



KATEDRA INFORMATIKY
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY
UNIVERZITA KOMENSKÉHO, BRATISLAVA

ŠÍRENIE SPRÁV V GRAFOCH
S PROPORCIONÁLNYM POČTOM CHYBNÝCH LINIEK

(Diplomová práca)

MIRKO ZIBOLEN

Vedúci: Doc. RNDr. Rastislav Kráľovič, PhD.

Bratislava, 2008

**Šírenie správ na grafoch
s proporcionálnym počtom chybných liniek**

DIPLOMOVÁ PRÁCA

Mirko Zibolen

**UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY
KATEDRA INFORMATIKY**

Informatika

Vedúci diplomovej práce
Doc. RNDr. Rastislav Kráľovič, PhD

BRATISLAVA 2008

Čestne prehlasujem, že som túto diplomovú prácu vypracoval samostatne s použitím citovaných zdrojov.

Ďakujem vedúcemu diplomovej práce Doc. RNDr. Rastislavovi Královičovi, PhD. za cenné rady, usmernenia a trpezlivosť pri mojich otázkach.

Abstrakt

Zibolen Mirko, Šírenie správ v grafoch s proporcionálnym počtom chybných liniek. Diplomová práca, Katedra informatiky fakulty matematiky, fyziky a informatiky Univerzity Komenského, vedúci diplomovej práce Doc. RNDr. Rastislav Kráľovič, PhD. Bratislava 2008, 72 strán.

Práca sa zaoberá analýzou rôznych aspektov šírenia správ v sieťach s chybnými linkami. Používaný model pripúšťa v každom kroku počet chybných liniek proporcionálne závislý od počtu poslaných správ. Pozornosť je sústredená najmä na všeobecné algoritmy šírenia správ, analyzujeme ich správanie sa na rôznych typoch grafov. Práca sumarizuje známe poznatky o greedy algoritme a hľadá prípady, v ktorých poskytuje zlé výsledky. Prezentujeme algoritmus spätného šírenia, ktorý garantuje asymptoticky optimálny čas šírenia správy na acyklických grafoch. Hlbšie si všimame aj ďalšie triedy grafov - stromy s konštantnou excentricitou iniciátora a hyperkocky - ktoré sú analyzované tak z pohľadu greedy algoritmu, ako aj z pohľadu optimálneho algoritmu.

Kľúčové slová: šírenie správ, proporcionálny počet chybných liniek, algoritmus spätného šírenia, algoritmus dopredného šírenia, odolnosť voči chybám

Abstract

In this work we discuss various aspects of broadcasting in point-to-point networks with faulty links. In the model we use the number of faulty links in a given step proportionally depends on the total number of links used in that step. Our attention is focused on generic broadcasting algorithms and their impact on broadcasting time for various networks. We summarize known results about greedy algorithm and search for bad-case examples. We present a backward-distance algorithm, which guarantees an asymptotic optimal broadcast time for every acyclic graph. In the last section some special graphs are more discussed - tree graphs with constant eccentricity of initial vertex and hypercubes.

Predhovor

Oblasť distribuovaných systémov už dávno nepatrí iba do teoretickej roviny. Výskum v tejto oblasti sa premieta do konkrétnych implementácií a pomáha riešiť široké spektrum najrôznejších úloh. Aj na Fakulte matematiky, fyziky a informatiky má štúdium distribuovaných systémov pevné korene, každoročne si ho za svoju špecializáciu (v bloku algoritmy a paralelné výpočty) volí mnoho študentov. Vedecky aktívni pracovníci prisievajú k rozširovaniu poznatkov v tejto bezpochyby aktuálnej oblasti.

Jednou z oblastí štúdia je aj šírenie správ v grafoch s chybnými linkami. Hoci dynamickými chybami liniek v distribuovanom systéme sa zaoberalo mnoho ľudí na celom svete, väčšina ľudí pracovala s modelom či už pravdepodobnostným, alebo ohraničeným konštantou. Až pred pár rokmi vznikol práve na fakulte model, ktorým sa budeme zaoberať v tejto práci. Snahou modelu je skombinovať výhody oboch princípov. V prvom rade je očakávateľné, aby počet chýb rástol spoločne s počtom odoslaných správ. Ohraničený model (v kontraste s pravdepodobnostným) zasa umožňuje formulovať tvrdenia o najhorších prípadoch, garantovať čas šírenia správy bez ohľadu na pravdepodobnosť.

Uvedomujeme si aj nedostatky predloženého modelu. Realite viac zodpovedá pravdepodobnostný model - nie je veľa prípadov, kedy môžeme naozaj garantovať počet chybných liniek. Napriek tomu, znalosť najhoršieho prípadu môže byť v mnohých prípadoch výhodou, predstavuje dodatočný pohľad na vec a vhodne dopĺňa poznatky z pravdepodobnostného modelu.

Samotný model spoločne s výsledkami pre niektoré triedy grafov bol publikovaný v [1] a venoval sa predovšetkým šíreniu správy pomocou greedy algoritmu na niektorých triedach grafov. Neskoršia diskusia priniesla ďalšie otázky (najmä ohľadom využiteľnosti iných algoritmov). Na niektoré z týchto otázok sa snažíme nájsť odpoveď v prekladanej práci.

Používaný model sa od pôvodnej myšlienky taktiež vyvíjal. V reálnych systémoch je očakávateľné, že lineárna závislosť počtu chybných liniek je badateľná až pri väčšom počte poslaných správ. Preto v snahe sťažiť podmienky pre algoritmus šírenia správy a obmedziť zneužívanie garancii modelu v prípade posielania malého počtu správ boli doplnené ďalšie pravidlá - počet bezchybných liniek je garantovaný len v prípade, že algoritmus posielala dostatočne veľa správ (viď [2]).

Obsah

1	Úvod	1
2	Modelovanie distribuovaných systémov	3
2.1	Modelovanie chýb v distribuovaných systémoch	4
2.2	Komunikácia v distribuovaných systémoch	5
2.3	Začlenenie študovaného modelu	6
3	Používaný model	8
3.1	Definícia modelu	8
3.2	Základné vlastnosti modelu	11
4	Všeobecné triedy algoritmov	14
4.1	Greedy algoritmus	14
4.2	Algoritmus dopredného šírenia	22
4.3	Algoritmus spätného šírenia	25
4.4	Neoptimálnosť greedy algoritmu	33
5	Stromy s konštantnou excentricitou iniciátora	36
5.1	Optimálny algoritmus	36
5.2	Dolný odhad pre greedy algoritmus	36
5.3	Horný odhad pre greedy algoritmus	41
6	Hyperkocky	44
6.1	Definícia topológie a základné vlastnosti	44
6.1.1	Vrcholová hranica na hyperkocke	45
6.1.2	Hranová hranica na hyperkocke	47
6.2	Optimálny algoritmus	48
6.3	Analýza greedy algoritmu	54
6.3.1	Diskrétna analýza greedy algoritmu.	54
6.3.2	Spojité analýza	56

7	Ďalšie oblasti štúdia	62
7.1	Výber iniciátora	62
7.2	Alternatívne modely	63
7.3	Informovanosť algoritmu	65
8	Záver	67

Zoznam obrázkov

1	Spojenie grafov.	15
2	Subaditivita pre greedy algoritmus I.	17
3	Subaditivita pre greedy algoritmus II.	18
4	Neoptimálnosť greedy algoritmu I.	20
5	Neoptimálnosť greedy algoritmu II.	21
6	Neoptimálnosť algoritmu dopredného šírenia.	25
7	Neoptimálnosť algoritmu spätného šírenia I.	30
8	Neoptimálnosť algoritmu spätného šírenia II.	31
9	Neoptimálnosť algoritmu spätného šírenia III.	31
10	Neoptimálnosť algoritmu spätného šírenia IV.	32
11	Neoptimálnosť greedy algoritmu III.	34
12	Neoptimálnosť greedy algoritmu IV.	34
13	Spojité analýza greedy algoritmu na hyperkocke I.	59
14	Spojité analýza greedy algoritmu na hyperkocke II.	60
15	Spojité analýza greedy algoritmu na hyperkocke III.	61
16	Model s orientovanými hranami.	65

1 Úvod

Pojem *distribučovaný systém* sa používa na označenie množiny samostatných, vzájomne prepojených entít (počítače, procesory, procesy, ...). Každá entita je schopná samostatne počítať a komunikovať s ostatnými. Práve autonómnosť uzlov a vzájomná komunikácia je charakteristickou črtou každého distribučovaného systému. Cieľom systému ako celku je spolupracovať pri riešení zadaného problému.

Využitie distribučovaných systémov v praxi je nespochybniteľné - výmena informácií, zdieľanie zdrojov, zvyšovanie výkonu a škálovanie, zvyšovanie odolnosti voči chybám a výpadkom (replikácia). Cenou za tieto významné prínosy sú vyššie nároky na návrh a implementáciu vhodných algoritmov. Mnohé úlohy, jednoduché z prostredia sekvenčných modelov, vyžadujú netriviálne prístupy. Uzly v distribučovanom systéme majú zväčša prístup iba k vlastnému stavu, nie ku globálnemu stavu celého systému. Vykonávanie algoritmu je obvykle do istej miery nedeterministické, kvôli možným rozdielom v rýchlosti vykonávania častí systému. Situácia je o to zložitejšia, že v skúmanej oblasti existuje veľa rôznych (opodstatnených) modelov, ktoré odlišným spôsobom popisujú správanie sa distribučovaného systému. O vzťahoch medzi modelmi (najmä čo sa týka zložitosti), nie je doposiaľ veľa známe. Základné smery, ktorými sa výskum v tejto oblasti uberal, stručne spomenieme v nasledujúcej kapitole.

V tejto práci sa venujeme teoretickému štúdiu šírenia správy (broadcasting) v distribučovaných systémoch. Šírenie správy predstavuje štandardný problém, ktorého riešenie výrazne ovplyvňuje mnoho úloh vykonávaných distribučovaným systémom. Dobré pochopenie vlastností šírenia správy má veľký praktický význam a prínos pre návrh efektívnych algoritmov.

Model, pomocou ktorého popisujeme distribučovaný systém, bol prvýkrát uverejnený v [1]. Distribučovaný systém je reprezentovaný komunikačným grafom. Správy sú posielané po linkách v krokoch (synchronný model). Šírenie správy je dané algoritmom, ktorý v každom kroku vyberá hrany, po ktorých sa má poslať správa. Oproti nemu stojí fiktívny súper, ktorý mu niektoré hrany blokuje. Maximálny počet blokovaných hrán je proporcionálne závislý od počtu vybraných hrán v danom kroku. Podrobnejšie zaradenie modelu a jeho popis je uvedený v nasledujúcich častiach.

V rámci práce nás budú zaujímať predovšetkým nasledovné otázky

1. Ktoré triedy grafov sú odolné voči proporcionálnemu počtu chybných liniek?

Hľadáme triedy grafov, na ktorých nedôjde k výraznému spomaleniu šírenia správy. Porovnáваме trvanie šírenia správy v prípade bez chybných liniek a v prípade s proporcionálnym počtom chybných liniek. Pre každý graf uvažujeme optimálny algoritmus a najhoršieho protivníka.

2. Aké všeobecné (generické) triedy algoritmov môžeme zostrojiť? Aké majú vlastnosti?

Snažíme sa skonštruovať univerzálne algoritmy, ktoré garantujú dobrý čas šírenia správy na mnohých (čím všeobecnejších) triedach grafov. Príkladom je algoritmus spätného šírenia, ktorý je asymptoticky optimálny na ľubovoľnom súvislom acyklickom grafe. Na rozdiel od predchádzajúceho bodu zameriavame našu pozornosť najmä na spôsoby, akými by mohli algoritmy pracovať.

3. Aký veľký môže byť rozdiel medzi časom šírenia pomocou všeobecných algoritmov?

Pri skúmaní všeobecných algoritmov sa zaoberáme nielen triedami grafov, na ktorých dávajú dobrý čas šírenia správy. Hľadáme tie triedy grafov, na ktorých jeden všeobecný algoritmus poskytuje dobré výsledky, zatiaľ čo iný algoritmus zlé.

4. Na ktorých triedach grafov je greedy algoritmus optimálny?

Greedy algoritmus predstavuje jednoduchý prístup k šíreniu správy. Aby sme preukázali opodstatnenosť či neopodstatnenosť takého prístupu, hľadáme tie triedy grafov, na ktorých je greedy algoritmus optimálny. Špeciálne nás zaujímajú tie z nich, ktoré sú odolné voči proporcionálnemu počtu chybných liniek.

Optimálnosť algoritmov nás bude zväčša zaujímať v asymptotickom ponímaní (vzhľadom na multiplikatívnu konštantu nezávislú od grafu).

2 Modelovanie distribuovaných systémov

Teoretické štúdium distribuovaných systémov vyžaduje formálny matematický model, ktorý abstrahuje od menej dôležitých detailov a zvýrazňuje požadované vlastnosti systému. Chápanie distribuovaného systému ako množiny kooperujúcich entít je pomerne široké, takisto aj spektrum riešených úloh. Vznik mnohých modelov je prirodzeným dôsledkom tohoto stavu. Uvádzame niektoré príklady odlišností, s ktorými sa môžeme v modeloch stretnúť. Podrobnejšie (a úplnejšie) porovnanie jednotlivých modelov je k dispozícii napríklad v [3] alebo priamo v práci venovanej rovnakému modelu [4].

1. **Synchrónne a asynchrónne modely.** Synchronný model predpokladá, že distribuovaný systém pracuje v krokoch. V každom kroku môžu vrcholy poslať správy, tieto budú doručené (a môžu byť spracované) v nasledujúcom kroku. Jednotlivé vrcholy poznajú globálne hodiny (v rôznych modeloch aj číslo taktu, alebo len tiky hodín), ktorými sa riadi distribuovaný systém a môžu ich využívať pri svojej práci. Naproti tomu v asynchrónnom modeli nevyužívame predpoklady o rýchlosti šírenia správy ¹.

V našom prípade sa budeme zaoberať synchrónnymi modelmi; zamerajme sa preto na rôzne varianty synchrónnych modelov. Asynchrónne modely sú študované napríklad v [5] alebo [6], iné alternatívy v [7].

2. **Rýchlosť komunikačných liniek.** Vzhľadom na rýchlosť komunikačných liniek môžeme rozdeliť modely do dvoch hlavných kategórií - modely s konštantným režimom (constant mode), v ktorých prenesenie správy trvá jednu časovú jednotku nezávisle od veľkosti správy, a modely s lineárnym režimom (linear mode), v ktorých čas prenesenia správy závisí lineárne od veľkosti posielanej správy. Väčšina prác sa orientuje na model s konštantným režimom; príklad šírenia správy s lineárnym režimom je ponúknutý napríklad v [8].
3. **Obojsmernosť liniek.** V prípade jednosmerných liniek (one way, half-duplex, telegraph) v každom kroku môže prechádzať správa iba jedným smerom (jeden vrchol slúži ako odosielateľ, druhý ako príjemca

¹iba predpoklad, že každá správa bude doručená v konečnom čase

- [5], [9], [10]). V prípade obojsmerných liniek môžu linkou v jednom kroku prechádzať správy obidvoma smermi ([11], [12], [13]) .

4. **Počet aktívnych liniek.** Rôzne obmedzenia boli kladené na počet aktívnych liniek incidentných s jedným vrcholom. V závislosti od modelu môže vrchol v jednom kroku posilať správy po jednej linke (one-port), ľubovoľnom počte liniek (multi-port), alebo obmedzenom počte liniek (k-port). Väčšina prác využíva one-port alebo multi-port model, k-port je spracovaný napríklad v [14].
5. **Znalosť topológie.** Znalosť topológie môže mať veľkú informačnú hodnotu pre distribuovaný algoritmus. Jednotlivé vrcholy môžu poznať komunikačný graf, svoju polohu a polohu svojich susedov. V obmedzenejších prípadoch poznajú napríklad slepú mapu komunikačného grafu bez svojej polohy ([15]), majú jednoznačné identifikátory ([5], [16]) alebo zmysel pre orientáciu ([17]).

2.1 Modelovanie chýb v distribuovaných systémoch

Samostatnou kapitolou pri modelovaní distribuovaných systémov je modelovanie chýb. Odolnosť voči poruchám je kľúčovou požiadavkou na mnohé aplikácie. Distribuovaný systém zväčša dokáže priniesť výhody aj v tejto oblasti. Mnohé modely pripúšťajú možnosť vzniku chýb. Odlisujú sa od seba najmä typom a množstvom chýb, ktoré môžu vzniknúť.

1. **Chyby vrcholov alebo hrán.** Distribuovaný systém pozostáva z objektov dvoch typov - vrcholov a hrán. Intenzívne študované sú oba typy porúch.
2. **Pravdepodobnostné alebo ohraničené modely.** Pri modelovaní chýb v distribuovaných systémoch rozlišujeme dva rozdielne prístupy. V pravdepodobnostnom modeli chyby vznikajú s istou (stanovenou) pravdepodobnosťou. Pri návrhu algoritmu požadujeme, aby riešil zadaný problém s dostatočne vysokou pravdepodobnosťou. Naproti tomu v ohraničenom modeli doplníme dodatočné predpoklady na maximálny počet chýb, resp. ohraničujeme iné udalosti. To nám umožňuje analyzovať najhorší prípad. Modelovanie chýb sa často neformálne prirovnáva k hre dvoch hráčov - algoritmus určuje, ktoré správy majú byť

poslané a súper chyby, ktoré v modeli vzniknú. V pravdepodobnostnom modeli súpera reprezentuje náhoda (pravdepodobnostné rozdelenie), v ohraničenom môže hrať ľubovoľne v súlade so stanovenými pravidlami modelu.

V prípade ohraničených modelov sa veľa úsilia venovalo modelu, ktorý umožňuje konštantou ohraničený počet chybných liniek v jednom kroku ([18], [19], [20], [21], [22], [23]).

- 3. Byzantínske chyby alebo zlyhania.** V modeli s byzantínskymi chybami môže súper ľubovoľne určiť správanie sa chybného objektu. V prípade chybných vrcholov to znamená, že vrchol nemusí pracovať na základe stanoveného algoritmu, ale môže vykonávať ľubovoľnú (škodlivú) činnosť. V prípade chybných hrán to znamená, že správy sa môžu strácať, meniť alebo vznikajú falošné. Byzantínske chyby poskytujú najväčší priestor súperovi.

Pri zlyhaní objektov (vrcholov, hrán) pokazený objekt prestane pracovať, súper však nijako nemôže pozmeniť jeho činnosť. V prípade chyby liniek to znamená, že správy prenášané po linke sa strácajú.

- 4. Statické alebo dynamické chyby.** Iným študovaným aspektom modelu je trvanie chýb. V prvom prípade statických chýb máme množinu chybných objektov, ktoré sú chybné počas celého výpočtu. Žiaden objekt sa nemôže pokaziť ani zotaviť z chyby počas výpočtu. V modeloch pracujúcich s dynamickými chybami sa objekty kazia a zotavujú počas výpočtu. V každom kroku môže byť množina chybných objektov odlišná (pokiaľ spĺňa ostatné ohraničenia modelu).

Dynamické chyby boli poprvýkrát predstavené v práci [18].

2.2 Komunikácia v distribuovaných systémoch

Spomedzi štandardných problémov riešených v oblasti spomeňme v prvom rade otázku komunikácie. Komunikácia medzi entitami je nevyhnutnou súčasťou všetkých netriviálnych výpočtov realizovaných distribuovaným systémom, vyplýva priamo z jeho podstaty. Jej efektívne riešenie býva kľúčovou požiadavkou pri návrhu algoritmov. Komunikačné schéma je veľmi úzko spätá s konkrétnym problémom; štúdium štandardných algoritmov však prinieslo niekoľko „generických riešení“, ktoré sa objavujú opakovane.

V rámci vývoja distribuovaných systémov boli študované viaceré problémy komunikácie²

1. **Broadcasting** spočíva v poslaní informácie z jedného procesora všetkým ostatným. V našej práci sa budeme zaoberať týmto typom problémom.
2. **Gossiping** vznikne doplnením predošlej požiadavky. Potrebujeme preniesť informáciu z každého procesora do všetkých ostatných. Informácia je závislá len od odosielajúceho procesora.
3. **Scattering** spočíva v poslaní informácie z jedného procesora všetkým ostatným. Informácia nie je rovnaká, pre každého príjemcu je určená samostatne.
4. **Multiscattering** vznikne opäť doplnením predošlej požiadavky. Potrebujeme preniesť informáciu z každého procesora do všetkých ostatných. Informácia je špecifická pre každého odosielaťela a každého príjemcu.

Podrobnejšie informácie sú k dispozícii napríklad v [24] alebo [25].

