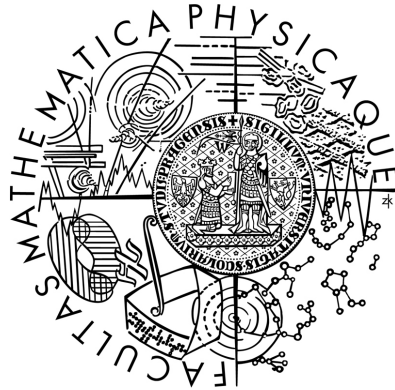


Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

BAKALÁŘSKÁ PRÁCE



Jiří Danihelka

Vizualizace trajektorie automobilů odhadnuté na základě úrovně signálu pro mobilní telefony

Katedra softwarového inženýrství

Vedoucí bakalářské práce: RNDr. Tomáš Kalibera

Studijní program: Informatika, Programování

2006

Poděkování

Na tomto místě bych rád poděkoval Výzkumnému a vývojovému centru RDC, které mi umožnilo přístup k technologiím bezdrátových sítí a poskytlo potřebné hardwarové vybavení.

Dále děkuji vedoucímu bakalářské práce za jeho cenné připomínky, které podstatným způsobem přispěly k zlepšení úrovně práce.

Prohlašuji, že jsem svou bakalářskou práci napsal samostatně a výhradně s použitím citovaných pramenů. Souhlasím se zapůjčováním práce.

V Praze dne 25.5.2006

Jiří Danihelka

Obsah

1	Úvod	5
1.1	Slovo úvodem	5
1.2	Rozsah práce	6
1.3	Popis kapitol	7
1.4	Možná využití lokalizace	8
1.5	Výzkumné a vývojové centrum RDC	9
1.6	Model šíření signálu.....	10
1.7	Princip lokalizačních algoritmů.....	11
2	Používání programu	12
2.1	Požadavky pro spuštění programu	12
2.2	Instalace a spuštění	12
2.3	Ovládání programu Guesstor Client	13
2.4	Popis ovládacích prvků	18
2.5	Popis tabulek.....	21
2.6	Ovládání pomocí klávesnice	25
2.7	Stručný popis lokalizačních algoritmů	26
3	Popis použitých technologií	28
3.1	Global Positioning System (GPS)	28
3.2	Global System for Mobile Communications (GSM)	30
3.3	System základnových stanic BTS.....	31
3.4	Souřadnicový systém UTM (Universal Transverse Mercator).....	32
3.5	Výhody souřadnicového systému UTM	33
3.6	Spring Framework	34
4	Programátorská dokumentace.....	35
4.1	Komunikace mezi serverem a klientem.....	35
4.2	Datové struktury přenášené mezi serverem a klientem.....	37
4.3	Popis tříd programu Guesstor Client	41
4.4	Generátor měření	46
4.5	Databázové technologie	47
5	Závěr	48
5.1	Zhodnocení přesnosti jednotlivých metod	48
5.2	Závěrečné ohodnocení jednotlivých metod	49
5.3	Zhodnocení dosažených výsledků	50
	Literatura.....	51

Název práce: Vizualizace trajektorie automobilů odhadnuté na základě úrovně signálu pro mobilní telefony

Autor: Jiří Danihelka

Katedra: Katedra softwarového inženýrství

Vedoucí bakalářské práce: RNDr. Tomáš Kalibera

e-mail vedoucího: kalibera@nenya.ms.mff.cuni.cz

Abstrakt:

Tato práce se zabývá zobrazováním polohy a trajektorie automobilu či jiného objektu, která byla určena pouze na základě informací o intenzitě signálu pro mobilní telefony. K určení polohy je použito lokalizačních algoritmů, údajů o poloze vysílačů a modelu šíření signálu. Součástí práce je program Guesstor Client, který přehlednou grafickou formou zobrazuje výsledky lokalizačních algoritmů, umožňuje srovnání přesnosti různých lokalizačních metod a dovoluje sledovat trajektorii automobilů v reálném čase. Program Guesstor Client se pomocí protokolu pro vzdálené volání metod připojuje k programu Guesstor Server, který obsahuje lokalizační algoritmy a spravuje databázi měření.

Klíčová slova: mobilní telefon, trajektorie, sledování

Title: Visualization of trajectories of cars estimated using mobile phones signal levels

Author: Jiří Danihelka

Department: Department of Software Engineering

Supervisor: RNDr. Tomáš Kalibera

Supervisor's e-mail address: kalibera@nenya.ms.mff.cuni.cz

Abstract:

This thesis covers the issue of location and trajectory displaying of a car or another object which were determined only on the basis of cell phone signal levels. Localization algorithms, information about the transmitter's positions and a signal propagation model are used to determine the location. A part of the thesis is the Guesstor Client program that displays the results of localization algorithms in a well arranged graphic form, makes it possible to compare the accuracy of different localization methods and allows us to follow the trajectory of automobiles in real time. The Guesstor Client program connects via a remote method invocation protocol the program Guesstor Server which contains the location algorithms and manages the measurements database.

Keywords: cell phone, trajectory, monitoring

1 Úvod

1.1 Slovo úvodem

V GSM sítích pro mobilní telefony je účastník sítě obsloužen signálem z jedné nebo více základnových stanic (BTS). Základnové stanice představují rádiové vysílače vybudované mobilním operátorem. Mobilní telefon monitoruje úroveň signálu z okolních vysílačů a vybere si ten, od kterého dostává nejsilnější rádiový signál. Informace o síle signálů z jednotlivých vysílačů je možné využít k zjištění polohy mobilního telefonu.

Myšlenka odhadovat polohu na základě intenzity signálu pro mobilní telefony se zrodila ve Výzkumném a vývojovém centru RDC. Toto centrum si klade za cíl hledat nové využití u stávajících bezdrátových technologií. Již v minulosti zde byly projekty, které se zabývaly lokalizací, využívaly však odlišné technologie. Vůdčími vývojáři projektu lokalizace na základě intenzity signálu byly Martin Orlich a Zbyněk Novák. Ti si stanovili jako svůj cíl vytvořit program Guesstor Server, který by uměl na základě intenzity signálů polohu vypočítat. Program pojmenovali Guesstor, protože již předem očekávali, že výsledky budou značně nepřesné.

V době, kdy jsem s touto prací začínal, byl už v RDC lokalizační Guesstor Server z části implementován. Tento server uměl přibližně určit polohu místa, ve kterém bylo měření provedeno, ale jeho výstup byl pouze textový do konzole. Výsledky ve formě řad čísel zeměpisných souřadnic byly značně nepřehledné a programátoři lokalizačních algoritmů měli jen velmi omezenou kontrolu nad tím, zda jejich program dává očekávané hodnoty. Proto jsem se rozhodl, že programátorům lokalizačního serveru pomohu a vytvořím program, který bude výsledky programu Guesstor Server graficky znázorňovat. Program jsem pojmenoval Guesstor Client.

Již od začátku bylo jasné, že výsledky lokalizace na základě intenzity signálu budou mnohem méně přesné než je tomu například u systému GPS. Cílem projektu lokalizace bylo umožnit přibližnou lokalizaci mobilního telefonu, aniž by bylo potřeba zasahovat do současného hardwarového vybavení mobilní sítě nebo telefonu samotného. Oproti systému GPS je možné určovat údaje o poloze pouze za pomoci změny software v mobilním telefonu. Díky tomu je tato technologie mnohem méně nákladná.

Původní cíl projektu

Automobily mobilního operátora Oskar/Vodafone jsou vybaveny zařízením, které je funkčně podobné mobilnímu telefonu. Čas od času se stane, že některé z těchto aut je odcizeno. Díky lokalizaci na základě síly signálu by bylo možné určit, kde se odcizený automobil pohybuje a pachatele podle toho vystopovat. Takovouto lokalizaci je možné uskutečnit bez nutnosti měnit jakýkoliv hardware, cena zavedení této technologie proto bude velmi nízká. Hodnota jediného nalezeného auta mnohonásobně převyší náklady na vývoj příslušného software.

1.2 Rozsah práce

V této podkapitole je popsáno, co bylo potřeba vyvinout jako součást bakalářské práce. Cílem je rozlišit to, co vytvořili ostatní vývojáři Guesstor Serveru od vývoje, který provedl autor této bakalářské práce.

Návrh a implementace grafického rozhraní

V rámci bakalářské práce bylo potřeba navrhnout a naimplementovat grafické uživatelské rozhraní programu Guesstor Client, které by zobrazovalo výsledky Guesstor Serveru. Bylo požadováno, aby grafické rozhraní umělo demonstrovat výsledky lokalizačních algoritmů, sloužilo jako praktická pomůcka při ladění lokalizačních algoritmů a umožňovalo srovnávání jejich přesnosti. Později přibyl požadavek na zobrazování trajektorií automobilů. Pro implementaci grafického rozhraní byla využita knihovna Swing, která je standardní součástí Javy.

Spojení Guesstor Clienta a Guesstor Server

Guesstor Server umožňuje spojení s klientem pomocí rozhraní Java RMI (Remote Method Invocation). Bylo tudíž potřeba nastudovat technologii RMI a implementovat komunikační rozhraní na straně Guesstor Clienta.

Rozšíření Guesstor Serveru

Jelikož Guesstor Server nepodporoval funkce nutné k sledování trajektorie, bylo zapotřebí ho rozšířit o novou funkčnost. Protože program Guesstor Server byl naprogramován pomocí technologie Spring Framework, bylo nutné se s touto technologií nejdříve seznámit. Guesstor Server původně umožňoval pouze dotazovat se na nějaké měření z databáze. Nově byla přidána možnost dotazovat se na polohu konkrétního automobilu. Aby bylo možné demonstrovat funkčnost sledování trajektorie i v době, kdy se žádný ze sledovaných automobilů nepohybuje, byl do Guesstor Serveru přidán generátor měření, který pohyb automobilů simuluje.

Přizpůsobení Guesstor Serveru pro běh v jiném prostředí

Původní návrh Guesstor Serveru počítal s jeho nasazením pouze v prostorách Výzkumného a vývojového centra RDC. Jelikož však program Guesstor Client vyžaduje pro svůj běh Guesstor Server, bylo zapotřebí rozšířit přenositelnost Guesstor Serveru tak, aby ho bylo možné spouštět i mimo RDC. Největší provedenou změnou v tomto směru byla výměna druhu použité databáze z MySQL na HSQLDB. Jelikož tyto databáze používají jinou variantu jazyka SQL, bylo také nutné pozměnit dotazy do databáze a přesunout data mezi databázemi.

1.3 Popis kapitol

Bakalářská práce je rozdělena do pěti kapitol. První kapitola popisuje **principy lokalizace na základě intenzity signálu pro mobilní telefony**. Je v ní vysvětlen princip lokalizačních algoritmů a použitý model šíření signálu. Dále obsahuje popis možných využití lokalizace polohy. Jedna z podkapitol je věnována Výzkumnému a vývojovému centru RDC.

Druhá kapitola se zaměřuje na praktické **používání programu Guesstor Client**. Obsahuje popis postupu, jak programy Guesstor Server a Guesstor Client nainstalovat na počítač a spustit. Dále je zde uveden návod, jak program Guesstor Client ovládat, jak ho využít pro sledování trajektorií objektů a pro srovnávání lokalizačních algoritmů. Následuje popis ovládacích prvků programu Guesstor Client včetně důležitých tabulek. Závěr kapitoly stručně popisuje dostupné lokalizační algoritmy.

Třetí kapitola **popisuje technologie**, které využívají programy Guesstor Server a Guesstor Client. Je zde uveden popis systému GPS (Global Positioning System), který umožňuje přesné určení polohy. Následuje popis mobilní sítě GSM (Global System for Mobile Communications). Lokalizační algoritmy využívají vlastností sítě GSM k určení místa, kde se mobilní telefon nachází. Jako hlavní souřadnicový systém pro stanovení zeměpisných souřadnic je v práci použit systém UTM (Universal Transverse Mercator). Třetí kapitola popisuje systém UTM včetně důvodů, proč je jeho použití vhodnější než použití jiných souřadnicových systémů. V závěru třetí kapitoly je popsána knihovna Spring Framework, která je základním kamenem programu Guesstor Server.

Čtvrtá kapitola obsahuje **programátorskou dokumentaci** k programu Guesstor Client. Jsou zde uvedeny nejdůležitější třídy programu spolu s popisem, jak tyto třídy vzájemně spolupracují. Dále je zde popsán způsob komunikace mezi programy Guesstor Client a Guesstor Server pomocí rozhraní Java RMI (Remote Method Invocation) včetně datových struktur, které jsou pomocí tohoto rozhraní předávány. V závěru této kapitoly jsou uvedena rozšíření programu Guesstor Server, které implementoval autor této bakalářské práce. Jedná se o generátor měření a rozšíření přenositelnosti programu Guesstor Server.

Poslední pátá závěrečná kapitola **popisuje dosažené výsledky** a porovnává jednotlivé lokalizační metody.

1.4 Možná využití lokalizace

Lokalizace nouzového volání

Pomocí lokalizace je možné alespoň přibližně určit polohu volajícího při volání na nouzové linky (policie, hasiči, záchranná služba) a využít ji při poskytování pomoci. Metody lokalizace na základě síly signálu dosahují přesnosti kolem 500 m, což může být na dálnici či venkově postačující.

Asistence na cestách

Lokalizace může pomoci i řidičům při orientaci na mapě. Podle odhadnuté polohy je možné určit přibližné místo, kde se řidič nachází.

Informační služby

Za pomoci lokalizace je možné upozorňovat vlastníky mobilních telefonů prostřednictvím SMS zpráv na zajímavé informace v okolí (turistické zajímavosti, nejbližší restauraci, ubytovací zařízení).

Zvýhodněné volání

Mobilní operátor může svým zákazníkům nabídnout výhodnější tarify, pokud volají z určitého geografického místa. Například voláním z okolí domova za cenu volání z pevné linky může zákazníky motivovat, aby dávali ve všech případech přednost mobilnímu telefonu. Operátor také může nabídnout například levné volání pro hovory v rámci firmy.

Sledování trajektorie

Metody lokalizace je možné využít při sledování trajektorie automobilů nebo i jiných dopravních prostředků. Při použití metod na základě síly signálu bude získaná trajektorie pouze přibližná, ale dá se z ní alespoň zjistit, v jakých městech či městských čtvrtích se automobil nacházel, a v jaký čas se pohyboval. Toto lze využít například pro monitorování zaměstnanců na služebních cestách nebo ke sledování pachatele v případě odcizení vozidla.

Podrobný popis možných využití lokalizace je uveden v [4].

1.5 Výzkumné a vývojové centrum RDC

Informace byly převzaty z internetových stránek centra [6].

Poznámka: Společnost Oskar změnila své jméno na Vodafone.

Co je RDC?

Výzkumné a vývojové centrum, podporující studenty a nadšence do bezdrátových technologií.



Oskar společně s dvěma dalšími společnostmi (Ericsson a České vysoké učení technické v Praze) založil v únoru 2001 Výzkumné a vývojové centrum (Research & Development Center) - RDC. Oskar vybavil centrum GSM technologií od společnosti Ericsson, ČVUT poskytlo své prostory, a tak vznikl "4. GSM operátor" již nic nebránilo. Vzniklo tím prostředí, které je studentům k dispozici pro výzkumné, vývojové a vzdělávací činnosti v oblasti bezdrátových služeb.

