Vol.2013-ARC-207 No.17
Vol.2013-HPC-142 No.17
2013/12/16

Blue Gene/Q における FFT の最適化と性能評価

土井　淳[†1]　　根岸　康[†1]

高速フーリエ変換(FFT)の性能は，多くの科学計算アプリケーションにおいて計算時間短縮のための重要な要素となる．また，並列計算機の通信特性を評価するための指標としても用いられている．本報告では，Blue Gene/Q に特化した並列 FFT 計算の最適化として，スレッド並列化によりパイプライン処理を行うことで全対全通信と計算を重ね合わせる手法を提案する．この手法を実装した並列 FFT ライブラリを用いて Blue Gene/Q を用いて性能評価を行い，全対全通信のバンド幅から計算される論理ピーク性能の 80%という高い実効性能を達成した．

Performance evaluation of parallel FFT on Blue Gene/Q

JUN DOI[†1]　YASUSHI NEGISHI[†1]

Fast Fourier Transform (FFT) is commonly used in the scientific computations on supercomputers, and its performance is one of the important factors to decrease the elapsed time of computer simulation. Also the performance of FFT is used to evaluate a parallel computer as a benchmark test. In this report, we propose an overlapping method of all-to-all communication and calculations using thread parallelization by pipelining the parallel FFT algorithm optimized for Blue Gene/Q supercomputer. We implemented this method and evaluated it on Blue Gene/Q, and the proposed method achieved 80% sustained performance of theoretical peak calculated from all-to-all bandwidth.

1. はじめに

並列計算機において科学技術シミュレーションを効率よく実行するために，高速フーリエ変換（FFT）の性能が重要になる場合がある．特に超並列の分散メモリ型計算機を用いる場合，FFT の計算量もさることながら，並列 FFT に必要な全対全通信（All-to-all 通信）の性能が非常に重要になり，並列化のスケーラビリティ向上の阻害になる場合も少なくない．しかしながら，汎用的な超並列計算機において，全対全通信にのみ特化した通信網を実装することは，コストの面からも電力消費量の面からも現実的ではない．近年の超並列計算機では，Blue Gene シリーズ[1]や京コンピュータ[2]のように，トーラス通信網が用いられている．トーラスは隣接するノード間のみを接続しているので，全対全通信を効率よく行うためには，ソフトウェア的な最適化を行う必要が出てくる．トーラス通信網における全対全通信では，トーラスの最長の軸の長さに反比例して全対全通信の転送速度が決まり，またトーラスの形状も転送速度に影響することが知られている．特に，トーラス形状が対称的ではない場合（トーラスの軸方向の長さが均等ではない場合）について，通信パケットの輻輳によって全対全通信の転送速度が落ちる問題があり[3]，我々は，ソフトウェア的な実装でパケットのスケジューリングを工夫することでこのような場合の転送速度を改善する手法[4]を提案している．

並列FFTの実行時間について全対全通信の実行時間が比較的多いとはいえ，ローカルな FFT 計算や配列の転置にかかる時間自体も無視できない．一般的な並列アプリケーションの最適化を行う際，通信と計算を平行して行うことで通信時間を隠蔽する手法が広く用いられるが，並列 FFT の場合，通常通信時間の方が長いため，通信と計算を平行して実行できれば，計算時間を隠蔽することができる．我々は，Blue Gene/P において並列 FFT 計算をパイプライン化しマルチスレッド実行することで計算時間と全対全通信の重ねあわせ実行をする手法[4]を提案している．

本報告では，HPC チャレンジベンチマークのテスト項目の 1 つである並列 1 次元 FFT と，科学技術シミュレーションで広く使われる並列多次元 FFT について，Blue Gene/Q を用いて最適化を行い，性能を評価した結果を報告する．特に，並列 FFT の全対全通信と計算の重ね合わせ実行について Blue Gene/P で行った最適化との違いについて述べる．以下、2 章では，一般的な FFT の最適化手法について，3 章では Blue Gene/Q についての概要と全対全通信の特性について, FFT の最適化手法について説明する. 4 章では Blue Gene/Q を用いた並列 FFT の性能測定の結果を示す.