2.3 Začlenenie študovaného modelu

Nami študovaný model spadá do nasledovných kategórií

1. Je synchronný, šírenie správ prebieha v krokoch.
2. Trvanie prenosu správy trvá konštantný čas, nezávisle od veľkosti správy.
3. Obojsmernosť liniek je nepodstatná, nakoľko je možné posilať správy iba z informovaných vrcholov do neinformovaných vrcholov (vid' definíciu modelu).
4. V každom vrchole môže byť v jednom kroku aktívny ľubovoľný počet liniek.
5. Algoritmus disponuje globálnou informáciou, rozhoduje sa na základe grafu a množiny aktuálne informovaných vrcholov.

²pojmy uvádzame v anglickom znení, ako sú používané v odborných textoch

6. Pripúšťame chyby zlyhania hrán, ktoré spôsobujú stratu správy.
7. Chyby hrán sú dynamické, v každom kroku môžu zlyhať iné hrany.
8. Počet chýb je ohraničený a nie je určený pravdepodobnosťou. Počet hrán, ktoré môžu zlyhať, závisí proporcionálne od počtu aktívnych hrán v danom kroku.

3 Používaný model

V prechádzajúcej časti sme priblížili používaný model zaradením do triedy študovaných modelov. Aby sme mohli podložiť a zdôvodniť dosiahnuté výsledky, potrebujeme model definovať formálnejšie.

3.1 Definícia modelu

Definícia 3.1 (šírenie správy na grafe G z vrchola w). Uvažujme graf $G = (V, E)$ a vrchol $w \in V$. Šírením správy (broadcast) z počiatočného vrcholu w je postupnosť

$$\{w\} = V_0 \subseteq V_1 \subseteq \dots \subseteq V_{t-1} \subset V_t = V$$

Číslo t nazývame čas šírenia správy (broadcast time) a množina V_i reprezentuje vrcholy, ktoré sú informované po i krokoch. Navyše požadujeme, aby sa správy mohli šíriť iba po hranách, teda pre každé i a pre každý vrchol $v \in V_{i+1} \setminus V_i$ existuje vrchol $u \in V_i$ taký, že $(u, v) \in E$.

Kľúčovou otázkou v nasledujúcich častiach je práve spôsob, akým vyberáme jednotlivé množiny V_i .

Definícia 3.2 (hranová hranica). Nech $G = (V, E)$ je ľubovoľný graf, $W \subseteq V$ je podmnožina vrcholov. Hranovou hranicou (edge boundary) množiny W označujeme

$$\partial W = \{e \in E \mid e = (u, v) \wedge u \in W \wedge v \in V \setminus W\}$$

Definícia 3.3 (vrcholová hranica). Majme graf $G = (V, E)$ a množinu vrcholov $S \subseteq V$. Vrcholovou hranicou (vertex boundary) množiny S nazývame

$$\delta(S) = \{v \in V \setminus S \mid \exists s \in S : (s, v) \in E\}$$

Definícia 3.4 (algoritmus šírenia správy). Algoritmus šírenia správy (broadcast algorithm) pre graf $G = (V, E)$ je ľubovoľné zobrazenie

$$A : 2^V \mapsto 2^E$$

ktoré množine vrcholov W priradí množinu aktívnych hrán $A(W)$, podmnožinu hranovej hranice ∂W . Každá aktívna hrana vedie z informovaného

vrchola z množiny W do neinformovaného vrchola z množiny $V \setminus W$.

Aktívne hrany sú práve tie, po ktorých by mala byť v nasledovnom kroku prenesená správa. V závislosti od zvoleného modelu šírenia správy môžu niektoré z aktívnych hrán zlyhať. Ako vidno z našej definície šírenia správy, nezaobráame sa niektorými ďalšími požiadavkami na algoritmus (bez znalosti topológie, bez potvrdenia o doručení, ...). Niektoré tieto vlastnosti si spomenieme pri konkrétnych skúmaných algoritmoch.

Definícia 3.5 (α -proporcionálne šírenie správy). Majme daný graf $G = (V, E)$, počiatočný vrchol $w \in V$ a algoritmus šírenia správy A pre graf G . α -proporcionálne šírenie správy na grafe G z vrcholu w popísané algoritmom A je ľubovoľné šírenie správy na grafe G z vrchola w , ktoré spĺňa nasledovnú podmienku pre každé i .

V_i nech označuje množinu vrcholov informovaných po i krokoch z definície šírenia správy na grafe G z vrcholu w . Označme v $(i + 1)$ -om kroku

- $E_{i+1}^* = A(V_i) \subseteq \partial V_i$ aktívne hrany
- $F_{i+1} \subseteq E_{i+1}^*$ chybné hrany
- $E_{i+1} = E_{i+1}^* \setminus F_{i+1}$ bezchybné hrany

Potom množina informovaných vrcholov v $(i + 1)$ -om kroku má podobu

$$V_{i+1} = V_i \cup \{v \in V \mid \text{existuje hrana } (u, v) \in E_{i+1} \text{ a vrchol } u \in V_i\}$$

Navyše požadujeme $|F_{i+1}| \leq \alpha |E_{i+1}^*|$ (α -proporcionalita).

Pokiaľ nedôjde k nejednoznačnosti, budeme v ďalšom texte niektoré z premenných α , G , w , A vynechávať. Ak nebude uvedené inak, budeme vždy spolu s grafom G implicitne považovať za zvolený aj niektorý z jeho vrcholov w za iniciátora šírenia správy. Koeficient α vždy vyberáme z intervalu $0 < \alpha < 1$.

Intuitívna predstava, o ktorú sa budeme v ďalšom opierať, je nasledovná. Na začiatku šírenia je informovaný jeden vrchol - w . Šírenie prebieha v krokoch, v ktorých sa (očakávane) rozširuje množina informovaných vrcholov. V každom kroku algoritmus šírenia správy vyberá niektoré hrany (aktívne hrany, množina E_i^*) vedúce z informovaných do neinformovaných vrcholov.

V ďalšom kroku mu súper niektoré z týchto hrán zablokuje (chybné hrany, množina F_i). Máme garantované, že chybných hrán bude najviac proporcionálne veľa k počtu aktívnych hrán, ohraničené dopredu daným koeficientom α .

Definícia 3.6 (používané pojmy). Pre účely tohoto textu budeme nazývať (všetky pojmy sa viažu k i -temu kroku šírenia správy)

- potenciálnymi hranami množinu ∂V_{i-1} .
- aktívnymi hranami množinu E_i^* .
- chybnými hranami množinu F_i .
- bezchybnými hranami množinu $E_i = E_i^* \setminus F_i$.
- informovanými vrcholmi množinu V_i .
- aktívnymi vrcholmi množinu všetkých vrcholov, ktoré sú incidentné aspoň s jednou aktívnou hranou a nie sú informované už v $(i - 1)$ -om kroku.
- novoinformovanými vrcholmi množinu $V_i \setminus V_{i-1}$. Novoinformované vrcholy sa vyberajú spomedzi aktívnych.
- Číslom aktivity aktívneho vrchola nazývame počet aktívnych hrán (v i -tom kroku), ktoré sú s ním incidentné.

Niektoré z pojmov sa vyskytli v rovnakom (intuitívnom) význame už v prechádzajúcich častiach.

Definícia 3.7 (čas algoritmu A). Nech G je graf, w počiatočný vrchol a A algoritmus šírenia správy pre graf G . Čas algoritmu A (na grafe G s počiatočným vrcholom w a α -proportionálnym modelom) je najväčšie t také, že existuje α -proportionálne šírenie správy na grafe G z vrcholu w popísané algoritmom A s časom t . Hovoríme, že algoritmus neskončí (čas je nekonečný, prípadne nedefinovaný) v prípade, že také t neexistuje. Čas značíme T_G^A , pokiaľ w a α je zjavné z kontextu.

Čas algoritmu A hovorí o tom, ako dlho môže v najhoršom prípade trvať šírenie správy popísané algoritmom A . V neformálnej predstave, ako veľmi môže súper blokujúci správy (v súlade s modelom) spomaliť šírenie správy týmto algoritmom.

Definícia 3.8 (čas grafu G). Nech G je graf a w počiatočný vrchol. Čas grafu G (najhorší garantovateľný, s počiatočným vrcholom w a α -proporcionálnym modelom) je najmenšie t také, že existuje algoritmus A pre graf G s časom t . Čas značíme T_G , pokiaľ w a α je zjavné z kontextu. Algoritmus A taký, pre ktorý platí $T_G = T_G^A$, nazývame optimálny pre graf G .

Definícia 3.9 (čas pre triedu grafov). Nech $G = (G_1, G_2, G_3, \dots)$ je postupnosť grafov, $A = (A_1, A_2, A_3, \dots)$ je im prislúchajúca postupnosť algoritmov. Potom T_G^A označujeme čas trvania triedy algoritmov (resp. zjednodušene algoritmu) A na grafoch G . T_G^A je funkcia $N \mapsto N$ definovaná nasledovne

$$T_G^A(n) = \max \left\{ T_{G_i}^A \mid G_i \text{ má najviac } n \text{ vrcholov} \right\}$$

Pokiaľ také G_i neexistuje, funkcia je v tomto bode nedefinovaná.

Funkcia $T_G^A(n)$ nám umožní popisovať čas šírenia správy pre „rastúce“ grafy - využívať asymptotické zápisy.

Definícia 3.10 (odolnosť triedy grafov). Hovoríme, že trieda grafov $G = (G_1, G_2, G_3, \dots)$ s iniciátormi $w_1, w_2, w_3 \dots$ je odolná voči proporcionálnemu počtu chybných liniek, ak existuje k také, že pre všetky i platí

$$T_{G_i} \leq k \text{ exc}_{G_i}(w_i)$$

Odolné triedy grafov sú veľmi zaujímavé - spoločne s vhodným algoritmom sa vieme veľmi dobre vysporiadať s chybami (bez toho, aby výrazne vzrástol čas šírenia správy). Nie každú odolnú triedu grafov možno (intuitívne) klasifikovať ako vhodnú - predlžujúca sa lúnia pospájaných vrcholov je odolnou triedou grafov, napriek tomu ju však zväčša nepovažujeme za vhodnú (šírenie správy trvá dlho aj v prostredí bez chybných liniek).

3.2 Základné vlastnosti modelu

V tejto časti vyslovíme niekoľko jednoduchých tvrdení, ktoré ohraničujú čas šírenia správy na grafe bez znalosti konkrétneho algoritmu. Tvrdenia aj ich dôkazy sú jednoduché, ich účelom je uvedenie si základných vlastností a stotožnenie sa s modelom.

Veta 3.11 (ohraničenie času zhora). *Pre ľubovoľný graf G , počiatočný vrchol w a algoritmus A platí: ak T_G^A je definované, potom $T_G^A < n$, kde $n = |V|$.*

Myšlienka väčšiny dôkazov na ohraničenie času zhora je podobná - potrebujeme ukázať, že nezávisle od zvoleného súpera rastie počet informovaných vrcholov dosť rýchlo (klesá počet neinformovaných vrcholov).

Dôkaz. Pokiaľ algoritmus skončí, musí byť v každom kroku informovaný aspoň jeden nový vrchol. Zoberme si najdlhšiu postupnosť V_0, \dots, V_t , predstavujúcu najhorší prípad šírenia správy a uvažujme dva prípady. Pokiaľ v každom kroku platí $V_i \subset V_{i+1}$, potom postupnosť pozostáva najviac n prvkov (a teda prezentuje najviac $n - 1$ krokov šírenia správy). Ak v nejakom kroku platí $V_i = V_{i+1}$, potom ale v $(i + 1)$ -om kroku nebola žiadna hrana bezchybná a teda ani žiadna hrana aktívna (viď definíciu 3.6). Keďže algoritmus je deterministický (daný ako zobrazenie $2^V \mapsto 2^E$), potom pre každé $j > i$ platí $V_i = V_j$ a teda T_G^A je nedefinované (šírenie správy nie je korektné, spor s podmienkou $V_{t-1} \subset V_t$). \square

Grafom, ktorého čas je $n - 1$, je napríklad línia pospájaných vrcholov s iniciátorom na okraji. V tomto prípade je v každom kroku novoinformovaný práve jeden vrchol. Postupnosť šírenia správy neexistuje (T_G^A je nedefinované) pre nesúvislé grafy alebo pre degenerované algoritmy, ktoré (aspoň v niektorých prípadoch) nevyberajú žiadnu hranu za aktívnu.

Veta 3.12 (ohraničenie času zdola). *Pre ľubovoľný graf G , počiatočný vrchol w a algoritmus A platí: ak T_G^A je definované, potom $T_G^A = \Omega(\log n)$, kde $n = |V|$.*

Pri ohraničení zdola je postup zväčša jednoduchší - stačí nájsť jedného súpera, pre ktorého platí požadovaná nerovnosť. Najzložitejšou časťou dôkazu býva ukázať korektnosť zvoleného súpera (α -proporcionálnosť) a spočítať dĺžku postupnosti šírenia správy. Uvedené tvrdenie je intuitívne - v každom kroku sa množina neinformovaných vrcholov môže zmenšiť najviac o proporcionálnu časť. Pár technickejších krokov si vyžiadala celá časť v definícii modelu.

Dôkaz. Algoritmus A v $(i + 1)$ -om kroku vyberie množinu aktívnych hrán $E_{i+1}^* = A(V_i)$. Z definície modelu je možné, že viaceré z týchto hrán smerujú

do toho istého aktívneho vrcholu. Nech súper blokuje maximálny možný počet hrán v súlade s modelom, pričom prednostne blokuje hrany vedúce do aktívneho vrchola s nižším číslom aktivity. Z konštrukcie vyplýva, že počet novoinformovaných vrcholov v tomto kroku bude najviac $\lceil \alpha' x \rceil$, kde x je počet aktívnych vrcholov a $\alpha' = 1 - \alpha$. Označme $v_i = |V_i|$ počet informovaných vrcholov v i -tom kroku a riešime nasledovnú rekurenciu

$$\begin{aligned} v_0 &= 1 \\ v_{i+1} &\leq v_i + \lceil \alpha'(n - v_i) \rceil \\ &\leq v_i + \alpha'(n - v_i) + 1 \end{aligned}$$

s preznačením $v'_i = (n - v_i)$, tzn. počítame neinformované vrcholy

$$\begin{aligned} v'_0 &= n - 1 \\ v'_{i+1} &\geq v'_i - \alpha'v'_i - 1 \\ &\geq \alpha v'_i - 1 \end{aligned}$$

dostávame riešenie

$$v'_i \geq \alpha^i \left(n - 1 - \frac{1}{\alpha'} \right) - \frac{1}{\alpha'}$$

položme pravú stranu rovnú jednej a dopočítajme i , kedy sa v grafe nachádza aspoň jeden neinformovaný vrchol, ktorý musí byť nevyhnutne informovaný v nasledujúcom kroku

$$i = \log_{\alpha} \left(\frac{\alpha' + 1}{\alpha'n - \alpha' + 1} \right) = \Theta(\log n)$$

Posledná rovnosť vyplýva z požiadavky $0 < \alpha < 1$.

□

Triedou grafov, ktorej čas je $\Theta(\log n)$, sú napríklad úplné grafy. Dôkaz je uvedený v sekcii venovanej greedy algoritmu.

Triviálnym dôsledkom tvrdenia je skutočnosť, že žiadna trieda grafov s priemermi $o(\log n)$ nemôže byť odolná voči proporcionálnemu počtu chybných liniek.

4 Všeobecné triedy algoritmov

Pripomeňme si otázky, formulované v úvode práce.

1. Aké všeobecné (generické) triedy algoritmov môžeme zostrojiť?
2. Aké majú vlastnosti?
3. Aký veľký môže byť rozdiel medzi časom šírenia pomocou všeobecných algoritmov?

Myšlienka všeobecných algoritmov spočíva vo formulovaní jednoduchých princípov a pravidiel, na základe ktorých má prebiehať šírenie správy. Tieto pravidlá môžu byť motivované hľadaním jednoduchých prístupov (posielanie správ po všetkých linkách - greedy algoritmus), získané relaxáciou modelu (simulujeme prostredie bez chybných liniek - algoritmus dopredného šírenia) alebo získané pozorovaním správania sa modelu (maximalizácia počtu aktívnych liniek v neskorších krokoch - algoritmus spätného šírenia).

V nasledujúcej časti definujeme tieto tri všeobecné algoritmy (greedy algoritmus, algoritmus dopredného šírenia a algoritmus spätného šírenia) a vysvetlíme motiváciu vedúcu k ich vzniku. Popíšeme ich správanie na jednotlivých triedach grafov. Pokúsime sa načrtnúť rozdiely, ktoré môžu vzniknúť šírením správy na rovnakom grafe s pomocou rôznych algoritmov.

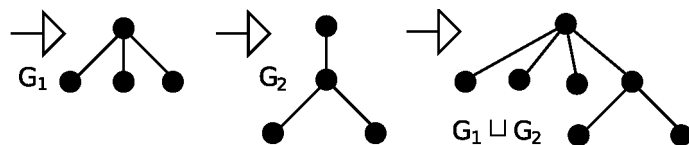
4.1 Greedy algoritmus

Prvým spomedzi študovaných algoritmov bol greedy (pažravý) algoritmus. Greedy algoritmus nerozlišuje medzi hranami a snaží sa v každom kroku poselať správy po všetkých prístupných hranách (všetky potenciálne hrany sú zvolené za aktívne hrany).

Definícia 4.1 (greedy algoritmus). Majme graf $G = (V, E)$. Greedy algoritmom pre graf G nazývame zobrazenie $A_G : 2^V \mapsto 2^E$ také, že pre ľubovoľnú množinu vrcholov $W \subseteq V$ platí

$$A_G(W) = \partial W$$

V predchádzajúcich prácach sa venovalo veľa úsilia štúdiu trvania greedy algoritmu na rôznych topológiach (niektoré výsledky o chvíľu spomenieme).



Obr. 1: Spojenie grafov.
Iniciátor je zobrazený so šípkou.

Motiváciou k jeho štúdiu je práve jednoduchosť zadania (a prípadne jednoduchosť implementácie). Algoritmus nepotrebuje väčšinu zo spomínaných dodatočných informácií (mapu komunikačného grafu, polohu, číslo tiky hodín), preto sú výsledky využiteľné aj v reštriktívnejších prostrediach. Ukazuje sa však, že šírenie správy pomocou greedy algoritmu môže v mnohých prípadoch dávať pomerne zlé výsledky. Analýza trvania pre netriviálne grafy môže byť obtiažna, niektoré (na pohľad intuitívne) vlastnosti pre šírenie správy pomocou greedy algoritmu totiž neplatia. Uvedme ako príklad subaditivitu v nasledujúcom zmysle

Definícia 4.2 (spojenie grafov). Majme graf $G_1 = (V_1, E_1)$ s iniciátorom w_1 a $G_2 = (V_2, E_2)$ s iniciátorom w_2 . Spojenie grafov $G_1 \sqcup G_2$ je graf $G = (V, E)$ taký, ktorý vznikne zlúčením jednotlivých iniciátorov. Formálne, predpokladajme navzájom dizjunktné množiny vrcholov V_1 a V_2 . Pre graf G potom platí

$$V = (V_1 \setminus \{w_1\}) \cup (V_2 \setminus \{w_2\}) \cup \{w\}$$

Dva vrcholy u, v rôzne od iniciátora sú spojené hranou, ak boli spojené hranou v niektorom z grafov G_1 alebo G_2 . Iniciátor w je spojený hranou s vrcholom u , ak je hranou spojený iniciátor w_1 s vrcholom u v grafe G_1 , resp. iniciátor w_2 s vrcholom u v grafe G_2 .

Veta 4.3 (subaditivita pre optimálny algoritmus). Pre ľubovoľné grafy G_1, G_2 platí $T_{G_1 \sqcup G_2} \leq T_{G_1} + T_{G_2}$

Dôkaz. Zostrojme algoritmus X taký, aby platilo

$$T_{G_1 \sqcup G_2}^X = T_{G_1} + T_{G_2}$$

Algoritmus najprv simuluje činnosť optimálneho algoritmu pre graf G_1 , všetky hrany pochádzajúce z grafu G_2 sú neaktívne. Akonáhle sú všetky

vrcholy grafu G_1 informované, simuluje činnosť optimálneho algoritmu pre graf G_2 . Hoci algoritmus X nemusí byť optimálny pre graf $G_1 \sqcup G_2$, zaručuje platnosť požadovanej nerovnosti

$$T_{G_1 \sqcup G_2} \leq T_{G_1} + T_{G_2}$$

□

Táto intuitívna vlastnosť platí pre optimálny algoritmus, neplatí však pre greedy algoritmus. Pridanie nových hrán a vrcholov ku grafu môže spôsobiť rozsiahlejšie spomalenie šírenia správy.