Co je cílem RDC?

Výzkumné a vývojové centrum nabízí standardizované interní procesy pro tvorbu, vývoj, testování a hodnocení produktů a služeb. Cílem centra je vytvořit, udržovat a rozvíjet komunitu lidí se společným technickým zájmem, která získá možnost dalšího vzdělávání a zapojí se do národních a mezinárodních projektů a aktivit.

Pro realizaci projektů a grantů v rámci Centra je k dispozici plně funkční síť GSM. Díky ní mohou v centru probíhat zajímavé analýzy a testy.

1.6 Model šíření signálu

Projekt Guesstor Server si kladl za cíl umožnit lokalizaci mobilního telefonu, aniž by bylo potřeba znát informace o zástavbě v prostředí, kde se mobilní telefon nachází. Vývojáři Guesstor Serveru na základě [2] odvodili model šíření signálu, který zástavbu města nahrazuje souhrnně pomocí většího útlumu signálu. Za cenu nepřesného modelu šíření signálu je možné používat lokalizační algoritmy i v místech, kde nejsou k dispozici přesné mapy architektury.

V libovolném prostředí platí, že velikost přijatého signálu klesá se zvětšující se vzdáleností od vysílače. Míru závislosti určuje bezrozměrný činitel útlumu prostředí α . Referenční vzdálenost d_0 je možno zvolit libovolně, musí být ovšem zvolena pevně.

$$R_x(d) = R_x(d_0) \cdot \left(\frac{d_0}{d}\right)^\alpha$$

kde

d vzdálenost mezi vysílačem a přijímačem v metrech

d_0 referenční vzdálenost v metrech

α činitel útlumu prostředí (bezrozměrný)

$R_x(d)$ přijatá energie ve vzdálenosti d ve wattech

$R_x(d_0)$ přijatá energie ve vzdálenosti d_0 ve wattech

Hodnota koeficientu prostředí α se pohybuje od 2 pro volný prostor do 6 pro hustou zástavbu velkoměsta. Referenční velikost přijatého signálu $R_x(d_0)$ můžeme ve vztahu nahradit pomocí výkonu zdrojového vysílače a útlumu $L(d_0)$.

$$R_x(d) = \frac{PWR}{L(d)}$$

d vzdálenost od vysílače v metrech

$L(d)$ útlum ve vzdálenosti d (bezrozměrný)

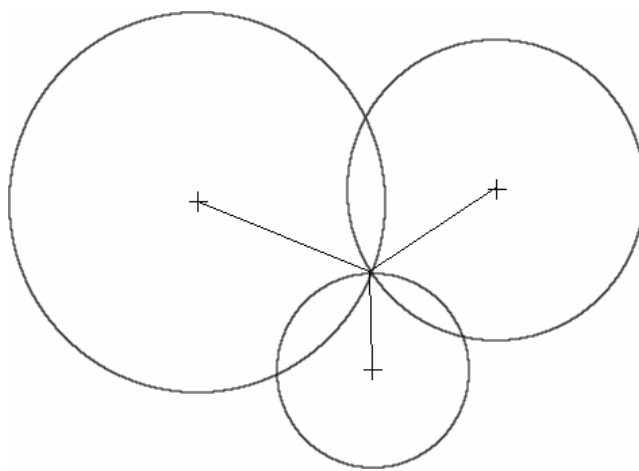
PWR vysílací výkon ve wattech

R_x přijatý výkon ve wattech

Hodnota útlumu $L(d)$ není závislá na výkonu zdrojového vysílače. Hodnotu lze dopředu vypočítat podle parametrů vysílače nebo experimentálně zjistit pro dané prostředí.

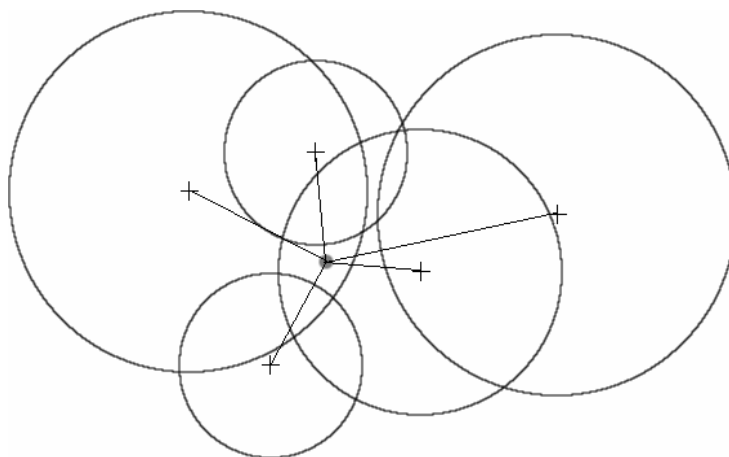
1.7 Princip lokalizačních algoritmů

Všechny lokalizační algoritmy uvedené v této práci používají stejný princip pro odhadnutí polohy, ve které bylo provedeno měření. Nejdříve se za pomoci naměřených hodnot a modelu šíření signálu určí předpokládaná vzdálenost místa měření od vysílače BTS. Následně se sestrojí pomyslné kružnice se středem ve vysílači a poloměrem odpovídajícím předpokládané vzdálenosti. V ideálním případě se všechny kružnice protnou v jediném místě. V tomto průsečíku se nachází místo, kde bylo provedeno měření.



Určení polohy v ideální případě

V reálném případě se však kružnice v jednom místě neprotnou, protože vlivem nedokonalého modelu šíření signálu je vzdálenost od vysílače určena pouze přibližně. Lokalizační algoritmus musí polohu měření odhadnout na základě vzdáleností od vysílačů, přestože vzdálenosti jsou někdy značně nepřesné.



Určení polohy v reálném případě

2 Používání programu

2.1 Požadavky pro spuštění programu

- Procesor alespoň 500 MHz
- Alespoň 20 MB volného místa na disku
- Pro spuštění je potřeba mít nainstalováno Java Runtime Environment alespoň ve verzi 1.4
- Pro kompilaci zdrojových kódů je potřeba mít nainstalováno Java Development Environment

2.2 Instalace a spuštění

Instalace

Na médiu se naleznete adresáře `Guesstor` a `GuesstorClient`, program se nachází v těchto adresářích. Program není potřeba nijak instalovat, stačí překopírovat adresáře `Guesstor` a `GuesstorClient` na pevný disk.

Spuštění pod systémem Windows

Guesstor Server spustíte pomocí dávkového souboru `run.bat` v adresáři `Guesstor`. Program **Guesstor Client** spustíte pomocí dávkového souboru `run.bat` v adresáři `GuesstorClient`. Pro správnou funkci programu **Guesstor Client** je potřeba, aby před jeho spuštěním již běžel **Guesstor Server**.

Spuštění pod systémem Unix nebo Linux

Guesstor Server spustíte pomocí dávkového souboru `run.sh` v adresáři `guesstor`. Program **Guesstor Client** spustíte pomocí dávkového souboru `run.sh` v adresáři `guesstorClient`. Pro správnou funkci programu **Guesstor Client** je potřeba, aby před jeho spuštěním již běžel **Guesstor Server**.

Pozor, pod systémy unixového typu může být zapotřebí přidat právo pro čtení a spuštění k souborům `run.sh`, aby je bylo možné spustit.

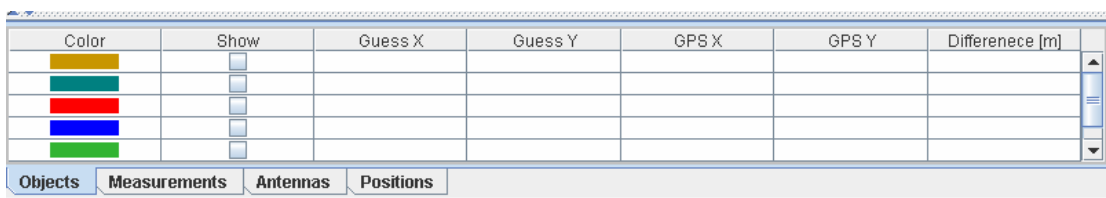
Možné problémy a jejich řešení






Nejčastějším problémem je opomenutí spustit **Guesstor Server** před spuštěním **Guesstor Clienta**. Další častou chybou je spuštění **Guesstor Serveru** více než jednou. V takovém případě vznikají problémy s přístupem do databáze. V obou těchto případech nemusí **Guesstor Client** pracovat správně. Pro nápravu těchto problémů ukončete všechny procesy **Guesstor Clienta** i **Serveru** a postupujte znova podle pokynů pro spuštění. **Guesstor Server** vypisuje chybové hlášky do konzole. **Guesstor Client** zobrazuje chybové hlášky v dialogovém okně a do konzole.

2.3 Ovládání programu Guesstor Client

Sledování trajektorie

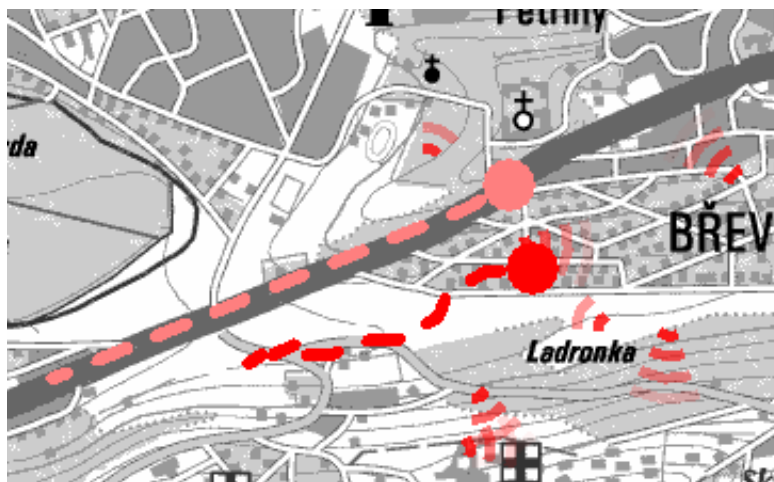
- a. Pokud byl program Guesstor Client správně spuštěn, v dolní části okna aplikace se objeví tabulka se seznamem dostupných objektů. Pokud tabulka v dolní části okna neobsahuje žádné řádky, spojení s Guesstor Serverem se nezdařilo. V takovém případě program vypíše chybové hlášení.



Color	Show	Guess X	Guess Y	GPS X	GPS Y	Diference [m]
	<input type="checkbox"/>					
	<input type="checkbox"/>					
	<input type="checkbox"/>					
	<input type="checkbox"/>					
	<input type="checkbox"/>					






tabulka dostupných objektů

- b. Pomocí myši zaškrtněte ve sloupci Show checkbox u jednoho z objektů. Tím se zahájí sledování trajektorie zvoleného objektu.
- c. Na mapě se začne vykreslovat trajektorie zvoleného objektu. Tmavou barvou se vykresluje trajektorie zjištěná pomocí úrovní signálů z vysílačů BTS. Světlou barvou se vykresluje trajektorie zjištěná pomocí systému GPS. Na mapě se též zobrazí vysílače BTS, na základě kterých byla určena poloha.



trajektorie objektu

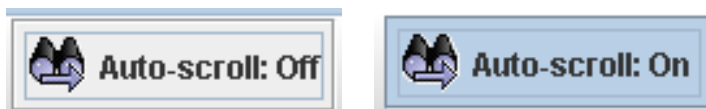
- d. V tabulce objektů se u zvoleného objektu doplní souřadnice pozice určené pomocí úrovní signálů (Guess X, Guess Y) a souřadnice pozice určené pomocí systému GPS (GPS X, GPS Y). Souřadnice v této tabulce jsou uvedeny ve formátu UTM. Rozdíl v metrech mezi těmito dvěma souřadnicemi je zobrazen v posledním sloupci tabulky.

Color	Show	Guess X	Guess Y	GPS X	GPS Y	Diference [m]
	<input type="checkbox"/>					
	<input type="checkbox"/>					
	<input checked="" type="checkbox"/>	458090	5549578	458356	5549683	285
	<input type="checkbox"/>					
	<input type="checkbox"/>					

Objects Measurements Antennas Positions

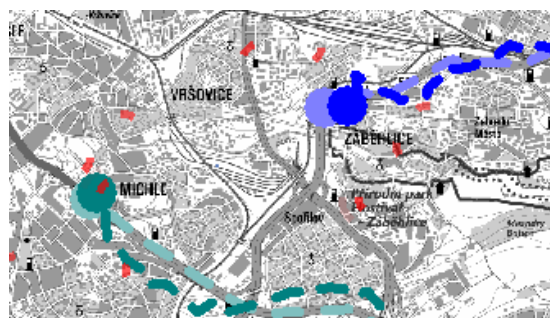
souřadnice objektu v tabulce

- e. Pomocí myši zaškrtněte ve sloupci Show checkbox u dalšího z objektů. Tím se zahájí sledování trajektorie druhého objektu. Trajektorie zvoleného objektu se začne vykreslovat a v tabulce Objects se zobrazí informace o jeho souřadnicích.
- f. Jakmile bylo zahájeno sledování dalšího objektu, na mapě se pravděpodobně zobrazila jiná část Prahy a trajektorie původního objektu přestala být vidět.
- g. V horní části okna se nachází tlačítko Auto-scroll. Pomocí tohoto tlačítka může uživatel nastavit zda chce, aby na mapě byly vidět stále všechny objekty. Implicitně je takovýto režim vypnutý, stisknutím tlačítka Auto-scroll se tento režim zapne.



Tlačítko Auto-scroll

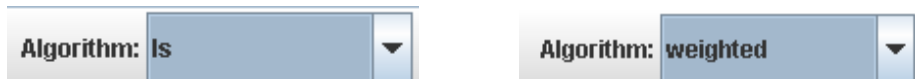
- h. V režimu Auto-scroll program Guesstor Client neustále mění zobrazenou část Prahy tak, aby všechny sledované objekty byly vidět v okně aplikace. V případě, kdy jsou objekty od sebe tak vzdálené, že není možné je všechny umístit do okna aplikace zároveň, se mapa oddálí a tím zobrazení objektů umožní.



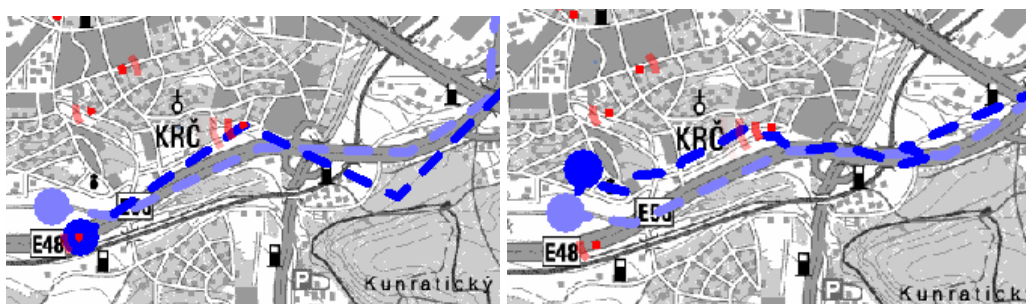
oddálená mapa s trajektoriemi dvou objektů

Změna lokalizačního algoritmu

- i. Pomocí listboxu v levém horním rohu okna aplikace je možné změnit lokalizační algoritmus, který bude použit na určení trajektorie podle úrovně signálů z vysílačů BTS. Vyzkoušejte si změnit lokalizační algoritmus z původního „ls“ na algoritmus „weighted“. Všimněte si, že na mapě zobrazené trajektorie se přepočtou podle nového algoritmu a překreslí.



změna lokalizačního algoritmu



změna vypočtené trajektorie

Práce s tabulkou provedených měření

- j. Před zahájením práce s tabulkou provedených měření je doporučeno v tabulce Objects vypnout sledování všech objektů. Zobrazené trajektorie mohou mapu při další práci znepřehledňovat. Pokud nechcete vypnout sledování objektů, vypněte alespoň režim Auto-scroll.
- k. Pomocí záložek s tabulkami ve spodní části aplikačního okna se přepněte na tabulku Measurements.

