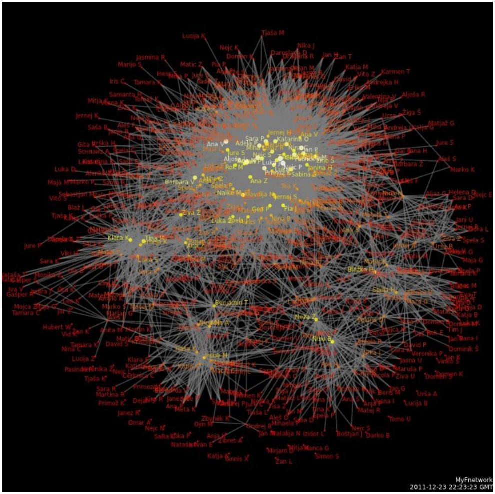
Gručavost

Uroš Kovač
Katarina Zadražnik
Maja Alif

4. junij 2012

Motivacija



Gručavost

Definicija

Gručevje $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$ je particija množice vozlišč V na neprazne podmnožice C_i, ki jim pravimo **gruče**.

Oznake:

- $E(C_i, C_j)$ …množica tistih povezav, ki imajo začetno vozlišče v gruči C_i in končno vozlišče v gruči C_j;
- $E(C_i)$ …okrajšava za $E(C_i, C_i)$;
- $E(\mathcal{C}) := \bigcup_{i=1}^{k} E(C_i)$ …množica povezav znotraj posameznih gruč;
- $\overline{E(\mathcal{C})} := E \setminus E(\mathcal{C})$ …množica povezav med gručami.

Merjenje kakovosti gručevij

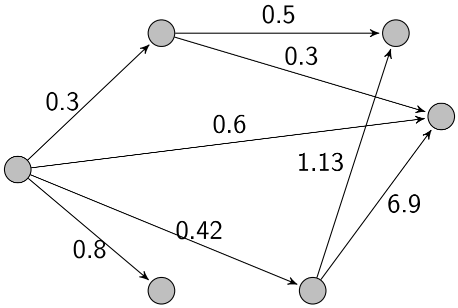
Kakovost gručevja danega grafa G izmerimo s pomočjo **strukturnega indeksa**.
$\mathcal{A}(G)$ naj bo množica vseh možnih gručevij grafa.
Strukturni indeks je sestavljen iz dveh neodvisnih funkcij:

- $f : \mathcal{A}(G) \to \mathbb{R}_0^+$ meri gostoto povezav znotraj posameznih gruč;
- $g : \mathcal{A}(G) \to \mathbb{R}_0^+$ g meri razpršenost povezav med gručami.

$$\text{indeks}(\mathcal{C}) := \frac{f(\mathcal{C}) + g(\mathcal{C})}{\max\left\{f(\mathcal{C}') + g(\mathcal{C}') \mid \mathcal{C}' \subseteq \mathcal{A}(G)\right\}}.$$

Model grafa

$G(V, E, \omega)$ ... enostaven, usmerjen, utežen graf brez zank, kjer je $\omega : E \to \mathbb{R}_0^+$ preslikava, ki vsaki povezavi priredi njeno utež.



Primeri strukturnih indeksov

Primeri strukturnih indeksov:

- pokritost;
- prevodnost;
- zmogljivost.

Pokritost

Definicija

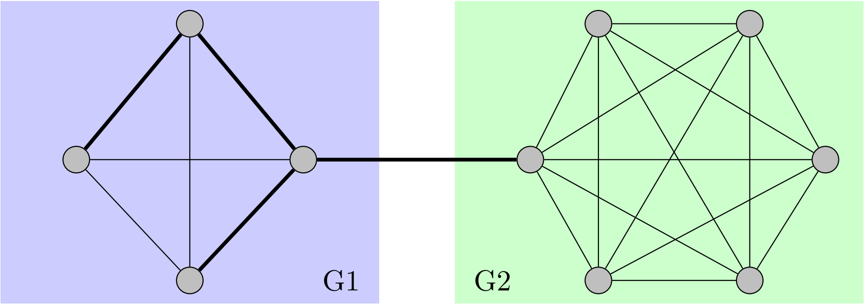
Pokritost $\gamma(\mathcal{C})$ je funkcija, ki meri skupno težo povezav znotraj gruč glede na težo vseh povezav.
Velja:

