# Blocking arborescences

Gyula Pap

Egerváry Research Group (EGRES)
Eötvös University, Budapest, Hungary

Joint work with Attila Bernáth, and Tamás Király

"blocking"

=

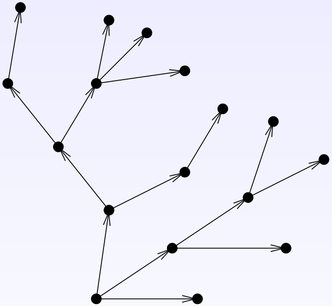
"deleting the minimum number/weights of edges/arcs/elements to destroy a certain property"

- Blocking spanning trees. (Global min cut.)
- Blocking min cost spanning trees. (Min cuts in bunch of minors.)
- Blocking $s$-$t$ paths. (Min cut.)
- Blocking min cost $s$-$t$ paths. (Min cut...)
- Blocking min cost bases of a matroid. (Depends on the matroid. We need minimum cuts in minors of the matroid.)
- Blocking perfect matchings in a bipartite graph. (NP-hard [Joret, Vetta, 2012])
- Blocking $r$-rooted arborescences - fixed root. (Min cut...)
- Blocking arborescences - arbitrary root. (Minimum double cut...)

Let $D = (V, A)$ be a digraph.

### Definition ($r$-rooted arborescence)

$T \subseteq A$ is a **(spanning) arborescence rooted in** $r$ if it is a tree in the undirected sense and every node is reachable from $r$.



### Definition ($k$-arborescence)

$T \subseteq A$ is a $k$-**arborescence** if it can be partitioned into the union of $k$ arborescences. $T \subseteq A$ is a $k$-**arborescence rooted in** $r$ if it can be partitioned into the union of $k$ arborescences rooted in $r$.

I will speak about:

- Blocking $r$-rooted arborescences. (fixed root)
- Blocking arborescences. (arbitrary root)
- Blocking $r$-rooted $k$-arborescences. (fixed root)
- Blocking $k$-arborescences. (arbitrary root)
- Blocking min cost $r$-rooted arborescences. (fixed root)
- Blocking min cost $k$-arborescences. (arbitrary root)
- Blocking $k$-arborescences under a matroid constraint. (arbitrary root)

## Problem A (Blocking $r$-rooted arborescences)

INPUT:

- Digraph $D = (V, A)$, root $r \in V$.

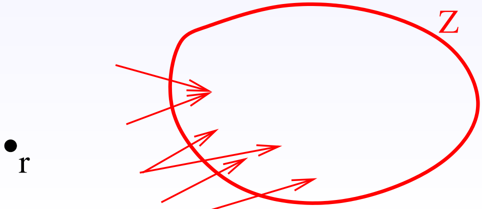
OUTPUT:

- Min $|B|$ s.t. $B \subseteq A$ a blocking of $r$-rooted arborescences.

## Solution — Problem A

Find a **minimum** $r$**-cut** $Z$, and let $B := \delta^{in}(Z)$, that is,

$$\min\{\delta^{in}(Z) : \emptyset \neq Z \subsetneq V - r\}$$



## Problem B (blocking of arborescences – root not fixed)

INPUT:

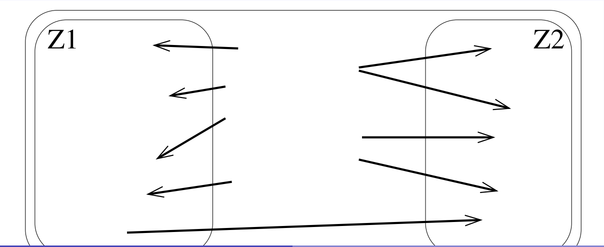
- Digraph $D = (V, A)$.

OUTPUT:

- Min $|B|$ s.t. $B \subseteq A$ a blocking of arborescences.

