

分子動力学専用計算機 MDGRAPE-4

大野 洋介^{1,a)} 泰地 真弘人¹ 大村 一太¹ 森本 元太郎¹ 長谷川 亜樹¹

概要: ミリ秒タイムスケールのタンパク質等の生体分子の古典分子動力学シミュレーションを達成するために開発中の分子動力学専用計算機 MDGRAPE-4 について報告する。汎用並列計算機での生体分子の古典分子動力学シミュレーションのストロングスケーリングは主に通信性能によって制限されている。これを解消するため MDGRAPE-4 では専用パイプライン演算器とネットワークインターフェイスを統合した SoC を採用した。計算プログラムとしては Gromacs を選択したが、専用 SoC で動作させるため大幅な変更が必要となり、ハードウェアの製造と並行して、MDGRAPE-4 用の Graomacs の開発を進めている。

1. Introduction

古典分子動力学シミュレーションでは、古典近似の結合力、クーロン力、分子間力で相互作用する原子の運動方程式を数値積分することで、原子・分子で構成される系の構造・エネルギー等の時間発展を計算する。原子の個数を N としたときの演算量は数値積分と結合力計算は $O(N)$ であるのに対し、遠距離二体相互作用であるクーロン力と分子間力は $O(N^2)$ となり、全体の計算時間の多くを占める。そのため原子数の多い系を扱うために様々なアルゴリズムが使用されている。代表的なものは FFT を利用し演算量を $O(N \log N)$ に減らす PME 法 [2] である。大規模並列環境では FFT より並列性の優れており $O(N)$ 法である FMM も利用されはじめて [1]。計算条件によっては、Wolf 法 [13]、zero-dipole summation [5]、isotropic periodic sum (IPS) [14] のような演算量が $O(N)$ のカットオフ法も使用される。これらは、アルゴリズムによって演算量を減らしているが、全く異なるアプローチとして GRAPE [3] のような専用計算機により二体相互作用計算を加速する方法もある。

GRAPE は元々は天文学の重力多体問題専用計算機として開発された。重力多体問題では銀河や球状星団のような恒星の集団を重力相互作用する質点系として扱う。重力はクーロン力と同様に大きさが距離の二乗に反比例する二体力であるため質点の個数を N とすると $O(N^2)$ の演算量

が必要となり、重力多体系の数値計算の計算量のほとんどを重力の計算が占める。1 ペアあたりの演算量は距離の逆数が必要なため 20 演算以上と多くペア数も多いため段数の深いパイプラインで高スループットが可能である。このような考えに基づき、重力パイプライン (GRAPE PipE) が開発された。クーロン力も符号の有無以外は重力と同じ形式であるため、距離だけに依存する二体相互作用である Lennard-Jone モデルの分子間力も扱えるようにパイプラインを拡張することで、MD にも使用できる GRAPE が開発された。その後 MD 用 GRAPE は MDGRAPE として独立に開発されるようになった。MDGRAPE-3 [6], [12] では、100 台のホスト計算機と 48000 個の専用 LSI で 1PFlops のピーク性能で、1700 万原子の計算で実効性能 185TFlops を達成した。初期の GRAPE/MDGRAPE は汎用計算機に接続するアクセラレーターとして実装された。 $O(N^2)$ 計算ではデータあたりの演算量が多いため $O(N)$ のデータ転送コストは汎用の I/O でもボトルネックにならなかったからである。また、二体相互作用計算部分のみを専用ハードウェア化し、既存の汎用計算機とその上で動作するソフトウェアを使用することで、設計コストを抑えることができた。しかしながら、並列化、演算性能の向上、演算量の少ないアルゴリズムへの対応により、アクセラレーターと汎用コアやノード間通信が問題となった。MDGRAPE-3 の場合、高い計算効率を達成するには専用 LSI あたり数千原子の規模が必要で、上述の計算例では、計算時間は 550ms/step であった。大きな系を計算するウィークスケーリングでの性能向上は可能であったが、小規模計算で計算時間を短縮するストロングスケーリングでの性能向上は困難であった。

