

創薬支援アプリ(ISLiMソフトウェア)講習会 2015年9月1日
理化学研究所 東京事務所, 公益財団法人都市活力研究所

REIN-K講習会

実習手順書

宮下 尚之

近畿大学 生物理工学部

理化学研究所 生命システム研究センター

理化学研究所 計算科学研究機構



実習手順書

SCLS FX-10での実習手順書

各自のPCにダウンロード

http://www.islim.org/isl-dl_j.html

ライフサイエンスのグランドチャレンジ・アプリケーション・プロジェクト



次世代生命体統合シミュレーションソフトウェアの研究開発
Next-Generation Integrated Simulation of Living Matter

English

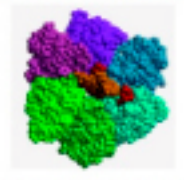
HOME概要研究チーム紹介開発アプリケーション行事・研究会情報ライブラリ問合せ先リンク

ダウンロード

(Updated on 2013/3/28)

●ダウンロードについて
本プロジェクトではその成果を社会に還元する一環として、研究開発の成果物である、スーパーコンピュータ「京」で稼働する開発ソフトウェアをプロジェクト期間中(2006年10月～2013年3月末)にソースコード・レベルで順次公開していきます。ただし一部の開発ソフトウェアについてはソースコードの公開ができないため、バイナリー・コードによる公開となります。また中間成果物である、高並列クラスター上で稼働するバージョンも同様に可能な範囲で公開いたします。産業界での利用、あるいは学生・大学院生の教育への活用など、広くご利用ください。(本プロジェクトでは複数の大学、研究機関から研究開発者が参加・連携してプロジェクトを進めています。そのため、研究開発者の所属機関のダウンロードサイトにリンクしているケースもありますので、ご了承ください。またプロジェクトの進捗に合わせ、ダウンロードについての内容も更新されますので、最新情報についてはこのページでご確認ください。)

分子スケール・アプリケーション

Cearse-Grained Model (CG)

Large spatial/temporal scale events in cellular context

量子科学計算(QM)、分子動力学計算(MM)、粗視化モデル計算(CG)の手法を結合したQM/MM、MM/CG法によって、マルチスケールシミュレーションを実現するためのプログラム。

クリック

番号	アプリケーション	コード名	ダウンロード	開発機関
M-1	マルチコピー・マルチスケール分子シミュレーション法 開発の基盤となるクラスライブラリ	mu2lib-K (開発コード名 Platypus-MM/CG)	ダウンロード	理化学研究所
M-2	レプリカ交換分子動力学計算インターフェイス	REIN-K (開発コード名 Platypus-REIN)	ダウンロード	理化学研究所
M-3	全原子分子動力学計算	MARBLE-K	ダウンロード	横浜市立大学
M-4	粗視化モデル計算	CafeMol-K	ダウンロード	京都大学

ダウンロード



● REIN-K (開発コード名: plutypus-rein) version 1.0.4

レプリカ交換インターフェースプログラム (Replica-Exchange Interface Program: REIN)

License: GPLv3

REIN(開発コード名:plutypus-rein)は、多次元レプリカ交換分子動力学計算を、NAMDなどの既存のMDプログラムを用いて行うインターフェースプログラムです。タンパク質などの生体分子の構造予測や、複数の生体分子の配置予測、2つの分子の結合自由エネルギープロファイル計算などを目的としています。

このversionではMDはNAMDのみに対応しております。

また、本プログラムは京コンピュータで動作しますが、コンパイルオプションを変更する事で、FX10やPCクラスターにも対応できる様になっています。プログラムに関する最新情報はtwitterで配信いたします。[rein_devel] (https://twitter.com/rein_devel)をfollowして下さい。

質問なども受け付けますが、込み入った内容はメール yukimya+rein [a] gmail.com にお願ひ致します。([a]は@に置き換えて下さい。)

