X_{25} = \{1,3,4,5,6\}$



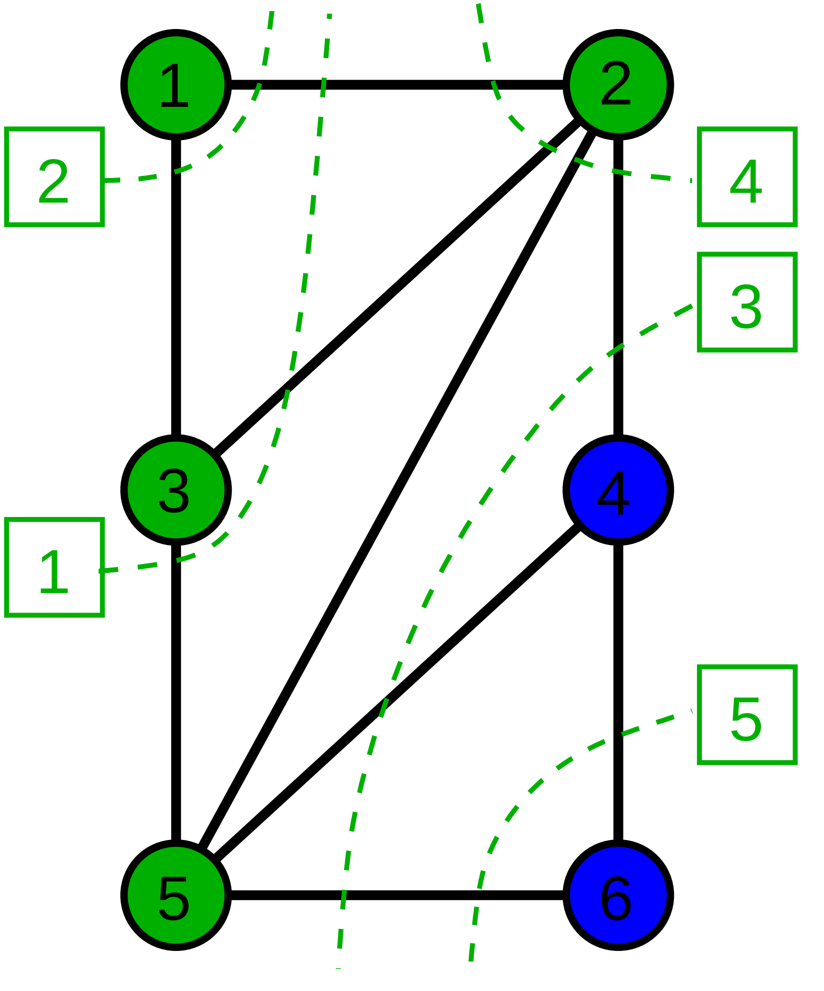


Gomory-Hu Example

Current partitioning of G along with $t(X,\bar{X})$

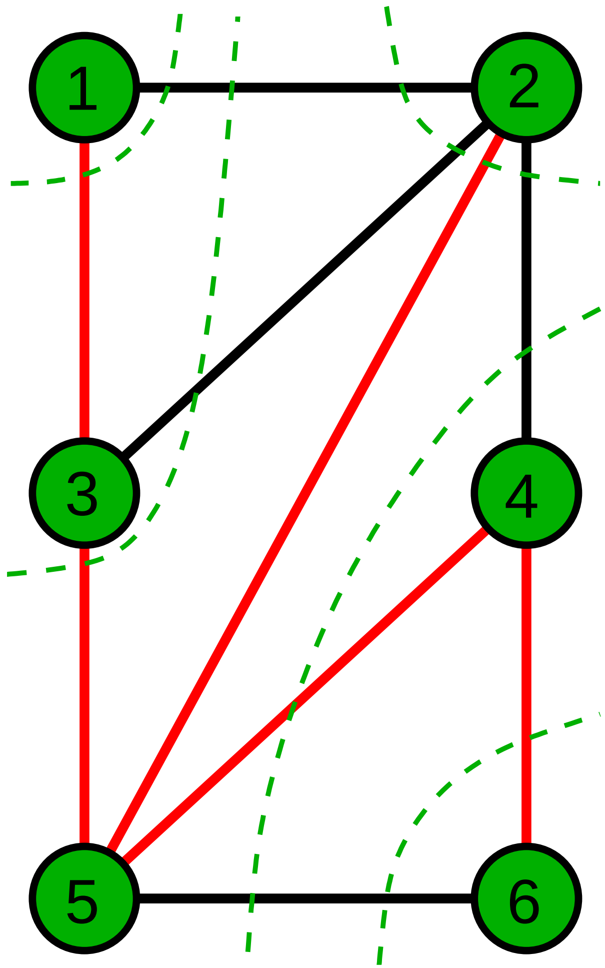


Step 5: $(p,q) = (4,6)$ and $X_{46} = \{1,2,3,4,5\}$

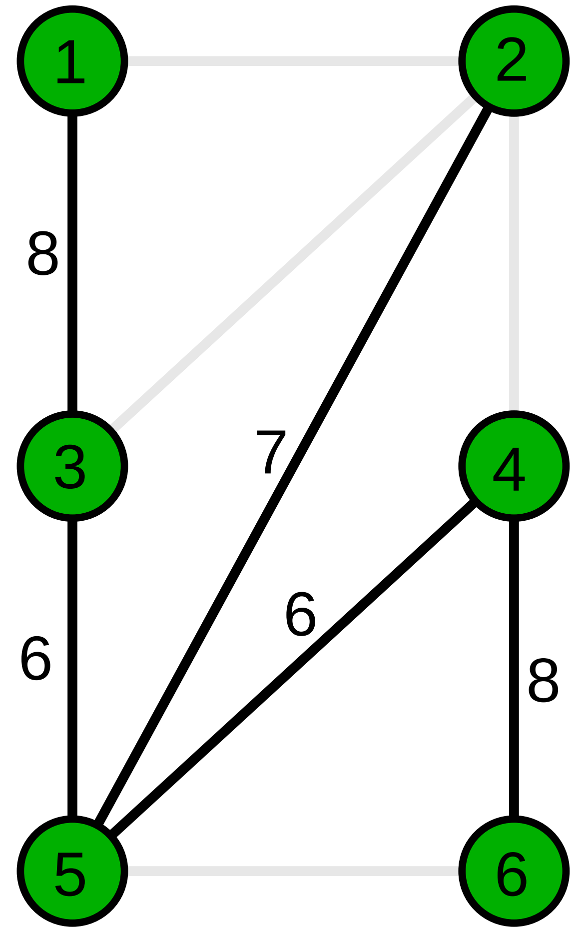


Gomory-Hu Example

Choose links in exactly one cutset



Final result for T also showing $f_e = t(X, \bar{X})$



Gomory-Hu Example: summary

SUMMARY:

(a) $\underline{1,2}$ $\mathcal{F}_1 = \{(X,\bar{X})\}$ where $X = \{1,3\}; \bar{X} = \{2,3,5,6\}$, $t(X,\bar{X}) = 6$.

(b) $\underline{1,3}$ $\mathcal{F}_2 = \mathcal{F}_1 \cup \{(X,\bar{X})\}$ where $X = \{1\}; \bar{X} = \{3,2,4,5,6\}$, $t(X,\bar{X}) = 8$.

(c) $\underline{2,4}$ $\mathcal{F}_3 = \mathcal{F}_2 \cup \{(X,\bar{X})\}$ where has $X = \{4,6\}; \bar{X} = \{1,2,3,5\}$, $t(X,\bar{X}) = 6$.

(d) $\underline{2,5}$ $\mathcal{F}_4 = \mathcal{F}_3 \cup \{(X,\bar{X})\}$ where has $X = \{2\}; \bar{X} = \{1,3,4,5,6\}$, $t(X,\bar{X}) = 7$.

(e) $\underline{4,6}$ $\mathcal{F}_5 = \mathcal{F}_4 \cup \{(X,\bar{X})\}$ where has $X = \{6\}; \bar{X} = \{1,2,3,4,5\}$, $t(X,\bar{X}) = 8$.