Timestamp	Object	Antennas	GPS X	GPS Y
23.5.2006 20:01:36		7	456205	5542811
23.5.2006 20:01:38		6	466259	5546480
23.5.2006 20:01:38		5	456353	5542791
23.5.2006 20:01:41		6	466124	5546415
23.5.2006 20:01:41		5	456501	5542770

Objects Measurements Antennas Positions

tabulka Measurements

- l. Tabulka Measurements obsahuje všechna měření úrovně signálu, která byla provedena pro sledování trajektorie objektů. Pokud je tabulka Measurements prázdná, nebyl zatím sledován žádný objekt.
- m. Pomocí myši vyberte libovolný řádek v tabulce Measurements. Vybraný řádek se zvýrazní modrou barvou.

- n. Na mapě se zobrazí červenou barvou všechny vysílače BTS, od kterých byl přijat signál ve vybraném měření.



měření zobrazené na mapě

- o. Na mapě se zobrazí zelený čtverec, který označuje místo, ve kterém se objekt nacházel v okamžiku měření. Toto místo bylo změřeno za pomoci GPS.
- p. Dále se na mapě zobrazí modrý kruh, který označuje místo vypočtené pomocí lokalizačního algoritmu na základě síly signálu z vysílačů BTS.
- q. Pomocí listboxu v levé horním rohu okna aplikace zvolte jiný lokalizační algoritmus. Výsledek se projeví změnou polohy modrého kruhu. Podle rozdílu mezi polohou modrého kruhu a zeleného čtverce lze posoudit přesnost algoritmu.
- r. Pomocí ovládacího prvku v levém horním rohu okna zvolte jiný lokalizační algoritmus. Výsledek se projeví změnou polohy modrého kruhu.

Práce s tabulkou vysílačů BTS

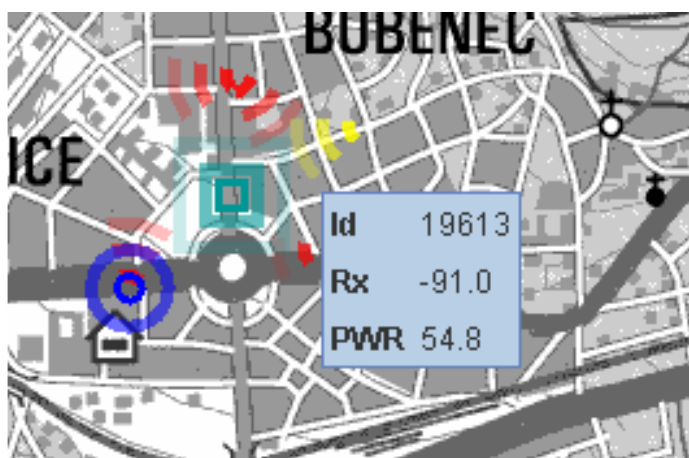
- s. Pomocí záložek s tabulkami ve spodní části aplikačního okna se přepněte na tabulku Antennas. Tabulka Antennas zobrazuje informace o naměřených signálech od vysílačů BTS pro vybrané měření. Pokud je tabulka Antennas prázdná, v tabulce Measurements nebylo vybráno žádné měření.

Id	X	Y	Rx [dB]	PWR [dBm]	Azimuth [°]	Beam width [°]
60233	455183	5548204	-84,56	51	208	45
60223	455792	5547713	-96,98	50	1	45
60261	456031	5549438	-98,58	50	72	45
60280	454629	5547996	-91,09	56	313	45
60191	454723	5547811	-96,55	52	8	45

Objects Measurements **Antennas** Positions

tabulka Antennas

- t. Pokud v tabulce Antennas myší vyberete nějaký řádek, vysílač, jehož údaje odpovídají vybranému řádku, se na mapě zvýrazní žlutou barvou.
- u. Funguje i postup opačný. Pokud na mapě stisknete levé tlačítko myši nad některým z vysílačů BTS, tento vysílač se zvýrazní žlutou barvou a v tabulce Antennas se vybere odpovídající řádek, který obsahuje informace o daném vysílači.
- v. Pokud umístíte kurzor myši na mapě nad některým z vysílačů BTS, za chvíli se zobrazí stručná informace o daném vysílači. Tímto způsobem lze zjistit identifikátor vysílače (Id), vysílací výkon (PWR) a míru přijatého signálu (Rx).



informace o vysílači zobrazená na mapě

Srovnávání lokalizačních metod

- w. Pomocí záložek s tabulkami ve spodní části aplikačního okna se přepnete na tabulku Positions. Tabulka Positions umožňuje srovnávat přesnost lokalizačních algoritmů pro měření vybrané v tabulce Measurements. Pokud je tabulka Positions prázdná, v tabulce Measurements nebylo vybráno žádné měření.

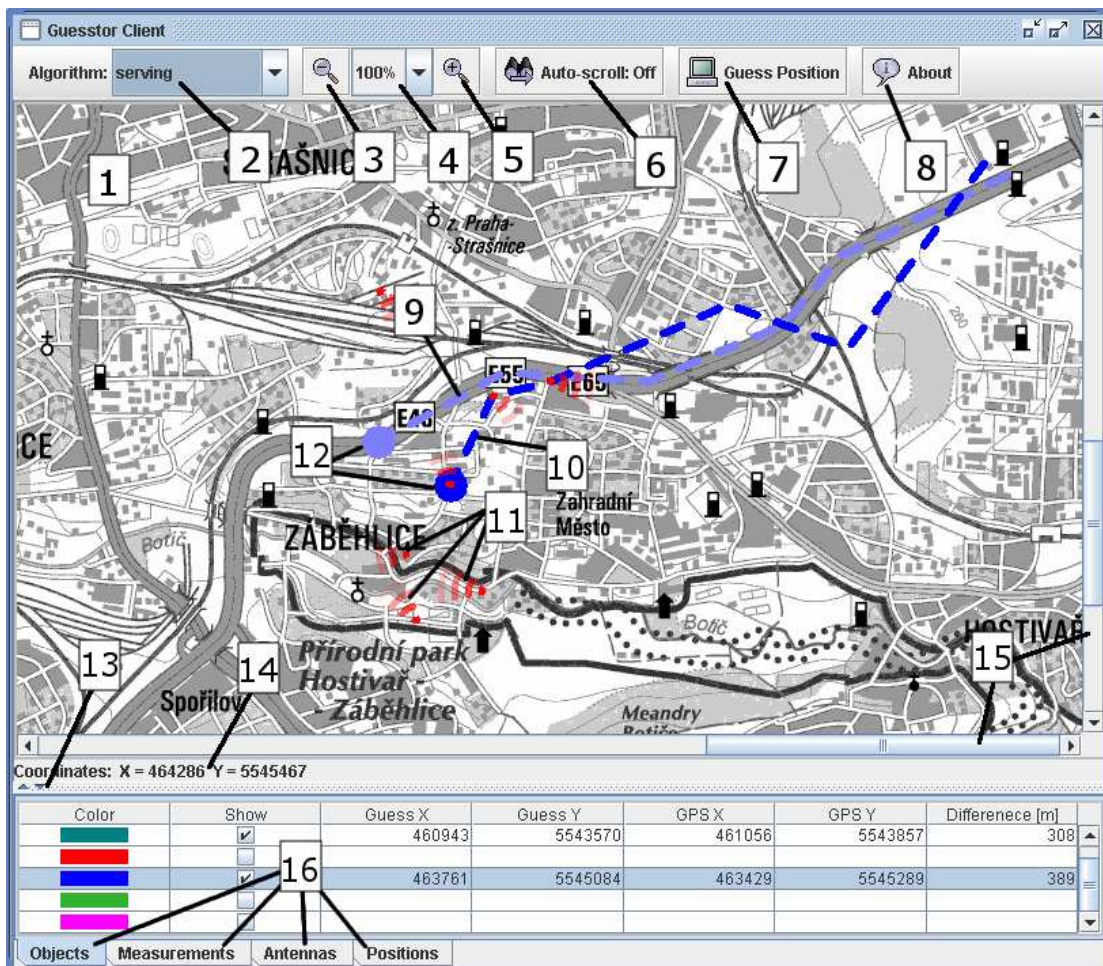
Compute	Name	Difference [m]	X	Y
<input checked="" type="checkbox"/>	GPS	0	455556	5548519
<input checked="" type="checkbox"/>	ls	104	455475	5548585
<input checked="" type="checkbox"/>	servng	488	455183	5548204
<input checked="" type="checkbox"/>	weighted	245	455423	5548313
<input checked="" type="checkbox"/>	iterative	92	455474	5548563

Objects Measurements Antennas Positions

Tabulka Positions

- x. Pomocí tabulky Positions můžete přehledně porovnat rozdíl polohy určené na základě síly signálu od polohy změřené pomocí GPS pro všechny dostupné lokalizační algoritmy.

2.4 Popis ovládacích prvků



1. Mapa

Mapa Prahy umožňuje uživateli vytvořit si lepší představu o tom, kde se nachází vysílače, a kudy vede trajektorie objektu.

2. Volba algoritmu

Pomocí tohoto ovládacího prvku je možné nastavit, jaký lokalizační algoritmu se použije pro odhad polohy. K dispozici jsou čtyři lokalizační algoritmy. Jejich stručný popis je uveden dále v této kapitole.

3. Tlačítko pro oddálení mapy

Po stisknutí tohoto tlačítka se mapa oddálí. Pokud mapu již není možné více oddálit, tlačítko je zašedlé.

4. Informace o oddálení či přiblížení mapy

Pokud je mapa Prahy zobrazena tak, že jeden pixel mapové předlohy odpovídá jednomu pixelu na obrazovce, je v tomto listboxu uvedena hodnota 100%. Pokud je zobrazená mapa zmenšena, pak je zde uvedeno číslo nižší než 100%. Pokud je zobrazená mapa naopak zvětšena, uvedené číslo je větší než 100%.

5. Tlačítko pro přiblížení mapy

Po stisknutí tohoto tlačítka se mapa přiblíží. Pokud mapu již není možné více přiblížit, tlačítko je zašedlé.

6. Tlačítko Auto-scroll

Pomocí tohoto tlačítka je možné zapnout či vypnout režim Auto-scroll. V režimu Auto-scroll se zobrazuje na mapě vždy taková část Prahy, na které jsou vidět všechny objekty. V případě potřeby se mapa automaticky oddálí či přiblíží.

7. Tlačítko pro opakování výpočtu

Toto tlačítko využívají převážně vývojáři Guesstor Serveru. Po jeho stisknutí se opakuje požadavek na server pro výpočet odhadnuté polohy. Je užitečné v případě, že server změnil chování (v důsledku zásahu programátora) a je potřeba zobrazit, jaký efekt měla změna.

8. Zobrazení informací o programu

Toto tlačítko zobrazí dialog se stručnou informací o programu a o jeho autorovi.

9. Trajektorie objektu určená pomocí systému GPS

Systém GPS umožňuje změřit polohu mnohem přesněji než lokalizační algoritmy, které polohu určují na základě intenzity signálu od vysílačů BTS. Trajektorie určená pomocí GPS vystihuje skutečnou trajektorii přesněji. Na mapě je trajektorie určená pomocí GPS zobrazena pro srovnání s trajektorií určenou pomocí lokalizačních algoritmů. Trajektorie určená pomocí systému GPS se na mapě zobrazuje světlejší barvou.

10. Trajektorie objektu určená pomocí lokalizačního algoritmu

Trajektorie určená pomocí lokalizačních algoritmů se na mapě zobrazuje tmavší a sytější barvou než trajektorie určená pomocí systému GPS.

11. Vysílače použité pro výpočet

Na mapě se též zobrazí vysílače, kterých je využito pro výpočet v lokalizačním algoritmu. Vysílače se zobrazují jako červené výseče. Jejich velikost je úměrná vysílacímu výkonu.

12. Aktuální poloha objektu

Aktuální poloha objektu je na mapě zobrazena jako blikající kruh. Poloha vypočtená pomocí lokalizačních algoritmů je zobrazena tmavší barvou. Poloha určená pomocí systému GPS je zobrazena světlejší barvou. Barevný odstín závisí na tom, o jaký objekt se jedná. Poloha určená pomocí lokalizačních algoritmů se obvykle liší od polohy určené pomocí systému GPS.

13. Oddělovací panel

Pomocí tohoto ovládacího prvku je možné nastavit jaká část okna aplikace bude zobrazovat mapu a jaká tabulky. Oddělovací panel může uživatel levým tlačítkem myši uchopit a přesunout do požadované polohy. Pokud uživatel stiskne levé tlačítko myši na některou z dvojice šipek při levé straně panelu, aplikace se přepne do režimu, ve kterém je zobrazena pouze mapa nebo pouze tabulka. (podle šipky, na které bylo stisknuto levé tlačítko)

14. UTM souřadnice

Ve stavovém řádku pod mapou se zobrazují zeměpisné souřadnice ve formátu UTM (Universal Transverse Mercator). Zobrazená souřadnice odpovídá místu, které se nachází pod kurzorem myši. Pokud uživatel pohne kurzorem myši na mapě, změní se i souřadnice ve stavovém řádku.

15. Posuvníky pro mapu

Pomocí posuvníků je možné pohybovat s mapou. Další možností posouvání mapy je pomocí myši. Pokud je levé tlačítko myši stisknuto kdekoli na mapě a následně je myš za stálého držení levého tlačítka přesunuta za okraj komponenty s mapou, mapa se bude stále posouvat, dokud nenarazí na okraj nebo dokud není puštěno levé tlačítko. Mapu lze též posouvat za pomoci kolečka myši.

16. Záložky s tabulkami

Pomocí těchto záložek je možné zvolit, která z tabulek je aktuálně viditelná. K dispozici jsou tabulky se seznamem dostupných objektů, se seznamem měření, s informacemi o odhadnutých pozicích nebo se seznamem vysílačů BTS.

2.5 Popis tabulek

Tabulka Measurements

Pomocí tabulky Measurement může uživatel prohlížet obsah databáze provedených měření. Jakmile pomocí myši nebo klávesnice vybere nějaký řádek tabulky, vybrané měření se zobrazí na mapě. Během vývoje lokalizačního software bylo po Praze provedeno větší množství měření úrovně signálů na mnoha místech. V této práci je obsažen pouze zlomek z nich, protože polohy a parametry vysílačů představují citlivé informace pro mobilního operátora a jejich zveřejnění není žádoucí. Jelikož se mnohé parametry vysílačů v průběhu času mění podle potřeb operátora, aktuální parametry vysílačů budou již patrně jiné než parametry zobrazené v databázi této práce.