下記にご記入の上送信ボタンを押して下さい。

お名前 (必須)	名前
所属 (必須)	所属
部門名	部門名
職業 (必須)	☐ 研究者・開発者 ☐ 学生・院生 ☐ その他
E-Mail	your@e-mail your@e-mail (確認)
REIN-Kについての情報をEmailで受取ることを了承しますが	☐ 了承します。 ☐ 必要ありません。

送信

リセット

入力された個人情報の扱いは、理化学研究所の個人情報保護規程に準拠します。

rein-v.1.0.4.tarをscls fx10へコピー

```
$ scp rein-v.1.0.4.tar.gz hoge@hpci-scls.riken.jp:
```

rein-v.1.0.4.tarを展開

```
$ tar -zxvf rein-v.1.0.4.tar.gz
```

```
$ cd rein-v.1.0.4
```

```
[miya@scls rein-v.1.0.4]$ ls  
AUTHORS.md  COPYING  README_v.1.0.4.md  examples  manual  notes_v1.0.4.md  src  tools  update_v.1.0.4.md  
[miya@scls rein-v.1.0.4]$
```

reinのコンパイル

```
$ cd src
```

```
[miya@scls src]$ ls  
Makefile  Makefile.lib  arch  configure.sh  lib  libmarble  libnamd  main  
[miya@scls src]$
```

reinのコンパイル続き

```
$ sh configure.sh sparc k.openmp
```

```
[miya@scls src]$ ls  
Makefile Makefile.comp Makefile.lib Makefile.machine arch configure.sh lib libmarble libnamd main  
[miya@scls src]$
```

```
$ make
```

```
[miya@scls src]$ ls  
Makefile Makefile.lib arch lib libnamd rein_k_openmp  
Makefile.comp Makefile.machine configure.sh libmarble main  
[miya@scls src]$
```

```
$ make install
```

```
$ cd ..
```

```
[miya@scls rein-v.1.0.4]$ ls  
AUTHORS.md COPYING README_v.1.0.4.md bin examples manual notes_v1.0.4.md src tools update_v.1.0.4.md  
[miya@scls rein-v.1.0.4]$
```

```
$ ls bin
```

```
[miya@scls rein-v.1.0.4]$ ls bin  
rein  
[miya@scls rein-v.1.0.4]$
```

サポーターティングソフトウェアのコンパイル

\$ cd tools

```
[miya@scls tools]$ ls  
Makefile arch auto_submitter batch_builder configure.sh input_builder lib remd_converter  
[miya@scls tools]$
```

\$ sh configure.sh x86_64 gfortran

```
[miya@scls tools]$ sh configure.sh x86_64 gfortran  
[miya@scls tools]$ ls  
Makefile Makefile.machine arch auto_submitter batch_builder configure.sh input_builder lib remd_converter  
[miya@scls tools]$
```

\$ make

\$ make install

\$ cd ..

\$ ls bin

```
[miya@scls rein-v.1.0.4]$ ls bin  
batch_builder input_builder rein remd_convert templates  
[miya@scls rein-v.1.0.4]$
```

全てrein-v.1.0.4/binに格納される

templatesには様々な設定のテンプレートが入っている

```
[miya@scls rein-v.1.0.4]$ ls bin/templates/  
t_archive.sh      t_batch_ricctar    t_namd.inp         t_rein_axis.inp    t_restart_ricc_upc  
t_batch_fx10      t_marble.inp       t_namd_colvar_ang  t_reininp_modify   t_restart_ricctar  
t_batch_ge        t_marble_atom_h    t_namd_colvar_dih  t_restart_ge        t_staging_k_marble  
t_batch_k         t_marble_copyfiles t_namd_colvar_dis  t_restart_k         t_staging_k_namd  
t_batch_nostfx10  t_marble_group     t_namd_copyfiles   t_restart_nostfx10 t_staging_ricc_namd  
t_batch_ricc      t_marble_group_h   t_namd_harmonic    t_restart_num       t_staging_ricctar_namd  
t_batch_ricc_upc  t_namd.colvar      t_rein.inp         t_restart_ricc  
[miya@scls rein-v.1.0.4]$
```

