

# **Bakalářská práce**

**Petr Lukeš**

**2007**

# Obsah

|                                |   |
|--------------------------------|---|
| <b>1. Rozbor problému</b>      |   |
| 1.1                            | Obecná struktura družicových polohových systémů.....6                                 |
| 1.1.1                          | Základní informace o satelitním systému GPS.....7                                     |
| 1.2                            | Metody určování polohy.....8  |
| 1.2.1                          | Kódová měření.....8   |
| 1.2.2                          | Fázová měření.....8   |
| 1.2.3                          | Dopplerovská měření.....9   |
| 1.3                            | Studium technik řešení nejednoznačnosti.....10  |
| 1.3.1                          | Úvod.....10   |
| 1.3.1.1                        | Fázová měření s dvojitými diferencemi.....10  |
| 1.3.1.2                        | Geometrická omezení.....10  |
| 1.3.2                          | Techniky řešení nejednoznačnosti.....12   |
| 1.3.2.1                        | Operační módy.....12  |
| 1.3.2.2                        | Hledání v doméně pozorování “Observation Domain“.....13                               |
| 1.3.2.3                        | Hledání v doméně souřadnic “Coordinate Domain“.....13                                 |
| 1.3.2.4                        | Hledání v doméně odhadovaných nejednoznačností<br>“Estimated Ambiguity Domain“.....14 |
| 1.4                            | Komunikace se senzorem AC12 – Datové zprávy protokolu NMEA.....15                     |
| 1.4.1                          | Formát Zpráv.....15   |
| 1.4.2                          | Zprávy NMEA.....17  |
| 1.5                            | Metoda nejmenších čtverců.....20  |
| <b>2. Řešení problému</b>      |   |
| 2.1                            | Zpracování navigačních zpráv.....23   |
| 2.2                            | Aplikace metody nejmenších čtverců.....25   |
| <b>3. Zhodnocení.....</b>      | <b>26</b>   |
| <b>Použitá literatura.....</b> | <b>27</b>   |
| <b>Přílohy.....</b>            | <b>28</b>   |

Katedra řídicí techniky

Školní rok: 2005/2006

## Zadání bakalářské práce

Student: Petr Lukeš  
Obor: Kybernetika a měření  
Název tématu: GNSS senzor orientace

### Zásady pro vypracování:

Vytvořte program pro PC pro senzor orientace v prostředí Builder C++.

1. Naprogramujte komunikaci po sériových linkách.
2. Implementujte protokol UBX.
3. Naprogramujte řešení přečtené soustavy lineárních rovnic metodou nejmenších čtverců.

*Seznam odborné literatury:* Dodá vedoucí práce

Vedoucí bakalářské práce: Dr. Ing. Pavel Kovář

Datum zadání bakalářské práce: zimní semestr 2005/06

Termín odevzdání bakalářské práce: 26. 5. 2006

Prof. Ing. Michael Šebek, DrSc.  
vedoucí katedry



V Praze, dne 6. 4. 2006

Prof. Ing. Zbyněk Škvor, CSc.  
děkan

## Prohlášení

Prohlašuji, že jsem svou bakalářskou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v příloženém seznamu.

V Praze, dne 17. 1. 2007 .....

*Michal Lukš*  
.....  
podpis

# Anotace

Globální družicový navigační systém – GNSS, má v dnešní době mnoho oborů využití. Jedním z těchto oborů je i možné použití do senzorů orientace. Tyto senzory orientace pak mají další využití například v letecké dopravě. Pro přesný senzor je však třeba využít maximálně možností satelitní navigace. Tyto možnosti jsou použití dálkoměrných metod určení polohy pro přesné kódové a fázové měření. Při užití v letecké dopravě je totiž potřebná co největší přesnost. Proto cílem této práce je řešení problému senzoru orientace s co možná nepřesnějším určení polohy.

Global Navigation Satellite System – GNSS, have in our period many domain of use. One of this domain, is using for the orientations sensors. This sensors have other improvement in air transporting. For precision sensor is necessary to use satellite navigations at maximum. This is using of telemetric method for precision code and carrier phase measurement because of in air transport is most important very high precision navigation. So the objective of this work is programming the precision orientation sensor.

# 1. Rozbor problému

## 1.1 Obecná struktura družicových polohových systémů

Družicové polohové systémy jsou tvořeny třemi základními segmenty:

- kosmickým segmentem
- řídicím segmentem
- uživatelským segmentem

**Kosmický segment** – jeho částí je soustava umělých družic Země, které obíhají po přesně definovaných a určených oběžných drahách. Tento segment je určen především :

- typem oběžných drah, zda se jedná o kruhové nebo eliptické dráhy a zda jsou tyto dráhy nízké, střední, vysoké nebo geostacionární
- sklonem, počtem a výškou oběžných drah
- rozmístěním a počtem družic na oběžných drahách

Konfigurace segmentu je určena požadavky uživatelského a také možnostmi řídicího segmentu.

**Řídicí segment** – sada pozemních stanic, plnící následující úlohy :

- řízení kompletního systému
- sledování a vyhodnocování stavu družic
- manévry družic
- monitorování družicových signálů
- údržba družic
- Určování parametrů oběžných drah jednotlivých družic a vyhodnocování jejich chování
- kontrola stavu hodin na družicích a určování korekčních parametrům
- vysílání aktualizovaných parametrů

Pro tyto účely se stanice rozlišují na 3 typy

- **Stanice pro komunikaci s družicemi** – většinou shodné s monitorovacími stanicemi. Slouží k přenášení nově zvolených parametrů oběžných drah a korekčních parametrů atomových hodin na družice a také se používají k ovládání.
- **Monitorovací stanice** – jejich rozmístění je takové aby bylo možno stále a po co nejdelší dobu sledovat největší počet družic. Monitorují družicové signály a přenášejí je do centra.
- **Hlavní řídicí stanice** – Většinou jedna (a jedna záložní), slouží ke zpracování signálů z monitorovacích stanic, provádí modelování chování kosm. segmentu, určuje parametry oběžných drah a korekční parametry družicových hodin a tyto výsledky posílá předává stanicím určeným pro komunikaci s družicemi

Konfigurace tohoto segmentu je dána možnostmi rozmístění a provozování stanic a to buď pouze na svém území (pokud se jedná o vojenský systém) nebo v případě civilního systému na území států které se na tomto systému podílejí.

**Uživatelský segment** – Jeho součástí jsou přijímače, veškeré technické zařízení které umožňuje využití družicového polohového systému, postupy měření a vyhodnocování. S řídicím segmentem uživatelé do přímého styku nepřicházejí.

### 1.1.1 Základní informace o satelitním systému GPS

- Konstelace družic: 6 drah, každá po 4 družicích
- Výška: 20 200 km
- Poloměr dráhy: 26 570 km
- Inklinace: 51°
- Doba oběhu 11h 58m

**Tabulka 1 – frekvence, modulace a vlnová délka signálu GPS**

| Nosná | Frekvence   | Vlnová Délka | Modulace | Frekvence |
|-------|-------------|--------------|----------|-----------|
| L1    | 1575.42 MHz | 19 cm        | C/A kód  | 1.023 MHz |
|       |             |              | P kód    | 10.23 MHz |
|       |             |              | Zpráva   | 50 Hz     |
| L2    | 1227.60 MHz | 24 cm        | P kód    | 10.23 MHz |
|       |             |              | Zpráva   | 50 Hz     |

Co se týká rozdělení signálů GPS podle předchozí tabulky, je nutno říci, že v současné době probíhá modernizace. Jedním z kroků je i nový způsob modulace nosné vlny L2 pomocí civilního kódu L2C vyvinutím technologie R-Track.

