



GENESIS

Generalized-ensemble simulation system

# GENESISでのMD計算ハンズオン

岩橋—小林 千草

理化学研究所 計算科学研究センター

2021/01/29 GENESIS講習会 -Cygnusを用いたハンズオン-



GENESIS

Generalized-ensemble simulation system

## 内容

- MD計算を高速に動かすために
- REMD計算のデモ

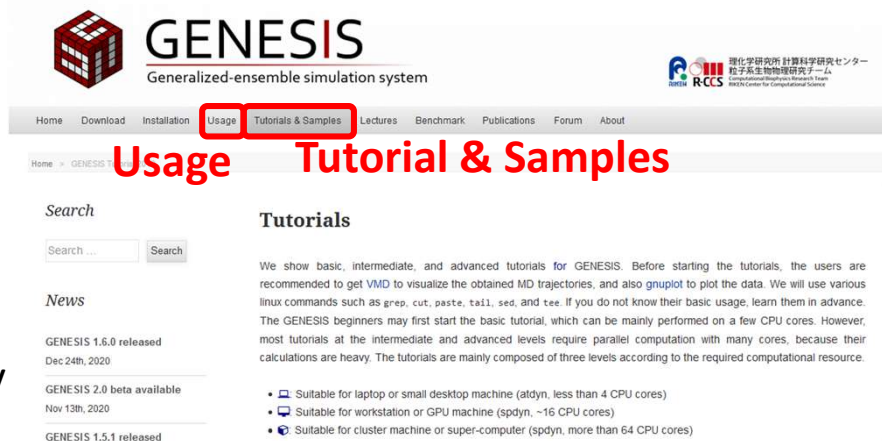


GENESIS

Generalized-ensemble simulation system

## 本ハンズオンの目的

- 今回のハンズオンはCygnusでのGENESISによるMD計算を高速に動かすためのやり方を学びます (MDの本計算を想定)
- MD/REMD計算の入力ファイルの作成方法や基本的なMD計算の手順は対象外です
- ハンズオンで取り扱わなかった件に関しては、GENESISのウェブページをご覧くださいか、ご質問をフリーディスカッションで受け付けます



<https://www.r-ccs.riken.jp/labs/cbrt/>



## ハンズオン資料の取得方法

1. ディレクトリからtutorial\_GENESIS\_2021.tar.bz2ファイルをダウンロードし、cygnusのワーク領域へ置いてください
2. cygnusにログインしていただき、ワーク領域で展開します

```
% cd /work/EDU4/<ユーザ名>
% tar xvhf tutorial_GENESIS_2021.tar.bz2
% cd tutorial_GENESIS_2021
% ls
  1_MD/ 2_REMD/
```



GENESIS

Generalized-ensemble simulation system

Hands-on 1

# MD計算を高速に動かすために



GENESIS

Generalized-ensemble simulation system

## 資料について

今回の資料はGENESIS websiteのTutorial 3.3に基づいています

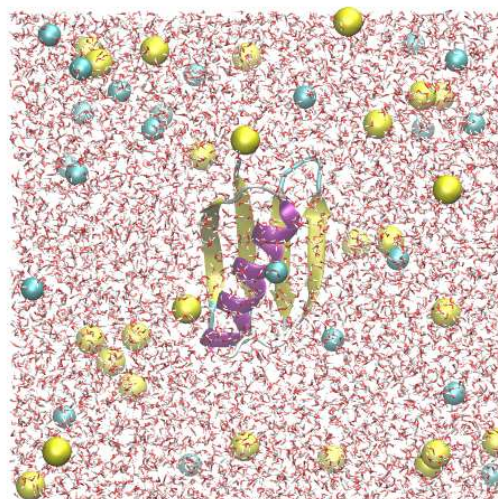
<https://www.r-ccs.riken.jp/labs/cbrt/tutorials2019/tutorial-3-3/>

MDの入力ファイルの作成方法、MD計算の準備(平衡化)などの手順は、上のウェブサイト詳しく記載されています

### 3.3 MD simulation of Protein G in NaCl solution with the CHARMM force field

#### Contents [hide]

- Preparation
- 1. Setup
- 2. Minimization
- 3. Equilibration
- 4. Production
  - Benchmark test
  - File size estimation
  - Production run
- 5. Analysis
  - 5.1 Make a trajectory file with PBC wrapping
  - 5.2 Make a trajectory file including protein only



