

Manuál k aplikácií a knižnici cheetah

Peter Mitura

Cheetah je aplikácia a knižnica umožňujúca efektívne riešiť problém konvexnej obálky. Tento manuál popisuje všetky možnosti aplikácie, funkcie knižnice a postup ich inštalácie.

Obsah

1	Inštalácia	2
1.1	Požiadavky	2
1.2	Kompilácia	2
2	Aplikácia	2
2.1	Formát vstupu	2
2.2	Formát výstupu	3
2.3	Základné použitie	3
2.4	Vstup zo súboru	3
2.5	Výstup do súboru	3
2.6	Meranie času	3
2.7	Použitie konkrétneho algoritmu	4
2.8	Voľba dimenzie	4
2.9	Paralelizácia	4
2.10	Test výkonu	4
3	Knižnica	5
3.1	Použitie	5

1 Inštalácia

Popis inštalčných procedúr, potrebných k sprevádzkovaniu projektu *Cheetah* na vašom počítači.

1.1 Požiadavky

Projekt je určený pre systém GNU/Linux, prípadne iné unix-like systémy. Pre inštaláciu je ďalej potrebné, aby ste na systéme mali nainštalované nasledovné komponenty:

- **GCC** verzia 5.0 alebo vyššia, s podporou OpenMP (<https://gcc.gnu.org/>)
- **GNU ar** (<https://sourceware.org/binutils/docs/binutils/ar.html>)
- **GNU Make** (<https://www.gnu.org/software/make/>)
- (*voliteľne*) **Google Test** (<https://github.com/google/googletest>)

Google Test je potrebný len k sprevádzkovaniu modulu jednotkových testov, aplikácia aj knižnica môže fungovať aj bez neho.

1.2 Kompilácia

Kompilácia je automatizovaná nástrojom GNU Make, na výber sú tri direktívy, ktorými sa kompilujú rôzne moduly projektu:

- **make**, **make app** – Aplikácia pre príkazový riadok, vytvorený je spustiteľný súbor **cheetah** v adresári **bin**
- **make lib** – Knižnica, do adresára **bin** je skompilovaný súbor **libcheetah.a** a vytvorený adresár **include**, obsahujúci príslušné hlavičkové súbory.
- **make test** – Jednotkové testy, po kompilácii sú automaticky aj spustené a vyhodnotené.

Upozorňujeme, že každý z týchto modulov používa zdieľané objektové súbory v adresári **build**. Je preto nutné pri každej zmene modulu zavolať príkaz **make clean**, ktorý vráti projekt do počiatočného stavu.

2 Aplikácia

Aplikácia sa po kompilácii nachádza v adresári **bin** pod názvom **cheetah**, je možné ju ďalej skopírovať do jedného z adresárov definovaných v premennej prostredia **PATH** a používať z ľubovoľného miesta v systéme.

2.1 Formát vstupu

Program prijíma textový vstup, začínajúci prirodzeným číslom n , označujúcim počet vstupných bodov, za ktorým nasleduje n dvojíc alebo trojíc reálnych čísel, reprezentujúcich súradnice bodov v \mathbb{R}^2 resp. \mathbb{R}^3 . Čísla sú oddelené bielymi znakmi. Príklad vstupu v \mathbb{R}^2 :

```
4
0 10
-10 0
9.8002 0
0 1
```

2.2 Formát výstupu

Výstup v \mathbb{R}^2 má rovnakú podobu, ako vstup. Body v ňom sú vypísané v poradí, v ktorom sa objavujú na obálke, teda proti smeru hodinových ručičiek.

Výstup v \mathbb{R}^3 je odlišný, začína sa prirodzeným číslom n , označujúcim počet stien. Za ním nasleduje popis n stien, pri čom popis jednej steny sa skladá z prirodzeného čísla m a po ňom m trojíc reálnych čísel, predstavujúcich súradnice vrcholov na danej stene. Vrcholy v rámci jednej steny sú zoradené proti smeru hodinových ručičiek

2.3 Základné použitie

Bez špecifikovaných prepínačov aplikácia hľadá konvexnné obálky v R^2 s použitím algoritmu Quickhull. Vstupné dáta sú očakávané na štandardnom vstupe, výstup je vypísaný na štandardný výstup. Ak by sme vyššie uvedený príklad vstupu uložili do súboru `data.in`, použitie by mohlo vyzeráť nasledovne:

```
$ bin/cheetah < data.in
3
9.8002 0
-10 0
0 10
```

Funkčnosť je ďalej možné rozšíriť prepínačmi, ktoré popíšeme v nasledujúcich sekciách.

