VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

ODHAD GEOGRAFICKÉ POLOHY STANIC V INTERNETU

LOCATION ESTIMATION OF INTERNET NODES

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE Bc. LADISLAV NĚMEČEK
AUTHOR

VEDOUCÍ PRÁCE doc. Ing. DAN KOMOSNÝ, Ph.D.
SUPERVISOR

BRNO 2014

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky a komunikačních technologií

Ústav telekomunikací

Diplomová práce

magisterský navazující studijní obor
Telekomunikační a informační technika

Student:	Bc. Ladislav Němeček	***ID:***	110612
Ročník:	2	***Akademický rok:***	2013/2014

NÁZEV TÉMATU:

Odhad geografické polohy stanic v Internetu

POKYNY PRO VYPRACOVÁNÍ:

Nastudujte a analyzujte dílčí faktory způsobující zpoždění při přenosu dat mezi stanicemi v IP sítích. Dále se seznamte s experimentální sítí PlanetLab – http://www.planet-lab.org/. Naprogramujte lokační algoritmus CBG (Constraint-Based Geolocalization). Ověřte správnou činnost realizovaného algoritmu pomocí stanic sítě PlanetLab.

DOPORUČENÁ LITERATURA:

[1] GUEYE, B. ZIVIANI, A., CROVELLA, M., FDIDA, S. Constraint-Based Geolocation of Internet Hosts. ACM Internet Measurement Conference 2004. ACM SIGCOMM, 2004.
[2] PERCACCI, R., VESPIGNANI, A. Scale-free behavior of the Internet global performance. The Europen Physical Journal B – Condensed Matter and Complex Systems. Springer-Verlag, 2003.
[3] PlanetLab Consortium. PlanetLab: An open platform for developing, deploying, and accessing planetary-scale services. URL: <http://www.planet-lab.org> [cit. 10. 10. 2011].

Termín zadání:	10.2.2014	***Termín odevzdání:***	28.5.2014

Vedoucí práce: doc. Ing. Dan Komosný, Ph.D.
Konzultanti diplomové práce:

doc. Ing. Jiří Mišurec, CSc.
Předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Tato diplomová práce se zabývá problematikou IP geolokace. Jsou rozebrány pasivní i aktivní geolokační techniky. Více se zaměřuje na aktivní vyhledávací metody. Ty jsou založeny na měření zpoždění v síti, a proto je zde vysvětleno, jak zpoždění vzniká a jakým způsobem ho lze měřit. Dále je část věnovaná vzniku a struktuře síťě PlanetLab, neboť její uzly jsou využívány pro geolokaci.

Je vysvětleno, jakým způsobem byl vytvořen vhodný soubor referenčních bodů, sloužících pro měření zpoždění v síti. Hlavním tématem je implementace metody Constraint-Based Geolocation v jazyce Java. Jsou popsány jednotlivé části a funkce vytvořeného programu, způsob výběru kalibračních přímek bestline, samotný algoritmus vyhledávání cílové polohy a možnosti zobrazení výsledků. Dále byly vytvořeny skripty zajišťující měření zpoždění a komunikaci se servery sítě Planetlab. V práci je také část věnovaná úpravě výsledného programu pro geolokační server.

V neposlední řadě je vytvořena množina testovacích cílů, za pomoci které je ověřena správná funkce a přesnost metody. Výsledky měření jsou prezentovány za pomoci statistik a grafů a jsou srovnávány s výsledky uváděnými v literatuře a také s dalšími metodami prostřednictvím geolokačního serveru.

KLÍČOVÁ SLOVA

IP geolokace, RTT, PlanetLab, CBG

ABSTRACT

This diploma thesis deals with subject of IP geolocation methods. It describes the methods of passive and active geolocation and it is more focused on active searching methods which use measuring the latency in the network. The factors causing delays in data transfer are discussed first, followed by discussion of the issue of measuring these delays. After that it is given a brief description of PlanetLab experimental network, which nodes were used for delay measurement.

The Dataset of PlanetLab network landmarks was created and it is used for requirement of geolocation method. The main topic is practical implementation of the method Constraint-Based Geolocation in Java programming language. The second part of the method package are bash scripts, that provide measuring network latency and communication with PlanetLab nodes.

The implemented method is tested using the created dataset of targets. Last but not least, the measurement results of CBG algorithm are compared to other papers and methods on geolocation server.

KEYWORDS

IP geolocation, RTT, PlanetLab, CBG

NĚMEČEK, Ladislav *Odhad geografické polohy stanic v Internetu*: diplomová práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2014. 59 s. Vedoucí práce doc. Ing. Dan Komosný, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Odhad geografické polohy stanic v Internetu“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno ....

(podpis autora)

PODĚKOVÁNÍ

Děkuji vedoucímu práce doc. Ing. Danu Komosnému, Ph.D. za velmi užitečnou metodickou pomoc a cenné rady při zpracování diplomové práce.

Brno ....

(podpis autora)

Faculty of Electrical Engineering
and Communication
Brno University of Technology
Technicka 12, CZ-61600 Brno
Czech Republic
http://www.six.feec.vutbr.cz

PODĚKOVÁNÍ

Výzkum popsaný v této diplomové práci byl realizován v laboratořích podpořených z projektu SIX; registrační číslo CZ.1.05/2.1.00/03.0072, operační program Výzkum a vývoj pro inovace.

Brno ....

(podpis autora)

MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY

EVROPSKÁ UNIE
EVROPSKÝ FOND PRO REGIONÁLNÍ ROZVOJ
INVESTICE DO VAŠÍ BUDOUCNOSTI

2007–13
OP Výzkum a vývoj pro inovace

OBSAH

Úvod **11**

1 Zpoždění vznikající při komunikaci v síti **12**
1.1 Zpoždění v síti a jeho vznik 12
1.1.1 Zpoždění vznikající v koncových zařízeních 13
1.1.2 Zpoždění na přenosových linkách 13
1.1.3 Zpoždění na mezilehlých zařízeních 14
1.2 Měření zpoždění v síti 14

2 Geolokace **16**
2.1 Pasivní metody 16
2.2 Aktivní metody 17
2.2.1 ShortestPing 18
2.2.2 GeoPing 18
2.2.3 Speed of Internet (*SOI*) 19
2.2.4 GeoWeight 19
2.2.5 Octant 20

3 Constraint-Based Geolocation (CBG) **21**
3.1 Převod zpoždění na hranice vzdálenosti 21
3.2 Automatická kalibrace 22
3.3 Odhad polohy hledaného cíle 23
3.4 Přesnost a omezení metody CBG 24

4 PlanetLab **25**

5 Implementace algoritmu CBG **26**
5.1 Využití serverů sítě PlanetLab 26
5.2 Formát vstupních dat a souborů 28
5.3 Proces odhadování pozice cíle 29
5.4 Výpočet kalibrační křivky 31
5.5 Hlavní část algorimu CBG 33
5.6 Možnosti zobrazení výsledků geolokace 35
5.6.1 Seznam výsledků 36
5.6.2 Grafická prezentace 37
5.6.3 Úprava programu pro geolokační server 38

6 Výsledky měření a porovnání metod **40**
6.1 Měření pomocí metody CBG 40
6.2 Srovnání CBG s ostatními metodami 43

7 Závěr **44**

Literatura **46**

Seznam symbolů, veličin a zkratek **49**

Seznam příloh **50**

A Seznam použitých referenčních bodů **51**

B Seznam cílů pro geolokaci **52**

C Textový výstup programu **54**

D Výsledky měření **57**

E Obsah přiloženého CD **59**

SEZNAM OBRÁZKŮ

1.1 Zdroje zpoždění v síti. [3] 12
1.2 Vliv směrovací politiky na délku přenosové cesty. 13
2.1 Princip metody GeoPing. 19
2.2 Princip metody GeoWeight. 20
2.3 Princip metody Octant. 20
3.1 Princip metody Constraint-Based Geolocation. 21
3.2 Graf závislostí zpoždění na vzdálenosti.[18] 22
3.3 Důsledky nevhodného výpočtu hranic vzdáleností: a) nadhodnocení hranic, b) podhodnocení hranic, c) kombinace předchozích situací. . . 24
4.1 Uzly sítě PlanetLab. [20] 25
5.1 Poloha vybraných referenčních bodů. 27
5.2 Vývojový diagram programu `cbgWorkstation.jar`. 31
5.3 Vývojový diagram algoritmu pro výběr přímky bestline. 32
5.4 Kolekce dat pomoci skriptu `komplet.sh`. 34
5.5 Okno zobrazující výsledky vyhledávání na mapě. 38
6.1 Úspěšnost geolokace cílů (odhad polohy do 400 km). 41
6.2 Chyba odhadu polohy cíle. 41
6.3 Kumulativní distribuční funkce metody CBG. 42
6.4 Kumulativní distribuční všech metod. 43

ÚVOD

Celosvětová síť Internet je prostředkem k šíření informací a ke komunikaci. Je ale také zdrojem zábavy nebo slouží jako nástroj pro výzkum či práci. Potřeby znát polohu zařízení v síti využívá dnes stále více aplikací. Tyto informace jsou zpracovány k nejrůznějším účelům. Ať už se jedná o cílenou reklamu, zpravodajství nebo jiné služby, je využívána znalost místa, na kterém se náš počítač nachází. Geolokace obecně znamená nalezení objektu v určitém prostoru. Pokud se omezíme na IP geolokaci, objekt představuje síťové zařízení a prostorem je myšlena síť Internet.

Mezi nejčastější metody vyhledávání patří jednoduché techniky založené na porovnávání dat s databázemi společností zabývajících se jejich tvorbou, či jejich správou. Nevýhodou je závislost na aktuálních údajích uvedených právě v těchto databázích. Druhou možností je vyhledávání stanic za pomoci metod pracujících se znalostí naměřeného zpoždění v síti.

Tato práce se věnuje implementaci jedné z aktivních geolokačních metod, konkrétně se jedná o *Constraint-Based Geolocation*. Z tohoto důvodu je vhodné popsat, co je to zpoždění, jak a kde vzniká a jakým způsobem ho lze meřit. Témata související se zpožděním v síti popisuje kapitola 1.

Kapitola 2 se zaměřuje na IP geolokaci a její rozdělení na pasivní a aktivní metody. Jsou popsány obě skupiny a z aktivních geolokačních technik jsou vybráni zástupci pro podrobnější popis.

V kapitole 3 je pak více do hloubky rozebrána metoda *Constraint-Based Geolocation*. Objasňuje automatickou kalibraci měřících bodů, převod zpoždění na vzdálenost a samozřejmě princip fungování algoritmu vyhledávání polohy stanice.

Kapitola 4 se zabývá historií a strukturou experimentální sítě PlanetLab, neboť její uzly jsou použity pro potřeby navržené metody CBG.

Hlavní praktická část je popsána v kapitole 5. Jedná se o konkrétní implementaci metody *Constraint-Based Geolocation*. Je vysvětleno, jak byl vytvořen soubor serverů zmíněné sítě PlanetLab a jakým způsobem jsou využity. Kapitola dále rozebírá postup při zpracování vstupních dat a jednotlivé části programu. Nechybí popis vytvořených skriptů zajišťujících měření zpoždění a komunikaci s referenčními body. Závěrečná část kapitoly je věnována možnostem zobrazení výsledků a hlavně pak úpravám programu pro potřeby geolokačního serveru umístěného na Ústavu telekomunikací Fakulty elektrotechniky a komunikačních technologií Vysokého učení technického v Brně.

Poslední kapitola 6 hodnotí výsledky vytvořené geolokační metody. Jsou prezentovány statistiky vybraných měření. Ty jsou porovnány navzájem a také s výsledky uváděnými v literatuře [18] a s dalšími metodami prostřednictvím výše zmiňovaného serveru.

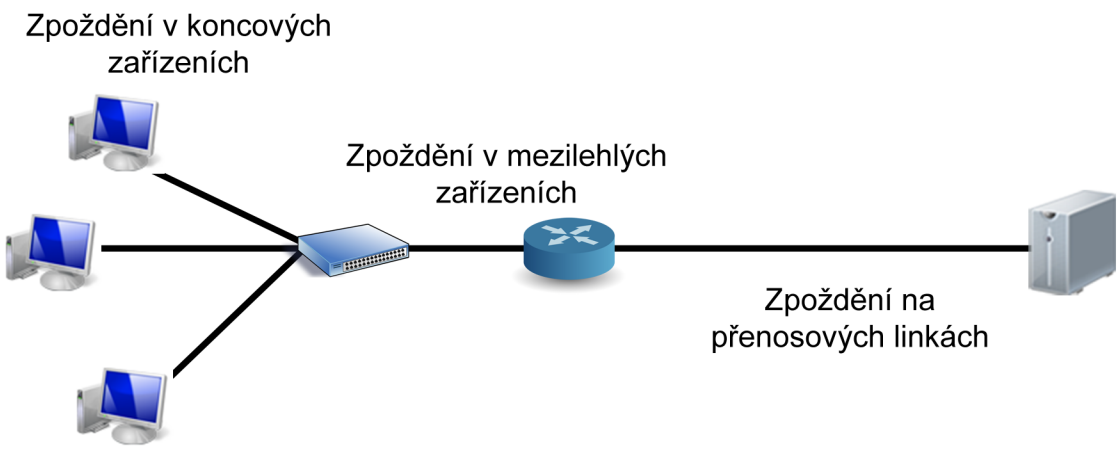
1 ZPOŽDĚNÍ VZNIKAJÍCÍ PŘI KOMUNIKACI V SÍTI

1.1 Zpoždění v síti a jeho vznik

Pojem *zpoždění v síti* nebo také *latence* definuje čas, který uplyne od odeslání zprávy zdrojovým uzlem až po přijetí této zprávy koncovým uzlem. Je tedy dáno součtem všech zpoždění vznikajících na jednotlivých částech přenosové cesty. Hlavní vliv na výslednou velikost má vzdálenost komunikujících uzlů, typ přenosové cesty a v neposlední řadě také doba potřebná ke směrování datových jednotek od zdroje k cíli. Na přenosové cestě má hodnota latence podobnou velikost i při opakovaném měření.[1]

Dle [2] lze zpoždění rozdělit na **stochastickou** a **deterministickou** část. Hodnotu deterministického zpoždění lze vypočítat nebo odhadnout z teoretických poznatků. Jedná se o minimální čas potřebný pro přenos datové jednotky mezi komunikujícími stranami. Zbývající část z celkové hodnoty latence představuje stochastické zpoždění. To je závislé na aktuálním stavu přenosové cesty a dalších náhodných procesech. Jeho velikost proto není konstantní a nelze ji předem určit.[1]

Druhou možností, jak rozdělit zpoždění, je dělení podle části přenosové cesty, na které vzniklo. Jedná se o zpoždění **v koncových zařízeních**, **na přenosových linkách** a **na mezilehlých zařízeních**.[3]



Obr. 1.1: Zdroje zpoždění v síti. [3]

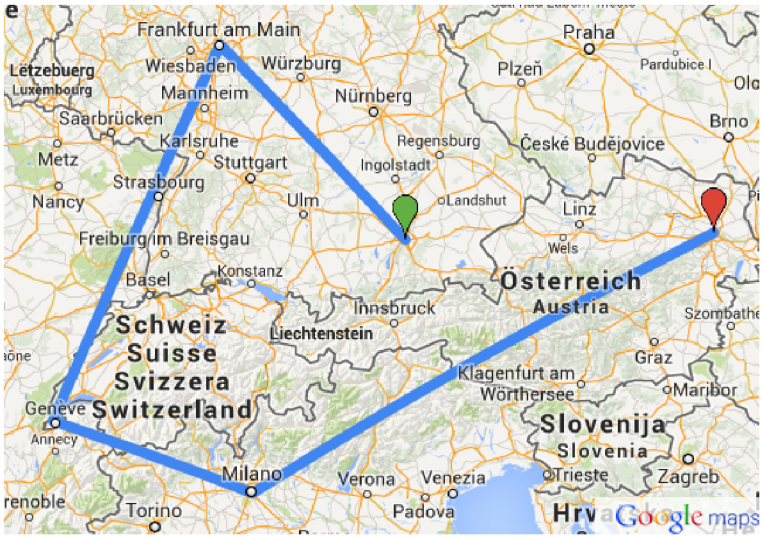
1.1.1 Zpoždění vznikající v koncových zařízeních

Za koncová zařízení jsou považovány zdrojový a cílový uzel, např. počítače. Zdrojová stanice připraví data pro přenos po transportním kanálu a odešle je příjemci. Data prochází postupně vrstvami ISO/OSI modelu a jsou připravovány na odeslání. Tento proces zahrnuje zapouzdřování dat do paketů, jejich následné opatření zdrojovou a cílovou adresou, vytvoření rámců a odeslání po bitech. Cílová stanice provádí tyto operace v opačném pořadí. Do této skupiny patří *paketizační* a *depaketizační* zpoždění způsobené vkládáním datových bloků do paketů a naopak. Další zpoždění vznikne metodami řízení přístupu ke sdílenému médiu.[4] Latence vznikající v koncových zařízeních představuje převážně stochastickou hodnotu, neboť nemá vždy stejnou velikost.

1.1.2 Zpoždění na přenosových linkách

Velikost tohoto zpoždění je závislá především na délce vedení a také na rychlosti šíření signálu v něm.

Skutečná délka přenosové cesty je často delší než je vzdálenost mezi zdrojem a cílem vzdušnou čarou. Trasy nevedou nejkratší přímou cestou, ale kabely jsou pokládány na vhodných místech podél silnic, železnic nebo vedení elektrické energie. Délka trasy je také dána směrovací politikou mezi jednotlivými autonomními systémy (AS) a sítěmi. Směrování probíhá na základě daných pravidel, jedním z nich je např. vhodné rozložení zátěže mezi směrovače. To může zapříčinit, že výsledná cesta nemusí být tou nejkratší možnou (viz Obrázek 1.2).[1]



Obr. 1.2: Vliv směrovací politiky na délku přenosové cesty.

