

GPUによる3D-RISMとレプリカ 交換法を結合したタンパク質の シミュレーションシステムの開発

丸山 豊、光武 亜代理 (慶應義塾大学物理)

1. レプリカ交換MD/3D-RISM法
2. GPU用3D-RISMプログラム
の並列化

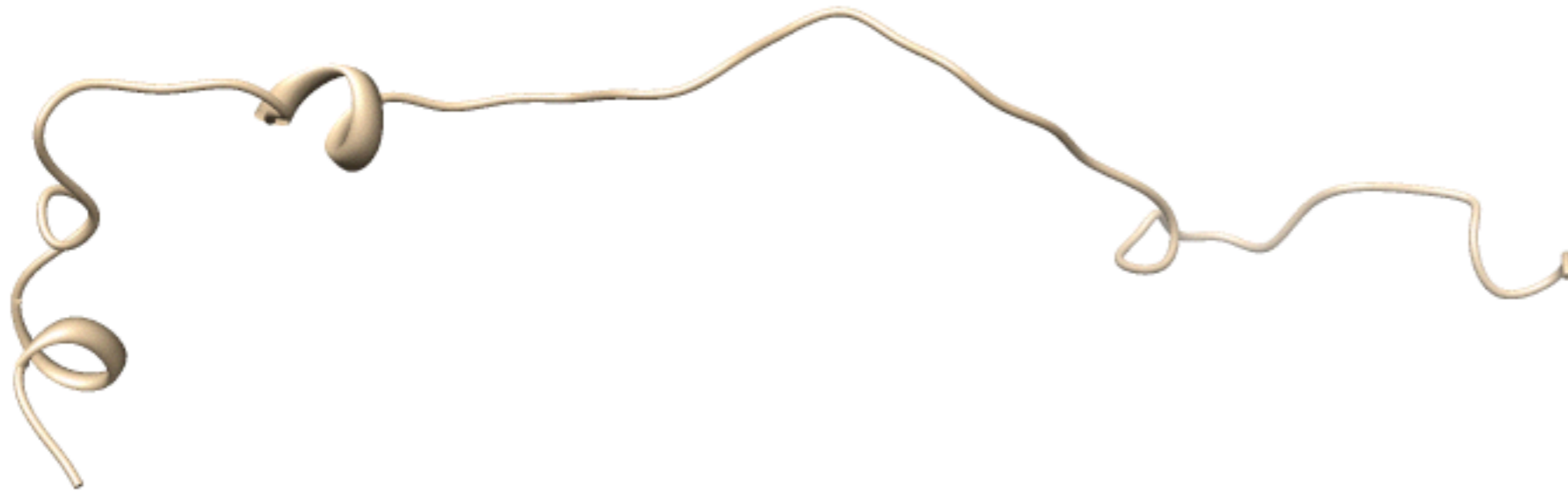
Contents

1. レプリカ交換MD/3D-RISM法

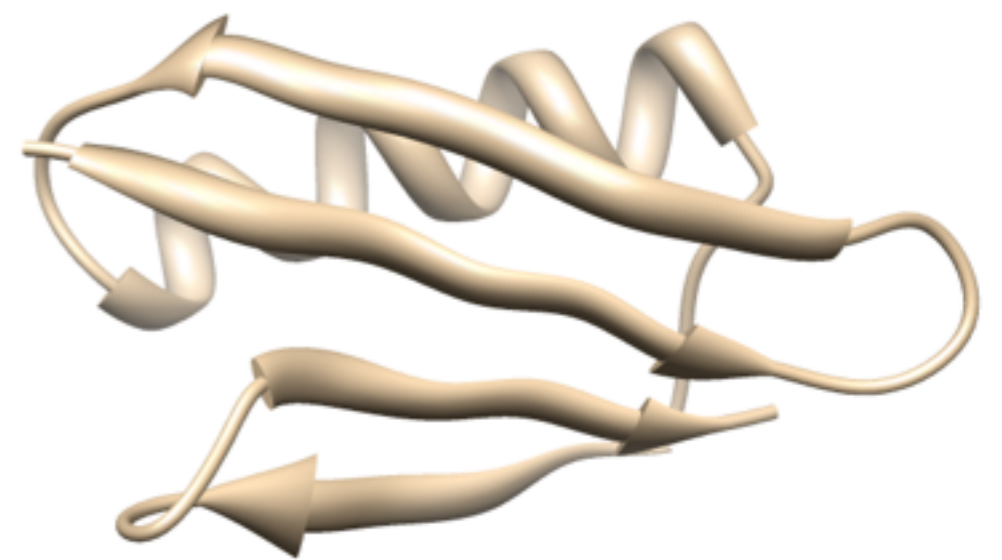
2. GPU用3D-RISMプログラム

の並列化

フォールディング問題



- ・ 20種類のアミノ酸が鎖状に重合
- ・ 室温水中で折り畳まれた構造を取る



分子動力学法

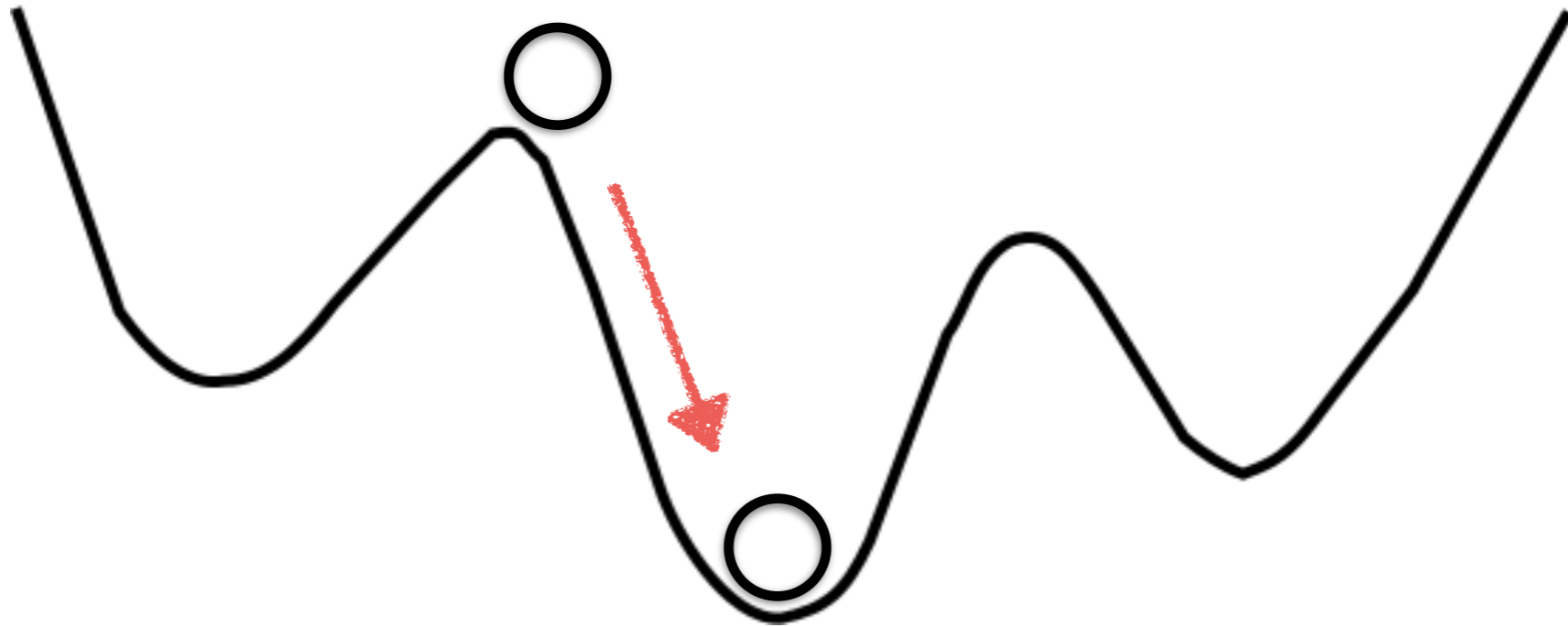
タンパク質は自発的にフォールドする。