生命科学では古典分子動力学シミュレーションは水中

¹ 理化学研究所 生命システム研究センター (QBiC) 生命モデリングコア 計算分子設計研究グループ

Laboratory for Computational Molecular Design Computational Biology Research Core Quantitative Biology Center (QBiC) RIKEN, 1-6-5 Minatojima-minamimachi, Chuo-ku, Kobe 650-0047, Japan

a) ohno@riken.jp

のタンパク質・DNA・糖脂質二重膜等の生体分子の構造変化・エネルギー等を計算するのに用いられる。計算規模は小型のタンパク質で数万原子（水を含む）であるが、多数の生体分子を含む系では 1000 万原子を越える規模になる。一方、タイムスケールは生体中の現象ではミリ秒以上のもが多い。時間刻みは結合力のタイムスケールにより 2fs 程度となるため、必要なステップ数は 5×10^{11} となる。通常の計算機環境では 1 ステップの計算時間は ms 程度であるので、1 日で 200ns、100 日かけても 20 μ s 程度の積分時間にしか到達しない。ms 積分のためには 20 μ s/step 程度と 50 倍近い計算時間の短縮が必要となる。必要な演算性能は 1PFlops 程度で京のような大規模並列計算機で達成されているが、通信性能、特に通信レイテンシによるストロングスケーリングの限界で計算時間の短縮は困難である。MDGRAPE-4 では分子動力学計算に特化した専用パイプラインとネットワークインターフェースを集積した SoC(System on Chip) でストロングスケーリング性能を追求しミリ秒オーダーの生体分子シミュレーションを可能とする計算機である [7]。同様の専用計算機としては Anton[11] がある。

2. 計算手法

Newton の運動方程式を全ての原子に関して数値積分することで系全体のダイナミクスを計算する (式 1)。生体分子の MD では leap-frog や velocity-Verlet が使われることが多い。運動方程式そのものは全エネルギーが保存される形式であるので、最も単純なアンサンブルは粒子数、体積、エネルギー一定 (NVE) で、平衡化や実験との比較のためには、粒子数、体積、温度一定 (NVT) や粒子数、圧力、温度一定 (NPT) が使われ、温度や圧力を制御する計算が追加される。いずれもアンサンブルでも $O(N)$ の計算量である。力は、結合力、分子間力、クーロン力である。結合力は調和振動子等の古典近似で表現する (式 2 はポテンシャル表記で、 r で微分したものが力である)。分子間力は生体分子を対象とする場合、Lennard-Jones モデルを使うことが多い (式 2)。

$$m_i \frac{d^2 \mathbf{r}_i}{dt^2} = -\nabla U(\mathbf{r}_1, \dots, \mathbf{r}_{N_{atom}}) \quad (1)$$

$$\begin{aligned} U(\{\mathbf{r}_i\}) = & \sum_{\text{bond}} \frac{1}{2} k_b (r - r_0)^2 + \sum_{\text{angle}} \frac{1}{2} k_a (\theta - \theta_0)^2 \\ & + \sum_{\text{torsions}} \frac{1}{2} V_n [1 + \cos(n\omega - \gamma)] \\ & + \sum_{i < j} \left\{ \epsilon_{ij} \left[\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - 2 \left(\frac{\sigma_{ij}}{r_{ij}} \right)^6 \right] \right. \\ & \left. + \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}} \right\} \quad (2) \end{aligned}$$

式 2 の第一項から第三項が結合力で、第一項、第二項

の結合距離、結合角の項は復元力が大きく高速な振動となり他の項による力より短い時間刻みが必要となる。このため、古典調和振動子の形式で計算せず、距離、角度の拘束条件として扱う場合もある。多数の原子が結合したタンパク質では全ての結合距離を拘束するのは困難なため、水素を含む結合のみ拘束条件で扱う場合が多い。結合の個数はほぼ原子数に比例するため計算量は $O(N)$ である。

第四項が遠距離二体力である分子間力 (Lennard-Jones モデル) とクーロン力である。分子間力は距離による減衰が早く、0.9-1.2nm でカットオフしても計算誤差は少ない。水中生体分子では粒子密度はほぼ一定なのである原子からカットオフ距離にある原子数もほぼ一定であり、カットオフを適用した場合の演算量は $O(N)$ である。クーロン力は減衰が遅いため短い距離での単純なカットオフでは誤差が大きいため、後述するようなアルゴリズムが使われる。従来の GRAPE/MDGRAPE では二体相互作用の計算が加速されるので長い距離でのカットオフを使用することが多かった。

