GreeM 簡易マニュアル

石山 智明
筑波大学計算科学研究センター
ishiyama@ccs.tsukuba.ac.jp

平成 25 年 5 月 2 日

目 次

文章中に太字で書かれた大部分のマクロは Makefile に、変数は param.h で定義されています。

1 実行環境

コンパイラは gcc が前提で、以下のライブラリが必要となります。

- fftw 3.3 以降 (MPI 並列に対応したもの)
- Phantom-GRAPE (http://code.google.com/p/phantom-grape/)

Phantom-GRAPE (Nitadori et al. 2006, Tanikawa et al. 2012, 2013) は、N 体シミュレーションで一番の高コスト部である、粒子間重力の計算を高速化するライブラリです。SSE や AVX を利用するなど、個々の CPU に特化したチューニングがなされているため、実行環境に応じてリンクするライブラリを変更する必要があります。AVX に対応している場合は (Intel sandy-bridge 以降) lib ディレクトリ内の libpg5.a.avx (=libpg5.a) を、それ以前の SSE のみ対応している場合は libpg5.a.sse を使用してください。前者は例えば、国立天文台の XC30、後者は筑波大学の T2K 等が該当します。他に京コンピュータや FX10 に対応したバージョンもありますが、リンクの仕方が異なるため、これらの計算機で使用する場合は別途相談ください。

2 使い方

2.1 コマンドライン引数

```
mpirun -n XXX ./GreeM initial_condition param #files (CPUTIME)
```

initial_condition

初期条件を指定します。Gadget フォーマットに対応しています。

param

スナップショットを出力する redshift を指定するためのファイルです。フォーマットは一行目、二行目は適当な数値で (互換性のため残っているだけです)、三行目にスナップショットの数を、それ以降に redshift を降順に列挙します。例えば

```
0
0
5
5
3
2
1
0
```

のようにすると、$z = 5, 3, 2, 1, 0$ の 5 つのスナップショットが出力されます。

#files

初期条件のファイル数を指定します。1 とした場合はそのファイル名のファイルが、1 より大きい場合は、ファイル名-X のファイルが読み込まれます。ファイル数とノード数が異なる場合は、**RESTART2** マクロを有効化する必要があります。

CPUTIME

指定しなくても動作します。指定した秒数プログラムが継続すると、リスタート用のファイルを出力し終了します。指定しない場合は、**MAX_CPUTIME** で指定されている時間になります。

2.2 特に重要なパラメータ

以下 4 つのパラメータは一番変更する頻度が高いため、グループ化してあります。Makefile 内の **MODEL_IDENTIFIER** で各グループを指定します。各グループの定義は param.h 内にあり、そこで以下の 4 つのパラメータを定義する必要があります。

NUMBER_OF_PART_ALL

シミュレーションの全粒子数です。

NUMBER_OF_PART

各ノードが粒子データのために用意する配列のサイズです。メモリサイズにもよりますが、**NUMBER_OF_PART_ALL** をノード数で割った数より、余裕を持たせる必要があります。大概は 1.3 倍程度にしておけば問題ありません。

SIZE_OF_MESH

PM の一次元方向のメッシュサイズを指定します。推奨値は $N^{\frac{1}{3}}/2$ です (1024^3 粒子なら 512)。

SFT_FOR_PP

ボックスサイズで規格化した、共動座標系における重力ソフトニングを指定します。例えば、2.5×10^{-4} で、ボックスサイズが 320Mpc/h なら、80kpc/h になります。

3 プログラムの終了フラグ

1. ステップ数が **MAX_STEPS** を超えた。
2. 経過した CPU 時間 (秒) が **MAX_CPUTIME**、もしくはコマンドライン引数で指定した CPUTIME を超えた。
3. redshift が **Z_FIN** より小さくなった。
4. 実行ディレクトリに **STOPFILE** で定義される名前のファイルが存在した場合 (デフォルトでは stop)。

4 リスタートの方法

プログラム終了フラグの 3 を除いて、**DUMPDIR** で指定されたディレクトリにその時点でのスナップショットを書き出して終了します (3 は終了するだけ)。ファイル名は **DUMPFILE** となります。また各ノードが個々にファイルを出力します。したがって DUMPDIR/DUMPFILE-(MPI rank) という名前のファイルがノード数分生成されます (デフォルトでは Dump/dump-?)。

リスタート時には 実行時引数の “initial_condition” に$DUMPDIR/$DUMPFILE を指定します (デフォルト設定では Dump/dump)。前回実行時とノード数を変更したい場合は、**RESTART2** マクロを有効にする必要があります。

5 ファイルフォーマット

GADGET_IO マクロを有効にすると、入出力ともに Gadget フォーマットになります。ユニットは gadget_param.h で指定します。

リスタートファイル、スナップショットともにフォーマットは同じです。

6 Makefile 内のオプション

ここにある以外のオプションは通常は有効にする必要はありません。

Parallel_IO

スナップショットの IO の際に、他のノードが IO しているかどうかに関わらず、個々のノードが IO を実行します。XC30 や T2K のように、大規模計算機では有効に、比較的小さい PC クラスタでは無効にするのが推奨です。

