Univerzita Komenského, Bratislava
Fakulta Matematiky, Fyziky a Informatiky

Konštrukcia grafov pomocou mobilných agentov

Diplomová práca

2014

Bc. Jakub Kováč

Univerzita Komenského, Bratislava
Fakulta Matematiky, Fyziky a Informatiky

Konštrukcia grafov pomocou mobilných agentov

Diplomová práca

Študijný program:	Informatika
Študijný odbor:	2508 Informatika
Školiace pracovisko:	Katedra Informatiky
Školiteľ:	prof. RNDr. Rastislav Kráľovič, PhD.

Bratislava, 2014

Bc. Jakub Kováč

Obsah

Zoznam obrázkov

Zoznam tabuliek

Kapitola 1

Úvod

Graf je vhodnou reprezentáciu údajov potrebných na riešenie mnohých problémov. V tomto smere našiel široké uplatnenie aj v informatike a oblastiach matematiky, ktoré sú s ňou úzko späté. My sa budeme zaoberať pojmom graf tak, ako je chápaný v Teórii grafov. V našom ponímaní je graf abstraktnou štruktúrou obsahujúcou vrcholy a hrany. Pričom každá hrana má práve dva konce - vrcholy, ktoré spája. Ak rozlišujeme počiatočný a koncový vrchol pri hranách grafu, hovoríme, že hrany sú orientované. Ak sú počiatočný a koncový vrchol hrany totožné, hovoríme, že táto hrana je slučka. Hrana vedie medzi počiatočným a koncovým vrcholom. Ka medzi niektorou dvojicou vrcholov vedie viacero hrán, hovoríme, že tieto hrany sú násobné. V rámci Teórie grafov sa skúmajú grafy z rôznych pohľadov a riešia sa na nich najrôznejšie úlohy. My sa budeme zaoberať grafmi bez slučiek, orientovaných a násobných hrán. Grafy boli na základe vybraných vlastností roztriedené do viacerých tried. Roztriedenie grafov medzi jednotlivé triedy umožnilo pozrieť sa na konkrétne prípady mnohých problémov pre ktoré nebolo známe všeobecné riešenie a dosiahnuť aspoň čiastočné výsledky. Taktiež to umožnilo optimalizovať všeobecné algoritmy a dosiahnuť na špecifickejších vstupoch rádovo lepšiu časovú náročnosť výpočtu.

Medzi významné problémy týkajúce sa grafov patria problémy ich konštrukcie a prehľadávania. V súčasntosti sa problému konštrukcie grafov venuje takmer výlučne teória grafových gramatík. Tento prístup používa podobné prostriedky a postupy ako známejšia teória formálnych jazykov a automatov s tým rozdielom, že grafové gramatiky sa týkajú grafov a sú teda o dosť zložitejšie. Napríklad pojem bezkontextovej grafovej gramatiky má viacero rozdielnych prirodzených interpretácií. V tejto teórii sa vyvinulo viacero prístupov a skúmaných modelov. Ďalším spomínaným problémom týkajúcim sa grafov je problém ich prehľadávania. V rámci teórie prehľadávania grafov sa skúmajú viaceré modely. Väčšina z nich má však spoločný základ. Týmto základom je entita nazývaná agent prípadne robot, automat atď., ktorá sa pohybuje po vrcholoch daného grafu a jeho úlohou je tento graf prehľať. Jednotlivé modely sa môžu líšiť vo viacerých vlastnostiach. Medzi najdôležitejšie z nich patrí vlastnosti agenta i spôsob jeho pohybu po grafe, vlastnosti grafu a definícia pojmu prehľadať. Prehľadanie grafu môže znamenať navštíviť každý vrchol grafu, ale tiež prejsť po každej hrane. Taktiež je dôležité, aké informácie získa agent o svojom okolí pri návšteve niektorého vrchola. Môže napríklad vidieť všetky vrcholy spojené hranou alebo len lokálne čísla portov. Naša práca skúma možnosti konštrukcie grafov pomocou mobilných agentov. Spája teda obe bližšie spomínané teórie. Teórie grafových gramatík sa týka v tom, že sa tiež zaoberá konštrukcio grafov. S prehľadávaním grafu pomocou agentov má spoločného viac. Ide totiž o akýsi duálny problém. Pri prehľadávaní ide často o získanie akejsi mapy neznámeho grafu a v našej práci sa naopak zaoberáme konštrukciou grafu pri znalosti mapy. V prvom prípade teda máme graf a chceme mapu, kdežto v druhom máme mapu a chceme graf. Keďže problém konštrukcie grafov pomocou mobilných agentov nebol dosiaľ skúmaný, zvolili sme si na skúmenie jednoduchý model. Na ňom sme skúmali možnosti konštrukcie grafov z jednotlivých tried. Zvlášť sme sa venovali skúmaniu konštrukcie hyperkocky a tried grafov $P_n \times P_m$ i $C_n \times C_m$. Výsledky na týchto triedach nájdete v hlavných kapitolách práce. Náš model sa skladá z agenta nachádzajúceho sa v počiatočnom vrchole grafu, ktorý ide konštruovať. Tento agent môže vždy vykonať jednu z troch možných operácií. Agent pozná lokálne značenie koncov hrán - portov vo vrchole, v ktorom sa nachádza. Toto značenie je dané poradím ich vzniku a agent nemá možnosť ho meniť. Operácie, ktoré môže agent vykonať, sú: otvorenie portu s danou značkou, prejdenie po hrane s lokálnou značkou portu k i vytvorenie novej hrany do dosiaľ neskonštruovaného vrchola; v ktorom sa agent vzápätí ocitne. K vlastnostiam modelu tiež patrí, že v momente, keď sú otvorené dva porty s rovnakou značkou, vznikne medzi nimi hrana. Krok agenta sme si definovali ako zmenu vrchola, v ktorom sa agent nachádza. Skúmali sme aký minimálny počet krokov agenta je potrebný na konštrukciu daného grafu pri rôzne veľkých množinách značiek, ktoré má agent k dispozícii.

Kapitola 2

Základné pojmy a defínície

2.1 Graf

V teórii grafov sa používa viacero rozdielnych definícii grafu, preto v tejto kapitole uvádzame základné definície pojmov použitých v našej práci.

Graf je obrazová reprezentácia údajov obsahujúca body - vrcholy a čiary, ktoré body spájajú - hrany. Hrana je úsek spojitej krivky začínajúci a končiaci v bode. Hrana vedie medzi vrcholmi v ktorých má konce. O vrchole, do/z ktorého hrana vedie a tejto hrane, hovoríme, že sú navzájom incidentné. Niektoré definície povoľujú aby hrana začínala a končila v tom istom vrchole; takúto hranu nazývame slučka. Pokiaľ vedie medzi dvoma vrcholmi viac hrán, hovoríme o násobných hranách. Striktné definície grafu nepovoľujú násobné hrany a slučky a ani my sa nimi nebudeme zaoberať.

Na reprezentáciu vrcholov sa v teórii grafov používajú prvky množiny a hrana je v množine hrán reprezentovaná dvojicou vrcholov, ktoré spája, teda dvojprvkovou podmnožinou množiny vrcholov. Ak rozlišujeme začiatočný a koncový vrchol hrany, hovoríme, že hrana je orientovaná. Takéto hrany potom zvykneme reprezentovať usporiadanou dvojicou vrcholov.

Nasledujú formálne definície dôležitých pojmov z teórie grafov. Ak chceme rozlíšiť, ktorý graf máme na mysli, používame označenie tohot grafu ako dolný index. Napríklad množina hrán E grafu G sa označuje E_G.

Definícia 1. *Graf G je usporiadaná dvojica (V, E), kde $E \subseteq \{u, v | u \in V, v \in V\}$.*

Množine V hovoríme množina vrcholov a množine E množina hrán.

Vrcholy označujeme malými písmenami prevažne z konca latinskej abecedy (okolo písmena v). Hrany označujeme malými písmenami zo začiatku latinskej abecedy (okolo písmena e) alebo ako dvojicu vrcholov, ktoré spája. Keďže sa zaoberáme neorientovanými grafmi, platí $(u, v) \equiv (v, u)$.

Definícia 2. *Hovoríme, že hrana $e \in E$ je incidentná s vrcholom $v \in V$, ak $\exists u \in V$: $e = (u, v)$*

Definícia 3. *Stupeň vrchola v je počet hrán s ktorými je incidentný; označujeme deg(v). Teda $deg(v) = |\{e|v \in e\}|$* [1]

Definícia 4. *Sled S v grafe G je postupnosť vrcholov a hrán končiaca a začínajúca vrcholom taká, že dva jej susedné prvky sú navzájom incidentné.*

Čiže $S = v_0, e_0, v_1, e_1, ..., e_{n-1}, v_n$; kde $\forall i : v_i \in V, e_i \in E$; $\{v_i, v_{i+1}\} = e_i$.

Definícia 5. *Ťah T v grafe G je taký sled S v tomto grafe, v ktorom sa neopakujú hrany. $T = S : \forall e_i, e_j \in S : e_i \equiv e_j \Rightarrow i = j$*

Definícia 6. *Hovoríme, že ťah alebo sled sú uzavreté, ak je ich posledný vrchol totožný s prvým vrcholom.*

Definícia 7. *Cesta P v grafe G je taký ťah T v tomto grafe, že sa v ňom neopakujú vrcholy.*

Definícia 8. *Cyklus C v grafe G je taký uzavretý ťah T v tomto grafe, že sa v ňom neopakujú vrcholy.*

Definícia 9. *Graf je súvislý, ak v ňom existuje sled obsahujúci všetky vrcholy.*

Definícia 10. *Podgraf H grafu G je taký graf, ktorého množina vrcholov V_H je podmnožinou vrcholov V_G grafu G. Množina hrán grafu H je podmnožinou tých hrán grafu G, ktoré sú incidentné len s vrcholmi z množiny V_H.*

Indukovaný podgraf je taký graf, ktorého množina hrán obsahuje všetky hrany, ktorých koncové vrcholy patria do jeho množiny vrcholov.

Definícia 11. *Most v grafe Gje taká hrana $e = \{u, v\}$ v grafe G, že ľubovoľný sled obsahujúci vrcholy u a v obsahuje hranu e.*

Definícia 12. *Hovoríme, že vrcholy v, u grafu G sú susedné vrcholy, ak $\exists e \in E_G$: $u, v, \in e$.*

Definícia 13. *Komponent K grafu G je taký jeho indukovaný súvislý podgraf, že v gafe G nevedie hrana medzi vrcholom v K a žiadnym vrcholom ležiacim vo zvyšku grafu G.*

Definícia 14. *Kostra súvislého grafu G je taký jeho súvislý podgraf, kde každá hrana je most.*

---

[1] Keďže sme si hranu definovali ako dvojprvkovú podmnožinu množiny vrcholov, môžeme si dovoliť tento zápis

Definícia 15. *Produkt kartézskeho súčinu grafov G a H označujeme $G \times H$, pričom platí:*

$V_{G \times H} = V_G \times V_H$

$E_{G \times H} = E_G \times V_H \cup E_H \times V_G$

Koncové vrcholy hrany $(d, v) \in E(G) \times V(H)$ sú vrcholy (x, v) a (y, v), kde $x, v \in d; d \in E(G)$, jedná sa teda o hranu $\{(x, v), (y, v)\}$. Pre hranu $(e, u) \in E(H) \times V(G)$ podobne.

2.2 Vybrané triedy grafov

Niektoré grafy majú spoločné vlastnosti na základe ktorých ich delíme do množín - tried grafov. Teraz si predstavíme niekoľko tried grafov potrebných v ďalších častiach diplomovky.

Definícia 16. *Cesta dĺžky n je graf, v ktorom existuje cesta na n vrcholoch a neobsahuje žiadne ďalšie hrany ani vrcholy. Takýto graf označujeme P_n.*

Podobne definujeme triedu grafov cykly.

Definícia 17. *Cyklus dĺžky n je graf s n vrcholmi, v ktorom existuje cyklus s n vrcholmi obsahujúci všetky hrany tohto grafu. Cyklus dĺžky n označujeme C_n.*

Kapitola 3

Grafové gramatiky

3.1 Úvod

Naša práca sa zaoberá vytváraním grafov a preto je nutné spomenúť najpoužívanejší formalizmus vytvárania grafov - grafové gramatiky. Cieľom tejto kapitoly je uviesť čitateľa do problematiky grafových gramatík a poskytnúť mu stručný prehľad tejto oblasti. Pri vypracovávaní tejto časti nám boli vo veľkej miere nápomocné práce týkajúce sa grafových gramatík, ktoré boli na našej fakulte vypracované v predošlých rokoch. Predpokladáme, že čitateľ má základné vedomosti z teórie formálnych jazykov a automatov.

3.2 Definície

Teória okolo grafových gramatík sa budovala podobne ako pri formálnych jazykoch a automatoch z viacerých prístupov pri ktorých boli neskôr dokázané určité ekvivalencie. Z toho vyplýva množstvo tried gramatík, prístupov a definícií. V konečnom dôsledku však možno rozlíšiť dva hlavné prístupy ku grafovým gramatikám - algoritmický (spájací) a algebraický (lepiaci). Pre ilustráciu uvedieme niekoľko tried a skúmaných problémov. Grafové gramatiky sa týkajú našej práce len okrajovo, v prípade hlbšieho záujmu oporúčame čitateľovi publikáciu Handbook of Graph Grammars [RE97] , z ktorej sme aj my čerpali.

