

Západočeská univerzita  
Fakulta aplikovaných věd  
Obor: IVT/PG

**Název úlohy, projektu:** Normály ve vrcholech trojúhelníkových sítí

**Zadání úlohy** (text zadání):

**Prohlášení:**

Prohlašuji, že všechny použité zdroje jsou řádně citovány a předkládaná práce byla vytvořena zcela mnou, pokud není uvedeno jinak.

Jméno studenta: Libor Váša	Jméno cvičícího: prof. V. Skala
e-mail: <a href="mailto:xrael@yahoo.com">xrael@yahoo.com</a>	Datum přijetí:
URL uložení všeho co se k úloze vztahuje včetně datových množin (prosím udělejte úvodní htm stránku): <a href="http://home.zcu.cz/~lvasa/apg">http://home.zcu.cz/~lvasa/apg</a>	Hodnocení:
Datum odevzdání: 15.11.2002	Počet bodů:
Podpis studenta:	Podpis cvičícího:

Použité prostředky .....	3
Implementované metody .....	3
Gouraud .....	3
Gouraud s normalizací .....	3
Thurmer .....	3
Vážení plochou .....	3
Inverzní vážení plochou .....	3
Metoda nejmenších čtverců .....	4
Metoda nejmenších čtverců $z=1$ .....	4
Testovací data .....	4
Funkce typu $z=f(x,y)$ .....	<b>Error! Bookmark not defined.</b>
Elipsoid .....	4
Přesné normály .....	4
Náhodné vzorkování .....	4
Umělá chyba .....	5
Metody testování .....	5
Testy přesnosti .....	5
Testy numerické stability .....	7
Testy rychlosti .....	7
Další testování .....	9

## **Použité prostředky**

V implementaci jsem vyšel z rendereru realizovaného pro předmět ZPG. Ten umožňuje mimo jiné načtení objektu ze souboru ve formátu .TRI, jeho zobrazení a vystínování Gouraudovou metodou a některé základní transformace objektu (translace, rotace, změna měřítka). Tyto funkce byly velmi užitečné pro aběžnou kontrolu správnosti genrovaných dat.

Z této volby rovněž vyplývá volba vývojového prostředí, kterým nutně musely být Delphi 5.0 implementující jazyk Object Pascal.

Procedury realizující výpočet normál jsou rozhraním vytvořené jednotky normals, kterou nyní renderer při realizaci Gouraudova stínování volá.

Druhou částí práce je generátor testovacích dat. Ten byl (především pro snadnou zajistitelnost kompatibility) realizován ve stejném vývojovém prostředí.

## **Implementované metody**

Po zvážení všech aspektů vyplývajících z teoretické části jsem se rozhodl implementovat následující metody:

- Gouraud
- Gouraud s normalizací
- Thurmer
- Vážení plochou
- Inverzní vážení plochou
- Metoda nejmenších čtverců
- Metoda nejmenších čtverců pro  $z=1$

### **Gouraud**

Gouraudova metoda tak jak je definována neříká nic o tom, jaká má být velikost normál, které jsou sečteny pro výslednou normálu ve vrcholu. V implementaci proto přímo sčítám výsledky Newellovy metody určující normály trojúhelníkům.

### **Gouraud s normalizací**

Jinou možností je všechny normály před sečtením normalizovat, což realizuje tato metoda. Úspěšnost Gouraudových metod závisí na povaze vstupních dat, proto jsem je ponechal implementované zvlášť.

### **Thurmer**

Implementace Thurmerovy metody se nijak neliší od své teoretické definice, nutno je pouze podotknout že jednotlivé normály jsou před vážením sečtením normalizovány.

### **Vážení plochou**

Metoda vážení plochou je implementována podle své definice, používá normalizované normály pro součet.

### **Inverzní vážení plochou**

Metoda je implementována stejně jako metoda předchozí, pouze zlomek určující váhu je brán převráceně, tj.  $S/S_i$ . Tento postup není převzat z žádného ze zdrojů, jeho myšlenka je proto prozatím považována za původní.

## Metoda nejmenších čtverců

Tímto názvem je míněna metoda aproximace tečné nadroviny metodou nejmenších čtverců. V tomto případě je metodou nejmenších čtverců minimalizována funkce  $d=ax+by+cz$ . Tím je dosaženo obecných řešení, nicméně tím v některých případech může utrpět přesnost (viz dále měření).

Jako problém se u této metody ukázalo být určení orientace získané normály. Bohužel se zatím nepodařilo najít lepší způsob než určit normálu k některému z přilehlých trojúhelníků a podle velikosti úhlu mezi touto a vypočtenou normálou pak případně orientaci vypočtené normály změnit. Stejný problém je i s následující metodou.