Veta 4.4 (subaditivita pre greedy algoritmus). *Existujú grafy G_1, G_2 také, že $T_{G_1 \sqcup G_2}^{AG} > T_{G_1}^{AG} + T_{G_2}^{AG}$*

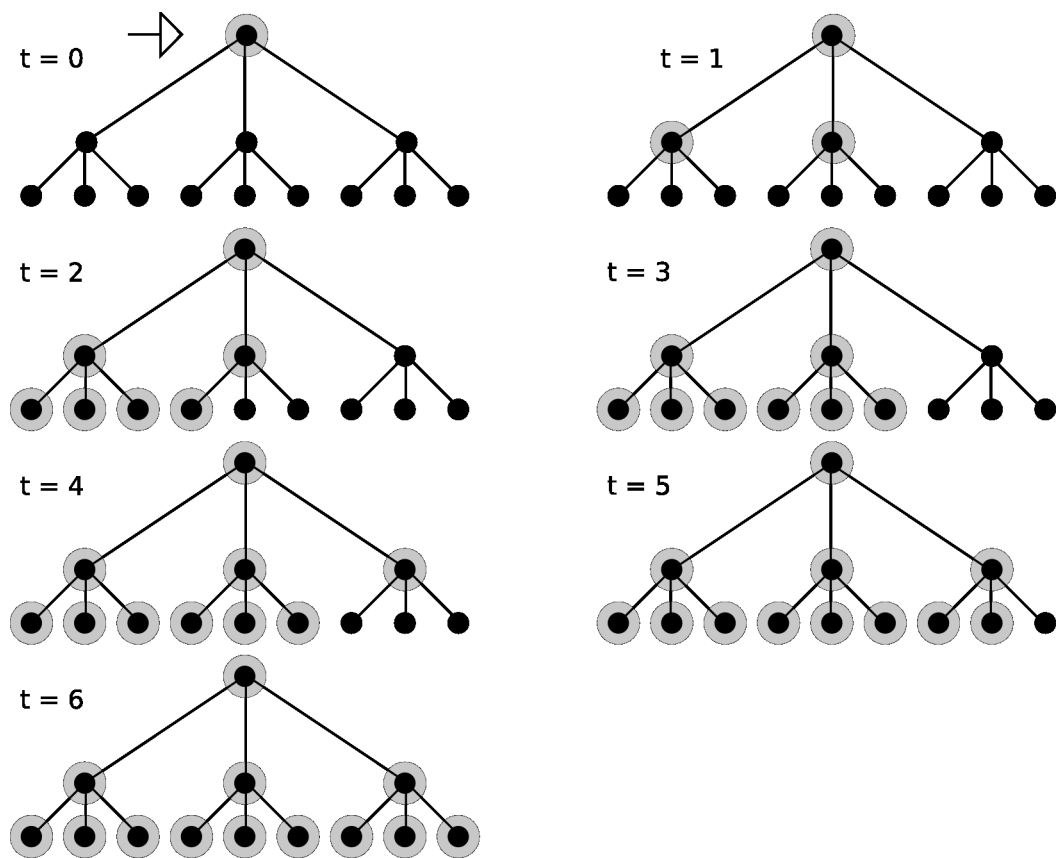
Dôkaz. Uvažujme dva grafy. Prvým grafom sú dva vrcholy prepojené hranou, druhý graf je zobrazený na obrázku 2 spoločne s najhorším prípadom šírenia správy s greedy algoritmom. Samotné grafy majú časy šírenia rovné 1 a 6, spojením grafov dostávame graf s časom šírenia 8, ako je zobrazené na obrázku 3. □

Predstavme ďalej niektoré výsledky o trvaní greedy algoritmu na vybraných štandardne študovaných triedach grafov. Ďalšie výsledky budú prezentované v kapitole o stromoch s konštantnou excentricitou iniciátora a v kapitole o hyperkockách.

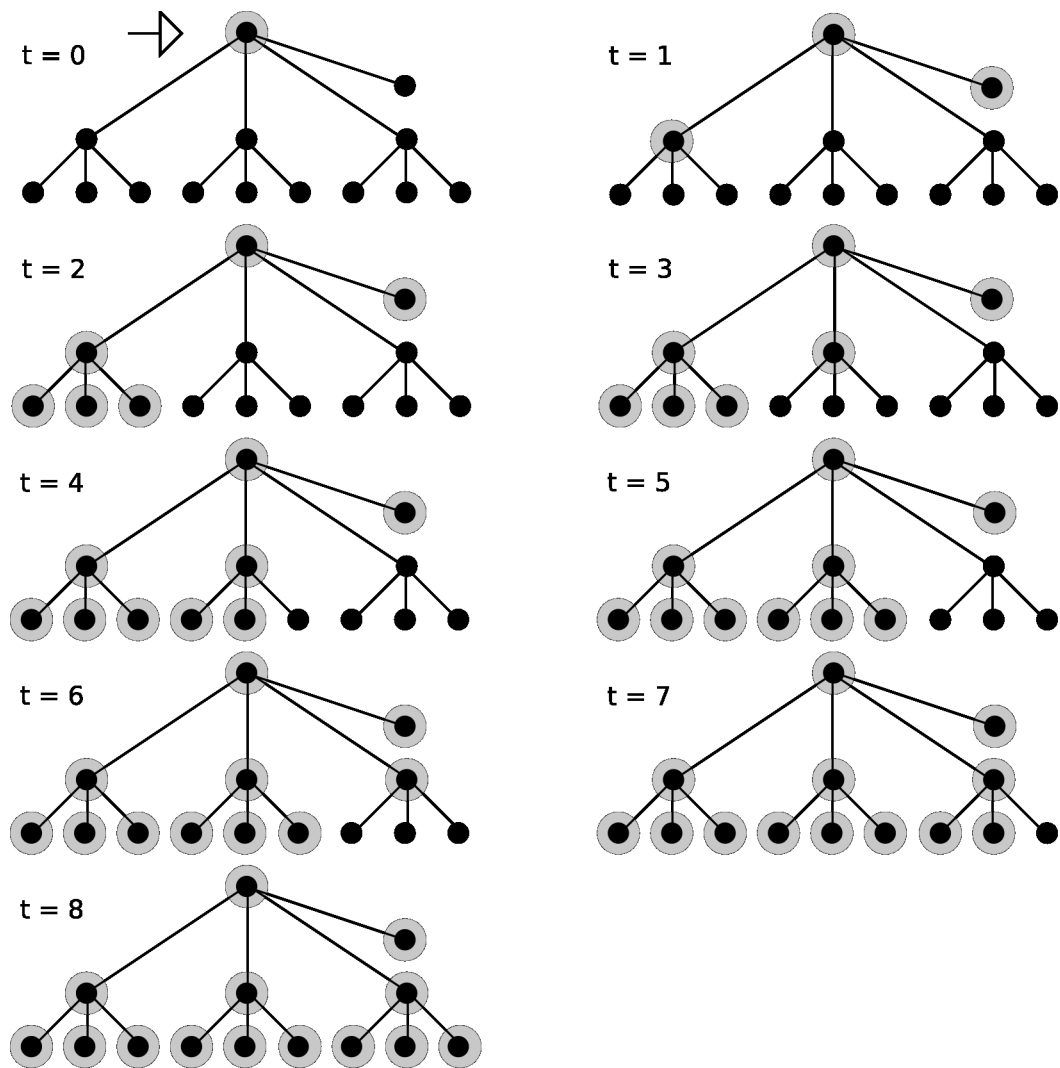
Veta 4.5 (úplné grafy K_n). *Nech $K = (K_1, K_2, K_3, \dots)$ je trieda úplných grafov a A_G prislúchajúca trieda greedy algoritmov. Potom platí*

$$T_K^{AG} = \Theta(\log n)$$

Dôkaz. Odhad $T_K^{AG} = \Omega(\log n)$ vyplýva zo všeobecných vlastností šírenia správy. Ďalej si stačí uvedomiť, že v každom kroku majú všetky aktívne vrcholy rovnaký stupeň aktivity, a teda platí obrátená nerovnosť oproti vete 3.12 (v_i označuje počet informovaných vrcholov v i -tom kroku).



Obr. 2: Subaditivita pre greedy algoritmus I.
 Najhorší prípad šírenia správy s greedy algoritmom na zvolenom grafe.



Obr. 3: Subaditivita pre greedy algoritmus II.
 Najhorší prípad šírenia správy s greedy algoritmom na zvolenom grafe.

$$\begin{aligned}
v_0 &= 1 \\
v_{i+1} &\geq v_i + \lceil \alpha'(n - v_i) \rceil \\
&\geq v_i + \alpha'(n - v_i)
\end{aligned}$$

dôkaz pokračuje rovnako, ako v prípade vety 3.12 substitúciou $v'_i = n - v_i$

$$\begin{aligned}
v'_0 &= n - 1 \\
v'_{i+1} &\leq \alpha v'_i
\end{aligned}$$

a teda $v'_i \leq (n - 1)\alpha^i$. Položme pravú stranu jednej a dostaneme, že po $\log_\alpha(1/(n - 1))$ krokoch sa v grafe nachádza najviac jeden neinformovaný vrchol, ktorý musí byť nevyhnutne informovaný v nasledujúcom kroku.

$$T_K^{AG} \leq \log_\alpha \left(\frac{1}{n - 1} \right) + 1 = O(\log n)$$

□

Veta 4.6 (úplné binárne stromy T_n). *Majme triedu úplných binárnych stromov $T = (T_1, T_2, T_3, \dots)$ - kde strom T_i ma hĺbku i a $2^i - 1$ vrcholov - a prislúchajúcu triedu greedy algoritmov A_G . Potom platí*

$$T_T^{AG} = \Omega(2^{\frac{i}{2}})$$

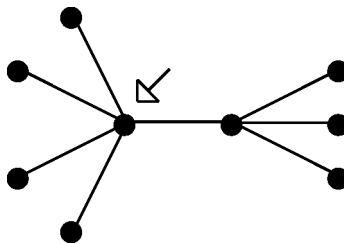
Dôkaz môže čitateľ nájsť v práci [1].

Definícia 4.7. (cyklická mriežka) N -rozmernou cyklickou mriežkou nazývame graf, ktorého vrcholy tvorí množina Z_k^n ; dva vrcholy sú spojené hranou práve vtedy, ak sa líšia práve na jednej pozícii práve o jedna (v zmysle grupy Z_k).

Veta 4.8 (cyklické mriežky). *Pre n -rozmernú cyklickú mriežku M s konštantnou hodnotou n a párnym k platí*

$$T_M^{AG} = \Theta(k)$$

Dôkaz môže čitateľ nájsť v práci [1].



Obr. 4: Neoptimálnosť greedy algoritmu I.
Graf publikovaný v práci [1], na ktorom je greedy algoritmus neoptimálny.

Vzniká prirodzená otázka, nakoľko dobrý čas šírenia správy greedy algoritmus ponúka. Z predchádzajúceho tvrdenia vyplýva, že na úplných grafoch je greedy algoritmus prinajmenšom asymptoticky optimálny. Je táto vlastnosť zachovaná aj na iných triedach grafov? Aká veľká medzera môže vzniknúť medzi trvaním greedy algoritmu a optimálneho algoritmu?

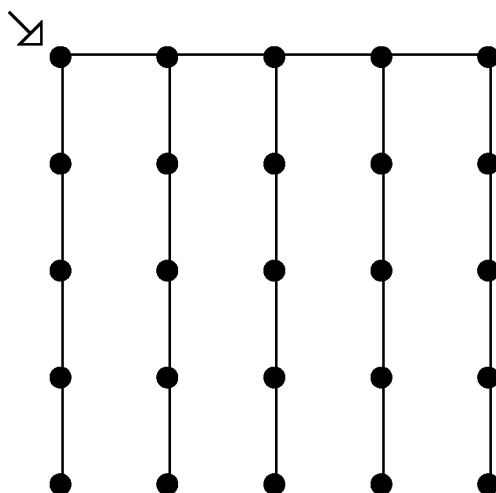
Príklad 4.9 (greedy algoritmus nie je optimálny). V práci [1], ktorá je venovaná analýze greedy algoritmu na viacerých topológiach, je uvedený graf zobrazený na obrázku 4. Trvanie greedy algoritmu na tomto grafe je 5, optimálny algoritmus dosiahne čas 4.

Príklad 4.10 (greedy algoritmus nie je asymptoticky optimálny). Uvažujme triedu grafov, ktorá pre zvolené m pozostáva z mriežky $m \times m$ vrcholov, pričom spojené sú po sebe nasledujúce vrcholy v prvom riadku a po sebe nasledujúce vrcholy v každom stĺpci. Príklad takéhoto grafu pre $m = 5$ je na obrázku 5.

Nájdime súpera, ktorý dokáže dostatočne spomaliť šírenie správy pomocou greedy algoritmu. Dobrou stratégiou je blokovať prednostne horizontálne hrany. V tomto prípade je v každom kroku informovaný práve jeden nový vrchol, trvanie greedy algoritmu na tomto grafe je $m^2 - 1 = n - 1$ krokov - čo je najhorší prípad, aký môžeme pri danom počte vrcholov dostať.

Nájdime teraz ľubovoľný dostatočne dobrý algoritmus (jeho čas bude horným odhadom pre trvanie optimálneho algoritmu). Zostrojme algoritmus tak, že

- V prvej fáze (pokiaľ existuje neinformovaný vrchol v prvom riadku), posiela správy výlučne po horizontálnych hranách.



Obr. 5: Neoptimálnosť greedy algoritmu II.
Príklad mriežky - hrebeňa pre $m = 5$, na ktorej je greedy algoritmus asymptoticky neoptimálny.

- V druhej fáze (všetky vrcholy v prvom riadku sú informované) nájde prvý riadok zhora, v ktorom sa nachádza nejaký neinformovaný vrchol. Následne zvolí za aktívne všetky tie hrany spomedzi potenciálnych, ktoré vedú do ľubovoľného vrchola v tomto (prvom neinformovanom) riadku.

V rámci prvej fázy je v každom kroku poslaná práve jedna správa, počet krokov je teda $m - 1$. Druhú fázu môžeme rozdeliť na etapy podľa riadkov. Je potrebné si uvedomiť, že ak sa aktívne vrcholy nachádzajú v i -tom riadku, potom

- všetky vrcholy v riadkoch $j < i$ sú (v tomto kroku) informované
- všetky vrcholy v riadkoch $j > i$ sú (v tomto kroku) neinformované

Na základe toho obdobnou rekurenciou ako vo vete 4.5 dostaneme ohraničenie na počet krokov jednej etapy (informovanie vrcholov v jednom riadku) na $\log_{\alpha}(1/m) + 1$, pre celkové trvanie dostávame horný odhad

$$(m - 1) + (m - 1) \left[\log_{\alpha} \left(\frac{1}{m} \right) + 1 \right] = O(m \log m) = O \left(n^{\frac{1}{2}} \log n \right)$$

Na konci kapitoly sa vrátíme k analýze medzery medzi greedy algoritmom a optimálnym algoritmom. Nateraz potrebujeme predstaviť ďalšie typy algoritmov, aby sme vedeli lepšie charakterizovať optimálny algoritmus.

4.2 Algoritmus dopredného šírenia

Algoritmus dopredného šírenia je definovaný pre ľubovoľný graf. Myšlienka algoritmu spočíva v simulácii algoritmu šírenia správy v modeli bez chybných liniek. V tomto prostredí sa správy šíria po vrstvách; každá vrstva predstavuje vrcholy s rovnakou vzdialenosťou od iniciátora. Algoritmus dopredného šírenia vynucuje rovnaké šírenie správ aj v modeli s proporcionálnym počtom chybných liniek. Samozrejme, nie je možné zaručiť, aby všetky vrcholy v jednej vrstve boli informované v priebehu jedného kroku (ako je to v prípade šírenia správ v modeli bez chybných liniek). Algoritmus teda v každom kroku vyhledá najbližšiu vrstvu k iniciátorovi, obsahujúcu nejaké neinformované vrcholy a označí za aktívne všetky potenciálne hrany vedúce do neinformovaných vrcholov na tejto vrstve.

Definícia 4.11 (algoritmus dopredného šírenia). Nech G je ľubovoľný súvislý graf s iniciátorom w . Algoritmom dopredného šírenia pre graf G nazývame každé také zobrazenie $A_D : 2^V \mapsto 2^E$, že pre ľubovoľnú množinu vrcholov $U \subseteq V$ platí - hrana (u, u') , kde $u \in U$ a $u' \in V \setminus U$ bude aktívna práve vtedy, ak vzdialenosť vrchola u' od iniciátora $d(w, u')$ je minimálna spomedzi všetkých vzdialeností neinformovaných vrcholov, t.j.

$$(u, u') \in A_D(U) \text{ práve vtedy, keď } d(w, u') = \min\{d(w, x) \mid x \in V \setminus U\}$$

Veta 4.12. Nech $G = (V, E)$ je ľubovoľný acyklický súvislý graf s iniciátorom w . Označme D_j množinu všetkých vrcholov grafu G so vzdialenosťou j od vrchola w a d_j jej mohutnosť. Potom platí

$$T_G^{A_D} \leq 2 \sum_{j=0}^{exc_G(w)} \left[\log_\alpha \left(\frac{1}{d_j} \right) + 1 \right]$$

Dôkaz. Ukážeme, že práve vrcholy množiny $D_0 \cup D_1 \cup D_2 \cup \dots \cup D_m$ sú

informované po T_m krokoch, pričom platí

$$T_m \leq \sum_{j=0}^m \left[\log_{\alpha} \left(\frac{1}{d_{j-1}} \right) + \log_{\alpha} \left(\frac{1}{d_j} \right) + 1 \right]$$

V okrajovom prípade kladieme d_{-1} rovné jednej.

Dôkaz vykonáme indukciou na m . Množina D_0 obsahuje iniciátora a je informovaná už na začiatku šírenia správy. V indukčnom kroku predpokladajme, že po

$$T_k \leq \sum_{j=0}^k \left[\log_{\alpha} \left(\frac{1}{d_{j-1}} \right) + \log_{\alpha} \left(\frac{1}{d_j} \right) + 1 \right]$$

krokoch sú informované práve vrcholy

$$D_0 \cup D_1 \cup D_2 \cup \dots \cup D_k$$

Ostáva ukázať, že všetky vrcholy v množine D_{k+1} (a žiadne iné) vieme informovať v čase najviac (ďalších)

$$\left[\log_{\alpha} \left(\frac{1}{d_k} \right) + \log_{\alpha} \left(\frac{1}{d_{k+1}} \right) + 1 \right]$$

krokov. Stačí si uvedomiť, že

- do každého vrchola z množiny D_{k+1} vedie aspoň jedna hrana z množiny D_k a nevedú žiadne hrany z množín s nižším indexom,
- pokiaľ nebudú informované všetky vrcholy množiny D_{k+1} , algoritmus nevyberá žiadne hrany vedúce do iného vrchola

Označme w_l celkový počet hrán vedúcich z niektorého vrchola množiny D_k do niektorého neinformovaného vrchola z množiny D_{k+1} čase $T_k + l$ (počet neinformovaných vrcholov v množine D_{k+1} sa znižuje, rovnako aj počet hrán). Potom platí

$$\begin{aligned} w_0 &\leq d_k d_{k+1} \\ w_{l+1} &\leq \alpha w_l \end{aligned}$$

Pre čas x dostávame horný odhad

$$w_x \leq \alpha^x d_k d_{k+1}$$

položme pravú stranu rovnú jednej - v čase

$$\left\lceil \log_\alpha \left(\frac{1}{d_k} \right) + \log_\alpha \left(\frac{1}{d_{k+1}} \right) \right\rceil$$

sa bude v grafe nachádzať maximálne jedna hrana medzi informovaným vrcholom množiny W_k a neinformovaným vrcholom množiny D_{k+1} - tento bude nevyhnutne informovaný najneskôr v najbližšom kroku.

Výraz, ako je uvedený v znení vety, je iba mierne zvýšeným horným odhadom. Každý zo sčítancov obsahujúci d_i sa v sume vyskytuje nanajvyš dvakrát.

□

Veta 4.13. *Existuje konštanta k taká, že pre ľubovoľný súvislý graf G o n vrchoch a iniciátorom w platí*

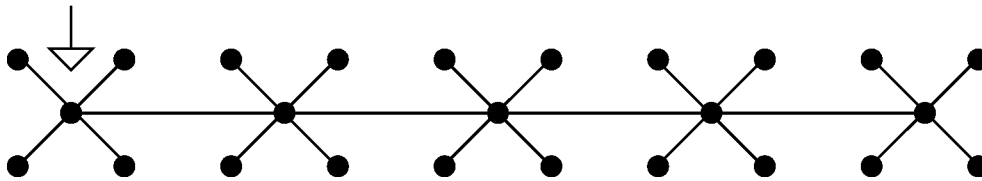
$$T_G \leq k \text{exc}_G(w) \log n$$

Výskyt proporcionálneho počtu chybných liniek môže (v prípade optimálneho algoritmu - alebo aj v prípade algoritmu dopredného šírenia) pridať prinajhošom multiplikatívny $\log n$ faktor k času šírenia správy v prostredí bez chybných liniek. Okrem iného to zhora ohraničuje „neoptimálnosť“ algoritmu dopredného šírenia. Tvrdenie je triviálnym dôsledkom vety 4.12.

