分子動力学専用計算クラスタの開発とそれを利用した計算資源提供型グリッドの試み

網崎孝志[†] 藤原伸一[†]

タンパク質など生体関連巨大分子系のための分子動力学計算専用 PC クラスタの開発と，そのグリッド上での利用の試みについて報告する．専用クラスタを構成する計算ノードは，通常の PC に二体相互作用専用計算ボードを装着したものである．この専用計算ボードと高速多重極法の併用により，分子動力学計算のボトルネックである Coulomb 二体相互作用の求値を，数値精度を維持したままで高速化した．4 ノードからなるクラスタで，通常の高速アルゴリズム計算のおよそ 46.8 倍の速度を達成した．また，そのような専用クラスタの計算パワーを，計算グリッド技術を用いて遠隔地のユーザに提供するための機能を 2 種類のアプローチで実現した．1 つは，Ninf-G を利用した Grid RPC 計算であるが，計算パワーの伝達において，ある程度のロスがみられた．しかしながら，遠隔地の利用者は桁違いの計算性能を入手できることに変わりはなく，また，RPC 方式の柔軟なプログラム開発が可能といった利点や今後の高速ネットワークの普及を考えると利用価値は高いと思われた．もう 1 つのアプローチは，計算と通信のオーバラップにより，エネルギーや原子位置の時間推移（すなわち座標トラジェクトリ）を遠隔地からモニタする方式で，Globus Toolkit を用いて実現した．この場合は，オーバヘッドをほとんど生ずることなしに，遠隔地の専用クラスタを利用することが可能であった．

Development of a Dedicated PC-cluster for Molecular Dynamics Simulations and Its Application in Computational Grids

Takashi Amisaki[†] and Shin-ichi Fujiwara[†]

This paper concerns the development of a special-purpose PC-cluster for large-scale molecular dynamics simulations of protein molecular systems. This paper also reports two approaches for grid-enabled use of the cluster. In such simulations, the most problematic part is that for the evaluation of Coulombic pair interactions. The dedicated cluster consists of ordinary PCs, each of which mounts special computation boards that calculate pair interactions at high speed. On the cluster, a parallel fast multipole method is executed in cooperation with the special boards to compress the time for the evaluation of the Coulombic interaction without compromising on accuracy. The cluster (with four PCs) was about 46.8 times faster as compared with fast-multipole-calculation on a PC. Using the cluster, we examined two approaches for grid-enabling molecular dynamics simulations. One approach is an Grid RPC method using Ninf-G, while the other is “simulation monitoring” approach implemented using Globus Toolkit. It is shown that the computation power of the specialized cluster can be effectively delivered to remote client sites, although considerable but tolerable level of power loss was observed during the delivery in one of the two classes of grid-enabled approaches.

1. はじめに

分子動力学（molecular dynamics; MD）計算は，未知タンパク質の構造や機能を理論的に予測/解析するための手法として期待されている．しかしながら，静電相互作用などの非結合性相互作用の求値に莫大な計算時間が必要であるため，現時点においては，計算環境に恵まれた一部の科学者がチャレンジしているにすぎない．また，長時間スケールのシミュレーションへの要求が高まってきているが，これは，時間ステップの反復数の増大を意味するだけでなく，各ステップにおいて相互作用のより精密な求値が不可欠となり，よりいっそうの計算パワーが必要とされる．それに見合っただけの計算資源を，一般のタンパク質科学者それぞれが用意することは非現実的であり，計算グリッド技術により，MD のための計算資源を遠隔地のユーザに提供する手立てが求められている．

† 鳥取大学
Tottori University

MD 計算の漸近計算量は $\mathcal{O}(N^2)$ である（N は系を構成する原子の個数）．これは，二体相互作用求値に起因する．実際，小規模なタンパク質分子系であっても，この部分が全計算時間の 99%以上を占める．そのため，漸近計算量を低減したアルゴリズムの開発もなされている．たとえば，Greengard の高速多重極法[1]（fast multipole method; FMM）や Saito の粒子粒子/粒子セル法[2] など多重極の分割統治的扱いに基づく方法や，Darden らの粒子メッシュEwald[3] など高速 Fourier 変換を利用するものがある．これらの高速アルゴリズムは，漸近計算量を $\mathcal{O}(N \log N) \sim \mathcal{O}(N)$ に低減するが，それは精度を犠牲にして何らかの近似を導入しているためである．我々の目的である精密な MD 計算を行うためには，依然として莫大な計算時間が必要であることに変わりはない．

実は，この精密計算で時間を要する部分は，多くの場合，高速アルゴリズムの近似が適用されない近距離二体相互作用の計算である．我々は，この残された部分の計算を，二体相互作用専用の演算プロセッサを搭載した「MD 専用計算ボード」に担当させることを提唱している．また，実際に，高速アルゴリズムと専用計算ボードの併用によるシステムを実現した結果，その有用性を確認することができた[4),5)]．このような専用計算ユニットは，対価格性能比の優れたシステムではあるが，単体での性能は PC の 20〜30 倍程度と限られている．大規模分子系の長時間スケール MD シミュレーションなどに対応するには，よりいっそうの高速化が求められる．そこで本研究では，このような専用計算ユニットのクラスタ化による高速化を行った．すなわち，計算ボードを装着した PC のクラスタ上で，遠距離/近距離二体相互作用をそれぞれ高速アルゴリズムと計算ボードに担当させるような並列計算方式を開発した．本論文の目的の 1 つは，この専用計算クラスタの開発と性能について報告することである．