2.1 クーロン力計算手法

クーロン力の計算量を削減するアルゴリズムが必要となる。現在、生体分子の MD では Particle Mech Ewald(PME) 法 [2] がよく使われている。PME 法は Ewald 法 [4] を元にしておりクーロン力を周期境界条件化で波数空間と実空間に分解して計算する (式 3)。

$$\begin{aligned} U_C = & \frac{1}{2} \sum_{i,j} q_i q_j \frac{\text{erfc}(\alpha r_{ij})}{r_{ij}} \\ & + \sum_{i,j} q_i q_j \sum_{\mathbf{k} \neq 0} \frac{\exp(-(\pi \mathbf{k} / \alpha)^2 + 2\pi i \mathbf{k} \cdot \mathbf{r}_{ij})}{2\pi V \mathbf{k}^2} \\ & - \frac{\alpha}{\sqrt{\pi}} \sum_{i=1}^N q_i^2 \quad (3) \end{aligned}$$

波数空間ではガウス分布で補正した電荷による力 (式 3 第二項) を計算し、実空間ではガウス分布補正と本来の点電荷の差である補正誤差関数の形の電荷による力 (式 3 第一項) を計算する。これにより波数空間でも実空間でも誤差を抑えてカットオフできる。元々の Ewald 法では電荷分布のフーリエ変換と波数空間で計算した力から原子への力への逆フーリエ変換の計算コストが問題であったが、PME 法では電荷分布を近傍メッシュ点に配分することでフーリエ変換に FFT を使い高速化している。実空間でのカットオフ計算、メッシュ点への電荷配分は $O(N)$ の演算量に抑えられ、メッシュ点数が原子数にほぼ比例する条件では FFT の演算量は $O(N \log N)$ となる。MDGRAPE-4 は 10 万原子程度の小規模計算を対象としており、512 ノード中 64 ノード使用して $32^3 - 64^3$ 程度の FFT を実行する。元々の Ewald 法ではガウス分布で電荷を広げているが、PME 法では精度と演算量のバランスからメッシュ点への電荷分

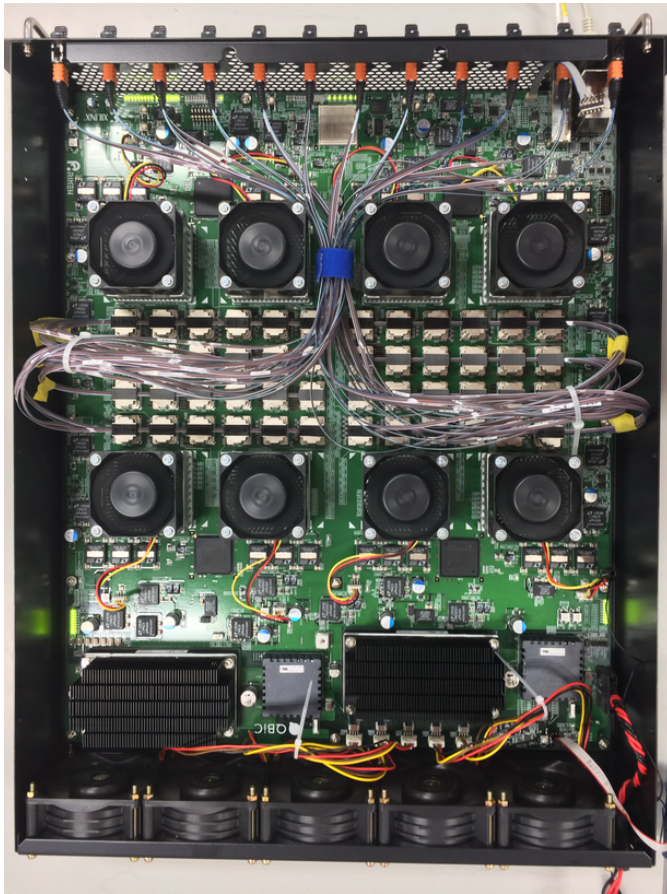


図 1 MDGRAPE-4 board

SoCがファン・ヒートシンク付きで8個実装されている。ボード上の8チップは2x2x2のメッシュ状に接続されている。残りのインターフェースは中央部で光信号に変換され、6方向送受信毎にまとめられ上部の光コネクタを経由して他のボードへ接続される。