## Metoda nejmenších čtverců $z=1$

Metoda vychází ze stejného teoretického základu jako metoda předchozí, ale metodou nejmenších čtverců je minimalizována funkce  $z=ax+by+c$ . Tím je dosaženo v některých případech (explicitní funkce typu  $z=f(x,y)$ ) větší přesnosti, nicméně metoda není schopna určit normálu v místech, kde by byla rovnoběžná s osou  $z$ . To je samozřejmě způsobeno tvarem rovnice, z něhož automaticky pro  $Z$  složku normály vyplývá velikost 1. Odtud také název.

## Testovací data

Testovací data jsou generována do souboru formátu TRI automaticky. Implementovaný program datagen rovněž automaticky počítá přesné normály ve vrcholech generovaných trojúhelníkových sítí.

## Explicitní funkce

Jako testovací funkci jsem zvolil (vycházejí z [3]) funkci  $z=0.2 \sin(15x)+0.2 \sin(15y^2)$ . Tato funkce poskytuje ve svých místech různou míru zvlnění (vlivem druhé mocniny argumentu  $y$ ). Funkce je vždy vzorkována přes interval  $\langle 0,1 \rangle$  u obou proměnných. Program datagen umožňuje libovolné vzorkování, pro testování byla použita vzorkování od  $10 \times 10$  (zjevně podvzorkovaná data) až po  $630 \times 630$ . Jako mezihodnoty byly použity hodnoty řady E5 vhodné pro zobrazení v logaritmickém měřítku.

## Elipsoid

Program datagen umožňuje rovněž generování standardně parametrizovaného elipsoidu s volitelnou hustotou vzorkování pro oba parametry a volitelnou excentricitou ve dvou osách.

## Přesné normály

Pro potřeby programu jsem zavedl generování normál do souborů s příponou .NM, ve kterých jsou vždy uloženy všechny normály postupně jako trojice čísel. Předpokládá se stejné pořadí jako pořadí vrcholů v souboru formátu TRI. Program používá pro výpočet normál vektorového součinu tečných vektorů získaných z parciálních derivací. Pro výpočet normál elipsoidu je použit vektorový součin normál k tečným vektorům ve směru růstu parametrů (podrobněji [5] str. 288-289).

## Náhodné vzorkování

Při zapnutí volby náhodné vzorkování v některé z os je příslušný z parametrů vzorkován náhodným krokem. Velikost náhodného kroku je minimálně  $1/10$  pravidelného kroku a maximálně  $19/10$  pravidelného kroku, tudíž je průměrná délka náhodného kroku rovna délce kroku pravidelného.

## Umělá chyba

Uživatel rovněž může zadat velikost umělé chyby. Číslo které zadá se bere jako maximální možná chyba zavedená do každé souřadnice vrcholu, skutečná chyba (vzdálenost přesného a chybného vrcholu v prostoru) pak může být samozřejmě větší. Chyba se zavádí na obě strany, tj. chybný element může mít záporné i kladné znaménko.

Program datagen vždy generuje soubor output.tri, popř. normals.nm do adresáře z něhož byl spuštěn.

## Metody testování

U každé metody jsem testoval celkem tři její aspekty – přesnost, rychlost a numerickou stabilitu. Dále jsem zavedl test podobnosti s ideálním obrazem, který se však neukázal jako spolehlivý.

