

Lagrangian Model (LM)

./LM/script

run.csh モデルを動かすためのスクリプトを格納
run_mpi.csh モデルの実行スクリプト(シリアル版)
combine_res_mpi.csh、delete_res_mpi.csh モデルの実行スクリプト(並列化版)
draw_grads.csh 並列計算後処理用
 GrADSを使って地上濃度を描画

./LM/source

makefile モデルのソースコードが入っている。
 コンパイラやコンパイルオプションの設定
main.F90 主プログラム(シリアル版)
main_mpi.F90 主プログラム(並列版) → 講習会はMPI環境が無いため、
 コンパイルしません(makefileの書き換えが必要)
main_traject.F90 **今回提供なし。**
\$subroutines.F90
 サブルーチン(副プログラム)
 (基本的に、1ファイルにつき1サブルーチン、
 サブルーチン名とファイル名は同一。)

./PM/\$output_name

計算結果が格納される。
\$output_nameは、run.cshで指定。

これまで報告されている注意すべき点

結果にコンパイラ依存性が見られる。

・g95とgfortranの結果でbudgetを比較したところ、若干の違いが見られる。大気負荷量、モデル領域外への流出量、湿性沈着量、乾性沈着量を比べると、大気負荷量の違いが最も大きいとその違いは、相関係数 R^2 で0.9997、RMSEは4.5[TBq]で、平均値140 [TBq]の3%程度であった。両者の差が最も小さいのは乾性沈着量で、 R^2 で1.000、RMSEは0.0478 [TBq] (平均値は22TBq)。

あるバージョンのPGIではrandomが[0:1]区間ではなく、1より大きい値を返す(v13.2で確認)

・[0:1]の範囲を超えると、もう一度乱数を発生させることで回避。

あるバージョン(6.1)のPGIでは、real, parameter :: pi = acos(-1.0)が通らない(v6.1で確認)

・main.F90、gran.F90、emit.F90にて、pi = 3.14159265358979324と指定する。

1 モデルのパラメータ制御 (run.csh)

`./LM/script/run.csh` (変更して、実行する(`./run.csh + enter`))

```
set meteo_path = ../../gpv_tools/GPVMSM_grads      (GrADS形式の気象解析データの格納場所)
set fort_name = fort                               (fortranコンパイラが指定するファイル名:コンパイラ依存)
set output_name = result01                        (計算名:任意)
```

#(main param) メインパラメータの設定

```
set ii = 41 # 130E - 145E with 0.125 deg resolution (切り出したGPV-MSM領域の緯度方向の数)
set jj = 51 # 30N - 45N with 0.100 deg resolution (切り出したGPV-MSM領域の経度方向のグリッド数)
set kk = 10 # 1000 975 950 925 900 850 800 700 600 500 (計算する気圧面の数;最大16まで)
set p_layers = "1000, 975, 950, 925, 900, 850, 800, 700, 600, 500" #consistent with kk (hPa) (具体的な気圧面hPa)
```

#(nuclear power plant location) 原子力発電所の位置を設定

```
set lon_npp = 141.033611 (原子力発電所の緯度E)
set lat_npp = 37.421389 (原子力発電所の経度N)
```

#(time param) 計算期間、ラグランジュ粒子の設定

```
set hindcast = 1 # hindcast mode (control start_date and end_date) (過去再現実験:解析時刻を繋ぎ合わせて計算を実行)
set start_date = 2016060100 # yyyymmddhh (UTC) need to be set every 3 hour (計算開始時刻UTC)
set end_date = 2016060300 # yyyymmddhh (UTC) need to be set every 3 hour (計算終了時刻UTC)
set forecast = 0 # forecast mode (control analy_date and fcst_hours) (予報モード:JPV-MSMの39時間予報を用いて計算を実行)
set analy_date = 2016060100 # yyyymmddhh (UTC) need to be set every 3 hour (解析時刻の設定)
set start_time = 0 # since 0 (=0 h) (計算開始ステップ:MSMの時間間隔は3hなので、3倍した数が予報時刻になる)
set end_time = 13 # up to 13 (=39 h) for GPV-MSM (計算終了ステップ:最大13まで)
```

注) hindcast=1ならforecast=0、hindcast=0ならforecast=1。hindcast=0なら再現計算用のstart_date, end_dateは計算に反映されない。同様にforecast=0なら予報計算用のanaly_date, start_time, end_timeは読まれない。

```
set dt = 60 # sec (計算時間間隔Δt → 1分推奨)
set dt_output = 60 # min (出力ファイルの時間間隔 → 1時間推奨)
set dt_meteo = 180 # min (気象解析値の時間間隔 → 3時間)
```