配 (Charge Assignment) とメッシュ点での力から原子に働く力への補完計算 (Back Interpolation) に xyz 軸毎の B-splie を使用している。MDGRAPE-4 の専用パイプラインは距離のみに依存する関数形を高速で処理できるため、原子の座標を中心としたメッシュ点でのガウス関数の計算も高速化できる。このため、ガウス関数でメッシュ点への電荷を分配する GSE(Gaussian Split Ewald)[10] を採用する。

3. ハードウェア

MDGRAPE-4 は 512 個の SoC を 3D トーラスで結合して構成される。MDGRAPE-4 の SoC は 2.5TFlops 相当のピーク性能で、512 個で 1PFlops 相当のピーク性能となる。8 個の SoC が一枚のボードに実装され (図 1)、ボード上の 8 チップは 2x2x2 の 3D メッシュ結合となっている。

3.1 チップ構成

MDGRAPE-4 の SoC は二体力専用パイプライン (以下

パイプライン) 64 個、汎用コア 64 個、ネットワークインターフェース、コントロール汎用コア、グローバルメモリで構成されている (図 2)。パイプラインと汎用コアはそれぞれ 8 個ずつ 8 ブロックにまとめられている。パイプラインは 32bit 固定小数点で分子間力とクーロン力を計算する。動作クロック周波数は 800MHz である。二体力計算の演算数が 50 演算相当とすると 40GFlops 相当のピーク演算性能で、チップ (64 パイプライン) で 2.5TFlops 相当となる。グローバルメモリの容量は 1.8MByte で各ユニットからアクセスできる。汎用コアからは 9.6GByte/s、パイプラインからはリード 14.4GByte/s、ライト 7.2GByte/s、ネットワークインターフェースからは 4.8GByte/s でアクセスできる。汎用コアは動作クロック周波数 600MHz の Xtensa™LX4 (Tensilica 社) で、8kByte の命令メモリ、8kByte のデータメモリ、4kByte の命令キャッシュ、4kByte のデータキャッシュを持つ。グローバルメモリはレイテンシが長いので、計算性能が要求される場合は命令メモリ・データメモリを使用する。命令メモリメモリにはフロントエンド計算機から直接書き込めない。フロントエンドから書き込み可能な 512kByte のインストラクションメモリが 32 コア毎にあり、複数の実行コードを配置しておくことができる。汎用コアは必要に応じて DMA 転送によりインストラクションメモリ上の実行コードを命令メモリへコピーして使用する。データメモリもフロントエンド計算機から直接アクセスできず、汎用コアが適時グローバルメモリとデータメモリの DMA 転送を行なう。コントロール汎用コアは基本性能は汎用コアと同じであるが、他のユニットやフロントエンドとのメッセージ送受信機能を持つこと、独自の 64kByte のインストラクションメモリを持つこと、フロントエンドとのデータ受け渡しのための 32kByte のバッファメモリへアクセスできることが異なる。

4. ソフトウェア

MDGRAPE-3 までは二体力専用パイプラインを汎用計算機 (ホスト計算機) に接続するアクセラレータ型であり、専用パイプラインは関数形を決定するパラメータ等が変更可能なだけで CPU のようなプログラムを実行する機能はなかった。MD プログラムはホスト計算機上で実行された。実行開始時に MDGRAPE のパラメータ設定をする、メインループ中の二体力計算部分を MDGRAPE を呼び出す関数に差し替える、といった変更のみで、既存の MD プログラムを使用していた (図 3 上)。

MDGRAPE-4 ではプログラムを実行できる汎用コアを搭載しており、専用パイプラインで計算する二体力以外の計算も実行できる。設計コストと計算性能の観点から MD 計算の実行のみを想定した設計で、フロントエンド計算機とのデータ転送以外の I/O 機能は持たず、OS も搭載しない。そのため、ユーザーインターフェース、ファイル入出

