

メニーコアプロセッサ SIMD 演算向けの データ並列 2 電子フォック行列計算 II

(九大情基センター^a, 九大シス情^b, 九州先端研^c) 本田宏明^a, 稲富雄一^b, 真木淳^c

【はじめに】

現在、ノートブックからサーバコンピュータに至る多くのコンピュータで複数個の計算コアを持つマルチコアプロセッサが使用されている。現状では 16 コア以下が大半であるが、今後もコア数は増加し、数年以内には非常に多くのコア (> 100 コア) を持つプロセッサも利用可能となると予想されている。高性能低消費電力化の要請からメニーコアプロセッサハードウェアの実装は、現在のマルチコアプロセッサと比較し、コアについては制御回路を簡素化する方向に、多数のコア間についてはネットワーク型通信方法の方向に向うと考えられている。そして主要な特徴だけでも、1. 100 コア以上のメニーコア構成、2. プロセッサあたりの主記憶量の低減ならびに主記憶に対するコアあたりの提供メモリバンド幅の低下、3. 命令処理におけるインオーダー実行、4. バス方式からパケットスイッチネットワーク通信型へのコア間通信方式の変更、5. 16 ウェイ以上の SIMD 演算器、を持つと予想されており、これらの特徴を考慮した有効なプログラム実装が必要になるといえる。

そこで本研究では、最後に挙げた多数のウェイ数を持つ SIMD 演算器の特徴を対象とし、2 電子フォック行列 (G 行列) 計算について入力データ並列計算手法の適用を試み、その有効性の検証を目的とした。また前回の報告に対し本報告では、SIMD ウェイ長の異なる新規のプロセッサについても比較対象とした。

【入力データ並列化法による SIMD 計算】

一般に、コンパイラの自動並列化機能を利用してコード中の多重ループ箇所の SIMD 化オブジェクトコード生成を行うには、最内ループが容易にループアンローリング可能であることならびにループボディ部計算にデータ依存関係が少ない事が必要である。しかしながら通常の G 行列計算コードでは、1. 最内ループが縮約計算であるためにそのループ長が計算実行時に初めて決定される。2. 原始積分計算のループボディ部に複雑な演算依存性がある、といった問題があり、コンパイラによる自動 SIMD 化は困難である。例えば 6-31G の基底関数等ではその最内ループ長は 1 ~ 6 と実行時に変動し、SIMD 向けループアンローリングは困難となる。また、STO-3G の基底関数専用のプログラムとして縮約ループ計算に固定のループ回転数の記述を行ない、ループアンローリングを可能とした場合についても、事前に行った (H₂O)₈ 分子の (ss, ss) 積分タイプのみからなる G 行列テスト計算からは、SIMD 化される浮動小数点演算数割合について 1% 以下との極めて低い結果のみ得られている。

そこで、既存の縮約ループの内側に更にデータ並列計算のループを形成することにより SIMD 化オブジェクトコード生成を試みた。このデータ並列間の計算には依存関係は全く存在しない。化学的にはこの並列計算はポテンシャルエネルギー面作成などの、分子座標が異なる複数の入力データの同時並列計算を対象としている。また、一般に SIMD 計算を効率的に実行するために考慮すべき別の課題として、SIMD 演算対象のデータが連続領域に確保されなくてはならないとの問題がある。通常のプログラミング作成ではオブジェクトコード内でのデータ配置はコンパイラ任せとなってしまうが、一般には SIMD 演算を効率的に行う様に連続には配置されず、SIMD 演算直前のデータ整列並びに SIMD 演算直後の整列データの書戻しオーバーヘッドが発生するために性能低下の原因となる。