3.3 Všeobecné pojmy

Za účelom uvedenia čitateľa do problematiky grafových gramatík sme si vybrali skupinu grafových gramatík založenú na prepisovaní vrcholov. Vo všeobecnosti má pravidlo odvodenia v grafovej gramatike tvar (M, D, E), kde M a D sú grafy a E je vkladací mechanizmus. Takéto odvodenie môže byť použité na ľubovoľný graf H, ktorý obsahuje podgraf M. Jeden výskyt podgrafu M je nahradený grafom D a spojený so zvyškom grafu H mechanizmom popísaným E. Zvyšok grafu H označujeme H^-. Na základe mechanizmu E sa rozlišujú spomínané dva prístupy. Pri lepiacom prístupe sa časti H^- stotožnia s časťami M a následne D. Graf D je skrz ne následne "vlepený"do grafu H^-. Pri spájacom prístupe vznikajú podľa E nové "spájacie"hrany medzi D a H^-. V spájacom prístupe sa v bezkontextových gramatikách skladá graf M len z jedného vrcholu. Teda vrchol je nahrádzaný grafom. Požaduje sa tiež, aby postupnosť nahrádzania vrcholov nemala vplyv na konečný výsledok. Pritom je viacero prirodzených spôsobov definovania mechanizmu E a obmedzení na "pravé strany"pravidiel - graf D.

3.4 NLC gramatiky

NLC (Node Label Controlled) grafové gramatiky sú jedným z najjednoduchších mechanizmov prepisovania grafov. Pravidlá NLC gramatiky sú tvaru $X \to D$, kde X je značka (label) prepisovaného vrchola a D je dcérsky graf. Pre použitie pravidie v NLC neexistujú žiadne obmedzenia, teda pravidlo sa môže použiť na ľubovoľný vrchol so značkou X. Vkladanie je v NLC lokálne, tj. vrcholy grafu D môžu byť spojené len s tými vrcholmi, s ktorými bol spojený prepísaný vrchol X. O NLC gramatikách, kde odvodenie je nezávislé na poradí použitia pravidiel odvodenia, hovoríme, že majú Church-Rosserovu vlastnosť - sú konfluentné (confluent). Vo všeobecnosti NLC túto vlastnosť nemajú a pri použití rovnakej sady pravidiel môžu vzniknúť rôzne grafy.

Definícia 18. *NLC grafová gramatika je 5-tica $G = (\Sigma; \Delta; P; C; S)$ kde $\Sigma - \Delta$ je abeceda neterminálnych znaciek uzlov a $\Delta(\Delta \subseteq \Sigma)$ je abeceda terminálnych znaciek uzlov. P je konecná množina NLC pravidiel, C je binárna relácia spojení nad Σ a S je pociatocný graf.*

Definícia 19. *Grafový jazyk generovaný G je $L(G) = \{H \in GR_\Delta | S \Rightarrow^* H\}, kdeGR_\Delta$ je množina neorientovaných grafov so značkami uzlov v Δ, $\Rightarrow$ reprezentuje jeden prepisovací krok, a $\Rightarrow^*$ reprezentuje odvodenie, t.j. postupnosť prepisovacích krokov.*

3.5 NCE

NCE (Neighborhood Controlled Embeding) grafové gramatiky sú rozšírením NLC grafových gramatík.V NCE mechanizme prepisovania je, na rozdiel od NLC, množina spájacích inštrukcií C definovaná pre každé pravidlo zvlášť a NCE gramatika môže k uzlom podgrafu D pristupovať priamo napr. ich očíslovaním; nemusí používať len značky týchto vrcholov. Toto už nie je možné pri vrcholoch hosťovského grafu (graf H), pretože v ňom môže byť počet susedov nahrádzaného vrchola neohraničený. Existujú rôzne rozšírenia NCE gramatík. edNCE gramatiky sú rozšírením NCE gramatík s dynamickým preznačovaním hrán pri prepisovaní. Písmenko e znamená, že sa budú značiť hrany a písmenko d označuje použitie orientovaných grafov.

Definícia 20. *edNCE grafová gramatika je definovaná ako 6-tica $G = (\Sigma; \Delta; \Gamma; \Omega; P; S)$. Kde Σ je abeceda značiek uzlov, $\Delta \subseteq \Sigma$ je abeceda terminálnych značiek uzlov, Γ je abeceda značiek hrán, $\Omega \subseteq \Sigma$ je abeceda terminálnych značiek hrán, P je konečná množina pravidiel a $S \in \Sigma - \Delta$ je počiatočný neterminál. Pravidlá sú tvaru $X \to (D, C), kdeX \in \Sigma - \Delta$ a $(D, C) \in GRE_{\Sigma,\Gamma}$. Relácia spojení je $C \subseteq \Sigma \times \Gamma \times \Gamma V_H \times in, out$.*

Vo všeobecnosti množinu grafov izomorfných s grafom H budeme značit $[H]$. Niekedy sa hovorí o H ako „konkrétnom“ grafe a o $[H]$ ako „abstraktnom“ grafe.

Definícia 21. *Množinu všetkých konkrétnych grafov nad Σ a Γ značíme $GR_{\Sigma,\Gamma}$ a množinu všetkých abstraktných grafov značíme $[GR_{\Sigma,\Gamma}]$. $GRE_{\Sigma,\Gamma}$ označuje množinu grafov s vnorením nad Σ, Γ. Obyčajný graf môžeme považovať za graf s prázdnym vnorením, teda $GR_{\Sigma,\Gamma} \subseteq GRE_{\Sigma,\Gamma}$.*

Definícia 22. *Grafový jazyk generovaný G je $L(G) = \{[H] | H \in GR_{\Sigma,\Omega}$ a $sn(S, z) \Rightarrow^* Hprenejaké z\}$ (t.j. všetky grafy v odvodení G, ktoré majú iba terminálne značky hrán aj uzlov).*

Teraz si definujeme hlavnú triedu bezkontextových grafových gramatík so spájacím prístupom. Označujeme ju C-edNCE. Táto je najrozšírenejšia v spájacom prístupe má značne všeobecný mechanizmus vkladania a žiadne reštrikcie na pravé strany.

Definícia 23. *edNCE gramatika $G = (\Sigma; \Delta; \Gamma; \Omega; P; S)$ je konfluentná (confluent), alebo C-edNCE gramatika, ak pre všetky pravidlá $X_1 \to (D1, C1)$ a $X_2 \to (D_2, C_2)$ z P, všetky uzly $x_1 \in V_{D_1} a x_2 \in V_{D_2}$, a všetky hrany so značkami $\alpha, \delta \in \Gamma$, platia nasledujúce ekvivalencie: $\exists \beta \in \Gamma : (X_2, \alpha/\beta, x_1, out) \in C_1 a(\lambda_{D_1}(x_1), \beta/\delta, x_2, in) \in C_2 \Leftrightarrow (\exists \gamma \in \Gamma : (X_1, \alpha/\gamma, x_2, in) \in C_2 a(\lambda_{D_2}(x_2), \gamma/\delta, x_1, out) \in C_1$.*

Poznámka 1. *Existuje charakterizácia C-edNCE nezávislá na gramatikách pomocou Monadic Second Order (MSO) logiky. Veľa výsledkov ju pre jednoduchosť využíva v dôkazoch.*

3.6 Rozličné známe výsledky

Hodno spomenúť, že vlastnosti, ktoré poznáme ako bezkontextové nemusia byť nutne konfluentné. [RE97] spomína pomerne jednoduchú lemu o konfluentnosti gramatiky, kde každý vrchol má "svoju"polovicu značky hrany a pri odvodení mení len tú. Následne nezáleží na poradí odvodenia dvoch vrcholov spojených hranou, lebo tieto menia len "svoju"časť značky hrany. Medzi zaujímavé veci v teórii o C-edNCE gramatikách patrí existencia takzvaných zakázaných (forbidden) hrán. Sú to hrany označené neterminálnym symbolom spájajúce dvojice terminálnych vrcholov. Z grafu obsahujúceho čo len jedinú takúto hranu sa už konečný graf nestane. Existujú mnohé modifikácie gramatík, ktoré sme si bližšie nespomínali. Skúmali sa aj problémy rozpoznávania, či graf patrí do jazyka; napríklad existuje NP-úplný LIN-A-edNCE jazyk. Taktiež sú preskúmané rôzne normálne formy. Veta o medzere hovorí, že existuje c pre ktoré sa nedá odvodiť v danom jazyku dosť grafov a tak ďalej.

3.7 Grafové gramatiky s nahrádzaním hyperhrán

Pre úplnosť venujeme krátky priestor aj vlepovaciemu prístupu v grafových gramatikách. Grafové gramatiky s nahrádzaním hyperhrán sú považované za grafovú ekvivalenciu k reťazcovým gramatikám. Ich začiatky siahajú do skorých 70. rokov. Základom je hyperhrana. Hyperhrana je atomický prvok s fixným počtom chápadiel - typov. Môže byť pripojená k ľubovoľnej štruktúre odvodenej z množiny vrcholov pripojením chápadla do vrcholu. Hyperhrana môže byť prepísaná štruktúrou R. R však musí mať, kvôli lepiacemu prístupu, rovnaký počet externých vrcholov ako nahrádzaná hrana typov. Gramatiky v tejto skupine majú konečný počet pravidiel a počiatočnú štruktúru. Jazyk tvoria množiny konečných štruktúr odvodených takouto gramatikou. Prepísanie hyperhrany je bezkontexotvé, pretože nezasahuje do okolia. Bolo dokázaných niekoľko výsledkov, ktoré majú blízko k podobným bezkontextovým gramatikám na reťazcoch. Ukázalo sa, že veľa zaujímavých problémov je pre tieto gramatiky rozhodnuteľných. Môžeme sa pýtať dva typy otázok: či hypergrafy generované príslušnou gramatikou spĺňajú vybrané vlastnosti alebo klasickú otázku príslušnosti do jazyka. Otázka príslušmosti je vo všeobecnosti NP-kompletná a len niektoré obmedzené triedy sú polynomiálne ťažké.

Kapitola 4

Prehľadávanie grafu

4.1 Úvod

Jednou z motivácií nášho prístupu ku konštrukcii grafu je možnosť využitia takejto konštrukcie ako duálneho problému k prehľadávaniu grafu. Náš model konštrukcie vychádza z modelov používaných pri prehľadávaní grafov s tým rozdielom, že kým v niektorých úlohách z prehľadávania je daný neznámy graf a úlohov agenta je skonštruovať jeho mapu, v našom modely je dané ako má graf vyzerať a úlohou agenta je skonštruovať ho pomocou operácií, ktoré má k dispozícii. V tejto kapitole uvádzame čitateľa do problematiky prehľadávania grafu, príslušných definícií a zaujímavých výsledkov z tejto oblasti. Nájde tu prehľad vybraných prístupov k tejto problematike a používaných modelov.

V problémoch prehľadávania má agent zostrojiť kompletnú mapu grafu (prostredia) bez akejkoľvek počiatočnej informácie o jeho topológii, prípadne len navštíviť každý vrchol alebo prejsť každou hranou. Tiež sú známe tri typy riešených úloh podľa toho, ako agent ukončí prehľadávanie. Prvým typom je prehľadávanie s návratom, keď agent po skončení musí zastať v počiatočnom vrchole. Druhým typom je prehľadávanie so zastavením, kedy agent musí v konečnom čase po prehľadaní celého grafu zastaviť. Posledným, tretím, typom úloh o prehľadávaní grafu je trvalé prehľadávanie. V tomto prípade agent pokračuje v prechádzaní grafu aj po jeho úplnom prehľadaní. Patria sem tiež úlohy, v ktorých má agent navštevovať každý daný prvok grafu (vrchol alebo hranu) s určitou periódou. Význam takejto úlohy je napríklad pri neustálej kontrole siete, ktorá sa môže kaziť počas prevádzky. Úspešnosť algoritmov prehľadávania grafov sa porovnáva s takzvanými offline problémami. Pri offline probléme agent graf pozná a jeho úlohou je navštíviť každý jeho vrchol, prípadne hranu podľa rovnakého zadania ako pri prehľadávaní. Napríklad ak má agent navštíviť každý vrchol v neznámom ohodnotenom grafe, online problémom je problém obchodného cestujúceho (TSP).

Prvý formálny model na prehľadávanie grafu uviedol [PY91]. Prvý raz bolo skúmané prehľadávanie celého grafu v [DP90], avšak uvažovali preskúmanie každej hrany v označenom orientovanom grafe. Teda, keď agent prišiel do vrcholu, dostal informáciu o počte neobjavených hrán odchádzajúcich z vrcholu ale už nie informáciu, kam tieto hrany vedú. Zodpovedajúcim offline problémom pre tento problém však nie je TSP ale polynomiálne riešiteľný Problém čínskeho poštára. Takéto zadanie problému prehľadávania má blízko k nášmu poňatiu problému konštrukcie grafu pomocou jedného mobilného agenta. V tejto oblasti prebieha rozsiahli výskum na orientovaných aj neorientovaných grafoch. Existuje aj iná trieda problémov, v literatúre označovaných ako online TSP, kde graf je známy dopredu a vrcholy, ktoré majú byť navštívené sa o bjavujú postupne. Zodpovedajúci offline problém pre túto triedu problémov je TSP s dátumami uvoľnenia (release dates).

