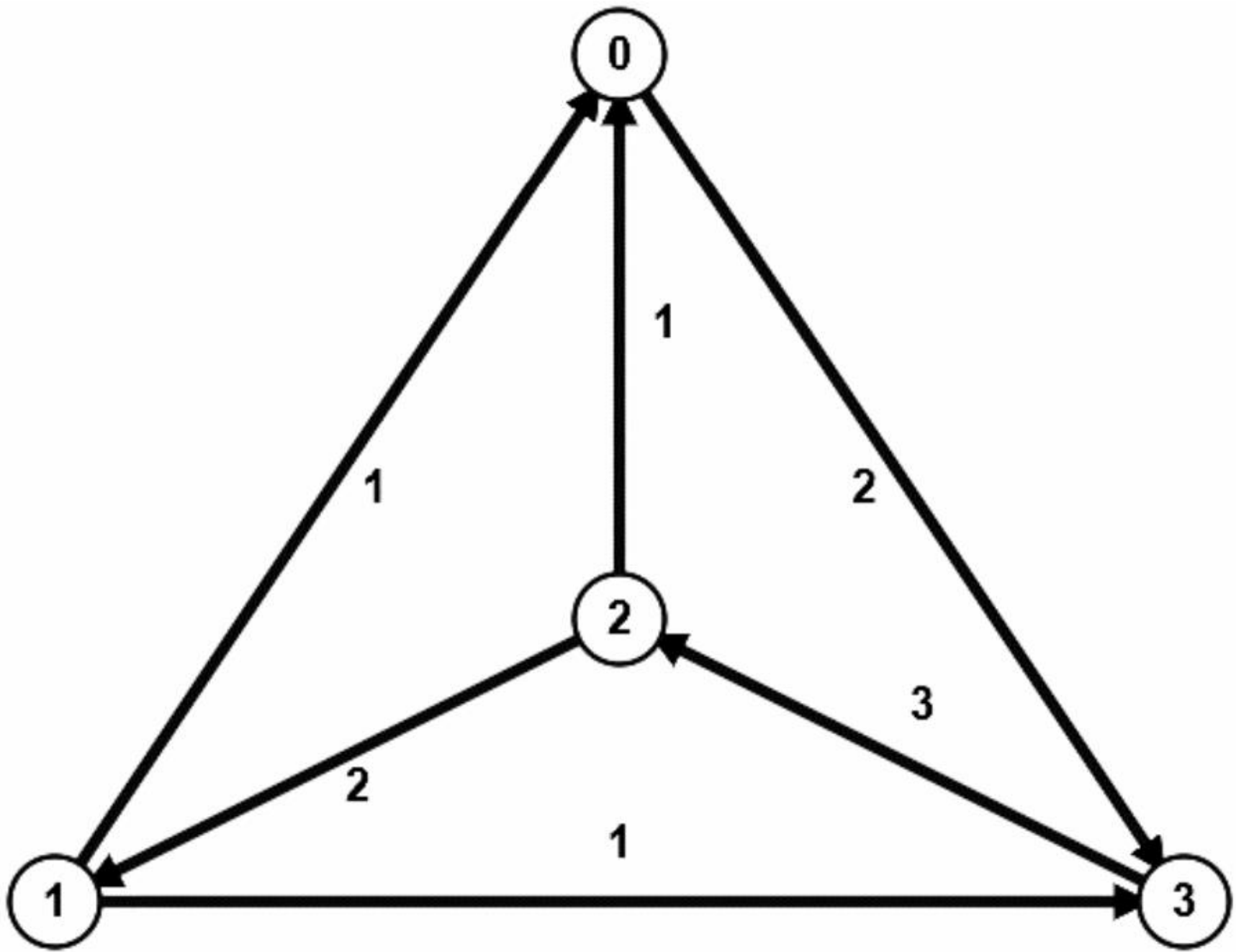


# Algoritmus na určenie cirkulárneho tokového čísla

- školiťel' : doc. RNDr. Robert Lukoťka, PhD.
- Samuel Vavrek

# Cirkulárny tok

- Cirkulárny nikde nulový  $r$ -tok - Priradenie reálnych čísel v intervale  $\langle 1, r-1 \rangle$  a orientácií hranám bezmostového grafu tak, že pre každý vrchol je súčet hodnôt vtekajúcich do vrcholu rovný súčtu hodnôt vytekajúcich z vrcholu.
- Cirkulárne tokové číslo – infimum množiny všetkých hodnôt  $r$ , pre ktoré má graf cirkulárny nikde nulový  $r$ -tok
- Pre toto infimum graf má cirkulárny nikde nulový  $r$ -tok a navyše je racionálne



# Problém hľadania cirkulárneho tokového čísla

- NP ťažký
- Súvisí s hranovým farbením
  - Kubický graf je hranovo 3-zafarbiteľný práve vtedy, keď má cirkulárne nikde nulový 4-tok.
- Dá sa sformulovať ako zmiešaný lineárny program
- Je známy algoritmus (Lukočka), ktorý pre kubický graf s cirkulárnym tokovým číslom najviac 5 nájde cirkulárne tokové číslo v čase  $O(2^{0.6|V(G)|})$ .

