

# Stručný návod k použití balíku ExplDGM <sup>1</sup>

Vít Dolejší  
dolejsi@karlin.mff.cuni.cz

## 1 Základní charakteristika

Balík ExplDGM slouží k numerickému řešení soustav Eulerových (Navierových-Stokesových) rovnic, které popisují proudění stlačitelné neviskózní (vazké) 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 Runge-Kuttových metod (máme tedy explicitní schéma). V balíku je naimplementována po částech lineární nespojitá aproximace DGM a Runge-Kuttových metod 1., 2. a 4. řádu. Testy byly prováděny především s metodou 1. řádu (dopředná Eulerova metoda). S tímto programem (resp. se starsími verzemi) byly spočteny výsledky prezentované např. v [1], [2], [3], [4]. V těchto pracích lze nalézt odvození metod a i poznámky k vlastní implementaci. Celý program je napsán v jazyce C a byl používán pod operačními systémy Linux a Unix. S použitím pod Windows zkušenost nemám.

## 2 Kompilace

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

```
tar xf ExplDGM.tar
```

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

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

Adresář **data/** obsahuje příklady 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' a soubor 'makefile'. (V balíku se mohou vyskytovat i soubory, které se nepoužívají, jedná se o vývojovou verzi.)

Kompilace programu se provádí příkazem

```
make
```

v adresáři **src/**. Při úspěšném překladu by výsledkem měly být dva programy

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

---

<sup>1</sup>Tento návod je určen především studentům, kteří pracují s tímto balíkem

### 3 Pouziti programu Dgm

Program Dgm se spousti s jedním parametrem, kterým je jméno datového souboru, který obsahuje parametry pro výpočet, tedy napr.

Dgm 0.ini

Příklad datového souboru '0.ini' je v adresáři **data**, který vypadá následovně:

```
* 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 (euler/ec/rk4)
euler

* local time stepping (global/local)
local
* global
```

```

* toleracne for limiting
1.0

*Reynolds and Prandtl number for viscous flow, Re=0 ==> inviscid
0E+04    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

* CFL number for time step restriction
0.60

```

### 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 znacicich 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 **stredu hrany trojuhelnika** v poradi danem souborem 'triang'. (Rozdil oproti baliku ImplDGM uzivajici hodnoty ve vrcholech 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 znakem '-'.
- **0.stat**: jmeno souboru, kam se zapisuji udaje o celkovem poctu iteraci, CPU casu, pouzite pameti apod. Jeho vypisovani lze potlacit znakem '-'.

- **1e-5**: hodnota stopping kriteria pro reziduum. Pokud reziduum (merici konvergence 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, ktery 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, tri moznosti, **euler** (1.rad), **ec** (2. rad, Euler-Cauchyho formula) a **rk4** (Runge-Kuttova metoda 4. radu).
- *zpusob volby casoveho kroku*: bud **global** (globalni) nebo **local** (lokalni). Lokalni znamena, ze na kazdem elementu site se voli jina delka casoveho kroku, pak sice mame nefyzikalni simulaci a hodnota o fyzikalnim case nema moc smysl, ale kdyz hledame stacionarni stav, konvergence na hodne zjemnenych siti je mnohem rychlejsi nez-li s pouzitim **global**.
- **1.0**: tolerance pro limiting, pokud reseni je nespojite (razove vlny), tak numericke reseni obsahuje oscilace, ty se potlacuji limitingem, viz [4]. Hodnotu 1.0 je doporuveno prilis nemenit.
- **0E+04 0.72**: hodnoty Reynoldsova a Prandtlova cisla, pro nevazke proudeni se klade prvni hodnota rovna 0 a program automaticky resi pouze Eulerovy rovnice. Hodnota Prandtlova cisla 0.72 je hodnotou pro vzduch, hodnota Reynoldsova cisla je dana rezimem proudeni.
- *inlet/outlet count*: pocet casti hranice, kde predepisujeme vstupni ci vystupni okrajove podminky. Ostatnich casti hranice se automaticky berou jako neprostupne steny. Casti hranice jsou definovany souborem 'triang'.
- *okrajove podminky*: pro kazdou z casti hranice, kde predepisujeme vstupni ci vystupni okrajove podminky se uvede napred jeji cislo (**ibc** ze souboru s 'triang'), pak typ OP (**fixed** nebo **physical** - doporuveno) a pak zase 4 realna cisla predstavujici hustotu, slozky rychlosti a tlak na vstupni ci vystupni stene. Pocet techto radku musi byt shodny s *inlet/outlet count*, viz predesly bod.
- *CFL cislo*: tato metoda je podminene stabilni, a proto predepisujeme konstantu omezujici velikost casoveho kroku. Tato hodnota musi byt z intervalu (0,1), doporuveno maximalne 0.6, u vice zjemnenych siti, ci pri "divokych" reseni musi byt mensi, napr. 0.1, 0.2, 0.4. V pripade, ze vypocet zhavaruje, tak lze doporučit vypocet s mensim CFL.

### 3.2 Beh programu Dgm

Pokud kompilace byla v poradku, tak program Dgm lze spustit z adresare **data** prikazem

```
../src/Dgm 0.ini
```

Pro uspesny beh programu Dgm musi v adresari, odkud se program spousti, byt soubor (viz priklad v adresari data) 'profiles', který popisuje krivou cast hranice. V pripade polygonalni hranice soubor 'profiles' obsahuje pouze jednu celociselnou '0'. Format souboru 'profiles' je stejný jako u baliku ANGENER.

