

Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

DIPLOMOVÁ PRÁCE



Štefan Šurina
Možnosti a metody geolokalizace serverů
Katedra softwarového inženýrství
Vedoucí práce: *Doc. Ing. Václav Jirovský, CSc.*
Studijní program: *Informatika*

Pod'akovanie

Ďakujem doc. Ing. Václavovi Jirovskému, CSc., vedúcemu svojej práce za cenné rady a pripomienky, ktoré dopomohli k realizácii tejto práce.

Prehlasujem, že som svoju diplomovú prácu napísal samostatne a výhradne s použitím citovaných prameňov.

V Prahe dňa

Štefan Šurina

OBSAH

OBSAH	2
1. ÚVOD	5
2. GEOGRAFICKÝ SYSTÉM	7
3. PREHĽAD DOSTUPNÝCH METÓD GEOLOKALIZÁCIE A PREDPOKLADY ICH PRAKTICKÉHO POUŽITIA	9
3.1 TECHNIKY ZALOŽENÉ NA MERANÍ ODOZVY (DELAY MEASUREMENT TECHNIQUES)	9
3.1.1 Najkratší ping (Shortest Ping)	10
3.1.2 SOI (Speed of Internet) Geolocation	11
3.1.3 CBG (Constraint Based Geolocation)	15
3.2 TBG (TOPOLOGY BASED GEOLOCATION)	19
3.3 TECHNIKY POUŽÍVAJÚCE ŠPECIÁLNE SÚBORY	22
3.3.1 DNS GPOS súbor	22
3.3.2 DNS LOC súbor	23
3.4 TECHNIKY POUŽÍVAJÚCE ŠPECIÁLNY HARDWARE	24
3.4.1 GPS	24
3.4.2 Place Lab	25
3.4.3 Cricket	25
3.5 TECHNIKY POUŽÍVAJÚCE MANUÁLNE UDRŽIAVANÉ DATABÁZY	26
3.5.1 Whois databázy	26
3.5.3 Komerčné databázy (GeoIP, IP2Geo, MaxMind, apod.)	27
3.6 TECHNIKY POUŽÍVAJÚCE ČIASTKOVÉ INFORMÁCIE	29
3.6.1 Obecne - mená DNS	29
3.6.2 Medzinárodné kódové označenia letísk	30
3.6.3 Konvencie pomenovaní pri dial-up pripojeniach vo Veľkej Británii a USA	30
3.7 OCTANT – RÁMKOVÁ TECHNIKA	31
4. IMPLEMENTÁCIA VYBRANÝCH TECHNÍK ZALOŽENÝCH NA MERANÍ ODOZVY A JEJ PROBLEMATICKÉ BODY	33
4.1 VŠEOBECNÁ CHARAKTERISTIKA POUŽITEJ IMPLEMENTÁCIE	33
4.2 MERANIE ODOZVY	35
4.3 PREPOČET ODOZVY (RTT) NA VZDIALENOSŤ	37
4.4 VÝPOČET PRIESEČNÍKOV PRI POUŽITÍ GEOGRAFICKÝCH UKAZATEĽOV	38
4.5 RIEŠENIE ÚLOHY LINEÁRNEHO PROGRAMOVANIA PRE BESTLINE	40
4.6 PROBLEMATIKA MISLOKALIZÁCIE A POKUS O JEJ VYRIEŠENIE	41
5. POROVNANIE TECHNÍK ZALOŽENÝCH NA MERANÍ ODOZVY POMOCOU ROZLOŽENIA PRAVDEPODOBNOSTI ICH ODCHÝLOK NA PODKLADE IMPLEMENTÁCIE A ZHODNOTENIE ICH EFEKTIVITY	43
5.1 ZÁKLADNÉ POROVNANIE VYBRANÝCH TECHNÍK	44
5.2 STRUČNÉ ZHODNOTENIE ČASOVEJ NÁROČNOSTI	48
5.3 VYLEPŠENIE ALGORITMU SHORTEST PATH A JEHO VPLYV NA EFEKTIVITU	49
6. VZNIK ODCHÝLKY ODOZVY A JEJ VZŤAH K CELKOVEJ CHYBE MERANIA	53
6.1 VZNIK ODCHÝLKY ODOZVY. VPLYV LINEARIZÁCIE SIGNÁLU	53
6.2 VZŤAH ODCHÝLKY ODOZVY K CELKOVEJ CHYBE MERANIA	60
7. ZÁVEREČNÉ ZHODNOTENIE A POJEDNANIE O MOŽNOSTI A PRAVDEPODOBNOSTI PRESNEJ LOKALIZÁCIE	74

8. LITERATÚRA	76
9. ZOZNAM POUŽITÝCH SKRATIEK	78
10. ZOZNAM PRÍLOH (V ELEKTRONICKEJ FORME).....	79

Abstract

This diploma thesis outlines the methods of server geolocalization, that is, determining the physical location of servers based on network measurements. Especially, it sums up several groups of methods, all more or less usable. What is more, it comprises the implementation of couple of them, mainly those based on round trip measurements from distributed environment of well known data points. According to this, statistical comparison of their ability to geolocate precisely is made. Finally, this thesis also tries to prove theoretically how to pick up geolocation servers in order to minimize the error of potential measurements.

Anotácia

Táto diplomová práca popisuje metódy geolokalizácie serverov, čo znamená stanovenie fyzickej lokality serveru na základe meraní v rámci siete. Zhrnuje hlavne niekoľko skupín metód, viac či menej použiteľných. Jej súčasťou je aj implementácia niekoľkých z nich, prevažne založených na meraní odozvy zo známych bodov v rámci distribuovaného prostredia. Na tomto základe je vykonaná štatistická analýza ich schopnosti presne určiť geolokalitu. A na záver sa tiež snaží teoreticky odvodiť spôsob ako vybrať geolokačné servery, aby bola chyba potenciálneho merania minimálna.

1. Úvod

Systém vzájomne prepojených sietí, ktorý dnes vytvára Internet, vznikol nekoordinovane, teda zdola nahor, a v rámci tohto vývoja nikdy nebola riešená otázka akéhosi prevodného mechanizmu medzi svetom virtuálnym v rámci tejto siete a reálnym svetom. Obzvlášť to platí pre lokalizáciu. Nepotrebnosť mechanizmu sa prejavuje i dnes, a dokonca pre ochranu akejkoľvek „slobody“ je nutná. Na druhej strane, potreba pre určité záujmové skupiny vychádza do popredia napr. s rozmachom reklamy – spoločnosti chcú cieľovať svoje produkty v bezmedznom Internete len na určité územie alebo okruh území. Ďalším impulzom, ktorý môže za vzostup záujmu o určité možnosti tohto prevodu je čím ďalej tým väčšie porušovanie právnych noriem a fakt, že to, čo je trestné v jednej krajine, inde byť nemusí. Preto je nutné, aby bolo možné zistiť s veľmi veľkou pravdepodobnosťou, na území ktorej krajiny sa tento subjekt, ktorý pôsobí protiprávne činy, nachádza (hľadisko tzv. teritoriality).

Cieľ tejto diplomovej práce nie je objasniť túto skutočne rozsiahlu a o to nemenej zaujímavú problematiku v celej svojej šírke, ale nahliadnuť do tejto problematiky, získať vedomosti o existujúcich technikách určovania geografickej lokality a zhodnotiť ich možnosti, príp. upozorniť na niektoré úskalía.

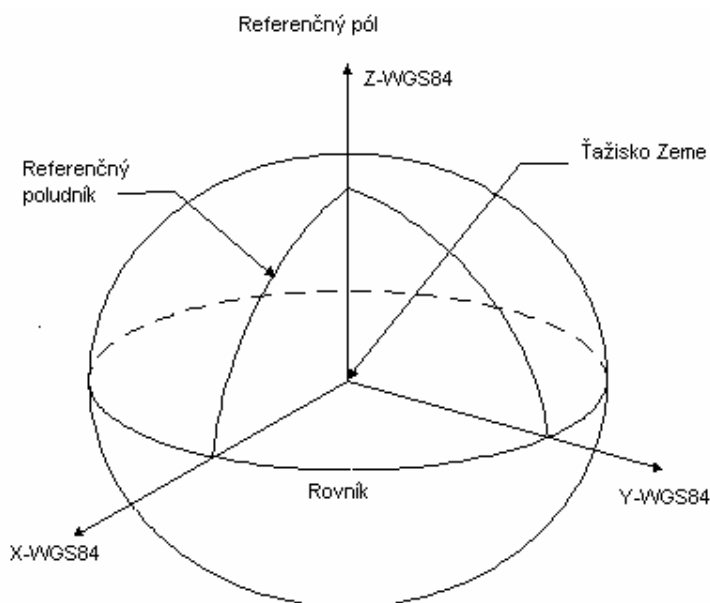
Prácu som rozdelil do viacerých funkčných celkov, ktoré sa zaoberajú rôznymi faktormi geografickej lokalizácie. Na počiatku celkového pojednania som zaradil popis geografického modelu, z ktorého pri zisťovaní geografickej lokality vychádzam a bez ktorého by táto práca nemohla obstáť. Ďalšia kapitola obsahuje popis všetkých metód, ktoré sa mi podarilo zozbierať, ktorým dominujú metódy založené na meraní odozvy zo stanovených serverov. Súčasťou tejto diplomovej práce je i program implementujúci práve vybrané z týchto metód. Kapitola č. 4 popisuje práve úskalía, ktoré bolo a je treba riešiť v prípade ich zachytenia (mi. obsahuje i pojednanie o tom, ako sa mi podarilo znížiť pravdepodobnosť nežiaduceho javu, ktorý nastával pri použití nízkeho počtu serverov, ktoré som mal k dispozícii – k mislokalizácii). Kapitola č. 5 taktiež vychádza z vykonanej implementácie a snaží sa zachytiť určité vývojové tendencie metód v porovnaní s ostatnými a činí tak prostredníctvom normálnych rozdelení pravdepodobností veľkostí ich chýb. Kapitola č. 6 je teoretickou časťou, kde sa pokúšam stanoviť príčiny odchýlok a matematicky ich popísať, a zároveň sa snažím matematicky odvodiť, ako vplýva rozostavenie serverov a ich vzájomné vzdialenosti na celkovú chybu merania. V tejto kapitole sa takisto pokúšam dať návod k tomu, ako je možné postupovať pri výbere serverov pri dôveryhodnej oblasti, aby táto chyba merania čo možno najmenšia. Na záver je zaradená kapitola o pojednaní o celkovej možnosti geolokalizácie a presnosti prípadných meraní.

Rozsah miery tvorivej činnosti je z princípu v jednotlivých kapitolách rozdielna. Popis geografického modelu a jednotlivých metód vychádza prevažne z článkov a publikácií, ktoré sú uvedené v zozname použitej literatúry. Kapitoly 4 a 5 sú vytvorené na podklade implementácie – tam zahŕňam tvorivé riešenie problémov implementácie, ako i porovnávanie metód na jej podklade. Konečne kapitola 6 predstavuje vlastné odvozenie nadpísaného vzťahu a so závermi z toho plynúcimi.

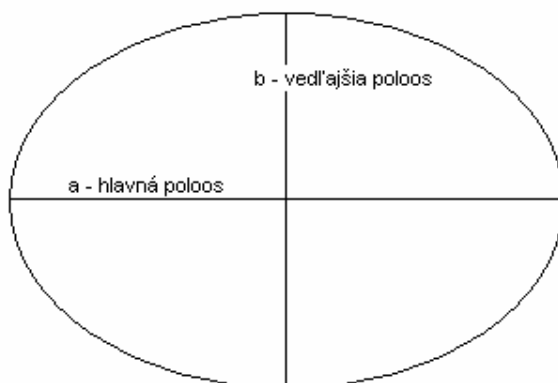
2. Geografický systém

Vzdialenosti, ktoré uvažujeme medzi jednotlivými servermi a cieľmi, sú na povrchu Zeme. Preto je nutné využívať pre uspokojivé výsledky zároveň dostatočne presný model Zeme alebo časti jeho povrchu.

Keďže v tejto práci spracúvam údaje a vzdialenosti presahujúce lokálne parametre, vhodnejšou voľbou je aplikácia modelu celého Zemského povrchu. Použitý referenčný model Zeme je označovaný ako WGS-84, ktorý je založený na rotačnom elipsoide. Ako štandardný ho využíva i satelitný navigačný systém GPS.



obr. 2.1



obr. 2.2

WGS-84 model využíva zemepisné súradnice, kde sú jeho body definované zemepisnou šírkou a dĺžkou. Začiatok sústavy súradníc a smery súradnicových systémov sú definované nasledujúcim spôsobom (viz. tiež obr. 2.1):

- a) začiatkom je Zemské ťažisko,
- b) os Z je definovaná ako smer ku konvenčnému terestrickému pólu (CTP) definovaného BIH (Bureau International de l'Heure) na základe súradníc staníc definovaných BIH,
- c) os X je priesečníkom referenčného poludníku WGS-84 a roviny rovníku vzťahujúceho sa k CTP,
- d) os Y dopĺňa systém na pravotočivý pravouhlý súradnicový systém, kde smer hlavnej osi je 90° východne vzhľadom k osi X.

Tento model sa považuje za jednu z najpresnejších aproximácií. Útvar je, ako už vyššie spomenuté, rotačný elipsoid s nasledovnými konštantami:

- hlavná poloos $a = 6378,137 \text{ km}$

- sploštenie $f = 1/298,257223563 (\approx 3,35\text{‰})$

Za pomoci týchto konštánt sme schopní vypočítať ďalší, pre implementáciu, potrebný údaj:

- vedľajšia poloos $b = (1-f)a = 6356,7523142 \text{ km}$

3. Prehľad dostupných metód geolokalizácie a predpoklady ich praktického použitia

V nasledujúcej kapitole sa pokúšam urobiť prehľad o skúmanej problematike z hľadiska možnosti použitia rôznych metód na zistenie informácií týkajúcich sa geolokalizácie. Metódy som zozbieral a rozradil ich do skupín, z hľadiska ich podobných znakov – hlavne spôsobu prístupu informáciám ohľadne pozície cieľového bodu.

Na úvod každej skupiny metód sa pokúšam načrtnúť predpoklady, za ktorých je vôbec možné (uspokojivo) ich použiť. Tieto predpoklady sú orientačné, pretože sa môžu meniť i v závislosti na použitom prostredí. (Viz. napr. kapitolu 5, v ktorej sú vidieť tieto diferencie pri empirickom skúmaní techník založených na meraní odozvy na území Severnej Ameriky a Európy).

Pri zhrnutí orientačných predpokladov uvádzam taktiež svoje úvahy, akú rolu by mohla hrať tá či oná skupina metód pri návrhu určitého „univerzálneho nástroja“ na geolokalizáciu, ktorý by mohol potenciálne zahŕňať všetky spomínané skupiny metód (a to napr. tú najefektívnejšiu variantu z každej z nich).

3.1 *Techniky založené na meraní odozvy (delay measurement techniques)*

Predpoklady použitia metód:

Pre užitie tejto skupiny metód je nutné mať k dispozícii prístup k aspoň 3 relatívne rovnomerne rozložených serverom, z ktorých je možné posielat' požiadavky na zisťovanie odozvy. Pre relatívne uspokojujúce výsledky (pod 100 km), ako uvádzajú viaceré zdroje¹, je vhodný počet 40-50 serverov na kontinenty severnej Ameriky či Európy. Tento nedostatok je možné pri niektorých metódach, ako sa snažím ukázať v kapitole 5.3 (v obmedzenej miere) odstrániť použitím pasívnych landmarkov.

Možnosť zahrnutia do univerzálneho nástroja:

Táto skupina je vďaka využitiu fyzikálnych vlastností prenosového média vhodná k prvotnému odhadu oblasti, v ktorej sa hľadaný bod nachádza. Vzhľadom ku kombinácii fyzikálnych vlastností a štatistických informácií je možné docieľiť nielen záruku tejto oblasti, ale aj relatívnej presnosti jej finálneho stanovenia.

¹ Napr. B. Gueye, M. Crovella, A. Ziviani, S. Fdida: Constraint-based Geolocation of Internet Hosts

3.1.1 Najkratší ping (Shortest Ping)

Táto metóda je pravdepodobne najjednoduchšia zo všetkých techník založených na meraní odozvy. Jej princíp spočíva v tom, že po zmeraní odozvy zo všetkých serverov, z ktorých posielame požiadavky, vezmeme najkratší RTT (round-trip time) a geografické koordináty tohto serveru (z neho bola odozva najrýchlejšia) prehlásime za naše hľadané miesto.

Je zrejmé, že táto metóda a jej presnosť bude veľmi závisieť od toho, nakoľko bude náš priestor, v ktorom hľadáme cieľovú pozíciu, husto posiaty servermi, z ktorých posielame zmienené požiadavky. Táto otázka súvisí s problémom naznačeným v kapitole 4.2 pod bodom iii). Malý počet týchto serverov nám robí túto metódu nepoužiteľnou. Z idey, ako z tohto algoritmu urobiť metódu, ktorá by bola presnejšia v situácii, keď nemôžeme zvyšovať počet serverov, je logická. Predpokladajme, že máme N serverov, potom obor hodnôt funkcie, ktorá priradí každému cieľovému miestu práve jednu z lokalít serverov, je N . Myšlienka vychádza z toho, že ak rozšírime jednoducho danú funkciu, a definujeme ďalšiu skupinu bodov o K prvkoch, môžeme docieľiť zväčšenie oboru hodnôt na $N+K$. Pri malom počte N , môže byť v tomto prípade pridanie určitej skupiny o K bodoch znamenať výrazné zefektívnenie.

Pre potreby možného zefektívnenia dodefinujem K tzv. landmarkov, čo znamená K určitých cieľových bodov, o ktorých s určitou presnosťou vieme, kde sa nachádzajú. Pozmeníme algoritmus takým spôsobom, že z našich serverov zisťujeme veľkosť odozvy nielen od cieľa, ktorého lokalitu chceme zistiť, ale i od landmarkov a všetkých ostatných serverov (ktoré sa v tomto prípade chovajú iba ako landmarky). Analogicky ako v obecnom prípade, sa snažíme získať na základe dostupných odoziev čo najpresnejšiu lokalitu cieľa. Teraz máme z každého serveru odozvy ku všetkým známym destináciám i k cieľu. Za lokalitu cieľa prehlásime lokalitu tej zo známych destinácií, ktorá bude mať zo všetkých najpodobnejšie odozvy, a teda súčet rozdielu odoziev bude najmenší. V kapitole 5.3 je viditeľné výrazné zefektívnenie tohto algoritmu pred obecným, vyššie popísaným algoritmom bez landmarkov.

3.1.2 SOI (Speed of Internet) Geolocation

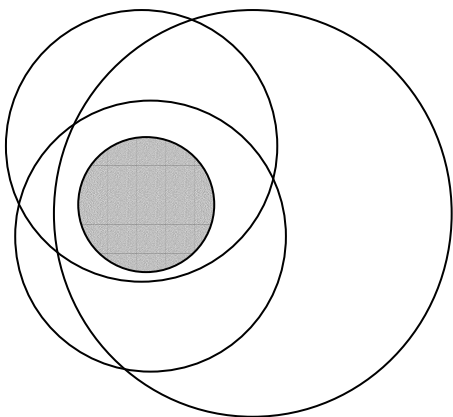
Metóda SOI sa na základe jej princípu tiež radí medzi metódy využívajúce odozvy z rôznych serverov a na ich podklade odhadne geografickú lokalitu cieľového počítača identifikovaného podľa IP adresy.

Prvým krokom je využitie serverov, ktoré máme k dispozícii, pričom vieme, kde sa nachádzajú a môžeme ich využívať na posielanie paketov a merať čas, za ktorý sa vrátia späť k danému serveru. Tieto odozvy získame zo všetkých serverov v oblasti, ktorú skúmame (t.j. v našom prípade v oblasti severnej Ameriky či Európy). V danom momente je nutné zaistiť prevod nejakým spôsobom zo získaných odoziev na vzdialenosti na zemskom povrchu. Pri tejto metóde sa obecné vychádza z premisy, že informácie prúdia cez optické káble maximálne rýchlosťou takmer $2/3$ rýchlosti svetla. Tento odhad je samozrejme horným odhadom, keď vezmeme do úvahy realitu – do prejdenej vzdialenosti za určitý časový úsek uvedený v odozve musíme zahrnúť taktiež napríklad možné spomalenie paketov kvôli nelinearite cesty, apod. Toto spôsobuje, že odhad naznačený vyššie by bolo možné vylepšiť. Z výsledkov empirických meraní uvedených v [1], je badateľné možné zlepšenie tohto odhadu až na $4/9c$.

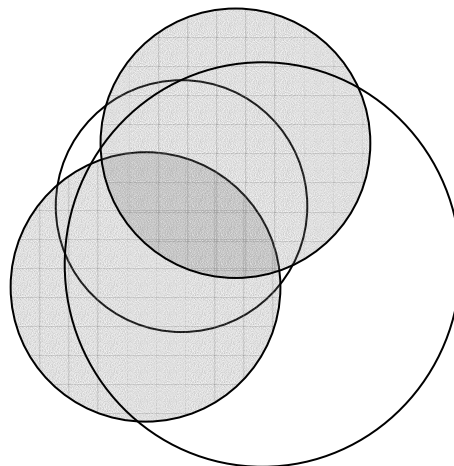
Následne, ako bolo už naznačené, prepočítame odozvy zo serverov na vzdialenosti, ktoré teda podľa vyššie uvedeného horné odhady pre tieto vzdialenosti. Tieto nám spolu s pozíciami príslušných serverov vytvárajú určitý počet „kruhov“ na povrchu Zeme, každá zo stredom na danej pozícii serveru, s polomerom viažucim sa k tomuto serveru, reprezentujúci zmienený horný odhad prepočítanej vzdialenosti k cieľovému bodu.

Vieme, že dané vzdialenosti sú horným odhadom, a tak všetky tieto „kruhy“ budú mať spoločný prienik. Tento nám vlastne tvorí oblasť, v ktorej by sa náš hľadaný cieľ mal s veľkou pravdepodobnosťou nachádzať. V ďalších úvahách rozvedieme spôsoby nájdenia najpravdepodobnejšieho bodu ako cieľa s ohľadom na tvar vzniknutej oblasti. Tento tvar je v zásade trojaký:

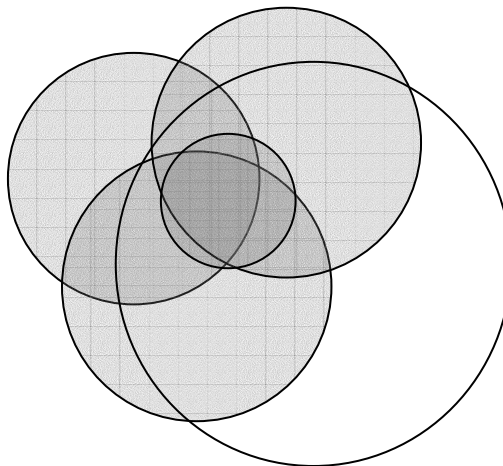
1. Prienik tvorí jediný „kruh“, pričom tento „kruh“ je obsiahnutý vo všetkých ostatných „kruhoch“
2. Prienik tvorí priesečník 2 „kruhov“, pričom je opäť nutne obsiahnutý vo všetkých ostatných „kruhoch“
3. Prienik tvorí priesečník viac než 2 „kruhov“



(obr. 3.1)



(obr. 3.2)



(obr. 3.3)

(Ad 1)

V tomto prípade označujeme za najviac pravdepodobný bod stred kružnice (viz obr. 3.1). (V prípade, že je v skúmanej oblasti dostatočné množstvo serverov, ktoré využívame ako brány, je veľká pravdepodobnosť, že jedna z brán bude v rovnakom meste ako cieľový stroj, a tak je logické prehlásiť za hľadaný bod, práve pozíciu tejto brány.)