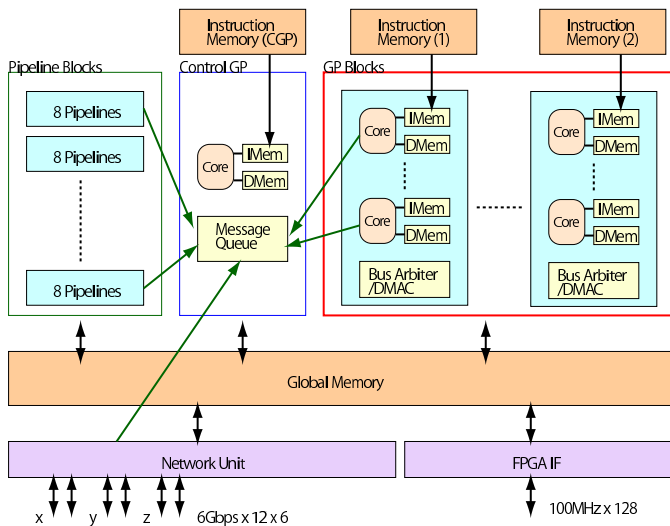


図 2 MDGRAPE-4 SoC

8x8 パイプライン (Pipeline Blocks)、8x8 汎用コア (GP Blocks)、ネットワークインターフェース (Network Unit)、コントロール汎用コア (Control GP)、グローバルメモリ (Global Memory) で構成されている。メッセージの送受信でコントロール汎用コアが他のユニットを管理する (緑線)。[8] Figure4

力等のメインループ以外の処理はフロントエンド計算機で実行し、MDGRAPE-4 では転送された初期条件・データをもとにメインループのみを実行する (図 3 下)。メインループ中の計算のうち、二体力計算、Charge Assignment、Back Interpolation はパイプラインが担当し、FFT、結合力、leap-frog は汎用コアが担当する。周辺チップとの通信はネットワークインターフェースが行う。メインループの制御、各処理の開始指示や終了確認はコントロール汎用コアが行う。分子動力学計算ソフトウェアは Gromacs[9] を選択した。フロントエンド実行部分はそのままだが、MDGRAPE-4 にオフロードするメインループ部分は大幅な変更が必要である。そのため Gromacs の開発チームと共同で MDGRAPE-4 用のメインループ部分の開発を行なっている。

4.1 メインループ

パイプラインが担当する計算は通常のソースコードではなく、目的の計算が行われるパラメータを設定してパイプラインを起動するコードとなり、汎用機用コードとは完全に異なるものとなる。

汎用コアは I/O を伴わなければ汎用機とほぼ同じコードを実行できるが、コード領域に入りきるようにサイズを切り詰める必要がある。そのため、計算手法やモデルを選択する機能は省略し、特定の手法・モデルのみを実装する。算術関数も入力範囲や精度を制限し、最小限の逐次近似や補完多項式を用いて、高速化とコードサイズの縮小を行なう。パイプラインの入出力が固定小数点であるため、浮動小数点との相互変換、汎用コアでの計算の一部も固定小数

