

GEODETIČKÝ a KARTOGRAFIČKÝ



**Český úřad zeměměřický a katastrální
Úrad geodézie, kartografie a katastra
Slovenskej republiky**

10/06

Praha, říjen 2006
Roč. 52 (94) ● Číslo 10 ● str. 177–192
Cena Kč 14,-
Sk 27,-

GEODETICKÝ A KARTOGRAFICKÝ OBZOR

odborný a vědecký časopis Českého úřadu zeměměřického a katastrálního a Úřadu geodézie, kartografie a katastra Slovenskej republiky

Redakce:

Ing. Stanislav Olejník – vedoucí redaktor

Ing. Ján Vanko – zástupce vedoucího redaktora

Petr Mach – technický redaktor

Redakční rada:

Ing. Juraj Kadlic, PhD. (předseda), **Ing. Jiří Černohorský** (místopředseda), **Ing. Svatava Dokoupilová**, **Ing. Dušan Fičor**,
doc. Ing. Pavel Hánek, CSc., **prof. Ing. Ján Hefty, PhD.**, **Ing. Štefan Lukáč**, **Ing. Zdenka Roulová**

Vydává Český úřad zeměměřický a katastrální a Úřad geodézie, kartografie a katastra Slovenskej republiky v nakladatelství Vesmír, spol. s r. o., Na Florenci 3, 111 21 Praha 1, tel. 00420 234 612 395. Redakce a inzerce: Zeměměřický úřad, Pod sídlištěm 9, 182 11 Praha 8, tel. 00420 286 840 435, 00420 284 041 656, fax 00420 284 041 416, e-mail: stanislav.olejnik@atlas.cz a VÚGK, Chlumeckého 4, 826 62 Bratislava, telefon 004212 20 81 61 75, fax 004212 43 29 20 28. Sází VIVAS, a. s., Sazečská 8, 108 25 Praha 10, tiskne Serifa, Jinonická 80, Praha 5.

Vychází dvanáctkrát ročně.

Distribuci předplatitelům v České republice zajišťuje SEND Předplatné. Objednávky zasílejte na adresu SEND Předplatné, P. O. Box 141, 140 21 Praha 4, tel. 225 985 225, 777 333 370, 605 202 115 (všední den 8–18 hodin), e-mail send@send.cz, www.send.cz, SMS 777 333 370, 605 202 115. Ostatní distribuci včetně Slovenské republiky i zahraničí zajišťuje nakladatelství Vesmír, spol. s r. o. Objednávky zasílejte na adresu Vesmír, spol. s r. o., Na Florenci 3, POB 423, 111 21 Praha 1, tel. 00420 234 612 394 (administrativa), další telefon 00420 234 612 395, fax 00420 234 612 396, e-mail vanek@msu.cas.cz, e-mail administrativa: vorackova@msu.cas.cz, nebo imlaufova@msu.cas.cz. Dále rozšiřují společnosti holdingu PNS, a. s. Do Slovenskej republiky dováža MAGNET – PRESS SLOVAKIA, s. r. o., Šustekova 10, 851 04 Bratislava 5, tel. 004212 67 20 19 31 až 33, fax 004212 67 20 19 10, další čísla 67 20 19 20, 67 20 19 30, e-mail: magnet@press.sk. Předplatné rozšiřuje Slovenská pošta, a. s., Účelové stredisko predplatiteľských služieb tlače, Námestie slobody 27, 810 05 Bratislava 15, tel. 004212 54 41 99 12, fax 004212 54 41 99 06. Ročné predplatné 324,- Sk vrátane poštovného a balného.

Toto číslo vyšlo v říjnu 2006, do sazby v září 2006, do tisku 27. října 2006. Otisk povolen jen s udáním pramene a zachováním autorských práv.

© Vesmír, spol. s r. o., 2006

ISSN 0016-7096
Ev. č. MK ČR E 3093

**Přehled obsahu
Geodetického a kartografického obzoru
včetně abstraktů hlavních článků
je uveřejněn na internetové adrese
www.cuzk.cz**

Obsah

Ing. Michal Reháč	OSOBNÉ SPRÁVY	191
Využitie programovacieho jazyka JAVA v geodetických aplikáciách	NEKROLOGY	192
MAPY A ATLASY	OZNÁMENÍ	192
		177

Využitie programovacieho jazyka JAVA v geodetických aplikáciách

Ing. Michal Rehák,
katedra vyšší geodézie FSV ČVUT v Praze

528:800.92 JAVA

Abstrakt

Programovací jazyk JAVA, ako vhodný prostriedok na súčasné geodetické aplikácie. Úvod do programovacieho jazyka JAVA, zhrnutie základných črt jednotlivých foriem aplikácií. Porovnanie rýchlosti algoritmov invertovania matic, napísaných v jazykoch JAVA a C++. Testované algoritmy sú Gaussova-Jordanova eliminácia, LU-rozklad a Choleského rozklad. Použité knižnice: Matvec zo systému GAMA (C++) a balík JAMA (java). Popis aplikácie, zameranej na spracovanie observácií GPS metódou Precise Point Positioning, ktorá je napísaná v jazyku JAVA.

Use of Programming Language JAVA in Geodetic Application

Summary

Programming language JAVA as a convenient mean for present – day geodetic application. Introduction into the programming language JAVA. Baic sketches of individual forms of application. Comparison of velocities of algorithms for matrix inversions written in the languages JAVA and C++. Tested algorithms were as follows: Gauss – Jordan elimination, LU decomposition and Cholesky decomposition. Libraries used: Matvec of system GAMA (C++) and JAMA packet (java). Application in case of processing GPS observations using Precise Point Positioning method which is written in JAVA language.

1. Úvod

Základné črty jazyka JAVA boli sformulované v 80. rokoch minulého storočia a v roku 1991 bol vo firme Sun micro-systems založený projekt s názvom Stealth, z ktorého neskôr tento jazyk vznikol.

Pôvodné cieľové prostredie nového jazyka bola heterogénna sieť užívateľov, v ktorej by boli zapojené rôzne zariadenia a prostredníctvom JAVY by boli okrem samotných dát šírené zároveň algoritmy na prácu s týmito dátami. Z tejto myšlienky vychádzajú nasledujúce požiadavky:

- nezávislosť na platforme a type procesora. Z tohoto dôvodu je JAVA interpretovaný jazyk. Pre uľahčenie práce interpretora bol stvorený „bytecode“, do ktorého je prevádzaný zdrojový kód a tento bytecode sa posiela po sieti a následuje interpretuje,
- stabilita, minimalizovanie programátorských chýb. Preto bolo upustené od dedenia po viacerých rodičovských triedach súčasne a preťažovania operátorov. Navyiac bol zavedený implicitný prostriedok na upratovanie pamäte po ukončení používania objektu a nakoniec, kvôli zamedzeniu nebezpečných konštrukcií, bol nový jazyk obmedzený len na objektové programovanie,
- bezpečnosť. Bezpečnosť je obzvlášť dôležitý aspekt pri programovaní v sieťovom prostredí. To malo za následok vylúčenie pointrov, aby pri použití nového jazyka nebolo možné získať prístup k ľubovoľnej časti operačnej pamäte,
- jedným z menej dôležitých cieľov bola jednoduchosť a rýchle zvládnutie nového jazyka. Dosiahnutie tohoto cieľa sa dodnes pokladá za otázne.

Dnes disponuje jazyk JAVA bohatou výbavou, ktorá okrem základov umožňuje tvorbu grafického rozhrania, prácu s grafickými formátmi, komunikáciu na sieti, maticové operácie, kryptovanie, prácu s databázou, parsovanie XML a mnoho ďalších oblastí.

Interpretátori JAVY existujú na najpoužívanejších platformách (linux, windows).

2. Základné typy aplikácií v JAVE

V súčasnosti má jazyk JAVA viacero základných možností implementovania. Každá má svoje výhody a nevýhody.

2.1 Klasická samostatná aplikácia

Samostatná aplikácia s možnosťou grafického rozhrania je najjednoduchší spôsob využitia tohoto jazyka. Je potrebné mať kompilátor a interpret jazyka JAVA.

Spustený kód má z hľadiska prístupu a bezpečnosti rovnaké možnosti, ako ktorýkoľvek iný lokálne spustený program. Môže využiť všetky dostupné triedy, súbory, pripojenia.

2.2 Aplikácia na strane servera

V tomto prípade je navyiac potrebný JAVA-aplikačný server, na ktorom sú k dispozícii javovské triedy. Filozofia práce je založená na princípe klient-server. Teda klient vyšle požiadavku serveru, tento zavolá patričnú triedu a výsledok pošle naspäť klientovi. Obvykle má klient k dispozícii webový prehliadač a zobrazovaný výsledok je klasické HTML.

Rýchlosť pri takomto využití JAVY je závislá na rýchlosti spojenia so serverom a vyťaženosti servera. Z hľadiska bezpečnosti je užívateľ vystavený nebezpečenstvu podobnému, ako pri bežnom prezeraní internetových stránok. Interpretácia JAVY sa odohrá iba na serveri.

Malou obmenou tohoto spôsobu je využitie súborov JSP (JAVA server pages), ktoré majú syntax porovnateľnú s jed-

noduchými skriptovacími jazykmi, ako napríklad PHP. Pri prvej požiadavke od klienta na zobrazenie JSP stránky je tento súbor preložený do javovského zdrojového kódu a po skomplikovaní je interpretovaný, ako klasická napísaná trieda.

Oproti lokálnej aplikácii má trieda spustená na serveri k dispozícii navyše triedy pre komunikáciu so serverom. Jej práva a teda potenciaľná nebezpečnosť limitovaná nastavením servera.

2.3 Applet

Applet, teda aplikácia, bežiaci v prostredí webového prehliadača. Tento spôsob pomaly vychádza z módy. Podmienkou spustenia appletu je zaregistrovaný interpret JAVY vo webovom prehliadači (najčastejšie v podobe zásuvného modulu – pluginu). Užívateľ má k dispozícii grafické rozhranie.

Rýchlosť naštartovania appletu je závislá na rýchlosti pripojenia užívateľa k sieti a rýchlosti samotného appletu stojí na výkone užívateľovej stanice.

Applet je teda celý interpretovaný na strane užívateľa, čím predstavuje bezpečnostné riziko. Práva appletu ovplyvňuje jednak nastavenie interpretera, ktoré býva inštaláciou JRE (JAVA runtime environment) prednastavené na najmenej nebezpečnú konfiguráciu. Druhým dôležitým faktorom bezpečnosti je typ appletu. Applet môže byť podpísaný elektronickým podpisom. Bez ohľadu na to, či sa tento podpis dá alebo nedá overiť u prednastavenej certifikačnej autority, je užívateľ vyzvaný, aby prijal alebo zamietol tento podpis. V prípade zamietnutia bude applet zastavený. V opačnom prípade získa práva podobné právam lokálnej aplikácie a bude môcť napríklad pristupovať na lokálny disk či na sieť.

Dôvodom pre podpisovanie appletu môže byť napríklad požiadavka na širšie kompetencie appletu. Bežný applet napríklad nemôže písať užívateľov disk a nemôže komunikovať s iným serverom, než s tým, odkiaľ pochádza. Pokiaľ teda autor appletu potrebuje k jeho činnosti napríklad dáta s presnými dráhami družíc GPS (Global Positioning System), nepodpísaný applet mu túto možnosť poskytne len v prípade, ak sa bude nachádzať priamo na IGS serveri alebo ak budú tieto dáta umiestnené na serveri autora.

Pomôckou pri často aktualizovanom applete je JNLP (JAVA network launch packet). Prehliadač často uloží triedy appletu do dočasnej pamäte (cache) a pri nasledujúcom otvorení adresy s appletom sa niekedy spustí neaktuálna verzia. Filozofiou JNLP je zostavenie zoznamu obľúbených appletov na strane užívateľa a pri každom spustení appletu je najprv lokálna verzia konfrontovaná s verziou na serveri.

3. Rýchlosť JAVY

V prípade využitia JAVY na výpočtové práce je zásadnou otázkou jej rýchlosť.

Z princípu JAVY (interpretovanie bytecode) nutne vyplýva, že musí byť pomalšia, ako natívne skompilovaný kód. Tento fakt sa výrazne prejavil pri pokusom počítaní MD5¹⁾ kontrolnej sumy zo súboru s veľkosťou niekoľko stoviek MB. Na strane favorita bol program md5sum v prostredí linuxu, balík OpenSSL. Súperom bol Checksum.java, program v JAVE s obdobnou funkcionalitou, ktorý využíval triedu zo základnej výbavy JAVY. Výsledok: JAVA dosiahla približne trojnásobne dlhší čas spracovania. Rýchlosť v tomto prípade je navyše silne závislá na spôsobe čítania dát z disku.

Na počiatku bola JAVA oproti FORTRAN alebo jazyku

C tak pomalá, že jej nasadenie vo výpočtových aplikáciách nebolo zaujímavé. Za posledných 10 rokov sa JAVA výrazne zrýchli a predpokladá sa jej ďalší postup.

Vela negatívnych názorov na rýchlosť JAVY je založených na faktoch do roku 1995. V tej dobe sa napríklad pri kompikovaní JAVA-aplikácií nevyužívala technológia JIT (prekompilátor), ktorá bola nasadená v roku 1996.