本論文のもう 1 つの目的は，そのような専用計算クラスタを使った計算資源提供型グリッドの試みについて報告することである．専用計算機や専用計算クラスタにグリッド技術を適用することは，単にその計算パワーを提供するというだけではなく，それらの可用性における問題の解決につながる．すなわち，長大性能のものは研究者個人で調達できるような価格帯にない；駆動・制御に高度のプログラミング技術が要求される；計算資源管理のための機構が未整備であるといったことから，一般の利用者は解放される．

しかしながら，強大な計算パワーが実現できたとして，それが有効に供給できるのかということが危惧される．最もシンプルなグリッドの形態は，たとえば，計算時間の大部分を占める相互作用求値の部分を，遠隔手続き呼び出しのような形態で利用するというものであろう．このような方式では，毎時間サイクルですべての原子座標と力の転送という大量の通信が発生し，広域的ネットワークを利用した遠隔計算には，いかにも不向きと思われる．しかしながら，本論文で報告するように，このような形態の利用法であっても，幸いにして，ある程度の効果的な利用が可能であることが示唆された．本論文の後半部分では，このような形態を含め，2 種類の計算サービス提供形態を提案し，その設計・実装とベンチマークの結果について報告する．

本論文の以降の構成は以下のとおりである．2 章では，高速アルゴリズムと専用計算ボードを使った MD 計算システムについて概観する．この計算システムは，基本的には文献 4), 5) で報告したものであるが，以後の議論で必要となる事項を中心に概略を説明する．続いて，3 章では，この計算システムを計算ノードとするような PC クラスタ上での二体相互作用計算の並列化について報告する．4 章では，グリッド化の試みについて実装面も含めて報告する．それらの有効性を検証するために行った数値実験について，5 章で報告し，最後に，若干の考察と結論を述べる．

2. 高速アルゴリズム/専用計算ボード 併用型 MD 計算

静電相互作用は代表的な遠距離相互作用である．N 個の電荷 $\{q_i\}$ で構成された系において，電荷 i の位置での Coulomb ポテンシャルは

$$\sum_{j \neq i}^{N} \frac{q_i q_j}{r_{ij}}$$

で与えられる．r_{ij} は i と j の間の距離である．したがって，N 個の原子で構成された系で，ポテンシャルエネルギーや原子に働く力をすべて求める場合の計算量は $\mathcal{O}(N^2)$ である．高速アルゴリズムは，何らかの近似を導入することにより，これを $\mathcal{O}(N \log N) \sim \mathcal{O}(N)$ に軽減する．

一般に，二体相互作用求値の高速アルゴリズムにおいては，空間を遠方と近傍の 2 つに分割し，高速化技法は遠方領域にのみ適用される．たとえば，FMM の場合は，階層的なセル構造において，ある原子［**図 1** (a) の大きな粒子］が，その遠方領域（図でグレーの部分）に存在するすべての原子から受ける力を 1 個の多項式で近似表現する．この多項式は，局所展開と呼ばれ，各セルの多重極/局所展開の分割統治的変換により生

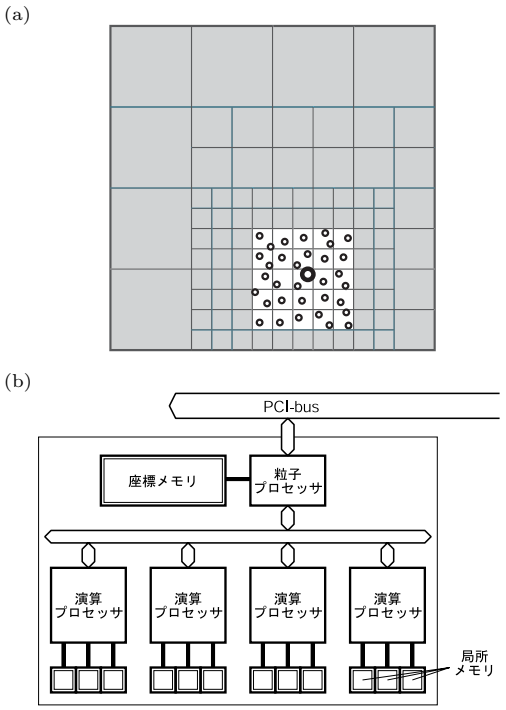

図 1 (a) 高速多重極法（FMM）における階層的セル構造の二次元の場合の例．大きく描いた粒子にその遠方領域（グレー部分）の粒子から働く力は，1 個の近似多項式で表す．近傍領域に存在する個々の粒子からの力は，それぞれの二体相互作用の和として求める．
(b) MD 専用計算ボード（MD Engine-II; MDE-II）の構成．4 個の演算プロセッサが二体相互作用の計算を行う．

Fig. 1 (a) An example of hierarchical cell structure of the fast multipole method (FMM) in two dimensions. (b) Organization of the specialized computation board (MD Engine-II; MDE-II).

