

Kratko poročilo – projektna naloga

Skupina 18: Število podpoti pri grafih z danim ciklomatskim številom

Uvod

V projektu obravnavamo povezane grafe $G = (V, E)$, za katere je definirano ciklomatsko število

$$\mu(G) = |E| - |V| + 1.$$

To število pove, koliko neodvisnih ciklov vsebuje graf oziroma koliko povezav bi morali odstraniti, da bi dobili gozd (acikličen graf).

CILJ NALOGE: raziskati, kako se obnaša število podpoti in geodetskih podpoti grafa, torej število vseh enostavnih poti (vključno s potmi dolžine 0), pri grafih z danim ciklomatičnim številom. Zanima nas, kateri grafi imajo za dano število vozlišč $|V| = n$ in dano ciklomatsko število $\mu = k$ najmanjše oziroma največje število poti.

Teoretično ozadje

Definicija 1 (Podpot). Naj bo $G = (V, E)$ povezan graf in $P = v_1 v_2 \dots v_k$ pot v grafu G . Podpot grafa G je vsaka zaporedna podzaporede oglišč te poti, torej vsaka pot oblike

$$v_i v_{i+1} \dots v_j, \quad \text{kjer } 1 \leq i < j \leq k.$$

Subpath number $p_n(G)$ štejemo kot skupno število vseh enostavnih poti v grafu, pri čemer se šteje tudi vsako vozlišče kot pot dolžine 0.

Definicija 2 (Geodetsko število). Naj bo $G = (V, E)$ povezan graf. Geodetsko število podpoti grafa G , označeno z $gp_n(G)$, definiramo kot skupno število vseh geodetskih enostavnih poti v grafu. Za vsaka dva vozlišča $u, v \in V(G)$ označimo z $d(u, v)$ razdaljo med njima, z $\sigma^*(u, v)$ pa število vseh enostavnih $u-v$ poti dolžine $d(u, v)$. Tedaj je

$$gp_n(G) = \sum_{u, v \in V(G)} \sigma^*(u, v).$$

Vračunane so tudi poti dolžine 0, torej $u = v$. Geodetsko število podpoti tako meri, koliko najkrajših enostavnih poti vsebuje graf.

Kaktusni graf je graf, v katerem se poljubna dva cikla stikata v največ enim vozlišču. To pomeni, da se cikli lahko dotikajo, vendar se ne prekrivajo.

Graf PTC(n,k) (angl. *Pseudo Triangle Chain*), slovensko psevdotrikotna veriga, je poseben primer kaktusnega grafa, v katerem so vsi cikli trikotniki, povezani v verigo, in dosega največje možno število podpoti med vsemi takšnimi grafi. Zato se v projektu pogosto uporabi kot osnovni primer (tj. zgornja meja) za primerjavo z drugimi grafi, ki imajo enako ciklomatično število $\mu(G)$.

Graf $PTC(n, k)$ ima naslednje lastnosti:

- je kaktusni graf z n vozlišči in k cikli;
- vsak notranji cikel je trikotnik (ima 3 vozlišča);
- cikli so med seboj povezani v verigo (en za drugim);
- prvi in zadnji cikel se lahko rahlo razlikujeta po velikosti (imata lahko 4 vozlišča namesto 3);
- vsi cikli se stikajo v natanko enim vozlišču, zato graf tvori verižasto strukturo.

Izrek (Knor, 2025) Graf $PTC(n, k)$ enolično maksimizira število podpoti med vsemi kaktusnimi grafi z n vozlišči in $k \geq 2$ cikli.

Iz literature (Knor, 2025): za kaktusne grafe veljajo naslednji ekstremni primeri

- maksimalni $p_n(G)$ doseže t.i. *pseudo triangle chain*, kjer so vsi notranji cikli trikotniki, oba končna pa se razlikujeta največ za eno vozlišče;
- minimalni $p_n(G)$ doseže graf, kjer so vsi cikli *end-trikotniki*, torej vsak trikotnik deli največ eno skupno vozlišče z ostalim grafom.