Total cost: $\sum_{e \in T} f_e = 8+6+7+6+8 = 36$

Gomory-Hu Complexity

- We have to find $|N| - 1$ non-crossing cutsets, i.e. there will be $O(|N|)$ steps
- each step requires minimization over all allowed cutsets
  - how do we find non-crossing cutsets?
  - Ford-Fulkerson Maximum Flow Labelling Algorithm (see Math Programming III)
    - max flow — min cut theorem gives the minimum cutset
  - but how do we test non-crossing (in reasonable complexity)?
    - non-trivial
- Gusfield's Algorithm is an alternative

Gusfield's Algorithm

How can we get away from needing non-crossing cutsets?

Gusfield's Algorithm

Objective: given a graph $G(N,E)$, and predicted traffic t_{pq}, find a min hop tree.

Principle: start with a star, and break off bits that can become substars

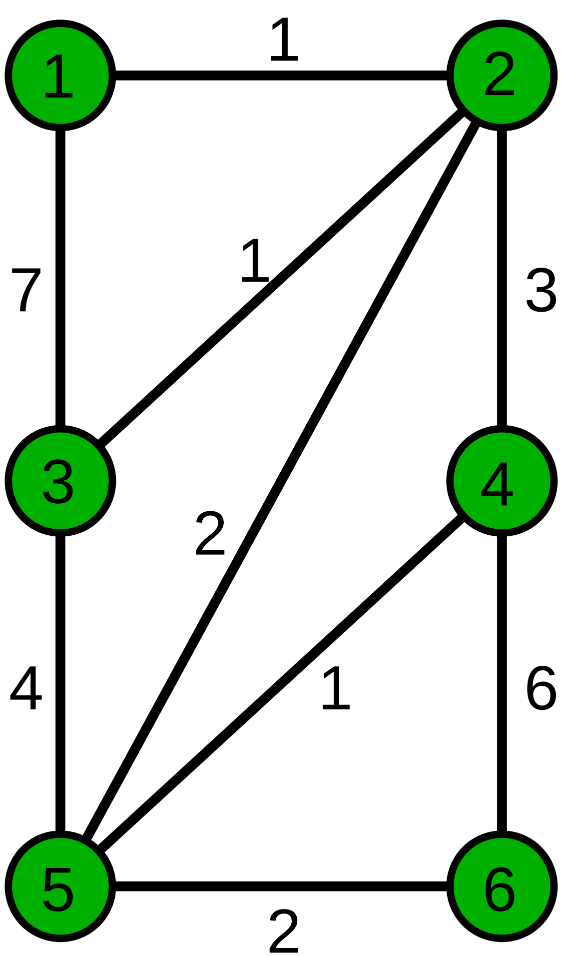
- WLOG we can choose initial hub to be node 1
- another greedy algorithm
  - for each node, test to see if the network is cheaper if we break it off the main hub
  - however, it does reach the optimum
- we have a spanning tree at each step
  - use $r(k)$ to denote the **parent** of node k
  - because its a spanning tree, this is a unique representation

