

Stručný návod k použití balíku ImplDGM ¹

Vít Dolejší
dolejsi@karlin.mff.cuni.cz

1 Základní charakteristika

Balík ImplDGM slouží k numerickému řešení soustav Eulerových rovnic, které popisují proudění stlačitelné neviskózní tekutiny. Rovnice se řeší na konformních trojúhelníkových sítích, diskretizace vzhledem k prostorovým souřadnicím se provádí pomocí nespojitých Galerkinových metod (DGM) a diskretizace vzhledem k časové souřadnici pomocí BDF metod za užití jisté linearizace (máme tedy semi-implicitní schéma). V balíku je naimplementována po částech lineární nespojitá aproximace DGM a BDF metoda 1. a 2. řádu s konstantním časovým krokem. S tímto programem (resp. se starsími verzemi) byly spočteny výsledky prezentované v [1]. V této práci lze nalézt odvození metod a i poznámky k vlastní implementaci. Celý program je napsán v jazyce C ke kterému je přilinkován řešič soustav lineárních rovnic napsaný ve Fortranu 77. Balík byl používán pod operačními systémy Linux a Unix, a použitím pod Windows zkušenost nemám.

2 Kompilace

Ke stažení je soubor 'ImplDGM.tar', který se pod operačními systémy Linux či Unix rozbíjí příkazem

```
tar xf ImplDGM.tar
```

Po rozbalení se vytvoří adresář ImplDGM/, který obsahuje tři podadresáře

```
data/  
doc/  
src/
```

Adresář **data/** obsahuje příklad datových souborů potřebných pro běh programu (jedná se o tzv. GAMM kanál), adresář **doc/** obsahuje stručný návod a adresář **src/** obsahuje zdrojové soubory '*.c', hlavičkové soubory '*.h', soubory '*.f', 'params.for' (zde se zadává dimenze polí pro řešič soustav lineárních rovnic, v případě potřeby je nutné jej upravit) a soubor 'makefile'. Kvůli snazšímu slinkování programu v jazyce C a F77 je hlavní program v souboru 'Frame.f', který obsahuje pouze jeden příkaz, kterým je volání programu v C. (V balíku se mohou vyskytovat i soubory, které se nepoužívají, jedná se o vývojové verze.)

Kompilace programu se provádí příkazem

¹Tento návod je určen především studentům, kteří pracují s tímto balíkem. Jedná se o mladší produkt než-li u ExplDGM a řada věcí je podobná, na odlišnosti je speciálně upozorněno.

make

v adresari `src/`. Muze se stat, ze pri prekladu da prekladac hlasku

ld:

Unresolved:

main_c_

Pak v souboru 'Dgm.c' je potreba nahradit radek

```
int main_c_(){          /* int main_c_(){ or int main_c_(){ */
```

radkem

```
int main_c_(){          /* int main_c_(){ or int main_c_(){ */
```

jak je naznaceno v komentari tohoto radku. Prvni varianta mi chodi pod operacnimi systemy Linux, druha varianta pak zase na Unixovskych strojich v Karline.

Pri uspesnem prekladu by vysledkem mely byt dva programy

- Dgm, který slouží k vlastnímu výpočtu,
- Plotdgm, který slouží k vizualizaci výsledku.

3 Pouziti programu Dgm

Program Dgm nacita vstupni data z datoveho souboru, priklad datoveho souboru '0.ini' je v adresari `data/`, jeho format je skoro tozny jako u `ExplDgm`, rozdily jsou uvedeny tucne.

* input .grid file

triang

* Poisson constant kappa

1.4

* initial condition (1) or .sol file (2)

* 1 rho v1 v2 v3 p kappa

* 2 file

1 1. 1.0 0 1.5911912

*2 0.rsol

* output .sol file

0.sol

* convergence file (- to suppress)

0.conv

* statistics file (- to suppress)

```

0.stat

* stopping criterion: Linfty norm (- to suppress)
1e-5

* stopping criterion: physical time (- to suppress)
2e+01

* numbered solution output physical time step (- to suppress)
-

* integrate proc (implicit/implicit2)
*implicit
implicit2

* local time stepping (G/L)
*local
global

* toleracne for limiting
1.0

*Reynolds and Prandtl number for viscous flow, Re=0 ==> inviscid
0E+05    0.72

* inlet/outlet count
2

* for each inlet/outlet: surfnr physical/fixed rho v1 v2 v3 p
2 physical 1. 1.0 0 1.5911912    ****  M=0.67
1 physical 1. 1.0 0 1.5911912

*2 physical 1. 1.0 0 2.857142857    *****  M=0.5
*1 physical 1. 1.0 0 2.857142857

* CFL number for time step restriction
10.0

