

CMSI計算科学技術特論A

# 古典分子動力学計算法の高速化1

2013年7月4日

名古屋大学大学院工学研究科  
附属計算科学連携教育研究センター  
吉井範行





# 講義内容

## MD計算とは

- MD計算の流れ
- 取り扱う原子数と周期境界条件
- 相互作用(力)
- 分子モデルと運動方程式
- MD計算の例
- MD計算の歴史
- MD計算の現状

## 非並列のMD計算

- 短距離相互作用 (bookkeeping, cell index法)
- 長距離相互作用 (クーロン相互作用) Ewald, PME法
- 効率化のための方法
  - 拘束動力学
  - On the flyによる最適化
  - Multiple time step  
(時間発展演算子法、シンプレクティック数値計算法)
- 高速化のために、キャッシュの有効利用、キャッシュミス

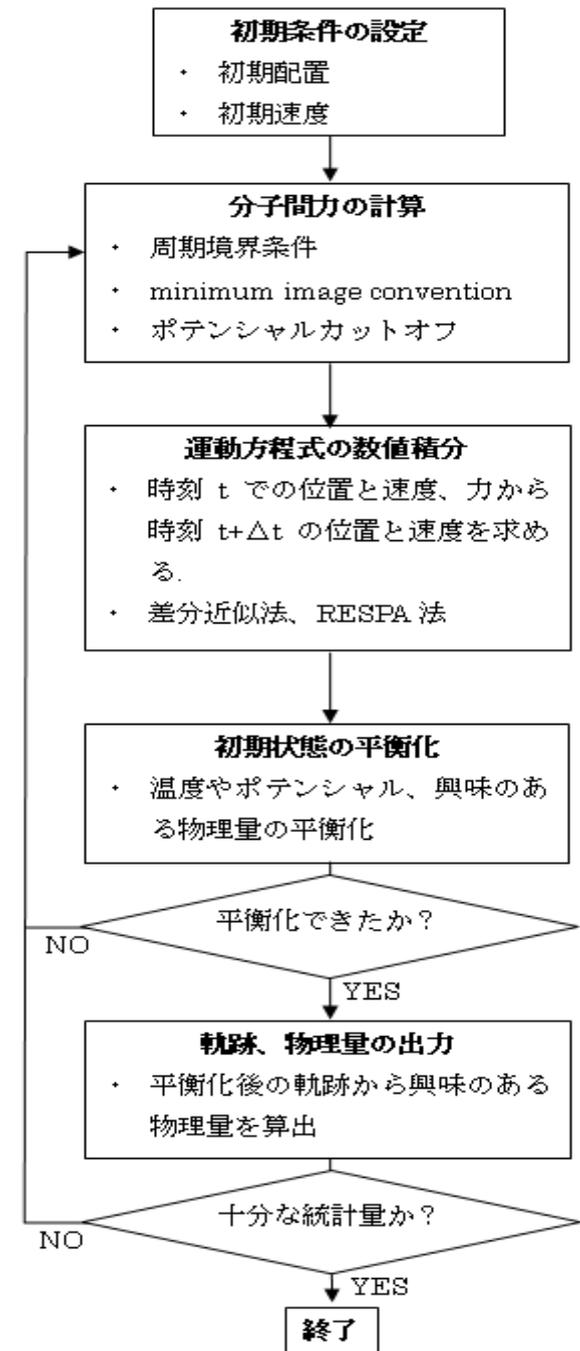
# 分子動力学(MD)計算とは



- 多数の原子や分子が集まってできた分子集団系(液体、固体、ガラス、高分子、生体分子)の熱力学、構造、ダイナミクスをコンピュータを用いて研究する手法である。
- 系を構成している全ての原子・分子の運動を運動方程式( $F=ma$ )を数値的に解くことにより、直接それら原子・分子の軌跡を追跡する。
- 原子・分子の運動方程式を解くために分子間に働く力が必要となる。この力が十分に良い精度であり、かつ運動方程式の数値解が十分な精度で得られているときに、分子の軌跡は実在系と同等であるとみなす。
- 得られた原子・分子の軌跡を用いることにより、熱力学関数、統計力学関数を求めることができる。

# MD計算の流れ

- 初期条件の設定  
初期座標・速度の入力
- 分子間力の計算  
LJ、クーロン相互作用(Ewald法, PME法, FMM) Hot Spot  
分子内相互作用(伸縮、変角、torsion, improper torsion, etc.)  
周期境界条件, minimum image convention
- 運動方程式の数値積分  
座標や速度を時間刻み $\Delta t$ だけ進める。  
(差分近似法、Gear predictor-corrector法、時間発展演算子法)  
温度一定(能勢-Hoover, 能勢-Poincaré法)  
圧力一定(Andersen, Parrinello-Rahman法)  
周期境界条件  
距離や速度の拘束(SHAKE法、RATTLE法、ROLL法)
- 初期状態の平衡化  
温度、圧力、ポテンシャルエネルギー、興味のある物理量
- 平衡化できたか？
- 軌跡、物理量の出力(熱力学、統計力学関数)  
座標、速度、温度、圧力、ポテンシャルエネルギー、  
運動エネルギー、ハミルトニアン、圧縮率、比熱等
- 十分に統計はとれたか？  
積算平均、誤差の評価
- 最終座標・速度の出力  
引き続き行う次のMD計算の初期座標・速度として



# 取り扱う原子数と周期境界条件



## ・原子数

数百~数億原子(長距離相互作用のある場合)  
対象系の規模や物理現象による。

## ・基本セルの一辺の長さ

原子間の相関がなくなる距離が $L/2$ より短くなるように

## ・周期境界条件

基本セル (立方体、直方体、平行六面体etc)

イメージセルを基本セルの周囲に配置。

→表面効果の除去

3次元のみでなく、2, 1次元方向にイメージセルを配置

2次元: 界面、平面膜、2枚の平板にはさまれた状態

1次元: 細孔、棒状分子

## ・ポテンシャルカット

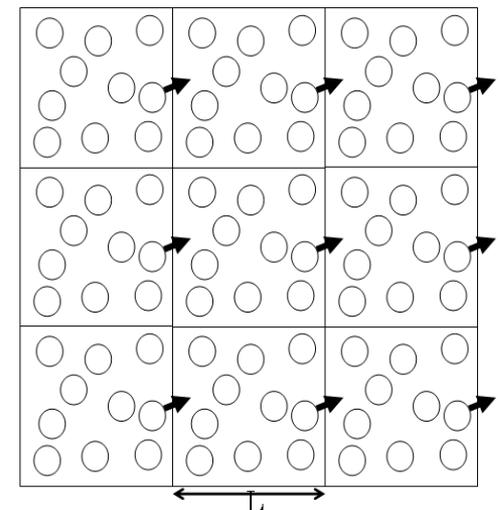
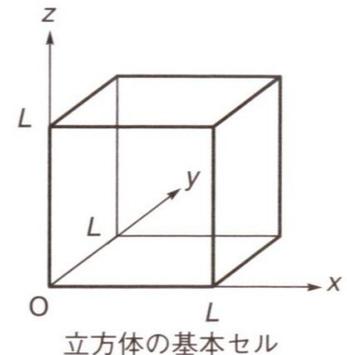
Lennard-Jones相互作用

Ewald法の実空間部分

通常、原子直径の3倍( $3\sigma$ )程度。

$L/2$ 以内が好ましい。

→ある原子とそのイメージ粒子と同時に相互作用



# 相互作用(力)

- 分子間相互作用(力)

2体力の和  $V_{\text{intermolecule}}(\mathbf{r}^N) = \sum_{i > j} V(r_{ij})$

Lennard-Jones(LJ)相互作用

$$V_{\text{LJ}}(r) = 4\epsilon \left\{ \left( \frac{\sigma}{r} \right)^{12} - \left( \frac{\sigma}{r} \right)^6 \right\}$$

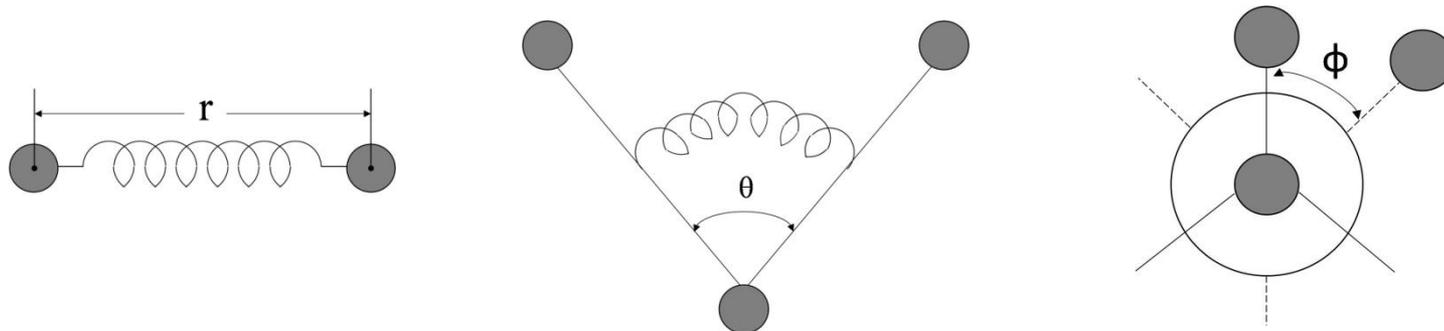
クーロン相互作用  $V_{\text{Coulomb}}(r) = \frac{q_1 q_2}{r}$

