

jh170024-NAH

分子動力学計算ソフトウェア MODYLAS の 大規模メニーコア・ワイド SIMD クラスタ対応並列化に関する研究

安藤 嘉倫 (名古屋大学工学研究科附属計算科学連携教育研究センター)

概要 本課題では平成 27, 28 年度 JHPCN 課題に引き続き、汎用分子動力学計算ソフトウェア MODYLAS について次世代のメニーコアおよびワイド SIMD クラスタ (FX100 および Oakforest-PACS) の性能を発揮させるための並列化チューニングを実施した。ポスト「京」を見据えて、分子動力学計算のホットスポットである粒子対計算および FMM 計算部分について、512bit SIMD を備えたハードウェア (Xeon Phi, Knights Landing (KNL)) 上での SIMD 処理性能評価およびコード最適化を行った。昨年度計測に使用した Xeon Phi (Knights Corner, KNC) に比べ、KNL では同じスレッド数で比較した際の処理速度に 5 倍程度の向上が見られたものの、Intel プロセッサ向けの標準的なプロファイラである VTune ではコードの詳細な性能測定が困難でありそれ以上のコード最適化が阻まれていた。しかしながら KNL と同様 512bit SIMD (AVX-512 命令) に対応した Skylake-SP 上で性能分析ライブラリ PMlib を用いることにより、ifort コンパイラにより発行される SIMD 命令の詳細解析を行うことに成功した。これにより、我々の開発したコードが 512bit SIMD に対応できていることの確認と、それが性能にどのように影響するかについての情報を得ることに成功し、今後のワイド SIMD 環境に向けた知見を得ることができた。

1. 共同研究に関する情報

(1) 共同研究を実施した拠点名

名古屋大学情報基盤センター
東京大学情報基盤センター

(2) 共同研究分野

■ 超大規模情報システム関連研究分野

(3) 参加研究者の役割分担

全体統括 : 安藤

MPI 並列化性能統括 : 荻野

分子動力学計算のアルゴリズム開発 :

安藤, 吉井, 藤本, 篠田, 岡崎

並列化コーディング (MD 計算全般) :

安藤, 坂下, 藤本

プログラム SIMD 性能評価 : 大島, 片桐

Xeon Phi 向け並列化コーディング : 大島

自動性能チューニング技術提供 : 片桐

並列化コーディングおよび並列アルゴリズム
開発 (OpenMP, SIMD) : 鈴木, 坂下

析ツールとして広く普及している。加えて工業分野においても分子の特性を活かしたナノ機能性材料や高分子材料を設計する際に MD 計算により得られる知見が不可欠になりつつある。分子内・分子間の相互作用には様々あるが、分子集団系の物性をシミュレーションにより正しく再現するには長距離性のある静電相互作用を精度よく計算する必要がある。しかしながら、長距離原子間静電相互作用を含めた実用の研究において MD 計算であつかえる原子数および計算時間は、「京」コンピュータといった最新鋭のスーパーコンピュータを用いたとしても 1 千万原子系 (空間サイズとして 50 ナノメートル立方程度) に対する数 100 ナノ秒の計算^[1]が限界である。より大規模かつ長時間な MD 計算を行うことで、上記学問分野におけるブレークスルーが期待できるだけでなく、より高精度な材料設計が可能になると期待される。

本研究では、昨年度に続き汎用分子動力学計算ソフトウェア MODYLAS^[2] に対して FX100 および Oakforest-PACS といった最新のメニーコア・ワイド SIMD 型の計算機、さら

