

2016年度 公開ソフト講習会 第16回

# GENESIS講習会(講義)

理化学研究所 計算科学研究機構  
粒子系生物物理研究チーム

講演者: 松永康佑  
ymatsunaga@riken.jp

2017年1月13日 FOCUS(神戸)

# 今日の内容

## ■ 10:00 – 11:30 講義

- 生体分子の分子動力学計算
- GENESISの紹介
- 実習に必要な理論の説明

## ■ お昼休憩(90分)

## ■ 13:00 – 16:00 実習

- GENESISのコンパイル
- 実行例(1) 分子動力学法計算
- GENESISのトラブルシューティング
- 実行例(2) レプリカ交換分子動力学法計算と自由エネルギー計算

# 今日の内容

## ■ 10:00 – 11:30 講義

- 生体分子の分子動力学計算
- GENESISの紹介
- 実習に必要な理論の説明

## ■ お昼休憩(90分)

## ■ 13:00 – 16:00 実習

- GENESISのコンパイル
- 実行例(1) 分子動力学法計算
- GENESISのトラブルシューティング
- 実行例(2) レプリカ交換分子動力学法計算と自由エネルギー計算

# 分子動力学法(Molecular Dynamics; MD)

- 粒子間の相互作用力を計算し、ニュートンの運動方程式を解く

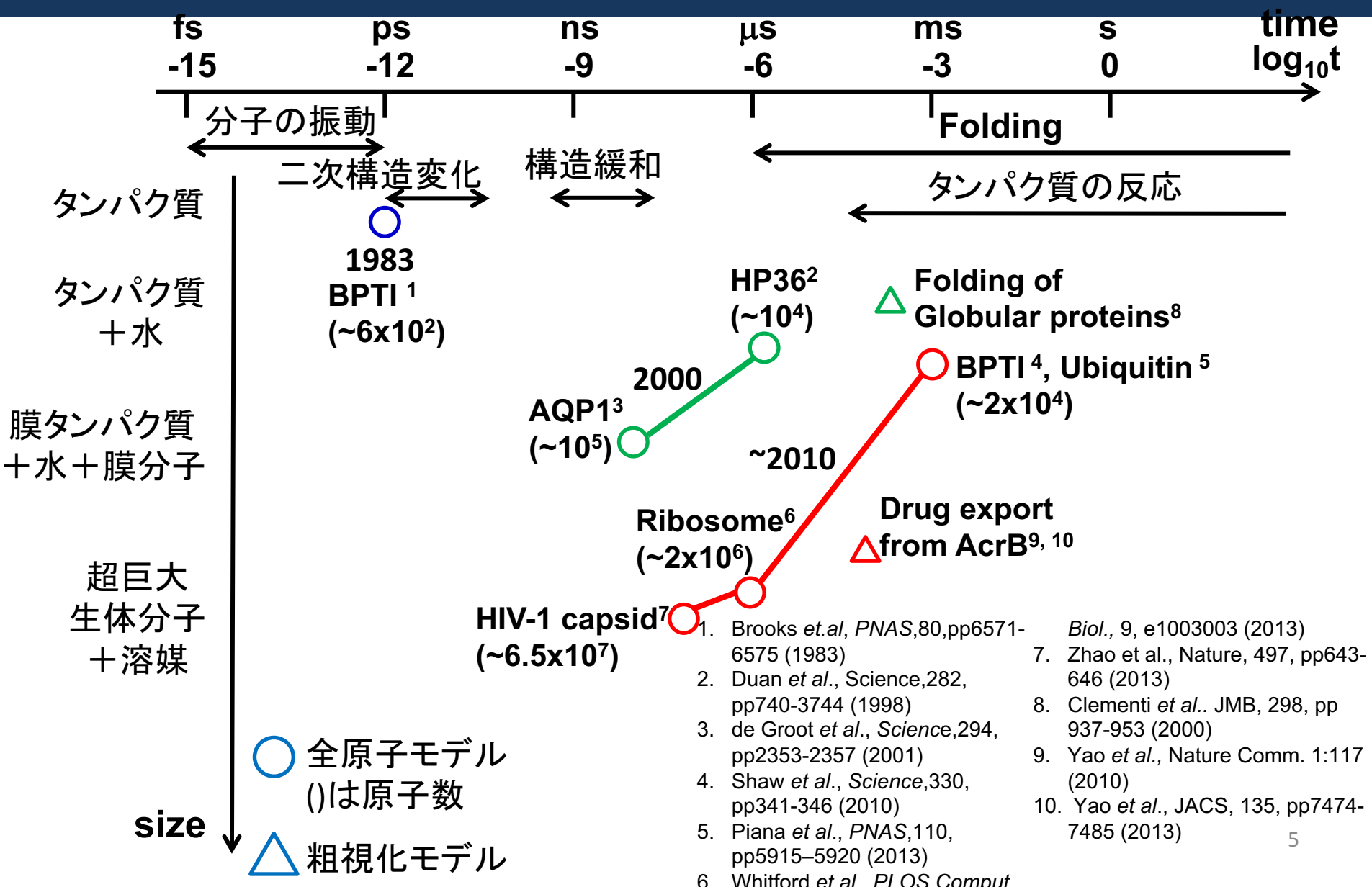
$$m_i \frac{d^2 \mathbf{r}_i(t)}{dt^2} = \mathbf{F}_i(t) = - \frac{dU(\mathbf{r}^N)}{d\mathbf{r}_i}$$

相互作用エネルギー  
粒子の座標

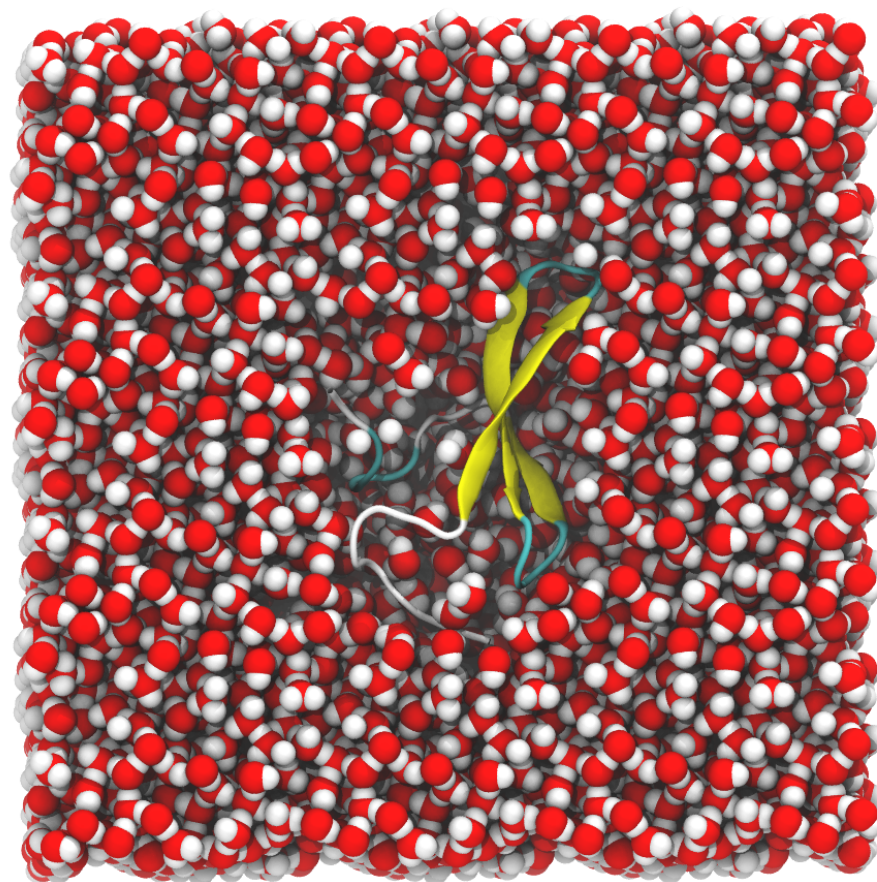
- 粒子の運動から、構造の安定性、構造ゆらぎ、更にアンサンブルを計算し、自由エネルギー面(結合エネルギーなど)を解析できる
- MD計算はとても時間がかかる...
  - タンパク質のダイナミクスをミリ秒計算するには、ニュートンの運動方程式を $10^{12}$ 回計算する必要がある
  - 例えば1stepに1msかかるなら、約31.7年必要！



# タンパク質のシミュレーション

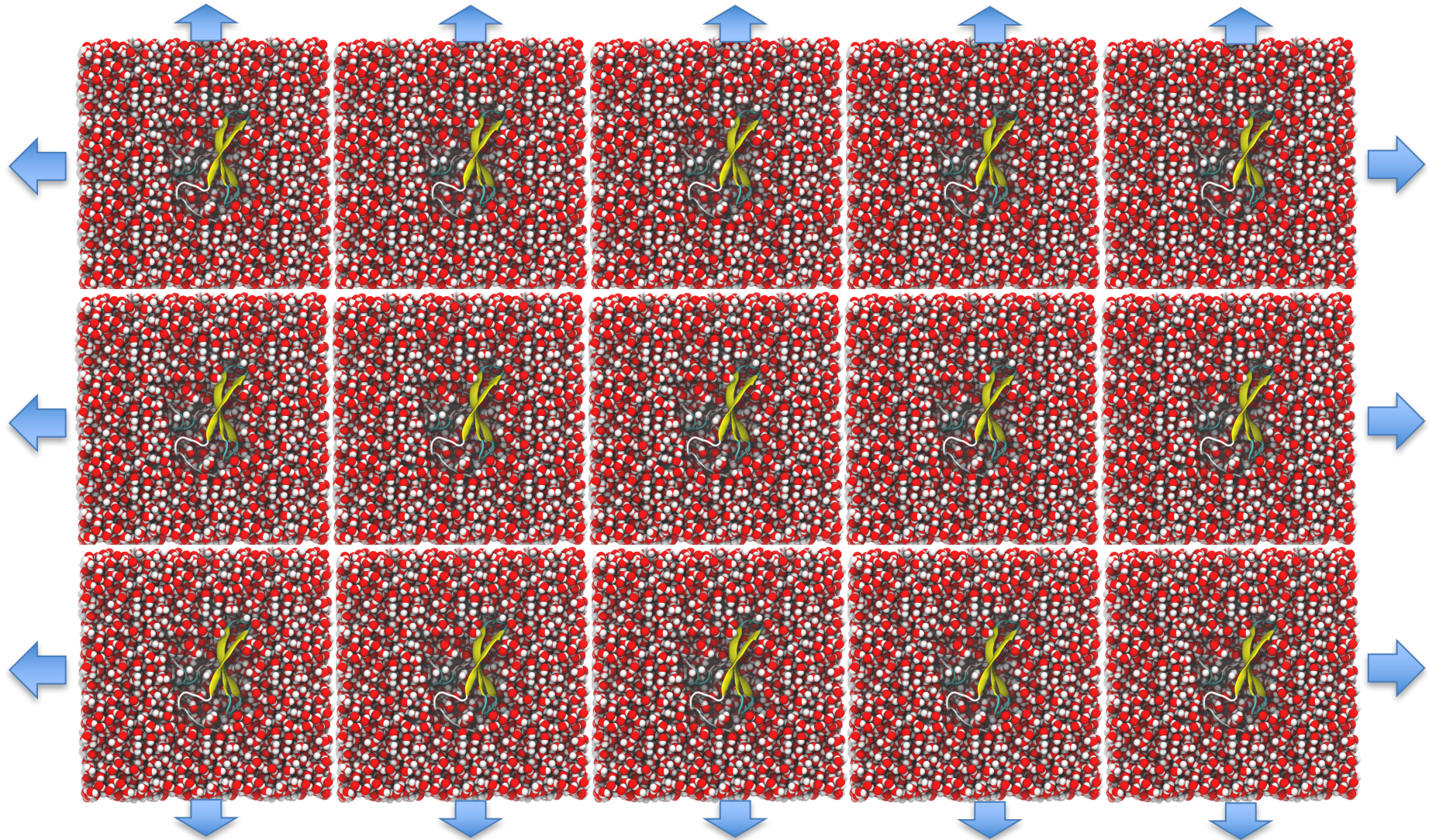


# 典型的なシミュレーション系



溶媒(水のモデル)を含めて全部で数万原子  
周期境界条件で計算

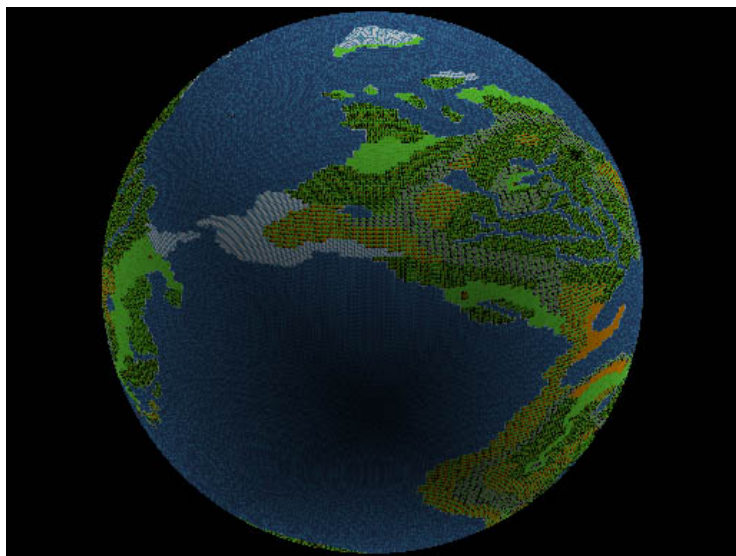
# 周期境界条件



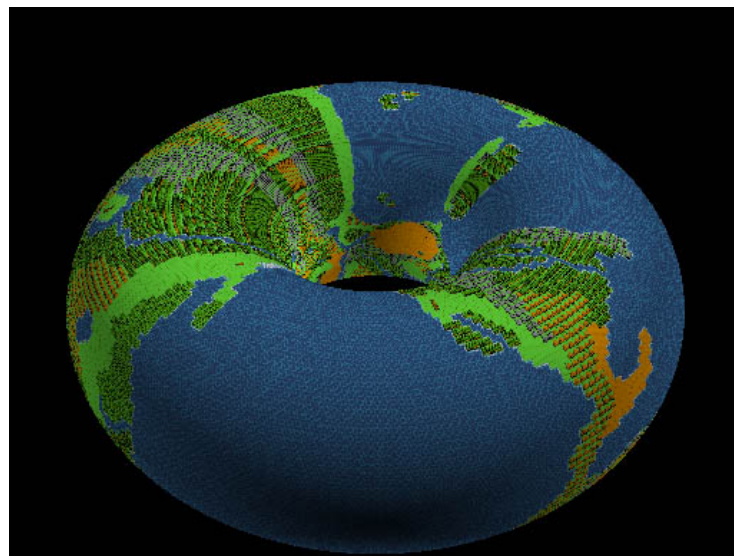
表面等の影響を排除して、バルクの性質を模倣する  
トポロジ的には3次元トーラス(次ページ)

# 小話: 二次元周期境界条件のトポロジーはトーラス

我々が住む世界



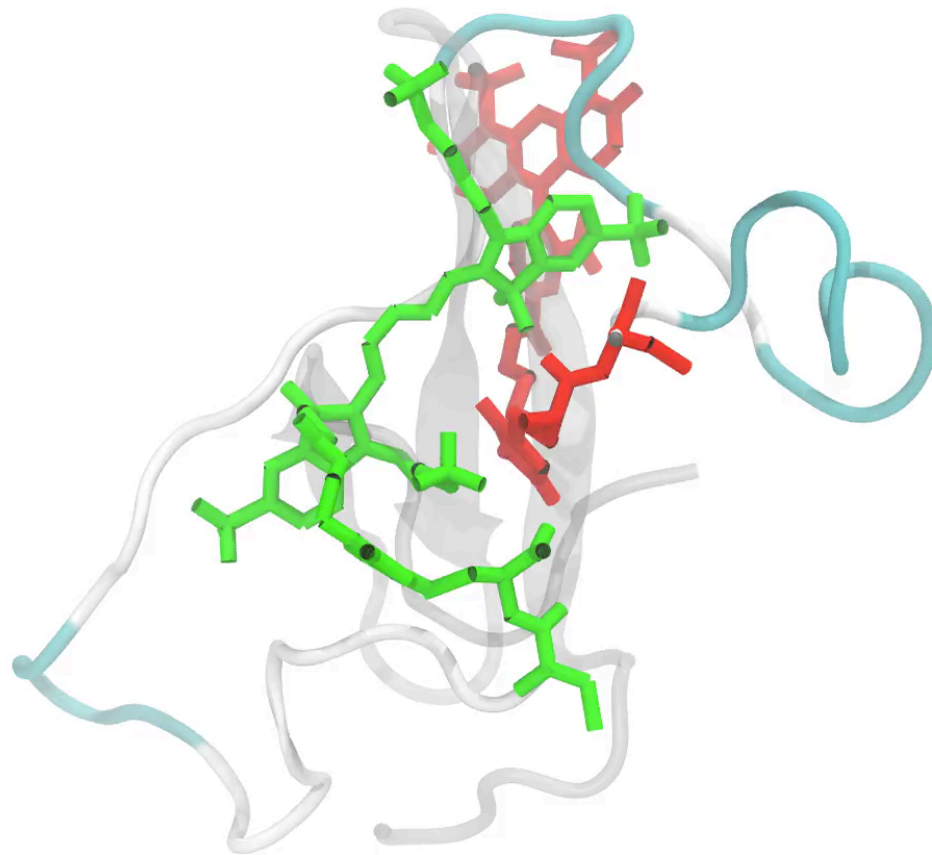
ドラクエの世界  
(二次元周期境界条件)



