GROMACS実習

独立行政法人理化学研究所
HPCI計算生命科学推進プログラム

チーム員　波内 良樹

yoshiki.namiuchi@riken.jp

実習内容について

この実習は、下記のGROMACS チュートリアルを元に構成しています。

GROMACS Tutorial

Lysozyme in Water

Justin Lemkul
Department of Biochemistry, Virginia Tech

http://www.bevanlab.biochem.vt.edu/Pages/Personal/justin/gmx-tutorials/lysozyme/

SCLS計算機システムのGROMACS

SCLS計算機システムにインストールされているGROMACS

GROMACS 4.5.6	GROMACS 4.6.2
• 倍精度 • MPI	• 倍精度 • MPI • OpenMP

※ GROMACSのコマンド名に、サフィックス「`_mpi_d`」が付きます。
※ すべてのコマンドで並列処理が可能な訳ではありません。
（`pdb2gmx`は逐次、`mdrun`はMPI並列、etc...）

GROMACSを使用するために

ジョブスクリプトに以下の記述を追加（環境変数の設定）

【GROMACS 4.5.6】

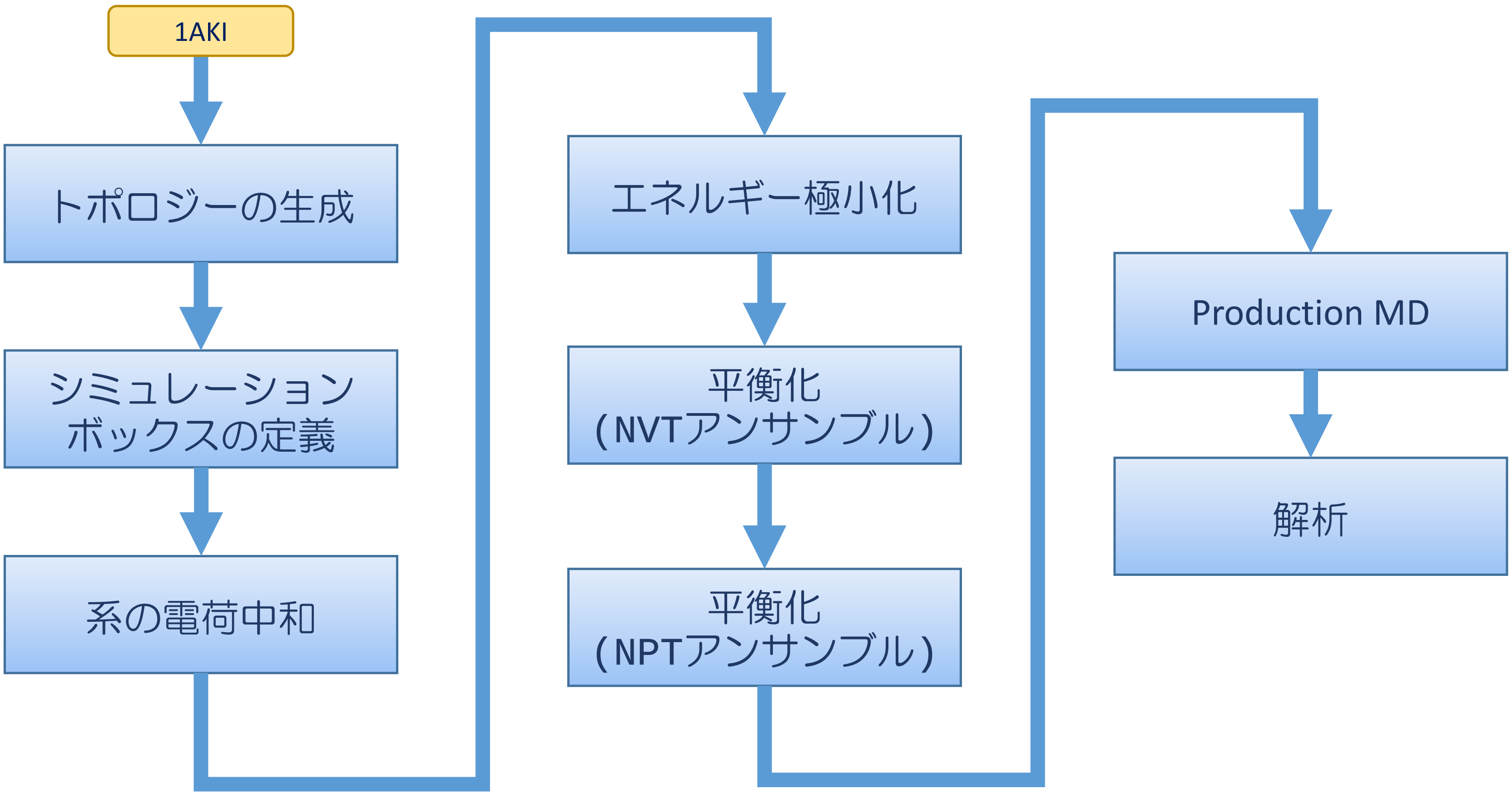
```
module load Gromacs/4.5.6
```

【GROMACS 4.6.2】

```
module load Gromacs/4.6.2
```

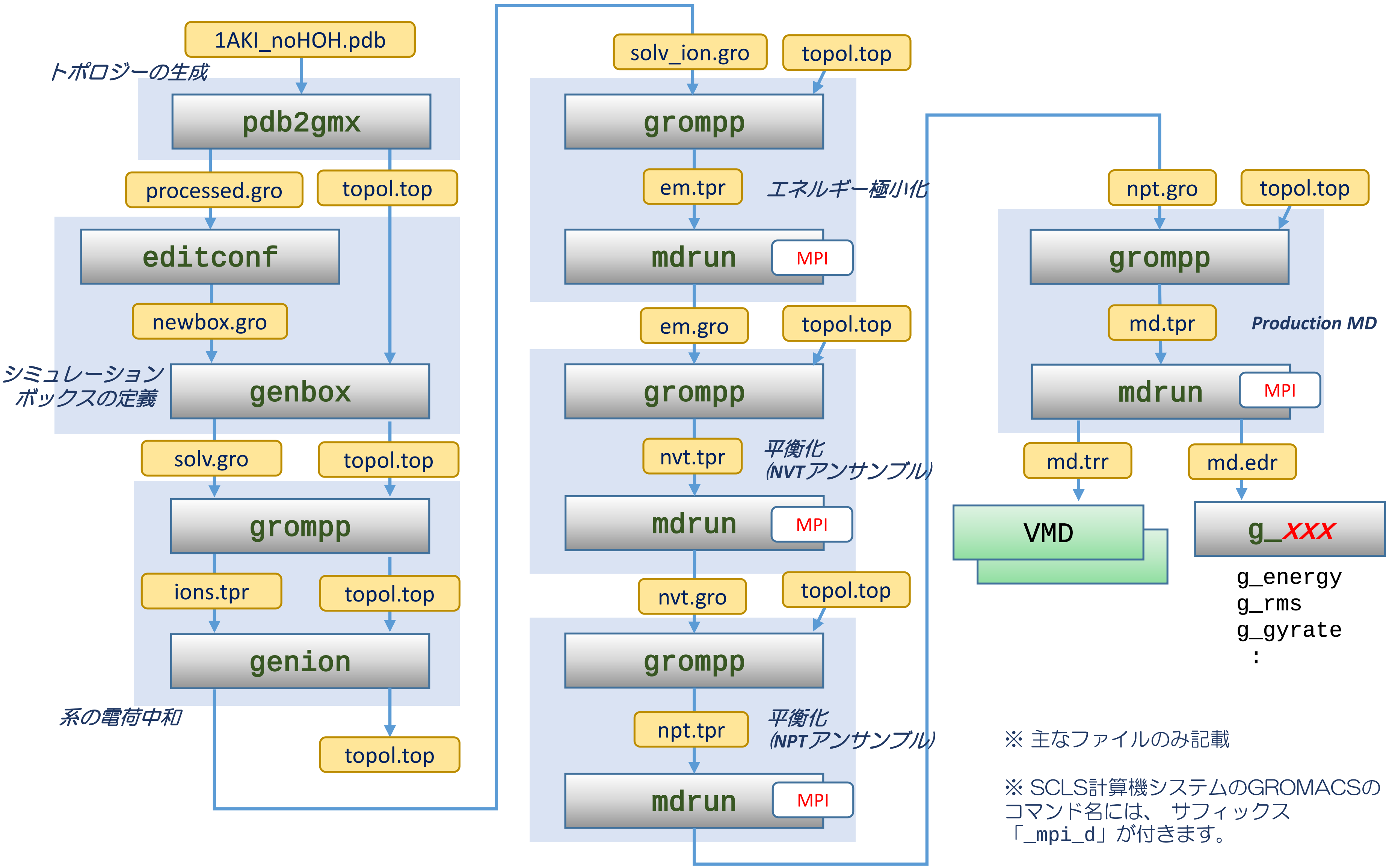
実習のながれ

PDB(Protein Data Bank)で公開されているニワトリ卵白リゾチーム(1AKI)を使用



PDB(Protein Data Bank) … http://www.rcsb.org/

GROMACSコマンドの実行のながれ



※ 主なファイルのみ記載

※ SCLS計算機システムのGROMACSのコマンド名には、 サフィックス「_mpi_d」が付きます。

実習の準備（1）

実習で使用するGROMACS

GROMACS 4.6.2

実習環境の準備

```
[user1@scls ~]$ cp -r ~namiuchi/lec/lec_gromacs .
[user1@scls ~]$ cd lec_gromacs
[user1@scls lec_gromacs]$ ls -lF
total 4
drwxrwxr-x 2 user1 group1   62 Aug  1 15:55 full/
drwxrwxr-x 2 user1 group1  122 Aug  1 15:48 mdp/
drwxrwxr-x 2 user1 group1   27 Jul 25 16:38 pdb/
drwxrwxr-x 2 user1 group1 4096 Aug  1 16:16 script/
[user1@scls lec_gromacs]$
```

- `cp -r ~namiuchi/lec/lec_gromacs .`：実習用ファイルを格納したディレクトリlec_gromacsをカレントディレクトリにコピー
- `cd lec_gromacs`：カレントディレクトリを、コピーしたlec_gromacsディレクトリに変更
- `ls -lF`：カレントディレクトリの内容を表示
- full/：事前に用意した結果を格納しているディレクトリ
- mdp/：MDパラメータファイルを格納しているディレクトリ
- pdb/：PDBファイルを格納しているディレクトリ
- script/：ジョブスクリプトを格納しているディレクトリ

実習の準備 (2)

実習で使用するジョブタイプ

処理	ジョブタイプ	処理	ジョブタイプ
トポロジーファイルの生成	会話型ジョブ	平衡化 (NVTアンサンブル)	バッチジョブ
シミュレーションボックスの定義	会話型ジョブ	平衡化 (NPTアンサンブル)	バッチジョブ
系の電荷中和	会話型ジョブ	Production MD	バッチジョブ
エネルギー極小化	バッチジョブ	解析 (g_energy等)	会話型ジョブ

注意点

※ GROMACSコマンド実行時に、以下のメッセージが表示されます。

```
jwe1050i-w The hardware barrier couldn't be used and continues processing using the software barrier.
```

FX10に実装されているハードウェアバリア機能が利用できない場合に表示されます。(FLIB_FASTOMPをFALSEに設定しているため)
動作に問題はありません。

会話型ジョブの開始と終了

```
[user1@scls lec_gromacs]$ pjsub --interact –L “node=1” --mpi "proc=1"
[INFO] PJM 0000 pjsub Job 6282 submitted.
[INFO] PJM 0081 .connected.
[INFO] PJM 0082 pjsub Interactive job 6282 started.
[user1@a01-008 lec_gromacs]$ module load Gromacs/4.6.2
[user1@a01-008 lec_gromacs]$ export FLIB_FASTOMP=FALSE

        ⋮

        ⋮

[user1@a01-008 lec_gromacs]$ exit
[INFO] PJM 0083 pjsub Interactive job 6282 completed.
[user1@scls lec_gromacs]$
```