- 分子内相互作用(力)

分子内の原子を剛体的に取り扱わず、フレキシブルに運動できるとする。(フレキシブルモデル)  
分子内ポテンシャルを伸縮、変角、torsionなどで表現する。

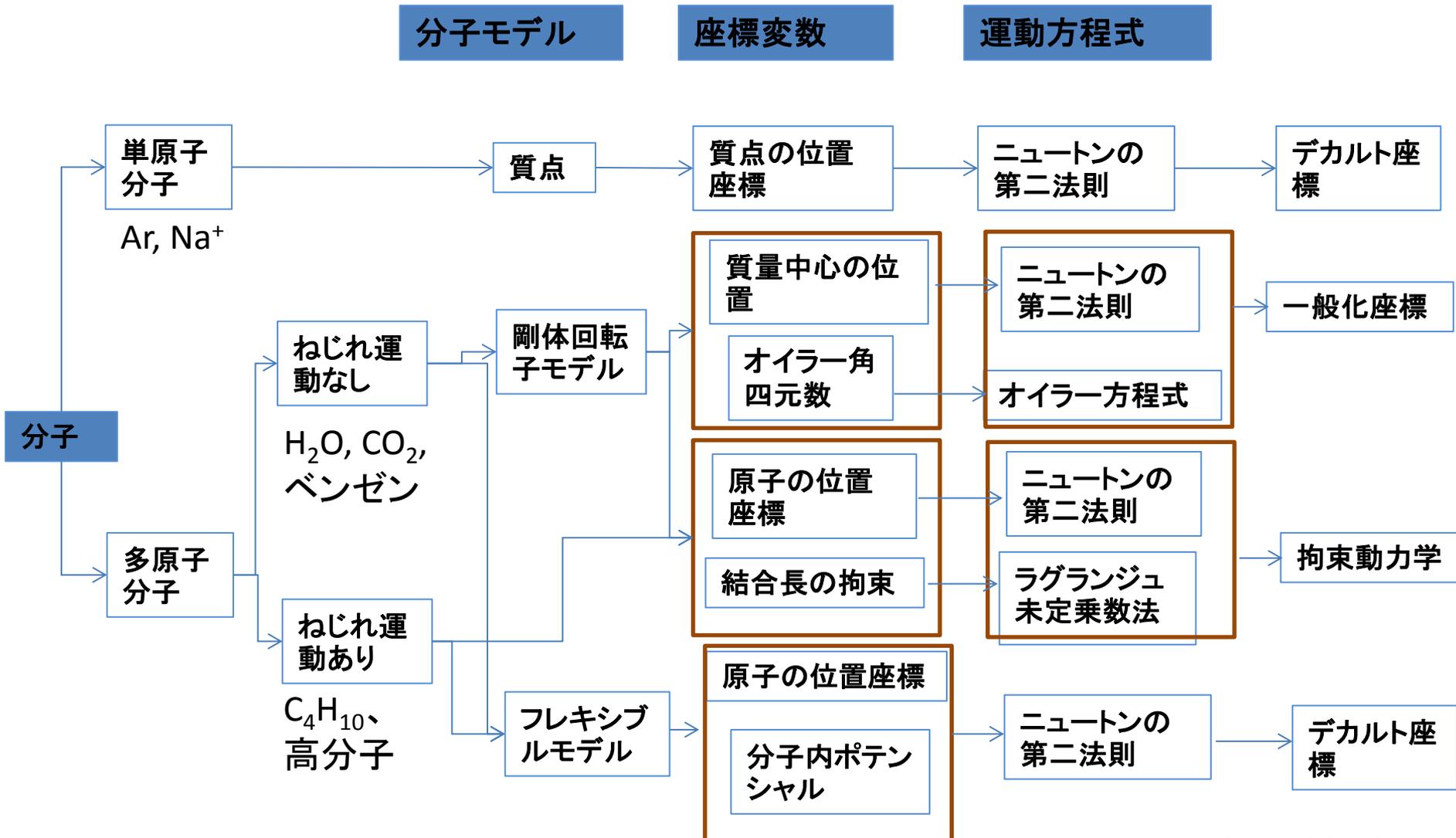
$$V_{\text{intramolecule}} = V_{\text{st}} + V_{\text{be}} + V_{\text{tor}}$$

$$= \sum k^{\text{st}} (r - r_0)^2 + \sum k^{\text{be}} (\theta - \theta_0)^2 + \sum_n \frac{1}{2} k_n^{\text{tor}} \{1 + \cos(n\phi - \phi_0)\}$$



分子間相互作用の精度がMD計算の質を決める

# 分子モデルと運動方程式

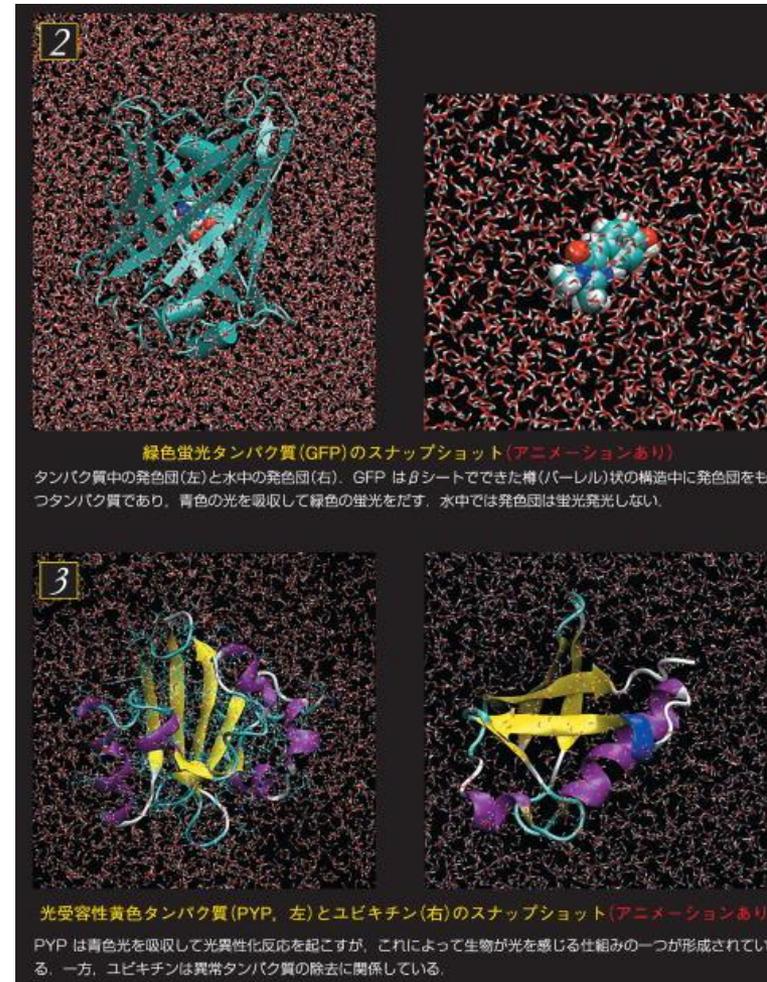
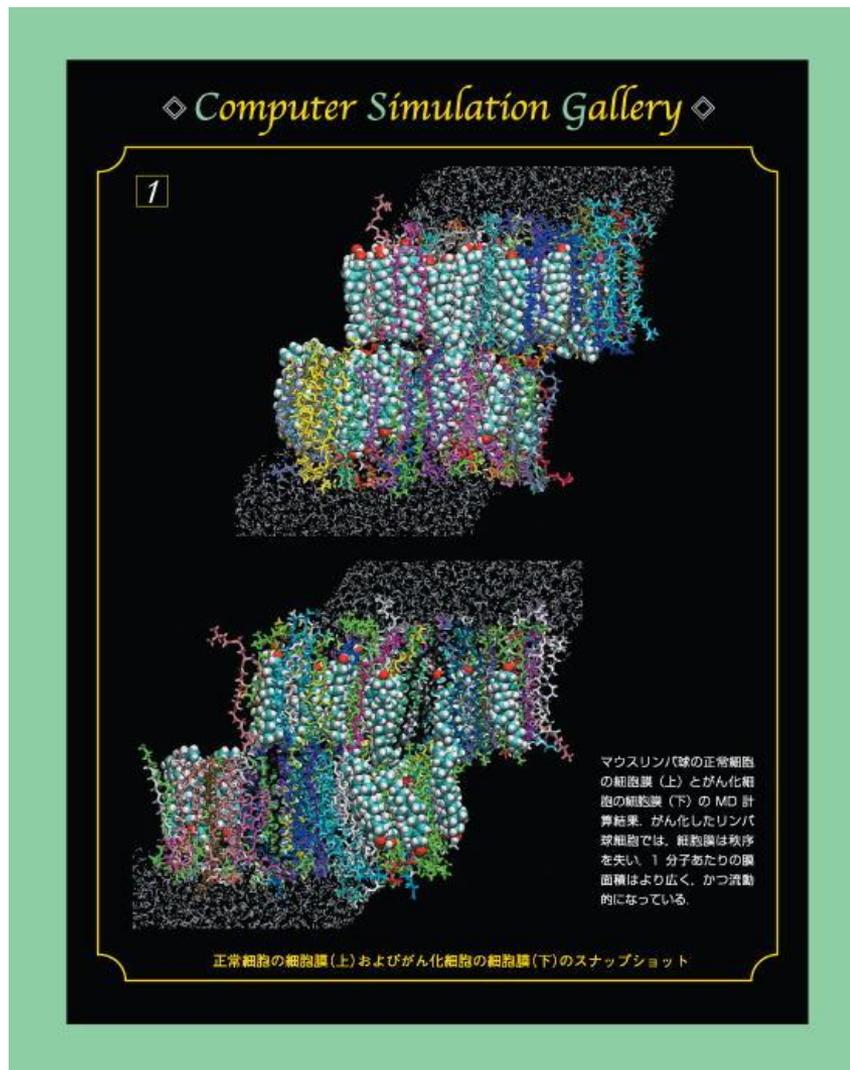