- $f(\mathcal{C}) = \omega(E(\mathcal{C}))$;
- $g \equiv 0$.

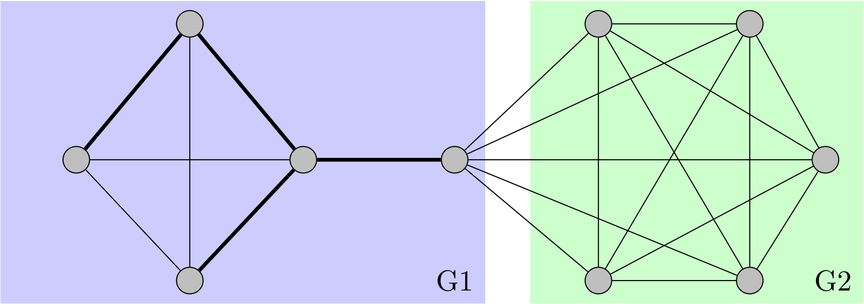
Pokritost izračunamo kot:

$$\gamma(\mathcal{C}) := \frac{\omega(E(\mathcal{C}))}{\omega(E)} = \frac{\sum_{e \in E(\mathcal{C})} \omega(e)}{\sum_{e \in E} \omega(e)}.$$

Zgled



Slika: Odebeljene povezave imajo utež enako 100. Ostale povezave imajo utež 1. Intuitivno gručevje ima pokritost $\gamma = 159/209 \approx 0.76$.



Slika: Odebeljene povezave imajo utež enako 100. Ostale povezave imajo utež 1. Optimalna vrednost, ki jo doseže pokritost pa je enaka $413/418 \approx 0.99$.

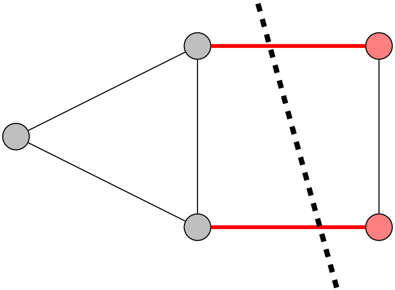
Trditev

Gručevje $\mathcal{C}$ *ima pokritost* $\gamma = 1$ *natanko tedaj, ko je množica povezav med gručami prazna ali pa imajo vse povezave v tej množici uteži enake* 0.

Pokritost in najmanjši prerez

Definicija

V teoriji grafov je **najmanjši prerez grafa** tisti prerez, pri katerem imamo najmanjše število prereznih povezav, če je graf neutežen. V primeru uteženega grafa pa je to prerez, pri katerem je vsota uteži na prereznih povezavah najmanjša.



Trditev

Naj bo $G(V,E,\omega)$ povezan graf, v katerem ima vsak možni prerez pozitivno vsoto uteži na prereznih povezavah. Netrivialna gručevja z optimalno pokritostjo so tista z dvema gručama, ki ju inducira najmanjši prerez.