フォールディングの時間スケール: 秒

Anton: ミリ秒

構造探索の問題点

自由エネルギー曲面

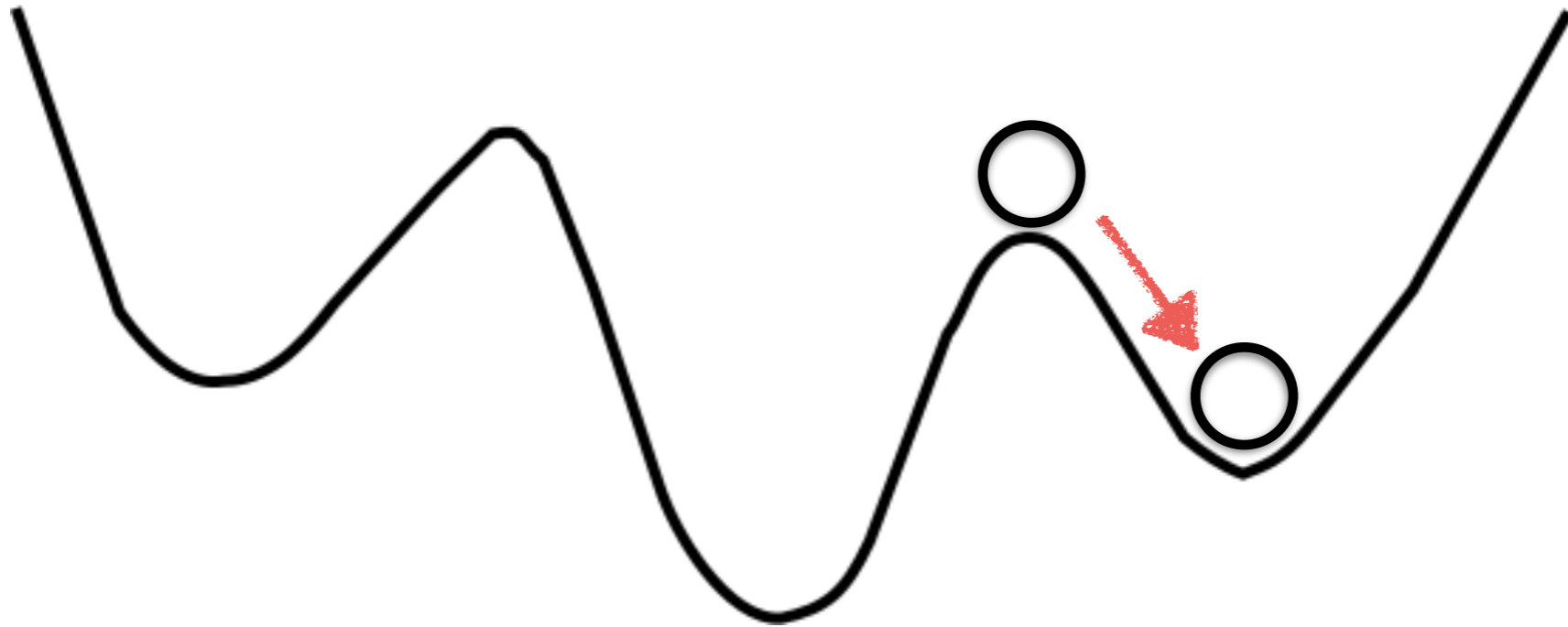


天然構造

出発点から天然構造に必ず到達？

構造探索の問題点

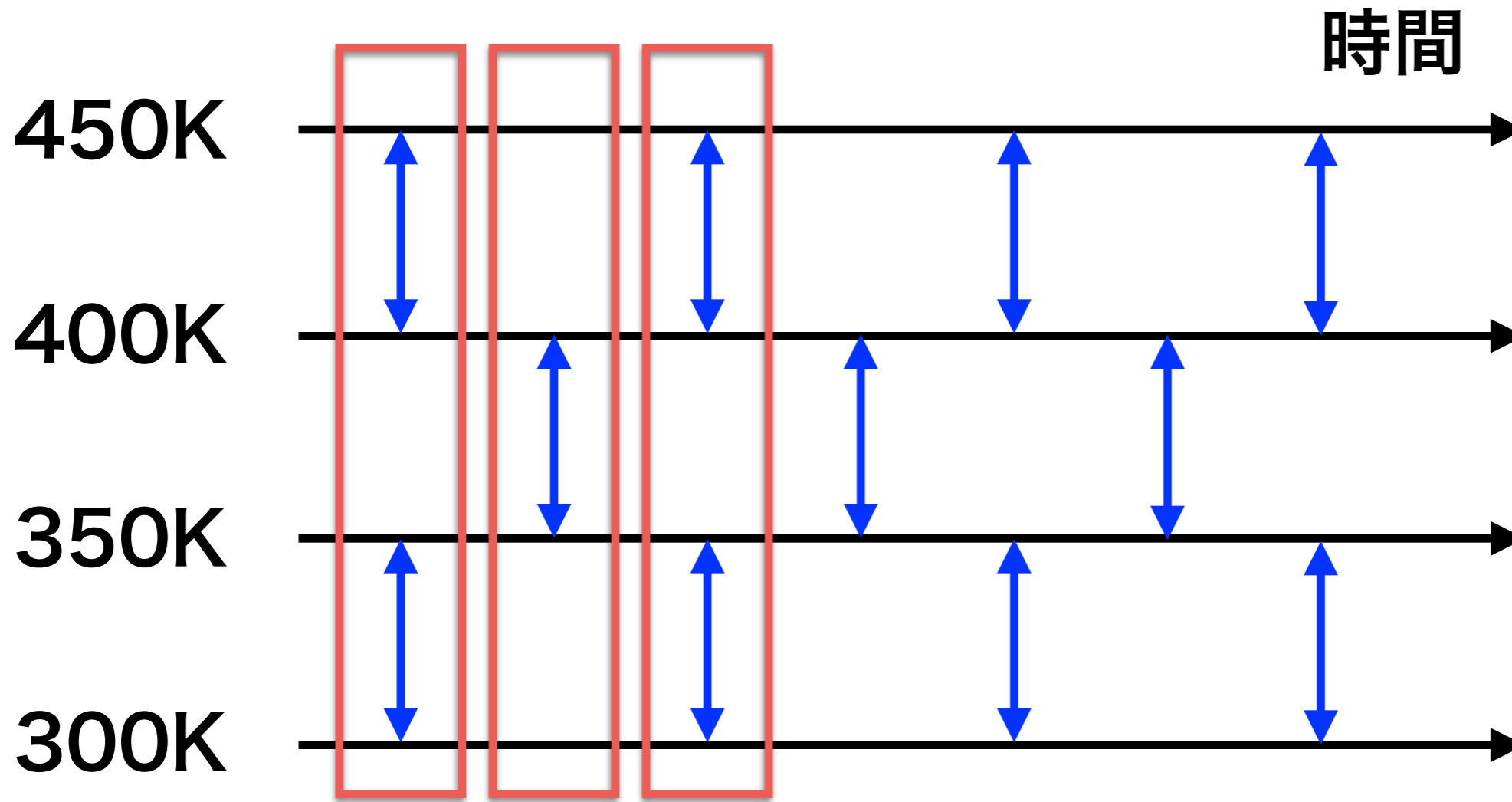
自由エネルギー曲面



天然構造

出発点から天然構造に必ず到達

レプリカ交換法

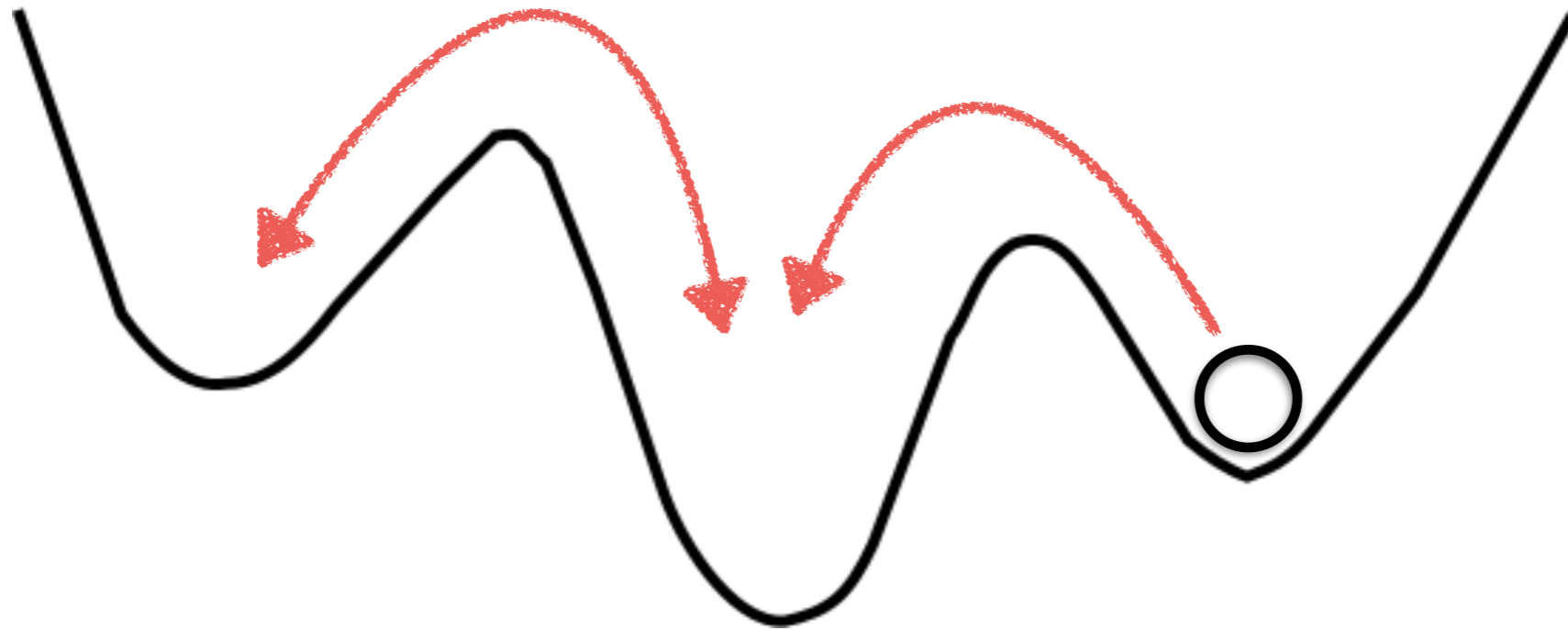


一定間隔でレプリカ間で交換を行なう。

エネルギーのメトロポリスで交換の判定をする。

構造探索の問題点

自由エネルギー曲面



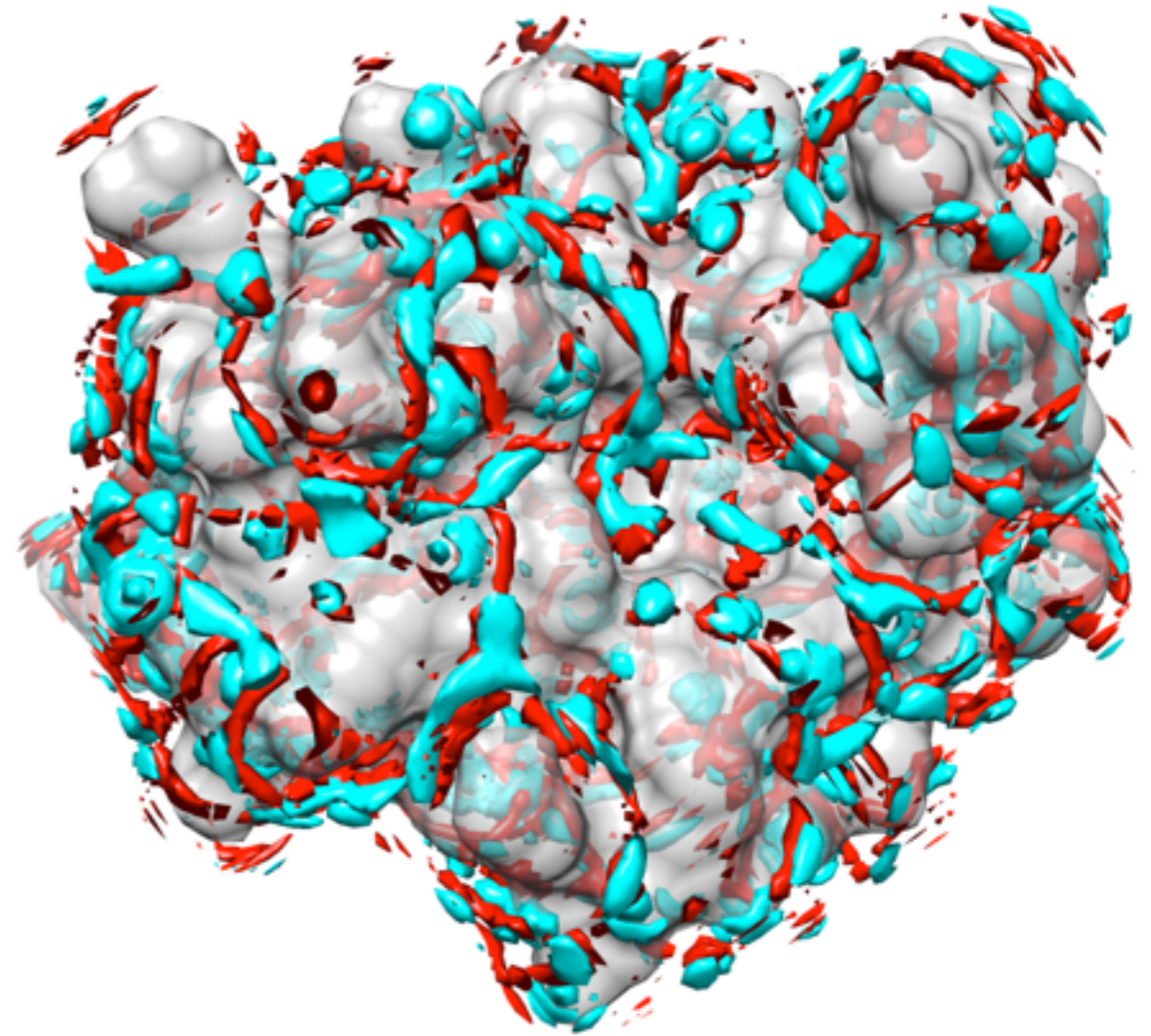
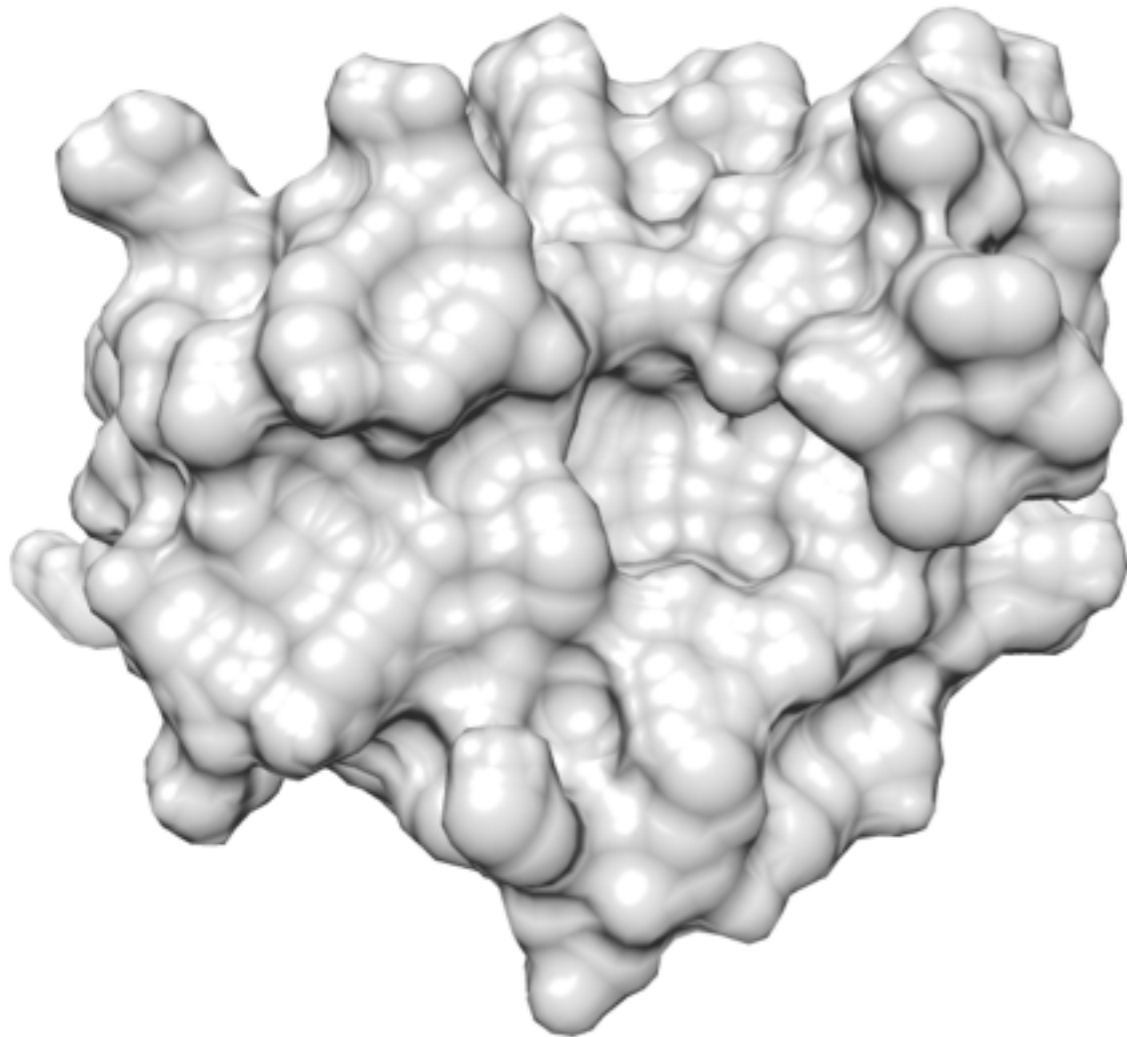
天然構造