Porovnávanie často nezohľadňuje použitie thread-save²⁾ technológie na strane JAVY, pričom je známe, že podobné ošetrenia algoritmov si nutne vyžadujú viac času. Podobné konštrukcie s časovou penalizáciou samozrejme existujú aj v jazyku C.

K ďalším nie práve korektným a aktuálnym základom pre hodnotenie rýchlosti JAVY patrí pomalosť a neúčinnosť GC³⁾, kompilovanie v čase exekúcie a iné.

Rýchlosť aplikácie však nie je daná iba verzitou kompilátora. Na zrýchľovanie aplikácií v JAVE existuje viacero stratégií [14, 15], napríklad:

Optimalizovanie kódu – znamená prepísanie väčšiny kódu. Napríklad zbytočne viacnásobné volania metód sa nahradia priradením výsledku prvého volania do lokálnej premennej a táto sa následne používa. Dokonale zoptimalizovaný kód ale býva horšie čitateľný a jeho vývoj a udržiavanie je náročnejšie. Často sa optimalizovaním vnesú drobné chyby, ktoré sa odhalia neskôr. Dokonca sa môže stať, že optimalizovaný kód bude na jednej platforme rýchlejší, ale na inej naopak stratí.

JNI – JAVA Native Interface, umožňuje volať natívne skompilované rutiny. Teda napríklad klasická JAVA-trieda spustí metódu alebo funkciu skompilovanej knižnice v jazyku C. Tento postup na jednej strane zrýchli JAVA-aplikáciu, ale zároveň ju pripraví o jednu z najcennejších vlastností – nezávislosť na platforme.

GCJ – kompilácia zdrojového kódu alebo bytecode do strojového kódu, teda natívne kompilovanej aplikácie. Tento postup nie je aplikovateľný na ľubovoľnú aplikáciu, pretože nie všetky triedy a balíky zo základnej výbavy sú kompilátorom GCJ podporované. Týmto postupom aplikácia opäť stratí svoju nezávislosť na platforme.

Pri spomenutých stratégiách sa zrýchľovanie aplikácie javí ako komplikovaná úloha, ktorej prínos je diskutabilný.

V nasledujúcej časti sa nachádza popis testu rýchlosti a jeho výsledky. Kedže veľmi častou matematickou úlohou v geodézii je inverzia matic, je tento test zameraný práve na výkon pri maticových operáciách.

3.1 Inverzia matice

Testovania sa zúčastnil algoritmus v JAVE SpeedTest_Ox.java, využívajúci balík JAMA: (<http://math.nist.gov/javanumerics/jama/>) a podobný algoritmus v jazyku C ++ speedTestOx.cpp, využívajúci knižnicu MATVEC, ktorej autorom je prof. Ing. Aleš Čepek, CSc., ka-

¹⁾ Message Digest algoritmus, bol vytvorený v roku 1991 s cieľom vytvárať digitálny podpis. Je to jednosmerná transformačná funkcia, ktorej výsledok je 16-bytová kontrolná suma zo vstupných dát. Malá zmena vo vstupných dátach má dramatický dopad na výsledok.

²⁾ Technológia zabezpečujúca súčasný chod viacerých vetiev algoritmu bez kolízií.

³⁾ Garbage Collector – technológia uvoľňovania pamäti po nepotrebných objektoch.

tetra mapování a kartografie FSv ČVUT v Praze. MATVEC je súčasťou balíka GAMA, (<http://www.gnu.org/software/gama/>) [13].

Algoritmy oboch balíkov neboli modifikované. Testy pod C++ boli spúšťané v dvoch variantoch. Raz bez optimalizovania a raz s využitím optimalizácie (O2) na úrovni kompilácie.

Porovnávanie obsahuje rôzne stratégie invertovania.

Balík JAMA využíva na inverziu štvorcovej matice metódu LU-dekompozície (LUD). Počet opakovaní vnútorného cyklu, ktorý pozostáva z jedného sčítania a jedného násobenia, je $\frac{2}{3}N^3$, kde N je rozmer matice. JAMA obsahuje aj metódu Choleského rozkladu (CHD), ktorá je aplikovateľná pri symetrickej pozitívne definitnej matici a jej počet vnútorných cyklov je $\frac{1}{3}N^3$. Teda Choleského rozklad možno použiť pri využití metódy najmenších štvorcov.

Obe metódy dokážu vyprodukovať riešenie – vektor odhadovaných parametrov – bez vyčíslenia inverznej matice.

V tomto prípade je inverzná matica potrebná na zostavenie kovariančnej matice parametrov a v praxi sa využíva na zostavenie štatistických vlastností výsledkov. Preto bola v tomto algoritme inverzná matica vyčíslená.

Gaussova-Jordanova eliminácia (GJE) sa v balíku JAMA nenachádza, ale kvôli korektnosti bola použitá trieda s touto metódou (<http://www.cs.umbc.edu/~squire/download/Matrix.java>).

V knižnici MATVEC je inverzia postavená na Gaussovej-Jordanovej eliminácii. Táto metóda opakuje vnútorný cyklus, ktorý obsahuje jedno odčítanie a jedno násobenie, N^3 krát.

Obe metódy využívajú úplnú pivotáž. Teoreticky, ak by bola JAVA na úrovni elementárnych algebraických operácií rovnako rýchla ako C++, tak balík JAMA s metódou Choleského rozkladu by mal dosiahnuť výrazne lepší čas oproti knižnici MATVEC s metódou GJE. Prakticky je známy fakt, že JAVA nemá polia a nahrádza ich objektami. Oproti jazyku C++ je rozdiel v tom, že vytvore-

nie inštalácie triedy je výrazne časovo náročnejšie v porovnaní s vytvorením poľa, aké je k dispozícii v C++. Táto indispozícia JAVY by sa mala postupne eliminovať s narástajúcim rozmerom inverznej matice, kde počet polí, potrebných na uskladnenie matice, bude veľmi malý v porovnaní s počtom následných operácií okolo inverzie.

Test obsahuje riešenie metódou najmenších štvorcov, teda zostavenie matice plánu, vektora absolútnych hodnôt, inverziu matice koeficientov normálnych rovníc a vynásobenia vektora ($A^T L$) invertovanou maticou.

V každom z testov bolo vykonaných 40 rokov kvôli eliminovaniu prípadných výkonnostných výkyvov. Pozorovaná veličina bol čas v [ms], spotrebovaný na výpočet. Tento čas neobsahuje počiatočnú inicializáciu matíc.

Pre overenie testovaných algoritmov z hľadiska správania sa pri zle podmienených maticiach bola využitá Hilbertová matica. Výsledky sú na obr. 1.

Základom pre porovnávanie sú výsledky z konfigurácie, ktorá obsahuje 512 MB operačnej pamäte, 1 GHz procesor a operačný systém linux (značenie lin). Ďalšia séria výsledkov vznikla na totožnej konfigurácii, ale pod operačným systémom windows (značenie win) a na výpočet tretej časti bol použitý opäť operačný systém linux, ale s výrazne menšou operačnou pamäťou (128 MB) a 800 MHz procesorom (značenie lin2).

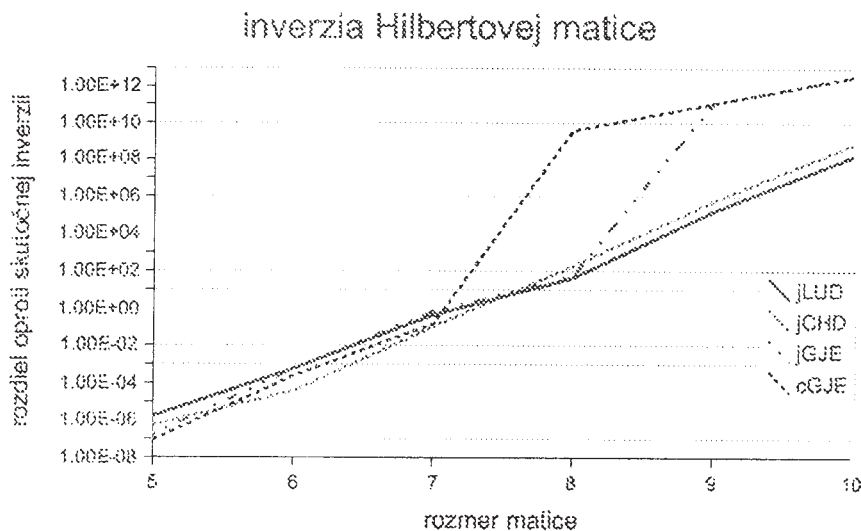
Výsledok by mal ukázať závislosť rýchlosti výpočtu od operačného systému a od množstva operačnej pamäte.

1. Invertovaná matica má rozmer 6×6 , vznikla z riešenia vyrovnania sprostredkujúcich meraní. Matica plánu mala rozmer 10×6 . Jeden testovací krok obsahoval 20000 opakovaní tohoto výpočtu.

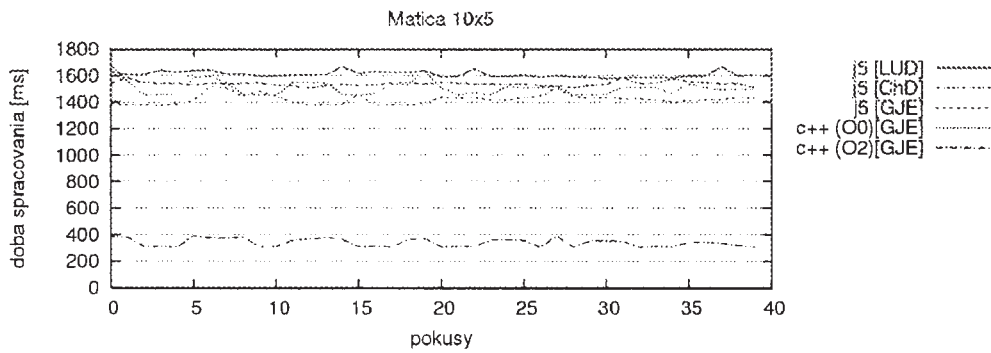
Výsledok (obr. 2, 3 a 4):

JAMA takmer nezaostáva za C++, dokonca pod OS Windows bola o 20 % rýchlejšia oproti MATVEC. Medzi jednotlivými konfiguráciami sú zachované časové proporcie. Po použití optimalizácie (-O2) pri kompilovaní v C++ sa výpočtový čas MATVEC znížil na 20 %. Teda JAMA prehráva približne 1:5.

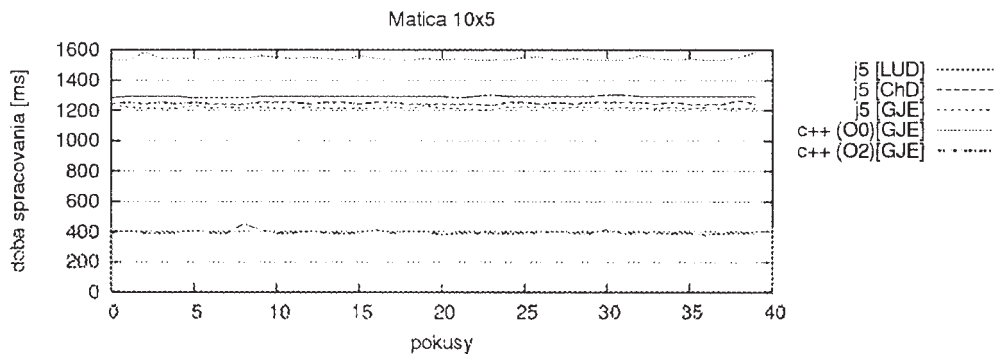
2. Nasadená matica má rozmer 35×35 . Odpovedajúca matica plánu mala rozmer 50×35 . Výpočet bol v jednom testovacom kroku zopakovaný 2000 krát.