会話型ジョブの投入

GROMACS環境変数の設定

GROMACSコマンドの実行

会話型ジョブの終了

トポロジーの生成


```
$ pdb2gmx_mpi_d -f pdb/1AKI_noHOH.pdb -o processed.gro
 :-)  G  R  O  M  A  C  S  (-:

        ⋮

Select the Force Field:
From '/usr/local/gromacs-4.6.2/share/gromacs/top':
15: OPLS-AA/L all-atom force field (2001 aminoacid dihedrals)
15
        ⋮

Select the Water Model:
 5: SPC/E  extended simple point charge
5
        ⋮

Total charge 8.000 e
Writing topology
        ⋮

$
```

力場の指定

水モデルの指定

系の電荷

トポロジーの生成 [実行結果の確認]

トポロジーファイル

```
$ less topol.top

      ⋮

; Include forcefield parameters
#include "oplsaa.ff/forcefield.itp"

[ moleculetype ]
; Name            nrexcl
Protein_chain_A     3

[ atoms ]

      ⋮
```

構造ファイル

```
$ less processed.gro
LYSOZYME
 1960
    1LYS      N    1   3.537   2.234  -1.198
    1LYS     H1    2   3.612   2.288  -1.236

      ⋮
```

トポロジーの生成 [実行結果の確認]

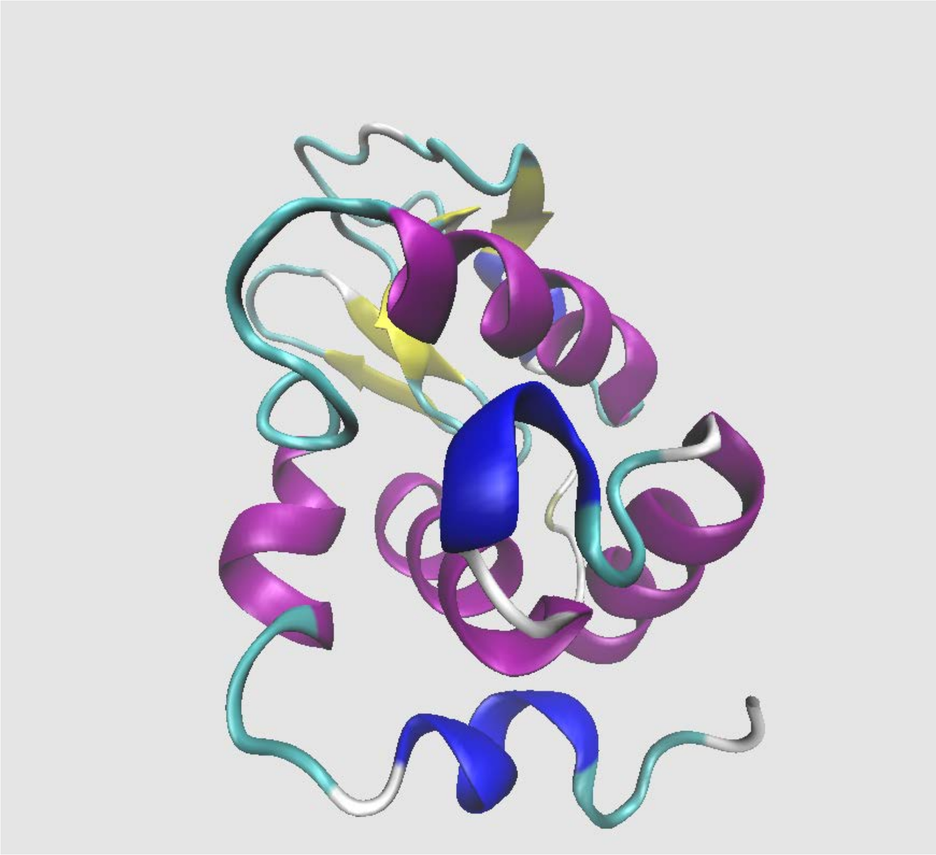
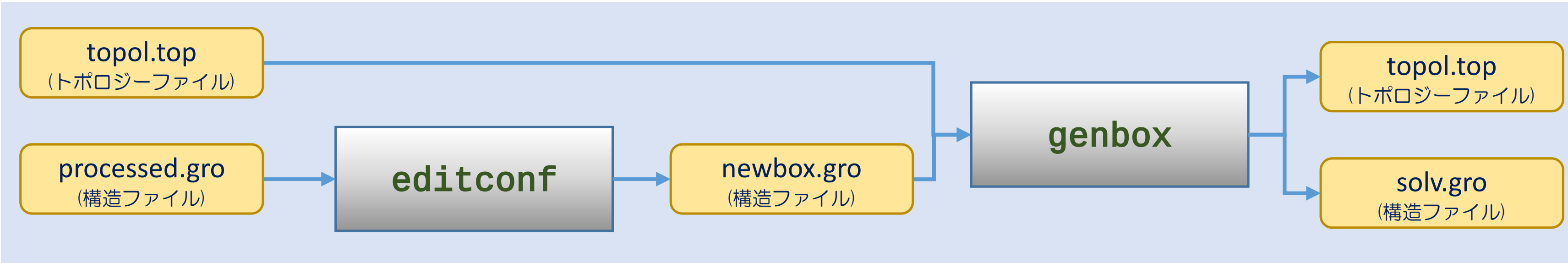


図１． processed.groをVMDで表示

シミュレーションボックスの定義




```
$ editconf_mpi_d -f processed.gro –c -bt cubic -d 1.0 -o newbox.gro
:-)  G  R  O  M  A  C  S  (-:

          ⋮

new center       :  3.505  3.505  3.505 (nm)
new box vectors :  7.010  7.010  7.010 (nm)
new box angles  :  90.00  90.00  90.00 (degrees)
new box volume  : 344.48               (nm^3)

$ genbox_mpi_d -cp newbox.gro -cs spc216.gro -p topol.top -o solv.gro
:-)  G  R  O  M  A  C  S  (-:

          ⋮

Grid: 16 x 16 x 16 cells
Adding line for 10832 solvent molecules to topology file (topol.top)
$
```

ボックスの形状 = 立方体
ボックスと溶質の最短距離 = 1.0nm

溶媒 = 水（spc216.gro）

シミュレーションボックスの定義 [実行結果の確認]

トポロジーファイル

```
$ diff ¥#topol.top.1# topol.top
18402c18402
< LYSOZYME
---
> LYSOZYME in water
18406a18407
> SOL             10832
$ less topol.top
:18402G
```

溶媒(水)を追加

構造ファイル

```
$ ls -l processed.gro newbox.gro solv.gro
-rw-rw-r-- 1 user1 group1   88246 Jul 11 17:22 newbox.gro
-rw-rw-r-- 1 user1 group1   88246 Jul 11 17:00 processed.gro
-rw-rw-r-- 1 user1 group1 1550566 Jul 11 17:23 solv.gro
$
```

【editconf】
processed.gro → newbox.gro … ボックスの中心になるように座標を更新

【genbox】
newbox.gro → solv.gro … 水を追加

シミュレーションボックスの定義 [実行結果の確認]

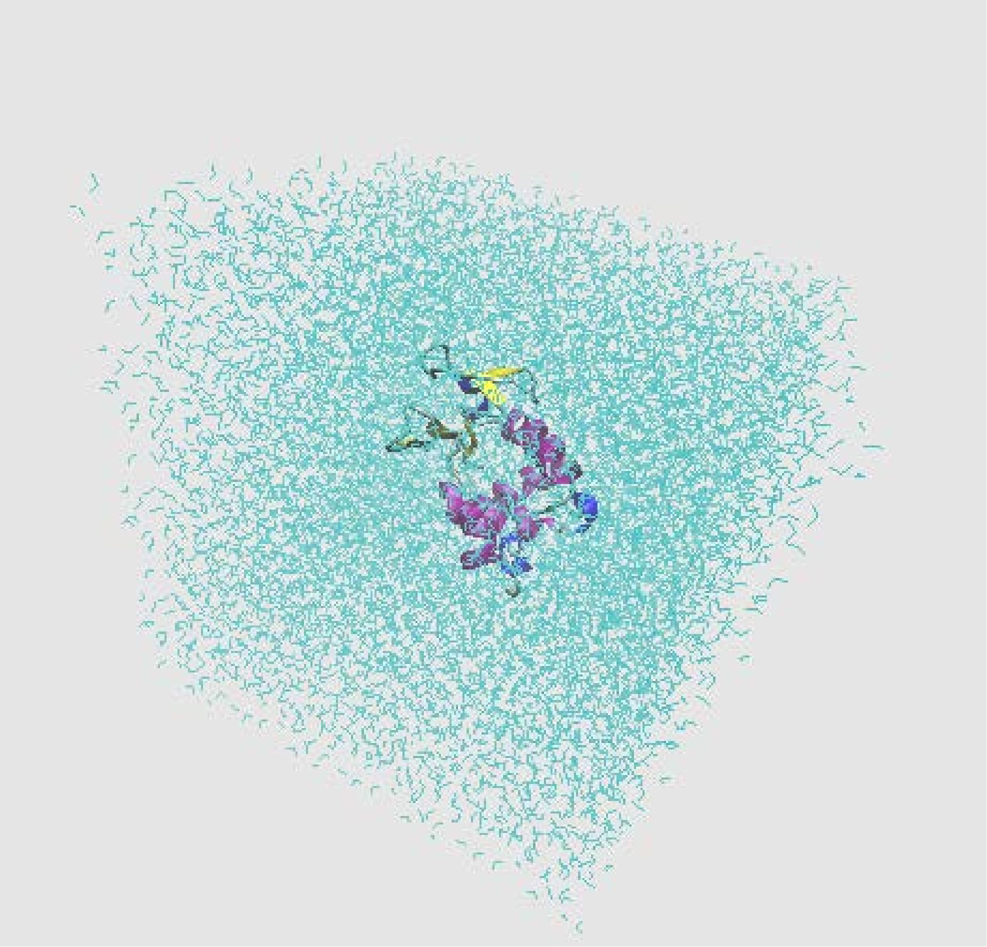
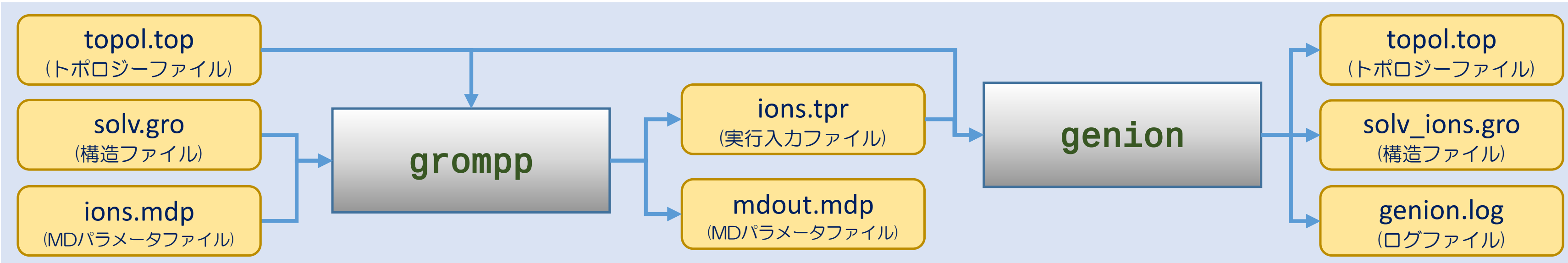