レプリカ交換法によって局所構造から脱出して
天然構造を探索する事が可能

3D-RISM理論

水の酸素

水の水素



水素結合などの分子的特徴を記述
統計力学的平均

分子液体の理論

3D-RISM方程式

$$\tilde{h}_\gamma^{Ub}(\mathbf{r}) = \tilde{c}_\alpha^{Ua}(\mathbf{r}) * \left[\tilde{\omega}_{\alpha\gamma}^{ab}(r) + \rho^a \tilde{h}_{\alpha\gamma}^{ab}(r) \right]$$

フーリエ空間

クロージャ方程式

$$h_\gamma^{Ub}(r) = \begin{cases} \exp(\chi) - 1 & \text{for } \chi \leq 0 \\ \chi & \text{for } \chi > 0 \end{cases}$$

$$\chi = -\beta u_\gamma^{Ub}(r) + h_\gamma^{Ub}(r) - c_\gamma^{Ub}(r).$$

実空間

3D-FFT

水和自由エネルギー

$$\Delta\mu = \rho k_B T \sum_\gamma \int_{V_{all}} dr \left[\frac{1}{2} (h_\gamma(\mathbf{r}))^2 \Theta(-h_\gamma(\mathbf{r})) - c_\gamma(\mathbf{r}) - \frac{1}{2} h_\gamma(\mathbf{r}) c_\gamma(\mathbf{r}) \right]$$

MD/3D-RISM

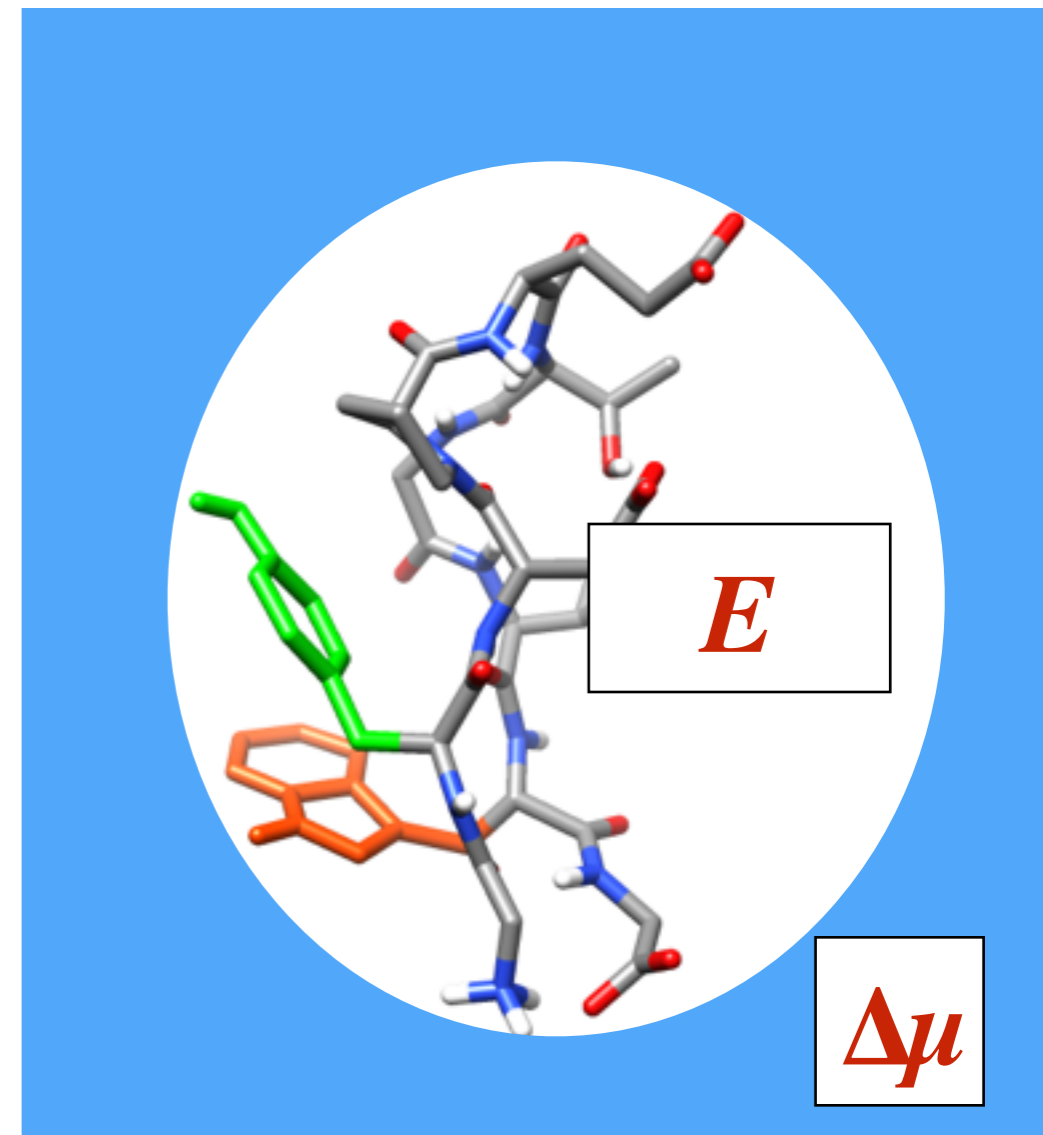
Force

$$\frac{\partial F_{tot}}{\partial \mathbf{R}_a} = \frac{\partial E_{conf}}{\partial \mathbf{R}_a} + \sum_{\gamma} \rho_{\gamma} \int d\mathbf{r} \frac{\partial u_{\gamma}^{uv}(\mathbf{r})}{\partial \mathbf{R}_a} g_{\gamma}^{uv}(\mathbf{r})$$