- t_batch : バッチキューのテンプレート
- t_marble, t_namd など : それぞれのMDのインプットのテンプレート
- t_rein : reinの基本設定
- t_restart : リスタート時の動作ルール
- t_staging : スパコンなどで、ファイル転送がある場合の転送ルールを記載

例題を試す

examplesにalanine dipeptideの例が格納されている

remd: 温度レプリカ交換

reus: end-to-end距離のレプリカ交換

mrem: 温度とend-to-end距離の二次元のレプリカ交換

シミュレーションの準備

```
$ cd examples/remd/namd
```

```
[miya@scls namd]$ ls
bb.inp          bb.inp.ricctar  ib.inp.nostfx10  initial.coor  initial.psf  initial.xsc  rein
bb.inp.nostfx10  ib.inp          ib.inp.ricc      initial.pdb   initial.vel  par_all27_prot_lipid.prm
```

ツールを全てコピーする

```
$ cp -r ../../../bin/* . #必要なプログラムをコピー
```

```
$ cp /home/miya/bin/namd2 . # namd2をコピー
```

```
[miya@scls namd]$ ls
batch_builder  bb.inp.ricctar  ib.inp.ricc  initial.psf  input_builder  rein
bb.inp         ib.inp          initial.coor  initial.vel  namd2          remd_convert
bb.inp.nostfx10  ib.inp.nostfx10  initial.pdb  initial.xsc  par_all27_prot_lipid.prm  templates
[miya@scls namd]$
```

Input builderによるreinのインプット作成

```
$ ./input_builder ib.inp.nostfx10
```

```
[miya@scls namd]$ ls
batch_builder  file_coef.d  ib.inp        initial.pdb   namd2         rep.1  rep.6
bb.inp         file_ex.d    ib.inp.nostfx10 initial.psf   par_all27_prot_lipid.prm rep.2  rep.7
bb.inp.nostfx10 file_on.d    ib.inp.ricc   initial.vel   rein          rep.3  rep.8
bb.inp.ricctar file_rank.d  initial.coor   initial.xsc   rein.inp      rep.4  templates
file_axis_ex.d file_value.d initial.inp    input_builder remd_convert  rep.5
```

それぞれのレプリカのディレクトリ(rep.#)とreinのインプットファイル(file_xx.d, rein.inp)、MDのテンプレート(initial.inp)が自動生成される。

batch builderによるreinのインプット作成

```
$ ./batch_builder bb.inp.nostfx10
```

```
[miya@scls namd]$ ls
batch_builder  file_coef.d  ib.inp        initial.pdb   namd2         remd_convert  rep.5
bb.inp         file_ex.d    ib.inp.nostfx10 initial.psf   par_all27_prot_lipid.prm rep.1         rep.6
bb.inp.nostfx10 file_on.d    ib.inp.ricc   initial.vel   rein          rep.2         rep.7
bb.inp.ricctar file_rank.d  initial.coor   initial.xsc   rein.inp      rep.3         rep.8
file_axis_ex.d file_value.d initial.inp    input_builder rein.sh        rep.4         templates
[miya@scls namd]$
```

reinの実行

```
$ pjsub rein.sh
```

jobの状態チェック

```
$ pjstat
```

```
[miya@scls namd]$ pjstat
```

	ACCEPT	QUEUED	STGIN	READY	RUNING	RUNOUT	STGOUT	HOLD	ERROR	TOTAL
	0	0	0	0	1	0	0	0	0	1
s	0	0	0	0	1	0	0	0	0	1

JOB_ID	JOB_NAME	MD	ST	USER	START_DATE	ELAPSE_LIM	NODE_REQUIRE
14306	rmd2	NM	RUN	miya	02/25 15:45:52	0000:20:00	10

```
$ pjstat -A
```

jobの出力状態チェック

```
$ pjcat -o #jobnumber
```

```
$ pjstat -e #jobnumber
```

```
[miya@scls namd]$ pjstat
```