図2. solv.groをVMDで表示

系の電荷中和




```
$ grompp_mpi_d -f mdp/ions.mdp -c solv.gro -p topol.top -o ions.tpr
:-)  G  R  O  M  A  C  S  (-:

         :

Calculating fourier grid dimensions for X Y Z
Using a fourier grid of 60x60x60, spacing 0.117 0.117 0.117
This run will generate roughly 3 Mb of data

$ mpiexec –stdin script/genion_in.txt genion_mpi_d -s ions.tpr -p topol.top
 -nname CL -nn 8 -o solv_ions.gro
:-)  G  R  O  M  A  C  S  (-:

         :

Select a continuous group of solvent molecules
Group    13 (             SOL) has 32496 elements
Select a group: 13

$
```

イオンと入れ替える対象として、溶媒(水)を指定
(script/genion_in.txtで13を指定)

ジョブ実行のエラーメッセージが表示されるが、処理は完了

系の電荷中和 [実行結果の確認]

トポロジーファイル

```
$ diff ¥#topol.top.2# topol.top
18407c18407,18408
< SOL             10832
---
> SOL         10824
> CL              8
$ less topol.top
:18407G
```

水分子を8個削除して、CLを8個追加

構造ファイル

```
$ less solv_ions.gro
:G
        ⋮
10953SOL    HW234432   5.585   6.577   5.918
10954CL      CL34433   0.980   2.978   0.295
10955CL      CL34434   0.865   4.072   3.919
10956CL      CL34435   2.476   1.984   1.979
10957CL      CL34436   2.674   5.310   0.687
10958CL      CL34437   5.182   0.735   0.728
10959CL      CL34438   6.302   0.565   1.708
10960CL      CL34439   6.356   3.192   2.163
10961CL      CL34440   6.452   4.178   0.642
```

水分子を8個削除して、CLを8個追加

系の電荷中和 [実行結果の確認]

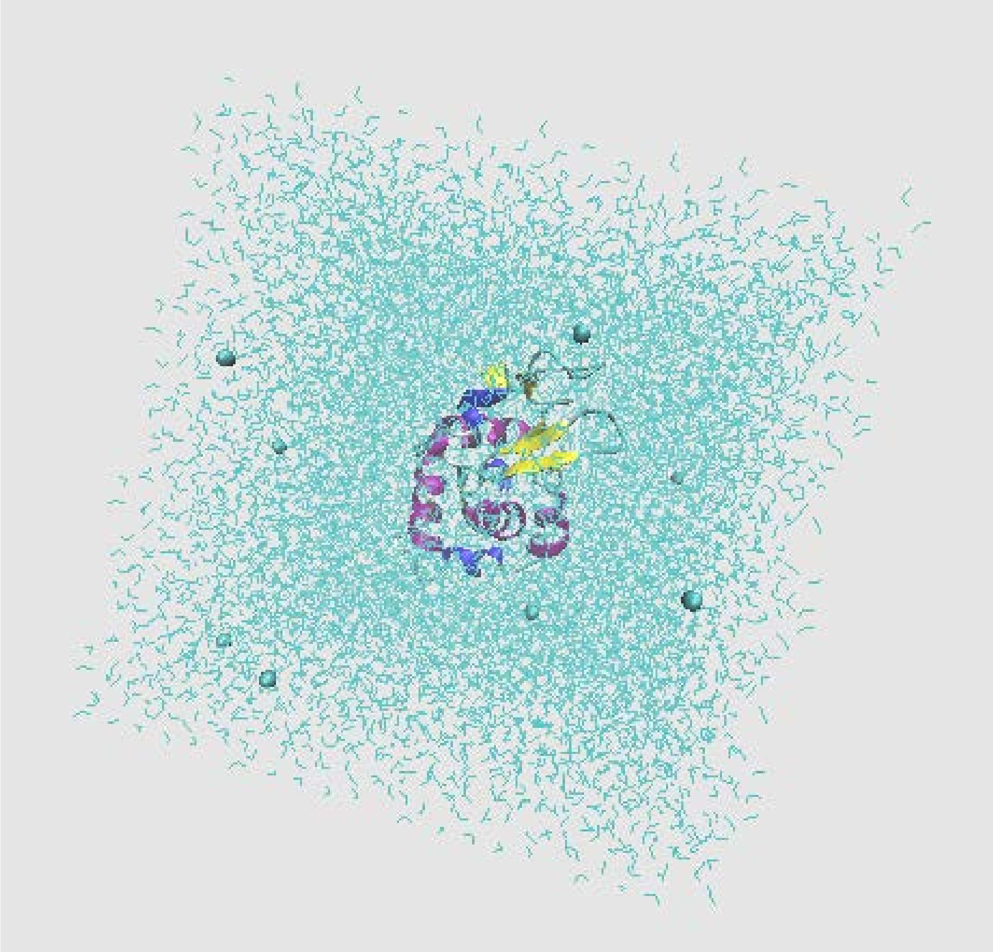
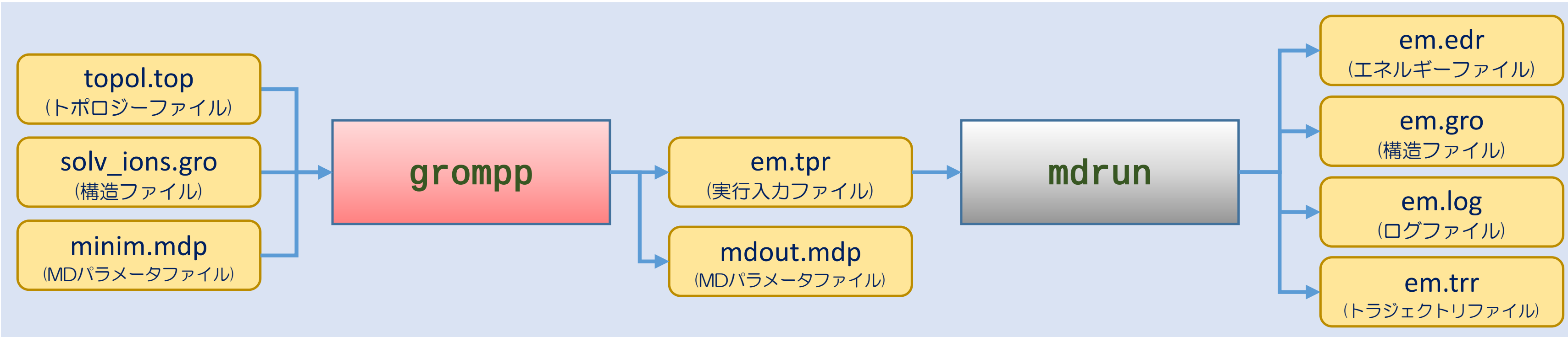


図3. solv_ions.groをVMDで表示

エネルギー極小化 [grompp]



MDパラメータファイル

```
[user1@scls lec_gromacs]$ less mdp/minim.mdp
```

integrator = steep	最急降下法 (Steepest descent)
emtol = 1000.0	Maximum forceが1000 KJ/mol/nmを下回ったら終了

エネルギー極小化 [grompp]

ジョブスクリプト（逐次ジョブ）

```
[user1@scls lec_gromacs]$ cat script/grompp_minim.sh
#!/bin/sh
#------ pjsub options ------#
#PJM -L "rscgrp=small"
#PJM -L "node=1"
#PJM -L "elapse=10:00"
#PJM -j
#------ Program Execution --------#
module load Gromacs/4.6.2

export FLIB_FASTOMP=FALSE

grompp_mpi_d -f mdp/minim.mdp -c solv_ions.gro -p topol.top -o em.tpr
```

リソースグループ「small」
使用ノード数「1」
最大経過時間「10分」
標準エラー出力を標準出力に向ける

GROMACS環境変数の設定

gromppの実行

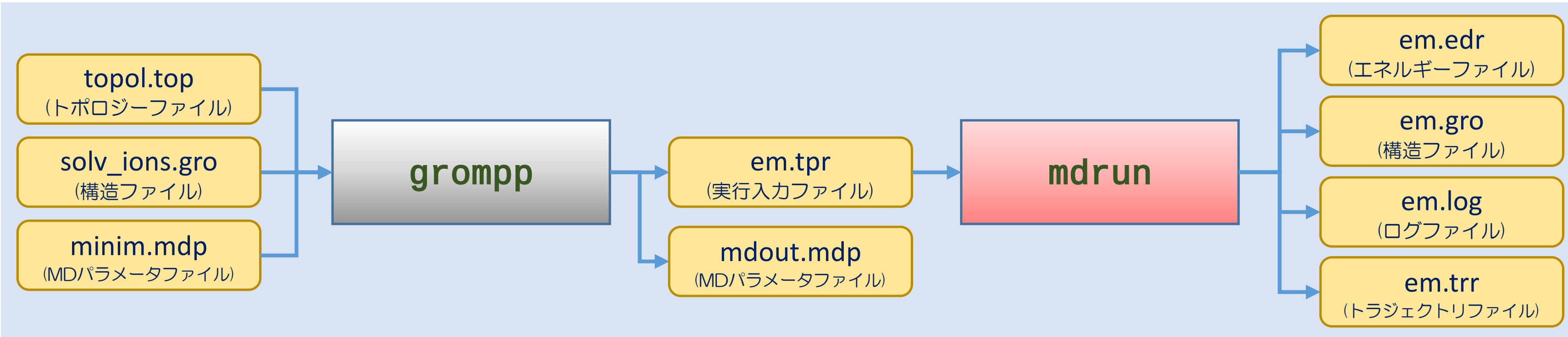
ジョブの投入と状態確認

```
[user1@scls lec_gromacs]$ pjsub script/grompp_minim.sh
[INFO] PJM 0000 pjsub Job 6293 submitted.
[user1@scls lec_gromacs]$ pjstat
```

ジョブ実行結果

```
[user1@scls lec_gromacs]$ less grompp_minim.sh.oXXXX
```

エネルギー極小化 [mdrun]



ジョブスクリプト (MPI並列ジョブ)

```
[user1@scls lec_gromacs]$ cat script/mdrun_minim.sh
#!/bin/sh
#------ pjsub options ------#
#PJM -L "rscgrp=small"
#PJM -L "node=1"
#PJM --mpi "proc=16"
#PJM -L "elapse=10:00"
#PJM -j
#------ Program Execution --------#
module load Gromacs/4.6.2

export FLIB_FASTOMP=FALSE

mpiexec mdrun_mpi_d -v -deffnm em
```

リソースグループ「small」
使用ノード数「1」
プロセス数「16」
最大経過時間「10分」
標準エラー出力を標準出力に向ける

GROMACS環境変数の設定

mdrunの実行

エネルギー極小化 [mdrun]

ジョブの投入と状態確認

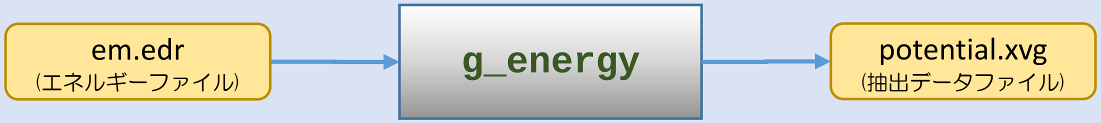
```
[user1@scls grolec]$ pjsub script/mdrun_minim.sh
[INFO] PJM 0000 pjsub Job 6293 submitted.
[user1@scls grolec]$ pjstat

  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
6293       mdrun_mini NM RUN user1    07/26 17:37:18  0000:10:00 1
```

ジョブ実行結果

```
[user1@scls lec_gromacs]$ less em.log
         ⋮

Steepest Descents converged to Fmax < 1000 in 884 steps
Potential Energy  = -6.03680902877939e+05
Maximum force     =  9.66569539375539e+02 on atom 736
Norm of force     =  2.11987691589411e+01
         ⋮
```

