

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

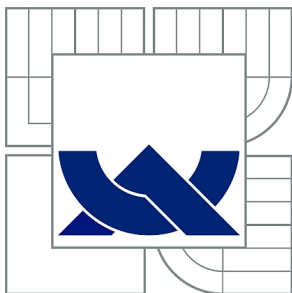
LOKALIZACE STANIC V SÍTI INTERNET POMOCÍ SYSTÉMU PRO
PŘEKLAD
DOMÉNOVÝCH JMÉN

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

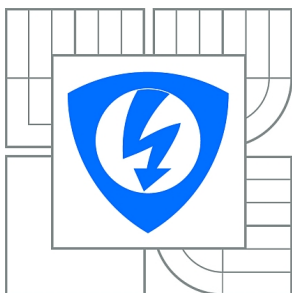
PETR DVOŘÁK

BRNO 2010



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

LOKALIZACE STANIC V SÍTI INTERNET POMOCÍ SYSTÉMU PRO PŘEKLAD DOMÉNOVÝCH JMÉN

NODE LOCALIZATION ON THE INTERNET USING DOMAIN NAME SYSTEM

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

PETR DVOŘÁK

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. DAN KOMOSNÝ, Ph.D.

BRNO 2010



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student: Petr Dvořák

ID: 106419

Ročník: 3

Akademický rok: 2009/2010

NÁZEV TÉMATU:

**Lokalizace stanic v síti Internet pomocí systému pro překlad
doménových jmén**

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s principy vyhodnocování polohy stanic v síti Internet. Zaměřte se na odhad pozice stanic pomocí metody King, která pracuje se systémem doménových jmén (DNS). Provedte simulace činnosti této metody ve zvoleném simulačním prostředí. Dále proveďte porovnání metody King s ostatními systémy pro určení pozice stanic v síti Internet. Při porovnání se zaměřte na dosahovanou přesnost, rychlost a náročnost odhadu pozice.

DOPORUČENÁ LITERATURA:

[1] GUMMADI, K. SAROIU, S. GRIBBLE, S. King: Estimating Latency between Arbitrary Internet End Hosts [online]. SIGCOMM Internet Measurement Workshop 2002. Association for Computing Machinery, 2002.

URL: <<http://www.mpi-sws.org/~gummadi/king/king.pdf>> [cit. 13. 10. 2009].

[2] KAASHOEK, M et al. The King data set [online]. Cambridge (USA):Massachusetts Institute of Technology, Computer Science and Artificial Intelligence Laboratory, 2003.

URL: <<http://pdos.csail.mit.edu/p2psim/kingdata/>> [cit. 13. 10. 2009].

[3] EUGENE, T. S., ZHANG, H. Predicting Internet Network Distance with Coordinates-Based Approaches. Proceedings of 21st Annual Joint Conference of the IEEE Computer and Communications Societies. IEEE, 2002.

Termín zadání: 29.1.2010

Termín odevzdání: 2.6.2010

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

prof. Ing. Kamil Vrba, CSc.

Předseda oborové rady

UPOZORNĚNÍ:

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

ABSTRAKT

Tato práce popisuje různé systémy, které slouží k predikci zpoždění mezi stanicemi. Zaměřuje se jak na systémy využívající k predikci zpoždění umělých souřadnicových systémů, tak na systémy, které využívají fyzická měření zpoždění. Primárně se však zabývá metodou King, která k predikci zpoždění využívá systém pro překlad doménových jmen DNS (Domain Name System). Dále je podrobně nastíněn princip metody King. Praktická část zahrnuje popis dvou vytvořených simulací v programu Matlab. První popisovaná simulace ukazuje princip metody King. Druhá simulace má poté za úkol vytvořit 2D geometrický prostor, ve kterém jsou náhodně generovány uzly a DNS servery. Jejím výsledkem je závislost přesnosti predikce RTT na počtu náhodně generovaných DNS serverů. Na základě této závislosti je metoda King v závěru porovnávána s dalšími metodami.

KLÍČOVÁ SLOVA

Metoda King, predikce RTT, systém DNS, DNS server, simulace, zpoždění

ABSTRACT

This paper describes various systems that are used to predict delay among nodes . It focuses on systems using for prediction delay artificial coordinate systems, and on the systems, which use physical measurements of delay. Primarily, however, it deals with the King method, which uses the system for translation of domain names - DNS (Domain Name System) to predict the delay. Next, the principle of the King method is outlined in detail. Practical part includes a description of the two created simulations in program Matlab. The first application described simulates the King method principle. The second simulation is then tasked with creating 2D geometric space in which nodes and DNS servers are randomly generated. The result is dependence of RTT prediction accuracy on the number of randomly generated DNS servers. Based on this dependency the method King is compared with other methods in the end.

KEYWORDS

Method King, prediction RTT, system DNS, DNS server, simulation, latency

DVOŘÁK, P. *Lokalizace stanic v síti Internet pomocí systému pro překlad doménových jmen*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií. Ústav telekomunikací, 2009. 43 s. Bakalářská práce. Vedoucí práce: doc. Ing. Dan Komosný, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma Lokalizace stanic v síti Internet pomocí systému pro překlad doménových jmen jsem vypracoval samostatně pod vedením vedoucího semestrální 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 bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne

.....

(podpis autora)

PODĚKOVÁNÍ

Děkuji vedoucímu bakalářské práce doc. Ing. Danu Komosnému, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce.

V Brně dne

.....

(podpis autora)

OBSAH

Seznam obrázků	vi
Úvod	1
1 Lokalizace stanic v síti Internet	2
1.1 Význam lokalizace v síti Internet	2
1.2 Lokalizace stanic pomocí metod využívajících umělé souřadnicové systémy	3
1.2.1 Systém GNP (Global Network Positioning).....	3
1.2.2 Systém Vivaldi.....	4
1.2.3 Systém Lighthouses	5
1.2.4 Systém PCoord.....	5
1.3 Lokalizace stanic metodami využívajícími přímá měření.....	6
1.3.1 Systém IDMaps.....	6
1.3.2 Systém Meridian	7
1.3.3 Systém Internet Iso-Bar.....	8
2 Systém DNS	10
2.1 Motivace k zavedení a popis systému DNS	10
2.2 Struktura a syntaxe systému DNS	10
2.3 Autoritativní DNS servery.....	11
2.4 Sekundární DNS servery	12
2.5 DNS protokol a DNS záznamy.....	13
3 Metoda King	15
3.1 Popis metody	15
3.2 Příklad použití metody	15
4 Aplikace metody King	19
4.1 Simulace principu funkce metody King.....	19
4.2 Simulace závislosti přesnosti predikce RTT na počtu DNS serverů	24
4.3 Porovnání přesnosti predikce s dalšími metodami	26
5 Závěr	30

Literatura	31
Seznam symbolů, veličin a zkratk	33

SEZNAM OBRÁZKŮ

Obr. 1.1 Schéma dílčích částí měření RTT.	3
Obr. 1.2 Vytváření kružnic uzlem v systému Meridian.	8
Obr. 2.1 Naznačení hierarchické struktury DNS systému.....	10
Obr. 2.2 Formát paketu protokolu DNS.....	14
Obr. 3.1 První měření, které se provádí jako přípravné kvůli zpoždění způsobeném vyhledáváním DNS serveru hledané domény.	16
Obr. 3.2 Druhé měření, které již změří hodnotu zpoždění RTT1.....	17
Obr. 3.3 Poslední krok, kdy je změřena latence mezi stanicí A a jejím DNS serverem.	17
Obr. 4.1 Inicializace stanic a odeslání dotazu na stanici B příslušnému DNS serveru....	20
Obr. 4.2 Odeslání dotazu na kořenový DNS server, následná odpověď a odeslání dotazu na zjištěnou adresu.....	21
Obr. 4.3 Zaslání odpovědi s adresou od DNS serveru domény .cz a odeslání dotazu na tuto adresu.	22
Obr. 4.4 Odeslání patričních DNS serverů DNS serverem domény .vutbr.cz.	23
Obr. 4.5 Grafický výstup simulace závislosti přesnosti predikce RTT na počtu generovaných DNS serverů.....	24
Obr. 4.6 Příklad nedostatečného počtu barev.	25
Obr. 4.7 Graf závislosti přesnosti predikce RTT metody King na počtu náhodně vygenerovaných DNS serverů.....	27
Obr. 4.8 Graf závislosti přesnosti predikce RTT metody GNP na počtu vygenerovaných landmarků.	28
Obr. 4.9 Graf závislosti přesnosti predikce RTT systému Lighthouses na počtu náhodně generovaných majáků.	28
Obr. 4.10 Závislost přesnosti predikce RTT na počtu prvků pro všechny porovnávané metody.....	29

ÚVOD

Tato bakalářská práce se zabývá problémem lokalizace stanic v síti Internet a to konkrétně metodou, která se nazývá King [11]. Tato metoda slouží k predikci RTT (Round Trip Time Delay) za pomoci systému serverů DNS (Domain Name Server) [12]. RTT je často měřený parametr, který je definován jako obousměrné zpoždění mezi stanicí A (zdroj) a stanicí B (příjemce).

Tato práce má za úkol simulovat jednoduchou síť uzlů, mezi nimiž je náhodně generován určitý počet DNS serverů a následně zhodnotit přesnost odhadu hodnoty RTT mezi danými uzly. Po provedení simulace metody je úkolem změřeni a zpracování hodnot, které reprezentují predikci zpoždění. Hlavním cílem práce by mělo být zjištění, zda je predikce zpoždění touto metodou dostatečně přesná na to, aby byla v praxi použitelná, popřípadě jaké odchylky mohou v rámci měření vzniknout.

V první kapitole je postupně probrán princip lokalizace stanic v síti Internet. Je zde vysvětlen důležitý pojem, který je využíván při predikci, a to zpoždění RTT. Následně jsou probrány metody predikce RTT. Nejprve jsou vysvětleny metody využívající umělých souřadnicových systémů. Poté metody, které využívají fyzická měření. Tyto metody jsou dnes velmi perspektivní hlavně systém Meridian [13].

Ve druhé kapitole je vysvětlen systém pro nalezení doménových jmen zvaný DNS, který je pro metodu King stěžejní. Základem je samotný princip tohoto systému a následné vysvětlení důležitých charakteristik systému.

Třetí kapitola se zabývá samotnou metodou predikce zpoždění zvanou King. Je zde podrobně nastíněn princip činnosti této metody, řešení problémů, které by mohly nastat díky samotnému systému DNS a také specifika této metody.

Čtvrtá kapitola popisuje dosažené výsledky simulace. Nejprve je popsána simulace činnosti metody King. Další popisovaná simulace se zabývá závislostí počtu DNS serverů na přesnosti predikce RTT mezi stanicemi. Je zde popsán průběh vytváření samotné simulace a také výsledky v podobě grafických závislostí měřených parametrů. Dosažené výsledky jsou na konci této kapitoly shrnuty a porovnány s dosaženými výsledky vybraných metod.

V závěru je zhodnocení dosažených výsledků a shrnutí použitelnosti některých v praktické části porovnávaných metod.

1 LOKALIZACE STANIC V SÍTI INTERNET

1.1 Význam lokalizace v síti Internet

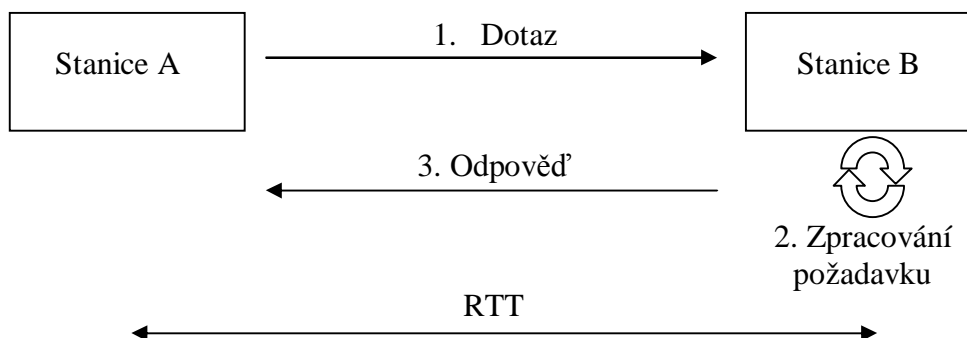
V dnešní době všudypřítomné výpočetní techniky, je Internet stále rozšířenější a využívanější. Tomuto faktu napomáhá zejména dostupné internetové připojení téměř ve všech lokalitách. S tím však také souvisí větší a na kvalitu náročnější datový provoz, který vyžaduje větší šířky pásma, kvalitnější kabely atd. Datová náročnost s nástupem výměnných sítí typu peer to peer, IPTV (Internet Protocol Television) a dalších datově náročných služeb značně stoupá. Většinu celkového datového toku tvoří právě tyto služby. Proto je třeba dbát na to, aby nebyly přenosové cesty zbytečně a příliš zatěžovány. K tomuto účelu je důležité obsáhnout fyzickou topologii sítě. Logická topologie sítě totiž nemusí kopírovat topologii fyzickou a některé uzly mohou být nadměrně zatěžovány. Aby nedocházelo k tomuto nadměrnému zatěžování uzlů, je vhodné využít metody sloužící k lokalizaci stanic v síti Internet. Slovo lokalizace se dá v tomto kontextu také vyjádřit jako predikce vzdálenosti dvou stanic. Vzdálenost mezi dvěma stanicemi v síti Internet však není možné chápat jako geografickou vzdálenost dvou uzlů. Nejčastějším parametrem určujícím vzdálenost dvou stanic je obousměrné zpoždění.

Obousměrné zpoždění označované jako RTT (angl. latency) je zpožděním, které nastává při komunikaci dvou uzlů v síti. Je zapříčiněno mnoha faktory. Některé tyto faktory nelze odstranit, např. fyzická vzdálenost mezi těmito uzly. Tato vzdálenost již sama o sobě zapříčiní určité zpoždění, což vyplývá ze samotných fyzikálních zákonů. Dalším důležitým faktorem je typ média, kterým jsou uzly k síti připojeny. Různé typy médií mají různou rychlost šíření, např. běžný koaxiální kabel má rychlost šíření 0,66 rychlosti světla [1]. Dalším důležitým faktorem, který však ovlivnit lze, je propustnost (šířka pásma) sítě, do které jsou uzly připojeny. Nízká propustnost sítě může zapříčinit vysokou hodnotu zpoždění. Jak již bylo řečeno v úvodu této kapitoly, logická topologie mezi uzly může být odlišná od topologie fyzické. Dalším faktorem ovlivňujícím zpoždění je tedy počet uzlů, přes které data proudí. Určitou úlohu zde hraje také interference na vedení.

