



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

NALEZENÍ FYZICKÉ POLOHY STANIC V SÍTI INTERNET POMOCÍ MĚŘENÍ PŘENOSOVÉHO ZPOŽDĚNÍ

GEOLOCATION IN INTERNET USING LATENCY MEASUREMENTS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. PETR HARTH

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. DAN KOMOSNÝ, Ph.D.

BRNO 2012



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Diplomová práce

magisterský navazující studijní obor
Telekomunikační a informační technika

Student: Bc. Petr Harth

ID: 106456

Ročník: 2

Akademický rok: 2011/2012

NÁZEV TÉMATU:

Nalezení fyzické polohy stanic v síti Internet pomocí měření přenosového zpoždění

POKYNY PRO VYPRACOVÁNÍ:

Nastudujte a analyzujte dílčí faktory způsobující zpoždění při přenosu dat mezi stanicemi v IP sítích. Dále se seznámte s experimentální sítí PlanetLab – <http://www.planet-lab.org/>. Nastudujte a naprogramujte lokalizační algoritmus CBG (Constraint-Based Geolocalization). Pomocí uměle vytvořených vstupních dat ověřte správnou činnost realizovaného algoritmu. Algoritmus nasadte v experimentální síti PlanetLab a proveďte lokalizaci stanic ve zvolených zemích Evropy.

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 European 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: 6.2.2012

Termín odevzdání: 24.5.2012

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

Konzultanti diplomové práce:

prof. Ing. Kamil Vrba, CSc.

Předseda oborové rady

ABSTRAKT

Tato diplomová práce se týká praktické realizace algoritmu CBG (*Constraint-based Geolocation*), což je jedna z metod řadících se mezi IP (*Internet Protocol*) geolokační techniky. IP geolokace se zabývá určením zeměpisné polohy počítačové stanice podle její IP adresy. Nejprve jsou popsány faktory způsobující zpoždění při přenosu dat mezi stanicemi v IP sítích. Dále se pojednává o problematice měření tohoto zpoždění. Následně je podrobně vysvětlen pojem IP geolokace se všemi jeho souvislostmi a popis aktivních IP geolokačních metod, což jsou techniky založené na měření výše uvedeného zpoždění.

Následuje stručné seznámení s experimentální sítí PlanetLab, která je využita pro měření geolokačních metod. Dále je popsána tvorba množiny referenčních bodů a cílů, což je další nutný předpoklad pro praktickou realizaci metody. Poté je vysvětlena praktická realizace algoritmu CBG v podobě programu *CBGfinder*. Následně je provedeno ověření správné činnosti na uměle zadaných vstupních datech a je uveden příklad IP geolokace cíle v Internetu.

V neposlední řadě jsou uvedeny výsledky měření algoritmu CBG, kdy je proveden rozbor vývoje parametrů přímky *Bestline* na určitém uzlu sítě PlanetLab v průběhu jednoho měsíce, dále je diskutována chyba odhadu zeměpisné polohy a rychlost výpočtu. Také je uvedena kumulativní distribuční funkce a jádrový odhad hustoty. Další část práce se věnuje diskusi naměřených výsledků s ohledem na porovnání s výsledky geolokačních technik, jež naprogramovali kolegové autora diplomové práce. Výsledky jsou porovnány za pomoci průměrné chyby odhadu geografické polohy a jejího mediánu, času výpočtu, kumulativní distribuční funkce a jádrového odhadu hustoty.

KLÍČOVÁ SLOVA

Geografická poloha, geolokace, IP, latence, Constraint-based Geolocation, PlanetLab.

ABSTRACT

This diploma thesis is concerned with practical realization of CBG (Constraint-based Geolocation) algorithm, which is one of the IP (Internet Protocol) geolocation technique. IP geolocation determines the localization of a computer workstation location on the basis of on its IP address. The factors causing delays in data transfer are discussed first, followed by discussion of the issue of measuring these delays. The detailed explanation of IP geolocation follows where its contexts as well as the active geolocation techniques (techniques based on delay measurement mentioned above) are described.

After that a brief description of PlanetLab experimental network, which was used for geolocation techniques measuring, is presented followed by a section explaining the creation of reference points and targets, which are another necessary prerequisite for practical realization of the method. Then the practical realization is explained in the form of *CBGfinder* program and its verification on the basis of artificial input data along with an actual example of IP geolocation of a point in the Internet are provided.

Last but not least the measurement results of CBG algorithm are introduced, based on the analysis of *Bestline* parameters of one of the PlanetLab nodes measured in the period of one month, followed by a discussion of the inaccuracy of geological position and the computation speed. The cumulative distribution function as well as the kernel density estimation are also described. Final part of the thesis consists of discussion on measured results compared to results of other geological techniques results implemented by colleagues of the author of this diploma thesis. The results are compared on the basis of average inaccuracy of geological position estimations and its median, computation time, cumulative distribution function and kernel density estimation are also taken into regard.

KEYWORDS

Geographic Location, Geolocation, IP, Latency, Constraint-based Geolocation, PlanetLab.

HARTH, P. *Nalezení fyzické polohy stanic v síti Internet pomocí měření přenosového zpoždění*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2012. 80 s. Vedoucí diplomové práce doc. Ing. Dan Komosný, Ph.D..

Prohlášení

Prohlašuji, že svou diplomovou práci na téma „Nalezení fyzické polohy stanic v síti Internet pomocí měření přenosového zpoždění“ 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 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.

V Brně dne

.....

podpis autora

Poděkování

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

Dále děkuji všem členům z týmu za pomoc a spolupráci. Jmenovitě Ing. Lukáši Vernerovi za pomoc při nasazení programu *CBGfinder* na webovém serveru a Ing. Jiřímu Balejovi za instalaci potřebných programovacích jazyků na počítači, kde běží webový server.

Také velmi děkuji Bc. Martinu Maděrovi za pomoc při programování, Ing. Lukáši Maláčovi za opravu anglického abstraktu a Mgr. Lence Juřenové za korekturu češtiny.

V Brně dne

.....

podpis autora

OBSAH

ÚVOD.....	12
1. ZDROJE ZPOŽDĚNÍ PŘI PŘENOSU DAT V INTERNETU	14
1.1 Zpoždění v koncových zařízeních.....	15
1.2 Zpoždění na přenosových linkách.....	15
1.3 Zpoždění v mezilehlých zařízeních.....	16
2. MĚŘENÍ ZPOŽDĚNÍ V INTERNETU	17
2.1 Měření zpoždění a dostupnosti na jednotlivých vrstvách OSI modelu	17
2.2 Program <i>Ping</i>	18
2.2.1 Protokol <i>Internet Control Message Protocol</i> (ICMP).....	18
2.2.2 Postup ICMP dotazu z pohledu programu <i>Ping</i>	19
3. IP GEOLOKACE	20
3.1 Oblasti využití geolokace	20
3.2 Názvosloví používané v IP geolokaci	20
3.3 Základní rozdělení IP geolokačních metod.....	21
4. GEOLOKAČNÍ METODY VYUŽÍVAJÍCÍ MĚŘENÍ.....	23
4.1 Geolokační algoritmus GeoPing	23
4.2 Geolokační algoritmus Shortest Ping.....	25
4.3 Geolokační algoritmus SGP (<i>Simple GeoPing</i>)	25
4.4 Geolokační algoritmus SOI (<i>Speed of Internet</i>).....	25
4.5 Geolokační algoritmus GeoWeight.....	26
4.6 Geolokační algoritmus Octant.....	27
4.7 Geolokační algoritmus CBG (<i>Constraint-based Geolocation</i>).....	28
4.7.1 Multilaterace s hranicemi geografické vzdálenosti	29
4.7.2 Postup převodu zpoždění na hranici vzdálenosti	29
4.7.3 Použití distribuovaných hranic vzdáleností.....	31
4.7.4 Geolokace cílové stanice.....	31
4.7.5 Důvěryhodnost odhadnutého regionu	33
4.7.6 Potenciální výsledky při vytváření hranic vzdáleností.....	33

4.7.7	Přesnost metody CBG v porovnání s algoritmem GeoPing.....	34
5.	EXPERIMENTÁLNÍ SÍŤ PLANETLAB.....	35
5.1	Názvosloví sítě PlanetLab.....	36
5.2	Technické řešení sítě PlanetLab.....	37
5.3	Vybrané projekty běžící v síti PlanetLab	37
5.4	CentOS – distribuce operačního systému GNU/Linux	38
6.	VYTVOŘENÍ MNOŽINY REFERENČNÍCH BODŮ A CÍLŮ	39
6.1	Referenční body	39
6.2	Cíle pro geolokaci	40
6.2.1	DNS záznamy LOC.....	40
6.2.2	Postup vyhledání cílů	40
7.	REALIZACE ALGORITMU CONSTRAINT-BASED GEOLOCATION.....	43
7.1	Postup vyhledání odhadované polohy cíle	43
7.2	Serverová část	44
7.2.1	Vypočítání přímky <i>Bestline</i>	45
7.2.2	Převod hodnoty zpoždění na hranici vzdálenosti	46
7.3	Komunikace serveru s klientem	46
7.4	Klientská část	47
7.4.1	Paralelní zpracování komunikace s referenčními body ze sítě PlanetLab ..	48
7.4.2	Vstupní a výstupní formát dat	50
7.5	Ověření správné činnosti algoritmu na umělých datech	50
7.6	Příklad geolokace cíle v Internetu	52
8.	NASAZENÍ ALGORITMU V SÍTI PLANETLAB	54
8.1	Správa referenčních bodů v síti PlanetLab.....	54
8.2	Problémy při měření a jejich ošetření v programu	54
8.2.1	Instalace <i>Java Runtime Environment</i>	54
8.2.2	Chybějící oprávnění správce	54
8.2.3	Doba SSH připojení	54
8.2.4	Spuštění démona po restartu uzlů referenčních bodů.....	55

8.2.5	Zajištění automatické kalibrace na referenčních bodech	55
8.2.6	Řešení problému se špatně zadanými polohami referenčních bodů.....	55
9.	VÝSLEDKY MĚŘENÍ A POROVNÁNÍ METODY S OSTATNÍMI GEOLOKAČNÍMI ALGORITMY	56
9.1	Výsledky měření metody CBG	56
9.1.1	Vývoj přímky <i>Bestline</i> v čase.....	56
9.1.2	Chyba odhadu zeměpisné polohy a rychlost výpočtu	58
9.1.3	Kumulativní distribuční funkce.....	58
9.1.4	Jádrový odhad hustoty.....	59
9.2	Srovnání výsledků měření s ostatními metodami	60
9.2.1	Chyby odhadu zeměpisné polohy a rychlosti výpočtu	60
9.2.2	Kumulativní distribuční funkce.....	62
9.2.3	Jádrový odhad hustoty.....	63
10.	ZÁVĚR.....	64
	LITERATURA.....	65
	SEZNAM ZKRATEK	68
	SEZNAM PŘÍLOH.....	69
A	SKRIPTY PRO VYHLEDÁNÍ DOMÉN S DNS LOC ZÁZNAMEM.....	70
A.1	Skript LOClookup.....	70
A.2	Skript DNSrotateLOClookup.....	71
B	SEZNAM REFERENČNÍCH BODŮ A CÍLŮ	73
B.1	Seznam použitých referenčních bodů.....	73
B.2	Seznam použitých cílů	74
C	VÝSLEDKY MĚŘENÍ.....	76
C.1	Výsledky měření IP geolokační metody CBG	76
C.2	Výsledky všech metod	77
D	OBSAH PŘÍLOŽENÉHO CD	80

SEZNAM OBRÁZKŮ

Obr. 1.1: Zdroje zpoždění a místa jejich vzniku ([1], str. 42-1).	14
Obr. 1.2: Směrování mezi autonomními systémy AS1 a AS3 [3].	15
Obr. 4.1: Princip metody GeoPing ([10], str. 21).	24
Obr. 4.2: Princip určování polohy metodou GeoWeight (převzato z [29]).	26
Obr. 4.3: Princip určování polohy metodou Octant (převzato z [30]).	28
Obr. 4.4: Graf závislosti zpoždění RTT na vzdálenosti, vytvořený v uzlu sítě PlanetLab.	29
Obr. 4.5: Princip tvorby polygonu a jeho těžiště.	32
Obr. 4.6: Možné případy vytváření hranic vzdáleností ([10], str. 32).	34
Obr. 5.1: Rozmístění PlanetLab organizací ve světě (převzato z [31]).	36
Obr. 5.2: Zjednodušený vrstevný model uzlu sítě PlanetLab ([33], str. 32).	37
Obr. 5.3: Princip rozmístění uživatelů na uzlech v síti PlanetLab ([33], str. 32).	37
Obr. 6.1: Zeměpisné polohy referenčních bodů v Evropě.	39
Obr. 6.2: Zeměpisné polohy domén s DNS LOC záznamem v Evropě.	42
Obr. 7.1: Komunikace při vyhledávání odhadované polohy cíle.	44
Obr. 7.2: Vývojový diagram automatické kalibrace.	45
Obr. 7.3: Vývojový diagram převodu hodnoty zpoždění na hranici vzdálenosti.	46
Obr. 7.4: Vývojový diagram démona starajícího se o komunikaci serveru s klientem.	47
Obr. 7.5: Vývojový diagram klientské části.	48
Obr. 7.6: Diagram datových toků paralelní komunikace a třída agenta.	49
Obr. 7.7: Ověření správného výpočtu pozice cílové stanice.	51
Obr. 7.8: Graf závislosti vzdálenosti na zpoždění.	52
Obr. 7.9: Zaměření pozice cíle <i>switch.ch</i> .	53
Obr. 9.1: Změna hodnoty směrnice <i>k</i> v uzlu <i>planetlab1.cesnet.cz</i> .	57
Obr. 9.2: Změna hodnoty směrnice <i>k</i> v uzlu <i>planetlab1.cesnet.cz</i> v průběhu 24 hodin.	57
Obr. 9.3: Změna hodnoty posunutí <i>q</i> v uzlu <i>planetlab1.cesnet.cz</i> .	58
Obr. 9.4: Kumulativní distribuční funkce algoritmu CBG.	59
Obr. 9.5: Jádrový odhad hustoty metody CBG.	60
Obr. 9.6: Graf kumulativní distribuční funkce všech metod.	62
Obr. 9.7: Graf jádrového odhadu hustoty pro všechny metody.	63

ÚVOD

Tato diplomová práce se týká praktické realizace algoritmu CBG (*Constraint-based Geolocation*), což je jedna z metod řadících se mezi IP (*Internet Protocol*) geolokační techniky. IP geolokace se zabývá určením zeměpisné polohy počítačové stanice podle její IP adresy.

Práce je také součástí výzkumu, jenž se zabývá praktickými realizacemi IP geolokačních technik na bázi měření zpoždění a jejich následném porovnávání z pohledu přesnosti a dalších zvolených měřítek. Členové výzkumné skupiny jsou vedoucí skupiny Doc. Ing. Dan Komosný, Ph.D., Ing. Jiří Balej, Ing. Lukáš Verner, Bc. Petr Harth, Bc. Filip Dvořák, Bc. Peter Tropp, Jakub Chmelař a Tomáš Sousedík.

Ing. Jiří Balej vytváří a spravuje geolokační server. Ing. Lukáš Verner má na starosti webové geolokační stránky. Bc. Petr Harth má za úkol zrealizovat algoritmus CBG. Bc. Filip Dvořák vytváří metodu Octant. Bc. Peter Tropp pracuje na geolokační technice zvané GeoWeight, Jakub Chmelař programuje algoritmus SOI (*Speed Of Internet*) a Tomáš Sousedík implementuje odhad geografické polohy za pomoci pasivních geolokačních technik.

Pro následné porovnání výsledků metod je nutné vytvořit seznam referenčních bodů a cílů. Jakub Chmelař má za úkol zpracovat seznam nejméně 40 referenčních bodů z uzlů sítě PlanetLab nacházejících se v Evropě. Bc. Petr Harth, Bc. Filip Dvořák a Bc. Peter Tropp se musí postarat o vytvoření evropské množiny cílových stanic. Tyto stanice musí obsahovat DNS (*Domain Name System*) záznam s názvem LOC. Po úspěšném vytvoření množiny referenčních bodů a cílů je důležité výše uvedené metody implementovat na geolokační server s webovými stránkami ve spolupráci s Ing. Jiřím Balejem a Ing. Lukášem Vernerem.

V první kapitole se tato diplomová práce zabývá dílčími faktory způsobujícími zpoždění při přenosu dat mezi stanicemi v IP sítích. Druhá kapitola popisuje techniky měření tohoto zpoždění. Ve třetí části je vysvětlen pojem IP geolokace, její základní rozdělení a používané názvosloví. Ve čtvrté kapitole jsou popsány IP geolokační metody využívající měření zpoždění. Důraz je kladen na podrobné vysvětlení metody CBG, kterou implementuje autor diplomové práce.

Pátá kapitola obsahuje popis experimentální sítě PlanetLab, která je využita pro měření IP geolokačních metod. V šesté části je zmíněna tvorba seznamu referenčních bodů a cílů, což je další nutný předpoklad pro provedení měření metod a jejich porovnání.

Hlavním cílem práce je nastudovat a naprogramovat geolokační algoritmus CBG, dále za pomoci uměle vytvořených vstupních dat ověřit jeho správnou činnost a také provést odhad geografické polohy stanic na území Evropy. Toto je splněno v sedmé kapitole, kde je vysvětlen vytvořený program *CBGfinder* a je uvedeno ověření správné činnosti programu na umělých

vstupních datech. V neposlední řadě je zde zmíněn příklad výsledků geolokace cíle v Internetu, jež využívá síť PlanetLab.

Další kapitola obsahuje popis nasazení algoritmu v síti PlanetLab, což znamená, že je zde zmíněna správa referenčních bodů a vysvětlení řešení problémů, které se vyskytly při implementaci programu *CBGfinder* do prostředí sítě PlanetLab.

Poslední, devátá kapitola uvádí výsledky měření algoritmu CBG, jenž je implementován programem *CBGfinder*. Také je zde uveden rozbor naměřených výsledků s ohledem na porovnání s výsledky geolokačních technik, jež naprogramovali ostatní členové v týmu. Metody jsou porovnány pomocí průměrné chyby odhadu geografické polohy a jejího mediánu, času výpočtu, kumulativní distribuční funkce a jádrového odhadu hustoty.

1. ZDROJE ZPOŽDĚNÍ PŘI PŘENOSU DAT V INTERNETU

Pojem zpoždění vyjadřuje v sítích na bázi internetového protokolu dobu trvání cesty informace od zdroje k cíli. Tento čas je závislý na vlastnostech celé přenosové cesty. Značný vliv na zpoždění má vzdálenost komunikujících stran, typ přenosového média a doba strávená při směrování k cíli [1].

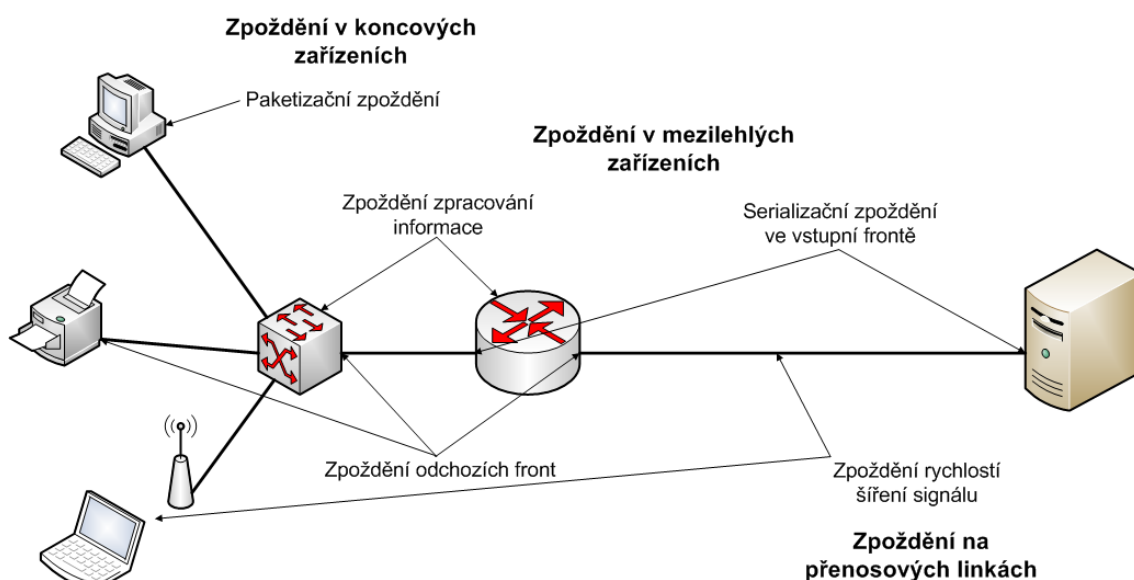
Synonymem ke zpoždění je latence, což je termín používaný v souvislosti s vlastností určitého prvku. Tato latence vzniká na různých částech komunikačního řetězce a má odlišný charakter, velikost i vliv na celkové zpoždění [1].

Při podrobnějším zkoumání latence bylo zjištěno, že zpoždění má podobnou velikost při opakovaných měřeních stejné přenosové trasy. Toto celkové zpoždění se dělí na **deterministickou** a **stochastickou** část [2].

Deterministická část je konstantní velikosti a lze ji vypočítat. Je to minimální čas potřebný pro přenos dané informace. Stochastická část zpoždění je náhodného charakteru a ovlivňuje ji aktuální stav sítě.

Faktorů zpoždění je celá řada, avšak dále se práce zabývá pouze zdroji zpoždění, které mají výrazný vliv na měření geolokace v IP sítích, neboť tato problematika je hlavní náplní této diplomové práce.

Zpoždění se z pohledu místa jeho vzniku dělí na **zpoždění v koncových zařízeních**, **zpoždění v mezilehlých zařízeních** a **zpoždění na přenosových linkách**. Na obr. 1.1 jsou vyobrazena místa vzniku výše uvedených zpoždění v komunikačním řetězci.



Obr. 1.1: Zdroje zpoždění a místa jejich vzniku ([1], str. 42-1).

1.1 Zpoždění v koncových zařízeních

Pojem koncové zařízení označuje zdroj a cíl přenášené informace, který přijímá a odesílá zprávu. Tato zpráva je na třetí vrstvě OSI (*Open System Interconnection*) modelu označena adresou zdroje a cíle a následně zabalena do paketu [1].

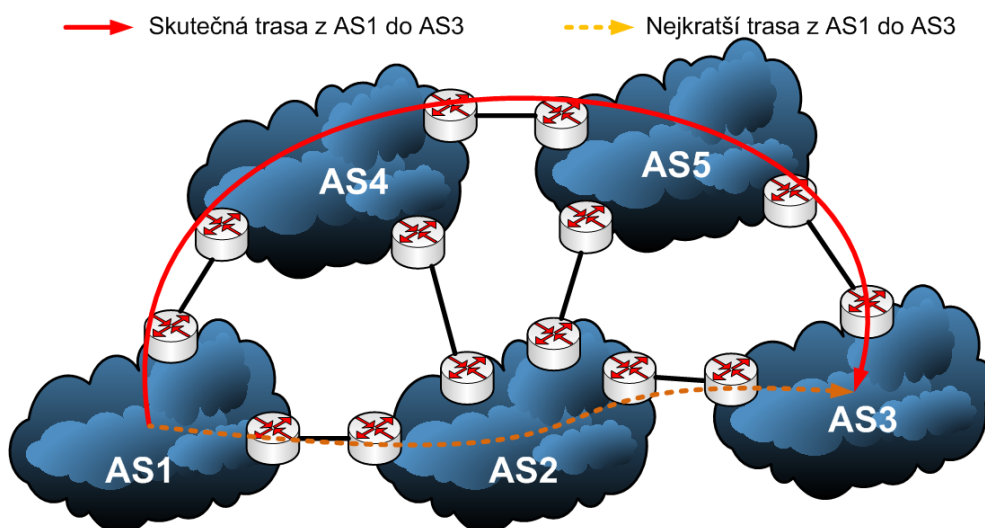
Na druhé vrstvě je paket zapouzdřen do rámce, či buňky a vyšle se bit po bitu přes přenosovou linku. Příjemací stanice provádí opačný postup pro vyčtení zprávy. Tato latence je považována za stochastickou část, neboť čas pro zpracování paketu nebývá stejný [2].

1.2 Zpoždění na přenosových linkách

Přenosové linky jsou součástí transportního systému, jenž je přímo závislý na fyzické poloze stanic. Skutečná délka trasy, kterou urazí zpráva, je značně delší než vzdálenost startu a cíle vzdušnou čarou. Kabely jsou totiž pokládány na vhodných místech podél silnic, železnic a dálkových vedení elektrické energie [1].

Další prodloužení cesty paketu způsobuje směrování. Směrovače totiž volí často jinou cestu k cíli než tu nejkratší, z důvodu administrativních rozhodnutí, cenové politiky či rychlosti linek. Tímto tedy vzrůstá i zpoždění způsobené rychlostí šíření signálu [2].

Internet používá směrování mezi autonomními systémy (AS), jež dělí IP síť na menší celky, mezi nimiž probíhá odlišné směrování od směrování uvnitř AS. Na obr. 1.2 je zachycena cesta zprávy mezi AS1 a AS3, která je ovlivněna směrovacími rozhodnutími, jež nekorespondují s požadavkem na nejkratší trasu paketu [3].



Obr. 1.2: Směrování mezi autonomními systémy AS1 a AS3 [3].

Výše uvedené zpoždění je tedy závislé nejen na typu použitého média, ale také na délce fyzické trasy. Pro transportní sítě se nejčastěji používá optické vlákno, jehož rychlost šíření signálu je 194 895 km/s, což je $0,65c$ (c je rychlost šíření světla ve vakuu¹). Pro metalické kabely platí hodnota $0,75c$. Při vzdálenostech nad 1000 km má tato latence velký podíl na celkovém zpoždění [1]. Velikost tohoto zpoždění je nejnižší hranice celkového zpoždění a patří do deterministické složky [2].

1.3 Zpoždění v mezilehlých zařízeních

Za mezilehlá zařízení se označují jakékoliv aktivní prvky na cestě mezi koncovými zařízeními. Tyto prvky plní mnoho úkolů zahrnujících různé vrstvy OSI modelu. Mezi nejdůležitější činnosti patří směrování a přepínání datových jednotek od zdroje k cíli.

Latence mezilehlých zařízení je složena ze zpoždění ve vstupní frontě, doby přesunutí dat ze vstupu na výstup a doby čekání ve výstupní frontě. Zpoždění ve vstupní frontě je ovlivněno příchozí rychlostí linky a velikostí paketu. Přesněji vzato je to čas mezi přijetím prvního a posledního bitu rámce [1].

Doba přesunutí datové jednotky ze vstupu na výstup určitého síťového prvku je závislá na rychlosti a funkcích prvku. Směrovače disponují latencí mezi 10 až 100 μ s, přepínače 1 až 10 μ s a zpoždění na opakovačích či aktivních rozbočovačích je zanedbatelné. Při velkém zatížení prvku se celkové zpoždění může zvýšit až o několikanásobek latence nezatíženého segmentu [1].

Čas čekání rámce v odchozí frontě je přímo závislý na aktuálním počtu datových jednotek směřujících na stejné rozhraní, jenž dorazil do fronty dříve. Z výše uvedeného vyplývá, že tato latence se řadí do stochastické složky celkového zpoždění.

Přesné rozmezí hodnot zpoždění pro určité prvky udává výrobce a různé testy těchto zařízení, které lze nalézt v publikacích [4], [5] a [6].

¹ Rychlost šíření světla $c = 299\,792\,458$ m/s.

2. MĚŘENÍ ZPOŽDĚNÍ V INTERNETU

Z důvodu měření zpoždění bylo vyvinuto mnoho různých aplikací a hardwarových zařízení. Názvosloví i metodiku měření zpoždění v IP sítích definuje RFC 1242 (*Request For Comments*) [8]. V této kapitole jsou popsány základní nástroje pro měření zpoždění a je vysvětlen jejich princip. Výpis metod není kompletní, protože podobných programů existuje celá řada.

Nejprve je nutno si uvědomit způsob měření latence a jaké faktory ovlivňují výsledek. Také je důležité pro dosažení přesných výsledků provést nejméně 10 až 15 měření, přičemž nejvíce vypovídajícím výsledkem měření je nejmenší hodnota [13].