水溶液中のProtein G

水中のタンパク質の  
等温等圧(NPT)条件下  
でのMD計算



## Step 1: 基本のMD

```
% cd
1_MD
% ls
1_VVER          3_RESPA_2.5fs          5_Benchmark    setup
2_VVER_2.5fs    4_RESPA_2.5fs_tb10    scripts        toppar
% cd 1_VVER
% ls
INP run.sh sample
```

vi等エディタでINPファイルを開いてください



**GENESIS**

Generalized-ensemble simulation system

## Step 1: GENESIS control file

GENESISのインプットは[]で囲まれたセクションに分けられており、各セクション毎に決められたパラメータが存在します

**[INPUT]** 入力ファイル

**[OUTPUT]** 出力ファイル

**[ENERGY]** 力場、エネルギー計算のパラメータ

**[MINIMIZE]** or **[DYNAMICS]** エネルギー最小法かMD計算か (排他的)

**[REMD]** or **[GAMD]** or **[RPATH]** or **[ALCHEMY]** 各計算手法でのパラメータ (排他的)

**[CONSTRAINT]** 分子内拘束

**[RESTRAINTS]** restraint計算

**[SELECTION]** restraintなどでの原子の選択

**[ENSEMBLE]** 温度、圧力の情報

**[BOUNDARY]** 周期境界条件の有無、Boxサイズなど





# Step 1: Control file

```
[INPUT]
topfile = ../toppar/top_all36_prot.rtf
parfile = ../toppar/par_all36m_prot.prm
strfile = ../toppar/toppar_water_ions.str
psffile = ../setup/ionized.psf
pdbfile = ../setup/ionized.pdb
rstfile = ../setup/eq3.rst
```

```
[ENERGY]
forcefield      = CHARMM
electrostatic   = PME
switchdist     = 10.0
cutoffdist     = 12.0
pairlistdist    = 13.5
vdw_force_switch = YES
pme_nspline    = 4
pme_max_spacing = 1.2
```

Coulomb相互作用で  
PMEを用いる

```
[DYNAMICS]
integrator      = VVER
nsteps         = 5000
timestep       = 0.002
eneout_period  = 100
nbupdate_period = 10
```

Velocity Verlet法

ステップ数  
ステップ幅(単位はps)

```
[CONSTRAINTS]
rigid_bond     = YES
```

```
[ENSEMBLE]
ensemble       = NPT
tpcontrol      = BUSSI
temperature    = 298.15
pressure       = 1.0
```

NPT計算

```
[BOUNDARY]
type           = PBC
```

周期境界条件を利用



## Step 1: 基本のMDの動かし方

vi等エディタでrun.shファイルを開いてください

```
#!/bin/bash -f
#
#PBS -A EDU4
#PBS -q genesis-b
#PBS -l elanstim_req=00:10:00 要求時間
#PBS -b 1 ノード数
#PBS -l cpunum_job=24
#PBS -T openmpi
#PBS -v OMP_NUM_THREADS=3 スレッド数
#PBS -v NQSV_MPI_VER=gdr/4.0.3/intel19.0.5-cuda10.2
module load openmpi/${NSQV_MPI_VER}
GENESIS_BIN=/work/EDU4/share/genesis-1.5.1/bin/cuda_single
cd ${PBS_O_WORKDIR}
mpirun ${NQSV_MPI_OPTS} -np 8 -npernode 8 --map-by ppr:8:node -bind-to socket
${GENESIS_BIN}/spdyn INP > LOG プロセス数
```

run.shファイルを実行します(20秒程度)

```
% qsub run.sh
```



## Step 1: 出力ファイルの見方

問題なく実行された場合はLOGという出力ファイルが出力されます

GENESISの出力ファイルは7段階([STEP 0]-[STEP 6])に分かれています

[STEP 0]: 計算環境の確認

[STEP 1]: 入力パラメータの確認

[STEP 2]: 並列数(プロセス数、スレッド数)の確認

```
[STEP2] Setup MPI
Setup.Mpi.Md> Summary of Setup MPI
number of MPI processes = 8
number of OpenMP threads = 3
total number of CPU cores = 24
```