Pri behu programu se napred vypisou nejake komentare ke vstupnim datum a pak postupne kazda 25. iterace. Tento vypis obsahuje cislo iterace, celkovy fyzikalni cas, reziduum (rozdil hustoty ve dvou nasledujich casovych krocich v  $L^1$ -norme deleny casovym krokem – timto se meri konvergence ke stacionarnimu stavu), maximalni a minimalni hodnoty Machova cisla v cele oblasti a pocet elementu, kde se provadel limiting. Beh programu na prilozenych datech by mel dat priblizne nasledujici vypis:

```
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
Numbered solution output physical time step: 0 s
Integrate using Euler formula
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
The 16 nodes detected on curved part of boundary
  Begin of Linear_Mapping_Computing
  END of Linear_Mapping_Computing
Solver_Init finished
```

Iter	Time	Reziduum	Mach: minimal	maximal	limit
0	0.0007	2.1923e-01	0.6672	0.6730	0(0)
25	0.0172	2.7604e-01	0.5405	0.7384	0(0)
50	0.0336	3.5036e-01	0.4725	0.7429	0(0)
75	0.0501	3.4920e-01	0.4544	0.7281	0(0)
100	0.0666	3.3826e-01	0.4652	0.7465	0(0)
125	0.0831	3.5673e-01	0.4546	0.7477	0(0)
150	0.0996	3.5905e-01	0.4512	0.7606	0(0)
175	0.1161	3.5273e-01	0.4384	0.7735	0(0)
200	0.1326	3.7414e-01	0.4292	0.7838	0(0)

.  
.  
.

Po uspesnem probehnuti 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).

## 4 Pouziti programu `Plotdgm`

Pro vizualizaci spoctenych vysledku se uziva program `Plotdgm`, který se spousti se dvema parametry, prvnim je soubor s triangulaci a druhym soubor s vysledky spoctenymi na teto triangulaci, t.j. v nasem pripade

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

V adresari, odkud se program `Plotdgm` spousti musi byt dale obsazen soubor `'plot.dgm'` (viz priklad v adresari `data/`, kam si uzivatel zada veliciny a typy zobrazeni, která chce vykreslit. Prikladem je soubor z adresare `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
```

- Prvni znak(y) znaci velicinu, která se ma zobrazit, jsou nasledujici moznosti: RO – hustota, RV – velikost momentu sily, V – velikost rychlosti, V1, V2 – složky rychlosti, P – tlak, PC – tlakovy koeficient, M – Machovo cislo, E – energie, T – teplota, CF – skin friction coefficient. EOD znamena konce datoveho souboru, co je za nim se jiz nenacita.
- Druhy symbol znamena vykreslovany typ:
  - WALL : vykresli rozlozeni na stene (stenach), napred se uvadi pocet sten, na kterych se ma soucasne vykreslovat a pak `ibc` techto sten (ze souboru `'triang'`). Defaultove je nastaveno, ze u sten s `ibc` rovno 1 nebo 2 (obvykle vstup nebo vystup) se vykresluje zavislost veliciny na  $x_2$  u ostatnich na  $x_1$  ( $x_1, x_2$  jsou kartezyke souradnice).
  - ISO : vykresli isokrivky, uvede se pocet (napr. 81) a pak se uvede bud 0, pak se isokrivky vykresli od maxima do minima dane veliciny, nebo se uvede 1 a minimalni a maximalni hodnota, které se maji zobrazit. Isokrivky jsou vzdy rozlozeny ekvidistantne.
  - CUT : vykresli se rez dane veliciny podel osy danou dvema body, jejich souradnice se uvedou za znaky CUT, napr. v zde uvedenem prikladu se vykresli rez podel osy  $x_2 = 0.2$ . Vznikly soubor obsahuje tri sloupce, souradnici  $x_1$ , spoctenou hodnotu na rezu a souradnici  $x_2$ .

Vysledkem behu programu `Plotdgm` jsou soubory `'gnu.01'`, `'gnu.02'`, `'gnu.03'`, ..., pode poradí zadanych vizualizaci v souboru `'plot.dgm'`. Jejich pocet by mel odpovidat poctu radku v `'plt.dgm'` pred EOD. Pokud jiz v adresari nejake soubory `'gnu.*'` existovaly, tak se automaticky prepisi. Vsechny tyto soubory jsou snadno zobrazitelne v gnuplotu, napr. prikazem

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

pri 3D vizualizaci isokrivek lze uzit

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

pri zobrazeni rezu v zavislosti na  $x_2$  lze uzit

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

(viz program gnuplot a jeho napoveda).

## 5 Zaverena poznámka

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

## References

- [1] V. Dolejší. A higher order scheme based on the finite volume approach. In R. Herbin and D. Kröner, editors, *Finite Volumes for Complex Applications III (Problems and Perspectives)*, pages 333–340. Hermes, 2002.
- [2] V. Dolejší. On the discontinuous Galerkin method for the numerical solution of the Navier–Stokes equations. *Int. J. Numer. Methods Fluids*, 45:1083–1106, 2004. (submitted).
- [3] V. Dolejší and M. Feistauer. On the discontinuous Galerkin method for the numerical solution of compressible high-speed flow. In F. Brezzi, A. Buffa, S. Corsaro, and A. Murli, editors, *Numerical Mathematics and Advanced Applications, ENUMATH 2001*, pages 65–84. Springer-Verlag, Italia, Milano, 2003.
- [4] 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.