極小化の結果

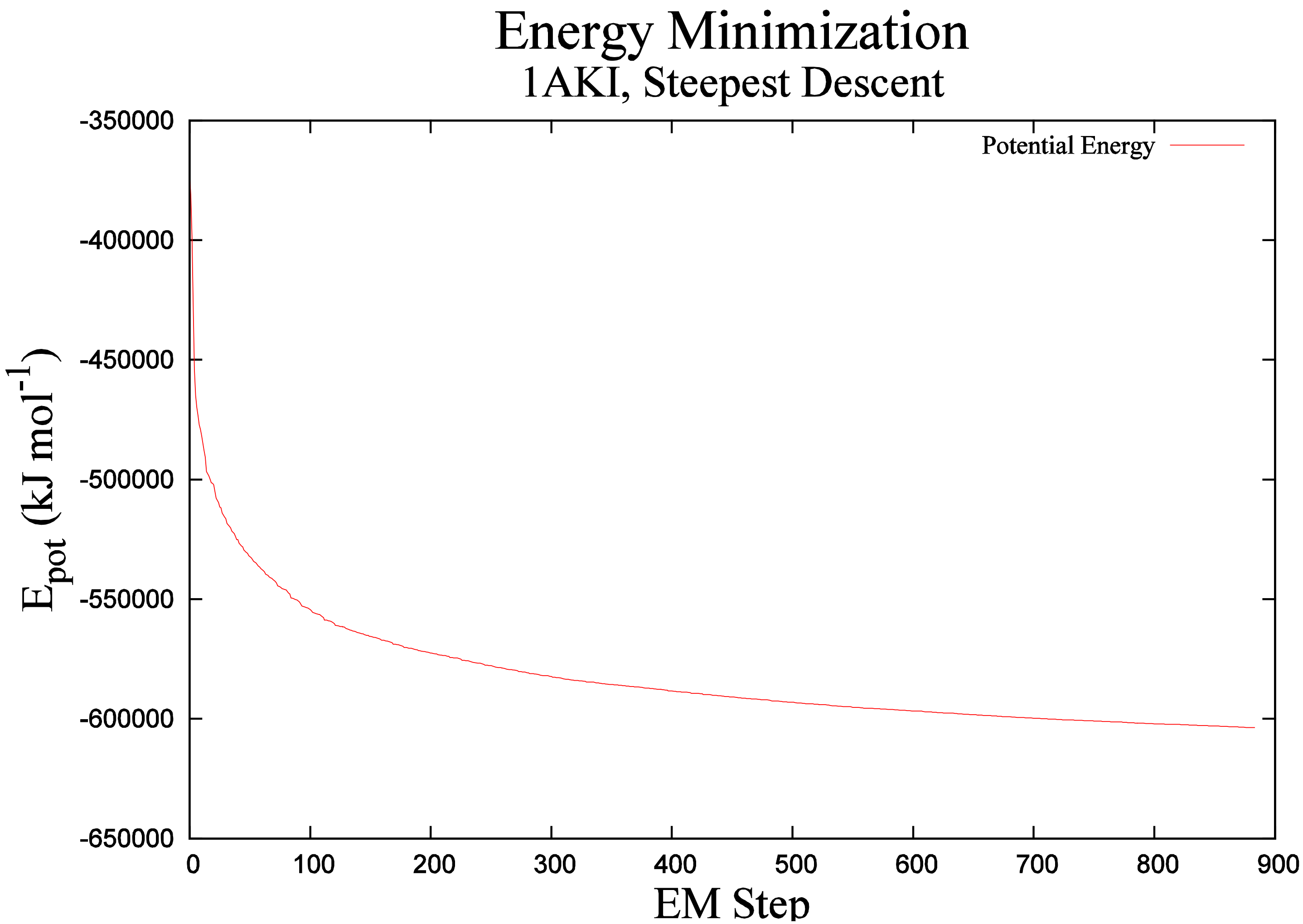
エネルギー極小化 [実行結果の解析]




```
[user1@scls lec_gromacs]$ pjsub --interact
[INFO] PJM 0000 pjsub Job 6282 submitted.
[INFO] PJM 0081 .connected.
[INFO] PJM 0082 pjsub Interactive job 6282 started.
[user1@a01-008 lec_gromacs]$ module load Gromacs/4.6.2
[user1@a01-008 lec_gromacs]$ g_energy_mpi_d –f em.edr –o potential.xvg
 :-)  G  R  O  M  A  C  S  (-:


-------------------------------------------------------------------
  1  Bond             2  Angle            3  Proper-Dih.      4  Ryckaert-Bell.
  5  LJ-14            6  Coulomb-14       7  LJ-(SR)          8  Coulomb-(SR)
  9  Coul.-recip.    10  Potential       11  Pressure        12  Vir-XX
 13  Vir-XY          14  Vir-XZ          15  Vir-YX          16  Vir-YY
 17  Vir-YZ          18  Vir-ZX          19  Vir-ZY          20  Vir-ZZ
 21  Pres-XX         22  Pres-XY         23  Pres-XZ         24  Pres-YX
 25  Pres-YY         26  Pres-YZ         27  Pres-ZX         28  Pres-ZY
 29  Pres-ZZ         30  #Surf*SurfTen   31  Mu-X            32  Mu-Y
 33  Mu-Z            34  T-rest
10 0

[user1@a01-008 lec_gromacs]$ exit
```

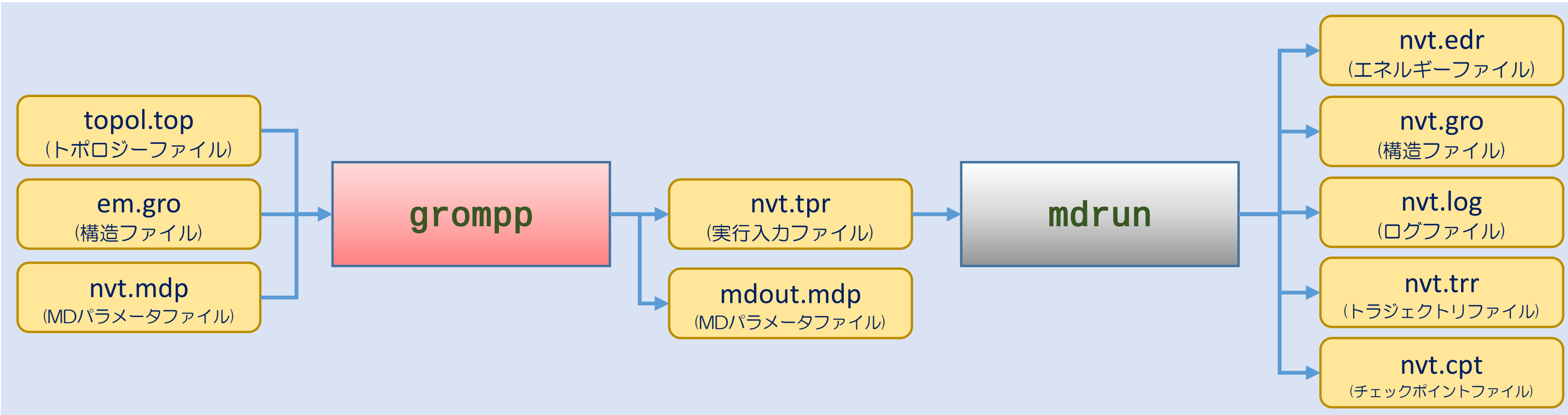
10番のPotentialを選択、0は入力終了

エネルギー極小化 [実行結果の解析]



グラフ1. potential.xvgをgnuplotで表示

平衡化 - NVTアンサンブル [grompp]



MDパラメータファイル

```
[user1@scls lec_gromacs]$ less mdp/nvt.mdp
```

`tcoupl = V-rescale`	速度スケーリング法
`pcoupl = no`	Pressure couplingはなし
`gen_vel = yes`	初期速度を生成
`nsteps = 5000`	実行時間は10ps (2fs x 5000) ※実習のために短く設定しています

平衡化 - NVTアンサンブル [grompp]

ジョブスクリプト（逐次ジョブ）

```
[user1@scls lec_gromacs]$ cat script/grompp_nvt.sh
#!/bin/sh
#------ pjsub option --------#
#PJM -L "rscgrp=small"
#PJM -L "node=1"
#PJM -L "elapse=10:00"
#PJM -j
#------ Program execution --------#
module load Gromacs/4.6.2

export FLIB_FASTOMP=FALSE

grompp_mpi_d -f mdp/nvt.mdp -c em.gro -p topol.top -o nvt.tpr
```

リソースグループ「small」
使用ノード数「1」
最大経過時間「10分」
標準エラー出力を標準出力に向ける

GROMACS環境変数の設定

gromppの実行

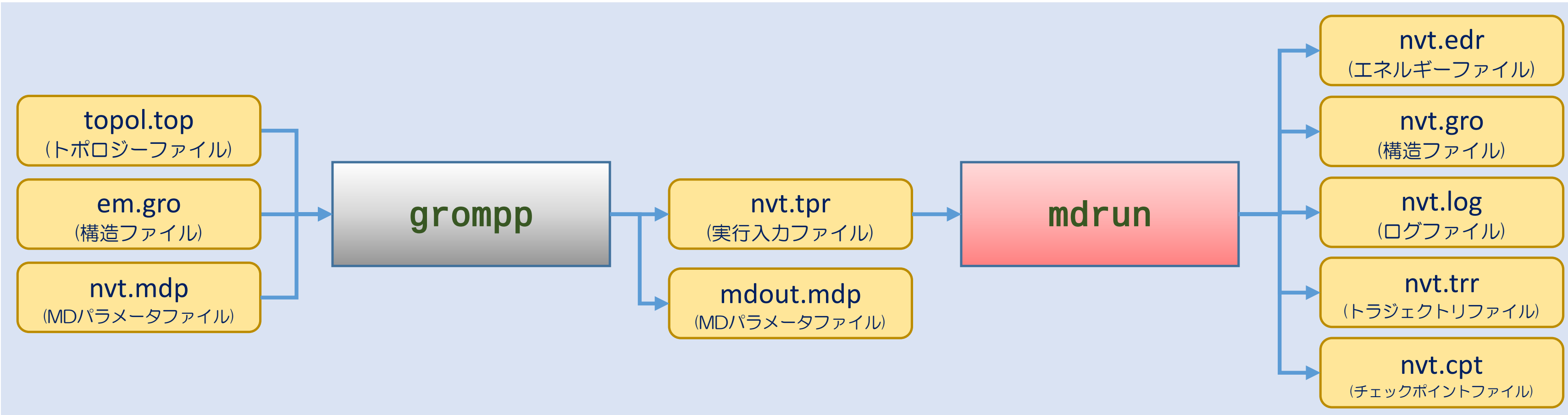
ジョブ投入

```
[user1@scls grolec]$ pjsub script/grompp_nvt.sh
[INFO] PJM 0000 pjsub Job 6293 submitted.
[user1@scls grolec]$ pjstat
```

ジョブ実行結果

```
[user1@scls lec_gromacs]$ less grompp_nvt.sh.oXXXX
```

平衡化 - NVTアンサンブル [mdrun]



ジョブスクリプト（MPI並列ジョブ）

```
[user1@scls lec_gromacs]$ cat script/mdrun_nvt.sh
#!/bin/sh
#------ pjsub options ------#
#PJM -L "rscgrp=small"
#PJM -L "node=2"
#PJM --mpi "proc=32"
#PJM --mpi "rank-map-bychip"
#PJM -L "elapse=10:00"
#PJM -j
#------ Program Execution ------#
module load Gromacs/4.6.2

export FLIB_FASTOMP=FALSE

mpiexec mdrun_mpi_d -v -deffnm nvt
```

リソースグループ「small」
使用ノード数「2」
プロセス数「32」
各ノードへのプロセスの割り振りの指定
最大経過時間「10分」
標準エラー出力を標準出力に向ける

GROMACS環境変数の設定

mdrunの実行

平衡化 - NVTアンサンブル [mdrun]