```

3.1 Popis datoveho souboru '0.ini' z adresare data

Vsechny znaky '*' a znaky za '*' az do konce radku, program ingoruje. Lze je tedy pouzit jako komentare, ci "schovani" zatim nepotrebnych dat. Nyni k jednotlivym parametrum uvadeneho souboru

- **triang**: jmeno souboru, kde je ulozena triangulace dane oblasti, na ktere se bude provadet vypocet. Format je shodny jako u baliku ANGENER (viz <http://www.karlin.mff.cuni.cz/~dolejsi/angen/angen.htm>) s drobnymi zmenami (2. radek), viz prilozeny priklad v adresari **data/**.
- **1.4**: hodnota Poissonovy konstanty, 1.4 je bezne uzivana hodnota pro vzduch.
- *typ pocatecni podminky*: V pripade, ze vypocet zacina, tak je na prvni miste 1 a pak ctverice realnych cisel znacich konstatni hodnoty hustoty, slozek rychlosti a tlaku. V pripade, ze vypocet pokracuje, tak je na prvni miste 2 a pak jmeno souboru, odkud se pocatecni podminky nacitaji (viz zakomentovany radek v souboru '0.ini'). Format tohoto vstupniho souboru je shodny s souborem vystupnich dat, viz dalsi bod.
- **0.sol**: jmeno souboru, kam se ukladaji vysledna data. Je to soubor realnych cisel se Ctyrmi sloupci (hustota, dve slozky hybnosti a energie). Pocet radku souboru je pocet trojuhelniku site krat 3. Kazdemu elementu odpovidaji 3 radky (nespojita, po castech linearni aproximace dava 3 stupne volnosti na kazdem elementu), kazdemu radku odpovidaji hodnoty ve **vrcholech trojuhelnika** v poradi danem souborem 'triang'. (Rozdil oproti baliku ExplDGM, který uziva hodnoty ve stredech stran trojuhelnika.)
- **0.conv**: jmeno souboru, kam se zapisuje historie konvergence ke stacionarnimu stavu (pokud existuje). Tento soubor je primo zobrazitelny v gnuplotu. Jeho vypisovani lze potlacit znakovym ' '.
- **0.stat**: jmeno souboru, kam se zapisuji udaje o celkovem poctu iteraci, CPU casu, pouzite pameti apod. Jeho vypisovani lze potlacit znakovym ' '.
- **1e-5**: hodnota stopping kriteria pro reziduum. Pokud reziduum (merici konvergenci ke stacionarnimu stavu) je mensi nez tato hodnota, tak se vypocet zastavi.
- **2e+01**: hodnota fyzikalniho casu, pri ktere se ma vypocet zastavit. Vypocet se zastavi, je-li splnena alespon jedna z uvedenych podminek.
- *numbered solution*: Program pravidelne uklada spoctena dat (priblizne kazdych 5 minut) do souboru, který je uveden vyse jako 4. parametr, tzn. ze napr. soubor '0.sol' se pravidelne prepisuje. Chceme-li si tyto soubory uchovavat (napr. pro ilustraci nekterych nestacionarnich jevu, rychlost ci zpusob konvergence), tak je potreba uvest casovy interval (fyzikalni cas, ne CPU), pri kterem se maji data uchovavat. Je-li uvedeno napr. 0.5 a jmeno vystupniho souboru je '0.sol', tak pak vznikaji soubory '0.sol0', '0.sol0.5', '0.sol1', '0.sol1.5', atd.
- *integracni procedura*: zpusob diskretizace vzhledem k casu, **dve moznosti**, **implicit** (1.rad), **implicit2** (2. rad).
- *zpusob volby casoveho kroku*: **pouze moznost global** ma smysl.
- **1.0**: tolerance pro limiting, pokud reseni je nespojite (razove vlny), tak numericke reseni obsahuje oscilace, ty se potlacuji limitingem, viz [2]. Hodnotu 1.0 je doporučeno prilis nemenit.