Důležité je také stanovit cíl měření a následně vybrat vhodný nástroj. Podle typu služby, jež se měří, se musí zvolit vrstva OSI modelu. Též záleží na komunikujících stanicích. Tyto stanice musí daný nástroj podporovat a musí být schopny komunikovat na zvolené OSI vrstvě. Měření je ovlivněno i mezilehlými zařízeními, neboť tato mohou například zprávy typu daného měření zahazovat. Problémem může být i interpretace dosažených výsledků, protože bývá nejednoznačná. Například existuje velký rozdíl mezi jednocestným a obousměrným zpožděním [7].

Jednocestné a obousměrné zpoždění

Jednocestné zpoždění (*one-way delay*) měří zdrojová i cílová stanice a výsledkem je hodnota zpoždění pro daný směr. Jeho nevýhodou je nutnost synchronizace stanic, které jsou od sebe zpravidla vzdáleny. Tato synchronizace musí být přesná v rámci desítek milisekund [2].

Tuto nevýhodu řeší obousměrné zpoždění RTT (*Round Time Trip*), což je velikost zpoždění změřená prostřednictvím jediné stanice. Zde zpoždění zahrnuje cestu paketu k cíli, dobu jeho zpracování ve stanici adresáta a cestu nazpět. Daná hodnota je tedy přibližně dvojnásobek jednosměrného zpoždění. Jelikož může být směrování asymetrické [9], což znamená, že se cesta paketu k cíli liší od cesty odpovědi na tento paket, takže i jejich hodnoty zpoždění jsou rozdílné, zcela přesnou latenci pro jeden směr nelze tímto způsobem získat [10].

Dle publikace [13] bylo empiricky zjištěno, že pokud má RTT zpoždění hodnotu do 5 ms, tak se stanice ve více než 90 % nachází ve vzdálenosti do 50 km. Při RTT menším než 10 ms jsou komunikující strany vzdáleny od sebe do 300 km.

2.1 Měření zpoždění a dostupnosti na jednotlivých vrstvách OSI modelu

Nejběžnější měření je prováděno na třetí vrstvě OSI modelu za pomoci programu *Ping*, jenž je podrobně rozebrán v kapitole 2.2. Nastávají ovšem různé případy, kdy je nezbytné použít jiné nástroje, neboť je nutno změřit zpoždění či dostupnost na ostatních vrstvách OSI

modelu [10]. Dále následuje výčet vrstev, na nichž lze měřit příslušnými aplikacemi s odkazy na podrobnější popisy uvedených technik:

- 7. vrstva: *Telnet* [14] – slouží k ověření, zda strany naslouchají na portu 23 a jsou tedy schopny navázat spojení na aplikační vrstvě,
- 4. vrstva: *TCPtracert* [15] – dotazuje se na určité číslo portu a příslušný protokol,
- 3. vrstva: *Ping* [16] – využívá ICMP (*Internet Control Message Protocol*) protokol pro ECHO dotaz na vzdálenou stranu,
- 2. vrstva: *Layer 2 traceroute utility* [17] – proprietární řešení firmy Cisco na bázi protokolu CDP (*Cisco Discovery Protocol*) pro zjištění přepínačů.

2.2 Program *Ping*

Jelikož se tato diplomová práce zabývá IP geolokační metodou, která používá právě *Ping*, tak je tento program popsán důkladněji. Ostatní nástroje měření zpoždění jsou zmíněny výše pouze obecně s příslušnými zdroji pro vyhledání vyčerpávajícího popisu.

Ping (*Packet InterNet Groper*) je základní nástroj k měření RTT zpoždění. Spustí se příkazem `ping` s IP či URL (*Uniform Resource Locator*) adresou stanice, ke které se měří zpoždění. Chování programu lze změnit určitými přepínači. V operačním systému (OS) Microsoft Windows se výpis statistik zobrazuje pouze v rámci milisekund, kdežto v OS GNU/Linux v řádu mikrosekund [10]. V OS Windows lze však použít přesnější program *True Ping* [18].

Program *Ping* využívá protokol ICMP, díky němuž dotazuje vzdálenou stanici. Z tohoto důvodu je tento protokol rozebrán dále.

2.2.1 Protokol *Internet Control Message Protocol* (ICMP)

Internet Control Message Protocol je protokol řídicích zpráv, jenž slouží pro informování o chybách při přenosu paketů. ICMP definuje několik typů odpovědí, jež konkretizují, proč nebylo možné doručit datagram cíli. ICMP zprávy jsou přenášeny v datové části IP datagramu, ale stává se, že jsou tyto zprávy záměrně zahazovány. Toto se děje z bezpečnostních důvodů. Tento stav tedy vede k zásadním problémům při měření zpoždění [10].

ICMP protokol v rámci programu *Ping* je také používán správci sítí jako diagnostický nástroj. Podrobněji o ICMP hovoří RFC 792 [11] a 1256 [12].

2.2.2 Postup ICMP dotazu z pohledu programu *Ping*

Proces ICMP dotazu, jenž využívá program *Ping*, proběhne v následujících v krocích [7].

1. Zdroj vytvoří ICMP zprávu typu 8 (*ECHO request*).
2. Zpráva je zapouzdřena do rámce a odeslána cíli.
3. Zdroj si uloží čas odeslání.
4. V síti každý uzel kontroluje, zda cílová adresa koresponduje s jeho adresou. Pokud ne, tak je zpráva odeslána dále dle jeho směrovací tabulky.
5. Cíl přijme paket, rozbalí ICMP zprávu a uloží si informace.
6. Cíl odesílá zprávu s odpovědí typu 0 (*ECHO reply*) na zdrojovou adresu přijatého ICMP dotazu.
7. Zdroj tento datagram přijme a odečte čas strávený cestou, čímž obdrží RTT zpoždění a vypíše statistiku.

3. IP GEOLOKACE

Pojem geolokace obecně označuje hledání geografické polohy objektu na základě znalosti jeho určité vlastnosti nebo známých údajů. V případě IP geolokace se jedná o hledání fyzické pozice počítače, či jiného síťového prvku, se známou IP adresou.

Avšak lokalizovat prvek jen podle IP adresy není snadný úkol, neboť tato informace v sobě nenese žádnou spojitost s geografickou polohou. Mezi zeměpisným umístěním stanice a IP adresou tedy neexistuje žádná přímá souvislost [19].

Pokud se dále v textu bude vyskytovat pojem geolokace, bude jím myšlena IP geolokace.

3.1 Oblasti využití geolokace

S rozrůstající se sítí Internet, kdy roste počet jeho uživatelů a množství služeb, které využívají této sítě, vzniká stále větší potřeba určování zeměpisné polohy komunikujících stran. Díky tomuto stavu lze najít mnoho odvětví a činností, kde se dají geolokační aplikace využít. Následují příklady použití geolokace [10]:

- **Reklama na Internetu:** Pokud se nabídne zákazníkovi produkt z jeho blízkého okolí, tak je větší pravděpodobnost, že jej tato nabídka osloví, než například reklama nabízející produkt z druhé strany zeměkoule.
- **Aktuální informace z blízkého okolí:** Zprávy, počasí či kulturní akce z místa pobytu uživatele Internetu jsou relevantnější a žádanější, než informace vzdálené stovky kilometrů.
- **Telefonie VoIP (*Voice over IP*):** V tomto oboru vzniká potřeba informace o poloze volajících stran. Například v případě tísňového volání, kdy volající je ohrožen na životě a neví, kde se nachází, mu může taková informace zachránit život.
- **Odhalování zneužití kreditních karet:** Pokud si zákazník zvolí obvyklou zeměpisnou oblast používání karty, může se lehce přijít na zneužití karty pocházející z místa mimo obvyklou oblast. Karta se může například zablokovat ještě dříve, než nepatřičná transakce proběhne.

3.2 Názvosloví používané v IP geolokaci

V každé práci zabývající se IP geolokací se lze setkat s následujícími pojmy, které jsou použity i v tomto pojednání. Tyto výrazy jsou dále uvedeny spolu s jejich významy [10].

- **Cíl** (T, anglicky *Target*): Cílem se rozumí stanice s neznámou geografickou pozicí. Tato poloha se zjišťuje pomocí geolokačních metod.

- **Referenční bod** (RB, anglicky *Landmark*): RB označuje stanici, jejíž zeměpisná poloha je známa. Některé metody vyžadují, aby RB byly schopny měřit RTT zpoždění. Tyto RB jsou označeny jako aktivní. Body, jež měřit zpoždění neumožňují, jsou nazvány pasivními referenčními body.
- **Sonda** (S, anglicky *Probe*): Je to stanice se známou polohou, jejímž úkolem je sondovat, tedy měřením zjišťovat zpoždění k ostatním RB či cílům. Rozdílem mezi aktivním RB a sondou je fakt, že u sondy v určitých případech není nutno znát její geografickou polohu.
- **Dataset**: Pod tímto termínem se skrývá soubor všech stanic, které jsou použity pro nalezení cíle.
- **Hranice vzdálenosti** (anglicky *Constraint Distance*): Tento termín zastupuje kruh se středem v podobě zeměpisné polohy referenčního bodu a poloměrem, jenž odpovídá hodnotě transformované z naměřeného zpoždění k cíli či dalšímu RB [23].
- **Bestline**: Přímka, jejíž směrnice a posunutí definuje v metodě CBG poměr naměřeného zpoždění RTT k příslušné geografické vzdálenosti. Tento poměr je pak využit pro vypočítání hodnoty hranice vzdálenosti z naměřeného zpoždění RTT [23].
- **Baseline**: Přímka, jejíž směrnice se rovná poměru 1 ms zpoždění RTT ku 100 km trasy. V metodě je CBG využívána pro určení maximální možné hranice vzdálenosti [23].
- **Automatická kalibrace**: Proces zavedený v metodě CBG, kdy se jednotlivé RB ptají na zpoždění k ostatním RB a podle těchto informací si upravují vlastní přímky *Bestline* [23].

3.3 Základní rozdělení IP geolokačních metod

Základní rozdělení geolokačních metod je provedeno podle toho, zda metoda využívá ke své činnosti měření zpoždění (**aktivní metody**) nebo jestli se informace pro nalezení zeměpisné polohy stanice vyhledávají pouze v určitých databázích či seznamech (**pasivní metody**).

Pasivní metody

Mezi pasivní metody patří všechny procesy využívající ke geolokaci informace o IP adrese. Tyto informace jsou uloženy v databázích, kde se nalézají IP adresy s příslušnými pozicemi. Seznamy mohou být soukromé (například [20]), či veřejné (například WhoIs databáze [21]). Dále se mezi pasivní metody řadí aplikace Geotrack [13], odhadující pozici na základě informace z DNS (*Domain Name System*) serveru cíle.

Další pasivní metoda je GeoIP [22], což je produkt firmy MaxMind. Pomocí její databáze lze podle IP adresy zjistit i zeměpisnou délku a šířku, rychlost připojení k Internetu či poskytovatele internetového připojení. V databázi se používají informace z webových stránek, které žádají jejich uživatele, aby uvedli svou polohu. Před použitím metody GeoIP je nutno si stáhnout její databázi, čímž se ale značně zrychlí odezva dotazu na polohu IP adresy, neboť dotaz probíhá lokálně v rámci počítače.

Tyto pasivní principy geolokace jsou dnes stále velmi rozšířené a oblíbené i přes nutnost manuální údržby databází, ze kterých čerpají informace [10].

Aktivní metody

Aktivní metody využívají měření RTT zpoždění v IP sítích. Pro jejich pochopení je nutno orientovat se v problematice zpoždění a jeho měření (viz výše kapitoly 1 a 2) a v oboru znalosti topologie sítě. Z tohoto pohledu se mohou aktivní metody dále dělit na procesy založené na měření zpoždění, či znalosti topologie sítě. Jako metody s nejlepšími výsledky se mohou označit kombinace obou metod, kdy je využito jak znalosti topologie sítě, tak i hodnot zpoždění [10].

Jelikož se tato práce primárně zabývá metodou využívající měření s názvem *Constraint-based Geolocation* (CBG) a tento algoritmus je následně srovnáván s ostatními podobnými metodami, tak je další kapitola věnována stručnému popisu principů konkrétních metod na bázi měření zpoždění. Pro jejich srozumitelné porovnání je nutno těmto metodám rozumět.

4. GEOLOKAČNÍ METODY VYUŽÍVAJÍCÍ MĚŘENÍ

V této kapitole je obecně popsána většina aktivních geolokačních metod využívajících měření zpoždění. Geolokační algoritmus CBG (*Constraint-based Geolocation*) je probrán podrobněji, neboť se následující kapitola týká jeho praktické realizace.

Geolokační techniky na bázi měření zpoždění využívají soustavu referenčních bodů (RB), jež musí být vybrány systematicky s ohledem na jejich zeměpisnou polohu, aby se dosáhlo nejlepších výsledků [23], [24]. Hodnoty zpoždění jsou metodami různě využity, také přesnost technik a náročnost uvedení do praxe se liší.

Velkou výhodou těchto algoritmů je, že jsou plně automatizované, čímž se myslí, že jediná manuálně zadávaná vstupní data jsou geografické souřadnice a IP adresy RB či sond. Funkčnost se stále stejnou přesností není omezena obsahem databáze a jediné omezení skýtá kolísající kvalita stavu sítě [10].

Princip měření zpoždění namísto hledání informací v ručně udržovaných databázích skýtá značnou výhodu, která se zřejmě projeví až s plným nástupem IPv6 (*Internet Protocol version 6*). Při použití tohoto protokolu enormně vzrůstá počet IP adres na 2^{128} . Díky tomuto faktu mohou poskytovatelé Internetu neustále měnit rozsahy přidělených adres podle potřeby a tímto se musí taktéž měnit záznamy v databázích, aby se zajistila jejich relevance. Tento stav je však pro metody na bázi zpoždění nepodstatný, neboť svou činnost mohou vykonávat bez ohledu na měnící se adresní prostory [25].

V současné době se pro připojení uživatelů do sítě často používá proxy serverů či techniky NAT (*Network Address Translation*) patřících poskytovatelé Internetu. Proto se při pohledu na IP adresu uživatele jeví jeho adresa shodná s adresou takového síťového prvku. Tento stav je podle [13] a [26] velká překážka všech geolokačních technik na bázi znalosti IP adresy cíle.

Pokud se cíl nachází blízko své brány, tak je toto zkreslení nepatrné. Ale jestliže je cíl vzdálen až stovky kilometrů, tak vzniká poměrně velká chyba. Takovýto nepřesný výsledek je pak mnohem horší, než kdyby pozice cíle zůstala neurčena. Tomuto jevu se určité metody snaží předcházet tím, že pokud zjistí pozici brány namísto cíle, tak odhad cíle raději neuvedou. Jako zástupce těchto metod se nabízí algoritmus TBG (*Topology-based Geolocation*), jenž je podrobně popsán v publikaci [19].

Následuje popis vybraných geolokačních algoritmů na bázi měření zpoždění v Internetu.

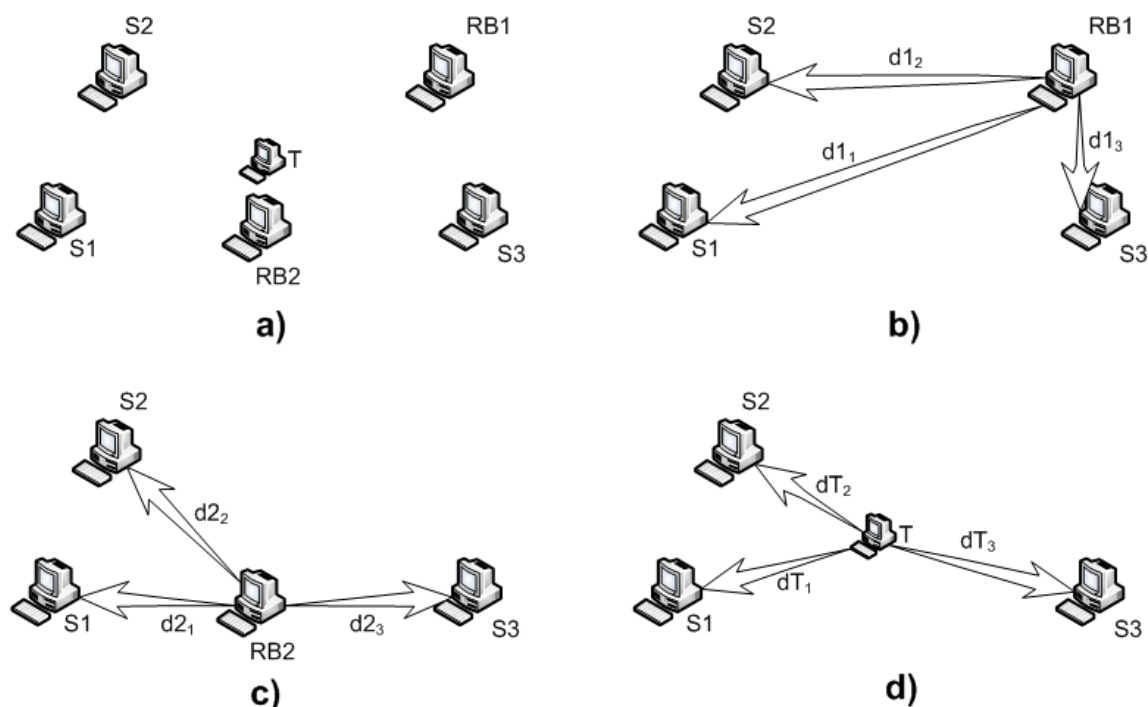
4.1 Geolokační algoritmus GeoPing

Geolokační metoda GeoPing [13] pracuje na principu znalosti zpoždění mezi sondami, referenčními body (RB) a cíli. Také zná zeměpisné polohy RB. Jako jediné možné odhadované polohy cílů se označují geografické pozice RB. Jinými slovy řečeno, výsledná pozice cíle může

korespondovat jen s pozicí jednoho RB. Tento fakt vede ke konečnému počtu odpovědí, jenž se logicky rovná počtu RB [10].

Tímto tedy také vzniká hlavní nepřesnost metody, neboť i ten nejvhodnější RB, jehož pozice byla vybrána jako nejvíce se shodující s cílem, může být stále velmi vzdálen od opravdové pozice cíle. Výše zmíněná chyba se daří zmenšovat zvýšením počtu RB a vhodným rozmístěním sond [10].

Na Obr. 4.1 je zachycen princip metody GeoPing. V sekci a) je zobrazeno rozmístění sond S1 až S3, referenčních bodů RB1 a RB2, a cíle T. V části b) jsou vyneseny jednotlivé složky vektoru zpoždění $\mathbf{d1}$, jenž produkuje RB1. V oddíle c) se nachází vektor $\mathbf{d2}$ referenčního bodu RB2. Vektor \mathbf{dT} produkující cíl T je v části d). Vektory $\mathbf{d1}$ a $\mathbf{d2}$ se porovnají s vektorem \mathbf{dT} a následně se pozice RB, jehož vektor je velikostně nejshodnější s \mathbf{dT} , prohlásí za polohu cíle T.



Obr. 4.1: Princip metody GeoPing ([10], str. 21).

Při měření přesnosti metody se autorům metody GeoPing [13] podařilo lokalizovat cíle s následující přesností. 90 % provedených pokusů odhadu obsahovalo chybu menší než zhruba 1350 km, 75 % pokusů chybu menší než přibližně 750 km, 50 % pokusů bylo zatíženo chybou odhadu pod 400 km a konečně 25 % pokusů bylo úspěšných natolik, že se chyba pohybovala pod hodnotou 150 km.

4.2 Geolokační algoritmus Shortest Ping

Metoda Shortest Ping [19] je jedna z nejjednodušších technik založených na měření RTT zpoždění. Způsob hledání cíle vychází z principu použití nejmenší hodnoty zpoždění. Po vytvoření datasetu RB, jejichž pozice jsou známy, se jako pozice cíle prohlásí poloha RB, ze kterého bylo k cíli naměřeno nejmenší RTT zpoždění [10].

Dle experimentálních výsledků je tato metoda úspěšnější než GeoPing, i když je principiálně jednodušší a dokonce se blíží přesnosti metod CBG a SOI. Nesmí ale nastat případ, kdy jsou všechny RB vzdáleny od cíle podobně daleko, nebo jsou velmi vzdáleny. V obou případech vzniká nepřesné měření, kdy se v první situaci algoritmus nemůže rozhodnout, jaký RB zvolit za cílovou polohu a ve druhém případě je odhadnutá pozice cíle velmi vzdálena reálnému zeměpisnému umístění [10].

4.3 Geolokační algoritmus SGP (*Simple GeoPing*)

Princip metody SGP představuje zjednodušení techniky GeoPing [26]. Rozdíl je v použití pouze jediné sondy, ze které měříme zpoždění ke všem referenčním bodům a cíli. Hodnoty zpoždění k RB se porovnají s velikostí zpoždění k cíli a pozice RB, k němuž je zpoždění od sondy nejshodnější se zpožděním k cíli, se označí jako odhadovaná poloha cíle.

Pro zvýšení přesnosti této techniky se měření provádí vícekrát a výsledná oblast cíle je určena průnikem jednotlivých výsledků. Pokud se výsledky neshodují, tak se odhad pozice považuje za neúspěšný [10]. Dle [26] je přesnost této metody velmi malá, neboť pouze 26,8 % cílů bylo odhadnuto správně.

4.4 Geolokační algoritmus SOI (*Speed of Internet*)

Metoda SOI vychází ze stejného principu jako technika CBG (viz kapitolu 4.7), tedy určování polohy cíle pomocí multilaterace využívající RB a jejich hranic vzdáleností. Oproti CBG ale SOI využívá faktu, že vzdálenost mezi stanicemi je z většiny případů kratší, než by se dalo odvodit ze zpoždění nebo z rychlosti šíření světla ve vakuu [19]. Z měření provedených v práci [19] vyplývá, že zpoždění mezi stanicemi je v takovém vztahu se vzdáleností, jakoby data cestovala rychlostí o velikosti $\frac{4}{9}c$ (c značí rychlost světla). Tento jev se potvrdil v 80 % měření, naopak ve 20 % měření vyšel faktor $\frac{2}{9}c$.

Výše uvedený poměr je tedy použit k transformaci zpoždění na vzdálenost při vytváření hranic vzdáleností jednotlivých ref. bodů. Tento způsob je mnohem jednodušší, než automatická kalibrace použitá u CBG. Také se nemusí měřit zpoždění mezi jednotlivými RB [10].

Naměřené výsledky v [19] potvrzují, že je koeficient pro převod zpoždění na vzdálenost přesný. I bez zatížení, kterými disponuje CBG v podobě složitého vytváření přímky *Bestline* a automatické kalibrace, dosahuje v podstatě stejných výsledků jako CBG.

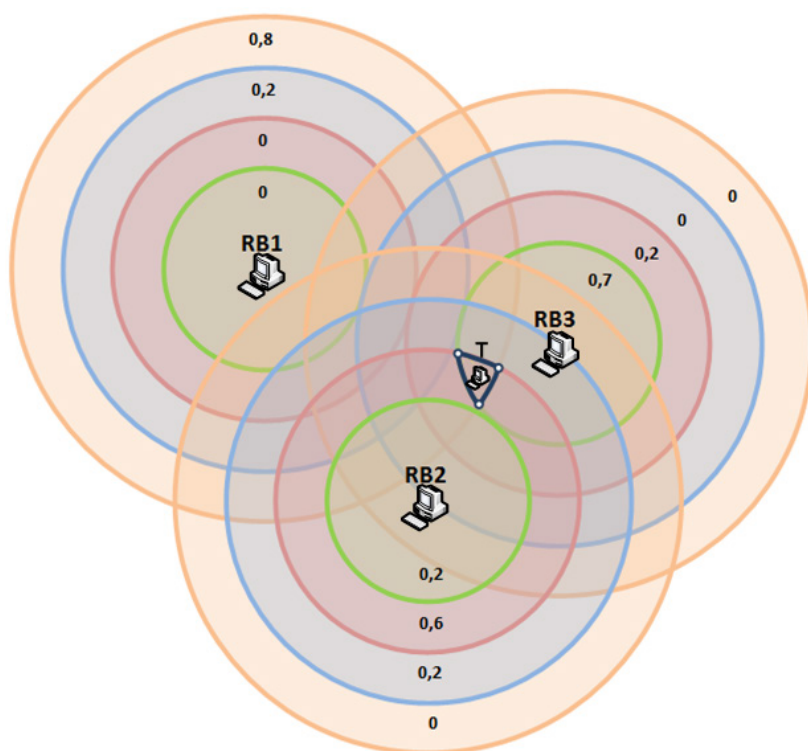
4.5 Geolokační algoritmus GeoWeight

GeoWeight je metoda založená na měření zpoždění RTT, jež také vychází při určování polohy cíle ze základních principů metody CBG, což je proces multilaterace hranic [29].

Případné rozdíly minimálních a maximálních hodnot zpoždění, tedy možných hranic vzdáleností, jsou řešeny tím, že se dané zpoždění rozdělí na stejné úseky a přiřadí se váhy těmto jednotlivým podoblastem. Jako mezní hodnoty pro výpočet vah se bere nejmenší a největší zpoždění. Tyto váhy pak vyznačují pravděpodobnost, kde by se cíl mohl nacházet. Větší váha znamená vyšší pravděpodobnost výskytu cílové stanice [29].

Při měření zpoždění z více referenčních bodů dochází k protínání se těchto oblastí pravděpodobností, což metoda GeoWeight počítá jako součet vah překrývajících se částí oblastí. Pozice cíle se určí jako těžiště protínajících se oblastí s nejvyšší vahou. Díky přiřazení vah jednotlivým podoblastem umožňuje metoda GeoWeight vymezit menší oblast pro hledání pozice cíle, a tím lze dosáhnout lepší přesnosti při odhadu geografické polohy. Pro podrobnější popis metody GeoWeight viz publikaci [29].

Na obr. 4.2 je zachycen princip určování polohy metodou GeoWeight, kdy je pozice cíle T odhadnuta pomocí měření zpoždění ze tří referenčních bodů (RB 1 až 3).



Obr. 4.2: Princip určování polohy metodou GeoWeight (převzato z [29]).

4.6 Geolokační algoritmus Octant

Metoda Octant [30] taktéž vychází z principů algoritmu CBG. Kromě toho využívá pozitivních a negativních informací. Pozitivní informace určují oblasti, kde se cílový bod může nacházet a negativní informace značí oblasti, kde cíl v žádném případě ležet nemůže.

V praxi tyto informace určují rozměry mezikruží se středem v RB. Vnější poloměr značí horní hranici, jež je odvozena z $\frac{2}{3}$ rychlosti světla, neboť za touto hranicí už cíl nemůže ležet. Spodní hranice určuje poloměr vnitřní kružnice mezikruží, v níž se také nemůže vyskytovat poloha cíle. Spodní hranice je vypočítána z negativních informací a oblast mezikruží zase z pozitivních informací.

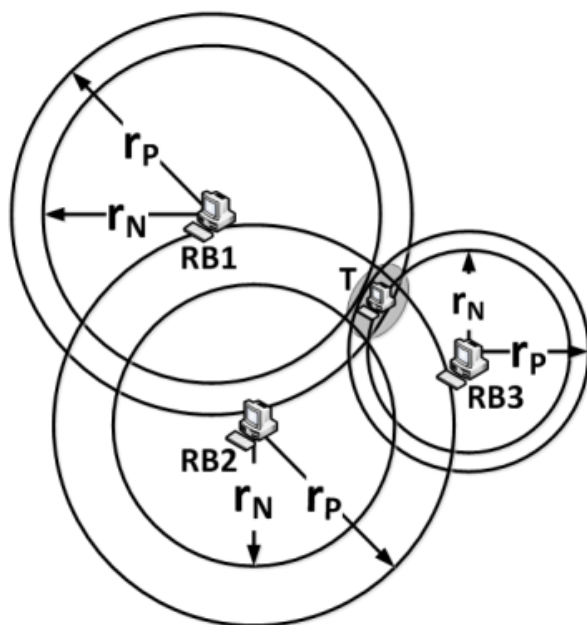
Metoda také využívá primární a sekundární RB. Primárními RB jsou myšleny RB se známou polohou. Geografická poloha sekundárních RB je neznámá a musí se vypočítat. Použitím sekundárních RB lze zmenšit chybu měření zpoždění vzniklou nepřímým směřováním (viz kapitulu 1), kdy jsou směrovače na cestě k cíli označeny jako sekundární RB a následně jsou lokalizovány pomocí metody Octant. Tímto se tedy cesta k cíli rozdělí na úseky ohraničené směrovači.

Pro další zlepšení přesnosti určení polohy sekundárních RB je využito informací z DNS jmen směrovačů, kdy tato doménová jména mohou obsahovat názvy měst či ulic, kde jsou směrovače instalovány.