1 モデルのパラメータ制御 (run.csh)

つづき

`$(grid param)` 計算格子の設定

```
set lon0 = 138.0 # lon at x = 1 (切り出した領域の経度(X=1))
set lat0 = 34.0 # lat at y = 1 (切り出した領域の緯度(Y=1))
set dx = 0.125 # degree (Δx; 経度方向の格子間隔(degree))
set dy = 0.100 # degree (Δy; 緯度方向の格子間隔(degree))
set kkout = 10 # number of output layers (出力濃度レベルの数)
set z_layers = " 50, 100, 200, 300, 400, ¥
                500, 1000, 1500, 2000, 3000" #grid top height (mMSL) consistent with kkout (出力濃度レベルの具体的な値)
set iflag_AGL = 1 # output layers, z_layers is in 1. mAGL or 0. not (1.なら上記出力濃度レベルを地上高度で規定: iflag_MSL=0にする)
set iflag_MSL = 0 # output layers, z_layers is in 1. mMSL or 0. not (1.なら上記出力濃度レベルを海拔高度で規定: iflag_AGL=0にする)
```

`$(point source emission option)` ポイントソースの排出源設定

```
set emis_opt = 0 # 0. continuous constant emission [Bq/h], 2. pulse emission [Bq] (0. 連続定常放出、2. パルス放出(計算開始時のみ)、
# 1. Katata et al., ACP (2015), negative: zero emission 1. Katata et al. (2015) のインベントリ(2011年3月-のみ)、負値は放出なし)
set emis_rat = 1e12 # constant emission rate [Bq/h] for $emis_opt = 0 (連続定常放出率[Bq/h]。パルス放出の場合は[Bq])
# emission amount [Bq] for $emis_opt = 2
set emis_hgt = 25 # constant emission height [m] for $emis_opt = 0,2 (プルーム中心高度: 中心緯度経度は$lon_npp, $lat_nppで指定済み)
set width_lat = 0.001 # plume width in latitude [deg] for $emis_opt = 0,2: 0.001 = ~100 m (緯度方向のプルーム幅)
set width_lon = 0.001 # plume width in longitude [deg] for $emis_opt = 0,2: 0.001 = ~100 m (経度方向のプルーム幅)
set width_hgt = 10.d0 # plume width in height [m] (プルーム高さ: 地下に潜らないように注意)
set npphr = 100 # number of Lagrangian particles emitted per hour for $emis_opt = 0,1 (粒子の放出率[#/h]。ただしパルス放出の場合は[#])
# initial total number of Lagrangian particles for $emis_opt = 2
```

`$(areal source emission option)` NOT SUPPORTED IN THE VERSION 面源の排出源設定 (別途様々な境界条件を準備する必要があるため今回は提供なし)

```
set area_emis_opt = -1 # 0. areal source emission option, negative: zero emission
set area_emis_spc = 1 # 1. Ishizuka et al. dust resuspension model
# 2. biogenic resuspension model (area_emis_fct * cs137dep * greenfraction * forest area)
# 3. biomass burning resuspension model (firespot) -- not implemented
set area_emis_fct = 1 # constant emission rate factor [/h] for $area_spec = 2 or 3
set area_emis_hgt = 25 # constant emission height [m]
set area_npphr = 10 # number of Lagrangian particles emitted per hour per grid
set area_emis_dt = 600 # frequency to call area_emis in order to save Lag particles
# mod(area_emis_dt,dt) should be zero
```

1 モデルのパラメータ制御 (run.csh)

つづき

```
 #(dynamics options) 診断的Kzのかわりに気象モデルで計算したKzを用いる。Kzは乱流拡散係数。今回は提供なし  
 set use_bound_kz = 0    # 0. diagnose Kz based on pressure level meteorology data  
                        # 1. use prognostic Kz from boundary file ($seddyz_path)
```

```
 #(physics / chemical parameters) 乾性沈着、落下速度、湿性沈着に関するパラメータ  
 set ddep_vel   = 1.e-3 # dry deposition velocity [m/s] (乾性沈着速度。Kajino et al. (2016) では海と陸で可変としているが今回は提供なし)  
 set setl_vel   = 0.    # gravitational settling velocity [m/s] (重力落下速度。微小粒子の場合無視できる)  
 set coll      = 0.05  # coalescence efficiency btw aerosol and rain (default=0.05) (エアロゾルと雨の衝突確率)
```

```
 #(output grid resolution ratio to model grid resolution) 出力結果の解像度をモデル解像度の何倍に設定するか。  
 set ratio_dx   = 2    (緯度方向の格子解像度がratio_dx倍に、格子数がratio_dx倍に)  
 set ratio_dy   = 2    (経度方向の格子解像度がratio_dy倍に、格子数がratio_dy倍に)
```

```
 #(domain info) (for GrADS description file)  
 set zdef      = "$kkout levels $lay" (変更の必要なし)  
 set tdef      = "481 linear 00z12mar2011 ${dt_output}mn" #20 days * 24 hours + 1 (JST) (出力時間数と、開始時間をベタ打ちする。)
```