Zpoždění, které je měřeno mezi dvěma uzly se nazývá jednosměrné. Toto zpoždění však nemá dostatečnou vypovídající hodnotu, jelikož při vzájemné komunikaci může jeden z uzlů vykazovat vyšší hodnotu zpoždění, které může být zapříčiněno zpracováním požadavku a následným odesláním odpovědi zpět. Proto je nejčastěji měřeným parametrem RTT. RTT je měřeno jako obousměrné zpoždění od odesílatele označeného jako stanice A k příjemci označenému stanice B. Do tohoto zpoždění je však také zahrnuto zpracování požadavku na straně stanice B a následné zpoždění vzniklé odesláním odpovědi ke stanici A. Funkce parametru RTT je lépe vidět na Obr. 1.1. Toto zpoždění je v konečném důsledku celkovým zpožděním mezi stanicí A a B.

Nástrojů, které slouží k měření RTT, je celá řada, jelikož je toto měření bráno jako jedno z nejpoužívanějších pro zjišťování stavu sítě. Jedny z nejznámějších aplikací, které toto zpoždění měří, jsou *ping* a *traceroute*. Aplikace *ping* využívá ke své činnosti ICMP (Internet Control Message Protocol) protokol definovaný specifikací RFC792 a

to konkrétně jeho zprávy *echo request* a *echo reply*. Zprávy tohoto protokolu slouží k signalizaci neobvyklých událostí v síti. Není vyžadováno jejich potvrzení, a jelikož jsou poměrně často rozesílány, velmi často jsou také zahazovány.



Obr. 1.1 Schéma dílčích částí měření RTT.

1.2 Lokalizace stanic pomocí metod využívajících umělé souřadnicové systémy

1.2.1 Systém GNP (Global Network Positioning)

Systém GNP [3][5] vytváří umělý systém souřadnic, které jsou tvořeny množinou tzv. orientačních bodů (landmark). V první části jsou nejdříve vypočítány souřadnice ve vybraném geometrickém prostoru pro všechny tyto orientační body. Tyto souřadnice poté slouží jako referenční pro ostatní uzly, které mají být do sítě připojeny. Pomocí v předešlém kroku vypočítaných souřadnic je následně možné připojit další koncový uzel, který získá své souřadnice dotazy na nejbližší orientační body.

Pro hlubší pochopení je třeba si Internet představit jako speciální geometrický prostor S . V tomto prostoru je určitý počet orientačních bodů $N = N_1, N_2, \dots, N_x$. Mezi všemi orientačními body je změřena pomocí ICMP zpráv latence. Hodnoty latence jsou poté vloženy do matice L , která obsahuje L_{ij} prvků, kde jednotlivé prvky představují latenci mezi N_i a N_j . Cílem je přiřadit orientačním bodům N takový bod c_i se souřadnicemi v eukleidovském prostoru S , aby byla odchylka od referenčních hodnot latence získaných pomocí měření a vypočtených hodnot co nejmenší. Toho je dosaženo použitím minimalizační funkce [5]:

$$E_1 = \sum_i \sum_j \varepsilon(L_{ij}, |c_i, c_j|), \quad (1.1)$$

kde ε je libovolná chybová funkce např. kvadratická odchylka. Výpočet souřadnic je poté proveden některým ze simplexních optimalizačních algoritmů pro nalezení extrému funkce např. Simplex Downhill [4].

Jakmile mají N přiřazeny své souřadnice, je jednoduše možné přidávat další klienty K . Pokud chce K zjistit své souřadnice, provede fyzická měření k několika jemu blízkým orientačním bodům N . V této fázi jsou orientační body pasivní a pouze odpovídají na příchozí ICMP zprávy. Poté jsou K na základě optimalizační metody přiřazeny souřadnice c_K . K predikci latence mezi klienty K_1 a K_2 pak jen stačí určit vzdálenost mezi jejich body c_{K1} a c_{K2} .

1.2.2 Systém Vivaldi

Algoritmus Vivaldi [6][5] funguje na podobném principu jako metoda GNP (viz kapitola 1.2.1). Je tvořen decentralizovaným souřadnicovým systémem. Tento algoritmus dokáže přiřadit stanicím souřadnice ve vybraném n -rozměrném souřadnicovém systému. Přiřazování je prováděno tak, aby se ve výsledku vzdálenost v souřadnicovém prostoru co nejvíce podobala vzdálenosti reálné. Toho je dosaženo minimalizací chybové funkce v našem případě kvadratické. Velikost kvadratické chyby v souřadnicovém systému je určena vztahem [6]:

$$E = \sum_i \sum_j (L_{ij} - \|x_i - x_j\|)^2, \quad (1.2)$$

kde L_{ij} je zpoždění mezi uzly i a j , x_i je souřadnice přiřazená uzlu i a x_j je souřadnice přiřazená uzlu j . Součástí algoritmu Vivaldi je poté jednoduchý centralizovaný algoritmus, který mění souřadnice jednotlivých uzlů do té doby, než dojde k minimalizaci chyby. Teoretickým základem tohoto algoritmu je potenciální energie pružin, což můžeme analogicky využít u uzlů v síti. Centralizovaný algoritmus poté vychází z Hookeova zákona [7] $\mathbf{F} = -k\mathbf{d}$, který popisuje přímou úměru natažení pružiny do vzdálenosti \mathbf{d} a na ni působící vratné síly \mathbf{F} . Analogicky je v souřadnicovém systému definován silový vektor \mathbf{F}_{ij} , který vyjadřuje sílu potřebnou k přeskoku mezi stanicemi i a j . Tato síla je pak definována [6]:

$$\mathbf{F}_{ij} = (L_{ij} - \|x_i - x_j\|) \times \mathbf{u}(x_i - x_j), \quad (1.3)$$

kde $L_{ij} - \|x_i - x_j\|$ představuje skok z pozice i do pozice j a vektor $\mathbf{u}(x_i - x_j)$ značí směr síly na uzlu i . Výsledná síla \mathbf{F}_i , která je silou uzlu i , je potom sumou jednotlivých sil \mathbf{F}_{ij} dalších uzlů [6]:

$$\mathbf{F}_i = \sum_{i \neq j} \mathbf{F}_{ij}, \quad (1.4)$$

Vypočtená síla působí na uzel v každé iteraci po dobu Δt a způsobí korekci souřadnic uzlu o hodnotu $\Delta t \mathbf{F}_i$. Výsledná souřadnice uzlu se poté rovná [6]:

$$x_i = x_i + \Delta t \mathbf{F}_i. \quad (1.5)$$

Počet iterací poté závisí na volbě požadované minimální chyby E .

Volbou časového kroku Δt je určena rychlost konvergence. Pokud je zvolen **konstantní** Δt , rozhodujícím faktorem konvergence je jeho velikost. Pokud je Δt zvolen velký, dochází k rychlejší konvergenci souřadnic. Na druhé straně však vzrůstá pravděpodobnost špatného odhadu a volba souřadnic je poté velmi nepřesná. Pokud je větší Δt nastaven i na ostatních uzlech, ke konvergenci vůbec nemusí dojít. Může se totiž stát, že dojde k oscilaci. Problémem konstantního Δt je také fakt, že pokud má některý uzel již přiřazeny souřadnice s určitou chybou, může se tato chyba u nových uzlů, které vypočítávají své souřadnice na základě těchto uzlů, ještě zvětšit. Čímž se

výrazně znelepší výsledná predikce.

Z tohoto důvodu je volen **proměnný** Δt . Proměnný Δt zajišťuje větší pravděpodobnost konvergence a menší pravděpodobnost zneužití, kdy při konstantním Δt a cílenému zavedení špatných souřadnic mohlo dojít k selhání predikce systému. Nový uzel má obvykle na začátku zvolen velký Δt , aby došlo k rychlejší hrubé konvergenci. Poté je zvolen kratší Δt , který zajistí zpřesnění konvergence. Vzorec algoritmu Vivaldi pro zvolení proměnného Δt je [6]:

$$\Delta t = c_c \times \frac{e_i}{e_i + e_j}, \quad (1.6)$$

kde c_c je stálý zlomek odhadované chyby, e_i je chyba bližšího uzlu i a e_j je chyba vzdálenějšího uzlu j , pomocí něhož určuje i svou polohu. Toto důmyslné řešení zajistí, že pokud uzel s přesnějšími souřadnicemi bude určovat svou polohu pomocí uzlu s méně přesnějšími souřadnicemi, bude zvolen menší proměnný Δt . Naopak pokud bude mít uzel méně přesnější souřadnice a bude určovat svou polohu pomocí uzlu s přesnějšími souřadnicemi, zvolí se větší Δt , aby došlo k rychlejší konvergenci.

1.2.3 Systém Lighthouses

Systém Lighthouses [5] funguje také na podobném principu jako systém GNP (viz kapitola 1.2.1). Ke své činnosti využívá orientační body, které se nazývají majáky (lighthouses). Je zjištěna vzdálenost mezi majáky a následně je dle změřené vzdálenosti každému majáku přiřazena souřadnice, která je vypočítána určitými jednoduššími algoritmy. Je zde lépe vyřešeno přiřazování souřadnic novým uzlům. Aby nový uzel získal souřadnice v systému, stačí mu zjistit vzdálenosti a souřadnice několika majáků, typická hodnota počtu těchto majáků je 3 [17]. Z těchto hodnot jsou určeny souřadnice nového uzlu a zároveň se tento nový uzel stává novým majákem v systému. Tento maják uchovává informace o majácích, na které se dotazoval při přiřazování souřadnic. Díky této vlastnosti je možné dále přidávat nové uzly, aniž by bylo třeba zjišťovat souřadnice mnoha majáků.

1.2.4 Systém PCoord

Tento systém [19][5] ke své činnosti využívá taktéž umělý souřadnicový systém. Algoritmus systému PCoord využívá několika mechanismů jako např. váhové ztrátové funkce, optimální úpravy souřadnic uzlů pro různé skupiny vzorků. Pro aktualizaci se používá vždy konstantní počet vzorků.

Váhová ztrátová funkce slouží k tomu, aby referenční uzly s velkou chybovostí příliš neovlivňovaly výsledek. Toho je docíleno přiřazováním váhy na základě relativní chybovosti pro souřadnice každého referenčního uzlu. Díky tomuto postupu je dosaženo toho, že uzly s menší chybovostí budou mít větší vliv na výsledek predikce než uzly s větší chybovostí.

Systém PCoord pracuje tak, že si nejprve uzel N_i zvolí nezávisle na ostatních uzlech M referenčních bodů. Na začátku se přitom předpokládá, že každý uzel má ve své blízkosti K sousedů. Při počáteční iteraci si tedy každý uzel náhodně vybere referenční uzly z těchto K sousedů. Tyto referenční uzly si označme jako N_j . V dalších iteracích jsou poté vybrané referenční uzly zpracovávány najednou jako celek.

U každého referenčního uzlu jsou postupně zjištěny souřadnice d_j , zpoždění Z_j , plovoucí průměr chyby predikce $e_{p,j} \in \langle 0, 1 \rangle$ a plovoucí průměr zbytkové chyby optimalizace e_f . Zbytková chyba optimalizace e_f daného uzlu nám určuje, do jaké míry je daný soubor referenčních uzlů soudržný. Pokud je tato chyba velká, je zřejmé, že uzel vykazuje vyšší chybovost, a proto je vhodné takovému uzlu přiřadit menší váhu. Naopak s malou chybou je možné docílit pravděpodobně (nemusí tomu vždy tak být) lepší minimalizace. Aby bylo možné hodnotit soudržnost, je nutné zavést poměr [5]

$$\rho = \min\left(\frac{e_f}{e_{f,n}}, 1\right), \rho \in \langle 0, 1 \rangle, \quad (1.7)$$

kde e_f je zbytková chyba předešlé iterace uzlu, $e_{f,n}$ je zbytková chyba právě prováděné iterace uzlu. Tento poměr nám poté určí míru posuvu souřadnic, aby mohlo dojít k rychlejší konvergenci.

Aby uzel nebyl příliš ovlivňován velkou chybou, uvažuje se také záznam uzlu N_i . Na základě zjištěné chybovosti referenčních uzlů je jim následně přiřazena váha [5]

$$\omega_j = \frac{a_j^2}{\sum_{k=1}^M a_k^2}, \quad (1.8)$$

kde $a_k = 1 - e_{p,j} + \tau$, kde τ je konstanta, která se blíží nule. Tato konstanta zapříčiní přiřazení malé váhy uzlům s velkou chybou.

V dalším kroku jsou vypočítány nové souřadnice d_n . Výpočet je možné provést libovolnou optimalizační metodou pro chybovou funkci. To znamená, že výběrem vhodné optimalizační funkce je možné docílit ještě přesnější predikce. Tvar chybové funkce je následující

$$\varepsilon = \omega_i (Z_i - \|d_j - d\|)^2 + \sum_{j=1}^M \omega_j (Z_j - \|d_n - d_j\|)^2. \quad (1.9)$$

1.3 Lokalizace stanic metodami využívajícími přímá měření

1.3.1 Systém IDMaps

Tato metoda [5][8] využívá podobnou strukturu jako metoda King (viz kapitola 3), která ke své činnosti vyžaduje síť DNS serverů. Namísto DNS sítě je zde síť tzv. tracerů. Pomocí takovýchto tracerů je poté predikována hodnota latence mezi stanicemi. Tato metoda nevyžaduje takový počet přímých měření jako u metod využívajících umělé souřadnicové systémy. Namísto toho jsou přímá měření delegována na určitý počet speciálních uzlů, v tomto případě to jsou tracery. Predikovaná latence je poté součtem latence změřené mezi stanicí A a jejím nejbližším tracerem, latence mezi stanicí B a jejím nejbližším tracerem a nakonec latence mezi určenými tracery. Predikovaná hodnota je poté uložena do databáze tzv. HOPS (Host Proximity Service) serverů, které slouží k distribuci predikované hodnoty latence ke koncovému uzlu.