Prevodnost

Definicija

Naj bo $\mathcal{C} = (C_1, C_2)$ prerez $(C_2 = V \setminus C_1)$. Potem sta **teža prevodnosti** $a(C_i)$ in **prevodnost** $\varphi(\mathcal{C})$ definirani kot:

$$a(C_i) := \sum_{(u,v) \in E(C_i, V)} \omega(u, v)$$

$$\varphi(\mathcal{C}) := \begin{cases} 1, & \text{če } C_1 \in \{\emptyset, V\}\,; \\ 0, & \text{če } C_1 \notin \{\emptyset, V\}\,, \omega(\overline{E(\mathcal{C})}) = 0; \\ \frac{\omega(\overline{E(\mathcal{C})})}{\min(a(C_1), a(C_2))}, & \text{sicer.} \end{cases}$$

Prevodnost grafa G pa je definirana kot:

$$\varphi(G) = \min_{\mathcal{C} \text{ prerez}} \varphi(\mathcal{C}).$$

Lema

Naj bo $G = (V, E, \omega)$ *neusmerjen graf s pozitivnimi utežmi na povezavah. Tedaj ima* G *največjo prevodnost, tj.* $\varphi(G) = 1$, *natanko tedaj, ko je povezan in ima največ 3 vozlišča ali pa je zvezda.*

- Vsi nepovezani grafi imajo prevodnost enako 0, saj zanje obstaja netrivialen prerez, ki ima na prereznih povezavah (teh ni) vsoto uteži 0. Torej ustrezajo drugi točki pri definiciji prevodnosti in res drži $\varphi = 0$.
- Za netrivialen prerez $\mathcal{C} = (C_1, V \setminus C_1)$ neusmerjenega grafa lahko težo prevodnosti $a(C_i)$ zapišemo drugače:

$$a(C_i) = \sum_{e \in E(C_i, V)} \omega(e) = \omega(E(C_i)) + \omega(\overline{E(\mathcal{C})}.$$

Tedaj se tretja točka pri definiciji prevodnosti prepiše v

$$\frac{\omega(\overline{E(\mathcal{C})}}{\min\Big(a(C_1), a(C_2)\Big)} = \frac{\omega(\overline{E(\mathcal{C})}}{\omega(\overline{E(\mathcal{C})}) + \min\Big(\omega(E(C_1)), \omega(E(V \setminus C_1))\Big)}.$$

Kako prepričati računalnik, da nam naredi gručevje?

Napišemo mu algoritem.

Osnovne tehnike, ki jih uporabljamo pri reševanju problema gručavosti, lahko razdelimo v tri razrede:

- požrešne metode;
- premične metode;
- splošni optimizacijski (aproksimacijski) pristopi, ki nam dajo približno rešitev.

Skupna ideja: *Problem delimo toliko časa, da dobimo obliko, v kateri problem znamo rešiti, nato pa ga razširimo nazaj do prvotnega problema.*

Korak delitve problema običajno sestoji iz dveh delov:

- Identifikacija podstruktur grafa.
- Modifikacija trenutnega grafa G. Običajne transformacije, ki jih izvedemo, so:
  - dodajanje povezav;
  - odstranjevanje povezav;
  - zamenjava korena v podgrafu, tj. predstavitev podgrafa z novim korenom.

Požrešna metoda

Oznake:

- L ... trenutna rešitev;
- $c(L)$ ... cena trenutne rešitve L;
- $N_g(L)$ ... množica vseh možnih rešitev, ki jih lahko dosežemo z izboljšavo rešitve L.

Algoritem (Požrešna metoda)

Naj bo L_0 dopustna rešitev.
$i \leftarrow 0$
while $\{L \mid L \in N_g(L_i), c(L) < c(L_i)\} \neq \emptyset$ **do**
  $L_{i+1} \leftarrow \operatorname{argmin}_{L \in N_g(L_i)} c(L)$
  $i \leftarrow i + 1$
end while
return L_i

Proces združevanja

Naj bodo dani:

- graf $G = (V, E, \omega)$ in začetna gručavost $\mathcal{C}_1$;
- globalna funkcija cene $c_{\text{globalna}} : \mathcal{A}(G) \to \mathbb{R}_0^+$
  ali
- lokalna funkcija cene $c_{\text{lokalna}} : \mathcal{P}(V) \times \mathcal{P}(V) \to \mathbb{R}_0^+$.