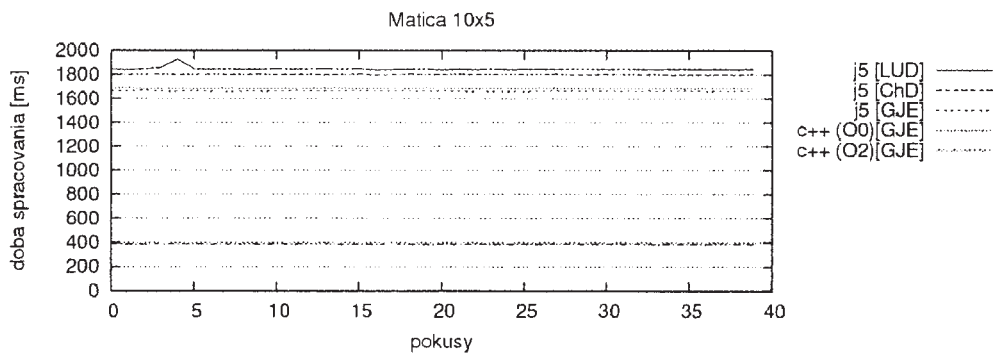
Obr. 1 Chyba pri inverzii Hilbertovej matice



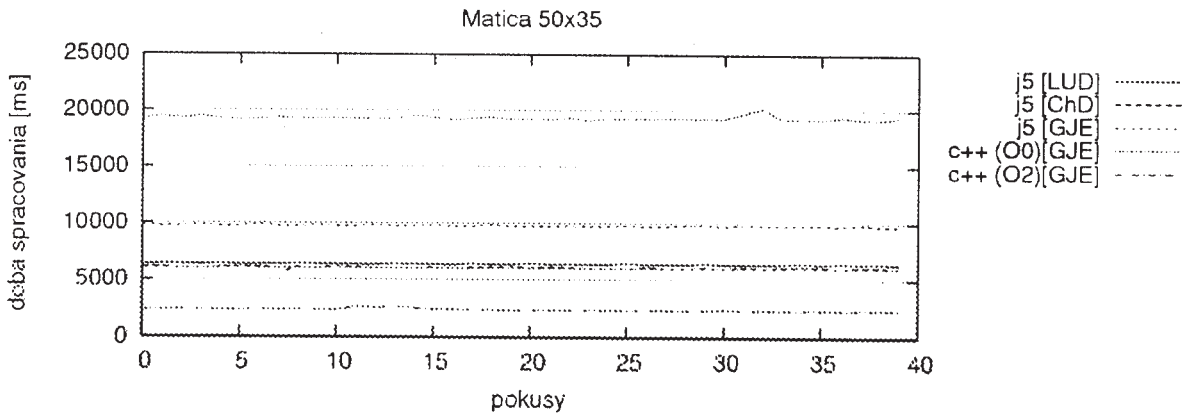
Obr. 2 Inverzia matic, 1. kolo, linux 1



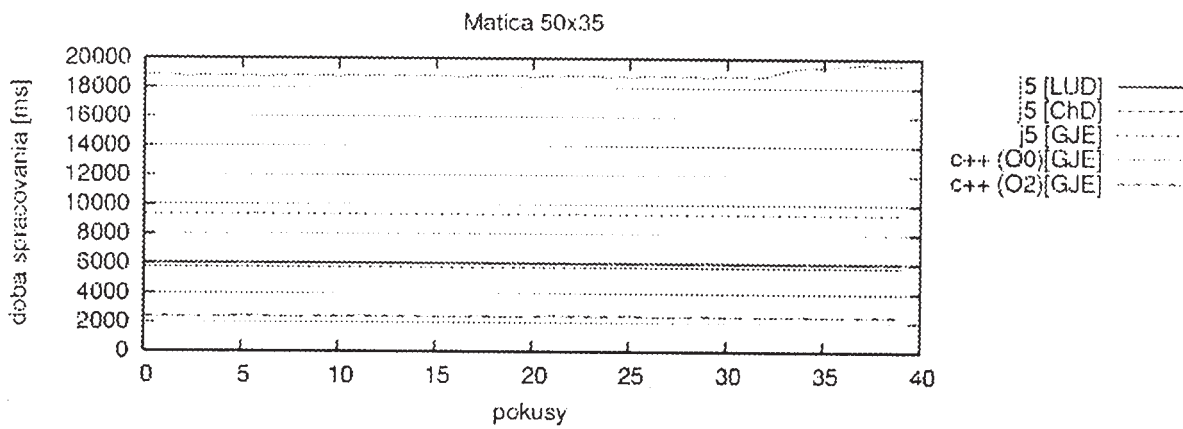
Obr. 3 Inverzia matic, 1. kolo, windows



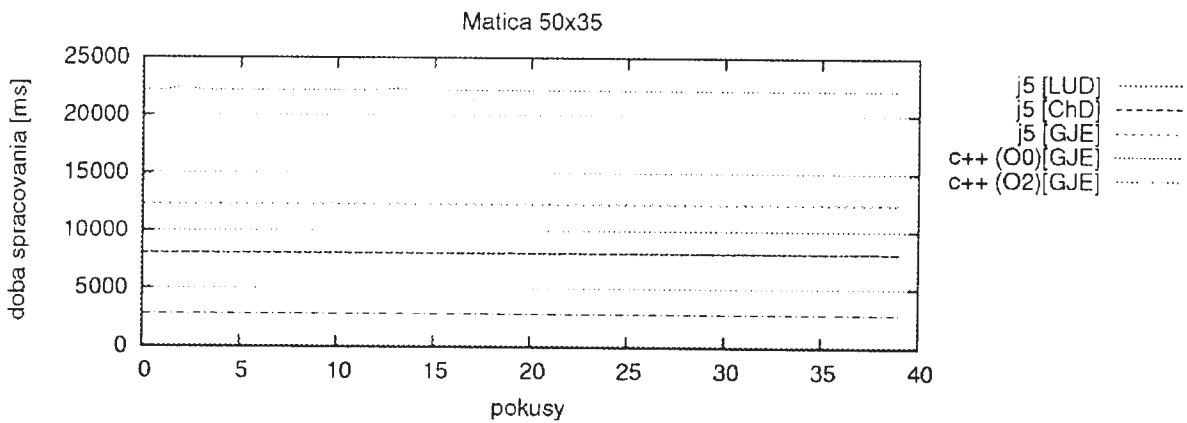
Obr. 4 Inverzia matic, 1. kolo, linux 2



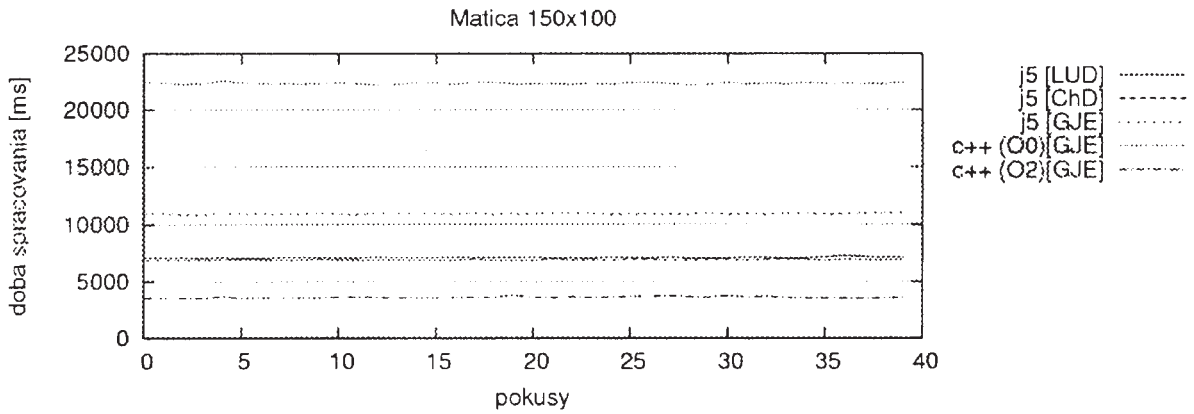
Obr. 5 Inverzia matic, 2. kolo, linux 1



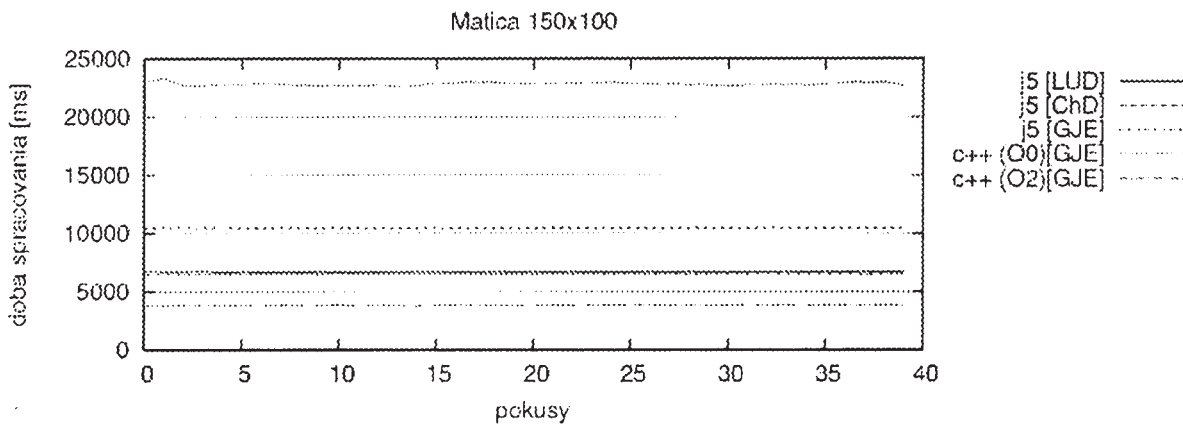
Obr. 6 Inverzia matic, 2. kolo, windows



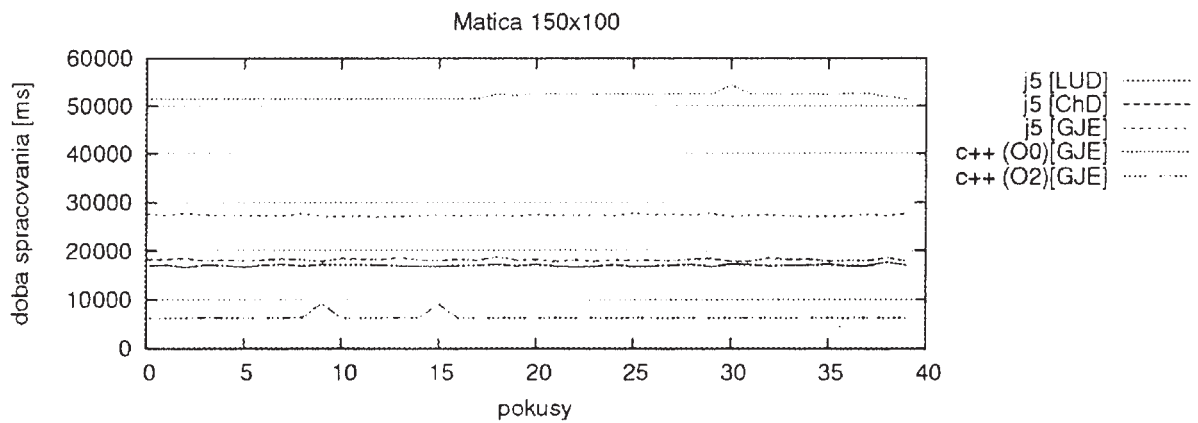
Obr. 7 Inverzia matic, 2. kolo, linux 2



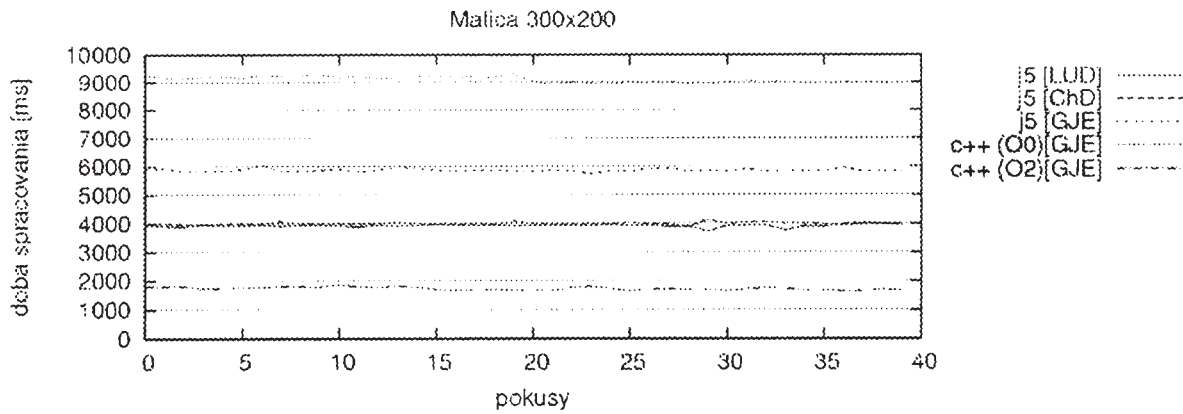
Obr. 8 Inverzia matíc, 3. kolo, linux 1



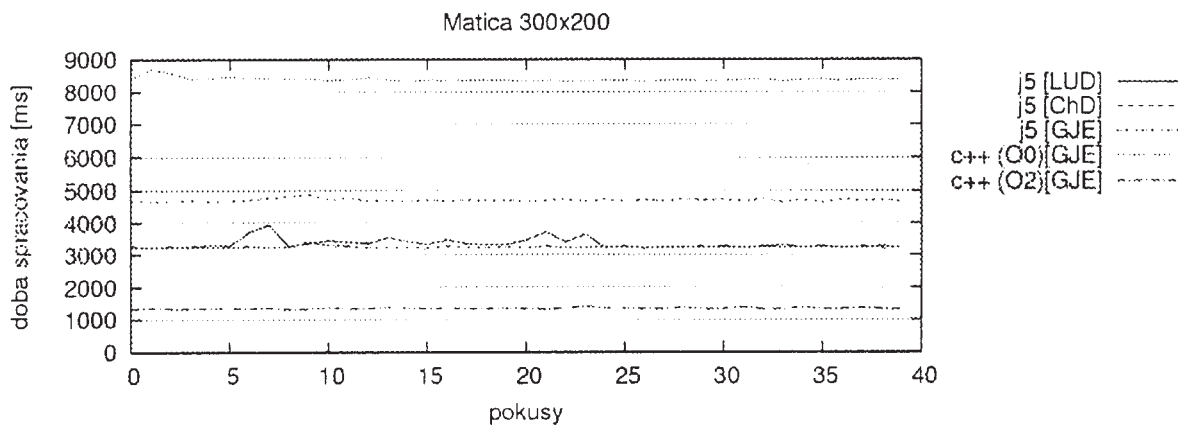
Obr. 9 Inverzia matíc, 3. kolo, windows



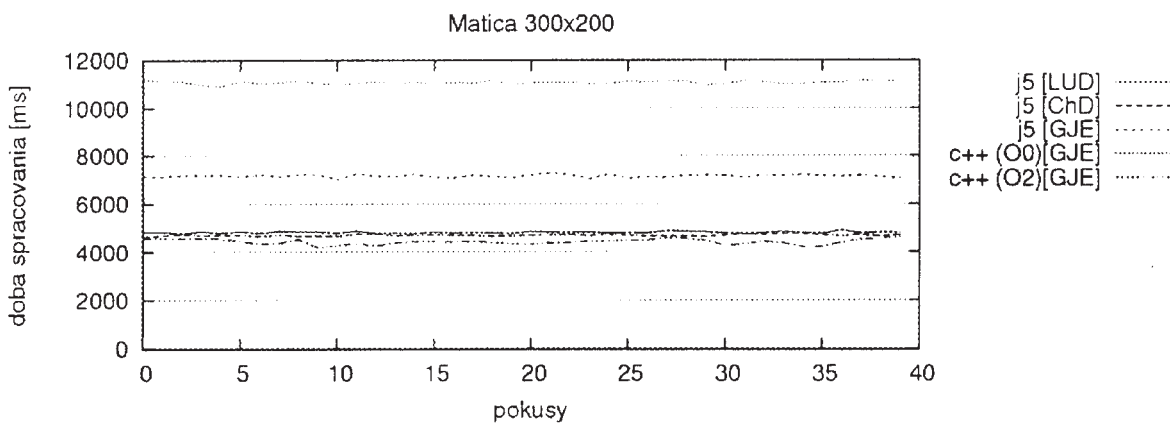
Obr. 10 Inverzia matíc, 3. kolo, linux 2