成される．一方，近傍領域に存在する粒子との相互作用は，通常の方法で，すなわち，定義どおりに直接的に計算する．タンパク質分子の精密なシミュレーションを行う場合は，通常，この高速化されない近傍領域の計算が，全計算時間のうち大部分を占める[5]．我々の提唱している高速アルゴリズム/専用計算ボード併用型計算システムにおいては，この近傍領域の計算を，専用計算ボードに担当させる[5]．

本研究では専用計算ボードに MD-Engine II（富士ゼロックス，以後 MDE-II と略）を利用した．計算ボード MDE-II には，各種のメモリの他，粒子プロセッサと 4 個の専用演算プロセッサが装備されている［図 1 (b)］．粒子プロセッサは，座標メモリから原子の座標や電荷を取り出して演算プロセッサに配信する．それをもとに，各演算プロセッサは，専用演算パイプラインを用いて二体相互作用を高速に計算する．我々は，このような機能を実現した MD-Engine の開発にかかわってきたが[4)~7)]，特に，MDE-II においては，高速アルゴリズムとの併用のための機構を導入することを提言した．粒子プロセッサは，各原子の近傍領域に存在する原子のみを抽出する機構を備えている．この機構により，FMM をはじめとする高速アルゴリズムとの，効率の良い併用が可能である．

高速アルゴリズムと専用計算ボードの併用の最大のメリットは，計算精度を維持したうえでの計算時間の短縮化にある．前述のように，高速アルゴリズムを用いて精密なシミュレーションを行う場合は，近傍領域の計算に膨大な時間が費される．逆に，専用計算ボード単独では，漸近計算量が改善されず，大きな分子系の計算を実用的な時間範囲で行おうとすると，きわめて大規模な計算システム（多数の計算ボード）が必要となる．併用により，これら両者の難点が解消される．

なお，本論文での議論においては，「二体相互作用の求値時間がその個数のみで定まり，原子座標のメモリ上での位置などの影響を受けない」という MDE-II の性質が非常に重要である．このため，複数の計算ボードを使った並列処理において，再帰二分法に基づく静的負荷分散法[4),5)] がきわめて有効であった．これは，通常の PC や超並列計算機のプロセッサなどキャッシュ機構があまり強力でないような計算機と比較して，特筆すべき特長である．後述のように，クラスタによる並列化においても，この静的負荷分散法をそのまま適用した．

3. 専用計算ユニットのクラスタ化による並列処理

これまで述べたように，FMM と MDE-II の併用計算においては，もともと FMM 計算の一部であった近傍領域に関する計算を MDE-II に担当させる．したがって，併用計算の並列化にあたっても，大枠は FMM 計算の並列化としてとらえることができる．

階層的セル構造を構成したうえで，FMM 計算は，概略，葉セルの多重極を生成，上位セルの多重極の再帰的生成，局所展開への変換と子孫セルへの再帰的継承，葉セルでの局所展開の求値と近傍領域直接計算という順で進行する．これらの計算には，階層の各レベルでセルやそれに含まれる原子に関する自明な並列性がある．事実，これまで報告されている FMM やその変法の並列化においては，それを利用した並列化が行われていた[8),9),☆]．このため，議論の対象は，並列ア

☆ 我々の知る限り，唯一の例外は，文献 10) のセル間ベクトルを単位とする並列化であるが，いずれにしても，この階層セル構造上での並列性を利用することに変わりはない．

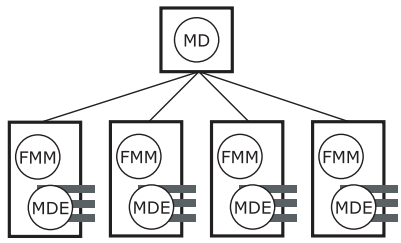

図 2 FMM/MDE-II 併用型計算ユニットのクラスタ化．各スレーブノード上で，FMM 計算により遠方領域との相互作用を計算する FMM プロセスと，近傍領域との相互作用を MDE-II を使って計算する MDE プロセスが動作する．マスタノードでは，MD 計算アプリケーションプロセスが動作する．

Fig. 2 A cluster computing system for MD simulations. The cluster is composed of FMM/MDE-II joint acceleration units.

ルゴリズムそのものではなく，プロセッサ間通信法や負荷分散法にある．ここでは，まず，前者に関連して，本研究での並列システムの構成と並列処理の概略について報告する．続いて，空間分割と負荷分散について述べる．

3.1 MD 計算の並列化

専用並列クラスタは，基本的に，**図 2** に示したようなマスタ/スレーブ型とした．各スレーブノードは，拡張スロットに MDE-II ボードを装着した PC である．このクラスタ上で，後述のような，“replicated data spatial decomposition”（RDSD）による並列処理を採用した．この方針は，マスタ部分が AMBER[11] などの MD プログラムの一部改変により製作できるなど，既存のソフトウェアとの親和性が高い．また，超並列処理においては，マスタ/スレーブや RDSD 形式はスケーラビリティに難点があるが，本研究ではノード単体の性能がきわめて高いため，比較的少数のノードで実用上十分な性能が達成できるものと判断した．

各プロセスの概略を**図 3** に示した．基本的に，スレーブプロセス群の役割は二体相互作用の求値である．より詳しくいうと，マスタノード上で動作する MD アプリケーションプログラムは，各時間ステップにおいて，全原子座標をスレーブノードにブロードキャストし，非結合性相互作用の求値を指示する．各スレーブノード上には 2 個ずつのプロセスが生成されており，一方が遠方領域との相互作用を FMM により計算し，他方は MDE-II を使って近傍領域との相互作用を計算する☆．最終的に，全プロセスの並列簡約（parallel reduction）により，マスタプロセス上に各原子に働く力を求める．