MD

+

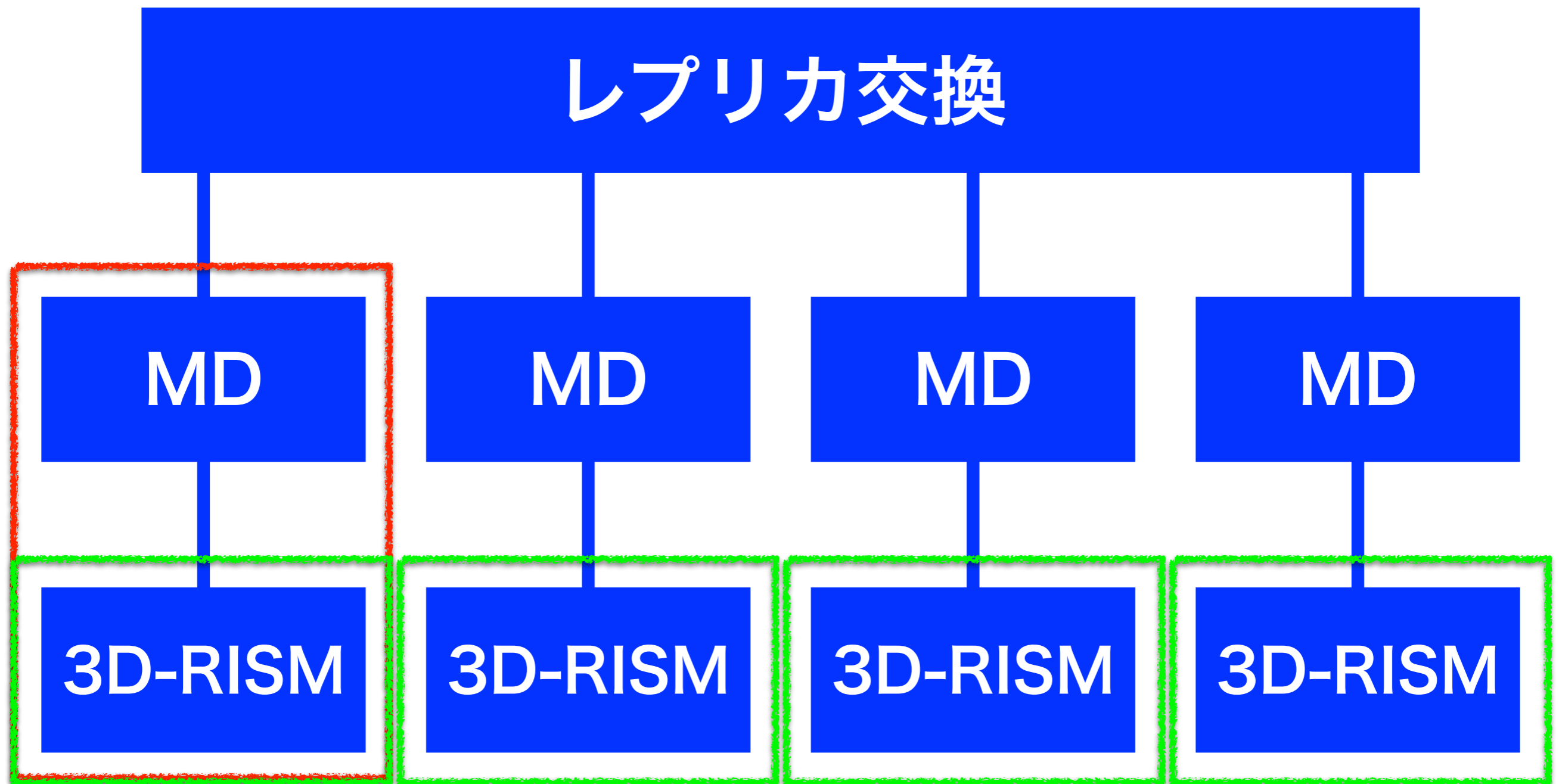
3D-RISM



$$F = E + \Delta\mu \quad \text{自由エネルギー}$$

分子動力学法と3D-RISM理論を結合

レプリカ交換MD/3D-RISM



レプリカ

GPU

レプリカ交換MD/3D-RISM法の特徴

1. 溶媒和構造の算出

タンパク質の構造によって一意に決まる。

自由エネルギー曲面を滑らかにする。

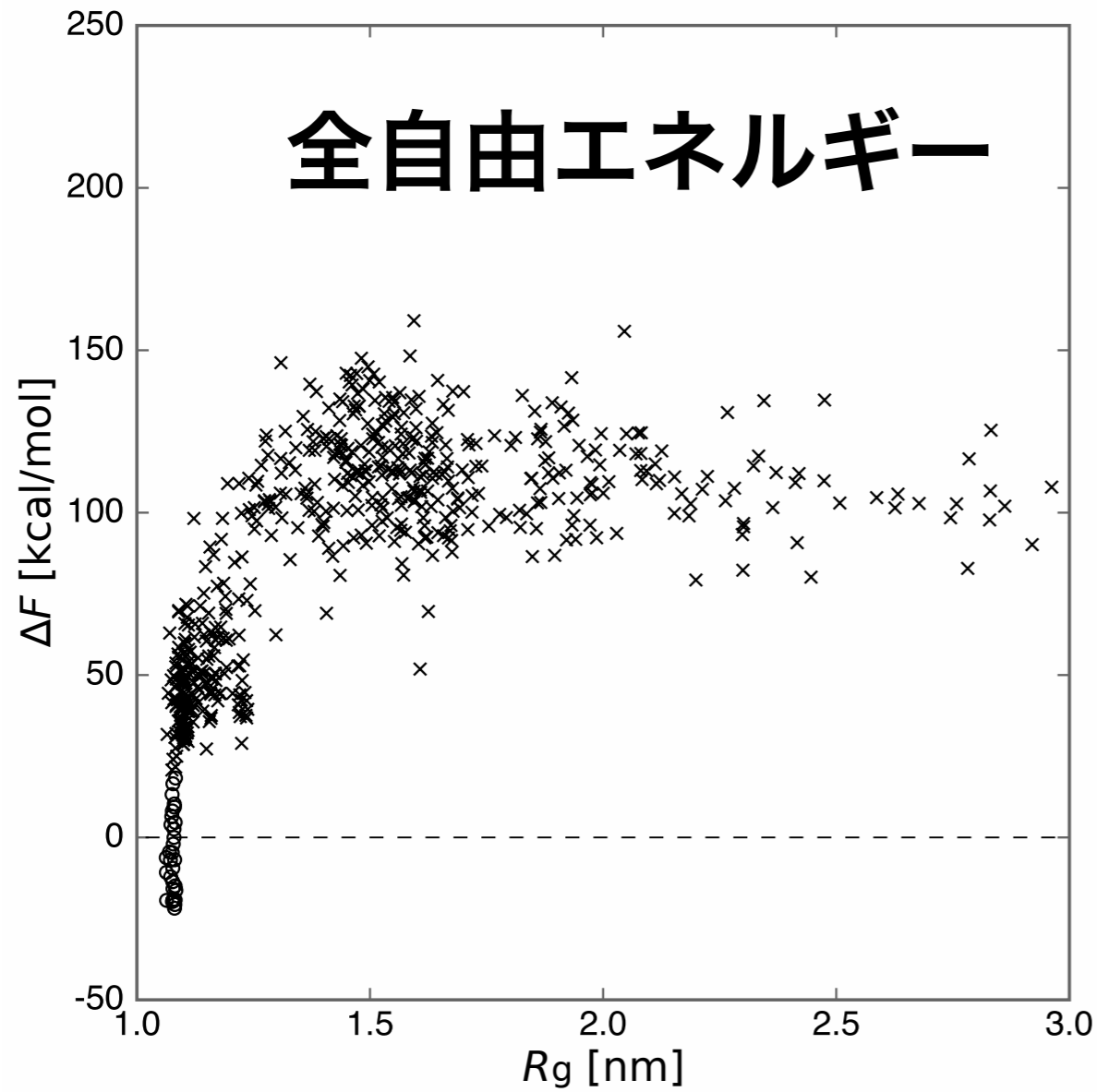
2. レプリカ数を減らす事ができる。

自由エネルギーでメトロポリスを行なう。

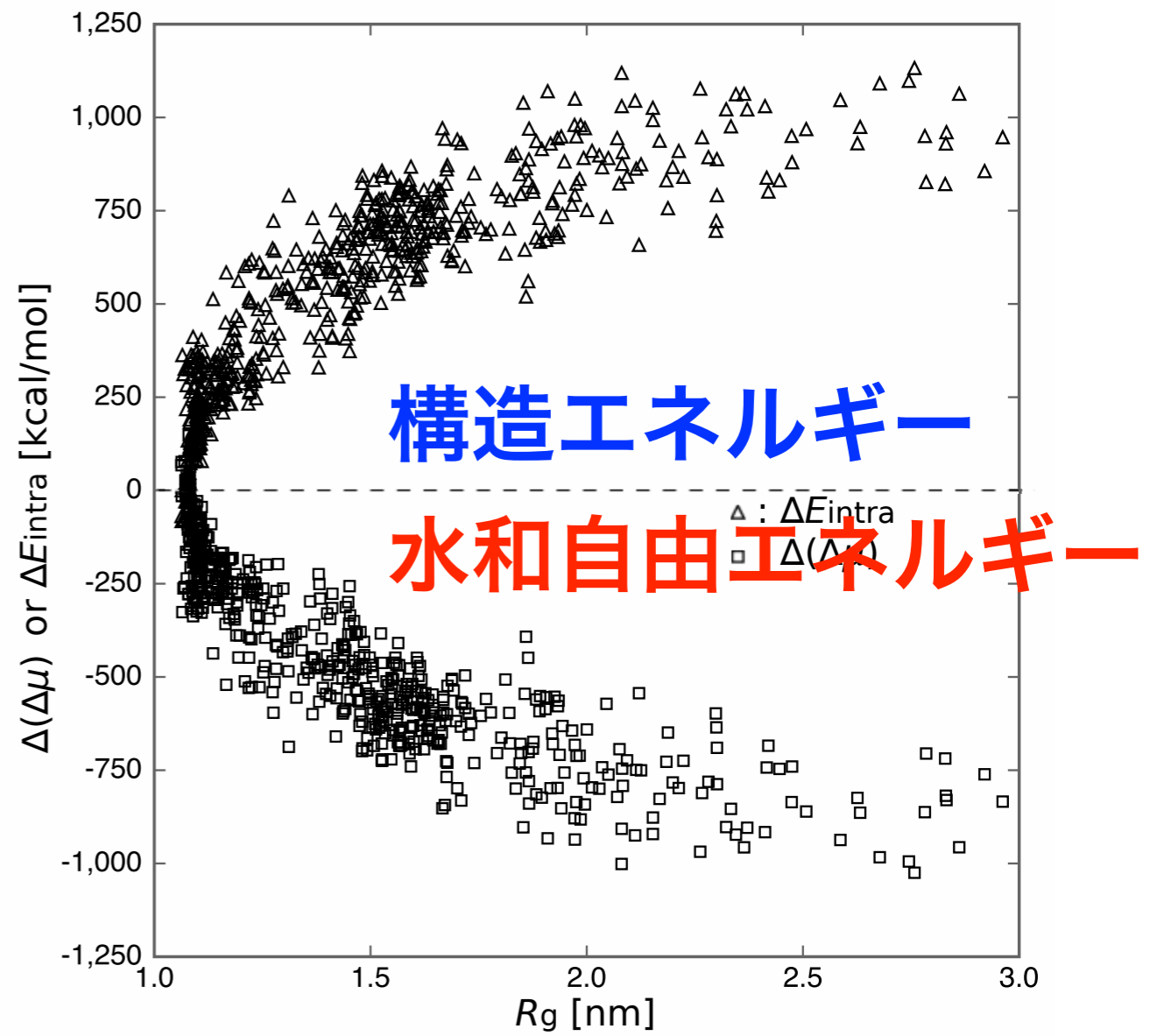
3. 3D-RISM部分が計算の大半を占める。

3D-RISMの高速化が重要

タンパク質のエネルギー曲面



300 kcal/mol



2500 kcal/mol

レプリカ交換MD/3D-RISM法の特徴

1. 溶媒和構造の算出

タンパク質の構造によって一意に決まる。

自由エネルギー曲面を滑らかにする。

2. レプリカ数を減らす事ができる。

自由エネルギーでメトロポリスを行なう。

3. 3D-RISM部分が計算の大半を占める。

3D-RISMの高速化が重要

パラメータの決定

2つのパラメータ

1. レプリカの温度分布

事前の計算(MD/3D-RISM)で対応可能

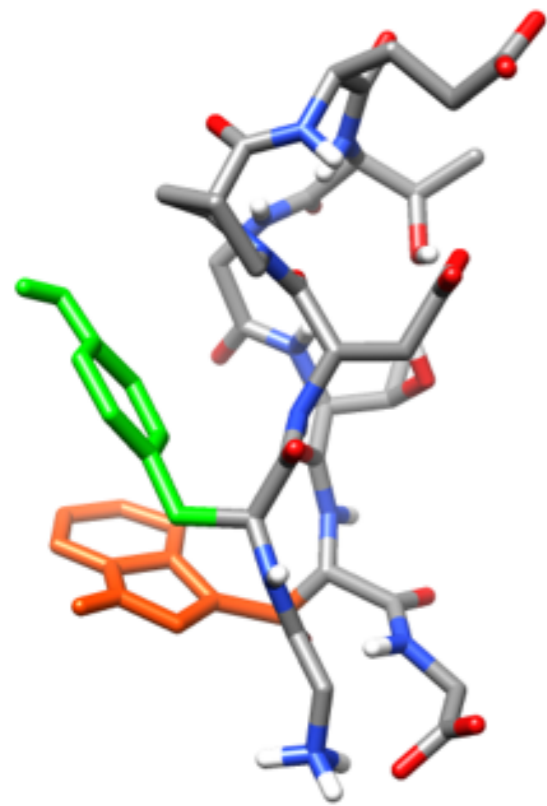
2. レプリカ交換の間隔

実際に計算しないとわからない。

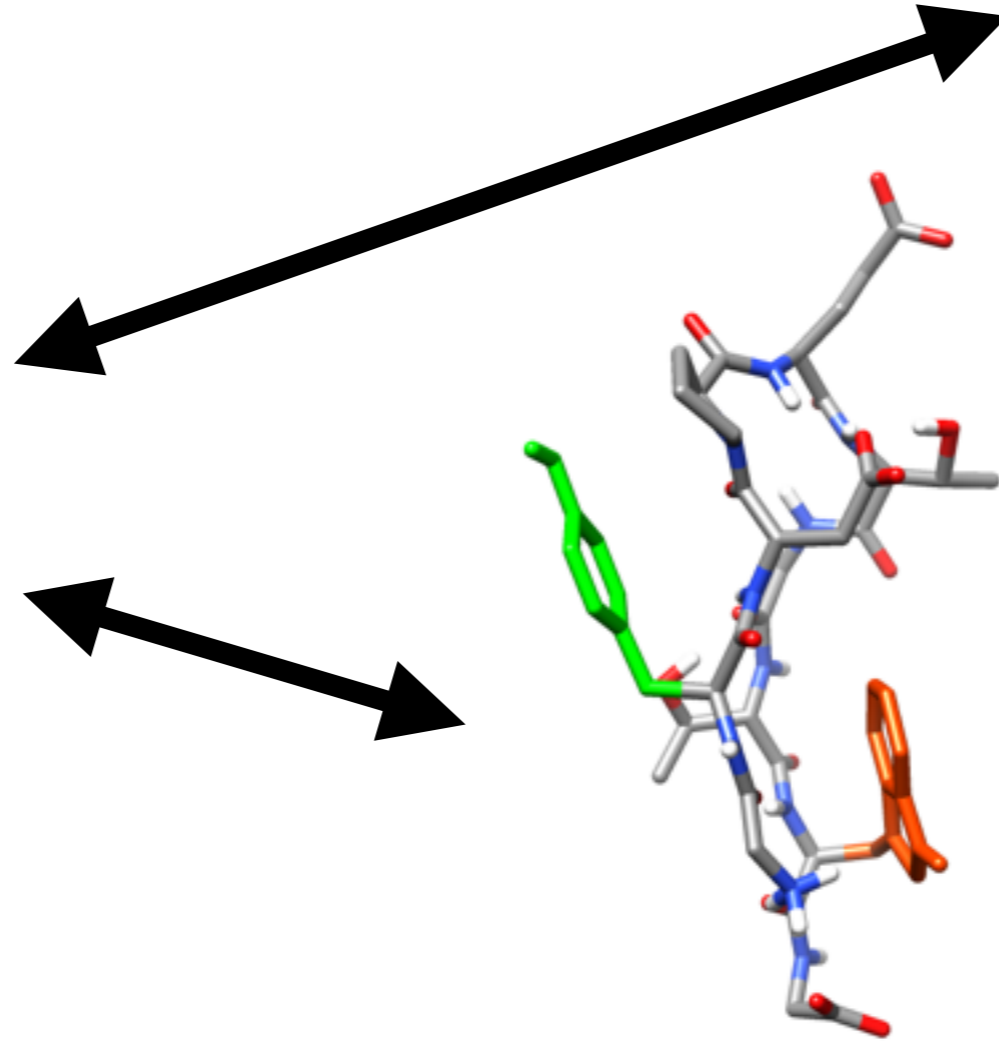
(MD/3D-RISMの時間スケールが通常のMDと違うため)