2. 研究の目的と意義

研究の目的 現在分子動力学 (MD) 計算は、化学、物理、生物、およびウイルス学といった様々な学問分野において実験とならぶ解

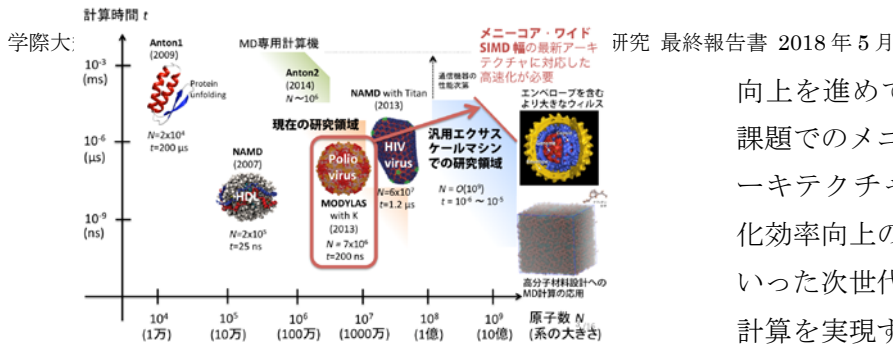


図 1 長距離静電相互作用計算を含む MD 計算で扱う原子数および時間スケールの変遷。

には将来のエクサスケールマシンの性能を發揮させるためのスレッド並列およびワイド SIMD, 特に 512bit SIMD を備えた最新の Xeon Phi Knights Landing (KNL) への最適化チューニングを行う。これら成果を有効に活用して, 将来的にポスト「京」コンピュータなどのエクサスケールマシンと MODYLAS によって数億から 10 億原子系での 1 μ 秒を超えた実用的な MD 計算を目標とする (図 1)。

研究の意義 スーパーコンピュータの進化はノード数の増加が頭打ちになり, ノード当たりのコア数および SIMD ベクトル長 (SIMD 幅) を増加させシステム全体の演算性能を向上させる方向にある。SIMD 幅は「京」において 128bit であったのに対し, FX100 では 256bit, 最新の Xeon Phi (Knights Landing (KNL)) では 512bit ある。ポスト「京」についても ARM アーキテクチャを用いた 512bit SIMD が採用される予定であることが発表されている。これら次世代のマシンでは, コア数および SIMD 幅の増加によって現行のプログラムの各所でスレッド並列性および SIMD 並列性が確保できなくなる。一方, 分子動力学ソフトウェア MODYLAS は「京」コンピュータの全ノード規模での MPI 並列計算に対応したチューニングが施されており, 平成 27 年度および 28 年度 JHPCN 課題において FX100 の 32 コアでの OpenMP 並列化チューニング, さらには Xeon Phi Knights Corner (KNC) を用いた 240 コアまでのスレッド並列性能検証, および準ホットスポットの性能

3. 当拠点公募型共同研究として実施した意義

メニーコア・ワイド SIMD といったハードウェアの日進月歩の進化に MD 計算ソフトウェアが対応していくためには, 最新のアーキテクチャの基本性能および対応コーディングに詳しいコンピュータ・サイエンスの研究者と MD シミュレーションを用い実際の研究を実施している者とが協力した学際領域分野の研究が不可欠である。本課題では両分野の専門家が共同研究者として参画している。本課題に参加するコンピュータ・サイエンスの研究者はハイパフォーマンスコンピューティング全般を専門とし, 特に自動性能チューニング(AT)に詳しい研究者 (片桐), 最新のメニーコア技術に詳しい研究者 (大島), 並列化技術全般に詳しい研究者 (鈴木, 坂下), および MPI 並列化技術に詳しい研究者 (荻野) から構成される。複合・階層的な最新のメニーコア・ワイド SIMD 型の計算機システムを使いこなす上で不可欠な人員構成である。一方, MD シミュレーションの研究者は, MD 計算ソフトウェア MODYLAS の基本設計に最初から関与しかつ MD 計算の基本原理に詳しい研究者 (安藤, 藤本, 吉井, 岡崎) および現行の MODYLAS 並列化の内容に詳しい研究者 (安藤, 藤本, 坂下) および粗視化 MD 計算に詳しい研究者 (篠田) から構成される。両者の協業により, プログラミングレベルでの並列性能チューニングだけでなく, 必要に応じて MD 計算のアルゴリズムレベルでの性能向上も期待される。

4. 前年度までに得られた研究成果の概要

MODYLAS では長距離静電相互作用計算に高速多重極展開法(図 2)を用いている. 過去 2 年間の JHPCN 課題において, 主に FX100 を用い, MD 計算における以下のホットスポット二箇所,