Druhým ovlivňujícím faktorem je rychlost šíření signálu v přenosovém médiu. Ta závisí na použitém materiálu vedení. Páteřní linky a dálkové spoje jsou ve většině případů tvořený vedením z optických kabelů. Rychlost šíření signálu v optickém vlákně je rovna $\frac{2}{3}$ c[1]. V případě vyjádření latence za pomoci obousměrného zpoždění a parametru RTT (viz kapitola 1.2) tato hodnota odpovídá dobře zapamatovatelnému poměru 100 km/ms.[5]

Ze známých hodnot délky vedení a rychlosti šíření signálu v něm lze tedy vypočítat velikost zpoždění na přenosové lince. Tuto hodnotu můžeme považovat za deterministické zpoždění.[2]

1.1.3 Zpoždění na mezilehlých zařízeních

Těmito zařízeními se rozumí všechny aktivní prvky na lince mezi zdrojovým a cílovým uzlem. Zabezpečují funkce na různých vrstvách modelu ISO/OSI, zejména pak přepínání (*linková vrstva*) a směrování (*síťová vrstva*) datových jednotek. Na nejnižší *fyzické vrstvě* je to funkce obnovy a zesílení signálu.

Celková latence každého aktivního prvku je složena ze třech dílčích částí. První z nich je zpoždění ve vstupní frontě. Tato část představuje dobu, kterou rámec čeká ve vstupní vyrovnávací paměti zařízení, než jsou načteny všechny jeho bity. Dále je to doba potřebná pro zpracování informace, neboli doba, za kterou jsou data přemístěny ze vstupu na výstup. Poslední částí je zpoždění ve výstupních frontách.[3]

Latence způsobená přepínači se pohybuje kolem 1-10 µs, směrovače pak vykazují několikanásobně vyšší hodnoty (10-100 µs). Zpoždění na fyzické vrstvě způsobené opakovači a huby je oproti dříve uvedeným zanedbatelné. Při nadměrném zatížení těchto prvků se však mohou hodnoty několikanásobně zvýšit.[1]

1.2 Měření zpoždění v síti

Možné způsoby měření latence v IP sítích definuje RFC 1242[6]. Zpoždění je definováno jako čas, který uplyne od odeslání zprávy po její přijetí. To je označováno jako jednosměrné zpoždění. Výsledkem je časová prodleva pro jeden směr toku dat. Měření tohoto parametru však není v praxi vhodné. Metoda vyžaduje velmi dobrou časovou synchronizaci koncových stanic. Dále je třeba zajistit možnost měření zpoždění v obou směrech, což zpravidla vyžaduje přístup ke vzdálené stanici.[7]

Z těchto důvodů se proto používá metoda měření obousměrného zpoždění představovaného parametrem RTT (*Round-Trip time*). RTT je měřen prostřednictvím jedné stanice a odpovídá času, za který je zpráva odeslána k cíli a doručena zpět.

[1] rychlost světla ve vakuu c = 299 792 458 m/s

Obousměrné zpoždění by mělo být přibližně dvojnásobné než jednocestné. Z výsledné celkové hodnoty však nelze přesně určit dobu pro jednotlivé směry komunikace. Z důvodů aplikace směrovacích politik (viz kapitola 1.1.2) není totiž zaručeno, že cesta od zdroje k cíli odpovídá cestě zpět.[3]

Ping

Nejrozšířenějším způsobem pro měření RTT je aplikace *ping*, využívající protokol ICMP [8], jež je standardní součástí protokolové sady TCP/IP.

Internet Control Message Protocol je služební protokol, což znamená, že nepřenáší žádná uživatelská data. Slouží k signalizaci mimořádných událostí v síti a umožňuje stanicím a směrovačům ohlašovat chyby a vyměňovat si omezené řídicí a stavové informace. K základním funkcím protokolu ICMP patří testování dostupnosti a stavu cílového uzlu sítě nebo např. řízení zahlcení sítě. V některých zařízeních jsou však tyto zprávy z bezpečnostních důvodů ignorovány, respektive zahazovány, což může právě při měření zpoždění vést k zásadním problémům.[9] ICMP zprávy jsou přenášeny v datové části IP datagramu. Vybrané druhy ICMP zpráv shrnuje tabulka 1.1

Tab. 1.1: Vybrané typy ICMP zpráv. [8]

Typ	Zpráva
0	odpověď na výzvu (*Echo Reply*)
3	nedoručitelný IP datagram (*Destination Unreachable*)
8	žádost o odpověď (*Echo*)
11	čas vypršel (*Time Exceeded*)

Ping, neboli *Packet InterNet Grouper*, využívá ICMP zprávy typu *Echo request*, na které cílová stanice musí odpovědět příslušnou zprávou typu *Echo reply*. Z měření času potřebného k doručení odpovědi jsou následně generovány výsledky a statistiky. Nástroj, který je součástí OS Windows i Unix, pracuje se standardním vstupem a výstupem. Program je spouštěn v terminálu (Unix) popřípadě příkazovém řádku (Windows) příkazem `ping`. Adresa cílového hostitele je udána jako vstupní parametr. Další parametry, které konfigurují možnosti měření a zobrazení, jsou závislé na OS a jejich význam lze najít v nápovědě programu.

2 GEOLOKACE

Geolokace obecně představuje výraz pro nalezení geografické polohy určitého objektu. Může se jednat např. o počítač, mobilní telefon, GPS přijímač nebo i o osobu. Přesnost, s jakou může být pozice určena, závisí na technologické výbavě hledaných a vyhledávajících zařízení i na metodě vyhledávání. Například v mobilním telefonu lze využít vestavěný GPS přijímač a systém satelitních družic. Pokud není vybaven touto technologií, je možné ho lokalizovat za pomoci BTS nebo okolních sítí WiFi.

V případě pojmu IP Geolokace je cílem hledání stanice nebo jiný aktivní prvek v síti. Tato metoda využívá protokol IP a jako identifikátor hledaného objektu slouží IP adresa. Ve skutečnosti však neexistuje žádný vztah mezi hledanou geografickou pozicí a IP adresou zařízení. Lokalizace pouze na základě znalosti IP adresy stanice tedy není jednoduchá.[10]

S rostoucím počtem stanic a s rychlým rozvojem sítě Internet jsou metody pro vyhledávání pozice využívány v mnoha aplikacích a oborech. Internet je velice příhodné medium pro šíření reklamy a je velmi žádoucí zaměřit ji na správnou cílovou skupinu. S využitím znalosti polohy uživatele je možné mu nabízet produkty a služby z jeho okolí. Stejně tak lze mimo produkty nabízet informace v podobě lokálního zpravodajství či počasí. Svou úlohu může geolokace hrát i v oblasti bezpečnosti, např. při omezování bankovních transakcí platební kartou. Znalost polohy může také ovlivňovat zobrazení webových stránek v různých světových jazycích. V neposlední řadě to mohou být i zábavné aplikace, jako např. *Foursquare*[1] nebo různé sociální sítě, které využívají polohu uživatele.[11]

Metody pro určení polohy stanice v síti lze rozdělit do dvou kategorií, a to na **pasivní** a **aktivní metody**.

2.1 Pasivní metody

Principem pasivních metod je získávání informací o hledaném cíli v databázích. Pokud požadujeme přesné a kvalitní informace, je třeba využívat komerční databáze, které jsou pravidelně aktualizovány. Jedním z klíčů pro vyhledávání může být IP adresa stanice nebo doménové jméno hledaného cíle, popřípadě brány, přes kterou je připojen. Údaj identifikující stanici je porovnáván se záznamy v databází organizace IANA[2], která je zodpovědná za přidělování veřejných IP adres.

[1] http://foursquare.com
[2] http://www.iana.org

IANA je v jednotlivých částech světa zastoupena regionálními registry (RIR) spravujícími příslušné databáze adres:

- AfriNIC pro Afriku,
- APNIC pro Asii a Pacifik,
- ARIN pro Severní Ameriku,
- LACNIC pro Latinskou Ameriku a Karibik,
- RIPE NCC pro Evropu.

Výsledkem hledání je záznam obsahující informace o organizaci využívající zadanou doménu či adresu.

Mimo komerční databáze existují i volně přístupné. Mezi nejznámější patří např. databáze *WhoIs* [12] nebo *GeoIP* od firmy MaxMind[3].

Hlavní nevýhodou pasivních metod vyhledávání je závislost na důvěryhodných záznamech v databázích. Pro správnou lokalizaci stanice je třeba tyto rozsáhlé databáze pravidelně manuálně aktualizovat, což je náročné časově i finančně.

2.2 Aktivní metody

Tyto způsoby zjišťování polohy oproti již popsaným pasivním metodám pracují na principu zpracování parametrů změřených v síti. Neporovnávají tedy vstupní data s žádnou databází, ale aktivně provádí měření, ze kterých následně pomocí algoritmů vypočítají odhadovanou polohu cíle.

Podle principu se mohou aktivní metody dále rozlišovat na algoritmy založené na **měření zpoždění v síti** (viz kapitola 1.2) nebo na **znalosti topologie sítě**. Dále pak existují metody kombinující obě předchozí možnosti a jejich výsledky dosahují ze všech tří metod nejlepších hodnot.[13] Více o využití topologie sítě k určování polohy lze nalézt např. v [10].

Následující kapitola se bude dále věnovat popisu geolokačních metod pracujících na základě měření zpoždění v síti. Pro tyto potřeby jsou využívány tzv. **referenční body RB** (angl. *Landmark*) a **sondy** (angl. *Probe*). Jedná se o stanice v síti, ve většině případů jde o servery, u nichž je známa jejich poloha. Prostřednictvím nich je pak měřeno zpoždění RTT k ostatním referenčním bodům a k cíli. Na základě takto získaných údajů je odhadnuta poloha hledaného cíle.

Výhodou všech těchto metod je to, že pracují zcela automaticky bez nutnosti ručního dohledávání údajů, aktualizací jákekoliv databáze, či jiných výpočtů. Je vyžadováno pouze počáteční vytvoření souboru referenčních bodů s jejich polohou.[13]

Mezi nevýhody lze počítat jistá omezení. Díky omezenému množství IPv4 adres (2^{32}) je v současné době mnoho stanic připojeno do sítě Internet přes bránu díky

[3] http://www.maxmind.com/en/geolocation_landing

technologii NAT (*Network Adress Translation*). Ty pak navenek vystupují pod jednou společnou veřejnou IP adresou. Ve většině případů poloha brány a stanice není totožná a do odhadu je tak zanášena chyba, která se může pohybovat v řadech stovek km. Stejný problém vzniká i v případě připojení stanice přes vzdálený proxy server nebo pomocí VPN.[14][15]

2.2.1 ShortestPing

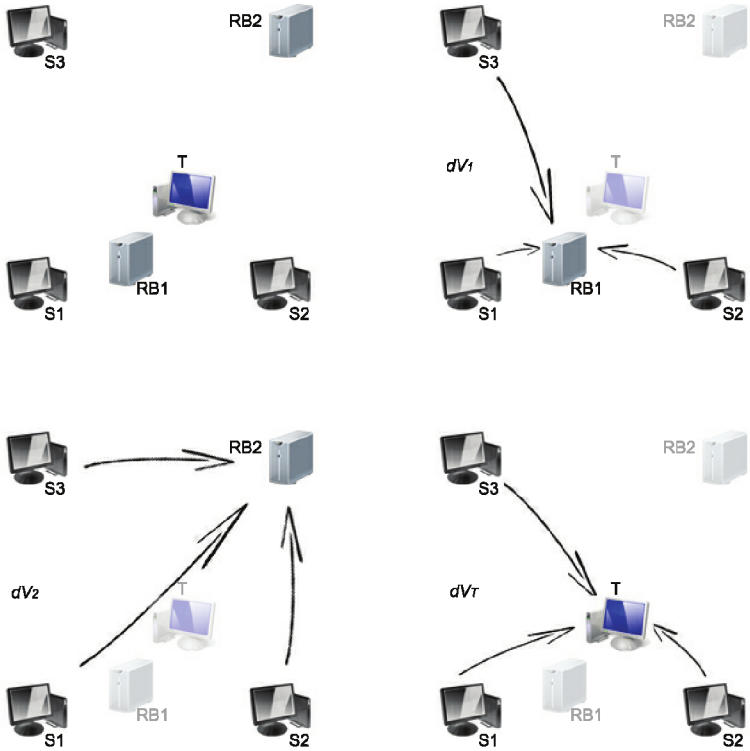
Metoda ShortestPing představuje nejjednodušší aktivní techniku určení pozice cíle na základě použití nejnižší naměřené hodnoty RTT. K činnosti je třeba soubor referenčních bodů, jejichž pozice je známa. Každý z nich měří velikost zpoždění k cíli. Referenční bod s nejnižší hodnotou je pak určen jako hledaný cíl.[10]

Výsledkem geolokace je tedy pozice jednoho z referenčních bodů. Tím vzniká odchylka, neboť poloha vybraného referenčního bodu může být i tak stále značně vzdálena od hledaného cíle. Dále tato metoda neřeší případ, kdy zpoždění od všech referenčních bodů je stejné nebo podobné, nebo kdy se cíl nachází ve velké vzdálenosti od nich.[13]

2.2.2 GeoPing

Metoda Geoping využívá ke své činnosti soubor pasivních referenčních bodů a dále pak sondy, pomocí nichž měří zpoždění. Poloha všech referenčních bodů musí být známa, naopak pro sondy to není bezpodmínečně nutné. Všechny sondy provádí měření zpoždění pro každý referenční bod a taktéž pro zadaný cíl. Výsledkem jsou tzv. *vektory zpoždění* dV. Obrázek 2.1 ilustruje příklad určení polohy cíle T. Vektory pro jednotlivé referenční body jsou porovnány s vektorem cíle a na základě nejmenší Euklidovské vzdálenosti je vybrán referenční bod, který je nejblíže cíli. Jinými slovy lze říci, že je vybrán ten, jehož vektor zpoždění (dV_1) se nejvíce shoduje s vektorem cíle (dV_T). Poloha tohoto referenčního bodu je zvolena jako odhadovaná pozice.[15]

Výsledkem je tedy stejně jako u metody ShortestPing poloha jednoho z referenčních bodů a projevuje se zde stejná chyba jako v předchozím případě. Tento rozdíl je možné eliminovat velkým množstvím referenčních bodů spolu s vhodným rozmístěním sond.[13]



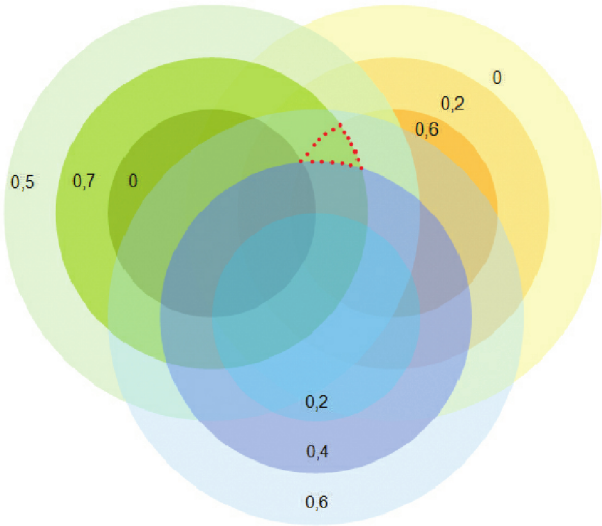
Obr. 2.1: Princip metody GeoPing.

2.2.3 Speed of Internet (*SOI*)

SOI vychází z principu fungování níže popisované metody *Constraint-Based Geolocation (CBG)* (kapitola 3). Metoda je založena nejen na výpočtu vzdálenosti na základě rychlosti šíření světla v optickém vláknu, ale dále zohledňuje zpoždění vznikající na mezilehlých uzlech (viz kapitola 1.1.3). Dle [10] bylo zjištěno, že je nejvhodnější pro výpočet hranic vzdáleností využít konstantu $\frac{4}{9}$ c. Výsledkem je opět průnik kružnic, jejichž středy odpovídají referenčním bodům. Jako odhadovaná pozice cíle je ve většině případů uvažováno těžiště takto nalezeného průniku.

2.2.4 GeoWeight

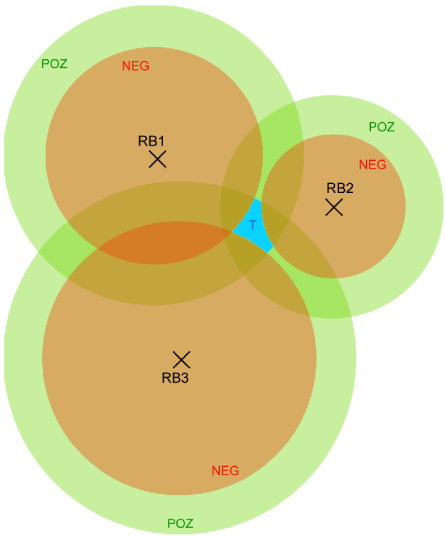
GeoWeight taktéž vychází z principů CBG a funguje na principu multilaterace pomocí referenčních bodů. Všechny vytváří hranice vzdáleností, které jsou následně rozděleny na regiony s různou váhou (Obrázek 2.2). Čím vyšší váha, tím je zde vyšší pravděpodobnost výskytu cíle. Výsledkem měření několika referenčních bodů je průnik oblastí s nejvyšší váhou. Podrobněji je metoda popsána v [16].



Obr. 2.2: Princip metody GeoWeight.

2.2.5 Octant

Tento algoritmus, stejně jako předchozí dvě metody, vychází z CBG a navíc ji rozšiřuje o tzv. negativní informace. Každý referenční bod vytváří oblast, v níž se cíl může nacházet a oblast, ve které ležet nemůže. Vzniknou tedy mezikruží, jejichž průnikem je oblast, ve které se nachází hledaný cíl.[17] Obrázek 2.3 ilustruje výše popsaný proces vytváření pozitivních a negativních oblastí metodou Octant.