Dve gruči v trenutnem gručevju $\mathcal{C}_i := \{C_1, \ldots, C_k\}$, kadar je to možno, združimo na naslednji način:

- **Globalna verzija**: Naj bo P množica vseh možnih rezultatov, ki jih dobimo, če združimo dve gruči iz trenutnega gručevja $\mathcal{C}_i$, tj.

$$P := \{\{C_1, \ldots, C_k\} \setminus \{C_\mu, C_\nu\} \cup \{C_\mu \cup C_\nu\} \mid \mu \neq \nu\}.$$

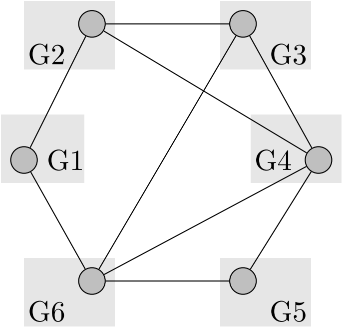
  Novo gručevje $\mathcal{C}_{i+1}$ je določeno z:

$$\mathcal{C}_{i+1} := \operatorname{argmin}_{\mathcal{C} \in P} c_{\text{globalna}}(\mathcal{C}).$$

- **Lokalna verzija**: Naj bosta μ in ν taka različna indeksa, da velja: c_{lokalna} ima en globalni minimum (C_μ, C_ν). Potem je novo gručevje $\mathcal{C}_{i+1}$ definirano z združitvijo C_ν in C_μ, tj.

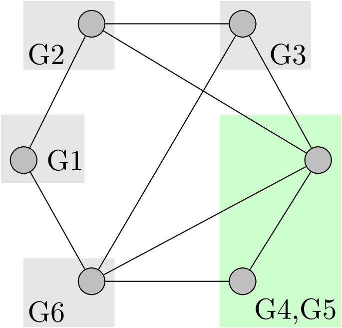
$$\mathcal{C}_{i+1} := \{C_1, \ldots, C_k\} \backslash \{C_\mu, C_\mu\} \cup \{C_\mu \cup C_\nu\}.$$

Zgled



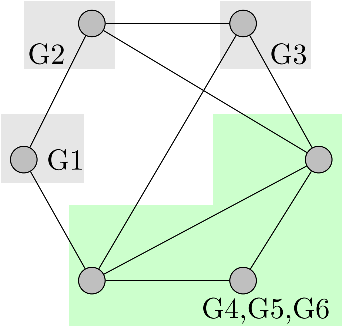
Slika: 1. korak

Zgled



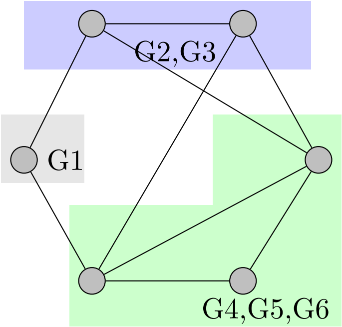
Slika: 2. korak

Zgled



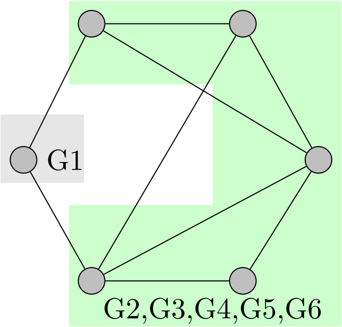
Slika: 3. korak

Zgled



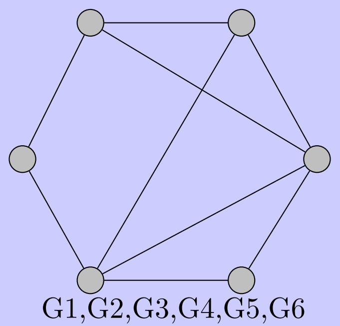
Slika: 4. korak

Zgled



Slika: 5. korak

Zgled



Slika: 6. korak

Proces delitve

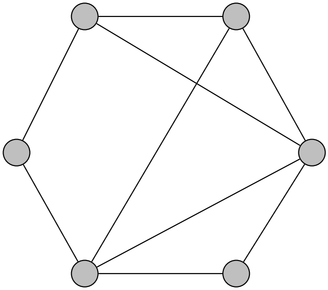
Naj bodo dani:

- graf $G = (V, E, \omega)$ in začetno gručevje $\mathcal{C}_1$;
- globalna ali semi-globalna ali lokalna ali semi-lokalna funkcija cene.

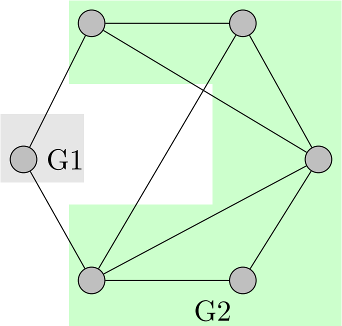
Definirajmo funkcije cene:

- *globalna:* funkcija cene $c_{\text{globalna}} : \mathcal{A}(G) \to \mathbb{R}_0^+$;
- *semi-globalna:* funkcija cene $c_{\text{globalna}} : \mathcal{A}(G) \to \mathbb{R}_0^+$ in pripadajoča funkcija prereza $c_{\text{lokalna}} : \mathcal{P}(V) \to \mathcal{P}(V)$;
- *lokalna:* funkcija cene $c_{\text{lokalna}} : \mathcal{P}(V) \times \mathcal{P}(V) \to \mathbb{R}_0^+$. ;
- *semi-lokalna:* funkcija cene $c_{\text{lokalna}} : \mathcal{P}(V) \times \mathcal{P}(V) \to \mathbb{R}_0^+$ in pripadajoča funkcija prereza $c_{\text{lokalna}} : \mathcal{P}(V) \to \mathcal{P}(V)$.

Ideja:
Pri procesu delitve eno gručo C_i v trenutnem gručevju $\mathcal{C} := \{C_1, \ldots, C_k\}$ razdelimo na dva dela. Proces se ustavi, ko se ne da nobene gruče več razdeliti.



Ideja:
Pri procesu delitve eno gručo C_i v trenutnem gručevju $\mathcal{C} := \{C_1, \ldots, C_k\}$ razdelimo na dva dela. Proces se ustavi, ko se ne da nobene gruče več razdeliti.



Vprašanji:

1. Kako izbrati gručo, ki jo bomo razdelili?
2. Kako izbrano gručo razdeliti?

Globalna verzija: Naj bo P množica vseh gručevij, ki jih dobimo iz gručevja $\mathcal{C}_i$, če razdelimo eno gručo C_i v dve novi gruči, glede na c_{globalna}; tj.

$$P := \Big\{ \{C_1, \ldots, C_k\} \backslash \{C_\mu\} \cup \{C'_\mu, C_\mu \backslash C'_\mu\} \mid \emptyset \neq C'_\mu \not\subseteq C_\mu \Big\}.$$

Novo gručevje $\mathcal{C}_{i+1}$ definiramo kot:

$$\mathcal{C}_{i+1} := \operatorname{argmin}_{\mathcal{C} \in P} c_{\text{globalna}}(\mathcal{C}).$$

Semi-globalna verzija: Naj bo P množica vseh možnih gručevij, ki jih dobimo iz gručevja $\mathcal{C}_i$, glede na S_{lokalna}; tj.

$$P := \{\{C_1, \ldots, C_k\} \backslash \{C_\mu\} \cup \{S_{lokalna}(C_\mu), C_\mu \backslash S_{\text{lokalna}}(C_\mu)\}\}.$$