Timestamp	Object	Antennas	GPS X	GPS Y
23.5.2006 20:01:36		7	456205	5542811
23.5.2006 20:01:38		6	466259	5546480
23.5.2006 20:01:38		5	456353	5542791
23.5.2006 20:01:41		6	466124	5546415
23.5.2006 20:01:41		5	456501	5542770

Objects Measurements Antennas Positions

tabulka Measurements

Význam sloupců

Timestamp – čas, kdy bylo měření přidáno do tabulky

Object – označuje objekt, ke kterému se měření vztahuje

Antennas – počet vysílačů BTS, od kterých byl změřen signál v tomto měření

GPS X, GPS Y – zeměpisné souřadnice místa, ve kterém bylo měřeno, ve formátu UTM, zjištěné pomocí systému GPS

Tabulka Positions

Pomocí tabulky Positions může uživatel programu Guesstor Client vzájemně porovnávat výsledky, které jsou výstupem jednotlivých lokalizačních algoritmů. V této tabulce má přehledně uspořádány všechny pozice, které byly odhadnuty, a může také srovnávat rozdíl mezi odhadnutou pozicí a pozicí změřenou pomocí systému GPS.

První řádek této tabulky má speciální význam. Tento řádek neurčuje výsledek žádného lokalizačního algoritmu, ale zobrazuje informace o skutečné poloze měření změřené pomocí GPS. Proto bude vždy ve sloupci Difference uvedena hodnota 0. V prvním řádku není možné pomocí myši či klávesnice změnit stav checkboxu. Tento checkbox pouze informuje o tom, zda je dostupná informace o poloze měření zjištěné pomocí systému GPS.

Compute	Name	Difference [m]	X	Y
<input checked="" type="checkbox"/>	GPS	0	455556	5548519
<input checked="" type="checkbox"/>	ls	104	455475	5548585
<input checked="" type="checkbox"/>	serving	488	455183	5548204
<input checked="" type="checkbox"/>	weighted	245	455423	5548313
<input checked="" type="checkbox"/>	iterative	92	455474	5548563

Objects Measurements Antennas **Positions**

tabulka Positions

Význam sloupců

Compute – Pomocí tohoto sloupce si může uživatel nastavit, zda se má provádět výpočet u daných algoritmů. Implicitně jsou výpočty u všech algoritmů zapnuté. Uživatel ocení možnost vypnout výpočet některého z algoritmů v případě, kdy je tento výpočet časově náročný a jeho výsledek uživatele nezajímá.

Name – název algoritmu

Difference – rozdíl v metrech mezi odhadnutou polohou a polohou určenou pomocí GPS

X, Y – zeměpisné souřadnice odhadnutého místa, ve formátu UTM

Tabulka Antennas

Tabulka Antennas zobrazuje informace o vysílačích BTS, od kterých byl změřen signál v měření zvoleném v tabulce Measurements. Význam tabulky Antennas spočívá hlavně v možnosti zobrazení vstupních údajů, podle kterých byla určena v lokalizačním algoritmu poloha. Vývojáři lokalizačních algoritmů si pomocí této tabulky mohou ověřit, zda jejich algoritmus vrací očekávané výstupy.

Id	X	Y	Rx [dB]	PWR [dBm]	Azimuth [°]	Beam width [°]
60233	455183	5548204	-84,56	51	208	45
60223	455792	5547713	-96,98	50	1	45
60261	456031	5549438	-98,58	50	72	45
60280	454629	5547996	-91,09	56	313	45
60191	454723	5547811	-96,55	52	8	45

Objects Measurements Antennas Positions

tabulka Antennas

Význam sloupců

Id – identifikátor vysílače BTS

X, Y – zeměpisné souřadnice udávající polohu vysílače BTS

Rx – intenzita přijatého signálu od vysílače BTS

PWR – vysílací výkon dané základnové stanice

Azimuth – Tento parametr udává jakým směrem je vysílač natočen. Údaj je zobrazen ve formátu azimutu. Azimut je orientovaný úhel, který svírá určitý směr od směru severního. Úhel je orientovaný, zaleží tedy na směru měření úhlu - měří se po směru pohybu hodinových ručiček, tj. od severu k východu. Měří se ve stupních.






Beam width – Tento parametr určuje vyzařovací úhel daného vysílače BTS. Mobilní operátor bohužel nedodal informace o vyzařovacích úhlech vysílačů, takže v současné době je v databázi u všech vysílačů hodnota 45°, která je obvyklá u směrového vysílání. U všesměrového vysílání je vyzařovací úhel 360°.

Tabulka Objects

Pomocí tabulky Objects může uživatel nastavit, jaké trajektorie pohybu objektů chce sledovat na mapě. V této tabulce také vidí informace o zeměpisných souřadnicích aktuální polohy objektu a může sledovat rozdíl mezi skutečnou a odhadnutou souřadnicí.

První řádek této tabulky obsahuje objekt s oranžovou barvou. Tento objekt simuluje pohyb automobilu, který byl naměřen v roce 2005. Vychází tedy ze skutečných měření. Trajektorie ostatních objektů je generována uměle pomocí generátoru měření podle modelu šíření signálu. Tyto trajektorie ze skutečných měření nevyhází. Slouží pouze pro demonstraci funkčnosti programu. Více reálních trajektorií nebylo možné uvést, protože mobilní operátor neposkytl informace o poloze a parametrech vysílačů v jiných částech Prahy než v Dejvicích.

Trajektorii objektu také ovlivňuje volba algoritmu. Pokud pomocí ovládacího prvku pro volbu algoritmu bude změněn algoritmus. Dojde k přepočtení již vykreslené trajektorie a k jejímu překreslení na mapě.

Color	Show	Guess X	Guess Y	GPS X	GPS Y	Diferencece [m]
	<input type="checkbox"/>					
	<input type="checkbox"/>					
	<input checked="" type="checkbox"/>	458090	5549578	458356	5549683	285
	<input type="checkbox"/>					
	<input type="checkbox"/>					

Objects Measurements Antennas Positions

tabulka Objects

Význam sloupců

Color – barva, kterou se trajektorie daného objektu vykresluje na mapě

Show – Pomocí checkboxů v tomto sloupci si může uživatel nastavit trajektorie jakých objektů chce zobrazit na mapě. Zobrazují se pouze trajektorie těch objektů, které mají příslušný checkbox zaškrtnutý.

Guess X, Guess Y – zeměpisné souřadnice ve formátu UTM odhadnuté pomocí lokalizačního algoritmu

GPS X, GPS Y – zeměpisné souřadnice zjištěné pomocí systému GPS ve formátu UTM

Difference – rozdíl mezi polohou určenou pomocí systému GPS a polohou odhadnutou na základě lokalizačního algoritmu

2.6 Ovládání pomocí klávesnice

Celý program Guesstor Client je možné ovládat za pomoci klávesnice. Tento způsob ovládání ocení zejména uživatelé, kteří nemohou nebo nechtějí používat myš.

Pomocí klávesy Tab, se uživatel může přepínat mezi čtyřmi částmi Guesstor Clienta: panelem nástrojů, mapou, tabulkou a záložkami s tabulkami. Vybraná část se orámuje černou barvou. Poté může uživatel s danou částí manipulovat, nebo se klávesou Tab přepnout na část následující.

Ovládání panelu nástrojů

Pomocí kurzorových šipek se uživatel může pohybovat mezi jednotlivými tlačítky v panelu nástrojů. Pomocí klávesy mezerník pak tlačítko může stisknout.

Ovládání mapy

Pokud je režim Auto-scroll vypnutý, uživatel může mapu posouvat pomocí kurzorových šipek. V režimu Auto-scroll se mapa posouvá automaticky a jiným způsobem se posouvat nedá.

Ovládání tabulky

Uživatel se může pohybovat v tabulce pomocí kurzorových šipek. Pokud tabulka obsahuje v některém ze sloupců checkbox, může uživatel změnit jeho stav pomocí klávesy mezerník.

Ovládání záložek s tabulkami

Uživatel se může pomocí záložek přepínat mezi jednotlivými tabulkami. V záložkách se uživatel pohybuje pomocí kurzorových šipek.

2.7 Stručný popis lokalizačních algoritmů

Tento popis lokalizačních algoritmů vznikl na základě osobních rozhovorů s vývojáři lokalizačních algoritmů, během kterých se koordinoval vývoj programů Guesstor Server a Guesstor Client. Lokalizační algoritmy vymysleli a implementovali studenti Českého vysokého učení technického v Praze Martin Olrich, Zbyňěk Novák a Michal Pravda. Jedná se pouze o neformální popis, který osvětluje princip fungování těchto algoritmů.

Algoritmus obsluhujícího vysílače (Serving cell algorithm)

Jedná se o velmi jednoduchý algoritmus, který byl v Guesstor serveru naprogramován jako první. Algoritmus vychází z toho, že přijímač dostává nejsilnější signál z vysílače, ke kterému je nejbližší. Pokud je v okolí dostatečné množství vysílačů, je možné pozici přijímače se značnou nepřesností odhadnout jako pozici vysílače, ze kterého má nejsilnější signál. Mobilní telefony navazují spojení s vysílačem, pro který mají nejsilnější signál. Takovýto vysílač pak telefon obsluhuje (Serving cell). Z toho vyplývá název algoritmu Serving cell algorithm. Navzdory jeho mimořádné jednoduchosti algoritmus má tu výhodu, že nikdy neurčí polohu špatně s mimořádně vysokou odchylkou. V grafickém rozhraní Guesstor Clienta můžeme algoritmus vybrat pod názvem „serving“.

Algoritmus využívající váhy (Weighted algorithm)

Tento algoritmus nejdříve pomocí modelu šíření signálu určí na základě intenzity přijatého signálu a vysílacího výkonu předpokládanou vzdálenost mobilního telefonu od vysílače BTS. Na základě této vzdálenosti pak vypočte takzvanou váhu daného vysílače. Váha je kladné reálné číslo. Vysílače, které mají předpokládanou vzdálenost malou mají větší váhu než vysílače s velkou předpokládanou vzdáleností. Výsledná poloha se určí jako vážený průměr poloh vysílačů, ve kterém se používají příslušné váhy. Díky tomu, že váhy vysílačů jsou vždy nezáporné, odhadnuté místo bude vždy ležet v konvexním obalu vysílačů, od kterých má přijímač signál. Tím je vyloučena možnost nějaké extrémní chyby v poloze. V grafickém rozhraní Guesstor Clienta můžeme algoritmus vybrat pod názvem „weighted“.

Iterativní algoritmus (Iterative algorithm)

Dalším implementovaným lokalizačním algoritmem je Iterativní algoritmus. Na jeho začátku se nejdříve, podobně jako u ostatních algoritmů, pomocí modelu šíření signálu určí na základě intenzity přijatých signálů z vysílačů předpokládaná vzdálenost přijímače od vysílače. Dále je vytvořena funkce, která pro souřadnici mapy určí počet trestných bodů. Trestné body se udělují podle toho, jak se liší vzdálenost mezi souřadnicí a vysílačem od předpokládané vzdálenosti přijímače od vysílače. Toto porovnání se provede pro všechny vysílače a výsledkem je suma trestných bodů pro všechny vysílače. V další fázi se algoritmus snaží zjistit, kde má daná funkce minimum. K tomu je použito techniky „slepého horolezce“. Nejdříve se podobně jako v algoritmu využívajícím váhy určí místo uvnitř konvexního obalu vysílačů. V tomto místě se pomocí výše popsané funkce vypočte množství trestných bodů. Dále se algoritmus snaží pohybovat souřadnicí místa různými směry a zjišťuje, zda se tím množství trestných bodů snížilo. Vzdálenost, o kterou se souřadnice posouvá, se postupně snižuje, aby se tím dosáhlo větší přesnosti v určení minima. Tento algoritmus dosahuje také dobrých výsledků. V grafickém rozhraní Guesstor Clienta můžeme algoritmus vybrat pod názvem „iterative“.

Algoritmus využívající metodu nejmenších čtverců (Least square algorithm)

Tento algoritmus využívá sofistikovaným způsobem algebry k učení minima a tím se odlišuje od Iterativního algoritmu, který využívá techniky „slepého horolezce“. Nejprve se na základě modelu šíření signálu a informací o vysílacím výkonu a přijatém signálu určí předpokládaná vzdálenost mezi vysílačem a přijímačem. Následně se pomocí algebraických metod určí takové místo na mapě, pro které je minimální součet druhých mocnin rozdílů skutečné vzdálenosti a předpokládané vzdálenosti. Přestože implementace tohoto algoritmu byla ze všech nejnáročnější, výsledky, které podává nejsou příliš přesné. V některých případech se stává, že odhadnutá pozice je od skutečné pozice velmi vzdálena. (je mimo konvexní obal vysílačů) Dle mého názoru toto nastává v případě, kdy je přijímač zastíněn zástavbou a podle slabého příjmu a modelu šíření signálu dojde algoritmus k závěru, že se od vysílačů nachází ve velké vzdálenosti. Následně pak umístí odhadnutou pozici daleko od vysílačů BTS. Dalším problémem je případ, kdy se přijímač nachází v těsné blízkosti u vysílače nebo pod vysílačem. V takovémto případě je také příjem signálu slabý, protože vysílač vysílá signál především do stran a nikoliv dolů. Algoritmus to však mylně interpretuje tak, že se od vysílače nachází někde hodně daleko. V případech kdy k těmto problémům nedochází dává algoritmus velmi dobré výsledky. Bohužel v průměrném případě jsou výsledky horší než u Algoritmu obsluhujícího vysílače. V grafickém rozhraní Guesstor Clienta můžeme algoritmus vybrat pod názvem „ls“.

Podrobný popis metody nejmenších čtverců je možné nalézt v [5].

3 Popis použitých technologií

3.1 Global Positioning System (GPS)

Zvýrazněný text byl převzat z encyklopedie [7] a upraven.

Global Positioning System, zkráceně *GPS*, je vojenský navigační systém provozovaný Ministerstvem obrany Spojených států amerických, který dokáže s několikametrovou přesností určit pozici kdekoliv na Zemi.

Vývoj *GPS* byl zahájen v roce 1973 a po postupném rozšiřování se stal plně funkčním a dostupným po celém světě 17. ledna 1994, kdy byla na orbitu umístěna kompletní sestava 24 satelitů. V roce 1983, kdy sovětská stíhačka ve vzdušném prostoru SSSR sestřelila civilní dopravní letadlo Korean Air Flight 007 (KAL 007), přičemž všech 269 lidí na palubě zahynulo, oznámil americký prezident Ronald Reagan, že po dokončení bude *GPS* k dispozici i pro civilní účely. V současné době se systém využívá i v mnoha oborech lidské činnosti, které s armádou nesouvisí, kdy civilní uživatelé mohou k určení polohy používat takzvaný civilní C/A kód.