2. 並列 FFT

2.1 1 次元 FFT の並列化

一般的に 1 次元の FFT を並列化するには，元の 1 次元配列を np 個のプロセスに均等に分割して各プロセスに配置する．この状態で式(1)に示す離散フーリエ変換（DFT）を直接計算するには，各プロセスの持つデータを随時参照する必要があるため，効率良く計算するのは困難である．

†1 日本アイ・ビー・エム株式会社 東京基礎研究所
IBM Research - Tokyo

Vol.2013-ARC-207 No.17
Vol.2013-HPC-142 No.17
2013/12/16

$$y_k = \sum_{j=0}^{n-1} x_j w_n^{jk} \quad \text{ここで} \quad w_n = e^{-2\pi i/n} \tag{1}$$

そこで，FFT の並列計算では、各プロセスの持つ配列を 2 次元配列とみなし，2 軸方向にそれぞれ FFT 計算をすることで並列化を可能とした，6-step FFT[5]が用いられる．元の配列の長さを n としたとき，$n=n_1*n_2$ として 2 次元配列を作った場合，式(2)のように計算できる．

$$y(k_2,k_1) = \sum_{j_1=0}^{n_1-1}\sum_{j_2=0}^{n_2-1} x(j_1,j_2) w_{n_2}^{j_2k_2} w_{n_1n_2}^{j_1k_2} w_{n_1}^{j_1k_1} \tag{2}$$

この 6-step FFT では、3 回の転置，2 回の FFT 計算，1 回のひねり係数の乗算の 6 回のステップから計算できる．

この計算には，n_1 および n_2 の並列度があるため，これらの公約数である np 個のプロセスを用いて並列実行できる[6]ことになる．このとき，初期状態では配列 x を np 個のプロセスで分割して持ち，$x(n/np) = x(n_1, n_2/np)$ と，2 次元配列の形で表す．この状態から式(2)を計算していくが，まず長さ n2 の配列の FFT を計算するために $x(n_2,n_1/np)$の形に全対全通信を使って 2 次元配列を転置する．次に n_1 の長さの FFT を計算するために，もう一度全対全通信を使って配列を転置する．最後に，出力 $y(n_2, n_1/np)$を得るためにもう一度全対全通信を行い，計 3 回の全対全通信が必要となる．6-step FFT の全対全通信と配列の転置の様子を図 1 に示す．

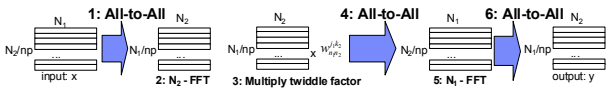


図 1　6-step FFT による並列 1 次元 FFT の計算

2.2 多次元 FFT の並列化

一般に、多次元 FFT の計算は，各次元について，1 次元 FFT を順に計算することで計算できる．各次元の FFT を計算するためには，その次元を一番内側にするような配列の転置を行う．並列化を行う場合，ある次元方向について，プロセス間で配列を分割して持っている場合，転置に全対全通信を伴うことになる．よって，配列を分割した次元の数 x2 回の全対全通信が必要となる．

3. Blue Gene/Q における並列 FFT の最適化

3.1 Blue Gene/Q 概要

Blue Gene/Q は Blue Gene シリーズの第 3 世代の超並列計算機システムで，米国ローレンスリバモア国立研究所に納入され 2012 年 6 月 Top500 ランキング 1 位のセコイアシステムにも採用された．Blue Gene/Q は，初代 Blue Gene/L からの設計思想を受け継ぎ，組み込み専用の Power プロセッサを使用し低消費電力で高密度な実装を実現しつつも，ノードごとに 16GB の主記憶と 16 コアの 64 ビット CPU を実装し，また各コアあたり 4 スレッドのハードウェアスレッド（SMT）を持ち，共有メモリ並列化をサポートすることで，より汎用的なスパコンとして活用できるようになった[7]．浮動小数点演算は，各コアあたり 4-way の SIMD 演算器を持ち，乗加算演算（FMA 演算）を用いると，コアあたり動作周波数の 8 倍の 12.8GFlops がピーク性能となる[8]．また，複素数演算においては専用の命令群を利用することで 2-way の SIMD として使用できる．