## 1.2 Metody určování polohy

Družicové polohové systémy jsou vybudované jako pasivní dálkoměrné systémy. Výjimkou však jsou systémy využívající dopplerovské měření. To znamená, že přijímač určuje svojí vzdálenost k několika družicím a polohu určuje protínáním. Určení vzdálenosti každého přijímače od družic lze provádět na základě následujících měření:

- kódové měření
- fázové měření
- dopplerovské měření

### 1.2.1 Kódová měření

Základním principem kódových měření je určení vzdálenosti mezi přijímačem signálu a různými družicemi. Pro tento účel se využívá dálkoměrného kódu který vysílají jednotlivé družice. Tyto kódy jsou v podstatě jakési přesné časové značky, které umožňují přijímači určit čas, kdy byla odvysílána kterákoliv část signálu vysílaného družicí. Přijímač pracuje tak, že ve vstupním signálu přicházejícím z antény, identifikuje dálkoměrný kód příslušné družice, zjistí čas odeslání a přijetí jedné sekvence kódu a ze zjištěného časového rozdílu  $\Delta t_i$  určí vzdálenost mezi přijímačem a družicí  $d_i$  dle vztahu:

$$d_i = \Delta t_i \cdot c ,$$

Kde  $c$  je rychlost šíření rádiových vln.

Důležité však je, že jelikož hodiny přijímače nejsou úplně synchronní se systémovým časem družic, je časový rozdíl  $\Delta t_i$  zatížen jistou chybou hodin přijímače. Proto při výpočtu vzdálenosti  $d_i$  nevypočítáme skutečnou vzdálenost přijímače od družice ale pouze **pseudorange – zdánlivou (pseudo) vzdálenost**.

Pro kódová měření můžeme odvodit reálně dosažitelnou přesnost. Čipová frekvence dálkoměrných kódů se pohybuje na úrovni jednotek MHz pro standardní přesnost a desítek MHz pro vysokou přesnost. Pro tyto frekvence platí vlnové délky  $\lambda_1 = 300$  resp.  $\lambda_2 = 30$ m. Při reálně dosažitelné přesnosti měření 1 – 2 % vlnové délky, pak vychází přesnost pro  $\lambda_1 = 3-6$  m  $\lambda_2 = 0.3 - 0.6$  m . Je však nutné říci, že tato přesnost nezahrnuje ostatní vlivy nepřesností jako například vlivy prostředí, nepřesnosti hodin a podobně.

### 1.2.2 Fázová měření

Fázová měření jsou založena na poněkud jiném principu. Nepracují s dálkoměrnými kódy, ale zpracovávají vlastní nosné vlny. Při fázových měřeních přijímač v podstatě spočítá počet vlnových délek nosné vlny, nacházejících se mezi přijímačem a družicí. Tento počet se pak skládá z celočíselného násobku nosných vln a z desetinné části. Protože však přijímač obtížně určuje celočíselný násobek, fázové měření vykazují určitou nejednoznačnost (ambiguity) rovnající se počtu celých vlnových délek nosné vlny signálu (**celočíslná**



**nejednoznačnost – integer ambiguity**). Pro určování celočíselné nejednoznačnosti byla vypracována celá řada postupů, umožňujících její stanovení buďto při následném zpracování, nebo i přímo v reálném čase.

Jakmile se jednou počáteční hodnota celočíselné nejednoznačnosti určí, není už problém průběžně sledovat změny fázového posunu a počet celých vln a tím i polohu nebo při pohybující se stanici změnu polohy. Přijímač prostě zachovává počáteční hodnotu celočíselné nejednoznačnosti a k ní připočítává celý počet vlnových délek, o které se změnila vzdálenost mezi přijímačem a družicí od počátku měření a dále desetinou část vlnové délky.

Pokud je signál náhle zeslaben nebo ztracen, například při zastínění antény, nebo pokud je družice příliš nízko, a dojde k přerušení sledování nosné vlny nastane tzv. **fázový skok** (cycle slip), což způsobí že přijímač není schopen počítat vlnové délky, o které se změnila vzdálenost mezi přijímačem a družicí. Musí se proto začít nový cyklus měření, od určení aktuálního fázového posunu až po nové určení počáteční hodnoty celočíselné nejednoznačnosti na počátku nového měření.

Vlnové délky nosných vln jsou v tomto případě velice krátké, řádově desítky centimetrů. Pokud budeme uvažovat stejnou přesnost zpracování signálů jako v případě kódových měření (1 - 2%) pak nám vyjde přesnost určení vzdálenosti řádově v milimetrech.

### 1.2.3 Dopplerovská měření

Pro měření polohy dále můžeme také využít metody počítající Dopplerův posun. Tato metoda, dnes už však téměř nevyužívaná, využívá skutečnosti, že v důsledku relativního pohybu družice vůči přijímači se průběžně mění frekvence přijímaného signálu – tento Dopplerovský posun se měří na nosné vlně signálu. Takovýto posun se měří po určitou dobu a pak se na základě těchto měření vypočte změna radiální vzdálenosti mezi oběma zařízeními. Typicky se toto měření provádělo u statického přijímače, který měřil frekvenci satelitu v několika časových obdobích a následně vyhodnotil polohu ze změn v této přijímané frekvenci. Dopplerovská měření lze použít k určení polohy, ale většinou se používá k určování rychlosti přijímače.

## 1.3 Studium technik řešení nejednoznačnosti

### 1.3.1 Úvod

Řešení nejednoznačnosti je matematický proces, který převádí nejednoznačné vzdálenosti (integrované nosné vlny) na vzdálenosti bez nejednoznačnosti a s milimetrovou přesností měření.

Je velmi obtížné, ne-li přímo nemožné, určit celočíselnou nejednoznačnost pro data proudící pouze jedním směrem, protože jsou nerozeznatelné od ostatních, neceločíselných rušivých vlivů, jako je například nepřesnost orbitální dráhy satelitu, vícečetná dráha signálu v atmosféře, lom a odraz v ionosféře a troposféře. Díky těmto nežádoucím vlivům je vlastně signál „kontaminovaný“ a tím nepřesný. Tudíž řešení nejednoznačnosti, tak jak je všeobecně známo, je možné zjistit až po eliminaci nebo jiného započítání rušivých vlivů.

#### 1.3.1.1 Fázové měření s dvojitými diferencemi

Obvykle využívanou metodou pro eliminaci uvedených rušivých vlivů je dvojitě diferencovaná data mezi satelity a mezi přijímači. Rovnice měření dvojitých diferencí, je odvozena z klasické rovnice měření, a má tvar:

$$\Delta \nabla \Phi = \Delta \nabla r + \Delta \nabla d r + l \bullet \Delta \nabla N - \Delta \nabla d_{ion} + \Delta \nabla d_{trop} + \Delta \nabla d_{mp}^{\Phi} + e_{\Delta \nabla \Phi} \quad r.1$$

Kde  $\Delta \nabla$  je operátor dvojitě diference,  $\Phi$  je měřená nosná vlna v jednotkách metrech,  $N$  je celočíselná nejednoznačnost,  $\lambda$  je vlnová délka nosné,  $e_{\Phi}$  je šum na nosné vlně, a ostatní veličiny jsou chyby dané zpožděním signálu v atmosféře. Z této rovnice je zřejmé, že vlivy hodin přijímače a satelitu byly eliminovány. Kromě toho ostatní rušivé vlivy ( dráha satelitu, ionosférické a troposférické zpoždění  $\Delta \nabla d r - \Delta \nabla d_{ion} + \Delta \nabla d_{trop}$  ) budou sníženy, ale jak moc záleží na vzdálenosti mezi přijímači. Ve většině případů je celková velikost těchto vlivů menší než polovina vlnové délky, pokud jsou přijímače od sebe vzdáleny méně než 15 km. Operátor dvojitých diferencí nesnižuje vliv vícecestného šíření a velmi pravděpodobně se zvyšuje ve srovnání s jednocestným měřením. Standardní odchylka dvojitých diferencí, je dvakrát větší než u jednocestného měření. Dvojitě diferencovaná nejednoznačnost zůstane celočíselná a může být dále určena pomocí speciální metody.