(GrADS ctlファイル用: 337は計算日数 $14 \times dt_output + 1$ 、00z12mar2011は、\$start_dateをJSTに変換したもの)

run.csh, run_mpi.cshに載せていない隠れたnamelistについて

スクリプト下方の、&emisから&endまでの箇所下記のパラメータを直書きすることで、放出期間を限定できる。

再現実験の場合 (2016/6/1から6/2までしか放出させない)

```
emis_st_date = 2016060100,  
emis_en_date = 2016060200,
```

予報実験の場合 (5ステップから10ステップまでしか放出させない: 15hから30hまで)

```
emis_st_time = 5,  
emis_en_time = 10,
```

1 モデルのパラメータ制御 (run_mpi.csh, etc.)

./LM/script/run_mpi.csh (変更して、実行する(./run_mpi.csh + enter)) 現在は、qsubバッチスクリプト用となっている。

#(mpi variables) 並列化の設定

set ntasks = 分割数 (実質上のパーティクル数は、npphr × ntasksとなる。)

ntasks = 16の場合、conc.datのかわりに、conc_ntask1.dat, conc_ntask2.dat, ... conc_ntask16.datが出来るため、これらをまとめて平均値を取り、conc.datとし、不要結果(conc_ntask*.dat)を削除する必要がある。(ddep, wdep, tdepも同様)。これらを、combine_res_mpi.cshおよび、delete_res_mpi.cshで行う。

./PM/script/combine_res_mpi.csh (変更して、実行する(./combine_res_mpi.csh + enter))

./PM/script/delete_res_mpi.csh (変更して、実行する(./delete_res_mpi.csh + enter))

set ntasks = (run_mpi.cshと同一)

set output_name = (run_mpi.cshと同一)

set ii_output, jj_output = ii*ratio_dx, jj*ratio_dy

set kk_output = (./\$output_name/conc.ctiのzdefと同一)

set nt_output = (./\$output_name/*ctiのtdefと同一)

./LM/script/draw_grads.csh (変更して実行する(./draw_grads.csh + enter))。../{計算結果フォルダ}/figs/以下に、地上最下層濃度と風ベクトルの画像とアニメーションが作成される)

set stdate = "2016/06/01 09 JST" #simulation start date

set nhours = 39 #simulation integration time

計算積分時間ステップ数(hourとなっているのはデフォルト出力間隔が1時間だから)

set fname = ../test #simulation result folder name

計算結果が収納されているフォルダの場所

set lonmin = 138 #same as ../gpv_tools/gpvmsm2grads_monthly.csh モデル西端経度

set lonmax = 143 #same as ../gpv_tools/gpvmsm2grads_monthly.csh モデル東端経度

set latmin = 34 #same as ../gpv_tools/gpvmsm2grads_monthly.csh モデル南端緯度

set latmax = 39 #same as ../gpv_tools/gpvmsm2grads_monthly.csh モデル北端緯度

set clevs = "0.1 0.3 0.6 1 3 6 10 30 60 100" # Bq/m3 reasonable range for emission of 1TBq/h (1e12 Bq/h) カラーバーのレベル指定(数は10が良い)

ここ(run.csh)で制御できないパラメータは
ソースコードレベルで制御

2 ソースコードの説明

./PM/source/*.F90

ソースコードに変更を加えた場合は、./PM/source/にて、%make clean, %make が必要

主プログラム main.F90
(重要なパラメータ)

(main.F90を書き換える場合は、main_mpi.F90も書き換えた方が、間違えなくて良い。逆も同じ。)

60行目 npmax

! ラグランジュ粒子の最大許容数

187行目 t_hlife

! 半減期

191行目 z_surflay

! 接地境界層高さ(乾性沈着を与える高度)

(重要な変数)

ltrac(npmax,6)