ネットワークは，これまでの 3 次元トーラスから 2 次元増えた 5 次元トーラス接続となり，1 リンクあたりの転送バンド幅は 2GB/s であり，ノードあたり 10 方向（双方向）合計で 40GB/s となる[1]. 5 次元の各次元は ABCDE で表す．A から D 軸では軸の長さが 4 以上になるとトーラス接続になりそれ未満の場合はメッシュ接続となる．一方 E 軸は常に長さが 2 のトーラス接続となる．2x2x2x2x2 が最小単位のネットワークで，32 ノードのパーティションを形成する．

3.2 Blue Gene/Q の All-to-all 通信

トーラスネットワークにおける全対全通信のバンド幅は，トーラスの最長の軸の長さに反比例する．そのため，使用するノード数が増えるにつれてパーティション全体のトーラスサイズが大きくなりバンド幅が減少する．式(3)は，トーラスの最長軸の長さとバンド幅の関係[3]を表しており，この式から計算される Blue Gene/Q と Blue Gene/P のバンド幅の違いを図 2 に示す．

$$A = e * B * (8/L) \tag{3}$$

なお，e はパケットヘッダ等オーバーヘッドによる損失を考慮した効率でここでは 0.9 とし，B はリンクあたりのバンド幅，L は最長軸の長さ（ただしメッシュ接続の軸の場合は 2 倍する）を表す．

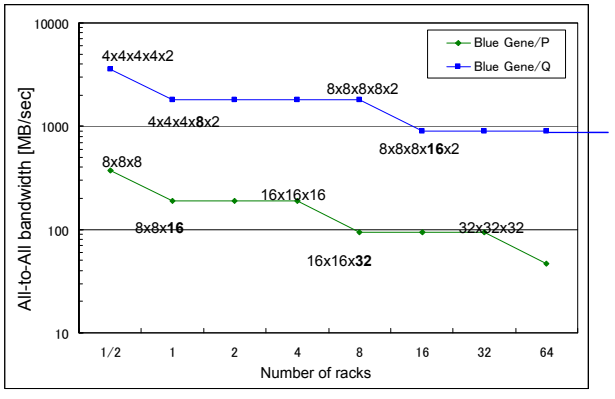


図 2　Blue Gene 2 機種によるトーラスの大きさによる全対全通信の論理バンド幅の比較

Blue Gene/Q は 5 次元トーラスではあるが 5 次元目は長さ 2 で固定なので，4 次元分の長さが倍になると，1 段階バンド幅が半減する 3 次元トーラスの Blue Gene/P が 64 ラッ

ク時にバンド幅が 1 ラック時の 1/4 になるのに比較すると，Blue Gene/Q では 128 ラック時に 1 ラック時の 1/2 を得られるのは大きなアドバンテージである．

また，トーラスネットワークにおける全対全通信のバンド幅はトーラスの形状にも影響される．長さが不均等なトーラスの場合，パケットのスケジューリングを工夫しないとバンド幅が理論性能よりも劣ることが知られている[3]．我々は，この問題を解決するために，トーラスの長い軸から順にたどるようにパケットをルーティングすることで，この性能低下を抑える手法を開発し，Blue Gene/P においてソフトウェア的に実装した[4]．一方 Blue Gene/Q では，ハードウェアによるパケットルーティングがサポートされており，任意の順番の軸方向にパケットをたどらせることが可能となり[1]，不均一な長さのトーラスネットワークにおいても，全対全通信の性能劣化はほとんど無くなった．

