KOMBINATORIKA

TEORIJA GRAFOV

Seminar Bloki

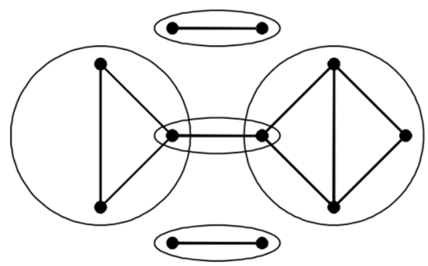
Špela Jezernik

Maribor, 17.5.2005

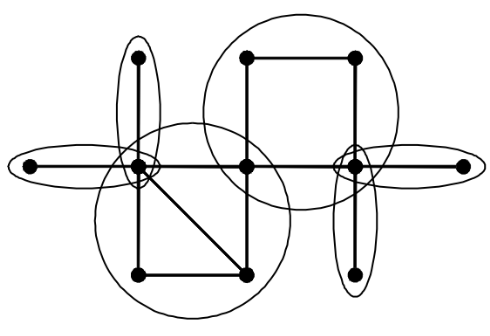
Povezan graf brez presečnih točk ni 2-povezan, ker je lahko K_1 ali K_2. Povezani podgrafi brez presečnih točk tvorijo uporabno dekompozicijo grafa.

DEFINICIJA 1. ***Blok*** *grafa G je maksimalen povezan podgraf grafa G, ki nima presečnih točk. Če je graf G sam povezan in nima presečnih točk, potem je G blok.*

PRIMER 1
Ta graf ima pet blokov; tri kopije K_2, eno K_3 in en podgraf, ki ni niti cikel niti poln graf.



PRIMER 2
Ta graf ima šest blokov; štiri kopije K_2, eno C_4 in en podgraf, ki ni niti cikel niti poln graf.



Če je H blok grafa G, potem H kot graf ne vsebuje presečnih točk, vendar lahko vsebuje presečne točke grafa G. Povezava v ciklu ne more biti blok, ker je v večjem podgrafu brez presečnih točk. Zato je povezava blok, če in samo če je presečna povezava. Bloki drevesa so njegove povezave. Če ima blok več kot dve točki, potem je 2-povezan. Bloki grafa brez cikla so njegove izolirane točke, presečne povezave in maksimalni 2-povezani podgrafi.

TRDITEV 2. *Dva bloka v grafu lahko imata največ eno skupno točko.*

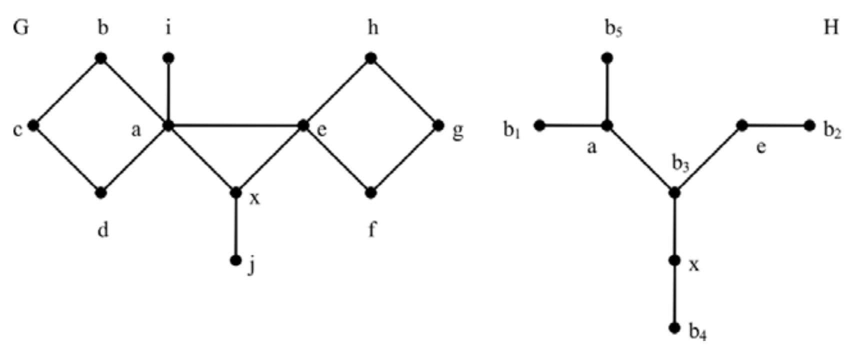
DOKAZ. Dokaz s protislovjem. Predpostavimo, da imata bloka B_1, B_2 najmanj 2 skupni točki. Pokažimo, da je $B_1 \cup B_2$ povezan podgraf brez presečnih točk, to bo v nasprotju z maksimalnostjo B_1 in B_2.

Ko zbrišemo eno točko iz B_i, je preostanek povezan. Saj ohranimo pot v B_i iz vsake preostale točke v vsako preostalo točko iz $V(B_1) \cap V(B_2)$. Ker imata bloka najmanj dve skupni točki, z brisanjem ene točke iz B_i obstaja v preseku vsaj še ena točka. Torej ohranimo poti iz vseh točk v to točko, zato $B_1 \cup B_2$ ne more biti nepovezan z brisanjem ene točke. $\square$

Vsaka povezava je sama po sebi podgraf brez presečnih točk in zato je v bloku. Bloki grafa razstavijo graf. Bloki v grafu se obnašajo podobno kot močne komponente digrafa, toda močne komponente nimajo skupnih točk. Bloki v grafu razstavijo množico povezav, močne komponente v digrafu razbijejo le množico točk in navadno izpustjo povezave.