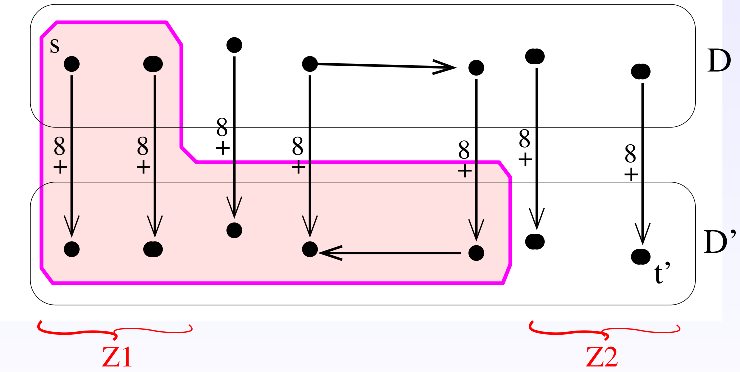
Find a **"minimum double cut"**, and let $B := \delta^{in}(Z_1) \cup \delta^{in}(Z_2)$, that is,

$$\mu(D) := \min\{\delta^{in}(Z_1) + \delta^{in}(Z_2) : Z_1 \cap Z_2 = \emptyset \neq Z_1, Z_2 \subsetneq V\}$$



Find a **"minimum double cut"**, that is,

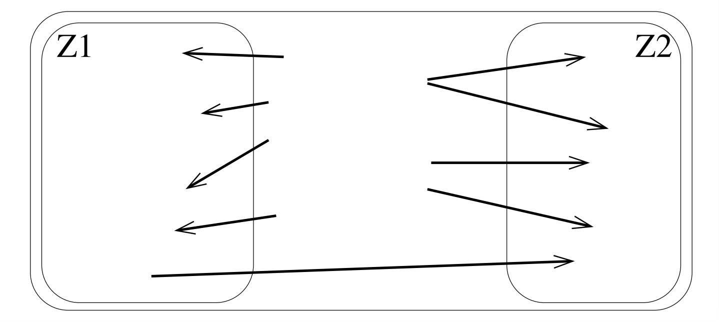
$$\mu(D) := \min\{\delta^{in}(Z_1) + \delta^{in}(Z_2) : Z_1 \cap Z_2 = \emptyset \neq Z_1, Z_2 \subsetneq V\}.$$




(Repeat for all $\binom{n}{2}$ choices of $s, t \in V$.)

Find a **"minimum double cut"**, that is,

$$\mu(D) := \min\{\delta^{in}(Z_1) + \delta^{in}(Z_2) : Z_1 \cap Z_2 = \emptyset \neq Z_1, Z_2 \subsetneq V\}$$




(Remark: already the dominant of double cuts has facets with large coefficients.)

## Problem C (blocking $r$-rooted $k$-arborescences)

INPUT:

- Digraph $D = (V, A)$, $r \in V$, $k \geq 1$.

OUTPUT:

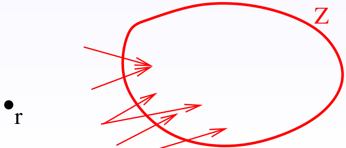
- Min $|B|$ s.t. $B \subseteq A$ a blocking of $r$-rooted $k$-arborescences.

## Solution — Problem C — by Edmonds' Arborescences theorem

Find a **minimum** $r$-**cut** $Z$, that is,

$$\min\{\delta^{in}(Z) : \emptyset \neq Z \subsetneq V - r\}$$

and make $B :=$ "$\delta^{in}(Z)$ minus $k-1$ arcs", or the emptyset.



## Problem D (blocking $k$-arborescences.)

INPUT:

- Digraph $D = (V, A)$, $k \geq 1$.

OUTPUT:

- Min $|B|$ s.t. $B \subseteq A$ a blocking of $k$-arborescences.

## Frank, 1978 - Existence of a $k$-arborescence

A $k$-arborescence exists if and only if the following inequality holds for every subpartition $\mathcal{Z}$ of $V$:

$$\sum_{Z \in \mathcal{Z}} \delta^{in}(Z) \geq k(|\mathcal{Z}| - 1).$$

## Solution — Problem D

$$k - \max\{\sum_{Z \in \mathcal{Z}} (k - \delta^{in}(Z)) : \mathcal{Z} \text{ a subpartition of } V, |\mathcal{Z}| \geq 2\}$$

and make $B :=$ "$\bigcup_{Z \in \mathcal{Z}} \delta^{in}(Z)$ minus $k(|\mathcal{Z}| - 1) - 1$ arcs", or the emptyset.