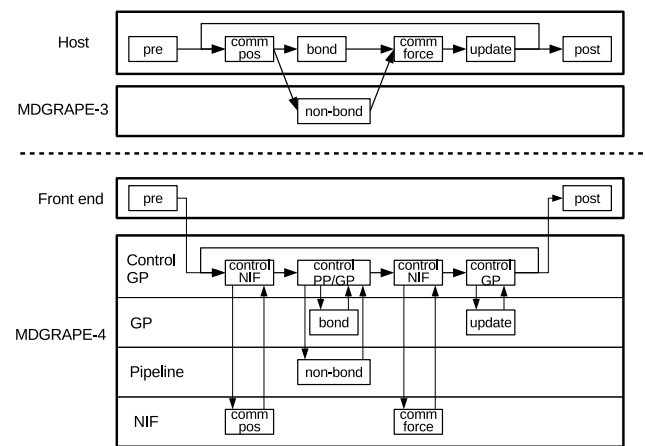


図 3 MD 計算の流れ

上: MDGRAPE-3 の場合、二体力計算 (non-bond) のみ MDGRAPE-3 が処理し、その他の部分は汎用のホスト計算機上で処理される。

下: MDGRAPE-4 の場合、ファイル入出力のような前処理・後処理をフロントエンド計算機で行ない、メインループが MDGRAPE-4 で処理される。MDGRAPE-4 の内部ではループ全体のフロー制御はコントロール汎用コア (Control GP) が行ない、結合力 (bond) と積分 (update) は汎用コア (GP) が、二体力計算はパイプライン (Pipeline) が、座標・力の通信 (comm pos/force) はネットワークインターフェース (NIF) が処理する。

(メモリアクセスや PME/GSE 処理は省略)

点演算化するといった変更も必要となる。

ネットワークインターフェースはメモリ上の通信コマンドに従って DMA を用いて自律的に通信を行なうが、コントロール汎用コアからメッセージ送信で通信コマンドの実行開始し、通信完了はメッセージ受信でコントロール汎用コアが検出する。メッセージ送受信でネットワークインターフェースを制御するコントロール汎用コア用のコードが必要となり、MPI のような高度な API で記述された汎用機コードとは完全に異なるものとなる。

依存関係に基づいて逐次処理させたりオーバーラップさせて並列処理させるような制御もコントロール汎用コアのコードで行なう。

4.2 データ構造

MDGRAPE-4 は 512 チップ、チップ内のパイプラインや汎用コアが 8 個 x 8 ブロックと多階層で並列化されている。8 ブロック間はグローバルメモリを単純に共有するだけでコヒーレンス等の機能がないため、ソフトウェア的にデータの分割と管理を行なう必要があり、汎用機のようなスレッド並列とは大きく異なる。また、パイプラインのバッファ長の 16 原子で最小分割単位が制限される。これらの理由から、MDGRAPE-4 の階層構造やデータ長に特化したデータ構造とデータ分割方式を検討する必要がある。

5. 開発状況

5.1 ハードウェア

現在、8チップボードの動作検証を進めている。チップ単体の検証としては、パイプライン等の個々のユニットの基本的な動作の確認、さらに、水分子600原子程度の単純化したカットオフ計算をコントロール汎用コア、汎用コア、パイプラインを連動させて実行できることを確認した。チップ間通信は、物理層に近い部分のパラメータ設定等の調整をし、データの送受信が正常に行われることを確認した。2x2x2のトーラスとなるようにボード間通信ポートを接続することで、光インターフェースを経由した通信に問題がないことを確認した。

5.2 ソフトウェア

新規に書き起こした単純化したカットオフ計算のメインループをMDGRAPE-4に移植した。まず、汎用機用コードと同じコードが汎用コアで動作することを確認した。次に、二体力計算、結合力計算、leap-frogといった単位の関数を、コントロール汎用コアが開始メッセージを送信することで汎用コアが実際の計算を行なう形に変更し、コントロール汎用コアによるメインループ制御と計算部分の汎用コアへのオフロードを確認した。続いて、二体力計算をパイプラインに行なわせるコードと合わせて、コントロール汎用コア、汎用コア、パイプラインでメインループ中の計算を進める最低限の基本コードを動作させた。汎用コアによるFFTは基本コードの開発を外注している。

5.3 性能評価

現状では部分的なMDコードしか移植できていないため、主にユニット毎の個別の性能を測定した。MDGRAPE-4のSoCにはパフォーマンス測定用のカウンターやリアルタイムクロックがないため、測定対象のユニットからの完了メッセージをポーリングするループの回転数から諸費したクロックサイクルを推定した。

1パイプラインブロック(8パイプライン)による600原子の計算は約30%の効率であった。データ構造・データ分割のコードが開発中のため実際の計算条件に近い測定は今後の課題であるが、効率30%であれば目標とする数万原子規模の計算には30 μ s程度かかる見込みである。charge assignment、back interpolationはそれぞれ10 μ sかかる演算量である。

通信速度は実測でレイテンシ600ns、転送速度2.6GByte/sであった。また、ホップ数あたりのレイテンシは450nsであった。目標原子数での近接通信量では座標等のデータ・力の交換に5 μ s・3 μ sかかる見込みである。