<http://www.ton.scphys.kyoto-u.ac.jp/~michikaz/crazy/rpgmap.html>



# MDで得られるタンパク質の動きの典型例



# タンパク質内外の相互作用

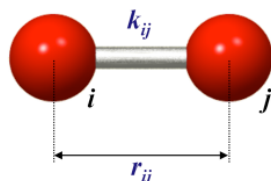
参照: 室温における熱エネルギー  $\sim 0.6 \text{ kcal/mol}$

non-covalent interactions	Interaction	Example	Typical distance	Energy
	Covalent bond	$\text{—C—C—}$	1.5 Å	$\sim 100 \text{ kcal/mol}$
	Salt bridge	$\begin{array}{c} \text{H} \\   \\ \text{—N—H}^+ \\   \\ \text{H} \end{array} \quad \begin{array}{c} \text{O} \\ \diagup \\ \text{—C—} \\ \diagdown \\ \text{O}^- \end{array}$	3.0 Å	$\sim -4.0 \text{ kcal/mol}$ (in water)
	Hydrogen bond	$\text{>N—H} \cdots \text{O=C<}$	3.0 Å	$\sim -1.0 \text{ kcal/mol}$ (in water)
	van der Waals	$\begin{array}{c} \text{H} \\   \\ \text{—C—H} \\   \\ \text{H} \end{array} \quad \begin{array}{c} \text{H} \\   \\ \text{H—C—} \\   \\ \text{H} \end{array}$	3.5 Å	$\sim -0.1 \text{ kcal/mol}$ (in water)
	Electrostatic	$\begin{array}{c} \text{H} \\   \\ \text{—N—H}^+ \\   \\ \text{H} \end{array} \quad \begin{array}{c} \text{O} \\ \diagup \\ \text{—C—} \\ \diagdown \\ \text{O}^- \end{array}$	various	Depend on distance and environment

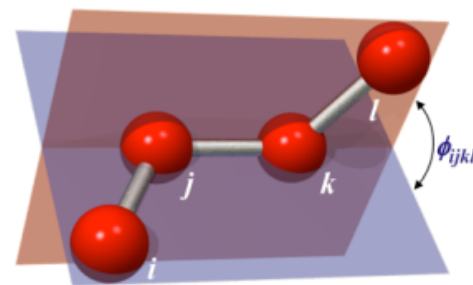
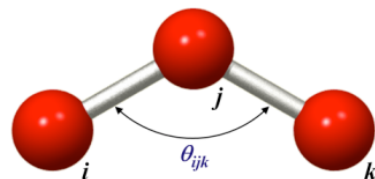
長距離相互作用が複雑に絡み合って生体分子の  
機能が発揮されている

# MDに使われる経験的なポテンシャル関数

$$V(\mathbf{r}) = \sum_{\text{bonds}} k_b (r_{i,i+1} - r_0)^2 + \sum_{\text{angles}} k_a (\theta_{i,i+1,i+2} - \theta_0)^2 + \sum_{\text{torsions}} k_t \left[ 1 + \cos(n\phi_{i,i+1,i+2,i+3} - \omega) \right]$$



Bonded interactions  
O(N) interactions



$$+ \sum_i \sum_{j>i} \varepsilon \left[ \left( \frac{\sigma}{r_{ij}} \right)^{12} - \left( \frac{\sigma}{r_{ij}} \right)^6 \right]$$

Non-bonded interactions  
vdW term **O(N<sup>2</sup>) interactions**

$$+ \sum_i \sum_{j>i} \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}}$$

Electrostatic term

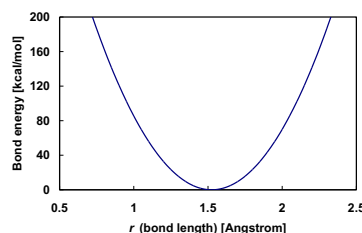
**Bottleneck**

# Bonded interactions

- Bond stretching term

$$\sum_{\text{bonds}} K_b (r - r_0)^2$$

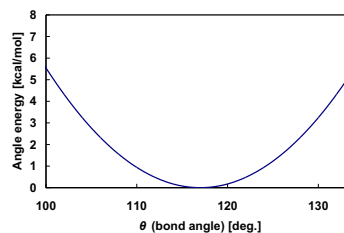
( $K_b$ ; force constant,  $r$ ; bond length,  
 $r_0$ ; equilibrium bond length)



- Angle bending term

$$\sum_{\text{angles}} K_\theta (\theta - \theta_0)^2$$

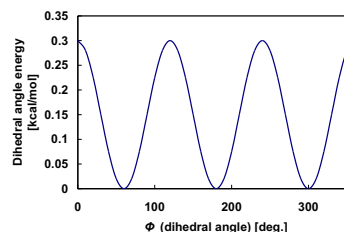
( $K_\theta$ ; force constant,  $\theta$ ; bond angle,  
 $\theta_0$ ; equilibrium bond angle)



- Torsion term

$$\sum_{\text{dihedrals}} K_\phi [1 + \cos(n\phi - \gamma)]$$

( $K_\phi$ ; force constant,  $\phi$ ; torsion angle,  
 $n$ ; multiplicity,  $\gamma$ ; phase)



The number of pairs examined in the bonded interaction terms grows only linearly with number of atoms,  $O(N)$ .

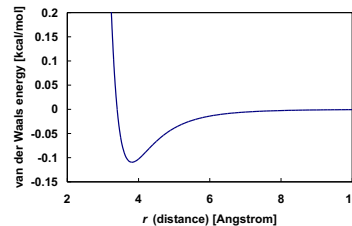


# Non-bonded interactions

- van der Waals term

$$\sum_{i, j \text{ pairs}} \left( \frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^6} \right)$$

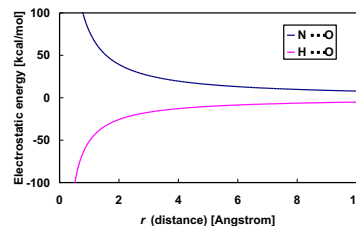
( $A_{ij}$  and  $B_{ij}$ ; force constants,  
 $r_{ij}$ ; distance between atoms  $i$  and  $j$ )



- Electrostatic term

$$\sum_{i, j \text{ pairs}} \frac{q_i q_j}{\epsilon r_{ij}}$$

( $q_i$  and  $q_j$ ; atomic charges of atoms  $i$  and  $j$ , respectively,  $\epsilon$ ; dielectric constant,  $r_{ij}$ ; distance between atoms  $i$  and  $j$ )



Opposite charges



Same charges

**Unlike the van der Waals interaction, the electrostatic term is long-range interaction!**

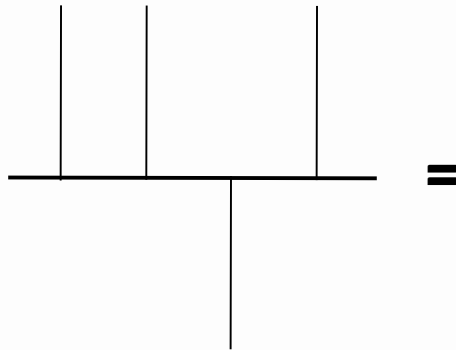
The number of pairs examined in the non-bonded interaction terms grows quadratically with number of atoms,  $O(N^2)$ .

# Non-bonded interactionsを効率よく計算するための工夫: Ewald summation

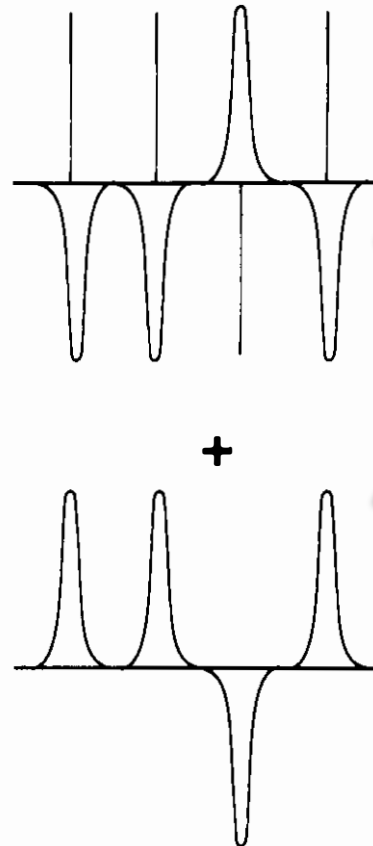
離散的な電荷分布を打ち消す滑らかな仮想電荷分布を導入することにより、  
クーロン相互作用を2つに分ける

Frenkel & Schmit, "Understanding Molecular Simulation"

離散的な電荷



=



Direct part

(遠い電荷の影響は打ち消されている、cutoffで解く)

仮想的な電荷分布の導入

Smooth part

(滑らかな電荷分布のみ存在。メッシュで近似して3D-FFTで解く)

# Direct partの計算法

Cutoff 距離を設定して、その距離以内の原子ペアについて計算する

並列化方法

原子分割 (atom decomposition)

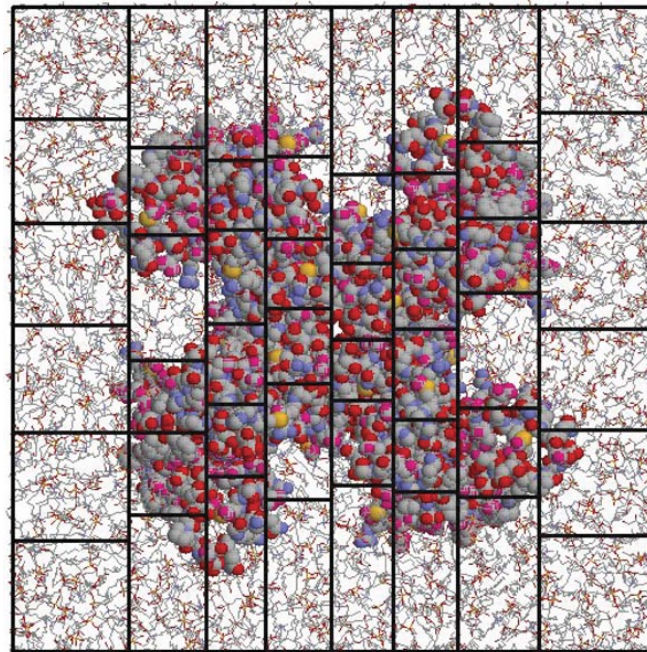
予め静的に原子を各ノードに割り振る

領域分割 (domain decomposition)

原子の存在する場所によってノードへ割り振る

→Cutoff内の原子が近くのノードに存在するため、並列化の  
パフォーマンスが良い

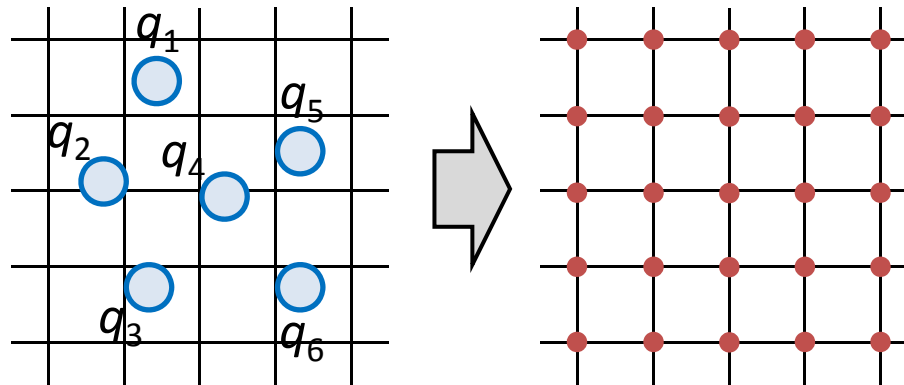
領域分割の例



Larsson, et al., *WIREs* (2011)

# Smooth partの計算法

- Particle Mesh Ewald法(Darden, et al., *JCP* (1993)) を使って3D-FFTにより解くのがスタンダード
  - 電荷分布をメッシュ上で近似して3D FFTする方法



しかし、一般に3D-FFTはtransposeの際の通信量が大きく、並列化に向かない

- そこで、real-space poisson solverを使う枠組みが提案されている。  
(real-space Gaussian Split Ewald法、Shan et al., *JCP* (2005))
- Multigrid法(階層的なグリッドを使ってimplicitに解く方法、収束が早い)が使えるそうだが、遠いノードとの通信を必要とするのでトーラスネットワークに不向き。
- まだ模索中。

# 今日の内容

## ■ 10:00 – 11:30 講義

- 生体分子の分子動力学計算
- **GENESISの紹介**
- 実習に必要な理論の説明

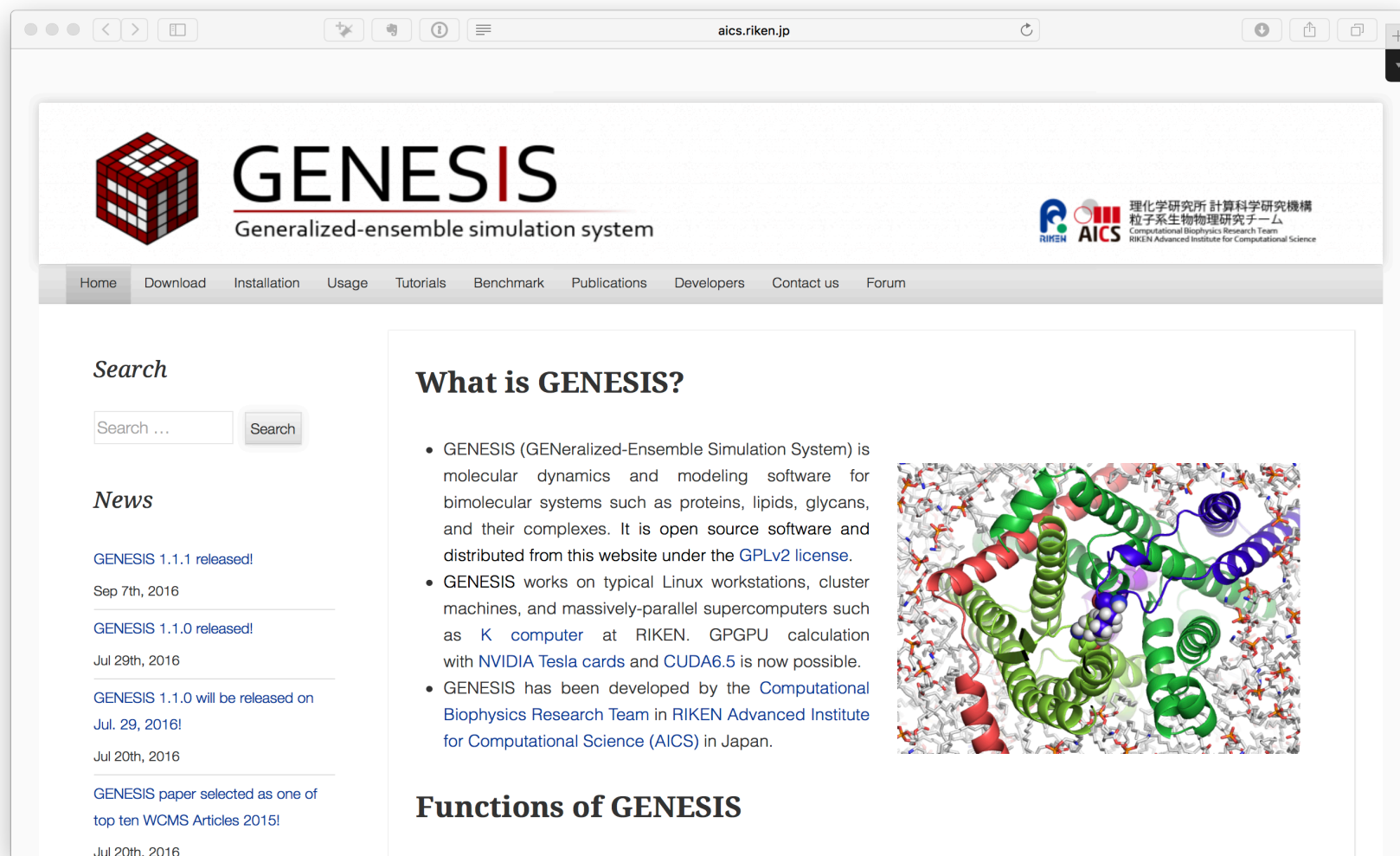
## ■ お昼休憩(90分)

## ■ 13:00 – 16:00 実習

- GENESISのコンパイル
- 実行例(1) 分子動力学法計算
- GENESISのトラブルシューティング
- 実行例(2) レプリカ交換分子動力学法計算と自由エネルギー計算

# GNESISの紹介

<http://www.aics.riken.jp/labs/cbrt/>



The screenshot shows the GENESIS website homepage. At the top, there is a navigation bar with the GENESIS logo (a red and white cube) and the text "GENESIS Generalized-ensemble simulation system". To the right of the logo is the AICS logo and the text "理化学研究所 計算科学研究機構 粒子系生物物理研究チーム Computational Biophysics Research Team RIKEN Advanced Institute for Computational Science". Below the navigation bar is a search bar and a "Search" button. The main content area is divided into two columns. The left column contains a "News" section with four items: "GENESIS 1.1.1 released!" (Sep 7th, 2016), "GENESIS 1.1.0 released!" (Jul 29th, 2016), "GENESIS 1.1.0 will be released on Jul. 29, 2016!" (Jul 20th, 2016), and "GENESIS paper selected as one of top ten WCMS Articles 2015!" (Jul 20th, 2016). The right column contains a "What is GENESIS?" section with three bullet points: "GENESIS (GENERALIZED-ENSEMBLE SIMULATION SYSTEM) is molecular dynamics and modeling software for bimolecular systems such as proteins, lipids, glycans, and their complexes. It is open source software and distributed from this website under the [GPLv2 license](#).", "GENESIS works on typical Linux workstations, cluster machines, and massively-parallel supercomputers such as [K computer](#) at RIKEN. GPGPU calculation with [NVIDIA Tesla cards](#) and [CUDA6.5](#) is now possible.", and "GENESIS has been developed by the [Computational Biophysics Research Team](#) in [RIKEN Advanced Institute for Computational Science \(AICS\)](#) in Japan." Below this is a "Functions of GENESIS" section. To the right of the text is a 3D molecular model of a protein structure, showing a green helix and a blue helix, with a red helix and a purple helix also visible. The background is a white grid.