[STEP 3]: 分子・エネルギー関数情報の確認

[STEP 4]: 初期座標のエネルギー計算結果

[STEP 5]: シミュレーション計算結果

```
[STEP5] Perform Molecular Dynamics Simulation
```

INFO:	STEP	TIME	TOTAL_ENE	POTENTIAL_ENE	KINETIC_ENE	RMSG	BOND	ANG
INFO:	100	0.2000	-68054.9248	-82706.9139	14651.9891	13.6665	155.6753	405.96
INFO:	200	0.4000	-68019.2779	-82723.5341	14704.2562	13.6456	163.1162	403.08

[STEP 6]: 終端処理と経過時間

```
Output_Time> Averaged timer profile (Min, Max)
total time = 23.005
setup = 1.164
dynamics = 21.841
energy = 13.200
integrator = 7.475
pairlist = 1.191 ( 1.118, 1.260)
```



# Step 1: 計算時間の確認の仕方

[STEP 6] : 経過時間を確認ください

```
Output_Time> Averaged timer profile (Min, Max)
total time      = 23.005
setup           = 1.164
dynamics       = 21.841
energy          = 13.200
integrator      = 7.475
pairlist        = 1.191 ( 1.118, 1.260)
energy
bond            = 0.023 ( 0.017, 0.030)
angle           = 0.084 ( 0.049, 0.126)
dihedral        = 0.228 ( 0.119, 0.349)
nonbond         = 11.725 ( 11.193, 12.128)
pme real        = 11.724 ( 11.192, 12.127)
pme recip       = 8.354 ( 8.341, 8.370)
solvation       = 0.000 ( 0.000, 0.000)
polar           = 0.000 ( 0.000, 0.000)
non-polar       = 0.000 ( 0.000, 0.000)
restraint       = 0.000 ( 0.000, 0.000)
qmmm            = 0.000 ( 0.000, 0.000)
integrator
constraint      = 1.331 ( 1.298, 1.397)
update          = 0.594 ( 0.585, 0.609)
comm_coord     = 0.796 ( 0.725, 0.903)
(skip)
```

単位=秒

MDのメインループの経過時間

‘total time’はsetupも含めた全体の経過時間です

計算時間の見積もりを行うためには“dynamics”で示された時間を利用してください

経過時間は基本的に最大値が使われます。

Performance (ns/day)=

各プロセスでの  
最大値

$24 \times 3600 \times 0.01 / \{\text{dynamics}\}$

最大値と最小値の差はプロセス間のインバランスとなるので、時間が遅いときは確認が必要です



GENESIS

Generalized-ensemble simulation system

## Step 2: 時間幅の変更

```
% cd ../2_VVER_2.5fs
% ls
run.sh sample
% cp ../1_VVER/INP .
```

GENESISのIntegratorは、2.5fsでの計算が可能です

[Jung et al. J. Chem. Phys. \*\*148\*\*, 164109 \(2018\)](#)

[Jung et al. J. Chem. Theory Comput. \*\*15\*\*, 84-94 \(2018\)](#)

vi等エディタでINPファイルを開いて、編集してください

```
[DYNAMICS]
integrator      =      VVER
nsteps          =      4000
timestep        =      0.0025
eneout_period   =      80
```

ステップ数 (5000 → 4000)  
ステップ幅 (0.002 → 0.0025)  
エネルギー表示回数 (100 → 80)

```
% qsub run.sh
```



## Step 3: RESPAの利用(1)

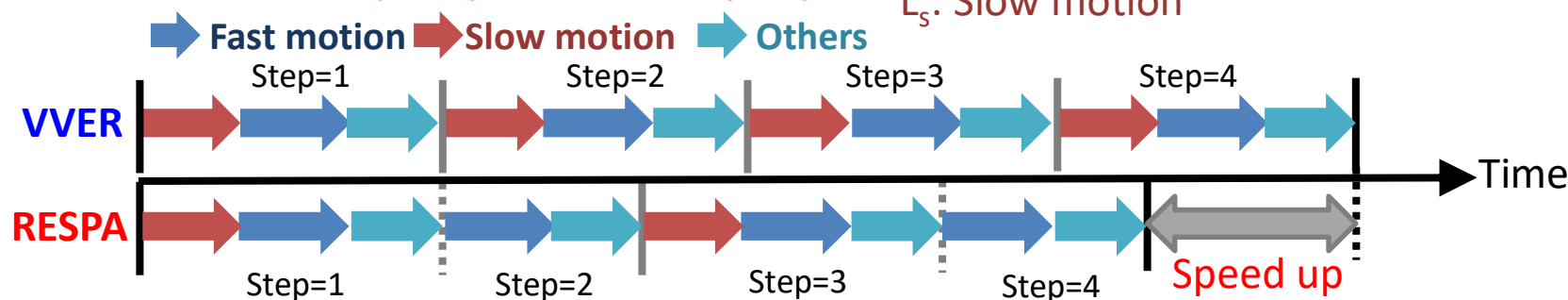
```
% cd ../3_RESPA_2.5fs
% ls
run.sh sample
% cp ../2_VVER_2.5fs/INP .
```