Predpoklad, že konečný automat s konečným počtom kamienkov nedokáže prehľadať každý graf bol dlho otvoreným problémom až [Rol79] dokázal o niečo širšie tvrdenie. Rollik [Rol79] dokázal, že žiadna konečná množina konečných automatov nedokáže prehľadať všetky kubické planárne grafy.

Labyrint je šachovnica Z^2 so zakázanými políčkami. Labyrint je konečný. Prehľadávaním labyrintov sa vo veľkej miere zaoberal Budach. Prehľadať konečný labyrint znamená, že robot je schopný odísť ľubovoľne ďaleko zo svojej štartovnej pozície pre každú štartovnú pozíciu. Hrany labyrintu sú konzistentne označené svetovými stranami (východ, juh, západ, sever).

4.2 Prehľadávanie jedným agentom

[MMS11] sa zaoberá problémom prehľadávania na neorientovaných spojitých grafoch s hranami ováhovanými nezápornými reálnymi číslami a označenými (labeled) vrcholmi (agent ich vie rozlíšiť). Efektivita online algoritmov sa zisťuje ich porovnávaním s prislúchajúcimi offline riešeniami. V tomto prípade je prislúchajúcim offline problémom Problém obchodného cestujúceho (TSP).

Ďalši článok [FIP$^+$05] skúma prehľadávanie grafu konečným automatom (robotom). Graf je neoznačený s hranami lokálne označenými v každom uzle; rovnako ako v našom modely konštrukcie. Úlohou robota je prehľadať každú hranu v grafe bez akejkoľvek predošlej znalosti o veľkosti a topológii grafu. Medzi zaujímavé výsledky tohto článku patrí dôkaz existencie planárnych grafov, pre ktoré sa to nedá. V prípade unikátneho značenia hrán a vrcholov môže byť prehľadanie dosiahnuté ľahko. Existujú neznáme prostredia, v ktorých takéto značenie nemusí byť k dispozícii, alebo robot nemusí byť schopný rozlíšiť od seba dve podobné značky. Z tohto dôvodu sa skúma prehľadávanie anonymných grafov; tj. grafov bez unikátne označených hrán a vrcholov. Je len samozrejmé, že bez možnosti lokálne rozlíšiť konce hrán by nebolo možné prehľadať dokonca ani hviezdu s tromi listami. Predpokladá sa teda v každom vrchole označenie portov $1..d$, kde d je stupeň vrchola. Avšak nepožaduje sa žiadna konzistencia tohto značenia. Keďže v mnohých aplikáciách sa požaduje, aby bol robot malé lacné zariadenie, autori článku optimalizujú pamäť. Takže hľadajú robota schopného preskúmať graf danej neznámej veľkosti s tak malou pamäťou, ako je to len možné. Robot s k-bitovou konečnou pamäťou sa modeluje konečným automatom. Trap je graf, ktorý nie je možné celý prehľadať automatom bez kamienkov (pebbles). Výsledok článku hovorí, že pre každý K-stavový automat a $d \geq 3$ existuje K+1 vrcholový graf - trap s max. stupňom d, ktorý sa nedá prehľadať týmto automatom.

4.3 Distribuované prehľadávanie

Autori článku [DFK$^+$05a] používajú pre prehľadávanie neznámeho prostredia viacero totožných agentov. Neznáme prostredie je v tomto prípade opäť modelované grafom a úlohou agentov je skonštruovať totožnú mapu grafu pre každého agenta. Bolo dosiahnuté deterministické riešenie za čo najslabších možných nastavení, pričom agenty [1] poznali iba veľkosť grafu alebo počet agentov.

Autori skúmajú distribuovanú verziu klasického problému prehľadania garfu s návratom do počiatočného vrchola, pričom agent má prejsť všetky hrany neoznačeného (anonymného) grafu. Počas tohto procesu je úlohou agenta vytvoriť mapu grafu. Model pozostáva z neoznačeného grafu a k agentov roztrúsených po n vrcholoch grafu. Komunikácia medzi agentmi prebieha prostredníctvom zápisov na malé tabuľky nachádzajúce sa v každom vrchole grafu. Tento problém je zložitejší od prehľadania grafu jediným agentom, pretože vyžaduje kooperáciu medzi agentmi. Keďže je tento problém vo všeobecnosti deterministicky neriešiteľný, článok predstavuje deterministický algoritmus za podmienky, že n a k sú nesúdeliteľné.

Vo svojej práci dávajú autori svoj výskum do súvisu s ostatnými problémami distribuovaných výpočtov, ako je napríklad voľba šéfa.

4.4 Prehľadávanie s radou

Z praxe sú známe situácie, keď graf, ktorý má agent prehľadať, nie je úplne neznámy. Teda je o ňom dostupná nejaká informácia. V [DKM12] skúmajú autori predovšetkým zložitosť rady. Teda aký je vzťah medzi veľkosťou informácie, ktorú agent o grafe

[1] Na návrh staršieho kolegu skloňujeme slovo agent v našom kontexte ako neživotné.

dostane a kvalitou riešenia. Úlohou agenta je prehľadať s návratom do počiatočného vrchola neznámy neorientovaný graf. Čiže agent má navštíviť každý vrchol grafu a vrátiť sa do počiatočného vrchola s čo najnižším súčtom cien prechodov po hranách. V tomto modely agent pri návšteve vrchola vidí ceny všetkých s ním incidentných hrán a kam tieto hrany vedú.

Výsledky sa týkajú dolnej hranice, kde autori ukázali, že $\Omega(nlogn)$ bitov rady je potrebných na dosiahnutie pomeru porovnania 1 oproti algoritmu pozanjúcemu topológiu grafu. Taktiež dokázali, že s radou $O(n)$ bitov pri ohraničených váhach je možné dosiahnuť konštantný pomer.

Autori sa venovali aj klasickej otázke prehľadania grafu a posunuli dolnú hranicu prehľadania grafu deterministickým algoritmom bez rady.

Kapitola 5

Konštrukcia grafu pomocou mobilného agenta

/TODO slušnejšie spracovanie/ Tu začína text samotnej práce
Agent predstavuje v drvivej väčšine prípadov subjekt vykonávajúci nejakú činnosť alebo skupinu činností. Túto činnosť vykonáva podľa určitých algoritmov, majú/nemajú na neho vplyv vonkajšie podnety a príkazy. Nezriedka ide o skupinu agentov, ktorý môžu ale nemusia byť inštanciami rovnakej triedy. Príkladom použitia agentov sú simulácie ekonomických systémov, v ktorých agenti predstavujú jednotlivých účastníkov trhového mechanizmu. Vo všeobecnosti sa na základe svojich výpočtových obmedzení dajú zaradiť do chomského hierarchie výpočtových modelov z Teórie formálnych jazykov a automatov. V súvislosti s grafmi sa agenti najčastejšie používajú na prehľadávanie (vo význame objavovanie - exploration) a mapovanie grafu, hľadanie čiernych dier v ňom a podobne. Problematika prehľadávania grafu (graph exploration) je v súčasnosti už v značnej miere preskúmaná a preskúmané sú aj rôzne špeciálne prípady a modifikácie. Cieľom našej práce bude preskúmať možnosti využitia agentov na vytváranie grafu. Začneme s čo možno najjednoduchším modelom na najkrajších triedach grafov (kompletný, hyperkocka, mriežka, ...) a budeme si všímať horné a dolné odhady a ohraničenia zdrojov, ktoré agent pri generovaní grafov danej triedy potrebuje vzhľadom na triedu a jej konkrétnu inštanciu.

5.1 Všeobecné pravidlá pre agenta

5.1.1 Obmedzenia agenta

Ako sme už spomínali sila agenta sa bude líšiť podľa obmedzení v závislosti od triedy. Niktoré pravidlá však budú pre všetky triedy (alebo aspoň väčšinu) rovnaké. Sú to tieto: - agent je v jednom konkrétnom vrchole - agent vidí hrany vychádzajúce z vrcholu (vo väčšine prípadov neorientované) - agent si môže zvoliť po ktorej hrane odíde (ak obmedzenia nehovoria inak) - z pohľadu agenta sa elementy jedného druhu v grafe (hrany, konce hrán, ...) navzájom líšia len značením definovaným buď spôsobom vzniku grafu alebo určeným agentom, teda nemajú žiadne ID - ak agent nevie rozlíšiť ktorú z možností si má vybrať, vyberie si nedeterministicky - agent vo všeobecnosti vie, po ktorej hrane do vrchola prišiel - agent môže značiť každý (zmysluplný) element grafu (najmä hrany, konce hrán, vrcholy) ak obmedzenia nehovoria inak - ak má medzi portmi vzniknúť hrana, vzniká hneď ako sú splnené podmienky - agent má pamäť - agent má algoritmus - ak nie je dané, čo má agent vykonať, nekoná nič

Agent môže vo všeobecnosti a vzhľadom na dané obmedzenia vykonať tieto akcie: - otvoriť port vo vrchole - prejsť po existujúcej hrane do iného vrcholu - vytvoriť hranu z vrcholu, v ktorom je a na jej konci nový vrchol incidentný len s touto hranou, po vykonaní tejto akcie sa agent automaticky nachádza v tomto novom vrchole (ak nie je povedané inak) - označiť, preznačiť, označiť podľa obmedzení hocičo na čo vidí z miesta kde sa nachádza -nemôže označiť vrchol "niekde v paži podľa obmedzení môže upravovať značenie hrán buď v incidentnom vrchole alebo pri prechode hranou - ak nie je povedané inak, viditeľnosť agenta sa vzťahuje na daný vrchol, konce incidentných hrán a poľažmä aj na tieto hrany

Definícia 24. *Agent je -ica /TODO nech sme kompatibilný aspoň s niečím, najprv dáme šancu definíciám z graph exploration/ (chceme aby agent mal variabilnú počiatočnú informáciu určujúcu, aký graf má zostrojiť, napr. o dimenzii hyperkocky ...)*

Definícia 25. *Konfigurácia agenta je dvojica (v,w) kde v je vrchol v ktorom sa agent nachádza a w je konfigurácia výpočtového stroja agenta.*

Definícia 26. *Krok výpočtu agenta je binárna relácia na množine konfigurácií /TODO formálne značenie/ medzi konfiguráciami* (v_1, w_1) *a* (v_2, w_2) *ak* $v_1 = v_2$ *a* w_2 *patrí do* $\delta - funkcie$ w_1 *výpočtového stroja agenta.*

Definícia 27. *Konfigurácia systému agentov je -ica* $(G, M_1, ..., M_k, n, A_1, ..., A_n, ...)$*, kde G je graf,* M_1 *až* M_k *sú značenia grafu (TODO spresniť asi zahrnúť do definície grafu alebo inak zapuzdriť), n je počet agentov,* A_i *je konfigurácia i-teho agenta, ...*

Definícia 28. *Dosiahnuteľná konfigurácia agenta ...*

Definícia 29. *Graf skonštruovaný agentom je graf*

TODO umiestnenie nasl. definície

Definícia 30. */TODO/ Kamienok (pebble) je značka, ktorú môže agent položiť do vrchola a zase ju zdvihnúť. (Agent ma k dispozícii multimnožinu značiek zväčša jedného druhu).*

5.1.2 Metriky

Základnou otázkou pri našich modeloch bude, aké grafy vôbec vedia skonštruovať. Ako ďalšia prirodzene vyvstáva otázka, ako sú v tejto konštrukcii dobré. Na to, aby sme to boli schopní určiť, si teraz stanovíme, podľa čoho budeme určovať, ako je ktorý model dobrý. (Aby sme sa mali s čím hrať a robiť trade-of, definujeme si takýchto metrík niekoľko, zo všetkého, čo sa čo i len trochu dá) Podrobne sa stanoveniu metrík venujeme v časti definície.

5.2 Definície

5.2.1 agent

5.2.2 Definície metrík

Prvá metrika, ktorú si zavedieme sa týka potrebného počtu prechodov z vrcholu do vrcholu potrebných na skonštruovanie grafov danej triedy a jeho závislosti od vstupu. Táto metrika meria samotný výpočet a je "známaäž po jeho skončení. Teda nie je to niečo, čo by sme nejakým spôsobom aktívne obmedzovali ani to nevyžaduje žiadne pamäťové alebo iné nároky na agenta (okrem maximálneho počtu dosiahnuteľných konfigurácií bez zacyklenia).

Definícia 31. *Počet prechodov agenta pri konštrukcii /TODO značenie/ je počet krokov výpočtu agenta, v ktorých zmení vrchol v ktorom sa nachádza, ktoré nastali počas konštrukcie grafu.*

Ďalšie metriky sa týkajú obmedzení agenta a sú "známe"pred výpočtom. Najdôležitejšia je výpočtová sila agenta. V tomto prípade si poslúžime rozpoznávacími modelmi z Chomského hierarchie (DKA, PDA, DTS, ...). Ako základným členením si poslúžime už spomínanou Chomského hierarchiou a na jemnejšie delenie môžeme použiť počítanie stavov (v gramatikách neterminálov) a ďalšími používanými metrikami. Obmedzenie na pamäť agenta zahrnieme do výpočtovej sily agenta a budeme ho definovať v rámci stroja.