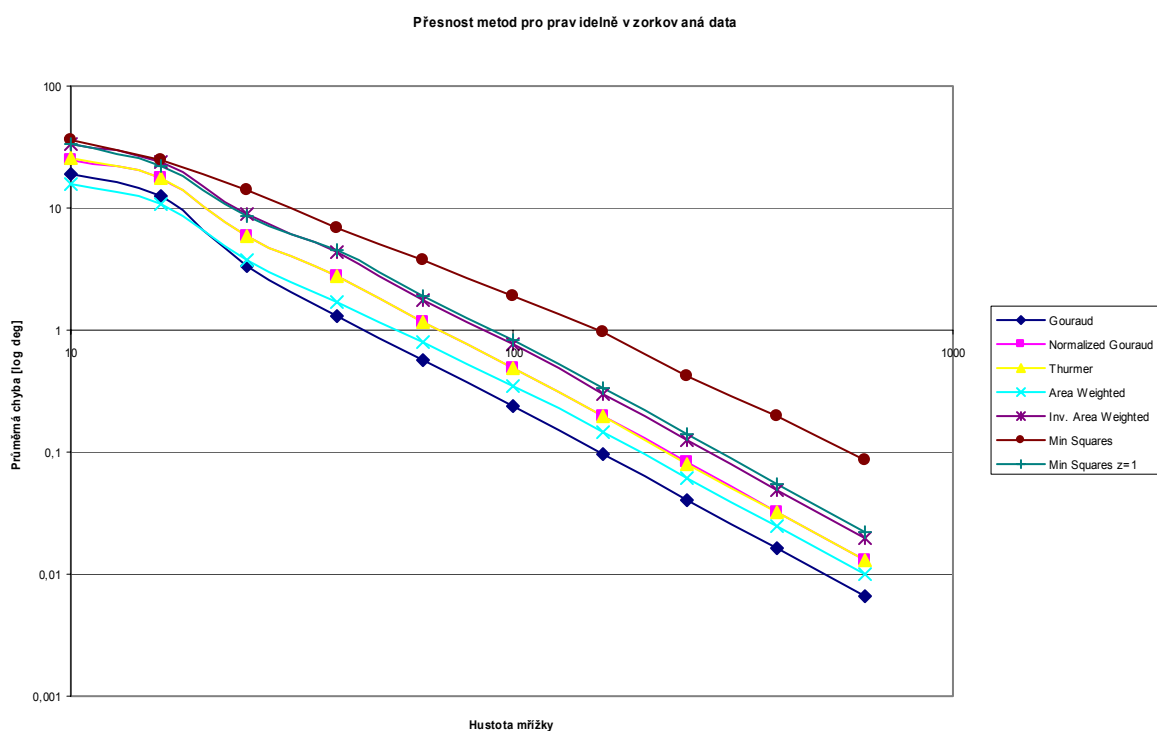
## Testy přesnosti

Testy přesnosti jsem realizoval jako testy úhlové odchylky přesné a vypočtené normály. Tato úhlová odchylka je průměrována přes všechny vrcholy. Průměrnou úhlovou odchylku jsem testoval v závislosti na hustotě vzorkování a excentricitě elipsoidu. Zvláštní testy jsem prováděl pro pravidelně a nepravidelně vzorkovaná data. Výsledky měření ukazují grafy 1 a 2, lépe čitelné a podložené daty jsou tyto grafy v příloženém souboru normals.xls.

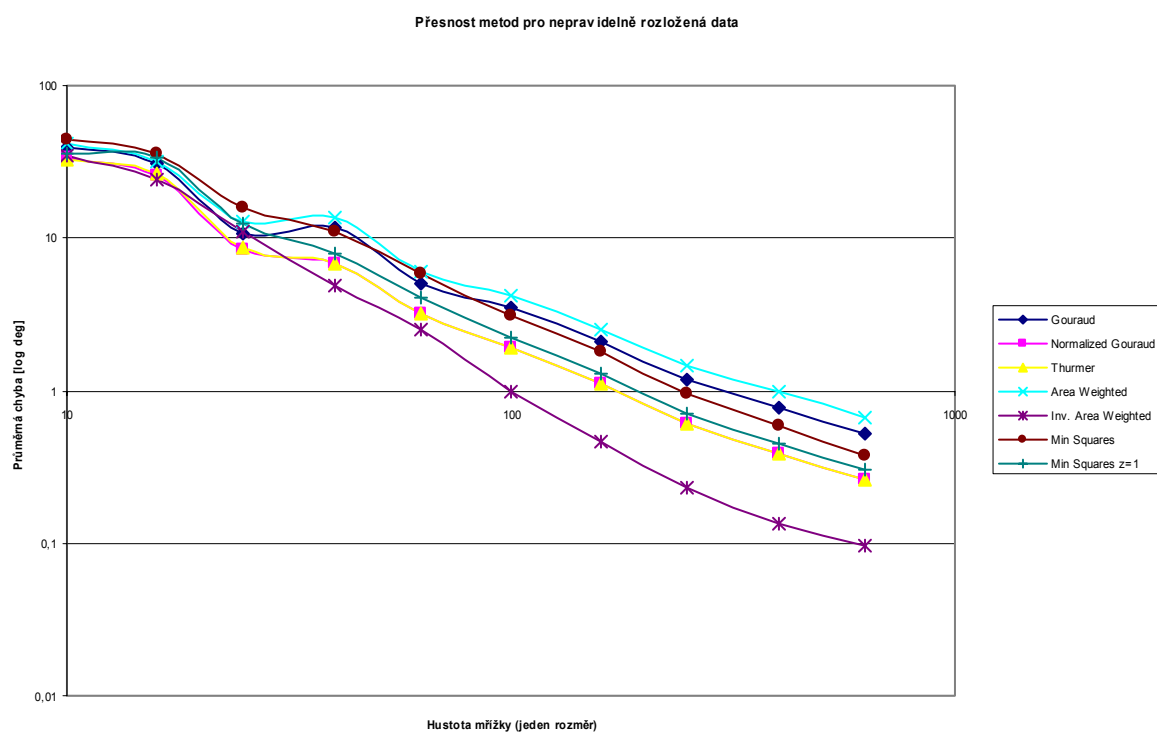
Grafy 3 a 4 pak ukazují výsledky na elipsoidu. V grafu 3 je test proveden na pravidelně vzorkovaném elipsoidu s excentrickou osou Y, v grafu 4 je pak použit nepravidelně vzorkovaný elipsoid s excentrickou osou Z. Z grafu 4 je patrný pokles přesnosti metody nejmenších čtverců pro  $z=1$ . To je způsobeno zvětšující se oblastí kde je tečná nadrovina téměř rovnoběžná s osou z.