(1) Lennard-Jones 相互作用および静電相互作用の粒子対計算 **p2p**

(2) 高速多重極展開法(FMM)での多極子から局所展開係数への変換 **M2L**

についてのメニーコア対応スレッド並列最適化を行い顕著な性能向上を得た. その詳細は JHPCN 最終報告書^[3,4]に譲る. 概要は, (1) の **p2p** についてオリジナルコードではスレッド並列対象 **do** ループ長不足 (表 1) により大きな負荷不均衡が生じていたところ, 新たに 4 つの並列化手法 (その 1: 該当 **do** ループ長の伸張, その 2,3: スレッド並列化位置の変更, その 4: それらの融合的方法) を提案することにより, スレッド間不均衡を 1%以下と劇的に縮小することができた. 同時にスレッド単体性能についても向上させ, 表 2 に示す顕著な性能向上を達成した. これら成果を J. Supercomput.誌に公刊した (7.成果リストの学術論文[2]). (2) の **M2L** については, **FMM** の 3 階層目以降での処理において不適切なチャンクサイズ設定により大きなスレッド間負荷不均衡が生じていたところ, チャンクサイズの最適化により 3 階層目以上においても均一な並列性を確保できるようになり高速化が達成された. **SIMD** 最適化についても, **p2p** および **M2L** のループ構成を活かすことで, FX100 の 256bit **SIMD** においてそれぞれ 95%, 85%以上の非常に高い浮動小数点命令演算率が達成された.

主に FX100 を用いたスレッド並列を含むこれらホットスポット二箇所の高速化により, 準ホットスポットであった **p2M**, **L2p** および二面角相互作用の計算などが律速段階となった. これら箇所での **SIMD** 化率が 0.6%,

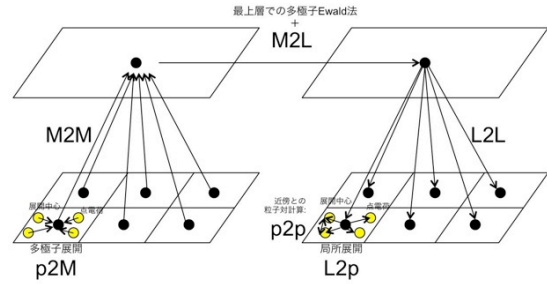


図 2 高速多重極展開法 (FMM)の主要演算. **p2p** および **M2L**のほか, 点電荷から多極子展開係数への変換 **p2M**, 多極子のマージ **M2M**, 多極子 Ewald 法, 局所展開中心の移動 **L2L**, および局所展開係数をもちいた点電荷上の電場の計算 **L2p** から構成される.

N×N×N: 計算領域セル数, N: スレッド数
セル内の原子数を40として評価

	粒度 (計算原子対数)	スレッド数Nの上限			
		N×N×N	2×2×2	4×4×4	8×8×8
オリジナル	cell内分割 8000/N _i	40	40	40	40
改良その1	column内分割 8000N _i	40N	80	160	320
その2	cell単位 8000	25N ²	200	1600	12800
その3	square単位 8000 ² ~200000	N(N+4)(N+4)	72	256	1152
その4	column単位 8000N	25N ²	100	400	1600

※ MPI領域境界ではi-squareがクリッピングされ, 有効なセル数が25から最低1まで減少する

表 1 **p2p** スレッド並列における粒度およびスレッド数上限. この表中での **N**は, 軸当たり担当サブセル数.

16コア実行 (1CMGIに1MPIプロセス)

	秒	GFLOPS	ピーク比	L1Dミス率	L2ミス率	SIMD化率
オリジナル	6.20	130.4	23.1%	0.40%	0.00%	96.56%
改良その1	4.83	168.2	29.9%	0.14%	0.00%	96.56%
その2	4.23	192.2	34.1%	0.25%	0.00%	96.56%
その3	4.02	200.9	35.7%	0.20%	0.00%	96.56%
その4	4.11	197.0	35.0%	0.08%	0.00%	96.56%

32コア実行 (1ノードに1MPIプロセス)

	秒	GFLOPS	ピーク比	L1Dミス率	L2ミス率	SIMD化率
オリジナル	4.74	170.4	15.1%	0.45%	0.02%	96.56%
改良その1	3.14	257.1	22.8%	0.30%	0.02%	96.55%
その2	2.87	284.9	25.3%	0.22%	0.04%	96.56%
その3	2.21	365.9	32.5%	0.09%	0.01%	96.56%
その4	2.32	349.8	31.1%	0.12%	0.02%	96.56%

GFLOPS値はMPIプロセスあたり, SIMD化率= SIMD浮動小数点演算命令率

表 2 FX100 での **p2p** 性能比較. オリジナルおよび改良コードでの 16および 32 スレッド実行. 表 1 での **N=4** の条件で計測.