Gusfield's Algorithm

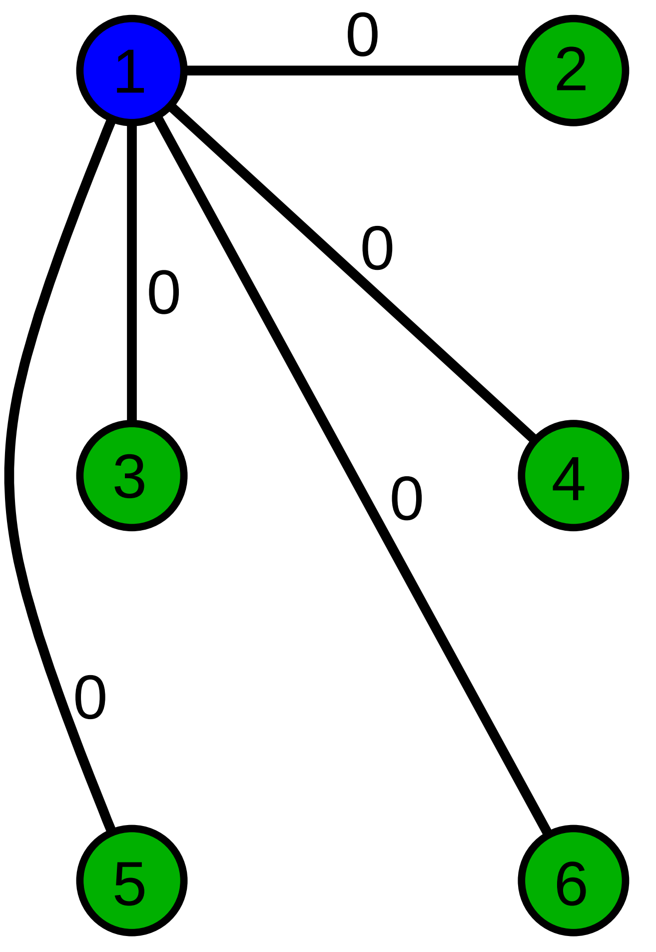
- **Initialize:** start with the tree T being star, with node 1 as the hub, i.e. $r(k) = 1$ for $k = 2, 3, \ldots, n$
  - also for each link $(k, r(k))$ assign $v_{k1} = 0$
- **For:** $k = 2, 3, \ldots, n$
  1. amoung all cutsets separating k from its parent $r(k)$, determine the cutset with smallest value of $t(X, \bar{X})$, i.e. choose $(X, \bar{X})$ that solves
  $$\min\{t(X, \bar{X}) | k \in X, r(k) \in \bar{X}\}$$
  2. assign $v_e = t(X, \bar{X})$ to the link $e = (k, r(k)) \in T$
  3. **For:** $i = 2, 3, \ldots, n$
     - if $i \in X$ and $i \neq k$ and $(i, r(k)) \in T$
     - then replace link $(i, r(k))$ in T by (i, k) with value equal to the old link, e.g. $v_{ik} = v_{i,r(k)}$

Gusfield's Algorithm Example

The traffic t_{pq}
(zero entries not shown)