2.4 Vstup zo súboru

Prepínač `-i <názov súboru>` umožní načítať vstup z určeného súboru. Príklad použitia:

```
$ bin/cheetah -i data.in
3
9.8002 0
-10 0
0 10
```

2.5 Výstup do súboru

Prepínač `-o <názov súboru>` presmeruje výstup zo štandardného výstupu do určeného súboru. Príklad použitia:

```
$ bin/cheetah -i data.in -o data.out
$ cat data.out
3
9.8002 0
-10 0
0 10
```

2.6 Meranie času

Prepínač `-t` na štandardný výstup čas, ktorý zaberie nájdenie obálky. Do času sa nezarátava doba potrebná k načítaniu vstupu, alebo vypísaniu výstupu. Príklad použitia:

```
$ bin/cheetah -i data.in -o data.out -t
Execution time: 0.000160927 ms.
```

2.7 Použitie konkrétneho algoritmu

Štandardným algoritmom pre riešenie je Quickhull. Na zvolenie iného algoritmu pre riešenie je možné použiť prepínač `-s <názov algoritmu>`. Dostupné voľby v \mathbb{R}^2 a ich asymptotické zložitosti sú:

- `jarvis` – Jarvis March ($\mathcal{O}(nh)$)
- `graham` – Graham Scan ($\mathcal{O}(n \log n)$)
- `quickhull` – Quickhull ($\mathcal{O}(nh)$ najhorší prípad, $\mathcal{O}(n \log n)$ v priemere)
- `chan` – Chanov algoritmus ($\mathcal{O}(n \log h)$)
- `andrew` – (*experimentálne*) Andrewov algoritmus ($\mathcal{O}(n \log n)$)

Pre \mathbb{R}^3 je dostupný len algoritmus Jarvis March, ktorý je použitý aj v základe.

2.8 Voľba dimenzie

V základe program používa rovinné algoritmy, dimenziu vstupu je možné špecifikovať prepínačom `-d <dimenzia>`. Súčasne podporované hodnoty sú 2 a 3.

2.9 Paralelizácia

Aplikácia štandardne púšťa sekvenčné verzie algoritmov, paralelizáciu je možné zapnúť prepínačom `-m <počet vlákien>`. Funguje len v \mathbb{R}^2 a maximálny počet vlákien je z dôvodu ochrany stability systému limitovaný na 24. Príklad použitia:

```
$ bin/cheetah -t -o /dev/null -i bigdata.in -s graham
Execution time: 0.700449 ms.
$ bin/cheetah -t -o /dev/null -i bigdata.in -s graham -m 2
Execution time: 0.467592 ms.
```

2.10 Test výkonu

Výkon je mimo prepínača `-t` možné otestovať aj priloženým generátorom dát, ktorý dokáže vytvoriť vstup s daným počtom bodov vo vstupnej množine a na obálke. Test sa spúšťa prepínačom `-p <n> <h> <s> <r> <t>`, kde n je počet bodov vo vstupnej množine ($n \in \mathbb{N}, n < 2^{31}$), h počet bodov na obálke ($h \in \mathbb{N}, h \leq n$), s určuje rozsah súradníc bodov $[-s, s]$ ($s \in \mathbb{N}$), r počet zopakovaní testu ($r \in \mathbb{N}$) (čas sa ráta ako súčet všetkých behov) a t počet vlákien ($t \in \mathbb{N}, 1 \leq t \leq 24$).

Upozorňujeme, že pri veľkom množstve bodov na obálke (typicky nad $5 \cdot 10^5$) sa začne prejavovať nepresnosť dátového typu `double` a na nájdených obálkach sa začne objavovať nižší počet bodov ako bol zadaný, čo môže ovplyvniť aj časy výpočtu. Testovač na túto skutočnosť upozorní varovaním. Príklad:

```
$ bin/cheetah -p 10000000 10 1000 1 1
Generating test instance...
...done, running test
time: 0.586616 ms
$ bin/cheetah -p 10000000 10000 1000 1 1
Generating test instance...
...done, running test
time: 1.82755 ms
$ bin/cheetah -p 10000000 1000000 1000 1 1
Generating test instance...
...done, running test
[WARNING] Precision errors occurred (h may be too high)
time: 2.60803 ms
```

3 Knižnica

Modul knižnice umožňuje pridať implementované algoritmy priamo do programov napísaných v jazyku C++. Oproti aplikácií v nej chýbajú výskumne založené vymoženosti ako testovanie rýchlosti, obsahuje však plné možnosti ohľadne výberu algoritmov a paralelizácie.

3.1 Použitie

Knižnica je kompilovaná pre statické linkovanie s cieľovou aplikáciou. Pri kompilácii výsledku je nutné použiť minimálne prepínače `-std=c++11` a `-fopenmp`. Ďalej je nutné direktívami `-I` a `-L` špecifikovať cestu k hlavičkovým súborom a knižnici a prepínačom `-lcheetah` prilinkovať knižnicu.