

課題名 (タイトル) :

FX100 を用いたチェックスイート性能評価

利用者氏名 : ○南 一生*, 黒田 明義*, 寺井 優晃*, 熊畑 清*, 宮本 健吾*, 北澤 好人*

所属 : *理化学研究所 計算科学研究機構 運用技術部門ソフトウェア技術チーム

1. 本課題の研究の背景、目的、関係するプロジェクトとの関係

本課題は、スーパーコンピュータ「京」(以下「京」と記す)開発時に作成したチェックスイートを用いて、FX100 での性能解析を行い、システム性能の評価、チューニング効果の評価、チェックスイートの問題点の洗い出しを行うことを目的に利用を行った。

ここで用いたチェックスイートとは、HPC で利用される計算科学アプリケーションから、主要な計算箇所をループ単位の粒度で切り出したものである。切り出しに用いたアプリケーションは、各分野から、アルゴリズム特性に応じて幅広く選択されている。また用途に応じて、単体チェックスイート、並列チェックスイートに分類出来る。

チェックスイートは、「京」開発当初、システム評価並びにアプリケーションのチューニング指針を得ることを目的に構築された。「京」の運用では、システム (OS、ミドルウェア等) のエンハンス/メンテナンスに伴うアプリケーションの性能劣化を「未然に防止する」「検知する」ことを目的とし、以下の機能整備を行った。

- (1) 複数システム(マシン, コンパイラ)動作
- (2) 計算結果のバリデーションチェック機能
- (3) 性能検証の比較結果レポート機能
- (4) 一括実行機能
- (5) ジョブ多重度管理機能
- (6) 評価カーネル選択機能

現在開発が進められているポスト「京」のスーパーコンピュータ開発の場では、最新システムで測定された性能をもとに次世代システムの性能評価を行う。このため、現段階で最新システムである FX100 での性能評価が重要である。本課題では、現状のチェックスイートを用いて、性能評価を行い、そこから問題点を抽出し、再整備を行っ

た。今回 HOKUSAI で実施した計算の多くは、FX100 の CPU アーキテクチャの評価が中心であるため、単体チェックスイートを用いている。

2. 具体的な利用内容、計算方法

(1) 京チェックスイートによる FX100 性能確認

京チェックスイートは、京コンピュータのシステム更新時に性能確認を行うためのツールで並列チェックスイートと単体チェックスイートの 2 種類がある。単体チェックスイートは、ユーザアプリケーションから抽出されたカーネルプログラムで 50 本・90 区間がある。本課題では、単体チェックスイートについて、FX100 での環境構築と実行確認を行い、各カーネルプログラムの性能を確認した。

(2) ルーフラインモデルによる性能評価との比較

プログラムの期待しうる実効性能(限界性能)の予測手法としてルーフラインモデルがある。ここでは、ループ特性を調整可能な、様々なループ集を用いて、メモリ、L1 キャッシュ、L2 キャッシュといった記憶領域の階層構造に着目した解析を行い、ループ構造と限界性能の因果関係について評価を行った。

(3) FrontFlow/blue カーネルの評価

FrontFlow/blue は東京大学革新的シミュレーション研究センターで開発された汎用熱流体シミュレーションコードであり、計算手法は有限要素法であり、用いる要素形状はテトラ、ピラミッド、プリズム、ヘキサなどの各種要素とそれらの混合を扱える。本アプリケーションはものづくりの現場で実用に供されており、我々は幾つかの主要演算部分(カーネルと呼称)に着眼して「京」と

比しての性能評価，高度化の検討を行った。

(4) MD カーネルの評価と再整備

本カーネルは，分子動力学計算に特徴的な二体力計算を行っており，原子数を n として， $O(n)$ で並び替えを行うことで， $O(n^2)$ のリストアクセス主体の計算を連続アクセスに置き換えるとともに，命令スケジューリングが効率的に働くようにループ形状を変形したものである。

単体チェックスイートのうち，MD カーネルの性能評価を行い，いくつかの問題が見つかった。このため今回カーネルの問題の洗い出しを行い，再整備を行った。

(5) グラフ探索に基づく経済シミュレーションコードの性能評価

近年，詳細な経済関連のデータベースが国内データバンクを中心に整備され公開されている。これらのデータベースを元に，金融機関や企業をノード，それらの信用関係や取引関係をエッジとしたグラフを作成することができる。このグラフを用いて，特定のノードからネットワーク全体への経済的なストレスの伝搬をシミュレーションすることで，経済システムの状態を定量化することができる。シミュレーションに用いられる DebtRank アルゴリズムは，最短経路探索問題と同様に，始点ノードから他の全ノードに対して探索を行う。

(6) 倍々精度行列と倍精度ベクトルの積を行うカーネルプログラムの高速化チューニング

産業界で重要とされるアプリケーションである Adventure から，主要演算である倍々精度行列と倍精度ベクトルの積を行うカーネルプログラムについて，性能評価を実施し，高速化チューニングを行った。

3. 結果

(1) 京チェックスイートによる FX100 性能確認

① 環境構築

京コンピュータで動作している単体チェックスイートを FX100 にて実行するために，京のためのステージング環境・スレッド並列の設定を修正して，リファレンスデータの取得と設定を，カーネルプログラム 50 件について行った。

② 実行・性能確認

カーネルプログラム 50 件について，通常実行・精密 PA について，それぞれ正常に実行出来ることを確認した。更に，精密 PA の結果から性能を確認した。FX100 は，京と比較してコア数が 2 倍，SIMD が 2 倍なので，約 4 倍の理論演算性能であるが，演算性能が 3 倍に満たない評価区間が 7 割以上を占めることがわかった。