(Ad 2)

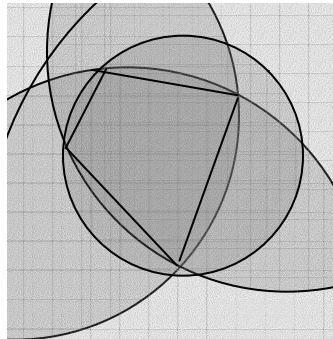
V tomto prípade sa ako najviac vhodný bod volí ako stred úsečky spájajúcej oba priesečníky (viz obr. 3.2).

(Ad 3)

Oblasť, ktorá vznikne, keď sa nám pretnú viac než dva „kruhy“, je útvarom ohraničeným z častí kružníc vymedzujúcich túto oblasť (viz obr. 3.3). Útvar bude konvexný. Táto vlastnosť vyplýva z toho, že vznikol ako prienik „kruhov“, a tieto sú všetky konvexné. Prienik konvexných útvarov je opäť konvexný. Vezmeme si akúkoľvek úsečku, ktorá spája dva okrajové body oblasti. Pre všetky kruhy platí, že tieto body budú ich okrajovými alebo sa budú nachádzať v ich vnútri. Akákoľvek úsečka kruhu sa bude nachádzať celá v jeho vnútri. Keďže toto platí pre všetky kruhy, bude sa nachádzať aj v ich prieniku, a teda v našej oblasti. Oblasť je teda konvexná.

Pri hľadaní bodu, ktorý bude naším odhadom pre hľadanú pozíciu cieľa, si trochu priblížime danú oblasť a aproximujeme ju mnohouholníkom, z dôvodu, aby sme mohli odhadnúť jej veľkosť a zároveň presnejšie určiť najpravdepodobnejšiu lokalitu cieľa ako ťažisko tohto mnohouholníka. Mnohouholník definujeme ako najväčší konvexný mnohouholník taký, že jeho vrcholy sú práve body tvoriace priesečníky kružníc a zároveň okrajové body našej oblasti. (viz obr. 3.4).

(obr. 3.4)



Tento polygón sa využije na približné zmeranie oblasti, v ktorej sa s veľkou pravdepodobnosťou nachádza hľadaný cieľ. Vďaka konvexnosti vzniknutej oblasti, je polygón celý vnútri. Definujeme si daný mnohouholník pomocou úsečiek vedúcich medzi jednotlivými vrcholmi. Pokiaľ máme n priesečníkov, kde každý označíme ako $v_i = (x_i, y_i)$, kde $0 \leq i \leq n-1$. Posledný vrchol je rovnaký ako prvý, a teda rovnako platí $v_0 = v_n = (x_n, y_n)$. Obsah tohto polygónu definovaného poradím vrcholov $v_0 = (x_0, y_0), \dots, v_{n-1} = (x_{n-1}, y_{n-1})$ vypočítame podľa vzorca:

$$A = \frac{1}{2} \sum_{i=0}^{n-1} \begin{vmatrix} x_i & x_{i+1} \\ y_i & y_{i+1} \end{vmatrix},$$

kde $|M|$ symbolizuje determinant matice M .

Ťažisko c tohto mnohouholníka spočítame podľa nasledujúcich rovností:

$$c_x = \frac{1}{6A} \sum_{i=0}^{n-1} (x_i + x_{i+1}) \begin{vmatrix} x_i & x_{i+1} \\ y_i & y_{i+1} \end{vmatrix}$$
$$c_y = \frac{1}{6A} \sum_{i=0}^{n-1} (y_i + y_{i+1}) \begin{vmatrix} x_i & x_{i+1} \\ y_i & y_{i+1} \end{vmatrix}$$

Toto ťažisko mnohouholníka (c_x, c_y) je teda naším odhadom na lokalitu cieľa.

3.1.3 CBG (Constraint Based Geolocation)

Táto metóda využíva odozvu zo serverov, teda je možné ju tiež zaradiť do nadpisanej skupiny metód. V podstate vychádza z myšlienky, ktorú využíva algoritmus SOI, a teda, že zmeria veľkosti odozvy z príslušných serverov v oblasti, „nejakým“ spôsobom prepočíta túto hodnotu na vzdialenosť, vytvorí okolo každého zo serverov oblasť, pre ktorú platí, že každý bod v tejto oblasti je nanajvýš tak ďaleko, ako je prepočítaná vzdialenosť, kde ako prienik vznikne dôveryhodná oblasť, ktorej ťažisko prehlásime za naše hľadané miesto.

Ďalej sa budeme zaoberať výhradne spôsobom prepočtu odozvy na vzdialenosť, pretože v iných aspektoch sa metóda od vyššie popísanej metódy SOI (časť 3.2) nelíši.

Zaujímavejší spôsob uvedeného prepočtu vychádza zo snahy zahrnúť oneskorenie spôsobené rôznymi dôvodmi do výsledného prepočtu. SOI vychádza z premisy, že vzdialenosť, ktorú prekoná paket od serveru až po cieľové zariadenie, je priamo úmerné času, ktorý paket na tejto ceste strávi (príp. RTT). Táto myšlienka, ktorú využíva SOI, je založená na ideálnych pomeroch, kde by server a cieľové zariadenie museli byť priamo napojené na jedno prenosové vlákno, a toto prenosové vlákno by muselo vytvárať priame prepojenie týchto dvoch strojov. Tento predpoklad je v reálnych podmienkach nerealizovateľný z viacerých dôvodov:

- i) Niektoré cesty od zdroja odozvy do jeho cieľa vykazujú ad hoc kruhový či inak nelineárny charakter.
- ii) Niektoré prenosové cesty sú zdieľané.
- iii) Niektoré časti siete vykazujú lokálne oneskorenie.

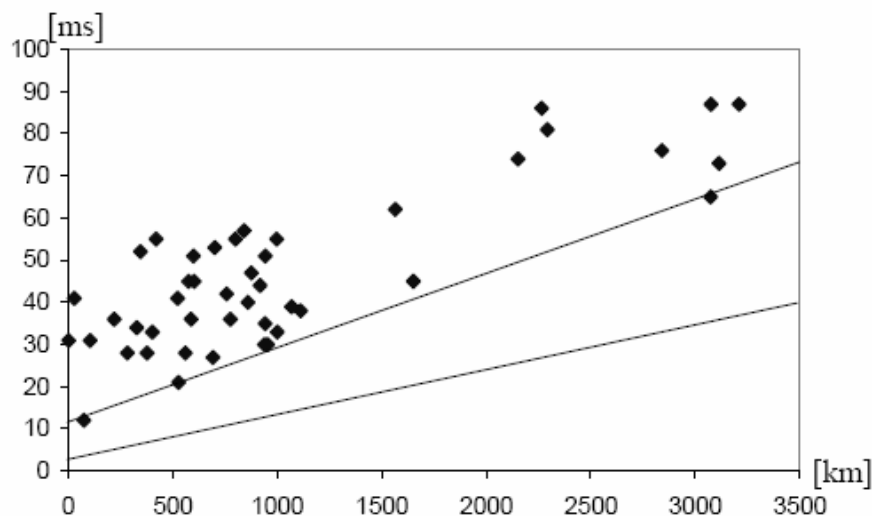
Ideou algoritmu CBG je odstrániť pevné prepočítavanie a stanoviť odlišný spôsob, ktorý bude odrážať aktuálnu situáciu v sieti. V porovnaní s algoritmom SOI, ktorý toto prepočítavanie má stanovené podľa fyzikálnych faktov a empirických zistení ako konštantné v čase i pre celú svetovú sieť bez rozdielu, CBG sa pokúša o diferenciáciu prepočtu s ohľadom na lokalizovanie určitých dôvodov oneskorenia, teda z hľadiska priestorového a zároveň i časového, pretože niektoré príčiny tohto oneskorenia sa môžu prejaviť v istých časových obdobiach, kedy je sieť viac zaťažená.

CBG sa pokúša o istú formu kalibrácie, ktorá sa prispôsobuje miestne i časovo podmienkam, a stanoví iné konštanty pre prepočítavanie.

Základný pohľad na prepočet tkvie v myšlienke, že každá odozva sa skladá z dvoch častí – jednej deterministickej (túto považujeme za pevnú časť) a stochastickej (túto považujeme za pohyblivú). Konkrétne, maximálna vzdialenosť, ktorú mohol paket prejsť, bude na základe istých fyzikálnych vlastností optického vlákna, v pohľade na istú odozvu pevne stanovená. Zároveň ale je jasné, že táto vzdialenosť bude často odhadom, ktorý ďaleko prevyšuje skutočnú vzdialenosť vzhľadom na oneskorenie, vytvorené hlavne ako je popísané v bodoch (i-iii) vyššie. Toto považujeme za deterministickú časť. Vylepšenie konštant prepočtu vzhľadom na konkrétne stochastické výsledky vzťahu (odozva, vzdialenosť) z určitého serveru na iné servery, konštituuje zmienenu stochastickú časť výpočtu. Konkrétny prepočet bude popísaný neskôr.

Metóda, ktorú využíva CBG, zaručuje, až na výnimočné prípady, že vzdialenosti, ktoré prepočtom dosiahne, budú odhadom, ktorý prevyšuje skutočnú vzdialenosť, a je zároveň menší než vzdialenosť, ktorú dosiahneme po použití prepočtu, ktorý využíva SOI a zároveň tvorí tzv. deterministickú časť výpočtu.

Graf g. 3.1 nám ukazuje závislosť medzi vzdialenosťami medzi jednotlivými servermi (resp. landmarkami) a príslušnými odozvami (Na ose x sú znázornené vzdialenosti v kilometroch a na ose y sú odozvy v milisekundách). Tieto údaje som získal pri zaznamenávaní jednotlivých testovacích výsledkov pri implementácii algoritmov. Na tomto grafe závislostí je tiež zakreslená priamka, ktorá reprezentuje základnú závislosť, ktorá nám tvorí deterministickú časť výpočtu. Algoritmus CBG sa pokúša priblížiť túto závislosť, aby lepšie reprezentovala reálne podmienky. Je tu snaha, aby skutočné vzťahy, ako sú vidieť na obrázku, boli dostatočne presne zachytené aj v danej funkčnej závislosti, ktorú použijeme na geolokáciu konkrétneho hľadaného cieľa. Pokiaľ chceme aproximovať lineárnou závislosťou, ako najpriamočiarejšia sa nám javí metóda najmenších štvorcov. Keď by sme sa ale pozreli na výsledok takejto implementácie, zistili by sme, že táto metóda nie je príliš vhodná, a to preto, že nám nezaručuje vôbec žiadnu korektnú informáciu o finálnej pozícii (to znamená, že pri prepočte podľa závislosti získanej na základe tejto metódy nevieme povedať s dostatočnou istotou, či je reálna vzdialenosť väčšia alebo menšia). Na obrázku je znázornené, ako toto priblíženie konštruuje CBG algoritmus. Vychádza sa z myšlienky čo najlepšej aproximácie pri zachovaní určitej informácie o pozícii (ako vidíme na nižšie na obrázku, CBG konštruuje najbližší možný horný lineárny odhad na základe získaných empirických údajov).



(g. 3.1)

Algoritmus CBG vychádza zo závislosti, ktorú sme vyššie odvodili pri popise algoritmu SOI a nazýva ju „baseline“. Pomocou nameraných hodnôt sa snaží prispôbiť konkrétnym podmienkam a tak, ako bolo vyššie spomenuté, posúva túto lineárnu závislosť čo najbližšie k týmto hodnotám. Spôsobom, ktorý popíšem nižšie, skonštruuje novú lineárnu závislosť, tzv. „bestline“, ktorú využije pri prepočte odoziev pri reálnej geolokácii z každého bodu.

Toto prispôbovanie a prepočet je nutné urobiť pre každý server, z ktorého posielame požiadavky diferencovane, a tým dostaneme pre každý z nich závislosť, ktorá bude rešpektovať lokálne podmienky a omeškania, ktoré môžu byť spôsobené špecifickými podmienkami v okolí tohto konkrétneho serveru. Ako je i možné spozorovať na obrázku, toto prispôbenie jednak posúva priamku „bestline“ vyššie a teda konštanta v jej predpise sa zväčší o kladnú hodnotu a jednak za zvýši jej sklon. Takáto konštrukcia má odrážať a súčasne do istej miery eliminovať dva hlavné dôvody omeškania:

- v okolí serveru je výraznejší zdroj omeškania (napr. môže byť napojený na sieť pomalšou linkou), a toto nám pri každej odozve vytvorí určité pevné omeškanie,
- sieť v určitom okamihu vykazuje nejakú hodnotu priechodnosti, ktorá závisí na momentálnych podmienkach a nedá sa dopredu určiť.

Dôvod označený ako a) je riešený pomocou posunu priamky o konštantu nahor, pričom do istej miery eliminovať dôvod označený ako b) je možný pomocou zvýšenia sklonu priamky oproti ideálnym podmienkam priamky „baseline“.

Po náznaku užitočnosti tejto úpravy závislosti pre každý server ešte ostáva odvodiť postup, ako dosiahnuť „bestline“ z „baseline“ za pomoci závislostí, ktoré máme namerané. Označme si odozvu zo i -tého serveru na j -ty ako d_{ij} a príslušnú geografickú vzdialenosť ako g_{ij} . Nech má priamka „baseline“ predpis $y = mx$. Hľadáme priamku $y = m_i x + b_i$, ktorá bude reprezentovať „bestline“. Vieme, že chceme vytvárať horný odhad vzdialenosti a vytvoriť dôveryhodný región, ako bol zobrazený v časti venovanej algoritmu SOI a to znamená, že priamka „bestline“ musí bezpodmienečne ležať pod všetkými bodmi, ktoré reprezentujú závislosti empiricky zistené. Teda musí platiť podmienka, že

$$\begin{aligned} \forall i \neq j : m_i x + b_i &\leq y \\ \forall i \neq j : y - m_i x - b_i &\geq 0 \end{aligned} \quad (\text{r. 3.1})$$

Ďalšia informácia v o priamke „bestline“ tkvie v tom, že to je priamka, pre ktorú platí, že ktorákoľvek iná priamka má v súčte vzdialeností od jednotlivých empiricky zistených bodov vyššiu hodnotu. Toto je možné formulovať ako funkciu

$$\min_{\substack{b_i \geq 0 \\ m_i \geq m}} \left(\sum_{i \neq j} y - m_i x - b_i \right) \quad (\text{r. 3.2})$$

V r. 3.2 máme funkciu, ktorú treba minimalizovať a v r. 3.1 máme obmedzujúcu nerovnicu. Takto môžeme formulovať úlohu lineárneho programovania s neznámymi premennými m_i a b_i . Po vyriešení tejto úlohy dostaneme hodnoty týchto premenných, ktoré použijeme ako parametre funkcie pri konštrukcii odhadu vzdialenosti $\tilde{g}_{i\lambda}$ serveru a hľadaného bodu.

$$\tilde{g}_{i\lambda} = \frac{d_{i\lambda} - b_i}{m_i} \quad (\text{r. 3.3})$$

3.2 TBG (Topology Based Geolocation)

Predpoklady použitia metódy:

Keďže táto metóda vychádza z predchádzajúcej skupiny, platí o nej čo sa týka predpokladov jej použitia mutatis mutandis to isté. Zisťovania určitých obmedzených topografických informácií je spojené so zložitejšími použitými nástrojmi a s nutnosťou širších dát (hlavne o pasívne landmarky z topograficky nekonvenčných oblastí).

Možnosť zahrnutia do univerzálneho nástroja:

Túto metódu možno využiť v rámci univerzálneho nástroja ako identifikátor možných problémových oblastí, kde by sa vďaka atypickej topografii siete mohli výsledky značne skresľovať.

Obecná charakteristika

Metóda nazývaná „Topology Based Geolocation“ tiež patrí do skupiny metód, ktoré využívajú meranie odozvy ako spôsob lokalizácie serverov, a snaží sa práve vnesením určitých informácií do preprocessingu eliminovať určité nepresnosti a nedokonalosti podobných systémov, ako napríklad CBG². Metóda v základoch z metódy CBG vychádza.

CBG vykazuje v určitých špecifických situáciách horšie a menej identifikovateľné výsledky. Špecificky, ak sa napríklad snažíme lokalizovať cieľ, ku ktorému vedie cesta paketu od dvoch alebo viacerých geolokačných serverov cez rovnaký router, potom toto bude znamenať, že výsledok bude oveľa nepresnejší, pretože sa nám tým pádom predĺžia cesty, a tzv. dôveryhodnú oblasť, ktorú by nám vytvoril napríklad hore uvedený algoritmus, bude oveľa väčšia, než tá, ktorú by sme mohli vytvoriť, ak by sme poznali, že uvedené cesty sa nestretajú až na koncovom serveri, ale už niekde po ceste. Algoritmus TBG sa snaží eliminovať tento problém tým, že diferencovanými spôsobmi sa snaží zistiť určité aspekty topografie siete. Nižšie je uvedený popis základných problémov, s ktorými sa TBG snaží vysporiadať, a akými nástrojmi na zistenie topografie tak činí.

I. Základný problém, s ktorým sa CBG potýka, je, ako už som spomenul pri analýze tohto algoritmu, že lepšie výsledky vykazuje v oblastiach, v ktorých má hustejšiu sieť serverov, z ktorých vysiela požiadavky na cieľový počítač. Pokiaľ nastane situácia, že chceme lokalizovať stroj, pre ktorý platí, že najbližší server je príliš ďaleko, alebo sa nachádza na okraji celkovej oblasti, kde sa pokúšame geolokovať a pokryť túto oblasť

² Pre hlbšie úvahy nad možnosťami, ako sa pokúsiť odstrániť nepresnosti pomocou topografických informácií viz. Basset, John and others: Towards IP Geolocation Using Delay and Topology Measurements

servermi, alebo sú tieto podmienky splnené naraz, algoritmus CBG môže vykazovať veľmi nepresné výsledky. Pri riešení tohto problému topograficky orientovaný TBG algoritmus vychádza zo zaujímavej myšlienky, že by bolo vhodné lokalizovať i všetky relevantné routery na ceste od všetkých serverov ku cieľovému počítaču. V tomto prípade by sme mohli docieľiť to, že pokiaľ sa vyskytne problém, ako bol naznačený vyššie a na ceste z viacerých serverov sa vyskytne router, ktorý bude pre tieto cesty spoločný, môžeme s veľkou presnosťou zistiť, kde sa práve tento router nachádza a z neho potom počítať zostávajúcu odozvu a tým vlastne môžeme simulovať chýbajúci server. Na druhej strane si ale musíme uvedomiť i skutočnosť, že toto by sme mohli použiť iba v prípade, že presnosť zistenia je dostatočná na to, aby sme túto pozíciu mohli prehlásiť za danú, inak môže tento spôsob viesť ku kumulácii nepresností.

II. Je faktom, že určité štrukturálne informácie o rozložení siete sú dostupné. Pokiaľ ich využijeme, môžeme sa vyhnúť nepresnostiam. Ako príklad je možné uviesť prípad, kedy určitý cieľový počítač alebo router má dve rozhrania. Rozhrania sa javia ako dve samostatné entity a spôsobom vyššie uvedeným by sme mohli dospieť k záveru, že sa nachádzajú geograficky na inom mieste, aj keď, samozrejme, opak je pravdou. S týmto problémom sa zaoberali i autori algoritmu TBG a dospeli k záveru, že je treba v rámci preprocessingu tiež zviazať rôzne rozhrania jedného počítača dohromady, aby k nepresnostiam z tohto dôvodu nedochádzalo.

III. Posledným problémom, ktorý sa pri svojej implementácii snaží vyriešiť TBG, je skutočnosť, že existuje veľké množstvo počítačov, ktoré komunikujú s vonkajším prostredím pomocou jediného routeru. Tieto počítače, pre ktoré táto podmienka platí, môžu pokrývať rôzne veľkú oblasť. Aby sme sa vyhli chybám, ktoré by pramenili z faktu, že každý paket vstupujúci do našej takto vykonštruovanej oblasti prechádza jediným serverom, musíme túto informáciu zistiť, čo znamená, že musíme zistiť veľkosť tejto oblasti a stroj, ktorý ako tento router slúži. Na zistenie tejto konkrétnej topografie využíva tento algoritmus spôsob, že posiela požiadavky na určité servery, o ktorých vieme ich geografickú lokalitu, a pri tejto činnosti je možné takúto informáciu zachytiť a zaznamenať.

Náznak konkrétneho riešenia

Obecne sa dá povedať, že postup geolokácie tak, ako bol popísaný vyššie a takisto implementovaný v pripojenom programe, je možné použiť i pre tento prístup. Navyše sa v rámci preprocessingu musia uskutočniť určité kroky k zisteniu topografických elementov siete.

Riešenie prvého z vyššie zmienených problémov bol načrtnutý už vyššie, a to, že sa v podstate zvýši počet serverov pomocou lokalizácie určitých routerov na trase, ktoré zistíme pri vzájomnom použití utility traceroute medzi servermi.

Riešenie druhého problému vo výsledku využíva prístup popísaný v [12]. Funguje tak, že sa posielajú UDP požiadavky na vysoko číslované porty rôznych rozhraní. Ako odpoveď posielajú tieto rozhrania obvyčajne ICMP paket „Port unreachable“ so zdrojovou adresou. Pokiaľ sa tieto zdrojové adresy viacerých rôznych rozhraní zhodujú, platí, že ide o rozhrania jedného počítača.

Riešenie tretieho problému spočíva nielen vo vyššie popísanom postupe, ale podobne je možné využiť informácie čiastkové (viz. ďalej kapitola 3.6), ktoré by samy osebe neboli schopné stanoviť určitú relevantnú a zároveň presnú informáciu. Ide v tomto prípade hlavne o to, že nevieme zaručiť správnosť týchto informácií. Túto neaktuálnosť by ale mohla kombinácia s metódou založenou na meraní odozvy odstrániť. Pokiaľ nám výsledok pokusu aplikácie čiastkovej informácie, napr. že `www.bos1123.com` sa bude nachádzať v Bostone, MA (pretože BOS je medzinárodná značka pre letisko v Bostone), takáto pozícia je vylúčená, pokiaľ bude odozva z niektorého serveru narušovať či prekračovať horný odhad vzdialenosti na základe prepočtu stanoveného vyššie pre SOI. V opačnom prípade (keď pre všetky servery tento odhad narušovať nebude) sa naopak pravdepodobnosť správnej domnienky zvyšuje (opäť záleží na hustote pokrytia servermi v danej oblasti).

Varianty

Existujú tri varianty na možnosť aplikácie uvedeného postupu rozdelené podľa počtu informácií, ktoré dostane algoritmus na začiatku behu k dispozícii, čo zároveň znamená odpoveď na otázku, že do akej využíva vyššie zmienené riešenia čiastkových problémov CBG:

(1) TBG-pure

Počíta sa iba so servermi, a žiadne iné lokality ani informácie nie sú stanovené.

(2) TBG-passive

Počíta sa so servermi a tiež pasívnymi landmarkami, ktorých pozíciu síce poznáme, ale nie sme schopní poslať akékoľvek požiadavky na odozvy,

(3) TBG-undns

Počíta sa so servermi, pasívnymi landmarkami a tiež s čiastkovými informáciami, ktoré je nutné otestovať, ako bolo vyššie uvedené, a tie, ktoré neprejdú, je treba vylúčiť.

3.3 Techniky používajúce špeciálne súbory

Tieto techniky používajú špeciálne súbory – rozšírenie hlavičky.