図 3 に Blue Gene/Q において測定された全対全通信のバンド幅をまとめる．

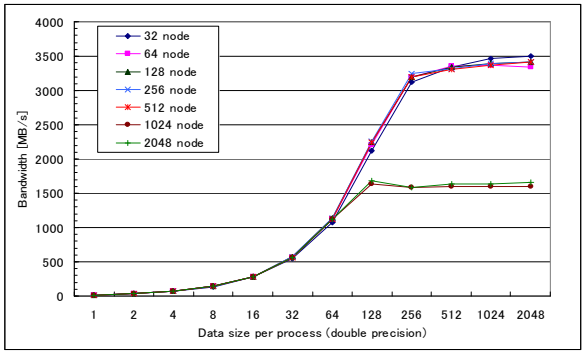


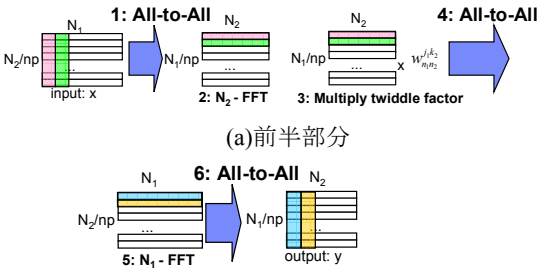
図 3　Blue Gene/Q における全対全通信のバンド幅の実測値の比較

3.3 並列 1 次元 FFT の最適化

Blue Gene/P における並列 1 次元 FFT の最適化においては，全対全通信のソフトウェアによるパケットルーティングを最適化した実装に FFT の実装を組み合わせることで任意のトーラス形状に対応する最適化 FFT を実現した[4]．これを実現するために，並列 6-step FFT をパイプライン化し，共有メモリ並列化を用いてノード内のスレッド並列化を行うことで全対全通信と，FFT のその他の処理とパケットルーティングの処理を重ね合わせ，実質的に全対全通信にかかる時間のみで FFT を行うことを可能とした．

Blue Gene/Q では，全対全通信自身の最適化は不要なので、FFT のその他の処理と全体全通信の並列化のみを行う．

並列 6-step FFT をパイプライン化するにあたり，n_2 の FFT を行う前半部分と，n_1 の FFT を行う後半部分の 2 つに分け，それぞれについて別々のパイプライン処理を行う．このとき，図 4 に示すように前半部分は n_1/np 個の，後半部分は n_2/np 個の独立した処理に分割することができるのでスレッド並列化が可能となる．



(a)前半部分

(b)後半部分

図 4　並列 6-step FFT のスレッド並列化

ここで，全対全通信を各スレッドが自分の分だけを行えば，前半部分と後半部分をつなぐ部分だけスレッド間の同期を取れば，スレッド並列化が完成するが，そのままだと，複数のスレッドがネットワークのバンド幅を奪い合ってしまい，効率が悪くなる．そこで，全対全通信を 1 つのスレッドのみにしかできないように，全対全通信の入り口で制御する．ここで注意しなければいけないのは，単に全対全通信にアクセスできるスレッドを 1 つにするだけでなく，各プロセスで，対応するスレッドが同時に通信しなければいけない点である．そこで，スレッド制御にカウンタを用いて，カウンタの値をスレッド数で割った余りが，スレッド ID に等しい場合に，そのスレッドが全対全通信にアクセスできるようにする．このカウンタの値の更新は，atomic な加算処理を行う必要があるが，Blue Gene/Q には，L2 キャッシュを利用した非常に高速な atomic 演算を行う実装があり，これを使用することで小さなオーバーヘッドで制御することができる．

図 5 に 4 スレッドを使用した場合の，6-step FFT の前半部分についてのパイプライン制御の様子を示す．このように全対全通信の制御を行うことで，全対全通信と他の FFT に係る処理を重ね合わせて実行することが可能となり，並列 1 次元 FFT の処理時間は，実質的に全対全通信の処理時間とほぼ等しくなる．