# Známe výsledky

- Každý graf má nikde nulový 6-tok, teda cirkulárne tokové číslo najviac 6 (Seymorova veta o 6-toku).
- Okrem racionálnosti je známe aj ohraničenie pre čitateľa a menovateľa
- Triedy grafov so známymi hodnotami cirkulárneho tokového čísla
  - Goldbergov snark  $G_{2k+1} - 4 + 1/(k+1)$
  - Isaacsov (flower) snark  $J_{2k+1} - 4 + 1/k$
  - Blanušove snarky – 4.5

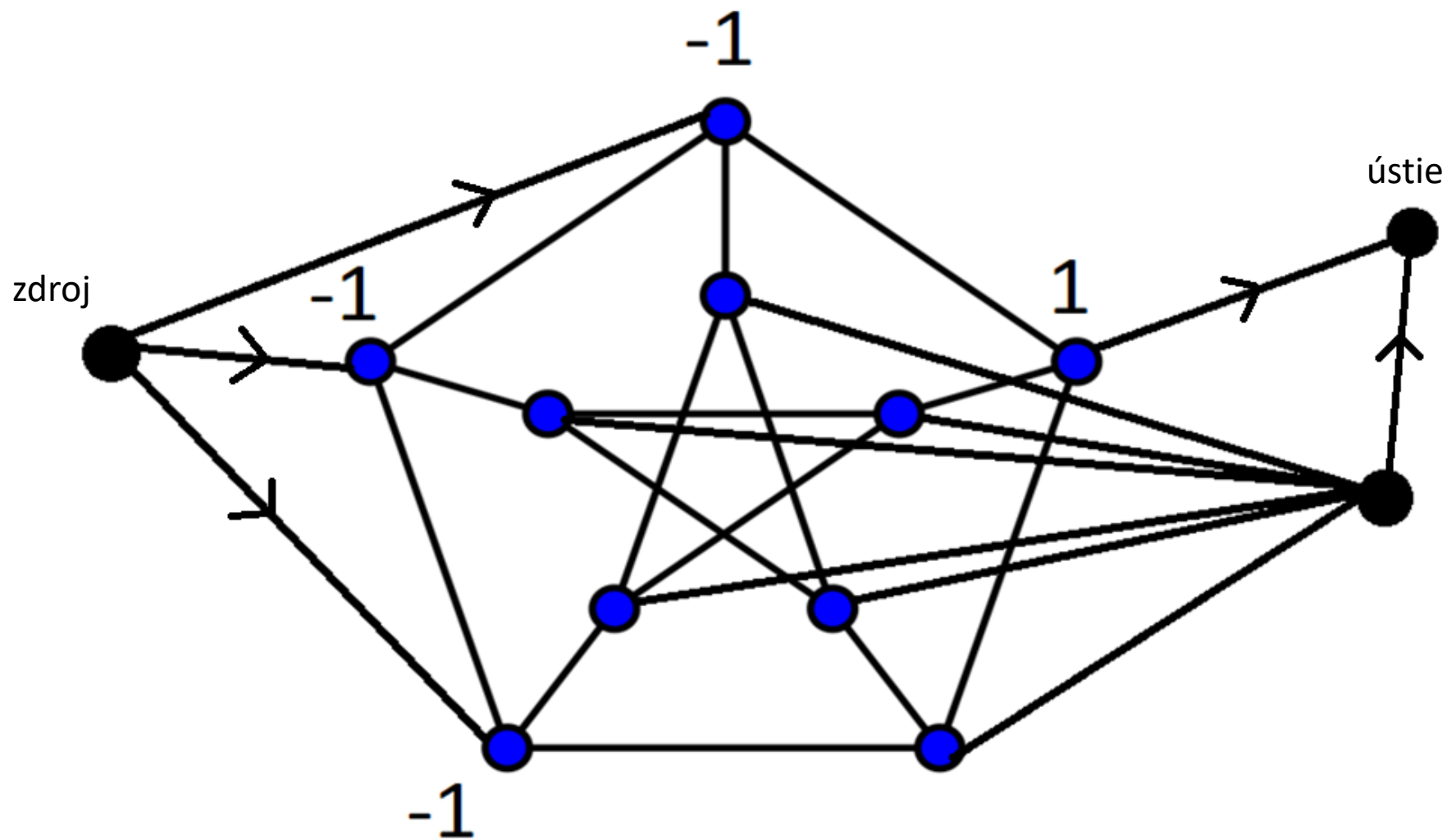
# Vyvážený tok

- Vrcholy majú priradené celočíselné hodnoty, vyvážené ohodnotenia, ktorých parita je rovná stupňu vrcholu
  - V absolútnej hodnote sú menšie ako stupeň vrcholu
  - Súčet všetkých vyvážených ohodnotení vrcholov grafu je rovný 0.
- Vyvážený tok je priradenie
  - vyvážených ohodnotení vrcholom
  - reálnych hodnôt v intervale  $\langle 2/r-1, 1-2/r \rangle$  a orientácií hranám grafu také, že pre každý vrchol je parita rozdielu vtekajúcich a vytekajúcich hodnôt rovná stupňu vrcholu.

# Branch and Bound algoritmus

- Vetvenie na vyvážených ohodnoteniach
- Pre výpočet dolného odhadu je použitý algoritmus pre maximálny tok
  - Google OR-Tools
- Ako najlepšie riešenie bude zvolená hodnota 6, keďže každý graf má nikde nulový 6-tok
- Heuristika
  - Ako ďalší sa vyberá vrchol s najväčším počtom susedov s pevným vyváženým ohodnotením
  - Ďalší sa vyberie vrchol ktorý má vypočítanú hodnotu najbližšie k celému číslu
- Algoritmus pracuje vnútorne s celými číslami

# Súvis cirkulárneho a maximálneho toku





# Výpočet dolného odhadu

Je známe že pre ak je  $r=p/q+1$  cirkulárne tokové číslo súvislého grafu, tak  $p+q-2 \leq E-V$

- Na základe tohto vzťahu (a ohraničení pre cirkulárne tokové číslo) vygenerujeme zoradené pole racionálnych čísel