- 0E+04 0.72: vzhledem k tomu, že program řeší **zasím pouze** nevazke proudění, tak tyto dvě hodnoty nemají vliv na běh programu.
- *inlet/outlet count*: počet částí hranice, kde předepisujeme vstupní či výstupní okrajové podmínky. Ostatních částí hranice se automaticky berou jako neprostupné stěny. Části hranice jsou definovány souborem 'triang'.
- *okrajové podmínky*: pro každou z částí hranice, kde předepisujeme vstupní či výstupní okrajové podmínky se uvede napřed její číslo (*ibc* ze souboru s 'triang'), pak typ OP (**fixed** nebo **physical** - doporučeno) a pak zase 4 reálná čísla představující hustotu, složky rychlosti a tlak na vstupní či výstupní stěně. Počet těchto řádků musí být shodný s *inlet/outlet count*, viz předeslý bod.
- *CFL číslo*: tato metoda je teoreticky **nepodmíněně stabilní**, byly prováděny testy až s CFL=150. Nicméně dlouhý časový krok vede k nepřesnému řešení, závisí to pak na problému. Navíc, ne vždy nutně delší časový krok vede k rychlejší konvergenci, jelikož pak trvá déle řešení soustavy lineárních rovnic. Na začátku si program automaticky upravuje CFL konstantu na 1 a pak ji postupně zvětšuje až na uživatelem zadane číslo. V případě, že výpočet zhavaruje, tak lze doporučit výpočet s menším CFL.

3.2 Beh programu Dgm

Pokud kompilace byla v pořádku, tak program Dgm lze spustit dvěma způsoby

- bez parametru, ale pak se hned ptá na jméno inicializačního souboru:

```
../scr/Dgm
Reading file
```

který se pak musí interaktivně zadat, např. '0.ini' z příkladu v **data/**

- nebo lze spustit neinteraktivně přesměrováním vstupu pomocí OS, t.j. např.

```
../scr/Dgm < Ini0
```

kde 'Ini0' je soubor, který obsahuje jméno souboru, který se užije jako datový. Viz příklad v **data/**, kde soubor 'Ini0' obsahuje řetězec '0.ini'.

Pro úspěšný běh programu Dgm musí v adresáři, odkud se program spouští, být soubor (viz příklad v adresáři **data**) 'profiles', který popisuje křivou část hranice. V případě polygonální hranice soubor 'profiles' obsahuje pouze jednu celociselnou '0'. Formát souboru 'profiles' je stejný jako u balíku ANGENER.

Při běhu programu se napřed vypisou nějaké komentáře ke vstupním datům a pak postupně každá 25. iterace. Tento výpis obsahuje číslo iterace, celkový fyzikální čas, reziduum (rozdíl hustoty ve dvou následujících časových krocích v L^1 -normě dělený časovým krokem – tímto se měří konvergence ke stacionárnímu stavu), maximální a minimální hodnoty Machova čísla v celé oblasti, počet elementů, kde se prováděl limiting, počet provedených **iterací** v jednom časovém kroku k vyřešení soustavy lineárních rovnic a **reziduum**, které bylo při řešení této soustavy dosaženo. Běh programu na příložených datech by měl dát přibližně následující výpis:

```

Reading file
Reading file 0.ini
Source grid file: triang
438 points, 784 volume elements, 90 surface elements
Initial condition: rho 1, v 1,0, p 1.5911912, kappa 1.4
Output solution file: 0.sol
Convergence file: 0.conv
Statistics file: 0.stat
Stopping criterion: L^infty norm: 1e-05
Stopping criterion: Physical time: 20 s
Implicit numerical scheme, 2nd order in time
Global time stepping
Tolerance for limiting:      1
Inlet/outlet sides: 2
Side 2: physical, rho 1, v 1,0, p 1.5911912
Side 1: physical, rho 1, v 1,0, p 1.5911912
Constant for CFL condition: 10
The 16 nodes detected on curved part of boundary
  Begin of Linear_Mapping_Computing_Vertex
  END of Linear_Mapping_Computing_Vertex
Solver_Init finished

```

Iter	Time	Reziduum	Mach: minimal	maximal	limit	iter	rez
0	0.0011	7.6399e-01	0.6584	0.6797	0	24	6e-13
25	0.0293	5.1096e-01	0.3701	0.7992	0	20	7e-13
50	0.0589	3.5785e-01	0.3474	0.7794	0	17	8e-13
75	0.0900	3.7311e-01	0.3419	0.8105	0	16	4e-13
100	0.1227	4.0045e-01	0.3427	0.8620	0	14	9e-13
125	0.1570	4.1844e-01	0.3394	0.9256	0	14	9e-13
150	0.1929	4.5421e-01	0.3365	0.9733	0	13	8e-13
175	0.2306	4.6192e-01	0.3331	1.0038	0	13	6e-13

.
.
.

Po uspesnem probehnutí programu Dgm (na dobrem pocitaci trva prilozeny priklad zhruba nekolik minut), vzniknou soubory uvedene v souboru '0.ini', v nasem pripade '0.sol', '0.conv', '0.stat' a soubor 'walls', který mel kdysi pomocny charakter (rozlozeni hustoty podel sten).