Plan dela

Projektno naloge bova ločila na dva dela.

1. Analiza za majhne grafe. V okolju SageMath bomo s pomočjo funkcije *gen* generirali vse povezane grafe z števili vozlišč $n \leq 8$ in za vsak graf izračunali: $|V|, |E|, \mu(G)$ in $p_n(G), gp_n(G)$. Pri majhnih grafih bomo $p_n(G)$ izračunali z iskanjem vseh enostavnih poti med pari vozlišč. Ker SageMath ne vsebuje vgrajene funkcije za štetje vseh enostavnih poti, smo uporabili rekurzivni pristop. Funkcija `all_simple_paths(G, start, end)` poišče vse poti med vozliščema `start` in `end`, tako da postopno obiskuje sosednja vozlišča, pri čemer se izogiba že obiskanih točk. Za vsak par vozlišč (u, v) se nato prešteje število poti in sešteje. Na podlagi rezultatov bomo določili grafe, ki imajo minimalni in maksimalni $p_n(G)$ pri enakem $\mu(G)$ ter preverili, ali so ti grafi kaktusnega tipa.

2. Analiza za večje grafe. Za grafe z več kot osmimi vozlišči bomo uporabili metodo simuliranega ohlajanja, saj popolna generacija vseh neizomorfnih grafov pri tej velikosti postane računsko neizvedljiva. Za dani vrednosti n in $\mu(G) = k$ bomo najprej ustvarili povezani graf z n vozlišči in točno $m = n - 1 + k$ povezavami, za katerega si mislimo da je podoben optimalnemu. Ta graf bo začetno stanje algoritma. V vsakem koraku bomo izvedli majhno lokalno spremembo grafa (t.i. *edge swap*), pri kateri odstranimo en obstoječi rob in ga nadomestimo z novim, tako da ohranimo število povezav in s tem tudi ciklomatsko število. Novo stanje sprejmemo vedno, kadar se vrednost $p_n(G)$ izboljša. Če se vrednost poslabša, takšno stanje sprejmemo le z verjetnostjo $e^{(\Delta/T)}$, kjer je Δ razlika med novim in trenutnim stanjem, T pa trenutna temperatura. Pri višjih temperaturah algoritmom pogosteje sprejema slabše korake, kar mu omogoča izogibanje lokalnim optimumom.

Ko temperaturo postopoma znižujemo po pravilu $T_{k+1} = \alpha T_k$, postane algoritem vedno bolj "konservativen" in sprejema skoraj izključno korake, ki izboljšajo rezultat. Po koncu iteracij kot končno rešitev vzamemo tisti graf, ki je med izvajanjem dosegel največjo (oziroma najmanjšo) vrednost $p_n(G)$, ter ga primerjamo z znanimi ekstremnimi kaktusnimi grafi za iste vrednosti n in $\mu(G)$.

Pričakovani rezultati in hipoteze

Pričakujemo, da bo vrednost $p_n(G)$ najnižja pri drevesih in bo naraščala z večanjem ciklomatskega števila $\mu(G)$, saj dodatni cikli omogočajo več različnih enostavnih poti med pari vozlišč. Na podlagi znane literature o kaktusnih grafih domnevamo, da bo minimalni $p_n(G)$ dosežen pri zaporedno povezanih ciklih, kjer so cikli razporejeni v verigo in tako le lokalno povečujejo število poti. Največji $p_n(G)$ pa pričakujemo pri strukturi, kjer imajo vsi cikli skupno vozlišče (*star cactus*), saj takšna razporeditev ustvarja veliko število alternativnih poti prek centralnega vozlišča.

Pri večjih grafih, kjer popolnega pregleda ni mogoče izvesti, pričakujemo pojav lokalnih ekstremov, zlasti v primerih, ko se cikli prekrivajo ali medsebojno vplivajo na način, ki ne povečuje večjega števila novih enostavnih poti. Takšne strukture lahko povzročijo le majhno rast $p_n(G)$ kljub večjemu številu povezav, kar želimo podrobno analizirati s pomočjo simuliranega ohlajanja.