- Skontrolujeme, či existuje cirkulárny  $r$ -tok pre najväčšie z nich, ak nie, potom hodnoty priradené vrcholom porušujú definíciu vyváženého ohodnotenia
- Inak hľadáme najmenšie racionálne číslo také, že graf má nikde nulový  $r$ -tok

# Sceločíslenie problému

- Ak  $r = p/q + 1$  potom
- $2/r - 1$
- $= 2q/(q+p) - 1$
- $= (q-p)/(q+p)$
- Interval  $\langle 2/r - 1, 1 - 2/r \rangle$  teda možno zapísať ako  $\langle \frac{q-p}{p+q}, \frac{p-q}{p+q} \rangle$
- Následne všetky hodnoty vynásobíme hodnotou  $p+q$
- Je známe, že pri hľadani cirkulárneho toku stačí uvažovať tokové hodnoty, ktoré sú celočíselným násobkom  $1/q$

# Výpočet bound lineárnym programom

- Lineárna relaxácia
- Testovanie
  - Musí byť rovný počet navštívených vrcholov
  - Dolný odhad pre dané (čiastočné) vyvážené ohodnotenie musí byť rovnaký
- Počíta s reálnymi číslami

# Výsledky

Goldbergove snarky				Isaacsove snarky				Blanušove snarky			
V	r	BnB	MIP	V	r	BnB	MIP	V	r	BnB	MIP
				44	$4+1/5$	5,189	3,333	58	4,5	2,696	10,957
40	$4+1/3$	0,816	3,044								
				60	$4+1/7$	43,230	44,731	82	4,5	29,832	168,073
56	$4+1/4$	12,215	37,217					90	4,5	58,435	188,772
72	$4+1/5$	171,802	333,435	68	$4+1/8$	183,501	194,02	98	4,5	135,863	187,636
								106	4,5	321,509	17m 51s

# Zhrnutie

- Implementovaný celočíselný Branch and Bound algoritmus na hľadanie cirkulárneho tokového čísla
  - Numerická stabilita
  - Presnosť
- Navrhnutý algoritmus je na vybraných grafoch rýchlejší, a to aj napriek tomu že v BnB strome navštívi viac vrcholov
- Ďalšou optimalizáciou by sme mohli dosiahnuť oveľa rýchlejší algoritmus ako zmiešaný lineárny program