Přepočet zpoždění na vzdálenost je podobný jako u CBG. Namísto *Bestline* je ale využita komplexní obálka (podrobněji viz [30]). Octant se také zabývá zpožděním na poslední míli. Toto zpoždění je totiž obvykle vyšší, než zpoždění ve zbývajících částech cesty a vyskytuje se na trase mezi poskytovatelem připojení k Internetu a cílovou stanicí.

Z odhadované oblasti, kde cílová stanice může ležet, je pak pomocí metody Monte Carlo (viz [42]) vypočítána zeměpisná poloha odhadovaného cíle. Pro podrobnější popis metody Octant viz publikaci [30].

Na obr. 4.3 je pro názornost zakreslen princip metody Octant. Mezikruží mají středy v referenčních bodech (RB1 až 3) s negativními (r_N) a pozitivními poloměry (r_P). Tam, kde mezikruží tvoří průnik, leží šedá odhadovaná oblast polohy cíle T.



Obr. 4.3: Princip určování polohy metodou Octant (převzato z [30]).

4.7 Geolokační algoritmus CBG (*Constraint-based Geolocation*)

Protože se tato práce zabývá praktickou realizací této metody, je zde tato technika popsána podrobně.

Základní myšlenkou metody CBG je, že odhad zeměpisné vzdálenosti cíle je možno stanovit pomocí multilaterace, stejně jako je to provedeno v GPS (*Global Positioning System*) [27]. Dalším klíčovým prvkem CBG je schopnost přesně transformovat změřené zpoždění na hranici vzdálenosti pomocí automatické kalibrace.

Výchozím bodem pro tento převod je fakt, že digitální informace cestuje optickými kabely téměř přesně $\frac{2}{3}$ rychlostí světla ve vakuu. To znamená, že tato míra zpoždění okamžitě udává horní hranici vzdušné vzdálenosti mezi koncovými body. Horní hranice je tedy míra zpoždění dělená rychlostí světla v optickém vlákne.

Při pohledu na konkrétní pár koncových bodů lze říci, že existuje také ideální teoretické minimální zpoždění při přenosu paketu, které je dáno vzdáleností mezi nimi. Proto skutečná míra zpoždění mezi body spočívá v doplňkovém zkreslení připočteném k ideálnímu zpoždění.

Nicméně pokud by CBG použilo jednoduché měření zpoždění k přímému odvození hranice vzdálenosti, tak by to nebylo dostatečně přesné. Pro přesné výsledky je důležité v co největší míře odhadnout a odstranit doplňkové zkreslení. CBG toto provádí pomocí automatické kalibrace, která měří zpoždění od každého RB k ostatním. To se provádí distribuovaným způsobem, jak je vysvětleno v kapitole 4.7.3. Díky automatické kalibraci může CBG přesněji transformovat sadu naměřených zpoždění k cíli na hranice vzdáleností.

Poté CBG používá multilateraci s těmito hranicemi vzdáleností k vypočítání odhadu geografické oblasti, která obsahuje cílového hostitele. Bodový odhad cíle je pak v těžišti takového regionu.

CBG také přiřazuje důvěryhodnost regionu ke každému odhadu cíle, čímž umožňuje aplikacím, aby dokázaly posoudit, zda je odhad umístění cíle dostatečně přesný s ohledem na jejich potřeby [23].

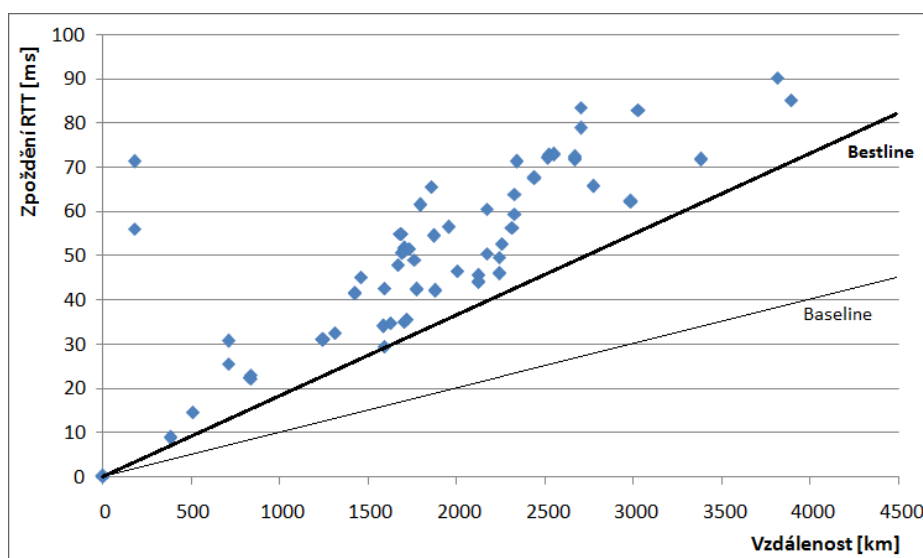
4.7.1 Multilaterace s hranicemi geografické vzdálenosti

Fyzická poloha určitého bodu může být odhadnuta pomocí dostatečného počtu vzdáleností nebo úhlů k jiným pevným bodům, jejichž pozice jsou známy. Pokud se hovoří o vzdálenosti, pak se tento proces nazývá multilaterace.

Hlavní problém, který vyplývá z použití multilaterace, je přesné měření vzdáleností mezi cílem a referenčními body. Například GPS používá multilateraci tří satelitů pro odhad polohy daného přijímače GPS. V případě GPS je vzdálenost mezi GPS přijímačem a satelitem měřena pomocí načasování, jak dlouho zabere signálu vyslanému z družice příchod do přijímače GPS. Přesné měření času a časový interval je základem přesnosti GPS. Na rozdíl od GPS je složité přesně transformovat změřené zpoždění v Internetu na zeměpisnou vzdálenost. Toto je pravděpodobně důvodem, proč přímá multilaterace zůstala dosud nevyužitá pro účely geolokace internetových hostitelů [23].

4.7.2 Postup převodu zpoždění na hranici vzdálenosti

Nejprve je nutno si povšimnout vzorkového rozptylového grafu závislosti zeměpisné vzdálenosti na zpoždění v síti na obr. 4.4, jenž byl vytvořen v referenčním bodu sítě PlanetLab s DNS jménem *aladdin.planetlab.extranet.uni-passau.de* a IP adresou 195.37.16.121.



Obr. 4.4: Graf závislosti zpoždění RTT na vzdálenosti, vytvořený v uzlu sítě PlanetLab.

Osa x popisuje geografickou vzdálenost a osa y je zpoždění v síti mezi určitým referenčním bodem RB_i a zbývajícími RB. Významy *Baseline* a *Bestline* jsou vysvětleny dále v této sekci.

Dle [23] přímka *Baseline* zachycuje ideální vztah mezi zpožděním v optickém vlákne a vzdáleností. Její rovnici lze napsat ve tvaru

$$y = mx + b, \quad (4.1)$$

kde $b = 0$, protože nejsou lokalizována žádná zkreslující zpoždění a m se vztahuje pouze na rychlost cestování bitů v komunikačním médiu. Jak již bylo uvedeno, digitální informace cestuje skrz optické kabely téměř přesnou $\frac{2}{3}$ rychlostí světla ve vakuu [28]. To poskytuje velmi vhodné pravidlo, že na 1 ms RTT zpoždění připadá 100 km délky kabelu. Takový vztah může být použit k získání absolutní fyzické dolní hranice RTT zpoždění mezi stanicemi, jejichž geografické umístění je dobře známo. Tato dolní mez je zobrazena jako *Baseline* na obr. 4.4.

Nicméně v praxi tyto ideální vztahy zřídka existují. Proto se musí počítat s cestami, které se odchyľují od tohoto idealizovaného modelu z mnoha důvodů (viz kapitulu 1). Tyto cesty obsahují kombinaci rozdílných zdrojů narušení ideálního zpoždění. Existuje tedy model vztahu mezi zpožděním v síti a geografickou vzdáleností definovaný následujícím způsobem.

Přímka *Bestline* pro daný RB_i je dána rovnicí

$$y = m_i x + b_i, \quad (4.2)$$

kteřá je položena níže než všechny body v grafu, ale zároveň je k těmto bodům nejbližší a má kladný průsečík s osou y . Také je nutno splnit podmínku, aby přímka *Bestline* neměla menší směrnici než *Baseline*, neboť by to jinak znamenalo, že reálné zpoždění je menší než ideální, což není možné.

Každý RB počítá svou vlastní *Bestline* s ohledem na všechny ostatní RB. Proto tato *Bestline* zachycuje nejméně zkreslený vztah mezi geografickou vzdáleností a zpožděním v síti z hlediska každého RB.

Hledání *Bestline* je formulováno jako lineární programovací problém. Pro daný RB_i je definováno síťové zpoždění d_{ij} a geografická vzdálenost g_{ij} ke každému RB_j , kde $i \neq j$. Pro každý RB_i se tedy musí najít směrnice m_i a posunutí b_i , jež určují *Bestline* výše uvedenou rovnicí (4.2). Podmínka, která říká, že *Bestline* pro každý RB_i by měla ležet pod všemi body grafu, je definována jako:

$$y - \frac{d_{ij} - b_i}{g_{ij}} x - b_i \geq 0, \forall i \neq j, \quad (4.3)$$

kde $m_i = \frac{d_{ij} - b_i}{g_{ij}}$ je směrnice *Bestline*. Funkce pro minimalizaci vzdálenosti mezi přímkou s kladným průnikem osy y a body označující naměřená zpoždění je uvedena jako:

$$\min_{\substack{b_i \geq 0 \\ m_i \geq m}} \left(\sum_{i \neq j} y - \frac{d_{ij} - b_i}{g_{ij}} x - b_i \right), \quad (4.4)$$

kde m je směrnice *Baseline*. Každý RB_i pak používá svou *Bestline* k převedení naměřeného zpoždění k cílovému hostiteli na zeměpisnou vzdálenost. To znamená, že odhadovaná hranice geografické vzdálenosti \widehat{g}_{iT} mezi RB_i a cílem T je odvozena z délky zpoždění \widehat{d}_{iT} za pomoci *Bestline* RB_i takto:

$$\widehat{g}_{iT} = \frac{d_{iT} - b_i}{m_i}. \quad (4.5)$$

Pokud jsou zpoždění mezi RB pravidelně shromažďována, vede to k automatickému kalibračnímu algoritmu, který určuje, jak každý RB nepřetržitě sleduje dynamický vztah mezi zpožděním a zeměpisnou vzdáleností v rámci sítě [23].

4.7.3 Použití distribuovaných hranic vzdáleností

CBG používá geometrický přístup používající multilateraci pro odhad umístění daného cílového hostitele T. Každý RB_i vyvozuje svou geografickou vzdálenostní hranici k cílovému hostiteli T, což je přídavná zkreslená vzdálenost $\widehat{g}_{iT} = g_{iT} + \gamma_{iT}$, viz rovnici (4.5). Proto každý RB_i odhaduje, že cíl T je někde uvnitř obvodu kruhu C_{iT} se středem v RB_i a poloměrem rovným odhadované geografické vzdálenostní hranici \widehat{g}_{iT} . Při daném K počtu referenčních bodů je cílový hostitel T umístěn v množině uzavřených křivek $C_T = \{C_{1T}, C_{2T}, \dots, C_{KT}\}$, kterou lze považovat za K -uspořádaný Vennův diagram². Z možných 2^K regionů, definovaných tímto diagramem, je nutno najít region R , jenž je vytvořen průnikem všech uzavřených křivek $C_{iT} \in C_T$, který je dán rovnicí

$$R = \bigcap_i^K C_{iT}. \quad (4.6)$$

Oblast R je konvexní, protože regiony C_{iT} jsou konvexní a průnik konvexních množin je také konvexní [23].

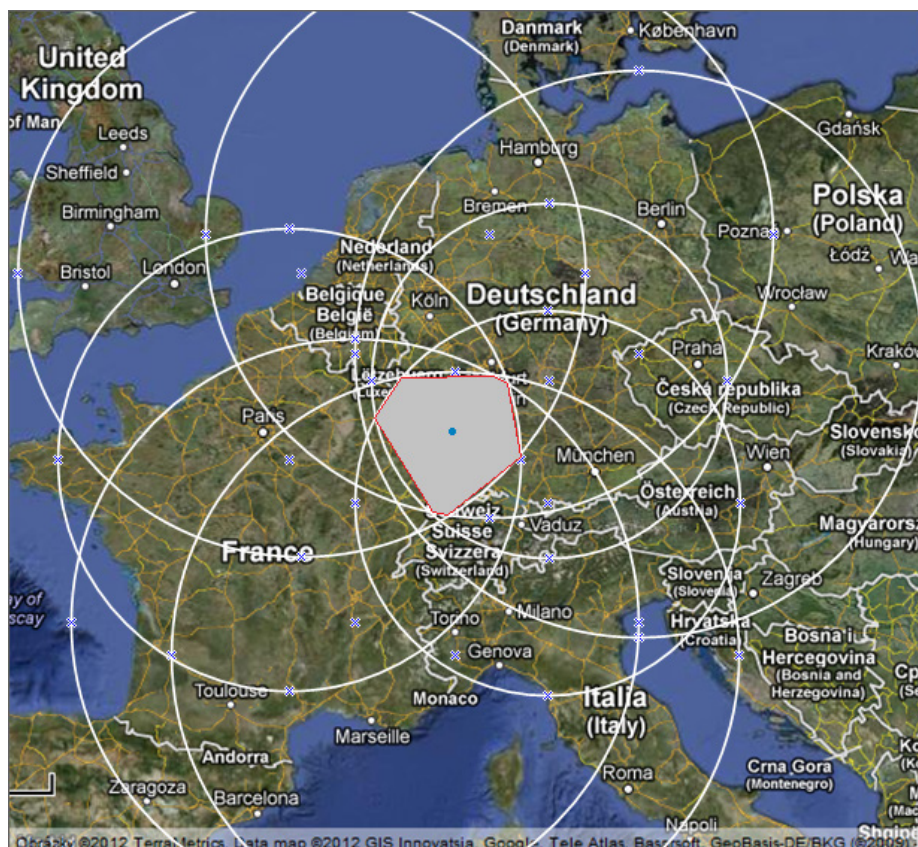
4.7.4 Geolokace cílové stanice

Region R je tedy odhad geografické oblasti, kde leží cíl a umístění cílového hostitele se pak nachází v těžišti tohoto regionu. Toto těžiště je pak bodový odhad cílové pozice. Pro popsání cílového regionu R je nutno přijmout heuristiku, že oblast se aproximuje na formu

² Informace o Vennově diagramu lze nalézt na webových stránkách: <http://astra.karlin.mff.cuni.cz/katedry/kdm/diplomky/moravecdp/mnoziny.php>.

polygonu. Výsledný polygon se pak používá k vypočítání odhadu bodové lokace cílového hostitele, což je těžiště polygonu.

Na obr. 4.5 je pro názornou demonstraci zobrazena tvorba polygonu a jeho těžiště na mapě Evropy z <http://maps.google.com/>. Bílé kružnice jsou obvody kruhů se středy v RB a poloměry rovnými hranicím vzdáleností k cíli. Oblast označená šedou barvou je výsledný polygon, jenž spadá pod všechny kruhy, a modrý bod je těžiště polygonu, tedy výsledný bodový odhad polohy cíle [23].



Obr. 4.5: Princip tvorby polygonu a jeho těžiště.

Vrcholy polygonu jsou průsečíky kružnic C_{iT} patřící do všech kruhů. Vzhledem k tomu, že je region R konvexní, je polygon podmnožina prostoru R . Následně dojde k aproximaci regionu R polygonem tvořeným úsečkami mezi N vrcholy $v_n = (x_n, y_n)$, $0 \leq n \leq N - 1$. Poslední vrchol $v_N = (x_N, y_N)$ je zároveň prvním vrcholem, polygon je tedy uzavřen [23].

Obsah A tohoto polygonu s vrcholy $v_0 = (x_0, y_0)$, ..., $v_{N-1} = (x_{N-1}, y_{N-1})$ je dán rovnicí

$$A = \frac{1}{2} \sum_{n=0}^{N-1} \begin{vmatrix} x_n & x_{n+1} \\ y_n & y_{n+1} \end{vmatrix}, \quad (4.7)$$

kde $|M|$ označuje determinant matice M . Těžiště c polygonu, tedy odhad pozice cílového hostitele T , je umístěn na pozici (c_x, c_y) , jež je dána rovnicemi

$$c_x = \frac{1}{6A} \sum_{n=0}^{N-1} (x_n + x_{n+1}) \begin{vmatrix} x_n & x_{n+1} \\ y_n & y_{n+1} \end{vmatrix} \quad (4.8)$$

a

$$c_y = \frac{1}{6A} \sum_{n=0}^{N-1} (y_n + y_{n+1}) \begin{vmatrix} x_n & x_{n+1} \\ y_n & y_{n+1} \end{vmatrix}. \quad (4.9)$$

4.7.5 Důvěryhodnost odhadnutého regionu

Z celkové plochy regionu R je odvozena důvěra, kterou CBG přiřadí výslednému odhadu. Tato oblast kvantifikuje geografickou oblast každého odhadu umístění v km^2 . Čím je menší oblast regionu R , tím je větší důvěryhodnost CBG odhadu.

Výše uvedená důvěryhodnost je důležitá pro aplikace využívající geolokaci, které chtějí vědět, do jaké míry se mohou spolehnout na daný odhad cíle. Tyto aplikace mohou mít rozdílné požadavky na přesnost a díky důvěryhodnosti poskytovaného odhadu umístění cíle mohou posoudit, zda má odhad dostatečné rozlišení s ohledem na jejich konkrétní potřeby.

Čím přesněji bude určena *Bestline*, tím tedy bude menší polygon R a odhad cíle bude přesnější. V nejideálnějším případě by hranice vzdáleností měly takovou velikost, že by se všechny takto vytvořené kružnice protínaly v jediném bodě, tedy v odhadované pozici cíle. Avšak v reálných podmínkách nastávají tři typy výsledků při stanovování hranic vzdáleností: nadhodnocení, podhodnocení nebo nadhodnocení i podhodnocení zároveň (anglicky *mismatch*) [23].

4.7.6 Potenciální výsledky při vytváření hranic vzdáleností

Nadhodnocení hranic

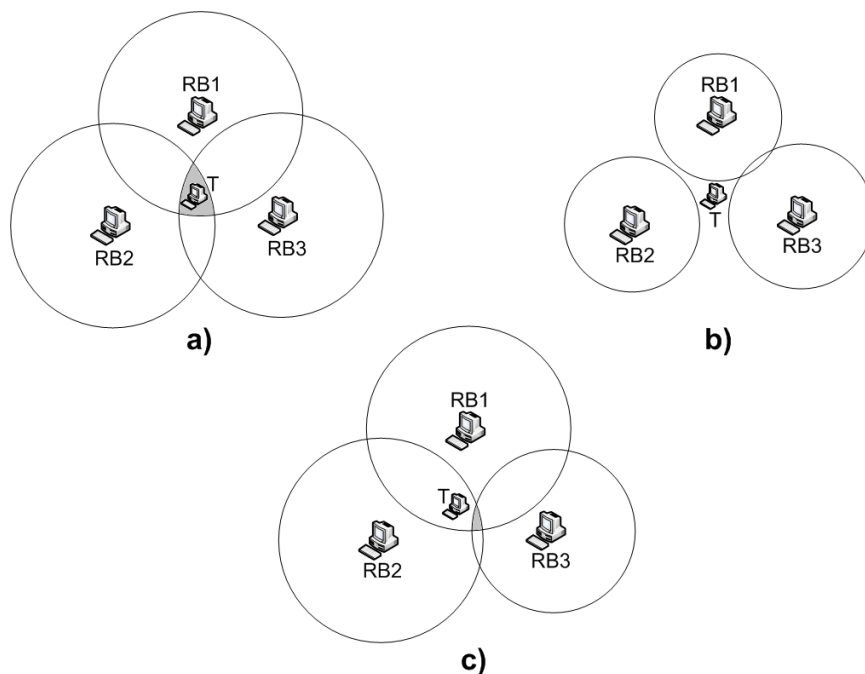
Tento případ uvedený na obr. 4.6 a) je nejběžnější. Hranice vzdáleností jednotlivých RB jsou poněkud větší, než je opravdová vzdálenost od RB k cíli. Důležité ale je, že tyto hranice dokážou vytvořit oblast R . Pokud toto nadhodnocení není příliš velké, tak ani oblast R není velmi rozlehlá, a CBG pozici cíle odhaduje relativně přesně [23].

Podhodnocení hranic

Tento stav, jenž je zachycen na obr. 4.6 b), je produktem chybné automatické kalibrace a jeho důsledkem je, že nedojde k odhadu pozice cíle. Tato situace nastane, pokud mají některé RB lepší vztahy mezi zpožděním a vzdáleností vůči cíli než oproti *Bestline*, tedy zbytku RB. Z toho plyne, že všechny nebo většina RB nevytvoří oblast R a tudíž se nemůže určit poloha cíle. Tato informace je důležitá pro rozhodnutí, zda vytvořit region R ze zbylých protínajících se hranic vzdáleností nebo radši prohlásit, že nelze určit pozici cíle. Vždy je totiž lepší polohu cíle neurčit vůbec, než ji odhadnout naprosto špatně [23].

Nadhodnocení i podhodnocení zároveň (anglicky *Mismatch*)

Na obr. 4.6 c) je zachycena situace, kdy RB_1 a RB_2 nadhodnotily své hranice vzdálenosti vůči cíli T , ale RB_3 svou hranici podhodnotil. Z této neshody pramení, že oblast R neobsahuje pozici cíle T . Tento stav se ale často nevyskytuje [23].



Obr. 4.6: Možné případy vytváření hranic vzdáleností ([10], str. 32).

4.7.7 Přesnost metody CBG v porovnání s algoritmem GeoPing

V publikaci [23] lze nalézt následující výsledky měření, které vytvořili autoři metody CBG. Na datasetu z USA byl medián chyby odhadu polohy cíle pomocí metody CBG v 80 % případů pod 100 km. Pro západoevropský dataset činil medián pod 25 km. Na stejných datasetech byl následně změřen medián i pro metody na bázi GeoPing. Tento činil pro americký dataset 150 km a pro západoevropský odpovídal hodnotě 100 km.

5. EXPERIMENTÁLNÍ SÍŤ PLANETLAB

Tato kapitola o síti PlanetLab je zde uvedena z důvodu využití jejích uzlů jako referenčních bodů. Na uzly lze prakticky neustále přistupovat a využívat jejich kapacit pro měření RTT zpoždění k cíli. Tyto uzly mají také známou zeměpisnou polohu, takže informace o pozici bude využita při praktické realizaci algoritmu CBG.

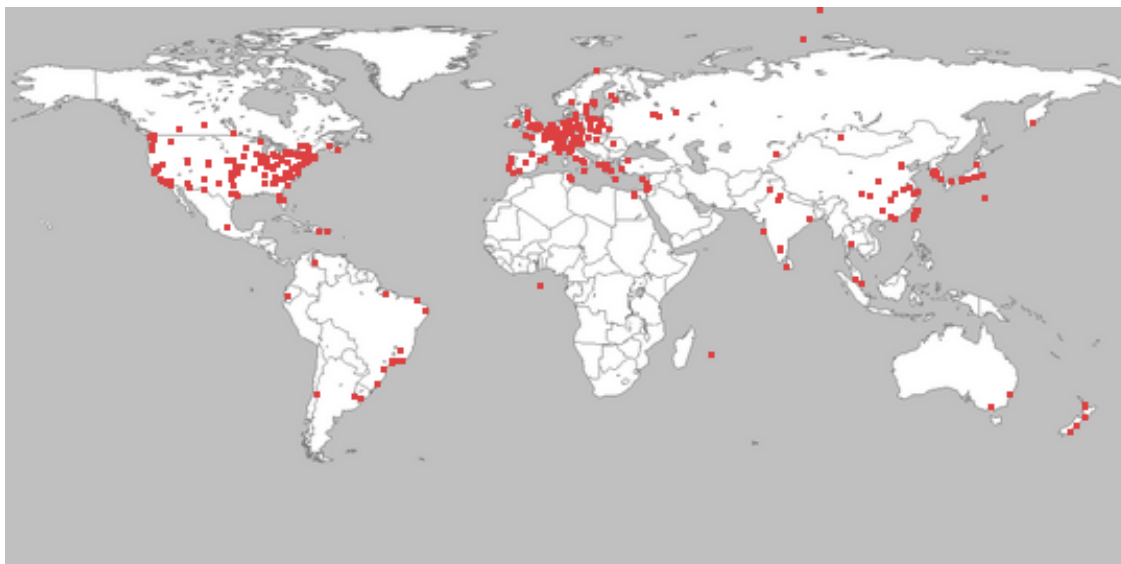
PlanetLab je celosvětová výzkumná síť, jež podporuje vývoj nových síťových služeb. Od roku 2003 se do tohoto vývoje zapojilo více než tisíc výzkumných akademických organizací a výzkumných průmyslových laboratoří [32].

Jejími členy z oblasti vývoje informačních technologií jsou například společnosti Google, Hewlett Packard, Bell Labs, Intel, France Telecom a dále také organizace zajišťující provoz Internetu v akademických kruzích [35]. Tyto firmy využívají síť PlanetLab k vývoji nových technologií pro distribuované ukládání dat, mapování sítě Internet, návrh nových komunikačních protokolů, P2P (*Peer To Peer*) systémy a podobně [33]. Provoz typu P2P probíhá přímo mezi samotnými účastníky Internetu bez nutnosti použití centrálního serveru [37].

Organizace, které se dlouhodobě angažovaly ve vývoji hardwaru a softwaru pro síť PlanetLab, vytvořily konsorcium poskytující podporu pro ostatní uživatele. Toto uskupení je také zodpovědné za další vývoj výzkumné sítě. Jejich spolupráce vedla k vytvoření softwarového balíčku MyPLC, díky němuž lze vytvořit vlastní PlanetLab servery členských výzkumných organizací. To umožnilo dále rozvíjet běžné balíčky služeb usnadňující vývoj nadstavbových služeb a aplikací nejen pro koncové uživatele, ale i pro zdokonalení samotné sítě PlanetLab [32].

Hlavním cílem sítě PlanetLab je kromě podpory výzkumu a testování v reálných podmínkách sítě i zpřístupnění vyvíjených služeb koncovým uživatelům pro usnadnění tvorby vlastních aplikací. Podpora obou úkolů je považována za nejdůležitější z hlediska dalšího vývoje a úspěchu systému [32].

Dnes se síť PlanetLab skládá z 1133 uzlů v 517 organizacích, které pracují na více než 600 aktivních projektech [34]. Jejich globální rozmístění ve světě lze vidět na obr. 5.1 [31]. V České republice se nacházejí 3 aktivní uzly provozované akademickým sdružením CESNET (*planetlab1.fit.vutbr.cz*, *planetlab1.cesnet.cz*, *planetlab2.cesnet.cz*) [34].



Obr. 5.1: Rozmístění PlanetLab organizací ve světě (převzato z [31]).

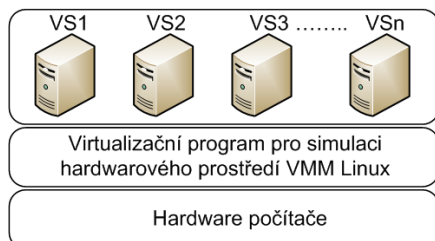
5.1 Názvosloví sítě PlanetLab

Níže jsou popsány termíny spjaté se sítí PlanetLab. Protože jsou dále v textu použity, je nutné uvést jejich význam pro jeho srozumitelnost [32].

- **Organizace** (anglicky *Site*): fyzické místo, kde se uzly v síti PlanetLab nacházejí (například CESNET – *Czech Education and Research Network*). Zjednodušená verze názvu místa je prefixem názvů všech projektů (viz dále).
- **Uzel** (anglicky *Node*): vzdálený server, na kterém jsou spuštěny jednotlivé služby sítě PlanetLab.
- **Projekt** (anglicky *Slice*): sada přidělených zdrojů distribuovaných v síti PlanetLab. Pro většinu uživatelů je projekt také konzolovým přístupem k serverům sítě PlanetLab. Vedoucí projektu je zodpovědný za jeho sestavení a přidělení jednotlivých projektů k uživatelům. Po přidělení k určitému projektu si k němu může uživatel přiřadit uzly. Následně po přidání uzlů k projektu je na každém přiděleném uzlu vytvořen příslušný virtuální server. Projekt má určitou dobu platnosti, která musí být pravidelně obnovována.
- **Sliver** (nemá český ekvivalent): přidělené zdroje na jednom uzlu sítě PlanetLab.
- **Virtuální server** (zkratka VS): Provedení sliverů je v podobě virtuálních serverů s OS GNU/Linux. VS realizuje oddělení jmenného prostoru a výkonu mezi slivery na jednom stroji. Síťovou virtualizaci sliverů umožňuje modul přístupu k virtuální síti (VNET).
- **Uživatel** (anglicky *User*): uživatelem je každý, kdo využívá, vyvíjí a instaluje aplikace na uzly sítě PlanetLab.