15.1%, 0.3%と非常に低いことが性能のボトルネックであった. 詳細は最終報告書^[4]に譲るとして, **IF** 文の削除, **SIMD** 化対象ループの伸長, プリファクター (定数) 計算部分の分離などにより, **SIMD** 化率をそれぞれ 49%,

55%, 69%に向上させることに成功した。結果としてMD計算プログラム全体の計算時間を25%程度削減することに成功した。さらにFX100で最適化したコードをXeon Phi (研究用KNC クラスタ) に移植し, 120 スレッドまで負荷均衡化および高速化を達成した。

しかしながら KNC では, 負荷均衡化したにも関わらず実際の計算時間が減少しない, さらにその原因を探るための詳細な性能プロファイルを KNC では採取できないなどの問題があった。そこで, 今年度正式運用を開始した最新の KNL で構成されたOakforest-PACS 上において詳細な性能プロファイルを取得し, メニーコアおよび 512bit SIMD へのさらなるコード最適化を進めるべく本年度の JHPCN 課題を提案した。

5. 今年度の研究成果の詳細

今年度は Oakforest-PACS の実機を用い, 主に p2p について KNL 上での性能評価を進めた。昨年度と同様独自タイマー挿入によりホットスポットでの経過時間を測定した。

図 3 にあるように, 同じスレッド数で比べた場合 KNL での p2p 実行時間は KNC にくらべ 1/5 倍程度に縮小された。KNL 上でのコード間の優劣をみると, 16 スレッドまではコードその 3 (図中 code3) がもっとも優れているのに対し, 32 スレッド以上ではその 1 (図中 code1) が優勢となった。このスレッド数の増加につれ code1 が優勢になる傾向は KNC と KNL とで変わらなかった。また, 64 スレッド以上の実行ではオリジナルコード, およびコード 2~4 の性能が大幅に劣化した。128 スレッド以上の実行ではハイパースレッディングが利用されており, p2p 処理は KNL 上におけるハイパースレッディング実行には適していないことが確認された。

しかしながら表 1 にあるように, 取り得るスレッド数上限 N_t の観点からは 32 スレッド以上であってもコードその 2~4 が優勢である

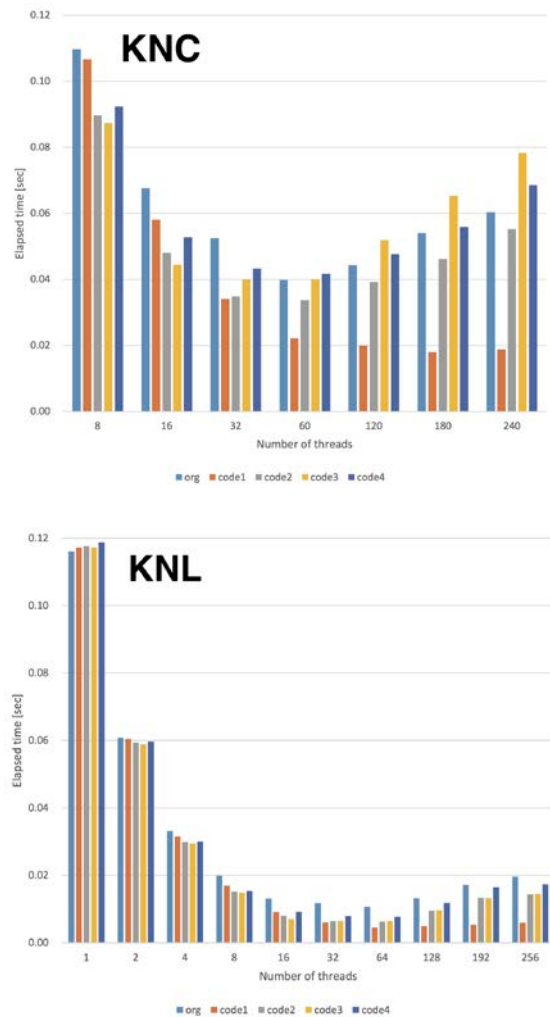


図 3 p2p の KNC および KNL でのベンチマーク結果。横軸スレッド数, 縦軸はタイマー挿入により測定した実行時間。 $N=4$ 。

べきである。実際, 表 2 にあるように, FX100 での測定では 32 スレッド実行時コードその 3 が最速になっている。すなわちスレッド数が大きい場合に FX100 と KNL で code1-4 間での性能についての傾向が異なる。その原因を探るべく, スレッドごとの実行時間および SIMD 化率について VTune プロファイラによる調査を試みた。