Initial star network
also showing v_{k1}



Gusfield's Algorithm Example

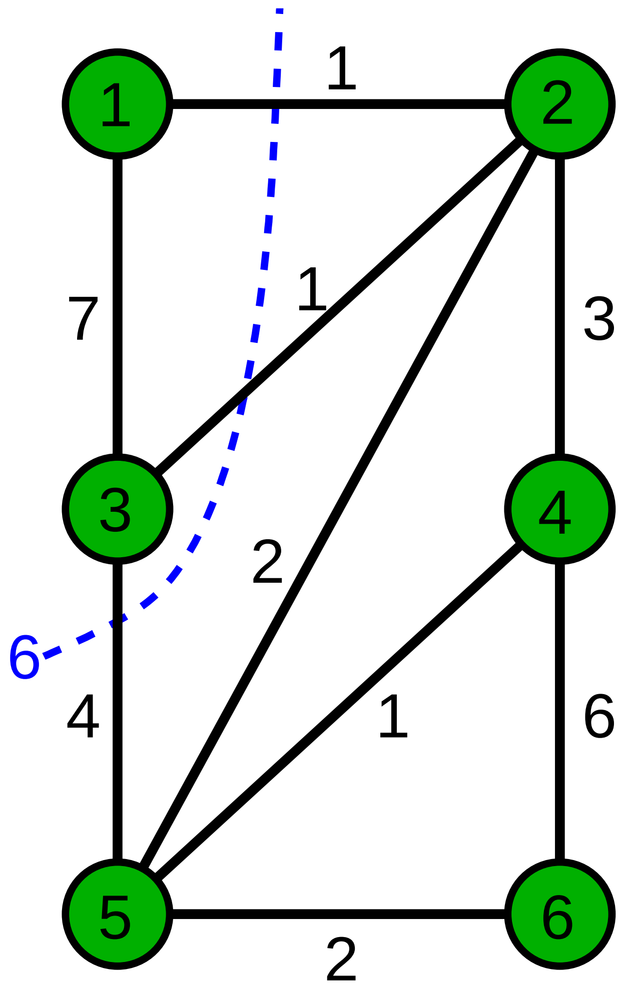
Iteration 1: $k = 2$

- $r(k) = 1$, so we find minimal cutset that separates node 2 from node 1
- this is just the same as step 1 of G-H, and so the minimal cutset is $X = \{2,4,5,6\}$ and $\bar{X} = \{1,3\}$
- $\nu_{2,1} = t(X,\bar{X}) = 6$
- for $i \in X = \{2,4,5,6\}$, we get $i \neq k$ and $i \in X$ for $i = 4,5,6$
- for $i = 4,5,6$, check whether $e = (i, r(k)) \in T$, e.g.

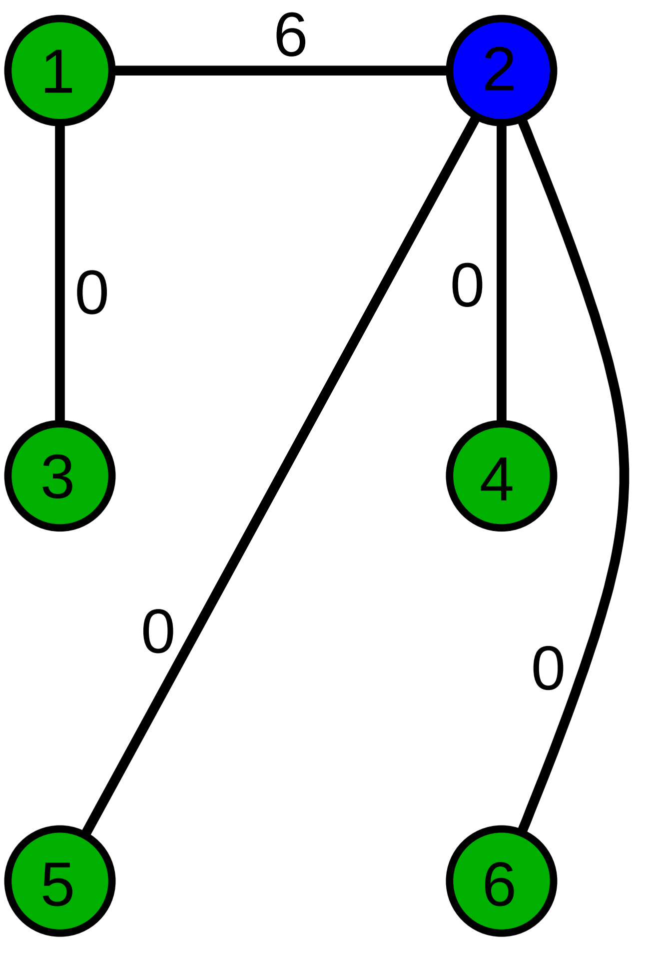
$$\begin{array}{ll} (4,1) \in T, & \text{so set } r(4) = k = 2 \\ (5,1) \in T, & \text{so set } r(5) = k = 2 \\ (6,1) \in T, & \text{so set } r(6) = k = 2 \end{array}$$