Ačkoliv počáteční hodnota nejednoznačnosti je neznámá, GPS přijímač dokáže kontinuálně sledovat změny nosné fáze od jedné epochy ke druhé. Trojitě diferencované měření realizuje diferencí mezi epochami, na dvojitě diferencované nosné, a chybné výsledky v termínu celočíselné nejednoznačnosti jsou eliminovány.

#### 1.3.1.2 Geometrické omezení

Procedura řešení nejednoznačností může být užita za předpokladu jistých omezení vyplývajících z měření. Hlavní geometrické omezení jsou :

*Celočíselné omezení* – Teoreticky počítáme s tím, že počáteční nejednoznačnost fázového měření na frekvencích L1 a L2 jsou celá čísla a toto omezení hraje hlavní roli při procesu řešení nejednoznačnosti při stanovení nejednoznačností jako známé číslo. Ve skutečnosti však existuje mnoho zdrojů systematických chyb, které nám nepříznivě ovlivní měření. Pokud tyto vlivy nemohou být odstraněny, nebo alespoň redukovány na úroveň kdy jsou nedůležité, ve vztahu k vlnovým délkám L1 a L2, nejednoznačnostem nemohou být spolehlivě přiřazeny celočíselné hodnoty.

*Omezení na krátké vzdálenosti* – Ionosférické a troposférické zpoždění a vlivy dráhy satelitu mohou být při použití dvojité diferencovaných měření zanedbány ve vztahu k procesu rozlišení nejednoznačnosti. Toto omezení činí rovnice měření pro krátké vzdálenosti mnohem jednodušší než u měření na velkou vzdálenost.

*Omezení nosné L1 a L2* – pokud mohou být zanedbány vlivy zpoždění v atmosféře, vícecestné šíření a šum měření, měření na nosné L1 a L2 a měření pseudo-vzdáleností jsou vystavené stejným nežádoucím vlivům. Toto omezení má vliv na to, že při měření na dvou frekvencích stačí polovina nejednoznačností. Pokud výše uvedené vlivy nemohou být zanedbány, nosné na L1 a L2 musejí mít velké korelace a tyto korelace mohou být užitečné pro rozlišení nejednoznačností. Toto omezení může být aplikováno přímo pomocí metody nejmenších čtverců, pomocí lineárních kombinací dvoufrekvenčního měření nosné, nebo pomocí LAMBDA transformací.

*Geometrické omezení skutečných satelitů* – Geometrické omezení satelitů jsou měřena pomocí různých faktorů jako jsou GDOP, PDOP, atd.. Tyto měření jsou v prvním případě využívána pro bodové (absolutní) měření polohy za užití pseudo vzdáleností. Pokud má referenční přijímač známé souřadnice, tyto měření mohou být také využity pro popis kvality skutečné geometrie satelitu vzhledem k relativní poloze, stejně jako pro měření nosné pokud jsou nejednoznačnosti předurčené (neboli „rozhodnuté“).

*Omezení změny geometrie satelitů* - Pokud se žádná pozice satelitu nemění, proces řešení nejednoznačnosti využívající pouze měření nosné, nemůže být provedeno pokud nemáme k dispozici velmi přesný přijímač a souřadnice satelitu ( typicky lepší než 5 cm). Naštěstí se pozice satelitu mění s časem a dopplerovské měření může být charakterizováno jako difference ve vzdálenostech od přijímače k satelitu mezi jednotlivými epochami. Této vlastnosti se používalo u navigačního systému Transit Doppler System.

## 1.3.2 Techniky řešení nejednoznačnosti

Nejjednodušší procedury pro zjištění nejednoznačností je buď využití trojitě diferencované měření, odstraňující podmínky nejednoznačnosti z rovnice měření, nebo proces měření dvojitých diferencí s odhadovanými nejednoznačnostmi ze skutečných parametrů (toto řešení se někdy označuje jako “ambiguity-free“ nebo “bias-float“ řešení). V důsledku aplikace pouze *Omezení změny geometrie satelitů* v těchto dvou procedurách, potenciální přesnost měření nosné není plně využívána. Dřívější metoda používaná k určení celočíselné nejednoznačnosti, vyžadovala odhad skutečné hodnoty nejednoznačnosti aby mohla být zaokrouhlena k nejbližšímu celému číslu, ale to záleželo na přesnosti s jakou mohla být odhadnuta. Odstranění problémů s nepřesností, se začalo dařit v poslední dekádě, kdy byly vynalezeny nové metody. Celkově lze říci, že následující metody řešení nejednoznačnosti (ambiguity resolution - AR), mohou být identifikované, s výjimkou výše uvedených trojitě diferencovaných řešení a dvojitě diferencovaných plovoucích řešení:

- AR pomocí speciálních operačních módů jako je inicializace výměny antén (antenna swap initialization) re-occupation a “stop&go“ procedury.
- AR užívající přímá data pseudo vzdálenosti které se používají jako lineární kombinace L a L2 nosné ( s dlouhou vlnovou délkou), a lineární kombinace L1 a L2 s P-kódem pro pseudo vzdálenost ( s nižším šumem).
- AR pomocí hledání v doméně souřadnic “Coordinate Domain“ jako například v případě funkčních metod nejednoznačnosti (Ambiguity Function Method).
- AR pomocí hledání v doméně odhadovaných nejednoznačností “Estimated Ambiguity Domain“, kde se prvně vezme odhadovaná hodnota jako reálný parametr, a dále jako určující celočíselná nejednoznačnost v procesu nejmenších čtverců.

### 1.3.2.1 Operační módy

*Antenna Swap* – Neboli výměna antén je popsána následovně. Umístíme první GPS anténu na místo A a druhou na místo B. Místa A a B jsou od sebe vzdálena několik metrů. Ve chvíli kdy sledujeme alespoň čtyři satelity (doporučováno je pět a více) přemístíme anténu 1 na místo B, A anténu 2 na místo A, a dáваме pozor aby se nepřerušilo sledování nosné (aby nenastal fázový skok). Nyní můžeme během několika sekund měřit vektor mezi dvěma neznámými body s přesností, která dosahuje několik milimetrů. Tímto způsobem může být nejednoznačnost efektivně eliminována z rovnic měření. Tato metoda využívá *Omezení změny geometrie satelitů* a *Omezení na krátké vzdálenosti*. Z důvodu nutnosti neztratit sledovaný signál během přesouvání antén, je tato technika používána pro statickou inicializaci pro metodu “stop&go“.

*Stop & Go* - Pokud GPS anténa dokáže zachytit a udržovat přijímaný signál, hodnota nejednoznačnosti se nezmění dokud se anténa nezačne pohybovat. Pokud je číselná nejednoznačnost určena na určitém místě, nejednoznačnost bude stejná na ostatních místech dokud neztratíme signál. Hodnota nejednoznačnosti na

prvním místě může být předurčena například předchozí metodou výměny antén, nebo jinou metodou AR.

*Re-Occupation* – Ke spolehlivému odhadu skutečné hodnoty nejednoznačnosti je třeba asi hodina pozorování. Důvod je znám už dlouho: důležité je, že se geometrie satelit-přijímač mění. A jelikož během čekání na změnu geometrie můžeme navštívit jiné místo, je možné na druhém místě také provést měření, atd... Tento mód vyžaduje, aby každá základna byla obsazena alespoň dvakrát a aby interval mezi dvěma obsazeními základen byl půl hodiny.