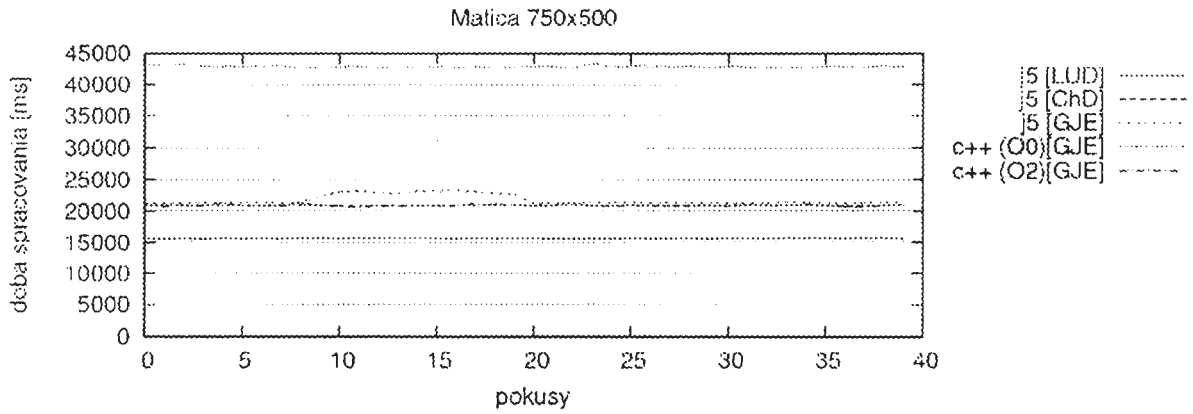
Obr. 11 Inverzia matíc, 4. kolo, linux 1



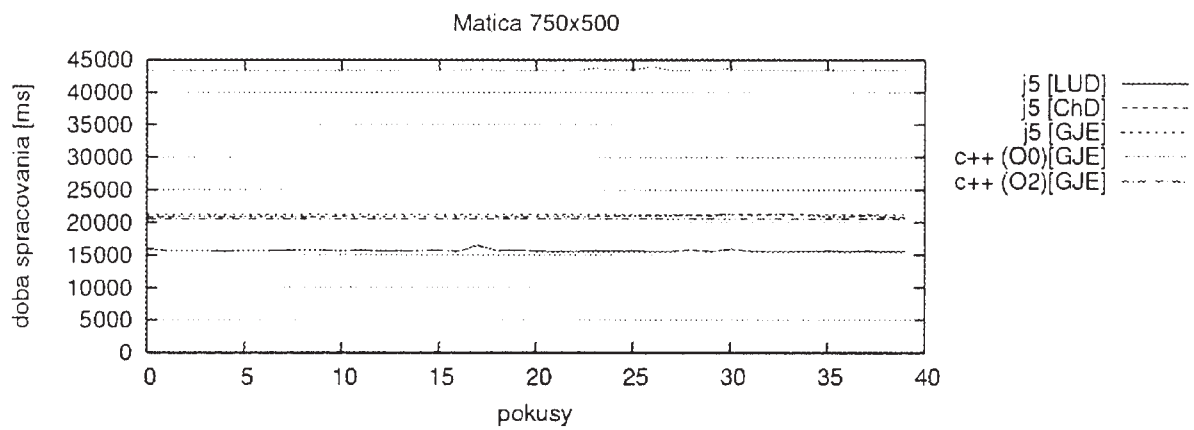
Obr. 12 Inverzia matíc, 4. kolo, windows



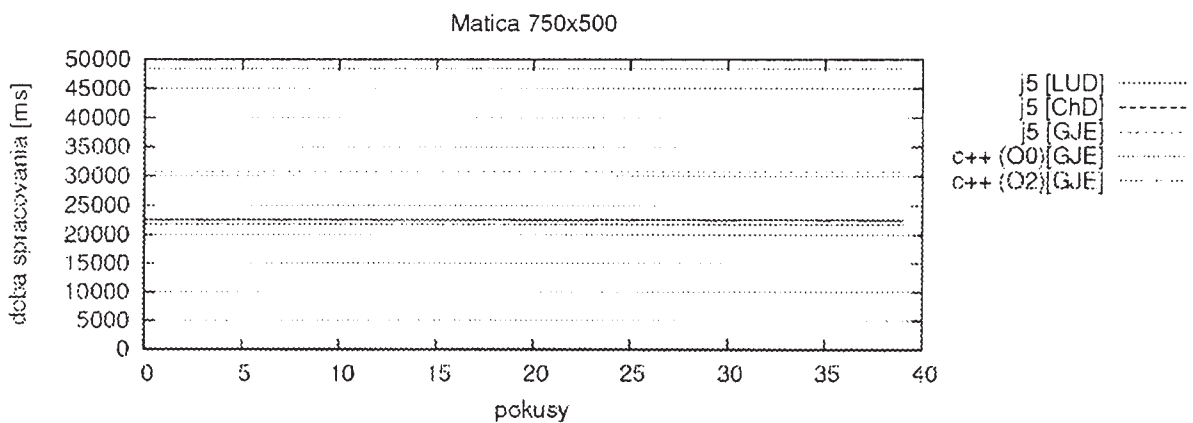
Obr. 13 Inverzia matíc, 4. kolo, linux 2



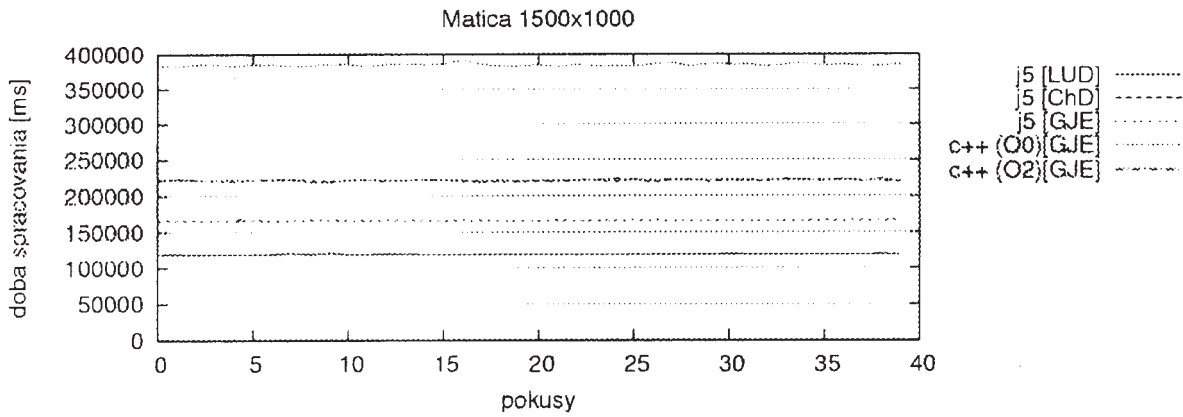
Obr. 14 Inverzia matíc, 5. kolo, linux 1



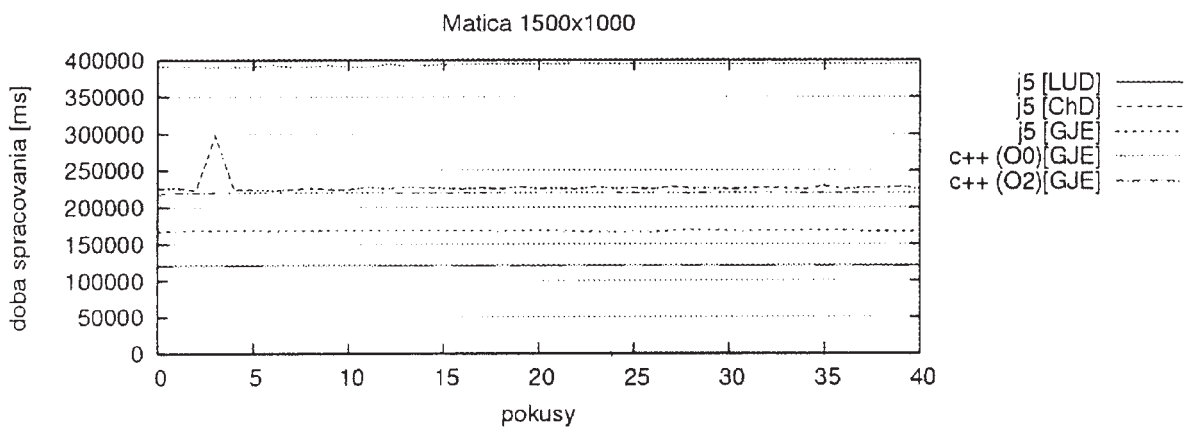
Obr. 15 Inverzia matíc, 5. kolo, windows



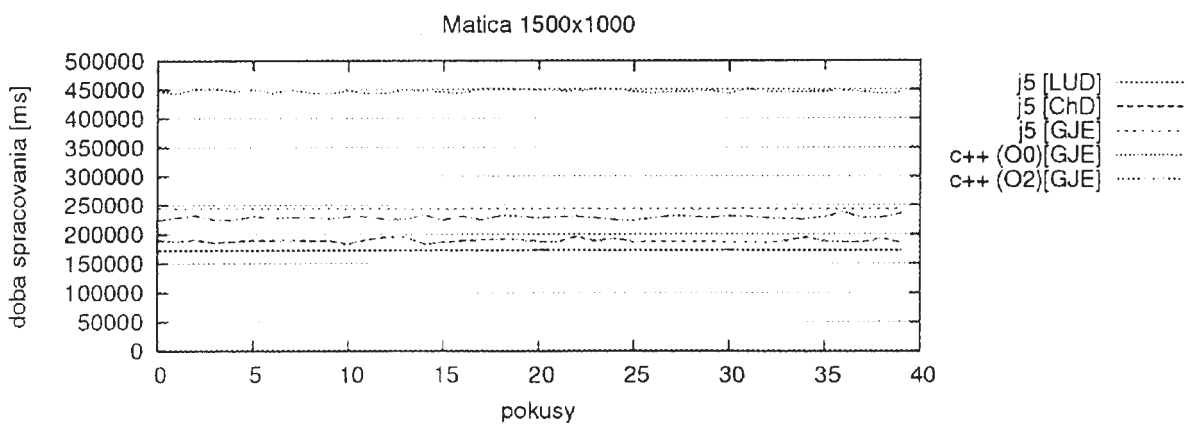
Obr. 16 Inverzia matíc, 5. kolo, linux 2



Obr. 17 Inverzia matíc, 6. kolo, linux 1



Obr. 18 Inverzia matíc, 6. kolo, windows



Obr. 19 Inverzia matíc, 6. kolo, linux 2

Tab. 1 Výsledky testov

N	OS	c++	c++[O2]	j LUD	j ChD	j GJE
6	lin	100.0%	22.6%	106.9%	102.2%	94.4%
	win	100.0%	25.3%	83.7%	80.8%	78.8%
	lin2	100.0%	23.1%	109.6%	106.8%	98.7%
35	lin	100.0%	12.2%	33.0%	31.3%	50.6%
	win	100.0%	12.8%	31.9%	30.3%	49.4%
	lin2	100.0%	12.5%	36.2%	36.0%	55.5%
100	lin	100.0%	16.0%	31.9%	31.0%	48.7%
	win	100.0%	16.7%	29.2%	28.5%	45.8%
	lin2	100.0%	12.3%	32.6%	34.8%	52.6%
200	lin	100.0%	18.9%	43.1%	44.0%	64.4%
	win	100.0%	16.0%	39.9%	38.6%	56.2%
	lin2	100.0%	40.0%	43.5%	42.4%	64.6%
500	lin	100.0%	48.8%	36.4%	48.8%	50.7%
	win	100.0%	47.4%	36.0%	48.3%	48.8%
	lin2	100.0%	45.0%	46.6%	44.9%	63.5%
1000	lin	100.0%	57.7%	30.9%	57.6%	43.2%
	win	100.0%	55.8%	30.7%	57.8%	42.6%
	lin2	100.0%	51.0%	38.5%	42.2%	54.4%

Výsledok (obr. 5, 6 a 7):

JAMA dosiahla v porovnaní s neoptimalizovaným C++ čas na úrovni 32 % (pod OS-Windows 30 %). Gaussova-Jordanova eliminácia začína výrazne zaostávať. Časové proporcie medzi konfiguráciami zostali opäť zachované. Po optimalizácii sa výpočtový čas MATVEC znížil na 12 %. JAMA prehráva 1:3, čo však na druhej strane znamená výrazne zlepšenie oproti prvému kolu.