GENESISのIntegratorは、RESPA(Tuckerman *et al.*, J. Chem. Phys. **97**, 1990 (1992))の計算が可能です

GENESISでのRESPAについてJung *et al.* J. Chem. Phys. **148**, 164109 (2018)

$$e^{iL\Delta t} = e^{iL_s\Delta t/2} \left( e^{iL_f\delta t} \right)^n e^{iL_s\Delta t/2} + O(\Delta t^3),$$

$L_f$ : Fast motion  
 $L_s$ : Slow motion



**Fast motion** : 結合性相互作用、非結合性相互作用(短距離)

**Slow motion** : 非結合性相互作用(長距離)、温度・圧力計算



GENESIS

Generalized-ensemble simulation system

## Step 3: RESPAの利用(2)

vi等エディタでINPファイルを開いて、編集してください

```
[DYNAMICS]
```

```
integrator      =      VRES
```

RESPAの利用 (VVER → VRES)

```
nsteps         =      4000
```

```
timestep       =      0.0025
```

```
eneout_period  =      80
```

```
nbupdate_period =      10
```

```
elec_long_period =      2
```

長距離非結合性相互作用計算の頻度(2 steps, 5fs毎)

```
thermostat_period =      2
```

温度計算の頻度(2 steps, 5fs毎)

```
barostat_period =      2
```

圧力計算の頻度(2 steps, 5fs毎)

```
% qsub run.sh
```



## Step 4: 温度・圧力計算の頻度を落とすRESPA

```
% cd ../4_RESPA_2.5fs_tb10
% ls
run.sh sample
% cp ../3_RESPA_2.5fs/INP .
```

GENESISの温度・圧力計算の頻度は10(25fs)へ落とすことが可能です

```
[DYNAMICS]
ntegrator          =      VRES
(skip)
elec_long_period   =      2
thermostat_period =      10
barostat_period   =      10
```

温度計算の頻度(2 → 10)

圧力計算の頻度(2 → 10)

```
% qsub run.sh
```





## 計算時間の比較

```
% cd ../  
% grep " dynamics " */LOG | sort -g -r -k 4 > timer.log
```

timer.logから、インプットを変更することで計算時間が短縮していることをご確認ください (sortコマンドによって計算時間が短いものが最後に来ます)

1_VVER/LOG:	dynamics	=	21.841
2_VVER_2.5fs/LOG:	dynamics	=	17.445
3_RESPA_2.5fs/LOG:	dynamics	=	16.335
4_RESPA_2.5fs_tb10/LOG:	dynamics	=	15.404

Tutorial 3.3で使われている入力パラメータ(Integrator, dt,...)は4\_RESPA\_2.5fs\_tb10と同じものです



## Step 5: ベンチマーク計算

効果的に研究を行うためには、ベンチマークを計測し効率的なコア数・並列パラメータを決定することが大事ですベンチマークの計算によって必要とされる資源量(時間、ノード時間)を知ることができます

Cygnusは1ノード24コア持つため、プロセスとスレッドの組み合わせは

(processes, threads)=(1,24), (2,12), (3, 8), (4, 6), (6, 4), (8, 3), (12, 2), (24, 1)

が理論上考えられます

数字を変えて試してみましょう (異なる名前のスクリプトを作成ください)

```
#!/bin/bash -f
#
(skip)
#PBS -b 1 ノード数(固定)
#PBS -T openmpi
#PBS -v OMP_NUM_THREADS=3 スレッド数
(skip)
mpirun ${NQSII_MPIOPTS} -np 8 -npernode 8 --map-by ppr:8:node -bind-to socket spdyn INP >
p8t3.log
```