Predpoklady použitia metód:

Táto skupina sa vyznačuje tým, že systém využitých informácií nie je obligatórny a tým pádom závisí na tom, či v konkrétnom prípade sú tieto informácie zadane v rámci týchto špeciálnych súborov. S tým zároveň súvisí riziko nepresných informácií s ohľadom na možnosť zlyhania ľudského faktora pri ich zadávaní.

Možnosť zahrnutia do univerzálneho nástroja:

S odkazom na vyššie uvedené môžu tieto metódy slúžiť ako upresňujúce a doplnkové. Na jednej strane môžu pomôcť presnejšie odhadnúť pozíciu v oblastiach, kde už iné informácie nie sú dostupné. Naopak, musia byť overované inými metódami kvôli ich možným pochybeniam (dostatočne malá dôveryhodná oblasť by ako overenie mohla byť v konkrétnom prípade postačujúca).

3.3.1 DNS GPOS súbor

Tento spôsob je popísaný v RFC 1712. Ide teda o zmienené rozšírenie hlavičky pre DNS – tzv. Resource Record v rámci BIND konfiguračných súborov, a teda spôsob, ako je možné z DNS spraviť nosiča geolokačných informácií. Mnemonický kód je tu rezervovaný na GPOS a typ (numerický kód) je 27. Tento systém bol zaradený do systému RFC ako pokusný a nejedná sa o žiaden štandard, ktorý by bolo nutné dodržiavať. Napriek tomu, myšlienka distribuovanosti informácií ohľadne geografickej polohy bola veľkým myšlienkovým pokrokom (ktorý bol práve možný s príchodom DNS), keďže dovtedy fungoval jedine systém uumap, ktorý sa ako jediný pokúšal zozbierať informácie ohľadne geografických polôh. Tento starší systém práve narážal na problém kvôli svojej centralizovanosti, a i už zanesené informácie sa mohli stať jednoducho neaktuálnymi.

Najväčšou nevýhodou tohto systému, ako systému, ktorý používa rozšírenie hlavičky, je fakt, že jeho funkčnosť a použiteľnosť závisí na tom, koľko administrátorov tieto informácie zadá, a tiež nakoľko dôveryhodné a presné tieto informácie sú.

Dáta, ktoré reprezentujú geografickú šírku, dĺžku a výšku sú reálne čísla s maximálnym počtom znakov 256. Tento systém, kde sú prijateľné práve iba reálne čísla, má predísť riziku obmeny.

Forma zápisu GPOS stanovuje niekoľko základných údajov, ktoré sa zapisujú, ako napr. typ DNS, vlastní, TTL a zmienené zemepisné dĺžka, šírka a výška. Takže by teoreticky mohol zápis vyzerat':

hinault IN GPOS -22.6882 116.8652 250.00

3.3.2 DNS LOC súbor

Tento spôsob je popísaný v RFC 1876. Je to obdoba systému GPOS a tiež sa zapisuje do vyššie zmieneného Resource Recordu. Je novší ako GPOS, ale rovnako je len experimentálnym počínom. Odchýlky sú prevažne v reprezentácii dát a v dátovom formáte. Teoreticky by zápis mohol vyzerat' zhruba:

LOC 42 21 54 N 71 96 18 W -24m 30m

a tak je vidieť, že formát je v stupňoch, minútach a sekundách, pričom je navyše špecifikovaný, o ktorú zo 4 častí (severná (N) a južná (S) zemepisná šírka a východná (E) a západná (W) zemepisná dĺžka) povrchu Zeme sa jedná.

Ďalším významným rozdielom je, že informácie do hlavičky je možné pridávať i do celých podsietí a tak eliminovať nutnosť zadávať IP adresu. Existujú určité miniprogramy, ktoré sú súčasťou zverejneného RFC, a tak je možné ich určitým spôsobom vyhľadávať tú najšpecifickejšiu adresu pre určité účely.

V reálnych systémoch je použité ako jedna z čiastkových informácií v NetGeo.

3.4 Techniky používajúce špeciálny hardware

Predpoklady použitia metód:

Nutná podmienka pre využitie tejto skupiny metód je špeciálny hardware všetkých uzlov využitých pri geolokalizácii, ale i cieľového miesta. Táto požiadavka robí tieto metódy použiteľnými len veľmi obmedzene.

Možnosť zahrnutia do univerzálneho nástroja:

Pre návrh obecného nástroja je možné využiť i tieto metódy, a to jednak na stanovenie určitých záchytných bodov, o ktorých vieme, že špeciálny hardware majú (viz. GPS), a jednak na geolokalizáciu vnútri určitých (napr. dôveryhodných oblastí z podkapitoly 3.1) oblastí, kde uzly už túto nutnú podmienku splňujú (viz. Cricket).

3.4.1 GPS

GPS³ je bezdrôtová technológia, ktorá umožňuje merať pozície uzlov, ktoré majú špeciálne zariadenie, tzv. prijímač (receiver). Princíp merania je podobný ako u vyššie zmienených techník založených na meraní odozvy signálu šíriaceho sa od prijímača k satelitom. Keďže pozícia štyroch satelitov je zistiteľná z navigačnej správy (Navigation Message – orbital position data), a rýchlosť signálu je merateľná, priesečník vzniknutých imaginárnych kružníc nám dá oblasť, v ktorej sa prijímač nachádza. Matematicky najpravdepodobnejšie miesto v tejto oblasti je prehlásené prijímačom ako zisťované miesto.

Chyba merania závisí od rýchlosti signálu. Obecne sa uvádza presnosť +/- 3 metre. Pri vysoko-rýchlostnom signále je možné doceliť presnosť +/- 30 centimetrov.

Nevýhodou tohto inak veľmi spoľahlivého systému je fakt, že je neefektívny vo vnútorných priestoroch. Niektoré systémy sú zamerané priamo na tieto prostredia, napríklad Active Badge⁴, Bat⁵ a Radar⁶, alebo systémy popísané nižšie.

³ Viz. napr. P. Enge and P. Misra, The Global Position System. Proc. of the IEEE, January 1999

⁴ Viac informácií možno nájsť v A. Harter, A. Hopper, P. Steggles, A. Ward, and P. Webster, The Anatomy of a Context-Aware Application, *ACM Mobicom*, August 1999

⁵ Andy Harter and Andy Hopper, A Distributed Location System, for the Active Office. *IEEE Network* Vol.8 No.1, January 1994

⁶ Viac informácií možno nájsť v A. Harter, A. Hopper, P. Steggles, A. Ward, and P. Webster, The Anatomy of a Context-Aware Application, *ACM Mobicom*, August 1999

3.4.2 Place Lab

Place Lab⁷ je technika, ktorá využíva rádiové signálne vysielacie (ako napr. 802.11 access points alebo GSM bezdrôtové vysielacie veže), aby lokalizovala klientskú jednotku. Tieto vysielacie sú vo svete veľmi rozšírené a majú jednoznačné alebo takmer jednoznačné identifikátory (napr. MAC adresy). Táto technika pracuje tak, že načúva jednému či viac vysielateľom, s tým, že vyhladá ich aktuálnu pozíciu v dočasne uloženej lokálnej mape, a odhadne ich pozíciu na základe času odozvy a prepočtu pozície.

Táto technika pracuje dobre i vo vnútorných i vo vonkajších prostrediach, a tak lokálne vie doplniť nepresnosti vyššie zmienenej techniky GPS.

3.4.3 Cricket⁸

Táto technika je prispôbená na použitie vo vnútorných prostrediach. Využíva vysielacie zasadené v stenách a v stropoch, ktoré distribuujú signál po RF kanáloch. Pri každej transmisii vyšle vysielateľ i ultrazvukový signál. Jednotlivé klientské prístroje prijímajú tento signál, identifikujú postupne pozíciu prístrojov, ktorá je presne stanovená, a vypočíta vzdialenosti od nich pomocou prepočtu násobkom rýchlosťou svetla (RF-signal) alebo rýchlosťou zvuku (ultrazvukový signál).

Technika je uspokojená len pre funkciu v rámci jednej budovy a teda nie je vhodná pre vonkajšie prostredie.

⁷ K ďalším informáciám viz. A. LaMarca, Y. Chawathe, S. Consolvo, J. Hightower, I. Smith, J. Scott, T. Sohn, J. Howard, J. Hughes, F. Potter, J. Tabert, P. Powledge, G. Borriello, and B. Schilit. Place Lab: Device positioning using radio beacons in the wild. In *Proc. of Pervasive*, Munich, Germany, 2005

⁸ Bližšie viz. N. Priyantha, A. Chakraborty, and H. Balakrishnan. The Cricket location-support system. In *Proceedings of MOBICOM*, Boston, MA, USA, 2000.

3.5 Techniky používajúce manuálne udržiavané databázy

Predpoklady použitia metód:

Možnosť použitia týchto databáz je spojená s nutnosťou platiť rôzne vysoké poplatky ich tvorcom (komerčné databázy) alebo disponovať s rozsiahlou informovanosťou ohľadne extrakcie dát vzhľadom na nízku úroveň štandardizácie (whois).

Možnosť zahrnutia do univerzálneho nástroja:

Tieto informácie sú v zásade použiteľné a niektoré systémy⁹ ich používajú ako základný zdroj informácií. Ja sa prikláňam k jeho použitiu len ako subsidiárneho nástroja, ktorého dôveryhodnosť je nutné vždy overiť.

3.5.1 Whois databázy

WHOIS je protokol z rodiny TCP, ktorý sa často využíva na poskytovanie informácií pre užívateľov Internetu. Špecifikáciu tohto protokolu môžeme nájsť v štandarde RFC 3912, ktorý nahradil predchádzajúce, zastaralé špecifikácie. Na základe uvedeného štandardu sa obecné využíva port 43, na ktorom servery, ktoré udržiavajú uvedené databázy počúvajú, a následne posielajú informácie na základe požiadaviek týchto klientov.

Pôvodne bol WHOIS systém koncipovaný ako centralizovaný, keďže všetky domény registrovala jedine DARPA samotná, ktorá zároveň spravovala server na odpovedanie požiadaviek ohľadne WHOIS informácií. Server dovoľoval veľké spektrum dotazov, napr. bolo možné vyhľadávať pomocou wildcards. Úroveň zabezpečenia bola veľmi nízka a bola zachovávaná vysoká liberalita služieb. Postupne, ako sa decentralizovala sieť registrátorov a vytváral sa určitý organizovaný systém, zodpovednosť za obsah whois databáz sa preniesla na nich, čo spolu prinieslo problémy s istou štandardizáciou údajov uchovávaných v týchto databázach, ako i so zabezpečením a ochranou týchto údajov.

Z týchto historických dôvodov protokol neobsahuje internacionalizáciu a ani nespĺňa požiadavky na bezpečnosť, ktoré sú vyžadované v novo vytváraných protokoloch. Čo sa týka internacionalizácie, protokol neobsahuje žiadny mechanizmus na zistenie znakovkej sady, ktorú konkrétny server využíva. Pôvodne sa využíval výhradne US-ASCII, ale postupne sa začali používať i iné, špeciálne u serverov nachádzajúcich sa mimo územia Spojených štátov amerických. Táto nemožnosť indikácie má neblahý vplyv

⁹ Napr. NetGeo používa databázu Whois ako primárny zdroj.

na interoperabilitu a tým pádom i na použiteľnosť. Čo sa týka bezpečnostných otázok, protokol neobsahuje mechanizmy na riadenie prístupu, integrity či dôvernosc' a z tohto dôvodu je možné používať tento protokol na informácie, ktoré nie sú nijako chránené a môžu byť používané širokou verejnosťou.

WHOIS databázy a systém má viacero problémov, ktoré ho pre naše účely robia menej použiteľným:

1. neexistencia zoznamu, ktorý by zahŕňal všetky servery, ktoré whois informácie poskytujú, a tak pri implementácii je treba tvoriť ad hoc zoznamy, čo vo výsledku znamená, že rôzne nástroje vychádzajú z rôznych zdrojov a tak dávajú rôzne výsledky,
2. niektorí registrátori nevidujú údaje tak, ako by mali, čo spôsobuje nepresnosť či úplnú absenciu údajov, ktoré hľadáme,
3. rozličné znakové sady (pôvodná US-ASCII doplnené inými sadami podľa voľby registrátora); toto a i celkovo iné formátovanie dát od rôznych registrátorov veľmi sťažuje až znemožňuje automatizáciu,
4. údaje o geografických polohách (resp. o adresách sídel registrovaných) sa časom môžu meniť, pričom tieto údaje nie sú menené, a ostávajú informácie v podobe, aké boli pri registrácii, pokiaľ s ohľadom na bod 2. boli zadané presne alebo vôbec.

3.5.3 Komerčné databázy (GeoIP, IP2Geo, MaxMind, apod.)

Komerčné databázy sú vytvárané rôznymi súkromnými firmami, ktoré potom ponúkajú svoje programy. Tieto výstupy, ktoré sú týmito spoločnosťami vytvorené, sú veľmi rýchle, pretože len prehľadajú túto databázu podľa príslušných IP adries a vydajú cieľovú adresu. Tomuto komerčnému vydaniu predchádza vždy určité zozbieranie dát, a následné udržiavanie a aktualizácie.

Spôsoby, ktorými sa tieto informácie zisťujú, sú rôzne a vo väčšej či menšej miere môžu využívať vyššie zmienené metódy. To nič ale nemení na veci, že drvivá väčšina z dôvodov nízkych nákladov získava jednoduchým spôsobom, a to na základe údajov, ktoré poskytujú registrátori, prípadne upresňujú svoje informácie prehľadávaním WHOIS databáz.

Tento prístup má napriek svojej vysokej rýchlosti vyhľadávania určitej informácie, určité úskalia. Prvým z nich, je nutnosť tieto databázy aktualizovať, teda vždy aplikovať určité metódy zbierania dát, ako bolo naznačené vyššie. To môže spôsobiť zastaralosť údajov, ktoré v konečnom dôsledku slúžia ako podklad pre konečnú lokalizáciu. Ďalším problémom je väčšinový spôsob získavania príslušných informácií, a to z tých od poskytovateľov (providers) pripojenia. Nezrovnalosti nastávajú v prípadoch veľkých poskytovateľov (providers). Títo (a samozrejme i menší, ale u nich to nespôsobuje taký problém) si zaregistrujú veľký úsek IP adries, a registrujú to napr. do svojho sídelného mesta. Potom pri pridelení týchto adries ďalej iným entitám a následnej lokalizácii bude výsledok práve v tomto registračnom meste, a nie tam, kde fakticky sa koncový počítač s danou IP adresou nachádza. Tieto vzdialenosti môžu byť dokonca až v tisíckach kilometrov. Ako príklad môžeme uviesť poskytovateľa americký poskytovateľ pripojenia AOL, ktorý má mnoho klientov mimo iné i v Európe. Pri požiadavke na lokalizáciu niektorého z týchto klientských počítačov, resp. pridelených IP adries, smeruje výsledok prehľadávania do USA, aj keď sa fakticky nachádza v Európe. Ešte je nutné spomenúť jeden trend, ktorý je v súčasnosti veľmi populárny a ktorý zároveň robí túto metódu menej použiteľnou, a to je fakt, že mnoho súčasných koncových užívateľov využíva hostingové služby v štátoch, kde je tento omnoho lacnejší (napr. v Číne), čo nám v konečnom dôsledku spôsobí mislokalizáciu.

3.6 Techniky používajúce čiastkové informácie

Predpoklady použitia metód:

Na použitie tejto skupiny metód nie sú kladené žiadne zvláštne nároky, je nutné mať k dispozícii aspoň čiastočnú databázu geograficky používaných skratiek, aby mohla byť dotazovaná na zhodu.¹⁰

Možnosť zahrnutia do univerzálneho nástroja:

Táto skupina metód môže hrať len výrazne subsidiárnu úlohu, a to hneď z niekoľkých dôvodov. Jednak je to fakt, že takéto informácie je možno extrahovať len z malého percenta DNS pomenovaní a často sú obvyklé výhradne len pre severoamerický kontinent. Druhým dôvodom je i skutočnosť, že tieto mená nie sú nijako štandardizované (ako vyplynie i z ďalšieho textu).

3.6.1 Obecne - mená DNS

Ak máme lokalizovať podľa doménového mena (v prípade máme iba IP adresu, toto meno si môžeme dohľadať reverzným DNS vyhľadávaním), črtá sa nám možnosť využiť čiastkové informácie, ktoré sú skryté v tomto pomenovaní určitého serveru či routeru. Zakódované často môžeme nájsť kódy miest, štátov, ale i medzinárodných letísk. Žiaľ, tieto kódy, ktoré sa občas vyskytnú, nie sú nijakým spôsobom štandardizované, a môžeme ich nájsť vo viacerých formách, podľa príslušného poskytovateľa (provideru), ktorý má väčšinou svoje štandardy nezávislé na ostatných. Napríklad Chicago, IL, USA má viac než 10 rôznych kódových označení, ktoré sa vyskytujú v doménových označeniach (napr. chcg, chcgil, cgil, chi, chicago).

Je možné použiť tieto čiastkové informácie ako spôsob dedukcie lokality napríklad tak, že porovnávame reťazce a ich zhodu, až pokiaľ sa nám nepodarí vytvoriť nejakú dôveryhodnú databázu týchto označení. Na druhej strane, je treba dbať na to, že táto implikácia nás často môže zaviesť tým, že DNS meno bude obsahovať určitý kód bez toho, aby plánovalo implikovať určitú geografickú lokalitu. Preto je vhodné si tieto informácie dodatočne overovať, ako to robí napríklad algoritmus TBG, ako som naznačil vyššie v časti 3.2.

¹⁰ Ide napr. o databázu skratkových pomenovaní miest alebo štátov – ktoré by mohli byť použité v rámci DNS pomenovania.

3.6.2 Medzinárodné kódové označenia letísk

Na rozdiel od vyššie uvedených pomenovaní, ktoré sú pre každého poskytovateľa rôzne, je možné s väčšou istotou konkrétneho určenia skúmať prítomnosť kódových označení letísk v jednotlivých miestach. Tieto kódové označenia sú totiž štandardizované a táto databáza je verejne prístupná. V tomto prípade nie je nutné, aby sme diferencovali prístup a databázu ku konkrétnemu poskytovateľovi, čo zlepšuje čas na implementáciu, ale i zužuje priestor doménových mien, ktoré toto obsahujú.

3.6.3 Konvencie pomenovaní pri dial-up pripojeniach vo Veľkej Británii a USA

Z čiastkových informácií je asi najdôveryhodnejšia metóda, kedy využívame štandardné pomenovania DNS v rámci dial-up pripojení vo Veľkej Británii alebo Spojených štátoch amerických. Toto je dané skutočnosťou, že pri pomenovávaní serverov či routerov sa zaužívala štandardná štruktúra, ktorá v sebe obsahuje i geografické informácie o svojej polohe. Na rozdiel od vyššie zmienených informácií, ktoré sú nedeterministicky zakomponované do DNS pomenovania, v tomto prípade ide o deterministickú výstavbu, a tak je jednoduché identifikovať, či sa jedná v konkrétnom prípade o túto výstavbu, a či je možné kódové označenie lokality považovať za dostatočne presvedčivé.

Výstavba DNS pri dial-up pripojeniach vo Veľkej Británii je nasledovná:

- aa-bb-cc-dd *IP adresa*
- dial/cable *typ technológie pripojenia (kábel, modem)*
- ubbryy *počet UBR, na ktoré je pripojený*
- area 4 *písmenový kód oblasti*
- blueyonder *prípona*

Potom napríklad užívateľ broadbandu, ktorý sa nachádza v Maidstone by mohol mať nasledovnú adresu DNS:

62-31-131-36.cable.ubr01.maid.blueyonder.co.uk

Tieto kódy oblastí sú štandardizované a verejne prístupné.

3.7 Octant – rámcová technika

Myšlienka techniky nazvanej Octant vznikla ako reakcia na nedokonalosti známych techník založených na meraní odozvy¹¹. Jej zmysel je možnosť využitia metód, ktoré boli známe pred ňou tým, že ich jednoducho do tohto rámca zahrnie. Môžeme si ju predstaviť ako určitú osnovu, do ktorej vložíme ostatné techniky, a táto osnova nám bude vravieť ako a kedy ktorá technika vytvorí geolokalizačnú informáciu, pričom ich súhrn znamená presnejšiu odpoveď ako ktorákoľvek z čiastkových techník zaradených do tejto osnovy. Octant teda znamená spôsob výberu relevantných informácií a elimináciu tých ostatných. V tomto zmysle by mohol plniť i úlohu „univerzálneho nástroja“, ako bol prezentovaný vyššie – bohužiaľ tento nástroj implementuje z vyššie uvedených iba často veľmi nespoľahlivé metódy využívajúce čiastkové informácie z DNS pomenovaní.

Podľa môjho názoru, a to s ohľadom na obmedzené použitie iných techník, je možné zaradiť ju do skupiny spôsobov založených na meraní odozvy. Nosným pilierom je vlastne metóda, ktorú vyššie nazývam CBG, ale určitým spôsobom zobecnená. Podobne ako pri CBG sú oblasti v okolí serveru vysielajúceho požiadavky obmedzené zhora – toto obmedzenie teda hovorí, kde sa hľadaný bod nachádza; na druhej strane sa snaží i o obmedzovanie zdola – t.j. stanoviť, kde sa hľadaný bod nemôže nachádzať. S týmto prostým faktom zároveň súvisí, že konečná dôveryhodná oblasť v tomto prípade už nebude predstavovaná konvexným útvarom. Tento nepravidelný útvar je nazývaný aj ako octant.

Všeobecnosť takéhoto zachytenia (a tiež zároveň jeho nepravidelnosť) umožňuje práve zahrnúť do výpočtov i iné metódy, ktoré sa môžu v konečnom dôsledku prejaviť ako obmedzenie celkovej dôveryhodnej oblasti, octantu. Za všeobecnosť tohto popisu je nutné zaplatiť a v tomto prípade sa tak deje prostredníctvom omnoho zložitejšej matematizácie takéhoto útvaru. Problém s týmto zachytením sa táto rámcová technika pokúša riešiť pomocou tzv. Bezierových kriviek, známych z počítačovej grafiky.

Metodika Octantu sa ale s takouto verziou nástroja neuspokojuje a dodáva ešte jednu zaujímavú myšlienku, ktorou sú tzv. sekundárne servery. Tieto servery sú určitými bodmi, o ktorých nevieme, kde sa nachádzajú, ale postup hľadania konkrétnej lokality cieľa nám dá informáciu o ich samotnej pozícii. Táto pozícia, resp. dôveryhodná oblasť je potom využitá ako „server“, a vytvorí ďalšiu oblasť pre celkový prienik. Problémom však je, že táto oblasť môže byť veľmi veľká a potom geolokalizačný okruh vytvorí príliš veľkú oblasť a tým vlastne nedodá žiadnu dodatočnú informáciu. Preto je možné používať tento postup len v prípade, že veľkosť oblasti je dostatočne overená.

¹¹ Spracovaná bola v rámci Cornell University, NY, pánmi Wongom, Stoyanovom a Sirerom. Pre ďalšie informácie viz Wong, Stoyanov, Sirer: Geolocalization on the Internet through Constraint Satisfaction

Obmedzením použitia tejto metódy je nutnosť väčšieho počtu aktívnych serverov, tak ako pri ostatných metódach založených na meraní odozvy. Ich nedostatočný počet často spôsobuje nepresné výsledky.¹² Podľa empirických výsledkov výskumu na podklade výpočtov¹³ je možné už pri počte zhruba 50 aktívnych serverov po Severnej Amerike dosiahnuť strednej odchýlky zhruba 35 km, čo je veľmi prijateľný výsledok. Bohužiaľ sa v rámci týchto výsledkov neuvádza i najhorší prípad, ktorý by mohol dotvoriť celkový obraz o fungovaní tejto metodiky v konkrétnom prevedení.