Definícia 32. */TODO/ Výpočtová sila agenta je určená množinou jazykov, ktoré dokáže rozpoznať výpočtový stroj používaný agentom.*

Ďalšími metrikami budú značenia, ktoré agent môže používať. Teda, či sa porty čísluјú za radom ako vznikali, inak deterministicky či nedeterministicky. Taktiež to či môže agent preznačovať porty a akú množinu značiek má k dispozícii , či sú čísla pevne dané od vzniku portu alebo sa menia (napr. podľa poradia kedy boli naposledy navštívené). Tiež obmedzenia na prípadné značenie vrcholov, množinu ich značiek a možnosť preznačovať ich.

Definícia 33. */TODO/ - značenie je množina dvojíc (objekt,značka) kde objekt* $\in$ *množina_objektov a značka* $\in$ *množina značiek - súčasť konfigurácie systému*

Ako už bolo spomenuté vyššie, ak chce agent vytvoriť hranu medzi vrcholmi, ktoré už oba existujú musí v oboch otvoriť porty s rovnakou značkou. Aby boli spojené požadované vrcholy, musí otvoriť porty tak, aby od okamihu otvorenia prvého z nich vrátane nebol, až po otvorenie prislúchajúceho portu v druhom zo spájaných vrcholov, otvorený port v inom vrchole s rovnakou značkou (teda taký, že by medzi ním a týmto vrcholom vznikla hrana a tento port by sa uzavrel). Na množinu značiek s ktorými môže tieto porty otvárať sa vzťahuje ďalšia metrika. Keďže môže byť otvorený naraz len jeden port s danou značkou (v okamihu otvorenia druhého vznikne medzi nimi hrana), táto metrika sa vzťahuje na mohutnosť množiny značiek portov.

5.3 Algoritmy na vybraných triedach

5.4

Kapitola 6

Konštrukcia mriežky

Definícia 34. *Mriežka veľkosti $n \times m$ je graf $P_n \times P_m$*

Definícia 35. *Cyklická mriežka veľkosti $n \times m$ je graf $C_n \times C_m$.*

V tejto kapitole sa budeme zaoberať mriežkou a cyklickou mriežkou. Na to, aby sme sa v týchto dvoch grafoch dokázali dobre orientovať, zavedieme pojem dimenzie a jej súradnice. Pričom využujeme, že oba grafy sú produktmi kartézskeho súčinu. Výsledky v tejto kapitole môžu byť užitočné pri hľadaní konštrukcií grafov, ktoré sú výsledkom kartézskyh súčinov zložitejších grafov. Cesta a cyklus totiž patria medzi najjednoduchšie podgrafy.

Označenie 1. *Pojem dimenzie v produkte kartézskeho súčinu G, grafov U a V zavedieme prirodzene vzhľadom na označenie vrcholov produktu ako usporiadanej dvojice (u, v), kde $u \in U$ a $v \in V$. i-tou dimenziou vrchola produktu tohto súčinu bude i-ty člen tejto dvojice. Súradnicou v dimenzii i je prvok na i-tej pozícii v usporiadanej dvojici. Pre zjednodušenie označíme vrcholy pôvodných grafov prirodzenými číslami od 1 po m respektíve od 1 po n. Vrcholy produktu kartézskeho súčinu týchto dvoch grafov budú pre dosť veľké m,n napríklad $(2, 4)$, $(5, 17)$, $(9, 1)$, atď.*

Poznámka 2. *Pri mriežke a cyklickej mriežke budeme číslovať vrcholy pôvodných grafov tak, aby sa označenia susedných vrcholov líšili o jedna, v prípade cyklickej mriežky budú v pôvodných grafoch - cykloch spojené aj vrcholy 1 a n, repektíve 1 a m.*

Lema 1. *Pri konštrukcii portovej hrany v mriežke aj v cyklickej mriežke, ktoré nemajú žiaden rozmer menší ako štyri, urobí agent medzi otvorením prvého a druhého portu najmenej tri kroky.*

Dôkaz. Najmenší cyklus v oboch grafoch má dĺžku štyri. Teda medzi otvorením dvoch portov hrany vykoná agent najmenej tri pohyby po hranách. Buď prejde po už existujúcich alebo vytvorí nové operáciou NH. □

6.1 Jedna značka pre porty

Lema 2. *Kostrové hrana je staršia, vznikla skôr, ako všetky portové hrany incidentné s jej mladším koncovým vrcholom.*

Dôkaz. Aby agent mohol otvoriť vo vrchole port, musel už tento vrchol existovať. Jediná operácia ktorou mohol vrchol vzniknúť je operácia NH. Mladší vrchol vznikol spolu s kostrovou hranou a preto doň predtým nemohli viesť žiadne hrany. □

Lema 3. *Tvoriaci cyklus hrany h obsahuje len hrany staršie ako h.*

Dôkaz. Tvoriaci cyklus obsahuje hrany, po ktorých agent prešiel, keď konštuoval hranu h. Museli teda v čase po prechode agenta už existovať a sú preto staršie ako hrana h, ktorú agent vytvoril až potom, ako po týchto hranách prešiel. □

Lema 4. *Každý tvoriaci cyklus je možné nahradiť tvoriacim cyklom, ktorý obsahuje len hrany kostry a príslušnú portovú hranu.*

Dôkaz. Majme tvoriaci cyklus hrany h. Každú portovú hranu, okrem h v ňom nahradíme jej tvoriacim cyklom, až kým nemáme uzavretý sled na kostrových hranách a hrane h. Z neho vieme vybrať tvoriaci cyklus hrany h. □

Označenie 2. *Graf prislúchajúci dimenzii d grafu G je d-ty činiteľ kartézskeho súčinu, ktorého je G produktom.*

Označenie 3. *Hrana v mriežke a cyklickej mriežke vedie medzi dvomi vrcholmi, ktoré sa líšia v práve jednej dimenzii a v tejto majú súradnice vrcholov, ktoré spolu susedia v grafe, ktorý prislúcha danej dimenzii. Teda ak je to prvá dimenzia jedná sa o prvý člen kartézskeho súčinu a ak je to druhá dimenzia o druhý. Súradnica hrany budeme volať súradnicu k, ak k je najvyššia súradnica v danej dimenzii a hrana spája vrchol s najnižšou a najvyššou súradnicou dimenzie, alebo nižšia zo súradníc v ktorých sa líšia vrcholy spojené touto hranou.*

Hranu so súradnocou k budeme volať niekedy hrana v súradnici k. Rovnako budeme používať pojem dimenzia hrany a hrana dimenzie.

Označenie 4. *Prechodom po súradnici budeme volať prechod agenta po hrane danej súradnice.*

Označenie 5. *Susedná súradnica súradnice k v dimenzii d, je každá súradnica vrchola spojeného hranou v grafe prislúchajúcom dimenzii d s vrcholom so súradnicou k.*

Lema 5. *Portová hrana v súradnici k v cyklickej mriežke má v tvoriacom cykle odlišnú paritu počtu prechodov po súradnici k ako majú počty prechodov po ostatných súradniciach V ostatnej dimenzii majú všetky počty prechodov po každej súradnici rovnakú paritu.*

Dôkaz. Aby otvoril druhý port hrany, agent musí svoju pozíciu zmeniť tak, aby sa nachádzal na susednej súradnici v diemnzii v ktorej sa líši s vrcholom s otvoreným prvým portom v dimenzii kam má viesť konštruovaná hrana. Keď rozdelíme pohyby agenta medi jednotlivé dimenzie, môžeme jeho pohyb premietnuť do grafov týchto dimenzií. Budeme teda sledovať len to ako sa mení tá, ktorá súradnica agenta pri jeho pohybe po tvoriacom cykle. Keďže vrcholy v ktorých otvára porty sa líšia iba v dimenzii konštruovanej hrany, v jednej dimenzii majú rovnakú súradnicu. Do grafu tejto dimenzie sa pohyb agenta premietne ako uzavretý sled, kde skončí vo vrchole, v ktorom začal. Poďme spočítať paritu prechodov po jednotlivých súradniciach v slede do ktorého sa v tomto grafe premietne prechod agenta. Vezmime tie výskyty toho istého vrcholu, medzi ktorými sú výskyty len iných vrcholov a to každého najviac raz. Takže aké pohyby mohol urobiť agent medzi dvomi najbližšími návštevami nejakého vrcholu? Mohol sa vrátiť zo susedného vrchola - zo vzdialenejšieho už nie, lebo susedný, cez ktorý by išiel by sa vyskytol na tomto slede viackrát ako raz. Alebo mohol agent prejsť po ceste, kde cieľom aj východiskom je tento vrchol. Takáto cesta je buď prázdna alebo obsahuje všetky iné vrcholy y je to vlastne prejdenie cyklu dookola. V prvom prípade sa parita prechodov po súradniciach zachová, lebo agent prešiel tam a spať po rovnakej súradnici. V druhom prípade sa zmení parita prechodov pre všetky súradnoce. V oboch prípadoch sa teda zachová rovnosť parít prechodov jednotlivých súradníc, ak platila aj predtým. Vieme teda, že medzi dvomi najbližšími výskytmi vrchola sa nič nepokazí. Na začiatku je počet prechodov pre každú súradnicu rovnaký - nulový a teda majú rovnakú paritu.Pri ďalšom počítaní môžeme teda tieto výskyty preskočiť - skontrahovať. Na takto upravenom slede postup opakujeme, až kým nemáme každý vrchol najviac raz okrem koncového a začiatočného, čo je ten istý. Preň platia rovnaké pravidlá a teda pre túto dimenziu je veta dokázaná.

V dimenzii konštruovanej hrany sa bude pri otváraní druhého portu nachádzať vo vrchole susednej súradnice k súradnici vrchola s prvým portom. Ak pridáme do sledu aj konštruovanú portovú hranu, dostaneme uzavretý sled, pre ktorý platí predošlý dôkaz. Z toho je jasné, že iba prechody po konštruovanej hrane majú odlišnú paritu počtu svojich výskytov. □

Lema 6. *Pri ľubovoľnej konštrukcii cyklickej mriežky G rozmerov $n \times m$ s jednou značkou pre porty, existuje aspoň jedna hrana pri konštrukcii ktorej agent vykoná aspoň $n-1$ krokov v prvej dimenzii a súčasne aspoň jedna portová hrana, pri konštrukcii ktorej agent vykoná aspoň $m-1$ krokov v druhej dimenzii, môže ísť o tú istú hranu.*

Dôkaz. Ako sme už povedali kostrové hrany sú hrany, ktoré vznikli operáciou NH a tieto hrany tvoria kostru grafu; ako bolo spomenuté grafom nazývame každý medziprodukt konštrukcie počnúc počiatočným vrcholom až po konečný skonštruovaný graf. Každej kinštrukcii grafu G zodpovedá nejaká kostra z kostrových hrán. Počet krokov, ktoré agent vykonal pri konštrukcii hrany h je súčet počtu krokov do vrchola v, v ktorom otvoril prvý port hrany a počet krokov z vrchola v do vrchola u, v ktorom otvoril druhý port hrany, tento sled z u do v tvoria s portovou hranou uzavretý sled. Tento sled obsahuje tvoriaci cyklus danej portovej hrany. Majme ľubovoľnú konštrukciu cyklickej mriežky G a k nej zodpovedajúcu kostru. Ukážeme, že pre ňu platí tvrdenie vety. Táto kostra určuje aj, ktoré hrany grafu sú portové - tie, ktoré nie sú v kostre. Pripomeňme si, že kostra zodpovedajúca konštrukcii je kostra tvorená práve všetkými hranami, ktoré vznikli operáciami NH. Ukážeme, že existuje aspoň jedna hrana, ktorej tvorivý cyklus bez tejto hrany obsahuje jej súradnicu párny počet krát - teda ostatné súradnice v tejto dimenzii nepárny počet krát. Rovnako to dokážeme pre obe dimenzie. Možné sú dva prípdy, buď ide o dve rôzne hrany, alebo o jedinú hranu. Keďže agent musel prejsť do susednej súradnice vrchola s prvým portom hrany a nemohol tak urobiť v jej súradnici, musel prejsť po všetkých ostatných súradniciach a teda vykonať aspoň n-1 respektíve m-1 krokov v prípade, že platí alternatíva s dvomi rôznymi hranami. Alebo n+m - 1 ak platí alternatíva s jedinou hranou. Nazvyme ich pre lepšiu orientáciu dotknuté hrany. Pozrieme sa na ľubovoľnú kostru. V tejto kostre sa nachádzajú všetky vrcholy grafu. Aký je tvorivý cyklus jednotlivých portových hrán. V tvorivom cykle portovej hrany h je možné nahradiť niektorú portovú hranu jej tvorivou cestou. Takýmto nahrádzaním postupne vieme nahradiť všetky portové hrany. Keď z výsledného sledu vyberieme konštrukčnú cestu, tento už obsahuje iba kostrové hrany. Pozrime sa na paritu počtu prechodov jednotlivých súradníc v tejto ceste. Tvrdíme, že ak sme nenahrádzali hranu, ktorej tvorivá cesta obsahovala súradnicu tejto hrany párny počet krát, výsledná tvorivá cesta na kostrových hranách bude obsahovať nepárny počet prechodov po súradnici hrany h. Toto je zrejmé. Ak výskyt hrany nahradíme cestou, kde sa táto hrana nachádza nepárny počet krát, počet prechodov po súradnici hrany sa nezmení. Toto zároveň implikuje, že prechody po ostatných súradniciach sa na tejto ceste nachádzajú párny počet krát zvlášť pre každú súradnicu. Teda nezmení sa ani parita prechodov po ostatných súradniciach. Čo sa týka parít počtov prechodov v ostatnej dimenzii, tak tu platí to isté, ak nahrádzame hranu jej tvorivou cestou, kde je počet prechodov po súradnici tejto hrany párny, tak aj vo výslednom slede je počet prechodov párny pre každú súradnicu tejti dimenzie.