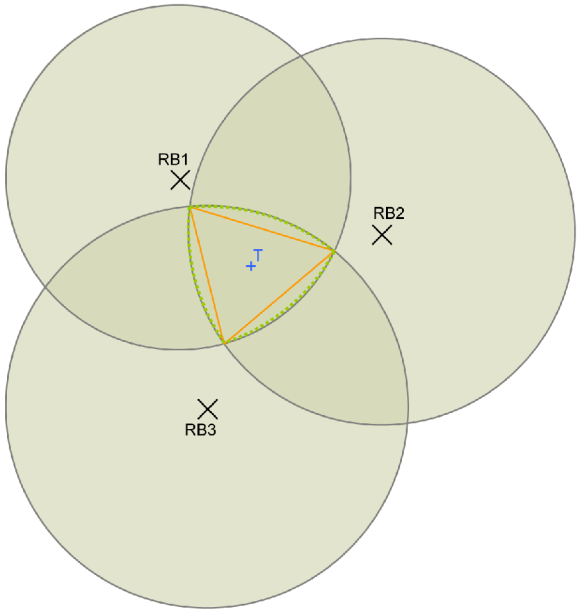
Obr. 2.3: Princip metody Octant.

3 CONSTRAINT-BASED GEOLOCATION (CBG)

Další ze skupiny metod založených na měření zpoždění je Constraint-Based Geolocation [18]. Tato práce se zabývá implementací právě tohoto algoritmu, a bude tedy popsán více podrobněji.

K odhadu pozice cíle se také využívá souboru referenčních bodů schopných měřit zpoždění RTT. Cíl je nalezen pomoci principu *multilaterace* podobně jako u systému GPS. Zde platí, že pokud známe polohu alespoň tří bodů a dále vzdálenost cíle od nich, pak jsme schopni vypočítat i pozici tohoto cíle.

Stejně jako u předchozích metod GeoWeight, Octant a SOI je výsledkem hledání průnik vytvořený z hranic vzdáleností jednotlivých referenčních bodů. Za bodový odhad cíle je považováno těžiště polygonu, který je vytvořen z takto nalezené oblasti.



Obr. 3.1: Princip metody Constraint-Based Geolocation.

3.1 Převod zpoždění na hranice vzdálenosti

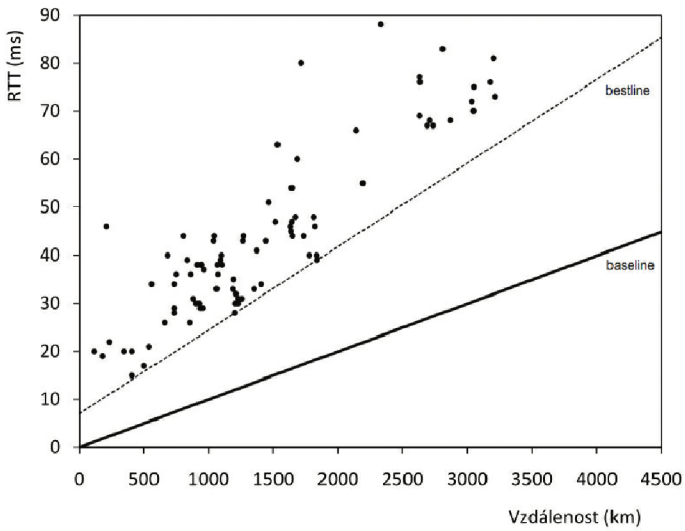
Převod zpoždění na vzdálenost je klíčovým prvkem metody CBG. Základní myšlenkou je vztah, kdy rychlost šíření signálu v optickém mediu se rovná $\frac{2}{3}$ c. Po přepočítání této konstanty lze uvažovat, že 1 ms RTT odpovídá vzdálenosti 100 km.[5] Výše zmíněné by však platilo pouze v ideálních podmínkách. Ve skutečnosti je nutné počítat s dalšími faktory ovlivňujícími převod zpoždění na vzdálenost. Výsledné hodnoty převodního poměru mohou být ovlivněny směrovací politikou jednotlivých

směrovačů na trase, proměnlivým zatížením linek či výpadky. CBG proto nepočítá s konstantní hodnotou rychlosti šíření signálu v přenosovém mediu, ale zjištění vhodně zvolené konstanty šíření dat po trase probíhá pomocí automatické kalibrace mezi jednotlivými referenčními body. Tímto způsobem je zajištěno eliminování nežádoucího zkreslení výpočtu a je zpřesněn odhad cílové oblasti.[18]

3.2 Automatická kalibrace

Procedura zjišťování aktuálního koeficientu rychlosti šíření signálu v přenosovém mediu je prováděna všemi referenčními body a je plně automatická. To každému z nich zaručuje v konečném vytváření hranic vzdáleností nejoptimálnější hodnoty v závislosti na stavu sítě. Postup kalibrace prováděný všemi referenčními body je podle [18] následující:

- Každý referenční bod změří RTT pro všechny ostatní referenční body.
- Jednotlivým zpožděním jsou přiřazeny vzdálenosti příslušných serverů. Z těchto hodnot lze vytvořit *graf závislosti zpoždění na vzdálenosti* (viz Obrázek 3.2).



Obr. 3.2: Graf závislostí zpoždění na vzdálenosti.[18]

- přímka nazvaná **baseline** odpovídá ideálním hodnotám poměru zpoždění a vzdálenosti (1 ms = 100 km). Tuto přímku lze vyjádřit směrnicovou rovnicí:

$$y = k \cdot x + q, \tag{3.1}$$

kde k je určeno rychlostí šíření signálu v optickém kabelu a $q = 0$, neboť neuvažujeme žádné další zkreslující zpoždění.

- Z grafu je vybrán nejlepší poměr vzdáleností a zpoždění. Tento poměr je vyjádřen jako přímka **bestline**. Její směrnicový tvar lze pak zapsat jako:

$$y = k_i \cdot x + q_i. \tag{3.2}$$

  Bestline je přímka ležící pod všemi body grafu a zároveň k nim je nejblíže. Současně musí mít kladný průsečík s osou y.

Tento postup určení bestline provádí každý z referenčních bodů. Tím je zaručeno, že bude ve výpočtu uvažován správný koeficient pro přepočet zpoždění na vzdálenost. Referenční body pak využívají koeficienty k_i a q_i pro výpočet hranice vzdálenosti podle vztahu:

$$g = \frac{d_i - q_i}{k_i}, \tag{3.3}$$

kde d_i je změřené zpoždění RTT a g je výsledná vzdálenost v km.

3.3 Odhad polohy hledaného cíle

V předchozí části textu bylo vysvětleno jakým způsobem probíhá převod naměřeného zpoždění na jednotlivé hranice vzdáleností. CBG využívá multilaterace pro odhad pozice cíle. Každý referenční bod měří zpoždění k hledanému cíli a následným vytvořením hranic vzdáleností na základě rovnice 3.3 vzniknou kruhové oblasti se středem v RB a o poloměru g.

Průnikem těchto kruhů vznikne oblast, v níž se nachází odhadovaná pozice cíle. Jako bodový odhad cíle je podle [18] uvažováno těžiště této oblasti, a proto je třeba ji aproximovat na polygon. Obsah S polygonu s vrcholy v_i o souřadnicích $[x_i, y_i]$ lze vypočítat podle vzorce: [19]

$$S = \frac{1}{2} \sum_{i=0}^{n-1} (x_i \cdot y_{i+1} - x_{i+1} \cdot y_i). \tag{3.4}$$

Jeho těžiště C o souřadnicích $[c_x, c_y]$ je pak dáno vztahy:[19]

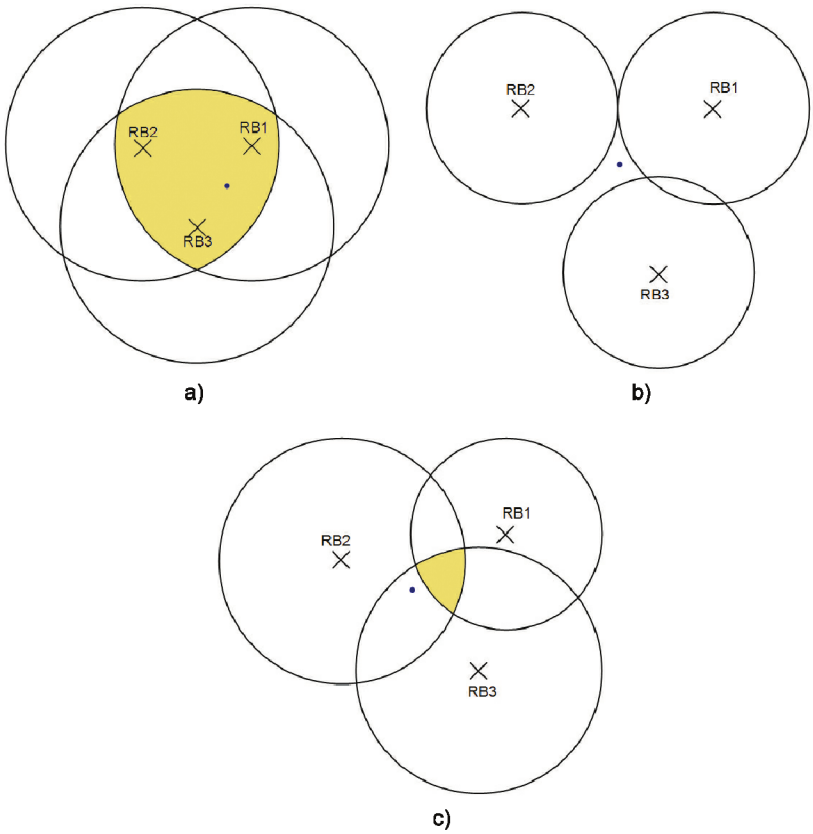
$$c_x = \frac{1}{6S} \sum_{i=0}^{n-1} (x_i + x_{i+1})(x_i \cdot y_{i+1} - x_{i+1} \cdot y_i) \tag{3.5}$$

a

$$c_y = \frac{1}{6S} \sum_{i=0}^{n-1} (y_i + y_{i+1})(x_i \cdot y_{i+1} - x_{i+1} \cdot y_i). \tag{3.6}$$

3.4 Přesnost a omezení metody CBG

V ideálním případě by se hranice vzdáleností vytvořené jednotlivými referenčními body protínaly v jediném bodě. Ve skutečnosti však mohou nastat situace, kdy vlivem parametrů rovnice přímky bestline je hranice nadhodnocena nebo podhodnocena. Případ nadhodnocení způsobuje zvětšení odhadnuté oblasti výskytu cíle. Naopak v případě podhodnocení není možné vytvořit průnik všech kružnic a není tedy možné odhadnout pozici cíle. Existuje i nevhodná kombinace obou předchozích situací, kdy některý z RB hranici vzdáleností podhodnotí a některý naopak nadhodnotí. I přesto, že je možné vytvořit odhadovaný průnik, ten nemusí obsahovat pozici hledaného cíle. Tento stav však není příliš častý.[13] Obrázek 3.3 ilustruje výše popisované situace.



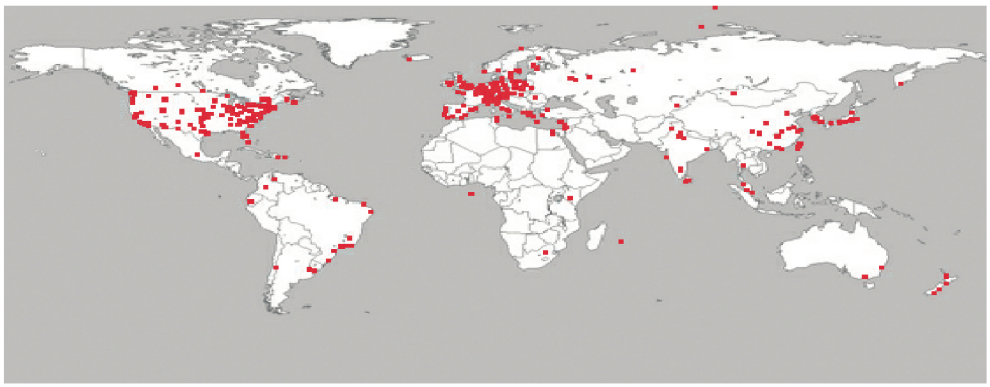
Obr. 3.3: Důsledky nevhodného výpočtu hranic vzdáleností: a) nadhodnocení hranic, b) podhodnocení hranic, c) kombinace předchozích situací.

4 PLANETLAB

Kapitola o síti PlanetLab je zde z důvodu jejího využití pro geolokační techniku CBG. Servery jsou využívány jako referenční body z důvodu relativně stálé dostupnosti a známé poloze. Dalším důvodem je možnost jejich konfigurace a využití k měření RTT.

PlanetLab je celosvětová experimentální síť, sloužící k testování a vývoji převážně akademických projektů a aplikací. Poskytuje prostor pro testování v reálných podmínkách sítě Internet. Vznikla v roce 2002 na území Spojených států amerických díky zapříčinění Larryho Petersona z Princetonské univerzity. Postupně se pak rozšířila po celém světě a provozuje ji řada světových univerzit a firem z oblasti informatiky a telekomunikací. Jsou to např. *Intel, Google, Hewlett Packard* nebo *AT&T*. Česká Republika je zastoupena organizací *CESNET*.[20]

V současné době je do sítě PlanetLab připojeno 1041 uzlů (*node*) soustředěných do 557 organizací (*site*). Pokud se zaměříme pouze na evropskou část sítě, je to 346 uzlů a 157 organizací. Na obrázku 4.1 je vidět rozmístění organizací po světě.[20]



Obr. 4.1: Uzly sítě PlanetLab. [20]

Síť je rozdělena na dvě části – **PlanetLab Europe** (*PLE*), obsahující evropské organizace a uzly a **PlanetLab Central** (*PLC*) pro ostatní. Základní rozdělení sítě je na organizace (*site*). Jedná se vlastně o sídlo společnosti, která poskytuje servery (*node*). Každý projekt (*slice*) má na serveru přidělen vlastní virtuální stroj s OS GNU/Linux v distribuci CentOS. Tato koncepce zajišťuje, že se jednotlivé projekty vzájemně neovlivňují.[20] Pro přístup k jednotlivým uzlům je využíván protokol SSH, který poskytuje zabezpečené šifrované spojení.

Jednotlivé uzly jsou ideální volbou pro využití geolokační metody CBG. Díky jejich známé pozici mohou být využity jako referenční body. Taktéž umožňují provádět měření RTT zpoždění v síti. Pro tuto práci budou zvoleny uzly náležící organizacím se sídlem v Evropě (*PLE*).

5 IMPLEMENTACE ALGORITMU CBG

Tato kapitola popisuje hlavní náplň práce, tj. vytvoření programu v jazyce *Java*, který implementuje algoritmus CBG. Za pomoci serverů sítě PlanetLab bude prováděna geolokace vybraných cílů a z takto vytvořených dat pak bude ověřena správná funkce programu a zhodnoceny výsledky měření.

Kapitola je rozdělena na několik částí. Nejprve bude přiblíženo, jakým způsobem jsou využity stanice sítě PlanetLab a bude popsán výběr vhodných serverů sloužících jako referenční body. Další část se věnuje struktuře souborů zpracovávaných v průběhu geolokace. Referenční body je třeba spravovat a získávat z nich data, která algoritmus dále využívá k měření. Pro tyto učely slouží konfigurační soubory. Kapitola dále popisuje algoritmus pro výpočet bestline jednotlivých referenčních bodů a s ním spojené konfigurační soubory. Dále je vysvětleno samotné vytvoření hlavní části algoritmu CBG a jakým způsobem je dosáhnuta konečná poloha odhadovaného cíle. Poslední část se pak zaměřuje na to, jakým způsobem jsou prezentovány výsledky geolokace. Bude popsána verze aplikace určená pro stanice s OS Linux a dále pak verze přizpůsobená pro nasazení na geolokační server.

5.1 Využití serverů sítě PlanetLab

Jak již bylo uvedeno tuto experimentální síť lze využít pro účely referenčních bodů hlavně z důvodu možnosti správy jednotlivých serverů. Dále pak díky tomu, že umožňují měření obousměrného zpoždění RTT, ať už mezi sebou navzájem, tak vůči hledanému cíli. V neposlední řadě je to pak poměrně velký výběr ze serverů umístěných v Evropě, kde budou probíhat jednotlivá měření.

Pro správnou funkci metody CBG bylo nutné vytvořit seznam vhodných referenčních bodů, které rovnoměrně pokrývají oblast Evropy. Jako podklady pro výběr slouží databáze serverů sítě PlanetLab, kterou vytvořili společně kolegové Pavol Iľko, Jakub Polášek a Bc. Ján Pružinský. Tato databáze obsahuje kromě všech serverů také další velmi důležité údaje nutné pro výběr vhodných referenčních bodů. Nechybí důležité rozdělení na evropské stanice (*PLE*) a ostatní (*PLC*). Mezi další hodnoty, které byly zohledněny při výběru, patří informace o tom, které uzly je možné spravovat prostřednictvím protokolu SSH, zda jsou stanice dostupné a v neposlední řadě to byla schopnost odpovědi programu ping. Ne vždy totiž platilo, že stanice splňovala současně všechny tyto podmínky. Seznam dále obsahuje polohu všech serverů a údaje o přesnosti této hodnoty.

Pro tuto práci bylo zvoleno celkem 20 serverů představujících referenční body. Všechny splňují výše popsané podmínky a umožňují tedy přístup přes zabezpečený protokol SSH a jsou dostupné z více než 90 % času. Obrázek 5.1 zobrazuje polohu

všech vybraných referenčních bodů na mapě. Kompletní seznam včetně doménových názvů, IP adres a zeměpisných souřadnic je uveden v příloze A.

Obr. 5.1: Poloha vybraných referenčních bodů.

Přístup ke stanicím je zajištěn zabezpečeným spojením. Tato práce spadá do projektu s názvem *cesnet_feec*, jehož název současně slouží jako přihlašovací jméno. Autentizace probíhá pomocí asymetrických SSH klíčů. Bylo tedy nutné vytvořit dvojici šifrovacích klíčů a ten veřejný umístit na stanice sítě PlanetLab. Manuální distribuce na všechny servery je prakticky velmi časově náročné a nevhodné řešení, proto ji automaticky zajišťuje sama síť. Prostřednictvím webového konfiguračního rozhraní[1] byl klíč naimportován do systému a automaticky rozšířen na všechny servery.