# MD計算の例

対象	系・現象	対象	系・現象
低分子液体、溶液	水、水溶液、 溶媒和、化学反応	タンパク質	折りたたみ、酵素、イオンチャネル、阻害剤、タンパク質複合体
固体	相転移、フォノン	DNA、RNA	
不均一系、界面	結晶成長、吸着、トライボロジー、蒸発、凝縮、透過、分配、界面増強スペクトル、界面張力	生体膜	脂質膜、ラフト、抗生物質、薬物透過、吸収
両親媒性分子水溶液 ミセル、単分子膜、二分子膜、バイセル	球状ミセル、棒状ミセル、 L膜、LB膜、ベシクル	ウイルス	カプシドタンパク質
化学反応	溶液内化学反応、 電解反応、 自由エネルギー曲面	ガラス	ガラス化、スローダイナミクス
超臨界流体	分子系、混合系、クラスターの生成	金属、半導体	電極、液体金属
高分子	分離膜、電解質膜、複雑系、スローダイナミクス	非平衡マクロ動力学	対流、ずり、核生成、流体力学

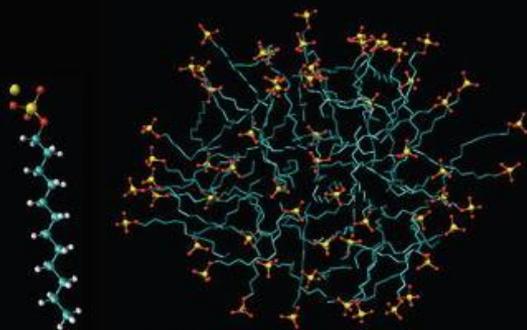
# MD計算の例



# MD計算の例



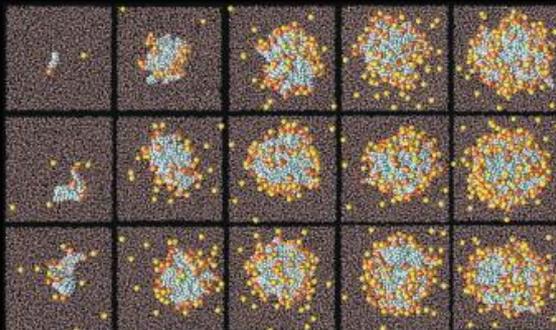
4



乱れたミセル構造のスナップショット

ドデシル硫酸ナトリウム(SDS)の分子構造(左)および SDS ミセルの MD 計算結果(右)。親水基を赤と黄色の球で、また疎水基を青色で示す。

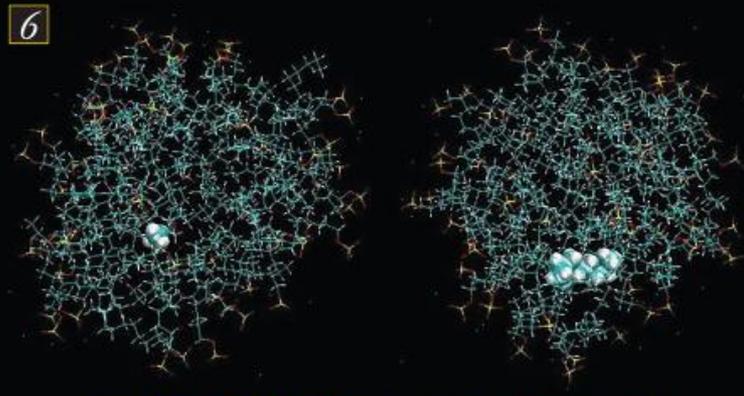
5



さまざまな大きさの球状ミセルのスナップショット

球状ミセル(SDS)に対する自由エネルギーの MD 計算から、水溶液中では約 60 個の分子からなる球状ミセルが安定に形成されることがわかった(下段中央)。

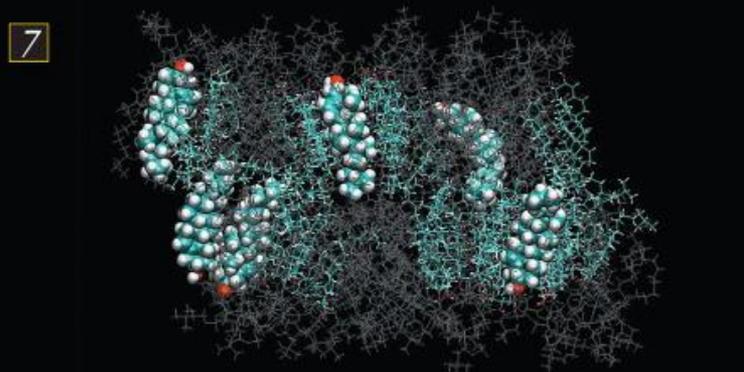
6



ミセルによる不溶性溶質の可溶化シミュレーション (アニメーションあり)

SDS 中に可溶化された不溶性溶質のメタン分子(左)とオクタン分子(右)。自由エネルギーの MD 計算から、水中よりもミセル中のほうが自由エネルギー的に安定であることが示される。

7

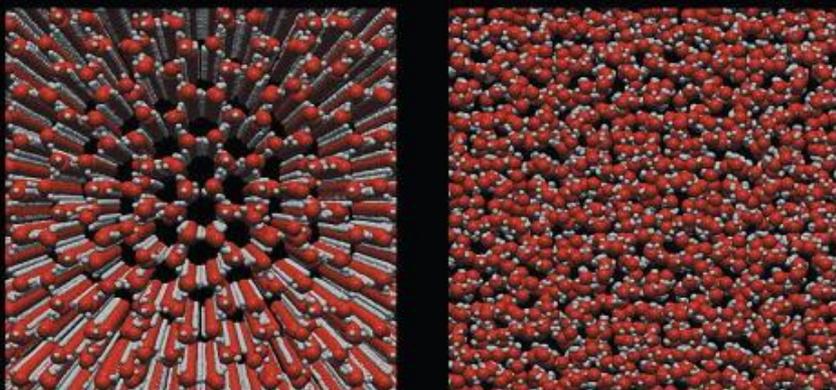


混合脂質膜のスナップショット (アニメーションあり)

ジパルミトイルフォスファチジルコリン (DPPC) + コレステロール (空間充填モデルで表示) 脂質混合系の MD 計算結果。コレステロールの添加により、膜はより秩序化し固くなる。

# MD計算の例

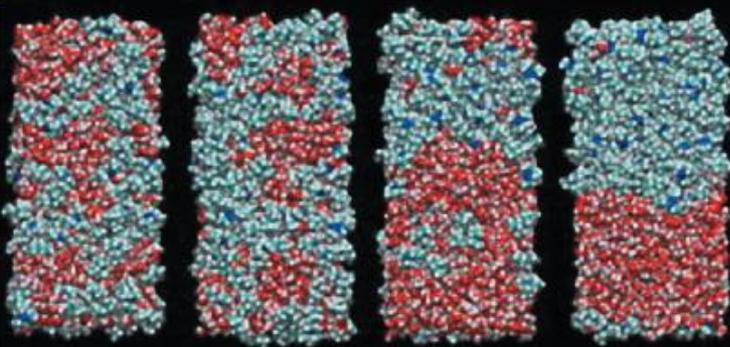
8



100万原子系のMD計算 (アニメーションあり)

氷(左)と水(右)。ここに示された原子数は少ないが、アニメーションは100万原子系に対するものである。現時点で、100万原子系のMD計算は一般的ではないが、数年後には日常的に計算されるようになるだろう。