Princip určení polohy

Zjednodušeně je *GPS* **družicový radiový dálkoměrný** systém. Vysvětlení jednotlivých částí tohoto označení:

- **Dálkoměrný systém** je takový, kdy se poloha nějakého objektu určuje ze vzdáleností od bodů se známou polohou. Např. v krajině lze určit polohu pomocí mapy a dalekohledu, který umí změřit vzdálenost od pozorovaného objektu. Dalekohledem změříme vzdálenost ke dvěma význačným objektům a kružítkem na mapě nakreslíme kolem každého objektu kružnici o změřeném poloměru. Zjišťovaná poloha je zřejmě v jednom z průsečíků obou kružnic.
- **Rádiový systém** pro měření určitého parametru využívá radiových vln. „Rádiový dálkoměrný“ systém k měření vzdálenosti využívá radiových vln takto: Do bodu se známou polohou je umístěn vysílač, který vysílá rádiové vlny s časovými značkami. V bodě, jehož poloha se měří, umístíme přijímač, který porovnává časové značky se svými „hodinami“. Tím je možno změřit zpoždění, tj. jak dlouho trvalo rádiové vlně, než k přijímači dorazila. Protože se rádiové vlny pohybují známou rychlostí (tj. rychlostí světla, která ve vakuu činí 299 792 458 m/s), stačí pro výpočet požadované vzdálenosti vynásobit změřené zpoždění touto rychlostí.
- **Družicový** je systém označován proto, že body se známou polohou jsou družice obíhající Zemi. Aby bylo možno určit polohu družic, musí být v jejich vysílání nejen časové značky, ale i parametry dráhy dané družice.

Kosmický segment GPS

Kosmický segment tvoří 24 družic a tři záložní družice, které obíhají Zemi na šesti drahách (sklon dráhy 55°) vzájemně posunutých o 60° ve výšce 20 200 km nad povrchem Země (oběžná doba je tedy 11h a 58 min - pozemský pozorovatel vidí každý den stejnou dráhu, avšak družice vychází vždy o 4 minuty dříve). Na palubě družic NAVSTAR jsou 3 až 4 velmi přesné (10^{-13}) atomové hodiny (nezbytné pro funkci systému), s cesiovým a rubidiovým oscilátorem, a dále pak detektory, kontrolující dodržování zákazu zkoušek nukleárních zbraní. Navigační signál je vysílán v pásmu L (2000-1000 MHz).

Uživatelský segment GPS

Uživatelský segment tvoří vlastní GPS přijímač, což je přijímač signálu s rozprostřeným spektrem. Přijímače GPS poslouchají tyto signály od minimálně tří a maximálně dvanácti satelitů a z těchto údajů vyhodnocují svojí přesnou pozici, spolu s rychlostí a směrem posunu. K určení polohy postačí signál tří satelitů, pro zjištění nadmořské výšky alespoň čtyř a pro zjištění přesného času pouze jeden satelit. Čím větší počet družic se daří současně zachytit, tím přesnější jsou udávané souřadnice. GPS přijímač obsahuje pseudonáhodný generátor shodný s generátory na satelitech (definováno 32 pseudonáhodných sekvencí), užitečný signál se začne přijímat při synchronizaci generátoru v přijímači s generátorem na satelitu (dochází ke korelaci signálů).

*Od počátku 90. let 20. století je systém zdarma přístupný i pro civilní uživatele po celém světě. Nejprve byla do přijímaného signálu systému zanášena umělá chyba. Toto opatření pod názvem **Selective Availability** (SA) mělo zabránit možnosti navádět vojenské dálkové rakety. SA většinou způsobovalo chyby v rozmezí 10 m horizontálně a 30 m vertikálně. SA bylo k 1. květnu 2000 zrušeno a přesnost zaměření zeměpisných souřadnic běžného civilního uživatele se tak zvýšila na 5 až 10 metrů, za příznivých okolností (otevřený terén) až na 3 metry.*

Přesnost určení polohy

Při určení polohy se nejprve změří vzdálenost ke družicím, dále se vypočtou polohy družic a nakonec je vypočítána poloha přijímače. Z uvedeného postupu vyplývá, že přesnost polohy ovlivňuje:

- *přesnost měření vzdálenosti*
- *přesnost určení polohy družic*
- *vlastní výpočet polohy*

3.2 Global System for Mobile Communications (GSM)

Zvýrazněný text byl převzat z encyklopedie [8] a upraven.

GSM (Globální Systém pro Mobilní komunikaci) je nejpoužívanější standard pro mobilní telefony na světě. GSM telefony používá přes miliardu lidí z více než 200 zemí.

Radiové rozhraní

GSM je buňková síť, což znamená že mobilní telefony se připojují do sítě prostřednictvím nejbližší buňky. GSM síť funguje na několika radiových frekvencích.

Jsou čtyři různé velikosti buněk - Makro, mikro, mikro a dešťníkové buňky. Oblast pokrytí každé buňky se liší podle prostředí. Za makro buňky jsou považovány ty kde je umístěna anténa základové stanice na stožáru nebo na budově nad úrovní střech. Mikro buňky mají anténu umístěnou pod úrovní střech; typické je použití v zastavěných oblastech. Pikobuňky jsou malé buňky s průměrem pár desítek metrů; používají se hlavně uvnitř budov. Na druhou stranu dešťníkové buňky se používají pro pokrytí oblastí ve stínech a na vyplnění mezer mezi buňkami.

Velikost pokrytí závisí na výšce antény, výkonu antény a na podmínkách šíření a pohybuje se od několika stovek metrů až do desítky kilometrů. Největší vzdálenost které se podle specifikace GSM prakticky používá je 35 km. Existuje však koncept rozšířené buňky kde může být oblast dvojnásobná i větší.

Pokrytí uvnitř budov podporuje GSM taky a dosahuje se ho děličem výkonu který přenáší radiový signál z vnějšku do odděleného systému antén uvnitř. To se používá hlavně když je potřeba velká kapacita hovorů, například v obchodních centrech nebo na letištích. I když to není nezbytně nutné, protože radiový signál z venku se šíří i skrz zdi.

Struktura sítě

Síť za systémem GSM je velká a složitá, aby mohla poskytovat veškeré vyžadované služby. Je rozdělena do několika sekcí.

- *Systém základnových stanic*
- *Síťový a přepínací podsystém (část sítě nejvíce podobná pevné síti). Většinou se jí říká prostě hlavní síť.*
- *Hlavní síť GPRS (volitelná část která umožňuje internetové spojení na bázi paketů).*
- *Všechny elementy se skládají dohromady, aby mohly poskytovat GSM služby jako hovory a SMS.*

3.3 Systém základnových stanic BTS

Zvýrazněný text byl převzat z encyklopedie [9] a upraven.

Systém základnových stanic (anglicky Base Station System, BSS) je část sítě GSM, která je zodpovědná za přenos a příjem radiových signálů z mobilního telefonu. Systém základnových stanic provádí překódování hovorových kanálů, přidělování radiových kanálů mobilním telefonům a mnoho dalších úkolů patřících k radiové síti.

Základnová převodní stanice

Základnová převodní stanice (Base Transceiver Station - BTS,) je vysílač a přijímač radiových signálů. Při použití směrových antén na základnových stanicích, kdy každá vysílá do jiného směru, je možné sektorizovat základnovou stanici, takže je několik buněk obsluhováno z jednoho místa. To zvyšuje provozní kapacitu základnové stanice (každá frekvence může obsahovat osm hlasových kanálů) zatímco moc nezvyší rušení způsobené na sousední buňky (v každém směru se šíří jen malý počet frekvencí.)

Buňky se rozdělují na několik poddruhů:

- **Makrocell** - tzv. makrobuňka je klasickým příkladem standardní základnové stanice. Má kapacitu 8 timeslotů a dosah až 35 km. Dříve se na stožár základnové stanice dával pouze jeden vysílač, nyní kvůli zvýšení kapacity obsahuje stožár vysílače makrocell až tři vysílače, které mají vysílací úhel 120°. Pokud je použito na stožáru tři směrových vysílačů, obsahuje jeden stožár tři makrocelly (a podobně u dvou či jedné)
- **Mikrocell** - tzv. mikrobuňka je vysílač mnohem menší jak rozměrově, tak svým dosahem. Obecně se používá jedné antény s vyzařovacím úhlem 360°, která má ovšem dosah pouze 27 km. Mikrobuňky se používají na vykrytí "problematických" částí. (Vnitřek budov, nebo například Metro)
- **Umbrella cell** - deštníková buňka je kombinací předchozích dvou typů. Jelikož mikrobuňky nemohou zajistit úplné pokrytí určitého prostoru, používá se na překrytí "hluchých míst" obvykle jeden vysílač typu makro cell (s dosahem max. 35 km) a vyzařovacím úhlem 360°

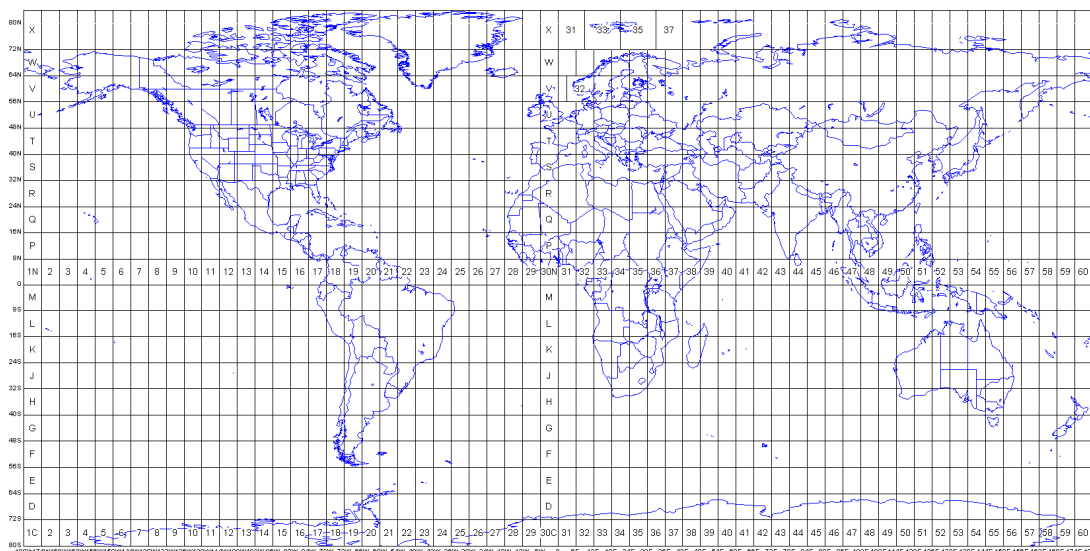
Programy Guesstor Server a Guesstor Client používají buněk typu makrocell, dokáží však pracovat i s ostatními druhy. Pozornému uživateli jistě neunikne skutečnost, že v některých případech je trojice vysílacích antén umístěna na stejné zeměpisné souřadnici a směry vysílání těchto antén jsou posunuty o 120°.

3.4 Souřadnicový systém UTM (*Universal Transverse Mercator*)

Zvýrazněný text byl převzat z webových stránek Geografického ústavu [10].

Zobrazení UTM (*Universal Transverse Mercator*) je konformní válcové zobrazení v příčné poloze. Je definováno na elipsoidu WGS84 a používá se pro území mezi rovnoběžkami 80° j.z.š. a 84° s.z.š. (na sever na jih od těchto rovnoběžek se používá zobrazení UPS - *Universal Polar Stereographics*).

Při zobrazení zemského povrchu do roviny pomocí UTM je podobně jako u Gaussova zobrazení zemský povrch rozdělen na poledníkové pásy. ČR leží v pásu 33 a 34. Poledníkový pás je definován osovým poledníkem se zeměpisnou délkou $\lambda \pm 3^\circ$. K zobrazení zemského povrchu je tedy použito 60 poledníkových pásů. Každý poledníkový pás je zobrazen na samostatnou válcovou plochu. Osa válcové plochy leží v rovině zemského rovníku a je kolmá na rovinu osového poledníku.



rozdělení povrchu Země na zóny UTM
<http://www.gpsinformation.org/utm-zones.gif>

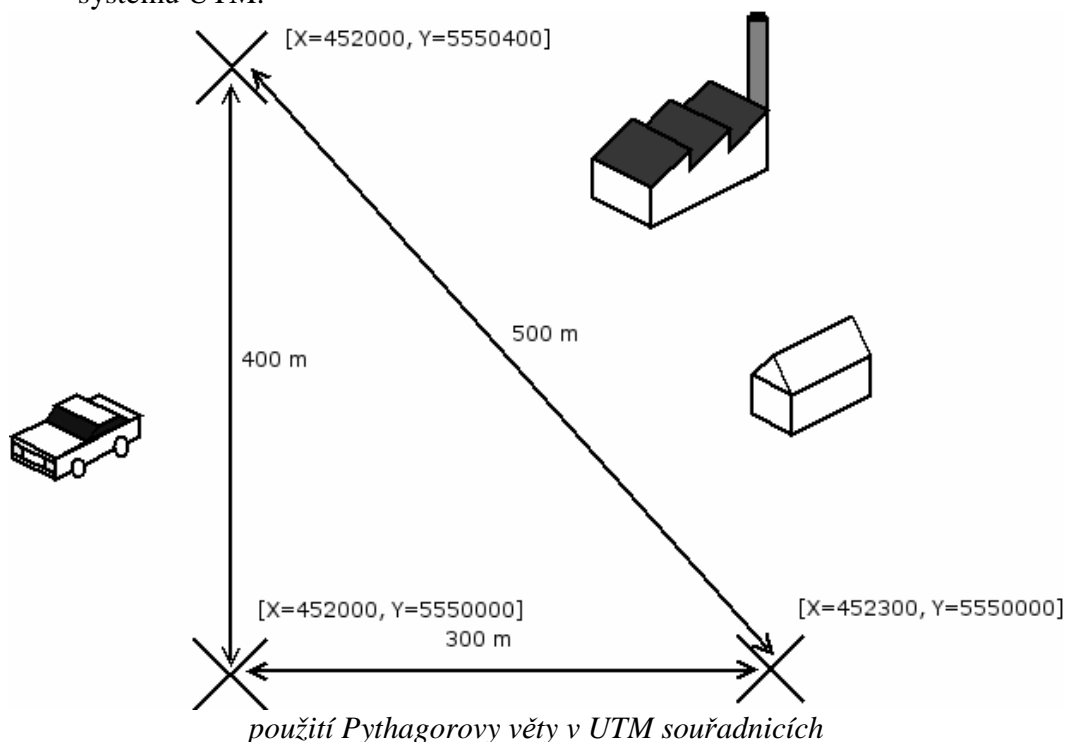
Každý poledníkový pás má svůj vlastní souřadnicový systém. Osa *N* je totožná s obrazem osového poledníku, osa *E* je rovnoběžná s obrazem rovníku. Počátek souřadnic leží v průsečíku obrazu rovníku a osového poledníku, ale je matematicky (stejně jako u Gaussova zobr.) posunut tak, aby všechny souřadnice v poledníkovém pásu měly kladné hodnoty.

3.5 Výhody souřadnicového systému UTM

Při návrhu Guesstor Clienta bylo potřeba rozhodnout, zda bude používat zeměpisné souřadnice ve formátu dvojice zeměpisná šířka a délka, nebo dá přednost jinému způsobu zápisu. Možných přístupů bylo několik, nakonec byla zvolena varianta s použitím souřadnic ve formátu UTM, protože formát UTM má několik významných výhod.