3. V tomto kole testov bolo pracované s maticou o rozmere 100×100. Výpočet bol v jednom testovacom kroku zopakovaný 100 krát.

Výsledok (obr. 8, 9 a 10):

JAMA dosiahla v porovnaní s neoptimalizovaným C++ čas na úrovni 30 % (pod OS-Windows 30,5 %). Časové proporcie medzi konfiguráciami stále vykazujú výraznú podobnosť.

Po optimalizácii sa výpočtový čas MATVEC znížil na 16 %. JAVA teda prehráva približne 1:2.

4. Matica má rozmer 200×200. Matica plánu odpovedajúceho vyrovnania sprostredkujúcich meraní mala rozmer 300×200. Výpočet bol v jednom testovacom kroku zopakovaný 4 krát.

Výsledok (obr. 11, 12 a 13):

JAMA dosiahla v porovnaní s neoptimalizovaným C++ čas na úrovni 44 % (pod OS-Windows 40 %).

Po optimalizácii sa výpočtový čas MATVEC znížil na 18 %. JAVA teda prehráva približne 2:5.

Pod konfiguráciou lin2 je optimalizované C++ na úrovni metód CHD a LUD v JAVE.

5. Invertovaná matica má rozmer 500×500. Matica plánu odpovedajúceho vyrovnania sprostredkujúcich meraní mala rozmer 750×500. Výpočet bol v jednom testovacom kroku zopakovaný 1 krát.

Výsledok (obr. 14, 15 a 16):

Najlepší výsledok dosiahla metóda LUD z balíka JAMA, v porovnaní s neoptimalizovaným C++ je to čas na úrovni 36 %. Metódy GJE a CHD dosiahli približne rovnaký čas u oboch balíkov JAMA a MATVEC – 48 % až 50 %.

JAVA teda nevýrazne vyhráva približne 3:4.

Pod konfiguráciou lin2 sú výsledky podobné predošlému kolu – LUD a CHD v JAME sú spolu s optimalizovaným C++ najrýchlejšie (45 %) a GJE – 63 % oproti neoptimalizovanému C++.

6. Invertovaná matica má rozmer 1000×1000. Matica plánu odpovedajúceho vyrovnania sprostredkujúcich meraní mala rozmer 1500×1000. Výpočet bol v jednom testovacom kroku zopakovaný 1 krát.

Výsledok (obr. 17, 18 a 19):

Najlepší výsledok dosiahla opäť metóda LUD z balíka JAMA, v porovnaní s neoptimalizovaným C++ je to čas na úrovni 30 %. Metóda GJE – 43 % a metóda CHD dosiahla spolu s optimalizovaným C++ približne rovnaký čas – 55 % až 57 %.

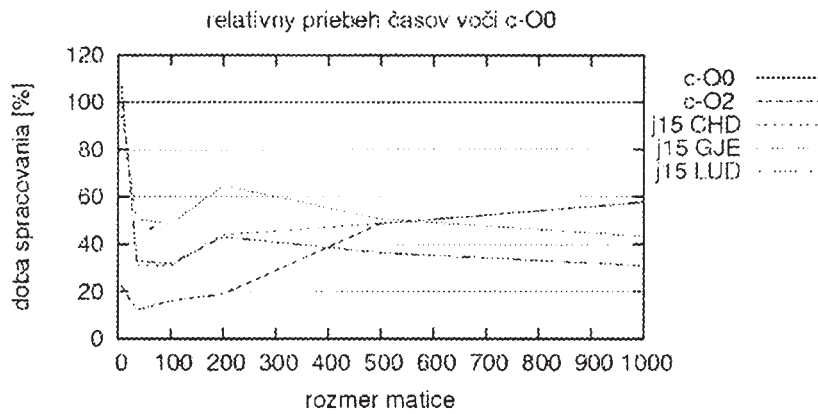
JAVA teda vyhráva približne 1:2.

Pod konfiguráciou lin2 je takisto vo vedení metóda LUD (38 %), za ňou nepatrne zaostáva CHD a metóda GJE v JAME a v C++ sa dostávajú na podobnú úroveň (51 % až 54 %).

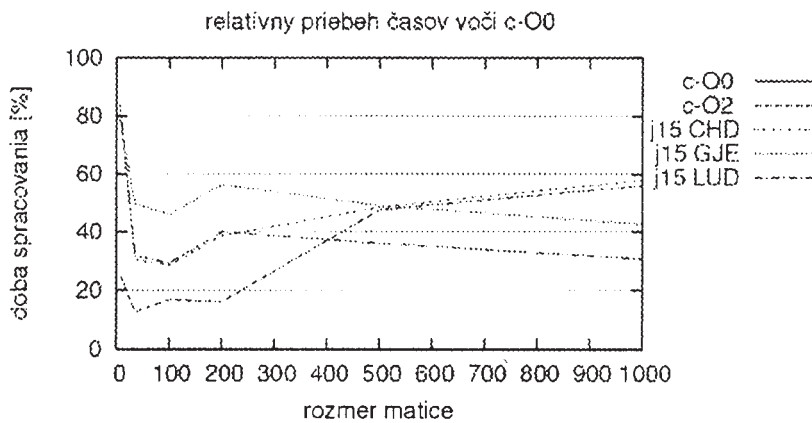
Doplnkom k rýchlostným testom bol test stability. Tento test pozostával z výpočtu inverzie Hilbertovej matice (HM). HM je zle podmienená matica (1), ku ktorej je známa presná inverzia (2). Indexy v týchto vzorcoch musia začínať od 1 a n je rozmer zvolenej HM:

$$H_{ij} = \frac{1}{i+j-1}, \quad (1)$$

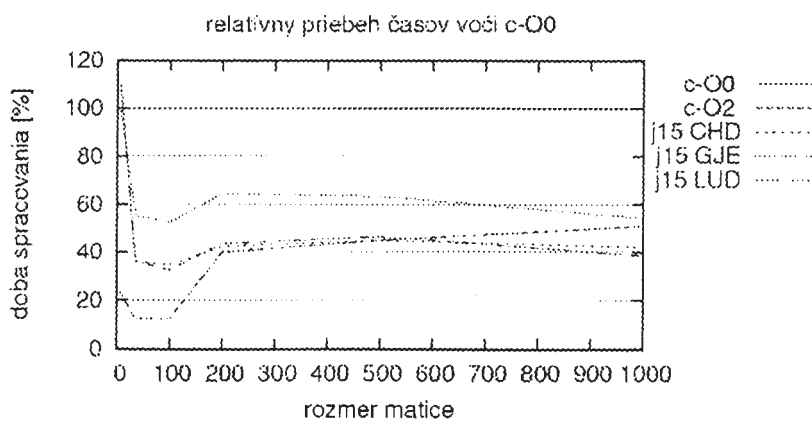
$$H_{ij}^{-1} = (-1)^{i+j} (i+j-1) \binom{n+i-1}{n-j} \binom{n+j-1}{n-i} \binom{i+j-2}{i-1}^2. \quad (2)$$



Obr. 20 Priebeh relatívnych časov spracovania– linux



Obr. 21 Priebeh relatívnych časov spracovania– windows



Obr. 22 Priebeh relatívnych časov spracovania– linux 2

3.2 Výsledky testu inverzie

V tab. 1 je N – rozmer matice, OS – operačný systém. Ostatné stĺpce obsahujú časy výpočtu: C++ – jazyk C++ bez optimalizácie, C++ [O2] – jazyk C++ s optimalizáciou O2, j | LUD – jazyk JAVA, metóda LU-rozkladu, j | ChD – jazyk JAVA, metóda Choleského rozkladu, j | GJE – jazyk JAVA, metóda Gaussovej-Jordanovej eliminácie (táto metóda je implementovaná v knižnici MATVEC na strane C++).

Obsah tab. 1 je graficky znázornený na obr. 20, 21 a 22.

Za normu bol vždy zvolený čas spracovania programu v C++ bez optimalizácie. Tento program dosahoval zpravidla najdlhšie časy a jeho výkon som pokladal za najstabilnejší.

Prínos optimalizácie v prípade jazyka C++ je zreteľný. So zvyšovaním rozmeru matice sa účinnosť optimalizácie O2 zlepšuje.

Na strane JAVY sa tiež objavilo zlepšenie výkonu pri rastúcom rozmere matice. Rozmanitosť rozmerov testovaných matíc je síce nízka, ale napriek tomu je vidieť, že do rozmeru 100×100 JAVA pomaly dobieha optimalizované C++, teda s rastúcim rozmerom sa pomer času JAVY oproti neoptimalizovanému C++ zlepšuje rýchlejšie, ako u O2-optimalizovaného C++. Tento trend sa ale neudrží a pri rozmere 200×200 si optimalizované C++ približne zachováva svoj časový pomer k neoptimalizovanému C++, ale JAVA začína spomaľovať.

Radikálna zmena trendov sa prejaví pri rozmere 500 (pri konfigurácii lin2 už pri 200), kedy optimalizované C++ spomalí natoľko, že je rovnako rýchle, alebo pomalšie, ako metóda LUD v JAME. Príčinou by mohlo byť vyčerpanie operačnej pamäte a nástup virtuálnej pamäte, ktorá je simulovaná pevným diskom.

Je zaujímavé, čo sa deje pri metóde Gauss-Jordanovej eliminácie (GJE) v JAVE. Pri najmenšom rozmere (6×6) dosahovala táto metóda v JAVE nepatrne lepší čas oproti ostatným metódam. Pri náraste rozmeru sa čas spracovania metódou GJE začal prudko zhoršovať a dosahuje približne 1,5 násobok ostatných metód v JAVE. Je pravdepodobné, že tak výrazný pokles výkonu má pôvod v konštrukcii GJE algoritmu. Cieľom týchto testov ale nie je porovnať najrýchlejšie existujúce spôsoby inverzie, ale bežne implementované algoritmy, ktoré síce nie sú dokonale optimalizované, ale zato sú prehľadnejšie a poskytujú širšie možnosti v udržiavaní a rozvoji.

Tak trochu sklamaním je metóda CHD, od ktorej som očakával vyšší výkon oproti LUD. Kratší čas spracovania dosahovala CHD len pri malých maticiach, ale rozdiel v porovnaní s časom LUD nebol významný.

Je badateľné, že nominálne časy spracovania sú pod operačným systémom windows väčšinou nepatrne lepšie ako pod systémom linux. Tento jav má viacero možných príčin. Za najpravdepodobnejšie pokladám množstvo bežiacich programov na pozadí a vyťaženosť operačnej pamäte. V oblasti testovania rýchlosti algoritmov sa obecné za solídny dôkaz pokladá až rádoový rozdiel vo výslednom čase. Menšie rozdiely sú často pokladané za šum alebo neobjektívnosť. Ideálny test by mal zasahovať čo najširšiu oblasť, teda samotné spracovanie matíc je iba zlomkom zpomedzi rutín, ktoré by bolo potrebné otestovať a dokázať ich podiel na celkovom čase spracovania konkrétnej aplikácie.

Z hľadiska numerickej stability vyzerajú podľa obr. 1 všetky použité metódy inverzie podobne. Najhoršie dopadla metóda Gaussovej-Jordanovej eliminácie, ale jej zhoršenie oproti ostatným metódam sa prejavilo pri rozmere 8×8, kde je podmienenosť Hilbertovej matice rádoovo $1e10$. Zaujímavé je, že táto metóda pod JAVOU zaznamenala podobné zhor-

šenie až pri rozmere 9×9 (podmienenosť = $5e11$). Z praktického hľadiska je tento skok ďaleko za bežne vyskytujúcou sa podmienenosťou matice v geodézii.

Pri porovnaní riešení matice o rozmere 200×200 sa ukazuje, že maximálny dosiahnutý rozdiel v absolútnej hodnote bol medzi výsledkom programu v C++ a výsledkom programu v JAVE (konkrétne s metódou LU-rozkladu). Dosiahol hodnotu 5e-7. Podmienenosť matice koeficientov normálnych rovníc z náhodne vygenerovaných hodnôt bola 102.1.

Pokiaľ je hlavným cieľom spracovania rýchlosť a významná časť spracovania stojí práve na výpočtoch, je použitie C++ s optimalizáciou výhodné. Teda napríklad pre periodicky opakované sekvencie výpočtov pri veľkom objeme dát a malých maticiach, s akými sa možno stretnúť napríklad pri spracovaní observácií z permanentných GPS staníc, by aplikácia v JAVE nebola najlepším riešením. Tu je ideálna konzolová aplikácia, bežiacia na pozadí a s minimálnou možnosťou interakcie užívateľa.