☆ スレーブは SPMD 方式でコーディングされている．

```
Master() {
    repeat                          /* MD 時間ステップの反復 */
        レベル L^par のセルを，各 Fmm プロセスに割当；
        レベル L のセルを，各 Mde プロセスに割当；
        全プロセスに座標などをブロードキャスト；
        並列簡約により，各原子に働く力を求める；
        力をもとに座標を更新;
        t ← t + δt;
    until t < T
}

Fmm() {
    座標など受信；
    担当する葉セルの多重極を求める；
    for l = L to 1 do      /* 上向きパス */
        レベル l の多重極からその親セルの多重極を求める;
    end for
    Fmm-プロセス間で多重極を全交換;
    for l = 1 to L do      /* 下向きパス */
        レベル l の遠方領域セルの多重極を局所展開に変換;
        親セルの局所展開をレベル l の局所展開に変換;
    end for
    担当する葉セルの各原子位置で局所展開を求値;
    並列簡約；
}

Mde() {
    座標など受信；
    for all ボード b do in parallel
        担当する葉セルの各原子の近傍相互作用の求値;
    end for
    並列簡約；
}
```

図 3 マスタープロセスと遠方領域担当スレーブプロセス（FMM），近傍領域担当スレーブプロセス（MDE）の動作概要．各スレーブノード上では，FMM-, MDE-プロセスが各々1 個ずつ存在し，並行に動作する．両プロセスが担当するセルは同一ではなく，別々の空間分割方式により定まる．また，複雑な通信を避けるため，各ノードの FMM-プロセスでは，レベル L^{par} より上位（$l < L^{\mathrm{par}}$）のセルに関する計算を重複して行う．

Fig. 3 Pseudo-code for master and slave processes. The two slave processes (FMM and MDE) are responsible for the computations regarding far and nearby regions, respectively.

3.2 2 種類の空間分割方式

問題は，FMM-プロセスと MDE-プロセスにどのセルを担当させるか，すなわち，遠方領域計算と近傍領域計算をどのように分割し，プロセスへマッピングするかである．

FMM の階層的セル構造は，セルを同じ大きさの子セルに分割することを再帰的に適用して得られるが，多重極の生成とその局所展開への変換などの遠方領域に関する計算は，この階層構造上のセルを単位とする空間分割が適している．そのような分割法を採用すると，並列アルゴリズムの骨格は図 3 の FMM-プロセ

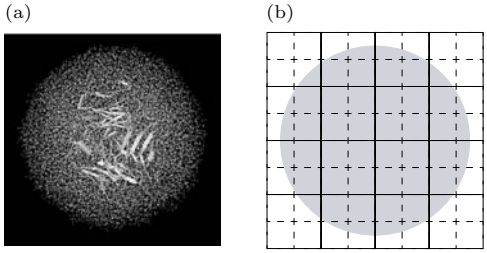

図 4 (a) 球状の水の中央に配置したタンパク質（免疫グロブリン Fab フラグメント，PDB ID 2mcp）．(b) 球状の分子系の空間分割（自由境界の場合）．たとえば，4×4 の部分セルに分割した場合，それに含まれる粒子数に著しい不均一性が生じる．なお，各セルの負荷量は，そのセルとその近傍セルのそれぞれの原子数の積で与えられる．

Fig. 4 (a) A protein molecule (immunoglobulin Fab fragment, PDB ID 2mcp) immersed in a sphere of water molecules. (b) Spatial decomposition of computation sphere (an example for free boundary condition).

スのようになる．そこでは，近傍領域の計算を行わないため，FMM の負荷量は，ほぼ，階層数 L によって決定され，各セルに含まれる原子数の影響をほとんど受けない．このため，本研究では，ある階層レベル（L^{par}）でのセルを各プロセスに均等に割り当て，各プロセスには，そのセルとその子孫のセルに関する計算を担当させた．本質的に，レベル L^{par} までの直交再帰二分法（orthogonal recursive bisection）を行うことに相当する．

一方，近傍領域の計算においては注意が必要である．タンパク質の MD シミュレーションは，周期境界条件あるいは自由境界条件で行われるが，いずれも，境界条件による人為的影響を受ける．その影響をできる限り排除した最も精密な計算には，**図 4** (a) に示したようなタンパク質を含む水の球をできるだけ大きくとった自由境界条件が適している．この場合，図 4 (b) の例からも分かるように，スレーブ間の負荷量に著しい偏りが生じる．この問題に対し，**図 5** のように，FMM の場合とは異なる空間分割方式を適用することにした．すなわち，空間順序が連続するセルは境界面を共有することが望ましいという専用計算ボードに起因する制約を考慮した結果，図 5 右図のように，交替行順序の上に一種の再帰二分法に基づく分割アルゴリズムを適用し，全ノード間で各ボードの負荷が均一となるように分割点を求めた．これは専用計算ユニット単体で複数の計算ボードを利用する場合の負荷分散法[4] と同じものである．なお，2 章の最後の段落で述べた計算ボードの特性から，葉セルごとに近傍領域に関する計算時間を予測することができる．この特別な性質のため，このような静的負荷分散が有効に機能する．

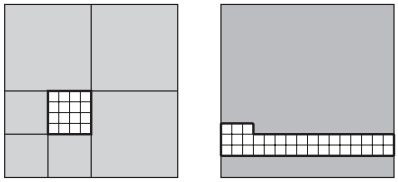

図 5 2 種類の空間順序の利用．（左）FMM による遠方領域計算では，その階層構造上でのあるレベル（L^{par}）でのセルを単位に仕事を割り当てる．図の白抜きの四角は，$L^{\mathrm{par}} = 2$ でのセル 1 個を表している．このような再帰的分割では，Morton 順序や Peano-Hilbert 順序が利用しやすいが，今回は実現の容易な前者を採用した．（右）MDE-II による近傍領域計算については，計算ボードの特性から，互いに隣接するセルからなる領域が要請される．その形状は任意であるが，今回はボード制御が容易な交替行順序を用いた．

Fig. 5 Two schemes for spatial decomposition used for the FMM (left) and MDE-II (right) computations.

