



GENESIS

Generalized-ensemble simulation system

GENESIS チュートリアル 1: GENESISのコンパイル、基本のMD

CBI学会2017年大会

2017/10/05

小林千草

(理化学研究所・計算科学研究機構)



GENESIS

Generalized-ensemble simulation system

14:05 – 15:30 GENESISチュートリアル1

- チュートリアルで使用するマシンの利用方法
- GENESISのコンパイル
- 基本のMD

クラウドマシンへのアクセス

この講習会ではマイクロソフトAzureを利用します

0. お持ちのマシンがインターネットにアクセスできるかをご確認ください

1. クラウドマシンへユーザアカウントを利用してログインください

```
% ssh userXX@riken.eastus.cloudapp.azure.com  
login node  
% ssh userXX@n0XX  
calculation node
```

userXX: アカウント
配布したカードからアカウント、
パスワードをご確認ください

2. ホームディレクトリに以下のファイルがあるかをご確認ください

```
% pwd  
/home2/userXX  
% ls -l  
drwxr-xr-x. 4 userXX user 46 Feb 25 09:35 Compile  
drwxr-xr-x. 2 userXX user 123 Feb 25 10:21 env  
drwxr-xr-x. 8 userXX user 121 Feb 21 19:16 Tutorial_1  
drwxr-xr-x. 3 userXX user 19 Feb 22 16:53 Tutorial_2  
...
```

2'. GENESIS siteからもチュートリアル内容物をダウンロードできます

<http://www.aics.riken.jp/labs/cbrt/news/genesis-tutorial-in-japanese-at-tokyo-oct-5-2017/>



GENESIS

Generalized-ensemble simulation system

14:05 – 15:30 GENESISチュートリアル1

➤ チュートリアルで使用するマシンの利用方法

➤ GENESISのコンパイル

➤ 基本のMD

この講習会で利用するファイル

Compile/: Compile of GENESIS

- genesis-1.1.6RC2
Recent GENESIS code
- tests-1.1.6RC2
Compile test

Tutorial_1/: Basic MD

- 1_setup
- 2_minimization
- 3_heating
- 4_equilibration
- 5_production
- 6_analysis

env/: Set of environment (消したり、変更しないでください!)

- setpath*

GENESISのコンパイル (1)

GENESISのsrcディレクトリに移動してください

```
% cd Compile/genesis-1.1.6RC2/src
```

GENESISのソースツリー

..		depcomp	
COPYING		install-sh	
README	簡単な説明	lib/	共通コード
src/	ソースコード	Missing	
bin/	バイナリ(コンパイル後)	spdyn/	spdynのコード
./src:		./src/analysis:	
GENESIS_VERSION		Makefile.am	
Makefile.am		Makefile.in	
Makefile.in		libana/	analysis共通コード
aclocal.m4		pcaana/	PCA解析
analysis/	解析ツールのコード	rpath_generator/	rpath生成コード
atdyn/	atdynのコード	rstcnv/	rst_convertのコード
bootstrap		trjana/	trj_analysisのコード
cleanup		trjcnv/	crd/pcrd/remd_convertの
config.h.in			コード
configure			
configure.ac			

GENESISのコンパイル (2)

推奨されるコンパイラ

Intel , Fujitsu compiler
gfortran 4.4.7より新しいもの

GENESISのコンパイルにはlapack, blasが必要

1. コンパイルはautoconfとmakeを使います

```
% ./configure  
% make  
% make install
```

2. 出来上がったバイナリはgenesis-1.1.6RC2/binに作成されます

GENESISはMD, トラジェクトリ変換プログラム、解析プログラム合わせて9つのアプリケーションを作成します

今回は、時間の都合で、すでにコンパイル済みのアプリケーションを使います

GENESISのコンパイル(3)

計算環境などによりコンパイルオプションを以下の様に設定ください

0. 可能なオプション一覧の確認

```
% ./configure --help
```

1. オプション無しでの実行はすべてのプログラムが倍精度となります

```
% ./configure  
% make install
```

2. spdynを混合精度でコンパイル(**SPDYN のみ**)

```
% ./configure --enable-single  
% make install
```