V prípade interaktívnej aplikácie s nutnosťou grafického rozhrania, kde samotné výpočty predstavujú iba zlomok práce s aplikáciou, je JAVA zaujímavá kvôli jednoduchosti implementácie sieťových zdrojov, databázového spojenia a nezávislosti na platforme.

Jednoznačnosť výhod optimalizovaného C++ je narušená pri väčších maticiach. Tu je namieste prehodnotiť rozšírenie operačnej pamäte, ktorým by sa mala rozšíriť oblasť, v ktorej je optimalizované C++ pri invertovaní výrazne rýchlejšie oproti JAVE.

Nie som presvedčený, že tento test je jasným dôkazom zaradenia JAVY do kategórie jednoduchých aplikácií či grafických rozhraní pre výkonnejšie aplikácie. Skôr bolo demonštrované, akú významnú rolu hrá optimalizácia, výber algoritmov a veľkosť operačnej pamäte v rýchlosti výslednej aplikácie.

Za najrýchlejšie výpočtové programy su považované fortranovské rutiny, pričom možnosť hierarchickej štruktúry algoritmov, ktorú ponúkajú napríklad objektovo orientované jazyky, je vo FORTRAN minimálna. FORTRAN využíva metódu návestia a skoku, od ktorej sa v modernejších jazykoch upustilo a je považovaná za zdroj veľmi nepríjemných logických chýb.

Tu je na každom programátorovi, či sa rozhodne pre maximálnu rýchlosť na úkor udržiavateľnosti programových knižníc (FORTRAN) alebo siahne po nepatrne menej výkonnom jazyku s podstatne širšími možnosťami (C++), či zvolí rozvíjajúci sa jazyk s nižším výkonom, ale s obrovskou výhodou v podobe nezávislosti na platforme.

V súčasnosti existuje názor, že najbúrlivejšia fáza vývoja jazyka JAVA už skončila a ďalší vývoj už neprinesie výrazne zmeny v tak krátkom časovom horizonte. Napriek tomu verím, že využitie JAVY v geodetických aplikáciách je dnes perspektívne.

Na záver sa chcem poďakovať prof. A. Čepkovi za pomoc a rady pri testovaní jeho knižnice MATVEC.

4. PPP

Precise Point Positioning alebo určovanie presnej absolútnej polohy bodu spracovaním fázových a kódových observácií GPS súčasne. Nasledujúca kapitola je malou demonštráciou využitia JAVY v geodetickej aplikácii. Metódou PPP sa zaoberá moja dizertačná práca a myšlienka využiť JAVU sa vyskytla náhodou. Systém, popísaný v nasledujúcej časti, dostal pracovný názov **JGPS**.

4.1 Vývoj riešenia

Mojím prvým cieľom bolo vytvorenie JAVA-applet, ktorý mal za úlohu spočítať SPP (single point position, výpočet absolútnej polohy bodu na základe kódového riešenia). Tento applet sa pri otvorení webovej stránky načítal a spustil na strane užívateľa. Následne užívateľ vybral:

- observačný súbor, ktorý sa nachádzal na serveri,
- stratégiu výpočtu dráh družíc (vysielané efemeridy alebo presné dráhy),
- súbor korekcií, ktoré využije nasledujúce spracovanie dát, e. g. relativistická korekcia, korekcia z rotácie systému WGS84 (World Geodetic System 1984) počas prenosu signálu a iné,
- model atmosféry, ktorý by mal spresniť výslednú polohu, pokiaľ sa lokálna atmosféra dostatočne blíži k modelu štandardnej atmosféry.

Po spustení výpočtu mohol užívateľ sledovať priebeh spracovania a po skončení sa objavil záznam z výpočtu a bolo otvorené nové okno s grafmi o priebehu súradníc a ich chýb počas spracovávanej kampane. Užívateľ mohol buď spracovať súčasný observačný súbor s iným súborom korekcií a modelom atmosféry [5, 10, 11], prípadne s inými dráhami, alebo zvolil iný observačný súbor, čo znamenalo zahodenie predchádzajúcich výsledkov. Grafy bolo možné poslať na pripojený tlačiareň a záznam z výpočtu sa dal skopírovať manuálne.

Tento applet bol zároveň spustiteľný ako lokálna aplikácia. JAVA obsahuje v základnej výbave aj triedy pre prácu so sieťovými zdrojmi a teda bolo napríklad možné pomerne jednoducho implementovať čítanie dát z ftp servera s vyrovnávacím zásobníkom. Aplikácia, ktorá nemá obmedzenia appletu, mohla automaticky na začiatku výpočtu načítať potrebné efemeridy priamo zo servera IGS.

Tu sa vyskytli prvé vážne komplikácie – efemeridy boli komprimované proprietárnym algoritmom (LZW kompresiou), ktorého patentová ochrana prestala platiť v roku 2004. Základná výbava JAVY neobsahuje triedy pre prácu s týmto typom dát.

Ďalší problém súvisiel s bezpečnosťou. Samotný applet musí byť podpísaný, aby umožňoval načítanie súborov s efemeridami zo servera IGS.

Ukončenie vývoja appletu bolo zavŕšené problémom s nedostatkom operačnej pamäti, ktorej množstvo je pridelené na strane užívateľa a nedá sa programovo ovplyvniť.

Ďalší vývoj sa týkal rutín vo forme aplikácie, ktoré by sa dali v prípade potreby relatívne jednoducho napojiť na aplikčný server.

Nasledovala ďalšia zmena v podobe preorientovania sa na PPP spracovanie. Táto metóda využíva väčšinu funkcií predchádzajúcej metódy (SPP) a navyše potrebuje spracovanie fázových meraní, odhad ambiguit, modelovanie slapového javu, excentricity fáxového centra a ďalšie spresňujúce modely.

4.2 Popis spracovania metódou PPP

Riešenie PPP je založené na spoločnom spracovaní kódových a fázových observácií GPS [7]. Pre každú epochu musia byť dostupné presné korekcie družicových hodín. Na dosiahnutie presnosti v polohe lepšej ako 0,1 m je zpravidla potrebné spracovať observácie z celého dňa. Najjednoduchšia stratégia spočíva v spracovaní len tých epoch, kde sú dostupné presné polohy a korekcie hodín satelitov zo súboru presných dráh. Presných dráh máme v súčasnosti viacero druhov, od ultra-rapid v reálnom čase až po final, ktoré sú dostupné

s oneskorením 13 dní a dosahujú presnosť dráhy 0,05 m a 0,5 ns hodín satelitov.

Z hľadiska objemu dát môže vzniknúť komplikácia s priveľkými maticami. V takom prípade je potrebné buď zredukovať počet neznámych, alebo observácie spracovávať sekvenčne – napríklad pomocou Kalmanovho filtra.

Zameral som sa na krátke observácie (do 48 h) a z tohoto dôvodu predpokladám, že rozmer matic pri výpočte nepreplní operačnú pamäť a teda hlavný výpočet prebehne v „jednom“ kroku.

Observačné rovnice majú tvar (3) a (4):

$$R_k^i(t) = \rho_k^i(t) - c \delta t^i(t) + c \delta t_k(t) + ZHD(t) m_{f_H}(z^i(t)) + ZWD(t) m_{f_w}(z^i(t)), \quad (3)$$

$$\Phi_k^i(t) \lambda = \rho_k^i(t) - c \delta t^i(t) + c \delta t_k(t) + ZHD(t) m_{f_H}(z^i(t)) + ZWD(t) m_{f_w}(z^i(t)) + n_k^i(t) \lambda, \quad (4)$$

kde je

$R_k^i(t)$ pseudovzdialenosť medzi satelitom (i) a prijímačom (k) v čase t ,

$\rho_k^i(t)$ geometrická vzdialenosť medzi satelitom (i) a prijímačom (k) v čase t ,

c rýchlosť svetla,

$\delta t^i(t)$ korekcia hodín satelitu (i) a v čase t ,

$\delta t_k(t)$ korekcia hodín prijímača (k) v čase t ,

$ZHD(t)$ suchá zložka ZTD (celkové oneskorenie signálu v smere zenitu) v čase t ,

$ZWD(t)$ mokrá zložka ZTD v čase t ,

$m_{f_H}(z^i(t))$ mapovacia funkcia suchej zložky ZTD ,

$m_{f_w}(z^i(t))$ mapovacia funkcia mokrej zložky ZTD ,

$z^i(t)$ zenitový uhol satelitu (i) a v čase t ,

$\Phi_k^i(t)$ observácia fázy satelitu (i) na prijímači (k) v čase t ,

λ vlnová dĺžka L1 alebo L2,

$n_k^i(t)$ ambiguita satelitu (i) pre prijímač (k) v čase t .

Ak sú k dispozícii observácie z oboch vlnových dĺžok L1 a L2, tak v jednej epoche budú predstavovať observácie z jedného satelitu práve štyri observačné rovnice. Vtedy je možné použiť lineárnu kombináciu L3, ktorá minimalizuje vplyv ionosféry na oneskorenie signálu.

Odhad parametrov troposféry je možné zrealizovať nasledujúco:

Predpokladáme, že mapovacie funkcie dostatočne presne vystihujú lokálnu troposféru (napríklad Niellove mapovacie funkcie) a že suchú zložku stačí modelovať pomocou modelu troposféry, ktorý využíva štandardnú atmosféru. Potom možno suchú zložku považovať za približnú hodnotu celkového oneskorenia signálu pod daným zenitovým uhlom a odhadujeme už len mokrú zložku. V derivácii mokrej zložky bude vystupovať iba mokrá mapovacia funkcia a zenitový uhol pre daný satelit a čas.

Ambiguita predstavuje najväčší problém. Jej hodnota je v princípe násobok celých vln signálu, teda celé číslo. Síce existuje stratégia na získanie celočíselnej hodnoty ambiguity, ale tento prístup je komplikovanejší a má využitie až pri dvojitom diferencovaní observáciách, preto budeme zaoberať sa s ambiguitou ako s reálnym číslom. Druhou komplikáciou je určovanie epochy, ktorou počínajúc je potrebné zaviesť novú ambiguitu pre konkrétny satelit. Príznak výskytu fázového skoku priamo v observačnom súbore (LLI) je často spoľahlivý a ak je k dispozícii, máme riešenie, ktoré zlyháva len pri problematických kampaniach. Pokiaľ tento príznak v observačnom súbore nie je, možno využiť lineárnu kombináciu L4 a z jej priebehu predpokladať výskyt nových ambiguit. Po-

kiaľ aj tento postup zlyhá, treba niektoré observácie či kompletne satelity zo spracovania vypustiť. Nie vždy sa podarí ambiguity vyriešiť a neexistuje univerzálne účinný postup.

Ostatné korekcie a modely, ktoré v observačných rovniciach nie je vidieť, budú aplikované buď priamo do observácií, alebo do polôh satelitov. V nasledujúcom zozname sú korekcie a ich vplyv na polohu určovaného bodu:

- ionosféra – takmer eliminovaná pomocou L3 (v extrémnych prípadoch do 100 m) [5, 2],
- rotácia systému WGS84 (30 m) [7],
- relativistický efekt, vyplývajúci z rozdielnych tiažových potenciálov satelitu a prijímača (10 m) [1],
- troposféra (do 3 m v smere zenitu, je silne korelovaná s výškou bodu) [3, 4, 6, 8, 9],
- excentricita fázového centra satelitu – anténa na starších satelitoch nie je v ťažisku (1 m) [7],
- vplyv rotácie satelitu na fázový posun – phase wind-up (0,2 m) [7],
- slapy oceánov vplyvajú na observácie v závislosti od vzdialenosti najbližšieho oceánu, pre stred Európy dosahuje tento vplyv rádovo 1 mm a menej [7],
- slapy pevnej Zeme, atmosféry a tektonické pohyby predstavujú korekcie rádovo v desatinách mm a menej [7].

Stratégia spracovania PPP by teda mohla vyzeráť takto:

1. vyhľadanie potrebných súborov presných dráh,
2. prvé parsovanie súboru observácií (detekcia ambiguít na základe príznaku LLI alebo L4 kombinácie a vylučovanie epoch podľa dostupnosti hodín satelitu v súbore dráh),
3. druhé parsovanie súboru observácií, naplnenie matice plánu a vektora absolútnych hodnôt pre odhad neznámych parametrov pomocou metódy najmenších štvorcov, aplikovanie spresňujúcich modelov a korekcií,
4. samotný výpočet, inverzia matice koeficientov normálnych rovníc,
5. zostavovanie výsledkov, v prípade nálezu odľahlých opráv nasleduje pridávanie ambiguít alebo vypúšťanie observácií z vyrovnania a opakovania postupu od druhého parsovania observácií.

4.3 Súčasný stav aplikácie JGPS

Aplikácia JGPS je v súčasnosti konzolová. Pri spustení je zadaný parametrický súbor, ktorý obsahuje umiestnenie observačných súborov a pracovný názov pre nasledujúci proces.