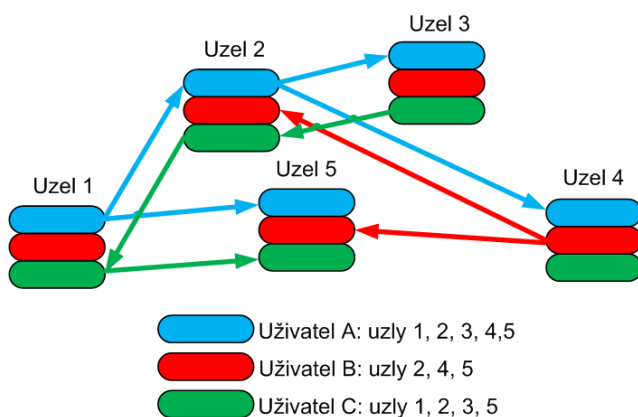
5.2 Technické řešení sítě PlanetLab

Všechny uzly v síti PlanetLab používají operační systém GNU/Linux v distribuci CentOS (*Community ENTERprise Operating System*) [34]. Nový uživatel této experimentální sítě, který se úspěšně zaregistruje a patří k určitému projektu (*slice*), se může vzdáleně připojit k jakémukoliv uzlu, kde má k dispozici vlastní sliver v podobě nezávislého virtuálního stroje (VS) s vlastním souborovým systémem a výpočetní kapacitou. Takovýto uzel lze popsat zjednodušeným vrstevným modelem dle obr. 5.2 [33].



Obr. 5.2: Zjednodušený vrstevný model uzlu sítě PlanetLab ([33], str. 32).

Na každém uzlu sítě PlanetLab může současně pracovat více uživatelů. To znamená, že na každém z nich je umožněn současný chod více projektů. Jednotliví uživatelé pracují ve vlastním virtuálním systému a mohou mít spuštěno více aplikací, viz obr. 5.3 [33].



Obr. 5.3: Princip rozmístění uživatelů na uzlech v síti PlanetLab ([33], str. 32).

5.3 Vybrané projekty běžící v síti PlanetLab

V síti PlanetLab v současné době probíhá více projektů. Několik hlavních projektů je zde dále představeno. Snahy těchto experimentů jsou především takové, aby každý uživatel Internetu s sebou nemusel nosit přenosný počítač, ale veškeré aplikace a dokumenty byly dostupné přímo na síti. Zároveň nastává potřeba vytvořit síť odolnou vůči virům [36].

- **CoDeeN:** systém, který v případě požadavku na určitý soubor od mnoha uživatelů vytvoří kopie souboru tak, aby těchto kopií byl vhodný počet. To znamená, aby nebyly přetížené servery při velkém počtu kopií, nebo nenastala dlouhá čekací doba

uživatelů při nízkém počtu kopií. CoDeeN může být využit například u často navštěvovaných webových stránek, které se replikují na jiné uzly. S tímto projektem souvisí i další podobné projekty, jež se většinou zabývají aplikacemi na správu a monitoring sítě PlanetLab, jmenovitě CoTop a CoVisualize [36].

- **OceanStore:** systém na globální uchovávání dat. Data uživatelů jsou rozdělena na fragmenty, které jsou neustále posouvány sítí z místa na místo. Taková data jsou potom v podstatě nezničitelná. Například pokud některý uzel v síti vypadne, najde se vždy kopie fragmentu někde jinde [36].
- **Scriptroute Network Measurement:** tento projekt nabízí celou řadu měřících nástrojů pro uživatele i pro developery. Slouží k měření odezev v síti, mapování směrovačů a podobně [36].

Mimo výše uvedené projekty, jež se v odborných kruzích označují jako zdrojové, jsou v síti PlanetLab stovky dalších, které se jednak pokoušejí vyvíjet nové síťové technologie, ale také například slouží pro monitoring a práci s uzly. Síť také funguje jako úložiště pro webové stránky celé řady univerzit a společností [36].

Pro nejrozšířenější světovou encyklopedii Wikipedia se zde testují možnosti efektivní distribuce velkých objemů dat k co nejširšímu okruhu uživatelů s použitím několikasupňové replikace. Společnost Google využívá PlanetLab pro vyhodnocování dostupnosti jejích serverů z různých částí světa. Hewlett Packard zase řeší jiný problém z této oblasti, který se týká vzájemné dostupnosti účastníků z různých domén [37].

5.4 CentOS – distribuce operačního systému GNU/Linux

Distribuce CentOS je postavena na RedHat distribuci, což je jedna z nejrozšířenějších distribucí v České republice. Konkrétně se jedná o distribuci RedHat Enterprise Linux, kterou firma RedHat vydává pouze komerčně, a to v binární podobě. Z licenčních důvodů otevřeného a volně šiřitelného softwaru jsou zdrojové kódy zcela volně přístupné na FTP severu [38].

CentOS je oproti distribuci RedHat volně k dispozici, ale s nevýhodou, že není provozován ani podporován firmou RedHat. Řadí se také mezi nejvíce využívané distribuce na webových serverech [38].

Vybrané vlastnosti distribuce CentOS [38]:

- Aktualizace jsou vydávány se zpožděním a provádí se pomocí nástroje *yum*.
- V distribuci CentOS neexistuje náhrada za placené služby u distribuce RedHat.
- Správu instalačních balíčků má na starost *RPM (RPM Package Manager)*.
- Při kompilaci balíčků se snaží o takzvaný *self-hosting* (distribuce je překládána sama do sebe).

6. VYTVOŘENÍ MNOŽINY REFERENČNÍCH BODŮ A CÍLŮ

Pro následnou realizaci algoritmu CBG, jeho správné měření a relevantní vyhodnocení přesnosti je nutné vytvořit kvalitní množinu, tedy dataset skládající se z dostatečného počtu referenčních bodů a cílů, jejichž zeměpisné polohy musí být známy a tato informace musí být pravdivá, aby nezkreslovala výsledky měření.

6.1 Referenční body

Jako referenční body byly zvoleny uzly sítě PlanetLab. Úkolem vybrat RB byl pověřen kolega Jakub Chmelař, který vytvořil množinu obsahující 140 RB. Její výčet se nachází na příloženém CD. Tento dataset byl vybírán pouze z uzlů v Evropě. Polohy vybraných uzlů také splňují podmínku souměrné rozprostřenosti. Pokud by stanice byly blízko sebe, tak by tvořily shluky, což by mělo za následek, že by výsledná měření provedená z RB ve shlucích byla velmi podobná, a tudíž by takové měření prakticky stačilo pouze jedno.

Tyto uzly také musí být většinu času dostupné, aby mohlo být prováděno více měření. Proto je většina RB zastoupena organizacemi (viz kapitulu 5.1), které obsahují více uzlů se stejnou geografickou pozicí. Tímto se tedy eliminuje riziko nedostupnosti RB, protože je velká pravděpodobnost, že aspoň jeden uzel bude vždy dostupný. Pro následné měření bylo vybráno Bc. Peterem Troppem 40 RB z výše uvedeného datasetu, které zastupují většinu organizací a jsou většinu času dostupné. Na obr. 6.1 lze vidět zeměpisné polohy referenčních bodů v Evropě na mapě z <http://maps.google.com/>. Jejich adresy spolu se souřadnicemi jsou uvedeny na webovém serveru³ [49], v příloze a také na příloženém CD.



Obr. 6.1: Zeměpisné polohy referenčních bodů v Evropě.

³ Přihlašovací údaje sdělí doc. Ing. Dan Komosný, Ph.D., email: komosny@feec.vutbr.cz.

6.2 Cíle pro geolokaci

Jako cíle byly zvoleny DNS (*Domain Name System*) uzly v Evropě, které obsahují DNS záznam LOC. Následuje popis tohoto typu DNS záznamu.

6.2.1 DNS záznamy LOC

Záznamy LOC jsou volitelnou součástí DNS systému a jsou podrobně definovány v RFC 1876 (*Request for Comments*), viz [39]. Jejich význam spočívá v tom, že informují o geografické poloze příslušného doménového jména.

Záznam LOC obsahuje informaci o zeměpisné šířce, délce a nadmořské výšce ve formátu souřadnicového systému WGS84 (*World Geodetic System 1984*), což je pravotočivá kartézská soustava souřadnic se středem v těžišti Země. Tímto modelem je popsána Země v podobě elipsoidu. Podrobněji viz [40].

Tato LOC informace pak může být vyžádána ostatními počítači připojenými do Internetu. Příklad DNS LOC záznamu pro doménové jméno *yahoo.com* se zeměpisnou polohou o hodnotách 37° 23' 30.9" severní šířky, 121° 59' 19" západní délky a 7 m nadmořské výšky je následující: LOC record yahoo.com. IN LOC 37 23 30.900 N 121 59 19.000 W 7.00m [39].

6.2.2 Postup vyhledání cílů

Vyhledáním cílů byli pověřeni Bc. Petr Harth, Bc. Filip Dvořák a Bc. Peter Tropp. Bylo zjištěno, že neexistuje databáze domén s vyplněnými DNS záznamy LOC. Webové stránky <http://www.ckdhr.com/dns-loc/sites.html> zabývající se danou problematikou obsahují pouze mapy s vyznačenými polohami domén bez jejich doménového jména či IP adresy. Také jsou zastaralé (poslední aktualizace v roce 2001) a jejich tvůrci neodpovídají na emaily s dotazy týkajícími se informací o názvech domén s LOC záznamy.

Další webová stránka <http://dns-loc.mapper.ofdoom.com/> obsahuje skript, jenž při ručním zadání jména domény vrátí obsah LOC záznamu, pokud jej daná doména obsahuje. Nalezené domény pak ukládá do souboru, který by měl být volně dostupný. Bohužel bylo zjištěno, že tento soubor obsahuje pouze 26 evropských domén s LOC záznamy, jež odpovídají na ECHO dotaz programu *Ping*. Proto autor této diplomové práce oslovil tvůrce uvedených stránek Jeffa Hungerforda, který na emaily s dotazy ohledně zmiňované problematiky odpověděl.

Tímto došlo při vytváření datasetu cílových stanic k mezinárodní spolupráci, neboť Jeff Hungerford uvolnil svůj skript pro volné použití a taktéž autorovi poradil, aby vyhledal domény ze seznamu DMOZ (*Open Directory Project*), což je největší lidmi spravovaný katalog odkazů [41].

Možným řešením tedy je zeptat se pomocí výše zmíněného skriptu s názvem *LOClookup* co největšího počtu domén, jestli nemají tento záznam vyplněn. Bohužel ale neexistuje volně zveřejněný seznam všech názvů domén, proto byly zvoleny již výše zmíněné domény ze seznamu DMOZ, který k datu 8. 12. 2011 obsahuje přibližně 4,5 milionu adres.

Skript Jeffa Hungerforda, jenž je napsán v programovacím jazyce *Pike*⁴, posílá dotazy na stále stejný veřejný DNS server. Jelikož je nutno odeslat počet dotazů rovný počtu doménových jmen, což je 4,5 milionu, může tím dojít k neúměrnému zatížení DNS serveru s následným prodloužením dob odpovědí.

Proto členové výzkumné skupiny Bc. Petr Harth, Bc. Filip Dvořák a Bc. Peter Tropp vytvořili skript napsaný v jazyce *Python*⁵ s názvem *DNSrotateLOClookup*, jenž používá skript *LOClookup* a z důvodu rozložení zátěže mezi více DNS serverů se střídavě dotazuje 16 veřejných DNS serverů⁶. Jeho jediný vstupní parametr je odkaz na soubor s doménami, jež mají být prohledány. Nalezená doménová jména s LOC záznamy jsou pak uložena jako výstupní parametry do souboru.

Okomentované skripty *LOClookup* a *DNSrotateLOClookup* jsou uvedeny v příloze, aby nenarušovaly celistvost a čitelnost diplomové práce.

Pomocí výše zmíněných skriptů bylo zjištěno, že se LOC záznamy často nepoužívají, neboť ze 4,5 milionu adres bylo nalezeno pouze 181 domén s LOC záznamy. Následně se tyto domény testovaly, zda reagují na ECHO dotaz programu *Ping*, což je důležitá vlastnost pro jejich použití v metodách na bázi měření. Tuto podmínku splňuje pouze 143 domén, z toho pouze 90 domén umístěných v Evropě. Následně byly uvedené geografické polohy domén zobrazeny na mapě a uzly, jež leží například v moři nebo v neobydlené krajině, byly ze seznamu vymazány. Tímto způsobem vznikla množina 76 cílů.

Příčina nízkého počtu domén s vyplněným záznamem může nabývat více podob, neboť LOC záznamy musí přidávat správci domén ručně, takže to pro správce znamená více práce. Tento parametr je nepovinný a obecně málo používaný, proto není potřeba jej mít vyplněn.

V neposlední řadě zde hraje roli bezpečnostní riziko v podobě zneužití takovéto informace útočníky, kdy může například zloděj vystopovat přesné umístění serverů a ukrást je, nebo útočník může v dané lokalitě přerušit dodávku proudu a pokud nejsou servery vybaveny alternativním zdrojem energie pokrývajícím výpadky veřejné elektrické sítě, tak dojde k vypnutí serverů a následně k jejich nedostupnosti.

⁴ Více o programovacím jazyce *Pike* viz <http://pike.ida.liu.se/>.

⁵ Podrobněji programovací jazyk *Python* viz <http://python.org/>.

⁶ Adresy veřejných DNS serverů jsou uvedeny na <http://www.tech-faq.com/public-dns-servers.html>.

Nalezené domény s LOC záznamy také musí splňovat podmínku časté dostupnosti pro opakovaná měření. Neméně důležitá je jejich vlastnost odpovídání na ECHO dotaz programu *Ping*, viz kapitolu 2.2.

Na obr. 6.2 lze vidět zeměpisné polohy domén s DNS LOC záznamem v Evropě na mapě z <http://maps.google.com/>. Jejich adresy spolu se souřadnicemi jsou uvedeny na webovém serveru [49], na příloženém CD a také v příloze.



Obr. 6.2: Zeměpisné polohy domén s DNS LOC záznamem v Evropě.

7. REALIZACE ALGORITMU CONSTRAINT-BASED GEOLOCATION

Hlavní část této diplomové práce spočívá v realizaci algoritmu CBG v podobě aplikace napsané v jazyce *Java*⁷, ověření její funkce na uměle zadaných vstupních datech a nasazení v experimentální síti PlanetLab z důvodu měření vlastností algoritmu CBG na reálných datech. Výsledky programu jsou následně v další kapitole srovnány s dalšími metodami, jež realizují ostatní autoři kolegové. Celý program i s jeho komentáři je psán v angličtině z důvodu srozumitelnosti [43], neboť je možné, že jej bude chtít přečíst i cizinec.

Implementace CBG metody je nazvána příznačně *CBGfinder*. *CBGfinder* je rozdělen na serverovou a klientskou část, neboť je důležité rozlišit operace, které budou zpracovávány servery, jež jsou umístěny na referenčních bodech v podobě uzlů sítě PlanetLab a operace vykonávané na klientovi, což je počítač umístěný v Ústavu telekomunikací Fakulty elektrotechniky a komunikačních technologií Vysokého učení technického v Brně. Tento počítač také sice obsahuje server s webovými službami [49], kde lze zadat IP adresu či doménové jméno cíle, ale z pohledu programu *CBGfinder* vystupuje v pozici klienta, neboť se bude dotazovat referenčních bodů na jejich vzdálenost k cíli, čili servery poskytují klientovi výše uvedené služby.

7.1 Postup vyhledání odhadované polohy cíle

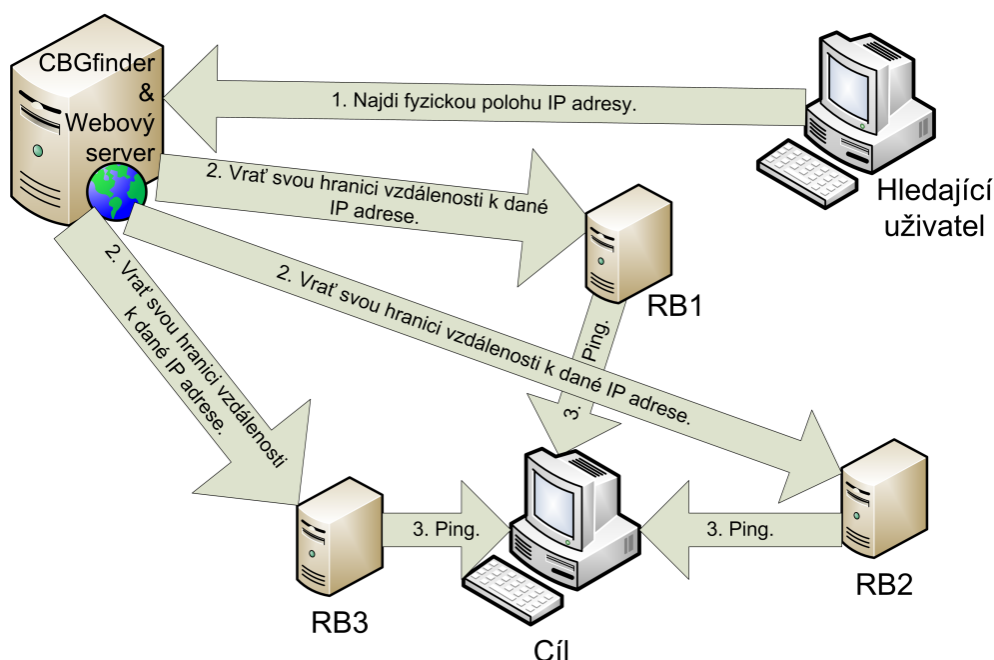
Před konkrétním vysvětlením jednotlivých částí aplikace je nutno získat obecnou představu, jak program celistvě funguje a jak spolupracují jeho části. Proto je v následujících krocích přehledně popsán postup vyhledání odhadované polohy cíle aplikací *CBGfinder*, který koresponduje s dále uvedenými vývojovými diagramy.

1. Spuštění programu *CBGfinder* se provede pomocí webového geolokačního serveru [49], do jehož formuláře hledající uživatel zadá IP adresu či doménové jméno cíle. Server tedy spustí klientskou část programu a předá jí vstupní parametry, tedy IP adresu cíle a informace o RB, jež mají být využity pro měření.
2. Klientská část programu *CBGfinder* se zeptá na hranici vzdálenosti daných RB k cíli, kdy jim pošle IP adresu cíle.
3. Serverové části programu *CBGfinder* na referenčních bodech IP adresu cíle přijmou a provedou na ni ECHO dotaz pomocí programu *Ping*.

⁷ Více o programovacím jazyce *Java* viz <http://www.oracle.com/technetwork/java/index.html>.

4. Jelikož referenční body provádí mezi sebou v určitém intervalu automatickou kalibraci podle kapitoly 4.7.2, jsou schopny převést RTT zpoždění k cíli na hranici vzdálenosti a tuto hranici vrátit klientské části programu *CBGfinder*.
5. Klient na počítači na univerzitě vzdálenosti přijme a následně vypočítá odhadovanou pozici cíle dle postupů a rovnic v kapitole 4.7.4.
6. Nakonec tuto polohu vrátí geolokačnímu serveru, který ji zobrazí uživateli na mapě pro názornou představu, kde cíl přibližně leží. Také dojde k předání hodnoty doby vyhledání cíle spolu s velikostí odhadované cílové oblasti.

Komunikace při vyhledávání odhadované polohy cíle je zachycena na obr. 7.1.



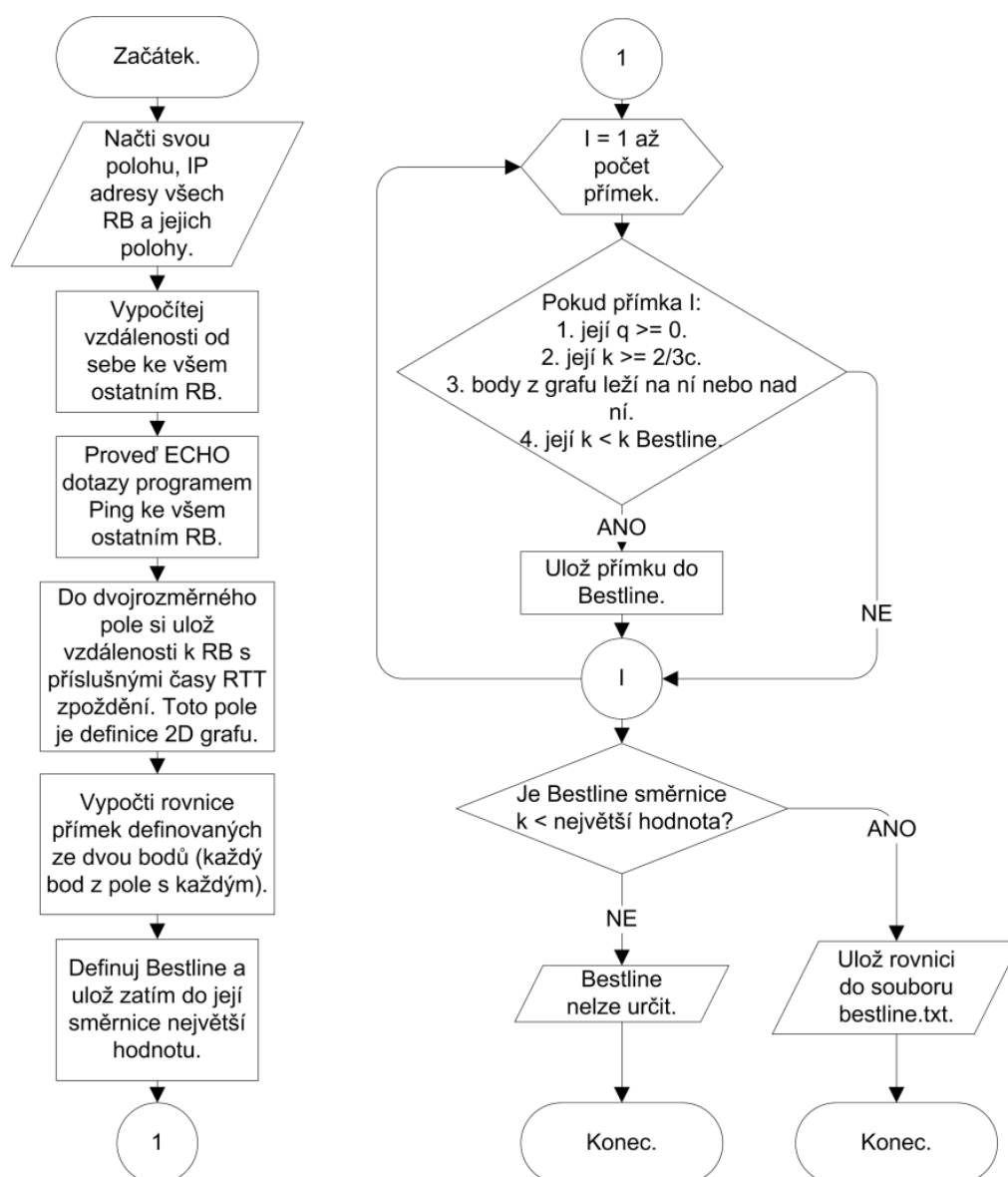
Obr. 7.1: Komunikace při vyhledávání odhadované polohy cíle.

7.2 Serverová část

Část programu *CBGfinder*, jež vystupuje v pozici serveru, je umístěna na uzlech sítě PlanetLab. Dle kapitoly 4.7 musí být referenční body (RB) schopny použít program *Ping* za účelem ECHO dotazu na adresu cíle. Takovýto dotaz je aktivován na vyžádání od klientské části (viz kapitolu 7.1), která poskytne IP adresu hledaného cíle. Také je vyžadováno automatické kalibrování mezi RB, jež spočívá v pravidelném dotazování se navzájem mezi RB pomocí programu *Ping*. Proto je důležité rozlišit následující dva podprogramy a náležitě je popsat.

7.2.1 Vypočítání přímky *Bestline*

Vypočítání přímky *Bestline* je součást automatické kalibrace, jejíž princip je popsán v kapitole 4.7.2 a je implementována programem *computeBestline.jar*. Na obr. 7.2 je znázorněn jeho vývojový diagram. Výsledkem automatické kalibrace je přímka *Bestline* s matematickým zápisem ve formátu $y = kx + q$, kde k je směrnice přímky a q je její posunutí. Parametry k a q se uloží do souboru *bestline.txt*, odkud je následně přečte program *cbgServer.jar*, jenž převádí naměřené zpoždění na hranici vzdálenosti. Program *cbgServer.jar* je popsán v následující kapitole.

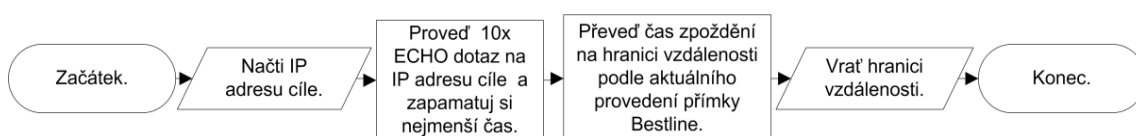


Obr. 7.2: Vývojový diagram automatické kalibrace.

7.2.2 Převod hodnoty zpoždění na hranici vzdálenosti

Převod hodnoty zpoždění na hranici vzdálenosti je implementován programem *cbgServer.jar*, jenž je spouštěn démonem, který je popsán v další kapitole. Tento démon předává IP adresu cíle, na níž se provede desetkrát ECHO dotaz programem *Ping*, a nejmenší hodnota zpoždění RTT je dále využita pro převod na hranici vzdálenosti podle rovnice $y = kx + q$, kdy se za y dosadí naměřené zpoždění, za k a q hodnoty ze souboru *bestline.txt* a vypočítá se x , tedy hodnota hranice vzdálenosti.

Stručný vývojový diagram převodu hodnoty zpoždění na hranici vzdálenosti je zachycen na obr. 7.3.



Obr. 7.3: Vývojový diagram převodu hodnoty zpoždění na hranici vzdálenosti.

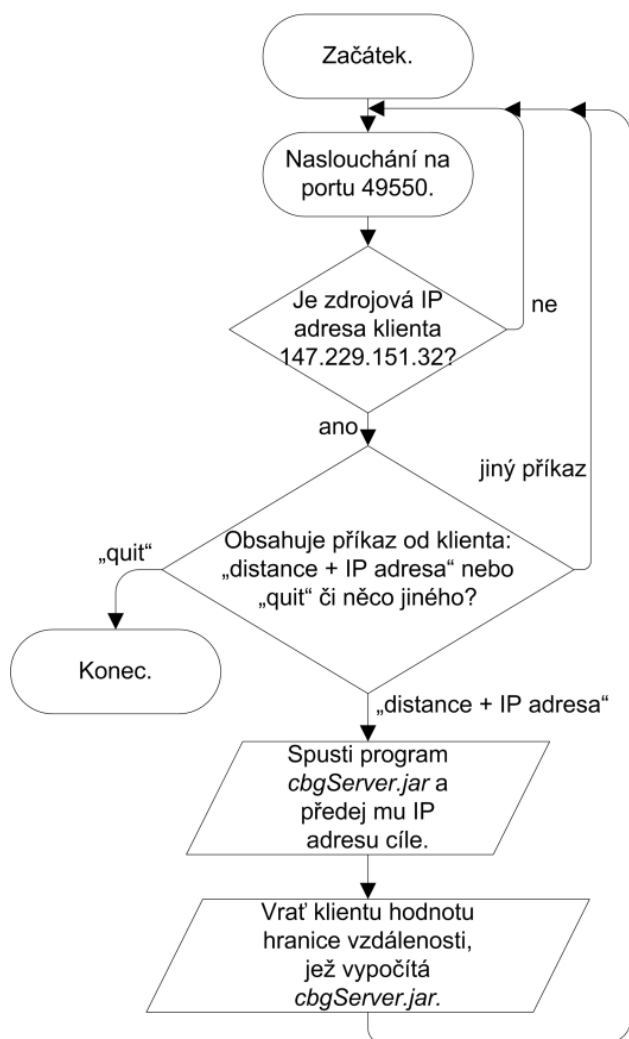
7.3 Komunikace serveru s klientem

O komunikaci serveru s klientem se stará program *tcpConnectionServer.jar* spuštěný na referenčních bodech. Je to program běžící na pozadí, což je v podstatě démon, který naslouchá na portu 49550, na nějž se připojuje klient. Číslo portu bylo vybráno z rozsahu dynamických, soukromých nebo dočasných portů⁸.