Vzdálenost mezi dvěma tracery a mezi tracerem a adresním prefixem AP se nazývá VP (Virtual Links). Problém však nastává v optimálním rozmístění tracerů. Toto rozmístění by mělo být takové, aby byl v každém adresním prefixu AP obsažen alespoň

jeden tracer, kde AP je množina IP adres uzlů, které se v rámci Internetu jeví, jako jeden uzel tzn. jednotlivé uzly mají téměř shodnou latenci. AP je v síti Internet však velký počet, proto je obtížné rozmístit potřebný počet tracerů. Možným řešením je umístění tracerů do autonomních systémů AS, což je zjednodušeně řečeno síť, která může obsahovat další síť a je spravována určitou organizací např. poskytovatelé připojení k internetu. Ideální řešení to však také není, jelikož se může stát, že uzel může být od druhého uzlu geograficky poměrně velmi blízko, avšak patří do jiného AS, což má za následek přiřazení odlišného traceru a z toho plynoucí méně přesnou predikci latence. Pro určení ideální pozice tracerů se používají dva algoritmy k-HST (k-hierarchically well-separated trees) a minimum K-center více o těchto metodách viz [9][10]. Tyto algoritmy se využívají i v obvyklé praxi např. pro ideální rozmístění ambulancí atd. K nejpřesnějším výsledkům predikce se lze dostat minimalizováním vzdálenosti AP a jeho nejbližším tracerem. Jelikož se Internet dynamicky mění, nejlepším řešením je umístění tracerů k internetovým poskytovatelům, kteří se však o tracer musí průběžně starat. To však nikdy není zaručeno. Také kvůli tomuto důvodu se tato metoda příliš nepoužívá.

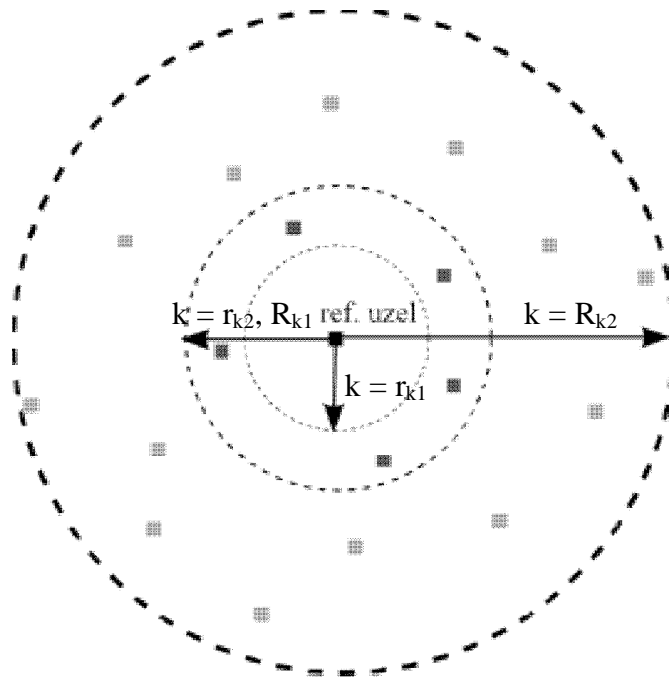
1.3.2 Systém Meridian

Tento systém [13][5] je vytvořen zejména pro překryvné síť. V dnešní době je to asi nejperspektivnější řešení predikce latence. K predikci je sice použito fyzické měření latence, ale díky systému, založenému na kružnicích, je počet takovýchto měření dostatečně malý a nezatěžuje tak zbytečně síť. S úspěchem se tento systém dá využít pro optimalizaci vyhledávání opravdu nejbližšího např. webového serveru ke klientovi, což má za následek významné snížení odezvy. Toho lze využít také u dnes hojně rozšířené služby hraní online her, kde lze za pomoci této metody najít nejbližší server k uživateli.

Principem tohoto systému predikce latence je rozdělení prostoru na kružnice viz Obr. 1.2. Zvolený uzel kolem sebe rozdělí prostor na kružnice, jejichž poloměr kvadraticky roste. V každé takovéto kružnici je fixní počet uzlů. Tím je zajištěno, že okolní uzly mají nejvyšší hustotu. Uzel nacházející se v určitém mezikruží má informace o ostatních nejbližších uzlech umístěných v této kružnici včetně latence k nim. Má také latence k několika uzlům ve vnějších prstencích, aby bylo možné předání dotazu o úroveň výše. Počet uzlů v mezikruží je také omezen. Pokud je zvolen větší počet uzlů, pak uzel, na který je posílán dotaz, bude mít větší počet informací o blízkých uzlech a tím pádem má možnost zvolit vhodnější uzel, kterému předá dotaz. Na druhou stranu je však nutné udržovat větší databázi, větší šířku pásma a také se zvýší vlastní režie.

Na začátku libovolný uzel sítě Meridian změří vzdálenost d_i k uzlu i , který je nejbližší k uzlu, ke kterému je predikováno zpoždění a podle této vzdálenosti umístí tento uzel i do příslušné kružnice k tak, aby platila podmínka $r_k < i < R_k$, kde r_k je nejmenší možný rádius kružnice k a R_k je největší možný rádius kružnice k . Výhoda tohoto systému je ta, že není potřeba znát fixní orientační body nebo distribuované souřadnice, jako je tomu např. u systému GNP (viz kapitola 1.2.1). Uzel i poté rozešle dotaz všem uzlům v mezikruží k a sousedním uzlům, které jsou vzdáleny $k + 1$ nebo $k - 1$ a jejich vzdálenost je mezi $\frac{d}{2}$ a $\frac{3d}{2}$ [13] původní měřené vzdálenosti d_i . Poté každý takovýto uzel odešle dotaz na latenci k uzlu i . Změřenou latenci poté odešle jako

odpověď. Ten uzel, který má latenci nejvíce podobnou uzlu i , je poté delegován jako nejbližší a latence k němu slouží jako predikce latence k cílovému uzlu. Zároveň je do této metody zaveden činitel β . Tento činitel slouží k odstranění uzlů s příliš odlišnou latencí. Činitelem je nastavena určitá prahová hodnota, nad níž jsou uzly považovány za neadekvátní. To vede k exponenciálnímu přibližování k nejbližšímu uzlu. Z toho vyplývá, že tímto způsobem lze výrazně omezit počet kroků měření.



Obr. 1.2 Vytváření kružnic uzlem v systému Meridian.

1.3.3 Systém Internet Iso-Bar

Tato metoda [18] je velmi podobná metodě IDMaps (viz kapitola 1.3.1), navíc však vylepšuje některé její nedostatky jako např. přesnější predikce uzlů, jež se nacházejí v jednom uskupení, které je spravováno jedním prvkem. Zlepšuje také celkovou přesnost predikce. Internet Iso-Bar navíc nevyžaduje speciální prvky, které by musel zajistit poskytovatel, jako tomu je u systému IDMaps.

Systém využívá určitých bodů zvaných landmarky, které slouží k prvotnímu přiřazení uzlů do určitých shluků. Tyto landmarky však slouží pouze k tomuto rozřazení, nepodílí se nijak na predikci zpoždění mezi uzly jako tomu je u metody GNP (viz kapitola 1.2.1). Jako landmark může vystupovat jakýkoliv koncový uzel popřípadě vnější server, který podporuje měření.

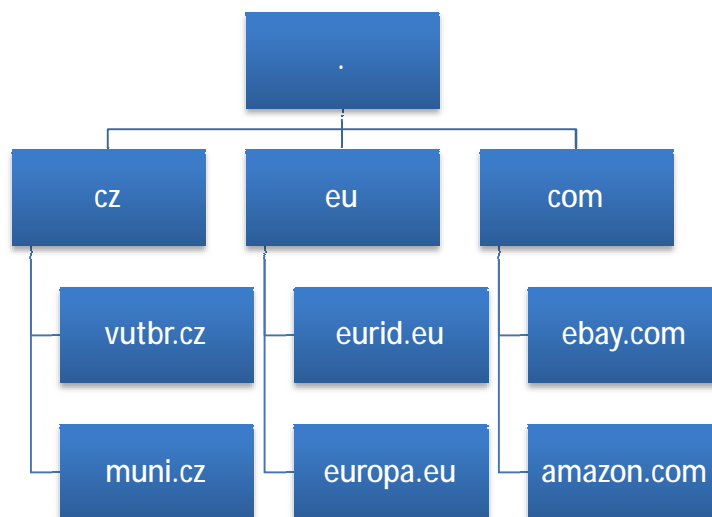
Všechny uzly i si na začátku změří vzdálenost mezi nimi a nejbližším landmarkem. Ze všech naměřených hodnot je následně vytvořen vzdálenostní vektor V_i . Tento vektor je m -dimensionální, kde m je počet uzlů měřících vzdálenost k jednomu landmarku.

Pomocí daných vektorů jsou následně vytvořeny shluky uzlů. Poloměr těchto shluků je poté upravován různými aproximačními metodami [20], aby byl zachován minimální možný poloměr. S koncových uzlů v jednom shluku je v dalším kroku vybrán centrální uzel zvaný monitor. Tento monitor periodicky měří zpoždění mezi uzly v daném shluku a mezi ostatními monitory. Toto měření poté průměruje. Tyto monitory tedy u systému Internet Iso-Bar slouží k měření predikce zpoždění. Predikce zpoždění mezi dvěma uzly i a j , které leží ve dvou různých shlucích, je rovna vzdálenosti mezi monitory M_i a M_j .

2 SYSTÉM DNS

2.1 Motivace k zavedení a popis systému DNS

Identifikace stanic v síti probíhá na základě jedinečného označení a to IP adresy. IP adresa je 32-bitová, přičemž se skládá ze čtyř číselných bloků po 8 bitech oddělených tečkou. Pro běžného uživatele však zapamatování takovéto adresy není příliš pohodlné. Proto byl zaveden systém, který umožní transformaci IP adresy na pojmenování (v našem případě doménové jméno), které je snáze zapamatovatelné. Tento systém se nazývá DNS [12][15]. Pro realizaci takovéto služby však vyvstává problém, jak udržet, dnes stále se rozrůstající, databázi doménových jmen aktuální. Proto se pro realizaci DNS systému použila hierarchická struktura DNS serverů viz Obr. 2.1.



Obr. 2.1 Naznačení hierarchické struktury DNS systému.

2.2 Struktura a syntaxe systému DNS

Aby bylo možné systém realizovat, byla použita, jak je napsáno výše, hierarchická struktura serverů. Toho je docíleno tak, že jsou mezi servery rozděleny převodní tabulky a servery zároveň mají informace o tom, který server a jaké informace má. Jedna IP adresa může mít přiřazeno více domén, to znamená, že IP adresa nemá pevně přiřazený doménový název, tím pádem se doménový název může měnit. Úkolem DNS databáze je poté zaznamenat příslušná doménová jména k určité IP adrese. Tato databáze je celosvětově distribuovaná.

Jako hierarchická struktura DNS bylo zvoleno rozdělení celosvětové sítě Internet na domény, které se dále mohou dělit na tzv. subdomény. Nejvyšší doménou je tzv. root doména, která má označení „.“. Doména v hierarchické struktuře pod root doménou se označuje doména 1. řádu a je označovaná jako TLD (Top Level Domain). Většinou označuje stát či území, ve kterém se příslušná doména nachází např. *cz*, *eu*, *sk* apod. nebo označuje účel této domény např. *edu* (education - vzdělání). Poté následuje doména 2. řádu označovaná SLD (Second Level Domain) např. *vutbr.cz*. Doména 2. řádu může mít další subdomény např. *feec.vutbr.cz*. Z toho plyne obecná struktura doménového jména:

...doména3.řádu.doména2.řádu.doména1.řádu.

Existují také tzv. **reverzní domény**, což není nic jiného, než překlad doménového jména na IP adresu. Tento překlad je možné si představit tak, že je známa IP adresa např. *142.145.78.17* a je třeba zjistit její doménové jméno. K tomuto účelu slouží pseudodoména *in.addr.arpa*. Za touto doménou jsou pak jednotlivé subdomény tvořeny čísly IP adresy, které se však zapisují v obráceném pořadí např. pro IP adresu *142.145.78.17* je to *17.78.145.142.in.addr.arpa*.

Doménové jméno je čteno zprava doleva, kdy jednotlivé domény jsou od sebe odděleny tečkou. Koncová tečka se může vynechat v aplikačních programech jako je internetový prohlížeč, neměla by však být vynechána v databázi doménových jmen.

2.3 Autoritativní DNS servery

Každá doména musí mít přiřazen autoritativní (name) DNS server. Těchto serverů může být pro jednu doménu i více. Obsahují důležité informace o své spravované doméně a znají adresy DNS serverů subdomén přidružených k této doméně. Pokud spravuje DNS server určitou doménu, je pro tuto doménu automaticky brán jako autoritativní server DNS. Informace o doméně, které jsou uchovány v autoritativním DNS serveru jsou závazné pro celý systém DNS a jsou brány za relevantní.

Každá doména může mít různé druhy autoritativních DNS serverů. Primární autoritativní DNS server má svou databázi o doméně uloženou na fyzickém médiu (HDD). Existují však také sekundární DNS servery, které si v určitém časovém intervalu kopírují databázi z primárního autoritativního DNS serveru. DNS servery spravující root doménu jsou brány jako kořenové servery. Tyto servery jsou spravovány organizací ICANN (Internet Corporation for Assigned Names and Numbers). Jejich názvy a adresy jsou všeobecně známé a všechny autoritativní DNS servery určené pro překlad doménových jmen musí jejich adresy znát. Tyto servery obsahují informace o všech doménách 1. řádu (*cz*, *com*, *eu* apod.) a také adresu jejich autoritativních DNS serverů.

Jednoduchý příklad činnosti při dotazu na doménové jméno může být následující: Je vyslán dotaz na doménu *www.feec.vutbr.cz*. Root DNS server ve své databázi nemá uloženu IP adresu pro tuto doménu, má však adresy autoritativních DNS serverů pro doménu 1. řádu *.cz*. Odešle tedy odpověď s adresami těchto serverů. Po zvolení jednoho z autoritativních DNS serverů domény *.cz* je na tento server vyslán opět dotaz na do-

ménu *www.feec.vutbr.cz*. Autoritativní server však ve své databázi také nemá příslušný záznam. Jako odpověď tedy zašle adresy autoritativních DNS serverů pro doménu 2. řádu v našem případě *vutbr.cz*. Po dotazu na jeden z těchto serverů opět obdržíme odpověď, že příslušný server danou doménu ve své databázi nemá, ale zná autoritativní DNS servery pro doménu 3. řádu *feec.vutbr.cz*. V dalším kroku po odeslání dotazu na příslušný DNS server je nalezen daný záznam v databázi a odeslána odpověď se záznamem.

Funkci dotazování k DNS serverům na straně stanice obsluhuje klient zvaný resolver. Obvykle je realizován množinou knihovnických funkcí, které se vybírají na základě typu tázající se aplikace. Tyto knihovní funkce poté vytvoří dotaz, který je poslán na server a očekávají odpověď. Pokud není do určitého časového okamžiku odpověď přijata, resolver uživateli odpoví, že příslušný záznam nebyl nalezen.