Gusfield's Algorithm Example

The traffic t_{pq}
and the first cutset



Iteration 1: $k = 2$
also showing values



Gusfield's Algorithm Example

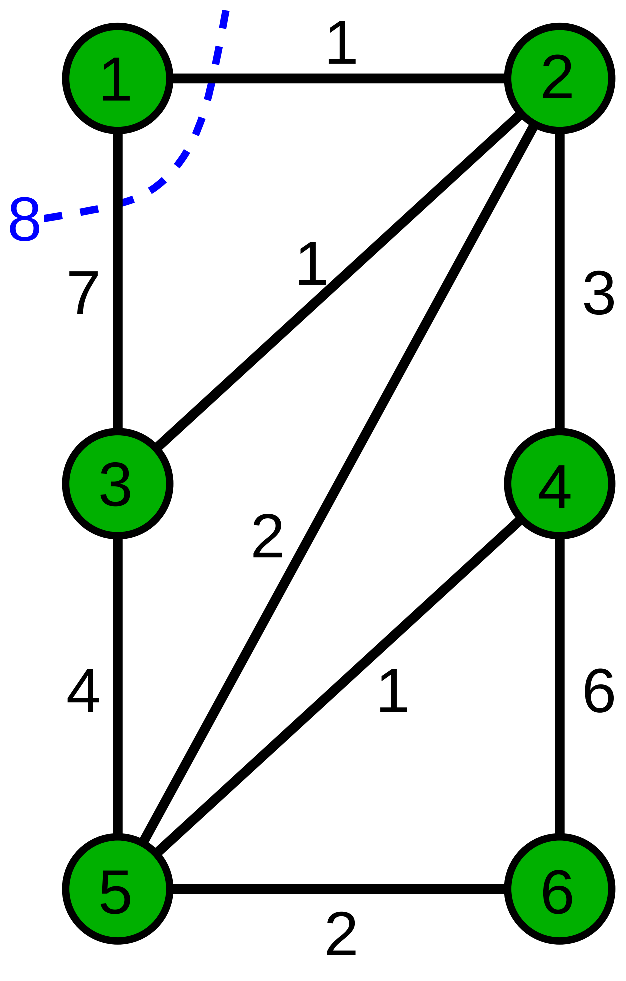
Iteration 2: $k = 3$

- $r(k) = 1$, so we find minimal cutset that separates node 3 from node 1
- this is just the same as step 2 of G-H, and so the minimal cutset is $X = \{2,3,4,5,6\}$ and $\bar{X} = \{1\}$
- $\nu_{3,1} = t(X, \bar{X}) = 8$
- for $i \in X = \{2,3,4,5,6\}$, we get $i \neq k$ and $i \in X$ for $i = 2,4,5,6$
- for $i = 2,4,5,6$, check whether $e = (i, r(k)) \in T$, e.g.

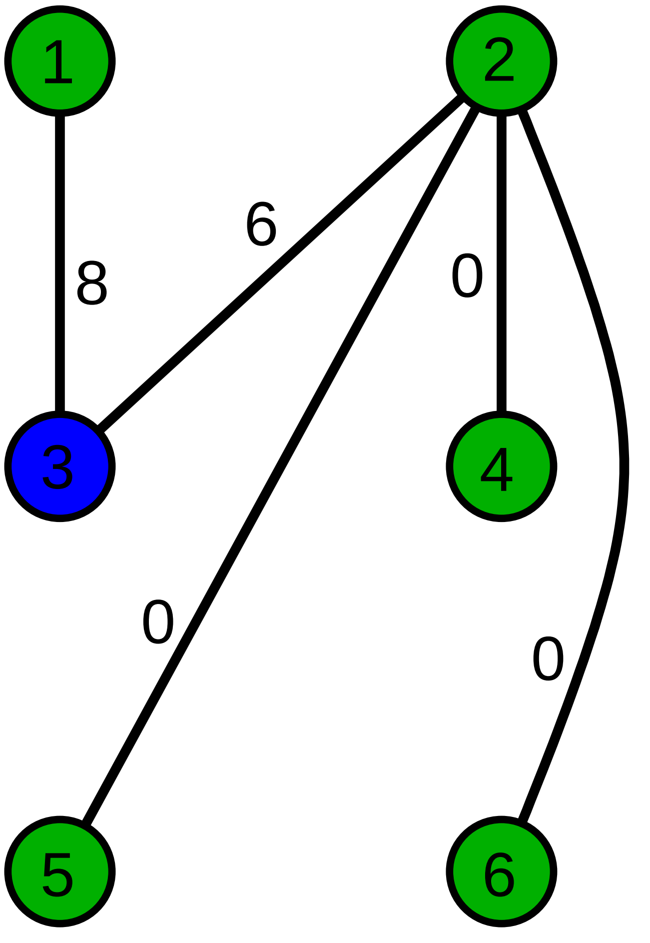
$$(2,1) \in T, \quad \text{so set } r(2) = k = 3$$
$$(4,1) \notin T, \quad \text{so take no action}$$
$$(5,1) \notin T, \quad \text{so take no action}$$
$$(6,1) \notin T, \quad \text{so take no action}$$