Novo gručevje $\mathcal{C}_{i+1}$ je določeno kot:

$$\mathcal{C}_{i+1} := \text{argmin}_{\mathcal{C} \in P} c_{\text{globalna}}(\mathcal{C}).$$

Lokalna verzija: Naj bo μ tak indeks in C_ν taka prava podmožica gruče C_μ, da ima cena funkcije c_{lokalna} globalni minimum v paru $(C_\nu, C_\mu \backslash C_\nu)$. Potem je novo gručevje $\mathcal{C}_{i+1}$ definirano z razdelitvijo gruče C_μ glede na S_{lokalna}, tj.

$$\mathcal{C}_{i+1} := \{C_1, \ldots, C_k\} \backslash \{C_\mu\} \cup \{C_\mu, C_\mu \backslash C_\nu\}.$$

Semi-lokalna verzija: Naj bo μ indeks, pri katerem ima c_{lokalna} globalni minimum v paru $(S_{\text{lokalna}}(C_\mu), C_\mu \backslash S_{\text{lokalna}}(C_\mu))$. Potem je novo gručevje C_{i+1} definirano z razdelitvijo gruče C_μ glede na S_{lokalna}, tj.

$$\mathcal{C}_{i+1} := \{C_1, \ldots, C_k\} \backslash \{C_\mu\} \cup \{S_{\text{lokalna}}(C_\mu), C_\mu \backslash S_{\text{lokalna}}(C_\mu)\}.$$

Premična metoda

Ideja: Izberemo začetno gručevje in ga iterativno spreminjamo dokler ne najdemo lokalnega optimuma.

Ponavadi imamo za to na voljo tri operacije:

- vozlišče premaknemo iz ene gruče v drugo že obstoječo gručo;
- vozlišče premaknemo iz ene gruče v novo gručo, ki smo jo predhodno na novo ustvarili;
- dve vozlišči zamenjata gruči.

Dovoljujemo tudi bolj kompleksne operacije: npr. odstranitev obstoječe gruče in reorganizacija vozlišč.

$N_s(L)$ ... množica vseh gručevij, ki jih dobimo, če na L izvedemo modifikacije

Algoritem (Premična metoda)

Naj bo L_0 začetna rešitev.
$i \leftarrow 0$
while $N_s(L_i) \neq \emptyset$ **do**
  Izberemo $L_{i+1} \in N_s(L_i)$.
  $i \leftarrow i + 1$
end while
return L_i

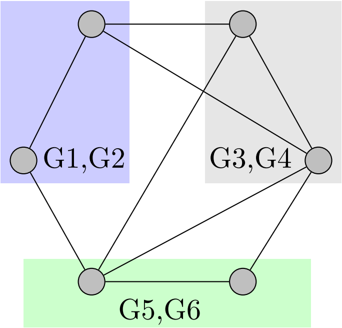
Definicija

Naj bodo dani graf $G = (V, E, \omega)$, začetno gručevje $\mathcal{C}_1$ in potencialna funkcija $\phi : \mathcal{A}(G) \times \mathcal{A}(G) \to \mathbb{R}$. **Premični proces** je izvedba neke operacije na trenutnem gručevju $\mathcal{C}_i$, ki nam da novo gručevje $\mathcal{C}_{i+1}$, tako da velja $\phi(\mathcal{C}_i, \mathcal{C}_{i+1}) > 0$.

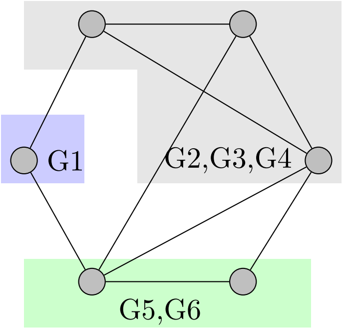
Opomba

Potencialno funkcijo ϕ premični proces uporablja kot kriterij za selekcijo.