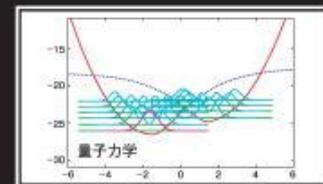
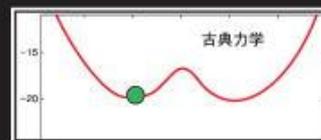
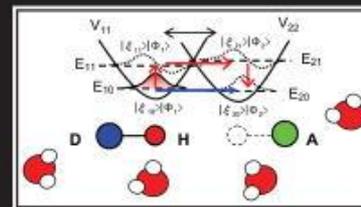
9



トリエチルアミン水溶液の相分離のスナップショット (アニメーションあり)

トリエチルアミン水溶液は下部臨界溶解温度をもつ二成分系であり、低温で完全に混合していた溶液(左端)は、時間の経過(右方向)とともに相分離する(右端)。水(赤色)、トリエチルアミン(水色)。

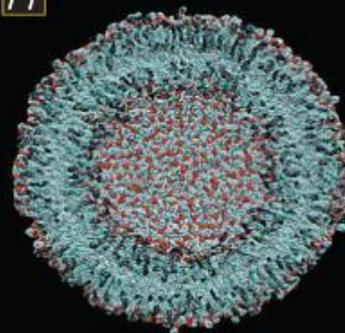
10



プロトン移動の量子動力学シミュレーション (アニメーションあり)

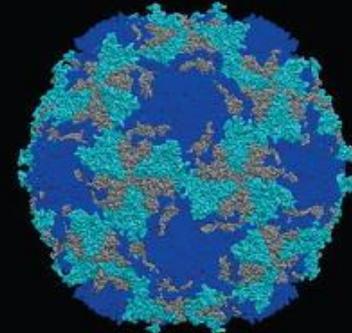
溶液中におけるマロンアルデヒドの分子内プロトン移動。量子効果が大きく、トンネル移動や振動励起、振動緩和など、運動は量子力学的に取り扱わなければならない。

11



リポソームの全原子MD計算によるスナップショット

水中に生成される球状の脂質二重層膜。多層のラメラ相と異なり、一枚膜である。計算は進行中で、300万原子系のMD計算が必要となる。



ウイルスカプシドの全原子MD計算によるスナップショット

小児マヒウイルスカプシド(タンパク質でできた球状の殻)。このなかにRNAが収められている。1000万原子系のMD計算が必要となる。現在、「次世代スパコン」を用いた計算が計画されている。

# MD計算の歴史



発表年	手法	対象	発表者
1953	MC	初めてのMC、剛体球	Metropolis et al.
1957	MC	球状分子	Wood and Parker
1957	MD	初めてのMD、剛体球	Alder and Wainwright
1964	MD	球状分子	Rahman
1969	MC	水	Barker and Watts
1971	MD	水(回転の運動方程式)	Rahman and Stillinger
1971	MC	イオン	Woodcock
1977	MD	拘束動力学	Rychaert et al.
1977	MD	タンパク質分子	McCammon et al.
1980	MD	圧力一定	Andersen
1980	MD	圧力テンソル一定	Parrinello and Rahman
1981	MD	経路積分、量子系	Chandler and Wolynes
1984	MD	温度一定	能勢
1985	MD	第一原理MD	Car and Parrinello
1986	MD	溶液内化学反応	Hynes et al.
1987	MD, MC	FMM	Greengard
1988	MD, MC	汎用ポテンシャル	Jorgensen et al.
1991	MD	非断熱量子動力学	Rosky et al.



# MDの現状

- ・汎用ポテンシャルセットの開発・・・対象系の分子のポテンシャルが簡単に構築できる
  - －多様な化合物に対応。生体分子(タンパク質、脂質、核酸、糖)など
  - －官能基が同じならポテンシャルパラメータも同じ
  - －同じ原子でも異なる官能基ではパラメータが異なる(アミノ酸の主鎖C $\alpha$ とC=OのC)
  - －異原子間のLJ相互作用では、本来組み合わせごとにパラメータ $\sigma$ 、 $\epsilon$ が決まるが、簡単のために2つの原子のパラメータの相加平均や相乗平均で近似してしまう。ローレンツ・ベルテロー則。

## ・粗視化モデル

- －複数の原子を一つにまとめて相互作用点を大幅に減らした粗視化モデルの汎用ポテンシャルセットも開発されている。
- －大規模系が取扱い可能に
- －両親媒性分子のポリモルフへの適用

ポテンシャル名	開発者
AMBER	Kollman
CHARMM	Karplus
OPLS	Jorgensen
GROMOS	Berendsen

プログラム名	開発者
AMBER	Kollman
CHARMM	Karplus
GROMACS	Hess
TINKER	Ponder
NAMD	Schulten
DESMOND	Shaw
MODYLAS	Okazaki



# 非並列のMD計算

シングルコアで高速化を進める

MD計算手法の選択

短距離力の計算 (bookkeeping, cell index)

長距離力の計算 (Ewald, PME, FMM)

拘束動力学の利用

Multiple time stepの導入

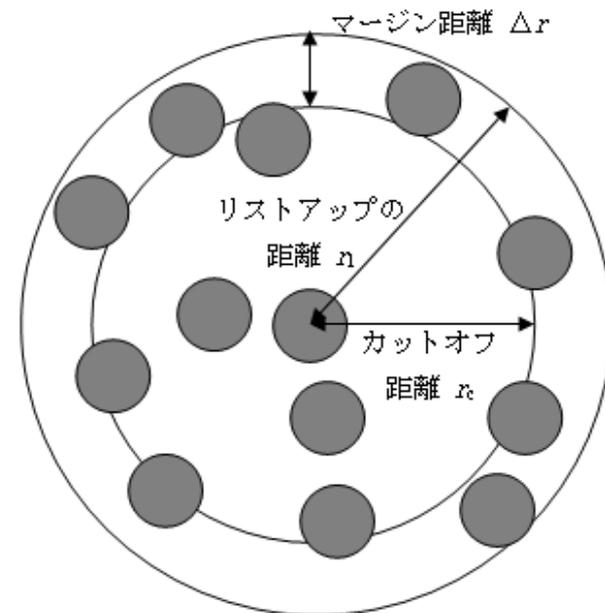
on the fly計算の利用

- 高速化のための工夫

# 短距離相互作用 bookkeeping法



- ある分子からの距離が $r_c$ よりも短い相手分子の番号すべてをリストアップ。
- リストの作成は何ステップかに1回毎。  
固体のような流動性のないものなら更新不要
- $r_c$ よりも $\Delta r$ だけ長い距離のものまで余分にリストに載せる。
- リストを用いて相互作用を計算する際には、リストを複数のノードやスレッドに割り振って並列計算させるループ分割法が容易に実装可能。



帳簿法.

L	N	N'	$L/r_1$	$N'/N$
$9\sigma$	600	150	2.6	0.25
$12\sigma$	1500	150	3.4	0.10
$15\sigma$	3000	150	4.3	0.05
$18\sigma$	5000	150	5.1	0.03



```
k=0
do i=1,n-1
  do j=i+1,n
    i, j間の距離が $r_l$ 以下
    k=k+1
    list(1,k)=i   リスト作成
    list(j,k)=j   リスト作成
  enddo
enddo
```

```
do i=1,k
  リストをもとに相互作用計算
enddo
```

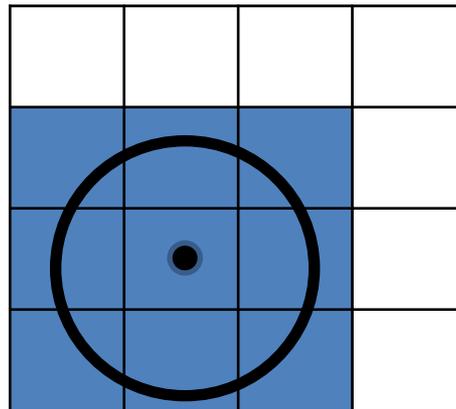
MPI並列なら  
相互作用の計算結果をノード間で通信  
スレッド並列  
通信不要

- 数ステップに一度相互作用対のリスト(list)を更新する
- 毎ステップリストを用いて相互作用計算する。

# 短距離相互作用 Cell index法



- 帳簿を作成, 更新する際の $N(N-1)/2$ 通りの距離計算が大きな負荷. 大きなメモリを要する.
- $L/r_i$ が4以上の場合には, 一辺の長さが $r_i$ よりも大きくなる範囲内で基本セルそのものをx, y, zそれぞれの方向に4分割以上の分割を行う.
- 注目している分子が属しているセルとそれに直接接しているセルあわせて27個の分割セル内にある分子との間だけでbookkeeping法を行う.
- 分割セルの一辺の長さをよりも短くする場合には, 第二近接のセルまで考慮に入れる
- Cell index法についても作成したリストを複数のノードやスレッドに割り振って相互作用の並列計算を行えばよい.



