



Eötvös Loránd Tudományegyetem

Informatikai Kar

Programozási Nyelvek és Fordítóprog-
ramok Tanszék

Interpoláció osztott rendszereken

Tejfel Máté
egyetemi tanár

Cselyuszká Alexandra
Informatika Bsc

ABCDEF GHIJKLM NOPQRSTUW VXYZ

Budapest, 2015

Tartalomjegyzék

1. Bevezetés	1
2. Felhasználói dokumentáció	3
2.1. Weboldal	3
3. Fejlesztői dokumentáció	4
3.1. Weboldal	4
3.2. Elosztott rendszer	4
3.2.1. Adat feldolgozás	4
3.3. Kalkulátor	4
3.4. Kommunikáció	5
3.4.1. Kalkulátor és az Elosztott rendszer közötti kommunikáció	5
4. Források	6

1. fejezet

Bevezetés

Ez a szoftver interpolációkat számít osztott környezetben.

Adott ponthalmazokból kívánunk egy közelítő polinómot becsülni, mellyel a többi pont várható elhelyezkedését is megadhatjuk. Ezeket különböző Interpolációs technikával meg tudjuk adni, ki tudjuk számolni. Viszont honnan tudjuk melyik a legjobb közelítő érték? Több Interpolációs technika létezik, melyekből könnyen meg tudunk adni akár több polinómot is egy adott ponthalmazhoz. De ezek a számítások előfordulhat hogy lassan futnak, főleg ha több Interpolációt kívánunk egyszerre számolni. Ebben az esetben optimálisabb több gépen számolni a különböző ponthalmazokat.

Egy grafikonon látható pontokat közelítünk meg egy polinómmal. Fel tudunk venni saját halmazokat, pontokat, melyekre kíváncsiak vagyunk. Vagy csak úgy megjelölni különböző pontokat, és megbecsülni hányad fokú lesz az eredmény polinóm. Elég látványos képet ad amikor az ember megkap egy 10ed- vagy 20ad fokú polinómot, melyet már kézzel számolni nem lenne olyan egyszerű, de akár ellenőrizni is az általánuk kalkulált eredményt.

Egy Web oldalon szereplő grafikon kirajzolásának mintáját, egy egyszerű elosztott számítást és szerver megvalósítást, és egy egyszerű Interpoláció számítást megvalósító részletekből össze rakott programot tekinthetünk meg, mely végül az alábbi feladatokat valósította meg.

A program egy meghatározott adatforrásból számít, amely tartalmazza nagy mennyiségű interpolálni kívánt függvény ismert értékeit. A számításokat több gépen végezzük elosztott módon.

Az kiindulási adatokat egy grafikus felület segítségével lehet megadni, illetve szerkeszteni grafikon, illetve táblázat segítségével. A felhasználó grafikonon kattintva fel tudja venni az adott pontot, és a táblázatot is tudja szerkeszteni.

A felhasználó, ha befejezte az adathalmazok szerkesztését, elindíthatja az interpolációt. Ekkor a szerverre felküldi az adatokat, ahol történik az elosztott, vagy párhuzamosított számítás. Eredményt vissza kapjuk a szervertől. A felhasználó kívá-

laszthatja az egyik pont halmazt és ekkor már az eredménnyel együtt megtekintheti azt. Az eredményt egyszerű felírt formában és a grafikonon megjelenítve is megtekintheti.

A felhasználók több Inteproláció közül is választhatnak: Lagrange-féle, Newton-féle, Hermite-féle interpolációs technikák. Valamint Interz interpolációt is választhatnak a Lagrange vagy a Newton interpoláció esetén.

A számításhoz használt maximális gépek száma paraméterként megadható, de a tényleges számítást csak annyi gépen tudjuk maximálisan végezni ahány gép felcsatlakozott a számításhoz.

2. fejezet

Felhasználói dokumentáció

A felhasználó megnyitja a weboldalt, megtekintheti a táblázatokat és a grafikonokat. Minden adatot szerkeszthet és új adathalmazt hozhat létre.

2.1. Weboldal

Weboldalon kattintgat majd szép eredményeket kap, pontokat lát aztán ha kiszámolja az eredményt még szebb polinómot kap eredményül ha sikeres volt a számítás.

3. fejezet

Fejlesztői dokumentáció

A program 3 fő részből áll a Weboldalból, az Elosztott rendszerből és a Kalkulátorból.

3.1. Weboldal

Weboldal felépítése HTML és JavaScript segítségével valósult meg. Egy oldalból áll melyen a felhasználó össze állítja a neki szükséges adathalmazt. Új adathalmazokat hozhat létre, a régieket szerkesztheti. A háttérben JSON-be formálódnak az adatok, melyeket a felhasználó is láthat, ha debug-módban lép be. Ha a felhasználó végzett egy gombra nyomással a program legenerálja a szükséges JSON-t.

3.2. Elosztott rendszer

Elosztott rendszer Erlang-ban lett megvalósítva. Az elosztást Interpolációnként végezzük, vagyis annyi node-ot hozunk létre amennyi Interpolációt kívánunk egyszerre kiszámítani.

3.2.1. Adat feldolgozás

Az elosztott rendszer először kap egy JSON adathalmazt melyből kinyeri a neki szükséges adatokat, és átkonvertálja.

3.3. Kalkulátor

A Kalkulátor részben számítódik ki egy-egy Interpolációnak az eredménye. A megkapott adatok alapján számol, ha kell létre hozza a kezdő mátrixot, kiszámolja az eredmény mátrixot, majd annak segítségével kiszámolja a polinómot.

```
DArray interpolateMain (  
    DArray &x, DMatrix &Y,  
    string type = "lagrange", bool inverse = false );
```

Kívülről meghívandó fő függvény mely elosztja és konvertálja a részeket.

DArray &x

Az x pontok listája

DMatrix &Y

Az x pontokhoz tartozó y pontok halmaza

string type

Interpoláció típusa: lagrange, newton, hermite

bool inverse

Inverz Interpoláció kell-e

3.4. Kommunikáció

A 3 különállóan megvalósított program részlet speciális módon kommunikál egymással.

3.4.1. Kalkulátor és az Elosztott rendszer közötti kommunikáció

Az elosztott rendszerben hívódó számítást Erlang - erl_nif"-el sikerült megoldanom. Az ezzel kapcsolatos dolgokat az Calculator/erlang.cpp tartalmazza.

4. fejezet

Források

- http://www.erlang.org/doc/man/erl_nif.html
- https://www.sharelatex.com/learn/Sections_and_chapters
- <https://github.com/mochi/mochiweb/blob/master/src/mochijson.erl>
- <http://tex.stackexchange.com/questions/137055/lstlisting-syntax-highlighting-for-c-like-in-editor>