**Search**

Search ...

**News**

[GENESIS 1.1.1 released!](#)  
Sep 7th, 2016

[GENESIS 1.1.0 released!](#)  
Jul 29th, 2016

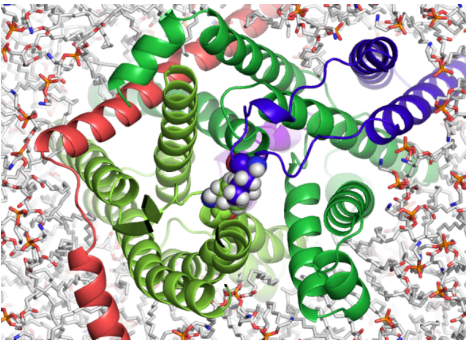
[GENESIS 1.1.0 will be released on Jul. 29, 2016!](#)  
Jul 20th, 2016

[GENESIS paper selected as one of top ten WCMS Articles 2015!](#)  
Jul 20th, 2016

**What is GENESIS?**

- GENESIS (GENERALIZED-ENSEMBLE SIMULATION SYSTEM) is molecular dynamics and modeling software for bimolecular systems such as proteins, lipids, glycans, and their complexes. It is open source software and distributed from this website under the [GPLv2 license](#).
- GENESIS works on typical Linux workstations, cluster machines, and massively-parallel supercomputers such as [K computer](#) at RIKEN. GPGPU calculation with [NVIDIA Tesla cards](#) and [CUDA6.5](#) is now possible.
- GENESIS has been developed by the [Computational Biophysics Research Team](#) in [RIKEN Advanced Institute for Computational Science \(AICS\)](#) in Japan.

**Functions of GENESIS**



GPL v2の下でソースコードを公開



# GENESIS

Generalized-ensemble simulation system



理化学研究所 計算科学研究機構  
粒子系生物物理研究チーム  
Computational Biophysics Research Team  
RIKEN Advanced Institute for Computational Science

- 生体系において、効率的で精度の良い自由エネルギー計算手法の開発が目的
- 高並列計算 - 京コンピュータなどでの超並列計算が可能  
(ただし、普通のPCクラスタでも動きます)
- 巨大な生体系も可能
- 全原子モデルのみでなく、粗視化モデル等異なる分子モデルへも応用できるアルゴリズムを採用
- レプリカ交換法(拡張アンサンブル法の一つ)による自由エネルギー計算も可能



# GENESIS

Generalized-ensemble simulation system



理化学研究所 計算科学研究機構  
粒子系生物物理研究チーム  
Computational Biophysics Research Team  
RIKEN Advanced Institute for Computational Science

## ■ 開発チーム

計算科学研究機構(AICS), 粒子系生物物理研究チーム

プロジェクトリーダー: 杉田有治

主開発者: 鄭載運、森貴治、小林千草、松永康佑、安藤格士(理科大)、神谷基司、依田隆夫(長浜バイオ大)、Michael Feig(ミシガン州立大)

公開バージョン : Ver. 1.1.0

## ■ 参考文献

J. Jung, T. Mori, C. Kobayashi, Y. Matsunaga, T. Yoda, M. Feig, and Y. Sugita, *WIREs Comput. Mol. Sci.*, 5, 310-323 (2015).

## ■ 問い合わせ先

genesis@riken.jp



# GENESISの高速化・高並列化

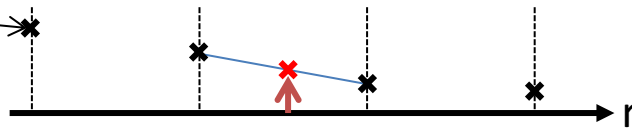
- 最も時間のかかる非結合相互作用計算を高速化するため  
GENESISでは下のアルゴリズムを新規に開発
  - **Inverse lookup table approach** (近距離相互作用計算の高速化)  
(Jung *et al.*, *J. Comput. Chem.*, **34**:2412–2420 (2013))
  - **Midpoint cell methods** (近・長距離相互作用計算の高並列化)  
(Jung *et al.*, *J. Comput. Chem.*, **35**:1064–1072 (2014))
  - **Parallelization of FFT** (長距離相互作用計算の高並列化)  
(Jung *et al.*, *Comp. Phys. Comm.*, **200**: 57-65 (2016))
  - **GPGPU化** (NEW in version 1.1.0)  
(Jung *et al.*, *J. Chem. Theory Comput.*, **12**: 4947-4958 (2016))
- 他にもトラジェクトリなどの書き出し・読み込みを高速化するため並列I/Oスキームを独自に持つ

# Inverse lookup table法

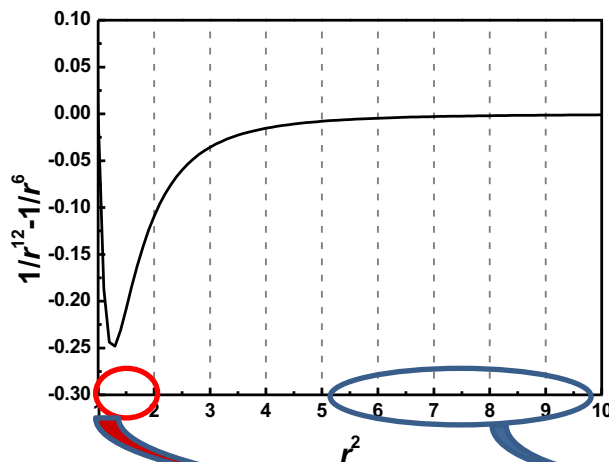
Jung et al., *J. Comput. Chem.*, **34**:2412–2420, (2013)

- 近距離相互作用は距離の関数であるため、カットオフ長までの距離の代表点での相互作用の値を計算、メモリ(table)に記憶しておく。

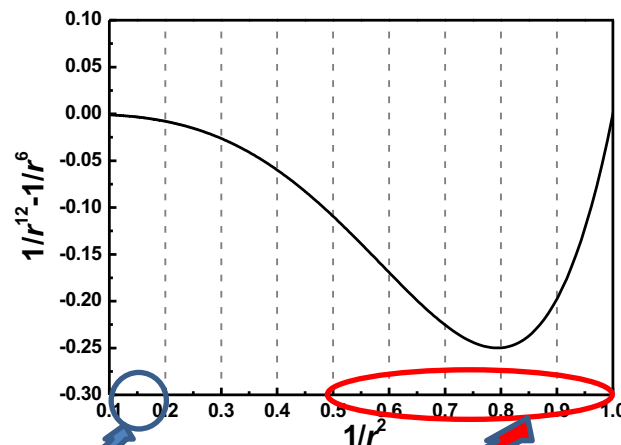
メモリに記録されている相互作用値



従来の方法



GENESISの方法



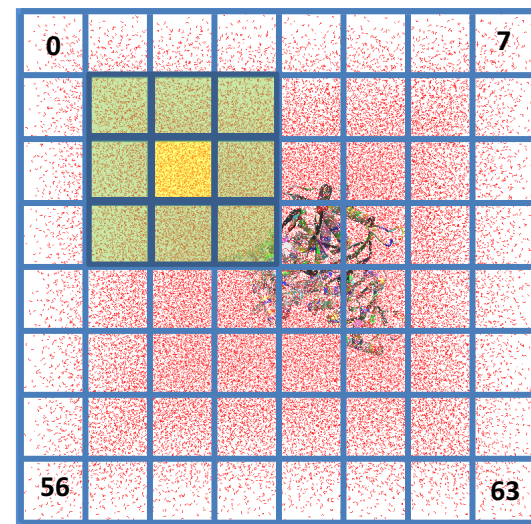
- 従来の方法では $r^2$ の線形関数、3次関数で内挿していた物を、GENESISでは $1/r^2$ の線形関数で内挿し、高速で精度の良い計算を可能とする

# Midpoint cell 法

Jung et al., *J. Comput. Chem.*, **35**:1064–1072, (2014)

## ■ ドメイン分割法 (ほとんどのMDプログラムで採用)

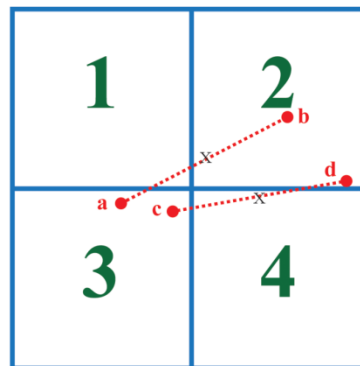
- 系全体をカットオフ長より長い一辺をもつセルで区切る
- エネルギー計算は隣接のboxのみを考えればよい (黄色のセル内の原子の相互作用は黄と緑のセル内の原子のみをカウントすればよい)
- 通信回数が減少されるため、並列度が良くなる
- ドメイン分割法で異なるセル間の相互作用計算をどのCPUコアに割り振るかが並列度に重要な問題になる



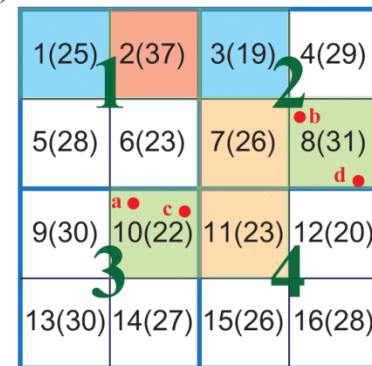
## ■ Midpoint cell法

- 従来のmidpoint法では、2つの原子の間地点のセルを受け持つコアが計算する
- それぞれの原子が存在する二つのセル (8, 10とする) の中間のセル (7, 11のどちらか) を受け持つコアが計算を担当する
- 通信回数が減少されるため、並列度が良くなる

(a) Midpoint法<sup>1</sup>

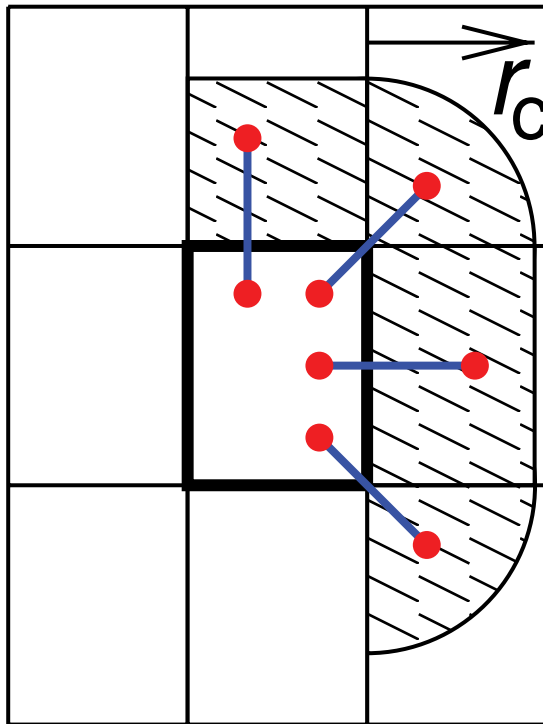


(b) Midpoint cell法

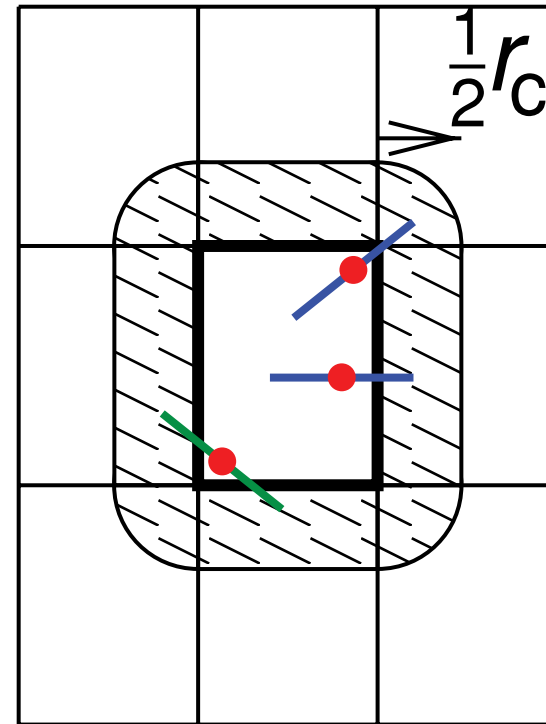


<sup>1</sup> KJ. Bowers et al, JCP 124, 184109 (2006)

# 補足: Midpoint だと通信量が減る理由



Half shell



Midpoint

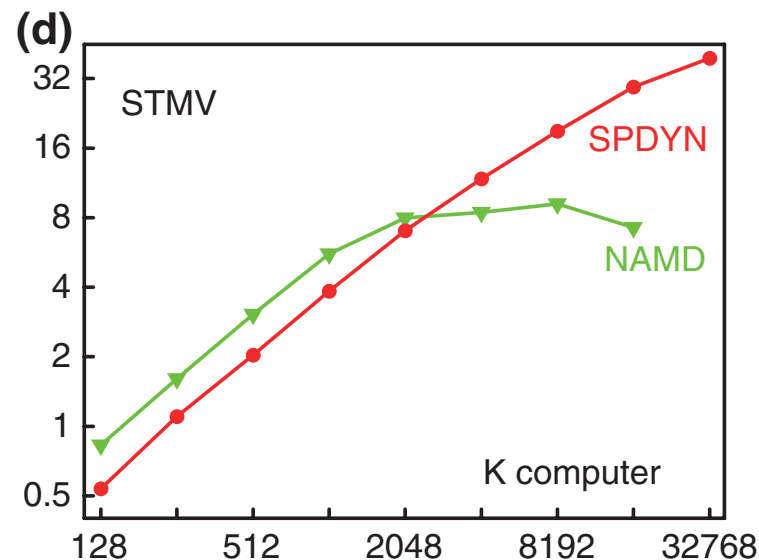
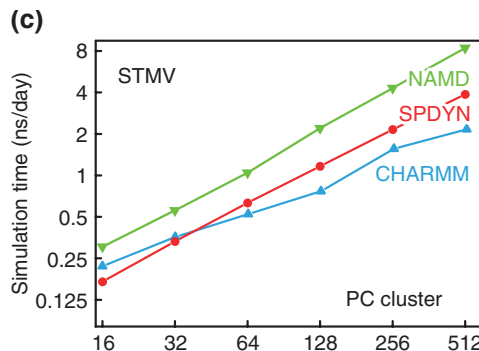
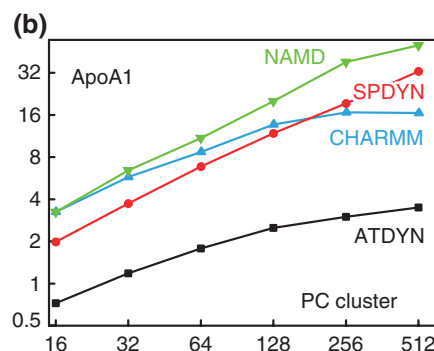
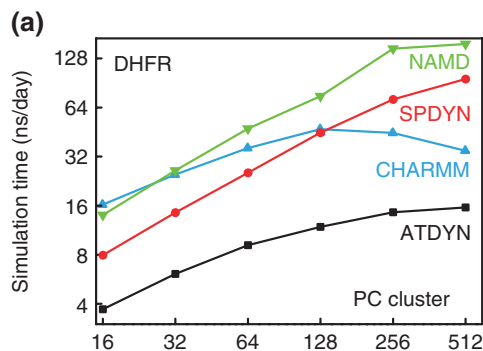
# GENESISのベンチマークデータ

## ■ ベンチマーク詳細

参照: <http://www.aics.riken.jp/labs/cbrt/benchmark/>

J. Jung, T. Mori, C. Kobayashi, Y. Matsunaga, T. Yoda, M. Feig, and Y. Sugita, *WIREs Comput. Mol. Sci.*, 5, 310-323 (2015).