Keďže výsledný sled, krorý vznikol nahradením tvorivej cesty portovej hrany h vedie len po kostrových hranách a na tých neexistuje cyklus, tvorivá cesta hrany h vybraná z tohto sledu zachováva paritu prechodov po jednotlivých súradniciach takú, aká je v tomto slede. Kostra je totiž strom a ak sa na slede vyskytne dvakrát ten istý vrchol, tak medzi dvomi jeho susednými výskytmi je párny výskyt prechodov po jednotlivých súradniciach a teda ho môžeme zo sledu vybrať - skontrahovať na jeden výskyt tohot vrchola. Jediná šanca ako pri pohybe po strome navštíviť opätovne niektorý vrchol je vrátiť sa doň.

Na kostre existuje pre každú portovú hranu jediná tvorivá cesta, inak by na nej bol cyklus, čo je spor. Vezmime si teraz dimenziu d a vrcholy, ktoré majú v ostatnej dimenzii rovnakú súradnicu, označme ju k. Indukovaný podgraf na týchto vrcholoch je cyklus. Niektoré jeho hrany sú kostrové, ostatné sú portové. Pokiaľ má každá portová hrana tvorivú cestu na kostre s nepárnym počtom prechodov po jej súradnici, existuje na kostre cyklus, čo je spor. Tento cyklus získame z uzavretého sledu vytvoreného z indukovaného grafu týchto vrcholov, kde portové hrany nahradíme ich tvorivými cestami na kostre. Vo výslednom slede sa každá hrana bude nachádzať nepárny počet krát a ako sme už ukázali, ak vyberieme niektorú portovú hranu f, vieme z tohto cyklu vybrať jej tvorivú cestu, ktorá zachová paritu prechodov po jednotlivých hranách. Keďže sa však v tomto slede nachádza prechod po súradnici hrany f párny počet krát, bude sa aj v ceste, čo je spor. Napríklad s tým, že táto hrana má tvorivú cestu na kostre s nepárnym počtom výskytov jej súradnice, alebo že v kostre existuje medzi dvomi vrcholmi - koncovými vrcholmi hrany f, práve jedna cesta. Z toho vyplýva, že niektorá nahrádzaná hrana mala tvorivú cestu s párnym počtom prechodov po jej súradnici. Teda pre každú dimenziu existuje hrana, ktorej tvorivá cesta obsahuje nepárne počty prechodov po súradniciach tejto dimenzie. A agent pri jej konštrukcii musí vykonať aspoň $n-1$ krokov pre prvú dimenziu a iných $m-1$ krokov pre druhú dimenziu. □

Lema 7. *Mriežku $m \times n$ je možné s použitím jedinej značky pre port zostrojiť pomocou $3m(n-1)+m$ pohybov, kde $m <= n$.*

Dôkaz. Agent vykonáva: OP,NH,NH,NH,OP,(n-2)krát (OP,S1,NH,NH,OP),S1,NH,HN,OP čím vytvorí prvý riadok mriežky. Následne vykoná (m-1)krát (NH,OP,S1,S2,NH,OP,(n-1)krát (OP,S1,S2,NH,OP)) □

Dôkaz. Pokrytie jednou dlaždicou stojí minimálne tri kroky plus cesta do vechola prvého portu. Dĺžka tejto cesty závisí od polohy agenta v predošlej dlaždici, jeho polohy v novej dlaždici a od vzájomnej polohy týchto dlaždíc.

Vzhľadom na dimenzie môžeme určiť štyri smery pohybu v grafe. Podľa toho, či súradnica v danej dimenzii stúpa alebo klesá.

Pri konštrukcii na tri kroky agent príde do vrchola s duhým portom a otvorí ho v inej dimenzii ako je tá v ktorej prišiel. Agent sa pohol v smere portovej hrany. Ak agent konštruuje dlaždice tak, že dve po sebe skonštruované majú jednu hranu spoločnú - jedna z dvoch ktoré vedú do vrchola s druhým portom predošlej portovej hrany, tak si zachováva smer v oboch dimenziách - stojí 0 krokov, každá zmena smeru stojí 1 krok.

Chceme konštrukciu na čo najmenej krokov. Teda hľadáme takú postupnosť dlaždíc, ktorou vie agent pokryť mriežku, že stojjí konštrukcia tejto postupnosti čo najmenej krokov. Čím sú od seba dlaždice ďalej, tým viac agent vykoná krokov pri ich konštrukcii, lebo medzi skonštruovaním týchto dlaždíc musí prejsť ich vzdialenosť. Ak agent vzápätí po konštrukcii dlaždice konštruuje ďalšiu, stojí ho pohyb medzi nimi 0 krokov. V tomto prípade sa tieto dlaždice dotýkajú rohmi. Ak niektorá dlaždica susedí s dvomi kockami spojenými rohom a konštruovanými hneď po sebe, je výhodnejšie vsunúť túto dlaždicu v konštrukcii medzi tieto dve. (Toto určite platí o dlaždiciach na okraji mriežky a odtiaľ sa to dá preniesť vyššie indukciou) Susedné kocky teda majú spoločný nielen roh ale aj hranu. Agent skončí na jednom z koncov portovej hrany. Portová hrana môže byť len na okraji pokrytej plochy. Po konštrukcii novej dlaždice sa agent ocitne v rovnakom rohu ako bol v predošlej dlaždici keď otváral prvý port portovej hrany novej dlaždice (keď z predošlej dlaždice odchádzal najkratšou cestou - pokiaľ portová hrana nesusedila s dlaždicou bude v susednom vrchole). TODO treba rozobrať všetky prípady najkratších ciest konštrukcie novej dlaždice ... Tam ukážeme susednosti ... tj. ktoré hrany sa opaltí vyberať ako portové ... (treba použiť argument o najkratšej konštrukcii, ak by portová hrana nemala susediť s predošlou dlaždicou, musí urobiť kroky navyše, ak všetky vrcholy dlaždice sú incidentné so 4 hranami, buď som skončil alebo sa niečo pokazilo ... ak dokazujeme konštrukciu postupne, do takéhoto stavu sa snáď nedostaneme ...)

Okraje mriežky sú vrcholy, ktoré majú jednu súradnicu buď minimálnu alebo maximálnu. Rohy mriežky sú vrcholy, ktoré majú maximálne alebo minimálne obe svoje súradnice. Rohy mriežky majú stupeň dva a vrcholy na okraji, ktoré nie sú rohmi, majú stupeň tri. Keď prvý raz príde agent do vrchola, tak hrana po ktorej prišiel musí byť kostrová. Každé prerušenie ťahu tvoreného portovými hranami, znamená, že agent okrem krokov medzi otvorením prvého a druhého portu niektorej portovej hrany, musel vykonať aspoň jeden krok medzi otvorením druhého portu niektorej hrany a prvého portu nasledujúcej. Počet týchto krokov nazvime cena prerušenia. Čím je menšia suma cien všetkých prerušení, na tým menej krokov agent skonštruoval mriežku (za predpokladu, že počet krokov na hranu je konštantný - 3). Vrchol s nepárnym počtom incidentných portových hrán - prerušenie vnútorný vrchol ako list - prerušenie Z tohto vyplýva, že v rohu je buď prerušený ťah, alebo doň nevedie portová - hrana. Vrchol na kraji má tri možnosti: - vedie doň jedna portová hrana -> prerušenie - žiadna portová hrana -> všetky tri sú kostrové - dve portové hrany

V druhom prípade, ak sú vedľa seba dva takéto vrcholy, tak vnútorné vrcholy s nimi spojené už sú v kostre a teda medzi nimi vedie portová hrana. Tiež je jasné, že vnútorný vrchol môže susediť najviac s jedným takýmto krajným vrcholom.

Vrcholov podľa druhého prípadu nevytvárajúcich prerušenia je obmedzene veľa. Ak totiž podgraf kostry spojený na hranu idúcu do vnútra mriežky neobsahuje vrchol s nepárnym počtom portových hrán, musí mať všetky listy na okraji. Na každý takýto podgraf teda pripadá minimálne jeden list na okraji a teda prípad dva môže platiť pre najviac polovicu vrcholov na okraji. Ukážeme, že v optimálnej konštrukcii je to menej.

V treťom prípade platí, že ak agent konštuuje portové hrany a príde do vrchola na okraji otvoriť port, tak na tri kroky otvorí port opäť len na okraji (ak je to možné). Teda až do prerušenie vedie ťah z portových hrán už len po okraji.

Je zrejmé, že spolu môžu susediť len rovnaké prípady alebo prvý prípad s ľubovoľným iným a druhý prípad s tretím ak to bude "prechod ťahu na okraj".

Tiež vidno, že ak by v treťom prípade boli vnútorné vrcholy navštívené prvý krát, išli by z nich tri kostrové hrany. Potom by v nich bolo buď prerušenie alebo by z nich musela vychádzať štvrtá kostrová hrana.

Ak má agent skonštruovať hranu na tri kroky, musí hrana v rovnakej súradnici o jedna ďalej buď existovať alebo byť kostrová. (Agent nemá k dispozícii značku pre portovú hranu). Ak je kostrová, musí viesť do ešte nevytvoreného vrchola.

Hľadáme teda spôsob pokrytia pri ktorom agent vykoná čo najmenej krokov, keďže počet krokov na dlaždicu je tri a je konštantný, zanedbáme ho. Najmenej agent potrebuje krokov pokiaľ portová hrana novej dlaždice začína vo vrchole, kde končí portová hrana predošlej. Keďže mriežka je osovo symetrická, bez ujma na všeobecnosti, nech skončil v pravom hornom rohu. Odtiaľ môže na 0 krokov skonštruovať tri rôzne kocky, vpravo hore a šikmo hore. Toto je možné len vtedy, ak sú potrebné vrcholy buď ešte nevytvorené, alebo vhodne spojené s predošlou dlaždicou. Pozrime sa na šikmý pohyb. Pri šikmom pohybe sa mení smer agenta. Ak šikmým pohybom obíde roh mriežky, bude drahé sa doň vracať. Je výhodnejšie vsunúť ho a vybrať z iného miesta v postupnosti. Vsunutie stojí 1 krok. Zmena smeru pri dotyku hranami stojí 1 krok. Ak agent skončí v rohu mriežky, bude stáť aspoň 1 krok ísť z neho konštruovať novú dlaždicu. Ak nikde doľava dole sa neoplatí šikmý pohyb, neoplatí sa šikmý pohyb ani tu, argumentácia podobná ako s rohom mriežky - vrchol šikmo doprava hore od ľavého dolného rohu má rovnaké postavenie ako ľavý dolný roh... totiž tú kocku, ktorú obíde šikmím pohybom stojí viac zapojiť do postupnosti ... všetko čo je drahšie ako 1 krok je drahé. Vľavo dole pod ňou už nie sú šikmé pohyby Teda šikmý pohyb sa neoplatí. Zmena smeru agenta stojí teda 1 krok. Na začiatku si agent "môže vybrať"v každej dimenzii ľubovoľný smer, môže totiž začínať v ľub. rohu prvej dlaždice. Ak agent konštruuje dlaždicu, ktorá má v niektorej dimenzii z oboch strán voľné miesta, znamená to, že bude musieť pokryť niektorú stranu a aby sa dostal do tej druhej, bude musieť meniť smer. Preto je najlepšie ak agent konštruuje dlaždice tak, aby z dvoch strán mal všetky štvorce pokryté.

Teda začína v rohu, ide po niektorom kraji a konštruuje po pásoch. Na konci vždy musí zmeniť smer, čo stojí 1 krok. Čím menej je teda pásov, tým menej musí agent meniť smer a menej pásov je po väčšej súradnici kocky. Spôsob konštrukcie mriežky $n \times m$ z vety TODO je optimálny.

Mriežka má štyri rohy. Pri akejkoľvek postupnosti plnenia dlaždicami, môže na začiatok a koniec minúť len dva z nich, cez zvyšné dva musí teda prechádzať postupnosť vypĺňania. Bez ujmy na všeobecnosti nech prechádza cez ľavý dolný roh a príde zľava. □

Lema 8. *Pri konštrukcii mriežky je najvýhodnejšie konštruovať portové hrany tvorivou cestou dĺžky 3.*

Dôkaz. Ukážeme, že pre každú kostru existuje taká postupnosť konštrukcie portových hrán, že ich tvorivá cesta má dĺžku 3. Tvorivá cesta po kostre je pre každú portovú hranu len jedna. Dajú sa úseky každej takejto tvorivej cesty postupne nahrádzať portovými hranami, ktorým sú tvorivými cestami tak, aby na konci mala tvorivá cesta dĺžku tri? Nie nutne. Nahradíme všetky vzory iki portovou hranou k. Tvorivá cesta obkolesuje vrcholy. Tieto tiež sú v kostre a teda kostrové hrany do nich vedúce sa musia napájať na túto tvorivú cestu.