4. 計算資源提供型 MD 専用グリッドの試み

我々は，MD 計算のための高性能計算システムを構築するには，前節で示した専用計算クラスタが対価格性能比の面で最も有望であると考えている．また，本論文の冒頭で述べたように，専用計算クラスタを構築し，その計算パワーを計算グリッドにより遠隔地ユーザに供給することができれば，専用計算ボードの可用性における問題点も解決できる．

MD 計算には種々の目的のものがあるが，それらをカバーするための MD 専用計算グリッドのサービス提供形態は，**表 1** に掲げたように整理されると考える．この章では，このうち最初の 2 つの形態について議論する（最後の形態については 6 章でふれる）．

4.1 相互作用提供型

「相互作用提供」型は，概略，クライアントマシン上で実行されている MD アプリケーションプロセスが，遠隔計算サイトのサーバマシンに対して，二体相互作用計算を行うライブラリ関数の実行を要求し，その結果を受け取るという作業を，MD 計算の各時間ステップにおいて行うものである（**図 6**）．このようなシステムは，以下に示すように，Ninf-G [12),13)] に用意されている GridRPC の機能で実現できた．

クライアントプロセス（すなわち MD アプリケーション）は，各時間ステップにおいて，`grpc_call()` により，二体相互作用を計算するためのライブラリ関数 `afmmdeng()` を遠隔呼び出しする．計算に必要となる（大量の）原子座標は，その引数という形で送信する．計算サイトのサーバ（Globus の gatekeeper も兼ねる）上では，GRAM により Ninf-G のスタブプログラムが起動される．これは，前述のクラスタ並列計算

表 1 本研究で計画している MD 専用計算サイトの機能
Table 1 Classification of services which may be provided by dedicated computation sites for MD simulations.

種別	サービス
相互作用提供型	二体相互作用のみを計算し返送
単一 MD 提供型	MD 計算全体を実行し結果を随時返送
多重 MD 提供型	レプリカ交換/温度並列 MD 計算を提供

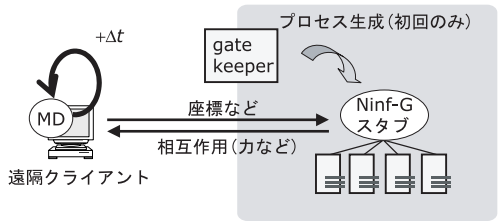

図 6 相互作用提供型システムの実装．MD アプリケーションはクライアント側で動作している．MD の各時間ステップにおいて，クライアントとサーバ（図中の “Ninf-G スタブ”）の間で大量の通信が必要である．Ninf-G スタブは gatekeeper 上で GRAM 要求により起動され，ジョブが完了するまで存在し続ける．このプロセスは，スレーブノードのマスタとしても機能する．
Fig. 6 An implementation of interaction-providing approach using Ninf-G.

でのマスタプロセスに相当する．このスタブが，以後，遠隔クライアントから `grpc_call()` があるたびに，ローカルに `afmmdeng()` を繰り返し呼び出すことになる．`afmmdeng()` の初回呼び出し時には，MPI/LAM の `MPI_Comm_spawn()` により，前述の相互作用計算スレーブプロセスが必要なだけ生成される．専用クラスタ上のスレーブ群により計算された結果（ポテンシャルエネルギーと各原子に働く力）は，Ninf-G の機構により，`afmmdeng()` の引数という形で遠隔地のクライアントプロセスに戻される．2 度目以降の呼び出し時には，新たにプロセスを起動することなく，すでに存在しているマスタ/スレーブプロセスを使って同様の計算を行う．

この相互作用提供型方式は，計算資源提供という語句からは最も連想しやすく，また，実現面でも Ninf-G などのグリッド型遠隔手続き呼び出し[12]（GridRPC）方式に明確に適合する．さらに，通常の MD アプリケーションプログラムでは，非結合性相互作用を計算する部分が独立したルーチンとなっており，アプリケーション側からも利用しやすいという柔軟性がある．反面，各時間ステップにおいて，座標と力についての大量の通信が発生するため，効果的な遠隔計算システムとして実現できるのか懸念される．しかし，後述のように，本研究の数値テストでは十分な有用性が確認

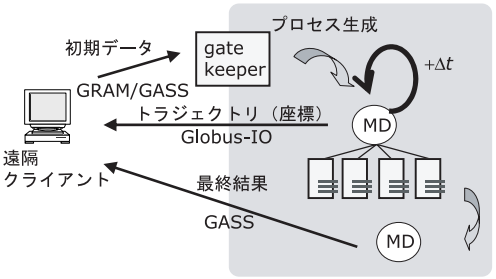

図 7 単一 MD 提供型システムの実装．MD アプリケーションはマスタ側（計算サイト）で動作する．初期データと最終データは GASS Copy で伝送する．スレーブノード上の相互作用求値計算とマスタ（サーバ）からクライアントへの通信は同時に進行する．
Fig. 7 An implementation of single-MD-providing approach using Globus Toolkit APIs.