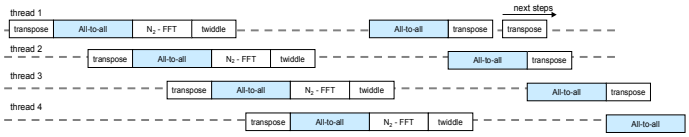


図 5　6-step FFT の前半部分についての全対全通信のスケジューリングによるパイプライン制御

ところで，Blue Gene/Q では，ノードあたり 16 コアあり，各コアあたり 4 つのハードウェアスレッドを持つため，最大で 64 スレッドの共有メモリ並列化が可能である．ノードあたりの性能を最大化するためには 16 スレッド以上の並列化が必要であるが，6-step FFT の並列度（n_1/np および n_2/np）が十分に大きくない場合，パイプラインの本数を増やしても十分な性能の向上を得ることはできない．

そこで，Blue Gene/Q では，スレッド並列化を 2 段階に

分け，1 段目でパイプライン並列化を，2 段目でノード内の FFT 計算の並列化を行う，ハイブリッドスレッド並列化手法を用いる．たとえば，32 スレッドを用いる場合，4 本のパイプラインを用い，それぞれのパイプライン毎に 8 つのスレッドを用いて計算処理を並列化する．

ノード内の FFT 計算のスレッド並列化は，式(2)に出てくる 1 次元 FFT を再帰的に 2 次元化することにより可能である．

3.4 並列多次元 FFT の最適化

トーラスネットワークで形成される並列計算機において，多次元の問題を並列化する場合，一般的に問題の次元とトーラスの次元が同じであれば，トーラスの長さで解こうとしている問題を分割すると効率良く並列化できる場合が多い．また Blue Gene/Q のような 5 次元トーラスでは，解こうとする問題の次元数のほうが小さい場合もある．この場合トーラス上のノードのマッピングを変えることで仮想的なより次元の低いトーラスを組むことにより問題の次元にあわせることでうまく並列化できる[9]．例えば，1/2 ラックの Blue Gene/Q（4x4x4x4x2 の 5 次元トーラス）を 3 次元の問題の並列化に適用する場合，例えば ABxCExD=16x8x4 という仮想的な 3 次元トーラスを形成できる．この例で，A 軸と B 軸を組み合わせて長さ 16 のトーラスを形成するには，図 6 に示すようなノードのマッピングをすれば良い．

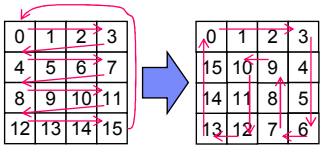


図 6　ノードのマッピングを変更することで複数次元を組み合わせてトーラスを形成する（4x4 の例）

このように 5 次元トーラスを仮想トーラスにマッピングした場合，全対全通信のバンド幅は，仮想トーラスの長さに反比例するのではなく，元のトーラスの最長辺の長さに反比例する．

多次元 FFT の最適化についても，1 次元 FFT と同様にパイプライン化を行い，全対全通信と計算を重ね合わせて実行する．例えば，三次元 FFT の場合，$x(n_1/px, n_2/py, n_3/pz)$ のように分割したとする．ここで px,py,pz は各軸方向の分割数で px*py*pz=np である．このとき，1 次元 FFT 同様に，前後 2 つのグループに分けてパイプライン化する．ここでは，(1)X と Y，(2)Z の 2 つのグループに分けるとすると，(1)は n3/np の並列度，(2)は n1*n2/np の並列度があり，それらの並列度の分だけパイプライン化，スレッド並列化が可能である．ここで注目すべきは，2 方向以上の FFT を 1 つのグループにした場合，パイプライン化によって全対全通信と計算を重ね合わせられるだけではなく，異なる軸方向の全対全通信同士も重ね合わせることが可能であるという点である．図 7 に示すように，それぞれの方向の全対全通信を 1 つのスレッドからのみアクセス可能になるように制御する．図 7 では，X 軸のトーラスの長さが Y 軸の倍であるため通信時間が倍になっており，結果 Y 軸方向の全対全通信は，X 方向の全対全通信に隠れてしまう．長さが同じであっても互いに重ねあわされる．