## ■ 例



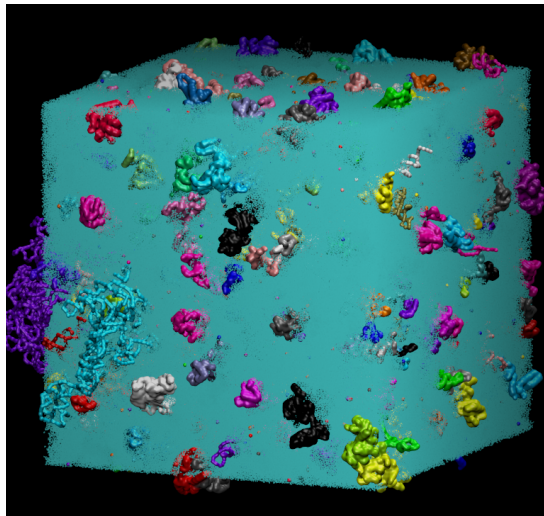
京コンピュータでの計算  
大規模資源での高並列効率を実現

# GENESIS+京コンピュータの成果紹介

I. Yu, T. Mori, T. Ando, R. Harada, J. Jung, Y. Sugita, and M. Feig, *Elife* 5, e19274 (2016).

## Cytoplasm of *Mycoplasma genitalium*

Model for MD simulation



**# of atoms: 11,737,298**

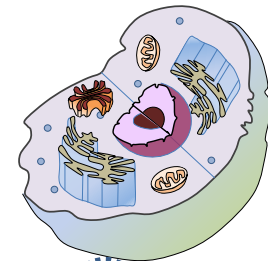
# of macromolecules: 216 (43 types)

# of metabolites: 4,212 (89 types)

**Conc. : 298 mg/ml**

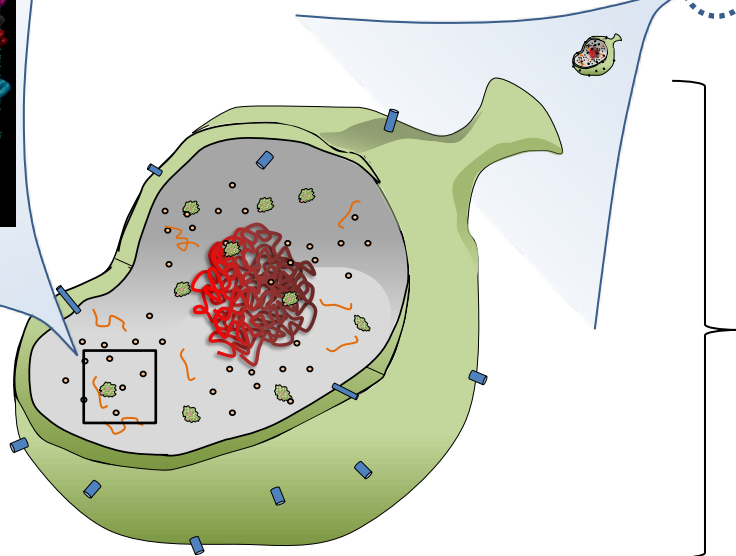
Vol. fraction: 0.315 (Calculated by 3V (Neil etc, 2010))

Animal cell



10  $\mu$ m

*Mycoplasma genitalium*

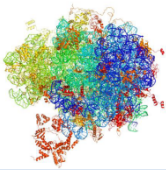
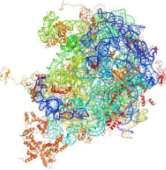
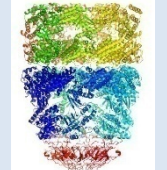


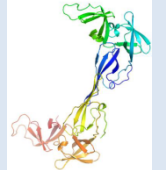


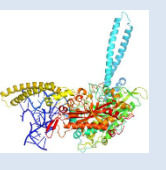
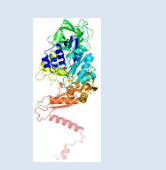
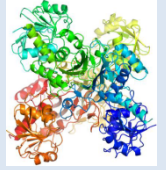
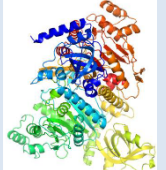

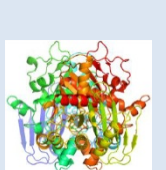



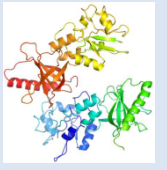
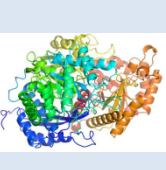
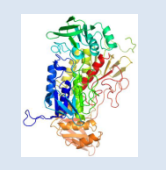
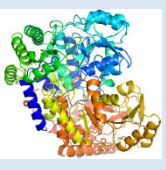



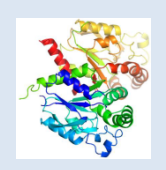


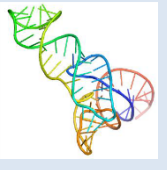














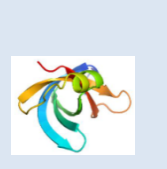


300 nm




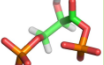


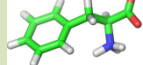
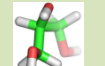
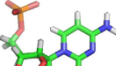
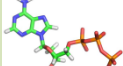

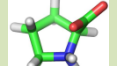
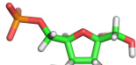
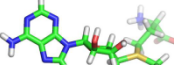
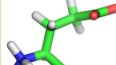
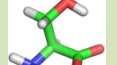


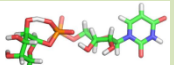
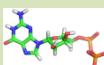
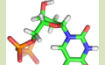
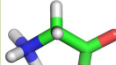
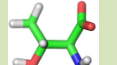


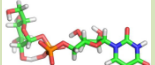
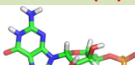
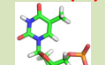
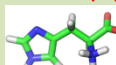


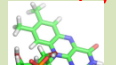
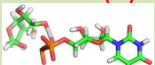
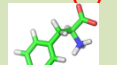
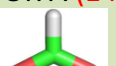
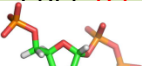
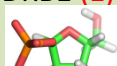
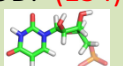

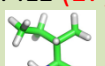
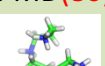

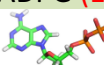
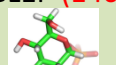
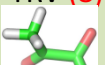
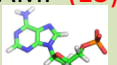
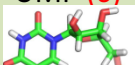
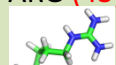



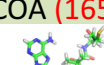


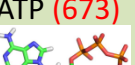
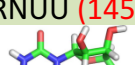
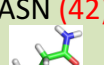


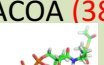


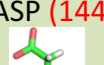


# List of macromolecules in the system

name (number)

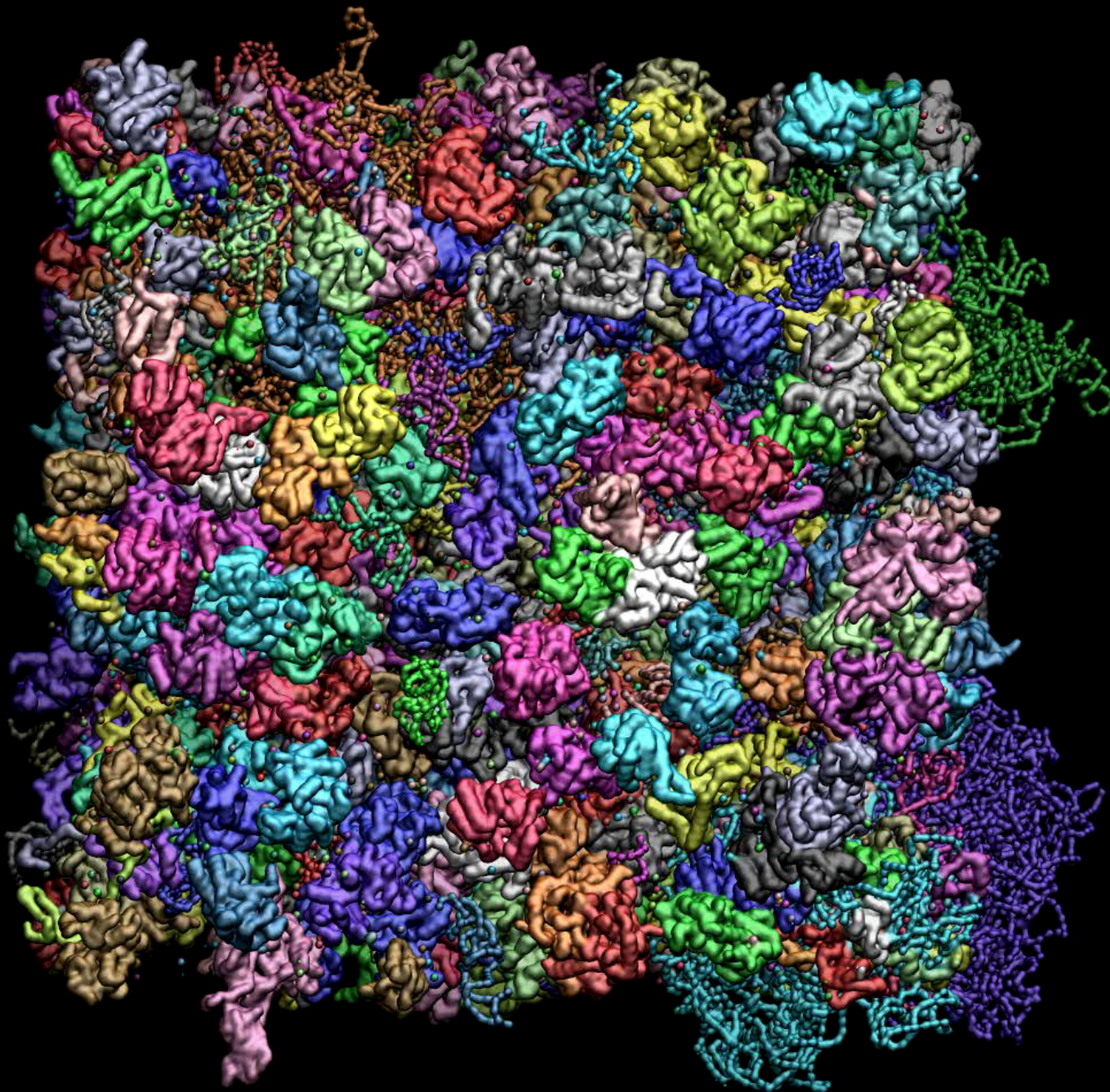
RP (2) 	R50P (1) 	GROL (3) 	PYK (2) 	IF2 (1) 	EFP (2) 	FUSA (3) 	APSO (1) 	SERS (1) 	LEPA (1) 
GAPA (5) 	LYSS (1) 	RR5 (1) 	ACKA(13) 	EFTU (1) 	PTA (4) 	FBA (4) 	RIBF (1) 	ENO (13) 	TRXB (1) 
FOLD (1) 	PGI (5) 	METK (1) 	MANB(3) 	UDK (1) 	NOX (20) 	RF (1) 	TRNA (2) 	ATRN(25) 	PGK (18) 
CDD (1) 	ACK1 (1) 	PDHC(28) 	PDHA(11) 	PDHD(21) 	PDHB(4) 	FBA1 (1) 	YID2 (1) 	TIG (2) 	CMK (2) 
ADK (2) 	MTHF (1) 	1F1 (3) 							

# List of metabolites (代謝物質) in the system

name (number)

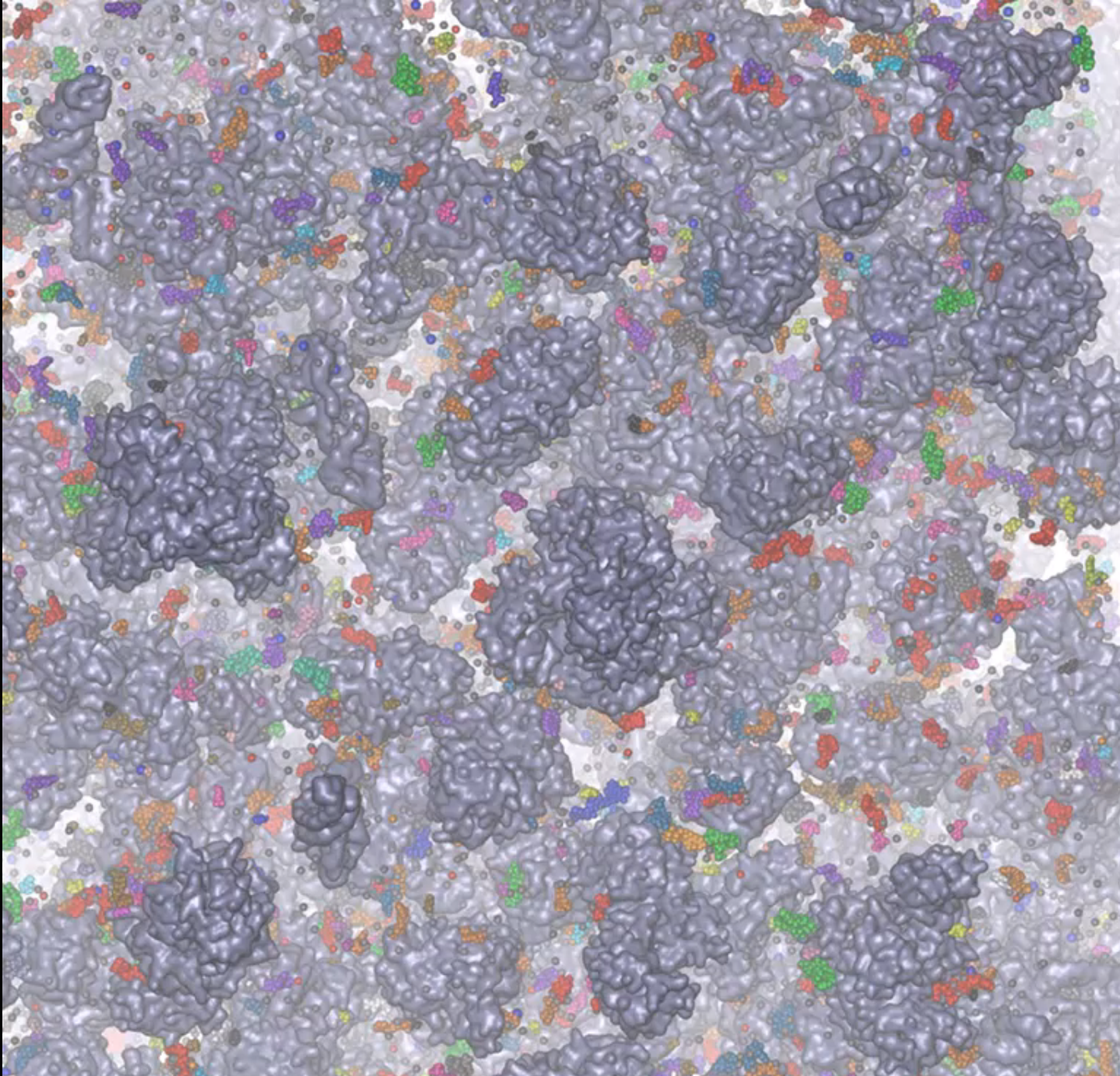
PO42 (161) 	GLBP (29) 	GLAP (3) 	RB5P (1) 	CDP (4) 	DAMP (2) 	PCYS (21) 	PPHE (2) 	water (2,944,143) Na <sup>+</sup> (1,505) Cl <sup>-</sup> (3,76) K <sup>+</sup> (20,792) Mg <sup>2+</sup> (376)
PO41 (194) 	DHAP (7) 	GLCO (7) 	SAHC (29) 	CMP (25) 	DATP 	PGLN (290) 	PPRO (28) 	
POP (30) 	FR6P (8) 	HCYS (25) 	SAMT (13) 	CTP (142) 	DGTP (6) 	PGLU (157) 	PSER (5) 	ADP(17) 
AMM1 (1) 	FAD (15) 	NAD (175) 	UDGL (5) 	GDP (6) 	TDP (28) 	PGLY (60) 	PTHR (13) 	GTP(334) 
H2O2 (1) 	FRBP (28) 	NDPH (6) 	UDPG (34) 	GMP (1) 	TMP (3) 	PHSE (9) 	PTPF (3) 	DCDP (1) 
GL3P (76) 	FMN (4) 	PYRP (7) 	UDGF (7) 	GUN (10) 	TTP (6) 	PHSD (4) 	PTYR (1) 	NUSC (1) 
ACET (63) 	FOR A (14) 	PRPP (7) 	DRB1 (1) 	UDP (134) 	PALA (162) 	PILE (17) 	SPMD(80) 	DCTP (1) 
ADPG (1) 	GL1P (140) 	PYRV (8) 	AMP (13) 	UMP (6) 	PARG (43) 	PLEU (18) 	ETOH(116) 	DGDP (1) 
COA (165) 	GOLP (16) 	RB1P (6) 	ATP (673) 	RNUU (145) 	PASN (42) 	PLYS (23) 	ACTP(73) 	DGMP(3) 
ACOA (38) 	GL6P (7) 	DRB5 (2) 	CYN (2) 	UTP (612) 	PASP (144) 	PMET (17) 	RBFL(4) 	PVAL(162) 