# 長距離相互作用 クーロン相互作用



- 直接計算  $O(N^2)$   
→計算方法はLJ相互作用と同じ  
カットオフの悪影響大のため用いない。
- エwald法  $O(N^2)$  あるいは  
Particle Mesh Ewald法  $O(N\log N)$   
→小、中規模系向け
- 高速多重極展開法 (Fast Multipole Method: FMM)  $O(N)$   
→大規模系向け

# 長距離相互作用 Ewald法



## Ewald法の要点

- ・減衰の遅いクーロン相互作用を、減衰の速い関数と遅い関数に分ける。
- ・遅い関数はフーリエ変換することにより逆格子ベクトルを用いて評価。
- ・速い関数は実空間で評価。

$$V_N(\mathbf{r}^N) = \frac{1}{2} \sum_n \sum_i \sum_j' \frac{1}{4\pi\epsilon_0} \frac{Q_i Q_j}{|\mathbf{r}_i - \mathbf{r}_j + L\mathbf{n}|}$$

(減衰の遅い $1/r$ )

$$= \begin{array}{l} \text{(減衰の速い関数)} \\ + \text{(減衰の遅い関数)} \end{array} \quad \frac{1}{r} = \frac{\text{erf}(\alpha r)}{r} + \frac{\text{erfc}(\alpha r)}{r}$$

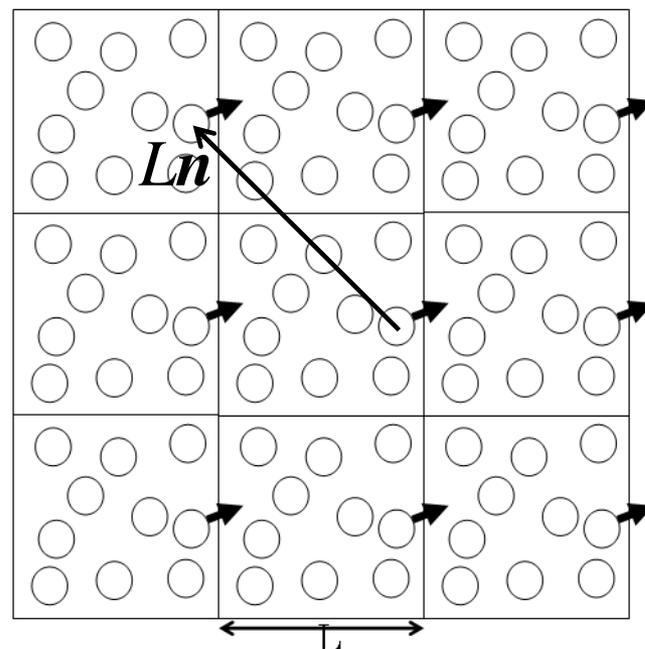
減衰の速い関数 ... この関数形で評価

$$\frac{\text{erfc}(\alpha r)}{r}$$

減衰の遅い関数 ... フーリエ変換したもので

$$\frac{\text{erf}(\alpha r)}{r}$$

評価



$\mathbf{n}$  整数ベクトル  $(-1, 2, 3)$

# 長距離相互作用 Ewald法



## • Ewald法によるポテンシャル

$$V_N = V_N^{(1)} + V_N^{(2)} + V_N^{(3)}$$

bookkeeping法  
セルインデックス法

$$V_N^{(1)} = \frac{1}{2} \sum_n \sum_i \sum_j \frac{Q_i Q_j}{4\pi\epsilon_0} \frac{\operatorname{erfc}(\alpha |\mathbf{r}_i - \mathbf{r}_j + L\mathbf{n}|)}{|\mathbf{r}_i - \mathbf{r}_j + L\mathbf{n}|}$$

実空間での相互作用  
Lと一緒に計算する

$$V_N^{(2)} = \frac{1}{2\epsilon_0 L^3} \sum_{\mathbf{G} \neq 0} \frac{\exp(-|\mathbf{G}|^2 / 4\alpha^2)}{|\mathbf{G}|^2} \left[ \left\{ \sum_i Q_i \cos(\mathbf{G} \cdot \mathbf{r}_i) \right\}^2 + \left\{ \sum_i Q_i \sin(\mathbf{G} \cdot \mathbf{r}_i) \right\}^2 \right]$$

Gについてのループ分割

$$V_N^{(3)} = -\sum_i \frac{Q_i^2}{4\pi\epsilon_0} \frac{\alpha}{\sqrt{\pi}}$$

通常このままでは用いない。



# Particle mesh Ewald法

逆格子ベクトルの項は

$$V_N^{(2)} = \frac{1}{2\varepsilon_0 L^3} \sum_{\mathbf{G} \neq 0} \frac{\exp\left(-\frac{|\mathbf{G}|^2}{4\alpha^2}\right)}{|\mathbf{G}|^2} \sum_i Q_i \exp(i\mathbf{G} \cdot \mathbf{r}_i) \sum_j Q_j \exp(-i\mathbf{G} \cdot \mathbf{r}_j)$$
$$= \frac{1}{2\varepsilon_0 L^3} \sum_{\mathbf{G} \neq 0} H(\mathbf{G}) S(\mathbf{G}) S(-\mathbf{G})$$

と書き表せる。ここで

$$H(\mathbf{G}) = \frac{\exp\left(-\frac{|\mathbf{G}|^2}{4\alpha^2}\right)}{|\mathbf{G}|^2} \quad S(\mathbf{G}) = \sum_i Q_i \exp(i\mathbf{G} \cdot \mathbf{r}_i)$$

$S(\mathbf{G})$ を毎ステップ計算しなければならない。

$S(\mathbf{G})$ を格子点 $\mathbf{u}$ における補間を用いて表現すると

$$S(\mathbf{G}) = \sum_{\mathbf{u}} \hat{S}(\mathbf{u}) \exp(i\mathbf{G} \cdot \mathbf{u})$$

となる。ここで $\hat{S}(\mathbf{u})$ は $S(\mathbf{G})$ の $\mathbf{u}$ における重み因子。これにより

$$V_N^{(2)} = \frac{1}{2\varepsilon_0 L^3} \sum_{\mathbf{u}} \hat{H} * \hat{S}(\mathbf{u}) \hat{S}(\mathbf{u})$$

$S(\mathbf{G})$ の計算に補間法を利用(格子点で評価)

高速フーリエ変換(FFT)利用  $O(N \log N)$

# 効率化のための方法



- 拘束動力学
- Multiple time step
- On the fly による最適化



# 拘束動力学

- 剛体分子

- (1) 剛体回転子

- 並進: 質量中心のデカルト座標に対してニュートンの運動方程式を解く。

- 回転: 配向を記述するオイラー角あるいは四元数についてのオイラー方程式を解く。

- (2) 拘束条件付きの運動方程式

- 分子を構成している原子のすべてに並進の自由度を与えつつ, 原子に拘束条件を課して運動させる。ラグランジュの未定乗数法を利用しつつニュートンの運動方程式を解く。



# 拘束動力学

- 原子間距離に関するホロノミックな拘束

一次元の2原子系

たとえば

$$g[x_1(t), x_2(t)] = 0$$

$$g = \{x_1(t) - x_2(t)\}^2 - r_0^2 = 0$$

系のラグランジュアン

$$L[x_1(t), x_2(t), \dot{x}_1(t), \dot{x}_2(t)]$$

作用

$$S = \int_{t_1}^{t_2} L[x_1(t), x_2(t), \dot{x}_1(t), \dot{x}_2(t)] dt$$

拘束 $g$ を満足させながら、なおかつハミルトンの原理から $S$ が停留値をとること、つまり $\delta S = 0$ を満足させるものである。

これから述べることは、このような軌跡がどのような微分方程式に従うか、ということ。



# 拘束動力学

## • 拘束条件付きの運動方程式

拘束条件

$$g_k = (\mathbf{r}_i - \mathbf{r}_j)^2 - d_{ij}^{(k)2} = 0$$

分子を区別せずに*i*番目の原子として

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\mathbf{r}}_i} \right) - \frac{\partial L}{\partial \mathbf{r}_i} = \sum_{k=1}^K \lambda_k \frac{\partial g_k}{\partial \mathbf{r}_i}$$

ラグランジュの未定乗数 $\lambda_k$

$$\frac{\partial g_k}{\partial \mathbf{r}_i(t)} = \begin{cases} 2\{\mathbf{r}_i(t) - \mathbf{r}_j(t)\} & (\text{拘束に原子}i\text{が関与しているとき}) \\ 0 & (\text{そうでないとき}) \end{cases}$$

$$L = \sum_i \frac{1}{2} m_i \dot{\mathbf{r}}_i^2 - V(\mathbf{r}^N) \text{ を}$$

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\mathbf{r}}_i} \right) - \frac{\partial L}{\partial \mathbf{r}_i} = \sum_{k=1}^K \lambda_k \frac{\partial g_k}{\partial \mathbf{r}_i} \quad \text{に代入}$$

$$m_i \ddot{\mathbf{r}}_i = \mathbf{F}_i + \sum_{k=1}^K \lambda_k \frac{\partial g_k}{\partial \mathbf{r}_i}$$

これが解くべき運動方程式

# 拘束動力学

- ベルレ法を用いて解く
- 拘束力をあらかじめ求めておいて, これを $F_i$ に加えあわせてから解く, というような計算はできない
- $t = t + \Delta t$ においても満足されるように $\lambda_k$ を決定する.