	ACCEPT	QUEUED	STGIN	READY	RUNING	RUNOUT	STGOUT	HOLD	ERROR	TOTAL
	0	0	0	0	0	0	0	0	0	0
s	0	0	0	0	0	0	0	0	0	0

```
[miya@scls namd]$ ls
```

batch_builder	file_ex.d	ib.inp	initial.psf	rein	rep.3	replica.trj
bb.inp	file_on.d	ib.inp.nostfx10	initial.vel	rein.inp	rep.4	rmd2.e14306
bb.inp.nostfx10	file_out_coef.d	ib.inp.ricc	initial.xsc	rein.sh	rep.5	rmd2.i14306
bb.inp.ricctar	file_out_rank.d	initial.coor	input_builder	remd_convert	rep.6	rmd2.o14306
file_axis_ex.d	file_rank.d	initial.inp	namd2	rep.1	rep.7	templates
file_coef.d	file_value.d	initial.pdb	par_all27_prot_lipid.prm	rep.2	rep.8	

```
[miya@scls namd]$
```

batch builderによるデータ処理と、次のインプット作成

\$./batch_builder bb.inp.nostfx10 restart

```
[miya@scls namd]$ ls
```

batch_builder	file_ex.d	ib.inp.ricc	initial.xsc	rein.sh	rep.5	rmd2.i14306
bb.inp	file_on.d	initial.coor	input_builder	remd_convert	rep.6	rmd2.o14306
bb.inp.nostfx10	file_rank.d	initial.inp	namd2	rep.1	rep.7	templates
bb.inp.ricctar	file_value.d	initial.pdb	par_all27_prot_lipid.prm	rep.2	rep.8	
file_axis_ex.d	ib.inp	initial.psf	rein	rep.3	results	
file_coef.d	ib.inp.nostfx10	initial.vel	rein.inp	rep.4	rmd2.e14306	

```
[miya@scls namd]$
```



```
[miya@scls namd]$ ls results/
```

```
result0000001_0000010
```

```
[miya@scls namd]$ ls results/result0000001_0000010/
```

file_axis_ex.d	file_ex.d	file_out_coef.d	file_rank.d	rein.inp	rep.1	rep.3	rep.5	rep.7	replica.trj
file_coef.d	file_on.d	file_out_rank.d	file_value.d	rein.sh	rep.2	rep.4	rep.6	rep.8	

```
[miya@scls namd]$
```

再度reinの実行

\$ pjsup rein.sh

トラジェクトリの整理

```
$ ./remd_convert -h ctrl > rc.inp
```

rc.inpファイルの変更

```
# control parameters in remd_convert
```

```
[INPUT]
```

```
psffile          = initial.psf
reffile          = initial.pdb
trj_format       = DCD           # (DCD/MARBLE/PDB)
```

```
directory001     = ./results/result0000001_0000010
directory002     = ./results/result0000011_0000020
directory003     = ./results/result0000021_0000030
```

```
repeat001        = 3           (→ 1に変更)
trj_filename001  = rep.{rep_no}/rep.{rep_no}.{step}.dcd
energy_trj001    = replica.trj
```

```
energy_column    =              # column of output energy, ex. = 1 2-4, empty: all column
```

```
[OUTPUT]
```

```
out_pdbfile      = out.pdb
trj_filename     = out.{type}.{number}.{extension}
trj_format       = DCD           # (DCD/MARBLE/AMBER/TINKER/PDB/GROMACS)
```

```
[CONVERT]
```

```
type             = REPLICA      # (REPLICA/CONDITION)
rank             =              # ex. = 1 2-8, empty: output all rank
```

```
[SELECTION]
```