Aby bylo zamezeno zneužití otevřeného portu, je definována podmínka, že komunikovat přes tento port lze jen se zdrojovou IP adresou 147.229.151.32, tedy IP adresou počítače na univerzitě, na níž se nachází klientská část aplikace *CBGfinder*.

TCP spojení typu klient-server bylo vybráno z důvodu urychlení celkové doby komunikace, neboť odlišný způsob výměny dat, a to přes SSH (Secure Shell) spojení, trval pro jednu relaci až 40 sekund. Kdežto komunikace typu klient-server je provedena během 10 sekund.

⁸Viz webové stránky <http://www.iana.org/assignments/port-numbers>.



Obr. 7.4: Vývojový diagram démona starajícího se o komunikaci serveru s klientem.

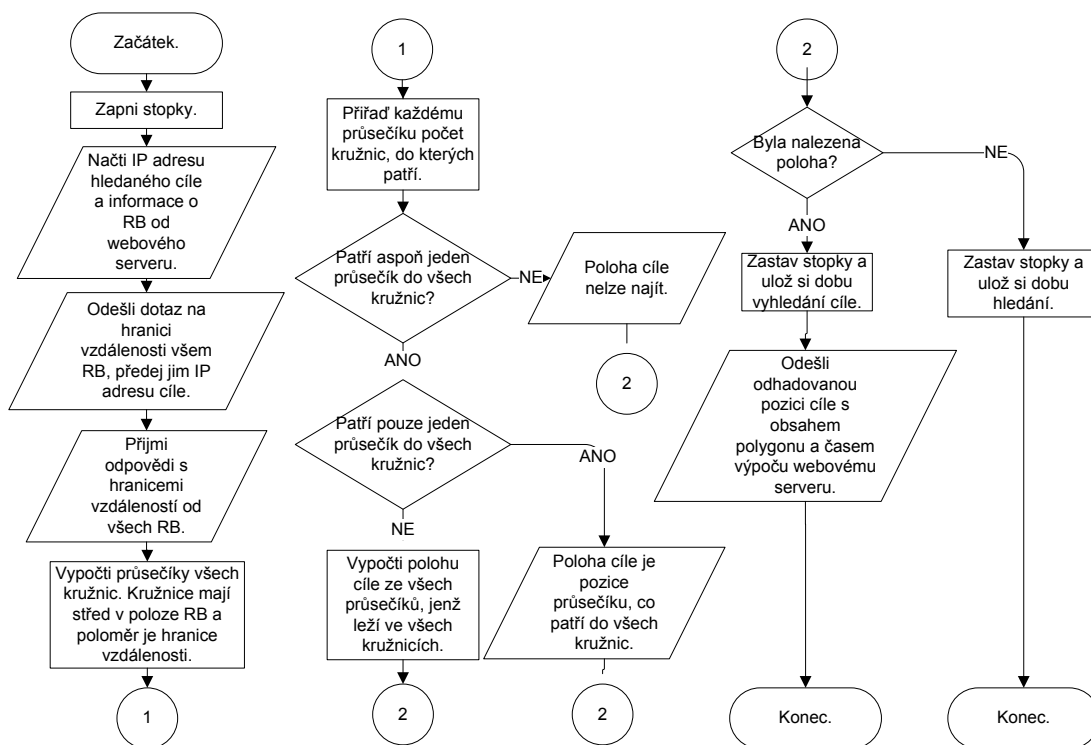
7.4 Klientská část

Za klientskou část programu *CBGfinder* je považována implementace programu na počítači umístěném v Ústavu telekomunikací Fakulty elektrotechniky a komunikačních technologií Vysokého učení technického v Brně. Tato stanice totiž musí zvládat dotazování se referenčních bodů na informaci o jejich vzdálenosti k cíli.

Nejprve je ale nutno přijmout IP adresu hledaného cíle od webového serveru [49], jenž vypracoval a udržuje Ing. Lukáš Verner. Na tomto webovém serveru běží webové geolokační stránky, kde jsou uvedeny informace o datasetech RB a cílů. Také je zde formulář pro zadání IP adresy či doménového jména stanice, jež je nutno lokalizovat.

Následně se klientská část postará o shromáždění potřebných dat z RB a vyhodnotí odhadovanou pozici cíle, kterou vrátí geolokačnímu serveru spolu s dobou hledání cíle a obsahem výsledného polygonu, v jehož středu leží zaměřená pozice cíle. Webové stránky také danou polohu zobrazí uživateli na mapách <http://maps.google.com/>.

Na obr. 7.5 je zakreslen vývojový diagram klientské části.



Obr. 7.5: Vývojový diagram klientské části.

7.4.1 Paralelní zpracování komunikace s referenčními body ze sítě PlanetLab

Z celkové doby výpočtu aplikace zabírá komunikace s referenčními body ze sítě PlanetLab asi 99 % času, proto je nutno tento děj paralelizovat. Implementace paralelizace využívá programovacího jazyku *Clojure*, jenž zahrnuje agenty, díky nimž je paralelní zpracování optimalizováno [44].

Agent je mechanismus pro řízení přístupu ke sdílenému stavu (sdíleným datům). Je podobný *actor* modelu z programovacího jazyka *Erlang*⁹. *Actor* je objekt, který dostává určité zprávy a vzápětí na ně reaguje posíláním zpráv ostatním objektům. Vše se děje ve stejný čas, tedy v samostatných vláknech. Rozdíl je v tom, že zatímco *actor* je autonomní (běží stále a většinu času čeká), agent je reaktivní, neboť na svůj vstup dostane funkci, která jako vstupní parametr bere současný stav (data) a jejím výstupem je nový stav (modifikovaná data). Tato funkce automaticky běží na pozadí a stav agenta (jeho data) se změní atomicky, tedy vždy se čtou konzistentní data, nikdy nedochází k přechodu nových a starých dat zároveň [44].

V implementaci aplikace *CBGfinder* obsahují data každého agenta seznam IP adres RB a hranic vzdáleností cíle. Funkce pro spočítání nového stavu agenta přebere tento seznam, funkci pro komunikaci s RB a IP adresu cíle. Následně se klient zeptá serveru na hranici vzdálenosti

⁹Webové stránky programovacího jazyku *Erlang* jsou <http://www.erlang.org/>.

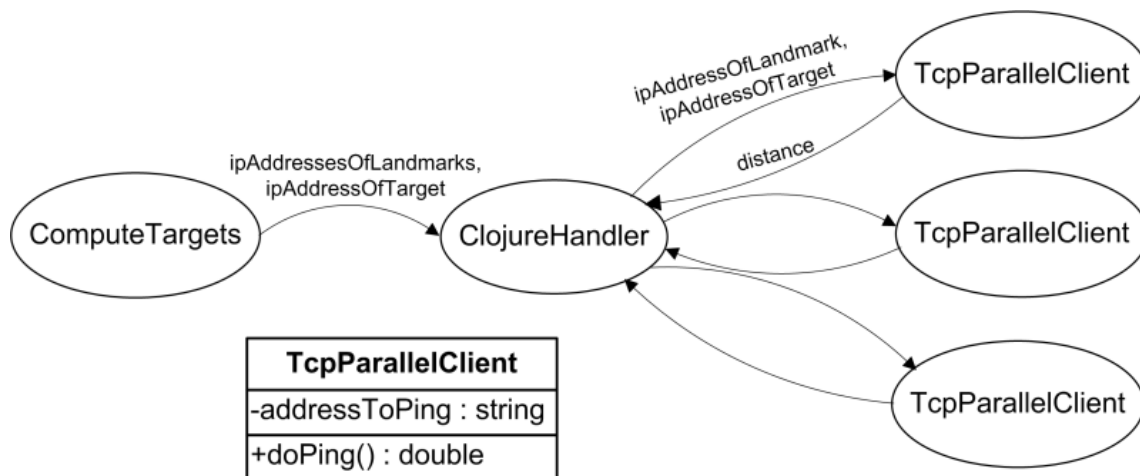
k cíli a pokud ji server poskytne, tak je přidána do soupisu a tento seznam s novou položkou je nový stav agenta.

V Javě by obdobný zápis vypadal následovně za předpokladu, že by bylo možno předat funkci jako parametr funkce a že by metoda *List.Insert* nemodifikovala původní seznam, ale vracela nový, což v jazyku *Java* implementovat nelze.

```
private List<IpAddress, Distance> getNewState(List<IpAddress, Distance>
    oldState, IpAddress newAddressToPing, Function<IpAddress, Distance>
    pingFunction)
{
    Distance newDistance = pingFunction.Invoke(newAddressToPing);
    List<IpAddress, Distance> newState = oldState.Insert(newAddressToPing,
        newDistance);
    return newState;
}
```

Pro názornou představu o implementaci paralelní komunikace je na obr. 7.6 uveden diagram datových toků paralelní komunikace se třídou agenta. Diagram datových toků slouží pro přehledné modelování funkcí informačních systémů [45]. Názvy funkcí korespondují s anglickým označením použitým v programu. Metoda *ComputeTargets* předá funkci *ClojureHandler* seznam IP adres RB (*ipAddressesOfLandmarks*) a IP adresu cíle (*IpAddressOfTarget*). Následně *ClojureHandler* vytvoří pro každý RB agenta *TcpParallelClient*, jenž se postará o komunikaci s příslušným RB. *ClojureHandler* příslušnému agentu předá IP adresu RB (*ipAddressOfLandmark*) a IP adresu cíle (*IpAddressOfTarget*). Agent posléze vrací hodnotu hranice vzdálenosti *distance*.

Třída *TcpParallelClient* obsahuje atribut typu *string* s názvem *addressToPing*, do nějž se ukládá IP adresa cíle. Dále třída zahrnuje metodu *doPing()*, díky níž probíhá komunikace se serverem na RB.



Obr. 7.6: Diagram datových toků paralelní komunikace a třída agenta.

7.4.2 Vstupní a výstupní formát dat

Klientská část aplikace *CBGfinder* načítá při svém spuštění dva vstupní parametry, a to seznam referenčních bodů, jež mají být využity při zaměření cílové pozice, spolu s množinou cílů. Tato vstupní data jsou zapsána ve formátu JSON (*JavaScript Object Notation*)¹⁰, jenž se často využívá pro výměnu dat mezi jednotlivými programy.

Pro ukázkou následuje tvar vstupního pole pro dva referenční body sítě PlanetLab:

```
[{"id":6,"idsite":1,"idmethod":4,"ip":"147.229.10.250","hostname":  
"planetlab1.fit.vutbr.cz","latitude":"49.22684","longitude":"16.596152",  
"name":"CESNET_-_Czech_Education_and_Research_Network","reliability":88.76,  
"method":"Planetlab","cost":100,"image":"planetlab.png"}, {"id":121,"idsite":  
53,"idmethod":4,"ip":"195.37.16.121","hostname":  
"aladdin.planetlab.extranet.uni-passau.de","latitude":"48.568333","longitude":  
"13.453333","name":"University_of_Passau","reliability":98.63,"method":  
"Planetlab","cost":100,"image":"planetlab.png"}].
```

Vstupní data například pro cíl s DNS jménem *switch.ch* vypadají následovně:

```
[{"id":286,"idmethod":6,"ip":"130.59.108.36","hostname":"switch.ch",  
"latitude":"null","longitude":"null","method":"CBG","image":"CBG.png",  
"comment":"null", "duration":"null"}].
```

Program po výpočtu zeměpisné polohy do pole s informacemi o cíli doplní naměřenou šířku a délku, čas výpočtu a do pole *comment* uvede obsah polygonu, jehož střed odpovídá odhadnuté pozici cíle. Výstupní pole dat pak pro cíl s DNS jménem *switch.ch* je ve formátu:

```
[{"id":286,"idmethod":6,"ip":"130.59.108.36","hostname":"switch.ch",  
"latitude":"47.33642240126","longitude":"8.5302530711867","method":"CBG",  
"image":"CBG.png", "comment":"44021","duration":"59.67"}].
```

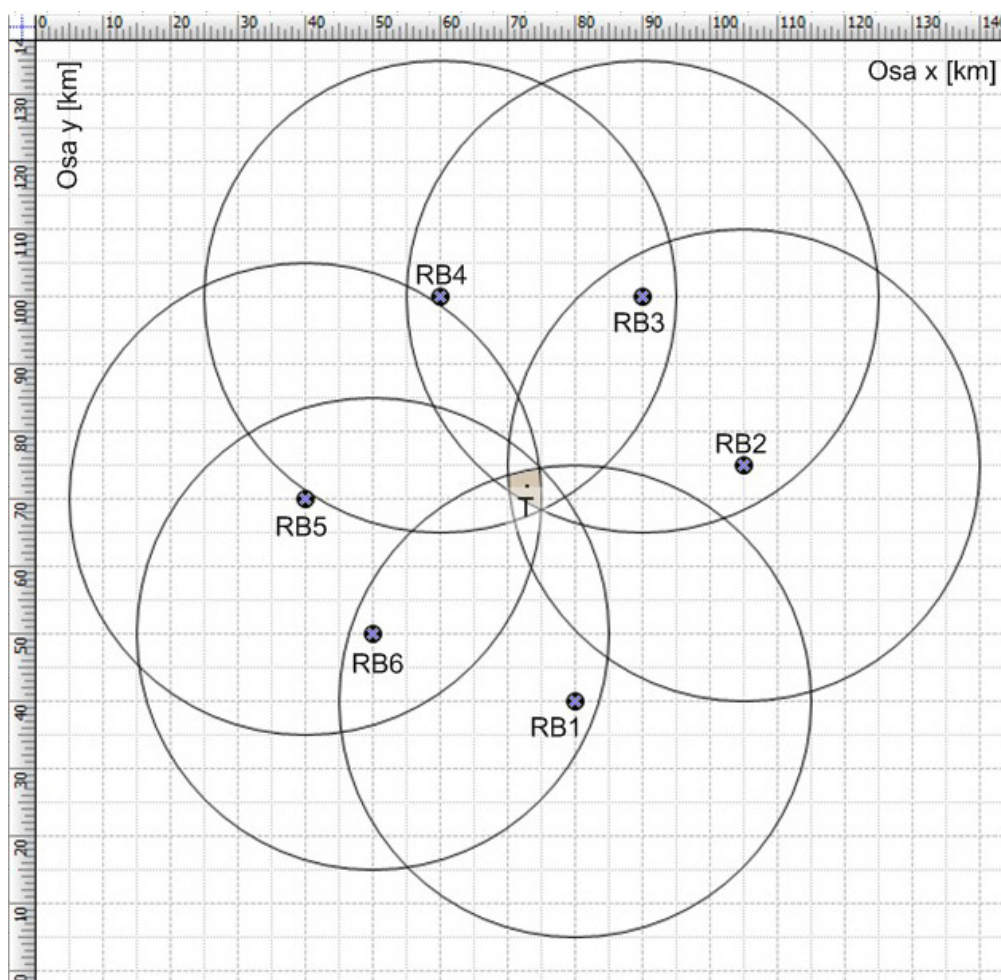
Daný výstup je pak zobrazen přehledně v tabulce na webových stránkách [49] spolu s vykreslením odhadnuté polohy na mapě. Pro průběžné informace o stavu výpočtu je zavedeno vypisování informací z programu do souboru *logFile.txt* na webovém serveru, díky němuž lze zjistit počet správných připojení k referenčním bodům, informace o hranicích vzdáleností a jiné informace.

7.5 Ověření správné činnosti algoritmu na umělých datech

Pro demonstraci ověření správné činnosti naprogramovaného algoritmu CBG byla zvolena umělá data dle obr. 7.7, kde je v kartézských souřadnicích zakreslena poloha 6 referenčních bodů RB1 až RB6 spolu s jejich hranicemi vzdáleností. Každý RB s příslušnou hranicí vzdáleností tvoří kruh. Průnik všech kruhů je znázorněn šedou oblastí a odpovídá hledané oblasti, kde se nachází poloha cílové stanice, jež je označena černým bodem a písmenem T.

¹⁰ Více o lehce čitelném formátu pro výměnu dat JSON lze nalézt na webových stránkách <http://www.json.org/json-cz.html>.

Souřadnice RB a příslušné velikosti hranic vzdáleností jsou zachyceny v tab. 7.1. Tyto hodnoty byly zadány do klientské části programu *CBGfinder*, která určila odhadovanou pozici cíle T [72,88; 71,87], čímž byl ověřen správný výpočet pozice cílové stanice. Velikost cílové oblasti byla spočtena na hodnotu 20,75 km², což také odpovídá nákresu. Přesnost odhadu je vypočítána z velikosti oblasti, kdy 100% přesnost odpovídá oblasti o rozloze 1 km² a 0% je určena pro nekonečně velkou oblast. V daném případě vychází přesnost 4,82 %.

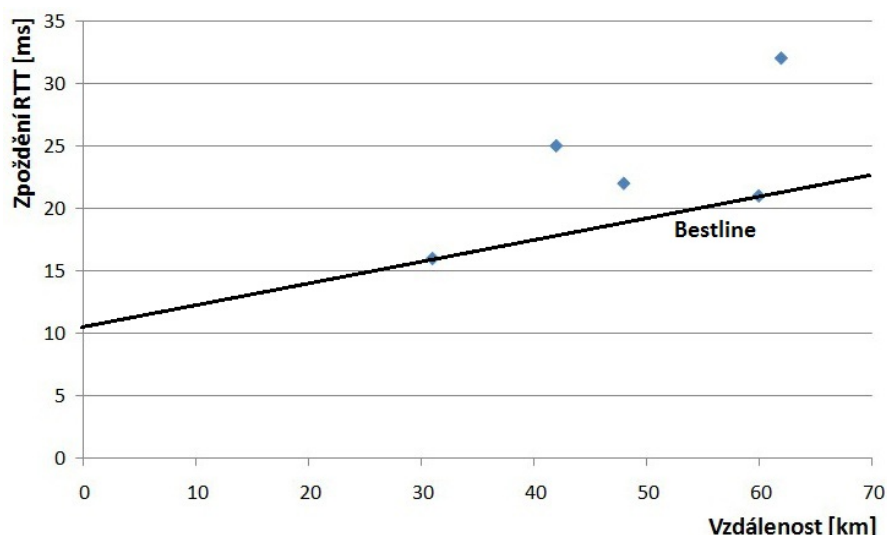


Obr. 7.7: Ověření správného výpočtu pozice cílové stanice.

Tab. 7.1: Souřadnice referenčních bodů a velikosti příslušných hranic vzdáleností.

Referenční body	Souřadnice x [km]	Souřadnice y [km]	Hranice vzdálenosti [km]
RB1	80	40	35
RB2	105	75	35
RB3	90	100	35
RB4	60	100	35
RB5	40	70	35
RB6	50	50	35

Ověření správného výpočtu automatické kalibrace proběhlo zadáním umělých dat do programu *computeBestline.jar*. Tato data jsou vynesena do grafu na obr. 7.8. Hodnoty vzdáleností odpovídají vzdálenostem od referenčního bodu RB1 k ostatním RB dle obr. 7.7. Přímka *Baseline* není do grafu zanesena, neboť by odpovídala ose x , na které je vynesena vzdálenost. Programem vypočtená přímka *Bestline* ve formátu $y = 0,17x + 10,6$ odpovídá *Bestline* z grafu na obr. 7.8 a je tvořena body o souřadnicích [31; 16] a [60; 21]. Tímto je tedy úspěšně ověřena funkce algoritmu CBG.



Obr. 7.8: Graf závislosti vzdálenosti na zpoždění.

Tab. 7.2: Hodnoty vzdáleností od RB1 a příslušné zpoždění RTT.

	RB2	RB3	RB4	RB5	RB6
Vzdálenost od RB1 [km]	42	60	62	48	31
Zpoždění RTT [ms]	25	21	32	22	16

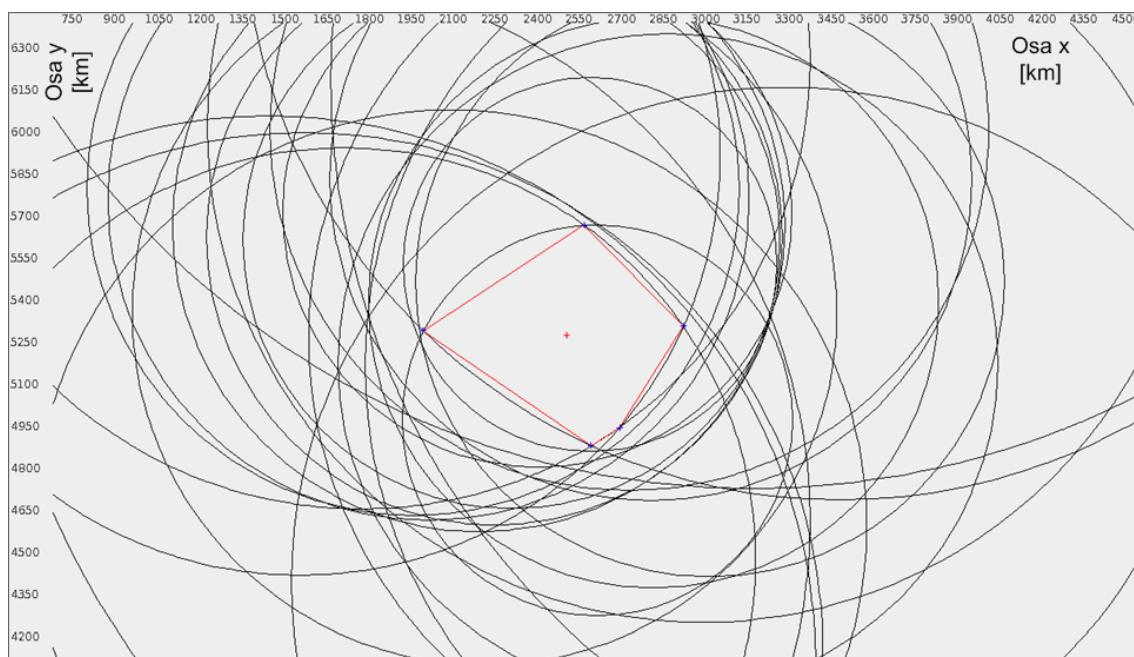
7.6 Příklad geolokace cíle v Internetu

Pro zobrazení reálného odhadu zeměpisné polohy cíle byla použita aplikace *CBGfinder*, jejíž klientská část je upravena pro použití na počítači bez webového serveru. Aplikace načítá vstupní data s informacemi o RB a cíli ze souborů a vykresluje výsledné hodnoty do nového okna, což při jejím použití na webovém serveru nelze. Pro možnost vlastního vyzkoušení je aplikace přiložena na CD spolu s jejím popisem nastavení.

Odhad zeměpisné polohy cíle byl proveden pro uzel s doménovým jménem *switch.ch* s využitím všech 40 referenčních bodů. Na obr. 7.9 lze vidět grafický výstup z aplikace. Výsledný polygon je umístěn uprostřed obrázku a v jeho těžišti je bod, jenž odpovídá odhadu geografické pozice cíle.

Zobrazená kartézská soustava koresponduje s kartografickou soustavou v určitém poměru. Díky převodu GPS souřadnic, jejichž jednotka je stupeň, na kartézské souřadnice s jednotkou kilometr dochází k drobnému zkreslení, neboť se pozice určuje pouze v Evropě, což v porovnání s celou zeměkoulí je relativně malá oblast. Svislá osa je zobrazena v převodním poměru $1^\circ = 111,330 \text{ km}$ a odpovídá zeměpisné šířce. Vodorovná osa odpovídá zeměpisné délce v poměru $1^\circ = 64,152 \text{ km}$ a je posunuta o 2000 km, aby byla zobrazovaná oblast více v prvním kvadrantu soustavy, neboť se vykresluje pouze kladná část grafu. Dané převodní poměry platí pro střed Evropy, jenž byl vybrán podle [46] a je určen místem v Litvě, nedaleko jejího hlavního města Vilnius s GPS souřadnicemi o šířce $54,9^\circ$ a délce $25,32^\circ$.

Výsledný polygon má obsah 376389 km^2 a jeho těžiště má GPS souřadnice o šířce $47,52^\circ$ a délce $8,64^\circ$, které v grafickém zobrazení na obr. 7.9 odpovídají bodu [2554; 5290]. Právě GPS souřadnice cíle jsou: šířka $47,38^\circ$ a délka $8,545^\circ$. Vzdálenost správné a odhadnuté polohy byla vypočítána dle [47] a určuje chybu odhadu zeměpisné polohy, jež je 17,33 km. Všechna tato data lze najít v souboru *logFile_zamereni_Switch_ch.txt* na příloženém CD.



Obr. 7.9: Zaměření pozice cíle *switch.ch*.

8. NASAZENÍ ALGORITMU V SÍTI PLANETLAB

V této kapitole je popsána problematika nasazení algoritmu v síti PlanetLab. Nejprve je stručně zmíněn program na správu RB a následně jsou uvedeny problémy, kterými se autor diplomové práce zabýval při implementaci programu *CBGfinder* v prostředí experimentální sítě.

8.1 Správa referenčních bodů v síti PlanetLab

Po vyzkoušení více programů určených ke správě referenčních bodů v síti PlanetLab¹¹ byla vybrána aplikace *PLMan*, která je obsáhle popsána v bakalářské práci [32]. Práce vysvětluje instalaci aplikace, její nastavení a možnosti. Také se věnuje ostatním programům, jež je možno využít při práci s RB.

PLMan se stará o paralelní SSH připojení na referenční body, kde poté lze současně na všech RB vykonat příkazy přes příkazový řádek. *PLMan* také umožňuje nahrání a stáhnutí souborů, což bylo využito pro distribuci aplikace *CBGfinder*. V neposlední řadě lze do *slice* (viz kapitolu 5.1) jednotlivé referenční body přidávat či odebírat.

8.2 Problémy při měření a jejich ošetření v programu

8.2.1 Instalace Java Runtime Environment

Na uzlech v síti PlanetLab běží operační systém CentOS, který implicitně neobsahuje *Java Runtime Environment* (JRE), což je prostředí pro běh programů napsaných v programovacím jazyku Java. Proto za pomoci příkazu `sudo yum -y java install` došlo k jeho instalaci.

8.2.2 Chybějící oprávnění správce

Na některých RB chybí pro použití *slice cesnet_vutbr2* oprávnění správce, a proto nelze vykonávat příkazy s právy správce. Tyto příkazy jsou uvozeny příkazem `sudo`, jenž je například použit při instalaci JRE. Takovéto uzly byly vynechány a nahrazeny jinými RB, kde výše zmíněný *slice* obsahuje práva správce.

8.2.3 Doba SSH připojení

Jeden z možných způsobů komunikace s RB je šifrované SSH spojení, jež bylo prvně zvoleno pro získávání informací o hranicích vzdáleností. Bohužel se ukázalo, že se doba jednoho zjištění hranice vzdálenosti pohybuje kolem 40 sekund. Proto se pro zrychlení celkové

¹¹ Informace o programech pro správu referenčních bodů v síti PlanetLab lze najít na webových stránkách <http://www.planet-lab.org/tools>.

doby komunikace se sítí PlanetLab zvolil systém typu klient-server, kdy na RB běží nepřetržitě program na pozadí (démon), jenž naslouchá na určitém portu a při dotazu klienta odpovídá příslušnou hranicí vzdáleností. Tímto bylo dosaženo urychlení výměny informací do 10 sekund.

8.2.4 Spuštění démona po restartu uzlů referenčních bodů

Démon aplikace *CBGfinder* musí být na uzlu RB stále spuštěn, aby mohl odpovídat na dotazy klienta. Jenže správci uzlů často provádějí údržbu RB, čímž dochází k restartování stanic či jejich vypnutí na delší čas. Při následném spuštění počítače je proto nutné inicializovat start démona.

Toho se dosáhne editací skriptu na každém RB, jenž je umístěn v souboru `/etc/rc.d/rc.local`, do něhož se запиše příkaz pro spuštění démona: `nohup java -jar /home/cesnet_vutbr2/xharth00/tcpConnectionServer.jar &`. Tento skript se pak automaticky vykoná při spouštění stanice [32].

8.2.5 Zajištění automatické kalibrace na referenčních bodech

Aby byla přímka *Bestline* na každém RB stále aktuální, spouští se program *computeBestline.jar* každou půlhodinu díky jeho registraci ve službě *cron*¹² za pomoci následujícího příkazu: `echo "*/30 * * * * java -jar /home/cesnet_vutbr2/xharth00/computeBestline.jar" | crontab -; sudo /sbin/service crond restart` [32].