¹² V obdobnom duchu sa vyjadrili i autori vyššie menovaného článku, Wong, Stoyanov, Sirer.

¹³ Bližšie viz. [13]

4. Implementácia vybraných techník založených na meraní odozvy a jej problematické body

Táto kapitola sa zaoberá implementáciou metód založených na meraní odozvy a jej problematickými bodmi, ktoré vyplynuli z ich praktickej aplikácie¹⁴. Popis týchto metód v dostupnej literatúre sa riadi často výlučne len akademicko-teoretickými aspektami a opomína zložitosti, ktoré je treba riešiť nielen v súvislosti s celkovou adaptáciou na model WGS-84, ale aj heterogenitou prostredia, do ktorého je skúmaná problematika zasadená. Na úvod sa budem zaoberať všeobecnou charakteristikou prístupu k implementácii týchto nástrojov s ohľadom na ich využitie a v ďalšom stručne načriem do problémov, ktoré bolo nutné riešiť a pokúsim sa zhrnúť navrhované a použité riešenie, ktoré tvorí súčasť implementačnej časti tejto práce.

4.1 Všeobecná charakteristika použitej implementácie

Jazyk a platforma

Na implementáciu uvedených algoritmov som zvolil pre zaistenie maximálnej efektivity jazyk C, pričom žiadne časti kódu neboli prebraté z iných programov. Výstup by mal byť schopný prekladu a použiteľnosti na väčšine „unix-like“ operačných systémoch. V rámci implementácie je vo výslednom balíku pripravený Makefile, ktorý tento preklad realizuje.

Paralelizácia

Pri pripájaní na server, posielaní požiadavku na zistenie veľkosti na odozvu, nastáva časová medzera, kedy procesor nevykonáva žiadne inštrukcie, a pasívnym čakaním čaká na odozvu zo serveru. Keďže podobnú informáciu je potrebné zistiť z viacerých rôznych serverov, logicky sa nám ponúka myšlienka paralelizovať túto činnosť a získať výrazné časové zlepšenie.

Táto paralelizácia prebieha tak, že zdvojnásobím otcovský proces toľkokrát, koľko je k dispozícii prístupných brán – prístupných serverov využitých na meranie odozvy. Každému takto vytvorenému procesu je pridelené jednoznačné číslo zo zoznamu

¹⁴ Táto implementácia tvorí elektronickú prílohu č. 5: „Geoloc-tools“

serverov, pričom tento proces bude obhospodarovat' práve tento server (hlavne, čaká na jeho odpoveď¹⁵). Keďže tieto procesy zapisujú zistené časy do jedného súboru, eventuálne komunikujú s jedným naväzujúcim procesom, bolo nutné vyriešiť reguláciu resp. synchronizáciu prístupu k tomuto zdieľanému zdroju. V priloženej implementácii som využil techniku vzájomného vylúčenia pomocou semaforov.

Využitie dáta a servery

Z dôvodu (ne)možnosti získania prístupu na servery a teda s tým súvisiacia obmedzenej (ne)možnosti realizácie celého projektu s ohľadom na uvedené metódy som sa rozhodol skúmať správanie uvedených algoritmov v dvoch geografických oblastiach, a to v oblasti Európy a severnej Ameriky. Toto obmedzenie je dané práve nedostatkom príslušných serverov, ktoré by bolo možné využiť pre toto skúmanie. Na ostatných územiach boli servery príliš riedko zastúpené, aby bolo možné urobiť relevantné závery ohľadne správania týchto metód¹⁶.

Použité servery tvoria prílohy k tejto práci, a to pre severnú Ameriku prílohu „NA-servers“ a pre Európu „EU-servers“. Použité landmarky, ktoré sú vlastne servery/koncové zariadenia, o ktorých vieme s istou dávkou presnosti, kde sa nachádzajú, tvoria prílohy „NA-landmarks“ (pre severnú Ameriku) a „EU-landmarks“ (pre Európu). Tieto dáta som zozbieral z rôznych zdrojov. Databázu „NA-landmarks“ napr. tvorí z veľkej časti výber z databázy knižníc a univerzít v rámci USA, pričom zoznam „EU-landmarks“ tvorí prevažne výber z databázy serverov, ktoré sa zapojili do výskumného projektu RIPE TTM, a sú zverejnené na tomto serveri¹⁷. Čo sa týka serverov, ich zdroje sú diferencované, ale najväčšou mierou prispel výskumný projekt v rámci Univerzity v Stanforde, kde sa ako servery poskytujúce službu cez protokol HTTP zapojilo niekoľko organizácií¹⁸.

¹⁵ Práve čakanie na odpoveď je dôvodom pre paralelizáciu, tj. využitie context-switchingu príslušného operačného systému.

¹⁶ I v odborných publikáciách sa sústreďí výskum maximálne na tieto geografické oblasti, viz. napr. B. Gueye, M. Crovella, A. Ziviani, S. Fdida: Constraint-based Geolocation of Internet Hosts.

¹⁷ Bližšie informácie viz. <http://www.ripe.net/ttm/>

¹⁸ Bližšie informácie o tomto projekte viz. <http://www.slac.stanford.edu/comp/net/wan-mon/traceroute-srv.html>

4.2 Meranie odozvy

Táto skupina implementovaných metód sa vyznačuje spoločným rysom, a to, že sa pozícia hľadaného bodu na povrchu Zeme počíta zo zmeraných odoziev medzi stanovenými dvojicami lokalít. Na to, aby sme boli schopní odhadovať pozíciu cieľa väčšinou nevystačíme s tým, že požiadavky vysielame z nášho počítača; je treba zapojiť aj iné počítače vo svete s pevne danou pozíciou, s tým, že tieto hodnoty nám až potom dodajú relevantnú informáciu pre naše výpočty. Implementované algoritmy sa teda pokúšajú o istú formu distribuovanosti.

V úvodných fázach návrhu tohto nástroja, a tým aj riešenia zadanej diplomovej práce, som uvažoval s určitou formou zapojenia tzv. source-routingu, a tým i možnosťou zapojenia veľkého rozsahu smerovačov (routerov) do tohto zisťovania. Tento systém je podporovaný protokolom IP a spočíva v možnosti dodať do hlavičky paketu informáciu nielen o tom, na akú IP adresu má byť doručený, ale zároveň o tom, ktoré smerovače má na tejto ceste využiť. Týmto systémom by bolo možné efektívne plánovať trasu každého paketu a teda zistiť mnoho informácií ohľadne odoziev z jednotlivých lokalít do iných. Po implementačných pokusoch, konzultáciách a preštudovaní materiálov som bohužiaľ zistil, že tento systém nie je použiteľný. Ba čo viac, do protokolu IP bol zavedený ako skúšobný a po určitej dobe testovania sa ukázala jeho neefektivita hlavne ohľadne plánovania a taktiež narušania fungovania algoritmov na priechodnosť siete. V súčasnosti takmer žiadny smerovač túto utilitu neimplementuje, takže pre reálne použitie nie je vhodný.

Po vyššie naznačenom zistení sa ako jediná vhodná varianta, ako potrebné odozvy merať, ukázalo využitie brán, ktoré sú založené na protokole HTTP. Po zadaní požiadavku, je možné posilať potrebné pakety z adresy, na ktorej sa nachádza tento server, ktorý teda bude v tomto prípade slúžiť ako spomínaná brána. Táto varianta zisťovania má hneď niekoľko úskalí, s ktorými sa bolo treba pri implementácii vysporiadať:

- i) zistiť dostatočne presne pozíciu tohto serveru slúžiaceho ako brána,
- ii) zaistiť spoľahlivosť a presnosť poslaných informácií zo serverov, ako aj
- iii) zabezpečiť dostatočný počet takýchto serverov, aby výsledky metód boli aspoň uspokojivé.

Vyriešiť tieto problematické body sa stalo jedným zo zásadných bodov pri riešení implementačnej časti tejto diplomovej práce, a stalo sa tak s väčšou či menšou úspešnosťou. Cez prizma týchto úskalí a nemožnosť ich úplného a dostatočného vyriešenia je treba nahliadať na celú časť piatu, kde sa snažím zhrnúť výsledky a závery, ktoré vyplynuli práve na podklade týchto implementačných prác.

(Ad i) Zisťovanie pozície brány

Pozície týchto serverov nie sú, a ani neboli známe. Ich odhad som vytvoril pomocou geolokačných nástrojov z časti 3., a to jednak hlavne pomocou komerčných, ručne udržiavaných databáz (3.5.3), a jednak DNS LOC súborov (3.3.2), ako aj doménových mien (3.6.1). Tieto odhady boli v niektorých prípadoch upravované empiricky pri implementácii a pri zistených odchýlkach. Podotýkam, že konečné pozície sú opäť iba odhadmi, i keď overenými viacerými, ale nedôveryhodnými zdrojmi, čo sa v určitom rozsahu mohlo prejaviť i v učených záveroch.

(Ad ii) Zaistenie spoľahlivosti a presnosti prijatých dát

Rozpoznať nespoľahlivosť a nepresnosť spôsobenú nekonštantným zaťažením siete nie je vždy jednoduchá záležitosť. A preto som sa aj tieto problémy pokúsil riešiť diferencovane. Každý server posielal smerom na cieľ, resp. na iný server určitý počet paketov, pričom pri každom pakete, ktorý sa vráti, zmeria jeho RTT. Priemer týchto časov potom som vzal ako požadovanú odozvu. Pokiaľ jeden či viac z časov vykazovali mimoriadne odchylné hodnoty, tieto časy som do výpočtov nezahrnul, a tým v podstate zmenšil rozptyl a docielil aspoň okrajovú elimináciu kolísavého zaťaženia (ďalší spôsob na evidenciu zaťaženia siete je popísaný v algoritme CBG (3.1.3)). Naopak, pokiaľ server vykazuje dlhodobo veľmi nekonzistentné hodnoty, je nutné ho zo zoznamu serverov vylúčiť. Tento krok by mal byť len výnimočný, pretože ovplyvňuje problém *iii*).

(Ad iii) Zaistenie dostatočného počtu brán

Počet takýchto serverov je veľmi dôležitý predpoklad pre uspokojujúce výsledky ďalej uvedených implementácií. Presnosť určenia presnej polohy daného cieľa je vždy závislá práve na tomto počte, a tiež i na hustote, s akou je pokrytá skúmaná cieľová oblasť.

4.3 Prepočet odozvy (RTT) na vzdialenosť

Problémom ostáva časť implementácie, kde máme k dispozícii odozvy zo všetkých serverov a potrebujeme ich nejakým spôsobom previesť na vzdialenosti. Tieto nám dajú hrubý horný odhad vzdialenosti (server, cieľ), ktorý je v danom algoritme považovaný za dostatočný.

Vyššie pri popise samotnej metódy SOI bolo uvedené, že v danej metóde sa uplatňuje jednoduchý prepočet, a to na 1 ms pripadá 0,01 km. Pri metóde CBG je tento prepočet zložitejší, a vychádza z bestline zostrojenej na základe dát získaných v preprocessingu v konkrétnej oblasti skúmania.

Vzdialenosti medzi rôznymi bodmi sú dané v kilometroch a my sa pohybujeme na povrchu elipsoidu WGS-84, ktorý bol definovaný v časti 2. Súradnice jednotlivých bodov sú dané zemepisnou šírkou a dĺžkou, ktoré vychádzajú práve z normy danej týmto elipsoidom. Aby sme boli schopní efektívne používať tieto geografické koordináty, je nutné, aby sme mali pripravený transformačný mechanizmus, ktorý budeme môcť využiť v dvoch základných situáciách:

- i) vypočítanie vzdialenosti medzi dvoma bodmi, ktorých súradnice sú zadané pomocou geografických koordinátov normy WGS-84
- ii) stanovenie množiny bodov (v geografických koordinátoch), ktoré sú v danej vzdialenosti (v kilometroch) od iného bodu (v geografických koordinátoch)

Na túto situáciu nám poslúžila inverzná varianta Vincentyho formule¹⁹, ktorú je možné realizovať i pomocou efektívneho iteratívneho algoritmu. T. Vincenty, autor tejto formule, sa vo svojej práci²⁰ snaží dokázať efektivitu tejto iteratívnej verzie, ktorá závisí od jej konkrétneho nastavenia a využitia. V našom prípade použitia vykazovala pri nastavení presnosti na 0,03 mm veľmi slušné výsledky: konvergovala veľmi rýchlo, pričom prebehlo od 20 do 30 iterácií cyklu.

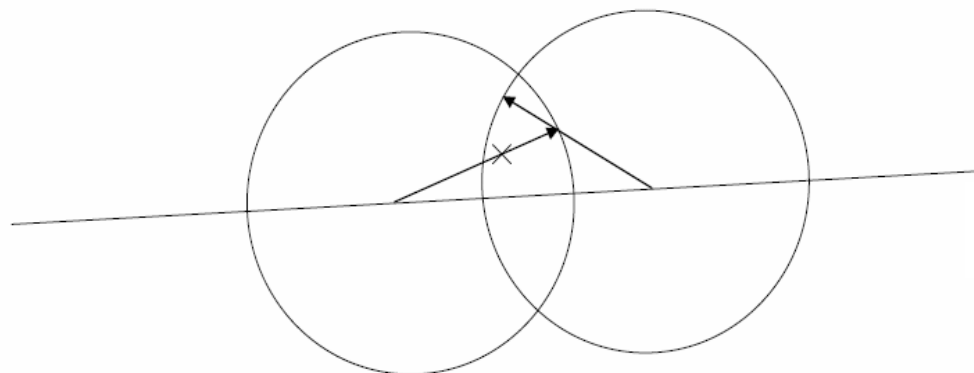
¹⁹ T. Vincenty: Direct and inverse solutions of geodetics on the ellipsoid with application of nested equations. In *Survey Review* No. 176, April 1975.

²⁰ Tam isto ako vyššie.

4.4 Výpočet priesečníkov pri použití geografických ukazateľov

Ďalší z problémov súvisí s vyššie zmienenou otázkou prepočtu odoziev na vzdialenosti. Ide o skutočnosť, že „kruhy“, s ktorými pracujeme v našej implementácii, v skutočnosti vďaka tomu, že sa pohybujeme na plášti elipsoidu podľa WGS-84, nie sú kruhmi v pravom slova zmysle. Je to vlastne *množina bodov, ktoré majú na povrchu Zeme rovnakú vzdialenosť od jedného konkrétneho bodu* (týmto bodom je jeden zo serverov). Bolo by možné analyticky vypočítať tieto priesečníky, ale toto by presahovalo možnosti tejto práce. V implementácii využívam toho, že pomocou Vincentyho formule dokážeme veľmi presne a efektívne merať vzdialenosti a definíciu, ktorú som vyššie vymedzil.

Pre túto konkrétnu situáciu som navrhol iteratívny algoritmus, ktorý sa pokúsi priblížiť postupne k bodu, ktorý znamená je jedným z priesečníkov. Pre tento algoritmus definujem jednoduchý invariant: *pokiaľ urobím polpriamku smerujúcu od jedného z dvoch serverov k zvolenému bodu a vezmeme bod, ktorý leží na tejto polpriamke, a pre ktorý platí, že je vzdialený od serveru presne o danú vzdialenosť* (t.j. polomer „kružnice“). Na tento bod aplikujeme to isté, ale z druhého z danej dvojice serverov. Potom bude platiť, že vzdialenosť tohto výsledného bodu od jedného z potenciálnych priesečníkov bude menšia. Myšlienka kroku algoritmu je naznačená na obrázku obr. 4.1.



(obr. 4.1)

Je zrejmé, že takýmto iteratívnym postupom je možné priblížiť sa k jednému z bodov na dostatočne malú vzdialenosť, aby sme tento bod mohli prehlásiť za hľadaný bod. Pri hodnotení toho, kedy je dané iteratívne prehľadávanie treba ukončiť, musíme skúmať splnenie podmienky pre vzdialenosť od oboch skúmaných serverov.

Takisto je potrebné mať na pamäti, že sa takto dostaneme iba k jednému z priesečníkov a je zásadne nutné zistiť oba, pokiaľ existujú. Z priebehu algoritmu je vidieť, že pre výber priesečníku je rozhodujúca voľba počiatočného bodu. Podľa toho, v ktorej polrovine určenej priamkou spájajúcou obe pozície serverov je tento bod zvolený, konverguje algoritmus k príslušnému priesečnému bodu. Túto voľbu určíme vedením kolmice na spojnicu serverov a posunutím sa v kladnom či zápornom smere o konštantný počet metrov po tejto kolmici.

Tento algoritmus počíta výlučne s prípadom, kedy vznikajú práve dva priesečníky. Zvláštne prípady sa dajú vylúčiť ešte pred behom tohto algoritmu testovaním podmienok súčtu „polomerov“ $R_1 + R_2$ a vzdialenosti medzi servermi d . To znamená, že pokiaľ:

- a) $R_1 + R_2 = d$, potom je jediné riešenie a síce bod ležiaci na tejto priamke a vo vzdialenosti R_1 , resp. R_2 od jedného zo serverov,
- b) $R_1 + R_2 < d$, potom žiadne riešenie neexistuje,
- c) $R_1 + R_2 > d$, potom existujú dve riešenia a to tie, ktoré zistíme aplikáciou vyššie zmieneného algoritmu.

4.5 Riešenie úlohy lineárneho programovania pre bestline

V rámci popisu metódy CBG bola stanovená rovnica stanoviaci problém lineárneho programovania. Na riešenie tejto úlohy definovanej pomocou nerovníc r. 3.1 a rovnice r. 3.2 som použil simplexovú metódu. Predtým bolo nutné upraviť tieto rovnice do štandardizovanej formy, a to tak, že bolo nutné zaistiť, aby boli vo vstupnej matici všetky prvky kladné. V nasledovných riadkoch odvodím výlučne vstupnú maticu do simplexovej metódy pre výpočet bestline pre i -ty server.

Bud' $a_i > 0$. Potom môžeme prepísať nerovnicu (r. 3.1) do nasledovnej formy rovnice:

$$\bar{y} - m_i \bar{x} - b_i - a_i = 0$$

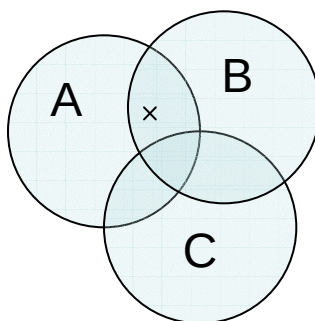
Cieľom je zaistiť kladnosť všetkých koeficientov, takže upravíme rovnicu:

$$\begin{aligned}\bar{y} - m\bar{x} - (m_i - m)\bar{x} - b_i - a_i &= 0 \\ \bar{x}(m_i - m) + b_i + a_i &= \bar{y} - m\bar{x}\end{aligned}\tag{r. 4.1}$$

Ako výsledok dostávame maticu (r. 4.1) so všetkými koeficientmi kladnými, kde $\bar{x}$ a $\bar{y}$ symbolizujú vektory odoziev a vzdialeností, ktoré boli namerané v rámci preprocessingu a kde hľadané hodnoty sú $m_i - m$ a b_i , pričom a_i je pomocná hodnota, ktorú chceme minimalizovať, to teda znamená, že chceme dosiahnuť, že $\sum a_i$ bude čo možno najmenšia. Na takto upravenú maticu môžeme už aplikovať zmienenu simplexovú metódu, ktorá nám určí správne koeficienty pri hodnotách b_i resp. $m_i - m$, pomocou ktorých pri aplikácii rovnice (3) dostaneme žiadaný odhad vzdialenosti.

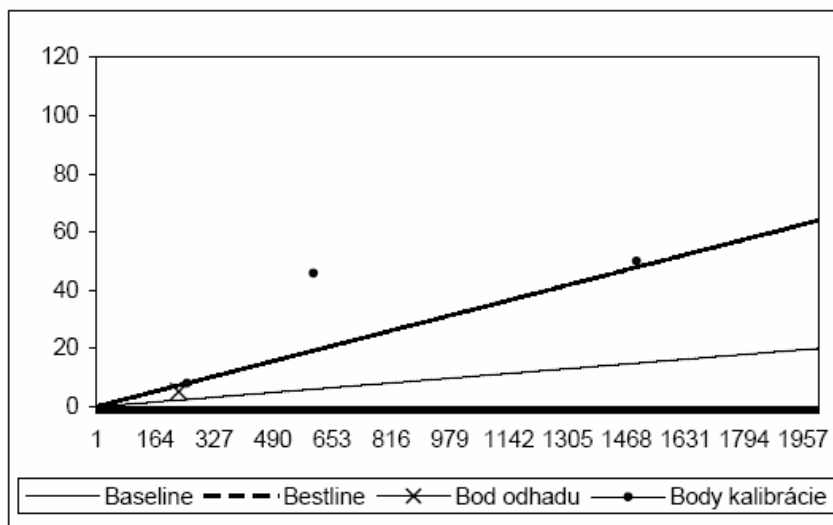
4.6 Problematika mislokalizácie a pokus o jej vyriešenie

Ak používame algoritmus CBG na lokalizáciu nejakého cieľového počítača v pôvodnej verzii, môže sa stať, že priestor, ktorý vytýči tento algoritmus ako tzv. dôveryhodnú oblasť, nebude obsahovať náš cieľový počítač. Táto situácia je vrcholne nežiadúca, ale bohužiaľ pri malom počte serverov (napr. pri počte, ktorý som mal k dispozícii), je to situácia obecné vcelku bežná. Pri empirickom zisťovaní správania sa tohto algoritmu na území Európy sa vyskytol priemerne v 20% prípadoch tento jav. To, že určitý koncový počítač bol mislokálny nastáva práve v situácii naznačenom na obrázku nižšie.



(obr. 4.2)

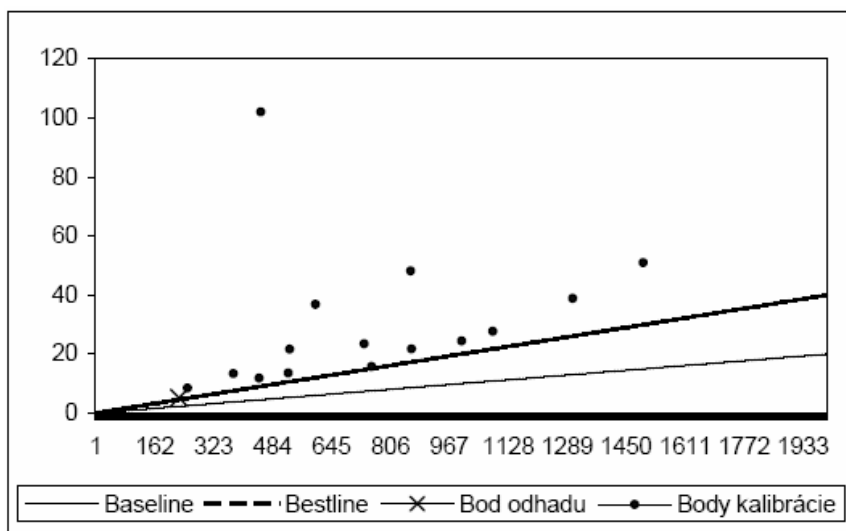
Oblasti A a B zahŕňajú cieľový bod, ale C ho nezahŕňa. Táto situácia je spôsobená faktom, že pri kalibrácii veľkosti polomeru „kružnice“ C bola bestline posunutá až príliš vysoko, a teda hľadaný bod sa dostal až pod túto priamku, čo potom pri prepočte znamená, že sa dostane mimo oblasť kružnice vytvorenej za pomoci tejto bestline, a teda bude i mimo prieniku a teda mimo oblasť označenú ako dôveryhodnú.