### 1.3.2.2 Hledání v doméně pozorování “Observation Domain“

Nejednoznačnosti mohou být předurčeny přímo použitím měření pseudo vzdálenosti a nosné. Bohužel přesnost C/A a P kódů u pseudo vzdáleností není příliš dobrá, je určená vlnovými délkami nosných L1 (19.03cm) a L2 (24.42cm). Proto tedy číselná lineární kombinace L1 a L2 pozorování, které mají relativně dlouhou vlnovou délku, má nízký šum a přiměřeně malé ionosférické zpoždění, jako například “wideline“ ( $\Phi_1 - \Phi_2$ , s vlnovou délkou 0.86 cm), “narrowline“ ( $\Phi_1 + \Phi_2$ , s vlnovou délkou 0.11 m), nebo “extra-wideline“ ( $-3\Phi_1 + 4\Phi_2$ , s vlnovou délkou 1.63 m), a kombinace s nejdelší vlnovou délkou ( $-7\Phi_1 + 9\Phi_2$ , které se používají. Pokud je základna dostatečně krátká, rozdílné ionosférické zpoždění může být ignorováno a P-kódová pseudovzdálenost může být použita pro předurčení počáteční číselné nejednoznačnosti kombinovaného měření, a následně rozlišit nejednoznačnosti na nosných L1 a L2.

Pro mnoho aplikací není základna natolik krátká, aby bylo možné zanedbat rozdíly ionosférického zpoždění, proto musí být použita tzv “ionosphere-free“ kombinace daná jako:

$$\Delta\nabla N_{1,-1} = \Delta\nabla\Phi_1 - \Delta\nabla\Phi_2 - \frac{f_1 - f_2}{f_1 + f_2} \left( \frac{\Delta\nabla R_1}{I_1} + \frac{\Delta\nabla R_2}{I_2} \right) \quad \text{r.2}$$

Kde  $\Delta\nabla$  je operátor dvojitě diference,  $\Phi_1, \Phi_2, R_1, R_2$  a  $N_{1,-1}$  jsou nosné frekvence L1 a L2, respektive pseudo vzdálenosti měřené na L1 a L2, a číselná nejednoznačnost “wideline“. Tato technika využívá *Celočíselné omezení, Omezení nosné L1 a L2* a dále vyžaduje přesné měření pseudo vzdáleností L1 a L2.

### 1.3.2.3 Hledání v doméně souřadnic “Coordinate Domain“

Pokud je odhad počátečních souřadnic dostatečně přesný, číselná nejednoznačnost může být počítána za použití počátečních souřadnic. To znamená, že pokud lze počáteční souřadnice dobře aproximovat, hodnota nejednoznačnosti může být předurčena pouze v jedné epoše, nebo několika málo epochách. Pro některé aplikace, jako například sledování deformací, je tato metoda dostačující. Poté co je množina nejednoznačností předurčena, může být základna odhadnuta s velkou přesností.

Nicméně pokud jsou počáteční souřadnice nepříznivě ovlivněny (na hodnotu několika decimetrů), nejednoznačnost nemůže být předurčena správně pomocí souřadnic. Eventuelně

však, hledací procedura může být aplikována. Často užívaná metoda je známá jako “Ambiguity Function Metod“ (AFM). Funkce nejednoznačnosti je definována jako:

$$AF(X) = \sum_{k=1}^m \sum_{l=1}^n \left| \sum_{j=1}^{m_{kl}} \exp\left(i \left( \Delta\Phi_{obs}^{jl}(k) - \Delta\Phi_{calc}^{jl}(k, X) \right)\right) \right| \quad r.3$$

Kde  $\Delta\Phi_{obs}^{jl}(k)$  a  $\Delta\Phi_{calc}^{jl}(k, X)$  jsou jednoduše diferencované nosné, měřené mezi přijímači a jejich spočítané hodnoty, za použití základního vektoru  $\mathbf{X}$ , vzhledem k satelitu  $\mathbf{j}$ , frekvenci  $\mathbf{f}$  a epoše  $\mathbf{k}$ .  $m_{kl}$  je počet jednoduše diferencovaných měření nosné, když  $\mathbf{m}$  a  $\mathbf{n}$  jsou počet epoch respektive frekvencí. Zvláštní základní vektor  $X$  představuje rozdílnou hodnotu funkce nejednoznačnosti. Při hledání se snažíme nalézt takový vektor  $X$ , který způsobí maximální hodnotu funkce nejednoznačnosti.

Tato technika využívá všechny typy omezení, ale *Geometrické omezení skutečných satelitů* je aplikováno jen málo, protože měření nosné je využíváno jen okrajově. Výhoda této metody je, že není citlivá na velikost číselné nejednoznačnosti nebo výsky fázových skoků.

#### 1.3.2.4 Hledání v doméně odhadovaných nejednoznačností “Estimated Ambiguity Domain“

Tato metoda hledání se skládá ze dvou kroků. První krok je získání plovoucího řešení “float“, kde jsou nejednoznačnosti získány pomocí odhadu metodou nejmenších čtverců : odhadovaná skutečná hodnota nejednoznačnosti  $\hat{X}_N$  a její kovarianční matice  $Q_{\hat{X}_N}$ . Druhý krok je hledání množiny nejednoznačností které splňují následující relaci:

$$R_k = (\hat{X}_N - N_k)^T Q_{\hat{X}_N}^{-1} (\hat{X}_N - N_k) = \min \quad r.4$$

Derivace a kritéria platnosti budou vysvětleny v následující kapitole.

Několik metod rychlého hledání nejednoznačností je navrhováno na základě této relace. Například tato relace je jako základ u metody která se nazývá “Least-squares AMBIGUITY Decorrelation Adjustment (LAMBDA) metod“.

Tato metoda využívá všechny omezení pro měření krátkých vzdáleností. Lze tuto metodu také využít i pro dlouhé vzdálenosti, ale pak již nelze využívat výhod *Omezení na krátké vzdálenosti*.

## 1.4 Komunikace se senzorem AC12 – Datové zprávy protokolu NMEA

### 1.4.1 Formát zpráv

Senzor AC12 má pro komunikaci s okolím k dispozici dva porty RS-232 (A a B), které slouží jak pro komunikaci s externím řídicím zařízením (např. PC), tak i pro získání diferenčních korekcí z referenční stanice.

#### Vstupní zprávy

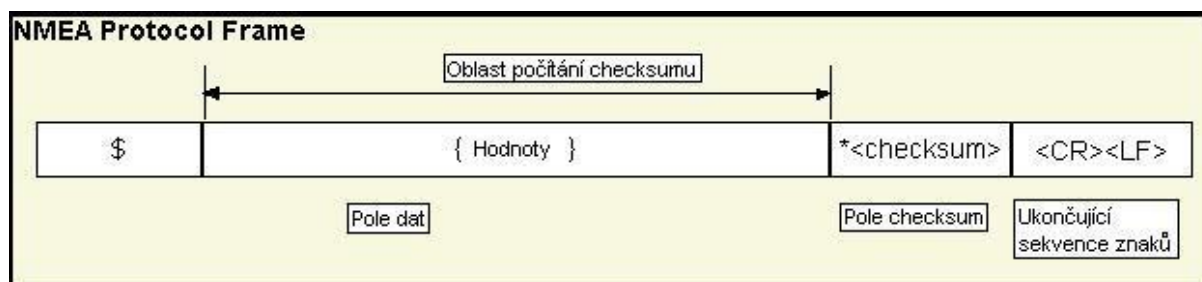
Vstupní zprávy obsahují dva typy zpráv. Jednak jsou to nastavující příkazy a za druhé jsou to příkazy pro dotaz. Nastavující příkazy dávají pokyny senzoru pro vykovávání určené aktivity, většinou kontinuálně, zatímco příkazy pro dotaz instruují senzor podávat zprávy v současném čase pouze jednotlivě. Celkově mají příkazy následující formát, který je v souladu se standardem NMEA 0183:

- NMEA 0183 ASCII řetězec začínající znakem \$
- Hlavičky jsou ve tvaru Ashtech NMEA
- ID zpráv jsou ve tvaru Ashtech NMEA
- Jednotlivé data jsou oddělovány čárkami
- Hexadecimální checksum je počítán jako součet XOR ze všech znaků zpráv mezi znaky \$ a \*
- Zprávy jsou ukončeny dle standardu NMEA znaky <CR> a <LF>

#### Výstupní zprávy

Výstupní zprávy jsou zprávy, které senzor posílá do počítače, nebo jiného elektronického zařízení jako odpovědi na řídicí zprávy. Zpráva se skládá ze souhrnné zprávy o stavu, řídicí zprávy oznamující potvrzení přijetí a zprávy dat GPS. Souhrnné zprávy o stavu mají volnou formu Ashtech NMEA. Další zprávy jsou již v souladu NMEA 0183 a to následovně:

- NMEA 0183 ASCII řetězec začínající znakem \$
- Hlavičky jsou ve tvaru Ashtech NMEA
- ID zpráv jsou ve tvaru Ashtech NMEA
- Standardní formát NMEA zpráv obsahuje byty hexadecimálního checksum
- Jednotlivá data jsou oddělovány čárkami, po sobě jdoucí čárky indikují špatná, nebo chybějící data(nulová pole)
- Zprávy jsou ukončeny dle standardu NMEA znaky <CR> a <LF>



**Obr.1 – NMEA Protokol**

Řídící zprávy NMEA (NMEA message commands) řídí všechny dotazy a nastavuje příkazy týkající se zpráv NMEA a různých ostatních zpráv ve formátu Ashtech NMEA. Všechny standardní NMEA zprávy jsou řetězec ASCII znaků oddělovaných čárkami, vyhovující standartu **NMEA 0183 ver. 3.0**. Ostatní (nestandardní) zprávy jsou také řetězec ASCII znaků oddělovaných čárkami, ale ve formátu Ashtech NMEA.

Pro každý typ NMEA zprávy existuje příkaz pro nastavení (set command), příkaz pro dotaz (query command) a zpráva obsahující odpověď (response message). Nastavující příkaz vyvolá odpovědní zprávu NMEA trvale ve specifické době. Dotaz vyvolá odpověď pouze jednou.

## Nastavující příkazy – Set commands

Základní struktura NMEA set commands je následující:

**\$PASHS,NME,str,x,s,d<Enter>**

Kde x.....sériový port na který má být odeslaná odpověď  
s.....ON /OFF – zapnutí /vypnutí zprávy  
d.....doporučený parametr pro specifikování intervalu odpovědí  
str.....3-znakový řetězec který identifikuje o jakou NMEA zprávu se jedná.

## Příkazy pro dotaz – Query commands

Základní struktura pro NMEA query command

**\$PASHQ,str,x,<Enter>**

Kde x.....sériový port na který má být odeslaná odpověď  
str.....3-znakový řetězec který identifikuje o jakou NMEA zprávu se jedná.

Možnosti výběru řetězců (NMEA zpráv) jsou následující : **ZDA, GGA, GLL, GSA, GSV, MSG, POS, RMC, SAT, VTG, ALM, RRE, UKO, CRT, XMG, UTM**



**Odpovědní zprávy** mají svůj tvar v závislosti na výběru řetězce (měření), pro který jsme psali nastavující nebo dotazovou zprávu.

Výše popsané možnosti jsou základem protokolu NMEA, který podporuje senzor AC12. Pro naše účely, tj. výpočet polohy pomocí fázového měření však potřebujeme takové zprávy, které obsahují údaje o azimutu, elevaci a měření fáze nosné vlny satelitu na který jsme zaměřeni. Takovéto informace však základní protokol neobsahuje. Proto pro naše měření musíme využít rozšiřující protokol NMEA, který umožňuje zobrazovat výstupní surová měření, mezi nimiž je právě i měření nosné frekvence.

Příslušné příkazy využívající rozšířené NMEA jsou následující:

**\$PASHS,NME,PBN**  
**\$PASHS,NME,MCA**  
**\$PASHS,NME,SNV**

V dalším textu se zaměřím na zprávy MCA, které obsahují všechny pro nás důležité údaje.

## 1.4.2 Zprávy MCA

Zprávy MCA jsou pro uživatele dostupné pouze sleduje li přijímač nějaký satelit. Tato vlastnost je logická, protože MCA zprávy obsahují měřicí informace o družici jako je např. dopplerovské měření, hrubá pseudovzdálenost, PRN satelitu, elevaci a azimut. Pro každý sledovaný satelit je samostatná MCA zpráva.

Struktura MCA zprávy:

**\$PASHR,MCA<datový řetězec Ashtech typ 3 + kontrolní součet>**

V tabulce 1 jsou definovány parametry zpráv MCA

| Typ pole                         | Velikost (bytes) | Obsah pole   |
|----------------------------------|------------------|--|
| unsigned short<br>[sequence tag] | 2                | Posloupnost ID čísel v jednotkách 50 ms, modulo 30 minut.  |
| unsigned char<br>[left]          | 1                | Počet zbývajících zpráv MCA, které mají být poslány v dané epoše. Senzor AC 12 vysílá zvláštní MCA zprávu pro každý sledovaný satelit. |
| unsigned char<br>[svprn]         | 1                | Číslo satelitu – PRN (1 – 56). Toto číslo je uvedeno ve všech následujících zprávách které obsahujících různá měřená data              |

|  |           |  |
|--|-----------|--|
|  |           | satelitu,takže lze okamžitě zjistit pro jaký satelit tato měřená data jsou.  |
| unsigned char<br>[elev]                            | 1         | Elevační úhel ( ve stupních ) sledovaného satelitu. Vztaženo k [svprn].  |
| unsigned char<br>[azim]                            | 1         | Azimut sledovaného satelitu po 2°. Vztaženo k [svprn].   |
| unsigned char<br>[chnind]                          | 1         | Kanál (1 – 12) přidělený satelitu. Vztaženo k [svprn].   |
| <b>Měřená data odvozená z C/A kódu ( 29 bytes)</b> |           |  |
| unsigned char<br>[warning]                         | 1         | Viz tabulka 2.   |
| unsigned char<br>[goodbad]                         | 1         | Indikuje kvalitu pozičního měření: <ul style="list-style-type: none"> <li>• 0 – Měření není dostupné, žádná další data nemohou být poslána</li> <li>• 21 – Satelit je pod elevační maskou PEM</li> <li>• 22 – Probíhá kódové a/nebo fázové měření</li> <li>• 23-Probíhá kódové a/nebo fázové měření, a byla získána navigační zpráva, ale měření není možné použít k výpočtu polohy.</li> <li>• 24-Probíhá kódové a/nebo fázové měření, a byla získána navigační zpráva, která je použita pro výpočet polohy.</li> </ul> |
| Char<br>[polarity_known]                           | 1         | Toto číslo je buď 0 nebo 5 <ul style="list-style-type: none"> <li>• 0 – satelit je již uzamčen</li> <li>• 5 – byla nalezena pouze první část navigační zprávy</li> </ul>   |
| unsigned char<br>[ireg]                            | 1         | Měření odstupu signál – šum (v db Hz) pro určený satelit. Vztaženo k [svprn].  |
| unsigned char<br>[qa_phase]                        | 1         | Nevyužíváno. Většinou je to nula   |
| double<br>[full phase]                             | 8         | Měření celé hodnoty vlnové délky (fáze) nosné vlny satelitu (měřeno v cyklech). Vztaženo k [svprn].  |
| double<br>[raw_range]                              | 8         | Hrubá vzdálenost (v sekundách) k satelitu. Vztaženo k [svprn].<br>Vzdálenost je počítána dle následujícího vztahu :<br>čas přijímače – vysílaný čas = hrubá vzdálenost   |
| long<br>[doppler]                                  | 4         | Dopplerovské měření (v 10 <sup>-4</sup> Hz) satelitu. Vztaženo k [svprn].  |
| long<br>[smoothing]                                | 4         | Bity 31-24 reprezentují [smooth_count]. Jsou neoznačené a normalizované a indikují činitel vyhlazení. <ul style="list-style-type: none"> <li>• 0 – nevyhlazený</li> <li>• 1 – nejméně vyhlazený</li> <li>• 100 – nejvíce vyhlazený</li> </ul> Bity 23-0 reprezentují [smooth_cor]. Bit 23 (most significant bit - MSB) je značkový a bity 22 až 0 jsou least significant bits(LSB) reprezentující velikost korekce v centimetrech.   |
| unsigned short<br>[checksum]                       | 1         | Kontrolní součet je logická operace exkluzive OR (XOR) všech bytů od sequence_tag ( hned pod hlavičkou) do posledního bytu [smoothing].  |
| <b>Celkem bytů</b>                                 | <b>37</b> |  |