Gusfield's Algorithm Example

The traffic t_{pq}
and the second cutset



Iteration 2: $k = 3$
also showing values



Gusfield's Algorithm Example

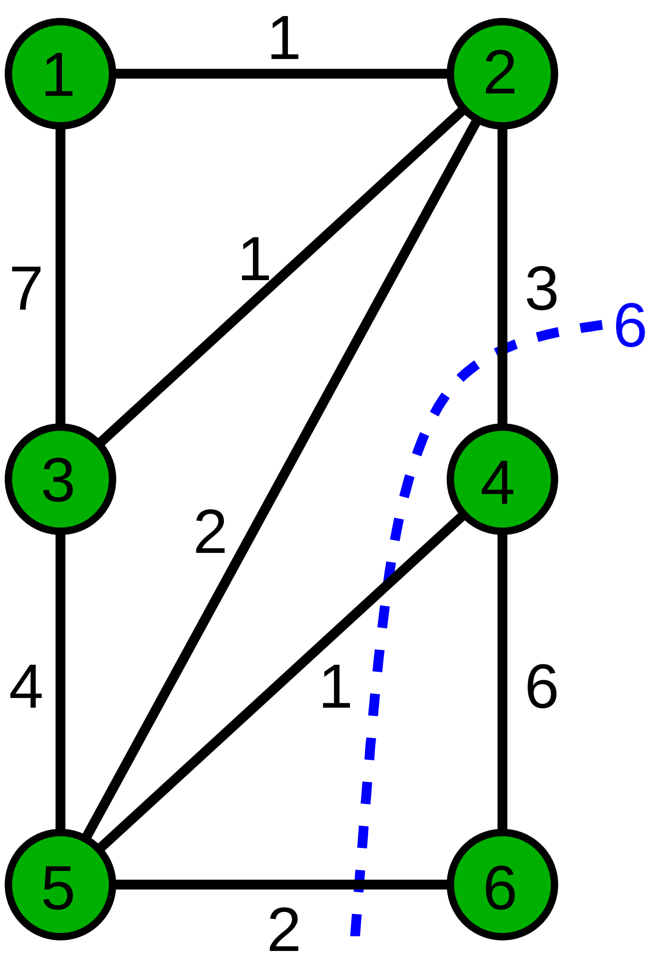
Iteration 3: $k = 4$

- $r(k) = 2$, so we find minimal cutset that separates node 4 from node 2
- minimal cutset is $X = \{4, 6\}$ and $\bar{X} = \{1, 2, 3, 5\}$
- $\nu_{4,2} = t(X, \bar{X}) = 6$
- for $i \in X = \{4, 6\}$, we get $i \neq k$ and $i \in X$ for $i = 6$
- for $i = 6$, check whether $e = (i, r(k)) \in T$, e.g.