Výhody UTM:

- Systém souřadnic UTM rozděluje zemský povrch na velké množství malých zón, v rámci kterých je možné zakřivení zemského povrchu zanedbat. Práce s souřadnicemi na kouli se tím zjednodušuje na práci v rovině.
- Zeměpisná poloha nikdy nemá záporné souřadnice ve formátu UTM. Záporné souřadnice mohou uživatele zmást.
- Souřadnice UTM jsou uvedeny v metrech. Oproti běžnému zápisu pomocí zeměpisné šířky a délky má uživatel mnohem lepší představu o vzdálenostech mezi pozicemi.
- K výpočtu vzdálenosti mezi souřadnicemi lze použít Pythagorovy věty.
- Téměř celá Česká republika leží v UTM zóně U33.
- Podstatná část údajů v databázi Guesstor Serveru byla již od začátku ve formátu systému UTM.



3.6 Spring Framework

Zvýrazněný text byl převzat z [3]. Podrobnější informace o technologii Spring Framework lze nalézt v [11].

Spring je vícevrstvý aplikační framework napsaný v jazyce Java. Patří do třídy tzv. lightweight (odlehčených) J2EE kontejnerů. Jeho posláním je zjednodušit a zpřehlednit vývoj J2EE aplikací. Základní myšlenky frameworku se poprvé objevily v ukázkovém kódu ke knize Roda Johnsona [1].

Autoři Spring frameworku se snaží držet myšlenky ne vynalézat kolo a nesnažit se nahradit stávající technologie, ale spíše pomoci vývojářům v jejich využívání.

Inversion of Control (IoC)

Spring ctí myšlenku neinvazivnosti tj. nepromíchání vašeho kódu se Spring API. Aby k tomuto nedocházelo, je celý framework protkán Inversion of Control designem. IoC je návrhový vzor, který řeší těsné vazby mezi aplikačními komponentami provázanými v aplikačním kódu.

Díky knihovně Spring Framework je možné vytvářet program jako skupinu vzájemně nezávislých komponent, které se propojí až po spuštění programu. K propojení používám Spring Framework návrhový vzor Inversion of Control. Způsob propojení se určí pomocí XML souboru, který popisuje vazby mezi komponentami. Závislosti mezi komponentami tudíž není nutné psát do zdrojového kódu.

Program Guesstor Server využívá knihovnu Spring Framework. Jelikož v rámci bakalářské práce bylo potřeba Guesstor Server rozšířit, musely být naprogramovány nové komponenty Guesstor Serveru a následně bylo potřeba doplnit XML soubor o vazby na vytvořené komponenty. Použití knihovny Spring Framework ulehčilo také výměnu použité databáze z MySQL na HSQLDB. Více o výměně databáze se píše v kapitole věnované programátorské dokumentaci.

```
<beans>
  <bean id="antennaGen" class="cz.rdc.guess.AntennaGenerator">
    <constructor-arg type="int"><value>300</value></constructor-arg>
  </bean>

  <!--
  Guessing service.
  -->
  <bean id="guesstor" class="cz.rdc.guess.remote.GuesstorFacadeImpl">
    <property name="dataService"><ref local="dataService-nonSpatial"/></property>
    <property name="measureLoader"><ref local="measureLoader"/></property>
    <property name="algorithmBag"><ref local="algorithmBag"/></property>
    <property name="anntenaGenerator"><ref local="antennaGen"/></property>
  </bean>
```

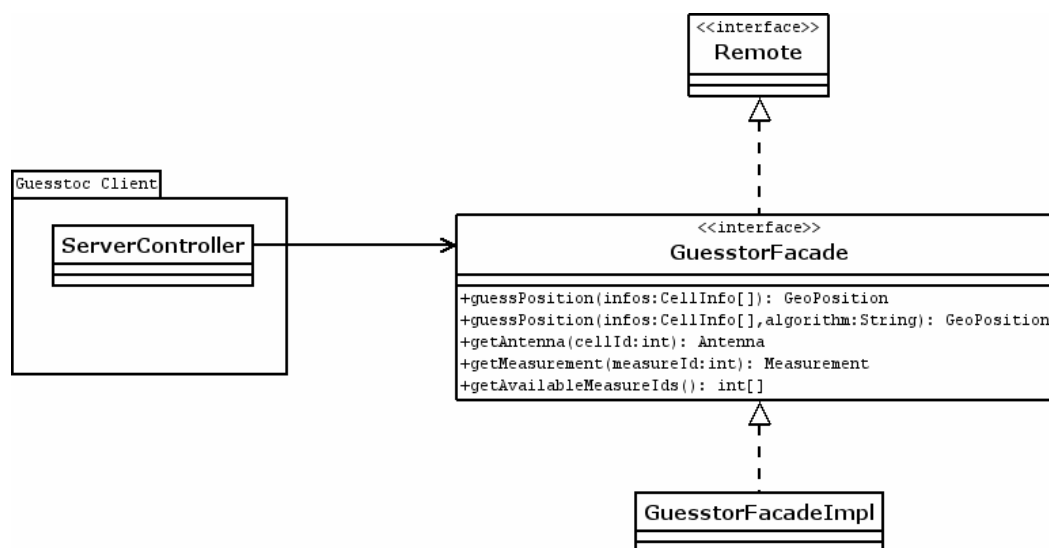
ukázka XML konfiguračního souboru

4 Programátorská dokumentace

4.1 Komunikace mezi serverem a klientem

Guesstor Client komunikuje s programem Guesstor Server pomocí rozhraní RMI (Java Remote Method Invocation). Díky tomu může Guesstor Client vzdáleně volat metody tříd Guesstor Serveru, který běží v jiné Java Virtual Machine třeba i na jiném počítači než Guesstor Client.

Komunikaci mezi Guesstor Serverem a Klientem osvětluje následující obrázek:



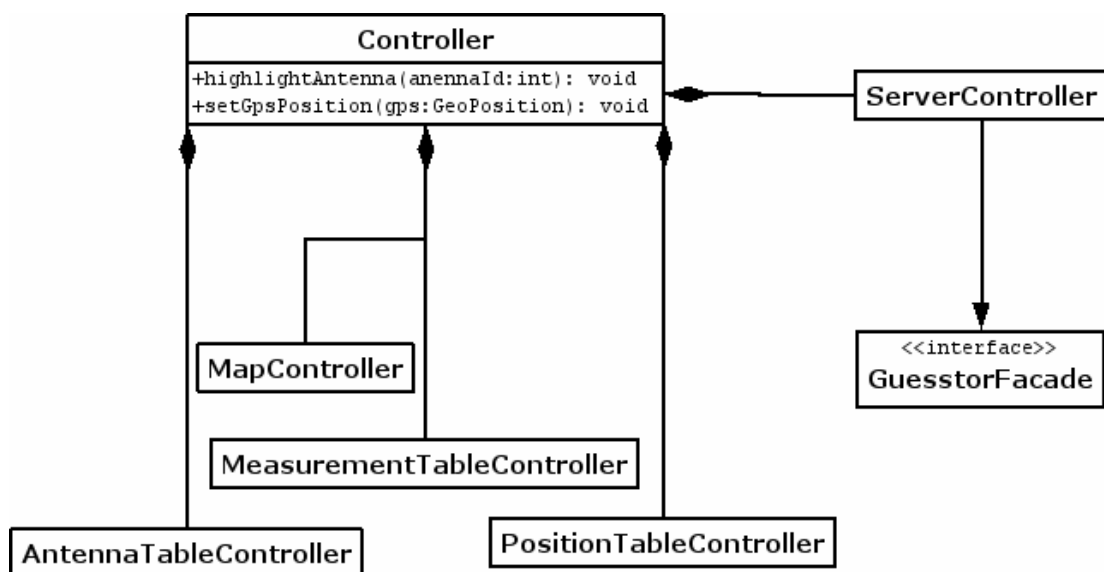
Aby se komunikace se serverem zjednodušila a zpřehlednila, bylo definováno rozhraní na vyšší úrovni GuesstorFacade, které umožňuje přístup k veškeré potřebné funkčnosti Guesstor Serveru a usnadňuje jeho použití. Při návrhu rozhraní bylo využito návrhového vzoru Fasáda (Facade), který má požadované vlastnosti.

Každé rozhraní, které je určeno pro vzdálené volání metod, musí být odvozeno od rozhraní Remote z balíčku `java.rmi`. Tudíž předkem rozhraní GuesstorFacade je rozhraní Remote. Guesstor Server obsahuje třídu GuesstorFacadeImpl, která rozhraní GuesstorFacade implementuje. Tato třída je grafickým klientem využívána pouze vzdáleně prostřednictvím rozhraní, nikdy ne přímo. Pro program Guesstor Client není dokonce ani viditelná.

Pro vzdálené volání metod pomocí rozhraní je zapotřebí, aby každá ze vzdáleně volaných metod měla možnost vyvolat výjimku RemoteException. Výjimka RemoteException není potomkem RuntimeException a tudíž je jí nutné vždy ošetřit nebo propagovat dále. Výjimka RemoteException vznikne v případě, když se nezdaří spojení se serverem, nebo nastane chyba v komunikačním protokolu.

Ze strany Guesstor Clienta je za komunikaci s Guesstor Serverem zodpovědná třída ServerController. Jelikož každá vzdáleně volaná metoda rozhraní GuesstorFacade může vyvolat výjimku RemoteException, je potřeba, aby došlo k odchytní a ošetření této výjimky v každé veřejné metodě třídy ServerController. Ostatní třídy programu Guesstor Client tak nejsou zatěžovány povinností ošetřovat výjimky vzniklé při komunikaci s Guesstor Serverem. Třída ServerController neví mnoho o kontextu volání svých metod. V důsledku toho výjimky v komunikaci se serverem ošetřuje velmi obecným způsobem, kdy informaci o vyjímce ohlásí prostřednictvím dialogového okna a do java konzole vypíše obsah zásobníku, ze kterého je možné poznat, co bylo příčinou výjimky. Dále je třída ServerController zodpovědná za uvedení vnitřní stavů všech objektů do konzistentní podoby v případě výjimky, aby mohl celý program pokračovat v činnosti.

Data, která třída ServerController získá vzdáleným voláním metod pomocí rozhraní GuesstorFacade, jsou dále předána třídě Controller, která rozhodne co se s daty bude dít dále. Třída Controller je zodpovědná na koordinaci ostatních tříd a přeposílá data ostatním obvykle vizuálním komponentám, které je zobrazují.



4.2 Datové struktury přenášené mezi serverem a klientem

Antenna

Tato třída se využívá při přenosu informací o vysílačích BTS (Base Transceiver Station).

Obsažené atributy:

Id jednoznačný identifikátor vysílače BTS
Position zeměpisné souřadnice polohy vysílače BTS
TransmitPower vysílací výkon vysílače BTS
Azimuth informace o směru natočení vysílací antény
BeamWidth velikost vysílacího kužele ve stupních.

Přístup k atributům je zapouzdřen pomocí přístupových metod set... a get... .

Pro snadnější výpis informací o vysílači za účelem ladění programu byla přetížena metoda toString() tak, aby vracela informace o vnitřním stavu třídy.

Antenna
-id: int -position: GeoPosition -transmitPower: double -azimuth: double -beamWidth: double
+setBeamWidth(beamWidth: double) +getBeamWidth(): double +setAzimuth(azimuth: double) +getAzimuth(): double +setTransmitPower(transmitPower: double) +getTransmitPower(): double +setPosition(position: GeoPosition) +getPosition(): GeoPosition +setId(id: int) +getId(): int +toString(): String

CellInfo

Jednoduchá třída CellInfo se využívá k předávání informací o intenzitě přijímaného signálu z vysílače BTS.

Obsažené atributy:

CellId jednoznačný identifikátor vysílače BTS

ReceivedPower..... intenzita naměřeného signálu

Hodnota atributů instance CellInfo je permanentně určena v konstruktoru třídy a následně se již nikdy nemění.

Přístup k atributům je zapouzdřen pomocí přístupových metod get... .

Tato třída má rovněž přetíženou metodu toString(), aby mohla pro účely ladění snadno vypsat do konzole obsah svých atributů.

<i>CellInfo</i>
-cellId: int -receivedPower: double
+CellInfo(cellId:int, double:receivedPower) +getCellId(): int +getReceivedPower(): double

GeoPosition

Třída GeoPosition se používá v případech, kdy je potřeba mezi Guesstor Serverem a Clientem přenášet informace o zeměpisných souřadnicích. Tato třída podporuje souřadnice ve formátu UTM, který je hlavním formátem zeměpisných souřadnic pro Guesstor Server.

Třída GeoPosition má všechny své atributy konstantní. Hodnota těchto atributů je stanovena během volání konstruktoru třídy a dále už není možné ji změnit. Tato skutečnost usnadňuje práci s objekty GeoPosition a umožňuje je sdílet bez obavy, že objekt bude nevhodně změněn.

Objekty GeoPosition navíc obsahují metody, které umožňují provádět aritmetiku se zeměpisnými souřadnicemi. Díky nim je možné zjistit rozdíl mezi zeměpisnými souřadnicemi, nebo k zeměpisné souřadnici přičíst či odečíst vektor.

Tato třída má rovněž přetíženou metodu toString(), aby mohla pro účely ladění snadno vypsat do konzole obsah svých atributů.

Obsažené atributy:

X..... souřadnice X ve formátu UTM

Y..... souřadnice Y ve formátu UTM

Zone..... zóna UTM

Northern..... příznak severní polokoule

Přístup k atributům je zapouzdřen pomocí přístupových metod get... .

GeoPosition
-X: double
-Y: double
-Zone: int
-Northern: boolean
+GeoPosition(x: double, y: double, zone: int, northern: boolean)
+difference(subtrahend: GeoPosition): Vector2D
+add(addend: Vector2D): GeoPosition
+subtract(subtrahend: Vector2D): GeoPosition
+getNorthern(): boolean
+getX(): double
+getY(): double
+toString(): String

Measurement

Třída Measurement v sobě zapouzdřuje informace z jednoho měření. Obsahuje seznam vysílačů, od kterých byl během měření přijat signál a intenzitu naměřených signálů. Na základě těchto údajů je pak možné odhadnout pozici, kde bylo měření provedeno. Třída Measurement navíc obsahuje souřadnice měření určené pomocí systému GPS, takže výsledek odhadu je možné porovnat se skutečností.

Tato třída má rovněž přetíženou metodu toString(), aby mohla pro účely ladění snadno vypsat do konzole obsah svých atributů.

Obsažené atributy:

Id..... jednoznačný identifikátor měření
Cell..... pole informací o úrovních signálů
Gps..... zeměpisná souřadnice měření zjištěná pomocí systému GPS
Timestamp..... čas, kdy bylo měření provedeno

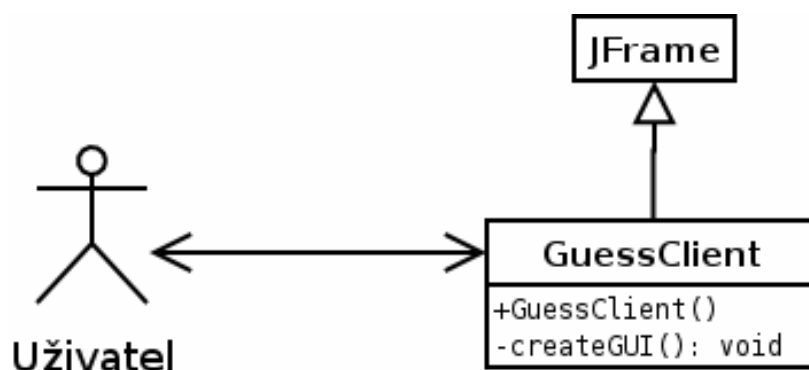
Přístup k atributům je zapouzdřen pomocí přístupových metod get... a set... .