(g. 4.1)

Vyššie na obrázku je znázornené tvorenie bestline v európskom priestore. Nasledovná situácia nastala pri pokuse lokalizovať počítač, označený v zozname landmarkov pod č. 11. Pri kalibrácii serveru označeného č. 3 sa podarilo zistiť odozvy iba na 3 zo štyroch ostatných serverov. Skonstruovaný bestline teda vyhovel požiadavkám iba týchto vzdialeností. Pri následnej lokalizácii sa cieľový bod ocitol pod priamkou bestline, a tak došlo k mislokalizácii tohto cieľa.

Na odstránenie tejto nežiaducej vlastnosti (aspoň jej najväčších nedostatkov) som využil zoznam landmarkov tak, ako som už popsoval vyššie. Tento zoznam slúži na spresnenie kalibrácie. Princíp, ako už bolo vyššie zmienené, je v tom, že bestline počítame i za pomoci dvojíc (vzdialenosť, odozva) i na každý z týchto landmarkov. Nižšie na obrázku je zaznamenaný výsledok výpočtu bestline už po aplikácii landmarkov. Zo zoznamu landmarkov bol odstránený lokalizovaný landmark č. 11.



(g. 4.2)

V tomto prípade sa podarilo mislokalizáciu odstrániť za pomoci pasívnych landmarkov. Samozrejme, nejde odstrániť túto nežiadúcu vlastnosť úplne, pretože táto závisí na aktuálnych pomeroch v sieti. Po zavedení pasívnych landmarkov do kalibrácie sa podarilo znížiť výskyt mislokalizácie na 6,6%. Z hľadiska charakteru tejto chyby sa dá ad hoc výrazne obmedziť až odstrániť rekalibráciou tesne pred samotnou geolokalizáciou. Pri testovaní som zistil, že sa takomto postupe vôbec nevyskytla. Na druhú stranu, toto znamená neúmernú časovú náročnosť, ktorú kalibrácia predstavuje.

5. Porovnanie techník založených na meraní odozvy pomocou rozloženia pravdepodobnosti ich odchýlok na podklade implementácie a zhodnotenie ich efektivity

V tejto kapitole sa zameriam na porovnanie metód, ktoré boli implementované v rámci tejto práce. Všetky tieto porovnania sa budem snažiť zachytiť prostredníctvom kumulatívneho rozloženia pravdepodobnosti jednotlivých odchýlok (resp. veľkosti dôveryhodných oblastí). Tzn., že všetky nižšie uvedené grafy budú ukazovať, s akou pravdepodobnosťou sa vyskytla daná alebo akákoľvek menšia odchýlka (resp. veľkosť dôveryhodnej oblasti). Všetky tieto porovnania budú ukázané vždy na dvojici grafov, kde prvý bude vždy reprezentovať merania vykonané v oblasti Európy²¹ a druhý merania vykonané v oblasti Severnej Ameriky²².

Pri meraniach som postupoval nasledovne (pre každý landmark z príslušnej skupiny EU, resp. NA, som vykonal práve jedno meranie):

- i) nastavil som v implementácii výskumný mód (tým, že som zrušil príznak MAPPED);
- ii) nastavil som príslušný landmark ako cieľ i s dátami o jeho skutočnej pozícii;
- iii) z príslušnej skupiny som tento landmark odstránil;
- iv) spustil príslušné prehľadávanie a výsledky zaznamenával²³.

V jednotlivých podkapitolách sa budem venovať najprv problematike základného porovnania jednotlivých metód (CBG, SOI, Shortest Path) – ich odchýlok a veľkostí ich dôveryhodných oblastí, potom sa stručne pokúsím naznačiť rozdielnosť časových nárokov príslušných častí týchto algoritmov a nakoniec ukážem opäť pomocou kumulatívnej pravdepodobnosti, ako sa podarilo zvýšiť efektivitu algoritmu Shortest Path po úprave, ktorú som v jeho rámci vykonal.

Pozn.: Všetky merania popísané v nasledovnej kapitole boli uskutočnené dňa 4.12.2007, 14-18 hod. a sú priložené v elektronickej podobe.

²¹ Použité servery a landmarky pre oblasť Európy je možné nájsť ako elektronickú prílohu pod č. 3, resp. 4.

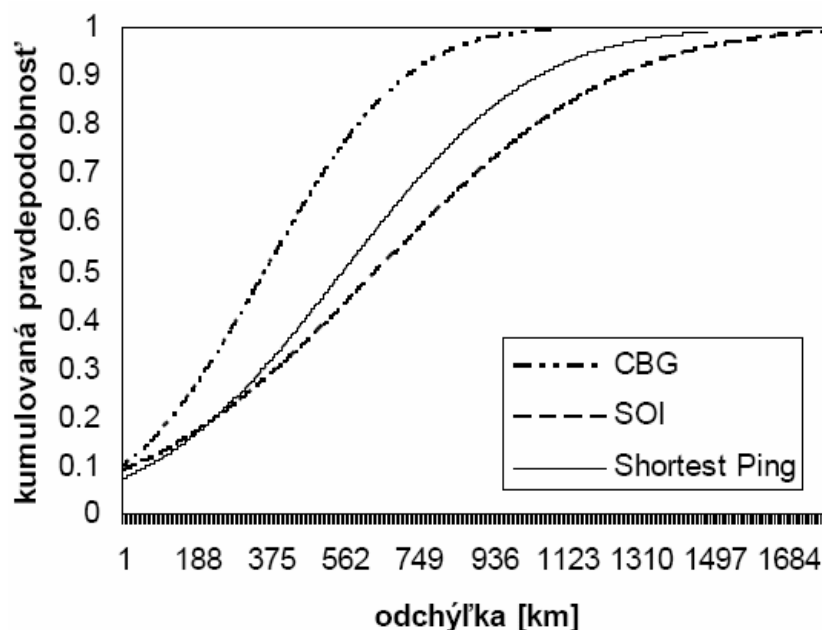
²² Použité servery a landmarky pre oblasť Severnej Ameriky je možné nájsť ako elektronickú prílohu pod č. 1, resp. 2.

²³ Zaznamenané dáta, ktoré boli využité i pre generovanie grafov nižšie v tejto kapitole, tvoria prílohu v elektronickej podobe pod č. 7

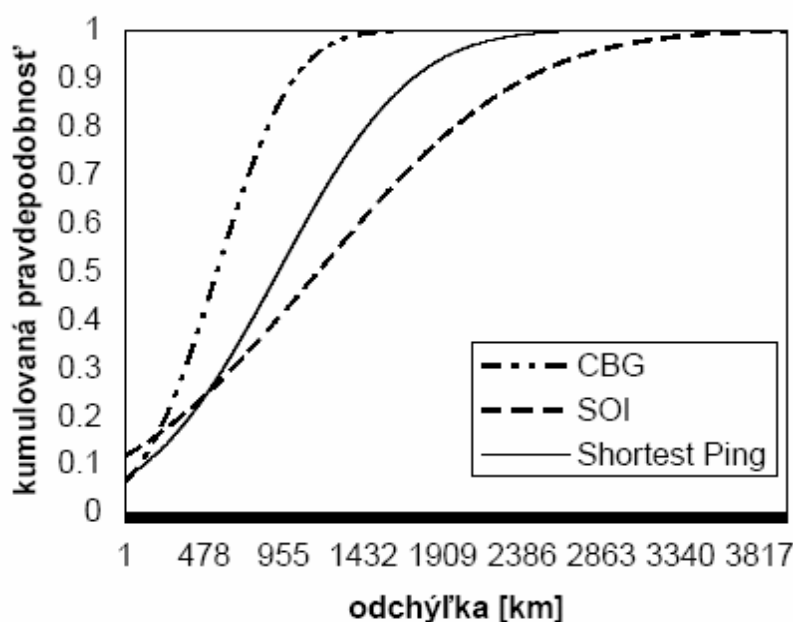
5.1 Základné porovnanie vybraných techník

V rámci tejto podkapitoly sa budem zaoberať vyššie zmieňovaným porovnaním techník. Najprv porovnam všetky tri implementované algoritmy s ohľadom na výskyt ich odchýlok. Potom sa budem venovať porovnaniu veľkosti dôveryhodných oblastí algoritmov CBG a SOI, keďže výhradne tieto dva generujú takéto oblasti.

Porovnanie odchýlok algoritmov CBG, SOI a Shortest Path



(g. 5.1)



(g. 5.2)

Graf g. 5.1 (Európa):

Jako už bolo vyššie zmienené, x-ová os znázorňuje zvyšujúcu sa odchýlku, pričom y-ová os zaznamenáva pravdepodobnosť príslušnej odchýlky a všetkých odchýlok od nej menších. Z grafu je možné vyčítať niekoľko zaujímavých informácií, a to jednak úrovne maximálnych odchýlok, úrovne minimálnych odchýlok a celkový priebeh pravdepodobnosti.

Pri porovnaní maximálnych odchýlok najlepšie výsledky vykazuje metóda CBG, ktorá dosahuje úroveň zhruba 1100 km, metóda Shortest Path je o niečo horšia, s výsledkami na úrovni 1400 km a najhoršie je na tom metóda SOI (1800 km). Naopak, pri porovnaní minimálnych odchýlok, najlepšieho výsledku dosiahla prekvapivo metóda CBG, potom metóda SOI a nakoniec Shortest Path.

Celkový priebeh pravdepodobnosti, i napriek nepochybne horším pravdepodobnostným odhadom pre menšie odchýlky, nastáva u metódy CBG (medián 362 km), horšie je to u metódy Shortest Ping (552 km), a najhoršieho výsledku dosiahla metóda SOI (medián 636 km).

Graf g. 5.2 (Severná Amerika):

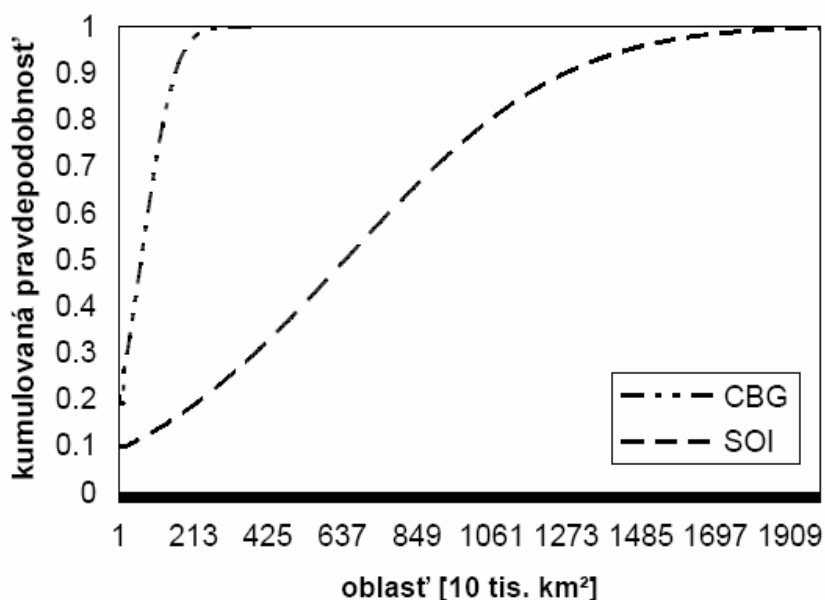
I v tomto grafe x-ová os znázorňuje zvyšujúcu sa odchýlku, pričom y-ová os zaznamenáva pravdepodobnosť príslušnej odchýlky a všetkých odchýlok od nej menších. Opäť budem porovnávať úrovne maximálnych odchýlok, úrovne minimálnych odchýlok a celkový priebeh pravdepodobnosti.

Pri porovnaní maximálnych odchýlok najlepšie výsledky vykazuje metóda CBG, ktorá dosahuje úroveň zhruba 1400 km, metóda Shortest Path je o niečo horšia, s výsledkami na úrovni 2400 km a najhoršie je na tom metóda SOI. Naopak, pri porovnaní minimálnych odchýlok, najlepšieho výsledku dosiahla prekvapivo metóda SOI, potom metóda Shortest Path a nakoniec CBG. Zaujímavou je skutočnosť, že hlavne pri minimálnych odchýlkach dosiahla metóda SOI lepšieho rozloženia (t.j. vyššej pravdepodobnosti) než metóda CBG, pričom práve táto metóda je priamym vylepšením nepresnejšej metódy SOI. K dôvodom, prečo nastáva práve tento jav, sa zamyslím v porovnaní dôveryhodných oblastí nižšie.

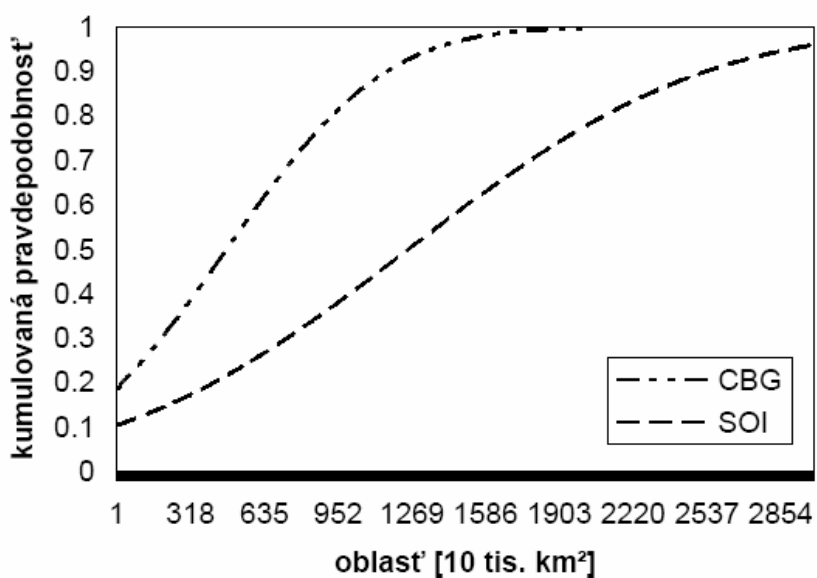
Celkový priebeh pravdepodobnosti, i napriek nepochybne horším pravdepodobnostným odhadom pre menšie odchýlky, nastáva u metódy CBG (medián 548 km), horšie je to u metódy Shortest Ping (908 km), a najhoršieho výsledku dosiahla metóda SOI (medián 1145 km).

Porovnanie grafov g. 5.1 a g. 5.2

Z vyššie uvedených čísel vyplýva, že všeobecne platí, že v oblasti Európy sú nižšie odchýlky vo všetkých vyššie uvedených ukazateľoch vzhľadom ku všetkým metódam. Dôvodom je nižšia rozloha skúmanej oblasti za súčasného obdobného počtu serverov, resp. landmarkov, a tým súčasne vyššia hustota²⁴ lokalít, kde je výsledná pozícia známa.

Porovnanie dôveryhodných oblastí algoritmov CBG a SOI:

(g. 5.3)



(g. 5.4)

²⁴ K efektu vyššej hustoty viz. kapitola 4.2 pod problémovým bodom iii)

Graf g. 5.3 (Európa):

Na tomto grafe, x-ová os znázorňuje zvyšujúcu sa veľkosť dôveryhodnej oblasti, pričom y-ová os zaznamenáva pravdepodobnosť príslušnej veľkosti a všetkých veľkostí od nej menších. Z grafu je možné zistiť úrovně maximálnych veľkostí týchto dôveryhodných oblastí, úrovně minimálnych veľkostí a celkový priebeh pravdepodobnosti.

Pri porovnaní maximálnych veľkostí lepšie výsledky vykazuje metóda CBG, ktorá dosahuje úroveň zhruba 2,8 m. km² než metóda SOI, s výsledkami až na úrovni 19 m. km². I pri porovnaní minimálnych veľkostí lepšieho výsledku dosiahla metóda CBG.

Z vyššie uvedeného je zrejmé, že i celkový priebeh pravdepodobnosti vyznie lepšie pre metódu CBG, ktorá dosiahla medián veľkosti 0,63 m. km², než pre metódu SOI s mediánom o veľkosti 6,4 m. km²). Práve na tomto ukazateli je zrejmé, ako veľmi dokáže metóda CBG v rámci preprocesingu vylepšiť metódu SOI, i keď z porovnaní odchýlok takéto vylepšenie nebolo zrejmé. V konkrétnom prípade sa v podarilo tejto metóde medián veľkosti dôveryhodnej oblasti znížiť na 1/10.

Graf g. 5.2 (Severná Amerika):

I v tomto grafe x-ová os znázorňuje zvyšujúcu sa veľkosť dôveryhodnej oblasti, pričom y-ová os zaznamenáva pravdepodobnosť príslušnej veľkosti a všetkých veľkostí od nej menších. Opäť budem porovnávať úrovně maximálnych veľkostí týchto dôveryhodných oblastí, úrovně minimálnych veľkostí a celkový priebeh pravdepodobnosti.

Pri porovnaní maximálnych veľkostí lepšie výsledky vykazuje metóda CBG, ktorá dosahuje úroveň zhruba 19 m. km² než metóda SOI, s výsledkami až na úrovni 30 m. km². I pri porovnaní minimálnych veľkostí lepšieho výsledku dosiahla metóda CBG.

I tu je zrejmé, že celkový priebeh pravdepodobnosti vyznie lepšie pre metódu CBG, ktorá dosiahla medián veľkosti 4,84 m. km², než pre metódu SOI s mediánom o veľkosti 12,61 m. km²). Zaujímavý poznatok vzíde z porovnania odchýlok a dôveryhodných oblastí v Severnej Amerike. Na jednej strane, pravdepodobnosť nízkych odchýlok je vyššia u inak vo všetkých ostatných ohľadoch horšieho algoritmu SOI, a na druhej strane pravdepodobnosť malých veľkostí dôveryhodných oblastí je už vyššia u CBG. Tento paradox by mohol byť spôsobený tvarom vytváraných oblastí, keď odhad vo väčšej oblasti bude bližšie než odhad v menšej z nich, a to z dôvodu, že tento odhad je počítaný práve ako ťažisko spomínaných dôveryhodných oblastí.

5.2 Stručné zhodnotenie časovej náročnosti

Pre všetky vyššie zmieňované algoritmy platí, že v porovnaní s ostatnými operáciami je jednoznačne časovo najnáročnejší prístup na vzdialený server a čakanie na jeho odozvu. Samotná geolokalizácia prebieha tak, že sa pristúpi na každý zo vzdialených serverov, ktorý je uvedený v príslušnom zozname a počká sa na jeho odozvu. Potom sa matematickou transformáciou dovodí odhad pozície. Tento čas transformácie je zanedbateľný oproti času nutného na čakanie. To znamená, že keďže všetky algoritmy vychádzajú z rovnakého zoznamu týchto serverov, bude tento čas strávený na každom algoritme podobný.

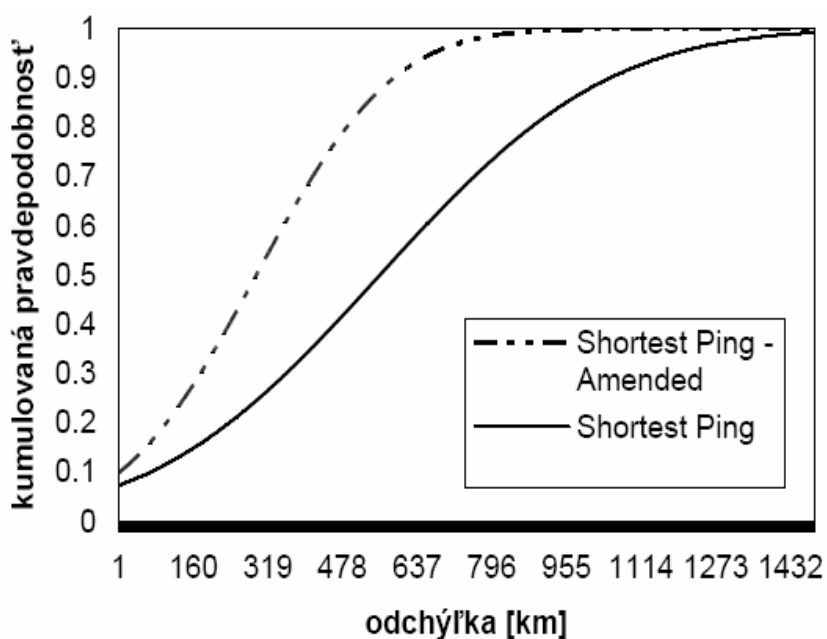
Za zmienku stojí fakt, že algoritmus CBG a pozmenený algoritmus Shortest Path potrebujú na optimálne fungovanie vykonať kalibráciu. Táto kalibrácia nemusí predchádzať každú geolokalizáciu, a preto nebola braná do úvahy v predchádzajúcom odstavci. Je nutné ju vykonať pri zmene podmienok. Jej náročnosť môžeme odvodiť pomocou časovej náročnosti geolokalizácie samotnej. Pokiaľ si tento čas označíme ako T a počet landmarkov N_L , potom bude výsledná náročnosť rovná $T \cdot N_L$. Je nutné teda zvažovať, kedy je kalibráciu naozaj nutné aplikovať.

5.3 Vylepšenie algoritmu Shortest Path a jeho vplyv na efektivitu

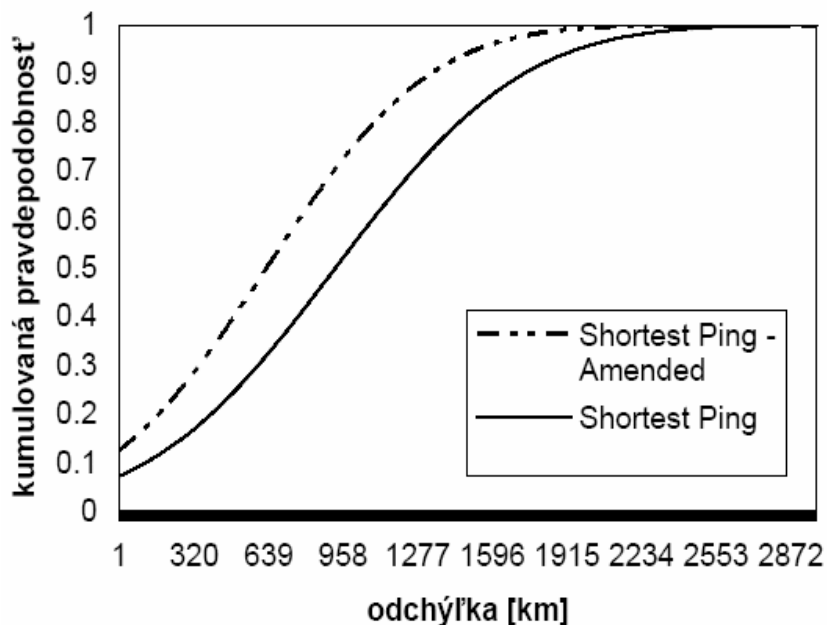
V predchádzajúcej kapitole bolo ukázané, že i napriek svojej algoritmickej jednoduchosti, dosahuje tento algoritmus veľmi uspokojivých výsledkov. Je zrejmé, a ako už bolo i v kapitole 3.1.1 popísané, jeho efektivita výrazne závisí na hustote, s akou servery, ktoré vysielajú pakety smerom s cieľovým počítačom, pokrývajú priestor, ktorý skúmame.

V rovnakej kapitole bola aj zmienená myšlienka, ako by sa mohol tento algoritmus vylepšiť v prípade, s akým som bojoval i ja pri implementácii, a to je nedostatok týchto serverov. Ide o myšlienku doplnenia systému dát landmarkami so známou pozíciou a mierne pozmenenie tohto inak veľmi jednoduchého algoritmu. Podobne ako vyššie som porovnal pomocou kumulatívneho rozloženia pravdepodobnosti tento pozmenený algoritmus s pôvodným algoritmom.