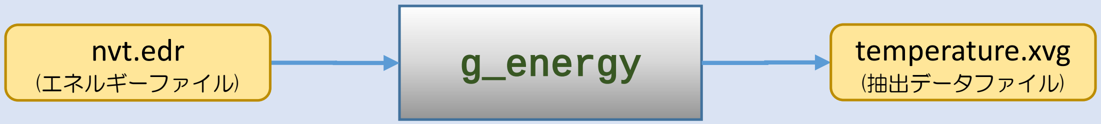
ジョブの投入と状態確認

```
[user1@scls lec_gromacs]$ pjsub script/mdrun_nvt.sh
[INFO] PJM 0000 pjsub Job 6293 submitted.
[user1@scls lec_gromacs]$ 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
6293       mdrun_nvt. NM RUN user1    07/26 17:37:18  0000:10:00 2
```

ジョブ実行結果

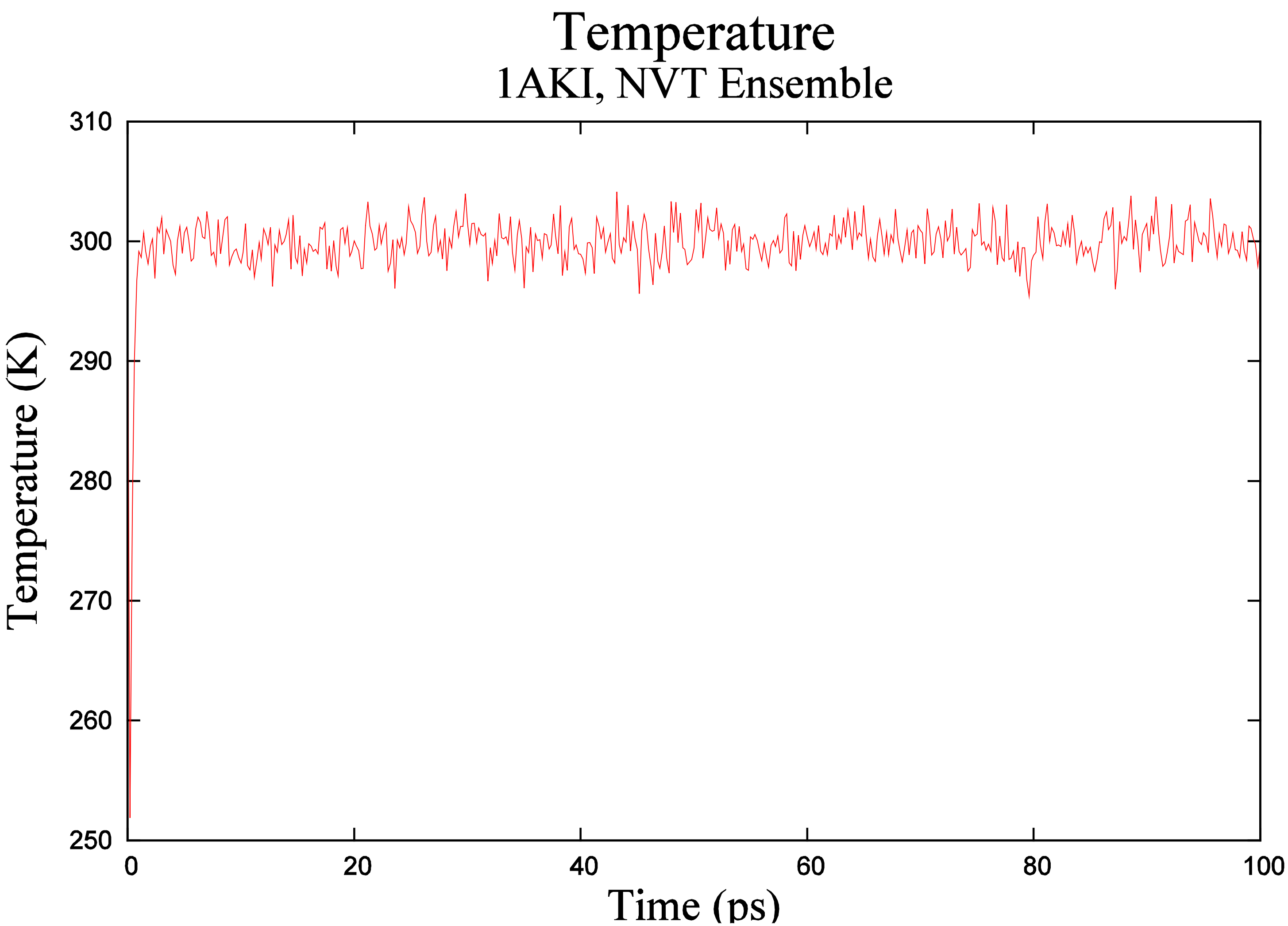
```
[user1@scls lec_gromacs]$ less nvt.log

         ⋮

               Core t (s)   Wall t (s)        (%)
       Time:    13805.400      575.713     2398.0
                 (ns/day)    (hour/ns)
Performance:       15.008        1.599
Finished mdrun on node 0 Fri Jul 26 17:47:17 2013
```

平衡化 – NVTアンサンブル [実行結果の解析]



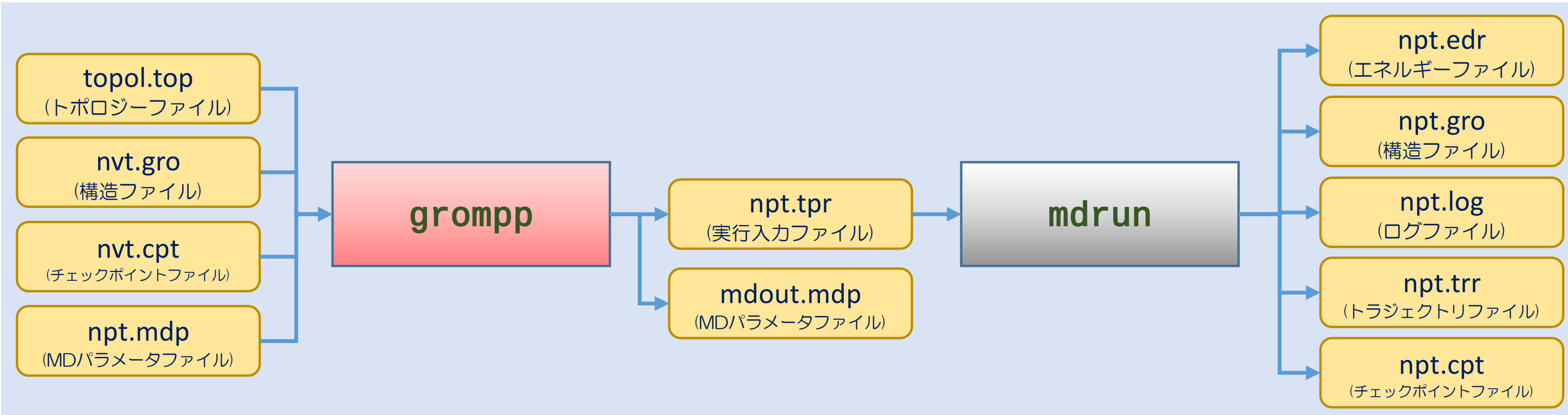

```
[user1@scls lec_gromacs]$ pjsub --interact
[INFO] PJM 0000 pjsub Job 6282 submitted.
[INFO] PJM 0081 .connected.
[INFO] PJM 0082 pjsub Interactive job 6282 started.
[user1@a01-008 lec_gromacs]$ module load Gromacs/4.6.2
[user1@a01-008 lec_gromacs]$ g_energy_mpi_d –f nvt.edr –o temperature.xvg
 :-)  G  R  O  M  A  C  S  (-:

-------------------------------------------------------------------
  1  Angle             2  Proper-Dih.       3  Ryckaert-Bell.    4  LJ-14
  5  Coulomb-14        6  LJ-(SR)           7  Disper.-corr.     8  Coulomb-(SR)
  9  Coul.-recip.     10  Position-Rest.   11  Potential        12  Kinetic-En.
 13  Total-Energy     14  Conserved-En.    15  Temperature      16  Pres.-DC
 17  Pressure         18  Constr.-rmsd     19  Vir-XX           20  Vir-XY

          ⋮

41  Lamb-non-Protein
15 0

[user1@a01-008 lec_gromacs]$ exit
```

15番のTemperatureを選択、0は入力終了



グラフ2. `temperature.xvg`をgnuplotで表示

平衡化 - NPTアンサンブル [grompp]



MDパラメータファイル

```
[user1@scls lec_gromacs]$ less mdp/npt.mdp
```

`tcoupl = V-rescale`	速度スケーリング法
`pcoupl = Parrinello-Rahman`	パリネロ・ラーマン法
`continuation = yes`	NVT平衡化から継続
`gen_vel = no`	速度は生成しない
`nsteps = 3000`	実行時間は6ps (2fs x 3000) ※実習のために短く設定しています

平衡化 - NPTアンサンブル [grompp]

ジョブスクリプト (逐次ジョブ)

```
[user1@scls lec_gromacs]$ cat script/grompp_npt.sh
#!/bin/sh
#------ pjsub options ------#
#PJM -L "rscgrp=small"
#PJM -L "node=1"
#PJM -L "elapse=10:00"
#PJM -j
#------ Program Execution ------#
module load Gromacs/4.6.2

export FLIB_FASTOMP=FALSE

grompp_mpi_d -f mdp/npt.mdp -c full/nvt.gro -t full/nvt.cpt -p topol.top -o npt.tpr
```

リソースグループ「small」
使用ノード数「1」
最大経過時間「10分」
標準エラー出力を標準出力に向ける

GROMACS環境変数の設定

gromppの実行
※ nvt.gro, nvt.cptは、事前に用意したNVTの結果を使用します。

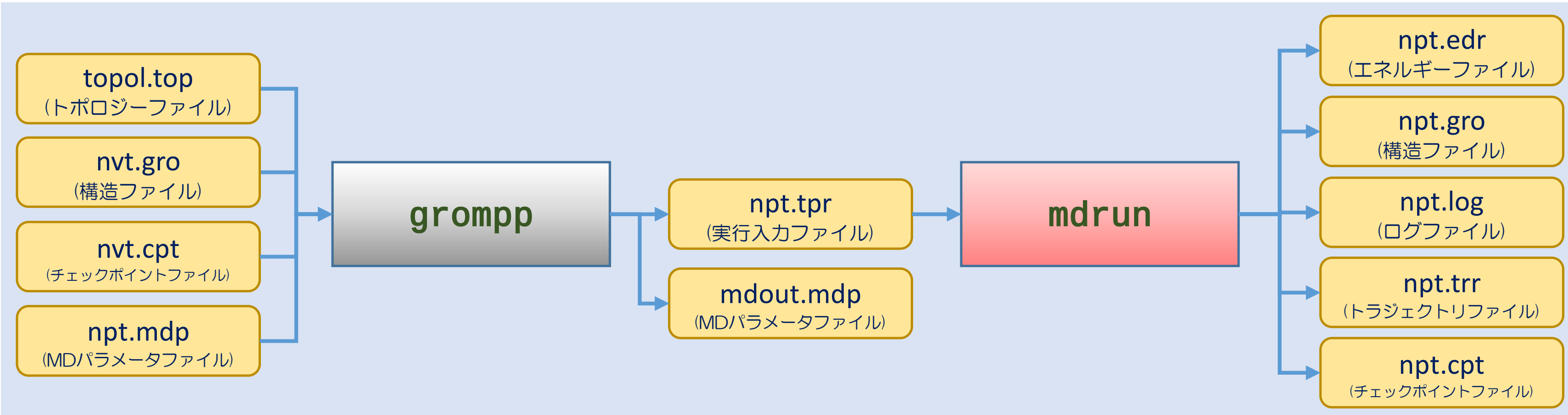
ジョブ投入

```
[user1@scls lec_gromacs]$ pjsub script/grompp_npt.sh
[INFO] PJM 0000 pjsub Job 6293 submitted.
[user1@scls lec_gromacs]$ pjstat
```

ジョブ実行結果

```
[user1@scls lec_gromacs]$ less grompp_npt.sh.oXXXX
```

平衡化 – NPTアンサンブル [mdrun]