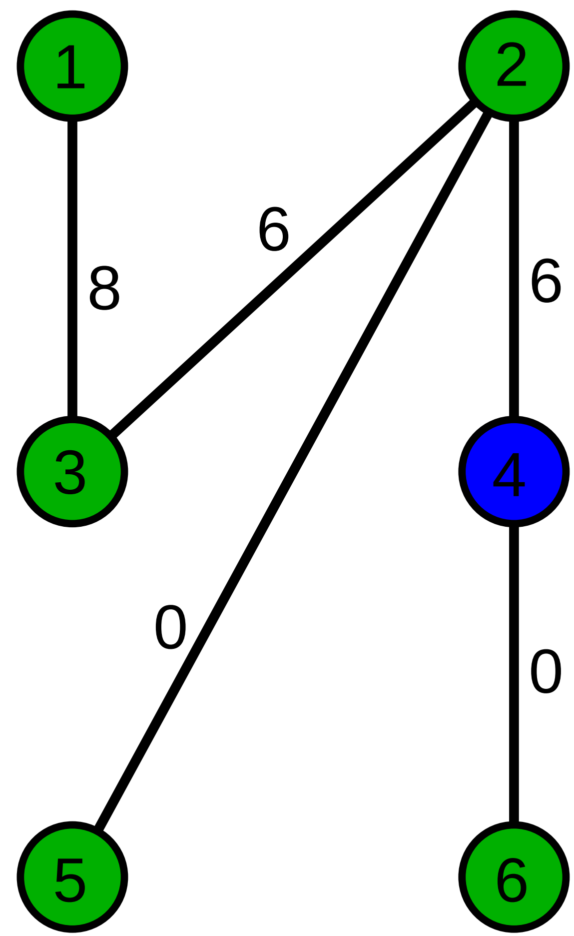
  $(6, 2) \in T,$ so set $r(6) = k = 4$

Gusfield's Algorithm Example

The traffic t_{pq}
and the third cutset



Iteration 3: $k = 4$
also showing values



Gusfield's Algorithm Example

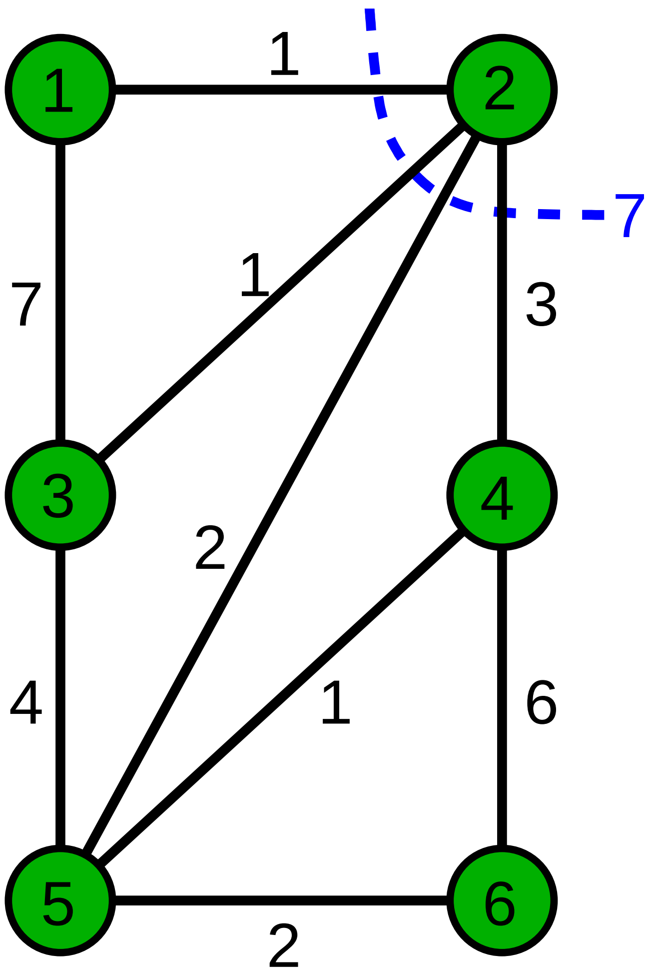
Iteration 4: $k = 5$

- $r(k) = 2$, so we find minimal cutset that separates node 5 from node 2
- minimal cutset is $X = \{1,3,4,5,6\}$ and $\bar{X} = \{2\}$
- $\nu_{5,2} = t(X, \bar{X}) = 7$
- for $i \in X = \{1,3,4,5,6\}$, we get $i \neq k$ and $i \in X$ for $i = 1,3,4,6$
- for $i = 1,3,4,6$, check whether $e = (i, r(k)) \in T$, e.g.

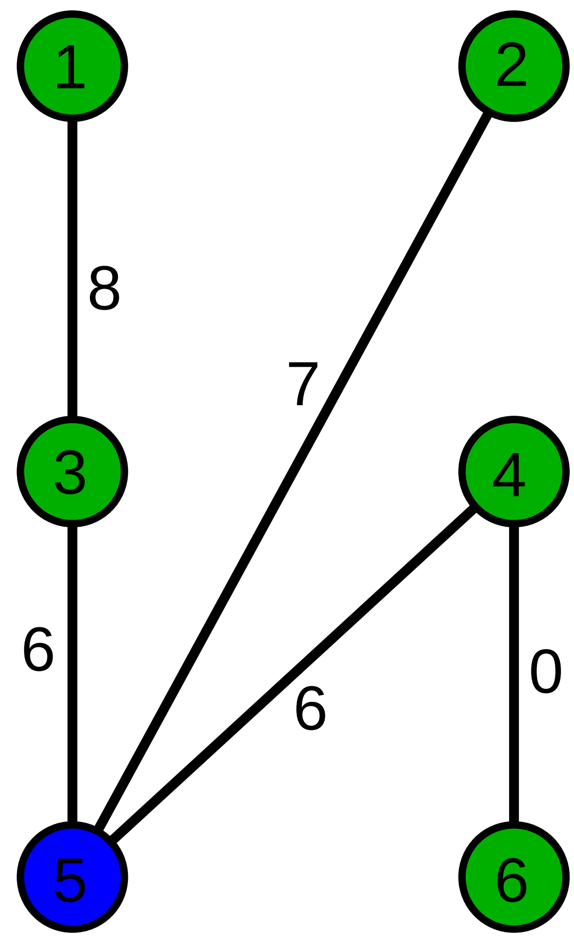
$$\begin{array}{ll} (1,2) \notin T, & \text{so no action} \\ (3,2) \in T, & \text{so set } r(3) = k = 5 \\ (4,2) \in T, & \text{so set } r(4) = k = 5 \\ (6,2) \notin T, & \text{so no action} \end{array}$$