3. spdynをGPU利用し、混合精度でコンパイル (**SPDYN のみ**)

```
% ./configure --enable-gpu --enable-single --with-cuda=/usr/local/cuda-8.0  
% make install
```

4. 'debug'モードでのコンパイル

```
% ./configure --enable-debug=3  
% make install
```

コンパイルテスト(1)

コンパイルが正しく実行されたことを確認するために、開発者が提供する入出力ファイルによるテストを行うことを推奨します

'GENESIS' websiteから'Test set'としてダウンロードが可能です

```
% cd Compile/tests-1.1.6RC2/regression_test
```

GENESIS-1.1.6RC2のテストセット一覧

./regression_test:		test_remd.py	scripts for REMD
build/	Inputs for regression tests	test_rpath.py	scripts for string method
charmm.py		test_rpath_atdyn/	tests for rpath(atdyn)
genesis.py		test_rpath_spdyn/	tests for rpath(spdyn)
param/	FF parameters	test_spdyn/	tests for functions only in
test.py	regression test's script		spdyn
test_atdyn/	tests for functions only in atdyn		
test_common/	tests for common functions in		
	spdyn & atdyn		
test_nonstrict.py			
test_parallel_IO/	tests for parallel I/O		
test_remd/	tests for REMD		
test_remd.csh			

コンパイルテスト(2)

‘regression tests’の実行方法

```
% cd ~/Compile/tests-1.1.6RC2/regression_tests
% export PATH_GENESIS=/home2/data/genesis/bin.CPU.dp
% export OMP_NUM_THREADS=3
% ./test.py "mpirun -np 8 ${PATH_GENESIS}/atdyn"
% ./test.py "mpirun -np 8 ${PATH_GENESIS}/spdyn"
```

プロセス数は必ず8にしてください

速度上の観点から、スレッド数はプロセス数以下が推奨です

GPU環境でのコンパイルテストの仕方

```
% ./test.py "mpirun -np 8 ${PATH_GENESIS}/spdyn" gpu
```

REMD, R-PATHの実行テスト

```
% export PATH_GENESIS=/home2/data/genesis/bin.CPU.dp
% export OMP_NUM_THREADS=3
% ./test_remd.py "mpirun -np 8 ${PATH_GENESIS}/spdyn"
% ./test_remd.py "mpirun -np 8 ${PATH_GENESIS}/atdyn"
% ./test_rpath.py "mpirun -np 8 ${PATH_GENESIS}/spdyn"
% ./test_rpath.py "mpirun -np 8 ${PATH_GENESIS}/atdyn"
```

注意: 倍精度・混合精度はテストスクリプトが自動判定しています

GENESISの利用方法

spdyn/atdynの利用方法

```
% export PATH_GENESIS=/home2/data/genesis/bin.CPU.dp
% export OMP_NUM_THREADS=3
% mpirun -np 8 ${PATH_GENESIS}/atdyn INP
% mpirun -np 8 ${PATH_GENESIS}/spdyn INP
```