Porovnanie pôvodného a pozmeneného algoritmu Shortest Ping



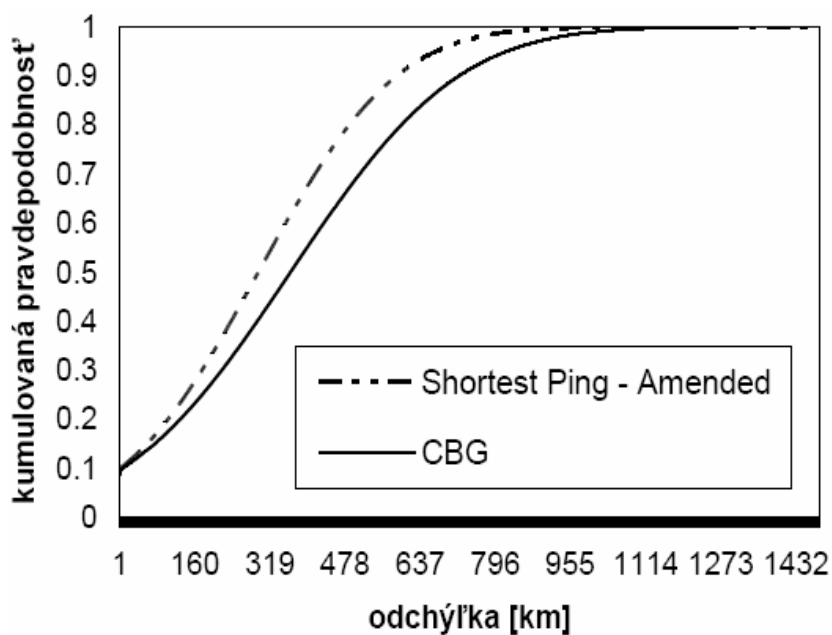
(g. 5.5)



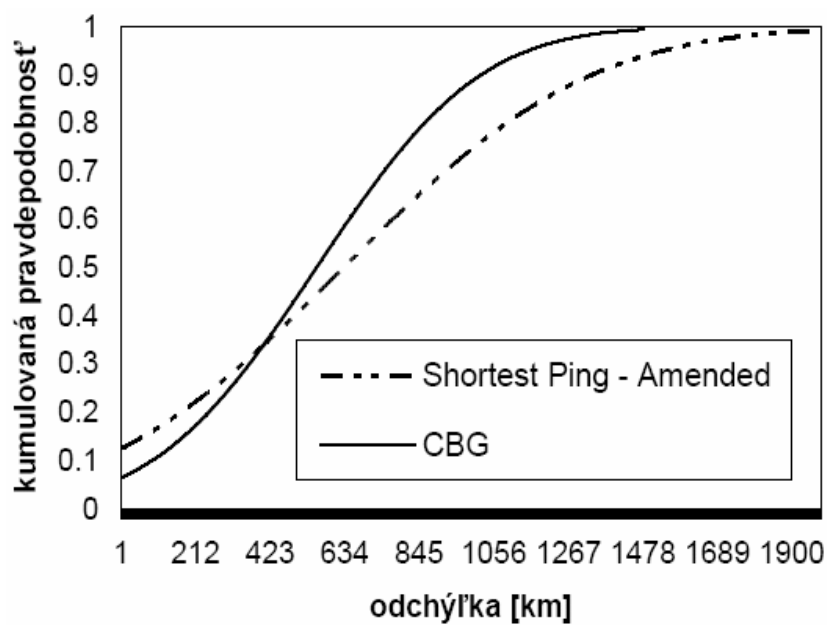
(g. 5.6)

Výsledky, ktoré sú zobrazené na grafoch g. 5.5 a g. 5.6 ukazujú, že presnosť pozmeneného algoritmu Shortest Ping sa výrazne zlepšila oproti pôvodnej verzii. V obidvoch oblastiach je toto vylepšenie platné bezvýnimočne pre celý priebeh rastu pravdepodobnostnej krivky. V Európe dosiahol medián odchýlky pôvodného algoritmu hodnotu 563 km, zatiaľ čo pozmenený algoritmus sa vyznačuje s výrazne lepším mediánom s hodnotou 296 km. Výraznejšie je vylepšenie v Európe, kde, ako bolo už vyššie rozoberané, je priestor hustejšie posiaty servermi a landmarkami.

Toto zistenie ma zároveň zaviedlo k myšlienke porovnať tento pozmenený Shortest Ping algoritmus s algoritmom CBG. Znovu som použil metódu zobrazenia kumulatívneho rozloženia pravdepodobnosti.

Porovnanie pozmeneného algoritmu Shortest Ping a metódy CBG

(g. 5.7)



(g. 5.8)

V porovnaní týchto dvoch metód vyplýva nasledovné:

Graf g. 5.8 (oblasť Severnej Ameriky)

V tejto oblasti sa zdajú byť metódy vcelku vyrovnané, s tým, že CBG sa javí byť výhodnejší. Jeho medián sa pohybuje na úrovni 554 km, pričom mediánu pozmeneného algoritmu Shorterst Path je na úrovni 642 km. Takisto maximum odchýlky činí zhruba 1400 km naproti 2000 km pozmeneného algoritmu ShP. CBG je praktickejší i z toho dôvodu, že poskytuje určitý údaj o tom, kde sa daný cieľ s istotou nachádza, tzv. dôveryhodnú oblasť, pričom pozmenený ShP takúto informáciu neposkytuje. Jediným kladným ukazateľom v prospech pozmeneného ShP algoritmu je vyššia pravdepodobnosť nižších odchýlok, ktorá ale v porovnaní s ostatnými kritériami hrá subsidiárnu rolu.

Graf g. 5.7 (oblasť Európy)

Naopak, v tejto oblasti výrazne dominuje pozmenený algoritmus ShP. Je lepši v rozhodujúcich kritériách. Hodnota jeho mediánu je 295 km, pričom hodnota mediánu algoritmu CBG je 367 km. I pri porovnaní maximálnych odchýlok dosahuje pozmenený ShP nižšej hodnoty, ktorá je zhruba na úrovni 900 km, pritom CBG je až na úrovni zhruba 1150 km. CBG sa môže pýšiť jedine svojou obecnou výhodou, a to, že poskytuje odhad dôveryhodnej oblasti, ktorý je pre túto oblasť veľmi dobrý²⁵.

Záver porovnania

Obecne by sa z tohto dal vytvoriť záver, že v hustejšie posiatej oblasti servermi a landmarkami (ako napr. oblasť Európy) môže dosiahnuť za daných podmienok lepšie výsledky pozmenený algoritmus Shortest Path, kdežto v redšej oblasti (ako napr. oblasť Severnej Ameriky) môže dominovať algoritmus CBG.

²⁵ Viz. graf g. 5.3

6. Vznik odchýlky odozvy a jej vzťah k celkovej chybe merania

Túto kapitolu som pojal viac teoreticky, kde skromným cieľom prvej časti je identifikovať príčiny vzniku odchýlky a načrtnúť matematické odvodenie jej veľkosti s odrazom spomenutých príčin; v druhej časti sa pokúsím popísať vplyv jedinej odchýlky na celkovú chybu merania, pričom budem polemizovať i vplyv celkového rozloženia serverov na túto celkovú chybu. Opäť sa pokúsím vytvoriť obecnú závislosť, ktorá by tento jav popisovala.

6.1 Vznik odchýlky odozvy. Vplyv linearizácie signálu

Na úvod tejto prvej časti sa pokúsím stručne charakterizovať základné príčiny vzniku tohto javu a potom sa bližšie pozriem na tie najdôležitejšie s tým, že sa pokúsím navodiť matematickú závislosť, ktorá je ale napojená na určitých parametroch dátových tokov.

1. Nepriamosť dátových tokov.

Prenos dátových správ po sieti v prevažnej väčšine prípadov (na rozdiel napr. od systému GPS) nie je priamy a paket pri svojej ceste od vysielateľa k príjemcovi často prejde geograficky dlhšiu vzdialenosť než je tá skutočná, „vzdušná“. Geolokalizačné systémy sa práve snažia cestu paketu zlinearizovať, z dôvodu, aby boli schopné prepočet a neskôr i odhad efektívne vyhodnotiť. Bližšie sa tejto príčine budem venovať nižšie.

2. Nerovnomernosť rýchlosti prenosu v rôznych sieťových segmentoch.

Vyššie popísaný problém nie je jediným. K nepriamosti jednotlivých dátových ciest môže, a vo väčšine prípadov naozaj pristupuje, nerovnomernosť rýchlosti prenosu v jednotlivých dátových segmentoch. Spôsobovať ich môže nielen rozdielna prenosová technológia, ale aj možný rozdielny stupeň zahltenia či priepustnosti siete vôbec. Opäť je v rámci geolokalizačných nástrojov nutné kvôli samotnej možnosti, ale aj efektívnosti výpočtov tieto nerovnomernosti linearizovať. I tomuto problému sa nižšie budem ešte venovať.

3. Lokálne oneskorenie.

Ďalšou príčinou odchýlok je lokálne oneskorenie. Jedná sa o jav, kedy určitý uzol v sieti vykazuje pravidelne dodatočné odchýlky, ktoré sú pripočítavané k meraniu. Táto skutočnosť môže byť spôsobená jednak nízkorychlostným pripojením tohto uzla do siete, a jednak lokálnym zahltením, či oboma zároveň. Geolokalizačné nástroje sa s týmto problémom pasujú, bohužiaľ nie vždy s maximálnou efektivitou. V ďalšom texte sa ešte tejto problematiky ešte dotknem.

4. Vplyv zdieľaných dátových ciest.

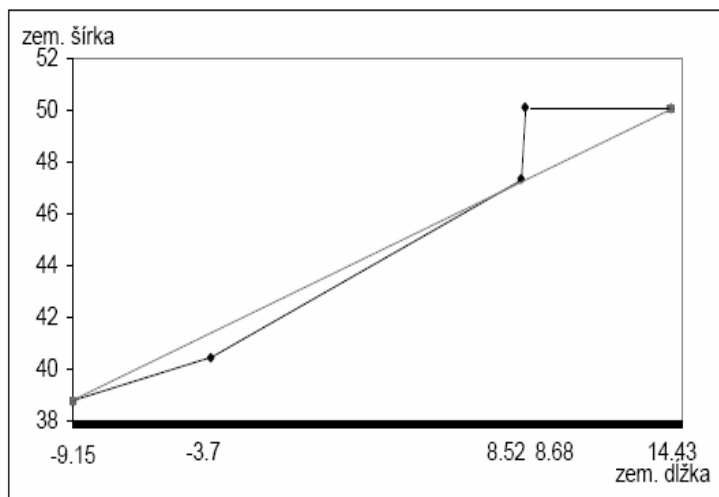
Zdieľané dátové cesty sú podľa môjho názoru okrajovou záležitosťou, pokiaľ skúmame vzniknuté odchýlky z pohľadu jediného uzla v sieti. Naopak, výrazne môžu vplyvať na celkovú presnosť geolokalizácie tým, že vlastne obmedzujú možnosti finálnej triangulácie, hlavne pokiaľ sa nachádzajú v okolí cieľového bodu. Vtedy totiž dôjde ku vzniku neopodstatnených redundancií u väčšiny zúčastnených serverov. Táto príčina môže byť považovaná za špeciálny prípad problému ad 1, a preto sa ním už ďalej nebudem zaoberať.

V nasledujúcom texte sa pokúsím postupne dopracovať k odvodeniu vplyvu týchto jednotlivých príčin na veľkosť odchýlky. Zároveň budem modelovať situácie v konkrétnom prípade prechodu paketu od serveru k cieľovému počítaču²⁶.

Problém ad 1

Na grafe g. 6.1 je model konkrétneho geografického priechodu paketu od serveru k cieľu. V tejto situácii sú zachytené výhradne hlavné smerovače, ktoré dátovú cestu určili. Prípadné iné zmeny trasy paketu nie sú zachytené.

²⁶ Toto meranie bolo uskutočnené 5. augusta 2008, 17h., požiadavka bola vyslaná zo serveru označeného pod č. 4 v zozname gateways_EU (geograf. lokalita Lisabon, PT) na cieľový počítač, server ms.mff.cuni.cz, lokalita Praha, CZ



(g. 6.1)

Z tohto grafu je vidieť, že dátová cesta je dlhšia, ako skutočná vzdialenosť medzi vysielajúcim a prijímajúcim uzlom. Aby sme zistili, aký vplyv má linearizácia dátových ciest na vznik odchýlky, budeme predpokladať, že rýchlosť prenosu paketu je konštantná²⁷. Budem sa snažiť odvodiť vzorec pre našu situáciu, aby sme boli schopní porovnať efekt linearizácie.

Podľa Pythagorovej vety môžeme elementárnu časť dĺžky funkcie $f(x)$ vyjadriť podľa vzorca:

$$\sqrt{(dx)^2 + (f'(x)dx)^2} \quad (\text{r. 6.1})$$

Z tohto potom hneď dostávame obecný vzorec pre dĺžku funkcie $f(x)$ od a do b :

$$\int_a^b \sqrt{1 + (f'(x))^2} dx \quad (\text{r. 6.2})$$

My ale máme špeciálny prípad, a to, že máme pospojované jednotlivé lineárne funkcie (pretože uvažujeme linearitu cesty od jedného smerovača k druhému), označím si preto pre tieto účely jednotlivé čiastkové funkcie ako $f_i(x)$, kde $x \in \langle c_{i-1}, c_i \rangle$ a $i = 1..n$. Tým, že ide o lineárne funkcie, môžeme zároveň označiť $f_i(x) = a_i x + b_i$.

Potom dovodíme nasledovný vzťah:

²⁷ Budeme uvažovať, že vo všetkých dátových segmentoch je možné priradiť pomer $1\text{km}=0.01\text{ ms}$. Tento prepočet definujem s ohľadom na metódu SOI, kde je odvodený tento odhad ako horný s ohľadom na fyzikálne vlastnosti svetla a jeho rýchlosti v optickom vlákne.

$$\sum_{i=1}^n \left(\int_{c_{i-1}}^{c_i} \left(\sqrt{1 + (f'_i(x))^2} \right) dx \right) = \sum_{i=1}^n (c_i - c_{i-1}) \sqrt{1 + a_i^2} \quad (\text{r. 6.3})$$

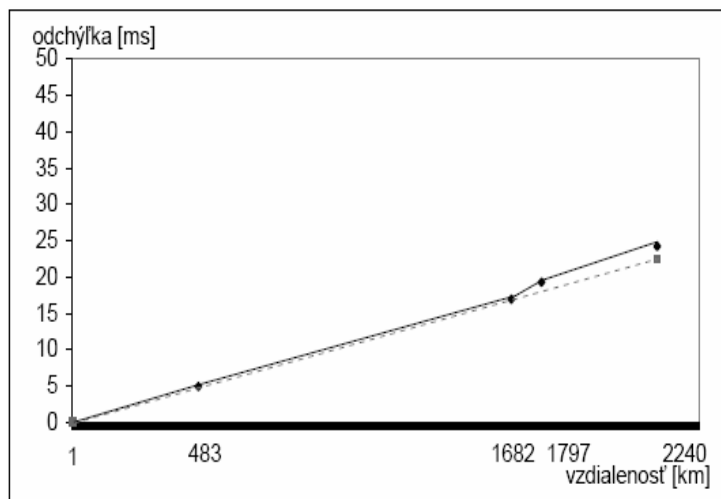
V rovnici r. 6.3 máme teraz obecný vzorec pre výpočet dĺžky celkovej cesty paketu. Pre výpočet celkového faktoru, ktorý nám povie, koľko percent z linearizovanej vzdialenosti bude činiť odchýlka, musíme vziať do úvahy i túto vzdialenosť. Linearizovanú vzdialenosť označím ako $f(x) = ax + b$:

$$\frac{\sum_{i=1}^n (c_i - c_{i-1}) \sqrt{1 + a_i^2} - (c_n - c_0) \sqrt{1 + a^2}}{(c_n - c_0) \sqrt{1 + a^2}} \quad (\text{r. 6.4})$$

Vo vzťahu r. 6.4 máme faktor stanoviaci pre jednotlivé prípady nárast odchýlky. Pre náš modelový prípad (paket putujúci z Lisabonu do Prahy) je tento faktor rovný 8.05%.

Problém ad 2:

Aby sme boli schopní podobne odvodiť i vzťah pre nerovnomernú rýchlosť jednotlivých dátových segmentov, najprv na konkrétnom prípade transformujeme graf g. 6.1, ktorý zachytáva samotné geografické vzdialenosti pri závislosti zemepisnej šírky a dĺžky na graf g. 6.2, kde už sú je zobrazená závislosť medzi vzdialenosťou a odozvou. Keďže stále uvažujeme, že rýchlosť je rovnomerná, t.j. 0.01 ms na 1 km, závislosti ostanú obdobné. Tento nový graf už ale stanoví kumulovanie odchýlok, pričom nové „dôležité“ body, t.j. $c_0, \dots, c_n$ už nebudú zemepisné šírky jednotlivých smerovačov, ale ich kolmé priemety na linearizovanú funkciu z grafu g. 6.1., ktoré budem naďalej označovať ako $d_0, \dots, d_n$.



(g. 6.2)

Čiastkové lineárne funkcie, ktoré odrážajú vzťah medzi vzdialenosťou a odozvou pre jednotlivé dátové toky medzi jednotlivými smerovačmi budem naďalej označovať ako $g_i(x) = p_i x + q_i$, kde $x \in \langle d_{i-1}, d_i \rangle$ a $i = 0..n$ a celkovú linearizovanú závislosť²⁸ ako $g(x) = px + q$, kde $x \in \langle d_0, d_n \rangle$. Pre výslednú veľkosť odchýlky teraz nie je smerodatná dĺžka funkcie, ale jej hodnota v bode d_n .

$$\begin{aligned} g_n(d_n) &= g_n(d_{n-1}) + p_n(d_n - d_{n-1}) = g_{n-1}(d_{n-1}) + p_n(d_n - d_{n-1}) \\ &= p_n(d_n - d_{n-1}) + p_{n-1}(d_{n-1} - d_{n-2}) + \dots + p_1(d_1 - d_0) \end{aligned} \quad (\text{r. 6.5})$$

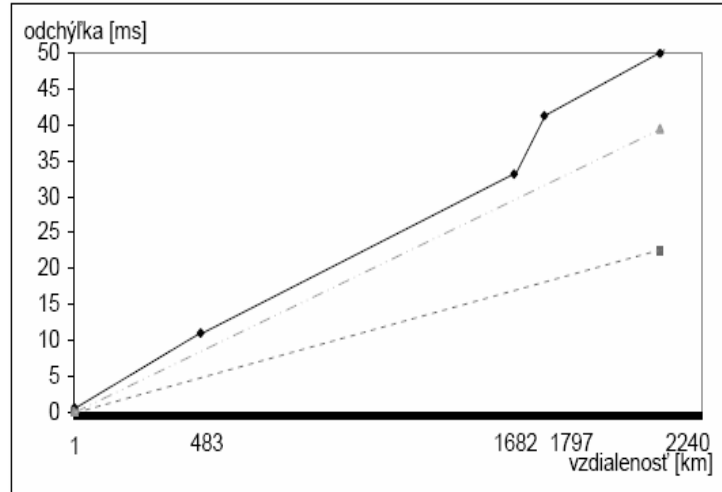
Potom teda faktor pre baseline pri rovnomernej rýchlosti prenosu, rovnaký ako v rovnici r. 6.4 bude vyzerať nasledovne:

$$\frac{\sum_{i=1}^n p_i(d_i - d_{i-1}) - (0.01(d_n - d_0))}{0.01(d_n - d_0)} \quad (\text{r. 6.6})$$

Teraz už prejdeme k samotnému výpočtu celkovej odozvy s tým, že budeme uvažovať i nerovnomernú rýchlosť v rámci prenosových segmentov²⁹. Na grafe g. 6.3 je ukážka takejto závislosti, pričom tento graf zobrazuje obdobne vzťah medzi vzdialenosťou a odozvou:

²⁸ Táto závislosť je v terminológii algoritmu CBG označovaná ako baseline.

²⁹ Pre účely tejto modelovej ukážky budem uvažovať, že v rámci jedného segmentu medzi smerovačmi je závislosť lineárna, a teda rýchlosť prenosu rovnomerná.



(g. 6.3)

Označíme si pre každý segment medzi smerovačmi index spomalenia ρ_i , kde $\rho_i \in \langle 1, \infty \rangle$, ktorý odráža spomalenie oproti ideálnej rýchlosti v optickom vlákne, 100 km/ms. Taktiež definujeme lokálne oneskorenie ako q_0 . Z tohto vyplýva, že celková odozva bude mať veľkosť:

$$g_n(d_n) = \sum_{i=1}^n p_i \rho_i (d_i - d_{i-1}) + q_0 \quad (\text{r. 6.7})$$

Celkový faktor bude vzhľadom k baseline veľmi podobný ako bol uvedený vo výraze r. 6.6, jediná zmena bude v doplnení faktoru spomalenia do definície čiastkových funkcií. Zaujímavejšie bude preto navodiť tento faktor voči bestline, známou z algoritmu CBG, a porovnať, ako veľmi dokáže znížiť tieto odchýlky. Bestline definujeme ako $g_b(x) = p_b x + q_b$. Potom faktor vypočítame ako:

$$g_n(d_n) = \frac{\sum_{i=1}^n p_i \rho_i (d_i - d_{i-1}) + q_0 - (p_b (d_n - d_0) + q_b)}{p_b (d_n - d_0) + q_b} \quad (\text{r. 6.8})$$

Zaujímavé bude porovnanie týchto celkových faktorov v našom konkrétnom prípade voči bestline³⁰ a voči baseline. Faktor k baseline je 123% (z toho 120.5% na

³⁰ Bestline použitý pre tento účel bol vytvorený kalibráciou európskych serverov v rámci priloženého programu Geoloc_tools dňa 5. srpna 2008, 17h15m

nerovnomernosť rýchlosti a nepriamosť dátovej cesty, 2.5% na lokálne oneskorenie), pričom faktor k bestline iba 26.66%.

I keď z vyššie uvedeného by sa dalo dovodiť, že najväčší vplyv na odchýlku má linearizácia prenosovej rýchlosti, nemusí tomu tak nutne byť. Keď totiž pozorne porovnáme grafy g. 6.2 a g. 6.3, zistíme, že tvary týchto zložených funkcií sú veľmi podobné. Z toho plynie, že práve pridaný faktor spomalenia prehĺbil nerovnomernosti už spôsobené nepriamosťou dátových ciest.

Všeobecne povedané, linearizácia prenosovej rýchlosti úzko súvisí s linearizáciou dátových ciest, obe sa vzájomne ovplyvňujú, a nedá sa teda stanoviť, ktorá z týchto dvoch nepravidelností má väčší vplyv – toto je záležitosťou konkrétneho prípadu.

Je vhodné si povšimnúť, že v konkrétnom prípade sa podarilo v rámci tvorby bestline, ako je popísané v algoritme CBG znížiť faktor odchýlenia takmer o 100% oproti baseline použitej v rámci algoritmu SOI. Faktor, ktorý ukázal modelový príklad, sa dá zaradiť do celkového faktoru, ktorý vychádzal v rámci skúmania vykonaného pre účely kapitoly 5, kde sa odchýlka pohybovala v priemere okolo 20%. Tento údaj je použitý i pre druhú časť v rámci tejto kapitoly.

Je jasné, že odchýlky vzniknuté vplyvom linearizácie sú často veľké³¹. Na druhej strane opačná tendencia, ktorá sa snaží zabrániť mislokalizácii, pričom sa ako bestline berie až najbližší horný lineárny odhad všetkých nameraných odchýlok z daného uzla, spôsobuje práve tieto nárasty. Bez ohľadu na to, odchýlku pre jeden server je možné relativizovať pri interakcii s inými servermi pri vytváraní dôveryhodnej oblasti, avšak iba v obmedzenom rozsahu. Touto problematikou, t.j. vplyvu odchýlky na celkovú chybu merania, sa budem venovať v nasledujúcej stati.