–TODO treba ukázať, že aj keď sa vezmú do úvahy presuny, tak sú trojkrokové cesty výhodnejšie ... □

Lema 9. *Na zostrojenie mriežky $m \times n$, kde $m <= n$ potrebuje agent s jednou značkou pre porty vykonať aspoň $3m(n-1) + m$ krokov.*

Dôkaz. Premietnime si mriežku do roviny (mriežka je planárny graf), potom každý minimálny cyklus ohraničuje časť roviny. Ak minimalizujeme dĺžky tvorivých cyklov hrán, nič nepokazíme./TODO - možno to vyplynie samo/ Medzi otvorením prvého a druhého portu hrany agent

□

Kapitola 7

N-rozmerná kocka

Vrcholy spojené hranou sa líšia práve v jednej súradnici. Každej hrane z vrcholu možno priradiť súradnicu v ktorej sa jej druhý vrchol od tohto odlišuje. Toto priradenie je injekcia. Susedia vrchola sa navzájom líšia v práve dvoch súradniciach. Pri pohybe po hrane mením jednu súradnicu svojej pozície. Z tohto vyplývajú obmedzenia pre úsporu pohybov pridávaním značiek pre porty. Napríklad pri použití dvoch značiek pre porty potrebuje agent na vytvorenie dvoch nových portových hrán aspoň 5 ťahov. Úspora nastáva, keď port pre druhú hranu otvorím už počas cesty za otvorením portu prvej hrany.

Lema 10. *Dva vrcholy, ktoré sa navzájom líšia v práve dvoch súradniciach majú práve dvoch spoločných susedov.*

Dôkaz. Susedné vrcholy sa líšia práve v jednej súradnici. Spločný sused dvoch vrcholov sa líši v práve jednej súradnici od každého z nich. Teda má všeky súradnice rovnaké ako jeden z nich a líši sa v jedej z tých dvoch. Teda ju má rovnakú ako druhý vrchol. Druhú z týchto súradníc má potom odlišnú od druhého vrchola. Čiže sa od každého z nich líši v práve jednej súradnici. Toto dáva práve dva rôzne vrcholy, ktoré susedia z oboma. Vrchol, ktorý sa od jedného z vrcholov líšiacich sa v dvoch súradniciach líši v nejakej inej, sa od druhého líši v troch súradniciah a teda s ním nesusedí. □

Otázkou ostáva, koľko hrán vie agent vytvoriť počas jednej návštevy vrcholu otváraním portov?

Lema 11. *Pri jednej návšteve vrcholu môže agent vytvoriť najviac* $(fracn - 12) >$ *#dimenzii hrán otvorením portu.*

Poznámka 3. *Rátajú sa hrany, ktoré vzniknú hneď tj. pri ktorých agent otvorením portov vytvorí druhý (posledný) koniec.*

Dôkaz. Je zrejmé: že vo všetkých vrcholoch, kam majú tieto hrany viesť, už agent bol; medzi dvomi vrcholmi vedie najviac jedna hrana. Z tohto a predošlej vety vyplýva, že po návšteve n vrcholov môže agent otvorením portu vytvoriť z vrcholu, kde sa nachádza, najviac $(fracn - 12) > \#dimenzii$ hrán. Z týchto $(fracn - 12)$ vrcholov sa od neho každý líši v inej súradnici, lebo ak by sa dva líšili v rovnakej, museli by to byť totožné vrcholy. □

Lema 12. *Existuje taká postupnosť n vrcholov, že pri návšteve posledného agent vytvorí* $(fracn - 12) >> \#dimenzii$ *hrán.*

Dôkaz. Ak vezmeme takúto postupnosť odzadu, vyzerala by až na substitúciu dimenzií takto: 1, 2, 1, 3, 2, 4, 3, 5, 4, 6, 5, ... □

Poznámka 4. *Avšak ľahko vidieť, že žiadne iné portové hrany by medzi vrcholmi v tejto postupnosti vzniknúť nemohli.*

Skúsme teda nájsť postupnosť (cestu v hyperkocke) s najväčším možný počtom portových hrán medzi jej vrcholmi.

Označenie 6. *O hrane povieme, že vedie v dimenzii k, ak sa vrcholy, ktoré spája, líšia v práve tejto dimenzii.*

Je jasné, že v akomkoľvek súvislom slede vrcholov povedie hrana medzi dvojicami nasledujúcich vrcholov, ide o hrany po ktorých agent prejde z vrcholu do vrcholu. Portová hrana môže vzniknúť medzi vrcholmi, ktoré sa líšia v práve jednej súradnici, lebo iné hrany v hyperkocke nie sú.

Označenie 7. *Poloha agenta je vyjadrená súradnicami v jednotlivých dimenziách kocky. v každej dimenzii môže nadobúdať jednu z dvoch súradníc. Označme ich* 0 *a* 1. *Pri prechode po hrane agent preklopý jednu z týchto súradníc v práve jednej dimenzii na druhú.*

Lema 13. *V ľubovoľnej postupnosti hrán medzi vrcholmi hyperkocky spojenými hranou, sa nachádza párny počet hrán v každej dimenzii okrem dimenzie v ktorej sa tieto vrcholy líšia a teda v nej vedie hrana medzi nimi.*

Dôkaz. Takéto dva vrcholy sa líšia v práve jednej dimenzii. Ak agent zmenil pri pohybe svoju polohu v nejakej dimenzii nepárny počet krát, tak je v tejto dimenzii v inej polohe ako bol na začiatku. Ak to nie je dimenzia v ktorej sa líši koncový vrchol od začiatočného - agent sa nenachádza v koncovom vrchole, lebo polohy agenta a tohto vrcholu nie sú rovnaké. □

Označenie 8. *Ak viem súradnice vrchola, môžem postupnosť ťahov agenta zapísať ako postupnosť zmien jeho súradníc v jednotlivých dimenziách.*

Ak sa nachádza v takejto postupnosti medzi vrcholmi, medzi ktorými vzniká portová hrana dvakrát hneď za sebou číslo nejakej dimenzie, znamená to, že agent šiel tam a späť a tieto dva prvky môžeme z postupnosti vylúčiť, lebo je to z pohľadu tejto dvojice vrcholov zbytočný pohyb. Ak takto zredukujeme niektorú postupnosť, získame inú, v ktorej ide agent po podmnožine hrán z pôvodnej a na rovný alebo menší počet pohybov.

Poznámka 5. *Pripomeniem, že hľadáme taký sled, že počet portových hrán medzi jeho vrcholmi je na daný počet značiek pre porty a počet portových hrán, ktoré vzniknú medzi jeho vrcholmi, najkratší možný.*

Ak má agent dostatok značiek pre porty, nezaujíma nás poradie v akom vrcholy postupnosti navštívil. Úlohu teda môžeme zredukovať na hľadanie takej množiny vrcholov veľkosti m, ktorá tvorí zo všetkých takto veľkých množín najmenší rez od zvyšku grafu. Pri konštantnom stupni vrcholov to znamená, že v takej vedie najviac hrán medzi vrcholmi v tejto množine. Keď sa pozrieme na vrcholy a pravidlá ich spájania hranami, zistíme, že tieto vedú len v nejakej kocke a čím viac rozmerov kocka má, tým viac hrán vedie medzi jej vrcholmi. Teda aj v n-rozmernej kocke povedie najviac hrán medzi vrcholmi takej množiny, ktorej indukovaný graf je izomorfný kocke. Vrchol k-rozmernej kocky sa nachádza v k (k-1)-rozmerných kockách, ktoré sú jej podgrafmi. Teda budeme hľadať množinu m bodov, ktoré sa dajú pokryť kockami čím vyššieho rozmeru a nemusíme si všímať kocky ktoré sú podgrafmi týchto kociek. tieto kocky sa však môžu prekrývať.

Lema 14. *S každou hyperkockou v ktorej vrchol neleží ho spája najviac jedna hrana. Ktorá vedie v inej dimenzii ako sú dimenzie tejto hyperkocky.*

Dôkaz. Každý vrchol hyperkocky má hrany vo všetkých jej dimenziách. Vo zvyšných dimenziách sú vrcholy hyperkocky totožné, lebo sa dva susedné líšia práve v jednej. Vrchol mimo hyperkocky spojený s vrcholom v nej sa od neho líši v inej dimenzii a v tejto sa líši teda aj od všetkých ostatných. Zároveň je vo všetkých dimenziách hyperkocky totožný s vrcholom z nej a teda sa lííši v aspoň jednej dimenzii hyperkocky od jej ostatných vrcholov a má teda s nimi aspoň dve rodielne dimenzie. Nemôže ho preto s nimi spájať hrana. □

Lema 15. *Bijektívna substitúcia dimenzií na všetkých vrcholoch podgrafu hyperkocky zachováva hrany medzi vrcholmi. Rovnako aj preklopenie súradníc v niektorej dimenzii.*

Dôkaz. Ak sa dva vrcholy líšia práve v jednej dimenzii, budú sa v práve jednej dimenzii líšiť aj po tejto substitúcii, lebo dimenzie v ktorých boli rovnaké sa zobrazia na rovnaké a tá v ktorej sa kíšili sa u oboch zobrazí na rovnakú a budú sa v nej líšiť aj naďalej;

teda medzi nimi povedie hrana. Pri preklopení súradníc sa medzi vrcholmi zachovajú rovnosti a rozdiely v danej dimenzii, teda aj hrany. □

Lema 16. *Vrcholy súvislého podgrafu hyperkocky s počtom vrcholov m sa navzájom líšia v najviac m-1 súradniciach.*

Dôkaz. Vezmime kostru tohto podgrafu. Táto má m-1 hrán. Každá vedie v niektorej dimenzii. Cesta v tejto kostre udáva v ktorých dimenziách sa daná dvojica jej koncových vrcholov líši. Všetky cesty v tejto kostre idú cez najviac m-1 rôznych dimenzií, čo je jej počet hrán. V ostatných dimenziách sa dané vrcholy nelíšia, majú v nich teda navzájom rovnaké súradnice. □

Lema 17. *Podmnožina m bodov hyperkocky, ktoré indukujú graf s najvyšším počtom hrán je súvislá.*

Dôkaz. Nech existujú dva komponenty. Na jednom z nich urobíme postupné popreklápanie súradníc tak, aby susedil niektorý jeho bod s bodom v druhom komponente a teda medzi nimi viedla hrana. Čiže vyberieme jednu dimenziu v ktorej sa líšia a vo všetkých ostatných dimenziách, kde majp rozdielne súradnice preklápame postupne súradnice jedného z vrcholov. V prípade, že počas tohot postupu vznikne hrana spájajúca vyhrané dva komponenty, skončíme skôr. Pôvodné hrany sa zachovajú a jedna nová pribudne, čo je spor s najvyšším počtom hrán. Tieto kroky môžeme urobiť bez toho, aby sa dva vrcholy zobrazili na seba. Ak budeme robiť preklápanie súradníc po jednotlivých dimenziách, v tom prípade, ak by sa nejaký vrchol transformovaného komponentu podgrafu zobrazil na niektorý vrchol druhého komponentu, musela by medzi nimi viesť predtým hrana, teda stav v ktorom by boli komponenty prepojené a preklápanie by teda už nepokračovalo. □

Potrebujeme vybrať m vrcholový indukovaný podgraf hyperkocky tak, aby počet hrán v ňom bol maximálny možný.

Lema 18. *Vrcholy podgrafu hyperkocky sa dajú rozdeliť na dve množiny A,B podľa súradnice vo vybranej dimenzii. Počet hrán vedúcich z vrcholov v jednej množine do vrcholov v druhej množine je rovný alebo menší ako mohutnosť menšej z nich. Čiže* $|\{e|e = (a, b); a \in A; b \in B\}| <= min\{|A|, |B|\}$

Dôkaz. Každý vrchol má v jednej dimenzii najviac jednu hranu a hrana vedie v danej dimenzii medzi dvomi vrcholmi vtedy ak je to jediná dimenzia v ktorej majú rozdielne súradnice. V inej dimenzii medzi A a B hrana viesť nemôže, lebo by musela viesť medzi vrcholmi líšiacimi sa v najmenej dvoch dimenziách - v dimenzii hrany a v dimenzii podľa súradníc ktorej sú A a B vytvorené. Teda hrany medzi A a B môžu viesť len v jednej dimenzii a do menšej z A,B môže viesť najviac toľko hrán v jednej dimenzii, koľko má vrcholov. □

Lema 19. *Keď vezmeme z podgrafu hyperkocky len vrcholy, ktoré majú hranu v jednej vybranej dimenzii a označíme nimi indukovaný podgraf G. Vieme vrcholy G rozdeliť na dva izomorfné indukované podgrafy hyperkocky G_1 a G_2 podľa súradnice vrchola v tejto dimenzii.*

Dôkaz. Vrcholy, medzi ktoými vedie hrana sa líšia v práve jednej dimenzii, v ostatných dimenziách sú rovnaké. Keď teda vezmeme hrany vedúce medzi G_1 a G_2 a vytvoríme podľa nich bijektívne zobrazenie medzi týmito podgrafmi tak, aby sa dva vrcholy medzi ktorými vedie hrana vo vybranej dimenzii zobrazili na seba, dostaneme izomorfizmus. Ak totiž viedla medzi dvomi vrcholmi hrana, bude hrana viesť aj medzi ich obrazmi, lebo tieto majú rovnaké súradnice ako ich vzory vo všetkých dimenziách okrem jednej, kde sa oba obrazy od svojich vzorov líšia a teda majú súradnicu v tejto dimenzii rovnakú; obaja sú v komponente, kde majú túto súradnicu rovnakú všetky vrcholy. Ak sa teda líšili v jednej súradnici vzory, budú sa aj obrazy a ak sa líšili vzory vo viacerých súradniciach, budú sa v rovnakých súradniciach líšiť aj obrazy. Zobrazenie je teda izomorfizmus. □

Lema 20. *Existuje indukovaný podgraf G hyperkocky na m vrcholoch, ktorý je možné postupne rozdeliť podľa dimenzií až po jednovrcholové indukované podgrafy tak, že v každom kroku rozdelíme vrcholy každého podgrafu s počtom vrcholov aspoň dva do dvoch podmnožín A a B. Takých, že $|A| - |B| <= 1$.*

Dôkaz. Dôkaz bude konštrukčný. Budeme postupne deliť m vrcholovú množinu i ďalšie z nej vzniknuté množiny na menšie, až kým nedostaneme m jednoprvkových množín vrcholov. Pritom budeme postupne určovať súradnice jednotlivých vrcholov.