Veta 4.14 (algoritmus dopredného šírenia nie je optimálny). *Existuje trieda grafov ξ taká, že platí $T_\xi^{AD} = \omega(T_\xi)$*

Dôkaz. Zoberme triedu grafov $\xi = (\xi_1, \xi_2, \xi_3, \dots)$ nasledovne (viď obrázok 6). Hviezdička z i vrcholov pozostáva z jedného centrálného vrchola a $i - 1$ vrcholov pripojených na centrálny vrchol (navzájom neprepojených). Graf ξ_i pozostáva z i hviezdíčiek, centrálny vrcholy jednotlivých hviezdíčiek sú prepojené do línie. Celý graf pozostáva z i^2 vrcholov. Iniciátorom grafu je centrálny vrchol na ľavej krajnej hviezdíčke. Štandardným prístupom dostaneme

$$T_\xi^{AD} = \Theta(i \log i) = \Theta(\sqrt{n} \log n)$$



Obr. 6: Neoptimálnosť algoritmu dopredného šírenia.
Príklad grafu ξ_5 z dôkazu tvrdenia 4.14. Iniciátor je označený šípkou.

Lepším prístupom je informovať najprv všetky centrálné vrcholy na línii a následne (paralelne) všetky ostatné vrcholy na hviezdíčke. Tým dostávame odhad

$$T^{AD} = O(i + \log i) = O(\sqrt{n})$$

□

Z vety 4.13 vyplýva, že prezentovaných $\log n$ je (asymptoticky) najväčšia medzera medzi trvaním algoritmu dopredného šírenia a optimálneho algoritmu, akú môžeme dosiahnuť. Okrem iného to znamená, že požiadavka informovať najprv centrálné vrcholy vedie k asymptoticky optimálnemu času šírenia správy.

4.3 Algoritmus spätného šírenia

Algoritmus spätného šírenia je definovaný iba pre acyklické grafy. Pracuje v etapách podobne ako predchádzajúci algoritmus dopredného šírenia, každá etapa môže trvať viac krokov. Rozdielny prístup je vo výbere vrcholov, ktoré majú byť informované v jednotlivých etapách. Kým v prípade algoritmu dopredného šírenia berieme do úvahy vzdialenosť od iniciátora, v prípade algoritmu spätného šírenia berieme do úvahy najdlhšiu cestu do listu (v zmysle orientovaného zakoreneného stromu).

V acyklických grafoch správy prechádzajú po hranách dopredu určeným smerom (smer závisí iba od grafu, nie od algoritmu ani súpera). Môžeme preto bez ujmy na všeobecnosti považovať hrany za orientované smerom ku vrcholom vzdialenejších od iniciátora.

Definícia 4.15 (spätná vzdialenosť vrchola). Nech $G = (V, E)$ je zakorenený strom s koreňom w . Spätnou vzdialenosťou $h(v)$ vrchola v rozumieme

délku najdlhšej orientovanej cesty z vrchola v do niektorého listu. Definované induktívne

- $h(v) = 0$, ak v je listom
- $h(v) = \max\{h(u) \mid u \text{ je synom } v\} + 1$, ak v nie je listom

Definícia 4.16 (algoritmus spätného šírenia). Nech $G = (V, E)$ je graf s počiatočným vrcholom w . Označme $h_i(V)$ množinu tých vrcholov grafu G , ktorých spätná vzdialenosť v grafe G je i . Potom algoritmus spätného šírenia pre graf G je zobrazenie $A_S : 2^V \mapsto 2^E$ také, ktoré v každom okamihu vyberá za aktívne tie hrany spomedzi potenciálnych, ktoré vedú do vrcholov s maximálnou spätnou vzdialenosťou. Formálne, pre informovaný vrchol $u \in U$ a neinformovaný vrchol $u' \in V \setminus U$ platí

$$(u, u') \in A_S(U) \text{ práve vtedy, keď } h(u') = \max\{h(x) \mid x \in V \setminus U\}$$

Veta 4.17. Nech $G = (V, E)$ je acyklický súvislý graf s iniciátorom w . Potom existuje konštanta c nezávislá od G taká, že platí

$$T_G^{A_S} \leq c \sum_{i=0}^{exc_G(w)} (\log |h_i(V)| + 1)$$

Dôkaz vety je analogický ako v prípade 4.12. Môžeme navyše využiť skutočnosť, že do každého neinformovaného vrchola vedie najviac jedna hrana z informovaného vrchola.

Veta 4.18 (asymptotická optimálnosť algoritmu spätného šírenia). Existuje konštanta c taká, že pre všetky acyklické grafy G platí

$$T_G^{A_S} \leq c T_G$$

Dôkaz. Nech O je optimálny algoritmus pre graf G s daným iniciátorom. Potrebujeme spraviť nasledovné

1. Na základe podoby grafu zostrojíme súpera X pre optimálny algoritmus, v súlade s proporcionalitou modelu.
2. Podľa šírenia správy na grafe G s algoritmom O a súperom X rozdelíme hrany na množiny O_1, O_2, \dots, O_k . Množina O_i obsahuje bezchybné hrany v i -tom kroku (hrany, po ktorých bola prenesená správa).

3. Každéj hrane v grafe G priradíme cenu $c(e)$. Priradenie ceny vykonáme na základe podoby grafu G a rozkladu E na množiny O_1, O_2, \dots, O_k .
4. Ukážeme, že s daným súperom X optimálny algoritmus prenesie v každom kroku správy po hranách v celkovej cene najviac 1, teda že platí

$$\sum_{e \in O_i} c(e) \leq 1$$

5. Na záver ukážeme, že existuje c nezávislé od G také, že platí

$$c \sum_{e \in E} c(e) \geq T_G^{As}$$

Z uvedených vlastností potom vyplýva

$$T_G^{As}(n) \leq c T_G(n)$$

Pre jednoduchosť zápisu predpokladajme $\alpha = 1/2$. Na záver dôkazu podáme diskusiu zmien pre všeobecné $0 < \alpha < 1$.

Usporiadanie podľa spätnej vzdialenosti. Uvažujme ľubovoľné (zafixované) usporiadanie vrcholov V . Zostrojme ku nemu usporiadanie hrán E , ktoré rešpektuje spätnú vzdialenosť. Usporiadanie je dané reláciou

$$(u_1, v_1) <_S (u_2, v_2)$$

čo platí práve vtedy, keď je splnená aspoň jedna z podmienok

- $h(v_1) > h(v_2)$
- $h(v_1) = h(v_2)$ a zároveň $v_1 < v_2$ v usporiadaní vrcholov

Zápis (u, v) symbolizuje orientovanú hranu z vrchola u do vrchola v (v zmysle uvedenej dohody).

Konštrukcia súpera. Pri usporiadaní hrán podľa algoritmu spätneho šírenia sme predpokladali iba znalosť grafu G . Využijeme teraz definované usporiadanie na to, aby sme skonštruovali súpera X pre optimálny algoritmus. Súper X v každom kroku bude blokovať maximálny počet hrán v súlade

s proporcionalitou modelu. Blokuje prednostne tie hrany, ktoré sú menšie v usporiadaní podľa algoritmu spätneho šírenia. Teda pre každú chybnú hranu e_1 a každú bezchybnú hranu e_2 (v danom kroku) platí $e_1 <_S e_2$. Nevyžadujeme, aby takto definovaný súper spôsobil maximálny možný čas šírenia správy pre algoritmus O .

Usporiadanie podľa optimálneho algoritmu. Na základe šírenia správy daného optimálnym algoritmom O so súperom X máme hrany rozdelené do množín O_1, O_2, \dots, O_n . Použijeme usporiadanie podľa spätnej vzdialenosti na to, aby sme hrany usporiadali v rámci jednotlivých množín O_i . Usporiadanie podľa optimálneho algoritmu je teda dané reláciou

$$e_1 <_O e_2$$

čo platí práve vtedy, keď je splnená aspoň jedna z podmienok

- $e_1 \in O_i, e_2 \in O_j$ a platí $i > j$
- $e_1 \in O_i, e_2 \in O_i$ a platí $e_1 <_S e_2$

Ocenenie hrany. Ocenenie hrany (u_1, v_1) prebieha nasledovne. Zoberme všetky hrany, ktoré patria do tej istej vrstvy v zmysle spätnej vzdialenosti - teda hrany (u_2, v_2) také, že $h(v_1) = h(v_2)$. Cena zvolenej hrany (u_1, v_1) je daná poradím v usporiadaní tejto množiny podľa relácie $<_O$. Nech k je počet „menších“ hrán, t.j.

$$k = \#\{(u_2, v_2) \in E \mid h(v_1) = h(v_2) \text{ a platí } (u_2, v_2) <_O (u_1, v_1)\}$$

Cenou hrany (u_1, v_1) nazveme hodnotu $1/(k+1)$.

Súčet cien v jednom kroku. Ukážme, že pre ľubovoľný krok optimálneho algoritmu O_i platí

$$\sum_{e \in O_i} c(e) \leq 1$$

Nech (u, v) je hrana z množiny O_i s najväčšou cenou. Cenou hrany (u, v) môžeme zapísať ako $1/(k+1)$ pre nejaké k . Ukážme, že počet hrán v množine O_i je najviac $k+1$. Využijeme pritom fakt, že v grafe je najviac k menších hrán vzhľadom na usporiadanie $<_O$

Dôkaz vykonáme sporom. Nech platí $|O_i| \geq k+2$. Potom súper zablokoval aspoň $k+1$ hrán³. Označme prvých $k+1$ blokovaných hrán

$$(u_1, v_1), (u_2, v_2), \dots, (u_{k+1}, v_{k+1})$$

a uvažujme, ktoré hrany mohli byť blokované

- Žiadna z blokovaných hrán nemohla viesť do vrchola s menšou spätnou vzdialenosťou ako $h(v)$. V opačnom prípade by súper uprednostnil zablokovanie hrany (u, v) .
- Ku každej blokovanej hrane vedúcej do vrchola so spätnou vzdialenosťou $h(v_i)$, kde $h(v_i) > h(v)$, existuje neaktívna hrana vedúca do vrcholu so spätnou vzdialenosťou rovnou $h(v)$. Táto hrana je neaktívna, preto je menšia v usporiadaní $<_O$ oproti hrane (u, v) .
- Každá blokovaná hrana vedúca do vrchola so spätnou vzdialenosťou $h(v)$, kde $h(v_i) = h(v)$, je v usporiadaní $<_O$ menšia oproti hrane (u, v) .

Z predpokladu, že súper blokuje aspoň $k+1$ hrán, dostávame aspoň $k+1$ hrán vedúcich do vrchola so spätnou vzdialenosťou $h(v)$, ktoré sú menšie v usporiadaní $<_O$. Cena hrany (u, v) by tak mohla byť najviac $1/(k+2)$. Z uvedeného vyplýva, že súčet cien hrán z množiny O_i je najviac 1.

Súčet cien všetkých hrán Spočítaním cien všetkých hrán v grafe dostaneme

$$\sum_{e \in E} c(e) = \sum_i \sum_{e \in E_i} c(e) = \sum_i H_{|E_i|}$$

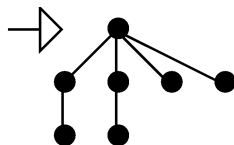
kde E_i je množina hrán vedúca do vrcholov so spätnou vzdialenosťou i a H_i je i -te harmonické číslo. S použitím známeho rozvoja pre harmonické čísla

$$H_n = \ln n + \gamma + O(n^{-1})$$

dostávame zápis

$$\sum_{e \in E} c(e) = \Theta \left(\sum_i (\log |E_i| + 1) \right)$$

³Nie $k+2$, pretože $\lfloor (1/2)(2k+3) \rfloor = k+1$



Obr. 7: Neoptimálnosť algoritmu spätného šírenia I.
Príklad grafu, na ktorom je algoritmus spätného šírenia neoptimálny.

Z tvrdenia 4.17 poznáme odhad pre trvanie algoritmu A_S v rovnakej podobe a dostávame požadovaný výsledok

$$c \sum_{e \in E} c(e) \geq T_G^{A_S}$$

pre nejaké c nezávislé od G (nie nutne rovnaké, ako vo vete 4.17).

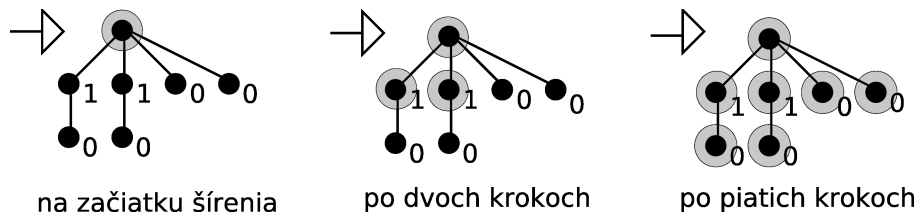
Zovšeobecnenie pre ľubovoľné α Hodnotu $\alpha = 1/2$ sme využili pri dôkaze súčtu cien hrán v jednom kroku šírenia správy. Pre inú hodnotu by sa tento súčet zmenil o multiplikatívnu konštantu závislú od α ; v konečnom dôsledku by to znamenalo iba inú hodnotu konštanty c v porovnaní časov.

□

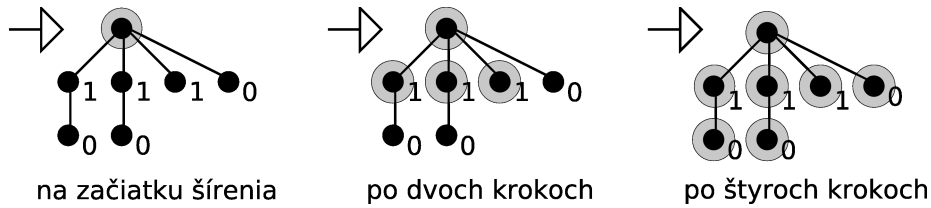
Aby sme doplnili predchádzajúce tvrdenie o asymptotickej optimálnosti algoritmu spätného šírenia, ukážme, že nemusí byť optimálny. Príkladom pre $\alpha = 1/2$ môže byť graf uvedený na obrázku 7. Čas dosiahnutý pomocou algoritmu spätného šírenia je 5, šírenie správy je zobrazené na obrázku 8. Alternatívny algoritmus pracuje podobne ako algoritmus spätného šírenia, v ktorom jeden z listových vrcholov preradíme do vyššej vrstvy, ako je zobrazené na obrázku 9. Takýto algoritmus dosiahne čas šírenia správy 4. Využívali sme fakt, že v definícii modelu sa implicitne nachádza dolná celá časť - maximálny počet blokováných hrán pri počte $2k$ a $2k + 1$ aktívnych hrán je rovnaký.

Nie je zložité zopakovať uvedenú štruktúru, aby sme zvýšili rozdiel medzi optimálnym algoritmom a algoritmom spätného šírenia. Pre $\alpha = 1/2$ a požadovaný rozdiel k môže vyzeráť nasledovne. Vrcholy grafu sú reprezentované množinou

$$\{(i, j) \mid i \in \{1, \dots, 2k\} \text{ a } j \in \{1, \dots, 2^i\}\} \cup \{w\}$$



Obr. 8: Neoptimálnosť algoritmu spätného šírenia II.
Šírenie správy na grafe s algoritmom spätného šírenia.



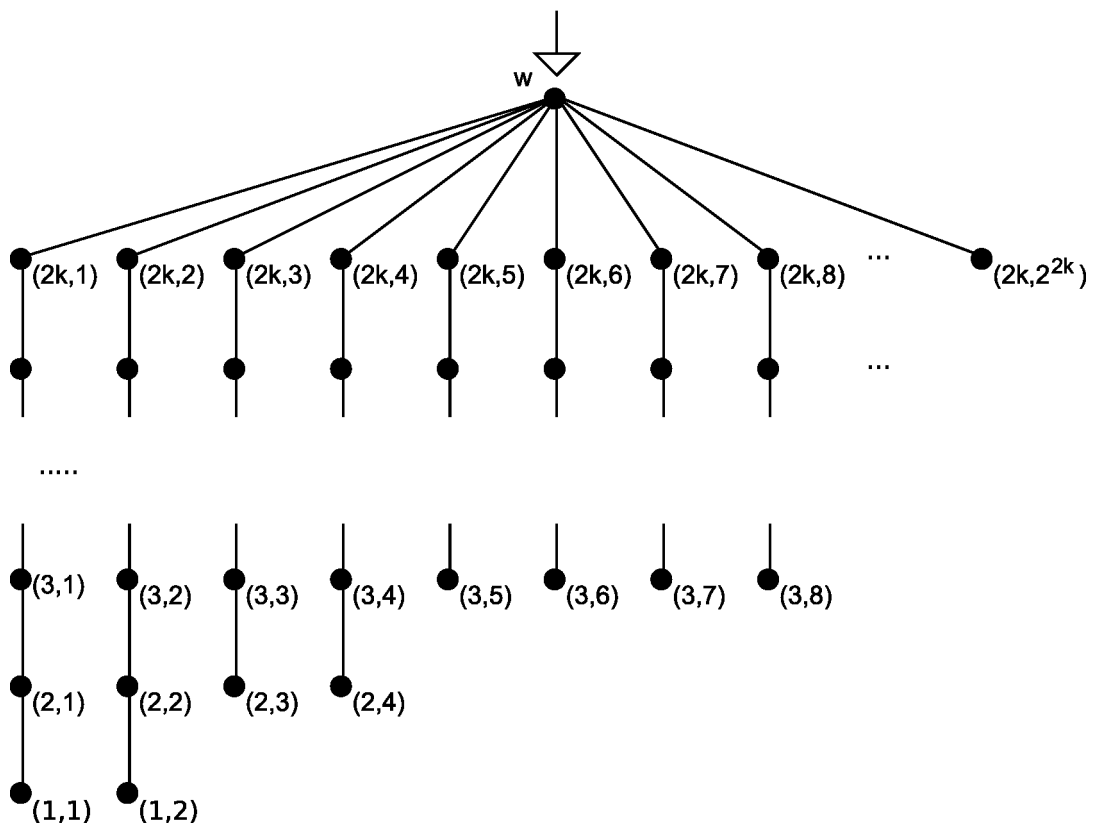
Obr. 9: Neoptimálnosť algoritmu spätného šírenia III.
Šírenie správy na grafe s alternatívnym algoritmom.

Medzi vrcholmi (i_1, j_1) a (i_2, j_2) sa nachádza hrana vtedy, keď $i_2 = i_1 + 1$ a $j_1 = j_2$. Iniciátor w je spojený hranou so všetkými vrcholmi vo vrstve $2k$. Príklad grafu je na obrázku 10. Čas šírenia správy na takomto grafe v prípade algoritmu spätného šírenia je

$$\sum_{1 \leq i \leq 2k} (i + 1) = k(2k + 3)$$

Alternatívny algoritmus - podobne ako v predchádzajúcom zjednodušenom prípade - pracuje rovnako ako algoritmus spätného šírenia s mierne upravenými množinami vrcholov $h_i(V)$. Z každej párnej vrstvy je odobratý jeden vrchol, ktorý je presunutý do nasledujúcej vrstvy. Pre každé i zoberieme jeden vrchol v z množiny $h_{2i}(V)$ a presunieme⁴ ho do množiny $h_{2i+1}(V)$. Informovanie vrstiev $h_{2i}(V)$ a $h_{2i+1}(V)$ v novej podobe trvá spolu o jeden krok kratšie oproti algoritmu spätného šírenia. Čas šírenia optimálneho algoritmu je teda najviac $k(2k + 2)$ krokov.

⁴Presunutie ovplyvňuje iba poradie, v ktorom budú vrcholy informované. V tomto okamihu prestáva platiť, že všetky vrcholy v množine $h_i(V)$ majú spätnú vzdialenosť i



Obr. 10: Neoptimálnosť algoritmu spätného šírenia IV.
 Zvyšovanie rozdielu medzi optimálnym algoritmom a algoritmom spätného šírenia - príklad konštrukcie grafu.

4.4 Neoptimálnosť greedy algoritmu

V predchádzajúcom texte sme ukázali, že greedy algoritmus nie je ani asymptoticky optimálny. Ponúkajú sa dve prirodzené otázky

1. Aký veľký môže byť rozdiel v trvaní greedy algoritmu a optimálneho algoritmu?
2. Aký zložitý graf potrebujeme, aby sme dokázali skonštruovať dostatočne veľkú „medzeru“ v trvaní?

V nasledujúcej časti sa pokúsime priblížiť k odpovedi na prvú otázku. Zo základných vlastností modelu vyplýva, že najhorší čas ľubovoľného algoritmu (na súvislom grafe) je $n - 1$, najlepší garantovateľný čas je naproti tomu $\Theta(\log n)$. Nie je nám známe, či existuje graf s takouto charakteristikou (greedy algoritmus, optimálny algoritmus).