された．

4.2 単一 MD 提供型システムの設計と実装

遠隔地の計算サーバに，1 個の MD ジョブ全体の実行を依頼し，その MD シミュレーションの実行状況（すなわちエネルギーや座標の変動）をモニタするような実行形態である．多くの MD 計算では，このようなモニタが不可欠である．**図 7** に示したように，単純な GridRPC 方式での実現は困難であるため，この計算方式は，Globus Toolkit [14),15)] を API（GRAM, GASS Copy，Globus IO など）を通して利用する形式で実装した．具体的には，以下のとおりである

先ほどの例とは異なり，クライアントは MD に関する計算を直接には行わない．クライアントプロセスは，起動後，`globus_gass_copy_register_url_to_url()` を使って初期座標などの入力データファイルを gatekeeper に非待機方式で送信する．同時に，`globus_client_job_request()` により GRAM 要求を送信し，gatekeeper 上で MD アプリケーションプロセスを起動する．このプロセスは，二体相互作用計算用サーバ兼マスタプロセスとしても働く．このプロセスとクライアントプロセスは，先のファイル転送が完了するまでブロックする．転送が完了すると，マスタプロセスは MD 計算本体の実行を開始する．すなわち，専用クラスタを用いて相互作用を計算し，その値をもとに原子位置を更新するというステップを繰り返す．各時間ステップにおいて，これらの座標は，いったん，送信用バッファに格納される☆．マスタプロセスは，専用クラスタ上のスレーブに対して，次の時間ステップにおける相互作用計算を指示したうえで，バッ

☆ 一般的な MD 計算であれば，これらの座標転送を必ずしもすべての時間ステップにおいて行う必要はない．0.1 ps に 1 回程度の頻度で記録したものをトラジェクトリとすることが多い．

ファ上の原子座標をクライアントに送信する．この送信は，あらかじめ確立しておいた TCP ソケットを使って，`globus_io_register_write()` により非待機方式で行う．このように座標（トラジェクトリ）の送信と，次ステップにおける相互作用計算を並列に行い，通信時間の隠蔽を図る．遠隔クライアントとサーバの間の通信では，この大量の座標トラジェクトリの転送が通信時間の大半を占めるため，このような計算と通信のオーバラップにより，遠隔分散計算でのオーバヘッドが軽減できるものと思われる．

この方式は，次章で示すように，ほとんどオーバヘッドを生じることなく効率的に実行できた．ただし，MD アプリケーションを特定してジョブを実行することになるため，利用者側からみて柔軟性を欠くことになるのは否めず，ポータルシステムを準備するなどして，定型的な計算（たとえば，自由エネルギー摂動計算による単塩基多型機能予測）での利用が適切であろう．

5. 数値実験

対象の系は，球形の水に沈めた免疫グロブリンフラグメント［PDB ID 2mcp，図 4 (a)］とした．総粒子数 N は 50,764 個．MD プログラムには Amber 5.0 [11] を用いた．コンパイラには gcc-3.2.1 を用いた．

5.1 専用クラスタによる並列処理

専用計算ノード 4 台からなるクラスタを使用した．専用計算ノードには，Pentium 4 PC（2 GHz，Red Hat 7.1）に MDE-II ボード各 2 枚を装着したものを用いた．それぞれのノードは Gigabit Ether で結合し，ノード間通信には MPICH-1.2.4 [16] を用いた．

100 ステップ（200 fs）のシミュレーションに費した時間を**表 2** に示した（1 ステップ平均，セットアップ時間は除く）．なお，表でカッコ付きで示した精度は，力の平均精度を十進数の有効桁数で表したものである．

今回は，ノードあたり 2 枚の計算ボードを用いたが，それでも，FMM 単独に比べ，単一ノードでおよそ 19 倍，4 ノードで 12 倍の高速化が達成できている．なお，平均 3.1 桁の有効精度は，タンパク質の精密シミュレーションに関する要求精度の我々の実験から，最低限必要なレベルであることが分かっている[5]．FMM/MDE-II 併用時の並列効率は，表で示した精度の順に，60.9%，67.1%，71.2%，71.5%であり，概して，精度が高いほど，効率が高い．また，この順は，ほぼ，近傍領域の計算量が増大する順となっている．また，MDE-II 単独の場合の並列効率は 89.8%であった．今回の数値実験で用いた程度の分子系のサイズであれば，きわめて高精度が要求される場合は，MDE-II 単独の並列処理の方が効率的であると考えられる．

表 2 $N = 50,764$ の系での MD 1 ステップあたりの平均所用時間（単位：秒）

Table 2 Average time spent in performing single MD step for the system of $N = 50,764$.

1 ノード	(精度)				
	(3.1)	(4.1)	(5.3)	(6.1)	(7.2)
FMM	74.9	75.2	77.9	127.4	—
MDE-II	—	—	—	—	27.3
FMM/MDE-II	3.9	5.1	7.4	14.3	—
4 ノード	(精度)				
	(3.1)	(4.1)	(5.3)	(6.1)	(7.2)
FMM	19.7	19.7	20.5	33.2	—
MDE-II	—	—	—	—	7.6
FMM/MDE-II	1.6	1.9	2.6	5.0	—

5.2 遠隔分散計算のベンチマークテスト

計算ノードには，先ほどと同じものを用いた．gatekeeper には Pentium 4（2 GHz）PC を，クライアントにも Pentium 4（2 GHz）PC を用いた．今回の実験では，すべて同一の実験室に配置したマシンを使ったが，遠隔分散計算をシミュレートするため，クライアントと gatekeeper の間は 10 Mbps の Ethernet で接続した．クラスタ並列化には MPI/LAM-6.5.8 [17] を，グリッド化には Ninf-G 1.1.1 [18] と Globus Toolkit 2.2 [15] を用いた．前節で述べたものと同じ分子系を対象とし，4 章で述べた単一 MD 提供型と相互作用提供型による 5 ps のシミュレーション（2500 ステップ）を，各々10 回ずつ行った．各シミュレーションで，50 ステップごとに全原子の座標を記録したものをトラジェクトリとした．なお，トラジェクトリと同容量のファイルを，通常の ftp により gatekeeper からクライアントへ転送すると 72 秒を要した．これより算出した実効帯域は 6.5 Mbps であった．