remd_convertの実行

```
$ ./remd_convert rc.inp
```

```
[miya@scls namd]$ ls
batch_builder  ib.inp          namd2           out.replica.5.crdtrj  rein           rep.7
bb.inp         ib.inp.nostfx10 out.pdb         out.replica.5.enetrj  rein.inp       rep.8
bb.inp.nostfx10 ib.inp.ricc     out.replica.1.crdtrj out.replica.5.enetrj  rein.sh        results
bb.inp.ricctar  initial.coor    out.replica.1.enetrj out.replica.6.crdtrj  remd_convert   rmd2.e14306
file_axis_ex.d  initial.inp     out.replica.2.crdtrj out.replica.6.enetrj  rep.1          rmd2.i14306
file_coef.d     initial.pdb     out.replica.2.enetrj out.replica.7.crdtrj  rep.2          rmd2.o14306
file_ex.d       initial.psf     out.replica.3.crdtrj out.replica.7.enetrj  rep.3          templates
file_on.d       initial.vel     out.replica.3.enetrj out.replica.8.crdtrj  rep.4
file_rank.d     initial.xsc     out.replica.4.crdtrj out.replica.8.enetrj  rep.5
file_value.d    input_builder   out.replica.4.enetrj rc.inp              rep.6
[miya@scls namd]$
```

out.replica.#.crdtrj : replicaのdcd file

out.replica.#.enetrj : energyなどのトラジェクトリ

番号はレプリカ。更に、[CONVERT] typeを"REPLICA"から"CONDITION"に変更するとコンディションに対するトラジェクトリが得られる。これらのデータから様々な解析ができる。

是非、VMDなどの画像ソフトウェアを使って動画などを見て下さい。

PCでの簡単な実行例

no spawn option 3レプリカの場合

インストール

```
$ cd src
$ ./configure.sh x86_64 gfortran local
$ make; make install
$ cd ../tools
$ ./configure.sh x86_64 gfortran
$ make; make install
```

必要なファイルのコピー

```
$ cd ../../test
$ cp -r ../rein/bin/* .
$ cp ../rein/example/input/namd/*
予めバイナリ形式のNAMDをnamd ディレクトリへ
```

ダウンロード

```
$ cp ../namd/Linux_x86_64/namd2 .
$ cp ../namd/Linux_x86_64/charmrun .
```

```
<ib.inp>
rein_input      = rein.inp    # rein input file name
spawn = no
openmp_rein = yes
```

```
$ ./input_builder ib.inp > ib.out
```

```
<bb.inp>
rein_input      = rein.inp    # REIN input file name
spawn = no
openmp_rein = yes
[BATCH]
batchfile_name   = rein.sh
job_scheduler    = GE
queue_name       = all.q
```

```
num_threads_md   = 1
num_mpirproc_md  = 1
cpu_time         = 01:00:00
num_core_cpu     = 1
num_mpirproc_rein = 3
```

```
$ ./batch_builder bb.inp > bb.out
```

```
$ sh ../rein.sh > rein.out1 &
$ ./batch_builder bb.inp restart > bb.out2
$ sh ../rein.sh > rein.out2 &
```