ジョブスクリプト (MPI並列ジョブ)

```
[user1@scls lec_gromacs]$ cat script/mdrun_npt.sh
#!/bin/sh
#------ pjsub options ------#
#PJM -L "rscgrp=small"
#PJM -L "node=2"
#PJM --mpi "proc=32"
#PJM --mpi "rank-map-bychip"
#PJM -L "elapse=10:00"
#PJM -j
#------ Program Execution ------#
module load Gromacs/4.6.2

export FLIB_FASTOMP=FALSE

mpiexec mdrun_mpi_d -v -deffnm npt
```

リソースグループ「small」
使用ノード数「2」
プロセス数「32」
各ノードへのプロセスの割り振りの指定
最大経過時間「10分」
標準エラー出力を標準出力に向ける

GROMACS環境変数の設定

mdrunの実行

平衡化 – NPTアンサンブル [mdrun]

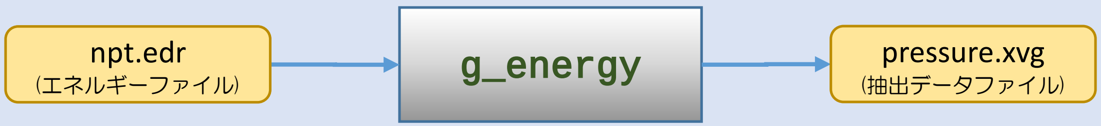
ジョブの投入と状態確認

```
[user1@scls lec_gromacs]$ pjsub script/mdrun_npt.sh
[INFO] PJM 0000 pjsub Job 6293 submitted.
[user1@scls lec_gromacs]$ 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
6293       mdrun_npt. NM RUN user1    07/26 17:37:18  0000:10:00 2
```

ジョブ実行結果

```
[user1@scls lec_gromacs]$ less npt.log

         ⋮

               Core t (s)   Wall t (s)        (%)
       Time:    32316.400      224.516    14393.8
                 (ns/day)    (hour/ns)
Performance:       38.484        0.624
Finished mdrun on node 0 Fri Jul 26 18:35:19 2013
```

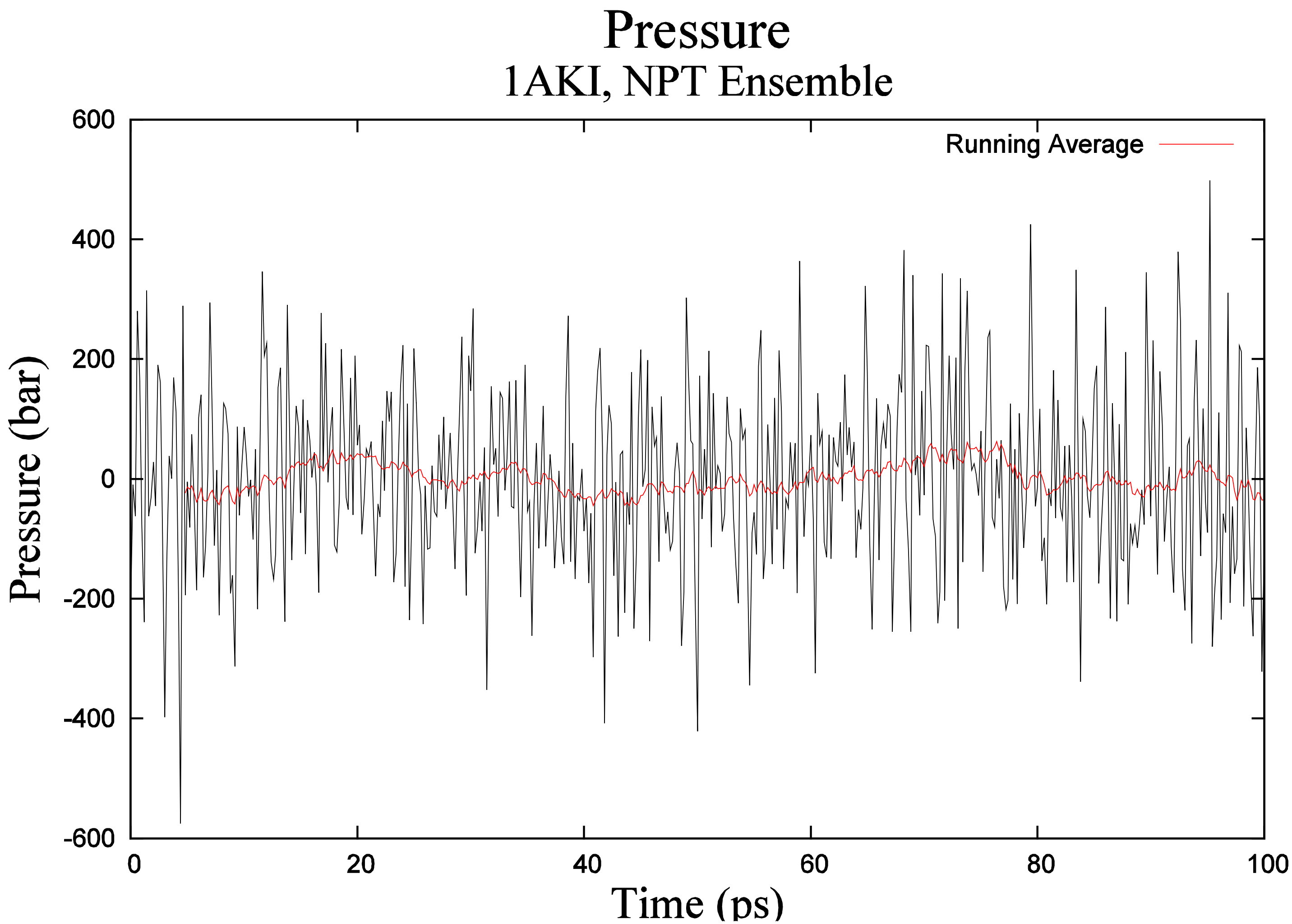
平衡化 – NPTアンサンブル [実行結果の解析]




```
[user1@scls grolec]$ pjsub --interact
[INFO] PJM 0000 pjsub Job 6282 submitted.
[INFO] PJM 0081 .connected.
[INFO] PJM 0082 pjsub Interactive job 6282 started.
[user1@a01-008 grorec]$ module load Gromacs/4.6.2
[user1@a01-008 grorec]$ g_energy_mpi_d –f npt.edr –o pressure.xvg
 :-)  G  R  O  M  A  C  S  (-:


-------------------------------------------------------------------
  1  Angle             2  Proper-Dih.       3  Ryckaert-Bell.    4  LJ-14
  5  Coulomb-14        6  LJ-(SR)           7  Disper.-corr.     8  Coulomb-(SR)
  9  Coul.-recip.     10  Position-Rest.   11  Potential        12  Kinetic-En.
 13  Total-Energy     14  Temperature      15  Pres.-DC         16  Pressure
 17  Constr.-rmsd     18  Box-X            19  Box-Y            20  Box-Z
         ⋮
 45  Box-Vel-YY       46  Box-Vel-ZZ       47  T-Protein        48  T-non-Protein
 49  Lamb-Protein                          50  Lamb-non-Protein
16 0

[user1@a01-008 grorec]$ exit
```

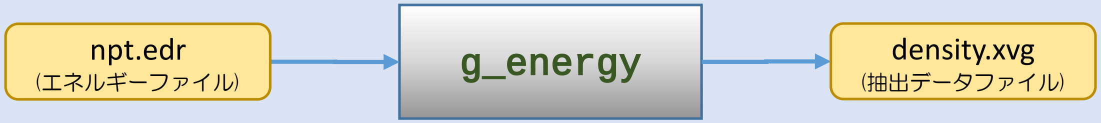
16番のPressureを選択、0は入力終了

平衡化 – NPTアンサンブル [実行結果の解析]



グラフ3. pressure.xvgをgnuplotで表示

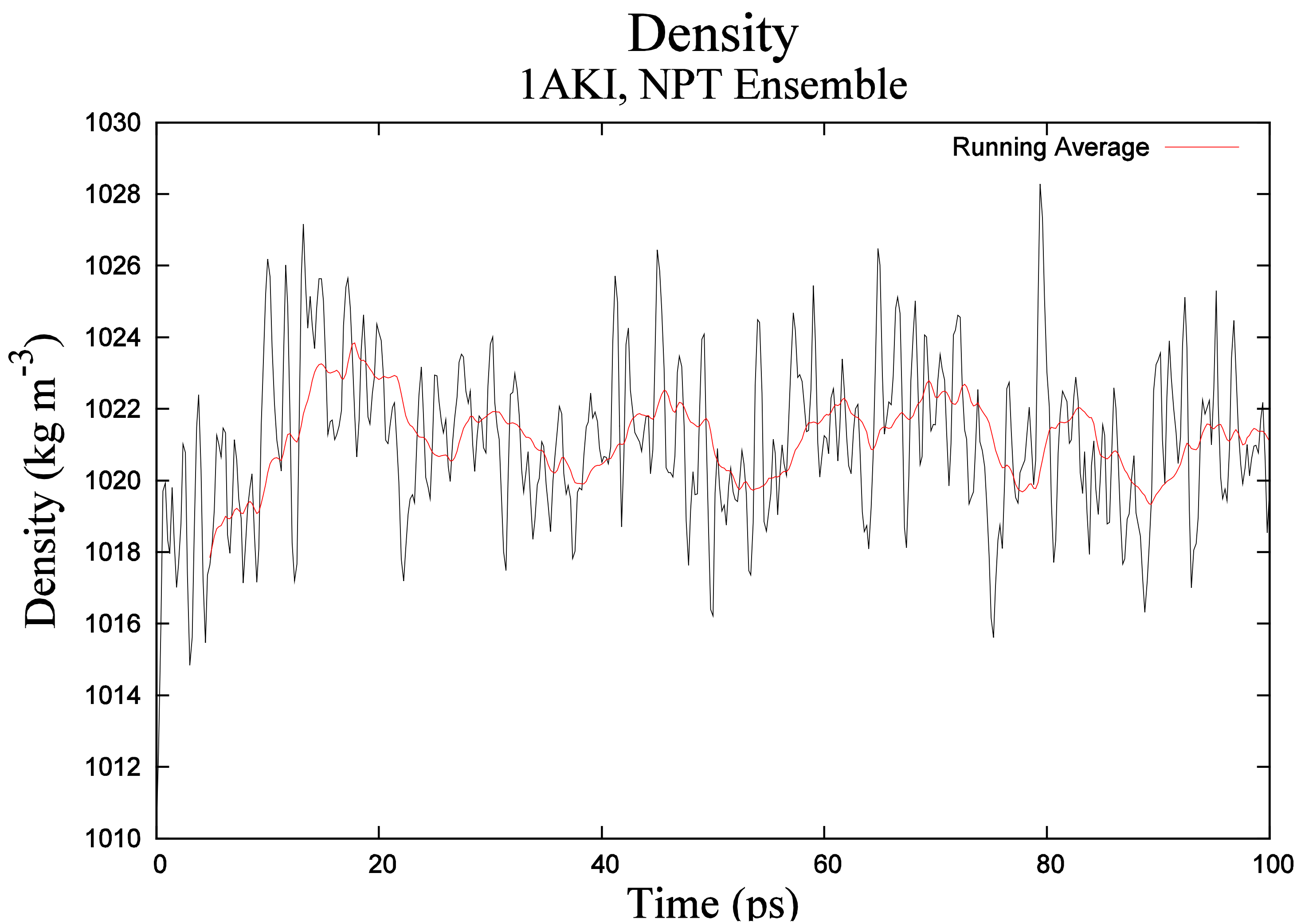
平衡化 - NPTアンサンブル [実行結果の解析]




```
[user1@scls grolec]$ pjsub --interact
[INFO] PJM 0000 pjsub Job 6282 submitted.
[INFO] PJM 0081 .connected.
[INFO] PJM 0082 pjsub Interactive job 6282 started.
[user1@a01-008 grorec]$ module load Gromacs/4.6.2
[user1@a01-008 grorec]$ g_energy_mpi_d -f npt.edr -o density.xvg
 :-)  G  R  O  M  A  C  S  (-:

------------------------------------------------------------------
  1  Angle             2  Proper-Dih.       3  Ryckaert-Bell.    4  LJ-14
  5  Coulomb-14        6  LJ-(SR)           7  Disper.-corr.     8  Coulomb-(SR)
  9  Coul.-recip.     10  Position-Rest.   11  Potential        12  Kinetic-En.
 13  Total-Energy     14  Temperature      15  Pres.-DC         16  Pressure
 17  Constr.-rmsd     18  Box-X            19  Box-Y            20  Box-Z
 21  Volume           22  Density          23  pV               24  Enthalpy
 25  Vir-XX           26  Vir-XY           27  Vir-XZ           28  Vir-YX
             :
 49  Lamb-Protein                          50  Lamb-non-Protein
22 0

[user1@a01-008 grorec]$ exit
```