2.4 Sekundární DNS servery

Tyto servery jsou neautoritativní a vznikly za účelem zjednodušení a zrychlení DNS systému. Z podkapitoly 2.3 je patrné, že záznamy v root DNS serverech nebudou měněny příliš často, což platí také pro záznamy některých dalších autoritativních DNS serverů. Je tedy zbytečné zatěžovat autoritativní DNS servery celým DNS cyklem stále dokola. Do systému DNS jsou tedy zavedeny tzv. **cachovací** servery, které mají za úkol provést celý DNS cyklus a odeslat uživateli výsledek. Tyto servery jsou většinou součástí autoritativních DNS serverů. Cachovací servery si ukládají do své paměti všechny provedené překlady. Neukládají si však jen výsledek překladu, ale také všechny výsledky jednotlivých provedených iterací tzn. jednotlivé autoritativní DNS servery na cestě k cíli. Důležitým parametrem cachovacích serverů je hodnota TTL (Time to Live), což je doba uchování cache záznamu. Nastavení této hodnoty je odesláno spolu s každým DNS záznamem.

Kromě výše popsaných typů serverů se v praxi využívají také **přeposílací** a **slave** servery. U těchto serverů nezáleží na jejich typu, mohou být jak sekundární tak primární. Přeposílací servery jsou využívány zejména tam, kde dochází k většímu vytížení sítě nebo je zde pomalejší linka. Pokud je např. odeslán na DNS server dotaz a tento server na něj neumí odpovědět, musí postupně rozesílat dotazy nejdříve na root DNS servery, poté na servery, jejichž adresa je odeslána tázanými root DNS servery atd. Proto je zaveden přeposílací server, což je server, který odešle rekurzivní dotaz DNS serveru, který tento dotaz zpracuje a odešle výsledek. Většinou se tedy jedná o cache server. Tento server je zpravidla umístěn v rychlé síti. Jakmile obdrží od přeposílacího serveru dotaz, sám postupně nalezne v systému DNS odpověď, kterou následně přepošle tázajícímu se přeposílacímu serveru. Pokud cache server po určitou dobu přeposílacímu serveru neodpovídá, přeposílací server již musí nalézt příslušný záznam opět sám. Pokud potřebujeme, ať už z důvodu nevytěžování sítě nebo jakéhokoliv jiného zajistit, aby se přeposílací server sám nedotazoval root DNS serverů, označíme ještě takovýto server jako **slave**. Takto označený přeposílací server přijímá pouze odpovědi. Tohoto je zejména využito v uzavřených sítích, kde není možné přímo kontaktovat root DNS servery.

2.5 DNS protokol a DNS záznamy

Informace, které DNS servery mají, se uchovávají v paměti serveru ve formě určitého záznamu, který můžeme nazvat jako tzv. RR věta [12], což je zdrojová věta. Pokud klient vyžaduje záznam od DNS serveru, je to právě tento záznam. Formát je vždy stejný, může se však měnit typ RR věty.

Pole jednotlivých RR vět jsou tvořena následujícími položkami:

- **NAME** – doménové jméno.
- **TYPE** – typ RR věty.
- **CLASS** – třída věty.
- **TTL** – 32-bitové číslo udávající dobu, po kterou může být RR věta uchována v paměti cache serveru.
- **RDLLENGTH** – 16-bitové číslo deklarující délku pole RDATA.
- **RDATA** – vlastní data, jejich formát je určen typem a třídou RR vět.

Nejdůležitější typy RR vět:

- **A** – tento typ RR věty obsahuje IP adresu, pole RDATA má formát 32-bitová IP adresa.
- **NS** – tento záznam obsahuje doménové jméno name serveru autoritativního pro danou doménu.
- **CNAME** – DNS záznam může mít také svůj alias, důvod je ten, že při změně záznamu např. IP adresy se nemusí přepisovat všechny záznamy jednotlivě. Ty, jež jsou aliasem pro daný záznam, se přepíší sami.

Komunikace mezi DNS servery probíhá pomocí UDP (User Datagram Protocol) i TCP (Transmission Control Protocol) protokolu ve formě dotaz-odpověď. Protokol DNS, jeho základní struktura je vidět na Obr. 2.2, pracuje na aplikační vrstvě a ke své činnosti využívá zpravidla port 53. Pro přenos UDP paketů mezi DNS servery využívá protokol UDP délku paketu 512 B, pokud je však přenášená informace větší, v paketu se přenesou část informace nepřesahující 512 B a v záhlaví se nastaví bit TC, který značí, že zpráva je delší než 512 B. Server přijme tuto zprávu a na základě nastaveného bitu TC ví, že si má vyžádat další zprávu, která je však již přenášena pomocí protokolu TCP, u něhož je možné přenášet větší objem dat. Paket protokolu DNS má stejný formát pro dotaz i pro odpověď. Tento paket musí povinně obsahovat záhlaví. V prvních 2 bytech je obsažen identifikátor, který slouží k jednoznačné identifikaci dotazu a odpovědi. Tento identifikátor je vygenerován klientem. Server, který odešle odpověď, musí jako identifikátor uvést klientem vygenerované číslo. Klient totiž po dotazu nečeká na odpověď, ale může odeslat dotaz nový. Jakmile je doručena odpověď, identifikátor dotazu je porovnán s identifikátorem uvedeným v odpovědi a pokud se sobě identifikátory rovnají, odpověď se přiřadí k příslušnému dotazu.

Záhlaví
Dotazy
Odpovědi
Autoritativní DNS servery
Doplňující informace

Obr. 2.2 Formát paketu protokolu DNS.

Za účelem co největší úspory datového toku se běžně v systému DNS používá komprese DNS paketu. Používá se zejména v případě, že se v DNS paketech vyskytují stejná doménová jména nebo jejich části. Komprese se provádí tak, že opakující se doménové jméno je zaznamenáno pouze jednou a v případě, že se objeví znovu, je na místo tohoto jména vložen ukazatel na první záznam s tímto doménovým jménem.

3 METODA KING

3.1 Popis metody

Metoda King [11] je poměrně perspektivní metodou predikce zpoždění mezi dvěma stanicemi. Vyplývá to hlavně z faktu, že ke své činnosti používá systém DNS. Některé jiné metody predikce ke své činnosti vyžadují speciální uzly, které musí být vytvořeny navíc. Například systém IDMaps (viz kapitola 1.3.1) potřebuje ke své činnosti tzv. tracery. Tyto tracery jsou obdobou DNS serverů, mezi kterými se predikuje zpoždění. Z toho plyne, že pro funkčnost metody je třeba do sítě zavést nové prvky. Naproti tomu infrastruktura DNS systému je již celosvětově rozšířená a jelikož je pro funkčnost celé sítě Internet stěžejním prvkem, je zaručena relativně malá vzdálenost DNS serveru ke každé stanici v Internetu. Výhodou této metody je nízká datová náročnost, jelikož je k predikci potřeba několika málo paketů, které jsou využívány při běžném DNS dotazování. Také nevyžaduje na rozdíl od některých metod aktivní kooperaci koncových hostů.

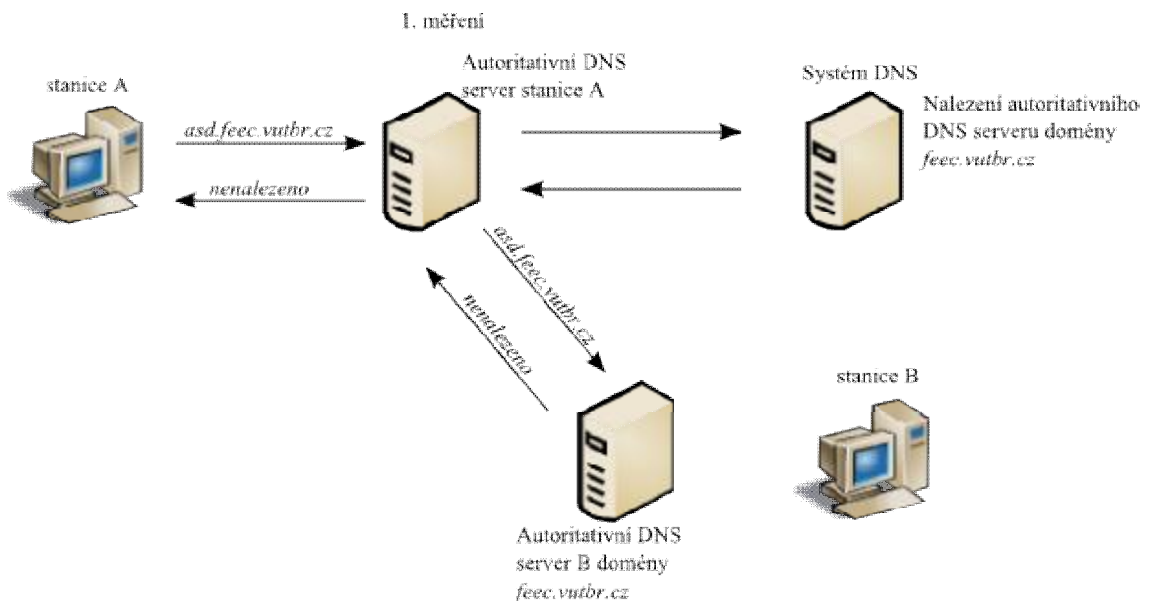
Princip této metody spočívá v tom, že namísto zpoždění mezi stanicemi je měřeno zpoždění mezi ke stanicím nejbližšími DNS servery. Velmi důležitým prvkem této metody je **rekurzivní dotaz**. Pojmeme rekurzivní rozumíme, že DNS server přijme dotaz a pokud není autoritativním serverem pro hledanou doménu, spustí algoritmus pro hledání doménového jména, tzn. začne u kořenových serverů a postupuje k nižším úrovním DNS hierarchie, až je příslušný DNS server nalezen. Klientovi odešle až samotný výsledek. Autoritativní servery pro danou doménu zpravidla rekurzivní dotazy nepodporují, mají na starosti databázi své domény. Provádění rekurzivních dotazů má tedy na starosti cachovací server. Jelikož predikce zpoždění pomocí metody King pochopitelně nelze měřit přímo, musí se měření provádět pomocí DNS dotazů.

3.2 Příklad použití metody

Pro pochopení této metody je uveden příklad. Predikce zpoždění mezi klientem A a klientem B se skládá ze tří částí. Nejdříve je potřeba zjistit DNS server, který spravuje doménu, v níž se nachází stanice B. Toho je docíleno odesláním rekurzivního dotazu klientem A viz Obr. 3.1. Tento dotaz by měl jako součást doménového jména obsahovat libovolné znaky nejlépe alfanumerické např. *asd.feec.vutbr.cz*, aby byl DNS server A donucen k vyhledávání serveru B, který spravuje doménu *feec.vutbr.cz*, pod kterou patří stanice B. Je totiž nepravděpodobné, že již server A takové doménové jméno hledal. Dotaz je tedy odeslán na DNS server A spravující doménu, do které náleží stanice A. Pokud DNS server A podporuje rekurzivní dotazy (ve většině případů tomu tak je [11]), spustí se algoritmus pro hledání a najde se nejbližší DNS server ke stanici B. Jelikož jsou zadány jako subdoména domény, pod kterou patří stanice B, libovolné alfanumerické znaky, obvykle DNS server B nezná odpověď. Predikce zpoždění, je zde velmi zkrácena vyhledáváním serveru B, proto je toto měření prováděno jako přípravné a slouží k vyhledání DNS serveru, pod který patří stanice B.

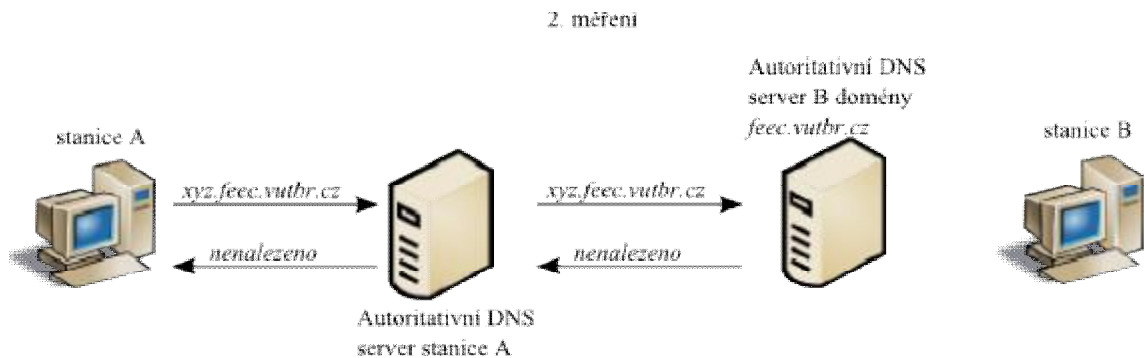
Druhá část měření je již prováděna mezi DNS servery A a B viz Obr. 3.2. Je vyslán nový rekurzivní dotaz, který by měl rovněž obsahovat libovolné alfanumerické znaky, které jsou však jiné než v prvním případě např. *xyz.feec.vutbr.cz*. Jiné proto, že cachovací server DNS serveru A má již v paměti uložený výsledek k dotazu na *asd.feec.vutbr.cz*, nezeptal by se tedy serveru B, ale zaslal by uložený výsledek. Jelikož DNS server A již zná adresu serveru B, probíhá komunikace přímo mezi nimi. Výsledný čas RTT_1 je pak roven obousměrnému zpoždění mezi těmito servery a době zpracování na straně DNS serveru B.

Posledním krokem je zaslání libovolného dotazu (nemusí být rekurzivní) na DNS server A viz Obr. 3.3. Tímto měřením je zjištěna doba RTT_2 , což je vlastně zpoždění mezi stanicí A a DNS serverem A. Výsledná predikovaná hodnota zpoždění se poté rovná $RTT = RTT_1 - RTT_2$.

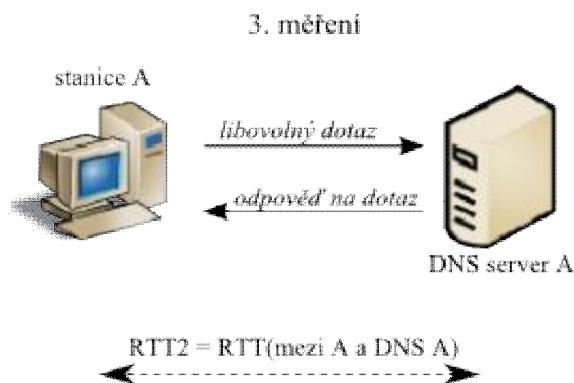


Obr. 3.1 První měření, které se provádí jako přípravné kvůli zpoždění způsobeném vyhledáváním DNS serveru hledané domény.