RESTART2

初期条件の入力の際に、このマクロが無効の場合、ノード数=初期条件ファイル数 (1 を除く) が仮定されます。そうでない場合、このマクロを有効にする必要があります。

GADGET_IO

ファイルフォーマットを Gadget フォーマットに切替えます。

USING_AVX -mavx, BUFFER_FOR_TREE, TREE2, TREE_PARTICLE_CACHE

XC30 のように、Intel sandy-bridge 以降の AVX が利用できる環境で、プログラムを少し高速化するためのオプションです。4 つ同時に有効にしてください。

VERBOSE_MODE

プログラムの経過情報を標準エラーに出力します。

VERBOSE_MODE2

プログラムの経過情報を出力する際に、バリア同期を行います。通信関係で謎のエラーが発生した時に使用すると、デバッグしやすくなることもあります。プログラムの実行速度が著しく低下します。

MY_MPI_BARRIER

プログラムの至るところでバリア同期を行います。通信関係で謎のエラーが発生した時に使用すると、デバッグしやすくなることもあります。プログラムの実行速度が著しく低下します。

7　param.h 内のオプション

ここにある以外のオプションは通常は変更する必要はありません。

7.1　精度

Z_SWITCH_ETA　　0.0 (デフォルト値)

タイムステップを調整するためのパラメータ η の値を切り替える redshift です。これより大きい redshift では後述の **ETA_HIGHZ** が、小さい redshift では **ETA_TIMESTEP** が用いられます。

ETA_TIMESTEP　　0.3

タイムステップを調整するためのパラメータで、Gadget の $\sqrt{2.0 \times \mathrm{ErrTolIntAccuracy}}$ に相当します。

ETA_HIGHZ　　0.3

タイムステップを調整するためのパラメータで、Gadget の $\sqrt{2.0 \times \mathrm{ErrTolIntAccuracy}}$ に相当します。

MAX_DLOGA　　0.03

タイムステップを調整するためのパラメータで、Gadget の MaxSizeTimestep と同じです。

Z_SWITCH_THETA　　0

ツリー法の精度を決めるパラメータである opening angle θ の値を切り替える redshift です。これより大きい redshift では後述の **THETA_HIGHZ** が、小さい redshift では **THETA** が用いられます。

THETA_HIGHZ　　0.5

ツリー法の精度を決定します。小さいほど精度が良くなります。

THETA　　0.5

ツリー法の精度を決定します。小さいほど精度が良くなります。

CUTOFF_RADIUS　　3.0

PM のメッシュ幅で規格化した、近距離力のカットオフ長を指定します。SIZE_OF_MESH $= \mathrm{N}^{\frac{1}{3}}/2$ なら 3.0 が推奨値です。

CONST_Z_VALUE **0**

ソフトニングを物理座標で固定したい場合使用します。ここで指定した redshift 以降は、ソフトニングが物理座標に換算した時に、SFT_FOR_PP × ボックスサイズ になるように固定されます。逆にこの redshift 以前は、共動座標で SFT_FOR_PP × (1.0 + CONST_Z_VALUE) になるように固定されます。例えば CONST_Z_VALUE = 5.0、SFT_FOR_PP = 2.5×10^{-4} で、ボックスサイズが 320Mpc/h なら、$z = 5$ 以降は、ソフトニングが物理座標で 80kpc/h に、$z = 5$ 以前では、共動座標で 480kpc/h に固定されます。

7.2 IO

MODEL **Test**

ログファイルのベースのファイル名を指定します。

SNAPSHOT **Snapshot**

スナップショットを出力するディレクトリ名を指定します。ディレクトリはプログラム実行時に自動生成されます。

DUMPDIR **Dump**

リスタートファイルを出力するディレクトリ名を指定します。ディレクトリはプログラム実行時に自動生成されます。

DUMPFILE **dump**

リスタートファイルのベースのファイル名を指定します。

LOADBALANCELOG_ON **0**

1 ならば通信やロードバランスの詳細についてのログを出力します。

ONELOGFLAG **1**

1 ならばログファイルをルートノードだけで出力します。それ以外の時全ノードで出力するので、場合によっては大変なことになります。

NUMBER_OF_COMM_PER_ALLTOALLV_FOR_INPUT_IC **16**

RESTART2 が有効な場合、前回実行時とはノード数を変更したリスタートが可能になります。この時はプログラムの最初に全粒子のシャッフルに近い通信を行いますが、その通信の分割数を指定します。デフォルト値で大概の場合問題ありませんが、1 ノードあたりの粒子数が大き過ぎる場合 ($> 10,000,000$)、このパラメータを大きくする必要があるかもしれません。

7.3 ロードバランス

NRATE_EXCHANGE 2500

NUMBER_OF_PART_ALL/NRATE_EXCHANGE の数の粒子をサンプリングして、domain decomposition を行います。小さい程ロードバランスが良くなりますが、domain decomposition の負荷や、メモリ使用量が大きくなります。1024^3 以下の時はデフォルトのまま、それ以上の時は、20000 程度が最適値です。