Upozorneni! Muze se stat a stava se, ze pri nekterych vypoctech dojde na nejakyh elementech k (nefyzikalnimu) zapornemu tlaku. Program tyto hodnoty "opravuje" (dava tam maly kladny tlak) a dava o tom hlasku, jaky tlak a v jakem miste oblasti vzniknul. Simulace je pak dosti nefyzikalni, ale v pomerne castem procentu pripadu se reseni zase ustali a vypocet je jiz v

poradku. Odstranění těchto problémů je součástí dalšího výzkumu.

4 Použití programu Plotdgm

Pro vizualizaci spočtených výsledků se užívá program `Plotdgm`, který je téměř identický s tím z `ExplDGM`, ale nelze je kombinovat, jelikož soubor s výsledky `'0.sol'` má jiné brány hodnoty řešení, viz komentář k datovému souboru `'0.ini'`. `Plotdgm` se spouští se dvěma parametry, prvním je soubor s triangulací a druhým soubor s výsledky spočtenými na této triangulaci, tj. v našem případě

```
../src/Plotdgm triang 0.sol
```

V adresáři, odkud se program `Plotdgm` spouští musí být dále obsazen soubor `'plot.dgm'` (viz příklad v adresáři `data/`, kam si uživatel zadá veličiny a typy zobrazení, která chce vykreslit. Příkladem je soubor z adresáře `data/`:

```
M WALL 2 3 4
RO WALL 2 3 4
P WALL 2 1 2
M ISO 81 1 0. 2.
RO CUT -1. 0.2 1. 0.2
EOD
```

- První znak(y) značí veličinu, která se má zobrazit, jsou následující možnosti: RO – hustota, RV – velikost momentu síly, V – velikost rychlosti, V1, V2 – složky rychlosti, P – tlak, PC – tlakový koeficient, M – Machovo číslo, E – energie, T – teplota, CF – skin friction coefficient. EOD znamená konec datového souboru, což je za ním se již nenachází.
- Druhý symbol znamená vykreslovaný typ:
 - WALL : vykreslí rozložení na stěny (stenách), napřed se uvádí počet stěn, na kterých se má současně vykreslovat a pak `ibc` těchto stěn (ze souboru `'triang'`). Defaultově je nastaveno, že u stěn s `ibc` rovno 1 nebo 2 (obvykle vstup nebo výstup) se vykresluje závislost veličiny na x_2 u ostatních na x_1 (x_1, x_2 jsou kartézské souřadnice).
 - ISO : vykreslí isokřivky, uvede se počet (např. 81) a pak se uvede buď 0, pak se isokřivky vykreslí od maxima do minima dané veličiny, nebo se uvede 1 a minimální a maximální hodnota, které se mají zobrazit. Isokřivky jsou vždy rozloženy ekvidistantně.
 - CUT : vykreslí se řez dané veličiny podél osy danou dvěma body, jejich souřadnice se uvedou za znaky CUT, např. v zde uvedeném příkladu se vykreslí řez podél osy $x_2 = 0.2$. Vzniklý soubor obsahuje tři sloupce, souřadnice x_1 , spočtenou hodnotu na řezu a souřadnice x_2 .

Výsledkem běhu programu `Plotdgm` jsou soubory `'gnu.01'`, `'gnu.02'`, `'gnu.03'`, ..., podle pořadí zadání vizualizací v souboru `'plot.dgm'`. Jejich počet by měl odpovídat počtu řádků v `'plt.dgm'` před EOD. Pokud již v adresáři nějaké soubory `'gnu.*'` existovaly, tak se automaticky prepíší. Všechny tyto soubory jsou snadno zobrazitelné v `gnuplotu`, např. příkazem

```
gnuplot> plot 'gnu.01' w l
```

pri 3D vizualizaci isokrivek lze uzit

```
gnuplot> splot 'gnu.03' w l
```

pri zobrazení rezu v závislosti na x_2 lze uzit

```
gnuplot> plot 'gnu.05' using 3:2 w l
```

(viz program gnuplot a jeho napoveda).

5 Zaverena poznámka

Balik ImplDGM predstavuje pouze jakousi vyvojovou verzi, bude zrejme obsahovat chyby a není ani optimalne napsan. Take dokumentace je velmi stroha, ale pro zacatek by mela stacit.

References

- [1] V. Dolejší and M. Feistauer. Semi-implicit discontinuous galerkin finite element method for the numerical solution of inviscid compressible flow. *J. Comput. Phys.*, 198(2):727–746, 2004.
- [2] V. Dolejší, M. Feistauer, and C. Schwab. On some aspects of the discontinuous Galerkin finite element method for conservation laws. *Math. Comput. Simul.*, 61:333–346, 2003.