100 ns MD  
performed by  
GENESIS on K  
computer





# GENESISの内容物(1)

## ■ MDプログラム

### – ATDYN (ATomic decomposition DYNamics simulator):

- atomic decompositionを使用
- 粗視化モデル(C $\alpha$ -GO, all atom GO)が計算可能
- わかり易いコードでユーザによる開発可能

### – SPDYN (SPatial decomposition DYNamics simulator):

- domain decompositionを使用
- 超並列・高速なルーチン (Midpoint cell/3次元分割FFT/並列 I/O)



実習ではこちらを使用

特徴	ATDYN	SPDYN
システムの分割法	原子分割	ドメイン分割
New lookup table	○	○
レプリカ交換法	○	○
粗視化モデル	○	×
3次元分割FFT	×	○
並列I/O	×	○
r-RESPA (1.1.0)	×	○

# GENESISの内容物(2)

## ■ ATDYN/SPDYN共通

- 最適化
  - ✧ Steepest Decent法
- Integrator
  - ✧ Leapfrog
  - ✧ Velocity Verlet
- アンサンブル
  - ✧ NVE
  - ✧ NVT
    - Langevin
    - Bussi
    - Berendsen
  - ✧ NPT
    - Langevin Piston
    - Bussi
    - (Isotropy of Simulation box: Isotropic, Semi-iso, An-iso,

XY-fixed)

- 拘束計算(constraint)
  - ✧ SHAKE (Leapfrog)
  - ✧ RATTLE (Velocity Verlet)
  - ✧ SETTLE
- FFT (PME)
  - ✧ FFTE
- Restraint functions
  - ✧ Position
  - ✧ Bond
  - ✧ Angle
  - ✧ Dihedral angle
  - ✧ RMSD
- REMD/**REUS**
- String method

赤字は午後の実習で使うもの

# GENESISの内容物(3)

## ■ その他解析ツール

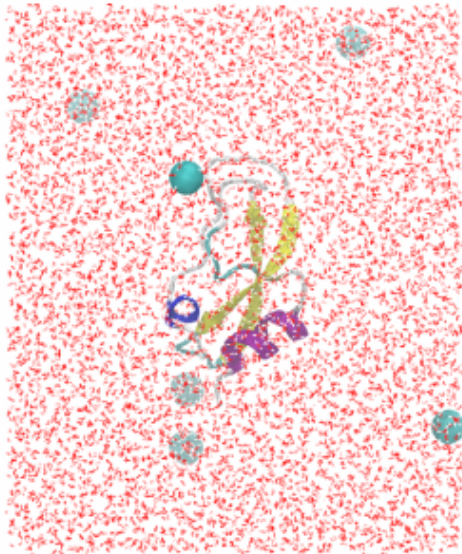
- トラジェクトリ変換(trjcnv)
  - ✧ crd\_convert
  - ✧ pcrd\_convert(並列I/O用計算用)
  - ✧ **remd\_convert**(REMD計算用)
- トラジェクトリ解析(trjana)
  - ✧ mbar\_analysis
  - ✧ pathcv\_analysis
  - ✧ pmf\_analysis
  - ✧ gval\_analysis
  - ✧ rmsd\_analysis(RMSD、RMSF)
  - ✧ trj\_analysis(距離、角度など)
  - ✧ **wham\_analysis**
- リスタートファイル変換(rstcnv)
  - ✧ rst\_convert
  - ✧ rst\_upgrade
- PCA解析(pcaana)
  - ✧ avecrd\_analysis
  - ✧ eigmat\_analysis
  - ✧ flccrd\_analysis
  - ✧ prjcrd\_analysis
- 反応経路探索関連(rpath\_generator)
- 解析共通(libana)



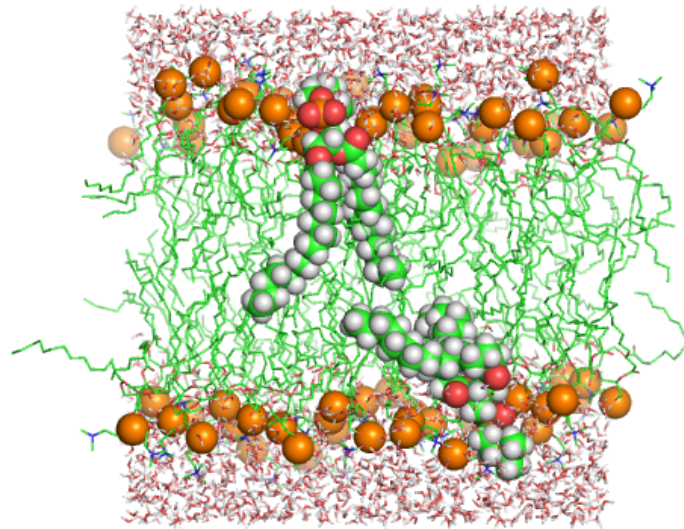
# これらを使ってできること (例1)

## 通常のMDシミュレーション

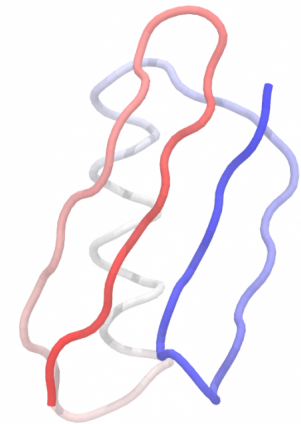
水溶性タンパク質



膜タンパク質



粗視化モデル



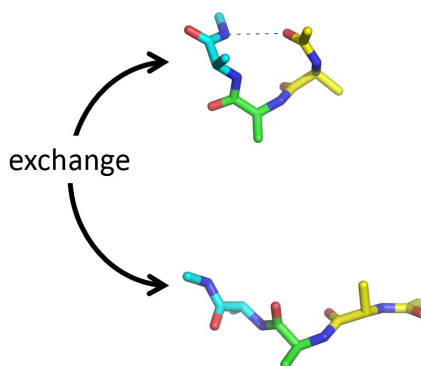
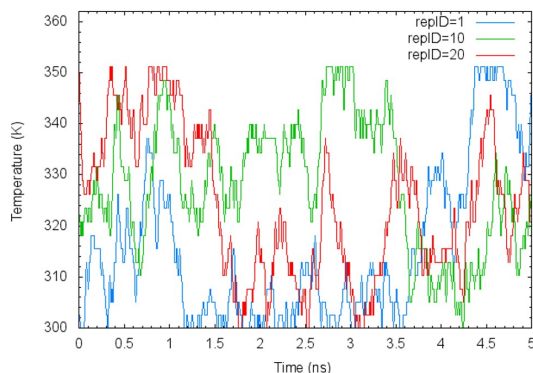
ホームページにチュートリアルがある

[http://www.aics.riken.jp/labs/cbrt/tutorial/basic\\_md\\_tutorials/](http://www.aics.riken.jp/labs/cbrt/tutorial/basic_md_tutorials/)

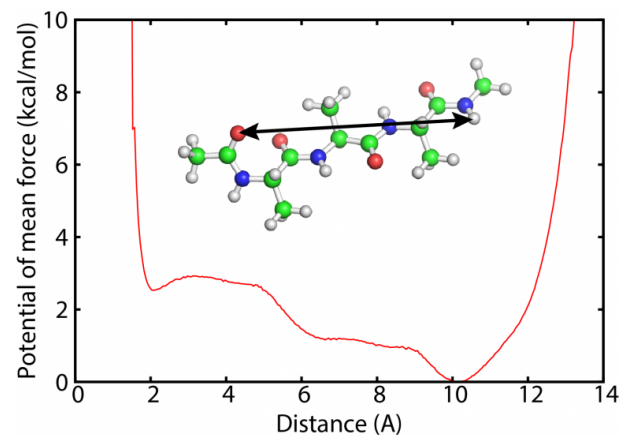
# これらを使ってできること (例2)

## レプリカ交換法による効率的なサンプリングと得られたデータ解析

温度レプリカ交換法



ポテンシャル(アンブレラ)  
レプリカ交換法

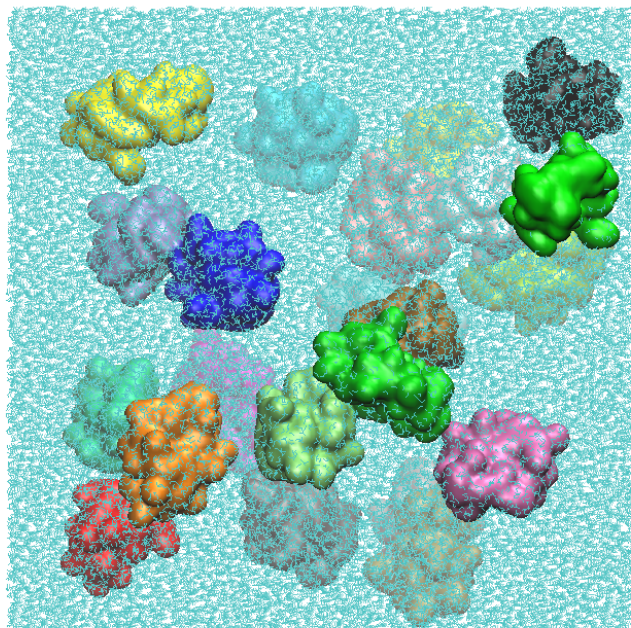


ホームページにチュートリアルがある

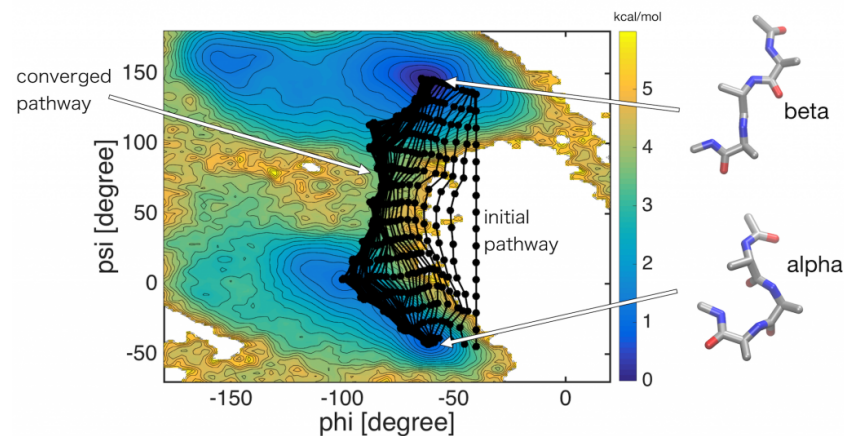
<http://www.aics.riken.jp/labs/cbrt/tutorial/remd-tutorials/>

# これらを使ってできること (例3)

並列I/Oによる巨大なシステムのMD



構造変化のパスウェイ探索



ホームページにチュートリアルがある

[http://www.aics.riken.jp/labs/cbrt/tutorial/advanced\\_md\\_tutorials/](http://www.aics.riken.jp/labs/cbrt/tutorial/advanced_md_tutorials/)



# 今日の内容

## ■ 10:00 – 11:30 講義

- 生体分子の分子動力学計算
- GENESISの紹介
- 実習に必要な理論の説明

## ■ お昼休憩(90分)

## ■ 13:00 – 16:00 実習

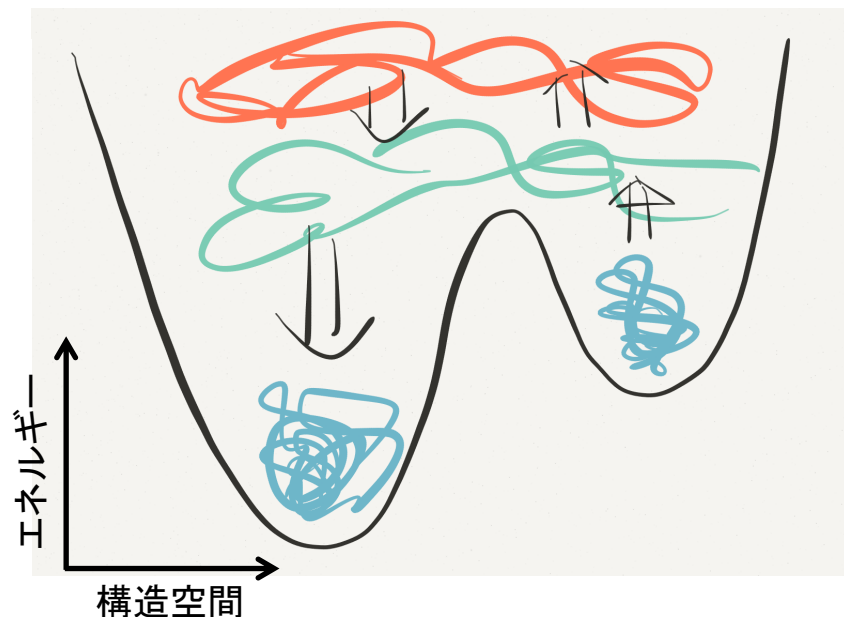
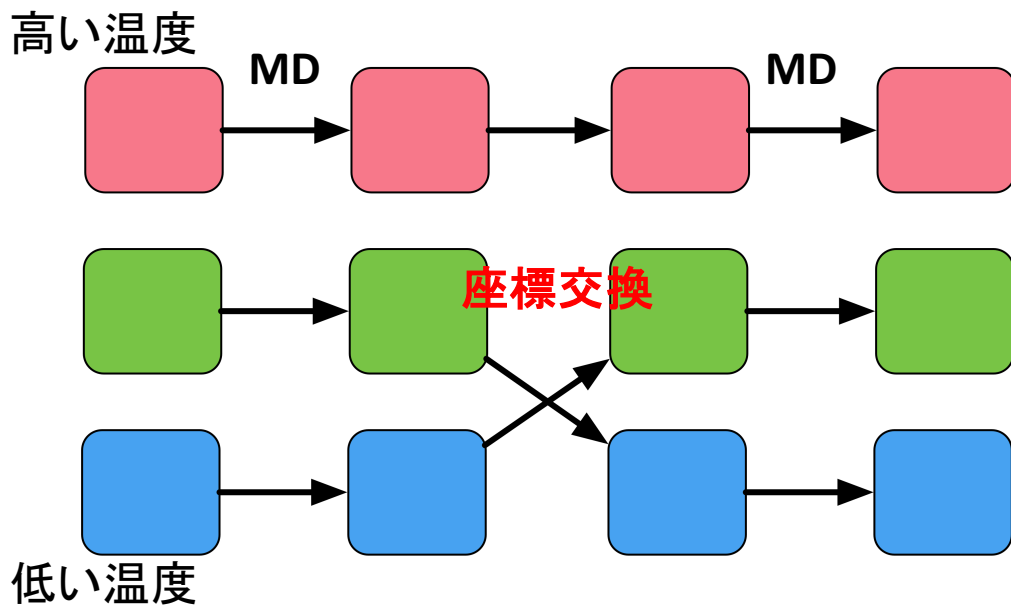
- GENESISのコンパイル
- 実行例(1) 分子動力学法計算
- GENESISのトラブルシューティング
- 実行例(2) レプリカ交換分子動力学法計算と自由エネルギー計算