2つのパラメータによってレプリカ交換の頻度が決まる。

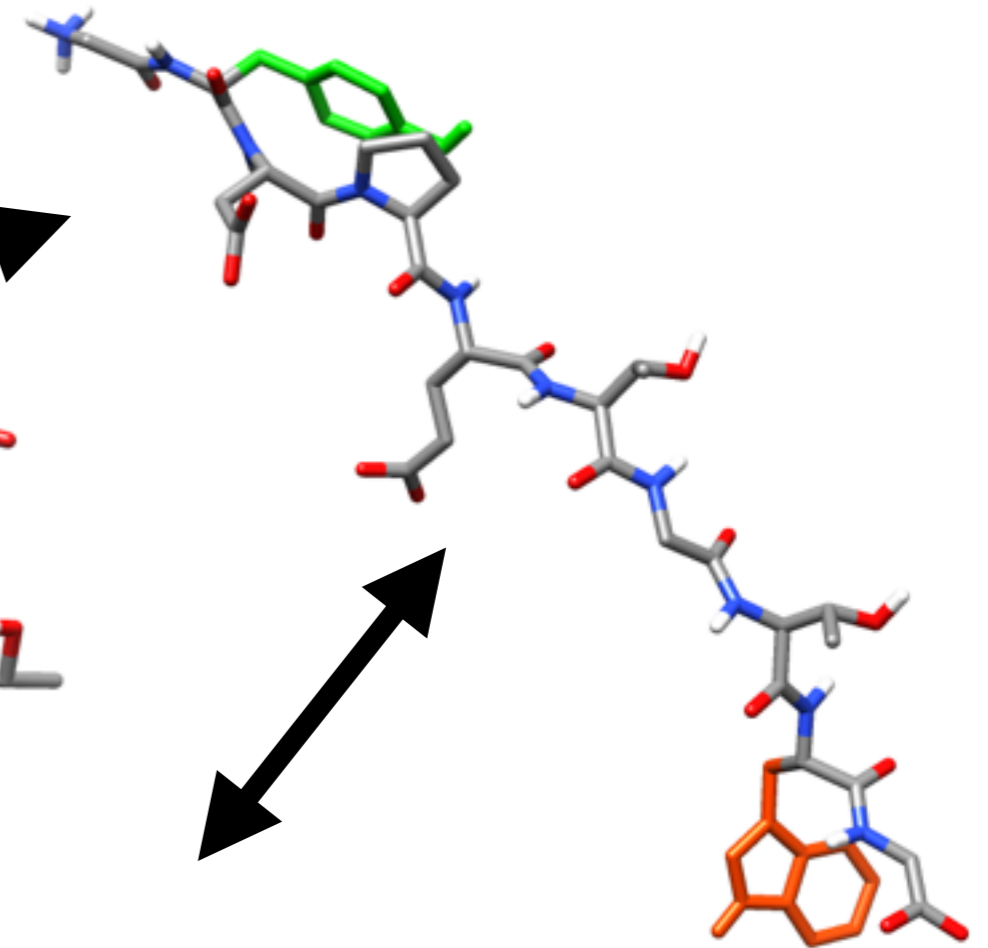
Chignolin



native



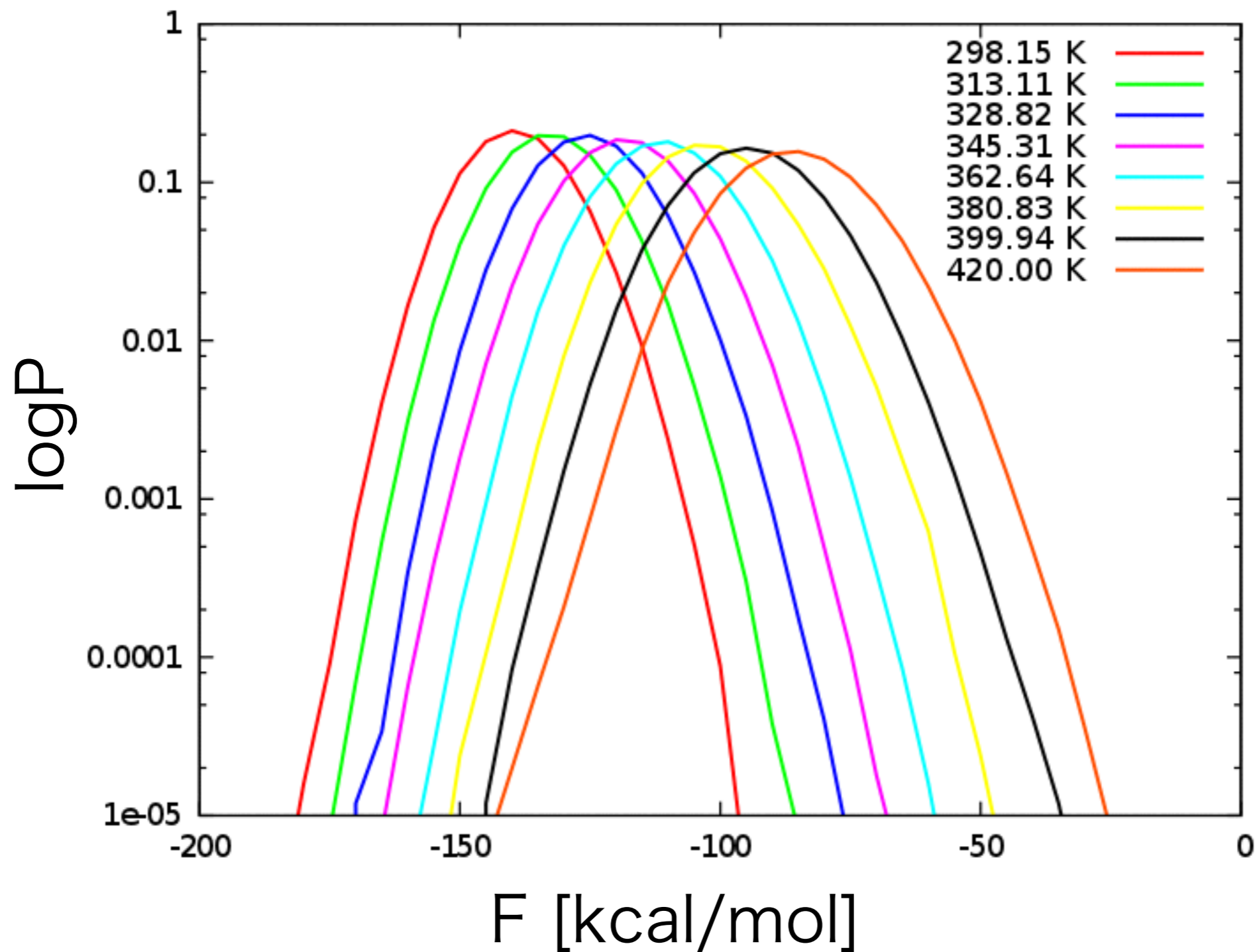
misfolded



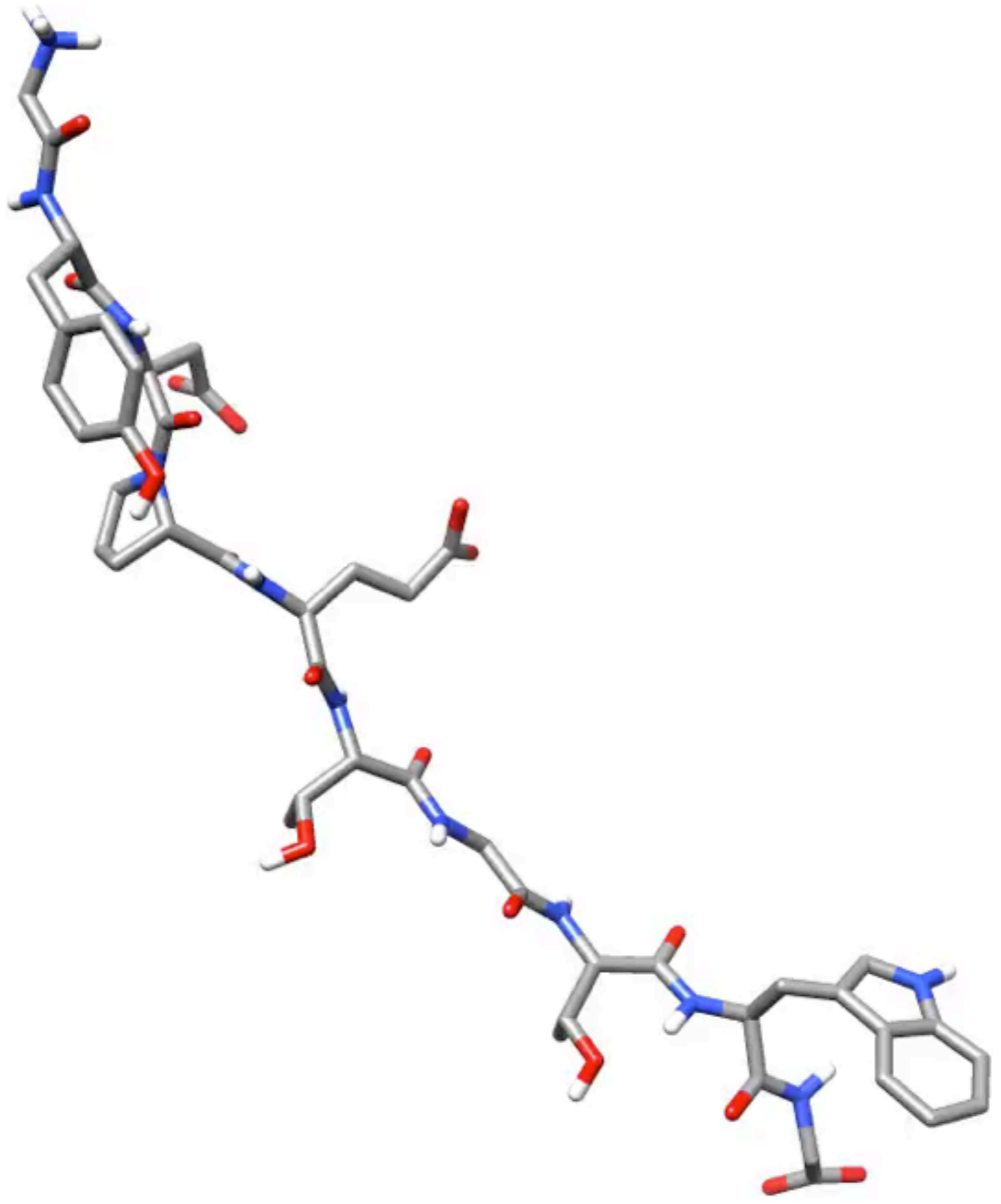
unfolded

アミノ酸10残基のミニ蛋白質
常温の水中で β ターンを形成

自由エネルギー分布

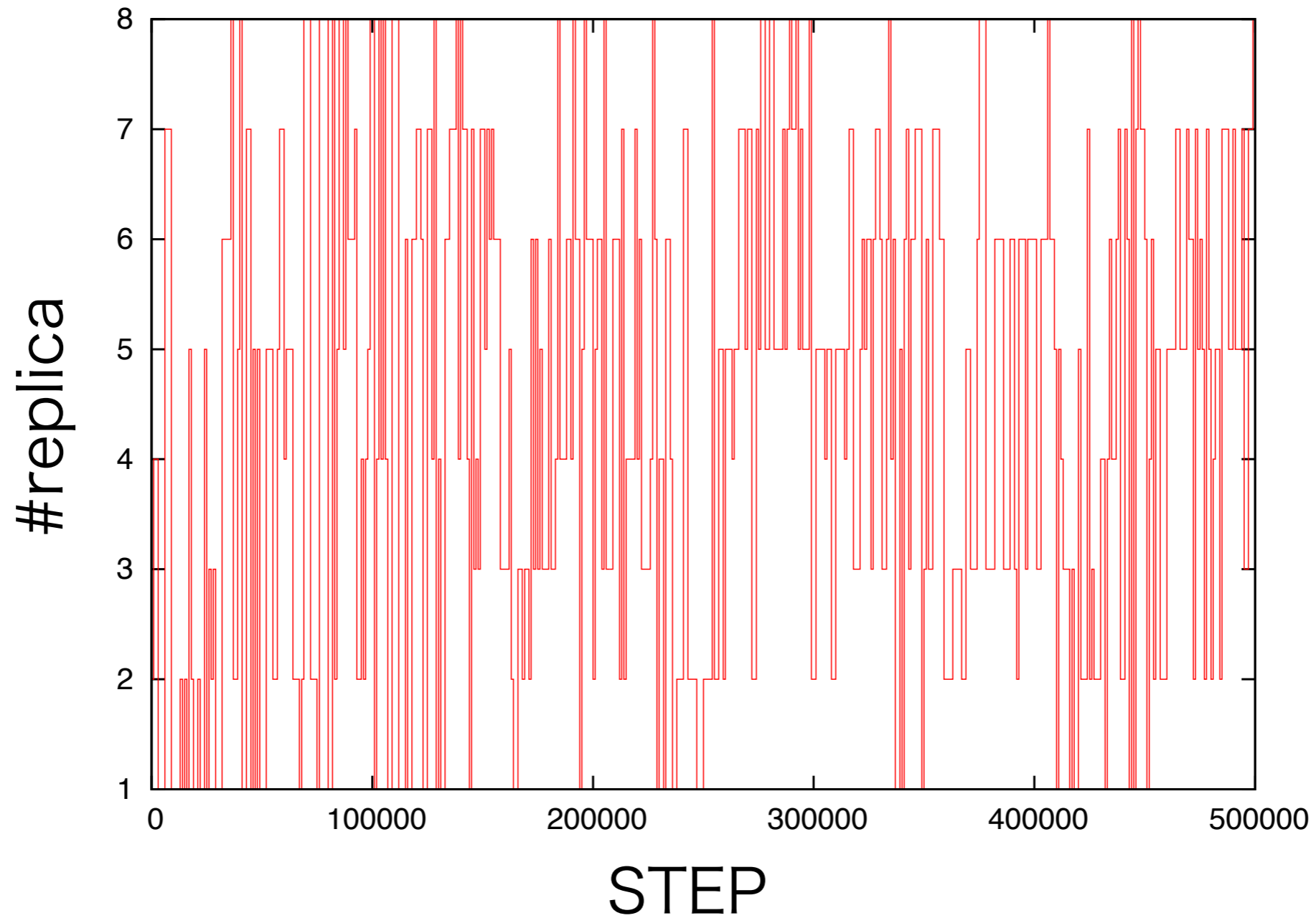


交換には各温度のオーバーラップが必要



レプリカの変遷

レプリカ交換頻度: 1 ps



1 ns相当、14回のランダムウォーク

まとめ(1)

レプリカ交換法とMD/3D-RISMを結合した。Chignolinでテスト計算を行い、動作確認を終えた。

展望

より大きなタンパク質への適用
タンパク質と分子の結合問題

Contents

1. レプリカ交換MD/3D-RISM法