Pro doménu koncové stanice, ke které je predikováno zpoždění, může existovat více autoritativních DNS serverů. Zde vyvstává otázka, jak metoda King vybere ten teoreticky nejbližší DNS server. Metoda King pracuje jak s IP adresou koncové stanice, tak s doménovým jménem této stanice. Pokud je zadáno doménové jméno, metoda vybírá z nalezených autoritativních DNS serverů ty, které mají nejvíce totožných prvků sufixu jako koncová stanice. Jak je vidět v tab. 1, pro stanici B je zadáno doménové jméno *staniceb.feec.vutbr.cz* a jsou nalezeny 4 autoritativní DNS servery. Pokud je zadána IP adresa, jsou autoritativní servery nalezeny pomocí reverzního doménového jména *in-addr.arpa*. IP adresa stanice B je *147.229.72.54*, její reverzní doménové jméno je *54.72.229.147.in-addr.arpa*. Reverzí je nalezeno doménové jméno stanice, které je *staniceb.feec.vutbr.cz*. Doména obsahující toto jméno je *72.229.147.in-addr.arpa*. Doménové jméno korespondující s touto adresou je *feec.vutbr.cz*, která je spravována 4 autoritativními DNS servery.



Obr. 3.2 Druhé měření, které již změří hodnotu zpoždění RTT1.



Obr. 3.3 Poslední krok, kdy je změřena latence mezi stanicí A a jejím DNS serverem.

Jakmile jsou pro metodu King zjištěny autoritativní DNS servery ke stanici B, přichází na řadu výběr vhodného DNS serveru, který probíhá na základě dvou kritérií. Jako první jsou vybrány autoritativní DNS servery, jejichž sufix se nejvíce shoduje s tím, který má koncová stanice. Jakmile jsou přípustné servery odseparovány, aplikuje se na ně druhé kritérium. Z odseparovaných autoritativních DNS serverů je vybrán server s nejvíce se shodující společnou částí prefixu IP adresy koncové stanice. První pravidlo slouží k odseparování autoritativních serverů, které překládají i jiná doménová jména a mohou být od stanice více vzdáleny. Druhé kritérium již jen vybere autoritativní DNS server, který by měl teoreticky patřit do stejné sítě. Po aplikování těchto dvou kritérií na autoritativní DNS servery uvedené v tab. 1 jsou odseparovány autoritativní servery *kos.feec.vutbr.cz* a *gate.feec.vutbr.cz*. Dále je zkoumána jejich IP adresa, následně je vybrán server s IP adresou *147.229.72.10*, což je *kos.feec.vutbr.cz*, jelikož má nejvíce společný prefix IP adresy s prefixem IP adresy stanice B *147.229.72.54*.

Tab. 1 Příklad nalezených autoritativních DNS serverů.

Typ zdroje	Doménové jméno	IP adresa
klient	staniceb.feec.vutbr.cz	147.229.72.54
Autoritativní DNS server	kazi.fit.vutbr.cz	147.229.8.12
Autoritativní DNS server	kos.feec.vutbr.cz	147.229.72.10
Autoritativní DNS server	rhino.cis.vutbr.cz	147.229.3.10
Autoritativní DNS server	gate.feec.vutbr.cz	147.229.71.10

4 APLIKACE METODY KING

V rámci praktické části bakalářské práce byla vytvořena simulace naznačující princip funkce metody King (kapitola 4.1). V této simulaci jsou postupně rozebrány jednotlivé iterace metody King. Dále byla vytvořena další simulace, která má za úkol zhodnotit, jak moc se hodnota predikovaného RTT přibližuje reálnému RTT v závislosti na počtu DNS serverů, které mohou být k predikci využívány.

Simulace metody King jsou provedeny v programu Matlab, který je určen převážně k matematicko-fyzikálním výpočtům. Tento program má velmi rozsáhlé možnosti využití a je proto hojně využíván k vysokoškolským studiím a k všemožným simulacím. V programu Matlab je s proměnnými pracováno výhradně jako s maticemi. Jelikož je Matlab velmi komplexní nástroj, obsahuje mnoho tzv. toolboxů, které bývají používány u rozličných problematik.

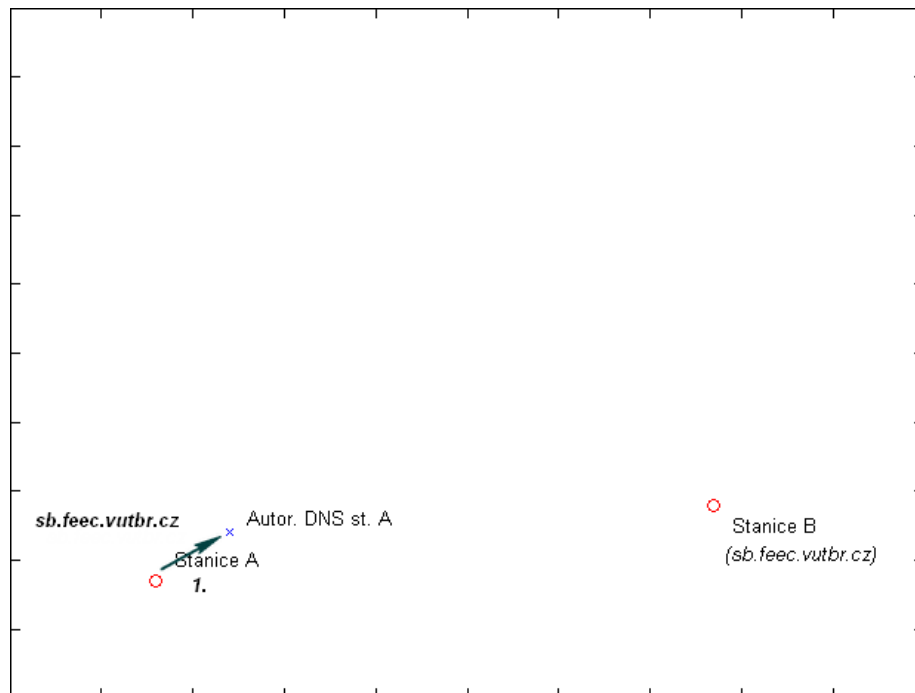
4.1 Simulace principu funkce metody King

Prvně popisovaná simulace metody King byla vytvořena v programu Matlab za účelem bližšího pochopení činnosti metody King. Tato simulace se nazývá *simulace.m*. Pro její spuštění je třeba v programu Matlab nastavit domovský adresář takový, kde je tento soubor umístěn. Poté již stačí zadat název do hlavního okna programu. U simulace je nastaveno krokování, pro jednotlivé iterace je tedy potřeba postupně stisknout libovolnou klávesu. Pracuje s reálnými doménovými názvy a také s reálnými IP adresami jednotlivých prvků, které jsou v simulaci používány. Z principu DNS systému vyplývá, že určitá doména je spravována zpravidla více DNS servery, tím pádem je možné, že jakmile se metoda dotazuje DNS serveru spravujícího určitou doménu, dotaz může být směrován na, v té chvíli, nejméně zatížený autoritativní DNS server, který však nutně nemusí být DNS serverem nejbližším. Provedená simulace však tento fakt nezohledňuje. Doménový název a k němu patřící IP adresa jsou tedy vybrány náhodně, např. autoritativních kořenových DNS serverů je 13 a každý takovýto DNS server má ještě navíc svá zrcadla [23]. Jejich doménové názvy a IP adresy jsou celosvětově známé a každý autoritativní DNS server musí tyto adresy znát. Dotaz vyslaný DNS serverem tedy může být teoreticky směrován na jakýkoliv kořenový DNS server, který v danou chvíli není zaneprázdněn. Do simulace je tedy náhodně vybrán jeden z těchto kořenových DNS serverů a po celý čas simulace se již tento DNS server nemění. Stejný princip platí i pro ostatní DNS servery spravující jednotlivé domény a subdomény, které se v simulaci vyskytují. Vzdálenosti jednotlivých bodů znázorňujících stanice a DNS servery nejsou odrazem vzdálenosti skutečné ani ve změněném měřítku. Vzdálenost je volena tak, aby bylo dobře patrné rozlišení jednotlivých domén, kdy se jednotlivými dotazy autoritativní DNS server přibližuje k autoritativním DNS serverům hledané domény. Jednotlivé iterace simulace jsou postupně znázorněny graficky.

Tab. 2 Doménové názvy a IP adresy jednotlivých DNS serverů a stanic použitých v simulaci.

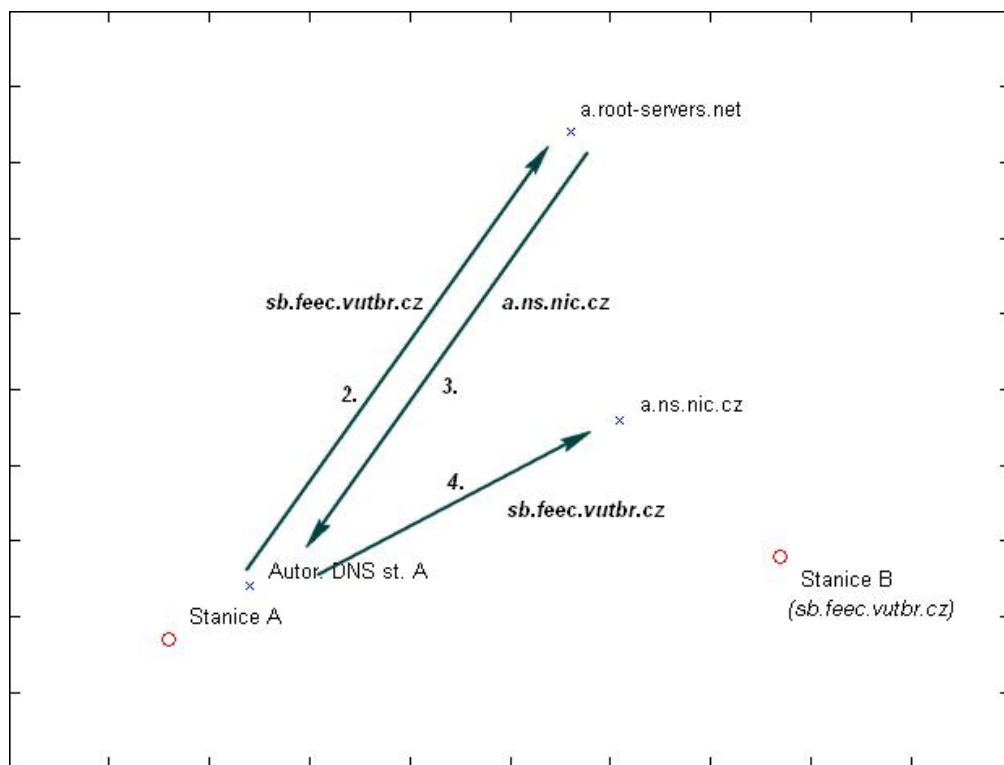
Doménový název	IP adresa	Popis
l.root-servers.net	199.7.83.42	kořenový DNS server
a.ns.nic.cz	194.0.12.1	Autorit. DNS server domény .cz
rhino.cis.vutbr.cz	147.229.3.10	Autorit. DNS server domény <i>vutbr.cz</i> a domény <i>feec.vutbr.cz</i>
kazi.fit.vubtr.cz	147.229.8.12	Autorit. DNS server domény <i>feec.vutbr.cz</i>
kos.feec.vutbr.cz	147.229.72.10	Autorit. DNS server domény <i>feec.vutbr.cz</i>
gate.feec.vutbr.cz	147.229.71.8	Autorit. DNS server domény <i>feec.vutbr.cz</i>

Při inicializaci simulace jsou nastaveny adresy jednotlivých DNS serverů. Máme stanici A, která má svůj poskytovatelem pevně daný autoritativní DNS server v simulaci označený jako *DNS A* a cílem je predikovat latenci ke stanici B s IP adresu 147.229.72.54, která leží v doméně *feec.vutbr.cz*.



Obr. 4.1 Inicializace stanic a odeslání dotazu na stanici B příslušnému DNS serveru.

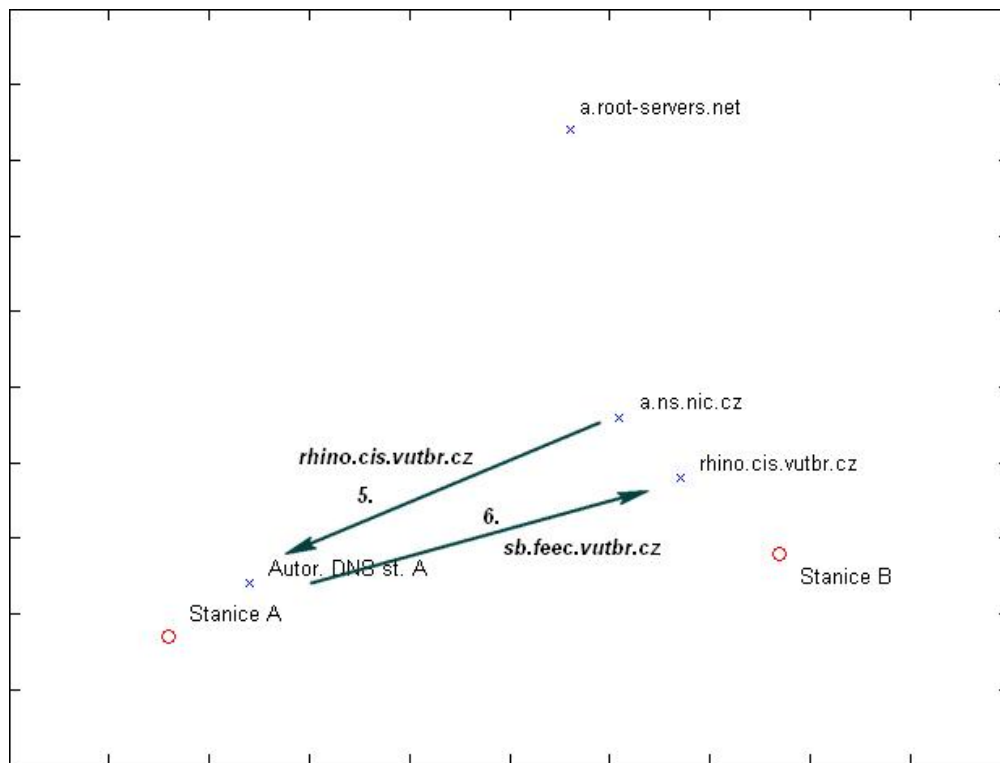
V prvním kroku je nejprve zjištěn autoritativní DNS server stanice B. Stanice A zasílá rekurzivní dotaz *sb.feec.vutbr.cz* svému autoritativnímu DNS serveru viz Obr. 4.1. První část sufixu „*sb.*“ je zvolena tak, aby bylo co nejméně pravděpodobné, že by takovýto dotaz byl autoritativním DNS serverem stanice A již v minulosti řešen. Docílí se tak toho, že tento DNS server se musí dotazovat ostatních DNS serverů. *DNS A* je tedy nucen se dotázat a tento dotaz přeposílá kořenovému DNS serveru *l.root-servers.net*. Je velmi pravděpodobné, že kořenový DNS server ve své databázi odpověď na příslušný dotaz nemá. Má však databázi všech autoritativních DNS serverů všech domén 1. řádu. Odešle tedy k serveru *DNS A* adresu jednoho z autoritativních DNS serveru domény *.cz*, což je *a.ns.nic.cz* s IP adresou *194.0.12.1*. Jelikož *DNS A* v této chvíli zná adresu DNS serveru domény 1. řádu, odesílá dotaz *sb.feec.vutbr.cz* na přijatou adresu viz Obr. 4.2.