Problem D (blocking $k$-arborescences) reduces to the following formula:

$$\max\{\sum_{Z\in\mathcal{Z}}(k-\delta^{in}(Z)) : \mathcal{Z} \text{ a subpartition of } V, |\mathcal{Z}|\geq 2\}$$

???

## Definition (Bárász, Becker, Frank, 2005)

Given a digraph $D = (V, A)$, a nonempty subset $X \subseteq V$ is called **in-solid** if $\delta^{in}(Y) > \delta^{in}(X)$ for every nonempty $Y \subsetneq X$.

## Theorem (Bárász, Becker, Frank, 2005)

*The family of in-solid sets satisfies the Helly property.*
*— Family of in-solid sets have a tree-representation.*
*— Polytime algorithm to construct the tree representation.*

(Originally, this result was used for a result on a Facility Location problem.)
(Bárász, Becker, Frank proved Helly property for solid sets.)

Problem D (blocking $k$-arborescences) reduces to the following formula:

$$\max\{\sum_{Z\in\mathcal{Z}}(k-\delta^{in}(Z)) : \mathcal{Z} \text{ a subpartition of } V, |\mathcal{Z}|\geq 2\}$$

$$=$$

$$\max\{\sum_{Z\in\mathcal{Z}}(k-\delta^{in}(Z)) : \mathcal{Z} \text{ in-solid subpartition of } V, |\mathcal{Z}|\geq 2\}$$

Let $V = V_1 \cup V_2 \cup V_3$ be a partition induced by edges $e_1, e_2$ edges in the tree representation (chosen arbitrarily).

$$\max\{\textstyle\sum_{Z \in \mathcal{Z}} (k - \delta^{in}(Z)) : \mathcal{Z} \text{ in-solid subpartition of } V, |\mathcal{Z}| \geq 2\}$$

$$=$$

$$\max_{e_1, e_2}\{\max\{(k - \delta^{in}(Z_1)) : Z_1 \subseteq V_1 \text{ in-solid } \}+$$

$$+\max\{(k - \delta^{in}(Z_2)) : Z_2 \subseteq V_2 \text{ in-solid } \}+$$

$$+\max\{\textstyle\sum_{Z \in \mathcal{Z}} (k - \delta^{in}(Z)) : \mathcal{Z} \text{ in-solid subpartition of } V_3\}\}$$

$$\max\{\sum_{Z\in\mathcal{Z}}(k-\delta^{in}(Z)) : \mathcal{Z} \text{ in-solid subpartition of } V, |\mathcal{Z}|\geq 2\}$$

$$=$$

$$\max_{e_1,e_2}\{\max\{(k-\delta^{in}(Z_1)) : Z_1\subseteq V_1 \text{ in-solid }\}+$$

$$+\max\{(k-\delta^{in}(Z_2)) : Z_2\subseteq V_2 \text{ in-solid }\}+$$

$$+\max\{\sum_{Z\in\mathcal{Z}}(k-\delta^{in}(Z)) : \mathcal{Z} \text{ in-solid subpartition of } V_3\}\}$$

$$=$$

$$\max_{e_1,e_2}\{\max\{(k-\delta^{in}(Z_1)) : Z_1\subseteq V_1\}+$$

$$+\max\{(k-\delta^{in}(Z_2)) : Z_2\subseteq V_2\}+$$

$$+\max\{\sum_{Z\in\mathcal{Z}}(k-\delta^{in}(Z)) : \mathcal{Z} \text{ subpartition of } V_3\}\}$$

**Theorem, Bernáth, P, '14**

Blocking $k$-arborescences is solvable in polynomial time. (Running time bounded by $O(n^3)$ minimum cut subroutines.)

### Problem E (blocking min cost $r$-rooted arborescences.)

INPUT:

- Digraph $D = (V, A)$, $c : A \to \mathbb{R}$, $r \in V$.