Če si dva bloka grafa G delita točko, mora biti le-ta presečna točka grafa G. Povezava med bloki in presečnimi točkami je opisana s specialnim grafom.

DEFINICIJA 3. ***Graf blokov in presečnih točk*** *grafa G je dvodelni graf H, v katerem je en kos biparticije sestavljen iz presečnih točk grafa G, drugi kos pa iz točk, ki ustrezajo blokom B_i. V graf H vključimo povezavo vb_i, če in samo če je $v \in B_i$. Pri tem je b_i točka grafa blokov in presečnih točk, ki ustreza bloku B_i.*

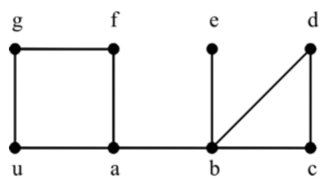


Če je graf G povezan, je njegov graf blokov in presečnih točk drevo, katerega listi so bloki grafa G. Graf G, ki ni sam blok, ima najmanj dva bloka (**list bloka**), da vsak vsebuje natanko eno presečno točko grafa G.

Bloke lahko najdemo z uporabo metode za iskanje grafov. **Iskanje v globino** (DFS-postopek) vedno išče od nazadnje odkrite točke, ki ima še neraziskane povezave (imenujemo tudi **sledenje nazaj**). Iskanje v širino (BFS-postopek) pa išče od najprej odkritih točk.

PRIMER 3

Eno iskanje v globino iz točke u najde točke v vrstnem redu u, a, b, c, d, e, f, g. Za BFS-postopek in DFS-postopek je vrstni red odkritih točk odvisen od vrstnega reda povezav iz preiskane točke.



Iskanje v širino ali globino iz točke u generira drevo, ki ima korenino v u. Ko raziskujemo točko x in najdemo novo točko v, vključimo povezavo xv. Tako dobimo vpeto drevo komponente, ki vsebuje u.

LEMA 4. *Če je T vpeto drevo povezanega grafa G, dobljeno z DFS-postopkom iz točke u, potem vsaka povezava grafa G, ki ni v T, sestoji iz dveh točk v, w, tako da v leži na u, w-poti v T.*

DOKAZ. Naj bo vw povezava grafa G, kjer smo v našli pred w z DFS-postopkom. Ker je vw povezava, ne moremo končati v v dokler w nismo dodali v T. Zato se w pojavi nekje v poddrevesu, ki je nastalo preden smo končali s preiskavo v in pot iz w v u vsebuje v. $\square$

ALGORITEM: Izračunanje blokov grafa G
Vhod: Povezan graf G (Bloki grafa so bloki njegovih komponent, ki jih lahko najdemo z DFS-postopkom. Zato lahko privzamemo, da je graf G povezan.)
Ideja: Zgradimo DFS-drevo T grafa G, odstranimo dele drevesa T, za katere odkrijemo, da so bloki. Ohranimo eno točko imenovano AKTIVNA TOČKA.
Začetek: Izberemo koren $x \in V(H)$, postavimo x AKTIVNA TOČKA in množica $T = \{x\}$.
Ponavljamo: Naj v označuje trenutno aktivno točko.

1. Če ima v neraziskane povezave vw, potem

   1A) Če $w \notin V(T)$, potem dodamo vw v T, označimo vw raziskana, postavimo w AKTIVNA TOČKA.

   1B) Če $w \in V(T)$, potem je w predhodnik v, označimo vw raziskana.

2. Če v nima več neraziskane povezave, potem

   2A) Če $v \neq x$ in je w predhodnik v, postavimo w AKTIVNA TOČKA. Če nobena točka v trenutnem poddrevesu T', ki ima korenino v v, nima raziskane povezave do predhodnika nad w, potem je $V(T') \cup \{w\}$ množica točk, ki predstavlja blok. To si zapomnimo in zbrišemo $V(T')$ iz T.

   2B) Če $v = x$, končamo.

PRIMER 4
Eno prečkanje v globino iz točke x obišče druge točke v vrstnem redu $a, b, c, d, e, f, g, h, i, j$. Tako najdemo bloke v vrstnem redu $\{a, b, c, d\}$, $\{e, f, g, h\}$, $\{a, i\}$, $\{x, a, e\}$, $\{x, j\}$.