Obr. 4.2 Odeslání dotazu na kořenový DNS server, následná odpověď a odeslání dotazu na zjištěnou adresu.

Autoritativní DNS server domény *.cz* samozřejmě nemá tak rozsáhlou databázi, aby znal autoritativní DNS servery stanice B. Má však adresy autoritativních DNS serverů domén 2. řádu, které spadají pod doménu *.cz*. Zná tedy adresu autoritativního DNS serveru domény *vutbr.cz*, odesílá tedy k *DNS A* adresu *rhino.cis.vutbr.cz* s IP adresou *147.229.3.10*. *DNS A* proto zasílá dotaz *sb.feec.vutbr.cz* na DNS server *rhino.cis.vutbr.cz* viz Obr. 4.3. Tento autoritativní DNS server již má ve své databázi adresy autoritativních DNS serverů stanice B a odesílá je k *DNS A* viz Obr. 4.4. Existuje-li více autoritativních DNS serverů jedné domény, metoda King obvykle vyžaduje adresy všech takovýchto DNS serverů a následně vybírá dle výše popsaných dvou kritérií (kapitola 3.2) teoreticky nejbližší DNS server. V našem případě spravují

doménu *feec.vutbr.cz* čtyři autoritativní DNS servery, které jsou uvedeny v Tab. 2.



Obr. 4.3 Zaslání odpovědi s adresou od DNS serveru domény .cz a odeslání dotazu na tuto adresu.

Ve vytvořené simulaci jsou tedy zavedeny algoritmy pro výběr teoreticky nejbližšího autoritativního DNS serveru ke stanici B. Nejprve je proveden algoritmus určený k separaci autoritativního DNS serveru na základě jeho doménového jména. Tímto krokem je zajištěno odseparování těch DNS serverů, které jsou pravděpodobně využívány jako záložní pro případ, že by byl hlavní autoritativní DNS server mimo provoz nebo přetížen a ve skutečnosti mohou spravovat další doménu, pro kterou mohou být primárním DNS serverem. To je patrné například u autoritativního DNS serveru *kazi.fit.vutbr.cz*, u něhož je velmi pravděpodobné, že bude sloužit jako autoritativní DNS server pro doménu *fit.vutbr.cz*. Algoritmus funguje tak, že zadaná doménová jména nalezených autoritativních DNS serverů porovnává s doménovým jménem stanice B. Porovnávání je realizováno postupně od domény 1. řádu směrem k doménám nižších řádů. Jelikož Matlab nepodporuje porovnávání více znaků najednou, čehož by bylo využito u jednotlivých subdomén, je nutné porovnávat každý znak zvlášť. Tato skutečnost však na dobu výpočtu téměř nemá vliv. Následně jsou vypsaný vždy ty DNS servery, jejichž část doménového názvu je shodná se stejnou částí domény, ve které se nachází stanice B. Algoritmus si také ukládá DNS servery, které byly vyřazeny. Výsledkem je poté odseparování určitých DNS serverů. Pokud po tomto kroku nezůstane pouze jeden DNS server, následuje druhá část algoritmu.

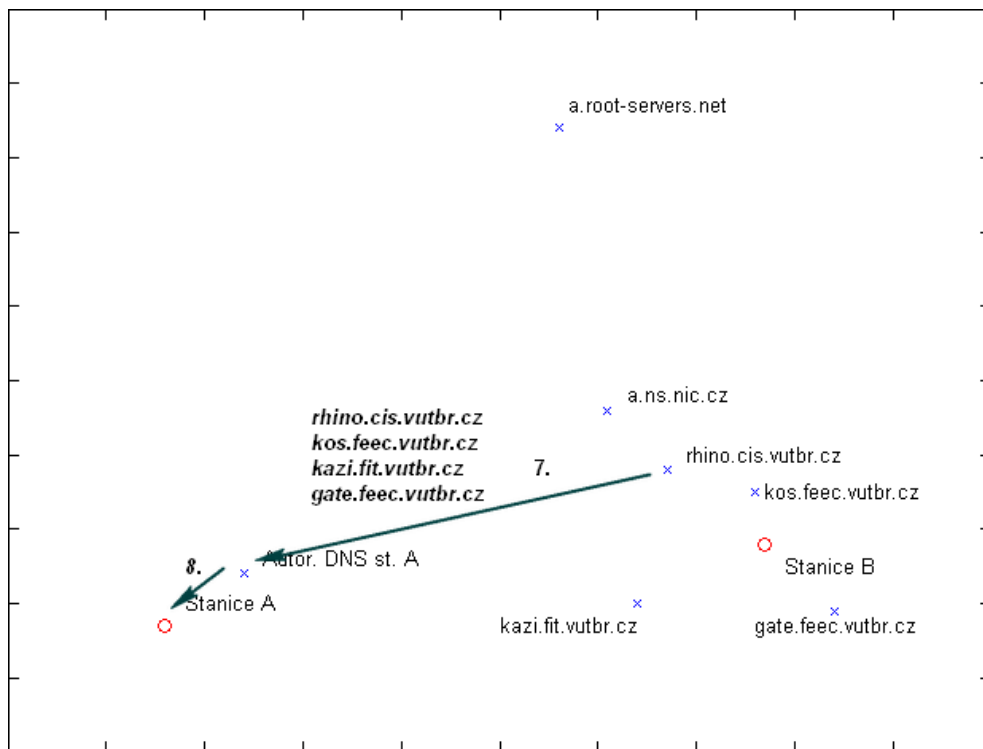
Druhá část algoritmu pracuje na velmi podobném principu jako algoritmus předešlý. Rozdíl je v tom, že porovnává namísto doménových jmen IP adresu stanice B s IP adresami zbylých nalezených autoritativních DNS serverů. Algoritmus pracuje s IP

adresami zadanými na začátku simulace. Postupně porovnává jednotlivé 8-bitové segmenty IP adresy těchto DNS serverů s IP adresou stanice B a vypisuje seznam DNS serverů, u nichž se porovnávaná část shoduje. Výstupem tohoto algoritmu by měl být jediný autoritativní DNS server, který by měl být teoreticky nejbližším DNS serverem ke stanici B.

Jakmile je vybrán teoreticky nejbližší autoritativní DNS server stanice B, přistupuje se k dalšímu kroku. Ke stanici A je zaslán jejím DNS serverem A výsledek jejího dotazu, který byl zaslán nalezeným DNS serverem. Stanice A, jelikož již zná adresu autoritativního DNS serveru stanice B, odesílá dotaz *2sb.feec.vutbr.cz* přímo na tento DNS server, což je *kos.feec.vutbr.cz*. Tento dotaz má jiný tvar než dotaz předešlý. Důvod, proč tomu tak je, je vysvětlen v kapitole 3.2. DNS server *kos.feec.vutbr.cz* na tento dotaz odpoví. Odpověď bude pravděpodobně záporná, tzn. že DNS server adresu pro tento doménový název nezná. To však pro samotnou metodu není důležité. Důležitý je fakt, že byla odeslána odpověď s určitým zpožděním. Toto zpoždění je stěžejní pro samotnou predikci. Jedná se o RTT mezi stanicí A a DNS serverem *kos.feec.vutbr.cz*.

Ve třetím kroku simulace je zjištěna vzdálenost mezi stanicí A a jejím DNS serverem. Toto zpoždění je poté odečteno od zpoždění mezi stanicí A a DNS serverem stanice B. Výsledkem je poté predikovaná hodnota zpoždění mezi stanicí A a stanicí B. Pro porovnání je ještě zjištěna samotná vzdálenost mezi stanicí A a stanicí B a vypočtena přesnost predikce.

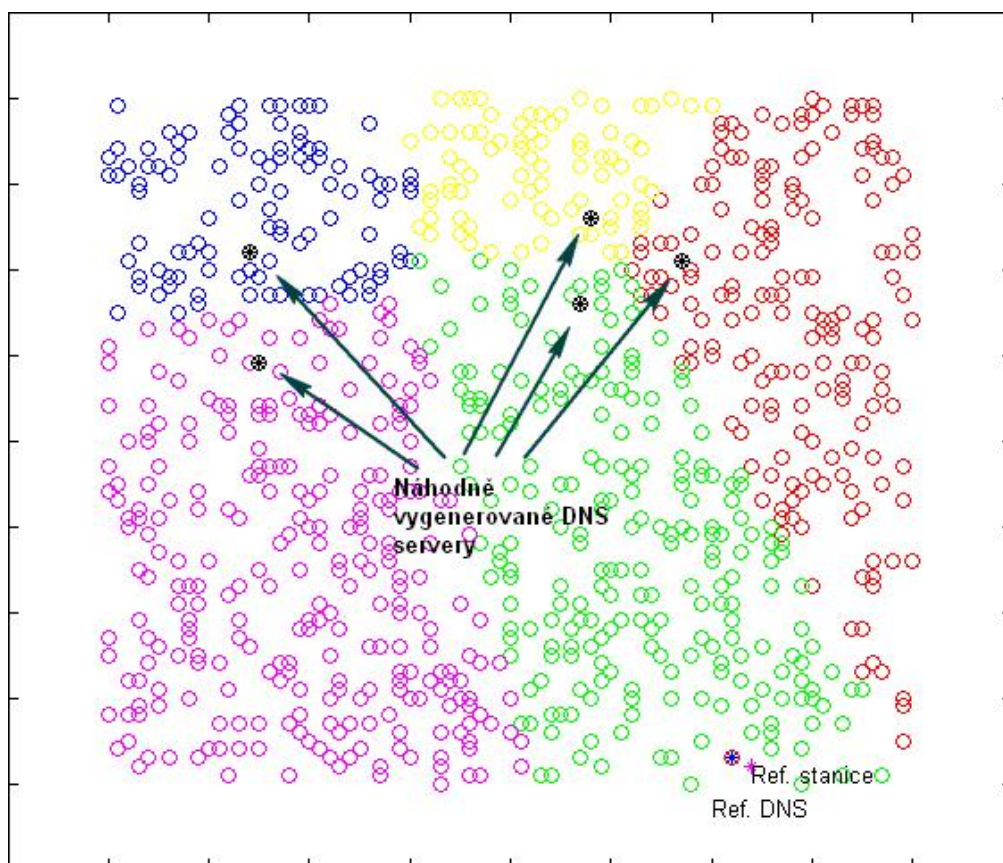
Vytvořená simulace může být libovolně upravována. Zejména mohou být dosazována jiná doménová jména popřípadě jiné IP adresy v poslední iteraci nalezených DNS serverů. Může se také zvětšovat jejich počet.



Obr. 4.4 Odeslání patřičných DNS serverů DNS serverem domény .vutbr.cz.

4.2 Simulace závislosti přesnosti predikce RTT na počtu DNS serverů

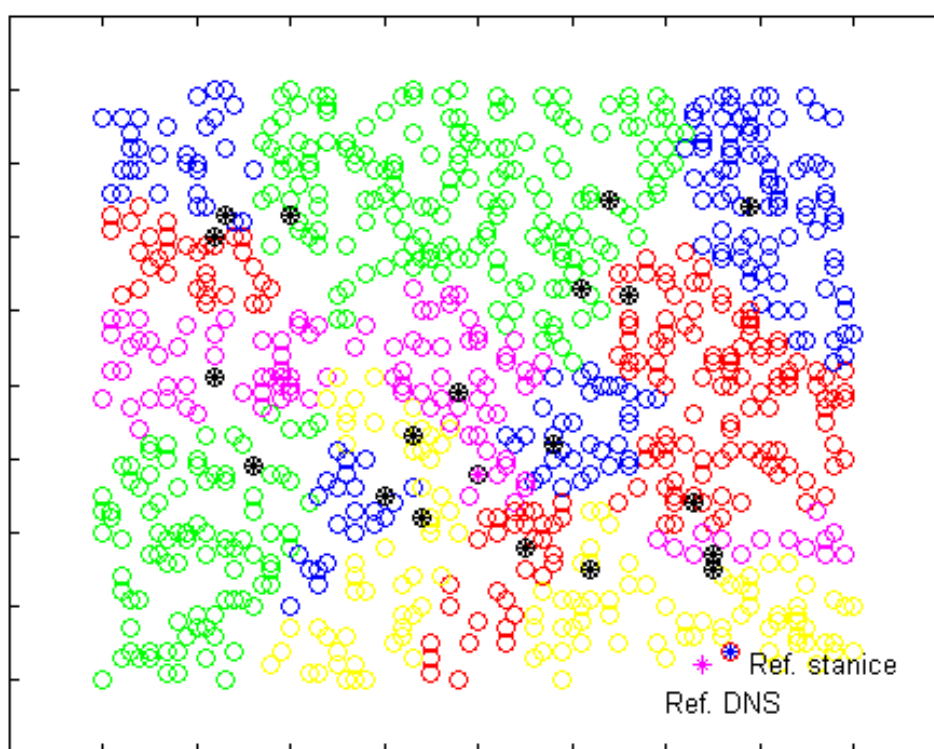
Jako druhá byla vytvořena simulace, která má za úkol zjistit přibližnou závislost přesnosti predikce RTT mezi stanicemi na základě počtu DNS serverů, které tyto stanice obsluhují. Tato simulace se nazývá *bakprojekt.m*. Na začátku simulace je nutné zvolit 2 vstupní parametry, kde první parametr představuje počet stanic a druhý parametr vyjadřuje počet DNS serverů. Příklad zápisu může být takovýto *bakprojekt(900,5)*. Samotné spuštění simulace je realizováno stejným způsobem jako simulace předešlá. Pro účel zjištění závislosti přesnosti predikce RTT na počtu DNS serverů je počet stanic nastaven na neměnnou hodnotu 900. Tato hodnota představuje přibližný počet stanic na fakultě Elektrotechniky. Této hodnoty bylo dosaženo tak, že byl nejdříve zjištěn počet zaměstnanců, od kterého se poté odvíjel počet stanic. K počtu stanic zaměstnanců se poté přičetl přibližný počet stanic v učebnách. Počet DNS serverů je v simulaci zvyšován vždy o hodnotu 1 v rozmezí od 2 do 20 DNS serverů. Tento interval byl volen tak, aby počet DNS serverů přibližně odpovídal skutečným podmínkám. Např. fakultu Elektrotechniky obsluhují 4 DNS servery. Výsledkem je poté graficky zpracovaná závislost vyjadřující, jak moc se zvyšuje přesnost predikce RTT mezi vybranými stanicemi, se zvyšujícím se počtem DNS serverů, které tyto stanice obsluhují.