$$\begin{aligned} \mathbf{r}_i(t + \Delta t) &= 2\mathbf{r}_i(t) - \mathbf{r}_i(t - \Delta t) + \frac{\Delta t^2}{m_i} \left\{ \mathbf{F}_i(t) + \sum_{k=1}^K \lambda_k \frac{\partial g_k}{\partial \mathbf{r}_i} \right\} \\ &= 2\mathbf{r}_i(t) - \mathbf{r}_i(t - \Delta t) + \frac{\Delta t^2}{m_i} \left\{ \mathbf{F}_i(t) + \mathbf{F}_i^{(c)}(t) \right\} \\ &= \mathbf{r}_i'(t + \Delta t) + \Delta \mathbf{r}_i(t + \Delta t) \end{aligned}$$

ただし

$$\begin{aligned} \mathbf{r}_i'(t + \Delta t) &\equiv 2\mathbf{r}_i(t) - \mathbf{r}_i(t - \Delta t) + \frac{\Delta t^2}{m_i} \mathbf{F}_i(t) \\ \Delta \mathbf{r}_i(t + \Delta t) &\equiv \frac{\Delta t^2}{m_i} \sum_{k=1}^K \lambda_k \frac{\partial g_k}{\partial \mathbf{r}_i(t)} = \frac{\Delta t^2}{m_i} \mathbf{F}_i^{(c)}(t) \end{aligned}$$

# 拘束動力学 SHAKE法



一方、拘束条件より

$$g_k = [\{\mathbf{r}'_i(t + \Delta t) + \Delta \mathbf{r}_i(t + \Delta t)\} \\ - \{\mathbf{r}'_j(t + \Delta t) + \Delta \mathbf{r}_j(t + \Delta t)\}]^2 - d_{ij}^{(k)2} = 0$$

これより $\lambda_k$ に対する $K$ の二次の連立方程式が得られる

- 二次の連立方程式を直接解いてもよいが、この方法だとタンパク質のように分子が大きくなると長いCPU時間を要するようになる。そこで反復法を用いる。これをSHAKE法という。

$k$ 番目の拘束 $g_k$ に原子 $i$ と $j$ が関与しているとして、仮に系にはこの拘束だけが単独に働いていると考える。

$$\Delta \mathbf{r}_i(t + \Delta t)^{(k)} = \frac{2\Delta t^2}{m_i} \lambda_k \{\mathbf{r}_i(t) - \mathbf{r}_j(t)\}$$

$$\gamma_k = 2\Delta t^2 \lambda_k \quad \text{定義}$$

原子 $i, j$ に対する拘束力による変位

$$\Delta \mathbf{r}_i(t + \Delta t)^{(k)} = \gamma_k \frac{\mathbf{r}_i(t) - \mathbf{r}_j(t)}{m_i}$$

$$\Delta \mathbf{r}_j(t + \Delta t)^{(k)} = \gamma_k \frac{\mathbf{r}_j(t) - \mathbf{r}_i(t)}{m_j} = -\gamma_k \frac{\mathbf{r}_i(t) - \mathbf{r}_j(t)}{m_j}$$

# 拘束動力学 SHAKE法



$$\left[ \{r'_i(t + \Delta t) - r'_j(t + \Delta t)\} + \left( \frac{1}{m_i} + \frac{1}{m_j} \right) \{r_i(t) - r_j(t)\} \gamma_k \right]^2 - d_{ij}^{(k)2} = 0$$

$$\left[ \left( \frac{1}{m_i} + \frac{1}{m_j} \right) \{r_i(t) - r_j(t)\} \right]^2 \gamma_k^2$$

$$+ \left[ 2 \left( \frac{1}{m_i} + \frac{1}{m_j} \right) \{r_i(t) - r_j(t)\} \{r'_i(t + \Delta t) - r'_j(t + \Delta t)\} \right] \gamma_k$$

$$+ \{r'_i(t + \Delta t) - r'_j(t + \Delta t)\}^2 - d_{ij}^{(k)2} = 0$$

$\gamma_k$ に対する二次方程式

- $K$ 個ある拘束条件のすべてを満足するように $\gamma_k$ の解を求めなければならない。
  - (1) 系には一番目の拘束力だけが働いているとして、上式から $\gamma_1$ を求め、一番目の拘束に関与している2個の原子の座標を、一番目の拘束だけが満足されるように置き直す。
  - (2) この新たな座標を $r'_i(t + \Delta t)$ と考えて、次に二番目の拘束だけが働いているとして、これに関与している2個の原子の座標を二番目の拘束条件を満足するように置き直す。
  - (3) そして、次に三番目、四番目というように、 $K$ 個の拘束が単独に存在しているとして、順次関与している原子の位置を補正していく。

# 拘束動力学 RATTLE法



- 速度ベルレ法と結びついた拘束の動力学
- SHAKE法では位置に関する拘束のみ

速度に関しても

$$\frac{dg_k}{dt} = 2(\dot{r}_i - \dot{r}_j) \cdot (r_i - r_j) = 0$$

まず位置に関して

$$\begin{aligned} r_i(t + \Delta t) &= r_i(t) + \Delta t v_i(t) + \frac{\Delta t^2}{2m_i} F_i(t) + \frac{\Delta t^2}{2m_i} \lambda_k \frac{\partial g_k}{\partial r_i} \\ &= r_i(t) + \Delta t v_i(t) + \frac{\Delta t^2}{2m_i} F_i(t) + \gamma_k(t) \frac{r_i(t) - r_j(t)}{2m_i} \end{aligned}$$

一方, 速度に関して

$$\begin{aligned} v_i(t + \Delta t) &= v_i(t) + \frac{\Delta t}{2m_i} \{F_i(t) + F_i(t + \Delta t)\} \\ &+ \frac{1}{2\Delta t} \left\{ \gamma_k(t) \frac{r_i(t) - r_j(t)}{m_i} + \gamma'_k(t + \Delta t) \frac{r_i(t + \Delta t) - r_j(t + \Delta t)}{m_i} \right\} \end{aligned}$$

$\gamma'_k(t + \Delta t)$  速度に対する拘束を満足するように新たに定めなければならない

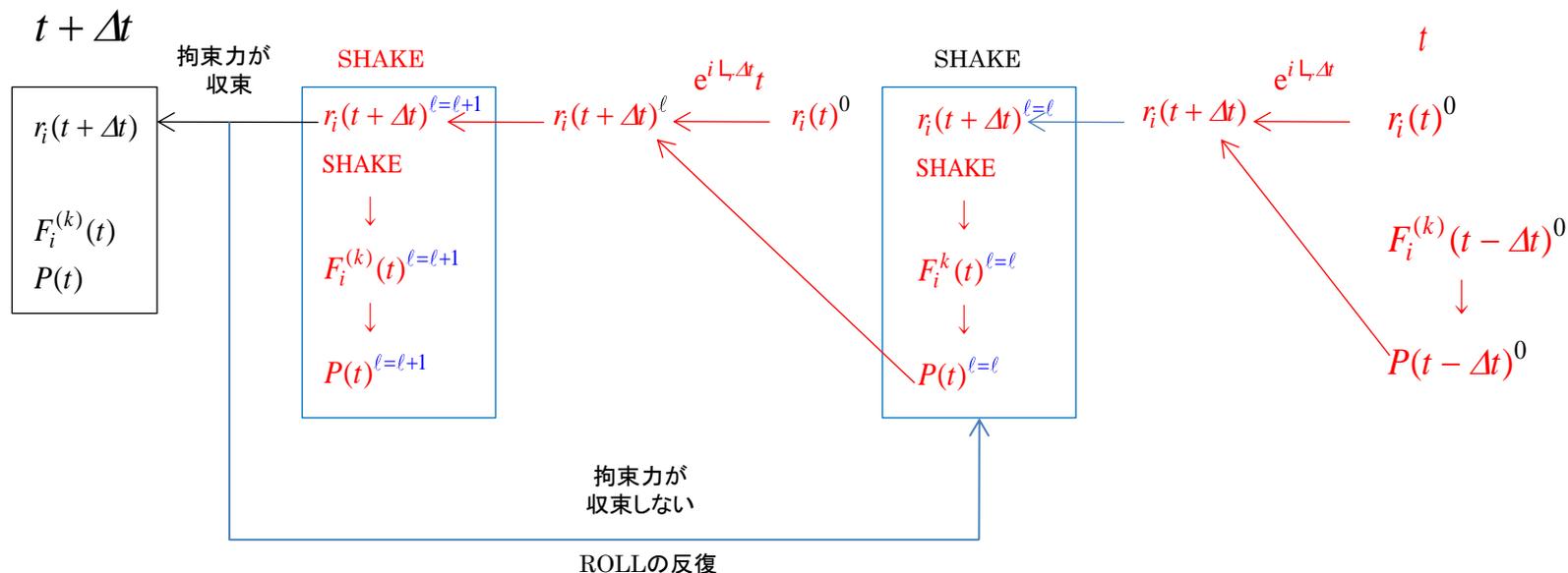
一次方程式 → 拘束が一つなら解析的、2つ以上の連立であればSHAKE法と同様に反復法で解く。