Druhou možností využití sítě PlanetLab je seznam cílů, pomocí kterých byla testována funkce a přesnost vytvořeného programu. Několik serverů tedy sloužilo jako cíl se známou polohou a díky tomu bylo možné vypočítat odchylku mezi odhadovanou a skutečnou pozicí. Seznam všech cílů mezi kterými se vyskytují i servery z této experimentální sítě lze nalézt v příloze B.

[1] `http://www.planet-lab.org/`

5.2 Formát vstupních dat a souborů

Data zpracovávaná při hledání jednotlivých cílů lze rozdělit do dvou skupin. Ty současně využívají rozdílný formát zápisu.

První takovou skupinou jsou vstupní data, která jsou přádávána programu jako parametr při spuštění (viz tabulka 5.4). Patří sem především seznam referenčních bodů, se kterým se dále pracuje. Tyto servery jsou zapsány jako pole *JSON*[2] *objektů*. *Pole* je seřazenou kolekcí *objektů*. Začíná znakem `"["` a končí znakem `"]"`, hodnoty jsou odděleny znakem `","`. *Objekt* představuje množinu párů *název* : *hodnota*. Jednotilvé páry jsou pak odděleny znakem `","` a celý objekt je uvozen znakem `"{"` a končí `"}"`. Pro ukázku je níže uveden formát vstupních dat pro dva referenční body:

```
[{"name":"ple1.cesnet.cz","ip":"195.113.161.13","lat":"50.102",
"lng":"14.3916"},{"name":"planetlab1.cs.vu.nl","ip":"130.37.193.141",
"lat":"52.35","lng":"4.9"}]
```

Program takto formátovaný seznam referenčních bodů akceptuje buď jako text, a nebo je možné ho předat jako samostatný soubor.

Druhou skupinou dat a současně tím i jiným formátem jsou data zpracovávaná za běhu programu. Ten vytváří celkem tři soubory s daty potřebnými k nalezení polohy cíle. Celým procesem vytváření souborů a jejich funkcí se zabývá kapitola 5.5, zde bude popsána pouze jejich struktura a obsah. Jedná se o jednoduchý formát *CSV* určený pro textové soubory obsahující tabulková data. Jednotlivé položky jsou odděleny znakem `";"`.

Prvním je soubor `servery` obsahující všechny referenční body spolu s jejich souřadnicemi. Ten je vytvořen programem ze vstupního JSON seznamu referenčních bodů. Každému serveru odpovídá jeden řádek obsahující jeho doménový název, souřadnice zeměpisné šířky a zeměpisné délky. Struktura souboru `servery` je zobrazena v tabulce 5.1.

Tab. 5.1: Struktura souboru `servery`.

Doménový název serveru 1	;	Latitude1	;	Longitude1
Doménový název serveru 2	;	Latitude2	;	Longitude2
Doménový název serveru 3	;	Latitude3	;	Longitude3
⋮		⋮		⋮

Pro účely kalibrace serverů (viz kapitola 5.4) je pak vytvářen soubor `serverybl` s identickou strukturou.

[2] *JavaScript Object Notation* – `http://www.json.org/json-cz.html`

Dalším je soubor `spoje`, který obsahuje data potřebná ke kalibraci referenčních bodů. Jedná se o výsledky měření RTT každého serveru pro všechny ostatní stanice. Soubor obsahuje název zdrojového serveru, následuje cílový server a jako poslední hodnota je uložena velikost RTT zpoždění mezi nimi. Tabulka 5.2 názorněji popisuje strukturu souboru `spoje`.

Tab. 5.2: Struktura souboru `spoje`.

Doménový název serveru1	;	Doménový název serveru2	;	RTT1
Doménový název serveru1	;	Doménový název serveru3	;	RTT2
Doménový název serveru2	;	Doménový název serveru1	;	RTT3
⋮		⋮		⋮

Posledním ze zpracovávaných souborů je `target`. Ten obsahuje hodnoty zpoždění RTT naměřené všemi servery vůči hledanému cíli. Jeho struktura je v tabulce 5.3.

Tab. 5.3: Struktura souboru `target`.

Doménový název serveru 1	;	RTT1
Doménový název serveru 2	;	RTT2
⋮		⋮

5.3 Proces odhadování pozice cíle

V předchozích kapitolách bylo vysvětleno, jakým způsobem je využita síť PlanetLab a byl popsán formát dat, se kterými program pracuje. Následující část se věnuje celému procesu geolokace stanice v Internetu a funkcím programu. Budou postupně popsány dílčí části od načtení dat až po prezentaci výsledků.

Program je vytvořen v jazyce *Java* a přizpůsoben pro OS Unix. Kromě hlavní aplikace obstarávající výpočty, samotný odhad a zobrazení výsledků jsou součástí programu i další konfigurační *Bash skripty*. Ty byly vytvořeny za účelem získávání dat z jednotlivých serverů a aktualizací referenčních bodů. Další skripty pak zajišťují měření RTT mezi jednotlivými servery navzájem a k cíli. Celá aplikace i skripty a jejich funkce budou popsány níže v textu. Spouštění programu probíhá pomocí příkazu v terminálu. Jednotlivé možnosti aplikace jsou nastavovány pomocí parametrů a přepínačů předávaných na vstupu. Níže vypsaná nápověda k programu znázorňuje výčet veškerých možností:

```
usage java -jar cbgWorkstation.jar -d|-D -b|-B [-o|-O output_format]
                                   [-u|-U ssh_user] [-pki|-PKI private_key]
                                   [-t|-T target] [-s|-S real_coordinates]
                                   [-l|-L JSON_landmarks]
```

Tabulka 5.4 pak popisuje funkci jednotlivých položek.

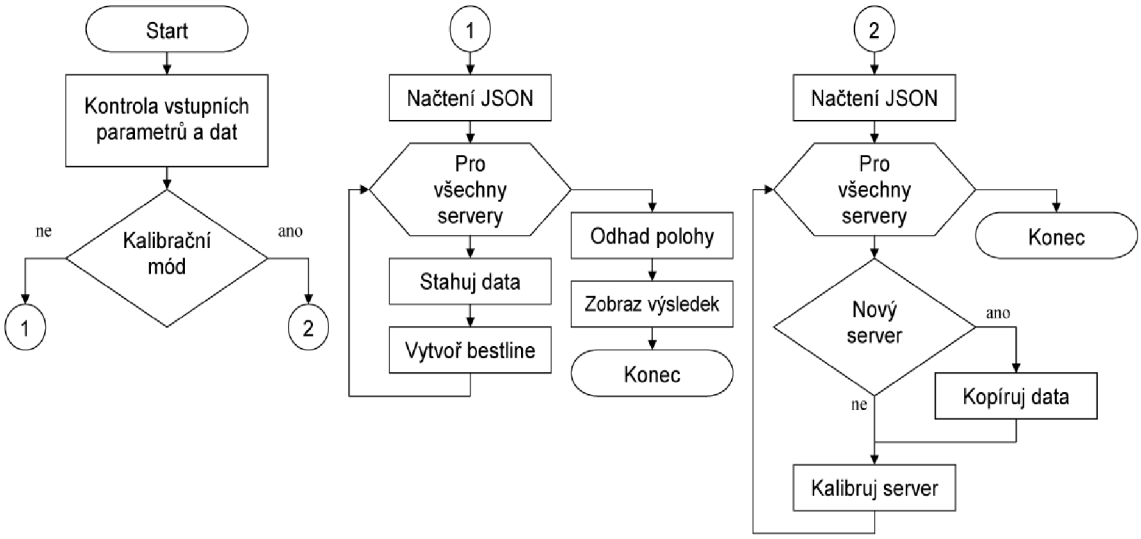
Tab. 5.4: Vstupní přepínače a parametry programu.

přepínač	parametr	popis funkce
`-d|-D`	–	debug mód vypisující důvod chyby při běhu programu
`-b|-B`	–	mód pro kalibraci referenčních bodů, viz kapitola 5.4
`-o|-O`	`mapa / vypis / server`	výsledky jsou zobrazeny v grafickém okně a do terminálu jsou vypsány seznamy a výpočty / vypsány pouze seznamy a výpočty do terminálu / výstup přizpůsobený pro geolokační server, viz kapitola 5.6.3
`-u|-U`	ssh_user	jméno uživatele pro ssh spojení
`-pki|-PKI`	private_key	cesta k privátnímu ssh klíči
`-t|-T`	target	doménový název nebo IP adresa hledaného cíle
`-s|-S`	lat,lon	skutečné souřadnice potřebné pro výpočet odchylky
`-l|-L`	JSON_landmarks	seznam referenčních bodů v JSON nebo cesta k souboru se servery

Parametry vypsané červenou barvou jsou povinné, pokud není tedy jakýkoliv z nich zadaný, je vypsána nápověda. Vyjimku tvoří kalibrační mód (kapitola 5.4), kde není povinný údaj o hledaném cíli. Program je tedy spuštěn například zadáním příkazu:

```
java -jar cbgWorkstation.jar -o mapa -u cesnet_feec -pki /files/klice/pkey
-t 91.123.196.1 -s 56.15,15.59 -L /files/test/serveryarray.txt
```

Tím je zahájen proces geolokace. Proběhne připojení na všechny referenční body ze seznamu a pomocí nich je provedeno měření obousměrného zpoždění vůči cíli. Tato data společně s kalibračním měřením jsou stažena a kompletována. Pro každý server je vypočítána přímka bestline a na jejím základě je z hodnoty RTT vytvořena hranice vzdálenosti. Průnikem těchto kruhových hranic vznikne oblast, která je aproximována na polygon z jehož vrcholů je vypočítáno těžiště. Výsledky jsou nakonec zobrazeny uživateli. Celý proces je ve formě vývojového diagramu znázorněn na obrázku 5.2.



Obr. 5.2: Vývojový diagram programu `cbgWorkstation.jar`.

5.4 Výpočet kalibrační křivky

Kalibrační mód zajišťuje všem referenčním bodům aktuální informace o stavu sítě a přispívá tak ke zpřesňování výpočtů hranic vzdáleností. Tím dochází i k celkovému změnšení chyby odhadu geolokačního algoritmu. Druhým využitím je pak možnost kopírování potřebných dat na nové servery sítě PlanetLab.

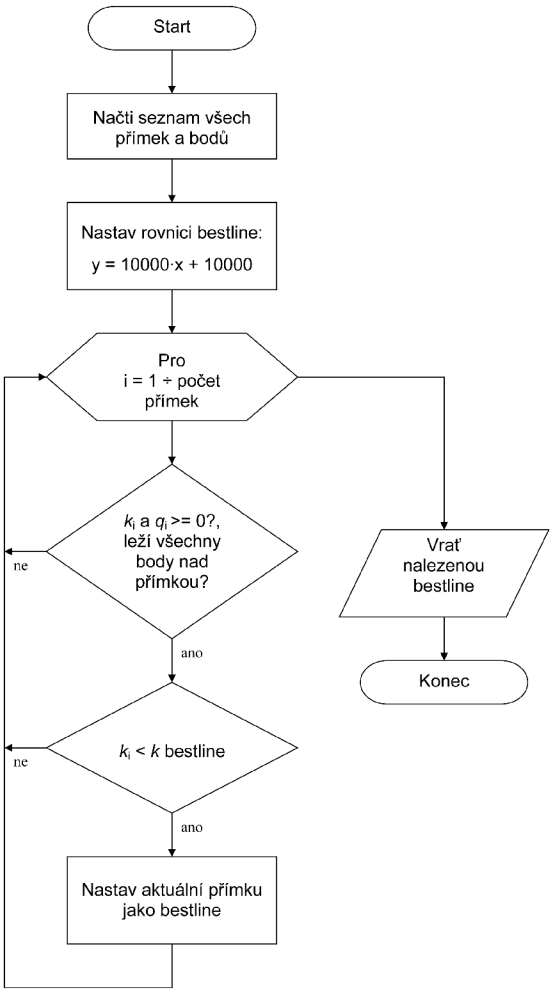
Nedílnou součástí procesu kalibrace je bash skript `kopirovani.sh`. Jeho obsah zde není uveden z důvodu rozsáhlé velikosti, která by narušovala text práce. Bude popsána pouze jeho funkce. Skript je spouštěn Java aplikací `cbgWorkstation.jar`, která mu jako vstupní parametry předává *privátní klíč*, *jméno uživatele* a soubor `serverybl` (viz tabulka 5.1). Pomocí SSH klienta je vytvořeno spojení na jednotlivé referenční body a zkontroluje se, zda existuje adresář s metodou CBG. Pokud tomu tak není, jsou na server nakopírovány potřebné soubory a je spuštěna kalibrace za pomoci bash skriptu `ping.sh` umístěného na serveru. Pokud adresář existuje, předpokládá se, že všechna potřebná data již na serveru jsou a pouze se spustí kalibrace. Tento postup zvyšuje univerzálnost vytvořené metody a v případě potřeby je možné měnit referenční body nebo jejich počet.

Skript `ping.sh` je umístěn na všech referenčních bodech. Slouží pro měření hodnot zpoždění RTT potřebných k automatické kalibraci. Postupně je procházen seznam serverů a na každý z nich je za pomoci programu *ping* provedeno 5 měření, ze kterých je vybrána minimální hodnota. Tyto údaje spolu s informacemi o zdrojovém a cílovém serveru jsou uloženy do souboru `DomenovýNazevZdroje.spoj` v adresáři `/res/bestline/` každého referenčního bodu.

Další část textu popisuje získání dat pro potřeby výpočtů nutných k nalezení jednotlivých přímek bestline všech referenčních bodů. Jako zdroj dat slouží soubory `servery` a `spoje`, jejich struktura je znázorněna v kapitole 5.2 a způsob jakým byly vytvořeny popisuje kapitola 5.5. Graf přímky bestline představuje závislost zpoždění na vzdálenosti. Je tedy zřejmé, že se při výpočtech bude využívat právě těchto dvou hodnot.

Samotný algoritmus pracuje ve dvou fázích. První z nich je příprava potřebných dat. Ta spočívá ve vytvoření několika dynamických polí. Jedním z nich je pole typu *ArrayList<Bod>* představující graf závislosti zpoždění na vzdálenosti. Toto pole je vytvořeno pro každý server a obsahuje množinu bodů reprezentovaných souřadnicemi [*vzdálenost, zpoždění*]. Z nich je pak vytvořen seznam všech přímek. Ty jsou uloženy ve formě parametrů k a q (viz kapitola 3.2).

Samotný výběr přímky bestline provádí metoda *vytvorBestline()*. Celý algoritmus výběru je na obrázku 5.3.



Obr. 5.3: Vývojový diagram algoritmu pro výběr přímky bestline.

Jako bestline je nejprve uložena přímka s parametry k a $q = 10000$. Hodnoty jsou dočasně zvoleny jako dostatečně vysoké a je tím zajištěno, že se všechny body nachází pod přímkou. Druhým důvodem pro takto přesně zadanou hodnotu parametrů je pozdější kontrola, zda byla či nebyla nalezena bestline příslušného referenčního bodu. Postupně je pak procházen celý seznam přímek vytvořený před samotným výpočtem a je zjišťováno, zda všechny body leží nad aktuální přímkou. Pokud tomu tak je, aktuálně kontrolovaná se uloží jako bestline a pokračuje se v testování zbývajících. Podstatou testování, zda bod leží nad přímkou, je řešení nerovnice:

$$y > k \cdot x + k, \tag{5.1}$$

kde k, q jsou parametry aktuální přímky a x, y jsou souřadnice testovaného bodu. Přesněji řečeno x odpovídá vzdálenosti mezi servery a y zpoždění naměřenému mezi nimi. Výsledkem celého procesu hledání přímek je pole *ArrayList<Bestline>* obsahující příslušný referenční bod a jeho nalezená bestline uložená jako instance třídy *Smernice*.

5.5 Hlavní část algorimu CBG

Kompletní vývojový diagram metody CBG lze nalézt na obrázku 5.2. Kalibrační mód, který zajišťují především bash skripty `ping.sh` a `kopirovani.sh`, popisuje předchozí kapitola, zbývá tedy podrobně vysvětlit proces samotné geolokace zadaného cíle, kterou představuje první větev diagramu začínající v bodě *1*. V prvním kroku vyhledávání polohy je třeba vytvořit soubory `servery`, `spoje` a `target`, které jsou nezbytné pro další pokračování procesu.

Vše začíná zpracováním seznamu referenčních bodů ve formátu JSON vloženého pomocí vstupního parametru. Program rozeznává dvě možnosti, jakými lze data předat. Jednou z variant je soubor obsahující seznam referenčních bodů, v tomto případě je jako parametr zadána cesta k němu. Druhou možností je seznam vypsaný přímo do terminálu. Parsování JSON dat ze vstupu a rozlišení varianty zadávání zajišťuje metoda *nactiJSON()* a předává je druhé metodě *parsuj()*, která vytvoří výsledný soubor `servery`.

Další průběh spočívá v komunikaci s referenčními body a získání všech potřebných dat, které jsou uloženy na nich. To zajišťují bash skripty `komplet.sh` ve spolupráci s `pingtar.sh` uloženým na všech serverech. Výsledkem je vytvoření zbývajících dvou CSV souborů. Skript `komplet.sh` potřebuje jako vstupní data seznam všech referenčních bodů ve formě souboru `servery` a lze ho rozdělit do tří částí.