V tabulce 2 jsou definovány formáty výstražných vlajek MCA

| Index Bitů |   | Popis  |
|------------|---|--|
| 1          | 2 | Kombinace bitu 1 a bitu 2.   |
| 00         |   | Probíhá Kódové a/nebo fázové měření pro příslušný satelit. Vztaženo k [svprn].   |
| 01         |   | Probíhá kódové a/nebo fázové měření pro příslušný satelit a navigační zpráva byla získána, ale tato data nemohou být použita pro výpočet polohy. Vztaženo k [svprn].   |
| 10         |   | Probíhá kódové a/nebo fázové měření pro příslušný satelit, navigační zpráva byla získána a data byly použity pro výpočet polohy.                                       |
| 1          | 2 | Kombinace bitu 1 a bitu 2.   |
| 3          |   | Symboly v navigační zprávě nebyly synchronizovány.   |
| 4          |   | Měření pseudo-vzdálenosti není vyhlazené.  |
| 5          |   | Rezervováno.   |
| 6          |   | Při kódovém a/nebo fázovém měření se objevila ztráta signálu   |
| 7          |   | Rezervováno.   |
| 8          |   | Vyskytla se ztráta kontinuity (tato výstražná vlajka se vyskytne když přijímač znovu získá signál pro kódové a/nebo fázové měření. Také se objeví po známé polaritě.). |

Výstup MCA zpráv je v binárním formátu, v souladu s tím jak byly nastaveny nahrávací intervaly (\$PASH,NME,PER). Jedna výstupní MCA zpráva je jedinečná pro každý sledovaný satelit.

## 1.5 Metoda nejmenších čtverců

### Metoda nejmenších čtverců obecně :

Je to matematická metoda, která je určena ke statistickému zpracování dat. Úkolem této metody je nalézt vhodnou aproximační funkci pro dané empiricky zjištěné hodnoty. Obecně lze říci, že metoda nejmenších čtverců slouží k nalezení takového řešení, aby součet druhým mocnin chyb nalezeného řešení byl minimální.

### Aplikace metody nejmenších čtverců na lineární závislost ( metody lineární regrese):

Uvažujeme funkční závislost :  $f(x) = ax + b$  r.5

Součet čtverců pak bude vypadat následovně:

$$S(a, b) = \sum_{i=1}^n [f(x_i) - y_i]^2 = \sum_{i=1}^n (ax_i + b - y_i)^2 \quad \text{r.6}$$

Abychom našli minimum součtu (koeficienty a,b tak, aby nalezená závislost vhodně aproximovala daná data), položíme obě parciální derivace součtu čtverců rovny nule:

$$0 = \frac{\partial S}{\partial a} = 2 \sum_{i=1}^n (ax_i + b - y_i) x_i \quad \text{r.7}$$

$$0 = \frac{\partial S}{\partial b} = 2 \sum_{i=1}^n (ax_i + b - y_i) \quad \text{r.8}$$

Po úpravách získáme soustavu:

$$a \sum_{i=1}^n x_i^2 + b \sum_{i=1}^n x_i = \sum_{i=1}^n x_i y_i \quad \text{r.9}$$

$$a \sum_{i=1}^n x_i + bn = \sum_{i=1}^n y_i \quad \text{r.10}$$

A jejím řešením pro konkrétní hodnoty  $x_i, y_i$  dostaneme námi hledané hodnoty parametrů  $a$  a  $b$ .

$$a = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{n \sum x_i^2 - (\sum x_i)^2} \quad \text{r.11}$$

$$b = \frac{\sum x_i^2 \sum y_i - \sum x_i \sum x_i y_i}{n \sum x_i^2 - (\sum x_i)^2} \quad \text{r.12}$$

Lze dokázat, že matice takovéto soustavy je regulární pro všechna  $n \geq 2$ , a má tedy právě jedno řešení. Obecně lze také ukázat, že v tomto bodě je součet čtverců minimum. Tento postup lze aplikovat na jakýkoliv druh závislosti i více proměných.

### Metoda nejmenších čtverců a soustava lineárních rovnic

Uvažujeme soustavu lineárních rovnic ve tvaru  $\mathbf{Ax}=\mathbf{b}$  ( $\mathbf{A}$  je známá matice,  $\mathbf{b}$  je známý vektor a  $\mathbf{x}$  je neznámý vektor). Za těchto podmínek mohou nastat následující tři situace :

- Matice  $\mathbf{A}$  je čtvercová, vektory  $\mathbf{b}$  a  $\mathbf{x}$  mají stejný rozměr, tak je pro výpočet k dispozici stejný počet neznámých jako rovnic. Zde lze  $\mathbf{x}$  vypočítat třeba pomocí inverzní matice  $x = A^{-1}b$
- Matice  $\mathbf{A}$  obsahuje více sloupců než řádků, vektor  $\mathbf{b}$  má menší rozměr než vektor  $\mathbf{x}$ . V tomto případě se jedná o *nedourčenou soustavu* rovnic. Tato soustava má více neznámých než rovnic
- Matice  $\mathbf{A}$  obsahuje více řádků než sloupců, vektor  $\mathbf{b}$  má větší rozměr než  $\mathbf{x}$ . Zde je více rovnic než máme neznámých a jedná se o *přeurčenou soustavu* rovnic.

U případů s nedourčenou a přeurčenou soustavou rovnic platí, že soustava má nekonečně mnoho (nebo žádné) řešení. To je však pro nás nedostačující zjištění a tak musíme definovat chybu řešení  $e$ .

$$e = Ax - b \quad \text{r.13}$$

Budeme se snažit najít takové řešení  $x$ , aby vektor  $e$  (čili chyba) byl minimální. Jedna z možností jak najít takové řešení je derivace matic a vektorů.