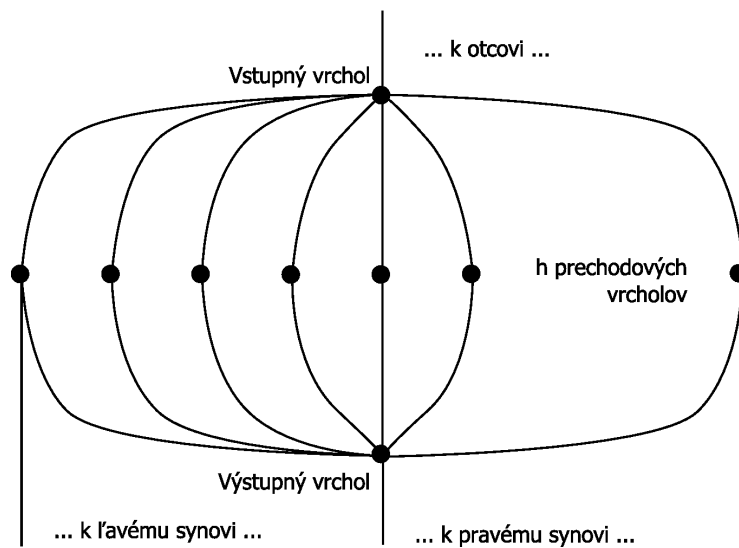
Predstavme aspoň ďalší - možno trochu umelo skonštruovaný - príklad, ktorého hlavným cieľom je posunúť medzeru bližšie k triviálnej hornej hranici n verzus $\log n$. Pre jednoduchosť predpokladajme, že hodnota α určujúca proporcionalitu modelu je $1/2$, zovšeobecnenie pre iné α prediskutujeme v závere dôkazu.

Základom pre konštrukciu grafu je úplný binárny strom hĺbky h . Zoberme nasledovný „glóbus“⁵ podľa obrázku 11, pozostávajúci z jedného vstupného a jedného výstupného vrchola, medzi ktorými sa nachádza h vrcholov. Nahraďme každý vrchol takýmto glóbusom, nahradenie v binárnom strome hĺbky h vytvorí graf s $n =_{nt} (2^h - 1)(h + 2)$ vrcholmi. Príklad grafu pre $h = 3$ je na obrázku 12. Pre optimálny algoritmus platí horný odhad získaný z algoritmu dopredného šírenia

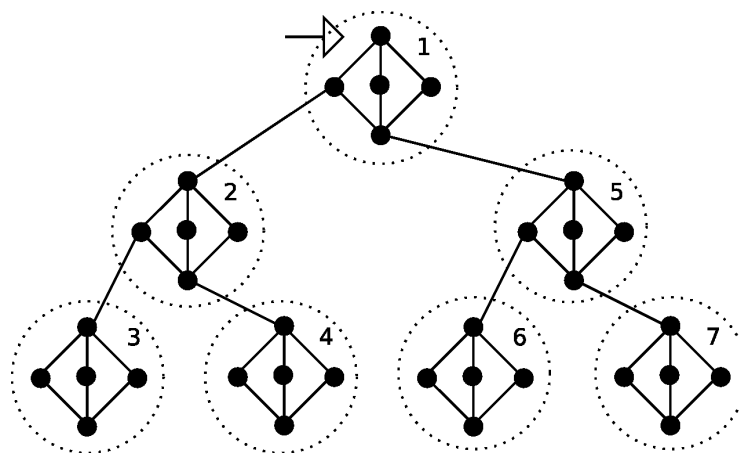
$$T_G = O((h - 1) \log[(2^h - 1)(h + 2)]) = O(h^2) = O(\log^2 n)$$

Ukážme, že trvanie greedy algoritmu na zostrojenom grafe je $\Omega(n / \log n)$. Súper nech pracuje nasledovne. Použijeme prefixové očíslovanie vrcholov pôvodného binárneho stromu (dané prehľadávaním do hĺbky). Po nahradení

⁵Pomenovanie glóbus vychádza z podobnosti grafu a glóbusu so znázornenými populníkmi



Obr. 11: Neoptimálnosť greedy algoritmu III.
 Obrázok schematicky zobrazuje glóbus použitý pri konštrukcii grafu.



Obr. 12: Neoptimálnosť greedy algoritmu IV.
 Príklad grafu s očíslovaním glóbusov pre $h = 3$.

vrcholov glóbusmi máme očíslované glóbusy, ktoré budú informované v poradí podľa rastúceho čísla. Informovanie jedného glóbusu g trvá dva kroky. Pred prvým krokom prislúchajúcim ku glóbusu g

- sú informované všetky vrcholy glóbusov s nižším číslom (vstupné, prechodové aj výstupné)
- je informovaný vstupný vrchol glóbusu g

Počas prvého kroku prislúchajúceho ku glóbusu g sú prenesené správy vedúce do prechodových vrcholov glóbusu g . Počas druhého kroku sú prenesené správy vedúce do výstupného vrcholu glóbusu g a jedna správa po hrane vedúcej do vstupného vrcholu nasledujúceho glóbusu g' (podľa zvoleného očíslovania), pokiaľ g nie je posledný. Po druhom kroku sú splnené uvedené podmienky pre glóbus g' . Trvanie šírenia správy je dvojnásobkom počtu glóbusov, teda $2(2^h - 1)$.

Ostávajú ukázať dve záležitosti. V prvom rade to, že takto popísaný súper je v súlade s proporcionalitou modelu. V každom kroku šírenia správy je prenesených h správ (v prvých krokoch, kedy sú informované prechodové vrcholy) alebo $h + 1$ správ (v druhých krokoch, kedy je informovaný výstupný vrchol a vstupný vrchol nasledujúceho glóbusu - s výnimkou druhého kroku prislúchajúceho k poslednému glóbusu). V žiadnom kroku nie sú blokované hrany v rámci glóbusu (iba hrany medzi glóbusmi, ktoré pochádzajú z pôvodného binárneho stromu) - z vlastnosti prefixového číslovania môže byť týchto najviac $h - 1$. Proporcionalita ostáva teda zachovaná. Pokiaľ je α zvolené inak ako $1/2$, stačí položiť počet prechodových vrcholov $\lceil h(1 - \alpha)/\alpha \rceil$. Na záver sa ešte uistíme, že v druhom kroku je možné informovať vstupný vrchol nasledujúceho glóbusu g' - vyplýva z toho, že

- predchádzajúci glóbus g má aspoň o dve menšie číslo (ak g' je pravým synom g) a jeho výstupný vrchol je informovaný
- predchádzajúci glóbus g má o jedno menšie číslo (ak g' je jeho ľavým synom) a všetky prechodové vrcholy g sú informované

Trvanie greedy algoritmu na tomto grafe je teda

$$T_G^{AG} \geq 2(2^h - 1) = \Omega\left(\frac{n}{\log n}\right)$$

5 Stromy s konštantnou excentricitou iniciátora

V nasledujúcej časti sa zamerajme podrobnejšie na triedu stromov s konštantnou excentricitou iniciátora. Porovnáme čas šírenia správy s jednotlivými algoritmami a budeme hľadať grafy v rámci uvedenej triedy, na ktorých je rozdiel výrazný. Keďže trieda obsahuje pomerne rôznorodé grafy, budeme sa zaujímať o horné a dolné odhady, prípadne „najťažších“ reprezentantov. Hľadáme tým odpoveď na otázku nastolenú v prechádzajúcej kapitole - aký zložitý graf potrebujeme na to, aby sme dosiahli „dost veľkú“ medzeru v trvaní greedy algoritmu a optimálneho algoritmu.

Pre jednoduchosť výkladu sa budeme miestami prikláňať k ponímaniu stromu ako štruktúry zakorenenej v iniciátorovi, ktorej hrany sú orientované. Takáto predstava nič nemení na pôvodnej definícii algoritmu a modelu (hrany sú obojsmerné). Hĺbka stromu v tomto prípade značí excentricitu iniciátora.

Definícia 5.1. Nech $T = (G_1, G_2, G_3, \dots)$ je trieda acyklických súvislých grafov s iniciátormi w_1, w_2, w_3, \dots a nech existuje k také, že pre každé i excentricita vrchola w_i v grafe G_i je najviac k . Triedu T nazývame triedou stromov s konštantnou excentricitou iniciátora.

5.1 Optimálny algoritmus

V predchádzajúcej časti sme predstavili algoritmus dopredného šírenia. Tento algoritmus nemusí byť vždy optimálny, dáva nám však dostatočný horný odhad pre trvanie optimálneho algoritmu. Navyiac, tento odhad ostáva rovnaký aj pre cyklické grafy s konštantnou excentricitou iniciátora. Keďže excentricita iniciátora je konštantná, na základe vety 4.12 a 4.13 platí

$$T_T^{AD} = \Theta(\log n)$$

5.2 Dolný odhad pre greedy algoritmus

Veta 5.2 (Dolný odhad pre greedy algoritmus). Nech T je trieda všetkých acyklických súvislých grafov s maximálnou excentricitou iniciátora k . Potom platí

$$T_T^{AG} = \Omega\left(n^{\frac{k-1}{k}}\right)$$

Dôkaz. Zoberme podtriedu triedy T - nech ξ sú stromy s hĺbkou k a všade rovnakým vetvením (každý nelistový vrchol má rovnaký počet synov). Inicialátorom je koreň stromu. Postupnosť grafov dostávame zvyšovaním vetvenia nelistových vrcholov. Stačí ukázať, že na takto zvolenej podtriede platí požadovaný odhad

$$T_{\xi}^{AG} = \Omega\left(n^{\frac{k-1}{k}}\right)$$

Nech G je graf z triedy ξ taký, že nelistové vrcholy majú i synov. Z dôvodu jednoduchšieho značenia v ďalšom pokračovaní dôkazu predpokladajme $\alpha = 1/2$ a označme

$$\vartheta =_{nt} \left\lfloor \frac{i}{k+1} \right\rfloor$$

Súper bude pracovať nasledovne:

Prvá fáza. V prvej fáze súper umožní informovať „ľavú časť“⁶ stromu. Blokuje ϑ hrán na prvej úrovni (vychádzajúce z koreňa). Ostatné hrany neblokuje, správa sa šíri po poschodiach a zaplavuje ľavú časť stromu smerom ku listom. V poslednom kroku, v ktorom by správy dosiahli listy, odblokuje súper aj jednu hranu na prvej úrovni a prechádza do druhej fázy.

Druhá fáza. V druhej fáze súper pracuje nasledovne

- **P1** - nájde najvyššiu úroveň (v zmysle najbližšie k listom), na ktorej sa nachádzajú aktívne hrany.
- **P2** - pokiaľ táto úroveň nie je posledná (vedúca k listom), zablokuje ϑ hrán na tejto úrovni a všetky ostatné aktívne hrany na iných (nižších) úrovniach. Ostatné hrany na tejto úrovni sú voľné a bude po nich prenesená správa.
- **P3** - pokiaľ táto úroveň je posledná (vedúca k listom), prepustí všetky správy na poslednej úrovni. Zároveň nájdi najväčšie $q < k$ také, že na úrovni q sa nachádza aspoň jedna aktívna hrana. Ak q existuje, prepusti správu po jednej hrane na úrovni q . Všetky ostatné hrany sú blokované. Ak q neexistuje, je to posledný krok algoritmu (všetky hrany v tomto kroku sú voľné a po vykonaní tohoto kroku po všetkých hranách prešla správa).

⁶Pojem „ľavá“ je použitý len z dôvodu lepšej predstavy.

Vlastnosti súpera. Pri dôkaze tvrdenia budeme využívať nasledovné vlastnosti

- **V1** - súper neblokuje žiadne hrany na poslednej úrovni (tzn. hrany vedúce do listov)
- **V2** - na každej úrovni s výnimkou poslednej blokuje súper najviac ϑ hrán, celkovo teda blokuje najviac $(k - 1) \vartheta$ hrán
- **V3** - nech p je najvyššia úroveň, na ktorej sa nachádzajú nejaké aktívne hrany. Potom sa na úrovni p nachádza aspoň i aktívnych hrán

Vlastnosť V1 vyplýva priamočiaro z definície činnosti súpera. Vlastnosti V2 a V3 dokážeme indukciou na počet krokov výpočtu druhej fázy. V počiatočnom stave prešla správa po jedinej hrane na prvej úrovni. Predpokladajme dostatočne veľké $k \geq 3$. Na začiatku druhej fázy je na druhej úrovni aktívnych i hrán, čím je splnená vlastnosť V3. Na žiadnej vyššej úrovni sa totiž aktívne hrany nenachádzajú. Súper aplikuje pravidlo P2, na základe ktorého bude blokovaných na druhej úrovni ϑ hrán a na prvej úrovni všetky hrany - z popisu prvej fázy algoritmu bude týchto $\vartheta - 1$, čím je splnený invariant V2. V indukčnom kroku uvažujme dva prípady

Prvý prípad - v prechádzajúcom kroku bolo aplikované pravidlo P2 na úrovni p . Potom sa ale na žiadnej inej úrovni vyššej ako p nenachádzala v predchádzajúcom kroku žiadna aktívna hrana. Keďže správy tiekli iba po úrovni p , ani teraz sa na úrovni vyššej ako $p + 1$ nenachádza žiadna aktívna hrana (greedy algoritmus) a na úrovni $p + 1$ sa ich nachádza aspoň i (vetvenie). Tým je dokázaný invariant V3. Súper v tomto kroku blokuje rovnaký alebo menší počet hrán ako v prechádzajúcom kroku na úrovniach menších ako p - korektné z IP. Ďalej blokuje $(k - 1) \vartheta$ hrán na úrovni p na základe pravidla P2 aplikovanom v minulom kroku a rovnaký počet hrán (alebo dokonca žiadnu hranu) na úrovni $p + 1$ na základe pravidla P2 alebo P3 aplikovanom v tomto kroku.

V druhom prípade rozoberme možnosť, kedy v predchádzajúcom kroku bolo aplikované pravidlo P3 a poslaná jedna správa na úrovni p . S takýmto označením je dôkaz vlastností V2 a V3 obdobný ako v predchádzajúcom odseku.

Korektnosť súpera. Na základe dokázaných vlastností je celkovo blokováných hrán najviac $(k - 1) \vartheta$ a správa prejde aspoň po

$$i - \vartheta = \left\lceil i \frac{k}{k+1} \right\rceil \geq (k - 1) \vartheta$$

hranách. Súper je korektný v zmysle 1/2 proporcionálnosti modelu.

Trvanie šírenia správy. Dôkaz trvania sa bude opierať o dodatočné vlastnosti druhej fázy algoritmu.

Veta 5.3. *Neexistujú dve rôzne úrovne $q_1 < k$, $q_2 < k$ a krok šírenia správy, pri ktorom by správa tiekla súčasne po hranách na úrovni q_1 aj q_2 .*

Dôkaz. Vlastnosť vyplýva priamo z definície súpera, vlastností P1 až P3. Na základe znenia vety (podmienok $q_1 < k$ a $q_2 < k$) neuvažujeme poslednú úroveň. \square

Veta 5.4. *Nech T_i označuje počet krokov šírenia správy v rámci druhej fázy, v ktorých prešla aspoň jedna správa po úrovni i . Potom pre prvú úroveň platí*

$$T_1 \geq \vartheta$$

Dôkaz. Zoberme si ľubovoľný krok druhej fázy, v ktorom prešla aspoň jedna správa po prvej úrovni. Ukážme si, že táto správa bola práve jedna. Z toho dostávame triviálne platnosť tvrdenia.

Podľa ktorého pravidla konal súper? Na základe pravidla P3 - pre netriviálne k - by prešla jediná správa. Pokiaľ konal na základe pravidla P2, musela byť najvyššia úroveň s aktívnymi hranami práve úroveň 1. Ktorá úroveň s aktívnymi hranami bola v predchádzajúcom kroku najvyššia? Pokiaľ to nebola úroveň k , súper pracoval v predchádzajúcom kroku na základe pravidla P2 a v tomto kroku nemôže byť najvyššou úrovňou s aktívnymi hranami úroveň 1. Pokiaľ to bola úroveň k , súper pracoval v prechádzajúcom kroku na základe pravidla P3 a v tomto kroku opäť existuje úroveň s aktívnymi hranami vyššia ako 1. \square

Veta 5.5. *Pre všetky $m < (k - 1)$ platí*

$$T_{m+1} \geq \vartheta T_m$$

Dôkaz. Vždy, keď prejde nejaká správa úrovňou m , vznikne na úrovni $m+1$ aspoň i nových aktívnych hrán. Postačuje ukázať, že „spracovanie“ týchto nových aktívnych hrán trvá aspoň ϑ krokov; formálnejšie povedané, že existuje aspoň ϑ rôznych krokov šírenia správ, v rámci ktorých tečú správy iba po týchto nových aktívnych hranách a žiadnych iných na úrovni $m+1$.

Najprv sa presvedčíme, že pokiaľ nebudú tieto hrany „spracované“, nepotečie žiadna správa hranou na nižšej úrovni. Toto vyplýva priamo z pravidiel P2 a P3.

V ďalšom uvažujeme, na základe ktorých pravidiel bude súper konať. Ďalší krok (prvý po vytvorení aktívnych hrán na úrovni $m+1$) bude vykonaný na základe pravidla P2 a ostane ϑ hrán. Z rovnakého dôvodu ako v prechádzajúcom tvrdení - všetky ďalšie kroky, v rámci ktorých potečie správa po niektorej zo zostávajúcich aktívnych hrán, budú vykonané na základe pravidla P3 - a teda v jednom kroku tečie maximálne jedna správa po úrovni $m+1$. Týchto krokov musí byť minimálne ϑ . \square

Pokračovanie dôkazu vety 5.2. Pre dolný odhad času šírenia správy postačuje zobrať hodnotu

$$T_{k-1} = \vartheta^{k-1} = \Omega\left(n^{\frac{k-1}{k}}\right)$$

pretože $i = \Theta(n^{1/k})$ a k je konštanta. V prípade inej hodnoty α postačuje príslušne upraviť hodnotu ϑ .

\square

Veta 5.6 (horný odhad). *Pre triedu ξ_k acyklických grafov s rovnomerným vetvením platí horný odhad*

$$T_{\xi_k}^{AG} = O\left(n^{\frac{k-1}{k}} \log n\right)$$

Dôkaz. Ukážme, že aspoň každých $\log n + 1$ krokov prejdú správy súčasne po $\frac{1}{2}i$ hranách.

Sporom, nech existuje postupnosť po sebe idúcich krokov šírenia správ, ktorá je dlhá aspoň $\log n + 1$ a v každom kroku prešla správa po menej ako $\frac{1}{2}i$ hranách. Zoberme prvých $\log n$ krokov. V rámci nich nemohla prejsť správa po žiadnej úrovni $q < k$. V ďalšom kroku by totiž bolo na vyššej úrovni aspoň i aktívnych hrán. Všetky správy teda museli ísť po najvyššej úrovni,

na ktorej muselo byť na začiatku postupnosti aspoň $2^{\log n} = n$ aktívnych hráčov a v prvom kroku z toho prejsť aspoň polovica. Tvrdenie vyplýva z toho, že $i = \Theta(n^{1/k})$.

Pre hodnotu α rôznu od $1/2$ postačuje požadovať prenesenie $(1 - \alpha)i$ správ v jednom kroku a prílušie upraviť základ logaritmu (čo neovplyvňuje prezentovaný výsledok v tvrdení).

□

Je potrebné si uvedomiť, že poskytnutý horný odhad hovorí o triede grafov ξ (rovnomé vetvenie), nie všeobecne o triede T . Význam tvrdenia je v zdôraznení „presnosti“ výsledku pre špeciálne zvolenú podtriedu grafov. Všeobecnejšie tvrdenie pre menej obmedzené triedy predstavíme v nasledujúcej časti.

5.3 Horný odhad pre greedy algoritmus

V predchádzajúcej časti sme ukázali, že na špeciálne vybratej triede grafov s konštantnou excentricitou iniciátora je greedy algoritmus asymptoticky neoptimálny. Táto neoptimálnosť rástla spoločne s priemerom, nikdy však nedosiahla hodnotu n . Ponúka sa prirodzená otázka - je tento jav spôsobený výberom našej špeciálnej triedy spomedzi všetkých tried grafov s konštantnou excentricitou iniciátora? Existuje trieda grafov s konštantnou excentricitou, na ktorej je trvanie greedy algoritmu asymptoticky $\Theta(n)$? V tejto časti zovšeobecníme tvrdenie 5.6 z predchádzajúceho odseku pre ľubovoľnú triedu acyklických grafov s konštantnou excentricitou iniciátora (aj keď s mierne slabším výsledkom).

Nasledujúce tvrdenie hovorí práve spomínané - pre ľubovoľnú triedu acyklických grafov s konštantnou excentricitou iniciátora je najhorší čas asymptoticky zhora ohraničiteľný funkciou n^t pre nejaké $t < 1$.