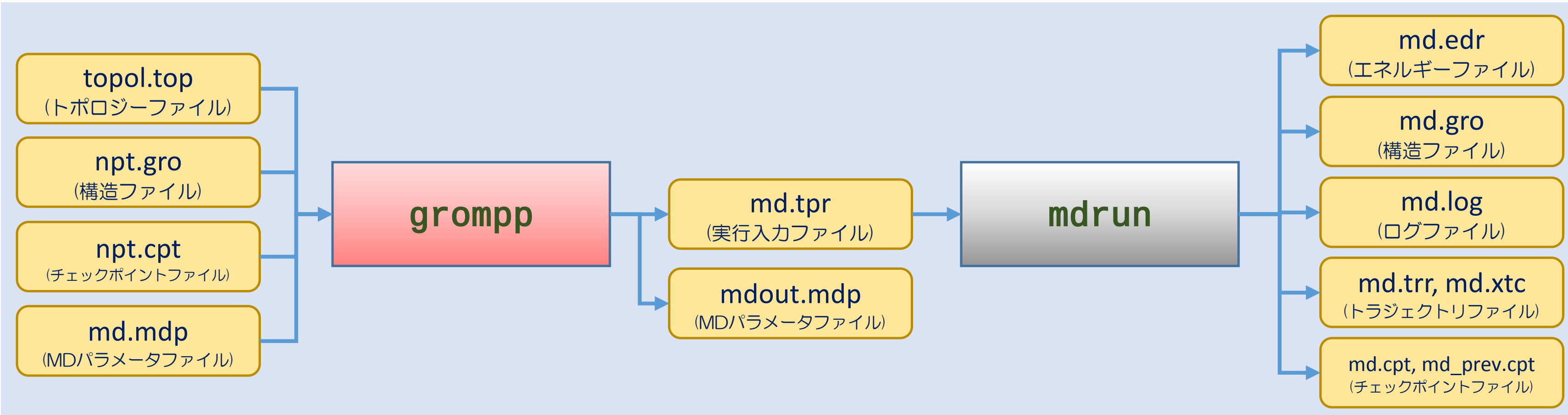
22番のDensityを選択、0は入力終了

平衡化 – NPTアンサンブル [実行結果の確認]



グラフ4. density.xvgをgnuplotで表示

Production MD [grompp]



MDパラメータファイル

```
[user1@scls lec_gromacs]$ less mdp/md.mdp
```

`tcoupl = V-rescale`	速度スケーリング法
`pcoupl = Parrinello-Rahman`	パリネロ・ラーマン法
`continuation = yes`	NPT平衡化から継続
`gen_vel = no`	速度は生成しない
`nsteps = 5000`	実行時間は10ps（2fs x 5000） ※実習のために短く設定しています

Production MD [grompp]

ジョブスクリプト（逐次ジョブ）

```
[user1@scls lec_gromacs]$ cat script/grompp_md.sh
#!/bin/sh
#------ pjsub options ------#
#PJM -L "rscgrp=small"
#PJM -L "node=1"
#PJM -L "elapse=10:00"
#PJM -j
#------ Program Execution ------#
module load Gromacs/4.6.2

export FLIB_FASTOMP=FALSE

grompp_mpi_d -f mdp/md.mdp -c full/npt.gro -t full/npt.cpt -p topol.top -o md.tpr
```

リソースグループ「small」
使用ノード数「1」
最大経過時間「10分」
標準エラー出力を標準出力に向ける

GROMACS環境変数の設定

gromppの実行
※ npt.gro, npt.cptは、事前に用意したNPTの結果を使用します。

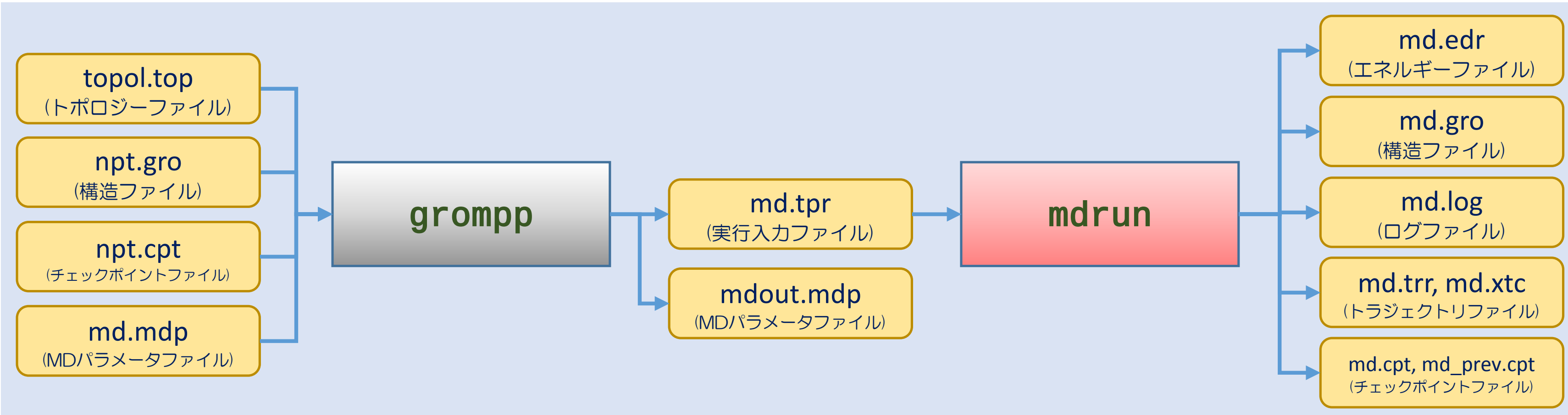
ジョブ投入

```
[user1@scls lec_gromacs]$ pjsub script/grompp_md.sh
[INFO] PJM 0000 pjsub Job 6293 submitted.
[user1@scls lec_gromacs]$ pjstat
```

ジョブ実行結果

```
[user1@scls lec_gromacs]$ less grompp_md.sh.oXXXX
```

Production MD [mdrun]



ジョブスクリプト（MPI並列ジョブ）

```
[user1@scls lec_gromacs]$ cat script/mdrun_md.sh
#!/bin/sh
#------ pjsub options ------#
#PJM -L "rscgrp=small"
#PJM -L "node=2"
#PJM --mpi "proc=32"
#PJM --mpi "rank-map-bychip"
#PJM -L "elapse=10:00"
#PJM -j
#------ Program Execution ------#
module load Gromacs/4.6.2

export FLIB_FASTOMP=FALSE

mpiexec mdrun_mpi_d -v -deffnm md
```

リソースグループ「small」
使用ノード数「2」
プロセス数「32」
各ノードへのプロセスの割り振りの指定
最大経過時間「10分」
標準エラー出力を標準出力に向ける

GROMACS環境変数の設定

mdrunの実行

Production MD [mdrun]

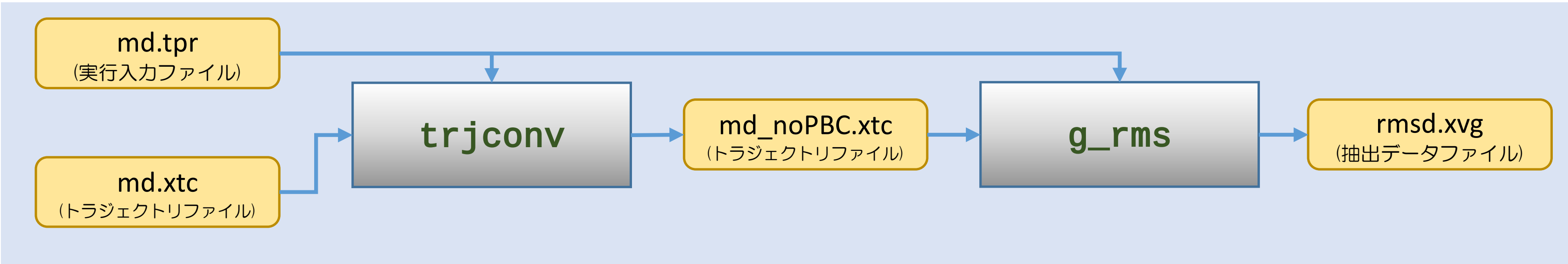
ジョブの投入と状態確認

```
[user1@scls lec_gromacs]$ pjsub script/mdrun_md.sh
[INFO] PJM 0000 pjsub Job 6293 submitted.
[user1@scls lec_gromacs]$ 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
6293       mdrun_md.s NM RUN user1    07/26 17:37:18  0000:10:00 2
```

ジョブ実行結果

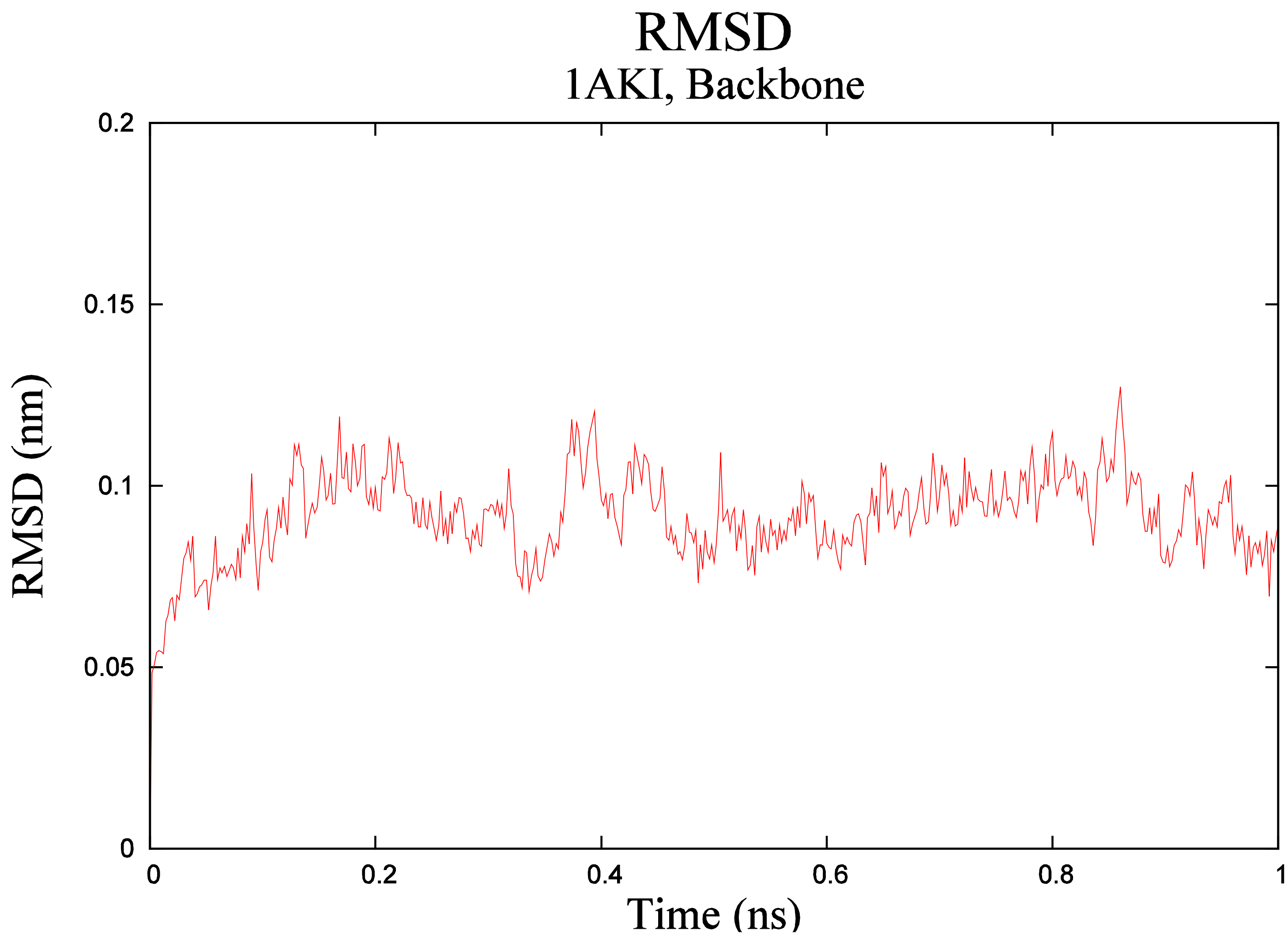
```
[user1@scls lec_gromacs]$ less md.log

         ⋮

               Core t (s)   Wall t (s)        (%)
       Time:   258551.600     1795.520    14399.8
                 (ns/day)    (hour/ns)
Performance:       48.120        0.499
Finished mdrun on node 0 Fri Jul 26 19:20:06 2013
```