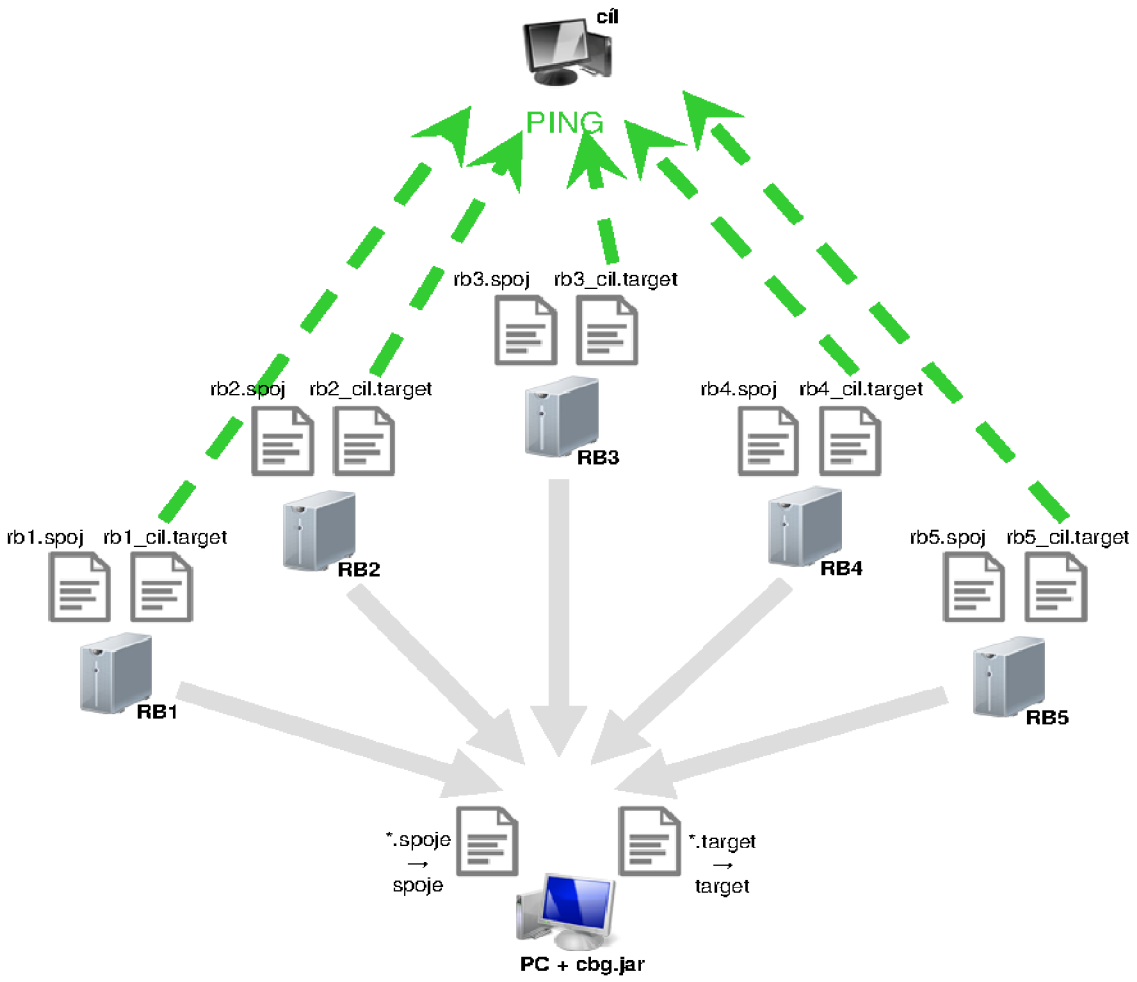
Nejprve je na každém serveru spuštěn skript `pingtar.sh`, kterému je předána jako vstupní parametr adresa hledaného cíle. Za pomoci něj je provedeno programem *ping* 5 měření vůči cíli a z nich je pak vybrána minimální hodnota RTT. Tyto

výsledky jsou na serveru uloženy do souboru `JménoServeru_AdresaCíle.target` v adresáři `/res/AdresaCíle/`.

Ve druhé fázi skriptu jsou ze všech referenčních bodů staženy soubory `.spoj` a dále pak výše popsaný soubor `.target` příslušného cíle.

Posledním krokem je tvorba zbývajících dvou souborů. To znamená, že všechna stažená kalibrační měření jsou spojena do jednoho souboru `spoje` a dále pak veškeré výsledky měření RTT k cíli do souboru `target`.

Výše popsaná funkce skriptu `komplet.sh` je znázorněna na obrázku 5.4



Obr. 5.4: Kolekce dat pomoci skriptu `komplet.sh`.

Po úspěšném vytvoření souborů má metoda CBG všechna data potřebná k lokalizaci cíle. Data jsou načtena a uložena do polí, se kterými se následně pracuje v programu. Pole spojů je navíc rozšířeno o vzájemnou vzdálenost referenčních bodů, která je vypočítána za pomoci metody *vypocetVzdalenosti()*. Zmíněná metoda implementuje Vincentiho algoritmus pro výpočet vzdálenosti dvou bodů na

zeměkouli.[21]. Vstupním argumentem je dvojice zeměpisných souřadnic a metoda vrací jejich rozdíl v km.

Následuje proces hledání bestline pro jednotlivé servery podle algoritmu na obrázku 5.3. Vše implementují metody *vytvorBestline()* a *vytvorSeznamBestline()*. Po nalezení přímek bestline všech serverů sloužících jako referenční body je možné začít odhadovat pozici cíle. Prvním krokem je výpočet hranic vzdáleností všech RB, dále je třeba vytvořit průnik těchto oblastí. Nalezený průnik je aproximován na polygon a je vypočítáno jeho těžiště, které je prohlášeno za výsledek geolokace. Pokud je známa skutečná pozice hledaného cíle, která lze zadat jako nepovinný vstupní argument, je navíc vypočten rozdíl mezi odhadnutou a skutečnou pozicí cíle.

Při výpočtu hranic vzdáleností je využíván již dříve zmíněný soubor `target`. Obsahuje pouze informace o zdrojovém serveru a hodnotě zpoždění, kterou tento server naměřil k cíli. Je tedy třeba ho přepočítat na vzdálenost. To zajišťuje metoda *vytvorCil()*, jejímž výstupem je pole záznamů obsahující zdrojový server, vzdálenost od cíle a zpoždění. Vzdálenost je přepočítávána dosazením známých parametrů přímky bestline k a q a hodnoty zpoždění RTT do rovnice 3.3.

Jsou tedy známy jak polohy serverů, tak jejich hranice vzdáleností, a proto je možné vytvořit jednotlivé kruhové oblasti. Výstupem metody *vytvorKruhy()* je pole instancí třídy *Area* obsahujících objekt *Ellipse2D.Double*. Tato metoda navíc provádí korekce geografických souřadnic a vzdálenosti na hodnoty korespondující s mapovým podkladem. Tento přepočet bude dále vysvětlen v kapitole 5.6.2.

Pro vytvoření průniku hranic vzdáleností je využívána metoda *intersect()*. Jedná se o metodu určenou přímo pro práci s oblastmi. Výstupem je opět instance třídy *Area*. Aproximaci oblasti na polygon zajišťuje metoda *vytvorPolygon()*. Vstupním argumentem je objekt třídy *Area*, který je převeden na množinu souřadnic představujících výsledný polygon. Tento převod obstarává rozhraní *PathIterator*. Z vytvořeného polygonu je pak za pomocí vzorců 3.4 a 3.5, 3.6 metodou *spocitejTeziste()* vypočten jeho obsah a následně těžiště.

5.6 Možnosti zobrazení výsledků geolokace

Výsledky výpočtů a odhadu cílové pozice jsou prezentovány třemi způsoby. Prvním z nich je textový výstup ve formě seznamů a výpočtů. Vše je zobrazováno na standardním terminálovém výstupu. Druhým typem je samostatné okno zobrazující mapu Evropy se zakreslenými hranicemi vzdáleností, pozicemi referenčních bodů a cíle. Do terminálu jsou pak vypsány výsledky odhadu. Třetí možností je výsledek výpočtů přizpůsobený pro nasazení programu na geolokační server. Výstupní formát je volen při spouštění programu přepínačem `-o|-O`. V závislosti na zvole-

ném parametru jsou volány metody *vypisSeznamy()* v případě parametru `vypis`, *ukazMapu()* zvolením `mapa` a nebo *vypisServer()* volbou `server`. Pokud není zadán žádný parametr, je automaticky uvažována prezentace výsledků ve formě grafického okna.

5.6.1 Seznam výsledků

Všechny kroky výpočtů a vytvářené seznamy jsou v průběhu geolokace zobrazeny na standardní výstup. Následující tabulka zobrazuje strukturu výstupních dat. Konkrétní příklad zobrazení průběhu geolokace je díky jeho velikosti v příloze C.

Tab. 5.5: Struktura seznamu výsledků zobrazovaných na standardní výstup.

Servery: počet ks		
Server1	Latitude1	Longitude1
⋮	⋮	⋮
Spoje: počet ks		
Zdrojový RB1 ⟶ Cílový RB2	Vzdálenost1	RTT1
⋮	⋮	⋮
Body pro server 1:		
[vzdálenost1, RTT1]		
⋮		
Body pro server 2:		
⋮		
Bestlines pro jednotlivé servery:		
Server1:	*k1*	*q1*
⋮	⋮	⋮
Vzdálenosti cíle od jednotlivých serverů:		
Server1:	poloměr kružnice1 [km]	
⋮	⋮	
Souřadnice polygonu:		
$[x_1, y_1]$		
⋮		
Souřadnice těžiště: $[x, y]$		
Vypočtená pozice: $[Latitude, Longitude]$		
Skutečná pozice: $[Latitude, Longitude]$		
Rozdíl: [km]		

5.6.2 Grafická prezentace

Jak již bylo zmíněno, pro grafické zobrazení výsledků je využito nové okno s mapovým podkladem. To je zobrazeno za pomoci grafického rozhraní *AWT* a celé vykreslení pak zajišťuje třída *Mapa*.

Oblast, ve které bude prováděna geolokace, je omezena na území Evropy. Při konverzi výsledků do mapového podkladu bylo třeba vyřešit několik problémů.

Prvním z nich byla prezentace trojrozměrného tělesa, jako je zeměkoule, do roviny. Tímto problémem se zabývají mapové projekce, konkrétně pro tuto práci byla vybrána **Mercatorova válcová projekce** [22]. Jejím výstupem je obdélníková plocha, poledníky jsou zobrazeny jako vertikální a rovnoběžky jako horizontální rovnoběžné čáry. To je pak velmi výhodné pro konverzi do kartézských souřadnic.

Jako mapový podklad slouží mapa ze serveru *http://maps.google.com*, která byla upravena do potřebných rozměrů. Zobrazovaná oblast je od 35° do 70° severní šířky a od −15° do 40° východní délky. Velikost mapy je $832 \times 529\,\mathrm{px}$ a tyto rozměry dávají vůči rozsahu poledníků a rovnoběžek stejné koeficienty px/°. Aby byly body zobrazeny ve správném měřítku, je nutné jimi vynásobit výsledné souřadnice.

Pro zobrazení pomoci Mercatorovy projekce bylo dále nutné vytvořit metodu pro konverzi souřadnic trojrozměrného tělesa na souřadnice pro rovinnou plochu. V případě souřadnic zeměpisné délky (*Longitude*) není třeba žádný přepočet a postačuje tedy pouze násobení koeficientem měřítka mapy. V případě hodnot zeměpisné šířky (*Latitude*) je však potřeba převod za pomoci vztahu:

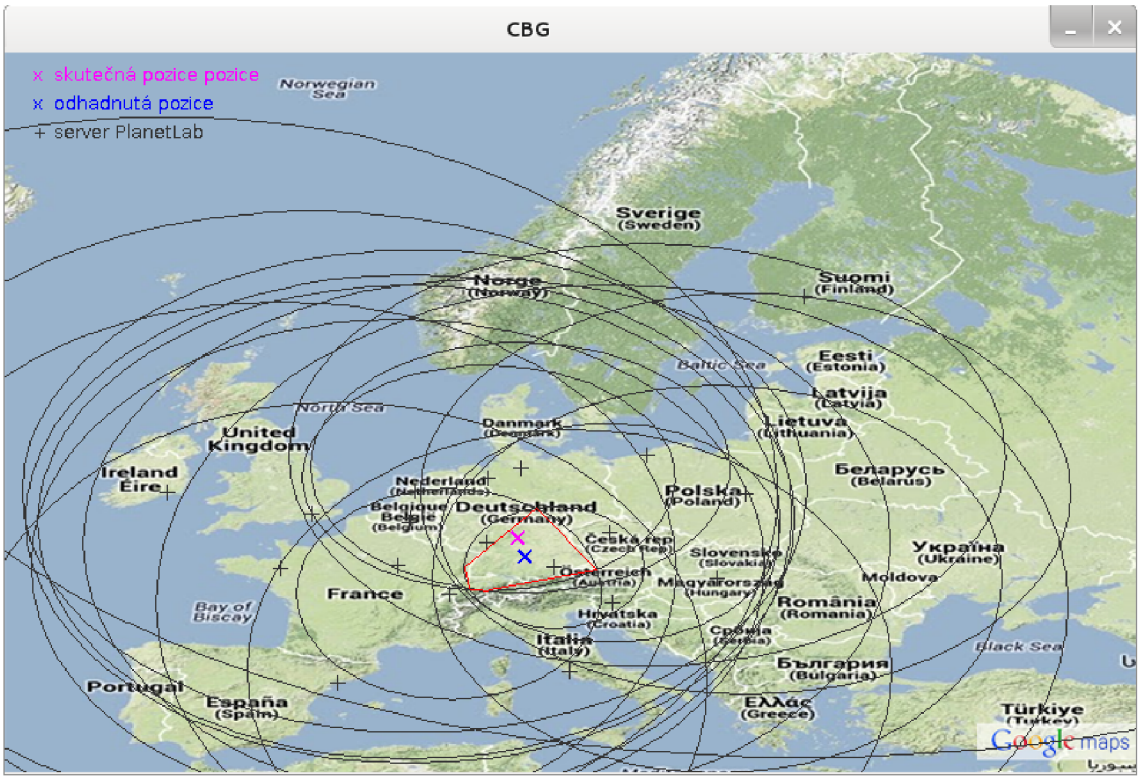
$$\ln\left[\tan\left(\frac{latitude}{2}+\frac{\pi}{2}\right)\right].[22] \tag{5.2}$$

Konverezi souřadnic do hodnot korespondujících s mapovým podkladem zajišťuje metoda *prepocitejLatLon()*.

Převod trojrozměrného tělesa do ploché mapy za pomoci Mercatorovy projekce způsobuje zkreslení ve směru rovnoběžek. Toto zkreslení se rostoucí zeměpisnou šířkou zvětšuje. Kružnice je pak zkreslena do tvaru elipsy. Z tohoto důvodu je třeba zajistit, aby hranice vzdáleností měly tvar elipsy. Metoda *vytvorKruhy()* je tedy upravena tak, aby neukládala hranice vzdálenosti s konstantním poloměrem. Vzdálenost v km je přepočítána s ohledem na polohu středu elipsy. Délka vedlejší poloosy je přepočítávána v poměru $1° = 110,54\,\mathrm{km}$. V případě hlavní poloosy se uplatní zkreslení a je nutné ji přepočítávat s ohledem na polohu. Výsledný poměr se rovná $1° = 111,32 \cdot \cos(Latitude)\,\mathrm{km}$.[23] Hodnoty je nutné nakonec vynásobit koeficientem měřítka mapy.

Po všech předchozích přepočtech je možné zobrazit mapu společně s vypočtenými objekty. Vykresleny jsou všechny referenční body a jejich elipsy hranic vzdáleností.

Dále program vykreslí výsledný polygon a jeho těžiště. Pokud budou zadány skutečné souřadnice cíle, bude vykreslena i tato pozice. Grafickou prezentaci je možno vidět na obrázku 5.5.



Obr. 5.5: Okno zobrazující výsledky vyhledávání na mapě.

Na mapovém podkladu Evropy jsou zobrazeny pozice referenčních bodů a jejich příslušné hranice vzdáleností reprezentované elipsami. Červenou barvou je zobrazen polygon, vytvořený z odhadované oblasti. Jeho těžiště a zároveň odhadovanou pozici pak představuje modrý kříž. Posledním prvkem je skutečná pozice serveru označena růžovou barvou.

5.6.3 Úprava programu pro geolokační server

Vytvořený program určený pro stanice s OS Unix byl dále upraven pro nasazení na geolokační server[3] nacházející se na Ústavu telekomunikací Fakulty elektrotechniky a komunikačních technologií Vysokého učení technického v Brně. Server je provozovaný Ing. Lukášem Vernerem a poskytuje webové rozhraní pro měření pomocí geolokačních metod.

[3] `http://betka.utko.feec.vutbr.cz/`

K dispozici je seznam uzlů sítě PlanetLab, které mohou být použity jako referenční body aktivních geolokačních metod. Dále nabízí rozsáhlý seznam cílů doplněných o skutečné zeměpisné souřadnice. Každému uživateli je umožněna správa vlastních metod a tvorba uživatelských testů, které mohou být spouštěny v různých konfiguracích. Server generuje statistiky naměřených hodnot a z nich následně vytváří grafy. Jednotlivé metody pak lze díky nim porovnávat.

Úpravy celého programu pro fungování společně s geolokačním serverem nebyly nijak zásadní, neboť již při návrhu bylo počítáno s jejím provozováním nejen na lokálních stanicích, ale také právě na něm. Formát vstupních dat a parametrů proto byly voleny s ohledem na podmínky kladené v nápovědě geolokačního serveru. Přizpůsobení tedy spočívalo převážně v úpravě formátu výstupních dat a dále nebylo nutné implementovat metody zajišťující výpočet odchylky, neboť ho provádí server sám. Program tedy výsledek vypisuje v očekávaném formátu JSON dat obsahujících zeměpisné souřadnice odhadované pozice. Výpis je ve tvaru:

```
{"lat":latitude,"lng":longitude}
```

V případě jekékoliv chyby v průběhu geolokace nebo pokud není možno polohu cíle určit, je výstup formátován následovně:

```
{"lat":null,"lng":null}
```

Vznikla tedy verze `cbgServer.jar`, která byla odlazena a uvedena do provozu. Další funkcí, kterou geolokační server plní, je pravidelné spouštění kalibračního módu, což zajišťuje stále aktuální data pro výpočet přímek bestline. Každé 2 hodiny je prováděno volání metody příkazem:

```
java -jar {METHOD_DIR}/nemecekcbg/cbgServer.jar -b -u {USER}
-pki {PKI} -L {METHOD_DIR}/nemecekcbg/files/test/serveryarray.txt
```

Za proměnné `{METHOD_DIR}`, `{USER}` a `{PKI}` jsou doplněny potřebné vstupní hodnoty. O to se stará sám geolokační server.