LOADBALANCE_METHOD 1

domain decomposition の方法を指定します。0 の場合は、ツリー法における interaction 数が等しくなるように分割します。1 の場合は、前ステップの PP+PM の計算時間に基づき、それが等しくなるように分割します。2 の場合は、粒子数が均等になるように分割します。3 の場合は、1 に加えて粒子数の不均一をある程度緩和させます (詳しくは、SAMPLING_LOWER_LIMIT_FACTOR を参照)。

数百ノード以上の場合、ロードバランスは $1 > 0 > 2$ で良くなります。ただし、1 は計測した時間に基づく領域分割を行うため、同じ初期条件、同じノード数のシミュレーションでも、試行毎に結果が多少変わります (問題ない程度に)。

SAMPLING_LOWER_LIMIT_FACTOR 1.2

LOADBALANCE_METHOD = 1 の場合、ロードバランスを最優先する結果、粒子数の不均一が大きくなる時があります。LOADBALANCE_METHOD = 3 にすると、この数値程度の不均一に抑えます。

1Gpc を超えるようなシミュレーションでは、粒子数の不均一はそれほど大きくはならないため、特に指定する必要のないオプションです。

7.4 終了フラグ

MAX_STEPS 100000000

ステップ数がこの値を超えると、リスタートファイルを出力し、プログラムが終了します。

MAX_CPUTIME 1.0e30

経過時間がこの値を超えると、リスタートファイルを出力し、プログラムが終了します。コマンドライン引数として CPUTIME を与えた場合、後者が優先されます。

Z_FIN 0.0

redshift がこの値を下回ると、プログラムが終了します。リスタートファイルは出力しません。

STOPFILE stop

実行ディレクトリにこのファイルが存在すると、リスタートファイルを出力し、プログラムが終了します。

7.5 イメージ出力

IMAGESTEP 32

この数値が 0 より大きいとき、密度分布を可視化した画像を出力します。この数値は出力するステップ間隔を設定します。画像のフォーマットはビットマップです。

IMAGEDIR Image

画像を出力するディレクトリ名を指定します。ディレクトリはプログラム実行時に自動生成されます。画像名は MODEL_(ステップ数).bmp となります。

NEWEST_IMAGEFILE newest.bmp

最新の画像は作業ディレクトリにも出力されます。そのファイル名を指定します。

IMAGEWIDTH, IMAGEHEIGHT 768

画像のサイズを指定します。

IMAGEFACA, IMAGEFACB, IMAGEFACS

密度の規格化のための値で、詳しくは ptobmp.cpp を参照してください。

COLORMAP 0

カラーマップを指定します。0 はグレースケールで、1 は青っぽく、2 は赤っぽくなります。

8 ログファイル

通常は簡易ログと詳細ログの 2 種類のログファイルが出力されます。LOADBALANCELOG_ON = 1 の時は、さらに通信やロードバランスの詳細についてのログを出力します。

簡易ログ

MODEL.log(MPI_rank) という名前のファイルです。ONELOGFLAG=1 の場合はルートノードだけ出力するので、MODEL.log0 という名前のファイルになります。

```
1 1.1224e-04  2.4943e-03  63.0000 1.5625e-02  6.250000e-05    2.5392  2.5392e+00
2 1.1346e-04  2.5213e-03  62.5435 1.5737e-02  6.250000e-05    1.8980  4.4372e+00
3 1.1469e-04  2.5486e-03  62.0887 1.5851e-02  6.250000e-05    2.0517  6.4890e+00
4 1.1593e-04  2.5763e-03  61.6355 1.5965e-02  6.250000e-05    2.1347  8.6236e+00
5 1.1720e-04  2.6044e-03  61.1840 1.6081e-02  6.250000e-05    2.0688  1.0692e+01
6 1.1848e-04  2.6329e-03  60.7341 1.6199e-02  6.250000e-05    2.0796  1.2772e+01
7 1.1978e-04  2.6619e-03  60.2858 1.6317e-02  6.250000e-05    2.1319  1.4904e+01
8 1.2110e-04  2.6912e-03  59.8392 1.6437e-02  6.250000e-05    2.1252  1.7029e+01
```

```
  9 1.2245e-04  2.7210e-03  59.3942 1.6558e-02  6.250000e-05    2.0745  1.9104e+01
 10 1.2381e-04  2.7513e-03  58.9509 1.6680e-02  6.250000e-05    2.0966  2.1200e+01
```

一番左の列から順に、シミュレーションのステップ、ステップ間隔 (単位はスケールファクター)、現在時刻 (コード内部の時間単位で規格化、宇宙論パラメータによるが 1 でだいたい宇宙年齢)、redshift、スケールファクター、共動座標でのソフトニング、このステップにかかった時間、累積時間となっています。

詳細ログ

MODEL.out(MPI_rank) という名前のファイルです。ONELOGFLAG=1 の場合はルートノードだけ出力するので、MODEL.out0 という名前のファイルになります。各パートにどれだけ時間がかかったかを出力します。

LOADBALANCELOG_ON=1 の時

MODEL.loadbalance という名前のファイルです。全ノードの通信等の統計を出力します。