速度ベルレ法と結びついたこのような拘束の動力学の数値解法をRATTLE法と呼んでいる

# 拘束動力学 ROLL法

- 圧力一定のMD計算を行う場合、 $t = t$ での力と圧力が必要
- 拘束力が働く場合、 $t = t$ での圧力を求めるためには $t = t$ で働く拘束力も必要となる。(後述)
- 一方、拘束力は $t = t + \Delta t$ で初めて求まるもの。
- このような場合は反復法を用いて自己無撞着に $t = t$ での拘束力と圧力を決める。

## • SHAKE/ROLL法



# On the fly による最適化



- 第一原理MD (Car-Parrinello法)
- 電荷揺らぎモデル (TIP4P-FQ、SPC-FQモデル)
- 分極モデル (点双極子モデル)
  
- 最適値を得るために繰り返し計算を必要とする変数を含む
- 繰り返し計算を行わず、最適値近傍を動的に揺らがせる近似解で満足できる場合
- その変数に仮想的な慣性量、運動エネルギーを与えて運動方程式を解く。
- 繰り返し計算がなくなるため高速化が可能。
- 運動エネルギーの相当する分だけ最適値からずれて、誤差が生じる。

## 分極モデル

$$U = U_0 + U_{\text{pol}}$$

$$U_{\text{pol}} = -\sum_i \mathbf{E}_i \cdot \boldsymbol{\mu}_i + \frac{1}{2\alpha} \sum_i \boldsymbol{\mu}_i \cdot \boldsymbol{\mu}_i$$

$$\mathbf{E}_i = \mathbf{T}_{ij} \cdot \boldsymbol{\mu}_j$$

$$\boldsymbol{\mu}_i = \alpha \mathbf{E}_i$$

## 拡張系のラグランジアン

粒子と双極子モーメントの自由度がある。

$$L(\mathbf{r}^N, \boldsymbol{\mu}^N) = \frac{1}{2} \sum_{i=1}^N m \dot{\mathbf{r}}_i^2 + \frac{1}{2} \sum_{i=1}^N M \dot{\boldsymbol{\mu}}_i^2 - U$$

## 双極子モーメントの運動方程式

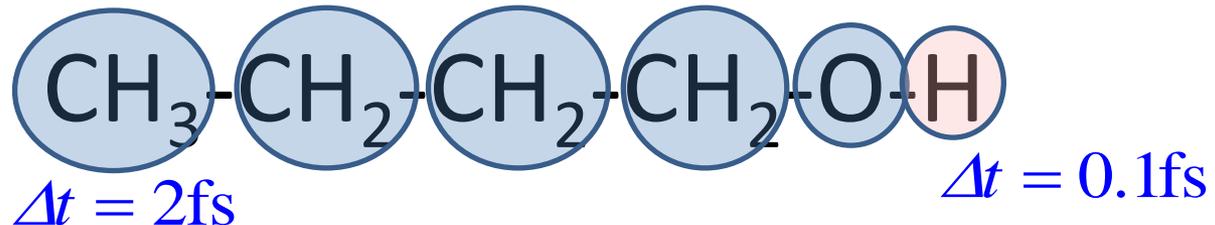
$$M \dot{\boldsymbol{\mu}}_i = \frac{\partial}{\partial \boldsymbol{\mu}} L(\mathbf{r}^N, \boldsymbol{\mu}^N) = -\frac{\boldsymbol{\mu}_i}{\alpha} + \mathbf{E}_i$$

# Multiple time step

- 同じMD計算のなかで変化の速いものと遅いものがある場合には、速い変化のほうに対応した小さな $\Delta t$ を選択しなければならない。

例

United atomのブチルアルコールのフレキシブルモデル



- Hに対する運動方程式とほかの重い原子に対する運動方程式を区別して、異なった $\Delta t$ で解くことができれば非常に都合のよいことになる。この考え方に基づいた方法がMultiple time step法である。

# 時間発展演算子法

- 時間発展演算子法により、重い(遅い)運動と軽い(速い)運動とを分離して数値積分する。
- 時間発展演算子法

リウビル演算子  $L$

$$iL = iL_r + iL_p = v \frac{\partial}{\partial r} + \frac{F}{m} \frac{\partial}{\partial v}$$

時間発展演算子

$$e^{iL\Delta t} \quad q_i(\Delta t) = e^{iL\Delta t} q_i(0)$$

異なる自由度など2つの部分に分離できると  
トロッター展開により

$$e^{iL\Delta t} = e^{iL_p\Delta t/2} e^{iL_r\Delta t} e^{iL_p\Delta t/2} + O(\Delta t^3)$$

$$r_i(\Delta t) = e^{iL_p\Delta t/2} e^{iL_r\Delta t} e^{iL_p\Delta t/2} r_i(0)$$



速度ベルレ法

$$\begin{cases} r_i(t + \Delta t) = r_i(t) + \Delta t v_i(t) + \frac{\Delta t^2}{2m_i} F_i(t) \\ v_i(t + \Delta t) = v_i(t) + \frac{\Delta t}{2m_i} \{F_i(t) + F_i(t + \Delta t)\} \end{cases}$$

時間発展演算子法では、リウビル演算子、あるいはそのもととなる運動方程式やハミルトニアンが与えられれば、機械的に数値計算法を作り出すことができる。



# 時間発展演算子法

- 時間発展演算子法により、重い(遅い)運動と軽い(速い)運動とを分離して数値積分する。

リウビル演算子  $L$

$$iL = iL_r + iL_{heavy} + iL_{light} = v \frac{\partial}{\partial r} + \frac{F_{heavy}}{m_{heavy}} \frac{\partial}{\partial v_{heavy}} + \frac{F_{light}}{m_{light}} \frac{\partial}{\partial v_{light}}$$

時間発展演算子

$$e^{iL\Delta t} q_i(\Delta t) = e^{iL\Delta t} q_i(0)$$

$$\begin{aligned} e^{iL\Delta t} &\approx e^{iL_{heavy}\Delta t/2} e^{i(L_r + L_{light})\Delta t} e^{iL_{heavy}\Delta t/2} \\ &\approx e^{iL_{heavy}\Delta t/2} \left\{ e^{iL_{light}\Delta t \delta t/2} e^{i(L_r)\delta t} e^{iL_{light}\delta t/2} \right\}^n e^{iL_{heavy}\Delta t/2} \end{aligned}$$

# シンプレクティック数値計算法



- ハミルトニアンがある系について時間発展演算子法を用いると、MD計算のハミルトニアンが長時間にわたって保存するという優れた性質が現れる
- 数値計算においてもハミルトニアン(影のハミルトニアン)が存在することによる
- このようなハミルトン系に対する数値計算をシンプレクティック数値計算法という

ハミルトニアンのあるもの  
フレキシブルモデル分子の運動  
剛体回転子の運動  
能勢・ポアンカレの方法  
アンダーソンの方法  
パリネロ・ラーマンの方法

$$H = T + V$$

$$i\mathbb{L}_H = -\{H, \quad \} \quad i\mathbb{L}_V = -\{V, \quad \} \quad i\mathbb{L}_T = -\{T, \quad \}$$

$$e^{i\mathbb{L}_H \Delta t} = e^{i\mathbb{L}_T \Delta t} e^{i\mathbb{L}_V \Delta t} + O(\Delta t^2)$$

$$e^{i\mathbb{L}_{\bar{H}} \Delta t} = e^{i\mathbb{L}_T \Delta t} e^{i\mathbb{L}_V \Delta t}$$

$$\bar{H} = H + \frac{\Delta t}{2} \frac{\partial H}{\partial q} \frac{\partial H}{\partial p} + \dots$$

# 高速化のために

遅い演算（除算と組み込み関数）

- 除算は他の四則演算より遅い。
- 組み込み関数も時間のかかる演算である。
- ホットスポット内ではできるだけこれらの演算を減らすように工夫する。

```
do i=1,n  
  b(i)=a(i)/x  
enddo
```

```
y=1.d0/x  
do i=1,n  
  b(i)=a(i)*y  
enddo
```

除算をループの外へ移動

```
do i=1,n  
  x(i)=a(i)/b(i)/c(i)  
enddo
```

```
do i=1,n  
  x(i)=a(i)/(b(i)*c(i))  
enddo
```

除算を乗算に変更

```
do i=1,n  
  x(i)=a(i)/b(i)  
  y(i)=c(i)/d(i)  
enddo
```

```
do i=1,n  
  tmp=1.d0/(b(i)*d(i))  
  x(i)=a(i)*d(i)*tmp  
  y(i)=c(i)*b(i)*tmp  
enddo
```