Budeme postupovať postupne po dimenziách, až kým budúd všetky množiny jednoprvkové. Vezmeme množinu M, rozdelíme ju na dve časti M_1, M_0 také že $|M_1|-|M_0| <= 1$. Vrcholy v M_1 budú mať v tejto dimenzii súradnicu 1 a vrcholy v M_0 budú mať v tejto dimenzii súradnicu 0. Ak je množina jednoprvková a delenie niektorých iných ešte pokračuje, dostane v každej ďalšej dimenzii súradnicu 1. Ak má kocka viac dimenzií ako je potrebných, aby obsiahla všetkých m vrcholov podgrafu, tak môžu dostať tieto vrcholy v týchto zvyšných dimenziách ľubovoľné súradnice, ale pre všetky vrcholy musí byť súradnica v každej z týchto dimenzií rovnaká. Tým sme zistili súradnice vrcholov vo všetkých dimenziách, podľa ktorých podgraf delíme. Pre účely tejto vety na súradniciach ostatných dimenzií nezáleží a môžu byť ľubovoľné. □

Lema 21. *Pri delení z vety 20 vedie medzi množinami vrcholov M_1 a M_0, ktoré vznikli rozdelením jednej množiny M podľa tohto delenia, práve toľko hrán, ako je mohutnosť* *menšej z týchto množín. Pričom, ak má kocka viac dimenzií ako je potrebných, aby obsiahla všetkých m vrcholov podgrafu, tak môžu dostať tieto vrcholy v týchto zvyšných dimenziách ľubovoľné súradnice, ale pre všetky vrcholy musí byť, na rozdiel od predošlej vety, súradnica v každej z týchto dimenzií rovnaká.*

Dôkaz. Stačí ukázať, že menší z indukovaných podgrafov má vo väčšom izomorfný obraz podľa vety 19. Vtedy vedie z každého vrchola v menšom podgrafe hrana do nejakého vrchola vo väčšom podgrafe a týchto hrán je presne toľko, koľko je vrcholov menšieho podgrafu. Teda potrebujeme dokázať, že vo väčšom podgrafe existuje skupina vrcholov, ktorej budú súradnice v dimenziách pridelené rovnako ako vrcholom v menšom podgrafe. Prideľovanie súradníc je deterministické, teda rovnako veľkým množinám vrcholov v rovnakej fáze budú poprideľované súradnice vo zvyšných dimenziách rovnako. Na menší a väčší podgraf sa rozdelia grafy s nepárnym počtom vrcholov. Použijeme úplnú indukciu. Báza indukcie bude množina veľkosti tri. Menšie podgrafy sú už len veľkosti dva a jedna. Jednovrcholový sa už ďalej nedelí a dvojvrcholový sa rozdelí na dva jednovrcholové. Predpokladáme, že pre delenie množín s mohutnosťou menšou ako n platí veta. Dokážeme, že potom musí platiť aj pre množinu M mohutnosti n: Ak je n párne, dá sa množina rozdeliť na dve množiny rovnakej mohutnosti, ktorých vrcholom sa zvyšné dimenzie poprideľujú rovnako. Teda každý vrchol v jednej bude mať iný vrchol v druhej množine s ktorým sa bude líšiť v práve jednej dimenzii a to v tej podľa ktorej sme M rozdelili. Ak n je nepárne, vzniknú dve množiny M_1, M_2. Obe z nich už podľa IP spĺňajú tvrdenie vety. Rozdelia sa každá na dve množiny, pričom z množiny s párnou mohutnosťou vzniknú dve rovnako veľké množiny M_{11},M_{12} a množina M_{21} tejto veľkosti vznikne aj z množiny s nepárnou mohutnosťou. Druhá množina M_{22} z tejto množiny bude mať o prvok viac alebo menej. Prvé tri množiny sa už budú deliť rovnako, teda viem povyberať trojice vrcholov, každý z inej množiny, ktoré budú mať vo zvyšných dimenziách rovnaké súradnice. Medzi dvomi vrcholmi z M_{11},M_{12}, označme ich m_{11},m_{12}, povedie hrana, lebo sa líšia práve v jednej dimenzii. Tretí vrchol, označme ho m_{21} bude spojený hranou s niektorý z tých dvoch, pretože pri druhom delení jeden z nich dostal rovnakú súradnicu ako on v dimenzii druhého delenia a teda sa líšia iba v dimenzii prvého delenia množiny M. Ak je prvok m_{21} spojený s prvkom z M_{22}, označme ho m_{22}, potom je m_{22} spojený aj s tým prvkom z dvojice m_{11},m_{12}, s ktorým nie je spojený m_{21}. Prvok m_{22} neexistuje iba vtedy, ak $|M_{22}| < |M_{21}|$. Ak $|M_{22}| > |M_{21}|$ potom v M_{22} majú všetky vrcholy z M_{21} izomorfný obraz a teda aj príslušná časť z párnej množiny. Z horeuvedeného vyplýva, že menšia množina z prvého delenia má obraz, izomorfný a spĺňajúci podmienky vety 19, vo väčšej množine vzniknutej z tohto delenia a keďže medzi vzorom a obrazom vedie hrana, veta platí. □

Lema 22. *Maximálny počet hrán, ktoré môžu viesť v podgrafe hyperkocky indukovanom* *m bodmi je daný postupnosťou a_m, kde $a_1 = 0; a_2 = 1$; a pre $\forall i < 2; a_i = \lfloor \frac{i}{2} \rfloor + a_{\lfloor \frac{i}{2} \rfloor} + a_{\lceil \frac{i}{2} \rceil}$*

Dôkaz. V dôkaze použijeme úplnú indukciu. Báza indukcie je očividná: hyperkocka nemá slučky a teda v jej jenovrcholovom indukovanom grafe nevedie hrana a hyperkocka nemá násobné hrany, teda v jej indukovanom dvojvrcholovom podgrafe môže viesť najviac jedna hrana. Predpokladáme, že pre všetky členy s indexom menším ako m platí, že udávajú najväčší možný počet hrán v indukovanom podgrafe hyperkocky zo všetkých množín jej bodov, ktorých mohutnosť je rovná indexu daného člena postupnosti. ďalej predpokladáme, že pre každého člena postupnosti s menším indexom platí tvrdenie vety. Ukážeme, že existuje indukovaný podgraf hyperkocky s m bodmi s najvyšším možným počtom hrán G taký, ktorý niektorá dimenzia delí na polovice G_1 a G_2, teda na množiny vrcholov, ktorých mohutnosti sa líšia najviac o jedna. To vyplýva z toho, že neexistuje taký podgraf hyperkocky, ktorý by delila inak a mal by zároveň vyšší počet hrán. Nech taký podgraf G' existuje. Potom ho vybraná dimenzia delí na podgrafy s počtom bodov a_i, a_j. Keďže sú obe neprázdne, sú menšie ako m a teda pre ne platí indukčný predpoklad. $a_i = \lfloor \frac{i}{2} \rfloor + a_{\lfloor \frac{i}{2} \rfloor} + a_{\lceil \frac{i}{2} \rceil}$ a $a_j = \lfloor \frac{j}{2} \rfloor + a_{\lfloor \frac{j}{2} \rfloor} + a_{\lceil \frac{j}{2} \rceil}$ Bez ujmy na všeobecnosti, nech i < j. Aby sme zjednodušili indexovanie označme $a_{\lfloor \frac{i}{2} \rfloor} = a_{i1}; a_{\lceil \frac{i}{2} \rceil} = a_{i2}; a_{\lfloor \frac{j}{2} \rfloor} = a_{j1}$ a $a_{\lceil \frac{j}{2} \rceil} = a_{j2}$. Je zrejmé, že $a_{i2} - a_{i1} <= 1$ a $a_{j2} - a_{j1} <= 1$. Ak teda rozdelíme G podľa dimenzie, ktorá ho delí na G_1 a G_2, ich mohutnosti budú bez ujmy na všeobecnosti $a_{i2} + a_{j1}; a_{j2} + a_{i1}$. Z IP zároveň platí, že G_1 a G_2 majú aspoň tak dobré rozdelenie ako na množiny veľkostí a_{i2}, a_{j1} respektíve a_{j2}, a_{i1}. Počet hrán, ktoré vedú vo podgrafe G' je teda $a_{m'} = i + \lfloor \frac{i}{2} \rfloor + \lfloor \frac{j}{2} \rfloor + a_{i1} + a_{i2} + a_{j1} + a_{j2}$. Je zrejmé, že menšia z množín na ktoré sa rozdelí G bude mať veľkosť $\lfloor \frac{m}{2} \rfloor$ a v týchto podmnožinách vedie aspoň $a_{i1} + a_{j2} + \lfloor \frac{i}{2} \rfloor$ respektíve $a_{j1} + a_{i2} + \lceil \frac{i}{2} \rceil$ hrán. Čiže $a_m >= \lfloor \frac{m}{2} \rfloor + a_{i1} + a_{j2} + \lfloor \frac{i}{2} \rfloor + a_{j1} + a_{i2} + \lceil \frac{i}{2} \rceil$ Z toho si ľahko ukážeme, že $a_m >= a_{m'}$. $a_m - a_{m'} >= \lfloor \frac{m}{2} \rfloor + \lceil \frac{i}{2} \rceil - i - \lfloor \frac{j}{2} \rfloor$ a keďže $\lfloor \frac{m}{2} \rfloor = \lfloor \frac{i+j}{2} \rfloor >= \lfloor \frac{i}{2} \rfloor + \lfloor \frac{j}{2} \rfloor$ potom platí, že $a_m - a_{m'} >= 0$. Tým sme dokázali, že neexistuje lepšie rozdelenie podgrafu hyperkocky indukovaného m vrcholmi ako rozdelenie na polovice. Z predošlých viet vyplýva, že takéto rozdelenie je možné. □

Lema 23. *Zoberme jednu dimenziu a rozdeľme body na dve množiny, podľa toho, či v nej majú 1 alebo 0. Platia dve veci: medzi týmito dvomi množinami vedie najviac toľko hrán, koľko má menšia z nich vrcholov; v iných dimenziách už medzi týmito vrcholmi hrany nevedú.*

Dôkaz. Prvá vec vyplýva z toho, že každý vrchol má v jednej dimenzii najviac jednu hranu (pri nekompletnej hyperkocke, pri kompletnej práve jednu). Druhá vyplýva z toho, že hrana spája vrcholy, ktoré sa líšia práve v jednej súradnici a vedie v dimenzii tejto súradnice. Ak by mala viesť medzi týmito dvomi množinami hrana v nejakej inej súradnici, musela by spájať vrcholy, ktoré sa líšia v najmenej dvoch súradniciach (v tej ktorá rozdeľuje vrcholy na dve množiny a v tej v ktorej vedie táto hrana), čo nemôže. □

- TODO lema

Lema 24. *Majme množinu množín vrcholov, označme ju M. Nech existuje taká postupnosť spájania týchto množín, ktorého súčet bude najvyšší možný. Nech sa v ňom nenachádza spojenie najmenšej množiny s inou v prvom kroku, ukážeme že existuje poradie, kde sa nachádza o krok skôr, ktoré je aspoň také dobré alebo pôvodné nie je najlepšie.*

Dôkaz. Ak sa najmenší prvok spája s množinou, ktorá bola v M, vieme spojenie minima s ňou presunúť do úplne prvého kroku spájania. Ak sa spája s množinou, ktorá vznikla spojením dvoch množín, premiestnime krok pripojenia minima hneď za krok pri ktorom táto množina vznikla. Označme množiny z ktorých táto množina vznikla A,B a mohutnosť minima označme C, |A| = a; |B| = b; |C| = c. Bez ujmy na všeobecnosti a =< b. Spájanie nebolo optimálne, lebo v jeho súčte sa nachádza c+a, keby sa však spojila najprv množina A s C tak by bol celkový súčet väčší, lebo by obsahoval buď c+b alebo c+(a+c).

□

Každý vrchol je 0-rozmernou kockou sám o sebe. Všetky hrany, ktoré medzi nimi môžu viesť v prvej dimenzii, v druhej, tretej atď. ... TODO výde z toho hyperkocka a elegantnejšie ako predtým :) – stále sa mi to mezdá dokázané, že iný výber vrcholov nedá viac hrán ...

Treba maximalizovať počet dvojíc líšiacich sa v práve jednej dimenzii ... V každej dimenzii má vrchol max jednu hranu

To že medzi vrcholmi vedie hrana znamená, že sú spolu v nejakej hyperkocke. V čím väčšej sú, tým viac hrán z nich ide.