Measurement
-Id: int
-Cell: CellInfo[]
-Gps: GeoPosition
-Timestamp: Date
+getCell(): CellInfo[]
+setCell(cell:CellInfo)
+getGps(): GeoPosition
+setGps(gps:GeoPosition)
+getId(): int
+setId(id:int)
+getTimestamp(): Date
+setTimestamp(timestamp:Date)
+toString(): String

4.3 Popis tříd programu Guesstor Client

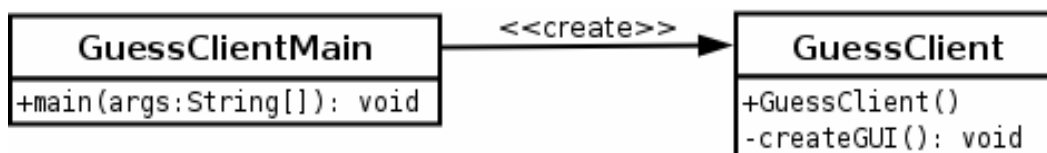
Třída **GuessClient**

Nejdůležitější třídou grafického klienta je třída **GuessClient**. Jedná se o třídu reprezentující hlavní okno programu. Prakticky veškerá interakce mezi uživatelem a aplikací probíhá v rámci formuláře vytvořeného z třídy **GuessClient** nebo některé komponenty vložené do tohoto formuláře. Během konstrukce objektu **GuessClient** se vytvoří veškeré komponenty a vloží se do formuláře. Na konci konstrukce je proveden pokus o spojení se serverem, v případě úspěchu se ze serveru získají a zobrazí informace o dostupných měřeních a vysílačích BTS. Za běhu aplikace se vytváří pouze jeden objekt (formulář) typu **GuessClient** na začátku aplikace. **JFrame** je třída obecných formulářů z knihovny Swing (`javax.swing.JFrame`).



Třída **GuessClientMain**

Třída **GuessClientMain** představuje pouze oddělený vstupní bod pro spuštění aplikace. Její jediná metoda **main** vytvoří instanci třídy **GuessClient** a vytvořený formulář zobrazí. Aplikace skončí po uzavření vytvořeného formuláře **GuessClient**.

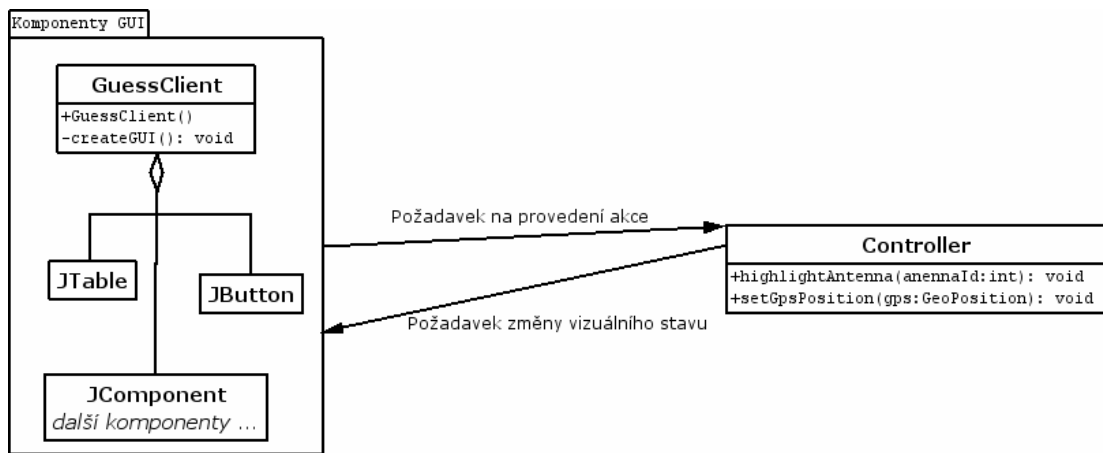


Poznámka: Původně byla zamýšlena i možnost spustit program jako applet v internetovém prohlížeči. Od této možnosti bylo upuštěno kvůli množství obtíží zejména s načítáním lokálních souborů, které je v appletu problematické.

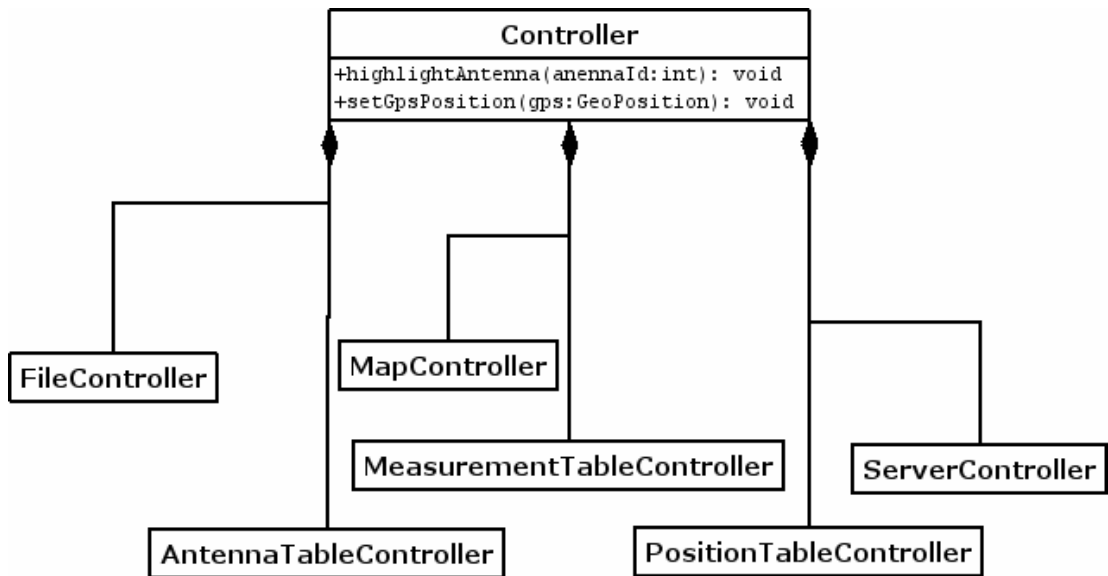
Třída Controller

Třída **Controller** představuje vnitřní logiku aplikace. Většina akcí uživatele vyvolá nějakou metodu objektu **Controller**. Tím se obvykle změní vnitřní stav aplikace. V důsledku změny vnitřního stavu **Controller** změní některé vlastnosti vizuálních komponent, aby dal uživateli změnu stavu najevo.

Například pokud uživatel klikne myší na mapu, o kliknutí je informován **Controller**. **Controller** zjistí zda byla vybrána kliknutím nějaká anténa. Pokud ano, upozorní tabulku Antennas a mapu s vysílači BTS, aby vybraný vysílač zvýraznil.



Při implementaci třídy Controller bylo třeba zabránit tomu, aby veškerá aplikační logika byla soustředěna v jediné třídě. Tato třída by pak obsahovala veliké množství metod, čímž by se stala špatně udržovatelnou. Specializované části kódu z třídy **Controller** proto byly rozděleny do více tříd. Na následujícím UML diagramu je zobrazeno šest tříd, do kterých je umístěna většina kódu aplikační logiky.

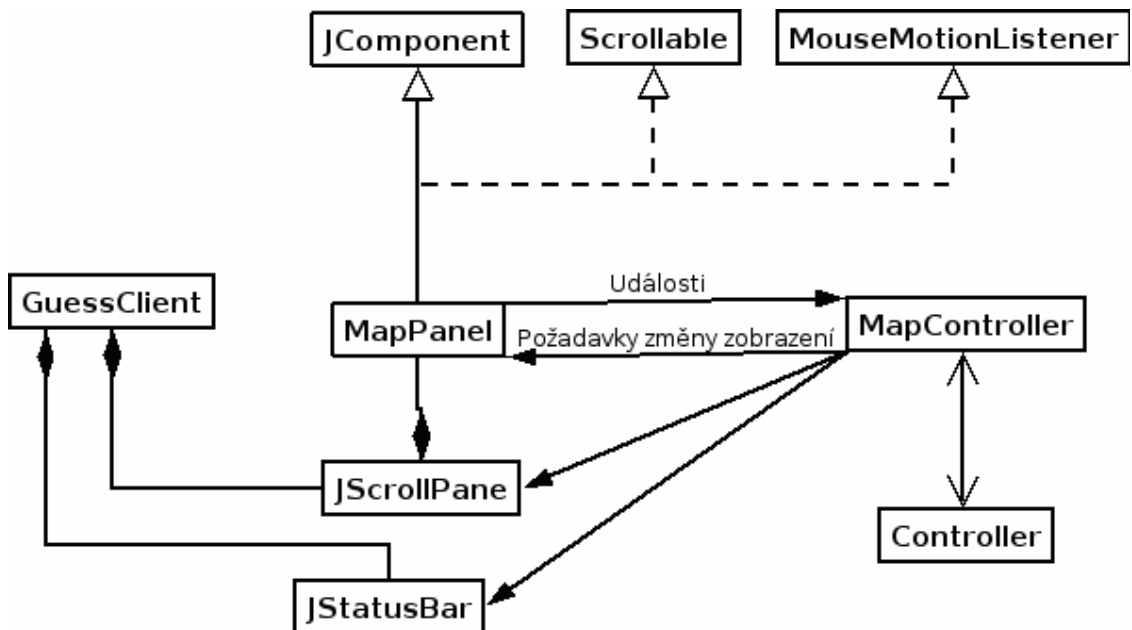


Význam třídy **Controller** spočívá hlavně zajištění komunikace mezi specializovanými controllery a vykonávání několika funkcí, které se nedaly zařadit do žádného specializovaného controlleru. Třída **Controller** se chová jako návrhový vzor Prostředník (Mediator).

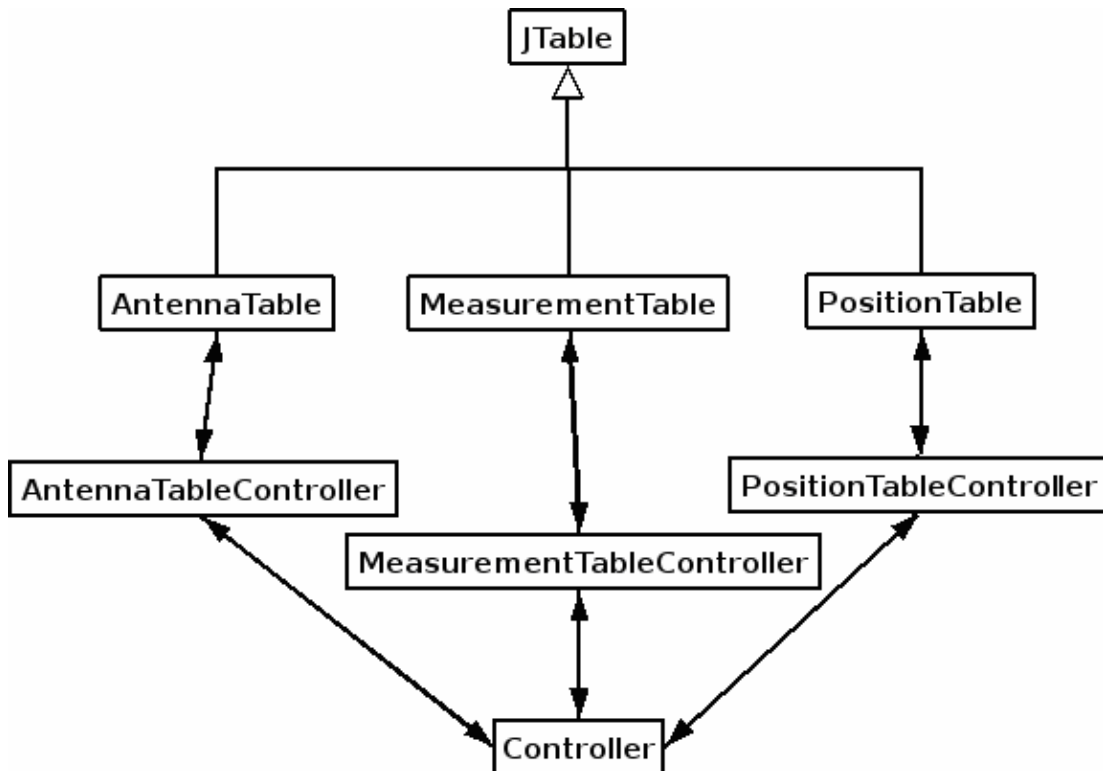
Například když uživatel stiskne levé tlačítko myši na mapě, událost zachytí specializovaný controller pro mapu **MapController**. Ten se postará o prvotní zpracování události. Projde seznam zobrazených vysílačů BTS a zjistí, zda stisknutím tlačítka byl některý vysílač vybrán. Pokud ano, zašle požadavek na zvýraznění vysílače hlavní řídicí třídě **Controller**. Na základě požadavku od třídy **MapController** rozešle **Controller** požadavek na změnu stavu všem specializovaným controllerům, které mají zájem o událost změny vybraného vysílače BTS. V našem případě jimi jsou **MapController** a **AntennaTableController**. **MapController** na požadavek zareaguje změnou barvy, kterou se vybraný vysílač bude vykreslovat. **AntennaTableController** na požadavek zareaguje zvýrazněním řádku s informacemi o vysílači BTS.

Třídy MapPanel a MapController

Třída **MapPanel** je vizuální komponenta, která zobrazuje mapu a vysílače BTS. **MapPanel** je navržen tak, aby mohl být bez problému použit i v jiné aplikaci, přestože se takové použití zatím nepředpokládá. Základní logika je implementována přímo v komponentě **MapPanel**. Komponenta implementuje rozhraní **Scrollable**, aby dokázala lépe spolupracovat s komponentou **JScrollPane**. Protože mapa může často být i větší než rozměry obrazovky, předpokládá se použití **JScrollPane** pro zobrazení pouze části mapy. Využití **JScrollPane** však není podmínkou. Komponenta **MapPanel** dokáže sama reagovat na některé události myši, proto implementuje rozhraní **MouseMotionListener**. Další události zpracovává **MapController**.



Třídy `AntennaTable`, `AntennaTableController`, `MeasurementTable`, `MeasurementTableController`, `PositionTable`, `PositionTableController`
 Některé třídy mají jméno končící **Table**. Jedná se o komponenty, které zobrazují tabulky s informacemi o vysílačích, naměřených hodnotách nebo odhadnutých pozicích. Třídy `xxxTableController` představují aplikační logiku příslušné tabulky.