通分の利用

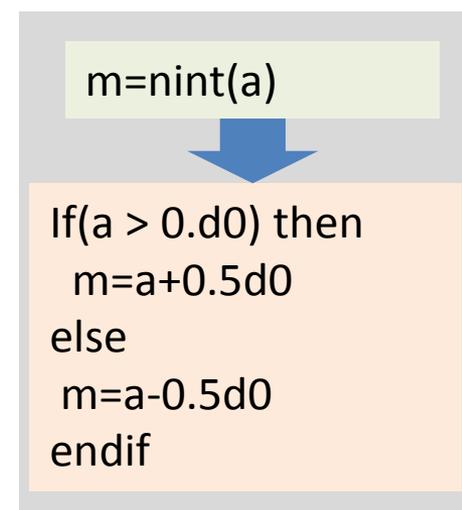
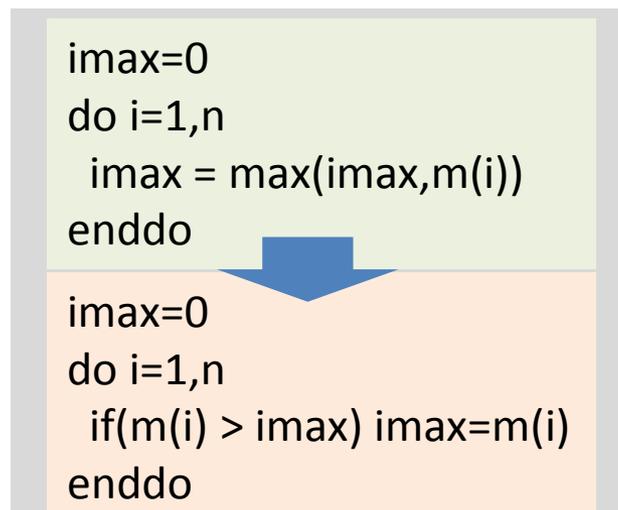
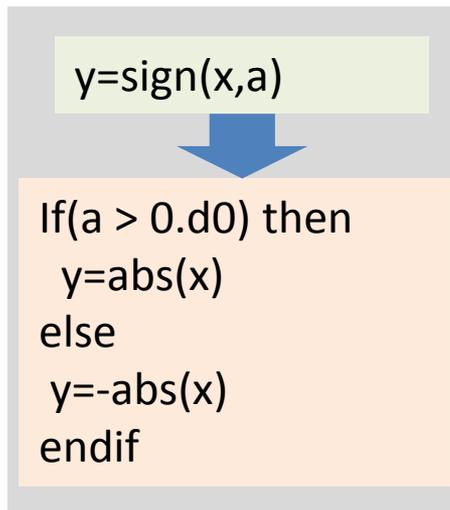


# 高速化のために

## 遅い演算(組み込み関数)

組み込み関数のfortranコード化 環境によって速くなる場合とそうでない場合がある。両者の比較が必要。

- $e^x \times e^y \longrightarrow e^{x+y}$
- $\log_a x + \log_a y \longrightarrow \log_a xy$
- $\sin\theta \times \cos\theta \longrightarrow 0.5\sin 2\theta$
- $\text{If}(\text{sqrt}(x^*x+y^*y) < R) \longrightarrow \text{If}(x^*x+y^*y < R^2)$



# 高速化のために

## 分子シミュレーションへの適用

- Lennard-Jones相互作用

$$V = \left(\frac{1}{r}\right)^{12} - \left(\frac{1}{r}\right)^6$$

$$r^6 = r^2 * r^2 * r^2$$

$$r^{12} = r^6 * r^6$$

$$V = 1.d0/r^{12} - 1.d0/r^6$$



$$r^6 = r^2 * r^2 * r^2$$

$$r^{12} = r^6 * r^6$$

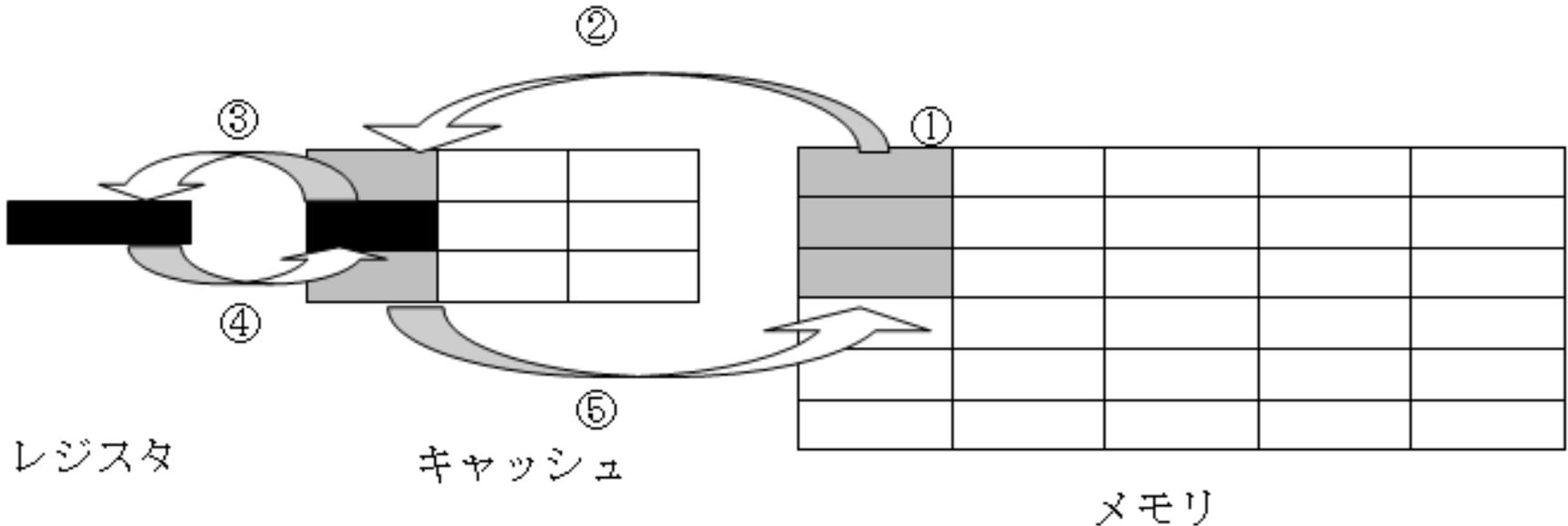
$$V = (1.d0 - 1.d0 * r^6) / r^{12}$$

LJ相互作用への適用

# キャッシュの有効利用

CPUの必要なデータに高速アクセス

- ① データはメモリに格納
- ② 必要に応じて隣接するデータと共にキャッシュ上に
- ③ 演算で必要となるとレジスタにコピー
- ④ 演算された結果は再びキャッシュに上書き
- ⑤ キャッシュ上で用いられなくなるとメモリ上に上書きコピーされる。



# キャッシュミス

- データ: レジスタ ← キャッシュ ← メモリ
- レジスタ ← キャッシュ: 高速アクセス
- キャッシュ ← メモリ: ← 低速アクセス
- 演算を行う時にデータがキャッシュ上にないときはメモリまで取りに行く。キャッシュミス。
- データがキャッシュに乗っているように工夫することが重要。



# キャッシュミスを防ぐ

- 1次元配列の変数: 引数の順にキャッシュに格納
- 1次元配列の変数へのアクセス→配列の引数について連続的にアクセスするように心がける。
- 2次元以上の配列の変数: Fortranでは配列左側の引数の順にキャッシュに格納。  
(C言語は右側から)
- 2次元以上の配列の変数へのアクセス→Fortranでは配列の左側の引数について連続的にアクセスするように心がける。A(l,j)  
(C言語は右側)

# 参考文献



## MD全般

- D. Frenkel, B. Smit, "Understanding Molecular Simulation," Academic Press, San Diego (1996).
- 上田 顕, 「コンピュータシミュレーション」, 朝倉書店(1990).
- M. P. Allen, D. J. Tildesley, "Computer Simulation of Liquids," Oxford Science, Oxford (1987).
- J. P. Hansen, I. R. McDonald, "Theory of Simple Liquids," Academic Press, London (1986).
- 岡崎 進, 吉井 範行, コンピュータ・シミュレーションの基礎(第2版), 化学同人(2011).

## 力学

- H. Goldstein, 「ゴールドスタイン新版古典力学(上), (下)」(瀬川富士, 矢野 忠, 江沢康生 訳), 吉岡書店(1983).

## 数値積分

- G. J. Martyna, M. E. Tuckerman, D. J. Tobias, M. L. Klein, Mol. Phys., 87, 1117(1996).

## Particle mesh Ewald法

- T. Darden, D. York, L. Pedersen, J. Chem. Phys., 98, 10089(1993).

## FMM

- L. Greengard, V. Rokhlin, J. Comput. Phys., 73, 325(1987).

## SHAKE

- G. Ciccotti, J. P. Ryckaert, Comp. Phys. Rep., 4, 345(1986).
- P. Gonnet, J. Comput. Phys. 220, 740(2007).

# まとめ



## MD計算とは

- MD計算の流れ
- 取り扱う原子数と周期境界条件
- 相互作用(力)
- 分子モデルと運動方程式
- MD計算の例
- MD計算の歴史
- MD計算の現状

## 非並列のMD計算

- 短距離相互作用(bookkeeping, cell index法)
- 長距離相互作用(クーロン相互作用) Ewald, PME法
- 効率化のための方法
  - 拘束動力学
  - On the flyによる最適化
  - Multiple time step  
(時間発展演算子法、シンプレクティック数値計算法)
- 高速化のために、キャッシュの有効利用、キャッシュミス

次週は並列環境下における高速化について解説する