全プロセス(MPI)数

ノード当たりの  
プロセス数



## Step 5: 計算が失敗する理由 (1)

いくつかの組み合わせでは、ログが以下の部分で止まります

```
Define_Molecule> Summary of molecules
  num_atoms      =      24552  num_bonds      =      16628
  num_angles     =      9438  num_dihedrals   =      2269
  num_impropers  =      141   num_cmap_terms  =      54
  num_residues   =      7986  num_molecules =      7931
  num_segments  =          3  num_deg_freedom =     73656
  total_charge   =     -0.000

Setup_Restart_Pre> Coordinates and velocities were replaced
```

その場合は(script名).eXXXXX (XはJOBID)をご覧ください



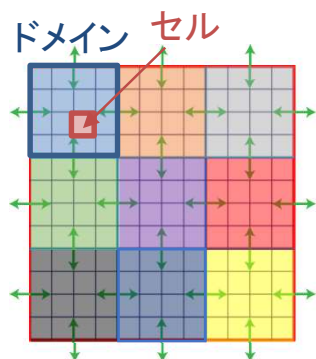
GENESIS

Generalized-ensemble simulation system

## Step 5: 計算が失敗する理由 (2)

Case 1:

```
Setup_Processor_Number> MPI Process number can not be defined, please set them manually  
rank_no = 2
```



GENESISのドメイン分割は、全体のMPIプロセスを各方向(X, Y, Z)に振り分けますが、自動で振り分けることができない場合にこのエラーが出ます (特に少ないプロセス数の時によく出ます)

```
[BOUNDARY]  
type = PBC  
domain_x = 2  
domain_y = 2  
domain_z = 3
```

各方向でのMPIプロセスの数  
 $\text{domain\_x} * \text{domain\_y} * \text{domain\_z} = \# \text{ of MPIs}$

ただし、 $\text{domain\_}[x,y,z] = 1$ の場合は

$(\text{domain\_x}, \text{domain\_y}, \text{domain\_z}) = (2, 1, 1)$  or  $(2, 2, 1)$  or  $(1, 1, 1)$ 以外は使えません  
つまり、3や6のMPIプロセスは計算できません



## Step 5: 計算が失敗する理由 (3)

### Case 2:

```
Setup_Processor_Number> Cannot define domains and cells. Smaller or adjusted MPI  
processors, or shorter pairlistdist, or larger boxsize should be used (see "Chapter:  
Trouble shooting" in the user manual). rank_no = 13
```

セルのサイズはドメイン数とペアリストの長さで決まります。適切なドメイン分割ができない場合にはこのようなエラーが出ます。MPIの数を減らすか、セットアップを見直してください。

ペアリスト長=13.5 Å の場合、最短のセル長( $R_{cell, min}$ )、実際のセル長( $R_{cell, i}$ )、セル数( $N_{cell, i}$ )は

$$R_{cell, min} = \frac{(R_{pairlist} + R_{buff})}{2} = \frac{(13.5 + 2.6)}{2} = 8.05$$

$$R_{buff} = 2.6 \text{ (NPT \& (const || water\_model))}$$

$$N_{cell, i} = \text{int}(B_i / R_{cell, min}) - \text{mod}(\text{int}(B_i / R_{cell, min}), \text{domain}_i) \quad i = x, y, z$$

$$R_{buff} = 2.0 \text{ ((const || water\_model))}$$

$$R_{cell, i} = B_i / N_{cell, i}$$

$N_{cell, i} \geq 5$ 、かつ、 $N_{cell, i} > 2 * \text{domain}_i$  である必要があります

$R_{cell, i} \gg R_{cell, min}$  の場合にはメモリ使用量が大きくなりすぎたり、シミュレーションが遅くなります

[STEP3]のSetup\_Boundary\_Cell>にこのシミュレーションでのドメインとセル数が出力されます

```
Setup_Boundary_Cell> Set Variables for Boundary Condition  
domains (x, y, z) =           2           2           1  
ncells (x, y, z) =           6           6           7
```



## Step 5: ベンチマーク計算

計算ができたものは以下の5セットになるはずです

(processes, threads)=(1,24), (2,12), (4, 6), (8, 3), (12, 2)

```
grep " dynamics " p*log | sort -g -r -k 4 > timer.log
```

計算時間がプロセス、スレッド数の組み合わせによって大きく変わっていることをご確認ください

p1t24.log:	dynamics	=	39.609
p2t12.log:	dynamics	=	22.062
p12t2.log:	dynamics	=	17.313
p4t6.log:	dynamics	=	15.892
<b>p8t3.log:</b>	<b>dynamics</b>	<b>=</b>	<b>15.404</b>