Obr. 4.5 Grafický výstup simulace závislosti přesnosti predikce RTT na počtu generovaných DNS serverů.

Po zadání vstupních parametrů se náhodně vygeneruje zadaný počet stanic a DNS serverů. Jeden DNS server je fixně nastaven na určitou fixní pozici, která je při každém spuštění simulace neměnná. Tento DNS server reprezentuje předem známý DNS server, který stanice zná a odesílá k němu své dotazy. Od tohoto DNS serveru se poté odvíjí celá simulace.

V prvním kroku je nejprve z nastaveného počtu vygenerovaných stanic vybrána stanice, která je nejbližší k výše popsanému fixně nastavenému DNS serveru. Jakmile je tato stanice nalezena, nastaví se jako referenční po celý průběh simulace. Od této referenční stanice je poté vypočítána vzdálenost představující zpoždění ke každé ze zbývajících stanic. Pokud tedy máme 900 stanic, simulace bude obsahovat 899 hodnot zpoždění. Tyto hodnoty budou poté sloužit k porovnání s hodnotami, které budou získány na základě predikce.



Obr. 4.6 Příklad překrývajících se barev uzlů.

Ve druhém kroku je realizováno nalezení nejbližšího DNS serveru ke každé stanici. Toho je dosaženo za pomoci algoritmu, který zná pozici každého DNS serveru. Tento algoritmus vybírá popořadě vždy jednu stanici z proměnné, ve které jsou uloženy pozice jednotlivých stanic. Jakmile je stanice vybrána, začínají být prohledávány DNS servery. Vždy je vypočítána vzdálenost mezi stanicí a právě prohledávaným DNS serverem. Vznikne tak soubor hodnot vzdáleností mezi vybranou stanicí a všemi vygenerovanými DNS servery. Nakonec se vybere DNS server s nejmenší hodnotou vzdálenosti. Stejný postup je poté aplikován na ostatní stanice. Aby bylo dobře patrné, kterou stanicí obsluhuje který DNS server, je v simulaci aplikováno barevné rozlišení stanic. DNS

server určitého barevné uskupení stanic je označen černou barvou a je ho možné identifikovat tak, že kolem něj leží stanice stejné barvy. Jelikož funkce Matlabu zvaná *plot()* podporuje pouze 7 barev, přičemž některé jsou hůře viditelné, bylo k barevnému rozlišení zvoleno 5 barev. Pokud je nastaven počet generovaných DNS serverů například na 20, stanice náležící k určitému DNS serveru mohou mít stejnou barvu jako stanice, které náleží již k DNS serveru jinému viz Obr. 4.6. Nejlépe je tedy barevné rozlišení vidět u 5 vygenerovaných DNS serverů.

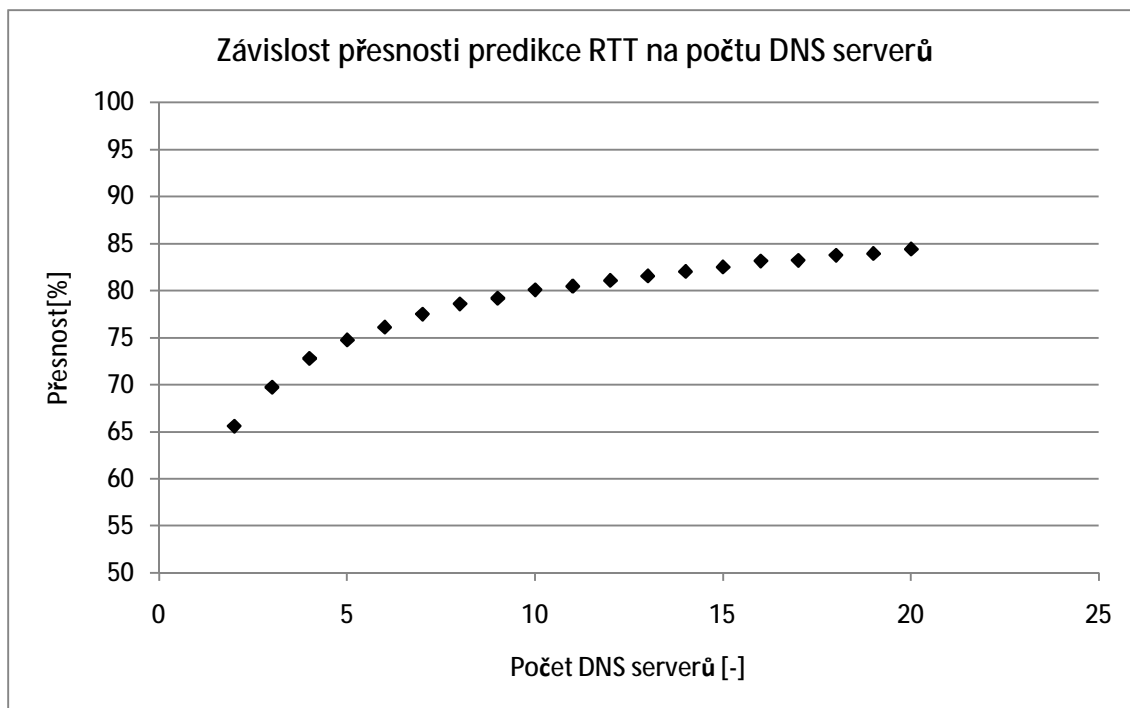
Jakmile jsou nalezeny nejbližší DNS servery ke všem stanicím, postoupí se ke třetímu kroku, kdy je vypočítána vzdálenost mezi nejbližšími DNS servery a fixně nastaveným DNS serverem. Tato hodnota poté odpovídá predikovanému RTT mezi stanicemi.

Nakonec jsou postupně porovnány všechny hodnoty predikovaného zpoždění s odpovídajícími hodnotami zpoždění reálného. Z vypočtených 899 hodnot je poté vypočítán aritmetický průměr. Tyto hodnoty jsou v závěru vyneseny do grafu.

4.3 Porovnání přesnosti predikce s dalšími metodami

Jednotlivé porovnávané metody ke své činnosti využívají jak umělých souřadnicových systémů, tak i přímých měření. Využívají různé strukturní prvky, pomocí nichž predikují zpoždění a využívají i odlišných algoritmů. Z tohoto faktu plyne zejména to, že predikce RTT různých metod může být značně odlišná. Navíc u jednotlivých metod nemusí platit přímá úměra počtu prvků metody a přesnosti predikce. Přímé porovnání tedy nemusí mít patřičnou vypovídající hodnotu. Tato kapitola se zaměřuje na porovnání přesnosti predikce RTT v závislosti na počtu určitých prvků typických pro danou metodu. U každé porovnávané metody je náhodně vygenerováno 900 uzlů, které stejně jako u simulace metody King, reprezentují teoretický počet stanic fakulty FEKT. Následně jsou postupně náhodně generovány prvky typické pro danou metodu. Počet je nastaven od 2 do 20 prvků, přičemž se počet zvyšuje vždy o hodnotu 1.

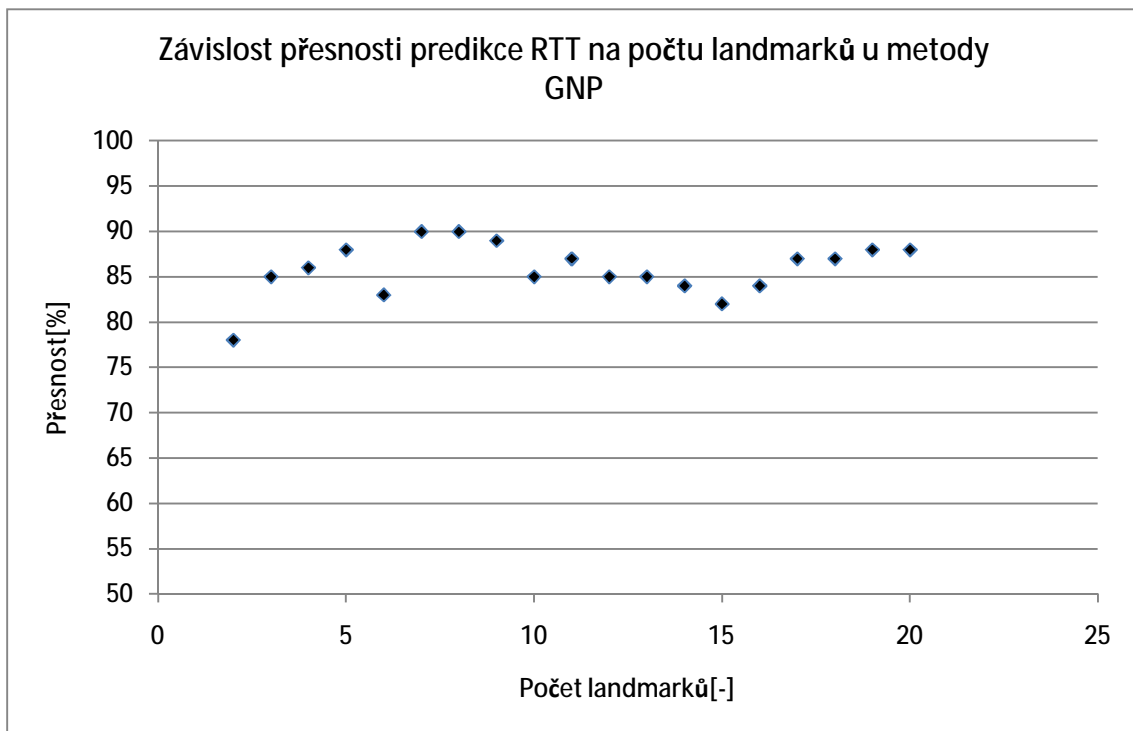
Dosahovaná přesnost predikce RTT metody King je vidět na Obr. 4.7 Přesnost predikce metody King je silně závislá na umístění DNS serveru. Jako nejideálnější řešení by bylo rozmístění co nejvíce DNS serverů v určitém prostoru a to nejlépe tak, aby byly umístěny ve středu určitých malých množin uzlů. Toto řešení však v praxi nelze realizovat, jelikož by byla správa takového počtu DNS serverů téměř nerealizovatelná a bylo by velmi složité udržovat takto rozsáhlou celosvětovou databázi DNS aktuální. Dosahovaná přesnost predikce RTT je tedy pouze teoretická a počítá s náhodným umístěním DNS serverů. Z toho také plyne fakt, že při generaci malého počtu DNS serverů může být predikovaná hodnota RTT jednotlivých měření značně odlišná. Proto bylo provedeno více iterací měření, z nichž byl vypočítán aritmetický průměr. Přesnosti predikce RTT okolo 70 % bylo u metody King dosaženo u počtu čtyř DNS serverů. U vyššího počtu DNS serverů již přesnost predikce RTT nemá takovou strmou tendenci vzrůstu. Například dosahovaná přesnost predikce RTT u 5 DNS serverů byla asi o 10 % větší než u 2 DNS serverů a přesnost predikce RTT u 20 DNS serverů byla větší také asi jen o 10 % než u 5 DNS serverů. Přesnosti 80 %, je dosaženo u počtu 10 DNS serverů. Z toho lze také vyvodit, že další zvyšování počtu DNS serverů je již ekonomicky nevýhodné, protože přidaná hodnota v podobě zvýšené přesnosti predikce RTT již není adekvátní vůči vynaloženým nákladům.



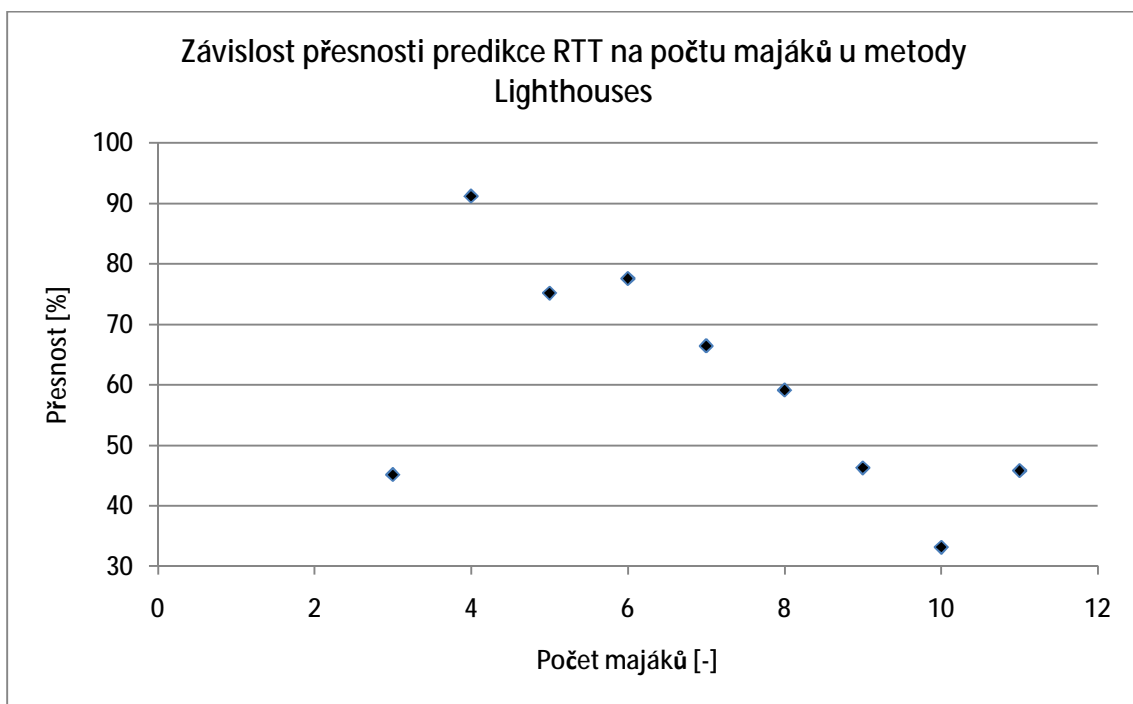
Obr. 4.7 Graf závislosti přesnosti predikce RTT metody King na počtu náhodně vygenerovaných DNS serverů.