結果として Oakforest-PACS 上で VTune でのプロファイル取得自体は行えたものの, コード 2~4 の性能が低下している理由付けになるようなプロファイル結果を得ることはできなかった。その理由には, 測定対象の p2p 一回あたりの実行時間が数 10 ms ないし数 ms と非常に短いこと, および FX100 上での

富士通社製の詳細プロファイラ `fapp` による測定に比べて `VTune` による測定では測定可能項目に制限が多く性能情報を把握しにくいことがある。入力問題規模の再設定や `VTune` の設定を調整しコード間の性能差の裏付けとなるプロファイル結果が取得できないかを継続的に調査しているが、これは我々のコード固有の問題ではなく `VTune` を用いた測定一般の問題であり、解決は容易ではなさそうである。また代替案として `Oakforest-PACS` にインストールされている `Allinea` 社製のツール `Forge` の利用も検討したが、デバッガである `DDT` のライセンスは確認できたものの、プロファイラである `Forge` はライセンスがないため利用できないことがわかった。利用するプロセッサにより性能解析の難しさに大きな差があることについては仕方がないが、利用者としては、利便性の大きな低下がないようハードウェアベンダーやスパコンセンターにも注意を払ってもらいたい。

さらなる代替案として、九州大学情報基盤研究開発センターで試験運用が開始された ITO システム上の `Xeon Gold (Skylake-SP, 18 コア)` 上での性能評価を試みた。`Skylake-SP` は `KNL` 同様に 512bit SIMD に対応している(ただし完全に同一の命令セットではない)ため、`Oakforest-PACS` 上での SIMD 化の状況について参考になる可能性を期待したものである。ITO も「京」や `Oakforest-PACS` 同様に富士通が導入ベンダーであるが、ITO で利用可能な富士通社製のコンパイラ等は `Skylake-SP` 向けの 512bit SIMD 命令に対応しておらず、「京」上で実績のある性能解析ツールを用いることはできないことが確認された。そこで我々は理研 AICS の開発した性能分析ライブラリ `PMLib` を ITO へ独自にインストールして `ifort` コンパイラの発行する SIMD 命令の詳細を解析し図 4 に示すような結果を得ることができた。なお現状 `PMLib` は

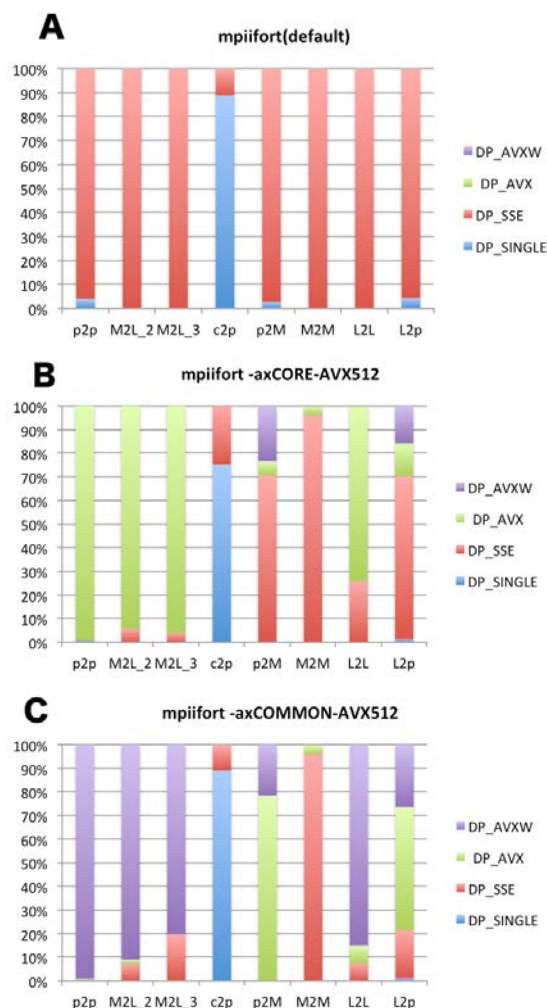


図 4 PMLib を用いた Skylake-SP 上での SIMD 命令の解析結果. A-C は `ifort` の最適化オプションを変えて測定している. A: デフォルト設定, B: `ifort -axCORE-AVX512`, C: `ifort -axCOMMON-AVX512`. 図中 `DP_SSE`, `DP_AVX`, `DP_AVXW` はそれぞれ倍精度 128bit, 256bit, 512bit SIMD 命令.