Základné vlastnosti:

- načítavanie súborov efemeríd priamo zo servera IGS v prípade ich lokálnej absencie,
- systém logovania troch typov údajov: informačné hlášky, varovné výpisy, chybové hlášky a oznamy pre ladenie s voliteľnou úrovňou podrobnosti,
- observačný súbor je spracovávaný pomocou dvoch prechodov, v prvom sú určené ambiguity a spracovateľné epochy, v druhom je zostavená matica plánu a vektor absolútnych hodnôt. Tieto procesy budú pravdepodobne spojené v prípade vyriešenia šetrného vytvárania matice o dynamickom rozmere v zmysle zabratej operačnej pamäte.
- výsledkom spracovania sú vektor odhadovaných parametrov, kovariančná matica parametrov a ich rotácia do topocentrického systému.

Medzi problémy, s ktorými momentálne JGPS zápasí, patrí detekcia fázových skokov v prípade, že tieto nie sú k dispozícii v observačnom súbore a robustný algoritmus na odstraňovanie odľahlých observácií. Momentálne je na detekciu odľahlých fázových observácií a nových ambiguít využité kódové riešenie, čo znamená výrazné spomalenie spracovania.

5. Záver

Dnes ešte stále existuje drvivá väčšina softvéru pre geodetické úlohy výhradne na platforme windows. Som toho názoru, že je to škoda. JAVA sa javí, ako ideálny prostriedok na písanie aplikácií, ktoré budú nezávislé na platforme, budú výkonnostne akceptovateľne zaostávať za natívne kompilovanými aplikáciami – napríklad v jazyku C++ a rozšíria možnosť voľby operačného systému.

Ďalším príjemným pozitívom JAVY je existencia vynikajúcich a voľne šíriteľných programátorských prostredí pre vývoj aplikácií a rozsiahla báza vedomostí v podobe internetových fór a elektronických publikácií.

Použitie konfigurácie:

- linux gentoo, windows 2000,
AMD 1GHz 266FBS, 512MB RAM,
J2SE 1.4.2, 1.5 (<http://java.sun.com>), eclipse 3.1.2 (<http://www.eclipse.org>)
gcc kompilátor 3.4.5 (linux), 3.4.2 (windows),
- linux debian 3.1,
Intel P3 800MHz, 128MB RAM,
J2RE, 1.5 (<http://java.sun.com>),
gcc kompilátor 3.3.

LITERATÚRA:

- [1] ASHBY, N.: Relativity in the Global Positioning System. Max Planck Institute for Gravitational Physics 2003.
- [2] BEUTLER, G.–BOCK, H.–BRÖCKMANN, E.–DACH, R.–FRIDEZ, P.–GURTER, W.–HABRICH, H.–HUGEN-TOBLER, U.–INEICHEN, D.–MEINDL, M.–MERVART, L.–ROTHACHER, M.–SCHAER, S.–SPRINGER, T.–URSCHL, C.–WEBER, R.: Bernese GPS Software Version 5.0 DRAFT. Bern, Printing Office of the University of Bern 1994.
- [3] DAVIS, J. L.–HERRING, T. A.–SHAPIRO, I. I.–ROGERS, A. E. E.–ELGERET, G.: Geodesy by radio interferometry: Effects of atmospheric modeling errors on estimates of baseline length. *Radio Science*, 1985, vol. 20, No. 6, pp. 1593–1607.
- [4] DOUŠA, J.: Využití GPS pro účely meteorologie. *Geodetický a kartografický obzor*, 46/88, 2000, č. 8, s. 153–159.
- [5] HOFFMAN-WELLENHOFF, B.–LICHTENEGGER, H.–COLLINS, J.: Global Positioning System Theory and Practice. Wien, New York, Springer-Verlag 1992.
- [6] KLEIJER, F.: Troposphere Modeling and Filtering for Precise GPS Levelling. Netherlands, Optima Grafische Communicatie 1992.
- [7] KOUBA, J.–HÉROUX, P.: GPS Precise Point Positioning Using IGS Orbit Products. Canada, National Resources 2000.
- [8] NIELL, A. E.: Global mapping functions for the atmosphere delay at radio wavelengths. *Journal of Geophysical Research*, 1996, vol. 101, No B2, pp. 3227–3247.
- [9] NIELL, A. E.: Improved atmospheric mapping functions for VLBI and GPS. *Earth Planets Space*, 2000, No. 52, pp. 699–702.
- [10] SAASTAMOINEN, J.: Contributions to the theory of atmospheric refraction, part I. *Bulletin Géodésique*, 1972, No. 106.
- [11] SAASTAMOINEN, J.: Contributions to the theory of atmospheric refraction, part I (Refraction correction in satellite geodesy). *Bulletin Géodésique*, 1973, No. 107.
- [12] SUDHIR MAN SHRESTHA: Investigations into the Estimation of Tropospheric Delay and Wet Refractivity Using GPS Measurements. Alberta, Calgary 2003.
- [13] ČEPEK, A.: gMatVec – C++ matrix / vector template library. [online]. Documentation version 0.15. 2001. <<http://gama.fsv.cvut.cz/cepek/gama/archive/matvec/old>>
- [14] BELL, D.: Make Java fast: Optimize! [online]. 1997. <<http://www.javaworld.com/javaworld/jw-04-1997/jw-04-optimize.html>>
- [15] ROULO, M.: Accelerate your Java Apps! [online]. 1998. <<http://www.javaworld.com/javaworld/jw-09-1998/jw-09-speed.html>>

Do redakce došlo: 21. 6. 2006

Lektoroval:
Ing. Marián Kováč, PhD.,
Katedra geodetických základov SvF STU v Bratislave

MAPY A ATLASY

TourMap 2006

912.43

Cestovatelé a turisté si každým rokem mohou vybírat z obsáhlejší, kvalitnější a rozmanitější nabídky kartografických děl a průvodců. Zatímco ty původní, které vznikly v České republice (ČR) v roce 2005, hodnotí komise Kartografické společnosti ČR, jiná komise TourMap složená ze zástupců obce novinářů, pedagogů, kartografů, knihkupců a manažerů z oblasti cestovního ruchu vybírala z celosvětové produkce nejlepšího průvodce a kartografické dílo s turistickým obsahem. Ocenění v této soutěži vydavatelé a tvůrci převzali na stejném místě jako se předávaly ceny za Mapu roku 2005 a to na veletrhu Svět knihy 2006 v Praze.

Z výběru nominovaných kartografických děl prestižní mezinárodní kartografické soutěže TourMap 2006 zvítězilo dílo polských kartografů Atlas Dolny Slask-Slask Opolski (nakladatel Ekograf), obr. 1, které získalo prvenství před New Zeland Motorcycle Atlas (vydaný nakladatelstvem Hema Maps (Austrálie) a dílem The Rough Guide Map to Spain and Portugal z dílny Rough Guides (Velká Británie).

Cenu poroty – Prix Jury získaly Autoatlas Evropa 1:750 000 od vydavatelství SHOCart (ČR), dále publikace Na kole celou Moravou a Slezskem (Fragment, ČR), dílo Biesz Czady (Compass, Polsko) a kolekce plastických map od Kartografie HP a SHOCart (ČR).

Nejvíce hlasů v kategorii turistických průvodců (mj. Jižní Afrika, obr. 1) získal soubor průvodců Lonely Planet (Svojtka & Co., ČR), před podobným souborem průvodců Rough Guides (JOTA, ČR) a Vodáckým průvodcem – Česko (SHOCart, ČR). Ceny poroty – Prix Jury si odnesli zástupci vydavatelů za publikace Centrální Kavkaz, Bezenski, Prielbusie, Dombaj (SKY, ČR), dále za titul Albánie – turistický a vlastivědný průvodce (Kudrna, ČR), Discover New Zeland (Hema Maps Austrálie) a Balaton – Felvidék Keszthelyi – Hegység (Cartographia, Maďarsko).

V kategorii elektronické mapy a průvodce převzali Cenu poroty – Prix Jury zástupci Seznam.cz a PLANstudio (ČR) za mapový produkt, který naleznete na adrese www.mapy.cz.

Při vyhodnocení soutěže bylo organizátory sděleno, že hlavní náplní TourMap není jen vlastní soutěž. Její prvotný cíl je jiný – představit veřejnosti přihlášené kartografické produkty a přiblížit tak ve-

řejnosti pro turisty nejkvalitnější mapy a průvodce. Na veletrhu Svět knihy si návštěvníci tak mohli přímo prohlédnout každou z přihlášených publikací. Podobnou výstavu shlédnou v říjnu v Karlových Varech i účastníci nejstaršího filmového festivalu turistických filmů na světě TourFilm 2006.

„Pořadatelé soutěžní přehlídky TourMap stojí na základním rozcestí, zda zůstat pouze v České republice, či potožmo ve střední Evropě, nebo se vydat za cenu velkých finančních investic směrem k mezinárodní prestižní soutěžní přehlídce turistických průvodců a map. Spojení s TourFilm a dalšími významnými uskupeními může významně napomoci ke druhé cestě.“ napsal po vyhodnocení soutěže TourMap 2006 člen soutěžní komise kartograf Jakub Zmrzlík.

Ing. Petr Skála,

Fakulta lesnická a environmentální ČZU v Praze

OSOBNÉ SPRÁVY

Jubileum Ing. Nadeždy Nikšovej

92.Nikšová:528

Ani se nechce verit, ale je tomu tak Ing. Nadežda Nikšová, riaditeľka odboru medzinárodných vzťahov Úradu geodézie, kartografie a katastra (ÚGKK) Slovenskej republiky (SR) v plnom rozmachu tvorivých síl oslávila okružle jubileum. Potvrdila, že i mladým sa posúva hranica veku.

Narodila sa 10. 9. 1955 v Nitre, kde v roku 1973 maturovala na strednej všeobecnovzdelávacej škole. Odbor geodézie a kartografia skončila na Stavebnej fakulte Slovenskej vysokej školy technickej v Bratislave v roku 1978 s vyznamenaním. V tomto roku nastúpila do Inžiniersko-projektovoj organizácie školských stavieb v Bratislave, kde ako vedúca geodet vykonávala práce inžinierskej geodézie: zakladanie bodového poľa, podklady na projektovú dokumentáciu stavieb, vytyčovanie stavieb, meranie skutočného vykonania stavieb a pod.

Ing. N. Nikšová prišla 1. 1. 1983 do Správy geodézie a kartografie v Bratislave do oddelenia na zadávanie a preberanie prác a od roku 1986 bola vedúcou oddelenia technickej dokumentácie. 1. 1. 1987 prešla do Slovenského úradu geodézie a kartografie (od 1. 1. 1993 ÚGKK SR) do technického odboru, ktorý sa postupne pretransformoval na odbor geodézie a kartografie, odbor geodézie, kartografie a medzinárodných vzťahov a napokon odbor geodézie, kartografie a geoinformatiky. Hlavnou náplňou Ing. Nikšovej bola koncepcná a normotvorná činnosť v oblasti tvorby základnej bázy geografického informačného systému, archivovanie a dokumentácia, vrátane koordinácie týchto činností na medzirezortnej úrovni s väzbou na medzinárodnú spoluprácu. Významná bola jej aktívna činnosť v poradných orgánoch, v pracovných skupinách a komisiách ÚGKK SR, činnosť v medzirezortných skupinách a komisiách a činnosť v medzinárodných skupinách expertov. Od 1. 1. 1997 vykonávala aj funkciu zástupcu riaditeľa odboru.

Ing. Nikšová sa význačne podieľala na tvorbe rezortnej Inštrukcie na dokumentačnú činnosť a skartačný plán dokumentačných fondov, ďalej na tvorbe zákona Národnej rady (NR) SR č. 215/1995 Z. z. o geodézii a kartografii, zákona NR SR č. 216/1995 Z. z. o Komore geodetov a kartografov a vyhlášky ÚGKK SR č. 178/1996 Z. z., ktorou sa vykonáva zákon NR SR o geodézii a kartografii. Je tiež publikačne činná.

V roku 2003 začala Ing. Nikšová utvárať nový odbor medzinárodnej spolupráce a európskej integrácie (od 20. 12. 2004 odbor medzinárodných vzťahov) ÚGKK SR. Funkciu jeho riaditeľky vykonáva od 28. 8. 2003. Tu ovplyvňuje tvorbu strategických zámerov ÚGKK SR a ich presadzovanie v komisiách alebo vo výboroch zriadených Európskou komisiou alebo Radou Európskej únie (EÚ), ďalej čerpanie prostriedkov európskych spoločenstiev so zameraním na oblasť geoinformatiky a katastra nehnuteľností, tvorbu legislatívnych opatrení na úseku kompetencií ÚGKK SR, súvisiacich s rámcovou pozíciou SR na rozhodovací proces EÚ, ako aj viaceré činnosti bilaterálnej a multilaterálnej spolupráce.

V rámci členstva v profesijných združeniach Ing. Nikšová vystupuje aktívne od roku 1998 ako dopisovateľka EuroGeographics, od



Obr. 1 Na veletrhu Svět knihy si všechny publikace mohli osobně prohlédnout návštěvníci veletrhu. Jedna z návštěvnic veletrhu má v rukách vítěznou publikaci

roku 1997 ako expert SR v skupine expertov na kvalitu EuroGeographics, v rokoch 1997 až 2004 bola expertom SR v skupine expertov na právne a komerčné záležitosti EuroGeographics, od roku 1999 je národným spracovateľom technickej normalizácie pre CEN technickú komisiu 287 Geografické informácie (GI) a technickej normalizácie pre ISO technickú komisiu 211 GI, od roku 1999 je členkou technickej komisie 89 Geodézia a geoinformatika, od roku 2004 je expertom SR na zasadnutiach pracovnej skupiny pre životné prostredie Rady EÚ (Working Party on Environment) k návrhu smernice INFrastructure for SPatial InfoRmation in Europe (INSPIRE), v rokoch 2000 až 2004 bola tajomníčkou pracovnej skupiny pre geografické informačné systémy v štátnej správe pri Rade vlády pre informatiku a v rokoch 1989 až 2002 bola technickou redaktorkou Spravodajcu ÚGKK SR. V stálej štátnej službe je od 30. 8. 2002.