### Derivace matic a vektorů

Jelikož  $e$  je vektor, upravíme tento požadavek tak, aby součet složek vektorů (neboli součet čtverců jednotlivých odchylek) byl minimální. Takto se vlastně jedná o minimalizaci skalárního součinu, který lze zapsat pomocí transpozice.

$$e \cdot e = e^T e \rightarrow \min$$

Součin bude minimální pokud jeho derivace podle proměnné  $\mathbf{x}$  bude rovna nule.

$$(e^T e)' = \left[ (Ax - b)^T \cdot (Ax - b) \right]' = 0 \quad \text{r.14}$$

Po dalších úpravách dle pravidel pro transpozici součinu a derivaci součinu vektorů a matic.

$$\left[ x^T A^T Ax - x^T A^T b - b^T Ax + b^T b \right]' = 2 A^T Ax \quad \text{r.15}$$

Z tohoto vztahu již lze vyjádřit výsledný vzorec pro  $\mathbf{x}$ .

$$x = (A^T A)^{-1} A^T b$$

r.16



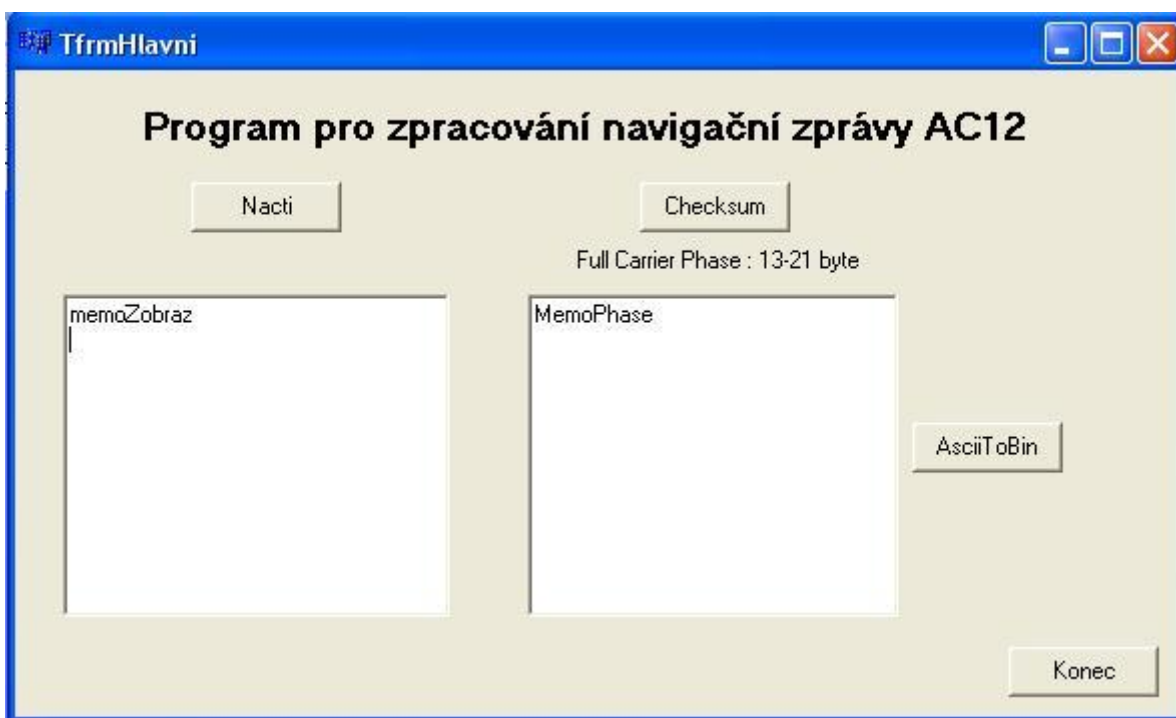
Dále následuje hlavní for cyklus jehož funkce je počítání součtu XOR z 36 načtených bytů.

Po tomto součtu následuje porovnání s 37 bytem, který bereme jako checksum a jestliže rovnost nesouhlasí program se vrací zpět k prvnímu cyklu for, který nám nastaví kurzor pro načtení znaků a jeden dál.

Jestliže rovnost souhlasí, program pokračuje načtením dat, která představují fázi nosné vlny. Tyto data podle protokolu NMEA začínají na 13 bytu a mají velikost 8 bytů. Program tedy nastaví kurzor pro čtení ze souboru na 12 byte a načte následujících 8 do bufferu. Tento buffer následně uloží do souboru určeného těmito datům (FullPhase.txt). Navíc se tato data ještě pro kontrolu zobrazí na komponentě Memo.

Na konci běhu programu tedy máme v souboru FullPhase.txt uloženy hodnoty fáze nosné vlny, které tak můžeme využít pro další práci, kterou je aplikace metody nejmenších čtverců.

Dále je ještě v programu funkce která převádí ascii znaky na binární, což lze použít například pro kontrolu či další zpracování.



- Tlačítko Nacti – načte navigační zprávu a zobrazí v komponentě memoZobraz. Lze nastavit kolik bytů chceme načíst, určitý počet nebo do konce souboru
- Tlačítko btnChecksum – nalezne fázi nosné pomocí synchronizace s checksumem a uloží ji do souboru, a zároveň ji zobrazí v komponentě MemoPhase. Lze nastavit kolik bytů chceme projít, určitý počet nebo do konce souboru
- Tlačítko AsciiToBin – převede načtená data v komponentě MemoPhase z řetězce ascii znaků na binární
- Tlačítko Konec – ukončí aplikaci



## 2.2 Aplikace metody nejmenších čtverců

Pro aplikaci metody nejmenších čtverců je důležité si uvědomit, že prostředí Builder C++ samo o sobě nepodporuje operace s maticemi. Proto je třeba využít některé speciální knihovny, které tuto možnost mají. Při práci probíhali pokusy se dvěma takovými knihovnami.

Jedna z nich je knihovna VectorSpace C++ Library. Tato velmi rozsáhlá knihovna přímo podporuje všechny algebraické operace včetně maticového počtu. Při použití této knihovny pak velmi jednoduše lze počítat metodou nejmenších čtverců podle definice a to například následovně:

```
#include "vs.h"                                     - Vložení knihovny
                                                    umožňující maticové
                                                    operace

int main() {

double b[] = {,,,,,,,,},
a[][] = { {,}, {,}, {,}, {.,}{,}, {,}, {,}, {,}, {,}, {,}, {,}};

C0 A(, a[]), B(,);

Matrix::Decomposition_Method = Matrix::Cholesky_Decomposition;

C0 AtA = ((~A)*A) ;

x = ((~A)*B) / AtA;

cout << x << endl;

return 0;
}
```

- definice matic  
přeurčené soustavy  
rovníc

- Využití choleského  
dekompozice pro  
výpočet inverzní  
matice

- výpočet  
( $A^T A$ )

- výpočet jako v  
r.16  
$$x = (A^T A)^{-1} A^T b$$

Další možností práce s vektory a maticemi je využití skupiny knihoven LAPACK C++. Tyto knihovny přináší do C++ všechny důležité algebraické operace. Po nahrání již teoreticky lze všechny tyto operace provádět v podstatě stejně jak bylo ukázáno v minulém případě protože obě knihovny by měly umožňovat stejné operace.

### 3 Zhodnocení

Jako první začala práce na vytvoření komunikace mezi počítačem a senzorem. Nakonec se po dohodě rozhodlo, že komunikace a nahrání dat bude prováděno pomocí oficiálního softwaru evaluate 5.0, který je výrobcem dodáván. Pomocí tohoto programu byla nahrána navigační zpráva MCA, k jejímuž zpracování byl vytvořen příslušný program v prostředí Builder C++.

Tento program při zpuštění nevykazuje žádnou chybu a bez problémů se zpusť. Také při stisknutí tlačítek se vykonají události, které se mají vykonat. Při testování byl spuštěn program krok po kroku a všechno fungovalo dle požadavků.

Nicméně ale výsledek, nahrání fáze nosné vlny, příliš úspěšný není. Jistá data se do souboru načtou, ale není jich tolik kolik by mělo. V navigační zprávě MCA by se tato data měla vyskytovat periodicky, a hlavně od prvního sesynchronizování (pomocí kontrolního checksumu) až do konce zprávy.

Bohužel se nepodařilo přijít na to, kde je chyba. V zásadě jsou dvě možnosti. Buď je možné, že i přes kontrolu je špatný algoritmus programu, či pouze některá jeho část. Druhou možností je, že již získaná navigační zpráva je chybná nebo poškozená.

Dalším úkolem bylo řešit soustavu lineárních rovnic metodou nejmenších čtverců také v prostředí Builder C++. Tato aplikace přímo nepodporuje takovéto algebraické operace, tak je nutné řešit tento problém pomocí některé z rozšiřujících knihoven.