`KNL` に対応していないため、`Oakforest-PACS` 上で同種の測定を行うことはできない。

図 4A は ITO にインストールされた `ifort` において特別な最適化オプションを設定しない場合 (`-O2` 相当) のコンパイル結果であり、`IF` 分岐の多く含まれる `c2p` (デカルト座標から極座標への変換操作) 以外のコードにおいて `DP_SSE` (倍精度 128bit SIMD 命令) が多く発行されているとわかる。さらに 512bit SIMD 命令を出力するオプション

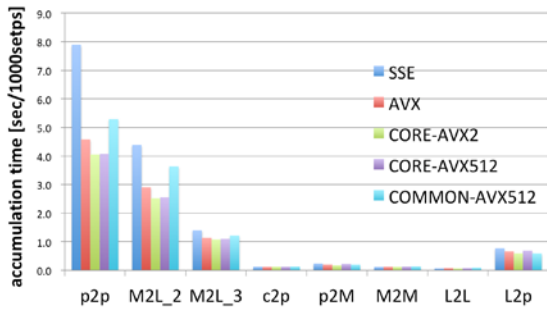


図 5 コードごとコンパイルオプションを変えた際の実行時間。

-axCORE-AVX512 を加えた結果が図 4B である。ホットスポットの p2p および M2L において DP_SSE が DP_AVX (倍精度 256bit SIMD 命令) に置き換えられたが、DP_AVXW (倍精度 512bit SIMD 命令) は p2M および L2p で一部発行されるに留まった。この時点でのコンパイラ最適化メッセージ(.qopt-report) から DP_AVXW 命令を発行できる可能性のある箇所が含まれることがわかったため、同命令をより強制的に生成するオプション -axCOMMON-AVX512 を加えた結果が図 4C である。ホットスポットである p2p および M2L において DP_AVX が DP_AVXW に置き換えられ、かつその割合が 90% 以上になった。

その一方で図 5 に示すように実行時間についてはほとんどの場合 -axCORE-AVX512 を加えた場合に最小になった。コンパイラへの -axCOMMON-AVX512 オプションにより DP_AVXW 命令を強制的に発行した場合に実行時間が減少する原因の 1 つとしては、Skylake-SP の仕様として DP_AVXW 命令発行時にクロック周波数が低下することが挙げられる (Xeon Gold 6154 では Turbo Boost が効いていない状態で通常時 3.0 GHz, AVX2 2.6 GHz, AVX512 2.1 GHz, Turbo Boost が効いている状態でも同程度に差が生じる)。図 4C での DP_AVXW 命令による実行時間の削減ぶんがクロック数低下による実行時間増分を下回るために図 5 での実行時間が増大した

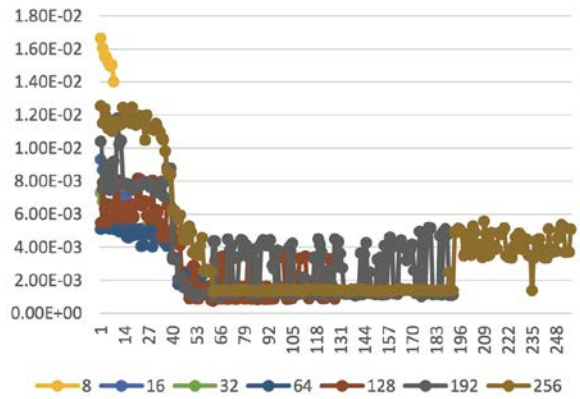


図 6 タイマー挿入による KNL でのスレッドごと実行時間測定結果 (オリジナルコード). N=4.

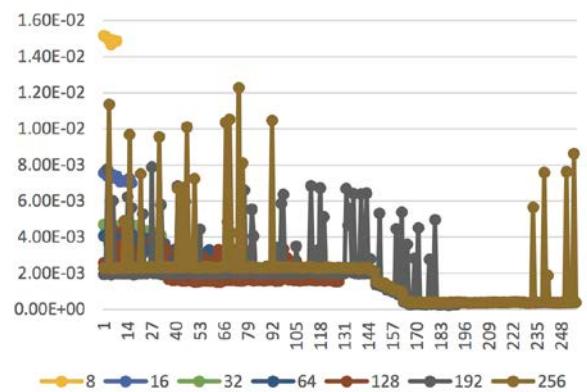


図 7 タイマー挿入による KNL でのスレッドごと実行時間測定結果 (コードその 1). N=4.