GENESIS

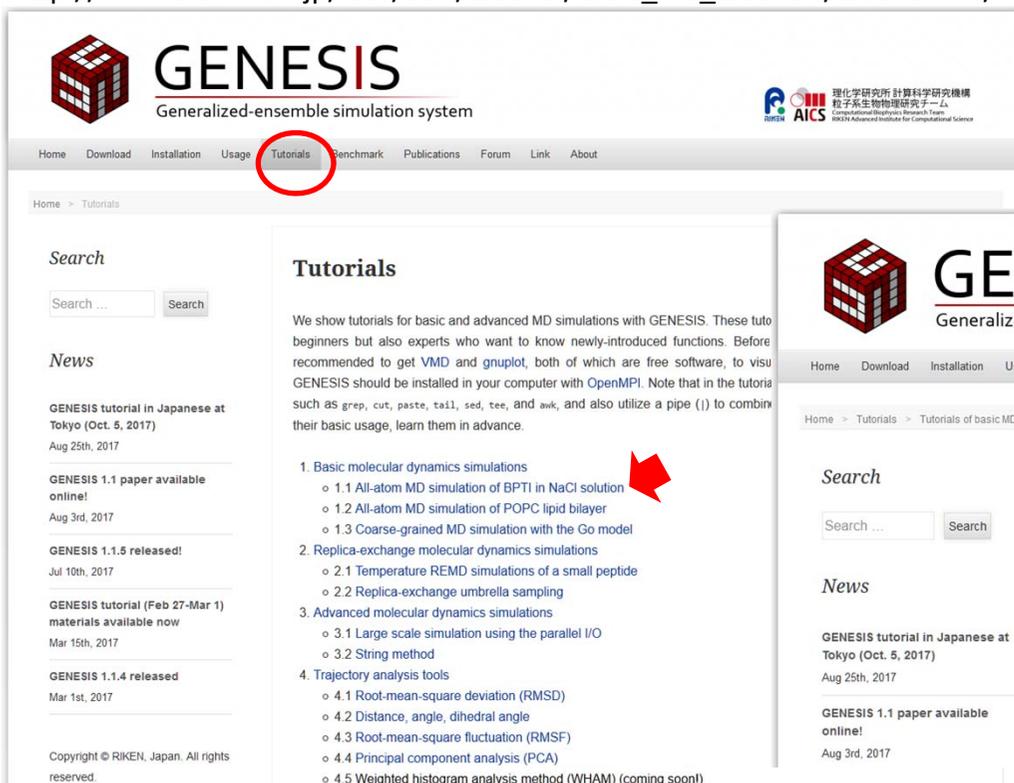
Generalized-ensemble simulation system

14:05 – 15:30 GENESISチュートリアル1

- チュートリアルで使用するマシンの利用方法
- GENESISのコンパイル
- 基本のMD

Tutorial 1: 水溶性タンパク質のMD

http://www.aics.riken.jp/labs/cbrt/tutorial/basic_md_tutorials/tutorial-1-1/



GENESIS
Generalized-ensemble simulation system

Home Download Installation Usage **Tutorials** Benchmark Publications Forum Link About

Home > Tutorials

Search

Search ... Search

News

GENESIS tutorial in Japanese at Tokyo (Oct. 5, 2017)
Aug 25th, 2017

GENESIS 1.1 paper available online!
Aug 3rd, 2017

GENESIS 1.1.5 released!
Jul 10th, 2017

GENESIS tutorial (Feb 27-Mar 1) materials available now
Mar 15th, 2017

GENESIS 1.1.4 released
Mar 1st, 2017

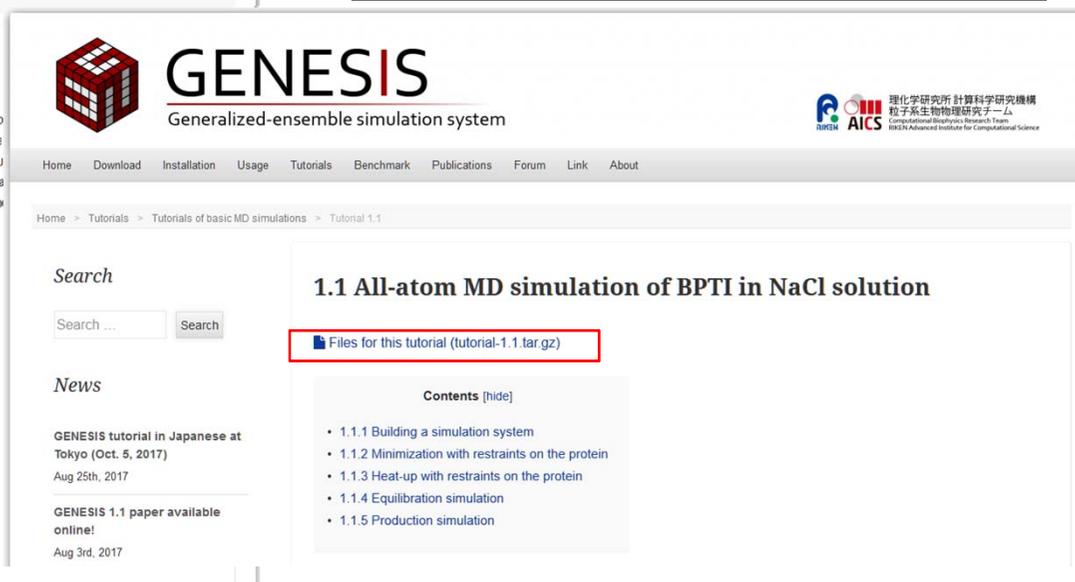
Copyright © RIKEN, Japan. All rights reserved.