Jako první pokusy probíhaly s množinou knihoven Lapack C++. Pokusy však nedopadly dobře, jelikož se nepodařilo tyto knihovny implementovat do našeho programu.

Dále se tedy hledala jiná možnost řešení problému a tak práce pokračovala alespoň teoreticky a to s knihovnou VectorSpace C++. To je knihovna, která umožňuje všechny algebraické operace, a hlavně maticový počet, což je pro metodu nejmenších čtverců nezbytné. Po nastudování možností této knihovny, tak byla metoda nejmenších čtverců řešena obecně, za předpokladu využití této knihovny.

Prakticky tento problém zatím řešen nebyl, protože tato knihovna není poskytována zdarma, ale za poměrně vysokou částku a tak zatím nebyla zakoupena licence.

## Použitá Literatura

- [1]: Rapant,P.: Družicové polohové systémy. Vysokoškolské skriptum, Vysoká škola báňská-Technická univerzita Ostrava, 2002, 200 stran.
- [2]: Hrdina, Z., Pánek,P., Vejražka, F. : Rádiové určování polohy. (Družicový systém GPS). Vysokoškolské skriptum, ČVUT Praha, 1996, 267 stran.
- [3]: Caroline Ericsson : Investigations of C/A Code and Carrier Measurements and Technique for Rapid Static Surveys. University of Calgary,1992,180 stran.
- [4]: Principles and Practice of GPS Surveying, School of Surveying & Spatial Informations Systéme, UNSW  
<http://www.gmat.unsw.edu.au/snap/gps>
- [5]: Oficiální stránky výrobce přijímače AC 12 – Thales Navigation  
<http://thalesgroup.com/navigation>
- [6]: Thales Reference manual - A12, B12, AC12
- [7]: Shaowei Han: Carrier Phase-Based Long-Range GPS Kinematic Positioning  
The University of New South Wales,Sydney

## Přílohy:

```
//-----  
//-----PROGRAM PRO ZPRACOVÁNÍ ZPRÁVY MCA PŘIJÍMAČE AC12-----  
//-----PETR LUKEŠ-----  
  
#include <vcl.h>  
#pragma hdrstop  
  
#include "NacteniAC.h"  
  
//-----  
  
#pragma package(smart_init)  
#pragma resource "*.dfm"  
TTfrmHlavni *TfrmHlavni;  
  
//-----  
__fastcall TTfrmHlavni::TTfrmHlavni(TComponent* Owner)  
    : TForm(Owner)  
{  
  
}  
//-----  
//-----FUNKCE-----  
//-----  
  
void showInBinary(unsigned char n) //funkce pro převod ASCII znaku na binární  
{  
    int i,pom;  
    char a,b;  
    for(i=7;i>=0;i--)  
    {  
        pom=1<<i;  
        a=(n&pom)==0?('0'),('1');  
        TfrmHlavni->MemoPhase->Lines->Text=TfrmHlavni->MemoPhase->Lines->Text + a ;  
    }  
}  
  
//-----  
//-----UDÁLOSTI NA BUTTONY-----  
//-----  
  
void __fastcall TTfrmHlavni::btnNactiClick(TObject *Sender) //Načtení řetězce ascii znaků  
//získaných ze senzoru ac12  
{  
int handle_souboru;  
  
if (!FileExists("ac12.txt"))  
    ShowMessage("soubor neexistuje");  
else
```

```

if((handle_souboru = FileOpen("ac12.txt",fmOpenRead))== -1)
    ShowMessage("Chyba pri otevirani souboru");
else{
    int precteno_znaku;
    int delka_souboru = FileSeek(handle_souboru,0,2);
    PChar buffer = StrAlloc(delka_souboru + 1);

    FileSeek(handle_souboru,0,0);

    FileRead(handle_souboru,buffer,5000);           //zobrazení 5000 bytů
                                                    //ze souboru ac12.txt

    memoZobraz->Lines->Text = buffer;

    StrDispose(buffer);
    FileClose(handle_souboru);

    }
}
//-----

void __fastcall TTfrmHlavni::btnAsciiToBinClick(TObject *Sender) //Převod ASCII znaků
{
    AnsiString EditText = MemoPhase->Lines->Text;           //na Binární formu
                                                            //Událost načte ascii znaky
                                                            //a převede na binární

    MemoPhase->Lines->Text = "";

    for (int i=1;i<=EditText.Length();i++)
    {
        showInBinary(EditText[i]);
    }
}
//-----

void __fastcall TTfrmHlavni::btnChecksumClick(TObject *Sender) //Výpočet checksum
{
    int handle_souboru;
    int handle_souboru_zapis;
    int a;

    if (!FileExists("ac12.txt"))
        ShowMessage("soubor neexistuje");
    else

    if((handle_souboru = FileOpen("ac12.txt",fmOpenRead))== -1)
        ShowMessage("Chyba pri otevirani souboru");
    else{
        int precteno_znaku;
        int j;
                                                    //deklarace proměnných

```

```

int delka_souboru = FileSeek(handle_souboru,0,2);
PChar bufferA = StrAlloc(delka_souboru + 1);
PChar bufferB = StrAlloc(delka_souboru + 1);
PChar bufferZapis = StrAlloc(delka_souboru + 1);
PChar pombuffer = StrAlloc(delka_souboru + 1);
PChar Spocchecksum = StrAlloc(delka_souboru + 1);
PChar Filechecksum = StrAlloc(delka_souboru + 1);

TfrmHlavni->EditChecksum->Text="";
MemoPhase->Lines->Text="";

FileSeek(handle_souboru,0,0);

for (j=0;j<=5000;j++){
    FileSeek(handle_souboru,j,0);
    //for cyklus který nastavuje kurzor
    //pro čtení ze souboru po každém cyklu
    //o jeden byte dal.Nastaveno 5000byte
    //ale lze dát do konce souboru

    for (int i=0;i<=35;i++)
    {
        //for cyklus pro načtení 36 bytů a
        //spočítání XOR z nich
        FileRead(handle_souboru,bufferA,1);

        *pombuffer=*bufferA^*pombuffer;

        *Spocchecksum=*pombuffer; //Spocchecksum obsahuje XOR z 36 bytů
        TfrmHlavni->EditChecksum->Text=*Spocchecksum;
    }

    FileRead(handle_souboru,bufferB,1); //Načtení 37 bytu jako checksum do
    *Filechecksum=*bufferB; //proměnné Filechecksum

    if (*Spocchecksum==*Filechecksum){ //pokud se checksum sečtený a nalezený
        //rovnají,načteme ze souboru byty 13-21
        FileSeek(handle_souboru,j+12,1); //což je námi hledaná fáze, dále tyto
        //byty ukládáme do souboru FullPhase a
        FileRead(handle_souboru,bufferZapis,8); //zároveň zobrazujeme v komponentě
Memo

        if (!FileExists("FullPhase.txt"))
            ShowMessage("soubor neexistuje");

        else

            if((handle_souboru_zapis = FileOpen("FullPhase.txt",fmOpenWrite))===-1)
                ShowMessage("Chyba pri otevirani souboru");
            else{
                //program zapíše byty 13-21 do souboru
                FileSeek(handle_souboru_zapis,0,2); //FullPhase.txt

                MemoPhase->Lines->Text = MemoPhase->Lines->Text +
                AnsiString(bufferZapis) + " ";
            }
        }
    }

```

```
FileWrite(handle_souboru_zapis,bufferZapis,8);

FileClose(handle_souboru_zapis);
}

}

}

FileClose(handle_souboru);
}
//-----
void __fastcall TTfrmHlavni::BtnKonecClick(TObject *Sender)
{
Application->Terminate();
}
//-----
```