参考情報

replicaの温度 generator

Van Der Spoel

<http://folding.bmc.uu.se/remd/>

WHAM: Alan Grossfield

<http://membrane.urmc.rochester.edu/content/wham>

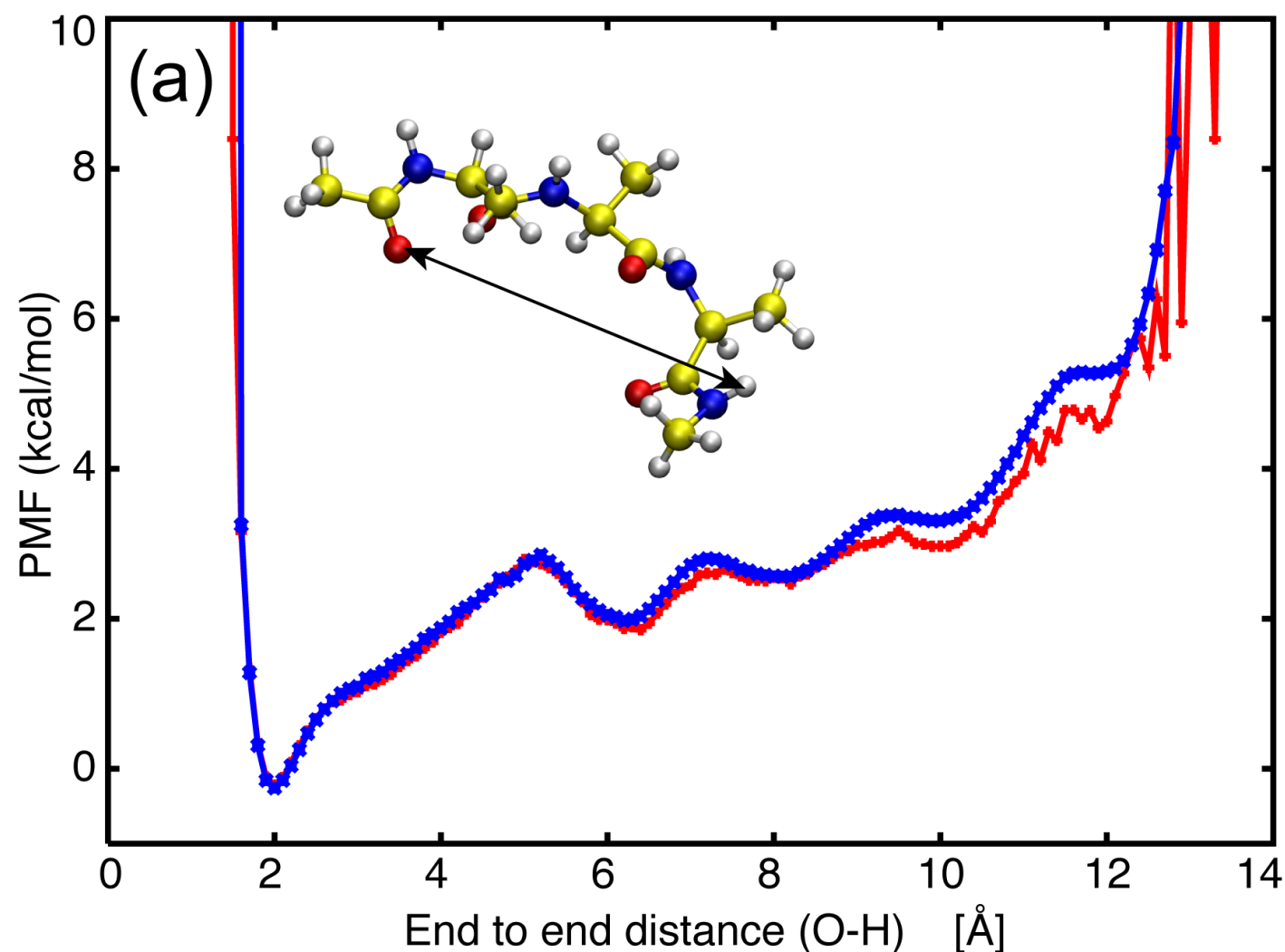
Potential Mean Force(PMF):

自由エネルギーを反応座標に沿って記述したもの

確率密度分布と関係がある

$$\text{PMF}(x) = -k_B T \ln (P(x))$$

確率密度分布はサンプリングから得られる



Red(REMD):

1D-16 replicas-T-REMD

Blue(REUS2):

2D-4 x 4 (16) replicas

REUS-T-REMD

Exchange steps: 50,000

Exchange: every 400 ps