Další porovnávanou metodou je metoda GNP (viz 1.2.1). Následující probírané hodnoty jsou převzaty z diplomové práce [21]. Tato metoda pracuje na odlišném principu činnosti, kdy nejsou používány fyzické prvky sítě. K predikci RTT je využito umělých souřadnicových systémů. Jak je vidět z Obr. 4.8, přesnost predikce RTT oproti metodě King značně kolísá. Je také nutno říci, že pro ideální volbu souřadnic orientačních bodů je možné využít různých minimalizačních funkcí, kdy každá může konvergovat poněkud odlišně. Z grafické závislosti však vyplývá, že zvětšující se počet orientačních bodů nemusí nutně znamenat zlepšení predikce RTT. Přesnost predikce RTT je však poměrně vysoká. Již při počtu 2 orientačních bodů dosahuje asi 75 %. Vysoké přesnosti 90 % je dosaženo při počtu 7 orientačních bodů. Další zvyšování počtu orientačních bodů pak již nemá patřičný vliv na zlepšování predikce RTT. Naopak se přesnost zhoršuje. Například u počtu 15 orientačních bodů byla přesnost predikce na hodnotě asi 82 %. Poté se postupně zlepšuje a od hodnoty 17 orientačních bodů se přesnost ustálila na asi 87 %. Z uvedené grafické závislosti tedy vyplývá, že metoda GNP je teoreticky přesnější než metoda King. Metoda King je však hodně závislá na fyzickém umístění jejích prvků. V určitých případech tedy může vykazovat lepší výsledky než metoda GNP.

Jako další porovnávaná metoda byl zvolen systém Lighthouses (viz 1.2.3). Tento systém pracuje na obdobném principu jako výše zmíněný systém GNP. Ke své činnosti využívá namísto orientačních bodů majáky. V grafické závislosti viz Obr. 4.9 je postupně zvyšován počet majáků od hodnoty 3 na hodnotu 11. Není zde tedy takový počet prvků, jako je tomu v případě předešlých dvou metod. Hodnoty této grafické závislosti jsou získány ze simulace metody Lighthouses provedené v diplomové práci [22].

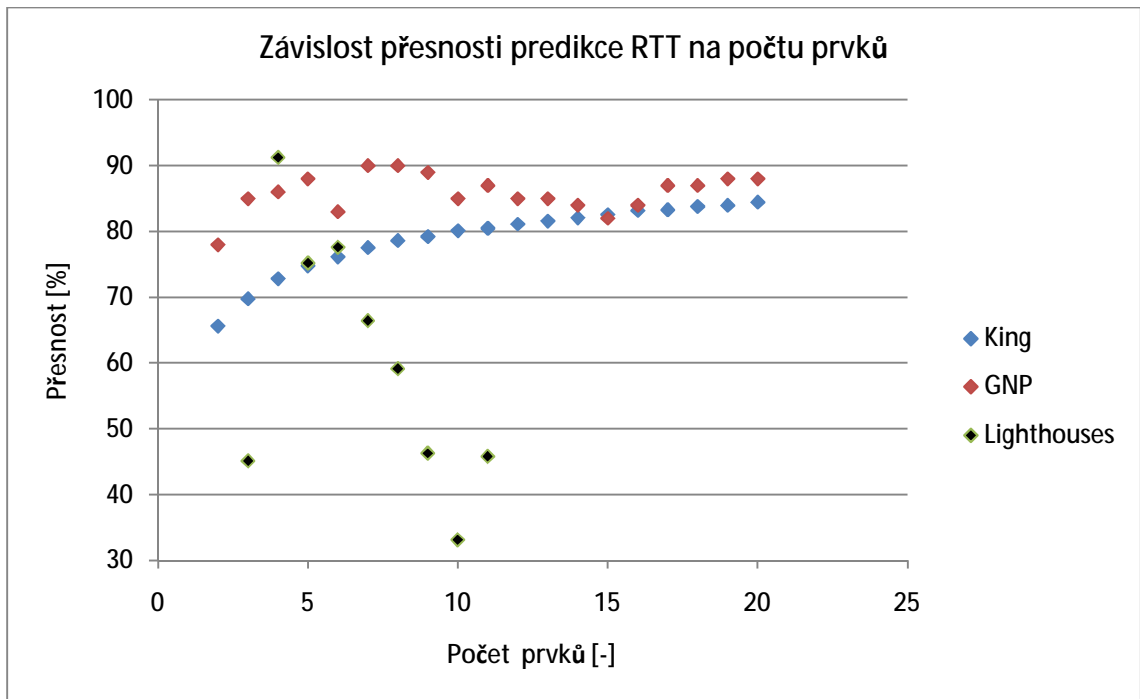


Obr. 4.8 Graf závislosti přesnosti predikce RTT metody GNP na počtu vygenerovaných landmarků.



Obr. 4.9 Graf závislosti přesnosti predikce RTT systému Lighthouses na počtu náhodně generovaných majáků.

Jak je vidět na Obr. 4.9, přesnost predikce RTT stejně jako u systému GNP značně kolísá. Zde jsou však výkyvy přesnosti ještě patrnější. Přesnost v závislosti na počtu majáků klesá i pod 50 %. U počtu čtyř majáků, což odpovídá 3D dimenzi, dosahuje přesnost hodnoty přes 90 %, z čehož lze vyvodit, že takovýto počet majáků a z toho plynoucí 3D dimenze je pro daný počet stanic dostačující. Postupným generováním více majáků přesnost téměř lineárně klesá až k hodnotě pod 40 % u počtu 10 majáků. Zvyšování počtu majáků má tedy opačný charakter průběhu, než by se dalo očekávat. Z toho plyne, že zvyšování počtu majáků v systému již nemá patřičný vliv na přesnost predikce RTT, naopak přesnost ještě zhoršuje.



Obr. 4.10 Závislost přesnosti predikce RTT na počtu prvků pro všechny porovnávané metody.

5 ZÁVĚR

Tato bakalářská práce byla zaměřena zejména na nastudování metod predikce RTT, kde byl největší důraz kladen na metodu King. Po patřičném seznámení se s touto metodou a nastudování systému DNS byla k nastínění funkčnosti této metody vytvořena simulace v prostředí Matlab. V provedené simulaci byly popsány jednotlivé iterace metody King spolu s doprovodným grafickým znázorněním. Jako další byla vytvořena simulace, výsledkem které je závislost přesnosti predikce RTT na počtu náhodně generovaných DNS serverů, která je následně podrobně okomentována. Z uvedené závislosti vyplývá, že přesnosti okolo 70 % je dosaženo průměrně u počtu 4 náhodně generovaných DNS serverů. Výsledná závislost je poté porovnána z hlediska přesnosti predikce RTT z dalšími metodami, kdy je nastaven stejný počet uzlů a poté jsou generovány prvky specifické pro danou metodu. Hodnocena je dosažená přesnost predikce při určitém počtu generovaných prvků.

Hodnoty u porovnávaných grafů byly získány z diplomových a bakalářských prací, které byly vytvářeny souběžně s touto bakalářskou prací. Tyto diplomové a bakalářské práce zpracovávají stejnou problematiku, zaměřují se však na odlišné metody řešení. Výsledky daných prací je poté možné porovnávat z více hledisek. Dosažené výsledky budou sloužit pro publikaci, která bude zaměřena na rozbor metod sloužících pro predikci RTT.

LITERATURA

- [1] ZDRÁHAL, L., KROPÁČEK, J. *Technologie Ethernetu*. Brno : Vysoké učení technické v Brně, Fakulta informačních technologií, [2008?]. 15 s. Seminární práce předmětu CC3 modul CCNA1.7. Vedoucí práce: Ing. Petr Matoušek Ph.D. Dostupný z WWW: <<http://netacad.fit.vutbr.cz/texty/ccna-moduly/ccna1-7.pdf>>[cit. 8.12.2009].
- [2] JAROŠOVÁ L. Měření vzdálenosti stanic prostřednictvím ICMP protokolu v IP sítích. Brno: Vysoké učení technické v Brně, Fakulta Elektrotechniky a komunikačních technologií, Ústav telekomunikací, Rok vydání: 2008. Počet stran 63. Vedoucí bakalářské práce Ing. Radim Burget.
- [3] NG, E. - HUI, Z. Predicting Internet network distance with coordinates-based approaches [online]. In INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies.. [s.l.]: IEEE, 2002. str. 170-179, svazek 1. URL: <<http://www.cs.rice.edu/~eugeneng/papers/INFOCOM02.pdf>>.
- [4] J.A. Nelder and R.Mead, "A simplex method for function minimization," Computer Journal, vol. 7, pp. 308–313, 1965.
- [5] ŠVÉDA, J. Nalezení pozice stanic v Internetu pomocí uměle vytvořených souřadnicových systémů. Brno: Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií. Ústav telekomunikací, 2009. 85 s. Diplomová práce. Vedoucí práce Ing. Dan Komosný, Ph.D.
- [6] HANDL, T. Algoritmus Vivaldi pro nalezení pozice stanice v Internetu. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 69 s. Vedoucí diplomové práce Ing. Dan Komosný, Ph.D.
- [7] HALLIDAY, D. - RESNICK, R. - WALKER, J. Fyzika. Brno: VUTIUM, 2003. Dotisk 1. českého vydání. ISBN 80-214-1868-0. Část 1, kapitola 7, s. 153.
- [8] FRANCIS, P. IDMaps: A Global Internet Host Distance Estimation Service. In IEEE/ACM Trans. on Networking [online], New York: [s.n.], říjen 2001. URL: <<http://http://idmaps.eecs.umich.edu/papers/ton01.pdf>>.
- [9] BARTAL, Y. "Probabilistic approximation of metric space and its algorithmic applications," in 37th Annual IEEE Symposium on Foundations of Computer Science, October 1996.
- [10] VAZIRANI, V, Approximation Methods, Springer-Verlag, 1999.
- [11] GUMMADI, K. - SAROIU, S. - GRIBBLE, S. King: Estimating Latency between Arbitrary Internet End Hosts [online]. In SIGCOMM Internet Measurement Workshop 2002. Seattle: University of Washington. Department of Computer Science & Engineering, 2002 URL: <<http://www.mpi-sws.org/~gummadi/king/king.pdf>> [cit. 25. 11. 2009].
- [12] DOSTÁLEK, L., KABELOVÁ, A. *Velký průvodce protokoly TCP/IP a systémem DNS*. Praha : Computer Press, 2000. 435 s.
- [13] WONG, B., SLIVKINS, A., SIRER, E. G. *Meridian: A Lightweight Framework for Network Positioning without Virtual Coordinates*. Dept. of Computer Science, Cornell University, Ithaca, NY 14853 : [s.n.], 2005. 20 s. Dostupný z WWW: <http://www.cs.cornell.edu/people/egs/meridian/papers/TR_meridian.pdf> [cit. 4. 12. 2009].
- [14] *MATLAB* [online]. Vysoká škola chemicko-technologická v Praze. Praha : Dostupný z WWW: <<http://uprt.vscht.cz/majerova/matlab/index.html>>.

- [15] ŠŤASTNÝ, P. *Průvodce DNS, DNS záznamy a protokol* [online]. 2007. Dostupný z WWW: <<http://www.pweb.cz/dns/index.html>>.
- [16] GUMMADI, K. - SAROIU, S. - GRIBBLE, S. *King : A tool to estimate latency between any two Internet hosts, from any other Internet host.* [online]. 2002. Dostupný z WWW: <<http://www.mpi-sws.org/~gummađi/king/>>.
- [17] PIAS, M. et al. Lighthouses for scalable distributed location [online]. URL: <<http://research.microsoft.com/~tharris/papers/2003-iptps.pdf>> [cit. 28. 11. 2009].
- [18] CHEN, Y. - OVERTON, C. - KATZ, R., *Internet Iso-bar: A Scalable Overlay Distance Monitoring System* [online]. In *Journal of Computer Resource Management, Computer Measurement Group. Spring Edition.* [s.l.]: [s.n.], 2002. URL: <<http://www.cs.northwestern.edu/~ychen/research/wmmms/isobar.ps>>.
- [19] LI-WEI, L. *PCoord: a decentralized network coordinate system for Internet distance prediction* [online]. Cambridge (USA): Massachusetts Institute of Technology, Dept. of Civil and Environmental Engineering, 2005. URL: <<http://dspace.mit.edu/bitstream/handle/1721.1/31134/61184096.pdf?sequence=1>>.
- [20] V. Varirani, *Approximation Methods*, Springer-Verlag, 1999.
- [21] ŠKVOR, M. *Lokalizace stanic v síti Internet pomocí umělých souřadnicových systémů*: Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2010. 58 s. Vedoucí diplomové práce doc. Ing. Dan Komosný, Ph.D.
- [22] ŠIMÁK, J. *Měření vzdáleností mezi stanicemi v IP sítích*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2010. 58 s. Vedoucí diplomové práce doc. Ing. Dan Komosný, Ph.D.
- [23] *Root Server Technical Operations Assn* [online]. 2009. Root-servers.org. Dostupné z WWW: <<http://www.root-servers.org/>>.

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

DNS	Domain Name System
GNP	Global Network Positioning
HDD	Hard Disk Drive
HOPS	Host Proximity Service
ICANN	Internet Corporation for Assigned Names and Numbers
ICMP	Internet Control Message Protocol
IP	Internet Protocol
IPTV	Internet Protocol Television
k-HST	k-Hierarchically well-Separated Trees
PING	Packet InterNet Grouper
RFC	Request For Comments
SLD	Second Level Domain
TCP	Transmission Control Protocol
TLD	Top Level Domain
TTL	Time To Live
UDP	User Datagram Protocol
VP	Virtual Links