そのため、G 行列計算部分での積分計算に必要な全ての浮動小数点データや D, G 行列の全要素をデータ並列分だけ連続領域に多重化して保持するよう変更を行なった。

【性能評価テスト】

上記の入力データ並列化について (ss, ss) , (ps, ss) , (ps, ps) 積分を対象とした G 行列計算の実装を行い、有効性の検証を行なった。積分アルゴリズムには小原法¹を使用した。測定条件として、前回の発表では Intel Xeon X5650 (2.67GHz, 6 コア, 倍精度浮動小数点 2 ウェイ SIMD 演算器) プロセッサからなるコンピュータを使用していたが、今回はより並列度の高い 4 ウェイ SIMD 演算器を持つ Intel Xeon E5-2650 (2.00GHz, 16 コア) からなるコンピュータについても測定し、両結果を比較した。コンパイルには Intel コンパイラ 13.0.1 による O3 最適化を適用し、詳細な性能計測のため Perfmon2 ライブラリ² によるハードウェアカウンターデータの取得を行なった。テスト計算対象として、6-31G 基底関数の $(H_2O)_8$ 分子について分子構造パラメータが同一な 32 個の入力データを準備した。この入力では積分カットオフや部分 G 行列への積分寄与の加算といった分子構造に依存した計算フローが入力間で全ての同一の挙動を示すことになる。

図 1 に 4 ウェイ SIMD プロセッサによる 1~32 の入力ファイルを対象としたデータ並列度あたりの速度向上比を示す。4 データ並列以上にておおよそ 2 倍程度の速度向上が得られている。これは通常のスカラー型の演算と 4 ウェイ SIMD 演算の性能比の半分程度である。積分タイプ毎では (ps, ps) が最も高い速度向上を示した。図 2 には全実行浮動小数点演算数における 2 ウェイならびに 4 ウェイ SIMD 浮動小数点演算数比を示している。データ並列が無い場合では、ほぼ 0% であった SIMD 化浮動小数点演算比が、並列度が 2 の場合にはほぼ 100% の 2 ウェイ SIMD 演算がなされ、4 並列以降では 2 ウェイ演算が 10% 以下となり、代わりに 4 ウェイ SIMD 演算が 90% 以上と増加している。また SIMD 演算を用いない通常のスカラー型演算は 1% 未満であり、全てのデータ並列計算において 100% に近い浮動小数点演算が SIMD 化される事が分った。この高い 4 ウェイ SIMD 演算化率にもかかわらず速度向上比が 2 程度となった原因として、全実行命令数における浮動小数点命令数の割合がおおよそ 30% との結果から、SIMD 計算の対象となる浮動小数点演算数以外のデータロードストア等の命令が大きな実行時間を占めていることが考えられる。

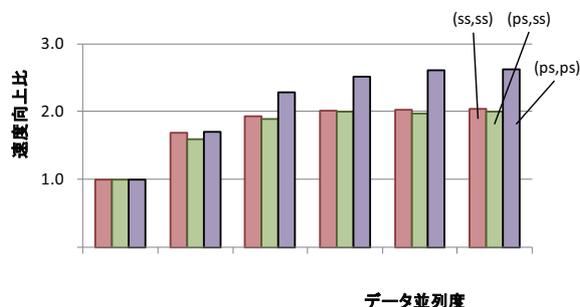


図 1: データ並列計算におけるデータ並列度あたりの速度向上比

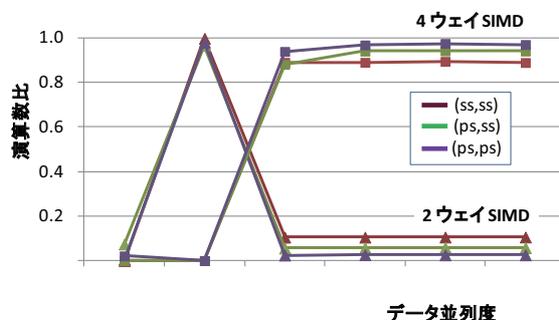


図 2: 全浮動小数点演算数における SIMD 演算数比

¹ S.Obara *et al.*, *J.Chem.Phys.*, Vol.84, pp.3963-3974, 1986.

² <http://perfmon2.sourceforge.net/>