Měřítko všech grafů jsou logaritmická.

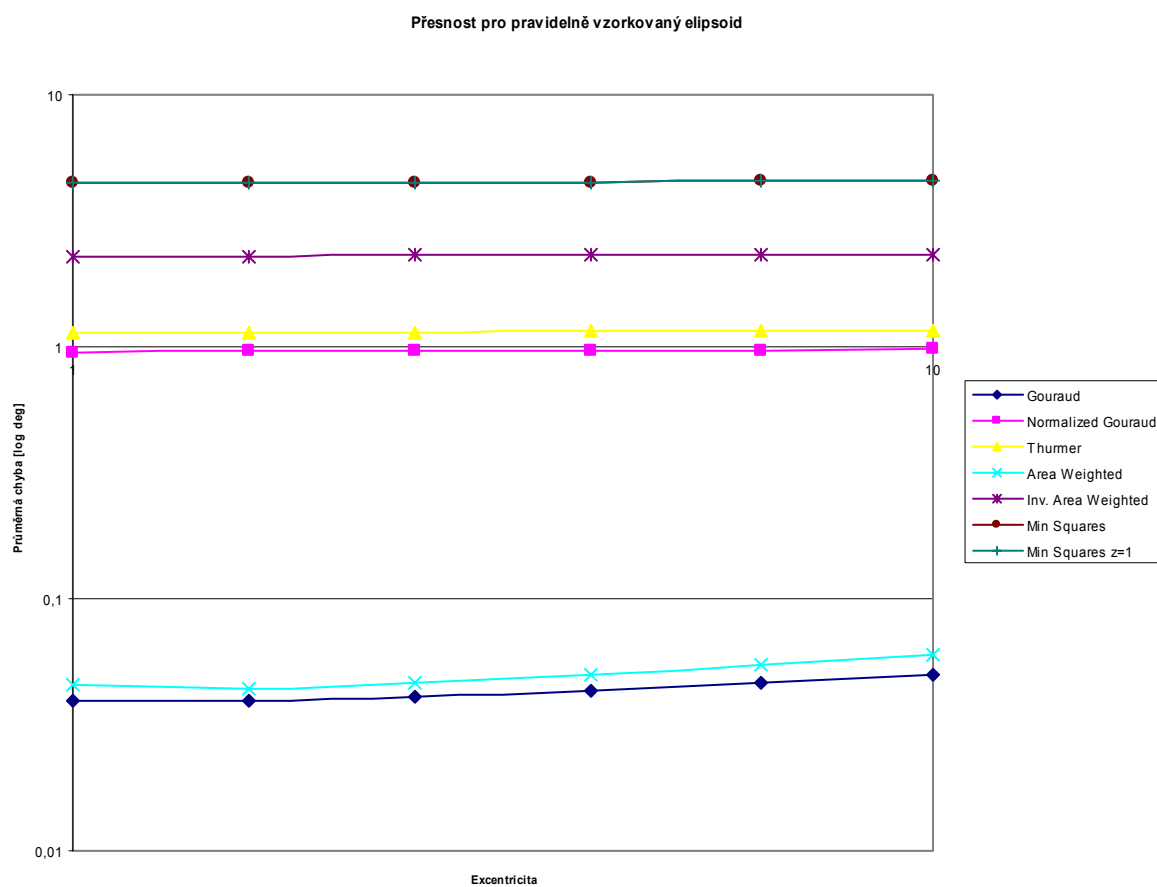
Graf 1:



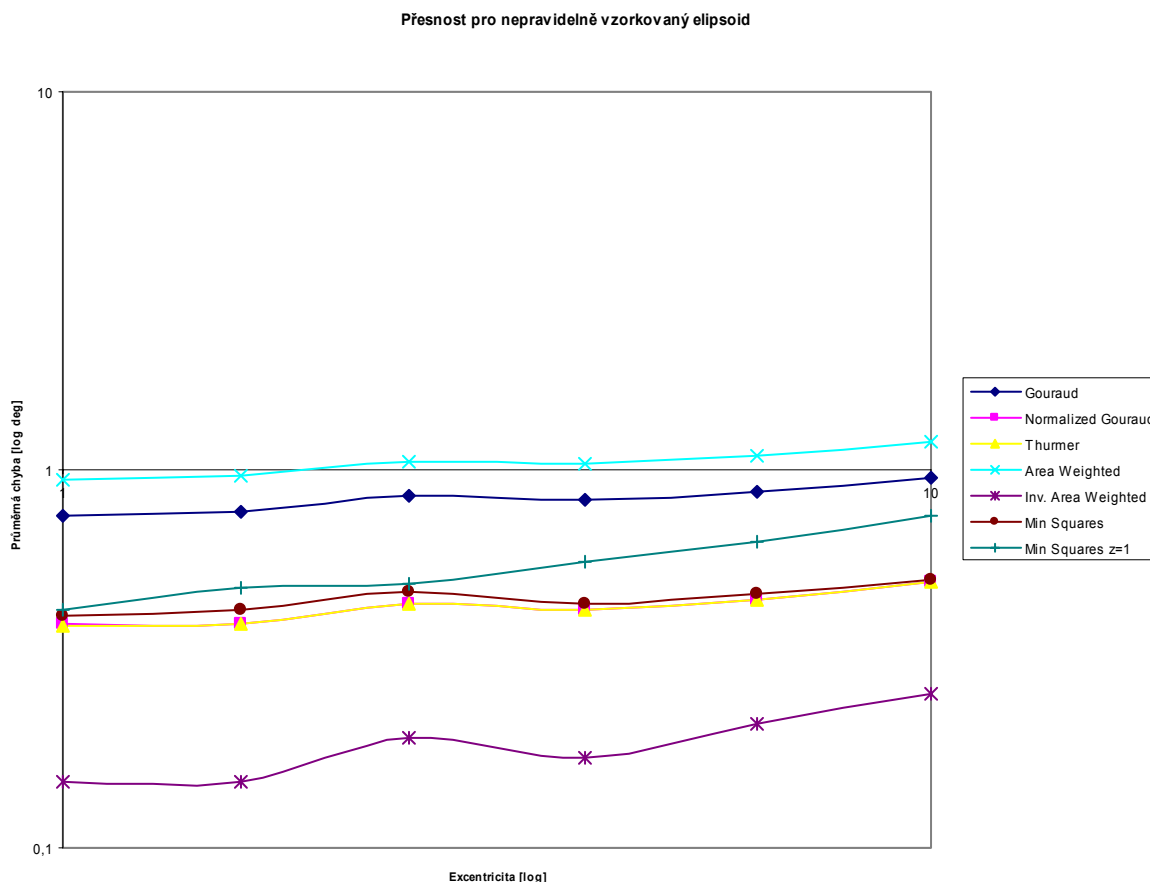
Graf 2:



Graf 3:



Graf 4:



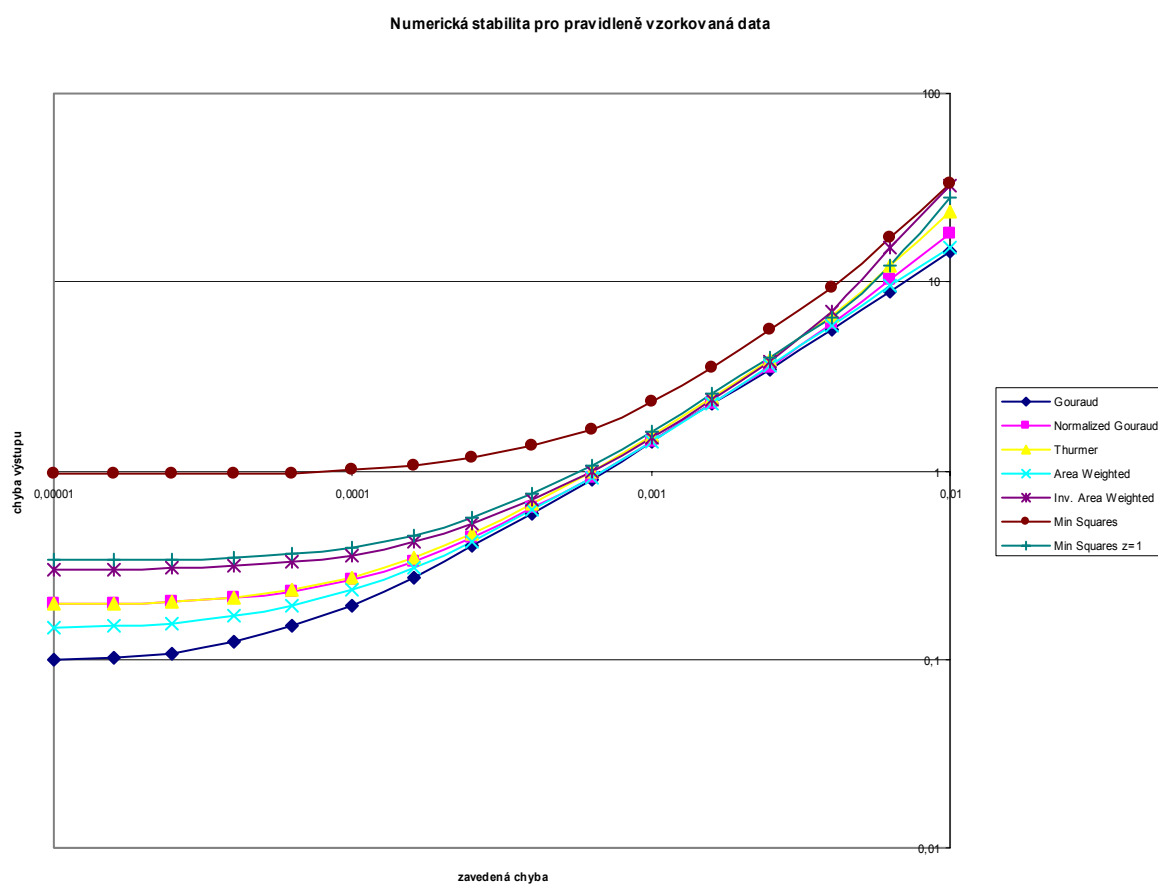
## Testy numerické stability

Pro testování numerické stability bylo použito zavedení umělé chyby do vstupních dat a sledování jejího vlivu na data výstupní. Pro testování byla použita explicitní funkce a mřížka rozměru 160x160. Pro tento rozměr je z předešlých grafů patrné, že vzniklá chyba není chybou podvzorkování. Zvláště byly provedeny testy pro pravidelné a nepravidelné vzorkování, maximální zavedená chyba byla volena tak, aby dokonce mírně přesáhla velikost okénka mřížky. Výsledky dokumentují grafy 5 a 6, měřítka jsou opět logaritmická. Z grafů mj. vyplývá, že i pro pravidelně vzorkovaná data od jisté velikosti zavedené chyby v zásadě nezáleží na použité metodě.

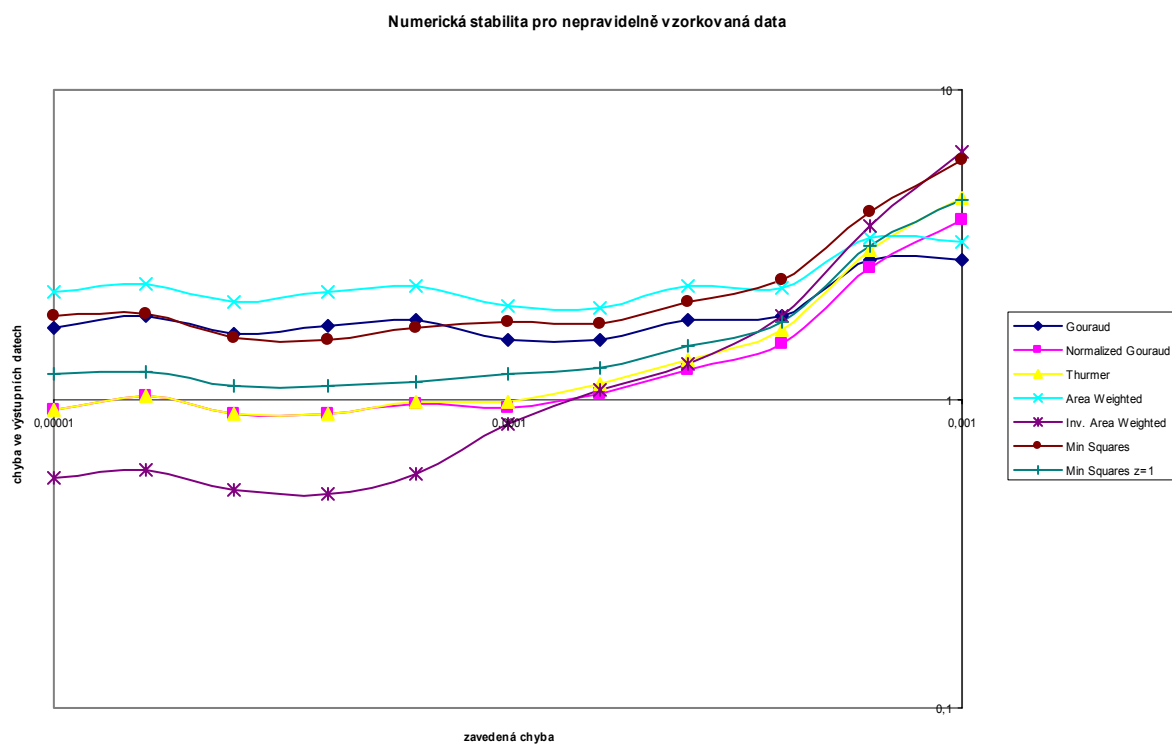
## Testy rychlosti

Testy byly provedeny pouze pro pravidelně vzorkovaná data, graf 7 ukazuje závislost doby výpočtu na počtu vrcholů trojúhelníkové sítě. Pro ostatní možná vstupní data se metody chovají obdobně. Všechny metody mají výpočetní složitost  $O(n)$ . Dle očekávání se jako nejpomalejší ukázala být metoda nejmenších čtverců která má jednak poměrně složitý aparát výpočtu a která navíc vyžaduje určení normály k některému z přilehlých trojúhelníků pro určení orientace vypočtené normály.