OUTPUT:

- Min $|B|$ s.t. $B \subseteq A$ a blocking of min cost $r$-arborescences.

- Problem raised by N. Kamiyama, who solved some special cases.

## Corollary (Fulkerson's Algorithm)

There is

- subset $A' \subseteq A$ of "tight arcs",
- laminar family $\mathcal{L} \subseteq 2^{V-r}$,

such that an $r$-rooted arborescence $T \subseteq A$ is optimal iff both of the following properties holds:

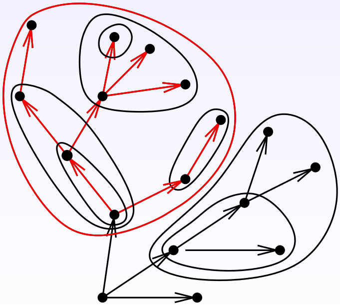
- $T \subseteq A'$, and
- $T \cap \delta^{in}(F) = 1$ for all $F \in \mathcal{L}$.

— i.e. an arborescence is optimal iff it is "$\mathcal{L}$-nice"

## Definition ($\mathcal{L}$-nice arborescence)

For a digraph $D = (V, A)$, and the laminar family $\mathcal{L} \subseteq 2^V$, an arborescence $T \subseteq A$ is called an *$\mathcal{L}$-nice arborescence* if it is rooted in a node $r \in V$, and for all $F \in \mathcal{L}$,

- $|\delta^{in}(F) \cap T| = 1$ for $r \notin F \in \mathcal{L}$, and
- $|\delta^{in}(F) \cap T| = 0$ for $r \in F \in \mathcal{L}$.



## Problem E' (blocking $\mathcal{L}$-nice arborescences)

INPUT:

- Digraph $D = (V, A)$, laminar family $\mathcal{L} \subseteq 2^V$.

OUTPUT:

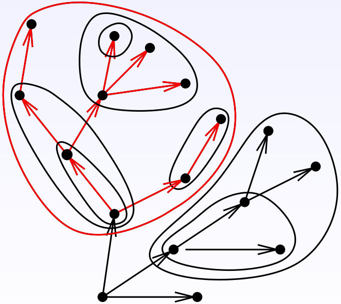
- Blocking $B \subseteq A$ of $\mathcal{L}$-nice arborescences

## Definition

For a laminar family $\mathcal{L}$ and some $F \in \mathcal{L}$, let $\mathcal{L}_F = \{X \in \mathcal{L} : X \subseteq F\}$.

## Observation

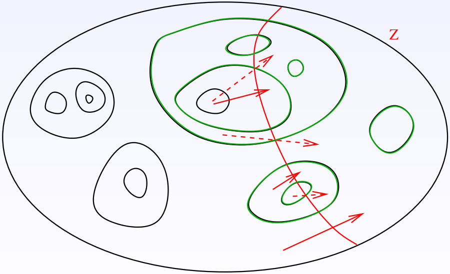
If $T$ is an $\mathcal{L}$-nice arborescence, then $T[F]$ is an $\mathcal{L}_F$-nice arborescence in $D[F]$.



## Definition

Consider a laminar family $\mathcal{L} \subseteq 2^V$ and a set $Z \subseteq V$. Define

- $\mathcal{L}_Z = \{X \in \mathcal{L} : X \cap Z \neq \emptyset\}$
- $M_Z = \delta_D^{in}(Z) - \bigcup_{F \in \mathcal{L}_Z} (\delta_D^{out}(F))$
- $f_{D,\mathcal{L}}(Z) = f(Z) := |M_Z|$.




## Claim

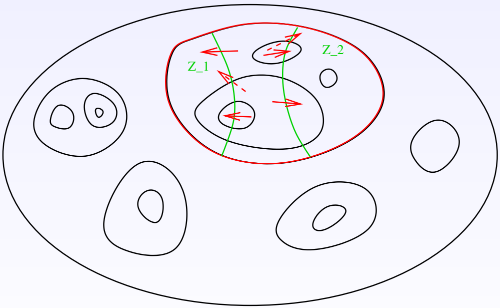
$M_Z$ blocks $\mathcal{L}$-nice arborescences rooted outside of $Z$.