6 VÝSLEDKY MĚŘENÍ A POROVNÁNÍ METOD

6.1 Měření pomocí metody CBG

Pro účely měření a testování přesnosti metody CBG byl vytvořen soubor cílů obsahující 65 serverů. Jejich seznam lze nalézt v příloze B. Geolokační měření probíhala v období od 23. 4. 2014 do 9. 5. 2014. V této kapitole budou prezentovány pouze výsledky měření za pomoci algoritmu CBG. Poloha jednotlivých cílů byla během uvedené doby několikrát odhadována a z jednotlivých pokusů byly pro učely této kapitoly vybrány celkem 3 měření.

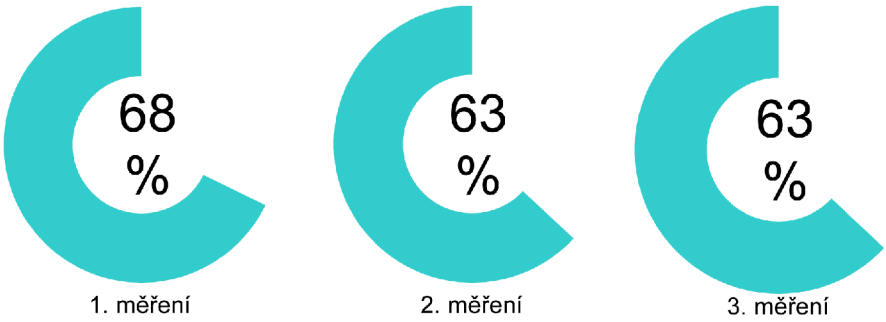
Výsledky těchto pokusů zobrazuje tabulka v příloze D. Jednotlivá měření lze rozdělit podle týdenní doby, ve které byla provedena. V případě první lokalizace se jednalo o měření na počátku období v nočních hodinách pracovního dne, naopak druhý pokus byl prováděn v průběhu víkendového dne. Třetí měření pak probíhalo na konci období v dopoledních hodinách a stejně jako v prvním případě se jednalo o běžný pracovní den.

Velikost chyby odhadu se pohybuje od hodnoty 0 km až po 3858,14 km. V tabulce 6.1 jsou uvedeny výsledky rozdělené pro jednotlivá měření. Z hodnot je patrné, že nejlepších výsledků podle mediánu chyby odhadu dosahuje měření č. 1.

Tab. 6.1: Statistiky jednotlivých měření.

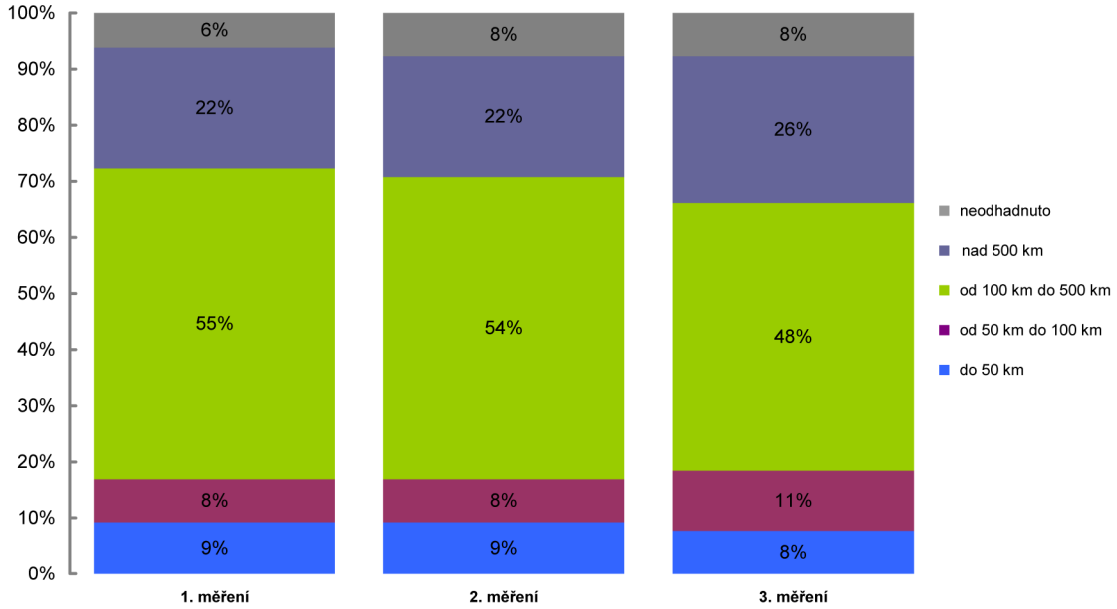
Měření	Minimální chyba [km]	Maximální chyba [km]	Průměrná chyba [km]	Medián [km]
1.	0,00	3548,95	371,02	200,64
2.	0,00	1436,27	348,19	231,70
3.	0,00	3858,14	443,92	240,17

Podrobnějším rozborem jednotlivých hodnot lze vytvořit další statistiky. Jednou z nich je úspěšnost metody při provádění odhadu polohy jednotlivých cílů. Hranici, kdy je rozhodnuto, zda byl pokus úspěšný či nikoliv, stanovuje vzdálenost 400 km. Výsledné statistiky pro jednotlivá měření jsou na obrázku 6.1. První měření dosahuje stále nejlepších výsledků s celkem 68 %, zbývající dva pokusy mají shodně 63 % úspěšně odhadnutých zeměpisných souřadnic. Hranice 400 km odpovídá odhadu na úrovni státu, což stále není úplně přesná hodnota. Pro prvotní odhad, který je dále upřesněn jinými metodami geolokace, je to dostačující údaj.



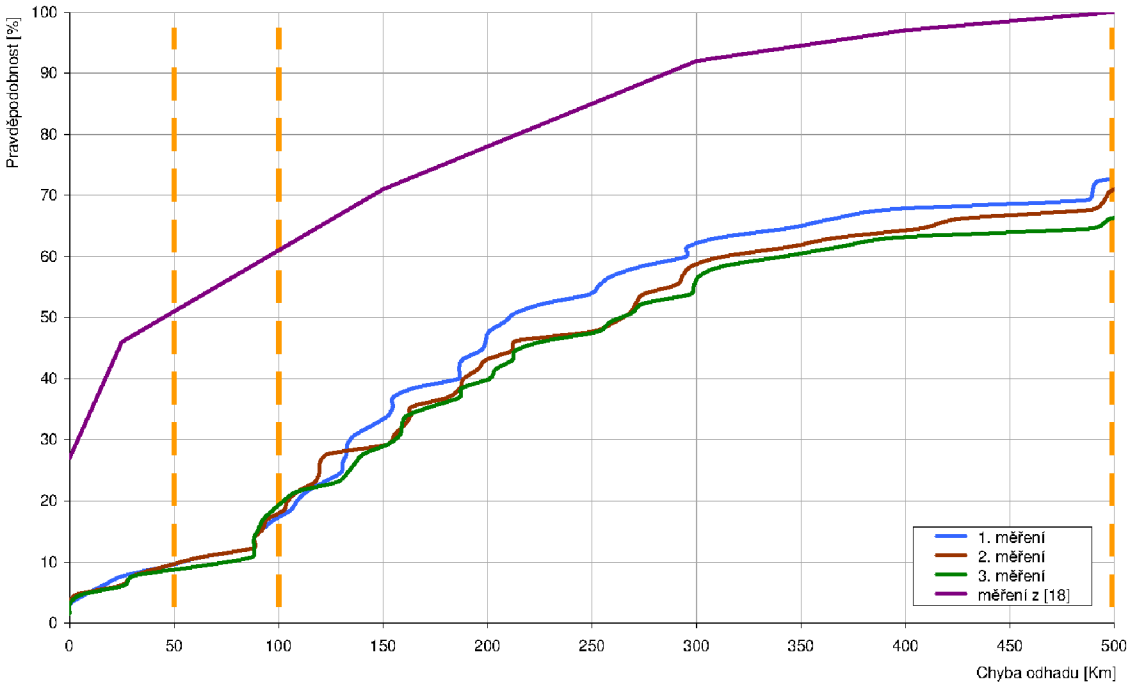
Obr. 6.1: Úspěšnost geolokace cílů (odhad polohy do 400 km).

Počet stanic, které metoda nedokázala vůbec lokalizovat, nepřekročil hodnotu 10 %. Jednalo se především o chyby, kdy nebylo možné vytvořit průnik kruhových oblastí. V příloze D jsou tyto chyby označeny *prunik_err* nebo *polygon_err*. Program je přizpůsobený tak, aby zajistil co nejvyšší pravděpodobnost odhadu polohy. Počet serverů, které vytváří hranice vzdáleností, může klesnout až na 3 referenční body. I takto malý počet zajistí vykreslení výsledného polygonu na úkor přesnosti odhadu. Obrázek 6.2 je vyjádřením procentuálního zastoupení hodnot odchylky pro jednotlivá měření. Z grafů je patrné, že se odchylka nejčastěji pohybuje v rozmezí od 100 km do 500 km. Procentuální zastoupení této kategorie je ve všech měřeních téměř 50 %. Z tohoto rozdělení však není jasné, zda je zastoupeno více hodnot z horní hranice intervalu, či převládají odchylky nižší.



Obr. 6.2: Chyba odhadu polohy cíle.

Poslední statistikou je kumulativní distribuční funkce (*anglicky Cumulative Distribution Function, CDF*), která každému reálnému číslu přiřazuje pravděpodobnost, že náhodná veličina nabude hodnoty menší nebo rovné než toto číslo.[24] Na obrázku 6.3 je rozložení pravděpodobnosti pro všechna 3 měření a současně zobrazuje výsledky dosažené v [18].



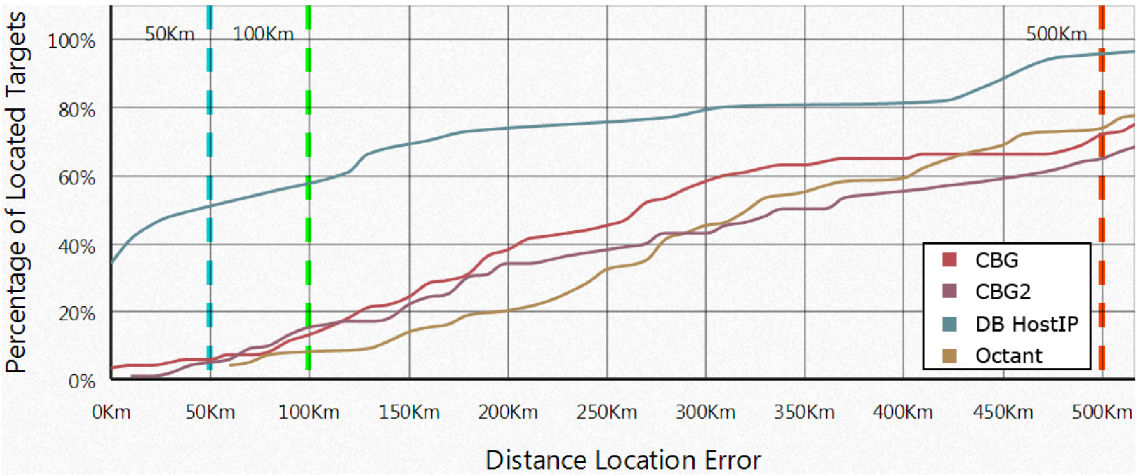
Obr. 6.3: Kumulativní distribuční funkce metody CBG.

Grafy jsou omezeny maximální hodnotou vzdálenosti 500 km. Vyšší odchylky měření již neodpovídají přesnosti na úrovni státu a pouze snižují rozlišení jednotlivých průběhů CDF. Pro všechna měření lze následně určit, kolik cílů bylo odhadnuto s hledanou odchylkou. Z výsledků vyplývá, že odhad chyby do 500 km je proveden průměrně pro 72 % cílů. S chybou menší než 100 km byla pozice odhadnuta u 18 % cílů.

Grafy implementované metody mají velmi podobné průběhy, ale od výsledků v [18] se odlišují výrazným způsobem. Autoři pro svá měření využívají celkem 57 referenčních bodů rozmístěných na území Spojených států amerických (24), Evropy (24) a Asie (5). Za pomoci nich lokalizovali 43 evropských cílů. Průměrná hodnota odchylky činila 106 km a medián 42 km. Výrazný rozdíl mezi CBG a [18] je způsoben zejména počtem použitých referenčních bodů. Dalším důvodem může být odlišný způsob implementace obou metod. Není také známo, které servery sítě PlanetLab byly využity a totéž platí o výběru testovaných cílů. Srovnání je tedy značně neobjektivní.

6.2 Srovnání CBG s ostatními metodami

Geolokační server nabízí srovnání naměřených výsledků. Myšlenkou tedy bylo porovnat úspěšnost a přesnost navržené metody s ostatními aktivními technikami. V době tvorby práce byly na serveru k dispozici metody *DB HostIP*, *Octant* a další verze CBG (bude označena *CBG2*). Výsledky měření všech metod lze zobrazit do grafů v podobě kumulativní distribuční funkce nebo histogramu odchylek. Srovnání v podobě statistik obsahujících např. údaje o minimální a maximální chybě odhadu nebo mediánu server bohužel nenabízí. Na obrázku 6.4 jsou průběhy CDF pro všechny nabízené metody.



Obr. 6.4: Kumulativní distribuční všech metod.

Z grafu vyplývá, že pasivní metoda DB HostIP dosahuje nejlepších výsledků v celém rozsahu. Do odchylky 50 km je jako druhá v pořadí vytvořená CBG následovaná CBG2, metoda Octant s touto přesností vůbec neměří.

Na hodnotě asi 70 km je přehozeno pořadí obou metod CBG, které je zachováno do hodnoty 115 km. V rozsahu od 115 km do 280 km je tedy pořadí: 1. CBG, 2. CBG2 a 3. Octant.

Další změna nastává na hranici 280 km, kdy se Octant stává účinnější než CBG2. Na první místo se však dostává až při odchylkách vyšších než 435 km.

Všechny zmíněné metody porovnávaly svou přesnost na výše popsaném souboru 65 testovacích cílů.

7 ZÁVĚR

Diplomová práce se věnuje možnostem geolokace zařízení v Internetu a s tím spojených témat. Hlavním cílem byla implementace algoritmu Constraint-Based Geolocation, který patří do aktivních metod vyhledávání stanic. Teoretický rozbor dále popisuje zdroje zpoždění v síti a možnosti jeho měření. Probíraným tématem je také experimentální síť PlanetLab, která byla využita pro potřeby měření a lokalizace. Část věnovaná implementaci se věnuje popisu vytvořeného programu a rozboru výsledků měření.

Na úvod bylo v práci popsáno zpoždění v síti. Za zdroje zpoždění v komunikačním kanále lze považovat jak koncové stanice, tak i veškeré aktivní prvky mezi nimi. Dalším zdrojem zpoždění je samotné přenosové médium. Celkové zpoždění může také dále ovlivňovat směrovací politika mezi autonomními systémy. Kapitola se dále věnuje způsobům měření zpoždění. Nejčastěji je využíváno zpoždění RTT.

V další kapitole byla zmíněna samotná geolokace, konkrétně IP geolokace. Jednotlivé metody jsou rozděleny na pasivní, využívající porovnávání záznamů v databázích a aktivní, které odhadují polohu cíle na základě měření. Byly popsány metody ShortestPing, GeoPing, Speed of Internet, Geoweight a Octant jako vybraní zástupci této skupiny. Metodě Constraint-Based Geolocation byla věnována samostatná kapitola. Zde bylo vysvětleno, jakým způsobem je prováděn převod zpoždění na hranice vzdáleností a dále pak důvody a princip automatické kalibrace referenčních bodů. V neposlední řadě bylo uvedeno, jakým způsobem se provádí konečný odhad cílové pozice.

Pro praktickou část byly využity servery náležící do sítě PlanetLab, neboť je známa jejich poloha a je možno je využít k měření RTT. Z tohoto důvodu je v práci kapitola věnovaná historii a struktuře této experimentální sítě.

Samotná implementace byla popsána v kapitole 5. Pro metodu CBG bylo z databáze serverů sítě PlanetLab vybráno celkem 20 referenčních bodů umístěných na území Evropy. Takto vytvořený soubor využívá metoda pro měření a geolokaci. Výsledkem práce je metoda CBG, představovaná Java aplikací `cbgWorkstation.jar` a bash skripty `ping.sh`, `pingtar.sh`, `komplet.sh` a `kopirovani.sh`. Program umožňuje zpracování seznamu referenčních bodů ve formátu JSON. Vytvořené skripty pro měření zpoždění mezi servery navzájem a mezi servery a hledaným cílem byly nahrány na jednotlivé referenční body. Další pak zajišťuje komunikaci se sítí PlanetLab a provádí kolekci naměřených dat. Posledním vytvořeným bash skriptem je `kopirovani.sh`, který jednak aktualizuje data potřebná pro kalibrační mód, ale také zvyšuje univerzálnost metody. Umožňuje totiž kopírovat potřebná data na nové servery sítě PlanetLab. Program výsledky geolokace zobrazuje v grafickém okně a pro výpis vypočtených údajů využívá terminálový výstup.

Vytvořená metoda byla dále modifikována a přizpůsobena pro nasazení na geolokační server umístěný na Ústavu telekomunikací Fakulty elektrotechniky a komunikačních technologií Vysokého učení technického v Brně. Zde mohou být její výsledky porovnávány s jinými metodami.

Pro účely měření polohy stanic v Internetu byl vytvořen soubor testovacích cílů obsahující 65 serverů se známou polohou. Pomocí nich byla otestována přesnost a funkce hotového programu.

Zavěrečná kapitola je věnována výsledkům měření. Ty jsou prezentovány za pomoci statistik a grafů. Velikost chyby odhadu se pohybovala od hodnoty 0 km až po 3858,14 km, medián chyby nejlepšího měření je roven 200,64 km. Byla stanovena vzdálenost 400 km, která rozhodovala, zda byl odhad polohy úspěšný či nikoliv. Pod touto hranicí bylo lokalizováno průměrně 65 % cílů. Celkově pak počet neodhadnutých souřadnic nepřekročil 10 %.