汎用コアによる結合力(結合距離、結合角のみ)の計算は10-20%の効率であった。ねじれ角も含めた典型的な

タンパク質の結合力の計算量を仮定すると、目標原子数では8 μ sかかる見込みである。積分計算は1 μ s程度になる。FFTはまだ実測できていないが、32³のFFTを64ノードで実行する場合の演算量からは実効速度で4 μ sとなる。転置通信を含めると波数空間計算では35 μ sかかる見込みである。

オーバーラップも考慮すると、最も時間がかかるパスはcharge assignment、二体力、back interpolationで、合計50 μ sとなり、パイプラインの実行効率を改善する必要がある。

6. 結論と今後の予定

現時点でのSoCの単体性能評価では50 μ s/step程度となる見込みで目標の倍の実行時間となっており、ミリ秒シミュレーションには半年かかることになる。ハードウェアの製造、ソフトウェアの開発途中であるが、最適化も同時に進めて実行時間を短縮する必要がある。

6.1 ハードウェア

まず、ボード二枚を接続してボード間通信の確認を行なう。これにより、ボードの全ての機能に問題がないことが確認されるので、64ボード製造する。その後、ボードの製造状況に応じてより多数の接続での動作確認を行なっていく。8ボードの時点で全方向がボード間接続された状態が検証できる。最終的に64ボード512チップが動作してハードウェアは完成となる。

6.2 ソフトウェア

MDGRAPE-4用のデータ構造への変更を行なう。これには、原子データを空間分割で再配置するコードの開発を伴う。続いて、このデータ構造でグローバルメモリに配置された原子データを使ってパイプラインで二体力計算させる。データの配置状況を管理し、適切にパイプライン設定をするコントロール汎用コア用コードが必要となる。汎用コア用コードでは、固定小数点で演算する範囲の決定と当該処理の固定小数点演算化、浮動小数点処理部との間でのデータ変換等を実装する。発注したFFTの基本コードを汎用コアで確認、最適化を進める。

ネットワークインターフェースには新しいデータ構造と空間分割情報に基づいて必要な隣接袖通信を行なうコードを開発する。隣接通信そのものは自立的に行われるので、コントロール汎用コアで転送元・転送先メモリアドレスを管理するコードとなる。FFT中の転置通信のコードは新規に開発する必要がある。

6.3 最適化

現時点の測定では初期メモリ配置で均等分割したデータを順次処理するだけの最も単純はパイプラインコマンドと

なっている。実際の計算では、空間分割でのグループ分けしカットオフを考慮することで、計算するグループペア数自体の削減や、グループペア中のカットオフヒット率を向上させる余地がある。また、パイプラインのロード・ストアと計算をオーバーラップさせる最適化も未実装である。現状の実測値ではパイプラインの処理時間で全体の時間かが決まっており、性能向上のためにはパイプラインの実行効率を上げることが必須である。

謝辞

本研究は日本学術振興会最先端研究基盤事業および the G8 Research Councils Initiative for Multilateral Research Funding の支援により実施されている。

GROMACS の移植は Prof. Erik Lindahl, Dr. Berk Hess, Dr. Szilárd Páll, and Dr. Mark Abraham (Stockholm University) と共同で行なっている。