! ラグランジュ粒子の番号

1:緯度、2:経度、3:高さ(地表からの高度)、

4: Δt 前の経度方向の水平風ゆらぎ、5: Δt 前の緯度方向の水平風ゆらぎ、6: 放射能(Bq)

lflag(npmax)

! ラグランジュ粒子が存在するかどうか

(まだ与えられていないか、モデル領域外に出た)

(注意すべき点、本計算の前に試行計算をしてください)

計算の途中でモデル領域内に存在する粒子の総数(nptot)が、最大許容数npmaxを超えると計算が途中で落ちる。まずは本計算の前に、npshr=100の最軽量の試行計算を実行し、計算にかかるCPU時間と計算期間中の最大粒子存在数(nptot_max)を調べる(\$output_name/main.printoutに出力されている)。計算にかかるCPU時間と、最大粒子存在数は、npshrに線形に比例し、計算精度は-1/2乗となることを勘案して、自分が許容できる計算時間と計算精度に最適なnpshrとnpmaxを勘案して、本計算を開始する。

例) run.csh で npshr = 100 として試行計算を実施したところ、nptot_max = 3850、CPU時間は32.5分であった。計算時間は、5時間程度の計算時間ならば許容できるため、本計算は、npshr = 1000、npmax = 50000 とした。

2-1 主プログラムmain.F90の構造

```
call read_namelist
call read_meteo
```

(run.cshで指定したパラメータを読み込む)
(初期時刻の気象解析値の読み込み)

```
555 continue      (気象解析値ループ:3時間ごと)
```

```
    call nextdate
    if (idate > end_date) goto 777
    call read_meteo
    do 501 nt1 = 1, nttt1
    do 502 nt2 = 1, nttt2
        call emit
        call transport
        call removal
```

(dateを3時間進ませる)

(3h先の気象解析値の読み込み)

(出力時間ループ:1時間)

(Δt ループ:1分)

(放出量の読み込み(1))

(移流・拡散計算(2))

(湿性・乾性沈着(3))

```
502     continue
```

```
        call conver2conc
        call write_output
```

(ラグランジュ粒子を格子濃度に変換)

(格子濃度と沈着量を出力)

```
501     continue
```

```
    goto 555
```

```
777 end
```

2-2 (1)副プログラムemit.F90

(発生量、発生高度、発生時間を与えるプログラム)

```
subroutine emit( emis_opt, emis_rat, emis_hgt, &  
               idate0, tfact1, e_cs137, e_hgt)
```

```
integer, intent(in) :: emis_opt           ! 0. constant emission  1. Katata et al., ACPD (2014)  
                                           ! 2. pulse emission  
real*8 , intent(in) :: emis_rat           ! constant emission rate [Bq/h] for emis_opt = 0  
real*8 , intent(in) :: emis_hgt          ! constant emission height [m] for emis_opt = 0  
integer, intent(in) :: idate0             ! yyyymmddhh (t, t=t+dt)  
real*8 , intent(in) :: tfact1             ! 0-1 between idate0 and idate1 (=3 hour)  
real*8 , intent(out) :: e_cs137,e_hgt     ! emission amount [Bq/h] and height [m]
```

15行目 idata (排出量データ(シナリオ)の数)

排出量の与え方

startdate(id)と、enddate(id)の間の放出量を、c37(id) [Bq/h]、高さ emh(id) [m]とする。

これは、JAEAの最新放出シナリオ(Katata et al., ACP (2015))を与えている。
emit.F90の編集によって、任意のシナリオを考慮できる。

ただし、startdateとenddateは実数なので、分については小数点で表現
(例)2011年3月12日00時30分 → 2011031200.5

2-3 (2)副プログラムtransport.F90 (移流・乱流拡散計算を実行するプログラム)

```
real*8, dimension(npmax,ll), intent(inout) :: ltrac
logical, dimension(npmax), intent(inout) :: lflag
integer, intent(inout) :: nptot
real*8, dimension(5), intent(inout) :: budget ! 収支について(単位は[Bq])
! 1. emitted activity
! 2. atmospheric activity burden
! 3. dry deposited activity
! 4. dry deposited activity over land →今回提供なし(3と同じ値)
! 5. wet deposited activity
! 6. wet deposited activity over land →今回提供なし(5と同じ値)
! 7. activity transported out of boundary

call uvkz_interp    水平風(u, v)と鉛直乱流拡散係数(kz)の内挿
call w_interp      鉛直風(omg[Pa/s]を内挿・変換して[m/s]に)
```