Graf5:

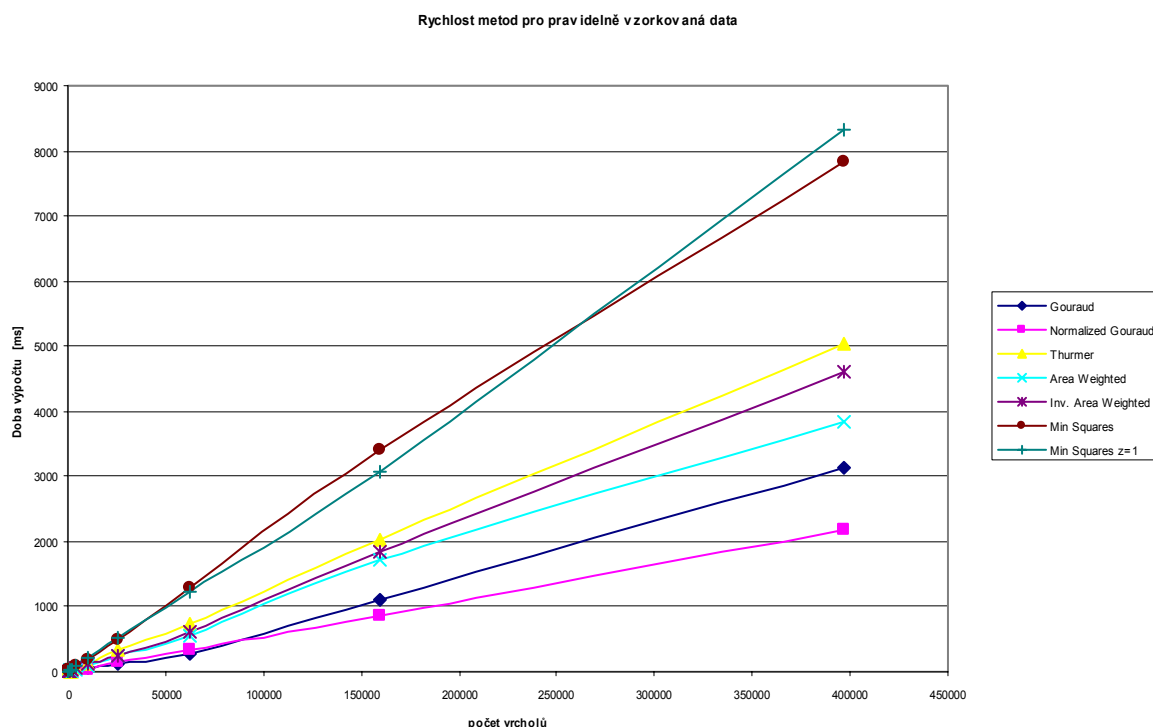


Graf 6:





Graf 7:



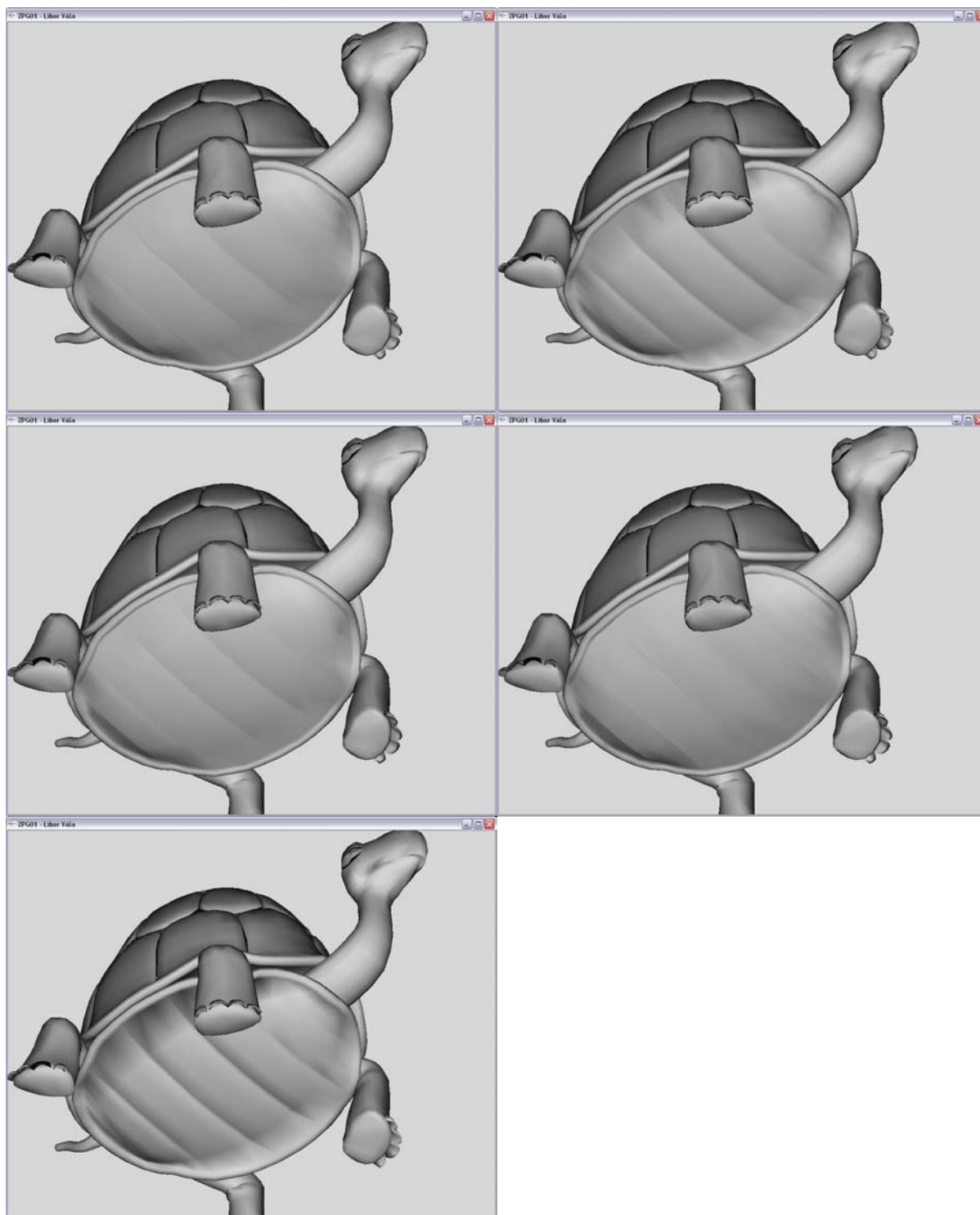
## Další testování

Kromě exaktních metod testování přesnosti jsem rovněž zkoumal intuitivní kvalitu metod na datech, u kterých přesné hodnoty normál nejsou známy. Z tohoto hlediska je především zajímavé jakým způsobem jednotlivé metody zacházejí z detaily objektů. Je možno konstatovat, že metody Gouraud a vážení plochou mají tendenci detaily spíše potlačovat, zatímco metody Gouraud s normalizací, metody aproximace tečné nadroviny a především metoda inverzního vážení plochou mají tendenci detaily zvýrazňovat. Tento jev dobře dokumentuje série obrázků 1, které byly vytvořeny metodami Gouraud, Gouraud s normalizací, Thurmer, vážení plochou a inverzní vážení plochou (v tomto pořadí).

Pokusil jsem se rovněž odpovědět na otázku jak daleko jsou použité metody od „dokonalosti“, tj. do jaké míry je obraz generovaný příslušnou metodou odlišný od obrazu generovaného s přesnými normály. Za tímto účelem jsem zavedl porovnávání jasových úrovní v obrazu (přepočítání RGB na grayscale jsem prováděl podle [6]) a zjišťování průměrné odchylky. Tato metoda se ukázala jako funkční, nicméně nevyhovující původnímu záměru, protože žádným způsobem nezohledňuje přítomnost „hran“<sup>1</sup> v obrazech, které působí velmi rušivě. Proto její výsledky zatím nedokumentuji, některá čísla mohou být nalezena v prezentaci doprovázející tento dokument.

www adresa projektu: <http://home.zcu.cz/~lvasa/apg>

<sup>1</sup> Tyto hrany jsou zvláště patrné u metod vážení plochou, naopak téměř mizí u inverzního vážení plochou.



### **Použitá literatura:**

- [1] **Gouraud, H.:** *Continuous Shading of Curved Surfaces*, IEEE Transactions on Computers, Vol. 20, No. 6, pp. 623-629, June 1971
- [2] **Hill, F. S.:** *Computer graphics usány OpenGL*, Upper Saddle River: Prentice Hall, 2001
- [3] **Jirka, T.:** *Vertex Normal Computation*, research paper, zatím nevydán
- [4] **Max, N.:** *Weights for computing verte normals from facet normals*, Journal of Graphics Tools, 4(2):1-6, 1999
- [5] **Rektorys, K.:** *Přehled užité matematiky*, SNTL 1968
- [6] **Žára, J.:** *Moderní počítačová grafika*, Computer Press, 1998