

eemParallel Ver.1.0

利用マニュアル

平成 26 年 3 月

合同会社 P E x P r o C S

1. インストール

まず、ソースコードのアーカイブを以下のコマンドで展開し、作成されたディレクトリに移動する。

```
$ tar zxvf eemParallel_1.0.tar.gz
$ cd eemParallel
```

次に、src ディレクトリに移動し、環境固有設定ファイル `local_settings.mk` を環境に合わせて適宜変更する。

```
$ cd src
$ vi local_settings.mk
```

なお、GNU コンパイラ、Intel コンパイラ、京コンピュータの富士通コンパイラ用には、それぞれサンプルとして、`local_settings.mk.gnu`、`local_settings.mk.intel`、`local_settings.mk.k`を用意しているので、適当なものを `local_settings.mk` に上書きコピーした上で、適宜コンパイラオプション等を変更する。

`local_settings.mk` の修正が終わったら、`make` コマンドにてビルドする。

```
$ make
```

これによって、`eem` ディレクトリに実行形式ファイル `eemParallel` が作成される。

2. 実行

コマンドラインは以下の通りである。

```
mpirun -np <NumProcs> /path/to/eemParallel [options] expfile gmtfile
Options:
--absrad, -A arg          absolute radius
--relrad, -R arg          comma separated list of relative radii (without space)
--pcut, -p arg            first P value cutoff
--itr, -i arg             iteration for second P value calculation
--outfile, -o arg         output file
--mingeneset, -m arg      min geneset size
--maxgeneset, -M arg      max geneset size
--log, -l arg             logFile
--recnull, -r             recycle null distribution
```

ただし、/path/to の部分は、実際の実行形式ファイル eemParallel へのパスで置き換えること。

-R オプションには複数の値をコンマで区切って指定することが可能となっている。ただし、コンマで区切った値のリストにはスペースを入れないように、例えば、“-R 0.05,0.1,0.15” のように指定する。また、相対半径を複数指定した場合には、それぞれの相対半径に対する計算に同数のプロセスを割り当てて計算する仕様となっているため、MPI のプロセス数は指定した相対半径の数の整数倍となるようにする必要がある。入力ファイルの書式、および、その他の対応するオプションの内容については、オリジナルの逐次版プログラムと同様である。

ソースコードのアーカイブを展開して作成された eemParallel ディレクトリの下に data ディレクトリにテストデータとして test.tab、test.gmt がある。これらを用いてテスト計算を行う場合は以下のように実行する。

```
$ cd data
$ ../src/eem/eemParallel test.tab test.gmt
```

なお、上記のように実行すると、進捗を表すメッセージと計算結果が全てターミナルに表示されることになる。計算結果をファイルに保存したい場合には、

```
$ ../src/eem/eemParallel test.tab test.gmt > test.out
```

のようになる。また、

```
$ ../src/eem/eemParallel test.tab test.gmt 1> test.out 2> test.err &
$ tail -f test.err
```

のようになることで、計算結果が `test.out` に、進捗メッセージが `test.err` に保存され、かつ、進捗結果がターミナルに表示される。

付録：ソースコード・アーカイブの内容

ソースコードのアーカイブ `eemParallel_1.0.tar.gz` には、`eemParallel_1.0` ディレクトリ以下に、下記のファイルが保存されている：