Třída `GUIStore`

Třída `GUIStore` představuje fasádu k rozhraní pro vytváření komponent. GUI (Graphics User Interface) komponenty je často potřeba vytvořit zároveň s příslušným controllerem. Navíc je třeba zajistit, aby každý controller i komponenta byl vytvořen právě jednou. Někdy je dokonce důležité i pořadí konstrukce komponent. Takovéto komplikované požadavky mohou vést k chybám, proto byla konstrukce většiny komponent zapouzdřena v třídě `GUIStore`. Třída vytvoří ve správném pořadí všechny komponenty a jejich controllery už ve svém konstruktoru a následně vrací reference na vytvořené objekty. Tím je zaručeno, že každá komponenta je vytvořena právě jednou.

4.4 Generátor měření

V rámci bakalářské práce byla rozšířena funkčnost Guesstor Serveru o generátor měření. Generátor je tvořen třídami MeasurementsGenerator a AntennaGenerator.

Třída **AntennaGenerator** vygeneruje požadované množství vysílačů, které jsou náhodně rozmístěny po mapě a jejich parametry jsou rovněž náhodné ovšem v rámci obvyklých rozsahů. Tato třída je zapotřebí, protože mobilní operátor dodal informace o vysílačích BTS pouze v pražských Dejvicích. Aby bylo možno demonstrovat činnost programu i v jiných částech Prahy, program si pomůže uměle vygenerovanými vysílači.

Třída **MeasurementsGenerator** je zodpovědná za generování trajektorie pohybujících se objektů. Tato třída využívá sady klíčových pozic, mezi kterými pak interpoluje. Tím získá množství bodů, v kterých potom podle modelu šíření signálu vygeneruje hodnoty o intenzitě přijímaného signálu. Výhodou této třídy je to, že umožňuje simulovat pohyb automobilu a tudíž je možné demonstrovat činnost GuesstorClienta i v případě, kdy se žádné reálné auto s vybavením pro měření intenzity signálu nepohybuje. Do měření, které je výstupem této třídy je zanášena záměrná odchylka, aby se tak lépe simulovala reálná situace.

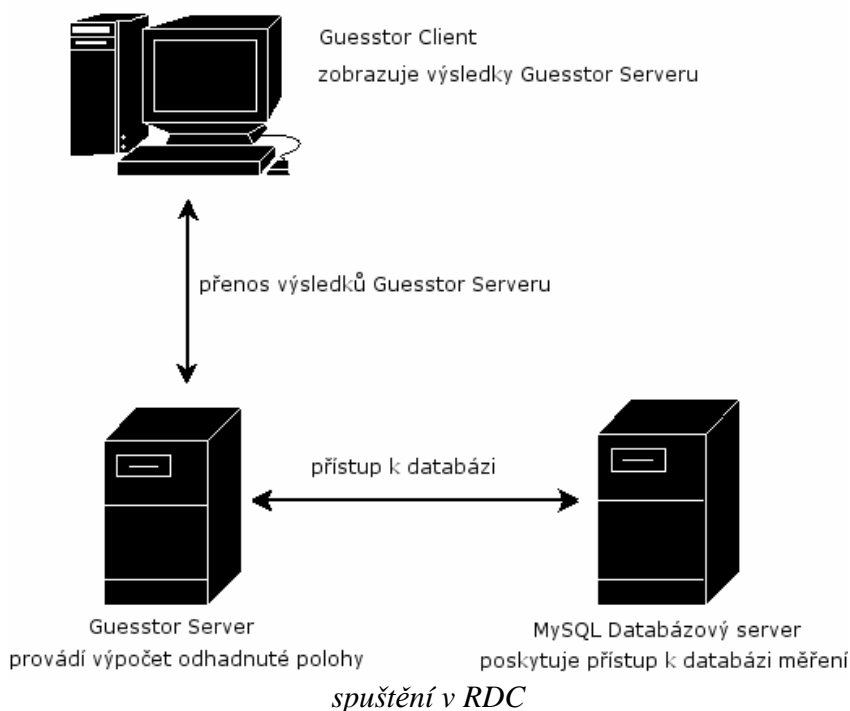
Následující obrázek zobrazuje trajektorii vytvořenou generátorem. Světlou barvou je zobrazena spojnice míst, pro které byly podle modelu generovány hodnoty měření. Tmavá trajektorie představuje spojnici míst, která byla odhadnuta pomocí lokalizačního algoritmu.



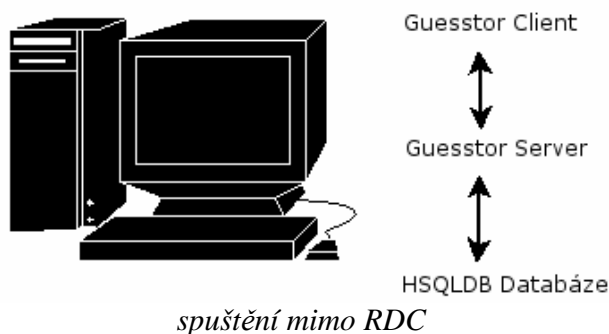
trajektorie vytvořená generátorem

4.5 Databázové technologie

Guesstor Server používá databázi pro načítání údajů o parametrech vysílačů a přístup k tabulkám měření. Ve Výzkumném a vývojovém centru RDC se Guesstor Client připojuje vzdáleně k počítači, na kterém běží Guesstor Server. Dále je v RDC umístěn databázový MySQL server, který zajišťuje přístup k tabulkám měření a vysílačů.



Pro demonstraci práce mimo RDC však takový přístup není žádoucí. MySQL server bývá někdy nesnadné nainstalovat a to by mohlo způsobit problémy při předvádění práce. Proto byl Guesstor Server upraven tak, aby umožňoval spojení i s databází HSQLDB, kterou není potřeba instalovat. Databáze HSQLDB však umí zpracovávat jen podmnožinu jazyka SQL, tudíž bylo potřeba upravit obsah databáze a dotazy do ní. Díky této změně je možné spouštět Guesstor Clienta i Guesstor Server na jediném počítači a není potřeba předem instalovat žádnou databázi.



5 Závěr

5.1 Zhodnocení přesnosti jednotlivých metod

Jednou z funkcí programu Guesstor Client je srovnávání přesností lokalizačních algoritmů v tabulce Positions. Díky této vlastnosti bylo možné statisticky zpracovat přesnost jednotlivých lokalizačních algoritmů a vzájemně je porovnat.

Pro srovnání přesnosti a kvality jednotlivých lokalizačních metod bylo provedeno statistické šetření na větším množství naměřených hodnot. Bohužel všechna měření pochází z okolí Dejvic v Praze, protože mobilní operátor neumožnil přístup k informacím o vysílačích pro jiné oblasti. Bylo by velice zajímavé zjistit jak se lokalizační algoritmy chovají v jiných podmínkách třeba na venkově, kde signál netlumí zástavba velkoměsta, ale síť vysílačů tam není zdaleka tak hustá jako v Praze.

Z množství měření, na kterých bylo prováděno statistické šetření, je možné si v programu Guesstor Client prohlédnout jen zlomek. Tyto údaje jsou pro mobilního operátora citlivé a jejich zveřejnění z tohoto důvodu není možné.

Pro srovnání jsou v tabulce uvedeny i výsledky algoritmu Spatial, který vznikl v rámci jiného projektu v RDC. Tento algoritmus také odhaduje polohu na základě informací o síle signálu z jednotlivých vysílačů, ale jeho model šíření signálu je komplexnější a výpočet je časově náročnější.

Název algoritmu	Směrodatná odchylka	Průměrná doba výpočtu
Serving cell	1534 m	132 ms
Least square	2313 m	216 ms
Weighted	695 m	218 ms
Iterative	508 m	216 ms
Spatial	261 m	20,4 s

Tabulka výsledků lokalizačních metod

5.2 Závěrečné ohodnocení jednotlivých metod

Algoritmus **Serving cell** byl vytvořen pouze pro otestování Guesstor Serveru v době, kdy ostatní algoritmy ještě nebyly implementovány. Tento algoritmus je zatížen příliš velkou průměrnou odchylkou při měření na reálných datech a pro praktické využití se nehodí. V případě, že jsou měření generována uměle podle zvoleného modelu šíření signálu k přílišnému zlepšení nedochází a odchylka je rovněž velmi velká.

Algoritmus **Least square** až příliš spoléhá na přesnost zvoleného modelu šíření signálu. V případě uměle generovaných měření přesně podle modelu má tento algoritmus nulovou odchylku, ale na reálných měřeních selhává. V průměrném případě je dokonce horší než triviální algoritmus Serving cell. Příčinou takto vysoké odchylky je značně zjednodušený model šíření signálu, který předpokládá homogenní prostředí v okolí vysílače. V praxi však něco takového nikdy nenastává a volným prostorem se signál šíří snáze než městskou zástavbou. Model dále nezohledňuje výškový rozdíl mezi vysílačem a bodem měření. Zohledňovat výškový rozdíl nebylo možné, protože v databázi nejsou informace o nadmořské výšce vysílačů. V případě, že je měření prováděno v těsné blízkosti vysílače (pod vysílačem), úroveň přijímaného signálu je zde nízká, protože vysílač šíří signál primárně do stran a nikoliv směrem dolů. Algoritmus Least square tento fakt ale mylně interpretuje jako měření ve velké vzdálenosti od daného vysílače. Ostatní algoritmy jsou vůči takovýmto problémům více odolné.

Algoritmus **Weighted** dosahuje na reálných měřeních poměrně dobrých výsledků. Pokud však naměřená data generujeme uměle podle modelu šíření signálu, přesnost se bohužel nijak zvlášť nezlepší. Další předností tohoto algoritmu je skutečnost, že odhadnutá pozice nikdy není zcela mimo očekávanou oblast tak, jak tomu občas bývá u algoritmu Least square.

Algoritmus **Iterative** dosahuje na reálných datech velmi dobrých výsledků v porovnání s předcházejícími algoritmy. V případě uměle generovaných dat podle modelu šíření se jeho přesnost výrazně zlepšuje a odchylka bude téměř nulová. Odhadnutá pozice také nikdy neleží zcela mimo očekávanou oblast. Tento algoritmus je patrně nejlepší, který Guesstor Server implementuje.

Algoritmus **Spatial** patří do jiné skupiny lokalizačních algoritmů než předcházející čtyři. Jeho vývojem se zabýval jiný projekt. Je zde uveden pouze pro srovnání odlišných způsobů lokalizace. Tento algoritmus využívá predikčních map, které určují intenzitu signálu v každém bodě mapy a následně hledá na mapě bod, který nejvíce vyhovuje naměřeným hodnotám. Guesstor Client, který vznikl jako součást této práce, výsledky Spatial algoritmu nezobrazuje. Pro zobrazování výsledků Spatial algoritmu vznikl v rámci RDC jiný program, který se specializuje na práci s predikčními mapami. Spatial algoritmus dosahuje poměrně přesného určení polohy, ale platí za to dlouhým výpočetním časem (v průměru přes 20 sekund). Zvýšení přesnosti je dosaženo pomocí realističtějšího modelu šíření signálu, který zohledňuje zástavbu. Na vstupu algoritmu jsou ale potřeba podrobné mapy architektury v okolí vysílačů, které se obtížně shání.

5.3 Zhodnocení dosažených výsledků

Projekt Guesstor přinesl množství zajímavých výsledků. Byla otestována metoda lokalizace mobilního zařízení na základě intenzity signálů z vysílačů BTS a to umožnilo ji porovnat s ostatními odlišnými možnostmi lokalizace. Výčet dosažených výsledků, úspěchů a neúspěchů následuje.

Vývoj lokalizačních algoritmů

Pomocí programu Guesstor Client byla usnadněna práce vývojářů Guesstor Serveru na lokalizačních algoritmech. Díky přehlednému grafickému rozhraní mají jistotu, že jejich algoritmy pracují dle očekávání a nemusí se dívat na množství těžce srozumitelných čísel. Díky programu Guesstor Client bylo opraveno několik chyb v lokalizačních algoritmech.

Porovnání přesnosti s jinými metodami lokalizace

Provedené statistické zhodnocení přesnosti metody lokalizace pomocí úrovně signálů bylo porovnáno s přesností metod založených na jiném principu. Bohužel se tato metoda ukázala v porovnání s ostatními jako značně nepřesná. Jedinou výhodou oproti ostatním metodám jsou její nízké náklady na zavedení.

Poskytování služeb zákazníkům

Metoda lokalizace pomocí úrovně signálu se ukázala jako málo přesná. Z tohoto důvodu se neočekává její nasazení pro poskytování služeb zákazníkům mobilní sítě.

Lokalizace odcizeného vozidla

Lokalizační programy Guesstor byly použity k odhalení krádeže vozidla mobilního operátora zatím dvakrát. V prvním případě bylo na základě lokalizace určeno místo výskytu vozidla. Při prozkoumání daného místa policií zde našli pouze z auta vymontované vysílající mobilní zařízení. Automobil tak nebyl vypátrán. Jednou z příčin tohoto neúspěchu bylo opožděné provedení lokalizace. Druhý případ použití byl o mnoho úspěšnější. Poloha automobilu byla určena včas a policii se na základě stanovených souřadnic podařilo ukradený automobil zajistit a pachatele dopadnout.

Sledování trajektorií vozidel

V rámci projektu Guesstor vznikl software, který umožňuje sledovat trajektorii služebních vozidel mobilního operátora. Mobilní operátor však pro nasazení technologie neprojevil zájem a dává přednost důvěře ve své zaměstnance před jejich monitorováním.

Přínos nových zkušeností pro programátory RDC

Díky projektu Guesstor všichni programátoři, kteří se na projektu podíleli, získali mnoho nových zkušeností a dovedností. Osvojili si práci s databázemi, vzdáleným voláním funkcí, programováním v týmu a v mnoha jiných oblastech. I toto je jedním z hlavních cílů Výzkumného a vývojového centra RDC.

Literatura

- [1] Johnson Rod: Expert One-on-One J2EE Design and Development
Wrox; New Ed edition (2002), ISBN: 0764543857
- [2] Körpeoğlu İbrahim: Radio Propagation Models
Bilkent University, Department of Computer Engineering
<http://www.cs.bilkent.edu.tr/~korpe/courses/cs515/lecture-slides/wireless-transmission-2s.pdf>
- [3] Malěj Jakub: Spring framework (2005)
<http://nb.vse.cz/~zelenyj/it442/eseje/xmalj35/spring.htm>
- [4] Olrich Martin: Služby lokalizace polohy
diplomová práce, České vysoké učení technické v Praze, Fakulta elektrotechnická,
Katedra telekomunikační techniky, 2005
- [5] Pravda Michal: Dokumentace metody nejmenších čtverců
Výzkumné a vývojové centrum RDC, 2005
- [6] Výzkumné a vývojové centrum RDC
<http://www.rdc.cz/>
- [7] Global Positioning System. (19. 05. 2006). Wikipedie: Otevřená encyklopedie.
<http://cs.wikipedia.org/wiki/GPS>
- [8] Global System for Mobile Communications. (18. 05. 2006). Wikipedie: Otevřená encyklopedie. <http://cs.wikipedia.org/wiki/GSM>
- [9] Systém základnových stanic. (13. 05. 2006). Wikipedie: Otevřená encyklopedie.
http://cs.wikipedia.org/wiki/Syst%C3%A9m_z%C3%A1kladnov%C3%BDch_stanic
- [10] Geografický ústav - Přírodovědecká fakulta Masarykovy univerzity
<http://www.geogr.muni.cz/>
- [11] Spring Application Framework
<http://www.springframework.org/>