2. GPU用3D-RISMプログラム

の並列化

3D-RISM理論

3D-RISM方程式

$$h_{\alpha}(\mathbf{r}) = \sum_{\gamma} \int d\mathbf{r}' c_{\gamma}(\mathbf{r} - \mathbf{r}') \chi_{\gamma\alpha}(\mathbf{r}')$$

↓ 畳み込み積分

$$h_{\alpha}(\mathbf{k}) = \sum_{\gamma} c_{\gamma}(\mathbf{k}) \chi_{\gamma\alpha}(\mathbf{k})$$

Closure方程式

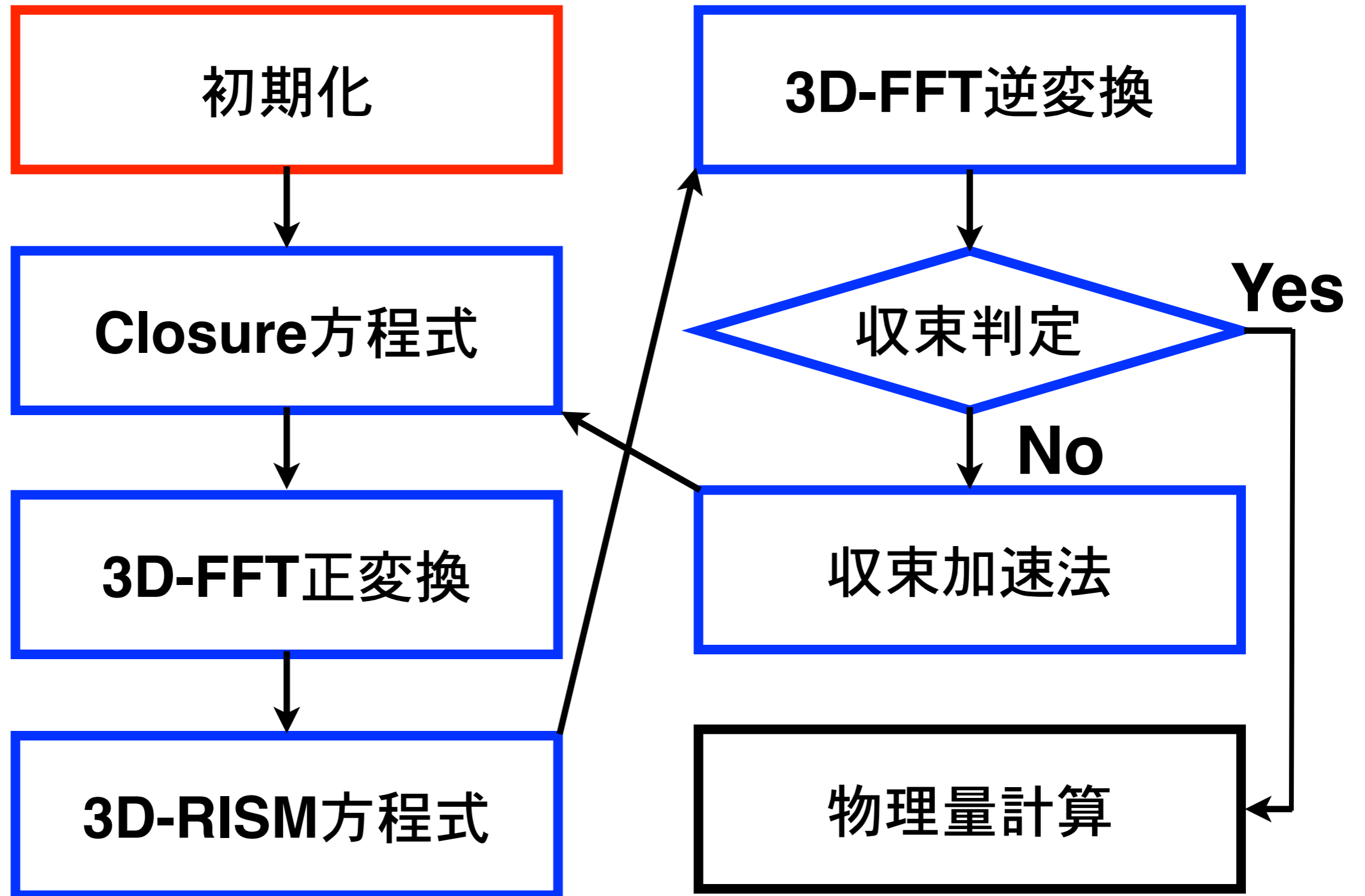
↑ 3D-FFT

$$h_{\gamma}(r) = \begin{cases} \exp(\chi) - 1 & \text{for } \chi \leq 0 \\ \chi & \text{for } \chi > 0 \end{cases}$$