Tutorials

We show tutorials for basic and advanced MD simulations with GENESIS. These tutorials are for beginners and also experts who want to know newly-introduced functions. Before recommended to get VMD and gnuplot, both of which are free software, to visualize GENESIS should be installed in your computer with OpenMPI. Note that in the tutorials such as `grep`, `cut`, `paste`, `tail`, `sed`, `tee`, and `awk`, and also utilize a pipe (`|`) to combine their basic usage, learn them in advance.

1. Basic molecular dynamics simulations
 - 1.1 All-atom MD simulation of BPTI in NaCl solution
 - 1.2 All-atom MD simulation of POPC lipid bilayer
 - 1.3 Coarse-grained MD simulation with the Go model
2. Replica-exchange molecular dynamics simulations
 - 2.1 Temperature REMD simulations of a small peptide
 - 2.2 Replica-exchange umbrella sampling
3. Advanced molecular dynamics simulations
 - 3.1 Large scale simulation using the parallel I/O
 - 3.2 String method
4. Trajectory analysis tools
 - 4.1 Root-mean-square deviation (RMSD)
 - 4.2 Distance, angle, dihedral angle
 - 4.3 Root-mean-square fluctuation (RMSF)
 - 4.4 Principal component analysis (PCA)
 - 4.5 Weighted histogram analysis method (WHAM) (coming soon!)

水中のBPTI(牛膝臓トリプシン阻害酵素)のMD計算



GENESIS
Generalized-ensemble simulation system

Home Download Installation Usage Tutorials Benchmark Publications Forum Link About

Home > Tutorials > Tutorials of basic MD simulations > Tutorial 1.1

Search

Search ... Search

News

GENESIS tutorial in Japanese at Tokyo (Oct. 5, 2017)
Aug 25th, 2017

GENESIS 1.1 paper available online!
Aug 3rd, 2017

1.1 All-atom MD simulation of BPTI in NaCl solution

Files for this tutorial (tutorial-1.1.tar.gz)

Contents [hide]

- 1.1.1 Building a simulation system
- 1.1.2 Minimization with restraints on the protein
- 1.1.3 Heat-up with restraints on the protein
- 1.1.4 Equilibration simulation
- 1.1.5 Production simulation

1. シミュレーションシステムの作成 (1_setup/)

現在のGENESISはシミュレーションシステムの作成するツールはありません
CHARMM-GUI/VMDなどを利用して作成してください

今回は時間の都合により、予め作成した分子ファイルを用います
詳しくは“1.1.1 Building a simulation system”をご覧ください

Tutorial 1: 構造最適化(2_minimization/)

初期構造では、原子間の距離が近すぎたり、不自然な原子結合を持つようなことがあります。不安定なシミュレーションを避けるため、本計算を行う前に“平衡化”と呼ばれる予備計算を十分に行います。

構造最適化は、その第1段階でエネルギー値が下がる方向に粒子を動かします

GENESISでは現在、最急降下法(Steepest Decent)を使うことができます

Tutorial_1/2_minimization/
run.inp : GENESIS入力ファイル
run.sh : ジョブスクリプト
output/ : 計算結果

構造最適化の入力ファイル

```
[INPUT]
topfile = ../1_setup/top_all27_prot_lipid.rtf # topology file
parfile = ../1_setup/par_all27_prot_lipid.prm # parameter file
psffile = ../1_setup/ionize.psf           # protein structure file
pdbfile = ../1_setup/ionize.pdb          # PDB file
reffile = ../1_setup/ionize.pdb          # reference for restraints

[OUTPUT]
dcdfile = run.dcd # DCD trajectory file
rstfile = run.rst # restart file

[ENERGY]
electrostatic = PME # [CUTOFF,PME]
switchdist = 10.0 # switch distance
cutoffdist = 12.0 # cutoff distance
pairlistdist = 13.5 # pair-list distance
contact_check = YES
pme_ngrid_x = 72 # grid size_x in [PME]
pme_ngrid_y = 80 # grid size_y in [PME]
pme_ngrid_z = 72 # grid size_z in [PME]

[MINIMIZE]
nsteps = 1000 # number of steps
eneout_period = 100 # energy output period
crdout_period = 100 # coordinates output period

rstout_period = 1000 # restart output period

[BOUNDARY]
type = PBC # [PBC,NOBC]
box_size_x = 70.8250 # box size (x) in [PBC]
box_size_y = 83.2579 # box size (y) in [PBC]
box_size_z = 69.0930 # box size (z) in [PBC]

[SELECTION]
group1 = an:CA | an:C | an:O | an:N # index of restraint group 1

[RESTRAINTS]
nfunctions = 1 # number of functions
function1 = POSI # restraint function 1
constant1 = 10.0 # force constant
select_index1 = 1 # restrained group
```