Production MD [実行結果の解析]



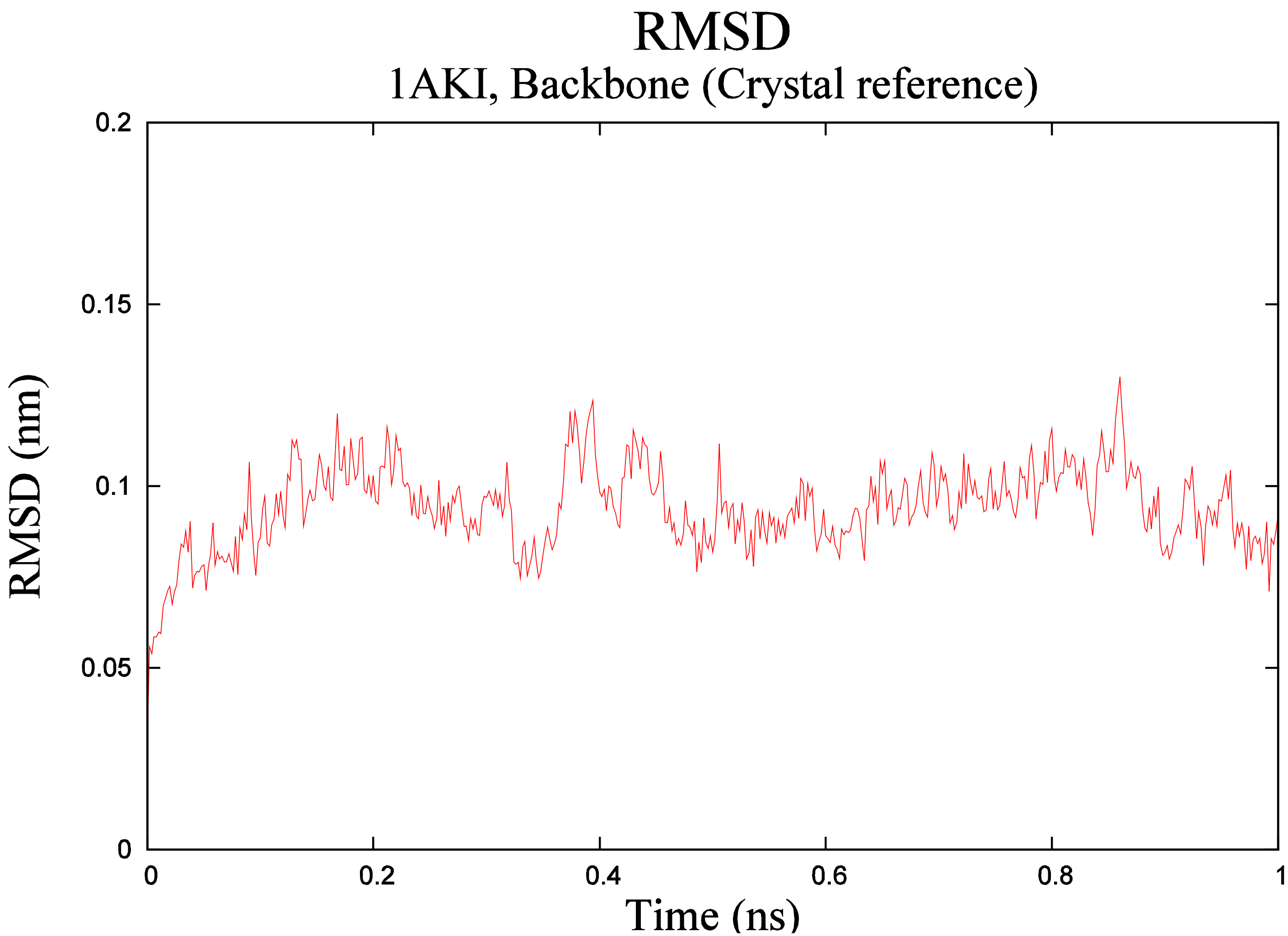

```
[user1@scls grolec]$ pjsub --interact
[INFO] PJM 0000 pjsub Job 6282 submitted.
[INFO] PJM 0081 .connected.
[INFO] PJM 0082 pjsub Interactive job 6282 started.
[user1@a01-008 grorec]$ module load Gromacs/4.6.2
[user1@a01-008 grorec]$ trjconv_mpi_d -s md.tpr -f md.xtc -o md_noPBC.xtc -pbc mol -ur compact
 :-)  G  R  O  M  A  C  S  (-:

Select group for output
Group     0 (         System) has 34440 elements
Select a group: 0

[user1@a01-008 grorec]$ g_rms_mpi_d -s md.tpr -f md_noPBC.xtc -o rmsd.xvg -tu ns
 :-)  G  R  O  M  A  C  S  (-:

Select group for least squares fit
Group     4 (       Backbone) has   387 elements
Select a group: 4

Select group for RMSD calculation
Group     4 (       Backbone) has   387 elements
Select a group: 4

[user1@scls grolec]$ exit
```

Production MD [実行結果の解析]



グラフ5. rmsd.xvgをgnuplotで表示

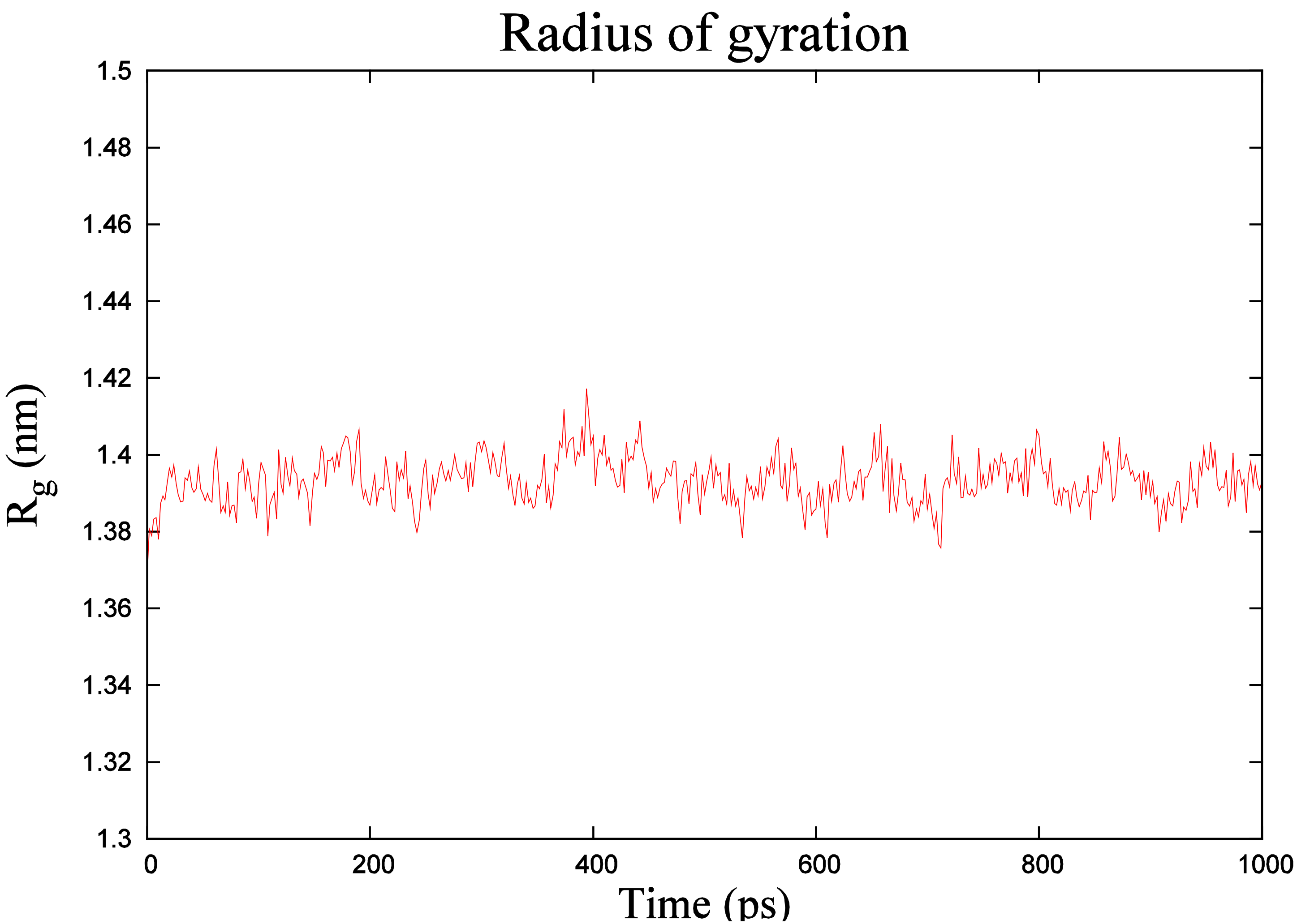
Production MD [実行結果の確認]



グラフ6. rmsd_xtal.xvgをgnuplotで表示

Production MD [実行結果の解析]


```
[user1@scls grolec]$ pjsub --interact
[INFO] PJM 0000 pjsub Job 6282 submitted.
[INFO] PJM 0081 .connected.
[INFO] PJM 0082 pjsub Interactive job 6282 started.
[user1@a01-008 grorec]$ module load Gromacs/4.6.2
[user1@a01-008 grorec]$ g_gyrate_mpi_d -s md.tpr -f md_noPBC.xtc -o gyrate.xvg
 :-)  G  R  O  M  A  C  S  (-:

         ⋮

Group     4 (       Backbone) has   387 elements
Select a group: 4

         ⋮

[user1@scls grolec]$ exit
```



グラフ7. gyrate.xvgをgnuplotで表示

GROMACSについて

GROMACSに関する情報、マニュアル、ソースコード入手など

http://www.gromacs.org/

SCLS
SUPERCOMPUTATIONAL LIFE SCIENCE

2013年8月6日（第2版）

独立行政法人理化学研究所
HPCI計算生命科学推進プログラム