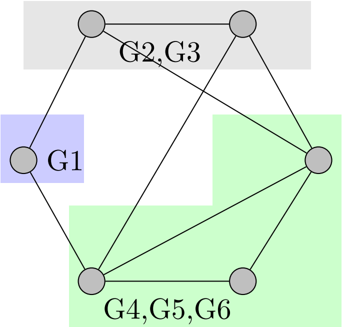
Zgled



Zgled



Zgled

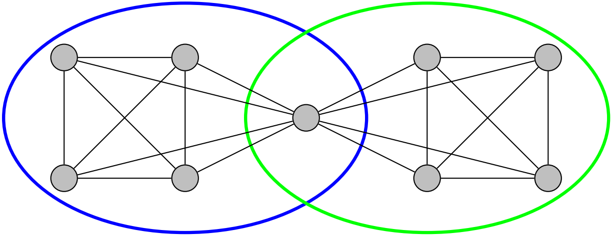


Splošni optimizacijski pristopi

- parametrično ocenjevanje;
- evolucijski pristop;
- $\dots$.

Prehodna gručavost

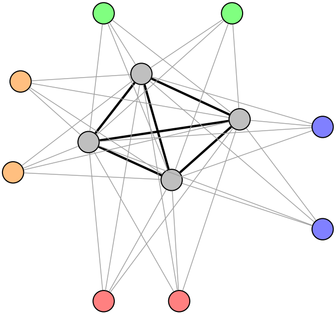
- Prerezno vozlišče pripada obema gručama.



Razširitve gručavosti

Problem: če majhna podmnožica vozlišč pripada mnogim gručam.

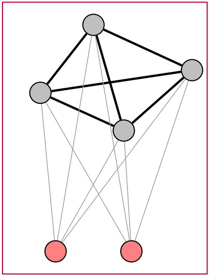
- Število gruč je omejeno s konstanto k.



Razširitve gručavosti

Problem: če majhna podmnožica vozlišč pripada mnogim gručam.

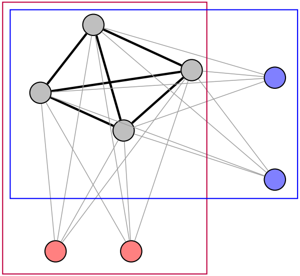
- Število gruč je omejeno s konstanto k.



Razširitve gručavosti

Problem: če majhna podmnožica vozlišč pripada mnogim gručam.

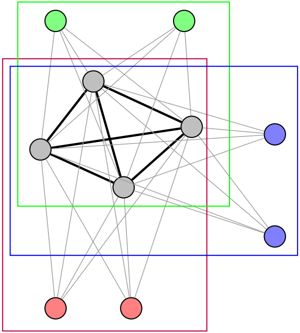
- Število gruč je omejeno s konstanto k.



Razširitve gručavosti

Problem: če majhna podmnožica vozlišč pripada mnogim gručam.

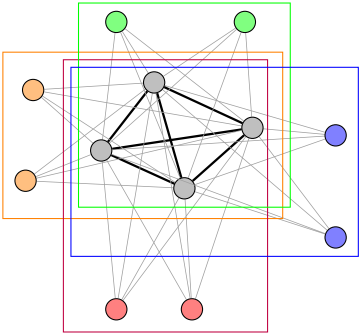
- Število gruč je omejeno s konstanto k.



Razširitve gručavosti

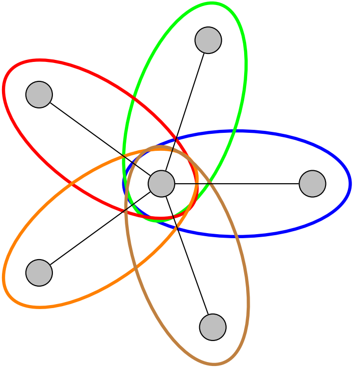
Problem: če majhna podmnožica vozlišč pripada mnogim gručam.

- Število gruč je omejeno s konstanto k.