Coulomb相互作用でPMEを用いる

位置拘束
どの粒子に拘束をかけるのかは
[SELECTION]で選ぶ
([SELECTION]に関してはManual 11、または巻末を参照)

原子間距離を確認し「Bad contact」や不自然な距離をレポートし、さらに力の値に制限をかける
(最適化・平衡化のみで使用してください)

長距離相互作用で用いるFFTのグリッド数
(2,3,5の倍数かつ、並列数によって最適値が変わる、一般的にはグリッドサイズが1 Å程度になるようにする。詳しくはマニュアルの5.3.1を参照)

ステップ数

構造最適化の実行

計算の実行

```
% cd ~/Tutorial_1/2_minimization  
% ./run.sh
```

Jobが終わったら以下のファイルが出力されているはずです

run.dcd : dcd形式のトラジェクトリファイル (binary)
run.out : GENESIS出力ファイル (ascii)
run.rst : GENESISリスタートファイル (binary)

出力ファイル(run.out)

```
*****  
*                                     *  
*      GENESIS SPDYN                 *  
*                                     *  
*      A Molecular Dynamics Simulator with  
*      Spatial Decomposition Scheme   *  
*                                     *  
*      Developed by RIKEN AICS       *  
*                                     *  
*****
```

出力ファイルの見方

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

[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

[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

INFO:	STEP	POTENTIAL_ENE	RMSG	BOND	ANGLE	UREY-BRADLEY	DIHEDRAL	IMPROPER	CMAP	VDWAALS	ELECT RESTRAINT_TOTAL	
INFO:	0	-101597.1585	30.2479	11247.1385	2939.9532	74.9561	260.8373	62.6065	-72.0093	11798.1692	-127908.8099	0.0000
INFO:	100	-114456.2091	5.4597	3977.1511	2353.8183	51.5709	256.4578	22.6346	-74.3462	9730.9841	-130775.5114	1.0317

[STEP 6] : 終端処理と計測時間

Output_Time> Averaged timer profile (Min, Max)	nonbond	=	109.986 (109.658,	110.141)				
total time	=	137.159	pme real	=	83.219 (81.500,	84.919)		
setup	=	5.843	pme recip	=	26.077 (24.516,	27.847)		
dynamics	=	131.316	restraint	=	0.221 (0.178,	0.251)		
energy	=	122.896	integrator						
integrator	=	0.838	constraint	=	0.000 (0.000,	0.000)		
pairlist	=	5.774 (5.629,	5.963)	update	=	0.000 (0.000,	0.000)
energy					comm_coord	=	0.000 (0.000,	0.000)
bond	=	0.657 (0.587,	0.708)	comm_force	=	0.000 (0.000,	0.000)
angle	=	0.717 (0.696,	0.751)	comm_migrate	=	0.000 (0.000,	0.000)
dihedral	=	1.275 (1.236,	1.295)					

Tutorial 1:温度上昇シミュレーション (3_heating/)

構造最適化で、ある程度の安定構造にした後、システムの温度をゆっくり変化させ、本計算で用いる温度まで上昇させる。

GENESISでは'Annealing' オプションを使ってゆっくり温度を変化させるようなシミュレーションができます

Tutorial_1/3_heating/
run.inp : GENESIS入力ファイル
run.sh : ジョブスクリプト
output/ : 計算結果

入力ファイル抜粋

[INPUT]

```
topfile = ../1_setup/top_all27_prot_lipid.rtf # topology file  
(skip)  
rstfile = ../2_minimization/run.rst # restart file
```