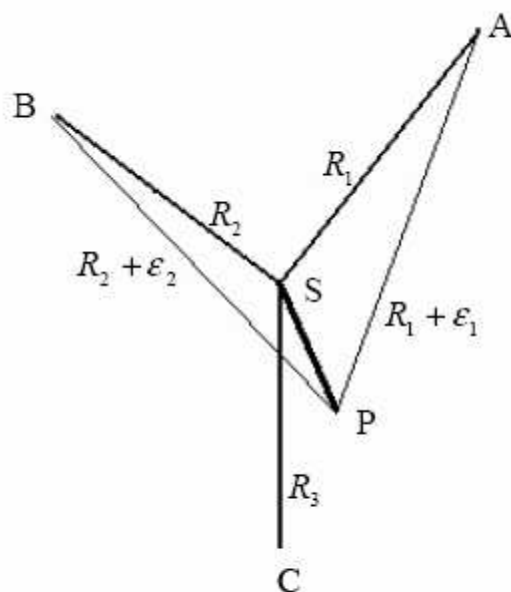
³¹ Toto platí hlavne, pokiaľ je server veľmi vzdialený od cieľa. Napr. v našom modelovom príklade je vzdialenosť 2240 km a pri zvýšení odhadu o 26.66% voči bestline, čo pri geolokalizácii prostredníctvom nástroja CBG znamená vyšší časový údaj o cca 10 ms, vytvorí odchýlku 550 km.

6.2 Vzťah odchýlky odozvy k celkovej chybe merania

Geolokalizačné metódy, ktoré sú postavené na meraní odozvy sú citlivé práve redundancie, ktoré vznikajú pri týchto meraniach³². Cieľom tejto state je skúmať, ako práve tieto redundancie vo svojom súhrne vplývajú na celkovú chybu merania. Logický predpoklad je čo najviac minimalizovať redundancie nameranej odozvy každého z geolokalizačných serverov. Túto stať som pojal ako pokus odvodiť matematickými metódami, že tento postup nemusí viesť za každých okolností k žiadanému výsledku, t.j. zmenšeniu celkovej chyby, ale môže nastať opačný jav, t.j. zvýšenie takejto chyby. Nástrojom k tomu mi bude odvodenie matematickej závislosti.

Obrázok 6.1 zjednodušene zobrazuje situáciu, kde geolokalizačnú oblasť vytvoria svojimi priesečníkmi redundantné odozvy práve z troch serverov. Body A, B, C na tomto obrázku naznačujú pozície týchto serverov. S je pozícia hľadaného bodu. ε_1 a ε_2 sú redundantné časti odchýlok pri odozvách postupne z bodov A a B. Celkovú veľkosť nameranej odozvy som označil ako $R_1 + \varepsilon_1$, resp. $R_2 + \varepsilon_2$, kde R_1 a R_2 sú odozvy bez spomínaných nameraných častí odchýlok. Priesečník takýchto odoziev (t.j. bez redundantných častí) by padol do bodu S, pričom priesečník celkových odoziev som označil ako P (ten bližší z oboch možných). Cieľom tejto state bude hľadanie veľkosti úsečky SP z obrázku a vplyv na celkovú chybu merania.

³² Bližšie viz. kapitola 3



Matematickým zápisom môžeme tento vzťah vyjadriť obecné ako funkčnú závislosť nasledovne:

$$X = f(R_1..R_n, \gamma_1..\gamma_n, \varepsilon_1..\varepsilon_n) \quad (\text{r. 6.9})$$

n - počet geolokačne použitých serverov,

$R_1..R_n$ - odozvy bez redundancií,

$\gamma_1..\gamma_n$ - uhly, ktoré medzi sebou uzavierajú „navzájom susediace“ serverové odozvy smerom ku geolokalizovanému miestu (na obrázku 6.1 zodpovedá uhol ASB uhlu γ_1),

$\varepsilon_1..\varepsilon_n$ - odchýlky prislúchajúce postupne k $R_1..R_n$,

pričom pri danom súbore geolokalizačných serverov budú parametre $R_1..R_n$ a $\gamma_1..\gamma_n$ konštantami, a $\varepsilon_1..\varepsilon_n$ budú premenné, teda funkcia bude dávať celkovú chybu merania v závislosti na veľkosti jednotlivých redundantných častí nameraných odoziev jednotlivých geolokalizačných serverov.

Aby sme získali žiadanú závislosť, budeme zisťovať, kam sa nám dostanú okrajové body tzv. dôveryhodnej oblasti, ktorú nám vytvoria priesečníky dvojíc kružníc (takýmto priesečníkom je napr. bod P z obrázku 6.1).

Bez ujmy na všeobecnosti vypočítam priesečník celkových odchýlok vychádzajúcich z miest serverov A a B a jeho vzdialenosť od cieľa geolokalizácie, bodu

S. Ďalej budem promiscue používať i pojmy „úsečka SP“ pre túto vzdialenosť, „úsečka AS“, resp. „úsečka BS“ pre vzdialenosti serverov A a B od cieľu geolokalizácie, bodu S, a konečne „úsečka AP“, resp. „úsečka BP“ pre vzdialenosti serverov A a B od bližšieho priesečníku kružníc so stredmi v mieste týchto serverov.

Ako vidno z obrázku 6.1, úsečka AS, úsečka PS a úsečka AP tvoria trojuholník. Podľa kosínusovej vety platí nasledujúca rovnosť:

$$|AP|^2 = |AS|^2 + |PS|^2 - 2|AS||PS|\cos(ASP) \quad (\text{r. 6.10})$$

Obdobne i úsečka BS, úsečka PS a úsečka BP tvoria trojuholník a teda platí obdobná rovnosť:

$$|BP|^2 = |BS|^2 + |PS|^2 - 2|BS||PS|\cos(BSP) \quad (\text{r. 6.11})$$

Voľba uhlov ASP a BSP nebola náhodná – dôvod je prostý: vo svojom súčte tvoria uhol ASB, ktorý je pri pevnom rozložení serverov konštantný. V súlade s vyššie zavedeným značením je to uhol γ_1 , v ďalších riadkoch ho budem označovať i ako γ . Ďalej zavediem označenie uhla ASP, a to ako α . Z toho zároveň vyplýva, že uhol BSP bude $\gamma - \alpha$. Celkovo teda môžeme vyššie naznačené rovnosti prepísať nasledovne ($|PS| = x$ ako hľadaná veľkosť):

Z rovnice r. 6.10:

$$(R_1 + \varepsilon_1)^2 = R_1^2 + x^2 - 2R_1x\cos\alpha \quad (\text{r. 6.12})$$

Z rovnice r. 6.11:

$$(R_2 + \varepsilon_2)^2 = R_2^2 + x^2 - 2R_2x\cos(\gamma - \alpha) \quad (\text{r. 6.13})$$

V týchto dvoch rovniciach pri pevnom rozložení serverov máme všetky neznáme konštanty až na hodnoty ε_1 , ε_2 , α a x . Cieľom ďalších krokov bude eliminovať premennú α , aby som mohol vyjadriť veľkosť neznámej x pomocou redundantných častí odchýlok ε_1 a ε_2 .

Z rovnice r. 6.12 dostaneme úpravou:

$$\frac{(R_1 + \varepsilon_1)^2 - R_1^2 - x^2}{2R_1x} = -\cos \alpha \quad (\text{r. 6.14})$$

Z rovnice r. 6.13 dostaneme aplikáciou rovnice pre kosínus rozdielu uhlov³³:

$$\frac{(R_2 + \varepsilon_2)^2 - R_2^2 - x^2}{2R_2x} = -\cos(\gamma - \alpha) = -\cos \gamma \cos \alpha - \sin \gamma \sin \alpha \quad (\text{r. 6.15})$$

Ďalšou úpravou rovnice 6.15 odstaneme:

$$\frac{1}{\sin \gamma} \frac{(R_2 + \varepsilon_2)^2 - R_2^2 - x^2}{2R_2x} + \frac{\cos \gamma}{\sin \gamma} \frac{(R_1 + \varepsilon_1)^2 - R_1^2 - x^2}{2R_1x} = -\sin \alpha \quad (\text{r. 6.16})$$

Po aplikácii rovnosti pre vzťah štvorcov sínusu a kosínusu rovnakého uhla³⁴ na rovnice r. 6.14 a r. 6.16 dostaneme rovnosť, v ktorej sa podarilo eliminovať premennú α . Táto rovnosť vyzerá nasledovne:

$$\begin{aligned} & \left(\frac{1}{\sin \gamma} \frac{(R_2 + \varepsilon_2)^2 - R_2^2 - x^2}{2R_2x} \right)^2 + \left(\frac{\cos \gamma}{\sin \gamma} \frac{(R_1 + \varepsilon_1)^2 - R_1^2 - x^2}{2R_1x} \right)^2 \\ & + 2 \frac{\cos \gamma}{\sin^2 \gamma} \frac{(R_2 + \varepsilon_2)^2 - R_2^2 - x^2}{2R_2x} \frac{(R_1 + \varepsilon_1)^2 - R_1^2 - x^2}{2R_1x} + \left(\frac{(R_1 + \varepsilon_1)^2 - R_1^2 - x^2}{2R_1x} \right)^2 = 1 \end{aligned} \quad (\text{r. 6.17})$$

Po eliminácii α bude účelom ďalších úprav vyjadriť neznámu x pomocou konštánt a premenných ε_1 a ε_2 . Týmto postupom by sme mali byť schopný dostať vyjadrenie funkčnej závislosti f_1 , pre ktorú platí, že:

$$x = f_1(R_1, R_2, \gamma_1 = \gamma, \varepsilon_1, \varepsilon_2) \quad (\text{r. 6.18})$$

Zložením rovníc $f_1 \dots f_n$ ³⁵ budeme schopní získať požadovanú celkovú závislosť. Po tejto odbočke na vysvetlenie smerovania týchto úprav sa vrátim k pôvodnému procesu vyjadrenia neznámej x . Rovnicu r. 6.17 vieme postupne upravovať, aby sme dostali rovnicu r. 6.19:

³³ Pre kosínus rozdielu uhlov platí: $\cos(\alpha - \beta) = \cos \alpha \cos \beta + \sin \alpha \sin \beta$

³⁴ Pre vzťah štvorcov sínusu a kosínusu rovnakého uhla platí: $\sin^2 \alpha + \cos^2 \alpha = 1$

³⁵ Funkcie $f_2 \dots f_n$ dostaneme obdobne ako f_1 .

$$\frac{1}{\sin^2 \gamma} \frac{((R_2 + \varepsilon_2)^2 - R_2^2 - x^2)^2}{4R_2^2 x^2} + \frac{\cos^2 \gamma}{\sin^2 \gamma} \frac{((R_1 + \varepsilon_1)^2 - R_1^2 - x^2)^2}{4R_1^2 x^2} + \frac{((R_1 + \varepsilon_1)^2 - R_1^2 - x^2)^2}{4R_1^2 x^2} =$$

$$1 - 2 \frac{\cos \gamma}{\sin^2 \gamma} \frac{((R_2 + \varepsilon_2)^2 - R_2^2 - x^2)((R_1 + \varepsilon_1)^2 - R_1^2 - x^2)}{4R_1 R_2 x^2} \quad (\text{r. 6.19})$$

Vzhľadom ku skutočnosti, že v rovnici r. 6.19 sa nachádza neznáma i v tetratickom tvare a jej vyjadrenie by bolo zložité, je prínosné si povšimnúť, že práve naopak sa táto neznáma nenachádza v lineárnom ani kubickom tvare. Keďže i očakávaný stupeň tohto polynómu v premennej x je maximálne 4, ponúka sa práve možnosť substitúcie $x^2 = y$, s tým, že dostaneme iba kvadratický polynóm v premennej y :

$$\frac{1}{\sin^2 \gamma} \frac{((R_2 + \varepsilon_2)^2 - R_2^2 - y)^2}{4R_2^2 y} + \frac{\cos^2 \gamma}{\sin^2 \gamma} \frac{((R_1 + \varepsilon_1)^2 - R_1^2 - y)^2}{4R_1^2 y} + \frac{((R_1 + \varepsilon_1)^2 - R_1^2 - y)^2}{4R_1^2 y} =$$

$$1 - 2 \frac{\cos \gamma}{\sin^2 \gamma} \frac{((R_2 + \varepsilon_2)^2 - R_2^2 - y)((R_1 + \varepsilon_1)^2 - R_1^2 - y)}{4R_1 R_2 y} \quad (\text{r. 6.20})$$

Po ďalších úpravách dostávame:

$$\frac{((R_2 + \varepsilon_2)^2 - R_2^2 - y)^2}{4R_2^2 y} + \frac{((R_1 + \varepsilon_1)^2 - R_1^2 - y)^2}{4R_1^2 y} =$$

$$\sin^2 \gamma - 2 \cos \gamma \frac{((R_2 + \varepsilon_2)^2 - R_2^2 - y)((R_1 + \varepsilon_1)^2 - R_1^2 - y)}{4R_1 R_2 y} \quad (\text{r. 6.21})$$

Po odstránení zlomkov a ďalších úpravách rovnice r. 6.21 dostávame postupne rovnice r. 6.22 a r. 6.23:

$$R_1^2 ((R_2 + \varepsilon_2)^2 - R_2^2 - y)^2 + R_2^2 ((R_1 + \varepsilon_1)^2 - R_1^2 - y)^2 =$$

$$4y R_1^2 R_2^2 \sin^2 \gamma - 2R_1 R_2 \cos \gamma ((R_2 + \varepsilon_2)^2 - R_2^2 - y)((R_1 + \varepsilon_1)^2 - R_1^2 - y) \quad (\text{r. 6.22})$$

$$R_1^2 (y^2 + \varepsilon_2^2 (2R_2 + \varepsilon_2)^2 - 2y\varepsilon_2 (2R_2 + \varepsilon_2)) + R_2^2 (y^2 + \varepsilon_1^2 (2R_1 + \varepsilon_1)^2 - 2y\varepsilon_1 (2R_1 + \varepsilon_1)) =$$

$$4y R_1^2 R_2^2 \sin^2 \gamma - 2R_1 R_2 \cos \gamma (y^2 - y\varepsilon_1 (2R_1 + \varepsilon_1) - y\varepsilon_2 (2R_2 + \varepsilon_2) + \varepsilon_1 \varepsilon_2 (2R_1 + \varepsilon_1)(2R_2 + \varepsilon_2)) \quad (\text{r. 6.23})$$

Z rovnice typu vyššie uvedenej rovnice r. 6.23 vieme získať koeficienty kvadratického polynómu, aby sme boli schopní vyjadriť jeho korene. Korene môžeme vyjadriť podľa známeho vzťahu:

$$y = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (\text{r. 6.24})$$

Koeficient a do vyjadrenia rovnice r. 6.24 dostaneme extrakciou kvadratických členov polynómu v x z rovnice r. 6.23 následovne:

$$a = R_1^2 + R_2^2 + 2R_1R_2 \cos \gamma \quad (\text{r. 6.25})$$

Podobne koeficient b dostaneme z lineárnych členov tohto polynóm:

$$b = -2R_1^2 \varepsilon_2 (2R_2 + \varepsilon_2) - 2R_2^2 \varepsilon_1 (2R_1 + \varepsilon_1) - 4R_1^2 R_2^2 \sin^2 \gamma - 2R_1 R_2 \cos \gamma (\varepsilon_1 (2R_1 + \varepsilon_1) + \varepsilon_2 (2R_2 + \varepsilon_2)) \quad (\text{r. 6.26})$$

A konečne to isté platí i pre koeficient c :

$$c = R_1^2 \varepsilon_2^2 (2R_2 + \varepsilon_2)^2 + R_2^2 \varepsilon_1^2 (2R_1 + \varepsilon_1)^2 + 2R_1 R_2 \cos \gamma \varepsilon_1 \varepsilon_2 (2R_1 + \varepsilon_1)(2R_2 + \varepsilon_2) \quad (\text{r. 6.27})$$

Hľadané x potom vyjadríme spätnou substitúciou nasledovne:

$$x = \pm \sqrt{y}, \quad y > 0 \quad (\text{r. 6.28})$$

Ak vychádzame z rovnakého predpokladu, že máme danú pozíciu všetkých geolokačných serverov i cieľového miesta, vypočítaním veľkosti x sme vypočítali vlastne pozíciu jedného okrajového bodu tzv. dôveryhodnej oblasti, a to vyčerpávajúcim spôsobom. Tento poznatok vyplýva z faktu, že sme schopní z tejto chybovej vzdialenosti (na obrázku 6.1 zobrazenú ako veľkosť úsečky SP) dopočítať i uhol, ktorý zvierá s pevne stanovenými úsekmi (na obrázku napr. s úsečkou SA – tá reprezentuje spojnicu medzi cieľovým miestom a prvým geolokalizačným serverom). Tento uhol je uhlom ASP, označeným vyššie tiež ako α . Môžeme ho veľmi jednoducho dopočítať napr. za pomoci rovnice r. 6.10. Výsledný vzťah bude vyzeráť nasledovne:

$$\alpha = \arccos \left(\frac{R_1^2 + x^2 - (R_1 + \varepsilon_1)^2}{2R_1 x} \right) \quad (\text{r. 6.29})$$

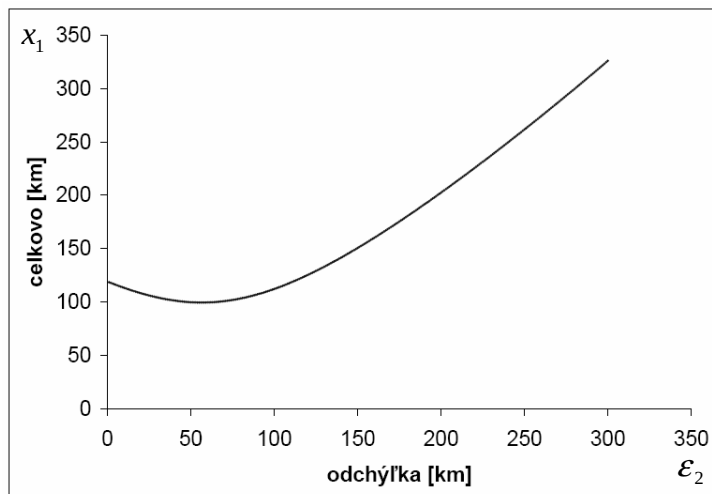
Na základe vyššie uvedených výpočtov som skúmal odchýlky jednotlivých odoziev a ich vzťah na celkovú chybu merania. Pokúsil som sa využiť reálne dáta, obdobne ako

pri implementácii a následných skúmaníach podľa kapitol 2 a 3 (s tým rozdielom, že v tejto kapitole pojednávam o hypotetických možnostiach na základe analytického výpočtu, pričom v spomínaných predchádzajúcich kapitolách sa jedná o čisto empirické výsledky na základe stanovených meraní). Pre dané pevné rozloženie som zvolil geolokačné servery 1-3 podľa prílohy č.3 „Pozície serverov určených pre skúmanie v Európe“, daný geolokalizovaný cieľ som určil server 12 podľa prílohy č.4 „Pozície landmarkov určených pre skúmanie v Európe“.

Pri skúmaní tejto problematiky som si pokladal otázku, ako sa zmena jednej odchýlky prejaví v konečnej, celkovej chybe merania. Bez ujmy na všeobecnosti som zvolil z obrázku 6.1 odchýlku pri meraní zo serveru B, pričom ostatné odchýlky som zafixoval na pevnú hodnotu, ktorú som odvodil od reálnej vzdialenosti serveru a skutočnej pozície geolokalizovaného bodu. (To znamená, že následné merania budú stanovené pre premennú ε_2 , ostatné, t.j. ε_1 a ε_3 budú pre účely tohto merania konštantné). Odvodenie od vzdialenosti pre tieto ostatné odchýlky som stanovil i s prihliadnutím na merania podľa kapitol 4 a 5 a 6.1 na 20%, tzn. $\varepsilon_1 = 0.2|AS|$ a $\varepsilon_2 = 0.2|CS|$. Ďalej som sa pokúsil skúmať taktiež vplyv rozloženia serverov na celkový graf tohto merania.

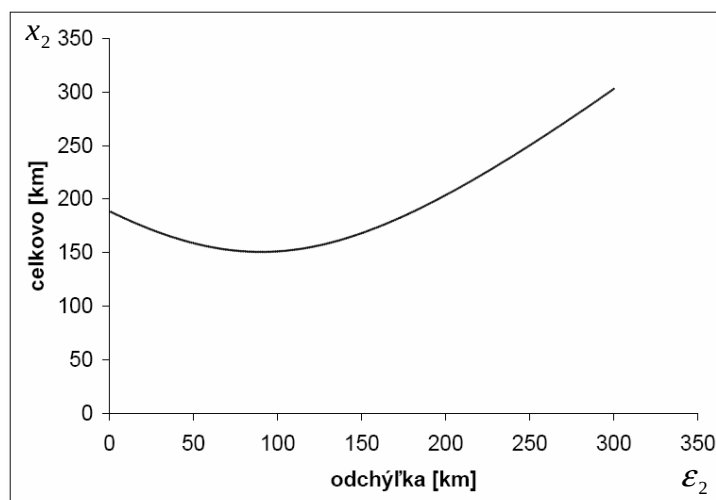
Pre nasledovné skúmanie som pre jednoduchosť simuloval pravidelné rozloženie serverov; toto možno vyjadriť pomocou uhlov v stupňoch, ktoré zvierajú vzájomne susediace úsečky, ako pomer 120:120:120; toto značenie rozloženia budem používať i naďalej. Nižšie v tejto kapitole ukážem vplyv aj iných rozložení. V celom nasledovnom texte sa pokúšam vykonať a popísať závery vo všeobecnom duchu, tak ako je to možné dovodiť z výpočtov uvedených vyššie – vzdialenosti, ktoré používam, sú reálnymi vzdialenosťami (vyššie špecifikované) a mali by len ilustrovať možnosť použitia tohto aparátu.

Na grafoch g. 6.4, g. 6.5 a g. 6.6 je priebeh toho, ako sa mení vzdialenosť okrajového bodu tzv. dôveryhodnej oblasti v závislosti na zmene vyššie zvolenej odchýlky.



(g. 6.4)

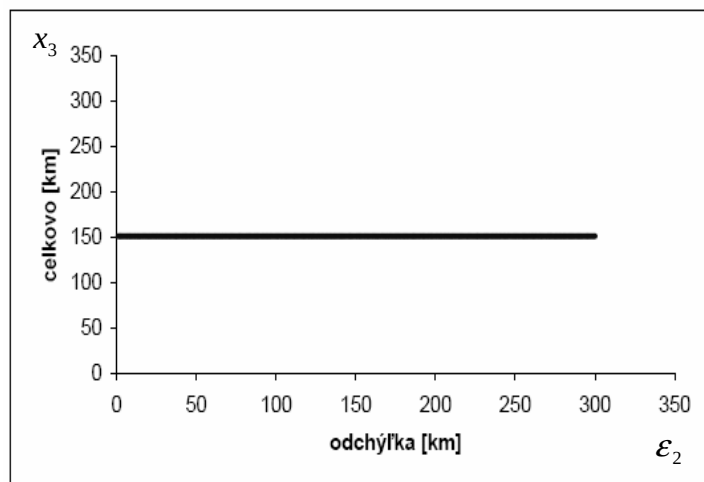
Graf g. 6.4 nám ukazuje práve okrajový bod určený kružnicami so stredmi A a B, a so simulovanými polomerami $R_1 + \varepsilon_1$ a $R_2 + \varepsilon_2$. Priebeh funkcie je hladký s lokálnym minimom u odchýlky 57 km s hodnotou vzdialenosti okrajového bodu 99.4 km. V konkrétnom prípade odlišnom od daného je možné toto minimum dopočítať ad hoc pomocou stanovením hodnoty prvej derivácie výslednej rovnice, ktorá vznikne spojením rovníc r. 6.24 až r. 6.29 rovnej 0.