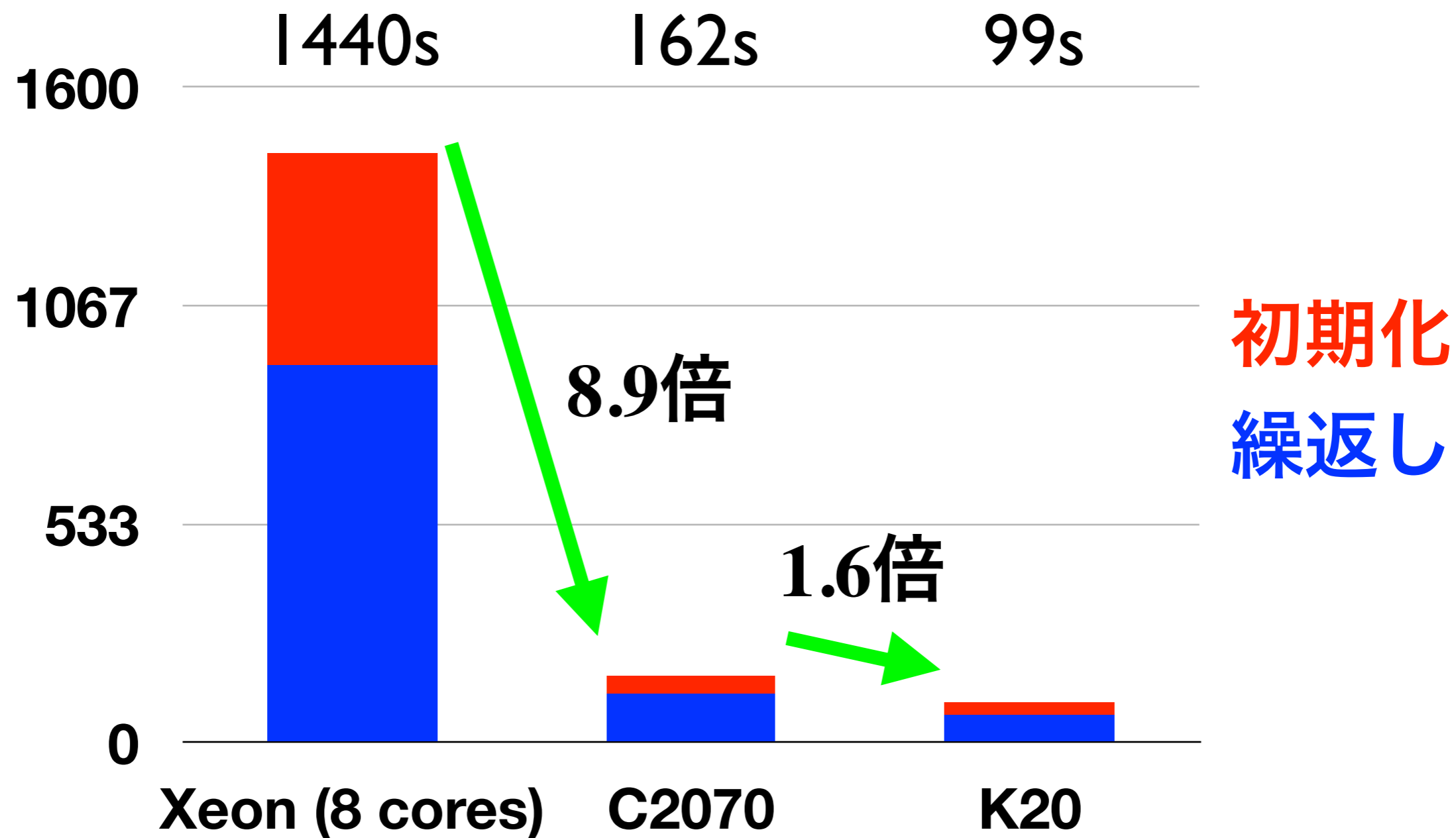
$$\chi = -\beta u_{\gamma}(r) + h_{\gamma}(r) - c_{\gamma}(r).$$

3D-RISM方程式とClosure方程式を交互に計算

解法の流れ



計算時間 (FKBPの周りの水)



Xeon E5-2650 2.5GHz, gcc 4.6.1, FFTW 3.2.2, CUDA5.0

計算時間の割合(%)

	C2070	K20
初期化	22.5	23.5
3D-RISM	31.6	36.9
収束加速法	12	14.7
FFT(cufft)	32.7	22.9
Data転送	0.5	1.2

並列化の必要性

1. GPUのメモリ容量の制限

256³グリッドで約5GiB

イオン水溶液 (メモリが倍必要)

膜タンパク質 (512³グリッド)