↑  
同じコア数でも2倍  
以上の差  
↓

プロセス、スレッド数の組み合わせは、システムのサイズ、計算条件で大きく変化します

まずは5000-10000ステップで最適な組み合わせを探してください



GENESIS

Generalized-ensemble simulation system

Hands-on 2

# REMD計算のデモンストレーション



GENESIS

Generalized-ensemble simulation system

## 資料について

今回の資料はGENESIS websiteのTutorial 10.1に基づいています

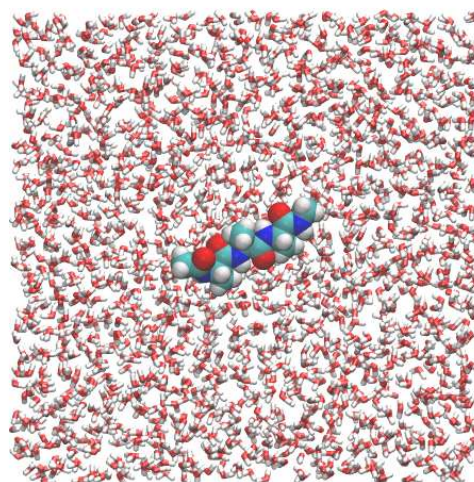
<https://www.r-ccs.riken.jp/labs/cbrt/tutorials2019/tutorial-10-1/>

初期構造作成などは、上のウェブサイトに詳しく記載されています

### 10.1 REMD simulation of alanine-tripeptide in water

#### Contents [hide]

- Preparation
- 1. Setup
- 2. Minimization
- 3. Heating
- 4. Equilibration
- 5. REMD Equilibration
- 6. Production run
- 7. Analysis
  - 7.1. Calculate the acceptance ratio of each replica



水中のペプチドの  
等温(NVT)条件下での  
REMD計算

水溶液中のペプチド





## ファイル

```
% cd ../../2_REMD
% ls
1_REMD_DEMO toppar setup
% cd 1_REMD_DEMO
% ls
INP run.sh sample
```

vi等エディタでINPファイルを開いてください



GENESIS

Generalized-ensemble simulation system

## コントロールファイル(抜粋)

```
[OUTPUT]
dcdfile  = step6_rep{}.dcd
rstfile  = step6_rep{}.rst
remfile  = step6_rep{}.rem
logfile  = step6_rep{}.log
(skip)
```

入出力ファイルは  
header.N.sufの様に  
レプリカ番号が入る  
それらの番号部分  
は{}で示す

```
[DYNAMICS]
integrator      = VRES
nsteps          = 4000
timestep        = 0.0025
eneout_period   = 500
crdout_period   = 500
rstout_period   = 2000
stoptr_period   = 10
elec_long_period = 2
thermostat_period = 10
barostat_period = 10
```

ステップ数はレプリカの交換頻度の倍数

出力頻度はレプリカの交換頻度の約数

```
[REMD]
```

```
dimension      = 1
exchange_period = 2000

type1          = temperature
nreplica1      = 8
parameters1    = 300.00 302.53 305.09 307.65
                 310.24 ¥
                 312.85 315.47 318.12
```

REMDの変数(次元)の数

レプリカの交換頻度

レプリカの種類  
レプリカ数

各レプリカが持つ値  
継続行は"¥"でつなぐ(ただし、¥の後に空白などをいれないこと、次行の先頭は一つ以上の空白を入れること)

注意:デモ利用のため、  
10.1のパラメータ(step6)からレプリカ数など  
変更しています



## スクリプトファイル

vi等エディタでrun.shファイルを開いてください

今回のデモでは1レプリカ当たり、4 プロセス \* 3 スレッドで計算を行います

全体のプロセス数 = レプリカ数 (8レプリカ) \* 4 プロセス = 32 プロセス

32 プロセス \* 3 スレッド = 96 コア → 4ノード 利用

```
#!/bin/bash -f
#
#PBS -A EDU4
#PBS -q genesis
#PBS -l elapstim_req=00:10:00
#PBS -b 4
#PBS -T openmpi
#PBS -v OMP_NUM_THREADS=3
#PBS -v NQSV_MPI_VER=gdr/4.0.3/intel19.0.5-cuda10.2
module load openmpi/${NSQV_MPI_VER}
GENESIS_BIN=/work/EDU4/share/genesis-1.5.1/bin/cuda_single
cd ${PBS_O_WORKDIR}
mpirun ${NQSV_MPIOPTS} -np 32 -npernode 8 --map-by ppr:8:node -bind-to socket ¥
${GENESIS_BIN}/spdyn INP > LOG
```