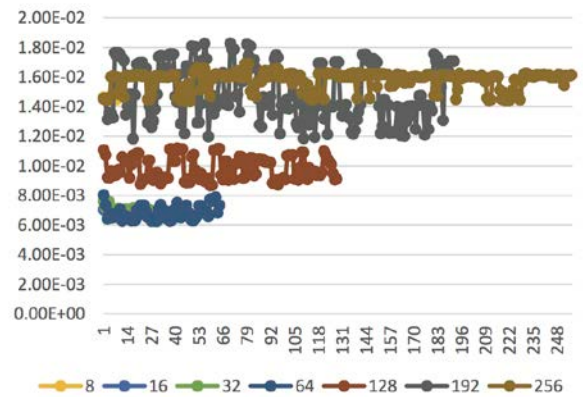


図 8 タイマー挿入による KNL でのスレッドごと実行時間測定結果 (コードその 3). N=4.

ものと考えられる。

また p2p についての最適化メッセージから -axCOMMON-AVX512 オプションを付けた際に peel ループも SIMD 化してしまっていることが速度低下の原因であることがわかった。オプションを -axCORE-AVX512 に戻した

上で該当ループの前にループ長ごとに SIMD 命令発行を制限するディレクティブ `!$OMP SIMD SIMDLLEN(16)` をくわえた結果、`DP_AVXW` 命令の発行率を 90% 以上に保ったまま実行時間を、`-axCORE-AVX512` 指定した場合に及ばないものの、同等程度まで削減することができた。さらに `M2L`, `p2M` および `L2p` についても `-axCORE-AVX512` とディレクティブ指定の組み合わせにより、`-axCORE-AVX512` を指定した場合よりもさらに短い実行時間を高い `DP_AVXW` 命令率とともに実現することができた。

以上から、我々の開発したコードは 512bit SIMD 命令に十分対応したものとなっていることや、ハードウェアの制約によりコードによっては 512bit SIMD 命令を利用することが必ずしも最大性能には結びつかないことが確認された。`PMlib` により図 4 に示すような SIMD 命令の詳細な解析結果が得ることで初めてこのような考察が可能になり、今後ワイド SIMD 環境にてコードを最適化するための指針を得ることができた。

一方、`KNL` 上でスレッドごとの実行時間を測定した結果を図 6~8 に示す。図 6 に示すオリジナルコードでは N_t の不足 ($N_t \sim 40$ 程度) により 64 スレッド以上で性能が飽和していることがわかる。図 7 に示すコードその 1 では $N_t \sim 160$ 程度までスレッド並列性能が保持される一方、実行時間 (1~3 ms) の 10 倍程度のスパイク状の遅延が不規則に観測され、これらの遅延が `p2p` を律速している可能性が明らかになった。このスパイクの原因を探るべく `VTune` によるプロファイルの取得と性能分析を試みてきたが、実行時間の短さや取得できる情報の不足からその原因を特定するには至っていない。しかしながら、これらスパイクを除くことが出来れば実行時間として数 ms が実現できることになり、来年度以降も引き続き原因究明に務める。

さらに、図 8 に示すコードその 3 では $N_t \sim$

256 であることから、256 スレッド実行での負荷均等化が実現されているにも関わらず、128 以上のスレッド数での実行時間は逆に増加した (これは `KNC` でも観測された^[4])。コードその 2, 4 においても同様な問題が生じることで性能が劣化しており、その原因を探るべく考察や詳細なプロファイル情報の取得を試みている。