参考文献

- [1] Andoh, Y., Yoshii, N., Fujimoto, K., Mizutani, K., Kojima, H., Yamada, A., Okazaki, S., Kawaguchi, K., Nagao, H., Iwahashi, K. et al.: MODYLAS: A Highly Parallelized General-Purpose Molecular Dynamics Simulation Program for Large-Scale Systems with Long-Range Forces Calculated by Fast Multipole Method (FMM) and Highly Scalable Fine-Grained New Parallel Processing Algorithms, *Journal of Chemical Theory and Computation*, Vol. 9, No. 7, pp. 3201–3209 (2013).
- [2] Darden, T., York, D. and Pedersen, L.: Particle mesh Ewald: An $N \log(N)$ method for Ewald sums in large systems, *J. Chem. Phys.*, Vol. 98, pp. 10089–10092 (1993).
- [3] Ebisuzaki, T., Makino, J., Fukushige, T., Taiji, M., Sugimoto, D., Ito, T. and Okumura, S. K.: GRAPE project: An overview, *Publ. Astron. Soc. Japan*, Vol. 45, pp. 269–278 (1993).
- [4] Ewald, P. P.: Ewald summation, *Ann. Physik*, Vol. 64, pp. 253–371 (1921).
- [5] Fukuda, I., Yonezawa, Y. and Nakamura, H.: Molecular dynamics scheme for precise estimation of electrostatic interaction via zero-dipole summation principle, *J. Chem. Phys.*, Vol. 134, No. 16, p. 164107 (2011).
- [6] Narumi, T., Ohno, Y., Okimoto, N., Koishi, T., Suenaga, A., Futatsugi, N., Yanai, R., Himeno, R., Fujikawa, S., Taiji, M. et al.: A 55 TFLOPS simulation of amyloid-forming peptides from yeast prion Sup35 with the special-purpose computer system MDGRAPE-3, *Proceedings of the 2006 ACM/IEEE conference on Supercomputing*, ACM, p. 49 (2006).
- [7] Ohmura, I., Morimoto, G., Ohno, Y., Hasegawa, A. and Taiji, M.: MDGRAPE-4: a special-purpose computer system for molecular dynamics simulations, *Phil. Trans. R. Soc. A* (accepted).
- [8] Ohno, Y., Yokota, R., Koyama, H., Morimoto, G., Hasegawa, A., Masumoto, G., Okimoto, N., Hirano, Y., Ibeid, H., Narumi, T. et al.: Petascale molecular dynamics simulation using the fast multipole method on K computer, *Computer Physics Communications* (2014).
- [9] Pronk, S., Pall, S., Schulz, R., Larsson, P., Bjelkmar, P., Apostolov, R., Shirts, M. R., Smith, J. C., Kasson, P. M., van der Spoel, D., Hess, B. and Lindahl, E.: GROMACS 4.5: a high-throughput and highly parallel open source molecular simulation toolkit, *Bioinformatics*, Vol. 29, No. 7, pp. 845–854 (2013).
- [10] Shan, Y., Klepeis, J. L., Eastwood, M. P., Dror, R. O. and Shaw, D. E.: Gaussian split Ewald: A fast Ewald mesh method for molecular simulation, *J. Chem. Phys.*, Vol. 122, No. 5, p. 54101 (2005).
- [11] Shaw, D. E., Deneroff, M. M., Dror, R. O., Kuskin, J. S., Larson, R. H., Salmon, J. K., Young, C., Batsion, B., Bowers, K. J., Chao, J. C., Eastwood, M. P., Gagliardo, J., Grossman, J. P., Ho, C. R., Ierardi, D. J., Kolossváry, I., Klepeis, J. L., Layman, T., McLeavey, C., Moraes, M. A., Mueller, R., Priest, E. C., Shan, Y., Spengler, J., Theobald, M., Towles, B. and Wang, S. C.: Anton, a special-purpose machine for molecular dynamics simulation, *Proceedings of the 34th annual international symposium on Computer architecture*, ISCA '07, New York, NY, USA, ACM, pp. 1–12 (online), DOI: 10.1145/1250662.1250664 (2007).
- [12] Taiji, M., Narumi, T., Ohno, Y., Futatsugi, N., Suenaga, A., Takada, N. and Konagaya, A.: Protein Explorer: A Petaflops Special-Purpose Computer System for Molecular Dynamics Simulations, *Proceedings of the 2003 ACM/IEEE conference on Supercomputing*, SC '03, New York, NY, USA, ACM, p. 15 (online), DOI: 10.1145/1048935.1050166 (2003).
- [13] Wolf, D., Keblinski, P., Phillpot, S. and Eggebrecht, J.: Exact method for the simulation of Coulombic systems by spherically truncated, pairwise r summation, *J. Chem. Phys.*, Vol. 110, pp. 8254–8282 (1999).
- [14] Wu, X. and Brooks, B. R.: Isotropic periodic sum: a method for the calculation of long-range interactions, *J. Chem. Phys.*, Vol. 122, No. 4, p. 44107 (2005).