(2) ルーフラインモデルによる性能評価との比較

ルーフラインモデルは，キャッシュアクセスが増える場合に，予測性能と実測性能が乖離することが分かった。キャッシュアクセスが増大した場合にも適用可能なプログラムの性能予測モデルを開発する研究に，FX100，HASWELL を使用した。性能予測モデルを開発し結果を論文として投稿中である。

(3) FrontFlow/blue カーネルの評価

FrontFlow/blue の主要演算カーネル①については FX100 の SIMD 幅及びメモリスループットが「京」よりも向上しているにも関わらず，元々のソースコードでは最内演算がアンロールされた形で書かれていたため，ベクトルのロードを SIMD ロードで処理できず，その結果 FX100 においてもメモリスループットが向上せず性能を十分に引き出せないことが判った。そこでアンロールされた形で書かれた最内演算を再度ループに戻すことにより SIMD 化が適用されるように修正した。その結果，「京」よりも広い SIMD 幅の効果が十分に発揮され，FX100 ではハードウェアのスペック上限の 80% 近くまでメモリスループットが向上した。「京」においてはループ内演算の 1 回転中の命令並列性を生かすためにアンロールを行っていたが，SIMD 幅が広い CPU では最内ループをアンロールせずループのままとした方が性能向上

するという知見を得た。

主要演算カーネル②についても同様に SIMD 幅の増大に合わせて最内演算をアンロールするのをやめ再度ループ化することで性能向上を望める。ただしこのカーネルを再度ループ化するとループ回転数が SIMD 幅の整数倍に一致していないため、リダクション演算を行う処理において SIMD 適応に起因する演算量の増加が発生することが判明した。今後はこの演算量の増加を抑制できる実装方法について調査を行う必要がある。

(4) MD カーネルの評価と再整備

単体チェックスイートのうち MD カーネルの性能を FX100 で測定すると、想定以上に性能が低いことが分かった。「京」で、76.5%ほどの実行効率が、FX100 では 58.2%程度まで低下した、これは、データアクセスの連続性、命令スケジューリングの問題であり、「L1 アクセス削減するため配列次元移動」、「演算待ち削減のため一時変数を利用」により、汎用的な性能回復を図った。また、他の性能評価手法と比較した際にも、問題があることがあり「ループ分割」、「ブロッキング」を行うことで、より汎用的に評価可能なカーネルとして再整備を行った。

(5) グラフ探索に基づく経済シミュレーションコードの性能評価

本研究では、プロセス並列に先立ってグラフ特有のタスクが不均一な場合のスレッド並列化手法の検討と性能評価を行った。評価のために、DebtRank アルゴリズムの主要なループに対して OpenMP の `omp parallel for/omp task` ディレクティブを用いた計 3 パターンのスレッド化を行い、その性能を評価した。さらに経済的なストレスの伝播に寄与しないノードを省略することで高速化を試みた。結果、SPARC64 XI_{fx}において 32 コアを使用した場合、ランダムに選択したグラフ中の 64 ノードの標本平均で約 17 倍の速度向上を達成した。HOKUSAI を利用した評価結果については研究会論文として報告した。

(6) 倍々精度行列と倍精度ベクトルの積を行うカー

ネルプログラムの高速化チューニング

行列の行数が 24 と少ない一方で、列数が 393,216 という、極端に細長い形状の倍々精度行列と、倍精度ベクトルの積を行うカーネルプログラムの高速化チューニングを行った。このカーネルプログラムの基本構造は、ベクトルの内積を行列の行数回繰り返すものとなっている。まずは、キャッシュライン競合を抑制することを目的として、行列の行番号のループのアンロールを止め、データストリーム数を減らした。次に、処理の並列性を高め、SIMD 化率を向上させることを目的として、内積計算に伴う総和計算の方法を、逐次方式から、前半後半 2 分割ベクトル加算繰り返し方式に変更した。結果として、24 行 393,216 列の倍々精度の行列と、長さが 393,216 の倍精度ベクトルの積を、FX100 システム (MPC) の 1CMG の 12 コアを用いて実行した際の実行性能を、7 倍以上に向上させた。

4. まとめ

「京」で構築されたチェックスイートを用いて、HOKUSAI での評価を行った。単体チェックスイートでの評価、ルーフラインモデルでの評価との比較を行うことで、システム評価上、様々な知見が得られた。

これらの解析を詳細に進めることで、今回新たにチェックスイート側の課題がいくつか顕在化した。FrontFlow/blue 主要カーネル、MD カーネルの評価例はその代表である。また、システム評価・チューニング効果の評価を実施するにあたり、チェックスイートの現行の品揃えでは、必ずしも十分ではないことがわかった。本課題実施したチューニングにより、性能向上の確認がされた「グラフ探索」、「倍々精度計算」などはその代表である。

5. 今後の計画・展望

今回の課題実施によって得られた知見をもと

平成 27 年度 利用報告書

に、更なる詳細の解析を行うとともに、より汎用かつ有用なチェックスイートの整備に力を入れる予定である。

6. 利用がなかった場合の理由

該当なし

平成 27 年度 利用研究成果リスト

【国際会議、学会などでの口頭発表】

寺井 優晃, 藤原 義久, 南 一生, 庄司 文由, “経済シミュレーションのための DebtRank アルゴリズムを用いたグラフ探索コードのスレッド並列化と性能評価”, 情報処理学会研究報告, ハイパフォーマンスコンピューティング, 2016-HPC-153(9), 2016-3-1, JAPAN.