(g. 6.5)

Graf g. 6.5 nám ukazuje práve okrajový bod určený kružnicami so stredmi B a C, a so simulovanými polomerami $R_2 + \varepsilon_2$ a $R_3 + \varepsilon_3$. Priebeh funkcie je hladký s lokálnym minimom u odchýlky 90 km s hodnotou vzdialenosti okrajového bodu 150.6 km. Opäť platí to isté, ako u prvého grafu, a teda, že v konkrétnom prípade je možné toto minimum

dopočítať ad hoc pomocou stanovením hodnoty prvej derivácie výslednej rovnice, ktorá vznikne spojením rovníc r. 6.24 až r. 6.29 rovnou 0.



(g. 6.6)

Graf g. 6.6 nám ukazuje práve okrajový bod určený kružnicami so stredmi A a C, a so simulovanými polomerami $R_1 + \varepsilon_1$ a $R_3 + \varepsilon_3$. Pribeh funkcie je konštantný, keďže obe odchýlky sú stanovené pevne a hodnota vyplýva z výpočtu rovnice r. 6.29.

Opäť vo všeobecnej rovine, ostatné okrajové body dôveryhodnej oblasti (mj. ktorých pribeh vzdialenosti bol v na konkrétnych údajoch simulovaný v grafoch g. 6.5 a g. 6.6) sa stanovujú obdobným spôsobom, ako som vypočítal prvý bod (bod P z obrázku 6.1). Všetky tieto okrajové nám spolu dotvoria tzv. dôveryhodnú oblasť, spomínanú v kapitole 2. Týmto máme dané všetky dôležité informácie, ktoré sú nutné na to, aby mohol byť vytvorený konštruktívny odhad celkovej odchýlky vzhľadom na dané parametre (či už vo forme konštant – ktoré sú reprezentované vzdialenosťami geolokačných serverov a cieľovým miestom a vzájomnými uhlovými veľkosťami medzi týmito spojnicami, alebo vo forme premenných – tieto sú reprezentované hlavne odchýlkami jednotlivých odoziev).

Aby som zachoval dostatočnú všeobecnosť tejto state (a aby som sa vyhol konkrétnym implementačným teóriám z kapitoly 2 – napr. počítanie odhadu ako ťažiska vzniknutého útvaru), budem počítať celkovú chybu ako súčet vektorov, ktoré vychádzajú z miesta geolokalizácie a končia v jednotlivých okrajových bodov (práve dopočítaných predošlým postupom). Tento postup je výlučne modelový, a nie je ho možné reálne použiť, keďže cieľ geolokalizácie dopredu nepoznáme. Aj napriek tomu, vytýčenému

účelu tejto kapitoly, obecné zachytiť vplyv odchýlky na celkovú chybu, poslúži dobre. Ak použijeme terminológiu obrázka 6.1 a trochu ju zovšeobecníme³⁶, bude celkový chybový vektor vyzeráť asi takto (symbolom $\vec{E}$ značím vektor vychádzajúci zo stredu S vedúci k odhadovanému miestu geolokalizácie) :

$$\vec{E} = \sum_{i=1}^n \vec{SP}_i, \quad (\text{r. 6.30})$$

pričom platí:

$$|\vec{SP}_i| = x_i \text{ a zároveň}$$

α_i reprezentuje uhol medzi $\vec{SP}_i$ a príslušnou spojnicou

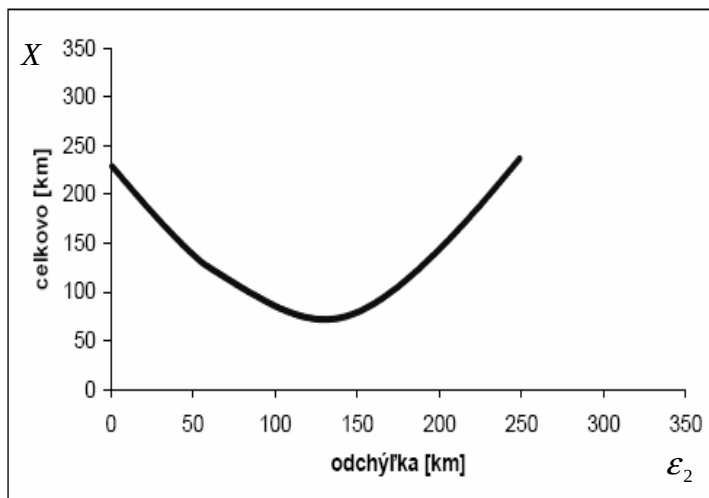
a teda jeho veľkosť obdobne:

$$X = f(R_1..R_n, \gamma_1..\gamma_n, \varepsilon_1..\varepsilon_n) = |\vec{E}| = \left| \sum_{i=1}^n \vec{SP}_i \right| \quad (\text{r. 6.31})$$

X reprezentuje vlastne celkovú chybu geolokalizácie, ktorá je daná zmieňovanou dôveryhodnou oblasťou, ktorá je vyčerpávajúco určená jednotlivými bodmi, ktoré ju ohraničujú (v predchádzajúcom texte zmienené ako $P_1..P_n$).

Nadviazaním na konkrétne dáta, tak ako vyššie, budem pokračovať v štúdiu týchto závislostí. Na nasledujúcom grafe, označenom ako g. 6.7 je naznačený vplyv odchýlky ε_2 na celkovú chybu X . Ostatné veličiny sú dané ako konštantné, ako som už špecifikoval vyššie. Zároveň tento graf predstavuje závislosť odchýlky, kde sa berú do úvahy iba situácie, v ktorých určujú výslednú dôveryhodnú oblasť práve tri servery (tak ako je to vyobrazené na obrázku obr. 6.1).

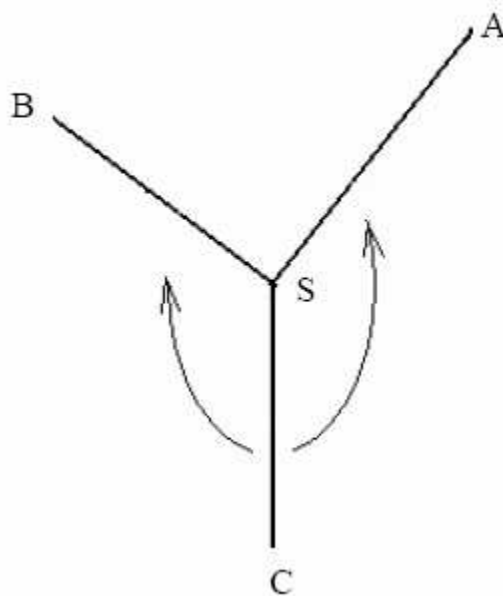
³⁶ Okrajové body budú viaceré a namiesto jediného bodu P zavediem obecné body v $P_1..P_n$, rovnako viaceré budú i vypočítané dĺžky, takže namiesto x budú zavedené obecné $x_1..x_n$, ako aj namiesto uhlu α bude $\alpha_1..\alpha_n$.



(g. 6.7)

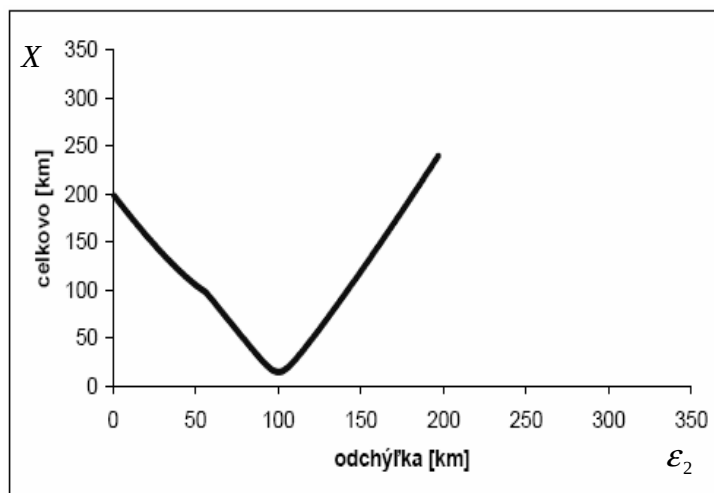
Funkcia, ktorej graf je zobrazený na g. 6.7, je hladkou s jedným (globálnym) minimom nachádzajúcim u odchýlky 130 km, kde hodnota celkovej chyby geolokalizácie je 71.9 km.

Čo sa stane, pokiaľ by sme mali možnosť rozloženie rozostavenia serverov, pričom zachováme vzdialenosti od cieľového miesta? Graf g. 6.7 zobrazuje rozloženie zhruba v pomere 120:120:120. Skúmal som, čo by sa z celkovou chybou stávalo, pokiaľ by sme práve toto rozloženie menilo.



(obr. 6.2)

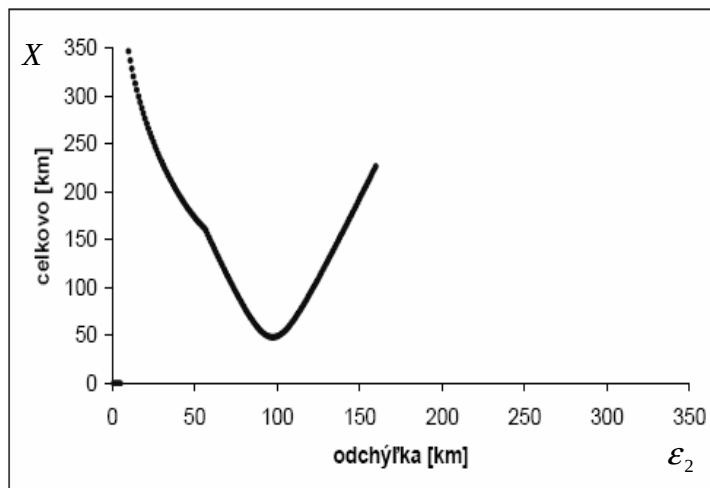
Na obrázku obr. 6.2 je znázornené, akým spôsobom budem skúmať zmeny daného rozloženia. Mojm cieom v ďalšom texte je simulovať a skúmať na podklade vyššie uvedených výpočtov, akým spôsobom sa priebeh tejto odchýlky bude meniť. Zafixujem opäť bez ujmy na všeobecnosti uhol ASB na stanovených 120 stupňov. Ostatné uhly budem pri zachovaní dĺžok zväčšovať eventuálne zmenšovať.



(g. 6.8)

Graf tejto funkcie zobrazuje rozloženie 120:105:135. Funkcia, ktorej graf je zobrazený na g. 6.8, je hladkou s jedným (globálnym) minimom nachádzajúcim u odchýlky 100 km, kde hodnota celkovej chyby geolokalizácie je 14.95 km. Nie je bez zaujímavosti, že toto minimum dosiahnutej chyby je v porovnaní s rovnomerným rozložením značne nižšie. Tu sa práve prejavuje fakt, že ani vzdialenosti od bodu geolokalizácie nie sú rovnomerné.

Pre lepšiu obraz o chovaní pri zmene rozloženia zobrazím ešte jeden graf funkcie:



(g. 6.9)

Graf tejto funkcie zobrazuje rozloženie 120:90:150. Funkcia, ktorej graf je zobrazený na g. 6.9, je hladkou s jedným (globálnym) minimom nachádzajúcim u odchýlky 98 km, kde hodnota celkovej chyby geolokalizácie je 48.05 km.

Pri porovnaní grafov funkcií g. 6.7, g. 6.8 a g. 6.9 (a teda pri porovnávaní rôznych rozložení v poradí od rovnomerného k nerovnomernému) je možné si všimnúť dve skutočnosti, a to, jednak zmeny minim (1.) a jednak zmeny rozsahu záberu (2.). V ďalších riadkoch sa pokúsim o podrobnejší rozbor:

1. Zmena minim:

Ukazovateľ minima vyššie zmienenej funkcie je veľmi dôležitý a dá sa interpretovať tým spôsobom, že nám vlastne stanoví, kedy a za akých podmienok vôbec môžeme dosiahnuť presnosť v určitom rozmedzí. V simulovanom prípade bola odchýlka pri rovnomernom rozložení 71.9 km, následne sa pri posúvaní menila v poradí na 14.95 km a na 48.5 km. Toto kolísanie odchýlky súvisí so skutočnosťou, že vzdialenosti server – geolokalizovaný bod boli stanovené náhodne a tým pádom i od nich odvodené pevné odchýlky majú v sebe nády náhodnosti. Samozrejme potom z toho vyplýva, že najmenšia hodnota minima bude daná pri určitom nepravidelnom rozložení závisiacom na hodnotách týchto náhodných vzdialeností.

2. Zmena rozsahu záberu:

Tri vyššie zmienené funkcie sa vyznačujú ešte jednou zaujímavou vlastnosťou, a to že ich strmosť nie je rovnaká. To nás ale vedie ku skutočnosti, že aj rozsah odchýlky (teda definičný obor týchto funkcií) bude iný. I tento ukazovateľ je nemenej dôležitý ako ten predchádzajúci. Voľne povedané, ide tu o to, do akej hodnoty je možné tolerovať odchýlku na x-ovej ose, aby nám dôveryhodnú oblasť vytváral očakávaný počet

priesečníkov (v našom simulovanom prípade tri). Zníženie počtu týchto priesečníkov vedie štatisticky k oveľa vyšším chybám, takže ide o jav nežiadúci. V simulovanom príklade sme pri rovnomernom rozložení mali maximum v bode 250 km, postupne sa toto maximum znižovalo na 197 km a 160 km; v tomto poslednom – najnerovnomernejšom rozložení bolo stanovené i minimum na 5 km. Je vidieť, že rozsah sa zmenšuje zároveň so znižovaním rovnomernosti a to bez ohľadu na stanovené vzdialenosti.

Z vyššie uvedených úvah si dovoľím pokus o návrh určitých čiastkových úprav algoritmov, ktoré práve využívajú veľkosti odozvy, ktoré by mohli viesť ku zníženiu priemernej odchýlky pri konanej geolokalizácii. Všetky vyššie uvedené úvahy boli robené s vedomosťou o lokalite, kde sa hľadaný bod nachádza. Napriek tomu, pri reálnom použití túto informáciu nemáme. Pokiaľ máme k dispozícii dostatočné množstvo serverov, z ktorých môžeme vysielat' požiadavky a tým ich využiť ku geolokalizácii, môžeme nahrubo odhadnúť približne dôveryhodnú oblasť a v druhej fáze selektujeme iba také servery, ktoré:

1. budú od tejto nahrubo vytvorenej oblasti obdobne ďaleko – tým budeme schopní priblížiť sa k nutnej podmienke pre to, aby sme mohli dosiahnuť malú chybu, a
2. budú od tejto nahrubo vytvorenej oblasti čo najviac rovnomerne rozložené – tak zaistíme čo najväčší rozsah, a tak znížime možnosť vyšších odchýlok, ako by bolo nutné.

Tieto a i podobné návrhy môžu byť iba čiastkové, pri reálnom behu nie je možné ovplyvniť všetky faktory, ktoré na možnú odchýlku môžu aktuálne vplývať. Na druhej strane, mohli by štatisticky tieto nástroje posunúť k lepším výsledkom.

7. Záverečné zhodnotenie a pojednanie o možnosti a pravdepodobnosti presnej lokalizácie

Cieľom práce bolo analyzovať možné riešenia problému geolokalizácie. V kapitole 3 sú popísané prístupné metódy a zároveň možnosti ich implementácie a uplatnenia. Ďalšou časťou bol návrh a implementácia nástroja, ktorý by vychádzal z tohto prehľadu. Ako riešenie tejto úlohy slúži nástroj zostrojený na základe algoritmu CBG, ktorý posielá výslednú lokalitu do prehliadača, ktorý ju zobrazí na príslušnej mape.

Pri implementácii bolo nutné riešiť niekoľko zásadných problémov. Základné myšlienky vychádzajú z literatúry, na ktorú bolo v priebehu odkazované. Detaily tejto implementácie a použitie konkrétnych programátorských techník (napr. používanie metód lineárneho programovania a jeho prispôbenie problému pre ich použitie) zhrnuté hlavne v kapitole 4 už boli mimo týchto textov. Ďalej sa bolo treba vysporiadať s vyššie naznačeným problémom nedostatku aktívnych serverov – brán, ktoré by slúžili ako stanice, ktoré posielajú požiadavky na koncový počítač, ktorý lokalizujeme. Tento nedostatok viedol v algoritme k problému mislokalizácie (podkapitola 4.6), ktorý som sa pokúsil v algoritme vyriešiť pridaním pasívnych landmarkov. To ale nemení nič na skutočnosti, že sa tento nedostatok prejavil v zníženej možnosti použiť tento nástroj na reálne použitie (teda na použitie s dostatočne nízkou pravdepodobnosťou nízkej odchýlky). Myšlienku pasívnych landmarkov som rozšíril i do vylepšenia výsledkov algoritmu Shortest Path (podkapitola 5.3).

Okrem algoritmu CBG v jeho pozmenenej podobe kvôli vyššie zmienenému nedostatku boli naimplementované i ďalšie techniky využívajúce rovnaký princíp, a síce meranie lokality na základe informácií z distribuovaných rozlične rozmiestnených serverov. Za daných podmienok som porovnal výsledky týchto algoritmov a pokúsil som sa zhodnotiť ich efektivitu. Ukázalo sa m. i., že i veľmi jednoduchý a priamočiary algoritmus Shortest Path môže konkurovať v jeho pozmenenej podobe i oveľa viac prepracovanému algoritmu, ktorým je CBG (podkapitola 5.1).

Obecne môžem skonštatovať, že geolokalizáciu je možné uskutočniť (toto je overené vytvorením nástroja, ktorý je súčasťou práce), ale jej výsledky závisia na rôznych faktoroch. Musím zároveň priznať, že sa mi nepodarilo implementovať nástroj, ktorý by podľa očakávaní odrážal geografickú pozíciu serverov s dostatočne nízkou odchýlkou natoľko, aby bolo napríklad možné s istotou tvrdiť, že sa ten ktorý server nachádza v určitej obci. Empirické výsledky mnou implementovaných nástrojov odrážajú strednú odchýlku na úrovni 200-500 km. Podľa autorov v [2] a [13] je možné stlačiť túto celkovú odchýlku pri väčšom počte serverov až na úroveň 80 km (CBG) a 35 km (Octant). Tieto odchýlky sú ale odchýlkami strednými a nič nevypovedajú o tom, aká je najhoršia odchýlka – a táto je práve tou odchýlkou, ktorá nám zabezpečí istotu pre geolokalizáciu. Z empirických výsledkov kapitoly 5 je vidieť, že najhorší prípad môže

byť u niektorých metód mnohonásobne vyšší. Táto skutočnosť je spôsobená hlavne nerovnomerným rozložením serverov a rozdielnymi vzdialenosťami geolokalizačných serverov, ako vyplýva mi. i z výsledkov kapitoly 6. Tieto skutočnosti, podľa môjho názoru vyjadreného v spomínanej kapitole, je možné do istej miery eliminovať.

Návrh univerzálneho nástroja odrážajúceho dostupné informácie, akým je Octant sa zdá byť očakávanou ideou, ale musím poznamenať, že za zlepšenie výslednej odchýlky o 45 km tým, že tento nástroj orezáva neprípustné oblasti (a tým zahŕňa množstvo získaných informácií) sa platí premrštená cena vyplývajúca z výrazne ťažkopádnejšieho mechanizmu získavania výsledného odhadu.

8. Literatura

- [1] E. Katz-Basset, J. John, A. Krishnamurthy, D. Wetherall, T. Anderson and Y. Chawathe, Towards IP Geolocation Using Delay and Topology Measurement, 2006
- [2] B. Gueye, M. Crovella, A. Ziviani, S. Fdida: Constraint-based Geolocation of Internet Hosts
- [3] P. Enge and P. Misra, The Global Position System. *Proc. of the IEEE*, January 1999.
- [4] Andy Harter and Andy Hopper, A Distributed Location System, for the Active Office. *IEEE Network* Vol.8 No.1, January 1994
(<http://www.cl.cam.ac.uk/research/dtg/publications/public/files/tr.94.1.pdf>)
- [5] A. Harter, A. Hopper, P. Steggles, A. Ward, and P. Webster, The Anatomy of a Context-Aware Application, *ACM Mobicom*, August 1999
(<http://citeseer.ist.psu.edu/cache/papers/cs/10242/ftp:zSzzSzftp.uk.research.att.comzSzpubzSzdocszSzattzSztr.1999.7.pdf/harter99anatomy.pdf>)
- [6] P. Bahl and V. N. Padmanabhan, RADAR: An In-Building RF-Based User Location and Tracking System. *IEEE Infocom*, March 2000
(<http://citeseer.ist.psu.edu/cache/papers/cs/11606/http:zSzzSzwww.research.microsoft.comzSz~padmanabzSzpaperszSzinfocom2000.pdf/bahl00radar.pdf>)
- [7] NETGEO <http://www.isoc.org/inet2000/cdproceedings/posters/226/>
- [8] A. LaMarca, Y. Chawathe, S. Consolvo, J. Hightower, I. Smith, J. Scott, T. Sohn, J. Howard, J. Hughes, F. Potter, J. Tabert, P. Powledge, G. Borriello, and B. Schilit. Place Lab: Device positioning using radio beacons in the wild. In *Proc. of Pervasive*, Munich, Germany, 2005 (<http://www.placelab.org/publications/pubs/pervasive-placelab-2005-final.pdf>)
- [9] N. Priyantha, A. Chakraborty, and H. Balakrishnan. The Cricket location-support system. In *Proceedings of MOBICOM*, Boston, MA, USA, 2000.
- [10] NIMA Technical Report TR8350.2, Department of Defense World Geodetic System 1984, Its Definition and Relationships with Local Geodetic Systems, Part 3 (<http://earth-info.nga.mil/GandG/publications/tr8350.2/wgs84fin.pdf>)
- [11] T. Vincenty: Direct and inverse solutions of geodetics on the ellipsoid with application of nested equations. In *Survey Review No. 176*, April 1975 (http://www.ngs.noaa.gov/PUBS_LIB/inverse.pdf)

- [12] R. Govindan and H. Tangmunarunkit. Heuristics for Internet map discovery. In Proc IEEE Infocom, 2000.
- [13] B. Wong, I. Stoyanov, E. G. Sirer: Geolocalization on the Internet through Constraint Satisfaction

9. Zoznam použitých skratiek

SOI	Speed of Internet - jedna z geolokalizačných metód založených na meraní odozvy zo serverov
CBG	Contraint Based Geolocation - jedna z geolokalizačných metód založených na meraní odozvy zo serverov
ShP	Shortest Path - jedna z geolokalizačných metód založených na meraní odozvy zo serverov
RTT	Round-trip time – čas odozvy (čas, ktorý je nutný pre prechod paketu od vysielateľa k prijímateľovi)

10. Zoznam príloh (v elektronickej forme)

1. NA-servers: zoznam aktívnych serverov v Severnej Amerike
2. NA-landmarks: zoznam pasívnych landmarkov v Severnej Amerike
3. EU-servers: zoznam aktívnych serverov v Európe
4. EU-landmarks: zoznam pasívnych landmarkov v Európe
5. Geoloc-tools: implementácia troch vybraných metód v C
6. Geoloc-documentation: dokumentácia k implementácii pod bodom 5
7. Measured_data: namerané hodnoty z 4.12.2007 10-18 hod. pre kapitolu 5