Ukážeme že maximalizácia n-rozmerných kociek maximalizuje počet n-1 rozmerných. Vezmeme najväčšiu maximálnu disjunktnú množinu n-1 rozmerných kociek. Aké je poprepájanie týchto kociek hranami pri ktorom je najviac n-1 rozmerných kociek?

Ak vedie hrana medzi vrcholmi a tieto sú spojené každý hranami rovnakej dimenzie s inými dvomi vrcholmi, vedie hrana aj medzi týmito dvomi vrcholmi. Je to dané tým, že ak odstránime dimenziu v ktorej vedie hrana, tieto vrcholy sa zobrazia na seba.

Kocka vyššej dimenzie vznikne, ak spojíme dve n-1 rozmerné kocky hranami podľa nejakého izomorfného zobrazenia dostaneme n-rozmernú kocku, ktorá má $nchoosen - 1$ n-1 rozmerných kociek.

Dôkaz. □

Izolované vrcholy -> jednorozmerné kocky -> dvojrozmerné kocky -> ... vždy dve spojíme na jednu s vyšším rozmerom ...

Čiže trade-off medzi počtom značiek a pohybov ... stačí jedna návšteva vrcholu ... otvorí pri nej všetky porty ... počet značiek = počet mimokostrových (portových) hrán ...

Čím vyššia dimenzia kocky, tým je menej kostrových hrán oproti portovým -> stačí menej pohybov na požitie daného množstva značiek.

Ak vrcholov nie je presne toľko, ako v nejakej n-rozmernej kocke tj. 2^n tak ich vyberáme z vrcholov najmenšej kocky s väčším počtom vrcholov tak, že vezmeme vrcholy nejakej (n-1) rozmernej kocky, ktorá je jej podgrafom, v ďalšom kroku zvyšok vrcholov vyberieme z ešte nepoužitých tak, aby tvorili kocku čo najväčšej dimenzie a tak ďalej, kým nevyberieme všetky vrcholy. Každá n-rozmerná kocka sa dá rozdeliť na dve disjunktné (n-1)-rozmerné kocky. Každý vrchol takejto kocky je spojený s práve jedným vrcholom z druhej kocky. Agent teda vždy nájde kocku ktorá je najmenšia taká, že má viac vrcholov. Jednu z týchto kociek pokryje a zvyšné vrcholy umiestni do druhej kocky rekurzívnym spôsobom.

Týmto sme vyriešili hľadanie postupnosti k vrcholov s nyjväčším počtom hrán medzi nimi. Teraz treba zistiť, ako veľká musí byť množina značiek potrebná na konštrukciu týchto hrán. Ak má agent minúť čo najmenej značiek pri tvorbe portových hrán, treba aby ostali porty otvorené čo najkratšie - čas meriame pohybmi agenta.

Počet značiek, ktroé agent potrebuje pre konštrukciu n-rozmernej kocky je suma dva na i minus jedna pre i od jedna po n. (po konštrukcii jednej kocky ostávajú otorené porty do druhej a agent potrebuje rovnaký počet značiek ako na predošlú ... - lenže postupne sa uvoľnujú) pozn. bude to teda asi # portov pre štvrtinovú kocku + dva na i munus jedna

7.1 2-rozmerná kocka

Elementárny prípad. Stačí jedna portová značka a tri pohyby - operácue NH.

7.2 3-rozmerná kocka

7.2.1 1 značka

Pri použití jednej zančky na porty vie agent trojrozmernú kocku zostrojiť na 15 pohybov: OP, NH, NH, NH, OP, OP, S1, NH, NH, OP, OP, S2, S2, NH, OP, OP, S1, S1, NH, OP, OP, S1, S2, S3, OP. Na menej sa to nedá. Agent potrebuje skonštruovať 5 mimokostrových hrán a na každú potrebuje aspoň 3 pohyby.

7.2.2 2 značky

Trojrozmernú kocku pri použití dvoch značiek na porty dokáže agent zostrojiť na 11 pohybov a menej sa nedá. Operácie agenta: OP1, NH, NH, OP2, NH, OP1, NH, OP1, NH, OP2, NH, OP2, NH, OP1, OP1, S1, S1, S2, OP1, S1, OP2 Na menej sa nedá, pretože porty treba otvoriť medzi dvojicou vrcholov líšiacich sa v práve jednej súradnici. Teda medzi otovrením portov jednej hrany potrebuje agent aspoň dve preklopenia na rovnakej súradnici a jedno na inej. Vie využiť najviac jedno preklopenie, ktoré nastalo pri konštrukcii predošlej hrany, čiže na konštrukciu hrany potrebuje najmenej dva ďalšie pohyby. Na konštrukciu prvej hrany potrebuje tri pohyby. $3 + 2 * 8 = 11$

7.2.3 3 značky

Trojrozmernú kocku agent za použitia troch portových značiek zostrojí na 9 pohybov. Na menej sa nedá. Operácie agenta: OP1, OP3, NH, NH, OP2, NH, OP1, NH, OP1, NH, OP2, NH, OP2, NH, OP1, OP3, S3(?), S1, OP2 S použitím menšieho množstva pohybov to agent urobiť nemôže. Kým vznikne prvá portová hrana - značka sa uvoľní na ďalšie použitie,vytvorí agent tri nové vrcholy. Teda je vo štvrtom vrchole. Súčet stupňov predošlých troch je deväť, z čoho päť zaberú nové hrany. To znamená, že do jedného z týchto vrcholov sa agent bude musieť vrátiť otvoriť port alebo z neho vytvoriť novú hranu. Keby tento vrchol bol tretí podľa vzniku, tak sa doň agent môže vrátiť na jeden ťah. Tu môžu nastať dve možnosti: pôjde tam otvoriť port alebo vytvoriť novú hranu. Ak otvoriť port, bude mu chýbať značka na štvrtom vrchole podľa vzniku a bude sa musieť vrátiť aj do neho. Ak otvorí port s novouvoľnenou značkou na štvrtom vrchole a z tretieho urobí NH. Ocitne sa vo vrchole v ktorom nemôže otvoriť port, lebo sa nelíši v práve jednej súradnici od žiadneho z vrcholov v ktorých sú porty pootvárané. Teda v oboch prípadoch potrebujem ďalšie dva pohyby, ktoré nie sú operáciou NH. Ak sa agent nevráti hneď do vrchola, v ktorom neotvoril port, bude musieť prejsť do niektorého z jeho dvoch susedov (lebo z nich už odišiel a z iného vrchola sa tam nedostane). To sú dva pohyby okrem operácií NH. Počet operácií NH je konštantný pre daný graf a rovná sa počtu hrán kostry. V tomto prípade je to sedem. Teda agent vybavený tromi rôznymi značkami na porty potrebuje aspoň deväť krokov na konštrukciu trojrozmernej kocky.

7.2.4 4 značky

4 rôzne značky portov agentovi na konštrukciu trojrozmernej kocky stačia: OP1, OP2, NH, OP3, NH, OP4, NH, OP1, NH, OP1, NH, OP4, NH, OP3, NH, OP2, OP1 Agent vzkoná sedeme pohybov, čo je nevyhnutné minimum na konštrukciu trojrozmernej kocky. Pridanie ďalších značiek už teda nič nevylepší.

7.3 4-rozmerná kocka

1 značka Ak chcem skonštruovať portové hrany použitím troch pohybov na hranu a žiadne ďalšie pohyby, tak musím vo vrchole, kam otvorením portu vznikne nová portová hrana, otvoriť port znovu. Portové hrany teda budú tvoriť súvislý ťah.

Všetkých hrán v štvorrozmernej kocke je 32, pričom kostrových je 15. Portových hrán je teda 17. Sled portových hrán sa nemôže pretínať, pretože nemôže existovať vrchol do ktorého nevedie kostrová hrana a každý vrchol má stupeň 4. Keďže stupeň vrcholov je párny, kostru treba zostrojiť tak, aby dva z nich mali nepárny stupeň a zvyšné párny. Čo je nutná podmienka zostrojenia jediného ťahu zo všetkých zvyšných hrán. Kostra má aposň dva listy a stupeň listov je nepárny. Ak teda kostra nemá mať väčší počet listov, musia mať zvyšné vrcholy stupeň dva. Kostra je teda cesta. Ukážeme, že nie je možné zostrojiť štvorrozmernú kocku na $3 * 17 = 51$ pohybov. Vrcholy v kotorých kostra-cesta začína majú tri portové hrany. Aby sme použili len tri pohyby na jednu portovú hranu, potrebujeme začať portové hrany konštruovať v jednom z týchto vrcholov. v týchto vrcholoch začína aj cesta-kostra. Keďže kostra vzniká tak, že na už existujúcom vrchole vykoná agent operáciu NH, v každom momente je práve jeden vrchol v ktorom môže byť agent, ak má skonštruovať graf na 51 pohybov. Tým však, že agent odišiel z prvého vrchola a tomu chýbajú ešte dve hrany, sa doň bude musieť vrátiť a otvoriť v ňom port. V konštruovanom grafe (medzivýsledku) sa však agent od prvého vrchola vzďaľuje každou portovou hranou. Pri skonštruovaní posledného vrchola bude teda od prvého v medzivýsledku konštrukcie ďalej ako tri a teda nie je schopný skonštruovať graf týmto spôsobom. Vrchol do ktorého ide agent konštruovať portovú hranu musí byť v medzivýsledku vzdialený práve tri od druhého vrchola s otvoreným portom a súčsne sa musí líšiť v práve jednej dimenzii. Pri tomto postupe žiadny takýto vrchol k poslednému skonštruovanému vrcholu v momente otvorenia portu nie je a jediný, ktorý sa tak dá vytvoriť je pomocou prechodu po novej portovej hrane a dvoch opráciách NH, alebo po troch operáciách NH. Ak použije prvý spôsob, kým sa dá, dostane sa do bodu z ktorého sa už tento nedá uplatniť a už nebude schopný skonštruovať ďalšiu hranu na tri pohyby. Pri použití druhého spôsobu z inicializačného vrchola trikrát sa dostane agent do situácie, kefy je od inicializačného vrchola vzdialený na medzivýsledku o tri hrany a zároveň sa ich polohy líšia v práve jednej dimenzii. Takže agent môže skonštruovať zvyšné dve hrany pre inicializačný vrchol. (TODO mohol by to urobiť aj neskôr, ale nejde to, lebo štvorrozmerná kocka má len 15 vrcholov a už by mu nevydalo vytvoriť tri portové hrany pomocou operácií NH) Keď má agent z vrcholu vytvoriť portovú hranu a odchádza z neho operáciou NH, odíde po jednej dimenzii a portová hrana sa otvorí v druhej. Teda vytvorí hrany v dvoch dimenziách vrcholu. Po skonštruovaní prvej portovej hrany má vrchol v ktorom agent skončí ešte "voľné"dve dimenzie. Z jednej má ísť portová hrana a z druhej kostrová. Teda po jednej agent odíde a v druhej sa po troch pohyboch bude líšiť. Rovnakáá situácia nastane po vytvorení druhej portovej hrany. Tretia portová hrana musí mať rovnakú dimenziu ako prvá, aby sa agent po jej konštrukcii líšil s inicializačným vrcholom v práve jednej dimenzii. štvrtá portová hrana je teda tiež daná. Dimenzie sú navzájom zameniteľné, takže na ich výbere nezáleží. (TODO Začiatok konštrukcie štvorrozmernej kocky je deterministický - možno je celá konštrukcia ...)

Literatúra

[DFK+05a] Shantanu Das, Paola Flocchini, Shay Kutten, Amiya Nayak, and Nicola Santoro. Distributed exploration of anonymous graphs by multiple agents. 2005.

[DFK+05b] Shantanu Das, Paola Flocchini, Shay Kutten, Amiya Nayak, and Nicola Santoro. Distributed exploration of unlabelled graphs by multiple agents. 2005.

[DKM12] Stefan Dobrev, Rastislav Královič, and Euripides Markou. Online graph exploration with advice. In *Structural Information and Communication Complexity*, pages 267–278. Springer, 2012.

[DP90] Xiaotie Deng and Christos H Papadimitriou. Exploring an unknown graph. In *Foundations of Computer Science, 1990. Proceedings., 31st Annual Symposium on*, pages 355–361. IEEE, 1990.

[FIP+05] Pierre Fraigniaud, David Ilcinkas, Guy Peer, Andrzej Pelc, and David Peleg. Graph exploration by a finite automaton. *Theoretical Computer Science*, 345(2–3):331 – 344, 2005. Mathematical Foundations of Computer Science 2004 Mathematical Foundations of Computer Science 2004.

[MMS11] Nicole Megow, Kurt Mehlhorn, and Pascal Schweitzer. Online graph exploration: new results on old and new algorithms. In *Automata, Languages and Programming*, pages 478–489. Springer, 2011.

[PY91] Christos H Papadimitriou and Mihalis Yannakakis. Shortest paths without a map. *Theoretical Computer Science*, 84(1):127–150, 1991.

[RE97] Grzegorz Rozenberg and Hartmut Ehrig. *Handbook of graph grammars and computing by graph transformation*, volume 1. World Scientific Singapore, 1997.

[Rol79] Hans-Anton Rollik. Automaten in planaren graphen. In *Theoretical Computer Science 4th GI Conference*, pages 266–275. Springer, 1979.