8.2.6 Řešení problému se špatně zadanými polohami referenčních bodů

V průběhu testování aplikace se vyskytl problém při automatické kalibraci v podobě nesmyslně malých hodnot naměřeného RTT zpoždění mezi referenčními body vzhledem k reálné vzdálenosti, což vedlo ke zkreslení přímky *Bestline* a tím i ke špatnému určení hranice vzdálenosti k cíli. Důvodem zpoždění, jež odpovídá rychlosti větší než je rychlost světla v optickém vlákne, může být špatné určení zeměpisné polohy referenčního bodu, kdy jsou RB umístěny ve skutečnosti blíže, než se předpokládá.

Proto byla při výpočtu *Bestline* zavedena podmínka, že když je RTT zpoždění menší než odpovídající zpoždění přenosu světla optickým vláknem, tak není tato hodnota dále uvažována. Pokud je za y dosazena hodnota RTT zpoždění v ms a x označuje vzdálenost v km, pak podmínka odpovídá rovnici $y \geq 0,01x$.

¹² Více informací o službě *cron* lze nalézt na webových stránkách <http://www.cyberciti.biz/faq/how-do-i-add-jobs-to-cron-under-linux-or-unix-oses/>.

9. VÝSLEDKY MĚŘENÍ A POROVNÁNÍ METODY S OSTATNÍMI GEOLOKAČNÍMI ALGORITMY

Společné měření aktivních metod CBG (Bc. Petr Harth), SOI (Jakub Chmelař), Octant (Bc. Filip Dvořák), Geoweight (Bc. Peter Tropp) a pasivní metody GeoIP (Tomáš Sousedík) probíhalo v časovém rozmezí 13. až 20. 4. 2012. Pro aktivní metody byly použity výše zmíněné stejné referenční body. Cíle, u nichž víme zeměpisnou polohu a můžeme tedy určit chybu měření, jsou pak společné pro aktivní i pasivní geolokační techniky.

Pro měření bylo využito webových stránek Ing. Lukáše Verneru [49], kde je umístěn formulář pro výběr RB, cílů a geolokačních technik, pomocí nichž má měření proběhnout. Výsledky se pak ukládají do tamější databáze. Při následném zpracování výsledků a zobrazení grafů bylo využito tabulkového editoru Microsoft Office Excel 2007.

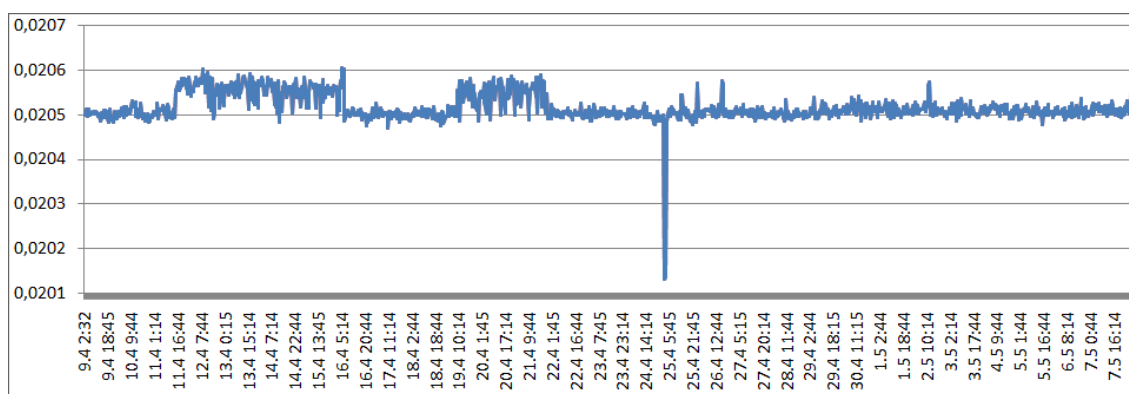
Jelikož měření probíhalo pro každý cíl vícekrát a každá metoda provedla různý počet měření, jsou výsledné chyby hodnot odhadu zeměpisné pozice zprůměrované, aby pro každý cíl nebyl uveden různý počet výsledků, což potom vede ke zkreslení grafů.

9.1 Výsledky měření metody CBG

9.1.1 Vývoj přímky *Bestline* v čase

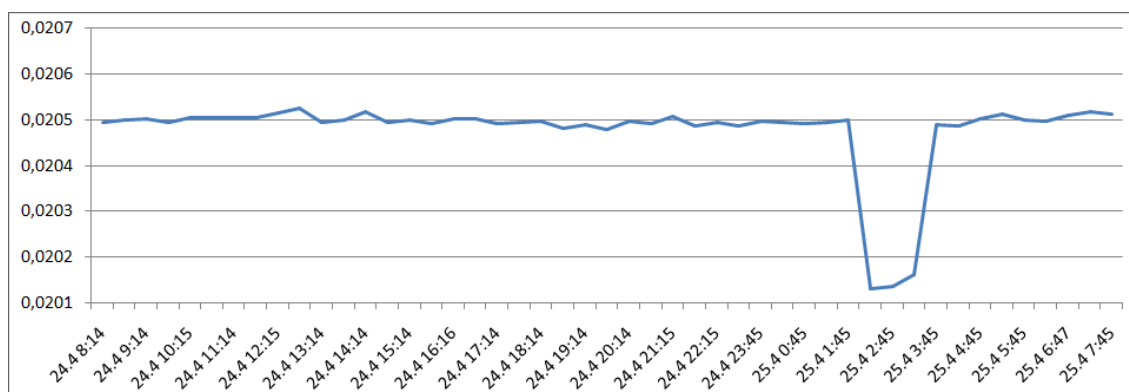
Po dobu jednoho měsíce (9. 4. až 8. 5. 2012) byly sledovány výsledky automatické kalibrace na referenčních bodech. V průběhu daného období byla zjištěna pouze nepatrná změna směrnice k a posunutí q . Například na uzlu *planetlab1.cesnet.cz* se směrnice k pohybovala v rozmezí hodnot 0,020131168 (dne 25. 4. v čase 2.15, rychlost 94,6767 km/ms) až 0,020604050 (dne 16. 4. v čase 5.14, rychlost 94,2533 km/ms), což činí rozdíl v rychlostech paketu 0,4234 km/ms. Hodnota posunutí q byla v rozmezí 0,0230 (dne 15. 4. v čase 14.45, rychlost 95,3465 km/ms) až 0,0670 (dne 26. 4. v čase 10.45, rychlost 90,9910 km/ms), což činí rozdíl v rychlostech paketu 4,3555 km/ms.

Na obr. 9.1 je zachycen graf změny hodnoty směrnice k v uzlu *planetlab1.cesnet.cz* za celý měsíc, z něhož vyplývá, že směrnice kolísá kolem určité hodnoty. Je zajímavé, že ve dnech 11. až 15. 4. a 19. až 22. 4. došlo ke zvýšení hodnoty, zatímco v jiné termíny je směrnice na nižší úrovni. Tento stav může být zapříčiněn vytížením RB jinými projekty, neboť prostředky stanic jsou sdílené, a proto se volná šířka přenosového pásma může měnit v závislosti na tom, jak moc využívají jednotlivé aplikace internetovou komunikaci.



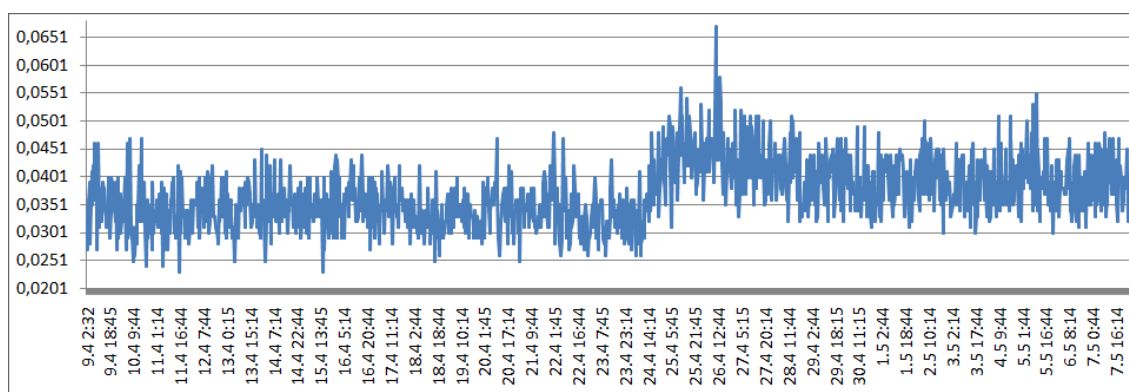
Obr. 9.1: Změna hodnoty směrnice k v uzlu *planetlab1.cesnet.cz*.

Také si lze povšimnout krátkého strmého poklesu dne 25. 4., kdy byl nejspíše internetový provoz na nižší úrovni. Pro názornost je uveden na obr. 9.2 graf průběhu směrnice k za časové rozmezí 24 hodin, kde lze vidět pokles v čase 25. 4. 1.45–3.45. Tento graf také dokazuje, že se směrnice v průběhu denní doby výrazně nemění, i když by bylo očekávatelné, že v noci, tedy v době pracovního klidu, bude na síti menší provoz.



Obr. 9.2: Změna hodnoty směrnice k v uzlu *planetlab1.cesnet.cz* v průběhu 24 hodin.

Na obr. 9.3 je graf změny posunutí q v uzlu *planetlab1.cesnet.cz* v průběhu celého měsíce. Hodnota posunutí závisí na hodnotě RTT zpoždění pro nulovou vzdálenost, neboť je při výpočtu *Bestline* zahrnuto i dotazování se programem *Ping* na vlastní IP adresu z důvodu zjištění, jak se mění vytížení systémových prostředků uzlu. Ode dne 25. 4. došlo k nárůstu hodnoty posunutí, a proto lze očekávat, že se spustila aplikace, jež je náročná na systémové zdroje. Graf změny posunutí q v průběhu určitého dne není uveden, neboť hodnoty jsou víceméně konstantní a nevykazují zajímavé odchylky, jako tomu bylo u grafů směrnice k .



Obr. 9.3: Změna hodnoty posunutí q v uzlu *planetlab1.cesnet.cz*.

Díky výše uvedeným grafům lze konstatovat, že zavedením automatické kalibrace sice dochází k určitému sledování stavu sítě z pohledu problematiky převodu zpoždění na hranici vzdálenosti, nicméně tyto změny nejsou zásadního charakteru.

9.1.2 Chyba odhadu zeměpisné polohy a rychlost výpočtu

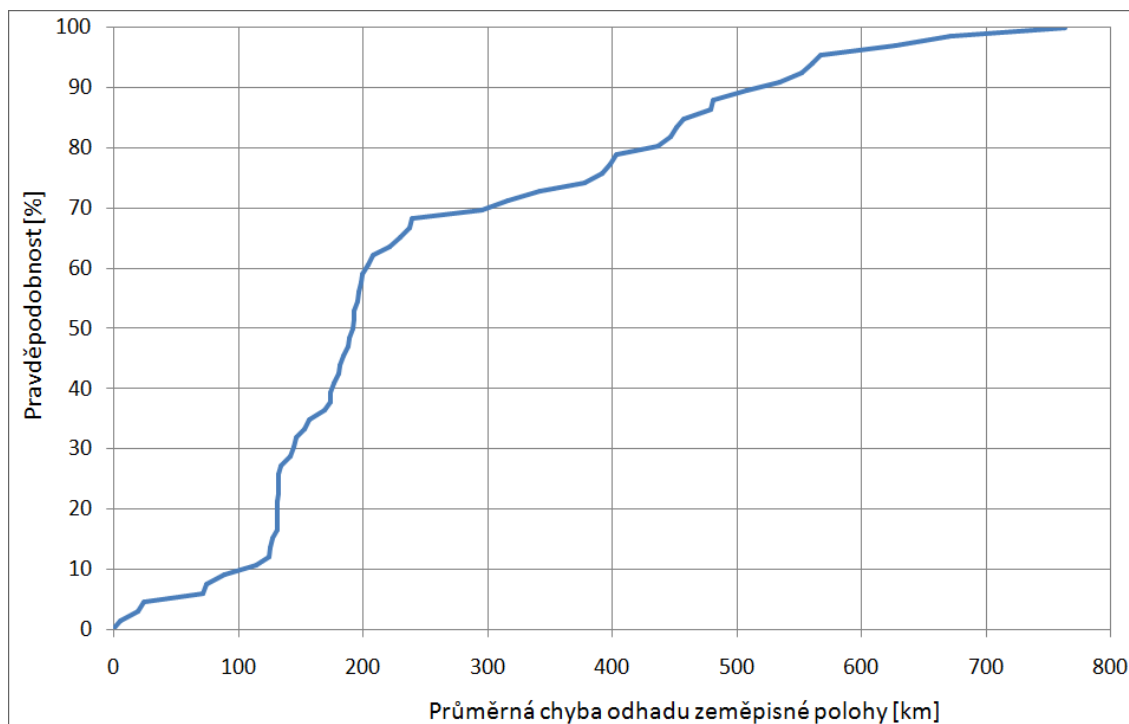
Výsledky měření geolokace pomocí metody CBG jsou uvedeny v příloze, neboť by zde vícestránková tabulka narušovala celistvost textu. Z těchto měření vyplývá, že průměrná chyba odhadu zeměpisné polohy je 284,69 km a medián chyby je 192,87 km. Průměrná chyba odhadu je velmi ovlivněna chybou při měření cíle s DNS jménem *koi8.pp.ru*, jenž nabývá hodnoty 2108,03 km. Ostatní měření jsou s chybou v rozmezí od 5,66 km do 763,37 km.

Rychlost výpočtu odhadu zeměpisné polohy velmi záleží na implementaci metody a paralelizaci zjišťování informací od referenčních bodů. Pro aplikaci *CBGfinder* je průměrný čas odhadu pozice vyčíslen hodnotou 78,48 s a medián je 59 s.

9.1.3 Kumulativní distribuční funkce

Kumulativní distribuční funkce (zkratka CDF z anglického názvu *Cumulative Distribution Function*) [48] udává pravděpodobnost, že náhodná veličina X nabude hodnoty menší, než je zvolená hodnota x , nebo je stejně velká. CDF je tedy definována rovnicí: $F(x) = P(X \leq x)$. Více o CDF lze nalézt v [48].

Na obr. 9.4 je tato CDF zachycena pro změřené výsledky algoritmu CBG, jenž je realizován aplikací *CBGfinder*. Data pro zobrazení CDF jsou omezena maximální hodnotou 1000 km, neboť by chyba měření cíle s DNS jménem *koi8.pp.ru*, jenž nabývá hodnoty 2108,03 km, snížila názornost grafu. Ve společném srovnání jsou pak také hodnoty pro vykreslení CDF omezeny do 1000 km.



Obr. 9.4: Kumulativní distribuční funkce algoritmu CBG.

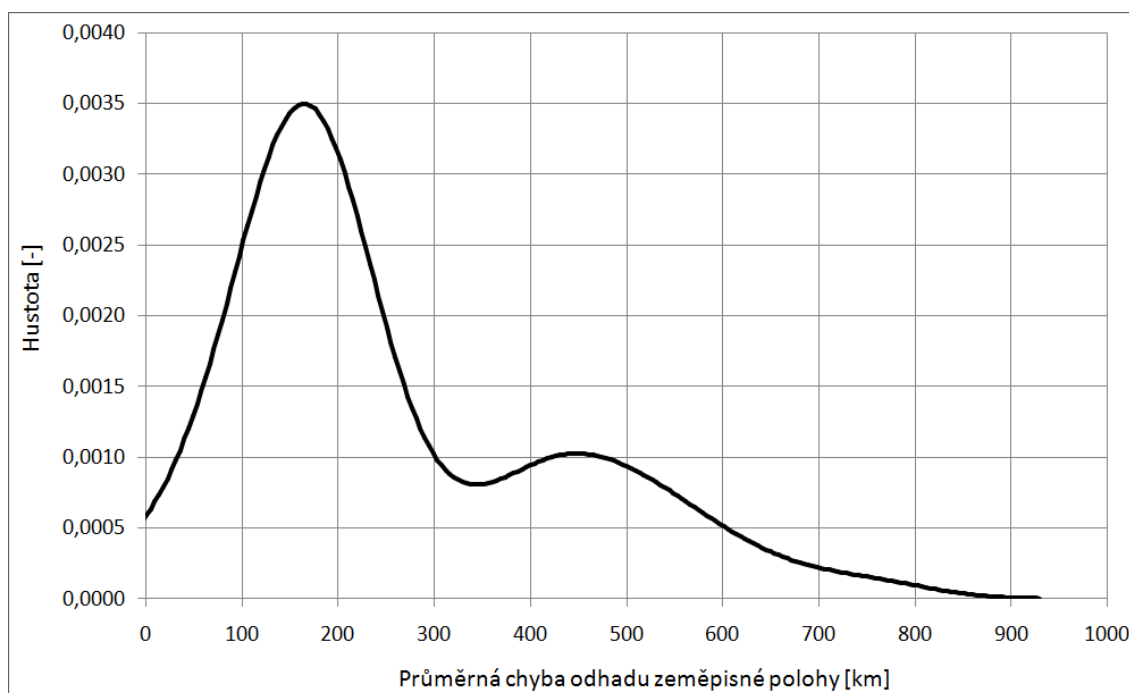
Z dané CDF vyplývá pro množinu 40 referenčních bodů a 76 cílů následující:

- 31,3 % cílům byla určena geografická pozice s chybou do 150 km,
- 68,7 % cílů bylo zaměřeno s chybou 300 km a méně,
- 86,6 % cílů bylo určeno s odchylkou menší než 500 km,
- pouze pro 11,9 % stanic nešlo určit polohu z důvodu podhodnocení hranic, čímž selhal výpočet výsledného polygonu, v jehož středu by ležel odhad zeměpisné pozice.

9.1.4 Jádrový odhad hustoty

Jádrový odhad hustoty (zkratka KDE z anglického názvu *Kernel Density Estimation*) [50] je neparametrický způsob, jak odhadnout hustotu pravděpodobnosti náhodné veličiny. Základním problémem u této metody je volba šířky okna, jež udává parametry výsledných hodnot. Pokud je šířka okna malá, tak se výsledek rovná průběhu pozvolně se měnící funkce. Avšak při velké šířce okna vznikají v místech naměřených hodnot špičky. Více k volbě okna viz [50].

Pro výpočet KDE jsem zvolil makro¹³ pro tabulkové editory, jež se také postará o optimálně zvolenou šířku okna. Na obr. 9.5 je pak zachycen KDE pro metodu CBG, z něhož vyplývá, že hodnoty kolem 170 km se objevují nejčastěji.



Obr. 9.5: Jádrový odhad hustoty metody CBG.

9.2 Srovnání výsledků měření s ostatními metodami

Pro srovnání výsledků měření s ostatními metodami byla vybrána diskuse nad průměrnou chybou odhadu zeměpisné pozice a jejím mediánem. Také je proveden rozbor rychlostí jejího výpočtu. Následně se porovnávají kumulativní distribuční funkce a jádrové odhady hustoty.

9.2.1 Chyby odhadu zeměpisné polohy a rychlosti výpočtu

V tab. 9.1 je uvedeno pořadí geolokačních technik podle mediánu chyby odhadu zeměpisné polohy, jelikož je medián lepší údaj pro srovnání než průměrná chyba. Nejlépe je na tom pasivní metoda GeoIP. Mezi aktivními metodami vede SOI. Třetí místo zaujímá algoritmus GeoWeight, následován metodou CBG. Jako poslední se umístil Octant.

¹³ Makro pro tabulkové editory, které vypočítá jádrový odhad hustoty je dostupné na <http://www.rsc.org/Membership/Networking/InterestGroups/Analytical/AMC/Software/kerneldensities.asp>.

Tab. 9.1: Pořadí geolokačních technik podle mediánu chyby odhadu zeměpisné polohy.

Pořadí podle mediánu	Metoda	Průměrná chyba [km]	Medián [km]
1.	GeoIP	120,39	13,22
2.	SOI	183,77	172,99
3.	GeoWeight	239,85	184,76
4.	CBG	284,69	192,87
5.	Octant	259,70	237,22

Z pohledu průměrné chyby i mediánu je nejlepší z metod pasivní algoritmus GeoIP, jelikož je v databázi správně uvedena většina zeměpisných pozic cílů. Nicméně pokud by informace o cílech v databázi chyběly, tak by pozice vůbec nešla určit.

Zajímavé je, že je z aktivních metod první právě SOI, neboť tento algoritmus nepoužívá automatickou kalibraci, jež by měla odhad polohy upřesňovat. Tento fakt potvrzuje správně určenou převodní konstantu metody SOI, čímž odpadá nutnost složité automatické kalibrace. Také to odpovídá výše uvedenému závěru ze sledování změn přímky *Bestline* v průběhu celého měsíce a to, že se parametry přímky *Bestline* výrazně nemění v závislosti na čase, a tudíž nedochází k výrazné změně přesnosti metody při zavedení automatické kalibrace.

Další důležitý poznatek je spatřen v informaci, že jako poslední algoritmus skončil Octant. Rozdíl v mediánu mezi metodami SOI a Octant je 64,23 km. Tento fakt vyvrací tvrzení, že komplexní obálka při automatické kalibraci (viz kapitolu 4.6) odhad zeměpisné polohy upřesňuje. Naopak zavedení vah pomocí metody GeoWeight (viz kapitolu 4.5) se ukázalo prospěšné, neboť GeoWeight je přesnější než CBG.

Z pohledu pořadí geolokačních technik podle mediánu doby výpočtu odhadu zeměpisné polohy dle tab. 9.2 vyplývá, že nejrychlejší je pasivní metoda GeoIP. Její okamžitý výpočet je zapříčiněn tím, že databáze s informacemi o poloze je uložena na lokálním počítači, takže dotaz na pozici probíhá jen v rámci webového serveru. Mezi aktivními metodami vede SOI a za ní následují Octant, CBG a poslední GeoWeight.

Tab. 9.2: Pořadí geolokačních technik podle mediánu doby výpočtu odhadu zeměpisné polohy.

Pořadí podle mediánu	Metoda	Průměrná doba [s]	Medián [s]
1.	GeoIP	0,00	0,00
2.	SOI	21,61	21,49
3.	Octant	58,76	28,00
4.	CBG	78,48	59,00
5.	GeoWeight	349,48	343,39

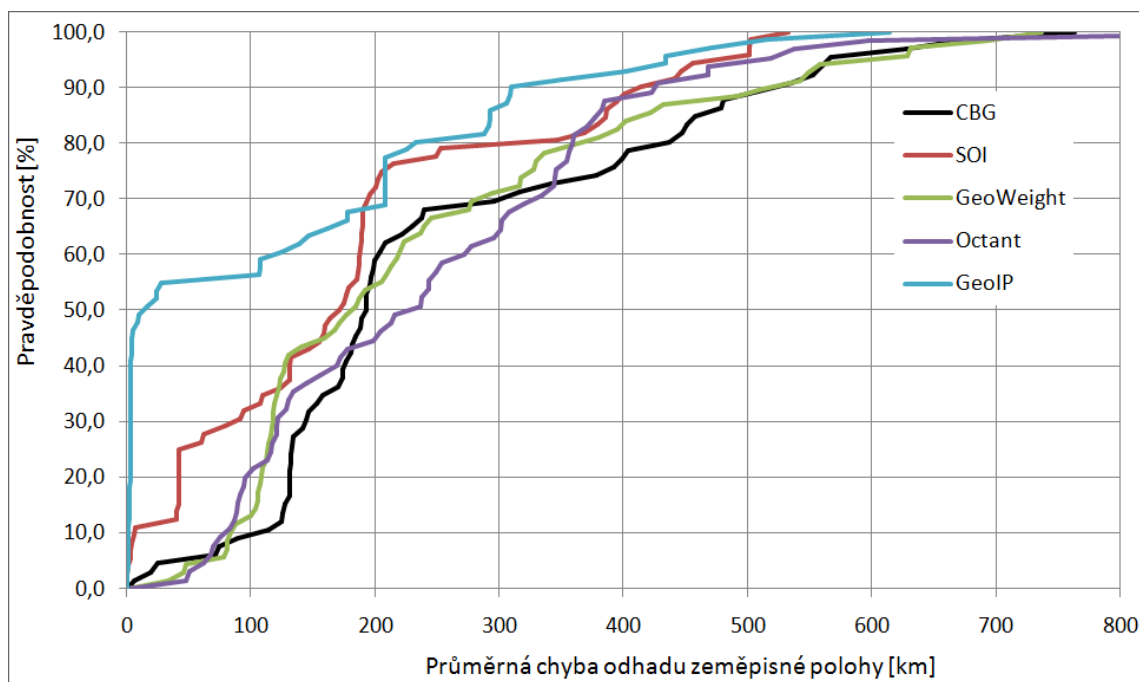
9.2.2 Kumulativní distribuční funkce

Nejnázornější srovnání geolokačních algoritmů nabízí graf kumulativní distribuční funkce na obr. 9.6. Podle tvaru křivek lze určit, jaká je pravděpodobnost, že chyba odhadu zeměpisné polohy bude menší nebo rovna určitému počtu km.

Z grafu vyplývá, že při chybě odhadu do 30 km je pořadí: 1. GeoIP, 2. SOI, 3. CBG a o 4. pozici se dělí GeoWeight s metodou Octant. Další zajímavá hodnota se nachází v 70 km, kde jsou pravděpodobnosti metod CBG, GeoWeight a Octant skoro stejné, ale zároveň SOI dává výrazně lepší výsledky. Pasivní algoritmus GeoIP však s přehledem vede nad aktivními metodami.

Při chybě do 100 km je zajímavá změna pořadí na 3. místě, jež zaujímá Octant následován metodou GeoWeight, a poslední skončila CBG. Z pohledu geolokační techniky GeoWeight je zajímavá hodnota 140 km, kde tato technika zaujímá 2. místo.

Při pohledu na chybu odhadu do 200 km lze konstatovat, že SOI dokonce poráží GeoIP a je na 1. místě. Třetí skončila metoda CBG, čtvrtá GeoWeight a poslední Octant. Další vývoj je rozdílný od předchozího stavu, neboť algoritmus Octant od 360 km poráží metody GeoWeight a CBG. V chybě odhadu do 380 km je Octant dokonce nepatrně lepší než SOI a je na druhém místě za pasivní metodou GeoIP.



Obr. 9.6: Graf kumulativní distribuční funkce všech metod.

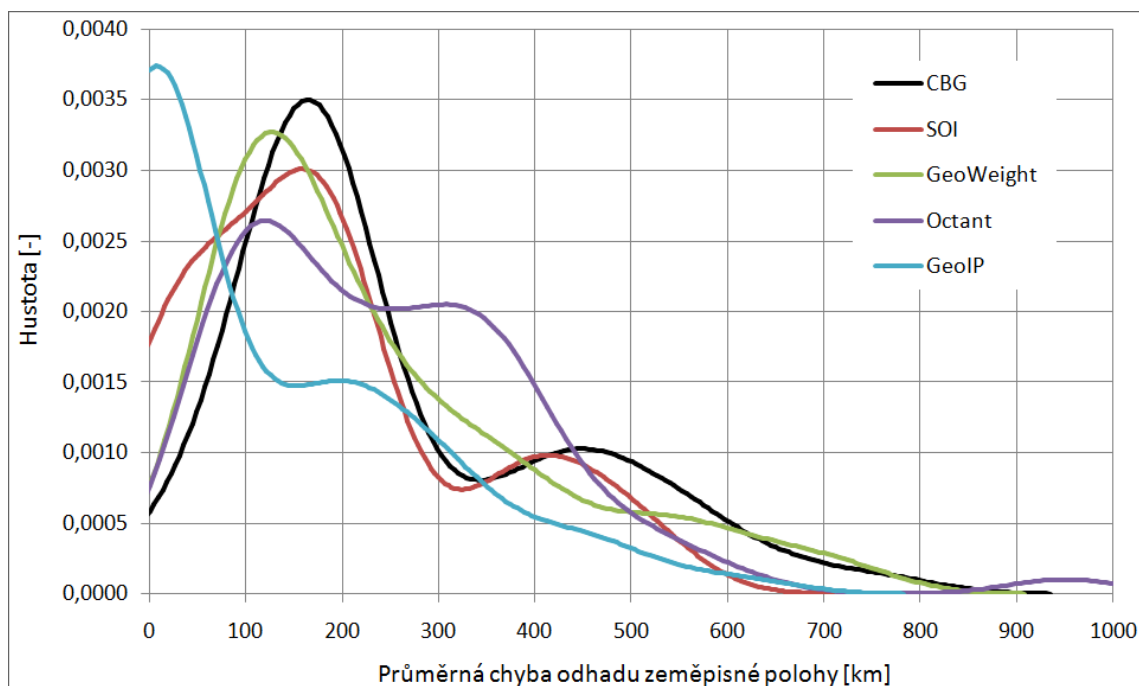
9.2.3 Jádrový odhad hustoty

Z grafu jádrového odhadu hustoty pro všechny metody, jenž je zobrazen na obr. 9.7, lze odečíst, kolem kterých hodnot se nejčastěji vyskytují průměrné chyby odhadu zeměpisné polohy.