Razširitve gručavosti

- Število k je primerljivo s stopnjo vozlišča.



Razširitve gručavosti

Gručavost s predstavnikom

Uporabne za pohitritev in aproksimativno računanje.

Gnezdena gručavost

Definicija

Gnezdena gručavost je gnezdeno zaporedje podmnožic vozlišč, tj. preslikava $\eta : \mathbb{N} \to \mathcal{P}(V)$ tako da:

1. množice so gnezdene, tj.

$$\forall i \in \mathbb{N} \,.\, \eta(i+1) \subseteq \eta(i);$$

2. zaporedje je končno, tj.

$$\exists k \in \mathbb{N} \,.\, \forall \ell > k \,.\, \eta(\ell) = \emptyset.$$

Razširitve gručavosti

- Najmanjši možni k je **velikost zaporedja**.
- Množica $\eta(k)$ je **vrhnji element**.
- Gostota podmnožic $\eta(i)$ je naraščajoča funkcija argumenta i.
- Dva ekstremna tipa: **hierarhije in vrhovi**.

Razširitve gručavosti

Hierarhija
Vsaka množica $\eta(i)$ inducira povezan graf.

Razširitve gručavosti

Vrhovi

Komplementaren pojem hierarhiji: vsaj ena podmnožica $\eta(i)$ inducira nepovezan graf.

Aksiomatika

Naj bo $K_n = (V, E)$ poln graf na n vozliščih in $\omega : E \to \mathbb{R}_0^+$ funkcija razdalj na povezavah. Množico vseh možnih funkcij razdalj označimo z D.

Definicija

Funkcija gručiranja f je preslikava $f : D \to \mathcal{A}(K_n)$, ki zadošča naslednjim aksiomom:
Invariantnost na merilo: $\forall \alpha \in \mathbb{R}^+, \omega \in D \,.\, f(\omega) = f(\alpha \cdot \omega)$, kjer je $(\alpha \cdot \omega) \cdot (u, v) := \alpha \cdot (\omega(u, v))$;
Bogatost: $f(D) = \mathcal{A}(K_n)$;
Konsistentnost: za vse $\omega, \omega' \in D$ velja

$$\left(\forall u, v \in V \,.\, \omega'(u, v) \begin{cases} \leq \omega(u, v) & u \sim_{f(\omega)} v \\ \geq \omega(u, v) & \text{sicer} \end{cases}\right) \Rightarrow f(\omega) = f(\omega').$$

Aksiomatika

Izrek

Za vse $n \geq 2$ *ne obstaja funkcija gručiranja.*

Lema

Obstaja veliko funkcij, ki zadoščajo samo dvema od treh aksiomov iz definicije.

Definicija

Naj bo dana množica elementov X. **Funkcija gručiranja grafa** f je preslikava $f : \Omega(X) \to \mathcal{A}(X)$, ki zadošča naslednjim aksiomom.

<u>*Invariantnost na merilo*</u>: $\forall \alpha \in \mathbb{R}^+, \omega \in \Omega(X) \,.\, f(\omega) = f(\alpha \cdot \omega)$, kjer je $(\alpha \cdot \omega) \cdot (u, v) := \alpha \cdot (\omega(u, v))$;

<u>*Bogatost*</u>: $f(\Omega(X)) = \mathcal{A}(X)$

<u>*Konsistentnost*</u>: za vse $\omega, \omega' \in \Omega(X)$, pri čemer je $E(\omega) \subseteq E(\omega')$, velja

$$\left(\forall u, v \in V \,.\, \omega'(u, v) \begin{cases} \leq \omega(u, v) & u \sim_{f(\omega)} v \\ \geq \omega(u, v) & \text{sicer} \end{cases}\right) \Rightarrow f(\omega) = f(\omega'),$$

kjer je $\omega(u, v) = \infty$ za $(u, v) \in E(\omega') \setminus E(\omega)$.