[DYNAMICS]

```
integrator = LEAP <# [LEAP,VVER]  
nsteps = 5000 # number of MD steps  
timestep = 0.002 # timestep (ps)  
(skip)  
annealing = YES # simulated annealing  
anneal_period = 50 # annealing period  
dtemperature = 3 # temperature change at annealing (K)
```

LEAP: Leapfrog
VVER: Velocity Verlet

50 stepに1回、3Kずつ
温度を上げる

[CONSTRAINTS]

```
rigid_bond = YES # constraints all bonds  
# involving hydrogen
```

[ENSEMBLE]

```
ensemble = NVT # [NVE,NVT,NPT]  
tpcontrol = LANGEVIN # thermostat  
temperature = 0.1 # initial temperature (K)
```

チュートリアルでは時間の関係でこのステップ数にしていますが、実際はもっと長い計算を行います

温度上昇シミュレーションの実行

計算の実行

```
% cd ~/Tutorial_1/3_heating  
% ./run.sh
```

計算が終わったら以下のファイルが出力されているはずです

run.dcd : dcd形式のトラジェクトリファイル (binary)

run.out : GENESIS出力ファイル (ascii)

run.rst : GENESISリスタートファイル (binary)

出力ファイル抜粋

[STEP 5]

```
INFO:  STEP      TIME    TOTAL_ENE  POTENTIAL_ENE  KINETIC_ENE      RMSG      BOND      ANGLE  UREY-  
BRADLEY  DIHEDRAL  IMPROPER    CMAP    VDWAALS      ELECT RESTRAINT_TOTAL  TEMPERATURE  
VOLUME  
-----  
(..skip..)  
Simulated_Annealing_Leapfrog> Anneal temperature from 213.100 to 216.100  
INFO:  3650      7.3000  -121015.8486  -134668.2442  13652.3956      14.8398      129.1861      316.5600  
34.9546      301.9510      20.3742      -76.0000      17444.0571  -152886.2023      46.8752      184.8766      407423.5098  
  
Simulated_Annealing_Leapfrog> Anneal temperature from 216.100 to 219.100  
INFO:  3700      7.4000  -120524.6696  -134365.7978  13841.1281      14.8065      135.7281      319.1936  
42.6740      291.7295      14.9113      -82.0820      17460.5756  -152587.8837      39.3559      187.4323      407423.5098
```

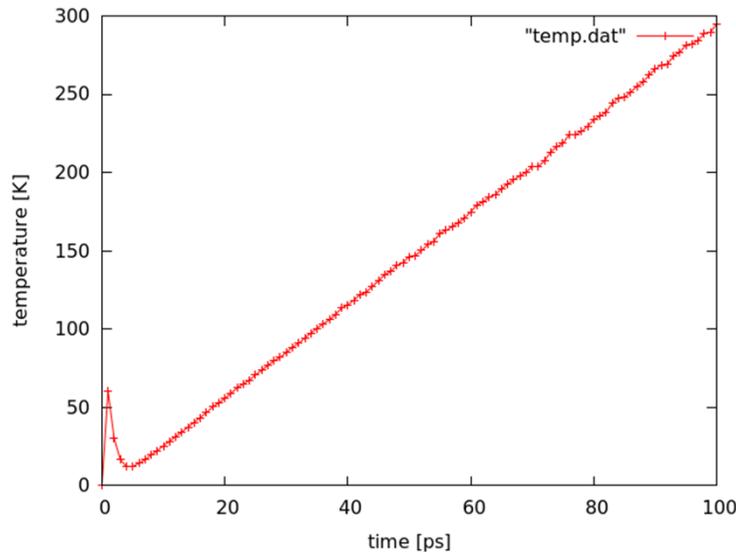
計算結果の確認

■ 温度の変化を確認します

- GENESIS出力ファイルから各ステップでの温度を抽出し、gnuplotでプロットします (今回は省略)

```
% sh temp.sh  
% gnuplot  
gnuplot> plot "temp.dat" w lp
```

実行結果



```
$ cat temp.sh  
#!/bin/bash  
  
grep "^INFO" output/run.out | tail -n +2 | awk '{print $3,  
$17}' >temp.dat
```