Veta 5.7 (horný odhad T_T^{AG}). *Nech $T = (T_1, T_2, T_3, \dots)$ je trieda acyklických grafov, pričom existuje k nezávislé od i také, že excentricita iniciátora v_i v grafe T_i je najviac k . Potom platí*

$$T_T^{AG} = O\left(n^{\frac{k}{k+1}}\right)$$

Dôkaz. Uvažujme ľubovoľný graf T_i a najpomalšie šírenie správy na tomto grafe. Každý krok šírenia správy „zaúčtujeme“ (podľa neskôr uvedeného kritéria) na jeden z dvoch zoznamov a ukážeme, že na žiadnom zozname sa nenachádza príliš veľa zaúčtovaných krokov. Prahovou hodnotou pre výber zoznamu bude práve $n^{1/(k+1)}$, ďalej označované ako ϑ .

Nazvime kvázipotenciálnou tú hranu, ktorá ide (v danom kroku) z informovaného do informovaného vrchola a existuje (orientovaná) cesta od iniciátora do listu, vedúca cez túto hranu, ktorá obsahuje aspoň jednu potenciálnu hranu. Kritérium pre „účtovanie“ krokov šírenia správy je nasledovné. V každom kroku zoberieme všetky aktuálne informované vrcholy. Ak niektorý z nich obsahuje v súčte aspoň ako ϑ kvázipotenciálnych alebo potenciálnych hrán, tento krok zapíšeme na červený zoznam. V opačnom prípade tento krok zapíšeme na zelený zoznam.

V každom kroku, ktorý je zapísaný na červenom zozname, bolo prenesených aspoň $(1-\alpha)\vartheta$ správ. Takýchto záznamov môže byť teda na červenom zozname najviac

$$\frac{n}{(1-\alpha)\vartheta} = O\left(n^{\frac{k}{k+1}}\right)$$

Ofarbíme na zeleno všetky hrany, po ktorých bola prenesená správa v niektorom z krokov zapísaných na zelenom zozname. Dostávame (nie nutne súvislý) podgraf grafu T_i . Ofarbíme preto aj minimálny počet ďalších hrán tak, aby sme dostali súvislý zelený podgraf grafu T_i . Ukážme, že vo vyfarbenom podgrafe má každý vrchol u najviac 2ϑ synov.

Zelené hrany mohli vzniknúť dvoma spôsobmi

- Zaradením kroku zo zeleného zoznamu. Zoberme prvý krok šírenia správy, kedy bola niektorá z jeho zelených hrán zaradená do zeleného zoznamu. V tomto čase mal každý vrchol menej ako ϑ potenciálnych hrán - z dôvodu zaradenia kroku zo zeleného zoznamu môže mať teda zafarbených najviac $\vartheta - 1$ hrán.
- Prefarbením hrany pri vytváraní súvislého grafu. Každá takáto hrana vedie do podstromu, ktorý obsahuje nejakú (aspoň jednu) zelenú hranu nachádzajúcu sa v zelenom zozname. Vyberme z každého podstromu jednu takúto hranu - dostávame množinu hrán E_Z . Predpokladajme, že vrchol u má prefarbených hrán viac ako ϑ . Zoberme krok, v ktorom prešla správa po prvej hrane spomedzi E_Z . V tomto kroku je ale

každá prefarbená hrana vrchola u potenciálna alebo kvázipotenciálna - hrana spomedzi E_Z , po ktorej ako prvej prešla správa, tak nemôže byť zaradená do zeleného zoznamu (spor).

Počet vrcholov zeleného podgrafu môžeme teda ohraničiť ako

$$\frac{(2\vartheta)^{k+1} - 1}{2\vartheta - 1} = O\left(n^{\frac{k}{k+1}}\right)$$

Počet krokov šírenia správy je daný súčtom počtu krokov zapísaných na červenom a zelenom zozname.

□

6 Hyperkocky

V nasledujúcej časti sa zamerajme podrobnejšie na triedu hyperkociek. Ukážeme, že hyperkocky (spoločne s vhodným algoritmom) sú odolnou topológiou v zmysle šírenia správy s proporcionálnym počtom chybných liniek. Podáme horný odhad na trvanie šírenia správy pomocou greedy algoritmu. Pri hornom odhade prezentujeme viacero zaujímavých techník (použitie izoperimetrických nerovností, diskrétna a spojitá analýza).

6.1 Definícia topológie a základné vlastnosti

Definícia 6.1 (d -rozmerná hyperkocka). d -rozmernou hyperkockou nazývame graf $G = (V, E)$, kde $V = \{0, 1\}^d$. Dva vrcholy sú spojené hranou práve vtedy, ak sa líšia v jednom znaku (jednom bite).

Často sa d -rozmerná hyperkocka definuje aj pomocou množiny mohutnosti d , pričom každý vrchol je reprezentovaný jednou jej podmnožinou. Dva vrcholy sú spojené hranou práve vtedy, ak symetrický rozdiel prislúchajúcich podmnožín obsahuje jeden prvok.

Definícia 6.2 (guľa s polomerom r). Majme graf $G = (V, E)$ a vrchol $w \in V$. Guľou s polomerom r a stredom w nazývame množinu

$$B_r(w) = \{v \in V \mid d(w, v) \leq r\}$$

Pokiaľ je stred gule zjavný z kontextu, budeme ho vynechávať. Ďalej hovoríme, že guľa $S(w)$ je takmer úplná guľa s polomerom r , ak platí

$$B_{r-1}(w) \subset S(w) \subseteq B_r(w)$$

V nasledujúcom texte budeme často využívať nasledovné rovnosti, ktoré triviálne vyplývajú z definície hyperkocky

- Počet vrcholov vo vzdialenosti k od iniciátora je $\binom{d}{k}$.
- Veľkosť gule o polomere r je $\sum_{i=0}^r \binom{d}{i}$.
- Veľkosť vrcholovej hranice gule o polomere r je $\binom{d}{r+1}$.

6.1.1 Vrcholová hranica na hyperkocke

Definícia 6.3 (minimálna veľkosť vrcholovej hranice). Majme graf $G = (V, E)$. Minimálnou veľkosťou vrcholovej hranice pre m vrcholov nazývame

$$d(m) = \min \{ |\delta(S)| \mid S \subseteq V \text{ a } |S| = m \}$$

Definícia 6.4 (protiľahlé vrcholy). Dva vrcholy hyperkocky u, v nazývame protiľahlé, ak im zodpovedajúce binárne vektory (z definície hyperkocky) sa líšia v každom bite.

Veta 6.5. *Nech Q_d je d -rozmerná hyperkocka a A, B sú množiny jej vrcholov. Označme*

$$d(A, B) = \min \{ d(a, b) \mid a \in A, b \in B \}$$

minimálnu vzdialenosť medzi týmito množinami vrcholov. Potom existujú dve takmer úplné gule umiestnené v protiľahlých vrcholoch A', B' také, že platí $|A| = |A'|$ a $|B| = |B'|$ a $d(A', B') \geq d(A, B)$.

Dôkaz môže čitateľ nájsť napríklad v [26].

Veta 6.6 (takmer úplná guľa s dokonalou hranicou). *Pre každé k existuje takmer úplná guľa B o k vrcholoch, pre ktorú platí*

$$|\delta(B)| = d(k)$$

Takúto takmer úplnú guľu nazvime takmer úplná guľa s dokonalou hranicou.

Dôkaz. Majme ľubovoľnú množinu S o k vrcholoch. Ukážme, že existuje takmer úplná guľa B_S o k vrcholoch taká, pre ktorú platí

$$|\delta(B_S)| \leq |\delta(S)|$$

Vytvoríme množinu $S' = V \setminus S \setminus \delta(S)$. Množina S' je konštruovaná tak, aby platilo $d(S, S') = 2$. Na základe tvrdenia 6.5 existujú dve takmer úplné gule $B_S, B_{S'}$ umiestnené v protiľahlých vrcholoch s vlastnosťou $d(B_S, B_{S'}) \geq 2$, pričom platí $|S| = |B_S|$ a $|S'| = |B_{S'}|$. Na dokončenie dôkazu stačí spomenúť, že každý vrchol z vrcholovej hranice B_S sa musí nachádzať v množine $V \setminus B_S \setminus B_{S'}$ a vrcholov v tejto množine je dostatočne málo, konkrétne

$$|V \setminus B_S \setminus B_{S'}| = |\delta(S)|$$

□

Nasledujúce dve tvrdenia nám poskytnú možnosť, ako ohraničiť veľkosť vrcholovej hranice pre takmer úplnú guľu.

Veta 6.7. *Nech $0 \leq i < \lceil d/2 \rceil$. Každá takmer úplná guľa S polomeru i má aspoň takú veľkú vrcholovú hranicu, ako množina B_{i-1}*

$$|\delta(S)| \geq |\delta(B_{i-1})|$$

Dôkaz. Tvrdenie intuitívne platí, nedá sa však priamočiaro dokázať matematickou indukciou tak, že pridávame vrcholy po jednom.

Bez ujmy na všeobecnosti predpokladajme, že stred guľí B_{i-1} a S je reprezentovaný nulovým vektorom. Označme $S' =_{nt} S \setminus B_{i-1}$ a zoberme ľubovoľný vrchol $x \in S'$. Vrchol x pozostáva z i jednotiek a $d - i$ núl. V hyperkocke susedí s $d - i$ vrcholmi, ktoré sa nenachádzajú v S ani vo vrcholovej hranici množiny B_{i-1} - môžeme invertovať každú z jeho núl. Na druhej strane, každý vrchol vzdialený $i + 1$ od iniciátora susedí najviac s $i + 1$ vrcholmi z množiny S' , pretože môžeme spätne invertovať každú z jeho jednotiek. Pre rozdiel teda platí odhad

$$|\delta(S)| - |\delta(B_{i-1})| \geq \left(\frac{d-i}{i+1} \right) |S'| - |S'|$$

Postačuje ukázať, že

$$d - i \geq i + 1$$

Čo platí práve vtedy, keď $i < \lceil d/2 \rceil$.

□

Veta 6.8. *Nech $\lceil d/2 \rceil < i \leq d$. Každá takmer úplná guľa S polomeru i má aspoň takú veľkú vrcholovú hranicu, ako množina B_i*

$$|\delta(S)| \geq |\delta(B_i)|$$

Dôkaz. Pri dôkaze tvrdenia použijeme symetriu hyperkocky a predchádzajúce tvrdenie 6.7. Označme

- $i' =_{nt} d - i$

- $S' =_{nt} V \setminus S \setminus \delta(S)$

Ak S je guľa, tvrdenie je triviálne. V opačnom prípade platí $0 \leq i' < \lceil d/2 \rceil$ a množina S' je takmer úplná guľa polomeru i' . Z predchádzajúceho tvrdenia dostávame

$$|\delta(S')| \geq |\delta(B_{i'-1})|$$

Na záver si postačuje uvedomiť, že vrcholové hranice množín S a S' , resp. množín B_i a $B_{i'-1}$ sú totožné. \square

6.1.2 Hranová hranica na hyperkocke

Definícia 6.9 (veľkosť hranovej hranice). Majme graf $G = (V, E)$. Minimálnou veľkosťou hranovej hranice pre m vrcholov nazývame

$$g(m) = \min \{|\partial S| \mid S \subseteq V \text{ a } |S| = m\}$$

Veta 6.10. *Nech Q_d je d -rozmerná hyperkocka s $n = 2^d$ vrcholmi. Potom pre funkciu $g(m)$ platí*

$$g(m) = g(n - m)$$

Dôkaz. Stačí si uvedomiť, že množiny S a $V \setminus S$ majú totožnú hranovú hranicu. \square

Veta 6.11. *Nech Q_d je d -rozmerná hyperkocka. Potom pre funkciu $g(m)$ platí*

$$g(m) \geq m(d - \log m)$$

Označme v ďalšom uvedený odhad $g'(m)$.

Tvrdenie predstavuje jeden zo štandardných výsledkov pre hyperkocky. Dôkaz možno nájsť napríklad v [27].

Veta 6.12. *Nech Q_d je d -rozmerná hyperkocka, ktorá má ku každému vrcholu priradený d -rozmerný binárny vektor (podľa definície). Označme C_k množinu prvých k vrcholov v lexikografickom usporiadaní podľa zodpovedajúcich binárnych vektorov. Potom pre ľubovoľnú množinu vrcholov V' mohutnosti k platí $|\partial C_k| \leq |\partial V'|$.*

Tvrdenie hovorí o tom, ako skonštruovať množinu s minimálnou hranovou hranicou. Dôkaz je možné nájsť napríklad v [27].

6.2 Optimálny algoritmus

V nasledujúcej časti predstavíme optimálny algoritmus šírenia správy na hyperkocke. Skonstruujeme postupnosť vrcholov, o ktorej ukážeme niektoré potrebné vlastnosti (najmä dĺžku postupnosti). Následne skonstruujeme algoritmus šírenia správy na hyperkocke a porovnáme ním získané šírenie správy s uvedenou referenčnou postupnosťou.

Referenčná postupnosť. Pre danú hyperkocku o n vrchoch definujme referenčnú postupnosť V_0, V_1, \dots, V_m nasledovne

1. $V_0 = \{w\}$
2. Pre každé i množina V_i je takmer úplná guľa s dokonalou hranicou.
3. Pre každé i počet vrcholov množiny V_{i+1} je rovný

$$\min(|V_i| + \lceil(1 - \alpha) |\delta(V_i)|\rceil, n)$$

Postupnosť končí prvým výskytom množiny V . Je potrebné si uvedomiť, že hoci platí $|V_i| \leq |V_{i+1}|$ nemusí nevyhnutne platiť $V_i \subseteq V_{i+1}$, táto postupnosť teda nemusí byť šírením správy. Pre každú guľu B_r však platí

$$B_r \subseteq V_i \Rightarrow B_r \subseteq V_{i+1}$$

Označme p_r počet množín V_i , ktoré sú takmer úplnou guľou o polomere r . Ďalej budeme používať označenie

- $V_{k(r)}$ je prvá takmer úplná guľa s polomerom r v referenčnej postupnosti
- $V_{l(r)}$ je posledná takmer úplná guľa s polomerom r v referenčnej postupnosti

Zjavne platí

- $p_r = l(r) - k(r) + 1$
- $|V_{l(r)}| - |V_{k(r)}| \leq |\delta(B_{r-1})|$

Dĺžka referenčnej postupnosti je daná súčtom hodnôt p_r . Ukážme, že tento súčet nie je príliš veľký

Veta 6.13. *Počet prvkov referenčnej postupnosti, ktoré sú takmer úplnými guľami s polomerom menším ako $\lceil d/2 \rceil$, je maximálne $O(d)$. Teda platí*

$$\sum_{i=0}^{\lceil d/2 \rceil - 1} p_i = O(d)$$

Dôkaz. Ukážeme, že počet množín každého polomeru i sa dá ohraničiť konštantou nezávislou od d (v dôkaze predpokladáme $0 \leq i < \lceil d/2 \rceil$). Konkrétne ukážme, že

$$p_i \leq \frac{1}{1-\alpha} + 2$$

Na základe tvrdenia 6.7 a definície referenčnej postupnosti dostávame pre každé m spĺňajúce $k(i) \leq m < l(i)$

$$\begin{aligned} |V_{m+1}| - |V_m| &\geq \lceil (1-\alpha) |\delta(V_m)| \rceil \\ &\geq \lceil (1-\alpha) |\delta(B_{i-1})| \rceil \end{aligned}$$

Ak v každom kroku pribudne aspoň $\lceil (1-\alpha) |\delta(B_{i-1})| \rceil$ vrcholov a celkovo pribudne najviac $|\delta(B_{i-1})|$ vrcholov, dostávame ohraňenie na počet krokov

$$\begin{aligned} p_i &\leq \left\lceil \frac{|\delta(B_{i-1})|}{\lceil (1-\alpha) |\delta(B_{i-1})| \rceil} \right\rceil + 1 \\ &\leq \frac{1}{1-\alpha} + 2 \end{aligned}$$

Potom platí

$$\sum_{i=0}^{\lceil d/2 \rceil - 1} p_i \leq \left\lceil \frac{d}{2} \right\rceil \left(\frac{1}{1-\alpha} + 2 \right) = O(d)$$

□

Veta 6.14. *Počet prvkov referenčnej postupnosti, ktoré sú takmer úplnými guľami s polomerom väčším ako $\lfloor d/2 \rfloor$, je maximálne $O(d)$. Teda platí*

$$\sum_{i=\lfloor d/2 \rfloor + 1}^d p_i = O(d)$$

Dôkaz. Označme $u_k(S)$ počet vrcholov z množiny $V \setminus S$ so vzdialenosťou k od iniciátora. Potom pre každé dve takmer úplné gule V_j, V_{j+1} , obe rovnakého polomeru r platí súčasne

1. $u_r(V_j) - u_r(V_{j+1}) \geq \lceil (1 - \alpha)u_r(V_j) \rceil$, teda $\alpha u_r(V_j) \geq u_r(V_{j+1})$
2. $u_r(V_j) - u_r(V_{j+1}) \geq \lceil (1 - \alpha)|\delta(B_r)| \rceil$

Bod 1 plynie z α -proporcionálnosti modelu a vlastností takmer úplnej gule, bod 2 z α -proporcionálnosti modelu a predchádzajúceho tvrdenia 6.8. Na základe bodu 1 môžeme sformulovať (k, l skrácujeme $k(r), l(r)$)

$$u_r(V_{l-1}) = u_r(V_{k+p_r-2}) \leq \alpha^{p_r-2} u_r(V_k)$$

Potom platí nasledovné

$$\begin{aligned} \lceil (1 - \alpha)|\delta(B_r)| \rceil &\leq u_r(V_{l-1}) - u_r(V_l) \\ &\leq \alpha^{p_r-2} u_r(V_k) \end{aligned}$$

Vyčíslime $|\delta(B_r)| = \binom{d}{r+1}$ a odhadnime $u_r(V_k) \leq \binom{d}{r}$

$$\left(\frac{1}{\alpha}\right)^{p_r-2} \leq \frac{r+1}{(1-\alpha)(d-r)}$$

Zlogaritmujeme obidve strany pri základe $\frac{1}{\alpha} > 1$

$$\begin{aligned} p_r &\leq \log_{\frac{1}{\alpha}}(r+1) - \log_{\frac{1}{\alpha}}(1-\alpha) - \log_{\frac{1}{\alpha}}(d-r) + 2 \\ &\leq \log_{\frac{1}{\alpha}}(r+1) - \log_{\frac{1}{\alpha}}(d-r) + O(1) \end{aligned}$$

Sčítaním pre všetky prípustné hodnoty

$$\begin{aligned} \sum_{i=\lfloor d/2 \rfloor + 1}^d p_i &\leq \sum_{i=\lfloor d/2 \rfloor + 1}^d \log_{\frac{1}{\alpha}}(i+1) - \sum_{i=\lfloor d/2 \rfloor + 1}^d \log_{\frac{1}{\alpha}}(d-i) + O(d) \\ &\leq \sum_{i=1}^d \log_{\frac{1}{\alpha}}(i) - 2 \sum_{i=0}^{\lfloor d/2 \rfloor + 1} \log_{\frac{1}{\alpha}}(i) + O(d) \end{aligned}$$

Pre prehľadnosť upravme osobitne prvú sumu

$$\begin{aligned} \sum_{i=1}^d \log_{\frac{1}{\alpha}} i &\leq \log_{\frac{1}{\alpha}} (d!) \\ &\leq \log_{\frac{1}{\alpha}} \left((d/e)^d O(d) \right) \\ &\leq d \log_{\frac{1}{\alpha}} d + O(d) \end{aligned}$$

A následne druhú sumu

$$\begin{aligned} \sum_{i=0}^{\lfloor d/2 \rfloor + 1} \log_{\frac{1}{\alpha}} i &\geq \log_{\frac{1}{\alpha}} (\lfloor d/2 \rfloor + 1)! \\ &\geq \log_{\frac{1}{\alpha}} \left[\left(\frac{\lfloor d/2 \rfloor + 1}{e} \right)^{\lfloor d/2 \rfloor + 1} O(d) \right] \\ &\geq \frac{d}{2} \log_{\frac{1}{\alpha}} d + O(d) \end{aligned}$$

Dosadením do pôvodnej nerovnice dostávame

$$\begin{aligned} \sum_{i=\lfloor d/2 \rfloor + 1}^d p_i &\leq d \log_{\frac{1}{\alpha}} d - d \log_{\frac{1}{\alpha}} d + O(d) \\ &= O(d) \end{aligned}$$

□

Potrebuje ešte posledné tvrdenie, aby sme skompletizovali sumu od 1 po d . V tomto prípade môžeme robiť odhady trochu benevolentnejšie.