## 出力ファイルの見方 (1)

問題なく実行された場合はLOGという出力ファイル、更に各レプリカ毎のエネルギー、トラジェクトリなどが別ファイルで出力されます

REMDの場合も出力ファイルは7段階([STEP 0]-[STEP 6])に分かれています  
並列度の情報はMDと同様に[STEP2]ですが、レプリカ毎の情報もあります

```
Setup_Mpi_Remd> Summary of Setup MPI
number of MPI processes           =          32
number of MPI processes in one replica =          4
number of OpenMP threads         =           3
total number of CPU cores        =          96
```

レプリカの情報を実際にどう使われているかは[STEP3]に表示されます

```
Setup_Remd> Replica information

ParmsetID
      1 =          1
(skip)
Parameters
Dim   1 =    300.000    302.530    305.090    307.650    310.240 ...
```



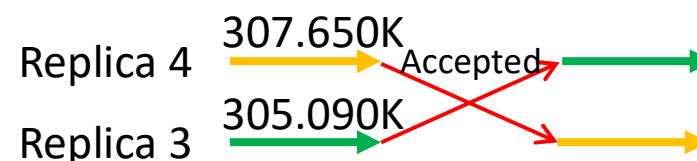
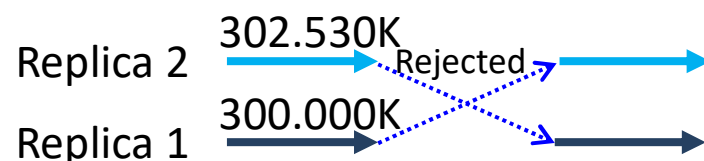
GENESIS

Generalized-ensemble simulation system

## 出力ファイルの見方 (2)

問題なく実行された場合はLOGという出力ファイル、更に各レプリカ毎のエネルギー、トラジェクトリなどが別ファイルで出力されます

REMDの場合も出力ファイルは7段階([STEP 0]-[STEP 6])に分かれています  
並列度の情報はMDと同様に[STEP2]ですが、レプリカ毎の情報もあります



REMD> Step:	4000	Dimension:	1	ExchangePattern:	1				
Replica	ExchangeTrial			AcceptanceRatio		Before	After		
1	1 > 2	R		0 / 1		300.000	300.000		
2	2 > 1	R		0 / 1		302.530	302.530		
3	3 > 4	A		1 / 1		305.090	307.650		
(skip)	3番と4番の交換		R: rejected A: accepted		各レプリカが担当する 変数(温度)の変化				
Parameter :	300.000	302.530	307.650	305.090	312.850	310.240	318.120	315.470	
RepIDtoParmID:	1	2	4	3	6	5	8	7	
ParmIDtoRepID:	1	2	4	3	6	5	8	7	



GENESIS

Generalized-ensemble simulation system

## 最後に

GENESIS開発チームへの質問、バグ報告等はGENESIS forumをお使いください

The screenshot shows the GENESIS website with the 'Forum' link highlighted in the navigation bar. A blue arrow points from the 'Forum' link to a table of forum topics.

Forum	Topics	Posts	Freshness
<a href="#">Announcement</a>	13	13	2 months ago
Announcement (read only)			
<a href="#">Bug Reports</a>	1	1	4 years ago
Forum for bug reporting			
<a href="#">Questions</a>	16	48	1 month ago
Questions about GENESIS			
<a href="#">Questions (Japanese language only)</a>	13	51	1 month ago
Questions about GENESIS			

日本語専用のフォーラムもあります

今日はありがとうございました！



## 質問への回答

当日(1/29)に回答できなかった質問に対する返答を記載します

1. FEPとgRESTを同時に組み合わせてできますか？

プログラム自体では可能となっていますが、公式にはサポートしていません

2. Implicit solventとREMDの組み合わせはできますか？

ATDYN でのみ使えます。まずは真空中での REMD 用のコントロールファイルを用意し、[ENERGY]セクションの `implicit_solvent = NONE` を GBSA に変えてください。