Tutorial 1:平衡化(4_equilibration/)

システムを計算させたい温度に上げた後、本計算と同じ条件の計算を行い、システムを平衡化させる (研究を行う上では、非常に重要なステップ)

Tutorial_1/4_equilibration/
run.inp : GENESIS入力ファイル
run.sh : ジョブスクリプト
output/ : 計算結果

入力ファイル抜粋

トラジェクトリ書き出しを行い、温度・エネルギーだけでなく、中の分子も平衡に達していることも確認する

(3)と同様にrun.shを実行してください

[OUTPUT]
dcdfile = run.dcd # DCD trajectory file

[ENERGY]
[CONSTRAINTS]
[RESTRAINS (今回はない)]
[ENSEMBLE]
これらのパラメータは基本的に本計算と同じものを使う

[ENSEMBLE]
ensemble = NPT # [NVE,NVT,NPT]
tpcontrol = LANGEVIN # thermostat and barostat
temperature = 300.0 # initial temperature (K)
pressure = 1.0 # target pressure (atm)

チュートリアルでは時間の関係でこのステップ数にしていますが、実際はもっと長い計算を行います (温度・エネルギーの安定性だけでなく、分子の観測したい性質が安定することも確認します)

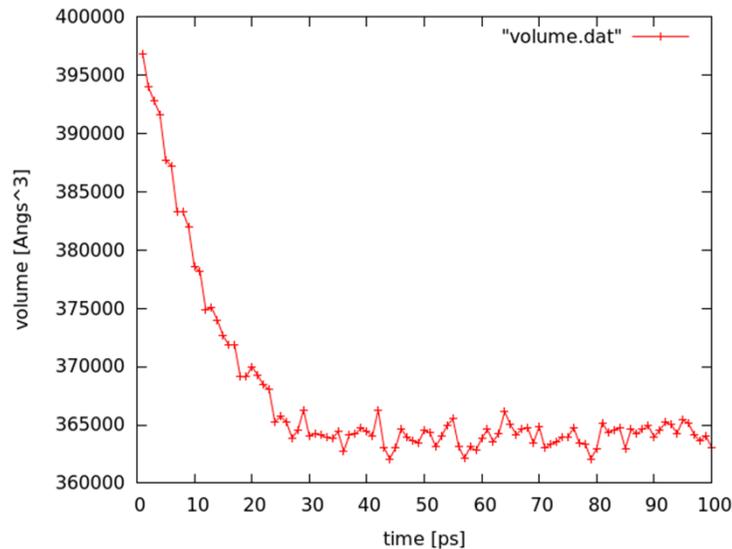
計算結果の確認

■ 体積の変化を確認します

- GENESIS出力ファイルから各ステップでの体積を抽出し、gnuplotでプロットします(今回は省略)

```
$ sh vol.sh  
$ gnuplot  
gnuplot> plot "vol.dat" w lp
```

実行結果



```
$ cat vol.sh  
#!/bin/bash  
  
grep "^INFO" output/run.out | tail -n +2 | awk '{print $3,  
$17}' > vol.dat
```

Tutorial 1:本計算(5_production/)

実際にデータを取るための計算

Tutorial_1/5_production/

run.inp : GENESIS入力ファイル

run.sh : ジョブスクリプト

output/ : 計算結果

(3)と同様にrun.shを
実行してください

入力ファイル抜粋

トラジェクトリ書き出しを行うことで、研究に
用いる様々な性質を計算する

```
[OUTPUT]  
dcdfile = run.dcd # DCD trajectory file
```

チュートリアルでは時間の関係でこのステップ数にしていますが、
実際はもっと長い計算を行います。

Tutorial 1:解析(6_analysis/)