Široký odborný rozsah pôsobnosti Ing. *Nikšovej* okrem odborných vedomostí umožňuje aj dobré ovládanie cudzích jazykov. V roku 1998 vykonala skúšku 2. základného stupňa na Rakúskom inštitúte v Bratislave z jazyka nemeckého a v roku 2003 štátnu záverečnú skúšku na Štátnej jazykovej škole v Bratislave z jazyka anglického.

Do ďalších rokov želáme Ing. *Nadežde Nikšovej* veľa zdravia, aby jej terajší optimizmus ešte dlho vydržal a v práci i v osobnom živote nové úspechy.

NEKROLOGY

Za Ing. Pavlom Vyskočilom, DrSc.

92. Vyskočil: 528



Dne 26. června 2006 zemrel v Praze ve věku 72 let náš kolega Ing. Pavel Vyskočil, DrSc., bývalý vedoucí vědecký pracovník ve Výzkumném ústavu geodetickém, topografickém a kartografickém (VÚGTK) a v letech 1976 až 1995 ředitel Mezinárodního centra pro recentní pohyby zemské kůry (International Center on Recent Crustal Movements – ICRCM).

Poslední rozloučení s Pavlem Vyskočilom se konalo v pátek 30. června 2006 v 10 hodin v bazilice Nanebevzetí Panny Marie

ve Strahovském klášteře v Praze-Břevnově.

Za účasti rodiny, známých a bývalých spolupracovníků se se znenulým rozloučil opat klášteřa řádu premonstrátsů Strahovského kláštera.

Pavel Vyskočil se narodil 9. 2. 1934 v Praze. Zeměměřičské inženýrství na ČVUT vystudoval v roce 1957. Ihned po nástupu do praxe se začal zabývat otázkami přesné nivelace. Zájem o tento obor jej v roce 1959 přivedl do VÚGTK, kde ho Ing. B. Kruis, CSc., inspiroval k vědecké práci. Rozbory a analýzy nivelačních sítí a bohaté zkušenosti s vývojem a výsledky zpracování nivelačních sítí jej přivedly ke studiu recentních pohybů zemské kůry. Tento obor se pak stal jeho předním a celoživotním zájmem.

Pavel Vyskočil se i v posledních letech svého života, přes fyzické omezení zaviněné krutou chorobou, nepřestával věnovat svému vědeckému oboru – geodetické geodynamice. Patřil k zakladatelům tohoto oboru u nás a prostřednictvím svých prací se stal známým i v zahraničí.

Z posledních let jmenujme jedno z mnoha ocenění a uznání jeho práce i v mezinárodním měřítku. Je to pamětní medaile Národního výzkumného ústavu astronomie a geofyziky (NRIAG) v Helwanu v Egyptě za jeho zásluhu o rozvoj egyptské geofyziky, která mu byla udělena při příležitosti 100. výročí založení tohoto ústavu. Medaili za něho převzal zástupce českého velvyslanectví v Egyptě shodou okolností právě v době jeho 70. narozenin v roce 2004. Této pocty,

kteř se dostalo jen několika světovým astronomům a geofyzikům, si velmi vážil. Připomínala mu dobu, kdy za podpory UNESCO v letech 1983 až 1993 vedl v tomto egyptském ústavu výzkum a studium recentních pohybů zemské kůry ve středním Egyptě.

Před tím, na konci roku 1984, nastalo totiž v Egyptě, na západním břehu Aswanského jezera, zemětřesení na zlomu Kalabsha. Toto zemětřesení vyvolalo obavy, že v případě jeho možného neočekávaného opakování by mohla být poškozena High Dam – Vysoká přehrada na Nilu. Protržení hráze 400 km dlouhého jezera, které zadržuje miliardy kubických metrů vody, by znamenalo ohrožení celé obydlené úrodné části Egypta.

Ústav NRIAG v Helwanu v Egyptě zintenzivnil svá seismická pozorování a požádal o pomoc i zahraniční vědecké organizace. Byl zakoupen systém telemetrických seismických stanic a Mezinárodní centrum pro recentní pohyby (ICRCM) se sídlem ve VÚGTK v Zdi-bech bylo požádáno o geodetické zabezpečení akce.

Pavel Vyskočil navrhl a postupně vybudoval v saharské pouštní oblasti Egypta vnitřní lokální síť, které byly zahrnuty do regionální vnější sítě, zasahující na oba břehy jezera. Podstatou celé práce byl ověřený předpoklad, že seismickému jevu předchází deformace zemské kůry a povrchu. V daném případě měly být předmětem sledování vodorovné pohyby a náklony – svislé pohyby v zájmovém prostoru. Vycvičil egyptské pracovníky pro přesná geodetická měření, sběr dat a jejich analýzy. Vedl aspiranty pro stupeň Mgr., CSc. a Ph.D., jakož i publikoval mnoho společných výzkumných pojednání. Zde se svými spolupracovníky na praktickém příkladu aplikoval výsledky svých teoretických prací.

Svými pracemi a tvorbou unikátních map recentních pohybů se stal mezinárodně uznávaným odborníkem. Ve VÚGTK byl jmenován vedoucím oddělení teoretické geodézie (1971 až 1983). Jeho zásluhou se VÚGTK stal sídlem Mezinárodního centra pro recentní pohyby zemské kůry (ICRCM). Na zasedání IAG (Mezinárodní geodetická asociace) byl Pavel Vyskočil v roce 1976 jmenován ředitelem tohoto centra.

V letech 1990 až 1993 byl zástupcem ředitele ústavu a předsedou vědecké rady VÚGTK. Byl dlouholetým funkcionářem IAG a členem i funkcionářem mnoha našich a mezinárodních odborných a vědeckých organizací a komisí. I jako důchodce byl vedoucím studijní skupiny mezinárodního projektu CERGOP (Central Europe Regional Geodynamics Project). Do posledních let odborně pracoval a udržoval kontakty i v zahraničí. Byl stále všestranně aktivní, vypracovával anotace článků a překlady odborné zahraniční literatury pro Odvětvové informační středisko VÚGTK atd.

Bohatá byla jeho činnost publikační, zahrnující 60 titulů. Z nich uvedeme alespoň monografii „Results on Recent Crustal Movements Studies“. Za svou celoživotní práci obdržel řadu významných domácích i mezinárodních ocenění. Velmi se zasloužil o dobré jméno naší geodézie na mezinárodním poli.

Pavel Vyskočil prožil celý život v Praze-Břevnově, který měl rád jako svůj domov a kde čerpal sílu pro práci do posledních týdnů svého života. Všichni, kdo jsme s Pavlem Vyskočilom spolupracovali, jsme si ho vážili pro jeho přátelský a lidský přístup, pochopení a ochotu každému pomoci.

Cest jeho památce!

OZNÁMENÍ

Seminář „Applikace družicových měření v geodézii“

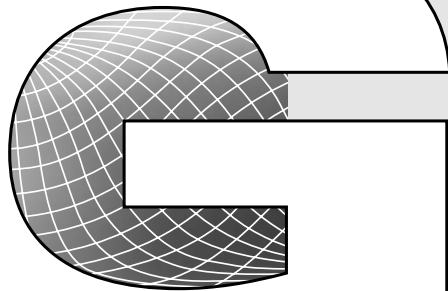
Ústav geodézie Stavební fakulty VUT v Brně pořádá seminář s mezinárodní účastí s názvem

APLIKACE DRUŽICOVÝCH MĚŘENÍ V GEODÉZII

ve čtvrtek dne 1. února 2007 od 9.30 h do 15.30 h v posluchárně D182 budovy D, na Veveří 95.

Přihlášky je možno posílat na adresy: Ústav geodézie, Veveří 95, Brno 602 00; e-mail <zmolova.v@fce.vutbr.cz>; fax 541 147 218

Prof. Ing. Zdeněk Nevošád, DrSc.,
FAST VUT v Brně



GEOS

2. mezinárodní veletrh geodézie, kartografie, navigace a geoinformatiky

1. - 3. 3. 2007 ■ PVA Letňany - PRAHA

■ ■ ■ Nomenklatura

1 GEODÉZIE

1.1. Geodetické přístroje

- teodolity
- dálkoměry
- totální stanice
- geodetické systémy GPS/ DGPS
- GPS navigační a lokalizační přístroje do aut, lodí, letadel a pro turistiku
- nivelační přístroje optické i digitální
- křížové, rotační a potrubní lasery
- přístroje a pomůcky v astronomii
- planimetry
- geofyzikální přístroje
- přístroje pro vyhledávání podzemních inženýrských sítí

1.2. Další geodetické pomůcky a příslušenství

- stabilizační znaky v geodézii
- pásma
- stativy
- odrazné hranoly

1.3. Geodetický software

- geodetické výpočty
- zeměměřičké a katastrální aplikace
- zpracování GPS dat
- transformace souřadnic
- vyrovnání sítí
- CAD
- zpracování rastrových dat
- digitální model terénu
- stavba a plánování pozemních komunikací
- stavba a plánování inženýrských sítí
- software pro realitní kanceláře

1.4. Měřické přístroje a pomůcky pro stavebnictví

1.5. Geodetické práce

- geodetické základy
- permanentní a virtuálních sítí referenčních stanic GPS a GLONASS (CZEPOS, by /S@T, TopNET, atd.), databáze geodetických bodů (DATAZ, ČSNS, atd.), výzkumné a experimentální sítě pro geodézii a geodynamiku (VESOG, atd.)
- stavební a inženýrská geodézie
- příprava stavby, tvorba mapových podkladů pro stavební projekty, vytyčovací práce na stavbách, dokumentace provedení stavby
- speciální práce
- geodetická bezpečnostní měření a měření v průmyslu, geofyzikální měřicí technika a technologie pro geofyzikální výzkum a průzkum, ekologické audity a monitoring
- důlní měřičství
- na hlubinných a povrchových dolech a lomech, při výstavbě metra, tunelů a kolektorů, zaměřování podzemních prostor (pod historickými jádry měst, jeskyní apod.)

1.6. Katastr nemovitostí

- informační systém katastru nemovitostí (ISKN)
- dálkový přístup do ISKN a služba „Nahlížení do katastru nemovitostí“
- tvorba geometrických plánů
- katastrální mapy
- oceňování nemovitostí
- služby realitních kanceláří
- software pro práci s údaji katastru nemovitostí

2 KARTOGRAFIE A MAPOVÁNÍ

2.1. Přístroje a pomůcky v kartografii

2.2. Mapová tvorba

- civilní státní mapová díla - základní mapy, SM5
- turistické a cyklistické mapy
- vojenské mapy
- ortofotomapy
- technické mapy měst
- tématické a účelové mapy pro hospodářskou, vědeckou a úřední potřebu
- plány a atlasy měst a glóby
- kartografická díla pro školy a veřejnost
- historické mapy
- digitální mapy (mapy v počítači)

3 FOTOGRAMMETRIE A DÁLKOVÝ PRŮZKUM

3.1. Přístroje

- letecké kamery, fotogrammetrické kamery, radary, skenery, lidary
- navigační a další pomocné systémy
- letecké snímkování, satelitní obrazové záznamy
- nosiče aparatur

3.2. Zpracovatelský a interpretační SW

- SW pro příjem a základní zpracování obrazu
- interpretační software
- tématicky orientovaný SW

3.3. Služby leteckého snímkování a snímání, distribuce satelitních dat

4 GEOINFORMACE A GIS

4.1. Informační systémy a databáze geoinformací a odborné literatury

- znalostní a informační systémy odborných geoinformací
- metadatové systémy
- odborná literatura

4.2. Geografické informační systémy (GIS)

- GIS-Software
- územně orientované informační systémy
- informační systémy měst a obcí
- tématicky orientované informační systémy
- zavádění informačních systémů
- sběr a zpracování dat pro GIS

4.3. Geodata

- databáze geodat
- technologie zpracování, distribuce a správy geodat

5 NAVIGAČNÍ SYSTÉMY PRO DOPRAVNÍ PROSTŘEDKY A TURISTIKU

5.1. Navigační systémy pro osobní automobily, lodě a letadla

5.2. Navigační systémy pro turistiku

5.3. Navigační systémy pro sledování vozů a nákladů a pro elektronické mýtné

5.4. Služby založené na lokaci (Location Based Services)

6 KANCELÁŘSKÉ VYBAVENÍ

6.1. Velkoplošné tiskárny a reprografické přístroje

6.2. Velkoplošné skenery

7 ODBORNÉ ORGANIZACE A ŠKOLY, VÝZKUM A VÝVOJ

www.geos.eu





GEOS

2. mezinárodní veletrh geodézie, kartografie, navigace a geoinformatiky

1. - 3. 3. 2007 ■ PRAHA

1. - 2. 3. 2007
Odborná konference

VÍCE INFORMACÍ:
www.geos.eu



www.terinvest.com