2. さらなる高速化

レプリカ交換MD/3D-RISM

3D-RISMプログラムの並列化

データ通信はFFTに集約

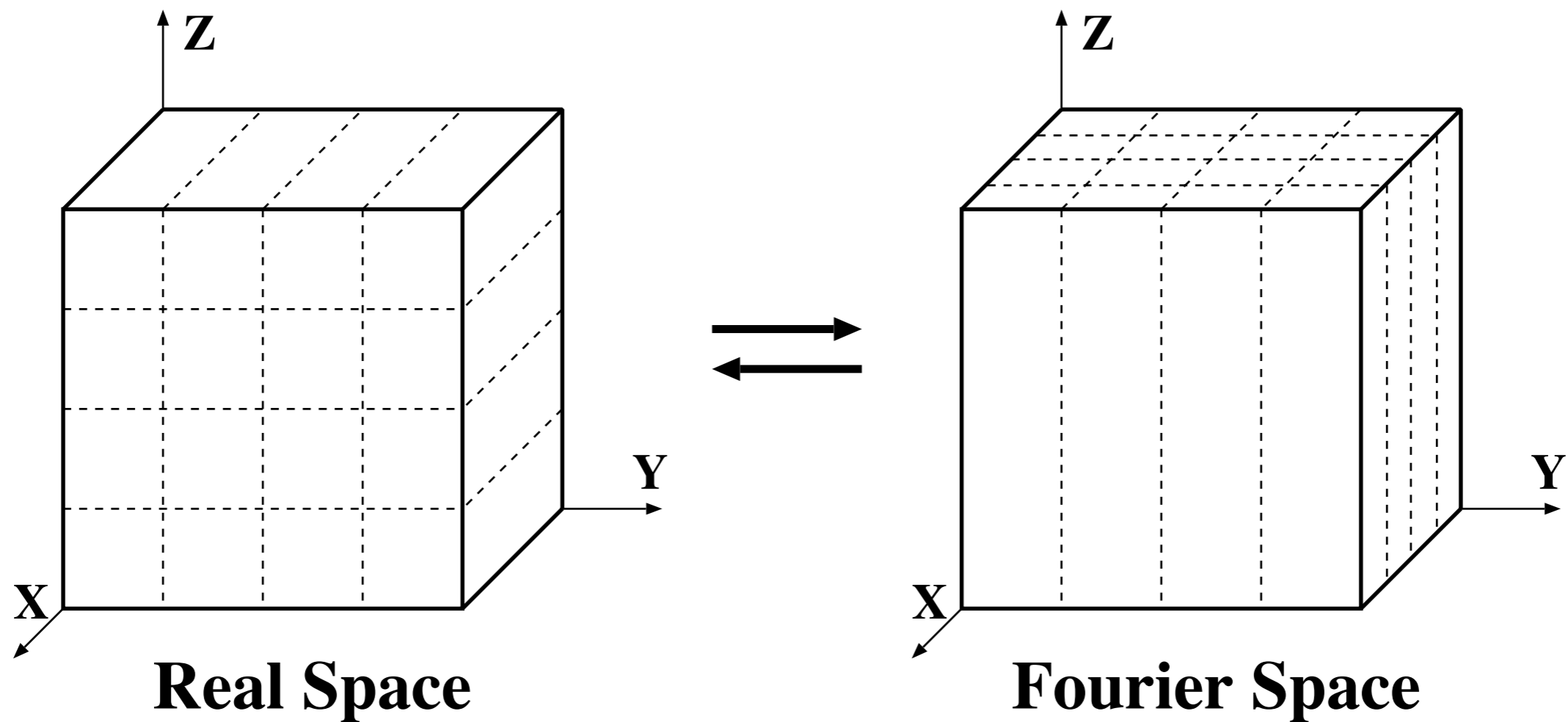
1. ノード内Multi-GPU化

QPIが問題 → NVLink待ち

2. ノード間MPI-Multi-GPU化

セルの分割が問題

3次元セルの分割



実空間でYZ分割、フーリエ空間でXY分割にして3D-FFTの通信量を半減

Volumetric 3D-FFT

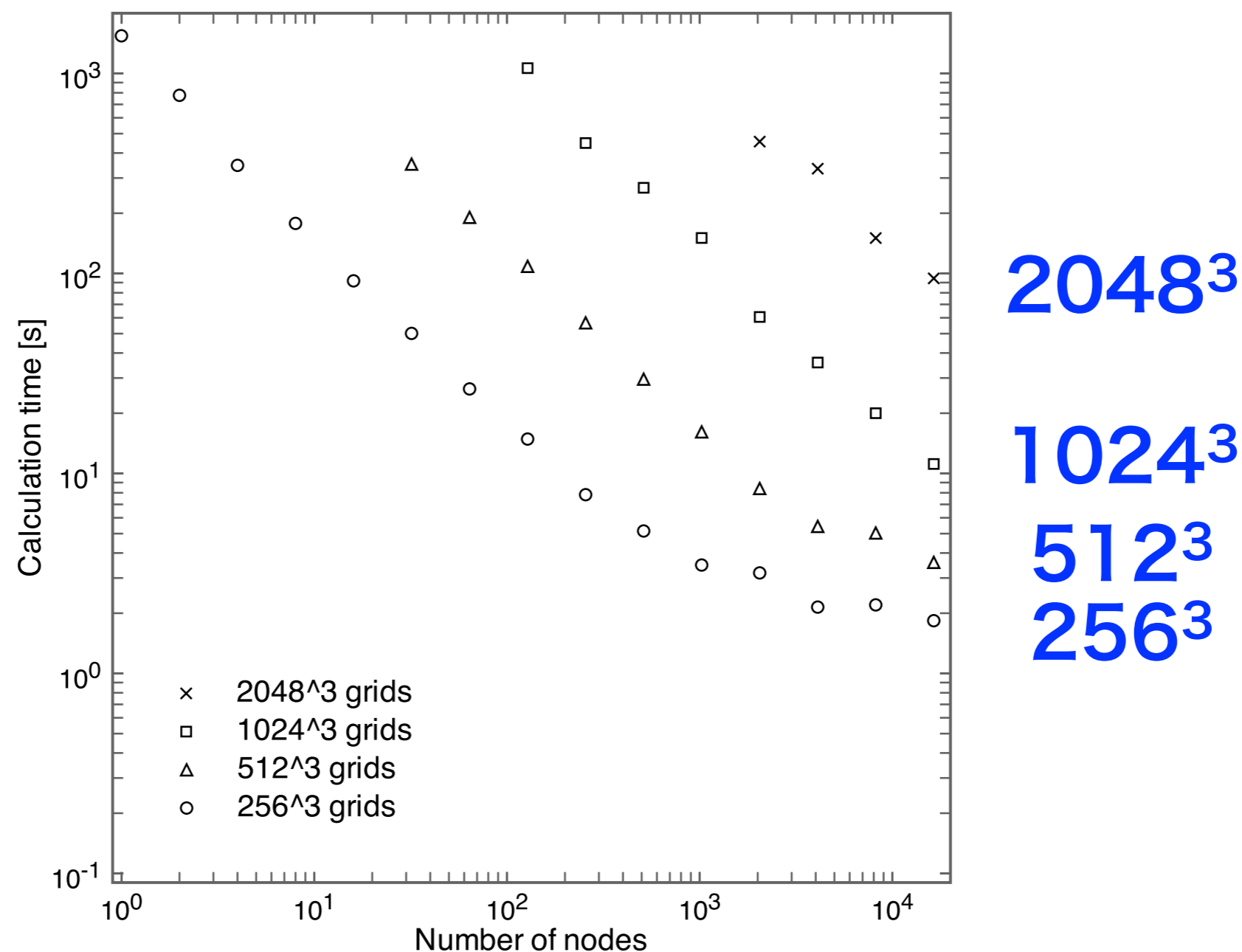
YZ分割の3D-FFTの手順

1. X軸のFFTを実行
2. Y軸のデータを揃える通信
3. Y軸のFFTを実行
4. Z軸のデータを揃える通信
5. Z軸のFFTを実行
6. Z軸のデータを戻す通信
7. Y軸のデータを戻す通信

← 実行しない

3D-RISMプログラムの京対応

筑波大学佐藤グループとの共同研究



計算セルサイズ毎の計算時間のノード数依存性

2次元分割FFTライブラリ

FFTE for GPU(筑波大、高橋教授)

CUDA Fortranで実装

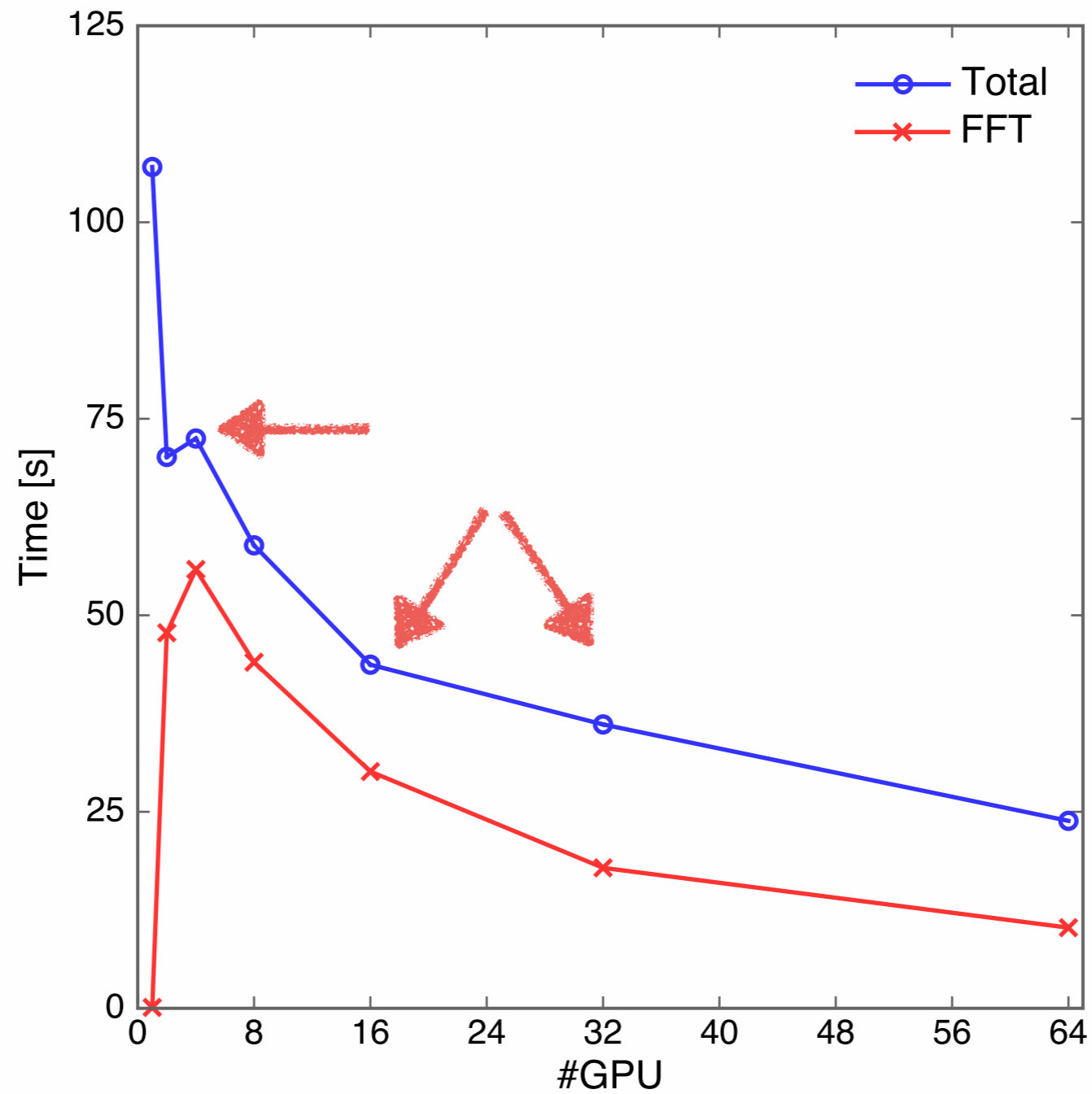
MVAPICH2 2.0b + CUDA5.5

→ CUDAで再実装

MVAPICH2 1.8.1 + CUDA6.0

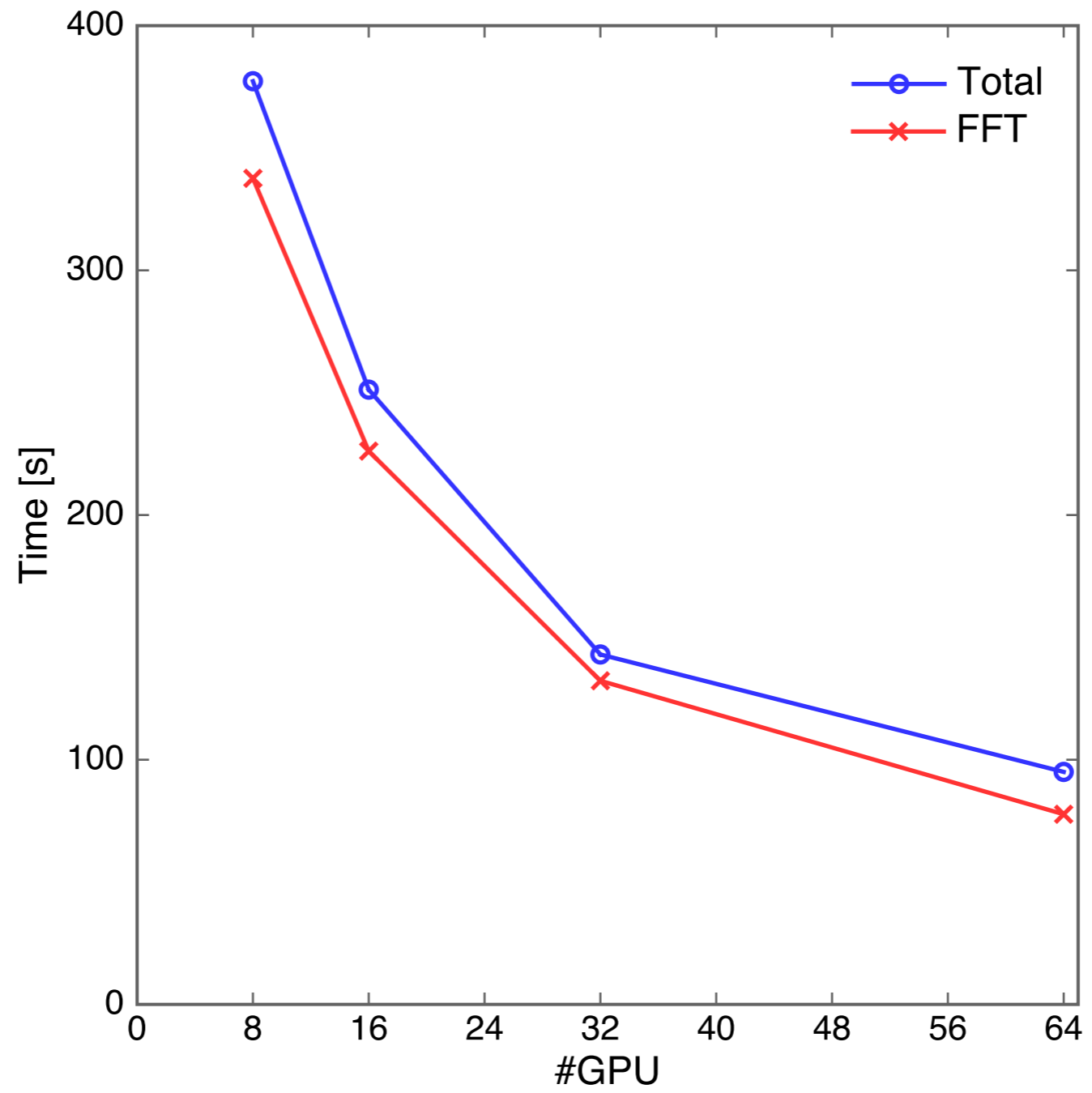
23秒 → 3.5秒

256³での結果



Persistence Modeがdisabledで6秒のオーバーヘッド

512³での結果



まとめ(2)

MPIで並列化したGPU用3D-RISMを作成して良好な結果を得た。

要望

32ノード以上のジョブの投入
Persistence Modeをenableに
MVAPICH2の最新バージョン