Gusfield's Algorithm Example

The traffic t_{pq} and the forth cutset



Iteration 4: $k = 5$ also showing values



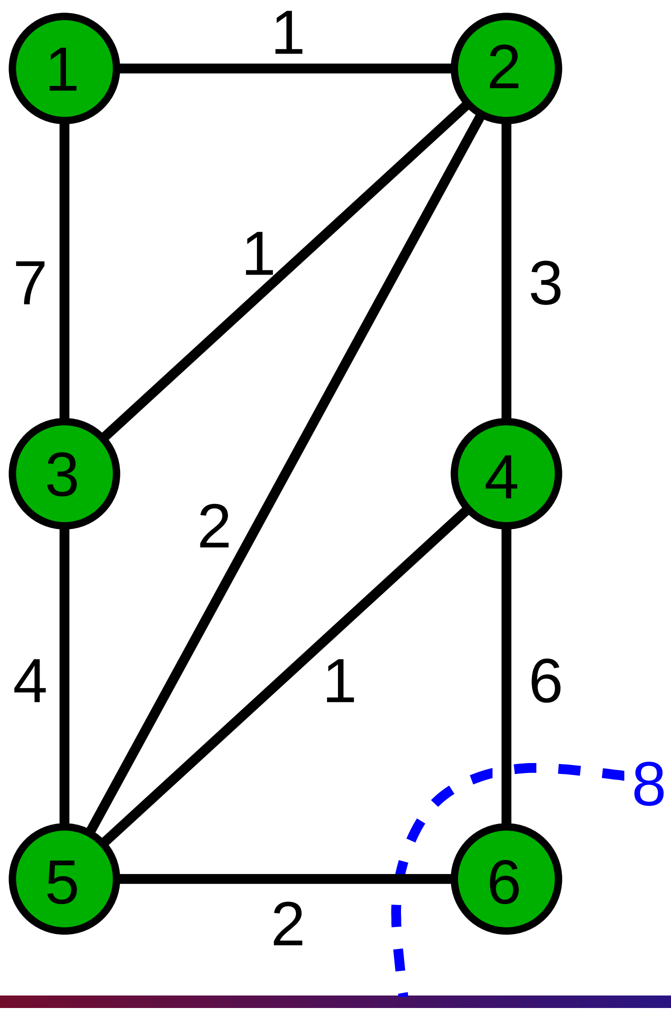
Gusfield's Algorithm Example

Iteration 5: $k = 6$

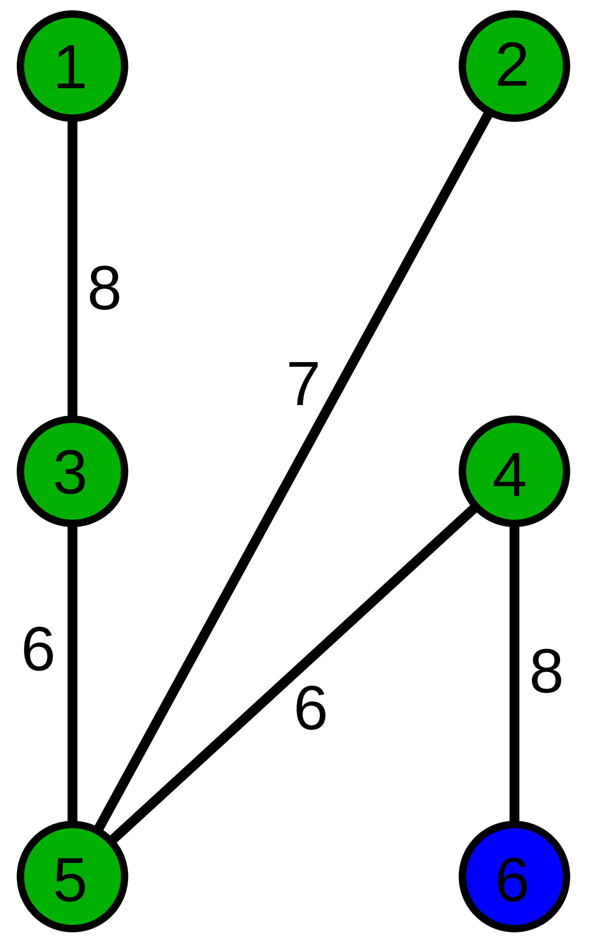
- $r(k) = 4$, so we find minimal cutset that separates node 6 from node 4
- minimal cutset is $X = \{6\}$ and $\bar{X} = \{1,2,3,4,5\}$
- $\nu_{6,4} = t(X, \bar{X}) = 8$
- for $i \in X = \{6\}$, we get $i \neq k$ and $i \in X$ for no values of i
- so there are no changes to the links

Gusfield's Algorithm Example

The traffic t_{pq}
and the fifth cutset

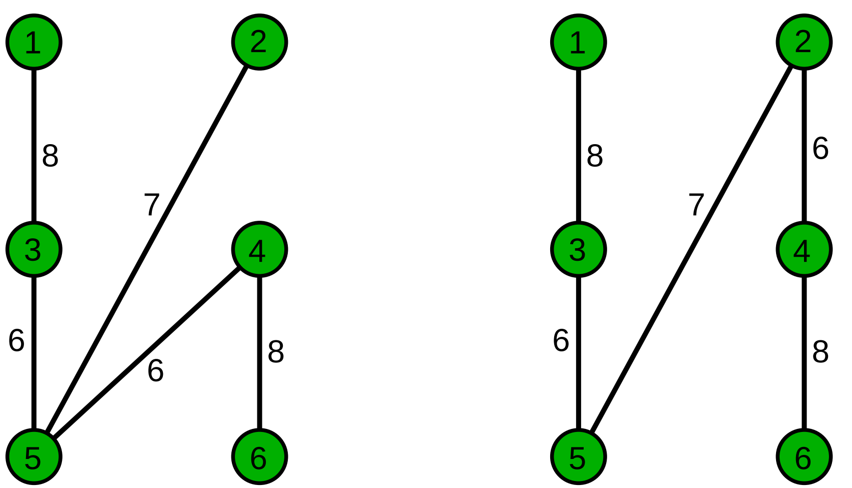


Iteration 5: $k = 6$
also showing values



Gusfield's Algorithm Example

- Final result is the same as for Gomory-Hu, which we expect
  - didn't need to look for non-crossing cutsets
- actually we could have used different cutsets
  - get a different tree
  - same cost though
  - non-unique solution to this particular problem



References