Výsledky měření byly také porovnány s dalšími geolokačními metodami DB HostIP, Octant a další verzí CBG. Srovnání bylo provedeno za pomoci vytvořeného souboru testovacích cílů. Výsledky jsou pak prezentovány pomocí CDF.

Dosavadní výsledky této diplomové práce a vytvořený program mohou dále sloužit pro odhad polohy pomocí geolokačního serveru nebo jako aplikace pro samostatnou stanici. Program je přizpůsoben tak, aby bylo možné přidávat nové referenční body nebo vyměnit ty současné. Není také zavislý na aktuálním projektu *cesnet_feec* a je možné ho kdykoliv v budoucnosti využívat pro jiná měření a uživatele sítě PlanetLab. Z těchto poznatků lze říci, že byly splněny všechny body zadání.

LITERATURA

[1] BALEJ, Jiří a Dan KOMOSNÝ. Zdroje zpoždění při komunikaci v Internetu. *Elektrorevue* [online]. 2010 [cit. 2013-10-13]. Dostupné z: <http://www.elektrorevue.cz/cz/clanky/komunikacni-technologie/15/zdroje-zpozdeni-pri-komunikaci-v-internetu>.

[2] BOVY, C. J., et al. *Analysis of End-to-end Delay Measurement in Internet* [online]. PAM 2002, 2002 [cit. 2013-11-10]. Dostupné z: <http://www.pamconf.net/2002/Analysis_of_End_to_end_Delay_Measurements_in_Internet.pdf>.

[3] BALEJ, Jiří. *Simulace zpoždění při přenosu dat mezi stanicemi v IP sítích:* diplomová práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. 2010. 64 s. Vedoucí práce byl doc. Ing. Dan Komosný, Ph.D.

[4] NOVOTNÝ, Vít. *Architektura sítí.* Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií. 2002. 136 s.

[5] PERCACCI, R. VESPIGNANI, A. Scale-free behavior of the Internet global performance. *The European Physical Journal B - Condensed Matter.* 32(4):411-414. 2003.

[6] BRADNER, S. *Benchmarking Terminology for Network Interconnection Devices.* RFC 1242 [online]. July 1991. Dostupné z: <http://www.rfc-editor.org/rfc/rfc1242.txt>.

[7] SHALUNOV, S. et al. *A One-way Active Measurement Protocol (OWAMP).* RFC 4656 [online]. September 2006. Dostupné z: <http://www.rfc-editor.org/rfc/rfc4656.txt>.

[8] POSTEL, J. *Internet Control Message Protocol.* STD 5. RFC 792 [online]. September 1981. Dostupné z: <http://www.rfc-editor.org/rfc/rfc792.txt>.

[9] JAROŠOVÁ L. *Měření vzdáleností stanic prostřednictvím ICMP protokolu v IP sítích:* bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. 2008. 63 s. Vedoucí práce Ing. Radim Burget

[10] KATZ-BASSETT, Ethan. et al. *Towards IP Geolocation Using Delay and Topology Measurements.* Internet Measurement Conference. 2006 [cit. 2013-12-02]. 2006. Dostupné z: <http://www-bcf.usc.edu/~katzbass/papers/geoloc-imc06.pdf>.

[11] VERNER, Lukáš a Dan KOMOSNÝ. Geolokace síťových zařízení v internetových sítích. *Elektrorevue* [online]. 2011 [cit. 2013-12-02]. Dostupné z: <http://www.elektrorevue.cz/cz/clanky/komunikacni-technologie/65/geolokace-sitovych-zarizeni-v-internetovych-sitich/>.

[12] GARGANO, J. and K. WEISS. *Whois and Network Information Lookup Service, Whois++*. RFC 1834 [online]. August 1995. Dostupné z: <http://www.rfc-editor.org/rfc/rfc1834.txt>.

[13] DOLEŽEL, P. *Současné možnosti nalezení fyzické pozice stanice v Internetu*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií. 2011. 54 s. Vedoucí bakalářské práce Ing. Jiří Balej.

[14] HOLZHAUER, Florian. *IP Geolocation*. 2007. [cit. 2013-12-02]. 11s. Dostupné z: <http://www.net.t-labs.tu-berlin.de/teaching/ss07/IM_seminar/aa_b1.pdf>.

[15] PADMANABHAN, Venkata N. SUBRAMANIAN, L. *An Investigation of Geographic Mapping Techniques for Internet Hosts*. In Internet Measurement Conference [online]. 2001. [cit. 2013-12-06]. Dostupné z: <http://conferences.sigcomm.org/sigcomm/2001/p14-pabmanabhan.pdf>.

[16] ARIF, M. J., et al. *GeoWeight: internet host geolocation based on a probability model for latency measurements*. ACSC '10. 2010. ISBN: 978-1-920682-83-5

[17] WONG, B., et al. *Geolocalization on the Internet through Constraint Satisfaction*. Proc. USENIX WORLDS. 2006

[18] GUEYE, B. ZIVIANI, A., CROVELLA, M., FDIDA, S. *Constraint-Based Geolocation of Internet Hosts*. Networking, IEEE/ACM Transactions on, vol.14, no.6. 2006. Dostupné z: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4032725&isnumber=4032714>.

[19] BOURKE, Paul. *Polygon Area and Centroid* [online]. 1998. [cit. 2013-12-10]. Dostupné z: <http://www.seas.upenn.edu/~sys502/extra_materials/Polygon%20Area%20and%20Centroid.pdf>.

[20] *PlanetLab / An open platform for developing, deploying, and accessing planetary-scale services* [online]. 2007 [cit. 2013-12-12]. About PlanetLab. Dostupné z: <http://planet-lab.org/>.

[21] VINCENTI, T. *Geodetic inverse solution between antipodal points (Technical report). DMAAC Geodetic Survey Squadron*. 1975. Dostupné z: <http://geographiclib.sourceforge.net/geodesic-papers/vincenty75b.pdf>.

[22] OSBORNE, Peter. *The normal and transverse Mercator projections on the sphere and the ellipsoid with full derivations of all formulae*. 2013 . Dostupné z: <http://www.mercator99.webspace.virginmedia.com/mercator.pdf>.

[23] RAPP, H. R. *Geometric geodesy, part I*. 1991. Dostupné z: <http://kb.osu.edu/dspace/handle/1811/24333>.

[24] LIMPOUCH, J. *Úvod do počtu pravděpodobnosti* [online]. 2000. [cit. 2014-04-25]. Dostupné z: <http://www-troja.fjfi.cvut.cz/~limpouch/sigdat/pravdh/node6.html>.

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

AS	Autonomní systém
AWT	Abstract Window Toolkit
BTS	Base transceiver station
CBG	Constraint-Based Geolocation
CDF	Cumulative distribution function
CESNET	Czech Education and Scientific NETwork
CSV	Comma-separated values
GNU	GNU's Not Unix!
GPS	Global Positioning System
IANA	Internet Assigned Numbers Authority
ICMP	Internet Control Message Protocol
IP	Internet Protocol
IPv4	Internet Protocol version 4
JSON	JavaScript Object Notation
NAT	Network address translation
OS	Operační systém
OSI	Open Systems Interconnection
PLC	PlanetLab Central
PLE	PlanetLab Europe
RB	Referenční bod
RFC	Request for Comments
RIR	Regional Internet registry
RTT	Round-Trip Time
SOI	Speed of Internet
SSH	Secure Shell
TCP	Transmission Control Protocol
VPN	Virtual private network

SEZNAM PŘÍLOH

A Seznam použitých referenčních bodů **51**

B Seznam cílů pro geolokaci **52**

C Textový výstup programu **54**

D Výsledky měření **57**

E Obsah přiloženého CD **59**

A SEZNAM POUŽITÝCH REFERENČNÍCH BODŮ

Doménový název serveru	IP adresa	Zeměpisná šířka [°]	Zeměpisná délka [°]
host2.planetlab.informatik.tu-darmstadt.de	130.83.166.245	49,53	8,4,2014
planetlab1.tmit.bme.hu	152.66.245.161	47,4725	19,6,2014
planetlab-4.imperial.ac.uk	193.63.75.21	51,1	-0,1
ple1.cesnet.cz	195.113.161.13	50,102	14,3916
planetlab1.cs.vu.nl	130.37.193.141	52,35	4,9,2014
prata.mimuw.edu.pl	193.0.109.25	52,211	20,981
inriarennes1.irisa.fr	131.254.208.12	48,1155	-1,65071
planetlab1.utt.fr	194.254.215.11	48,2693	4,06739
planet-lab-node1.netgroup.uniroma2.it	160.80.221.37	41,8556	1,12,1943
planetlab1.eecs.jacobs-university.de	212.201.44.81	53,04	1,8,1949
onelab2.info.ucl.ac.be	130.104.72.201	50,6833	4,61667
planetlab2.lkn.ei.tum.de	138.246.99.250	48,1493	1,11,1969
planetlab2.urv.cat	193.144.21.131	41,07	1,1,2015
planetlab2.unineuchatel.ch	192.42.43.23	46,59	1,6,1956
planetlab2.extern.kuleuven.be	193.190.168.51	50,8625	4,68599
planetlab-13.e5.ijs.si	193.138.2.13	46,0423	14,488
planetlabeu-2.tssg.org	193.1.201.27	52,2441	-7,15825
ple2.tu.koszalin.pl	62.108.171.76	54,2047	16,1972
planetlab2.rd.tut.fi	193.166.167.5	61,449	23,855
merkur.planetlab.haw-hamburg.de	141.22.213.34	53,5686	10,0386

B SEZNAM CÍLŮ PRO GEOLOKACI

Doménový název serveru	Zeměpisná šířka [°]	Zeměpisná délka [°]
2009.nwerc.eu	49,45	11,0833
69lovesongs.info	51,3758	-2,36056
80.167.187.0	51,69777	10,58423
91.123.196.1	56,15	15,59
93.157.192.1	44,96	18,3,2014
aaisp.net.uk	52,21068	0,09259
aguila1.lsi.upc.edu	41,3897	2,11273
alpha.science.unitn.it	49,38691817	9,603633495
bosna.utic.net.ba	46,61639	14,265
carleos.epv.uniovi.es	43,5244	-5,62472
dfn-ple1.x-win.dfn.de	49,575	11,0287
disch-online.de	49,60106797	10,1043328
dns1.uni-hannover.de	52,37	1,9,1974
dns1.up.pt	37,3801	-5,9916
dplanet1.uoc.edu	41,3909	2,16791
elrino.co.uk	52,21068	0,09259
games.sugardas.lt	55,6	26,44
gc2x22d.nordwal.de	48,78167	9,17528
golias.ruk.cuni.cz	49,22684	16,59615
host2.planetlab.informatik.tu-darmstadt.de	47,37642	8,5481
chronos.disy.inf.uni-konstanz.de	51,12	1,10,1954
iraplab1.iralab.uni-karlsruhe.de	49,015	8,405
node1pl.planet-lab.telecom-lille1.eu	50,6103	3,13525
ns.cvt.stuba.sk	48,15	17,11,2014
ns.hs-nb.de	53,56	13,26
ns.informatik.uni-bremen.de	52,45545	13,29774
ns1.editions.it	45,12506	7,67369
ns1.lut.fi	56,39621	17,93957
ns1.nostra.by	53,31	31,44
ns1.ugd.edu.mk	41,75	22,2,2014
ns1.u-strasbg.fr	48,66461	6,15788
ns1.wit.ie	53,30751	-6,22226
ns2.nbg.de	49,78806	9,93528
ns2.utu.fi	60,46939285	8,404745419
pl2.uni-rostock.de	54,13	12,145
planet1.unipr.it	44,8009	1,10,1955
planet2.inf.tu-dresden.de	51,026	13,725
planetlab1.ci.pwr.wroc.pl	51,1	16,93
planetlab1.diku.dk	55,7014	12,12,2014
planetlab1.ifi.uio.no	59,93	1,10,1975

Doménový název serveru	Zeměpisná šířka [°]	Zeměpisná délka [°]
planetlab1.informatik.uni-wuerzburg.de	49,8	1,9,1993
planetlab-1.iscte.pt	38,43	-9,08
planetlab-1.research.netlab.hut.fi	60,19	24,93
planetlab1.tmit.bme.hu	47,47443	19,06172
planetlab1.urv.cat	41,07	1,1,2015
planetlab1.xeno.cl.cam.ac.uk	52,2112	0,0917
planetlab2.cs.uit.no	69,6813	18,977
planetlab2.ionio.gr	39,6205	19,9148
planetlab2.polito.it	45,07	1,7,1967
planetlab2.s3.kth.se	59,3495	18,067
planetlab2.u-strasbg.fr	48,5237	7,73833
planetlab3.upc.es	41,3897	2,11273
planetlab3.xeno.cl.cam.ac.uk	52,2112	0,0917
planetlab4.hiit.fi	60,1	25
planetlab-node-01.ucd.ie	53,3092	-6,22169
planetvs2.informatik.uni-stuttgart.de	48,7835	9,17517
ple1.dmcs.p.lodz.pl	51,7461	19,4556
pluton.utt.fr	48,26916	4,06709
rektor.uniri.hr	45,33	14,44
utet.ii.uam.es	40,4363	-3,69413
w3projns.ze.tu-muenchen.de	52,43143	13,27514
www.ens.fr	48,52282	4,14744
www.lkb.ens.fr	48,8414	2,34556
www.whra.org.uk	52,5903	0,0583333
zoi.di.uoa.gr	37,9684	23,7669

C TEXTOVÝ VÝSTUP PROGRAMU

Servery: 19 ks

Server	[Lat]	[Lon]
host2.planetlab.informatik.tu-darmstadt.de	49,530000	8,400000
planetlab1.tmit.bme.hu	47,472500	19,600000
planetlab-4.imperial.ac.uk	51,100000	-0,100000
ple1.cesnet.cz	50,102000	14,391600
planetlab1.cs.vu.nl	52,350000	4,900000
prata.mimuw.edu.pl	52,211000	20,981000
inriarennes1.irisa.fr	48,115500	-1,650710
planetlab1.utt.fr	48,269300	4,067390
planet-lab-node1.netgroup.uniroma2.it	41,855600	12,430000
planetlab1.eecs.jacobs-university.de	53,040000	8,490000
planetlab2.lkn.ei.tum.de	48,149300	11,690000
planetlab2.urv.cat	41,070000	1,150000
planetlab2.unineuchatel.ch	46,590000	6,560000
planetlab2.extern.kuleuven.be	50,862500	4,685990
planetlab-13.e5.ijs.si	46,042300	14,488000
planetlabeu-2.tssg.org	52,244100	-7,158250
ple2.tu.koszalin.pl	54,204700	16,197200
planetlab2.rd.tut.fi	61,449000	23,855000
merkur.planetlab.haw-hamburg.de	53,568600	10,038600

Spoje: 361 ks

Spojeni	vzdalenost[Km]	zpozdeni[ms]
host2.planetlab.informatik.tu-darmstadt.de->host2.planetlab.informatik.tu-darmstadt.de	0,000000	0,016000
host2.planetlab.informatik.tu-darmstadt.de->planetlab1.tmit.bme.hu	857,795000	28,165000
host2.planetlab.informatik.tu-darmstadt.de->planetlab-4.imperial.ac.uk	629,694000	19,793000
host2.planetlab.informatik.tu-darmstadt.de->ple1.cesnet.cz	435,755000	19,942000
host2.planetlab.informatik.tu-darmstadt.de->planetlab1.cs.vu.nl	398,571000	7,678000
host2.planetlab.informatik.tu-darmstadt.de->prata.mimuw.edu.pl	933,045000	32,118000
host2.planetlab.informatik.tu-darmstadt.de->inriarennes1.irisa.fr	753,996000	21,382000
host2.planetlab.informatik.tu-darmstadt.de->planetlab1.utt.fr	347,156000	27,082000
host2.planetlab.informatik.tu-darmstadt.de->planet-lab-node1.netgroup.uniroma2.it	908,550000	29,189000
host2.planetlab.informatik.tu-darmstadt.de->planetlab1.eecs.jacobs-university.de	390,551000	24,513000
host2.planetlab.informatik.tu-darmstadt.de->planetlab2.lkn.ei.tum.de	286,143000	36,653000
host2.planetlab.informatik.tu-darmstadt.de->planetlab2.urv.cat	1097,620000	59,028000
host2.planetlab.informatik.tu-darmstadt.de->planetlab2.unineuchatel.ch	354,479000	9,951000
host2.planetlab.informatik.tu-darmstadt.de->planetlab2.extern.kuleuven.be	303,751000	13,477000
host2.planetlab.informatik.tu-darmstadt.de->planetlab-13.e5.ijs.si	598,400000	16,709000
host2.planetlab.informatik.tu-darmstadt.de->planetlabeu-2.tssg.org	1133,170000	31,952000
host2.planetlab.informatik.tu-darmstadt.de->ple2.tu.koszalin.pl	746,840000	33,328000
host2.planetlab.informatik.tu-darmstadt.de->planetlab2.rd.tut.fi	1638,893000	37,971000
host2.planetlab.informatik.tu-darmstadt.de->merkur.planetlab.haw-hamburg.de	463,449000	18,490000
inriarennes1.irisa.fr->host2.planetlab.informatik.tu-darmstadt.de	753,996000	21,433000
inriarennes1.irisa.fr->planetlab1.tmit.bme.hu	1588,682000	39,122000
inriarennes1.irisa.fr->planetlab-4.imperial.ac.uk	350,331000	13,549000
inriarennes1.irisa.fr->ple1.cesnet.cz	1189,496000	28,977000
inriarennes1.irisa.fr->planetlab1.cs.vu.nl	663,051000	18,462000
inriarennes1.irisa.fr->prata.mimuw.edu.pl	1672,276000	47,627000
inriarennes1.irisa.fr->inriarennes1.irisa.fr	0,000000	0,017000
inriarennes1.irisa.fr->planetlab1.utt.fr	425,371000	18,330000
inriarennes1.irisa.fr->planet-lab-node1.netgroup.uniroma2.it	1307,343000	38,114000
inriarennes1.irisa.fr->planetlab1.eecs.jacobs-university.de	901,943000	40,365000
inriarennes1.irisa.fr->planetlab2.lkn.ei.tum.de	991,766000	41,625000
inriarennes1.irisa.fr->planetlab2.urv.cat	813,741000	43,147000
inriarennes1.irisa.fr->planetlab2.unineuchatel.ch	642,761000	16,133000
inriarennes1.irisa.fr->planetlab2.extern.kuleuven.be	551,176000	18,415000
inriarennes1.irisa.fr->planetlab-13.e5.ijs.si	1244,684000	44,242000
inriarennes1.irisa.fr->planetlabeu-2.tssg.org	604,340000	26,034000
inriarennes1.irisa.fr->ple2.tu.koszalin.pl	1414,887000	48,920000
inriarennes1.irisa.fr->planetlab2.rd.tut.fi	2186,559000	50,774000
inriarennes1.irisa.fr->merkur.planetlab.haw-hamburg.de	1020,614000	34,539000
merkur.planetlab.haw-hamburg.de->host2.planetlab.informatik.tu-darmstadt.de	463,449000	18,490000
merkur.planetlab.haw-hamburg.de->planetlab1.tmit.bme.hu	957,420000	22,936000
merkur.planetlab.haw-hamburg.de->planetlab-4.imperial.ac.uk	742,864000	23,039000
merkur.planetlab.haw-hamburg.de->ple1.cesnet.cz	488,481000	15,758000
merkur.planetlab.haw-hamburg.de->planetlab1.cs.vu.nl	370,883000	15,240000
merkur.planetlab.haw-hamburg.de->prata.mimuw.edu.pl	751,044000	34,429000
merkur.planetlab.haw-hamburg.de->inriarennes1.irisa.fr	1020,614000	34,530000
merkur.planetlab.haw-hamburg.de->planetlab1.utt.fr	723,230000	39,471000