# 効率的なサンプリング法： レプリカ交換法

# レプリカ交換法の概要

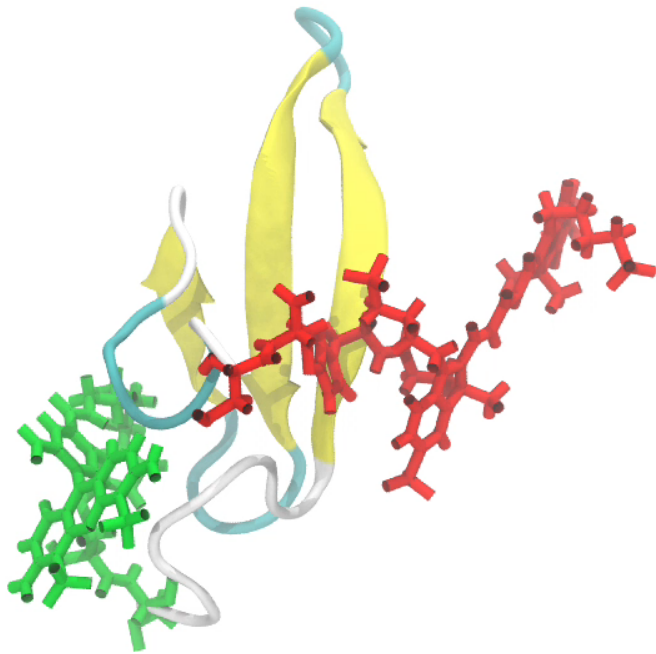
レプリカ交換法とは、異なる温度・ポテンシャルの**レプリカ**を並列に分子動力学シミュレーションし、一定間隔でレプリカ間の座標を一定のルールで交換させる手法。自由エネルギー上のローカルミニマムを脱出し、次の安定状態を探索するのに威力を発揮する。

GENESISのメイン機能のひとつ・並列計算機を有効に使用できる

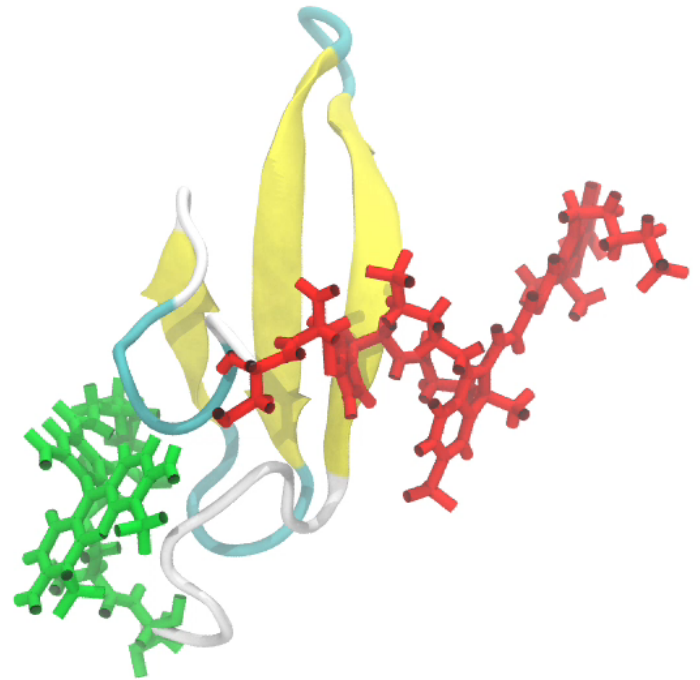


# レプリカ交換法の適用例：タンパク質にラベルした蛍光色素分子の動き

普通のMDシミュレーション



レプリカ交換法



レプリカ交換法により蛍光色素分子の多くの状態をサンプリングすることができる

# 理論の説明: 準備

システムのポテンシャルエネルギーを  $V(x)$  とする。  
ここで  $x$  は全原子のデカルト座標。

$A(x)$  を座標  $x$  に依存する物理量とすると、シミュレーションでは  
 $A(x)$  の期待値を求めたいことが多い。

$$\langle A(x) \rangle = \int A(x) P(x) dx = \frac{\int A(x) e^{-\beta V(x)} dx}{\int e^{-\beta V(x)} dx} = \frac{\int A(x) e^{-\beta V(x)} dx}{Z}$$

$$\beta = \frac{1}{k_B T}$$



# MDとMCMC

## 分子動力学法(Molecular Dynamics; MD)

分子システムの力場(ポテンシャルエネルギー)に基づいて、ニュートン方程式を解いてサンプルを生成する手法

サンプル生成法 = ニュートン方程式 + 熱浴

## マルコフ連鎖モンテカルロ法(Markov Chain Monte Carlo; MCMC)

定常分布(平衡分布)を満たすマルコフ連鎖によりサンプルを生成する方法

サンプル生成法 = 候補構造の生成 + メトロポリス判定

両者とも、エルゴード性が成り立つならば

$$\langle A(x) \rangle \approx \frac{1}{T} \sum_{t=1}^T A(x_t)$$

# MCMCの解説

**マルコフ連鎖**とは、次の状態で座標  $x'$  をとる確率が、直前の状態  $x$  のみに依存するルールで動くシステムのこと

遷移確率  $\pi(x \rightarrow x')$  で記述できる

定常分布(平衡分布)をサンプルするマルコフ連鎖を設計できるか？  
遷移確率が満たすべき性質・要請はあるか？

# MCMCの解説

答え: **詳細釣り合い**を満たせば良い

詳細釣り合い  $P(x)\pi(x \rightarrow x') = P(x')\pi(x' \rightarrow x)$

$$\frac{\pi(x' \rightarrow x)}{\pi(x \rightarrow x')} = \frac{P(x)}{P(x')} \quad \leftarrow \text{Zがキャンセルされる!}$$
$$P(x) = \int e^{-\beta V(x)} dx / Z$$

詳細釣り合いを満たす遷移確率設計の代表手法: **メトロポリス法**

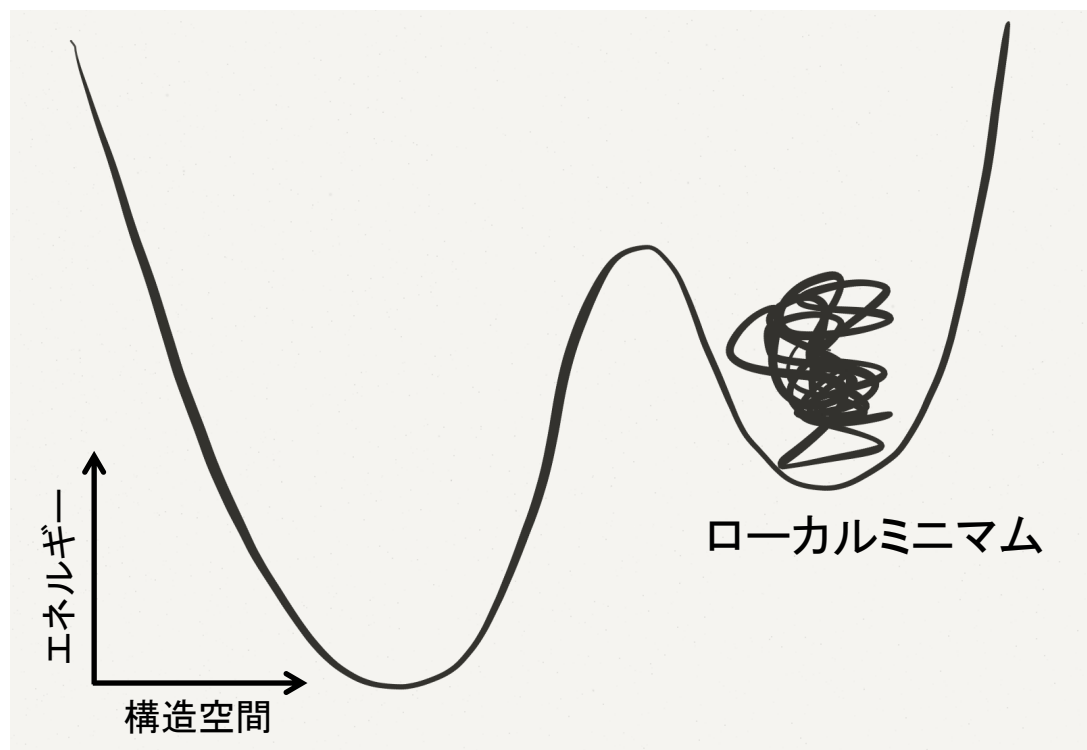
1. 対称性 $Q(x, x') = Q(x', x)$ を満たすような確率で次の構造候補を作成
2.  $r = P(x')/P(x) > 1$ ならば 必ず候補を受理  
 $r = P(x')/P(x) < 1$ ならば 確率 $r$ で候補を受理

これが詳細釣り合いを満たすことは容易に確認できる

# 通常のMDのサンプリングにおける問題点

自由エネルギー地形が凸凹していると、ローカルミニマムにトラップされることで緩和時間が長くなり、シミュレーション時間の範囲内でエルゴード性が成り立たない

$$\langle A(x) \rangle \neq \frac{1}{T} \sum_{t=1}^T A(x_t)$$



# 温度レプリカ交換法：温度の異なるレプリカの導入

温度の異なる分子システムのレプリカ  $M$  個をまとめた同時分布を導入する。

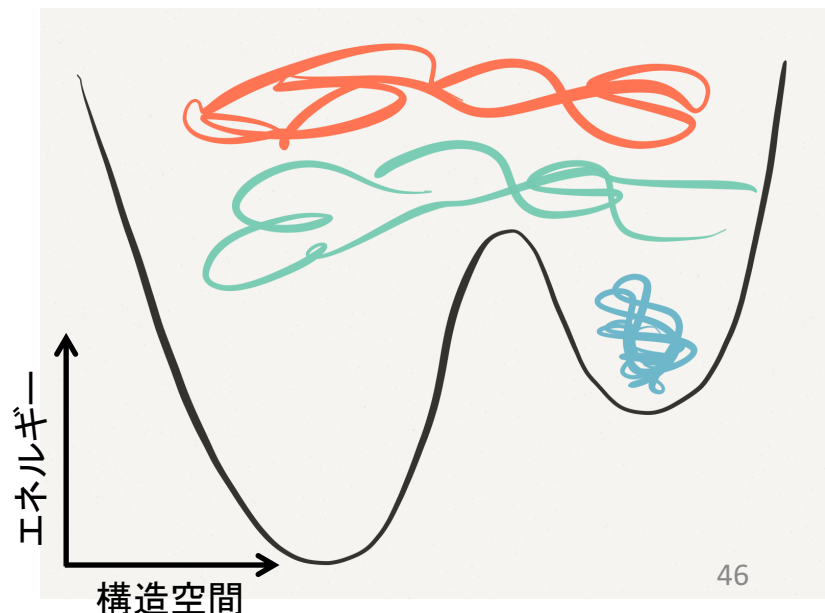
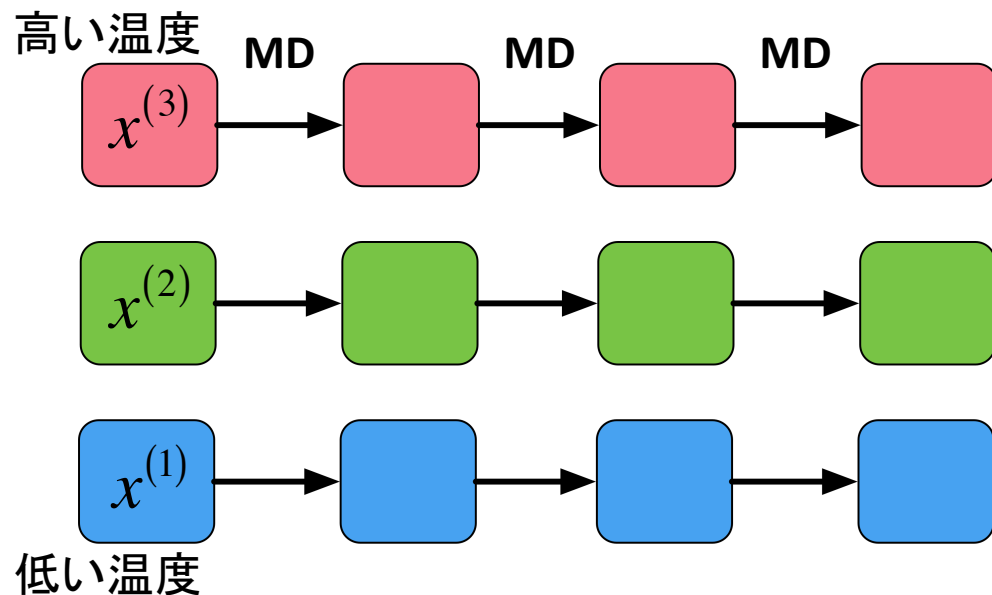
各レプリカ間には相互作用はないので、同時分布の定常分布(平衡分布)は単なる掛け算

同時平衡分布 
$$P\left(x^{(1)}, x^{(2)}, \dots, x^{(M)}\right) = \prod_{m=1}^M P_m\left(x^{(m)}\right) = \prod_{m=1}^M \frac{e^{-\beta^{(m)} V\left(x^{(m)}\right)}}{Z^{(m)}}$$

個々のレプリカについて分子動力学法(MD)を適用してサンプリングすると

→ 同時定常(平衡)分布についてサンプリングしていることになる

→ 低い温度のトラジェクトリは依然としてトラップされる可能性がある





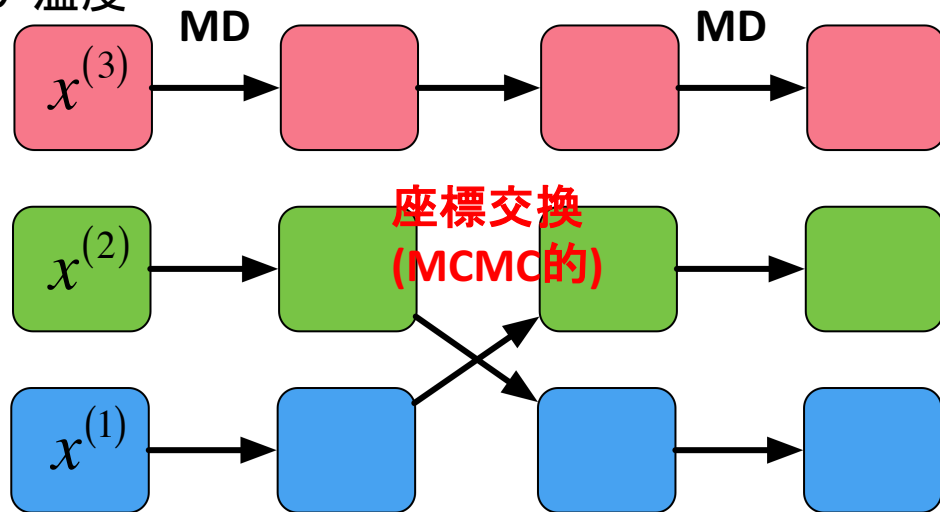
# レプリカ交換をMCMCとして導入

MDの途中で、MCMC的な時間発展ルールを導入する。ただし、候補構造を新たに生成するのではなく、レプリカ $m$ と $m+1$ の座標を交換することを考える。このとき、メトロポリス法の交換確率 $r$ は、以下のように書ける

$$r = \frac{P(x^{(1)}, \dots, x^{(m+1)}, x^{(m)}, \dots, x^{(M)})}{P(x^{(1)}, \dots, x^{(m)}, x^{(m+1)}, \dots, x^{(M)})} = \frac{P_m(x^{(m+1)})P_{m+1}(x^{(m)})}{P_m(x^{(m)})P_{m+1}(x^{(m+1)})} = \exp\left[(\beta^{(m+1)} - \beta^{(m)})(V(x^{(m+1)}) - V(x^{(m)}))\right]$$

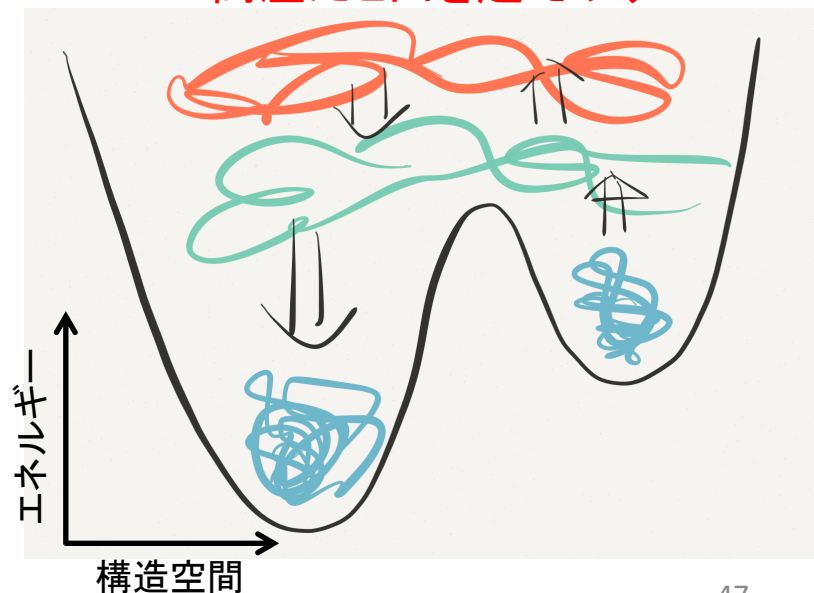
拡張アンサンブルの詳細釣り合いは満たしているので、平衡分布のサンプリングになっている。個々のレプリカの履歴に着目すると、一旦高温に上って、低温におりる可能性がある。つまりこの場合、トラップを短時間で脱出できる。