- `data-large/`
 - `G0.gmt` サンプル入力ファイル
 - `breastMiller.tab` サンプル入力ファイル
 - `auto_pjsub_k.sh` 「京」でのジョブ自動投入スクリプト
 - `extracttime.pl` 計算時間抽出用補助スクリプト
 - `extracttime.sh` 計算時間抽出用スクリプト
 - `P0NxM/` Nプロセス×M探索パラメータ用ディレクトリ
(N,M)=(1,1),(2,1),(4,1),(8,1),(1,4),(2,4),(4,4),(8,4)
 - ◇ `run_test_k.sh` 「京」用実行スクリプト
- `data/`
 - `test.gmt` サンプル入力ファイル
 - `test.tab` サンプル入力ファイル
 - `auto_pjsub_k.sh` 「京」でのジョブ自動投入スクリプト
 - `P0NxM/` Nプロセス×M探索パラメータ用ディレクトリ
(N,M)=(1,1),(4,1),(8,1),(1,3),(4,3),(8,3)
 - ◇ `run_test.sh` 一般環境用実行スクリプト
 - ◇ `run_test_k.sh` 「京」用実行スクリプト
 - ◇ `test-gnu.err` GNU コンパイラ使用時のエラー出力
 - ◇ `test-gnu.out` GNU コンパイラ使用時の結果出力
 - ◇ `test-intel.err` Intel コンパイラ使用時のエラー出力
 - ◇ `test-intel.out` Intel コンパイラ使用時の結果出力
 - ◇ `test-k.out` 「京」での結果出力
- `src/`
 - `Makefile` メイクファイル
 - `local_settings.mk` 環境固有設定ファイル
 - `local_settings.mk.gnu` GNU コンパイラ用設定ファイル
 - `local_settings.mk.intel` Intel コンパイラ用設定ファイル
 - `local_settings.mk.k` 「京」用設定ファイル

➤ eem/	
✧ AbstractEEMsearch.cpp	AbstractEEMsearch クラス (実装)
✧ AbstractEEMsearch.h	AbstractEEMsearch クラス (宣言)
✧ AbstractGeneSetAnalysis.cpp	AbstractGeneSetAnalysis クラス (実装)
✧ AbstractGeneSetAnalysis.h	AbstractGeneSetAnalysis クラス (宣言)
✧ CoherenceBasedEEM.cpp	CoherenceBasedEEM クラス (実装)
✧ CoherenceBasedEEM.h	CoherenceBasedEEM クラス (宣言)
✧ CoherenceBasedEEMsearch.cpp	CoherenceBasedEEMsearch クラス (実装)
✧ CoherenceBasedEEMsearch.h	CoherenceBasedEEMsearch クラス (宣言)
✧ DistConverter.cpp	DistConverter クラス (実装)
✧ DistConverter.h	DistConverter クラス (宣言)
✧ EEM.cpp	EEM クラス (実装)
✧ EEM.h	EEM クラス (宣言)
✧ EEMsearch.cpp	EEMsearch クラス (実装)
✧ EEMsearch.h	EEMsearch クラス (宣言)
✧ ExpressionModule.cpp	ExpressionModule クラス (実装)
✧ ExpressionModule.h	ExpressionModule クラス (宣言)
✧ ExpressionModuleSet.cpp	ExpressionModuleSet クラス (実装)
✧ ExpressionModuleSet.h	ExpressionModuleSet クラス (宣言)
✧ GeneSetAnalysis.cpp	GeneSetAnalysis クラス (実装)
✧ GeneSetAnalysis.h	GeneSetAnalysis クラス (宣言)
✧ Main.cpp	メイン関数
✧ Makefile	メイクファイル
✧ Makefile.depend	メイクの依存関係
✧ wgetopt.h	Windows 用 getopt ヘッダ
➤ utility/	
✧ Date.cpp	Date クラス (実装)
✧ Date.h	Date クラス (宣言)
✧ Dist.cpp	Dist クラス (実装)
✧ Dist.h	Dist クラス (宣言)
✧ HypergeometricDistributionImpl.cpp	超幾何分布クラス (実装)
✧ HypergeometricDistributionImpl.h	超幾何分布クラス (宣言)

◇	Makefile	メイクファイル
◇	Makefile.depend	メイクの依存関係
◇	MyException.cpp	MyException クラス (実装)
◇	MyException.h	MyException クラス (宣言)
◇	MyFunc.cpp	MyFunc クラス (実装)
◇	MyFunc.h	MyFunc クラス (宣言)
◇	MyMat.cpp	MyMat クラス (実装)
◇	MyMat.h	MyMat クラス (宣言)
◇	StopWatch.cpp	StopWatch クラス (実装)
◇	StopWatch.h	StopWatch クラス (宣言)

以上