※CFL条件(クーラン数 $cfl=0.5$)を越えないようにtime splittingを実施している。
gausは規格化された正規分布に従う乱数(平均値0、標準偏差1)を発生するプログラム

2-4 (3)副プログラムremoval.F90 (乾性沈着、湿性沈着、放射性壊変)

```
real*8, dimension(npmax,ll), intent(inout) :: ltrac  
logical,dimension(npmax), intent(inout) :: lflag  
integer,          intent(inout) :: nptot  
real, dimension(ii,jj), intent(inout) :: ddep, wdep  ! [Bq/m2]
```

66行目 湿性沈着パラメータ(beta)

67行目 併合確率(coll)

79行目 半減期(t_hlife)※ここでは単位は年だが、114行目で壊変率の単位が(/s)となるよう調節する。

ここで重要なのは、沈着過程では、ラグランジュ粒子の量は減らずに、ラグランジュ粒子が持つ質量(放射能)が減る、ということ。

2-5 (4)副プログラムconvert2conc.F90 (ラグランジュ粒子をオイラー濃度に変換)

モデルは地表からの高さ(AGL)で計算しているが、VDVGEは海拔高度(MSL)を採用している都合上、MSLで出力するようにしている。しかし、地上濃度の分布を描画したいときは、AGLで表示の方が都合が良いため、iflag_AGL = 1, iflag_MSL = 0とすると、海拔高度ではなく、地表面からの高さでconc.datを出力するようになる。

3 モデルの出力について

./PM/\$output_name/conc.ctl

3次元濃度 (Bq/m³) (conc.dat) をGrADSやVDVGEで読み込むためのdescriptionファイル。VDVGEで対数表示を可能にするため、0ではなく下限値 (tiny; write_output.f90で制御) を代入。

./PM/\$output_name/ddep.ctl, wdep.ctl, tdep.ctl

乾性、湿性、トータル (乾性+湿性) 沈着量 (Bq/m²) (ddep.dat, wdep.dat, tdep.dat) をGrADSやVDVGEで読み込むためのdescriptionファイル。VDVGEで対数表示を可能にするため、0ではなく下限値 (tiny; write_output.f90で制御) を代入。

./PM/source/zs.ctl

モデル標高 (GPV-MSMの地上気圧からの推定値)
(./PM/\$output_name/zs.dat) を読み込むためのdescriptionファイル。

./PM/\$output_name/main.printout (標準出力。並列計算の場合は、rank=0の結果のみ出力)

プログラムmain.exeの標準出力結果を格納 (**計算のコツ**: まずは軽い試計算 (npshr=100) を実施、計算時間とnptotの最大値nptot_maxを見て、自分が許容できる時間の限りnpshrを多くした本計算を実施する) → スライド7でも説明している。

バジェットのチェックも重要。航空機観測から得られたエリア内Cs-137の降下量は**2.68** PBq、本デフォルト計算では**2.53** PBqで、バジェットとして整合的 → 水平分布は異なるものの、LMモデルによる収支解析の妥当性の根拠としている (Kajino et al., 2016において) Katata et al. (2015) の2011年3月31日までの総放出量は約 **14.1** PBq。

3 モデルの出力について(標準出力結果)

run.csh(福島事故計算)の出力例

計算の経過中に出力されるもの。

```
write_output_here (output time) =      481
      (nptot, nptot_max) =      1170      2087
```

出力時間
L粒子の数、L粒子の最高値

budget output

```
emission          [TBq] 14064.2699999959
atmos. burden     [TBq]  31.8093343996941
dry deposition    [TBq]  778.674893191254
dry deposition over land [TBq] 517.343180173955
wet deposition    [TBq] 3512.76272704940
wet deposition over land [TBq] 2070.88276927904
out of boundary   [TBq] 9782.11455266721
```

バジレットの表示
時間積算排出量
領域内の大気中の存在量
領域内の時間積算乾性沈着量
陸域のみの時間積算乾性沈着量
(今回提供なし。領域全体と同じ値)
領域内の時間積算湿性沈着量
陸域のみの時間積算湿性沈着量
(今回提供なし。領域全体と同じ値)
領域外への流出量

計算の終了時に出力されるもの。

```
num of particles per hour=      100
cpu time (min) =      9.559896
npxmax          =      600000
maximum nptot  =      2087
average nptot  =      970.1660
```

1時間あたりのL粒子放出量
CPU経過時間
設定した最大許容L粒子量
L粒子の最高値
L粒子の平均数