Note:
$f$ is **NOT** submodular!

## Theorem ("Min-min formula")

*The minimum cover of all $\mathcal{L}$-nice arborescences is equal to*

$$\min\{f_{D[F],\mathcal{L}_F}(Z_1)+f_{D[F],\mathcal{L}_F}(Z_2) : F \in \mathcal{L}\cup\{V\}, \emptyset \neq Z_1, Z_2 \subseteq F, Z_1\cap Z_2 = \emptyset\}.$$




## Note

As $f$ is **NOT** submodular, submodular minimization does **NOT** help out.

## Lemma

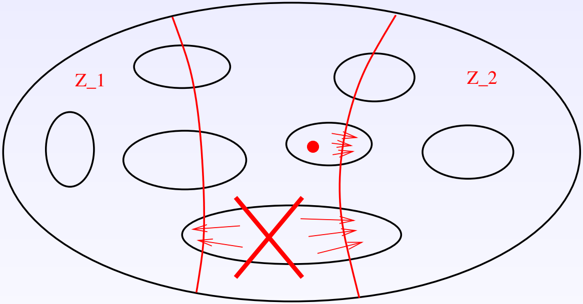
*In every digraph $D$ there is a node $r_D$ (called* **anchor node***) such that $\delta^{in}_D(X) \geq \frac{\mu(D)}{2}$ for every nonempty $X \subseteq V - r_D$.*

## Proof of Lemma.

Consider the subfamily $\{X : X$ is in-solid and $\delta^{in}_D(X) < \frac{\mu(G)}{2}\}$.
Theorem follows from Helly-property. □

# Motivation for anchor nodes

Consider $F_i$ inclusionwise minimal not considered yet. Find minimum double cut in $D[F_i]$, and anchor node $r_i \in F_i$.




$\Rightarrow$ no $F_i$ intersects both $Z_1, Z_2$, and

$\Rightarrow$ $r_i \notin Z_1 \cup Z_2$ and $F_i \cap (Z_1 \cup Z_2) \neq \emptyset$ holds for at most one $i$

## Algorithm for Problem E'

1. Let $D_0 = D$, $i = 1$ and $\mathcal{L}' = \mathcal{L}$.
2. Choose an inclusionwise minimal set $F \in \mathcal{L}'$
    1. $D_i$ is obtained from $D_{i-1}$ by relocating the tails of arcs leaving $F$ to $r_{D_{i-1}[F]}$.
    2. Let $i = i + 1$ and $\mathcal{L}' = \mathcal{L}' - F$.
3. Let $D' = D_{|\mathcal{L}|}$. For any $t \in \cup\mathcal{L}$ let $D_t$ be obtained by relocating the tails of every arc leaving a set $F \in \mathcal{L}$ with $t \in F$ to $t$.
4. The optimum is $\min\{\min\{\mu(D'[F]) : F \in \mathcal{L} \cup \{V\}\}, \min\{\mu(D_t[F] : t \in F \in \mathcal{L} \cup \{V\}\}\}$.

Naive implementation: $O(n^4)$ min cut computations

Problem E:

**Theorem, Bernáth, P, '13**

Blocking min cost arborescences is solvable in polynomial time. (Running time bounded by $O(n^3)$ minimum cut subroutines.)

Problem F:

**Theorem, Bernáth, T. Király, '15**

For **fixed** $k$, blocking min cost $k$-arborescences is solvable in polynomial time.

Problem G:

**Theorem, Bernáth, T. Király, '15**

Blocking min cost arborescences under a matroid constraint.

# Conclusions

- Better running time?!? ($O(n^3)$ min cuts seems too much...)
- Polyhedral approach?
- Polynomial time algorithm to block min cost $k$-arborescences ($k$ part of the input, not fixed)? (Only fixed $k$ is known to be polynomial time solvable.)

# Conclusions

- Better running time?!? ($O(n^3)$ min cuts seems too much...)
- Polyhedral approach?
- Polynomial time algorithm to block min cost $k$-arborescences ($k$ part of the input, not fixed)? (Only fixed $k$ is known to be polynomial time solvable.)

**Thank you for your attention!**