Pro pasivní metodu GeoIP se hodnoty objevují nejvíce kolem 10 km. Strmost křivky reflektuje fakt, že se v databázi vyskytují z valné většiny přesné a pravdivé informace o zeměpisných polohách cílů.

Metoda Octant disponuje grafem jádrového odhadu hustoty, jenž je nejvíce zploštělý. To znamená, že se hodnoty nevyskytují ve velké hustotě na jednom místě, ale jsou více rozprostřeny. Přesto jeho největší hustotu lze nalézt kolem hodnoty 120 km.

GeoWeight má největší hustotu kolem bodu 130 km, SOI ve 160 km a geolokační algoritmus CBG má hodnoty chyb odhadů zeměpisného polohy nejméně rozprostřeny kolem 170 km.



Obr. 9.7: Graf jádrového odhadu hustoty pro všechny metody.

10. ZÁVĚR

Tato diplomová práce se zabývala vysvětlením dílčích faktorů způsobujících zpoždění při přenosu dat mezi stanicemi v IP sítích. Dále byla popsána experimentální síť PlanetLab. Také byly popsány IP geolokační techniky a naprogramován geolokační algoritmus CBG (*Constraint-based Geolocation*). Pomocí uměle vytvořených vstupních dat byla ověřena správná činnost realizovaného algoritmu, program byl nasazen v experimentální síti PlanetLab a byla provedena lokalizace stanic na území Evropy.

Dle vypracování diplomové práce lze říci, že byly úspěšně splněny všechny body zadání. V neposlední řadě je důležité zmínit, že je tato práce součástí výzkumu, jenž se zabývá praktickými realizacemi IP geolokačních technik na bázi měření zpoždění a jejich následném porovnávání z pohledu přesnosti a dalších zvolených měřítek.

Autor práce se podílel na vytvoření seznamu cílů ležících v Evropě, jež obsahují DNS LOC záznam. Tyto uzly slouží jako cílové stanice při měření přesnosti určování polohy pomocí IP geolokačních technik. Při hledání cílů došlo k mezinárodní spolupráci, neboť tvůrce skriptu vyhledávajícího domény s DNS LOC záznamy Jeff Hungerford jej uvolnil pro jeho modifikaci a použití. Bylo zjištěno, že se LOC záznamy často nepoužívají, neboť ze 4,5 milionu adres bylo takových nalezeno pouze 181. Následně se tyto domény testovaly, zda leží v Evropě a jestli reagují na ECHO dotaz programu Ping. Tyto podmínky splňuje pouze 90 domén, z nichž pouze 76 cílů obsahuje geografické polohy, které neleží například v moři nebo v neobydlené krajině.

Praktická realizace algoritmu CBG proběhla v podobě jeho implementace v aplikaci s názvem *CBGfinder*, jež byla náležitě popsána i vývojovými diagramy. Následně proběhlo ověření správné činnosti programu na uměle zadaných vstupních datech i na příkladu geolokace cíle v Internetu. Tento cíl byl geograficky zaměřen pomocí 40 referenčních bodů ze sítě PlanetLab.

Dále jsou uvedeny výsledky měření algoritmu CBG. Nejprve je proveden rozbor vývoje parametrů přímky *Bestline* na určitém uzlu sítě PlanetLab v průběhu jednoho měsíce. Za dané období byla zjištěna pouze nepatrná změna směrnice k a posunutí q . Dále je popsána chyba odhadu zeměpisné polohy a rychlost výpočtu pro 76 cílů za pomoci 40 referenčních bodů sítě PlanetLab. Poté je uvedena kumulativní distribuční funkce a jádrový odhad hustoty.

Výsledky metody CBG jsou také porovnány s výsledky ostatních IP geolokačních technik, jež naprogramovali a změřili kolegové autora diplomové práce. Konkrétně se jednalo o metody SOI, GeoWeight, Octant a GeoIP. Měření probíhalo ve stejném časovém období i se stejnou množinou referenčních bodů a cílů. Proto lze výsledky efektivně porovnat za pomoci průměrné chyby odhadu geografické polohy a jejího mediánu, času výpočtu, kumulativní distribuční funkce a jádrového odhadu hustoty. Z daných porovnání byly vysloveny závěry, jež jsou podrobně vysvětleny.

LITERATURA

- [1] KOMOSNÝ, Dan; BALEJ, Jiří. Zdroje zpoždění při komunikaci v Internetu. *Elektrorevue* [online]. 2010, 2010, 42, [cit. 2011-11-16]. Dostupný z WWW: <<http://www.elektrorevue.cz/cz/clanky/komunikacni-technologie/35/zdroje-zpozdeni-pri-komunikaci-v-internetu/>>. ISSN 1213-1539.
- [2] BOVY, C. J., et al. *Analysis of End-to-end Delay Measurement in Internet* [online]. PAM 2002, 2002 [cit. 2011-11-17]. Dostupný z WWW: <http://www.pamconf.net/2002/Analysis_of_End_to_end_Delay_Measurements_in_Internet.pdf>.
- [3] Cisco Systems, Inc. *CCNP: Building scalable Internetworks v5.0.3.0* [online]. San Jose, USA : Cisco Press, 2006 [cit. 2011-11-17]. 6.1.8 BGP Routing Policies. Dostupné z WWW: <<http://www.cisco.com/web/learning/netacad/index.html>>.
- [4] KRUMNIKL, Michal. *Měření latence síťových prvků* [online]. 2005 [cit. 2011-11-17]. 15 s. Dostupný z WWW: <www.cs.vsb.cz/gragarek/SPS/>.
- [5] White paper. *Cisco Nexus 5020 Switch Performance in Market-Data and Back-Office Data Delivery Environments* [online]. 2009 [cit. 2011-11-17]. 9 s. Dostupný z WWW: <[origin-www.cisco.com](http://www.cisco.com)>.
- [6] *Internet Router Test* [online]. 2001 [cit. 2011-11-17]. Dostupný z WWW: <www.lightreading.com>.
- [7] 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.
- [8] Network Working Group. *Request for Comments: 1242 : Benchmarking Terminology for Network Interconnection Devices*. In BRADNER, S. RFC [online]. Harvard : Harvard University, July 1991 [cit. 2011-11-17]. Dostupné z WWW: <<http://www.ietf.org/rfc/rfc1242.txt>>.
- [9] StoneBeat Technical Documentation: What is asymmetric routing?. *Stonesoft* [online]. 2011 [cit. 2012-05-15]. Dostupné z WWW: <<https://my.stonesoft.com/support/document.do?docid=1377>>.
- [10] DOLEŽEL, Pavel. *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.
- [11] Network Working Group. *Request for Comments: 792 : INTERNET CONTROL MESSAGE PROTOCOL*. In POSTEL, J. RFC [online]. ISI, September 1981 [cit. 2011-11-17]. Dostupné z WWW: <<http://www.ietf.org/rfc/rfc792.txt>>.
- [12] Network Working Group. *Request for Comments: 1256 : ICMP Router Discovery Messages*. In DEERING, S. RFC [online]. Xerox PARC, September 1991 [cit. 2011-11-17]. Dostupné z WWW: <<http://www.ietf.org/rfc/rfc1256.txt>>.
- [13] PADMANABHAN, Venkata N.; SUBRAMANIAN, Lakshminarayanan. *An Investigation of Geographic Mapping Techniques for Internet Hosts*. In Internet Measurement Conference, 2001 [cit. 2011-11-18]. [s.l.] : ACM/SIGCOMM, 2001. s. 13. Dostupné z WWW: <<http://conferences.sigcomm.org/sigcomm/2001/p14-pabmanabhan.pdf>>. ISBN 1-58113-411-8.
- [14] PETERKA, Jiří. Telnet - I.. *EArchiv.cz* [online]. 1993, 67, [cit. 2011-11-18]. Dostupný z WWW: <<http://www.earchiv.cz/a93/a325c110.php3>>. [e-článek]
- [15] TOREN, Michael C. *Michael C. Toren's Webspace* [online]. 2008 [cit. 2011-11-18]. Tcptraceroute. Dostupné z WWW: <<http://michael.toren.net/code/tcptraceroute/>>.

- [16] MUUSS, Mike. *The Research Interests of MIKE MUUSS* [online]. 1997 [cit. 2011-11-18]. The Story of the PING Program. Dostupné z WWW: <<http://ftp.arl.mil/mike/ping.html>>.
- [17] Cisco Systems, Inc. *Cisco Systems, Inc* [online]. c2011 [cit. 2011-11-18]. Using the Layer 2 Traceroute Utility. Dostupné z WWW: <<http://www.cisco.com/en/US/docs/switches/lan/catalyst6500/ios/12.2SXF/native/configuration/guide/l2trace.html>>.
- [18] DENZLER, Andrea. *AndreaPlanet* [online]. 2004 [cit. 2011-11-18]. True Ping. Dostupné z WWW: <<http://www.andreaplanet.com/tping/>>.
- [19] KATZ-BASSETT, Ethan, et al. Towards IP Geolocation Using Delay and Topology Measurements. In *Internet Measurement Conference*, 2006 [cit. 2011-11-18]. [s.l.] : ACM/SIGCOMM, 2006. s. 13. Dostupné z WWW: <<http://conferences.sigcomm.org/imc/2006/papers/p7-bassett.pdf>>.
- [20] *IPinfoDB* [online]. 2010 [cit. 2011-11-19]. IP address geolocation database. Dostupné z WWW: <http://ipinfodb.com/ip_database.php>.
- [21] RFC 1834. *Whois and Network Information Lookup Service Whois++*. University of California: J. Gargano, K. Weiss, 1995. 7 s. Dostupné z WWW: <<http://www.faqs.org/rfcs/rfc1834.html>>.
- [22] MaxMind's IP Intelligence Solution. *MaxMind* [online]. 2012 [cit. 2012-05-14]. Dostupné z WWW: <<http://www.maxmind.com/app/ip-locate>>.
- [23] GUEYE, Bamba, et al. *Constraint-based Geolocation of Internet Hosts*. In *Internet Measurement Conference*. 2004. [s.l.] : ACM/SIGCOMM, 2004. s. 14. Dostupné z WWW: <<http://www.cs.bu.edu/fac/crovella/paper-archive/imc04-geolocation-full.pdf>>. E00-108534666.
- [24] ZIVIANI, Artur, et al. Demographic Placement for Internet Host Location. In *Global Telecommunications Conference*, 2003. [online]. San Francisco, USA : GLOBECOM, 2003-12, 2004-12-14 [cit. 2011-05-28]. Dostupné z WWW: <<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.2.7487>>. ISBN 0-7803-7974-8, 10.1109/GLOCOM.2003.1258945.
- [25] SVANTESSON, Dan. *Geo-location Technologies*. [s.l.], 2004. 10 s. . Faculty of Law, Bond University (Australia). Dostupné z WWW: <www.svantesson.org/svantesson20040906.doc>.
- [26] CHATZOPOULOU, Doxa; KOKKODIS, Marios. *IP Geolocation* [online]. [s.l.], . 6 s. . Computer Science and Engineering Department, UC Riverside. Dostupné z WWW: <<https://www.cs.ucr.edu/~mak/grad/geolocation.pdf>>.
- [27] ENGE, P.; MISRA, P. Special issue on GPS. *Proceedings of the IEEE*, 87(1): 3-15, Jan. 1999.
- [28] PERCACCI, R.; VESPIGNANI, A. Scale-free behavior of the Internet global performance. *The European Physical Journal B - Condensed Matter*, 32(4):411-414, Apr. 2003.
- [29] ARIF, M.; KARUNASEKERA, S.; KULKARNI, S. GeoWeight: Internet Host Geolocation Based on a Probability Model for Latency Measurements. In *Proc. 33rd Australasian Computer Science Conference (ACSC 2010)*. Brisbane, Australia : University of Melbourne, 2010.
- [30] WONG, B.; STOYANOV, I.; SIRER, E. *Octant: A Comprehensive Framework for the Geolocation of Internet Hosts*. Ithaca : Dept. of Computer Science, 2007. 14 s.
- [31] *PlanetLab / An open platform for developing, deploying, and accessing planetary-scale services* [online]. c2007 [cit. 2011-11-26]. About PlanetLab. Dostupné z WWW: <<http://planet-lab.org/>>.
- [32] CHUPÁČ, M. *Správa stanic Planetlabu*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2011. 100 s. Vedoucí bakalářské práce Ing. Jakub Müller.
- [33] KOPEČEK, T. *Nalezení fyzické pozice stanice v síti Internet*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2010. 57 s. Vedoucí diplomové práce doc. Ing. Dan Komosný, Ph.D.

- [34] MAŠÍN, J. *Lokalizace stanic v Internetu pomocí metody Vivaldi s adaptivním časovým krokem*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2011. 57 s. Vedoucí diplomové práce doc. Ing. Dan Komosný, Ph.D.
- [35] Navrátil, J. *PlanetLab - model budoucího Internetu*. Zpravodaj ÚVT MU. ISSN 1212-0901, 2006, roč. XVI, č. 5, s. 1-5. Dostupný z WWW: <http://www.ics.muni.cz/zpravodaj/articles/525.html>.
- [36] CIMBÁLEK, P. *Grafické zobrazení relací mezi počítači v Internetu*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2008. 74 s. Vedoucí diplomové práce doc. Ing. Dan Komosný, Ph.D.
- [37] RELOVSKÝ, J. *Zjištění fyzické pozice počítače v Internetu*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2008. 50 s. Vedoucí diplomové práce doc. Ing. Dan Komosný, Ph.D.
- [38] www.centos.org - *About CentOS* [online]. Dostupné z: <http://www.centos.org/modules/tinycontent/index.php?id=2> [cit. 14. 12. 2010].
- [39] RFC 1876. *A Means for Expressing Location Information in the Domain Name System*. University of Warwick, Davis; C. January 1996. Dostupné z WWW: <http://www.ietf.org/rfc/rfc1876.txt>.
- [40] JURKINA, M.; PICK, M. Numerické výpočty ve světovém geodetickém referenčním systému 1984 (WGS84). *Vojenský geografický obzor* [online]. 2006, č. 1, [cit. 2011-12-04]. Dostupný z WWW: http://www.army.cz/images/id_7001_8000/7162/VGO_1_2006_priloha_2.pdf.
- [41] *Open Directory Project - DMOZ : Největší lidmi spravovaný katalog odkazů* [online]. 2011 [cit. 2011-11-29]. DMOZ.cz > Úvodní stránka. Dostupné z WWW: <http://www.dmoz.cz/>.
- [42] Univerzita Hradec Králové. *Metoda Monte Carlo : Generátory náhodných čísel* [online]. Hradec Králové : Pedagogická fakulta, Katedra fyziky a informatiky, 2007 [cit. 2011-12-07]. Dostupné z WWW: <http://www.black-hole.cz/soubory/mc.pdf>.
- [43] MCCONNEL, Steve. MICROSOFT. *Code Complete: A Practical Handbook of Software Construction* [online]. 2004 [cit. 2012-05-15]. Dostupné z WWW: <http://xa.yimg.com/kq/groups/32249099/966630514/name/Microsoft.Press.Code.Complete.Second.Edition.eBook.pdf>.
- [44] HICKEY, Rich. *Clojure: Agents and Asynchronous Actions*. *Clojure* [online]. 2012 [cit. 2012-05-06]. Dostupné z WWW: <http://clojure.org/agents>.
- [45] YOURDON, Edward. *Just Enough Structured Analysis* [online]. 2006. vyd. [cit. 2012-05-06]. Dostupné z WWW: <http://www.yourdon.com/jesa/jesa.php>.
- [46] KROGH, Jan. The Centre of Europe. *Jan S. Krogh's Geosite: Other Places of Interest* [online]. 2012 [cit. 2012-05-12]. Dostupné z WWW: <http://geosite.jankrogh.com/other.htm>.
- [47] NAQPAL, Robin. Working with GeoLocations. In: *Xebee* [online]. 2010 [cit. 2012-05-12]. Dostupné z WWW: <http://xebbie.xebbie.in/2010/10/28/working-with-geolocations/>.
- [48] SYNEK, Václav. UJEP. *Rozdělení* [online]. Ústí nad Labem, 2011 [cit. 2012-05-06]. Dostupné z WWW: <http://fzp.ujep.cz/~synek/statistika/prednasky/less3aRoz.doc>.
- [49] VERNER, Lukáš. IP Geolocation. *IP Geolocation of Internet Nodes* [online]. 2012 [cit. 2012-05-09]. Dostupné z WWW: <http://betka.utko.feec.vutbr.cz>.
- [50] HLAVÁČ, Václav. *Neparametrické odhady hustoty pravděpodobnosti* [online]. Praha, 2008. Prezentace. Elektrotechnická fakulta ČVUT. [cit. 2012-05-10]. Dostupné z WWW: <http://cmp.felk.cvut.cz/~hlavac/TeachPresCz/31Rozp/18NeparametrOdhady.ppt>.

SEZNAM ZKRATEK

AS	Autonomous System
CBG	Constraint-based Geolocation
CDF	Cumulative distribution function
CDP	Cisco Discovery Protocol
CENTOS	Community ENTERprise Operating System
CESNET	Czech Education and Research Network
DMOZ	Open Directory Project
DNS	Domain Name System
GNU	GNU's Not Unix!
GPS	Global Positioning System
ICMP	Internet Control Message Protocol
IP	Internet Protocol
IPv6	Internet Protocol version 6
JRE	Java Runtime Environment
JSON	JavaScript Object Notation
KDE	Kernel Density Estimation
NAT	Network Address Translation
OS	Operating System
OSI	Open Systems Interconnection
P2P	Peer To Peer
Ping	Packet InterNet Groper
RB	Referenční bod
RFC	Request for Comments
RPM	Red Hat Package Manager
RTT	Round-Trip Delay Time [ms]
S	Sonda
SGP	Simple GeoPing
SOI	Speed of Internet
SSH	Secure Shell
T	Target
TBG	Topology-based Geolocation
TCP	Transmission Control Protocol
URL	Uniform Resource Locator
VNET	Virtual Network
VoIP	Voice over IP
VS	Virtual Server
WGS84	World Geodetic System 1984

SEZNAM PŘÍLOH

A	SKRIPTY PRO VYHLEDÁNÍ DOMÉN S DNS LOC ZÁZNAMEM.....	70
A.1	Skript LOClookup.....	70
A.2	Skript DNSrotateLOClookup.....	71
B	SEZNAM REFERENČNÍCH BODŮ A CÍLŮ	73
B.1	Seznam použitých referenčních bodů.....	73
B.2	Seznam použitých cílů	74
C	VÝSLEDKY MĚŘENÍ.....	76
C.1	Výsledky měření IP geolokační metody CBG	76
C.2	Výsledky všech metod	77
D	OBSAH PŘILOŽENÉHO CD	80

A SKRIPTY PRO VYHLEDÁNÍ DOMÉN S DNS LOC ZÁZNAMEM

A.1 Skript LOClookup

Tento skript napsal Jeff Hungerford v programovacím jazyce *Pike* a slouží k vyhledání DNS LOC záznamu domény, jejíž IP adresa či doménové jméno je vstupem skriptu.

```
#!/usr/local/bin/pike

int main()
{
    #Iteruje nad řádky ze standardního vstupu.
    foreach(Stdio.stdin.line_iterator();; string domain)
        #Spouští funkci lookup se vstupní proměnnou ze standardního vstupu.
        lookup(domain);
}

void lookup(string domain)
{
    #Deklarace proměnné typu Position z balíku Geography.
    Geography.Position location;
    #Do proměnné result typu mapping se uloží odpověď z dotazovaného serveru.
    mapping result = Protocols.DNS.client("8.8.8.8")->
        do_sync_query(Protocols.DNS.client()->
            mkquery(domain, Protocols.DNS.C_IN, Protocols.DNS.T_LOC));
    #Pokud je velikost odpovědi > 0, tak server odpověděl s DNS LOC záznamem.
    if (sizeof(result->an)>0)
        #Volání funkce na uložení DNS LOC záznamu ve formátu GPS.
        location = Geography.Position(result->an[0]->lat, result->an[0]->long);
        if(location)
        {
            #Vypíše na standardní výstup souřadnice z LOC záznamu.
            Stdio.stdout.write(domain + " " + location->lat + "," +
                location->long + "\n");
        }
}
```

A.2 Skript DNSrotateLOClookup

Tento skript napsal Bc. Petr Harth v programovacím jazyce *Python* a slouží pro rozložení zátěže na 16 DNS serverů při hledání LOC záznamů pomocí skriptu *LOClookup*.

```
#-----začátek konfigurace -----
#Počet skriptů LOClookup, kdy má každý z nich nastavený jiný DNS server.
pikeScripts = 16
pikeInterpret = "pike"
pikeScriptName = "./LOClookup" #Název skriptu bez jeho pořadového čísla.
statusLine = 50 #Po kolika řádcích se vypíše status na stderr.
#-----konec konfigurace-----

import sys
from subprocess import Popen, PIPE, STDOUT
actualScript = pikeScripts #globální proměnná

def getNextScriptName():
    '''Vrátí jméno dalšího volaného pike skriptu. Použije při tom 1 globální
    proměnnou a matematický výraz modulo + 1.'''
    global actualScript
    actualScript = actualScript + 1
    if (actualScript > pikeScripts):
        actualScript = 1
    #Jméno skriptu se skládá z konstantní části (pikeSkriptName),
    #potom z čísla a z přípony, která je opět pro všechny stejná.
    #Číslo se formátuje na 2 číslice, zleva doplněné nulou (01, 02, ..., 99).
    return pikeScriptName + "{0:02d}".format(actualScript) + ".pike"

def findLoc(address):
    '''Zavolá skript LOClookup pro nalezení LOC záznamu.'''
    #Vytvoří proces pro skript. Odchytne standardní vstup, výstup a chybový
    #výstup.
    dnsProc = Popen([pikeInterpret, getNextScriptName()], stdout=PIPE,
        stdin=PIPE, stderr=STDOUT)
    #Zavolá ho a vrátí výsledek.
    return dnsProc.communicate(address)[0]

def printOkLoc(loc):
    """Vytiskne location, pokud není prázdná nebo neobsahuje nuly."""
    if (len(loc) > 5):
        if (loc.find("0.0,0.0") == -1):
            print loc
```

```

def printLineNumToStdErr(num):
    '''Vypíše aktuální řádek (pokud je každý x-tý; nastavuje se v globální
        proměnné) na chybový výstup, aby nevadil v cílovém souboru.'''
    if ((num % statusLine) == 0):
        sys.stderr.write("current line: {0}\n".format(num))

def dnsRotateLookup(inputFileName):
    '''Postupně čte inputFile a zpracovává každý řádek zvlášť.'''
    inputFile = open(inputFileName, "r")
    currentLine = 1
    for line in inputFile:
        loc = findLoc(line)
        printOkLoc(loc[:-1]) #Poslední znak byl nový řádek.
        printLineNumToStdErr(currentLine)
        currentLine += 1

if (__name__ == "__main__"):
    if (len(sys.argv) < 2 or sys.argv[1] == "-h"):
        print "dnsRotateLookup.py <filename>\n\t V souboru by měly být adresy,
            pro které je nutno nalézt DNS LOC záznamy. Co řádek, to adresa.\n\t
            Tento skript rotuje několik skriptů LOClookup podle konfigurace.\n\t
            Výsledek je na stdout."
    else:
        dnsRotateLookup(sys.argv[1])

```


B SEZNAM REFERENČNÍCH BODŮ A CÍLŮ

B.1 Seznam použitých referenčních bodů

Organizace	DNS jméno	IP adresa	Šířka [°]	Délka [°]
Max Planck Institute for Software Systems	planetlab05.mpi-sws.mpg.de	139.19.142.5	49,26	7,05
CESNET - Czech Education and Research Network	planetlab1.fit.vutbr.cz	147.229.10.250	49,23	16,60
Space Monitoring Data Center SINP MSU	plab-1.sinp.msu.ru	213.131.1.101	55,70	37,53
Universite Catholique de Louvain	onelab2.info.ucl.ac.be	130.104.72.201	50,67	4,61
Warsaw University of Technology Europe	ple2.ise.pw.edu.pl	194.29.150.132	52,22	21,01
Universidad Carlos III Madrid	planetlab2.uc3m.es	163.117.253.23	40,33	-3,77
delab-auth	planet2.itc.auth.gr	155.207.48.52	40,63	22,96
Polish-Japanese Institute of Information Technology	planetlab1.pjwstk.edu.pl	148.81.140.193	52,22	20,99
Universitat Pompeu Fabra - NeTS	planetlabpc2.upf.edu	193.145.46.130	41,40	2,19
ETH Zuerich	planetlab01.ethz.ch	192.33.90.195	47,38	8,55
Troyes University of Technology	planetlab1.utt.fr	194.254.215.11	48,27	4,07
University of Essex	uoep1.essex.ac.uk	155.245.47.225	51,88	0,94
Eotvos Lorand University	planet1.elte.hu	157.181.175.249	47,47	19,06
IBBT	planck228ple.test.ibbt.be	193.191.148.228	51,04	3,74
University of Parma	planet2.unipr.it	160.78.253.32	44,80	10,33
Technical University of Lodz	ple1.dmcs.p.lodz.pl	212.51.218.235	51,75	19,45
The Koszalin University of Technology	ple2.tu.koszalin.pl	62.108.171.76	54,20	16,20
Universite de Liege	planetlab1.montefiore.ulg.ac.be	139.165.12.211	50,58	5,56
Petru Maior University of Targu Mures	planetlab1.upm.ro	193.226.19.30	46,55	24,57
Tampere University of Technology	planetlab1.rd.tut.fi	193.166.167.4	61,45	23,86
University of Wuerzburg	planetlab2.informatik.uni-wuerzburg.de	132.187.230.2	49,79	9,94
Technische Universitaet Berlin	planetlab01.tkn.tu-berlin.de	130.149.49.136	52,51	13,33
Netgroup at University of Rome Tor Vergata	planet-lab-node1.netgroup.uniroma2.it	160.80.221.37	41,85	12,62
CCTC / Universidade do Minho	planetlab-um00.di.uminho.pt	193.136.19.13	41,56	-8,40
University of Cambridge	planetlab1.xeno.cl.cam.ac.uk	128.232.103.201	52,21	0,09
ITEC, Klagenfurt University	plab1-itec.uni-klu.ac.at	143.205.172.11	46,62	14,27
University College Dublin	planetlab-node-01.ucd.ie	193.1.170.135	53,31	-6,22
Universidad Publica de Navarra	planetlab1.tlm.unavarra.es	130.206.158.138	42,80	-1,64
University of Stuttgart - IPVS	planetvs2.informatik.uni-stuttgart.de	129.69.210.97	48,78	9,18
University of Innsbruck	plab2-c703.uibk.ac.at	138.232.66.174	47,26	11,38
Universidade do Algarve	planetlab2.fct.ualg.pt	193.136.227.164	37,02	-7,94
University of Sevilla	ait05.us.es	193.147.162.166	37,38	-5,99
University of Passau	aladdin.planetlab.extranet.uni-passau.de	195.37.16.121	48,57	13,45
University of Ioannina	planetlab2.cs.uoi.gr	195.130.121.205	39,62	20,84
University of Duisburg-Essen	planetlab1.exp-math.uni-essen.de	132.252.152.193	51,46	7,01
Dipartimento di Informatica di Torino	planetlab2.di.unito.it	130.192.157.132	45,09	7,66
University of St Andrews	planetlab3.cs.st-andrews.ac.uk	138.251.214.77	56,34	-2,80
University of Tromso	planetlab1.cs.uit.no	129.242.19.196	69,68	18,97
University of Oslo	planetlab2.ifi.uio.no	193.157.115.251	59,94	10,72
University of Rostock	pl1.uni-rostock.de	139.30.240.191	54,08	12,11