Veta 6.15. *Nech d je párne. Počet prvkov referenčnej postupnosti, ktoré sú takmer úplnými guľami s polomerom $d/2$, je maximálne $O(d)$. Teda platí*

$$p_{d/2} = O(d)$$

Dôkaz. V tvrdení 6.13 sme odvodili vzťah

$$u_r(V_{l-1}) = u_r(V_{k+p_r-2}) \leq \alpha^{p_r-2} u_r(V_k)$$

toto odvodenie nevyužívalo predpoklady o polomere takmer úplných guľí. Použime teraz hrubé odhady

- $u_r(V_{l-1}) \geq 1$
- $u_r(V_k) \leq \binom{d}{r}$

Dostávame

$$1 \leq \alpha^{p_r-2} \binom{d}{r}$$

Dosaďme za r aktuálnu hodnotu $d/2$ a zlogaritmujme obe strany

$$p_r \leq \log_{\frac{1}{\alpha}} \binom{d}{d/2} + 2 = O(d)$$

□

Veta 6.16 (dĺžka referenčnej postupnosti). *Referenčná postupnosť pre d -rozmernú hyperkocku Q_d má $O(d)$ prvkov.*

Dôkaz. Veta vyplýva priamo z predchádzajúcich tvrdení 6.13, 6.14 a 6.15.

□

Algoritmus W. Majme d -rozmernú hyperkocku Q_d . Algoritmus W množine informovaných vrcholov V_i priradí hrany spomedzi ∂V_i tak, že do každého vrcholu z $\delta(V_i)$ smeruje práve jedna hrana. Pre analýzu algoritmu nie je podstatné, ktorá z (potenciálne) viacerých hrán to je. Pre dôkaz trvania algoritmu W budeme potrebovať ešte pomocné tvrdenia

Veta 6.17. *Pre každú množinu $S \subseteq V$, pre každú takmer úplnú guľu B s dokonalou hranicou takú, že $|B| \leq |S|$ a pre každé $0 \leq \gamma \leq 1$ platí*

$$|B| + \lceil \gamma |\delta(B)| \rceil \leq |S| + \lceil \gamma |\delta(S)| \rceil$$

Dôkaz. Zoberme ľubovoľnú $S' \subseteq S$ takú, že $|S'| = |B|$. Na základe definície takmer úplnej gule s dokonalou hranicou platí

$$\lceil \gamma |\delta(B)| \rceil \leq \lceil \gamma |\delta(S')| \rceil$$

Na základe výberu množiny S'

$$|B| + \lceil \gamma |\delta(B)| \rceil \leq |S'| + \lceil \gamma |\delta(S')| \rceil$$

Pre dôkaz tvrdenia si postačuje uvedomiť, že

$$|S'| + \lceil \gamma |\delta(S')| \rceil \leq |S| + \lceil \gamma |\delta(S)| \rceil$$

pretože pridaním ľubovoľného prvku do množiny S' zvýšime prvý sčítanec o jedna a druhý nanaajvýš o jedna znížime. Opakujeme pridávanie pre všetky prvky z $S \setminus S'$. \square

Veta 6.18 (porovnanie s ref. postupnosťou). *Nech*

$$\{w\} = W_0 \subseteq W_1 \subseteq \dots \subset W_k = V$$

je ľubovoľné šírenie správy na grafe Q_d dané algoritmom W a α -proporcionalným modelom. Potom pre každé i platí

$$|V_i| \leq |W_i|$$

kde V_i je prvok z referenčnej postupnosti.

Dôkaz. Dôkaz vykonáme matematickou indukciou na i . Pre $i = 0$ porovnáваме počiatkové množiny obsahujúce výlučne vrchol w .

Dokážme tvrdenie pre hodnotu $i + 1$. Z indukčného predpokladu platí

$$|V_i| \leq |W_i|$$

Na základe definície referenčnej postupnosti platí

$$|V_{i+1}| \leq |V_i| + \lceil (1 - \alpha) |\delta(V_i)| \rceil$$

Z tvrdenia 6.17 (V_i je takmer úplná guľa s dokonalou hranicou) dostávame pre $\gamma = 1 - \alpha$ a indukčného predpokladu

$$|V_i| + \lceil (1 - \alpha) |\delta(V_i)| \rceil \leq |W_i| + \lceil (1 - \alpha) |\delta(W_i)| \rceil$$

Záverom z α -proporcionality

$$|W_i| + \lceil (1 - \alpha) |\delta(W_i)| \rceil \leq |W_{i+1}|$$

V tomto okamihu sme využili požadovanú vlastnosť algoritmu W - do kaž-

dého vrchola vedie najviac jedna aktívna hrana. Poskladaním jednotlivých nerovností (tranzitivita usporiadania \leq)

$$|V_{i+1}| \leq |W_{i+1}|$$

□

6.3 Analýza greedy algoritmu

Nasleduje tvrdenie, na ktorého dôkaze budeme prezentovať viacero (rozlične silných) techník

Veta 6.19 (trvanie greedy algoritmu). *Označme*

$$Q = (Q_1, Q_2, Q_3, \dots)$$

triedu hyperkociek, kde Q_d má rozmer d a $n = 2^d$ vrcholov. Potom platí

- *Dolný odhad $T_Q^{AG} = \Omega(\log n)$*
- *Horný odhad $T_Q^{AG} = O(\log n \log \log n)$*

Dolný odhad triviálne vyplýva zo základných vlastností modelu, dôkaz horného odhadu ponúkžeme v nasledujúcej časti.

6.3.1 Diskrétna analýza greedy algoritmu.

Predvedieme jednoduchú analýzu greedy algoritmu. Analýzu nazývame diskrétnou (v konfrontácii s nasledujúcou časťou), pretože proces šírenia správy rozdeľujeme na tri časti a odhadujeme na každej časti samostatne. Cieľom našej snahy je prezentovať tvrdenie

$$T_Q^{AG} = O(\log n \log \log n)$$

Pri dôkaze budeme vychádzať z nerovnosti pre veľkosť hranovej hranice. Základný problém, ktorý vzniká - ako prevedieme počet novoinformovaných hrán na počet novoinformovaných vrcholov? Hrubý odhad môžeme vykonať

vd'aka tomu, že stupeň vrchola v hyperkocke je $\log n$. Pre $g(m)$ hrán v hranovej hranici množiny informovaných vrcholov bude počet novoinformovaných vrcholov v nasledujúcom kroku aspoň

$$\left\lceil \frac{[\alpha'g(m)]}{\log n} \right\rceil$$

Rozdelíme dôkaz na tri časti

- informovaných je menej ako $\frac{n}{4}$ vrcholov, odhad na počet hrán v hranovej hranici s pribúdajúcim počtom vrcholov rastie
- informovaných je $\frac{n}{4}$ až $\frac{3n}{4}$ vrcholov, odhad na počet hrán v hranovej hranici najprv rastie a potom klesá, nahradíme ho minimom na tejto množine
- informovaných je viac ako $\frac{3n}{4}$ vrcholov, odhad na počet hrán v hranovej hranici s pribúdajúcim počtom vrcholov klesá

Informovanie prvých vrcholov. V počiatočných krokoch šírenia správy je počet informovaných vrcholov malý - potrebujeme ukázať, že tento počet dostatočne rýchlo rastie. Rozdelíme preto túto časť šírenia správy na etapy. i -ta etapa obsahuje tie kroky šírenia správy, kedy počet informovaných vrcholov je v intervale

$$\left[\frac{n}{2^i}, \frac{n}{2^{i-1}} \right)$$

Keďže v celej časti predpokladáme, že počet informovaných vrcholov je menší ako $\frac{n}{4}$, funkcia $g'(m)$ je rastúca. Pre i -tu etapu je mohutnosť hranovej hranice v každom kroku aspoň

$$g' \left(\frac{n}{2^i} \right) = i \frac{n}{2^i}$$

Počet hrán, po ktorých prešla správa na konci i -tej etapy nie je viac ako

$$\frac{n}{2^{i-1}} \log n$$

t.j. maximálny počet vrcholov vynásobený maximálnym stupňom. Počet krokov i -tej etapy je najviac

$$\frac{n \log n}{2^{i-1}} / \frac{\alpha' i n}{2^i} = \frac{2 \alpha' \log n}{i}$$

Keďže etáp v rámci prvej časti je menej ako $\log n$, dostávame celkové trvanie prvej časti menej ako

$$\sum_{i=0}^{\log n} \frac{2 \alpha' \log n}{i} = O(\log n \log \log n)$$

Informovanie posledných vrcholov. Princíp dôkazu pre tretiu časť (na počiatku informovaných aspoň $\frac{3n}{4}$ vrcholov) je obdobný ako pri dôkaze prvej časti. i -ta etapa obsahuje kroky, kde počet neinformovaných vrcholov je v intervale

$$\left[\frac{n}{2^i}, \frac{n}{2^{i+1}} \right)$$

Mohutnosť hranovej hranice je aspoň

$$g' \left(\frac{n}{2^{i+1}} \right) = \frac{(i+1)n}{2^{i+1}}$$

a celkové trvanie je $O(\log n \log \log n)$ krokov.

Prostredná časť. V každom kroku prostrednej časti sa na hranici nachádza aspoň n hrán, celkový počet hrán v grafe je $n \log n$. Trvanie prostrednej časti je preto najviac $O(\log n)$ krokov.

Sčítaním jednotlivých častí dostávame horný odhad na počet krokov šírenia greedy algoritmu na hyperkocke v podobe

$$O(\log n \log \log n)$$

6.3.2 Spojitá analýza

Pri diskretnej analýze sme sa dopustili dvoch zásadných zanedbaní. V prvom rade sme uvažovali, že na informovanie jedného vrchola je potrebných $d =_{def} \log n$ správ. Takýto predpoklad je samozrejme korektným horným odhadom, nemusí byť však tesný. V samotnej analýze sme nahrádzali veľkosť

hranovej hranice dolným odhadom na zvolenom intervale. V prípade spojitaj analýzy sa pokúsime vysporiadať s druhým zanedbaním. Vopred by sme radi upozornili, že v tomto prípade spojitá analýza nepomôže zbaviť sa dodatočného $\log \log n$ faktora.

Myšlienka spojitaj analýzy je jednoduchá - vyjadríme závislosť veľkosti hranovej hranice od počtu informovaných hrán. V prípade diskkrétnej analýzy sme rozdelili šírenie správy na intervaly a odhadli trvanie šírenia na každom intervale. V prípade spojitaj analýzy abstrahujeme od skutočnosti, že šírenie správy prebieha v krokoch a odhadneme čas za pomoci prostriedkov matematickej analýzy. Súčasťou dôkazu musí byť opodstatnenosť takéhoto prístupu. Podáme odhad maximálnej chyby, ktorej sme sa mohli dopustiť.

Analógia. Pre lepšie pochopenie problému vytvoríme nasledovnú analógiu. Predstavme si automobil pohybujúci sa po dráhe, okolo ktorej máme v pravidelných (nekonečne malých) intervaloch nainštalované zariadenia merajúce jeho rýchlosť. Pri pohybe automobilu získame graf rýchlosti v závislosti od dráhy $y = v_s(x)$. Čas pohybu môžeme z uvedeného vzťahu vyjadriť ako

$$\int_0^s \frac{1}{v_s(x)} dx$$

kde s je celková dráha, ktorú automobil prešiel. Pre pohyb automobilu a šírenie správy na grafe máme nasledovné analógie

Pohyb auta	Šírenie správy
vzdialenosť	počet hrán v indukovanom podgrafe
rýchlosť	počet bezchybných hrán
čas jazdy	počet krokov šírenia správy

V tomto okamihu je potrebné uvedomiť si dva rozdiely

1. počet hrán v podgrafe indukovanom informovanými vrcholmi sa nezvyšuje iba o bezchybné hrany (môže sa zvýšiť viac)
2. celkový počet hrán, počet aktívnych hrán, počet krokov a pod. sú diskkrétne veličiny

s obidvoma ťažkosťami sa postupne vysporiadame.

Závislosť rýchlosti od dráhy. Majme na pamäti analógiu s automobilom a pokúsme sa vypočítať počet krokov šírenia správy. Prvým predpokladom bolo vyjadrenie závislosti rýchlosti od dráhy, v našom prípade vyjadrenie závislosti veľkosti hranovej hranice (krát α') od počtu informovaných hrán. Nech po i krokoch prešla správa po m hranách. Platí nerovnosť

$$m \leq v \log v$$

kde m je počet hrán v podgrafe indukovanom nejakou množinou vrcholov mohutnosti v . S použitím Lambertovej W funkcie, pre ktorú platí

$$W(x)E^{W(x)} = x$$

dostávame odhad na počet vrcholov v závislosti od počtu informovaných hrán

$$\begin{aligned} m = W(m)e^{W(m)} &\leq v \log v \\ e^{W(m)} &\leq kv \end{aligned}$$

kde k je konštanta nezávislá od m a v , daná rôznym základom logaritmov.

Pre závislosť počtu hrán v hranovej hranici od počtu vrcholov platí izoperimetrická nerovnosť

$$g(v) \geq v(\log n - \log v)$$

v ďalšom sa zamerajme iba na tú časť postupnosti, kde počet informovaných vrcholov neprekročil $\frac{n}{4}$. Na tejto časti je uvedený odhad pre hranovú hranicu rastúci, teda môžeme do izoperimetrickej nerovnosti dosadiť dolný odhad na počet vrcholov

$$v(\log n - \log v) \geq e^{W(m)}(\log n - \log e^{W(m)}) O(1)$$

čím dostávame dolný odhad veľkosti hranovej hranice v závislosti od počtu informovaných hrán, ktorý je na zvolenom intervale rastúcou funkciou. Označme

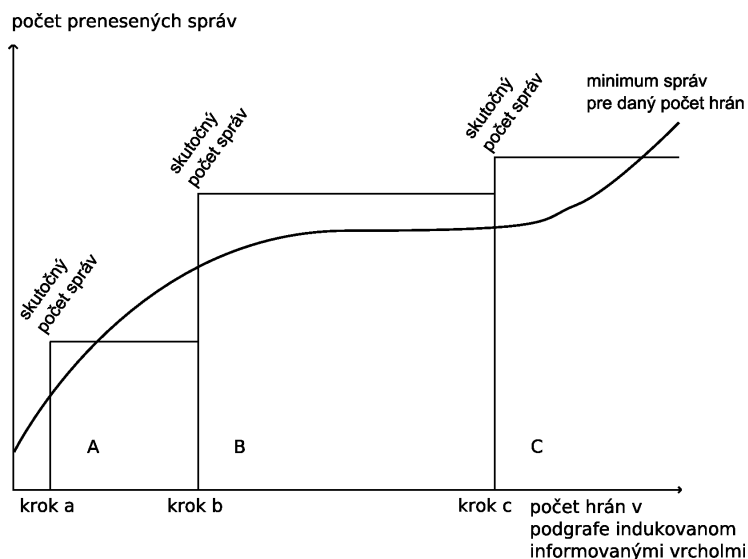
$$h(m) = e^{W(m)}(\log n - W(m)) O(1)$$

dolný odhad na počet hrán, ktorými prešla správa - hľadanú závislosť medzi

rýchlosťou a vzdialenosťou. Koštanty k a α' z proporcionality modelu sú zakomponované v multiplikatívnom faktore $O(1)$.

Uvedený vzťah znamená nasledovné - ak sa v niektorom kroku nachádza medzi informovanými vrcholmi aspoň m hrán, potom v nasledujúcom kroku prejde správa aspoň po $h(m)$ hranách.

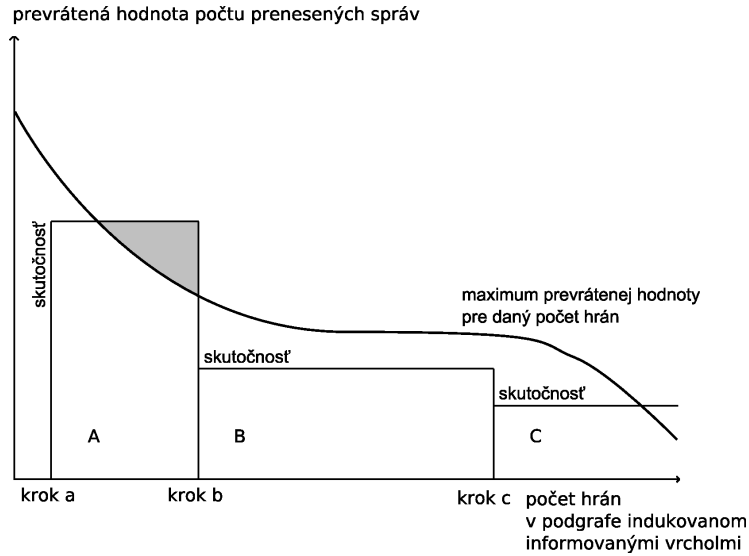
Vyjadrenie času a vysporiadanie sa s chybami. Vráťme sa k nastoleným problémom - počet hrán v podgrafe indukovanom informovanými vrcholmi sa nezvyšuje iba o bezchybné hrany; navyše šírenie správy v skutočnosti nie je spojité, ale prebieha v krokoch. Skutočný počet prenesených správ konfrontovaný s grafom rýchlosti môže vyzeráť napríklad ako na obrázku 13



Obr. 13: Spojitá analýza greedy algoritmu na hyperkočke I. Príklad šírenia správy. Uvedená závislosť je iba ilustratívna.

Body a, b, c reprezentujú tri po sebe idúce kroky šírenia správy. V každom kroku šírenia je počet hrán, ktoré pribudli do podgrafu indukovaného informovanými vrcholmi väčší alebo rovný krivke. Ďalší krok (na osi X) je posunutý aspoň o toľko, koľko hrán v skutočnosti pribudlo - každá hrana, po ktorej bola prenesená správa, vedie medzi dvoma informovanými vrcholmi; môže však medzi nimi viesť aj hrana, po ktorej nebola prenesená správa. Znáznornené obdĺžniky A, B, C majú teda šírku aspoň takú, ako výšku. Vytvoríme prevrátený graf s prevrátenými hodnotami na osi Y - výška obdĺžnika

bude prevrátenou hodnotou počtu prenesených správ.



Obr. 14: Spojitá analýza greedy algoritmu na hyperkočke II. Príklad šírenia správy, prevrátené hodnoty. Uvedená závislosť je iba ilustratívna.

Počet obdĺžnikov zodpovedá počtu krokov šírenia správy, obsah každého obdĺžnika je aspoň jedna. Dobrým odhadom na spočítanie počtu obdĺžnikov je vyjadrenie plochy pod krivkou, problémy vznikajú iba s potenciálnymi presahmi (zobrazené šedou). Celkový počet hrán v grafe je $n \log n$ a k je ľubovoľná konštanta (uvedená z technických dôvodov, znamená „informovanie prvých k hrán“). Požadovanú závislosť prevrátejnej hodnoty počtu prenesených správ od počtu hrán v indukovanom podgrafe informovanými vrcholmi nahradíme odhadom $1/h(m)$.

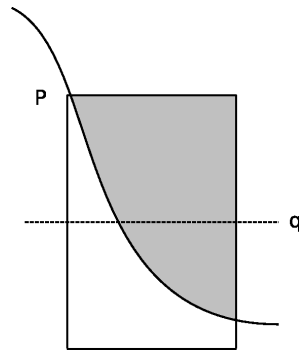
$$\begin{aligned}
 T &\leq \int_k^{(n/4) \log n} \frac{1}{h(m)} dm + k \\
 &\leq \left[-\frac{(\log n + 1) \log(W(m) - \log n) + W(m)}{\alpha'} \right]_k^{(n/4) \log n} O(1) \\
 &= O(\log n \log \log n)
 \end{aligned}$$

Získaný čas je horným odhadom času potrebného na informovanie štvrtiny vrcholov pomocou greedy algoritmu.

Ostáva ukázať, že sme sa nedopustili príliš veľkej chyby - teda že plocha presahov (zobrazené šedou) nie je príliš veľká oproti celkovej ploche pod krivkou. Rozdeľme obdĺžniky do dvoch skupín

- obdĺžniky, ktorých presah je menej ako $1/2$ plochy
- obdĺžniky, ktorých presah je viac ako $1/2$ plochy

Obdĺžniky v prvej skupine majú pod krivkou plochu aspoň $\frac{1}{2}$ kroku šírenia správy, postačuje teda náš odhad vynásobiť dvomi. Ukážeme, že obdĺžnikov v druhej skupine nemôže byť veľa. Ilustratívny príklad obdĺžnika z druhej skupiny je uvedený na obrázku 15. Keďže ľavý horný bod (bod P na obrázku) každého obdĺžnika sa nachádza pod- alebo na úrovni krivky a krivka je klesajúca, krivka musí vpravo vychádzať v dolnej polovici výšky obdĺžnika (pod osou q). Všetky nasledujúce obdĺžniky sa nevyhnutne nachádzajú pod osou q a majú najviac polovičnú výšku. Keďže triviálny odhad na počet hrán v hranovej hranici je $n \log n$, minimálna výška obdĺžnika je $1/(n \log n)$. Minimálny počet aktívnych hrán je 1, čo je zároveň maximálna výška obdĺžnika. Počet obdĺžnikov v druhej skupine je teda maximálne $\log(n \log n) = \log n + \log \log n$.