結果を**表 3** に示した．この表には，参考のために，同じ計算を MD 専用クラスタ上でオンサイトで実行した場合の所要時間も示した．「MD 本体実行時間」とは，AMBER のサブルーチン `runmd()` の実行に要した時間で，MD の時間ステップの繰り返しに要した時間である．それ以外の，MD 専用クラスタの初期化や，Globus や Ninf-G の初期化などに要した時間は含めていない．「相互作用求値時間」とは，専用クラスタ上で二体相互作用計算に費した時間を，全ステップにわたって合算したものである．ここには，相互作用提供型の場合についても，グリッド化に起因するオーバヘッドは含んでいない．表にあげた 3 種の方法で，この値に有意な違いがみられないことから，その他の所要時間での差異は，それぞれのグリッド化方式に起因すると思われる．

表 3 遠隔分散計算でのベンチマークテスト結果 [a]
Table 3 Benchmark results for the remote distributed computing.

	オンサイト	単一 MD 提供型	相互作用提供型
総経過時間 [b]	3986	4110	17768
	2	26	24
MD 本体実行時間 [b]	3978	3992	17744
	2	21	21
相互作用求値時間 [b]	3851	3855	3864
	2	20	23
平均実行時間/ステップ [c]	1.59	1.60	7.10
平均転送量/ステップ	—	24	3749

[a] タンパク質–水分子系（$N = 50,764$）の 5 ps シミュレーションの結果．所要時間の単位は秒．転送量の単位は KB.
[b] 上段は 10 回の計測の平均値．下段は標準偏差．それぞれの時間区分の詳細については本文参照．
[c] MD 本体実行時間を総時間ステップ数で除したもの．

単一 MD 提供型での総実行時間は，オンサイト計算に比べ 3.1%だけ増大したにすぎない．このオーバヘッドのうち 68%は，初期座標などの入力データファイルの送信と最終結果の受信によるものであった．表 3 に示した MD 本体実行時間は，これらファイル転送や GRAM 要求送信などの準備作業を除いたものであるが，オンサイト計算に比べて 14 秒だけ増大した．トラジェクトリ転送はこの部分に含まれているが，ftp による一括転送での所要時間（72 秒）を考えると，通信時間の隠蔽がある程度効果的に行われているのが分かる．

一方，相互作用提供型の場合は，総経過時間のみをオンサイト計算と単純に比較すると，およそ 4.5 倍と，予想されたように大幅に延長した．相互作用求値の所要時間は，3 つの方法で同程度であったため，相互作用提供型での大幅な延長は大量のデータ転送が原因と思われる．ステップあたりでみてみると，オンサイト計算との差は 5.51 秒である．前述の実効帯域とデータ転送量とから通信時間を推定すると，ステップあたり 4.48 秒となり，若干のひらきはあるものの，相互作用提供型での所要時間延長は，その大部分が大量のデータ通信によるものであることが確認できる．ただし，相互作用提供型の遠隔分散計算により，クライアントの視点では計算が大幅に高速化されたということは重要である．すなわち，ここで用いたクライアントマシン単独で FMM のみを利用した計算を行った場合と比較すると，1/10 未満の時間で計算が完了している．

6. おわりに

MD 計算は，その自明な並列性のため，古くから並列処理の例として取り上げられてきた．しかしながら，一般に，遠距離相互作用を無視しない限り，通常の PC クラスタや超並列処理計算機で高い効率を達成するのは困難である．FMM などを採用すれば，比較的高い効率を達成できるが，それでも，30–70%程度である．効率向上のため，本論文で示した空間分割ではなく，多重極–局所展開変換のシフトベクトルの分類に基づく分割なども提案されているが，それでもたかだか 8 台の PC で 86%の並列効率が達成されたにすぎない[10]．それを考えると，本研究の MD 専用クラスタでは，4 ノードの併用計算で，1 ノードの FMM 計算の 46.8 倍の性能を達成しており☆，専用クラスタという方策がきわめて効果的であるのが分かる．

また，本論文では，そのような専用クラスタの計算パワーを，計算グリッド技術を用いて遠隔地のユーザに提供することも現実的に可能であることを示した．特に，単一 MD 提供型では，ほとんどロスなしに計算パワーを伝達できた．一方，相互作用提供型では，大量のデータ通信が障害となり，計算パワーの伝達において少なからぬ減弱がみられた．しかしながら，PC 1 台で FMM 計算を行う場合と比較すると 10 倍以上の高速化が実現され，その有用性が確認できた．なお，この方式において，クライアントが 1 プロセスのみの場合は通信の隠蔽が困難であるが，複数クライアントでの利用の場合は，通信と計算のオーバラップが可能となる．本手法の持つ汎用性・柔軟性という利点を考えると，今後，複数クライアント利用時の利用率向上や，通信量削減のための転送データの厳選などの検討を行うことには十分な価値があるものと思われる．また，今後の広域的な高速ネットワークの普及も，本手法に有利に作用するであろう．

これまで，表 1 の 3 つの形態のうち「多重 MD 提供」型には言及しなかった．これは，単一 MD 提供型以上に柔軟性を犠牲にして，特定の計算，すなわち，レプリカ交換 MD や温度並列 MD など多数の MD スレッドを並列に実行するような拡張アンサンブル MD に，用途を限定するものである．すでに，この方式としては，Stanford 大学のインターネット・コンピューティング・プロジェクト Folding@home プロジェクト[19] などが知られている．ただし，そこでは，1 個のレプリカに関する MD を通常の非力な PC で完結させるため，溶媒（水）効果を簡略化するなどしており，本研究の精密計算の方向性とは若干異なる．