B.2 Seznam použitých cílů

IP adresa	DNS jméno	Šířka [°]	Délka [°]
131.188.32.37	2009.nwerc.eu	49,45	11,08
80.68.89.248	69lovesongs.info	51,38	-2,36
81.187.30.81	aaisp.net.uk	51,42	-0,77
193.205.194.23	alpha.science.unitn.it	46,07	11,15
156.35.96.172	bellman.ciencias.uniovi.es	43,36	-5,85
129.132.98.37	cam.switch.ch	47,38	8,55
156.35.173.54	carleos.epv.uniovi.es	43,52	-5,62
129.199.32.25	culturesciences.chimie.ens.fr	48,84	2,35
81.187.11.27	dell.wensley.org.uk	52,25	-2,14
192.134.6.120	electron.nic.fr	48,79	2,05
80.68.89.248	elrino.co.uk	51,38	-2,36
129.199.115.128	fip.phys.ens.fr	48,84	2,35
193.201.200.77	firedrake.org	51,53	-0,73
193.109.80.1	fk.lutsk.ua	50,75	25,33
160.45.170.10	fu-berlin.de	52,46	13,29
81.187.223.197	garyhawkins.me.uk	51,83	-0,81
83.175.178.18	goldengate.fydryk.com	50,09	19,94
156.35.119.120	horru.lsi.uniovi.es	43,52	-5,62
134.96.102.171	is.uni-sb.de	49,26	7,05
193.175.80.161	j3e.de	51,53	9,93
194.87.13.69	koi8.pp.ru	55,75	37,58
91.200.128.13	linkwave.it	42,78	11,10
195.220.94.165	listes.cru.fr	48,12	-1,64
129.199.80.1	lumiere.ens.fr	48,84	2,35
83.218.25.88	mcb.net	54,17	-4,47
82.238.220.82	nanardon.zarb.org	48,73	2,29
81.31.33.35	nomi.cz	49,20	16,62
188.40.65.74	nona.net	48,19	16,33
212.111.4.214	ns3.cngroup.cz	49,23	17,69
156.35.94.1	petra.euitio.uniovi.es	43,36	-5,85
129.199.129.64	quatramaran.ens.fr	48,84	2,35
134.225.1.128	rdg.ac.uk	51,44	-0,95
89.207.56.140	rte.ie	53,32	-6,21
77.68.170.97	simplifiedns.com	57,45	10,54
91.220.127.245	sittingbourne.bellgrim.se	51,31	0,73
193.219.33.197	soften.ktu.lt	54,90	23,96
213.73.92.40	star.trek.org	52,48	13,32
81.187.252.184	stat.wensley.org.uk	52,25	-2,14
87.121.7.244	stolitz.com	42,67	23,37
193.219.28.2	sunsite.icm.edu.pl	52,21	20,98
130.59.108.36	switch.ch	47,38	8,55
129.242.5.90	uit.no	69,68	18,98
134.21.213.189	unifr.ch	46,81	7,15
80.81.76.82	village.albourne.com	51,50	-0,02
195.39.234.226	wunderweib.de	53,55	10,00
129.199.30.64	www.biologie.ens.fr	48,84	2,35
80.87.131.127	www.bluemantle.co.uk	52,59	0,06
129.199.2.23	www.bourbaki.ens.fr	48,84	2,35
195.64.165.26	www.cg13.fr	43,31	5,41
80.87.131.127	www.combinationfireplaces.co.uk	52,59	0,06
195.220.94.166	www.cru.fr	48,12	-1,64
80.193.133.173	www.cyberquest.co.uk	52,60	-0,72
129.199.13.75	www.delta.ens.fr	48,84	2,35
129.199.2.22	www.di.ens.fr	48,84	2,35
78.46.105.215	www.disch-systems.de	50,47	12,37
129.199.2.29	www.dmi.ens.fr	48,84	2,35
129.199.1.129	www.eleves.ens.fr	48,84	2,35
129.199.13.15	www.ens.fr	48,84	2,35
213.245.116.198	www.erasme.org	45,75	4,43
80.87.131.127	www.fletchingfc.co.uk	52,59	0,06
129.199.13.25	www.geographie.ens.fr	48,84	2,35
80.87.131.127	www.hcidata.co.uk	52,59	0,06
217.29.35.149	www.chatnbrett.de	49,01	8,43
129.199.13.24	www.ihmc.ens.fr	48,84	2,35
46.183.8.138	www.kentalc.gov.uk	51,50	-0,10
80.87.131.127	www.la-potiniere.co.uk	52,59	0,06
129.199.119.106	www.lkb.ens.fr	48,84	2,35

129.199.120.253	www.lps.ens.fr	48,84	2,35
80.87.131.127	www.pevenseyparishcouncil.gov.uk	52,59	0,06
194.63.248.26	www.pgpi.org	59,93	10,67
129.199.115.253	www.phys.ens.fr	48,84	2,35
78.47.160.16	www.roehnfried-hesse.de	53,55	10,00
80.87.131.127	www.shereparishcouncil.gov.uk	52,59	0,06
212.60.1.2	www.shlink.de	53,75	9,66
80.193.133.173	www.uk1.co.uk	52,60	-0,72
80.87.131.127	www.whra.org.uk	52,59	0,06

C VÝSLEDKY MĚŘENÍ

C.1 Výsledky měření IP geolokační metody CBG

IP adresa	DNS jméno	Pravá šířka [°]	Pravá délka [°]	Odhadnutá šířka [°]	Odhadnutá délka [°]	Průměrná chyba odhadu [km]	Obsah polygonu [km2]	Průměrný čas výpočtu [s]
131.188.32.37	2009.nwerc.eu	49,45	11,08	49,75	10,05	89,13	44021	53,00
80.68.89.248	69lovesongs.info	51,38	-2,36	53,03	2,48	377,95	128870	82,00
193.205.194.23	alpha.science.unitn.it	46,07	11,15	44,07	11,16	221,67	1097026	127,00
156.35.96.172	bellman.ciencias.uniovi.es	43,36	-5,85	39,78	-1,56	534,46	283165	141,00
129.132.98.37	cam.switch.ch	47,38	8,55	47,18	8,46	24,40	0	98,50
156.35.173.54	carleos.epv.uniovi.es	43,52	-5,62	39,71	-1,35	552,69	336031	57,00
81.187.11.27	dell.wensley.org.uk	52,25	-2,14	53,06	-2,18	114,30	6491298	94,00
80.68.89.248	elrino.co.uk	51,38	-2,36	53,26	2,72	403,69	135730	52,00
129.199.115.128	fip.phys.ens.fr	48,84	2,35	48,14	0,69	144,77	40304	58,00
193.109.80.1	fk.lutsk.ua	50,75	25,33	55,76	30,99	671,62	994888	59,00
160.45.170.10	fu-berlin.de	52,46	13,29	53,75	11,28	196,76	79314	52,00
83.175.178.18	goldengate.fydryk.com	50,09	19,94	50,17	11,98	567,20	3942765	52,00
156.35.119.120	horru.lsi.uniovi.es	43,52	-5,62	40,82	-2,57	392,45	363861	56,00
134.96.102.171	is.uni-sb.de	49,26	7,05	48,23	7,96	132,62	80682	59,00
193.175.80.161	j3e.de	51,53	9,93	52,12	10,43	74,06	70628	55,00
194.87.13.69	koi8.pp.ru	55,75	37,58	50,95	9,80	2108,03	9315834	67,50
91.200.128.13	linkwave.it	42,78	11,10	44,40	14,01	295,47	0	62,00
195.220.94.165	listes.cru.fr	48,12	-1,64	48,09	1,16	207,73	486006	52,00
129.199.80.1	lumiere.ens.fr	48,84	2,35	48,19	0,69	141,45	29942	56,00
83.218.25.88	mcb.net	54,17	-4,47	52,70	3,63	560,40	513008	108,00
82.238.220.82	nanardon.zarb.org	48,73	2,29	50,45	0,34	237,28	2547759	69,00
81.31.33.35	nomi.cz	49,20	16,62	49,09	17,58	71,38	169445	70,00
188.40.65.74	nona.net	48,19	16,33	49,95	10,31	480,74	181985	55,00
212.111.4.214	ns3.cngroup.cz	49,23	17,69	47,40	17,92	203,96	496455	150,00
156.35.94.1	petra.euitio.uniovi.es	43,36	-5,85	39,34	-3,81	478,89	122791	170,00
129.199.129.64	quatramaran.ens.fr	48,84	2,35	48,18	0,63	146,30	33846	233,00
134.225.1.128	rdg.ac.uk	51,44	-0,95	51,83	0,58	124,60	104021	202,50
89.207.56.140	rte.ie	53,32	-6,21	52,40	5,07	763,37	841771	67,00
77.68.170.97	simplifiedns.com	57,45	10,54	53,55	11,35	436,73	290129	53,00
193.219.33.197	soften.ktu.it	54,90	23,96	52,75	18,13	451,34	0	147,00
213.73.92.40	star.trek.org	52,48	13,32	49,85	9,42	399,08	905093	149,00
81.187.252.184	stat.wensley.org.uk	52,25	-2,14	53,96	-0,21	229,51	7345396	86,00
87.121.7.244	stolitz.com	42,67	23,37	45,77	29,95	627,53	0	73,00
193.219.28.2	sunsite.icm.edu.pl	52,21	20,98	52,14	20,75	19,13	0	65,67
130.59.108.36	switch.ch	47,38	8,55	47,34	8,53	5,66	0	59,67
134.21.213.189	unifr.ch	46,81	7,15	45,68	7,20	125,57	58	230,00
80.81.76.82	village.albourne.com	51,50	-0,02	50,81	6,30	446,99	647600	71,00
195.39.234.226	wunderweib.de	53,55	10,00	52,44	12,23	193,21	118355	58,00
129.199.30.64	www.biologie.ens.fr	48,84	2,35	48,07	0,08	188,04	95216	57,00
80.87.131.127	www.bluemantle.co.uk	52,59	0,06	51,67	1,73	153,08	218352	56,00
129.199.2.23	www.bourbaki.ens.fr	48,84	2,35	48,15	-0,14	198,89	40837	141,00
195.64.165.26	www.cg13.fr	43,31	5,41	47,10	9,04	507,74	0	55,00
80.87.131.127	www.combinationfireplaces.co.uk	52,59	0,06	51,88	1,65	134,14	235628	58,00
195.220.94.166	www.cru.fr	48,12	-1,64	48,06	0,64	169,77	533471	52,00
80.193.133.173	www.cyberquest.co.uk	52,60	-0,72	50,45	-0,86	239,09	22696	59,00
129.199.13.75	www.delta.ens.fr	48,84	2,35	48,16	-0,05	192,42	45530	50,00
78.46.105.215	www.disch-systems.de	50,47	12,37	49,79	9,77	199,82	147497	64,00
129.199.2.29	www.dmi.ens.fr	48,84	2,35	48,16	-0,10	195,87	40973	55,00
129.199.1.129	www.eleves.ens.fr	48,84	2,35	48,18	-0,02	189,09	38338	58,00
129.199.13.15	www.ens.fr	48,84	2,35	48,14	0,14	180,58	43856	59,00
213.245.116.198	www.erasme.org	45,75	4,43	48,64	2,92	341,48	1070943	98,00

80.87.131.127	www.fletchingfc.co.uk	52,59	0,06	51,85	1,57	131,47	232354	50,00
129.199.13.25	www.geographie.ens.fr	48,84	2,35	48,14	0,12	181,53	43882	49,00
80.87.131.127	www.hcidata.co.uk	52,59	0,06	51,92	1,65	131,62	234224	51,00
217.29.35.149	www.chatnbrett.de	49,01	8,43	49,42	10,51	157,33	1091454	51,00
129.199.13.24	www.ihmc.ens.fr	48,84	2,35	48,16	0,06	184,65	47637	61,00
46.183.8.138	www.kentalc.gov.uk	51,50	-0,10	53,07	-0,44	176,49	165074	50,00
80.87.131.127	www.la-potiniere.co.uk	52,59	0,06	51,94	1,68	132,42	227917	111,00
129.199.119.106	www.lkb.ens.fr	48,84	2,35	48,30	0,13	173,94	63923	62,00
129.199.120.253	www.lps.ens.fr	48,84	2,35	48,21	0,19	173,78	55789	63,00
80.87.131.127	www.pevenseyparishcouncil.gov.uk	52,59	0,06	51,92	1,65	131,38	229053	85,00
129.199.115.253	www.phys.ens.fr	48,84	2,35	48,19	-0,08	192,87	50656	49,00
78.47.160.16	www.roehnfried-hesse.de	53,55	10,00	49,44	9,93	457,68	107353	52,00
80.87.131.127	www.shereparishcouncil.gov.uk	52,59	0,06	51,90	1,65	132,48	230168	49,00
212.60.1.2	www.shlink.de	53,75	9,66	53,61	14,45	316,24	122667	52,00
80.193.133.173	www.uk1.co.uk	52,60	-0,72	51,53	-1,37	127,33	163366	50,00
80.87.131.127	www.whra.org.uk	52,59	0,06	51,90	1,62	131,17	228852	61,00

C.2 Výsledky všech metod

ID na webu	IP adresa	DNS jméno	SOI		CBG	
			Průměrná chyba odhadu [km]	Průměrný čas výpočtu [s]	Průměrná chyba odhadu [km]	Průměrný čas výpočtu [s]
238	81.187.30.81	aaisp.net.uk	107,52	21,43		
246	129.199.32.25	culturesciences.chimie.ens.fr				
249	192.134.6.120	electron.nic.fr	174,54	16,19		
252	193.201.200.77	firedrake.org	94,38	21,54		
280	91.220.127.245	sittingbourne.bellgrim.se	108,92	22,14		
288	129.242.5.90	uit.no				
301	129.199.2.22	www.di.ens.fr				
320	194.63.248.26	www.pgpi.org	3,17	23,38		
286	130.59.108.36	switch.ch	0,47	26,23	5,66	59,67
285	193.219.28.2	sunsite.icm.edu.pl	2,53	27,00	19,13	65,67
243	129.132.98.37	cam.switch.ch	0,32	19,28	24,40	98,50
270	81.31.33.35	nomi.cz	3,64	22,83	71,38	70,00
261	193.175.80.161	j3e.de	60,38	21,96	74,06	55,00
247	131.188.32.37	2009.nwerc.eu	158,49	24,97	89,13	53,00
277	81.187.11.27	dell.wensley.org.uk	132,13	21,28	114,30	94,00
236	134.225.1.128	rdg.ac.uk	90,87	19,18	124,60	202,50
289	134.21.213.189	unifr.ch	123,24	21,69	125,57	230,00
325	80.193.133.173	www.uk1.co.uk	39,53	19,97	127,33	50,00
326	80.87.131.127	www.whra.org.uk	42,29	24,23	131,17	61,00
319	80.87.131.127	www.pevenseyparishcouncil.gov.uk	42,29	18,66	131,38	85,00
307	80.87.131.127	www.fletchingfc.co.uk	42,29	23,89	131,47	50,00
309	80.87.131.127	www.hcidata.co.uk	42,29	23,19	131,62	51,00
314	80.87.131.127	www.la-potiniere.co.uk	42,29	19,56	132,42	111,00
323	80.87.131.127	www.shereparishcouncil.gov.uk	42,29	19,77	132,48	49,00
260	134.96.102.171	is.uni-sb.de	61,58	23,44	132,62	59,00
297	80.87.131.127	www.combinationfireplaces.co.uk	42,29	18,38	134,14	58,00
266	129.199.80.1	lumiere.ens.fr	189,94	24,35	141,45	56,00
251	129.199.115.128	fip.phys.ens.fr	192,88	20,67	144,77	58,00
275	129.199.129.64	quatramaran.ens.fr	185,80	23,40	146,30	233,00
294	80.87.131.127	www.bluemantle.co.uk	42,30	24,80	153,08	56,00
310	217.29.35.149	www.chatnbrett.de	131,60	20,16	157,33	51,00
298	195.220.94.166	www.cru.fr	130,88	21,05	169,77	52,00
316	129.199.120.253	www.lps.ens.fr	202,35	20,99	173,78	63,00
315	129.199.119.106	www.lkb.ens.fr	200,29	20,28	173,94	62,00
313	46.183.8.138	www.kentalc.gov.uk	146,56	18,54	176,49	50,00
305	129.199.13.15	www.ens.fr	184,94	23,38	180,58	59,00
308	129.199.13.25	www.geographie.ens.fr	187,06	26,01	181,53	49,00
311	129.199.13.24	www.ihmc.ens.fr	187,59	19,12	184,65	61,00
293	129.199.30.64	www.biologie.ens.fr	195,65	16,11	188,04	57,00

304	129.199.1.129	www.eleves.ens.fr	188,83	21,90	189,09	58,00
300	129.199.13.75	www.delta.ens.fr	190,44	20,74	192,42	50,00
321	129.199.115.253	www.phys.ens.fr	188,29	20,14	192,87	49,00
292	195.39.234.226	wunderweib.de	163,75	21,81	193,21	58,00
303	129.199.2.29	www.dmi.ens.fr	190,24	20,55	195,87	55,00
254	160.45.170.10	fu-berlin.de	6,64	21,48	196,76	52,00
295	129.199.2.23	www.bourbaki.ens.fr	188,68	19,52	198,89	141,00
302	78.46.105.215	www.disch-systems.de	205,67	22,22	199,82	64,00
272	212.111.4.214	ns3.cngroup.cz	79,16	19,12	203,96	150,00
265	195.220.94.165	listes.cru.fr	130,85	21,36	207,73	52,00
240	193.205.194.23	alpha.science.unitn.it	154,66	22,33	221,67	127,00
283	81.187.252.184	stat.wensley.org.uk	171,44	19,81	229,51	86,00
269	82.238.220.82	nanardon.zarb.org	159,69	21,97	237,28	69,00
299	80.193.133.173	www.cyberquest.co.uk	39,94	23,20	239,09	59,00
263	91.200.128.13	linkwave.it	346,86	24,99	295,47	62,00
324	212.60.1.2	www.shlink.de	252,88	22,30	316,24	52,00
306	213.245.116.198	www.erasme.org	368,45	23,04	341,48	98,00
237	80.68.89.248	69lovesongs.info	178,49	21,49	377,95	82,00
258	156.35.119.120	horru.lsi.uniovi.es	386,88	21,45	392,45	56,00
282	213.73.92.40	star.trek.org	394,01	19,74	399,08	149,00
250	80.68.89.248	elrino.co.uk	177,04	19,12	403,69	52,00
279	77.68.170.97	simplifiedns.com	214,99	21,13	436,73	53,00
291	80.81.76.82	village.albourne.com	501,17	19,39	446,99	71,00
281	193.219.33.197	soften.ktu.lt	385,92	21,99	451,34	147,00
322	78.47.160.16	www.roehnfried-hesse.de	445,97	19,50	457,68	52,00
273	156.35.94.1	petra.eutio.uniovi.es	379,00	23,99	478,89	170,00
271	188.40.65.74	nona.net	501,21	25,98	480,74	55,00
296	195.64.165.26	www.cg13.fr	532,70	24,76	507,74	55,00
242	156.35.96.172	bellman.ciencias.uniovi.es	400,50	20,72	534,46	141,00
244	156.35.173.54	carleos.epv.uniovi.es	413,98	23,22	552,69	57,00
268	83.218.25.88	mcb.net	456,34	23,63	560,40	108,00
257	83.175.178.18	goldengate.fydryk.com	248,83	18,80	567,20	52,00
284	87.121.7.244	stolitza.com	441,90	20,11	627,53	73,00
253	193.109.80.1	fk.lutsk.ua	502,77	22,36	671,62	59,00
278	89.207.56.140	rte.ie	1,31	22,84	763,37	67,00
262	194.87.13.69	koi8.pp.ru	6,21	20,10	2108,03	67,50
Průměr			183,77	21,61	284,69	78,48
Medián			172,99	21,49	192,87	59,00

ID na webu	IP adresa	DNS jméno	GeoWeight		Octant		GeoIP	
			Průměrná chyba odhadu [km]	Průměrný čas výpočtu [s]	Průměrná chyba odhadu [km]	Průměrný čas výpočtu [s]	Průměrná chyba odhadu [km]	Průměrný čas výpočtu [s]
238	81.187.30.81	aaisp.net.uk	236,75	374,23			471,47	0,01
246	129.199.32.25	culturesciences.chimie.ens.fr					2,95	0,01
249	192.134.6.120	electron.nic.fr	184,76	352,46	133,82	30,00		
252	193.201.200.77	firedrake.org	328,11	343,39	215,97	27,00	287,91	0,01
280	91.220.127.245	sittingbourne.bellgrim.se	158,36	325,71	128,61	28,00	350,84	0,01
288	129.242.5.90	uit.no					1,69	0,01
301	129.199.2.22	www.di.ens.fr					2,95	0,01
320	194.63.248.26	www.pgpi.org	81,16	370,63	295,00	27,00	232,67	0,01
286	130.59.108.36	switch.ch	45,79	320,79	177,42	29,00	1,54	0,01
285	193.219.28.2	sunsite.icm.edu.pl	33,72	333,31	346,12	32,00	4,89	0,01
243	129.132.98.37	cam.switch.ch	209,11	311,11	320,23	30,00	1,41	0,01
270	81.31.33.35	nomi.cz	213,09	372,28	116,35	27,00	1,19	0,01
261	193.175.80.161	j3e.de	491,73	336,07	91,21	26,00	0,00	0,00
247	131.188.32.37	2009.nwerc.eu	316,16	378,92			107,24	0,01
277	81.187.11.27	dell.wensley.org.uk	220,62	345,75	383,09	28,00	3,81	0,01
236	134.225.1.128	rdg.ac.uk	80,39	350,94	93,79	349,00	16,56	0,01
289	134.21.213.189	unifr.ch	105,12	320,94	89,34	28,00	0,74	0,01
325	80.193.133.173	www.uk1.co.uk	77,60	433,45	359,25	28,50	177,54	0,01
326	80.87.131.127	www.whra.org.uk	171,24	348,46	345,07	26,75	208,09	0,01
319	80.87.131.127	www.pevenseyparishcouncil.gov.uk	120,92	335,95	212,61	26,75	208,09	0,01
307	80.87.131.127	www.fletchingfc.co.uk	117,27	392,07	236,69	26,00	208,09	0,01

309	80.87.131.127	www.hcidata.co.uk	111,91	361,90	242,87	27,33	208,09	0,01
314	80.87.131.127	www.la-potiniere.co.uk	115,81	354,84	537,90	27,50	208,09	0,01
323	80.87.131.127	www.shereparishcouncil.gov.uk	204,96	314,23	204,10	26,50		
260	134.96.102.171	is.uni-sb.de	329,84	328,80	307,82	30,00	4,23	0,01
297	80.87.131.127	www.combinationfireplaces.co.uk	122,15	395,43	271,97	26,50	208,09	0,01
266	129.199.80.1	lumiere.ens.fr	187,20	393,66	237,75	30,00	2,95	0,01
251	129.199.115.128	fip.phys.ens.fr	239,38	346,48	116,98	28,00	2,95	0,01
275	129.199.129.64	quatramaran.ens.fr	107,00	315,36	130,60	25,00	2,95	0,01
294	80.87.131.127	www.bluemantle.co.uk	117,58	338,23	242,86	28,60	208,09	0,01
310	217.29.35.149	www.chatnbrett.de	275,24	380,64	83,15	27,00	224,78	0,01
298	195.220.94.166	www.cru.fr	191,45	400,08	113,43	28,00	434,50	0,01
316	129.199.120.253	www.lps.ens.fr	99,63	352,75	253,77	29,00	2,95	0,01
315	129.199.119.106	www.lkb.ens.fr	123,49	321,53	122,11	30,00	2,95	0,01
313	46.183.8.138	www.kentalc.gov.uk			249,02	27,00	306,00	0,01
305	129.199.13.15	www.ens.fr	109,05	317,76	69,43	27,00	2,95	0,01
308	129.199.13.25	www.geographie.ens.fr	103,19	320,69	50,06	27,00	2,95	0,01
311	129.199.13.24	www.ihmc.ens.fr	118,39	347,02	67,89	26,00	2,95	0,01
293	129.199.30.64	www.biologie.ens.fr	127,57	368,02	156,14	26,50		
304	129.199.1.129	www.eleves.ens.fr	108,74	331,25	88,60	26,00	2,95	0,01
300	129.199.13.75	www.delta.ens.fr	117,59	331,20	95,22	27,00	2,95	0,01
321	129.199.115.253	www.phys.ens.fr	166,78	352,06	198,58	30,00	2,95	0,01
292	195.39.234.226	wunderweib.de	126,42	316,19	360,41	26,00	0,36	0,01
303	129.199.2.29	www.dmi.ens.fr	113,86	308,69	121,06	29,00	2,95	0,01
254	160.45.170.10	fu-berlin.de	317,36	331,18	344,17	29,00	9,87	0,01
295	129.199.2.23	www.bourbaki.ens.fr	113,08	353,42	61,81	27,00	2,95	0,01
302	78.46.105.215	www.disch-systems.de	176,59	362,91	144,32	33,00	146,54	0,01
272	212.111.4.214	ns3.cngroup.cz	217,30	336,65	169,29	27,00	8,69	0,01
265	195.220.94.165	listes.cru.fr	431,93	361,36	101,36	30,00	434,50	0,01
240	193.205.194.23	alpha.science.unitn.it	401,92	334,25	334,09	605,00	1,33	0,01
283	81.187.252.184	stat.wensley.org.uk	245,32	345,50	468,19	68,00	107,23	0,01
269	82.238.220.82	nanardon.zarb.org	223,03	320,51	597,81	29,00	0,43	0,01
299	80.193.133.173	www.cyberquest.co.uk	83,96	316,54	171,95	28,50	177,54	0,01
263	91.200.128.13	linkwave.it	277,79	375,38	354,58	34,00	3,36	0,01
324	212.60.1.2	www.shlink.de	336,68	341,43	75,39	26,00	309,45	0,01
306	213.245.116.198	www.erasme.org	380,16	326,77	951,24	27,00	106,10	0,01
237	80.68.89.248	69lovesongs.info	548,84	329,35	422,38	818,00	292,89	0,01
258	156.35.119.120	horru.lsi.uniovi.es	692,96	333,13	384,29	28,00	24,32	0,01
282	213.73.92.40	star.trek.org	87,44	318,94	427,79	26,00	27,90	0,01
250	80.68.89.248	elrino.co.uk	558,63	324,39	377,28	28,00	292,89	0,01
279	77.68.170.97	simplifiedns.com	294,35	357,47	47,27	34,00	2,29	0,01
291	80.81.76.82	village.albourne.com	512,27	453,66	468,42	28,50	308,62	0,01
281	193.219.33.197	soften.ktu.lt	140,21	347,76	370,12	30,00	3,73	0,01
322	78.47.160.16	www.roehnfried-hesse.de	395,08	338,13	356,25	28,00	291,67	0,01
273	156.35.94.1	petra.euitio.uniovi.es			300,67	28,00	1,79	0,01
271	188.40.65.74	nona.net	542,38	320,92	302,19	28,00	614,06	0,01
296	195.64.165.26	www.cg13.fr	130,26	376,75	1039,92	26,00	402,16	0,01
242	156.35.96.172	bellman.ciencias.uniovi.es	628,41	399,15	120,52	313,00	1,79	0,01
244	156.35.173.54	carleos.epv.uniovi.es	631,71	353,22	86,70	30,00	24,32	0,01
268	83.218.25.88	mcb.net	421,97	328,44	277,41	28,00	162,00	0,01
257	83.175.178.18	goldengate.fydryk.com	360,22	328,96			1,39	0,01
284	87.121.7.244	stolitz.com					138,28	0,01
253	193.109.80.1	fk.lutsk.ua	736,75	442,83			515,76	0,01
278	89.207.56.140	rte.ie	105,58	341,61	518,28	37,00	124,56	0,01
262	194.87.13.69	koi8.pp.ru	48,05	370,00			2,04	0,01
Průměr			239,85	349,48	259,70	58,76	120,39	0,01
Medián			184,76	343,39	237,22	28,00	13,22	0,01

D OBSAH PŘILOŽENÉHO CD

Příložený disk obsahuje následující složky:

Diplomova_prace – elektronická verze diplomové práce.

Vysledky – tabulky se všemi výsledky měření.

Referencni_body_a_cile – seznamy referenčních bodů a cílů.

CBGfinder:

- *Klient_CBGfinder* – zdrojové kódy klientské části programu *CBGfinder*.
 - *CBGClientForNormalComputer* – upravená klientská část pro spuštění na počítači bez webového serveru.
 - *CBGClientWebServer* – klientská část běžící na webovém serveru.
- *Server_CBGfinder* – zdrojové kódy serverové části programu *CBGfinder*.
 - *CBGServerOnPlanetLabNode* – převod zpoždění na hranici vzdálenosti.
 - *ComputeBestlineOnNode* – počítání přímky *Bestline*.
 - *TCPconnectionServer* – démon starající se o komunikaci s klientem.