6. 今年度の進捗状況と今後の展望

本年度は 4 期に分け研究を進める計画であった。第一期 (4-6 月) において `Oakforest-PACS` 上でのコード事前評価 (スレッド並列, SIMD), 第二期 (7-9 月) においてスレッド並列に関するコード改良を進めてきた。図 3 にあるように、`MD` 計算でのホットスポット `p2p` について `KNC` にくらべ `KNL` では 5 倍程度高速に計算できることが確認された。その一方、32 スレッド以上での実行時に性能劣化が確認された。詳細な解析によりその原因は図 7 にあるようなランダムなスパイク状の遅延、および図 8 にみられる不可解な現象 (負荷均等化にも関わらず実行時間が低下) にあることがわかった。その原因を突き止めるべく `VTune` プロファイラを用いての性能調査を試みてきたが、`VTune` により得られる情報は各コードの性能差を説明するに足らなかった。これを受けて第三期 (10-12 月) および第四期 (1-3 月) においては、`VTune` に代えて性能分析ライブラリ `PMlib` による詳細な性能測定を試みた。`PMlib` が `KNL` に対応していないため、九州大学の `ITO (Skylake-SP)` を利用することで `ifort` コンパイラにより発行される SIMD 命令の詳細を調べた。図 4 に示すようにコンパイルオプションを変えることで 512bit SIMD 命令が高い割合で発行されることがわかり、今後のコード最適化の指針を得ることができた。特に、`peel` ループに対して 512bit SIMD 命令を発行させない最適化を施すことで実行時間が短

縮されることがわかった。将来的に PMLib が KNL に対応した場合は同種の解析を Oakforest-PACS 上で行いコード最適化を進める計画である。第三期以降ではさらに新たな研究として、FX100 を用いて、ノード間の MPI 通信にアシスタントコアを明示的に用いることにより OS ジッタによる通信性能劣化の低減を図った。MD 計算は 1 ステップあたりの実行時間が数 ms~10 ms 程度と非常に短く OS ジッタによる MPI 通信への影響は大きい。特にストロングスケール問題において、実用上これを除くことによる恩恵は非常に大きいと期待され、来年度の JHPCN 課題においても同課題に継続的に取り組む。

今年度与えられた計算機資源は、コード改良のためのテスト、長時間 MD 計算テスト、および脂質二重膜系など実際の MD 計算研究への応用に用いてきた(学術論文[1,3], 国際会議発表[1], 国内会議発表[1-3]。FX10 は 100%, FX100 および Oakforest-PACS とともにほぼ 100%消費した。

7. 研究成果リスト

(1) 学術論文

- [1] Y. Andoh, S. N. S. Mohamed, S. Kitou, S. Okazaki, “Structural ordering of lipid bilayers induced by surfactant molecules with small hydrophilic head group”, *Mol. Siml.*, **43**, 1247-1255 (2017).
- [2] Y. Andoh, S. Suzuki, S. Ohshima, T. Sakashita, M. Ogino, T. Katagiri, N. Yoshii, S. Okazaki, “A thread-level parallelization of pairwise additive potential and force calculations suitable for current many-core architectures”, *J. Supercomput.*, in press (2018).
- [3] Y. Andoh, S. Kitou, S. Okazaki, “Difference in molecular mechanisms governing changes in membrane properties of phospholipid bilayers induced by addition of non-ionic and zwitterionic surfactants”, *J. Mol. Liq.*, 投稿中.

(2) 国際会議プロシーディングス

該当なし

(3) 国際会議発表

[1] ○ Y. Andoh, S. Hayakawa, S. Okazaki, Molecular dynamics study of differences in membrane properties between outer and inner leaflets of cell plasma membranes, Joint 2017 EMLG/JMLG Meeting, Wien, Austria, Sept. 11-17 (2017). ポスター

[2] ○ Y. Andoh, Recent developments of molecular dynamics calculation software MODYLAS, 5th ADAC Workshop, Ookayama Feb. 15 (2018). Oral (Invited)

(4) 国内会議発表

[1]早川, ○安藤, 岡崎, 脂質組成の非対称性を考慮したモデル脂質二重膜の分子動力学計算, 第 11 回分子科学討論会, 仙台 (2017). 口頭

[2]○早川,安藤, 岡崎, 細胞膜の外単層膜と内単層膜間での膜物性の違いに関する研究, 第 11 回分子科学討論会, 仙台 (2017). ポスター

[3]○安藤,早川,岡崎, 脂質組成の非対称性を考慮したモデル脂質二重膜の分子動力学計算, 第 40 回溶液化学シンポジウム, 姫路 (2017). 口頭

[4]○中島, 安藤, 坂下, 片桐, 荻野, 永井, Modylas を用いた FX100 のアシスタントコアの性能評価, 情報処理学会第 80 回全国大会, 早稲田(2018). 口頭

(5) その他 (特許, プレス発表, 著書等)

[1] 下司雅章(編), 計算科学のための HPC 技術 2, 第 6 章:大規模 MD 並列化の技術 (安藤), 大阪大学出版会(2017).