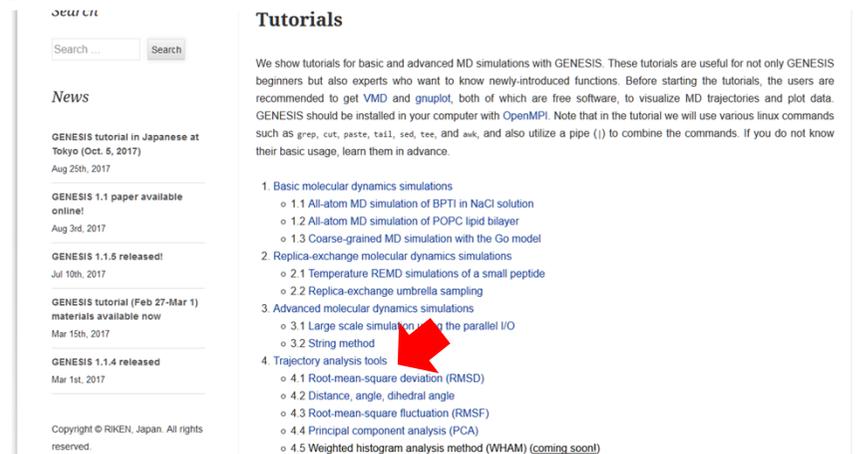
■ 求めたトラジェクトリから様々な解析を行います

- GENESISは、得られたトジェクトリーを必要に応じて加工し、基本的な解析を行う独自のツールを提供します

(例)

```
./6_analysis:
```

```
1_RMSD          # RMSDの計算
2_DIST          # 距離の計算
3_RMSF          # RMSFの計算
4_PCA           # PCA解析
```



The screenshot shows the GENESIS website with a search bar and a 'News' section. The 'Tutorials' section is visible, listing various simulation and analysis tutorials. A red arrow points to the 'Trajectory analysis tools' section, which includes sub-items like '4.1 Root-mean-square deviation (RMSD)', '4.2 Distance, angle, dihedral angle', '4.3 Root-mean-square fluctuation (RMSF)', '4.4 Principal component analysis (PCA)', and '4.5 Weighted histogram analysis method (WHAM) (coming soon!)'.

- 今回は一例として、「trjana/rmsd_analysis」ツールを用いて、システムの中からC α 原子のみ抜き出しそのRMSD(根二乗平均変位:基準の座標からどれだけずれたかを見る)を計算します

```
./1_RMSD:
```

```
run.inp          # GENESISの入力ファイル
run.sh           # バッチスクリプト
```

解析の入力ファイル

■ Trajectory、selection、fittingで解析対象と内容を指定します

```
[INPUT]
reffile = ../../1_setup/ionize.pdb # PDB file

[OUTPUT]
rmsfile = run.rms # RMSD file

[TRAJECTORY]
trjfile1 = ../../5_production/output/run.dcd # trajectory
file
md_step1 = 500000 # number of MD steps
mdout_period1 = 500 # MD output period
ana_period1 = 500 # analysis period
repeat1 = 1
trj_format = DCD # (PDB/DCD)
trj_type = COOR # (COOR/COOR+BOX)

trj_natom = 0 # (0:uses reference PDB atom count)

[SELECTION]
group1 = an:CA # selection group 1

[FITTING]
fitting_method = TR+ROT # method
fitting_atom = 1 # atom group
mass_weight = YES # mass-weight is applied

[OPTION]
check_only = NO # (YES/NO)
analysis_atom = 1 # atom group
```

Annotations:

- trj_natom = 0: C α のみ選択
- group1 = an:CA: C α のみ選択
- fitting_atom = 1: 基準構造に対する重ね合わせを行う (TR+ROT: 並進と回転で重ね合わせする)
- check_only = NO: YES: 本計算しない
- repeat1 = 1: 複数trjfileを扱う場合に指定簡略化

■ run.shの内容は基本的にanalysisツールを実行するものです

```
rmsd_analysis run.inp > run.out
```

'trj_analysis'を用いる事で、距離・角度・二面角などを計算することも可能

GENESISのトラブルシューティング

- GENESISが異常終了する
 1. “contact_check” optionをYES, nstep=20で計算
MDステップに入る前の段階で構造のチェックを行います
ログに”too short distance”などのメッセージが表示された場合は、構造が不安定ですので構造最適化・平衡化を行ってください
 2. ./configure --enable-debug=3を用いて再コンパイル
原子数などを確認しながら計算を行います
配列外アクセスが起きた際にはエラーを返しますので、原子数の最大値を変更するなどしてください
 3. GENESISのForumに投稿してください