Obr. 15: Spojitá analýza greedy algoritmu na hyperkocke III. Príklad obdĺžnika s nadpolovičným presahom. Funkcia $1/h(m)$ je iba ilustratívna

7 Ďalšie oblasti štúdia

V nasledujúcej časti stručne priblížime ďalšie oblasti, ktoré by mohli byť zaujímavé v súvislosti s proporcionálnym počtom chybných liniek.

7.1 Výber iniciátora

V predloženej práci sme implicitne predpokladali, že spoločne s grafom máme vždy zvolený niektorý vrchol za iniciátora. Zaujímá nás, ako veľmi závisia jednotlivé dosiahnuté výsledky od konkrétnej voľby iniciátora

Veta 7.1. *Nech G je ľubovoľný súvislý graf a $u, v \in V$ sú vrcholy. Potom čas šírenia správy z vrchola u a čas šírenia správy z vrchola v (v oboch prípadoch optimálnym algoritmom)*

- sa líši maximálne o priemer grafu G

$$T_{G,u} \leq T_{G,v} + \text{diam}(G)$$

- sa líši maximálne o multiplikatívnu konštantu nezávislú od grafu

$$T_{G,u} \leq k T_{G,v}$$

Dôkaz. Keďže zo všeobecných vlastností vyplýva $T_G \geq \text{diam}(G)$, postačuje ukázať prvú časť tvrdenia.

Označme $A_{G,u}$ optimálny algoritmus šírenia správy v grafe G z vrchola u . Skonstruujme (nie nevyhnutne optimálny) algoritmus $A_{G,v}$ pre šírenie správy v grafe G z vrchola v . Algoritmus bude pracovať nasledovne

1. V prvej fáze zvolí ľubovoľnú najkratšiu cestu z vrchola v do vrchola u . Pokiaľ je vrchol u neinformovaný, volí za aktívne hrany nachádzajúce sa na tejto ceste. V každom kroku je aktívna práve jedna hrana, preto musí byť bezchybná. Trvanie prvej fázy je maximálne $\text{diam}(G)$ krokov.
2. V rámci druhej fázy simuluje optimálny algoritmus $A_{G,u}$. Označme I silne informované vrcholy - tie, ktoré boli informované v rámci druhej fázy (môžu tu patriť niektoré - nie nutne všetky - spomedzi vrcholov informovaných v rámci prvej fázy). Ostatné informované vrcholy

označíme za slabo informované. Na začiatku druhej fázy je silne informovaný iba vrchol u a slabo informované sú všetky ostatné vrcholy informované v rámci prvej fázy.

- (a) Algoritmus v každom kroku simuluje $A_{G,u}$ na silne informovaných vrchoch, získa množinu aktívnych hrán $A_{G,u}(I)$. Vyberie z nich tie, ktoré vedú do neinformovaných vrcholov.
- (b) Súper mu zablokuje maximálne proporcionálny počet hrán. Tieto hrany blokuje v simulácii optimálnemu algoritmu $A_{G,u}$. Keďže skutočne aktívnych hrán bolo najviac toľko, ako požadoval algoritmus $A_{G,u}$, je zachovaná proporcionalita modelu. Množina silne informovaných vrcholov sa rozšíri o novoinformované vrcholy a tie slabo informované vrcholy, do ktorých viedla aktívna hrana v simulácii $A_{G,u}$.

V druhej fáze algoritmus $A_{G,v}$ robí korektného súpera (v zmysle α -proporcionálnosti modelu) k algoritmu $A_{G,u}$. Z predpokladu optimality je preto počet krokov maximálne $T_{G,u}$

□

Zaujímavá otázka je, či spomínané tvrdenie platí aj pre prípad greedy algoritmu. Odpoveď na túto otázku nám zatiaľ nie je známa.

7.2 Alternatívne modely

Pripustíme drobné zmeny modelu a všímajme si, ako to ovplyvní dosiahnuté výsledky.

Násobné hrany. Ak pripustíme v modeli násobné hrany, definovaný čas šírenia správy greedy algoritmom ako o funkcia závislá od počtu vrcholov stráca svoj intuitívny základ.

Definícia 7.2 (podobné grafy). Nech G_1 a G_2 sú grafy s násobnými hranami. Hovoríme, že grafy sú podobné, ak $V_1 = V_2$ a vrcholy u a v sú susedné v grafe G_1 práve vtedy, keď sú susedné v grafe G_2 (odhliadnuc od počtu hrán, ktoré ich v jednotlivých grafoch spájajú).

Veta 7.3 (šírenie správy na podobných grafoch). *Ku každému súvislému grafu G_1 s jednoduchými hranami o n vrcholoch existuje podobný graf G_2 s násobnými hranami taký, že čas šírenia správy greedy algoritmom na tomto grafe je $n - 1$ krokov.*

Dôkaz. Pre jednoduchosť predpokladajme $\alpha = 1/2$. Zoberme ľubovoľné usporiadanie hrán v grafe G_1 . Každú hranu z grafu G_1 , ktorá je v našom usporiadaní na pozícii⁷ i , nahradíme 2^i hranami v grafe G_2 . Označme $e(u, v)$ počet aktívnych hrán medzi vrcholmi u, v v danom kroku. Súper v každom kroku preposiela správy po všetkých hranách vedúcich z vrchola a do vrchola b , kde a, b je dvojica s maximálnou hodnotou $e(a, b)$. Ostatné hrany súper blokuje. Je zjavné, že počet blokovaných hrán je menší ako počet bezchybných hrán. Pre iné α stačí príslušne upraviť základ 2. \square

Orientované hrany Ak pripustíme v modeli orientované hrany, taktiež prichádzame o mnoho pekných vlastností.

Veta 7.4 (medzera medzi greedy a optimálnym algoritmom). *Existuje trieda grafov G taká, že šírenie správy*

- *pomocou greedy algoritmu trvá $\Theta(n)$ krokov*
- *pomocou optimálneho algoritmu trvá $\Theta(\log n)$ krokov*

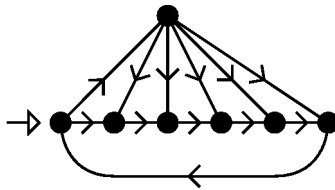
Dôkaz. Zoberme triedu grafov podľa obrázku 16. Ďalšie vrcholy sú vždy pridávané na pravú stranu línie.

Súper pre greedy algoritmus blokuje hranu vedúcu k hornému vrcholu, pokiaľ je niektorý vrchol zo spodnej línie neinformovaný. Takýmto prístupom dosiahne čas šírenia správy rovný $n - 1$.

Alternatívny algoritmus v prvom kroku zvolí za aktívnu iba hranu vedúcu do horného vrchola, v nasledujúcom potom môže paralelne informovať veľké množstvo vrcholov, čím dosiahne čas šírenia správy $\Theta(\log n)$. Presnejší výpočet je možné realizovať obdobne ako pri dôkaze vety 4.5. \square

Veta 7.5. *Pre ľubovoľné (dosť veľké) n existuje graf G s n vrcholmi a vrcholy $v_1, v_2 \in V$ také, že*

⁷pozície číslujeme od 1



Obr. 16: Model s orientovanými hranami.

Príklad grafu so 7 vrcholmi. Iniciátor je zobrazený so šípkou.

- šírenie správy na grafe G s iniciátorom v_1 a greedy algoritmom trvá aspoň $n - 1$ krokov
- šírenie správy na grafe G s iniciátorom v_2 a greedy algoritmom trvá najviac $k \log n$ krokov pre k nezávislé od G

Dôkaz. Použijeme n -vrcholový graf z triedy definovanej v dôkaze vety 7.4. Iniciátor v_1 je vrchol zobrazený so šípkou a trvanie sme ukázali v dôkaze vety 7.4. Iniciátor v_2 je horný vrchol (podľa obrázka 16). Rovnakou technikou, ako pri dôkaze tvrdenia 4.5 dostávame horný odhad na počet krokov

$$\log_{\alpha} \left(\frac{1}{n-2} \right) + 2$$

□

7.3 Informovanosť algoritmu

Definícia algoritmu šírenia správy ako zobrazenia z množiny informovaných vrcholov do množiny aktívnych hrán je pomerne „silná“. Je to dané snahou ukázať obmedzenia modelu - pokiaľ neexistuje dobre informovaný algoritmus, umožňujúci šírenia správy v prostredí s proporcionálnym počtom chybných liniek dosahujúce dostatočne dobrý čas, nie je možné očakávať dobré výsledky v prípade, kedy možnosti algoritmu budú viac obmedzené. Na druhej strane stojí za zváženie, či dobré výsledky (asymptotická optimálnosť, horné odhady pre šírenie optimálnym algoritmom) je možné dosiahnuť aj v prípade, kedy algoritmus musí byť do väčšej miery „distribovateľný“. Z tohto dôvodu sme ponúkli analýzy greedy algoritmu, ktorý je pomerne obmedzený. Mnohé s predložených výsledkov ostávajú s rovnakými argumentami v platnosti aj v prípade rôznych obmedzení kladených na algoritmus

(napr. algoritmus dopredného a spätného šírenia potrebuje mapu topológie, svoju pozíciu a globálne hodiny; nepotrebuje informáciu o tom, ktoré správy iným vrcholom súper zablokoval).

8 Záver

Zopakujme témy, ktorým sme sa v predkladanej práci venovali a dosiahnuté výsledky.

V rámci **štvrtej kapitoly** sme predstavili tri rôzne typy algoritmov šírenia správy.

Greedy algoritmus, ktorý je jednoducho zadaný a realizovateľný aj v obmedzujúcejších modeloch, ale postráda niektoré intuitívne vlastnosti. Šírenie správy pomocou greedy algoritmu bolo najviac študované - spomedzi známych výsledkov zopakujme

- $T_K^{AG} = \Theta(\log n)$ pre úplné grafy
- $T_T^{AG} = \Omega(\sqrt{n})$ pre úplné binárne stromy
- $T_M^{AG} = \Theta(k)$ pre n -rozmernú cyklickú mriežku a párne k

Greedy algoritmus môže byť neoptimálny, na zvolenej triede grafov sme prezentovali rozdiel

- $O(\log^2 n)$ optimálny algoritmus
- $\Omega(n/\log n)$ greedy algoritmus

V neskoršej časti sme navyše ukázali, že postačujú pomerne jednoduché triedy grafov, aby sme dosiahli veľký rozdiel v časoch šírenia správy.

Algoritmus dopredného šírenia simuluje šírenie správy na grafe bez chybných liniek, pracuje v etapách. Často poskytuje prvotný horný odhad na čas šírenia správy na zadanom grafe. Existujú acyklické grafy, na ktorých je asymptoticky neoptimálny. Čas oproti optimálnemu algoritmu (a rovnako aj oproti šíreniu správy v prostredí bez chybných liniek) je väčší maximálne o multiplikatívny $\log n$ faktor.

Algoritmus spätného šírenia definovaný pre acyklické grafy využíva na rozdelenie do skupín spätnú vzdialenosť vrchola do listu. Čas šírenia správy pomocou algoritmu spätného šírenia sa dá odhadnúť na základe rozdelenia vrcholov do skupín, podobne ako v prechádzajúcom prípade. Algoritmus je asymptoticky optimálny pre každú triedu acyklických grafov, aj

Typ grafu	Dolný odhad	Horný odhad
ξ - acyklické, rovnaké vetvenie	$\Omega\left(n^{\frac{k-1}{k}}\right)$	$O\left(n^{\frac{k-1}{k}} \log n\right)$
T - acyklické	-	$O\left(n^{\frac{k}{k+1}}\right)$

Tabuľka 1: Zhrnutie výsledkov pre grafy s konštantou excentricitou iniciátora k

keď rozdiel v trvaní šírenia správy medzi ním a optimálnym algoritmom môže byť ľubovoľne veľký (pre dostatočne veľký graf).

V **piatej kapitole** sme našu pozornosť zamerali na šírenie správy na grafoch s konštantnou excentricitou iniciátora. Ukázali sme, že greedy algoritmus na vybranom grafe môže byť pomerne neefektívny. Skonstruovali sme podtriedu grafov, na ktorých bol výrazný rozdiel medzi optimálnym algoritmom a greedy algoritmom.

Ďalej sme ukázali, že na inak skonstruovanej podtriede nemôžeme dostať výrazne kvalitatívne významnejší výsledok (za ktorý pokladáme hodnotu $\Theta(n)$).

Zhrnutie výsledkov je uvedené v tabuľke 1.

V rámci **šiestej kapitoly** sme sa zaoberali šírením správ na hyperkocke. V prvej časti sme analyzovali optimálny algoritmus - ukázali sme, že hyperkocky sú odolnou triedou grafov voči proporcionálnemu počtu chybných liniek. Analýza šírenia správy si vyžiadala využitie mnohých vlastností hyperkociek, najmä znalostí o veľkosti vrcholovej hranice a vlastnostiach gúl.

V druhej časti kapitoly sme prezentovali analýzu greedy algoritmu na hyperkocke. Na dôkaze horného odhadu $O(\log n \log \log n)$ sme prezentovali diskretnú a spojitú analýzu šírenia správy. Hoci sa nám nepodarilo rozhodnúť optimálnosť algoritmu na hyperkocke, prezentovaná technika s využitím izoperimetrických nerovností môže mať využitie aj v analýze šírenia správ na iných grafoch.

V **siedmej kapitole** sme sa stručne venovali ďalším oblastiam štúdia. Ukázali sme, že v prípade optimálneho algoritmu má výber iniciátora iba malý vplyv na čas šírenia správy. V krátkosti sme načrtli, ako môže zmena modelu pôsobiť na dosiahnuté výsledky (násobné hrany, orientované hrany).

Na záver zhrňme odpovede na otázky prezentované v úvode práce

- **Ktoré triedy grafov sú odolné voči proporcionálnemu počtu chybných liniek?**

- Žiadna trieda grafov s excentriciou iniciátora asymptoticky menšou ako $\log n$ nemôže byť odolná (úplné grafy, stromy s konštantnou excentricitou iniciátora).
- Cyklické mriežky s párnym k sú odolné (aj s použitím greedy algoritmu)
- Úplné binárne stromy nie sú odolné, pretože algoritmus spätného šírenia je asymptoticky optimálny s časom $\Theta(\log^2 n)$
- Hyperkocky sú odolné; či je greedy algoritmus asymptoticky optimálny, nie je zatiaľ známe.

- **Aké všeobecné (generické) triedy algoritmov môžeme zostrojiť? Aké majú vlastnosti?**

Definovali sme tri triedy algoritmov s rôznou motiváciou - zjednodušenie algoritmu šírenia správy, relaxácia modelu alebo snaha o paralelné posielanie veľkého množstva správ.

- **Aký veľký môže byť rozdiel medzi časom šírenia pomocou všeobecných algoritmov?**

Pre každú triedu algoritmov sme prezentovali grafy, na ktorých sú neoptimálne. V prípade algoritmu spätného šírenia sme dokázali jeho optimálnosť na ľubovoľnej triede acyklických grafov. Zvlášť v prípade greedy algoritmu sme sa snažili maximalizovať rozdiel v čase šírenia. Ukázali sme, že postačujú pomerne jednoduché grafy (acyklické s konštantnou excentricitou iniciátora) na to, aby vznikol značný rozdiel v čase šírenia greedy algoritmom a optimálnym algoritmom.

- **Na ktorých triedach grafov je greedy algoritmus optimálny?**

Greedy algoritmus je optimálny napríklad na úplných grafoch a cyklických mriežkach s párnym k . Naproti tomu, neoptimálny je na stromoch (a viacerých pomerne umelo skonštruovaných príkladoch). V prípade hyperkociek nám výsledok nie je známy.

Literatúra

- [1] Rastislav Kráľovič, Richard Kráľovič, and Peter Ružička. Broadcasting with many faulty links. In Jop F. Sibeyn, editor, *SIROCCO*, volume 17 of *Proceedings in Informatics*, pages 211–222. Carleton Scientific, 2003.
- [2] Stefan Dobrev, Rastislav Kráľovič, Richard Kráľovič, and Nicola Santoro. On fractional dynamic faults with threshold. In Paola Flocchini and Leszek Gasieniec, editors, *SIROCCO*, volume 4056 of *Lecture Notes in Computer Science*, pages 197–211. Springer, 2006.
- [3] Pierre Fraigniaud and Emmanuel Lazard. Methods and problems of communication in usual networks. *Discrete Appl. Math.*, 53(1-3):79–133, 1994.
- [4] Richard Kráľovič. *Broadcasting with Dynamic Faults*. PhD thesis, Comenius University, 2008.
- [5] A. Bagchi and S. L. Hakimi. Information dissemination in distributed systems with faulty units. *IEEE Trans. Comput.*, 43(6):698–710, 1994.
- [6] Krzysztof Diks, Evangelos Kranakis, and Andrzej Pelc. Broadcasting in unlabeled tori. *Parallel Processing Letters*, 8(2):177–188, 1998.
- [7] André Schiper. Group communication: From practice to theory. In Jirí Wiedermann, Gerard Tel, Jaroslav Pokorný, Mária Bieliková, and Julius Stuller, editors, *SOFSEM*, volume 3831 of *Lecture Notes in Computer Science*, pages 117–136. Springer, 2006.
- [8] Pierre Fraigniaud. Asymptotically optimal broadcasting and gossiping in faulty hypercube multicomputers. *IEEE Trans. Comput.*, 41(11):1410–1419, 1992.
- [9] Shimon Even and Burkhard Monien. On the number of rounds necessary to disseminate information. In *ACM Symposium on Parallel Algorithms and Architectures*, pages 318–327, 1989.
- [10] Leszek Gasieniec and Andrzej Pelc. Broadcasting with a bounded fraction of faulty nodes. *Journal of Parallel and Distributed Computing*, 42(1):11–20, 1997.

- [11] Leszek Gasieniec and Andrzej Pelc. Adaptive broadcasting with faulty nodes. *Parallel Computing*, 22(6):903–912, 1996.
- [12] Krzysztof Diks and Andrzej Pelc. Almost safe gossiping in bounded degree networks. *SIAM J. Discret. Math.*, 5(3):338–344, 1992.
- [13] V. E. Mendia and D. Sarkar. Optimal broadcasting on the star graph. *IEEE Trans. Parallel Distrib. Syst.*, 3(4):389–396, 1992.
- [14] Jehoshua Bruck. On optimal broadcasting in faulty hypercubes. *Discrete Appl. Math.*, 53(1-3):3–13, 1994.
- [15] Anders Dessmark and Andrzej Pelc. Optimal graph exploration without good maps. *Theor. Comput. Sci.*, 326(1-3):343–362, 2004.
- [16] Andrzej Pelc and David Peleg. Broadcasting with locally bounded byzantine faults. *Inf. Process. Lett.*, 93(3):109–115, 2005.
- [17] Paola Flocchini and Bernard Mans. Optimal elections in labeled hypercubes. *Journal of Parallel and Distributed Computing*, 33(1):76–83, 1996.
- [18] Nicola Santoro and Peter Widmayer. Distributed function evaluation in the presence transmission faults. In *SIGAL '90: Proceedings of the international symposium on Algorithms*, pages 358–367, New York, NY, USA, 1990. Springer-Verlag New York, Inc.
- [19] Chlebus, Diks, and Pelc. Broadcasting in synchronous networks with dynamic faults. *NETWORKS: Networks: An International Journal*, 27, 1996.
- [20] Gianluca De Marco and Ugo Vaccaro. Broadcasting in hypercubes and star graphs with dynamic faults. *Inf. Process. Lett.*, 66(6):321–326, 1998.
- [21] Gianluca De Marco and Adele A. Rescigno. Tighter time bounds on broadcasting in torus networks in presence of dynamic faults. *Parallel Processing Letters*, 10(1):39–49, 2000.
- [22] Stefan Dobrev and Imrich Vrto. Optimal broadcasting in hypercubes with dynamic faults. *Inf. Process. Lett.*, 71(2):81–85, 1999.

- [23] Stefan Dobrev and Imrich Vrto. Optimal broadcasting in tori with dynamic faults. *Parallel Processing Letters*, 12(1):17–22, 2002.
- [24] Hedetniemi S. M. (1) ; Hedetniemi S. T. ; Liestman A. L. ;. A survey of gossiping and broadcasting in communication networks. *Networks*, 18(4):319–349, 1988.
- [25] J. Hromkovič, R. Klasing, A. Pelc, P. Ružicka, and W. Unger. *Dissemination of Information in Communication Networks: Broadcasting, Gossiping, Leader Election, and Fault-Tolerance (Texts in Theoretical Computer Science. An EATCS Series)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.
- [26] Steven Butler. The isoperimetric problem on the hypercube. Lecture notes of course Random Walks on Directed and Undirected Graphs at University of Carolina, San Diego, November 2005. http://www.math.ucsd.edu/~phorn/math261/11_2_notes.pdf.
- [27] Ross M. Richardson. Edge isoperimetric inequalities. Lecture notes of course Random Walks on Directed and Undirected Graphs at University of Carolina, San Diego, November 2005. http://www.math.ucsd.edu/~phorn/math261/11_7_notes.pdf.