高い温度



低い温度

高温だと山を越えやすい！



# 温度レプリカ交換法の利点と欠点

- 利点

- 汎用的である: システムについての事前知識なしに使える

- 欠点

- システムサイズ(原子数)が大きくなると、交換頻度が下がるため、必要とされるレプリカ数が大きくなる

解決法: ある自由度に絞って拡張アンサンブルを考える  
→ポテンシャルレプリカ交換法(REUS)

# ポテンシャルレプリカ交換法(REUS): 異なるレプリカの導入

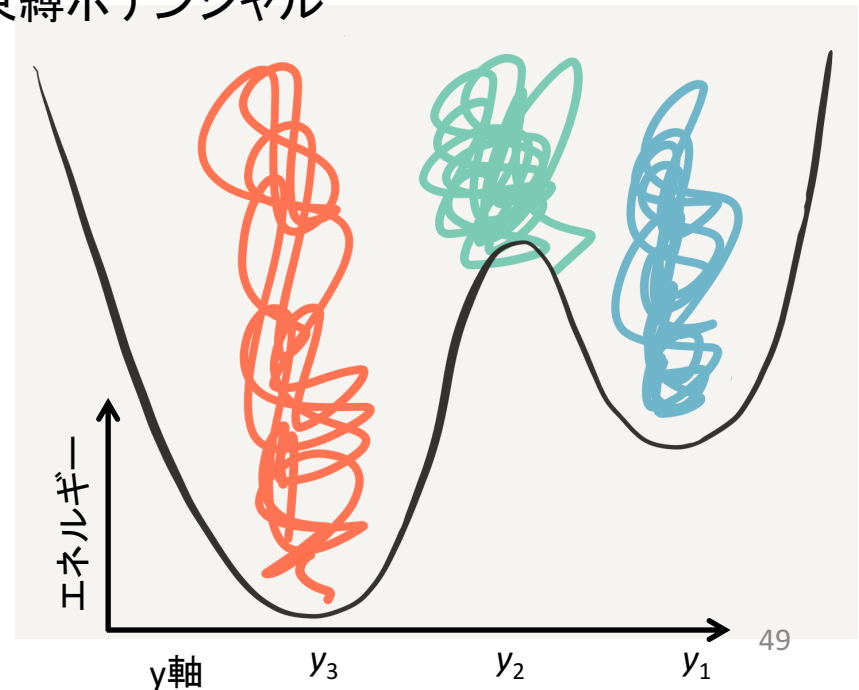
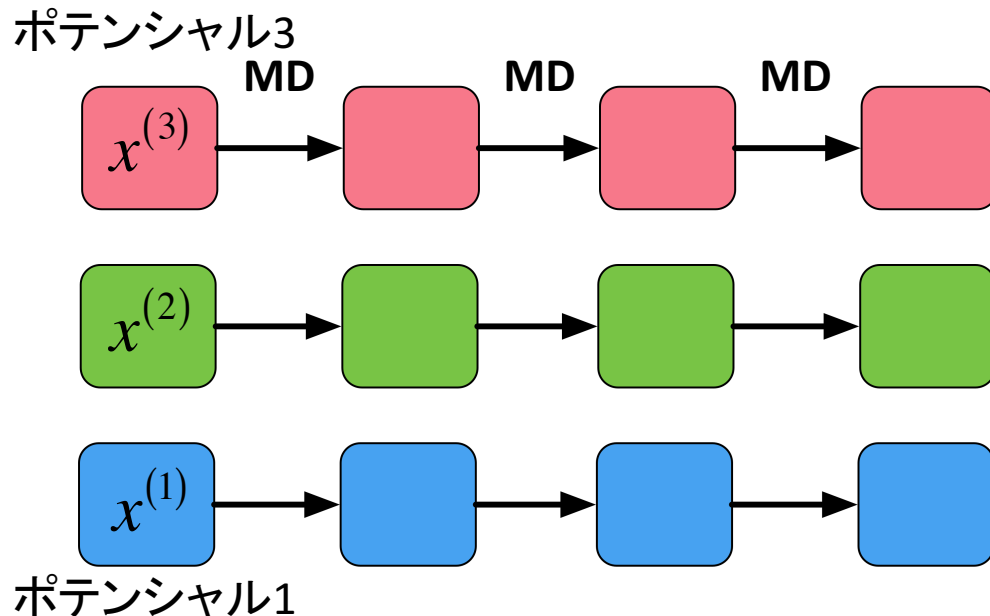
ある1次元座標 $y$ 上に自由エネルギーバリアがあると分かっている場合  
温度ではなく、異なる $y$ 軸上の束縛ポテンシャルをもつレプリカを導入する。

具体的には、レプリカ $m$ に対して、 $y_m$ を中心とする調和ポテンシャルを想定する:

$$V_m(x) = \underbrace{V(x)}_{\text{システムのポテンシャル}} + \underbrace{\frac{1}{2}(y(x)-y_m)^2}_{\text{束縛ポテンシャル}} = V(x) + U_m(y(x))$$

システムのポテンシャル

束縛ポテンシャル



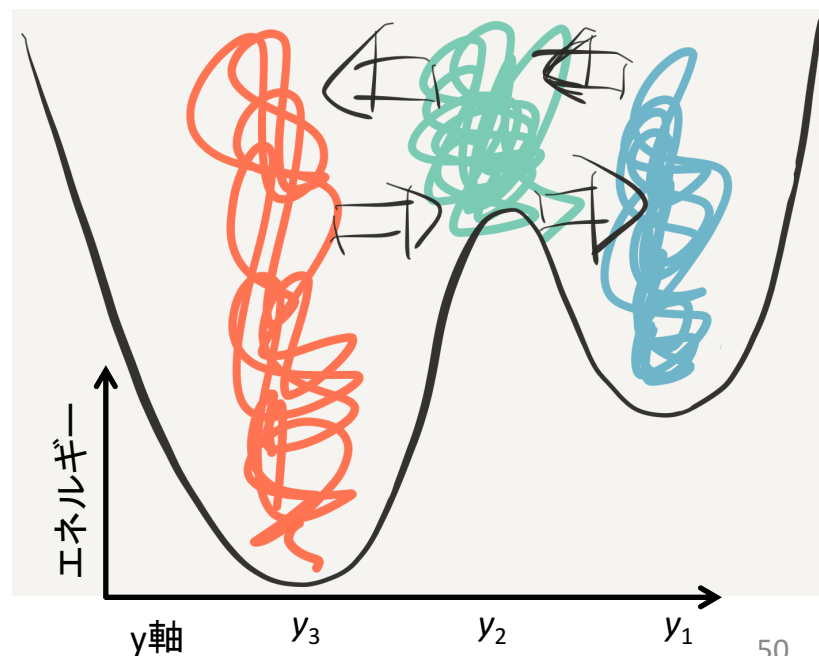
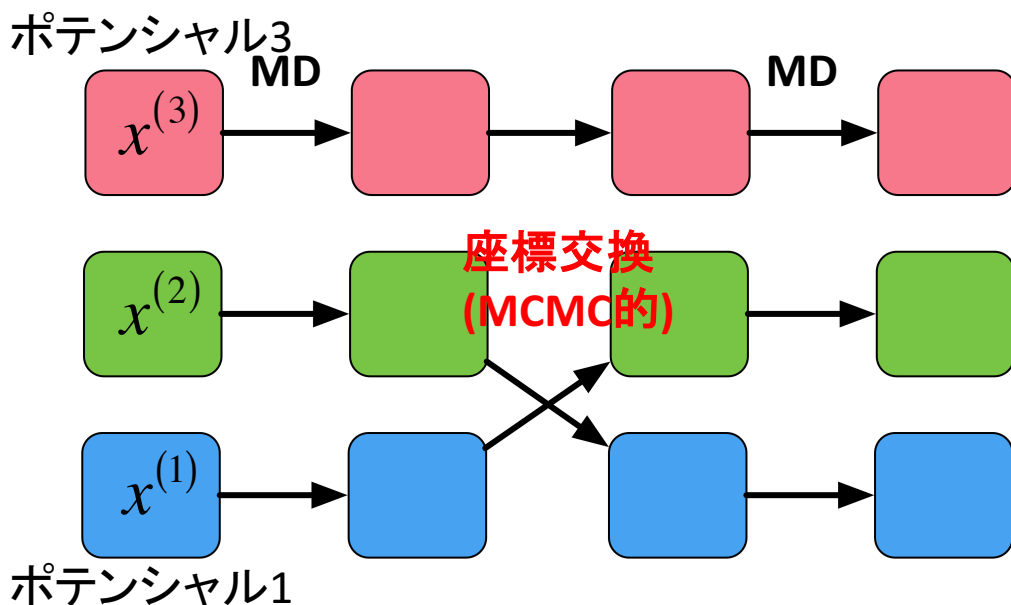
# レプリカ交換をMCMCとして導入

前と同様にして、レプリカmとm+1の座標を交換することを考える。

$$r = \frac{P(x^{(1)}, \dots, x^{(m+1)}, x^{(m)}, \dots, x^{(M)})}{P(x^{(1)}, \dots, x^{(m)}, x^{(m+1)}, \dots, x^{(M)})} = \frac{P_m(x^{(m+1)})P_{m+1}(x^{(m)})}{P_m(x^{(m)})P_{m+1}(x^{(m+1)})} = \exp\left[\beta\left(-U_m(y^{m+1}) - U_{m+1}(y^m) + U_m(y^m) + U_{m+1}(y^{m+1})\right)\right]$$

$r$ は $y$ のみに依るので、システムサイズが大きくても交換頻度は下らない

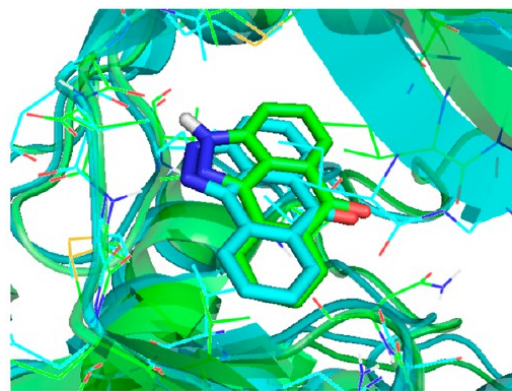
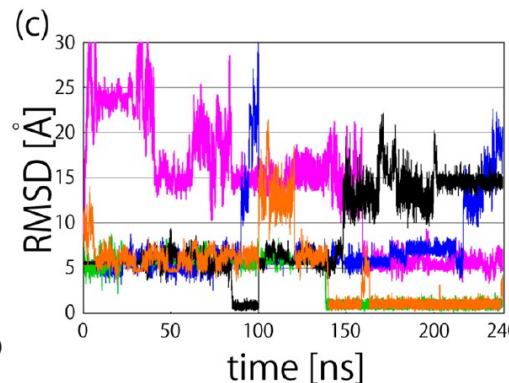
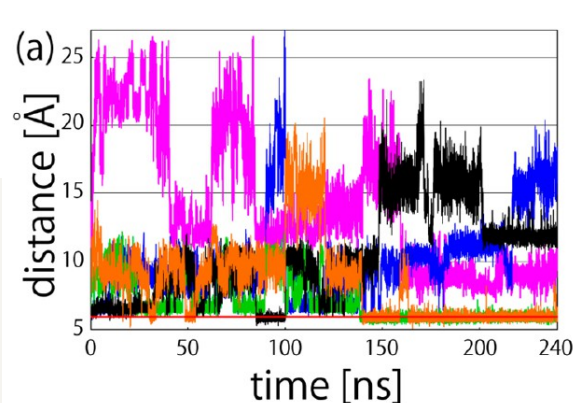
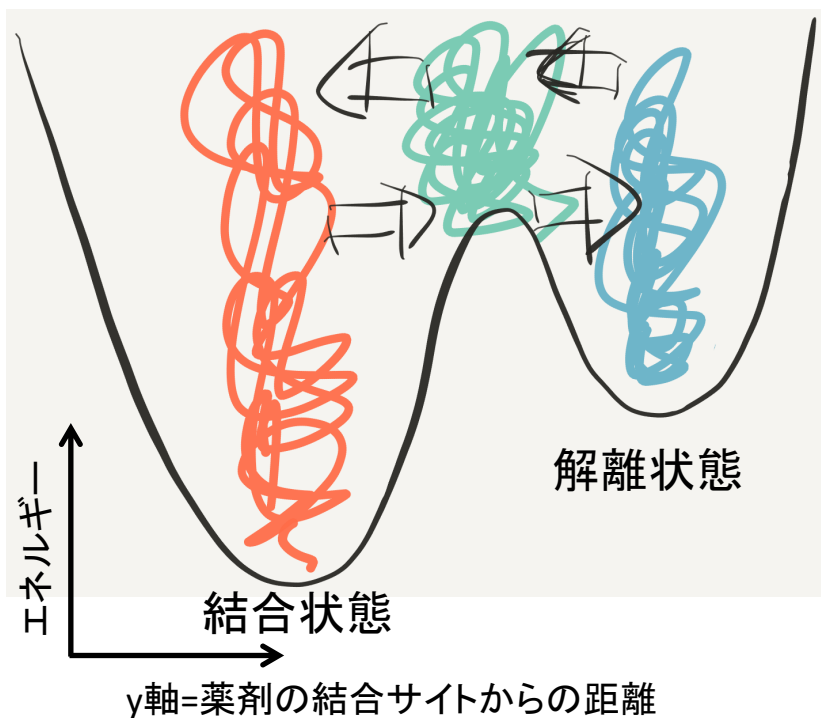
束縛ポテンシャルで山を越える！



# ポテンシャルレプリカ交換(REUS)が役に立つ場面: 薬剤の結合モードの探索

y軸として「薬剤の結合サイトからの距離」ととると  
結合モードの探索に有効であることが示されている

結合・解離を何度も繰り返して結合モードを探索！

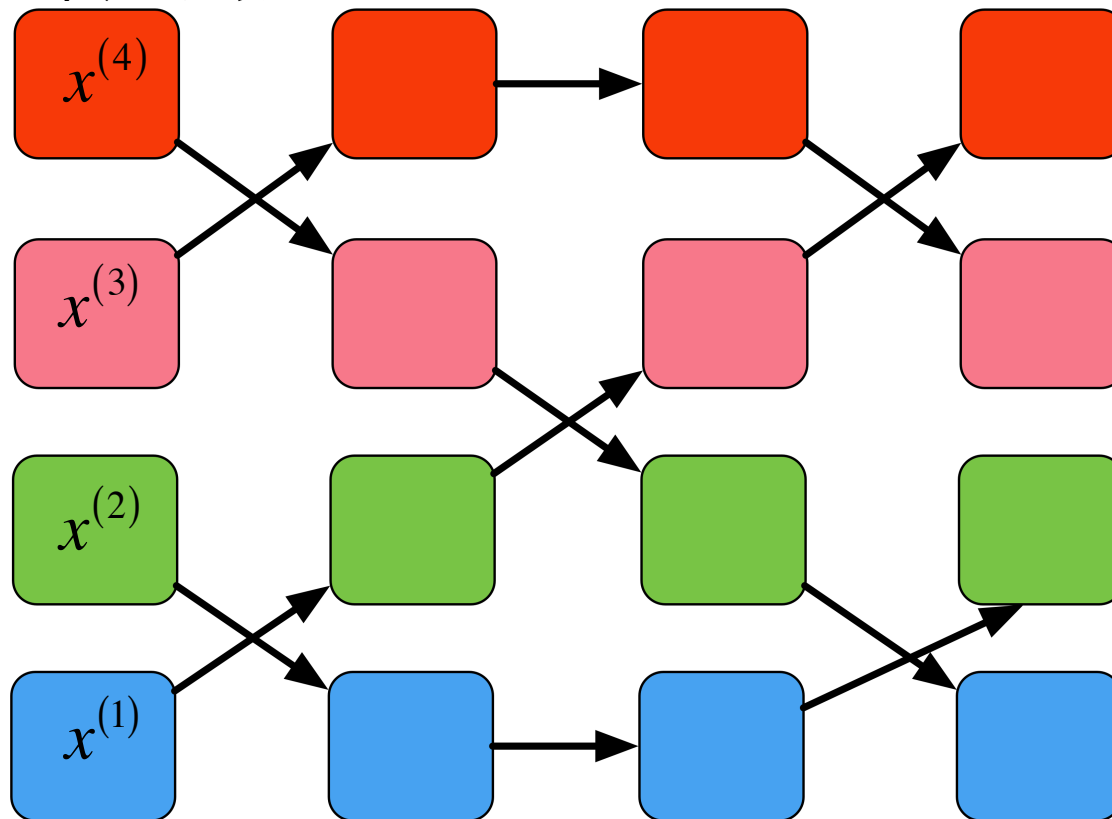


H. Kokubo, T. Tanaka, and Y. Okamoto, *J. Chem. Theory Comput.* **9**, 4660 (2013).  
H. Kokubo, T. Tanaka, and Y. Okamoto, *J. Comput. Chem.* **32**, 2810 (2011).