.
.
...

```
Body pro server host2.planetlab.informatik.tu-darmstadt.de:
0.0, 0.016
857.795, 28.165
629.694, 19.793
435.755, 19.942
398.571, 7.678
933.045, 32.118
753.996, 21.382
347.156, 27.082
908.55, 29.189
390.551, 24.513
286.143, 36.653
1097.62, 59.028
354.479, 9.951
303.751, 13.477
598.4, 16.709
1133.17, 31.952
746.84, 33.328
1638.893, 37.971
463.449, 18.49
Pocet: 19

Body pro server planetlab1.tmit.bme.hu:
857.795, 28.176
0.0, 0.021
1483.817, 29.024
481.46, 7.613
1184.086, 21.362
536.296, 39.134
1588.682, 39.116
1163.387, 43.973
843.627, 27.777
1003.721, 21.249
596.928, 22.41
1630.373, 61.372
994.85, 30.094
1148.692, 25.095
421.571, 7.693
1983.211, 41.613
786.107, 40.448
1579.196, 51.148
957.42, 22.933
Pocet: 19

Body pro server planetlab-4.imperial.ac.uk:
629.694, 19.867
1483.817, 29.149
0.0, 0.035
1030.268, 21.911
372.333, 6.512
1458.919, 40.556
350.331, 13.554
435.254, 18.507
1403.998, 38.368
626.852, 28.959
911.618, 40.708
1119.002, 43.068
699.816, 16.09
337.078, 6.885
1211.664, 44.198
504.371, 13.752
1152.722, 41.866
1860.565, 39.118
742.864, 23.014
Pocet: 19

Body pro server ple1.cesnet.cz:
435.755, 19.934
481.46, 7.615
1030.268, 21.938
0.0, 0.028
707.989, 14.285
517.008, 20.521
1189.496, 28.937
779.02, 33.911
929.022, 34.031
.
.
...
```

```
Bestlines pro jednotlive servery:
host2.planetlab.informatik.tu-darmstadt.de:  0.019223660224240113 + 0.016
planetlab1.tmit.bme.hu:  0.015768689699094934 + 0.021
planetlab-4.imperial.ac.uk:  0.017395721195181717 + 0.035
ple1.cesnet.cz:  0.015758304629482776 + 0.02799999999999958
planetlab1.cs.vu.nl:  0.017473608475505983 + 0.036999999999999034
prata.mimuw.edu.pl:  0.025869167159786245 + 0.019
inriarennes1.irisa.fr:  0.02321318431001532 + 0.017
planetlab1.utt.fr:  0.028773794088267853 + 0.024
planet-lab-node1.netgroup.uniroma2.it:  0.026968647488108417 + 0.021
planetlab1.eecs.jacobs-university.de:  0.021198113684977735 + 0.047999999999998266
planetlab2.lkn.ei.tum.de:  0.02718475576575388 + 0.01699999999999946
planetlab2.urv.cat:  0.029238475631109085 + 0.02
planetlab2.unineuchatel.ch:  0.02296033151703822 + 0.02499999999999858
planetlab2.extern.kuleuven.be:  0.021081779594707348 + 0.031
planetlab-13.e5.ijs.si:  0.018231805049954704 + 0.017999999999999794
planetlabeu-2.tssg.org:  0.020976087912483576 + 0.045999999999999375
ple2.tu.koszalin.pl:  0.025699482700181204 + 0.02999999999999936
planetlab2.rd.tut.fi:  0.020979649564227598 + 0.036
merkur.planetlab.haw-hamburg.de:  0.02377590251509458 + 0.023


Vzdalenosti cile od jednotlivych serveru:
Server host2.planetlab.informatik.tu-darmstadt.de, kruznice 704.809 Km
Server inriarennes1.irisa.fr, kruznice 1556.228 Km
Server merkur.planetlab.haw-hamburg.de, kruznice 834.122 Km
Server planetlabeu-2.tssg.org, kruznice 1836.091 Km
Server planet-lab-node1.netgroup.uniroma2.it, kruznice 907.684 Km
Server planetlab1.cs.vu.nl, kruznice 1045.749 Km
Server planetlab1.eecs.jacobs-university.de, kruznice 858.048 Km
Server planetlab1.tmit.bme.hu, kruznice 267.112 Km
Server planetlab1.utt.fr, kruznice 1415.385 Km
Server planetlab-13.e5.ijs.si, kruznice 442.688 Km
Server planetlab2.extern.kuleuven.be, kruznice 1031.886 Km
Server planetlab2.lkn.ei.tum.de, kruznice 713.93 Km
Server planetlab2.rd.tut.fi, kruznice 2287.312 Km
Server planetlab2.unineuchatel.ch, kruznice 1178.337 Km
Server planetlab2.urv.cat, kruznice 1992.409 Km
Server planetlab-4.imperial.ac.uk, kruznice 1493.298 Km
Server ple1.cesnet.cz, kruznice 341.534 Km
Server ple2.tu.koszalin.pl, kruznice 704.178 Km
Server prata.mimuw.edu.pl, kruznice 643.198 Km


Souradnice polygonu [x,y] :
445.993, 367.454
471.238, 337.501
471.238, 337.501
476.187, 340.167
476.187, 340.167
476.224, 342.293
471.48, 366.001
471.48, 366.001
460.159, 372.078
460.159, 372.078
448.105, 372.35
446.496, 372.345
446.496, 372.345
445.993, 367.454

Obsah:541.985
Teziste [x,y]: 463.106, 358.053


Vypoctena pozice [Lat, Lon]: 48.251, 17.242
Skutecna pozice [Lat, Lon]: 48.15, 17.11
Rozdil: 14.897km
```

Výstup programu byl z důvodu rozsáhlé velikosti zkrácen. Kompletní text je možno nalézt na přiloženém CD.

D VÝSLEDKY MĚŘENÍ

Doménový název cíle	skutečná pozice [Lat, Lon]	vypočtená pozice [Lat,Lon]			chyba odhadu [Km]		
		1. měření	2. měření	3. měření	1. měření	2. měření	3. měření
2009.nwerc.eu	49.45, 11.0833	49.717, 9.567	49.733, 9.638	49.734, 9.641	113,611	109,146	108,967
69lovesongs.info	51.3758, -2.36056	50.838, 1.763	50.718, 1.757	50.704, 1.766	294,918	297,844	298,873
80.167.187.0	51.69777, 10.58423	56.052, 9.276	56.098, 9.171	56.083, 9.107	492,184	498,495	497,578
91.123.196.1	56.15, 15.59	58.167, 15.04	57.756, 13.717	57.754, 13.712	227,101	212,090	212,023
93.157.192.1	44.96, 18.3	48.908, 14.256	47.739, 14.489	47.564, 6.85	536,015	425,891	927,954
aaisp.net.uk	52.21068, 0.09259	51.255, 2.556	*polygon_err*	*polygon_err*	200,639	—	—
aguila1.lsi.upc.edu	41.3897, 2.11273	41.07, 1.15	41.07, 1.15	41.07, 1.15	88,181	88,181	88,181
alpha.science.unitn.it	49.38691817165, 9.6036334951721	45.005, 9.681	44.932, 9.656	64.451, -6.613	487,174	495,310	1933,726
bosna.utic.net.ba	46.61639, 14.265	45.742, 14.121	45.779, 14.076	45.842, 14.087	97,846	94,221	87,177
carleos.epv.uniovi.es	43.5244, -5.62472	41.352, 0.892	41.457, 0.995	41.431, 0.959	587,806	590,423	588,927
dfn-ple1.x-win.dfn.de	49.575, 11.0287	49.744, 9.589	49.755, 9.61	49.745, 9.634	105,669	104,330	102,445
disch-online.de	49.601067974306, 10.104332803281	49.51, 8.112	48.921, 8.871	48.336, 10.027	144,524	117,344	140,764
dns1.uni-hannover.de	52.37, 9.74	52.57, 9.154	52.559, 9.149	59.943, 0.675	45,650	45,370	1012,034
dns1.up.pt	37.3801, -5.9916	41.07, 1.15	41.07, 1.15	41.07, 1.15	739,950	739,950	739,950
dplanet1.uoc.edu	41.3909, 2.16791	41.07, 1.15	41.07, 1.15	41.07, 1.15	92,485	92,485	92,485
elrino.co.uk	52.21068, 0.09259	50.769, 1.735	50.808, 1.787	50.711, 1.778	196,823	195,445	203,885
games.sugardas.lt	55.6, 26.44	51.343, 12.214	52.365, 12.639	52.528, 12.625	1054,137	972,306	964,938
gc2x22d.nordwal.de	48.78167, 9.17528	49.601, 7.847	58.309, -0.414	58.266, -0.366	132,912	1233,562	1227,991
golias.ruk.cuni.cz	49.22684, 16.59615	50.102, 14.392	50.117, 14.372	50.102, 14.392	186,548	188,618	186,548
host2.planetlab.informatik.tu-darmstadt.de	47.37642, 8.54810	*polygon_err*	*polygon_err*	*polygon_err*	—	—	—
chronos.disy.inf.uni-konstanz.de	51.12, 10.54	46.781, 6.76	46.77, 6.759	46.815, 6.771	556,100	557,238	552,438
iraplab1.iralab.uni-karlsruhe.de	49.015, 8.405	48.596, 7.826	48.598, 7.826	48.612, 7.844	63,070	62,878	60,853
node1pl.planet-lab.telecom-lille1.eu	50.6103, 3.13525	47.493, 1.559	47.473, 1.567	47.466, 1.566	365,309	367,214	367,976
ns.cvt.stuba.sk	48.15, 17.11	48.192, 17.186	48.175, 17.13	48.176, 17.172	7,303	3,116	5,406
ns.hs-nb.de	53.56, 13.26	53.569, 10.039	53.569, 10.039	53.569, 10.039	213,429	213,429	213,429
ns.informatik.uni-bremen.de	52.45545, 13.29774	53.108, 8.817	53.102, 8.782	53.105, 8.779	310,854	313,054	313,305
ns1.editions.it	45.12506, 7.67369	43.867, 10.436	44.797, 11.077	44.702, 11.028	260,393	270,969	268,987
ns1.lut.fi	56.39621, 17.93957	60.931, 20.01	60.943, 20.143	60.917, 20.146	519,155	522,330	519,542
ns1.nostra.by	53.31, 31.44	52.119, 21.54	52.104, 21.502	52.083, 21.463	681,409	684,405	687,586
ns1.ugd.edu.mk	41.75, 22.2	47.572, 19.19	47.575, 19.154	47.578, 19.104	689,501	690,739	692,514
ns1.u-strasbg.fr	48.66461, 6.15788	47.642, 2.492	47.68, 2.541	48.491, 2.141	295,449	290,430	297,003
ns1.wit.ie	53.30751, -6.22226	51.86, -5.707	51.878, -5.69	51.887, -5.775	164,823	163,072	160,946
ns2.nbg.de	49.78806, 9.93528	49.287, 8.274	48.834, 8.223	48.705, 8.236	132,486	163,613	172,626
ns2.utu.fi	60.46939284534, 8.4047454190894	60.972, 20.129	60.981, 20.107	60.992, 20.206	641,495	640,294	645,692
pl2.uni-rostock.de	54.13, 12.145	53.569, 10.039	53.569, 10.039	53.569, 10.039	152,051	152,051	152,051
planet1.unipr.it	44.8009, 10.55	42.532, 10.898	42.567, 10.923	42.539, 10.898	253,611	249,977	252,856
planet2.inf.tu-dresden.de	51.026, 13.725	53.206, 10.171	53.222, 10.19	53.191, 10.191	343,613	343,910	341,466
planetlab1.ci.pwr.wroc.pl	51.1, 16.93	53.182, 19.084	53.187, 19.073	53.192, 19.057	274,588	274,632	274,559
planetlab1.diku.dk	55.7014, 12.12	55.559, 12.074	55.73, 12.598	55.757, 12.594	16,145	30,212	30,432
planetlab1.ifi.uio.no	59.93, 10.75	58.896, 13.799	59.023, 15.613	58.952, 15.713	207,901	293,502	301,823
planetlab1.informatik.uni-wuerzburg.de	49.8, 9.93	50.007, 9.774	49.972, 9.717	49.987, 9.71	25,558	24,486	26,163
planetlab-1.iscte.pt	38.43, -9.08	40.246, 1.02	41.07, 1.15	40.462, 0.712	893,250	923,778	872,043
planetlab-1.research.netlab.hut.fi	60.19, 24.93	*prunik_err*	*prunik_err*	*prunik_err*	—	—	—

Doménový název cíle	skutečná pozice [Lat, Lon]	vypočtená pozice [Lat,Lon]			chyba odhadu [Km]		
		1. měření	2. měření	3. měření	1. měření	2. měření	3. měření
planetlab1.tmit.bme.hu	47.47443, 19.06172	*prunik_err*	*polygon_err*	*prunik_err*	—	—	—
planetlab1.urv.cat	41.07, 1.15	41.07, 1.15	41.07, 1.15	41.07, 1.15	0,000	0,000	0,000
planetlab1.xeno.cl.cam.ac.uk	52.2112, 0.0917	51.14, -0.647	51.188, -0.612	51.079, -0.674	129,722	123,830	136,630
planetlab2.cs.uit.no	69.6813, 18.977	57.119, 24.664	57.635, 24.344	57.156, 24.445	1427,284	1367,584	1421,206
planetlab2.ionio.gr	39.6205, 19.9148	8.506, 28.397	43.113, 9.242	5.899, 29.544	3548,948	972,603	3858,136
planetlab2.polito.it	45.07, 7.67	44.085, 9.593	44.135, 9.537	43.337, 9.186	187,869	181,023	227,491
planetlab2.s3.kth.se	59.3495, 18.067	59.261, 16.164	59.183, 16.303	58.95, 16.691	108,863	102,308	90,420
planetlab2.u-strasbg.fr	48.5237, 7.73833	47.715, 2.605	47.676, 2.41	47.67, 2.589	392,497	407,787	395,024
planetlab3.upc.es	41.3897, 2.11273	41.07, 1.15	41.07, 1.15	41.07, 1.15	88,181	88,181	88,181
planetlab3.xeno.cl.cam.ac.uk	52.2112, 0.0917	51.135, -0.663	51.218, -0.576	51.122, -0.696	130,597	119,759	132,911
planetlab4.hiit.fi	60.1, 25	*prunik_err*	*polygon_err*	*polygon_err*	—	—	—
planetlab-node-01.ucd.ie	53.3092, -6.22169	52.012, -5.393	51.933, -5.541	51.946, -5.51	154,848	159,967	159,187
planetvs2.informatik.uni-stuttgart.de	48.7835, 9.17517	48.071, 7.38	48.065, 7.374	48.043, 7.353	154,657	155,392	158,051
ple1.dmcs.p.lodz.pl	51.7461, 19.4556	53.293, 17.98	53.293, 17.973	53.299, 17.942	199,157	199,411	201,067
pluton.utt.fr	48.26916, 4.06709	48.269, 4.067	48.269, 4.067	48.269, 4.067	0,027	0,027	0,027
rektor.uniri.hr	45.33, 14.44	46.476, 15.071	47.55, 15.529	47.511, 15.607	136,437	260,568	258,533
utet.ii.uam.es	40.4363, -3.69413	42.559, 1.916	42.33, 1.945	42.426, 1.923	524,278	516,300	518,744
w3projns.ze.tu-muenchen.de	52.43143, 13.27514	48.149, 11.69	48.149, 11.69	48.149, 11.69	489,499	489,499	489,499
www.ens.fr	48.52282, 4.14744	47.519, 1.163	47.478, 0.909	47.81, 3.391	248,989	268,110	97,188
www.lkb.ens.fr	48.8414, 2.34556	47.789, 1.89	47.793, 1.992	47.734, 1.88	121,784	119,465	127,941
www.whra.org.uk	52.5903, 0.0583333	51.054, 1.122	51.047, 1.124	51.027, 1.084	185,977	186,810	187,739
zoi.di.uoa.gr	37.9684, 23.7669	46.865, 9.812	45.227, 9.46	45.561, 9.34	1511,184	1436,271	1462,783

E OBSAH PŘILOŽENÉHO CD

```
/
├── text ..................................... Elektronická verze práce
└── prilohy
    ├── cbg
    │   ├── landmarks ........................ Datasety referenčních bodů
    │   ├── app ............................. Programový adresář a archiv s aplikací
    │   └── src ............................. Zdrojové kódy java a bash skripty
    └── vysledky_mereni
        ├── cile ............................ Datasety použitých cílů
        └── data ............................ Výstupní soubory všech měření
```