☆ この数値は，各ノード上の計算ボードの枚数（今回は各 2 枚）を増やせば，さらに向上するものと思われる．

MD 計算は，必ずしも完成された方法ではなく，その有用性と限界を明らかにするためには，多くのタンパク質科学者が，これらの手法の試用・評価・改良に参画することが必要である．その意味からも，本論文で述べたような計算グリッド技術による計算資源提供型計算が広く普及することが期待される．残念ながら，今のところ，本研究のプロトタイプは，実用的グリッドとして供するには不備な点が多い．今後，専用クラスタ上での資源管理システムの構築，ポータルシステムの構築，GSI などセキュリティ関連機能の実装などが必要であろう．

謝辞 本研究の一部は，文部科学省科学研究費補助金特定領域研究（2）課題番号 15017266 による．

参 考 文 献

1) Greengard, L.: *The Rapid Evaluation of Potential Fields in Particle Systems*, MIT Press, Cambridge (1988).
2) Saito, M.: Molecular dynamics simulations of proteins in water without the truncation of long-range Coulomb interactions, *Molecular Simulation*, Vol.8, pp.321–333 (1992).
3) Darden, T., York, D. and Pedersen, L.: Particle mesh Ewald: An $N \cdot \log(N)$ method for Ewald sums in large systems, *J. Chem. Phys*, Vol.98, pp.10089–10092 (1993).
4) 網崎孝志，豊田新次郎，宮川博夫，北村一泰：高速多重極法と専用計算機の併用による分子動力学計算高速化のための二つのアルゴリズム，*J. Computer Chemistry, Japan*, Vol.1, pp.73–82 (2002).
5) Amisaki, T., Toyoda, S., Miyagawa, H. and Kitamura, K.: Development of hardware accelerator for molecular dynamics simulations: A computation board that calculates nonbonded interactions in cooperation with fast multipole method, *J. Comput. Chem.*, Vol.24, pp.582–592 (2003).
6) Amisaki, T., Fujiwara, T., Kusumi, A., Miyagawa, H. and Kitamura, K.: Error evaluation in the design of a special-purpose processor that calculates non-bonded forces in molecular dynamics simulations, *J. Comput. Chem.*, Vol.16, pp.1120–1130 (1995).
7) Toyoda, S., Miyagawa, H., Kitamura, K., Amisaki, T., Hashimoto, E., Ikeda, H., Kusumi, A. and Miyakawa, N.: Development of the MD Engine: A high-speed accelerator with a parallel processor design for molecular dynamics simulations, *J. Comput. Chem.*, Vol.20, pp.185–199 (1999).
8) Singh, J.P., Holt, C., Hennessy, J.L. and Gupta, A.: A parallel adaptive fast multipole method, *Proc. 1993 ACM/IEEE conference on Supercomputing*, Portrland, Oregon, pp.54–65 (1993).
9) Board, J.J.A., Hakura, Z.S., Elliott, W.D. and Rankin, W.T.: Scalable variants of multipole-accelerated algorithms for molecular dynamics applications, *Proc. 7th SIAM Conference on Parallel Processing for Scientific Computing*, Philadelphia, pp.295–300, SIAM (1995).
10) Choi, C.H., Ruedenberg, K. and Gordon, M.S.: New Parallel Optimal-Parameter Fast Multipole Method (OPFMM), *J. Comput. Chem.*, Vol.22, pp.1484–1501 (2001).
11) *AMBER 5*, San Francisco (1997).
12) Seymour, K., Nakada, H., Matsuoka, S., Dongarra, J., Lee, C. and Casanova, H.: Overview of GridRPC: A Remote Procedure Call API for Grid Computing, *Grid Computing — Grid 2002*, pp.274–278 (2002).
13) 田中良夫，中田秀基，平野基孝，佐藤三久，関口智嗣：Globus による Grid RPC システムの実装と評価，情報処理学会ハイパフォーマンスコンピューティング研究会，Vol.2001, No.77, pp.165–170 (2001).
14) Foster, I. and Kesselman, K.: Globus: A metacomputing infrastructure toolkit, *Journal of Supercomputer Applications*, Vol.11, pp.115–128 (1997).
15) http://www.globus.org/.
16) http://www-unix.mcs.anl.gov/mpi/mpich/.
17) http://www.lam-mpi.org/.
18) http://ninf.apgrid.org/.
19) http://FoldingAtHome.Stanford.edu.

(平成 15 年 10 月 6 日受付)
(平成 16 年 1 月 5 日採録)

網崎 孝志（正会員）

昭和 34 年生．昭和 58 年大阪大学大学院薬学研究科前期課程修了．同年鳥取大学医学部附属病院技官．平成 3 年より島根大学理学部情報科学科助手，同講師，同総合理工学部数理・情報システム学科助教授を経て，平成 12 年より鳥取大学医学部保健学科教授．分子シミュレーション等の計算科学的研究，ならびに薬物動態解析等の医療データ解析に関する研究に従事．博士（薬学）．ACM，日本生物物理学会，日本薬学会等各会員．

藤原 伸一

昭和 51 年生．平成 15 年京都大学大学院薬学研究科医療薬科学専攻博士後期課程修了．同年より鳥取大学医学部保健学科助手．現在，薬物動態関連タンパク質の分子動力学シミュレーションに関する研究に従事．博士（薬学）．日本薬学会，日本薬物動態学会各会員．