# GENESISでのレプリカ交換法の実装内容

高い温度 or ポテンシャル4



低い温度 or ポテンシャル1

1.  $(m, m+1)$  と  $(m-1, m)$  を入れ違いにして交換を試みる
2. 実際は、座標ではなく温度/ポテンシャルを交換している  
トラジェクトリの温度/ポテンシャルが交換ごとに変化  
ユーザがソートして、同一温度/ポテンシャルのデータにする必要

# 細かな話: 温度レプリカ交換法における交換時の速度の取り扱い

細かいことを言えば、MDと組み合わせる場合、交換時に座標だけでなく、速度をどのように取り扱うかも考えなければならない。

詳細釣り合いの式に、ポテンシャルエネルギーだけでなく、速度を含めたハミルトニアンに書くこともできるが、式が煩雑になるとともに、座標+速度交換直後のシステムの安定性等に疑義があった。

そこで、Sugita and Okamoto (1999)は交換の際に、速度を

$$\sqrt{\frac{\beta^{(m)}}{\beta^{(m+1)}}}$$

Y. Sugita and Y. Okamoto, *Chem. Phys. Lett.* **314**, 141 (1999)

またはその逆のファクターでスケールリングすることを提案した。このようにすれば、速度はキャンセルして、メトロポリスの交換確率  $r$  は前ページのようにシンプルになる。

速度の取り扱いに関しては、この他に交換時にマクスウェル・ボルツマン分布からサンプリングする手法等が提案されている。  
GENESISでは、スケールリングする手法が実装されている。

# 細かな話: 温度レプリカ交換法における 温度割り当て・レプリカ数を如何に決めるか?

## 経験的な事実

1. レプリカ  $m$  と  $m+1$  が交換する確率は  $m$  に寄らず一定となる温度割り当てが良い。→ 温度空間をランダム・ウォークするようになるため。
2. 交換確率は0.2~0.3付近が良い (経験則)

## 温度割り当ての見積り方

事前に異なる温度で交換なしのシミュレーションを流して、  
各温度でのポテンシャルエネルギーの分布を得る。

以下の式を使って交換確率を見積もる

$$P_{exchange}(\beta_m, \beta_{m+1}) = \int_{\beta_m} \int_{\beta_{m+1}} P_m(V^{(m)}) P_{m+1}(V^{(m+1)}) \min[1, r] dV^{(m)} dV^{(m+1)}$$

平均値+補完による近似 K. Y. Sanbonmatsu and A. E. García, *Proteins* **46**, 225 (2002).

ガウシアンによる近似 N. Rathore, et al., *J Chem Phys* **122**, 024111 (2005).

上限温度は、システムが壊れない かつ シミュレーションアルゴリズムが破綻しない温度

タンパク質フォールディング  $T_{\max} = 600\text{K}$  付近

膜タンパク質シミュレーション  $T_{\max} = 400\text{K}$  付近

交換確率・上限温度を基にして、レプリカ数を決定する

お手軽なのは以下のサーバ <http://folding.bmc.uu.se/remd/>

# 細かな話: レプリカ交換法におけるその他の話題

- 温度一定アンサンブル以外でのREMD
  - NPT条件 (GENESISに実装済)
  - NP $\gamma$ T条件  $\gamma$ T REMD (GENESISに実装済、膜タンパク質に有効)  
T. Mori, J. Jung, and Y. Sugita, *JCTC* **9**, 5629 (2013).
- REMDのトラジェクトリの統計処理
  - WHAM, Parallel tempering WHAM (PTWHAM)による複数トラジェクトリを使った処理  
J. D. Chodera, et al., *JCTC* **3**, 26 (2007).
  - Markov State Modelによるlong time-scale kinetics情報の推定  
N.-V. Buchete and G. Hummer, *Phys. Rev. E* **77**, 030902 (2008).

# レプリカ交換法の参考文献

- MCMC・レプリカ交換法の理論解説  
伊庭幸人, 種村正美 他, 計算統計 2 マルコフ連鎖モンテカルロ法とその周辺 (統計科学のフロンティア 12) 岩波書店 (2005)
- レプリカ交換法の実論文  
K. Hukushima and K. Nemoto, *J. Phys. Soc. Jpn.* **65**, 1604 (1996).
- 温度レプリカ交換法と分子動力学法を組み合わせた論文  
Y. Sugita and Y. Okamoto, *Chem. Phys. Lett.* **314**, 141 (1999)



# 自由エネルギーの統計推定： WHAM

# ポテンシャルレプリカ交換法のデータ解析

- 問題

- 束縛ポテンシャルがかかった系のシミュレーションデータから、どのようにして束縛ポテンシャルなしの系のy軸に沿った自由エネルギー変化をもとめるか？
- 異なる束縛ポテンシャルがかかった系のデータをどのようにして統合するか？

解決法: 重み付け(尤度定式化)して統計推定(最尤推定)  
→ Weighted Histogram Analysis Method (WHAM)

# 最尤推定

最尤推定とはモデルパラメータ推定法のひとつ。

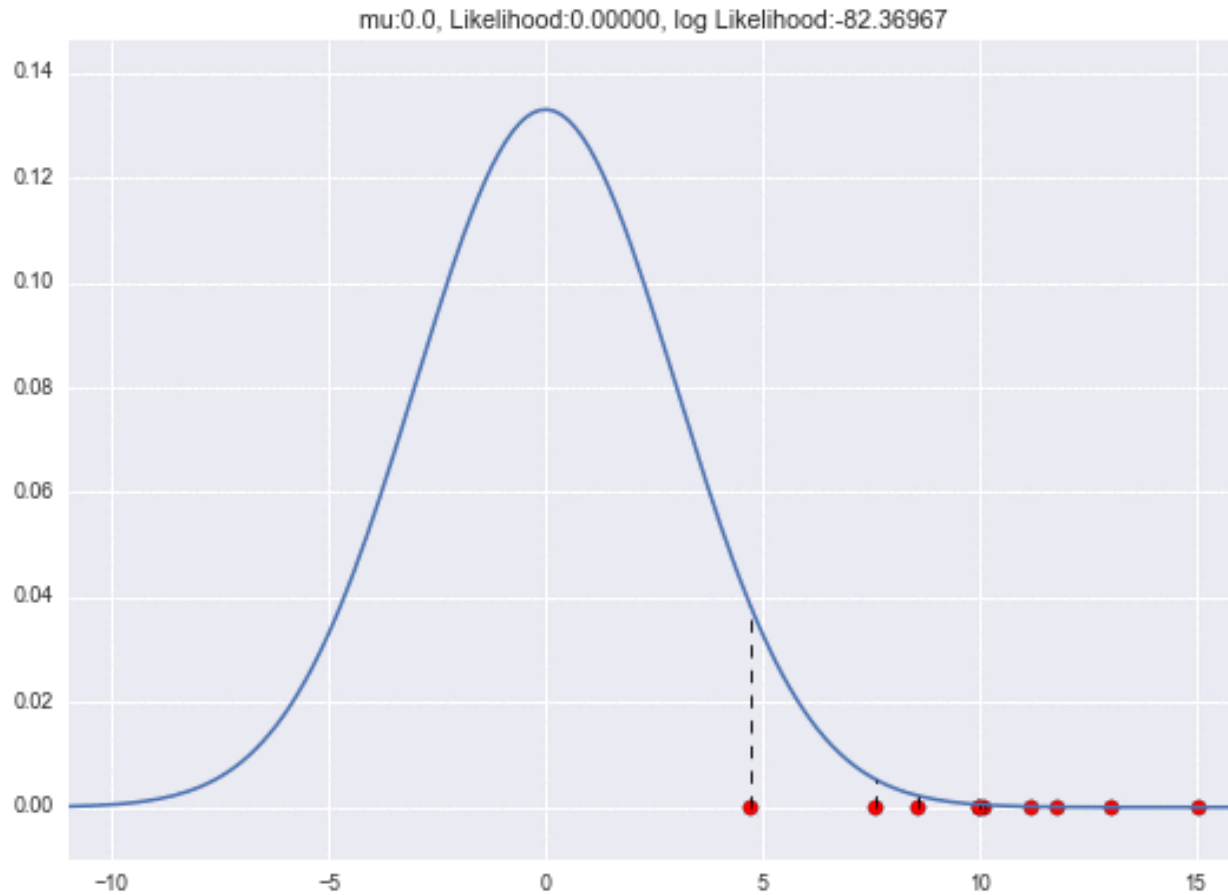
あるパラメータのもとで、データ (訓練標本) が生起する確率を考えて、それをパラメータの関数と見たものを尤度(Likelihood)とよぶ。  
尤度が最大になるようにパラメータを決定するほうほうが最尤推定である。

$$\text{尤度} \quad P(x_1, x_2, \dots, x_n | \theta)$$

例) データが10個、1次元ガウス分布のパラメータの尤度

$$\begin{aligned} & P(x_1, x_2, \dots, x_{10} | \mu, \sigma) \\ &= \prod_{i=1}^{10} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2} \frac{(x_i - \mu)^2}{\sigma^2}\right) \end{aligned}$$

# 最尤推定



# 最尤推定

- 最尤推定の理論的性質
  - 漸近普遍性: データ数が無限大の極限で、推定値のバイアスはゼロになる
  - 漸近有効性: データ数が無限大の極限で、推定値のエラー(バリエーション)が最小になる

統計的に優れたパラメータ推定法のひとつである

# 束縛ポテンシャルがある場合の尤度関数

$y$ 軸を $N$ 個のビンで区切り、  
束縛のない場合、 $i$ 番目のビンがもつ確率を $p_i$ とする。  
すると、束縛ポテンシャル $U_m(y)$ のもとでの確率 $p_i^{(m)}$ は、

$$p_i^{(m)} \propto p_i \exp(\beta U_m(y_i)) \quad y_i \text{はビン} i \text{の中心座標}$$

また、束縛ポテンシャル $U_m(y)$ のもとで観測された各ビンの  
ヒストグラムを $h_i$ とすると、パラメータ $p_i$ の尤度は

$$\begin{aligned} p^{(m)}(h_1, h_2, \dots, h_N | p_1, p_2, \dots, p_N) &= \frac{(h_1 + h_2 + \dots + h_N)!}{h_1! h_2! \dots h_N!} (p_1^{(m)})^{h_1} (p_2^{(m)})^{h_2} \dots (p_N^{(m)})^{h_N} \\ &\propto \frac{(h_1 + h_2 + \dots + h_N)!}{h_1! h_2! \dots h_N!} (p_1 \exp(\beta U_m(y_1)))^{h_1} (p_2 \exp(\beta U_m(y_2)))^{h_2} \dots (p_N \exp(\beta U_m(y_N)))^{h_N} \end{aligned}$$



# Weighted Histogram Analysis Method (WHAM)

前ページの尤度 $p^{(m)}$ を、全ての束縛ポテンシャル $m=1,\dots,M$ について独立事象として積をとると、トータルの尤度 $p$ が導出される

$$p(h_1, h_2, \dots, h_N | p_1, p_2, \dots, p_N) = \prod_{m=1}^M p^{(m)}(h_1, h_2, \dots, h_N | p_1, p_2, \dots, p_N)$$

この尤度をパラメータ $p_i$ について最大化して、最尤推定するのがWHAMである

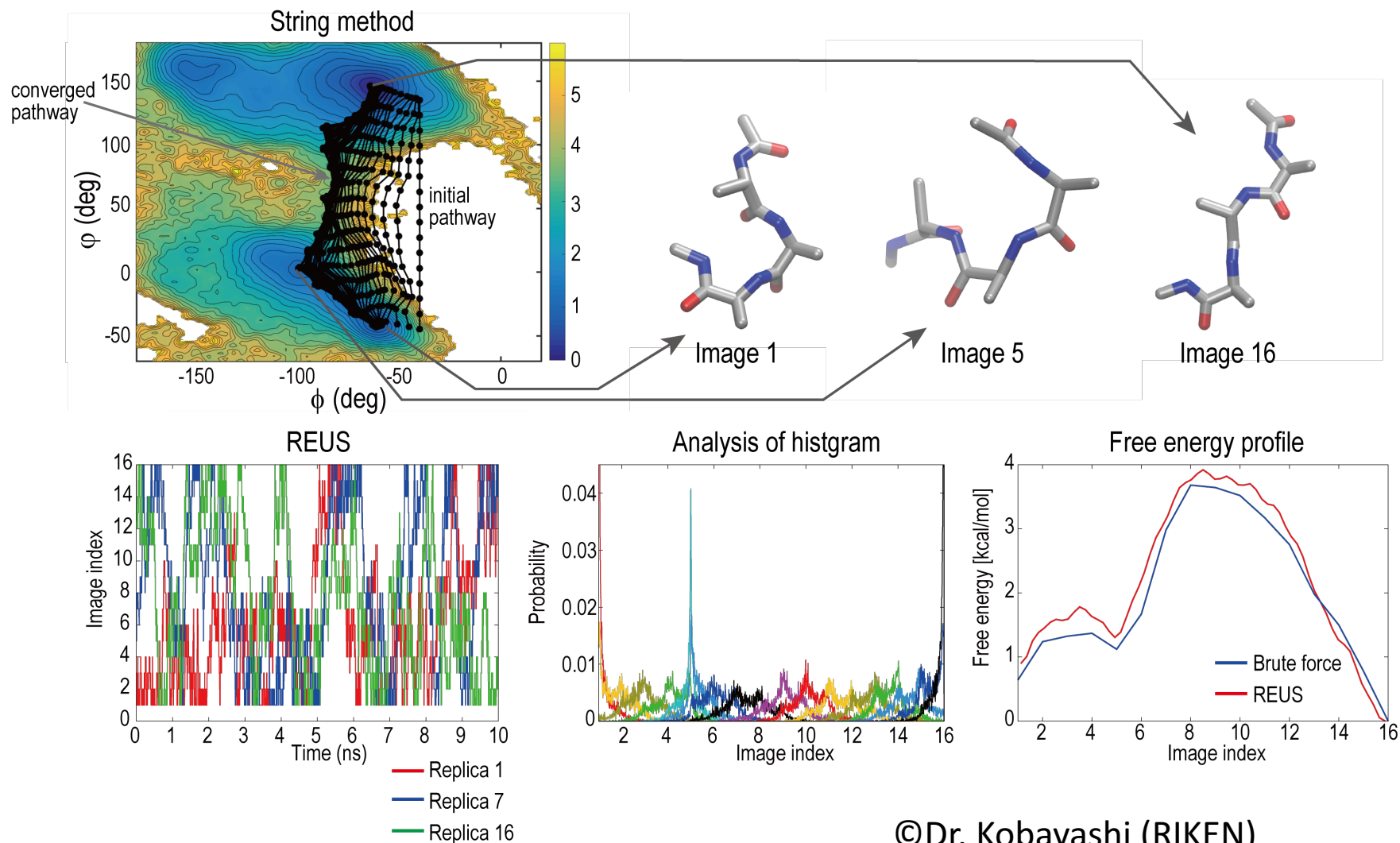
$p_i$ を推定した上で、 $y$ 軸における自由エネルギー変化は以下の式で計算できる

$$f_i = -\beta^{-1} \ln p_i$$

C. Bartels and M. Karplus, *J. Comput. Chem.* **18**, 1450 (1997).

S. Kumar, D. Bouzida, R. H. Swendsen, P. A. Kollman, and J. M. Rosenberg, *J. Comput. Chem.* **13**, 1011 (1992).

# WHAMの応用例



# まとめ

- 分子動力学法とGENESISについて簡単に紹介した
- 実習に必要なレプリカ交換法・WHAMの理論について説明した
- 問い合わせ先:
  - スライド: [ymatsunaga@riken.jp](mailto:ymatsunaga@riken.jp)
  - GENESIS: [genesis@riken.jp](mailto:genesis@riken.jp)