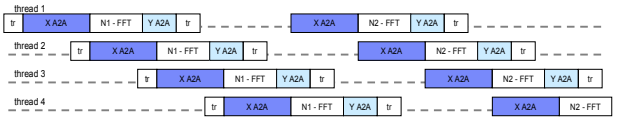


図 7　多次元 FFT のパイプライン化における 2 方向の全対全通信の重ね合わせ

この手法を用いると，本来グループ化した次元数 x2 回の全対全通信が必要であるが，見かけ上たった 2 回の全対全通信で処理できることになる．

4. Blue Gene/Q における性能評価

4.1 1 次元 FFT の性能評価

まず，パイプラインの本数を変えた場合の 1 次元 FFT の性能の違いを調べた．Blue Gene/Q 1/2 ラック（512 ノード）を使用し，ノードあたりのスレッド数=パイプラインの本数として 1 スレッドから 64 スレッドまで変えた場合の 1 次元 FFT の実効性能の測定値を図 8 に示す．

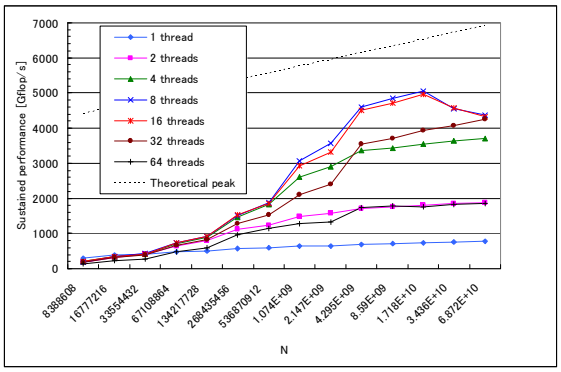


図 8　Blue Gene/Q 1/2 ラックにおけるパイプラインの本数を変えた場合の 1 次元 FFT の実効性能

ここで，Theoretical peak（論理ピーク性能）は，1 次元 FFT の計算量を 3 回分の全対全通信の通信時間の合計で割ったものである．この結果では，8 スレッドと 16 スレッドを使用した場合の性能がもっとも高く，それ以上のスレッドを使用しても十分な性能が得られないことが分かる．逆に少ないスレッド数では，計算部分の処理時間が通信時間よりも増えてしまい重ね合わせが十分ではない状態になる．

続いて，使用するスレッド数をノードあたりのスレッド数を 32 スレッドに固定し，パイプラインの本数 x パイプラ

インあたりのスレッド数=32 となる組み合わせを変えて行った場合の 1 次元 FFT の実効性能の測定値を図 9 に示す.

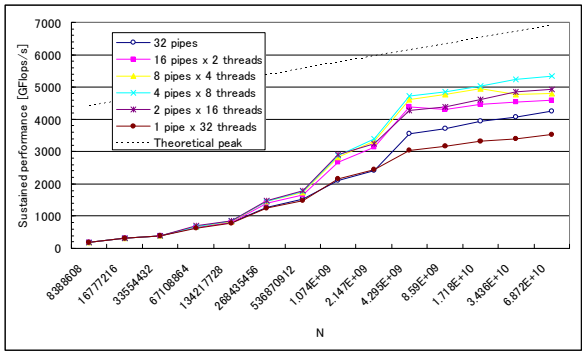


図 9 Blue Gene/Q 1/2 ラックにおけるパイプライン数と計算スレッド数を変えた場合の 1 次元 FFT の実効性能

図 9 でパイプラインが 1 本の場合は, 通信と計算の重ね合わせが一切行われていない状態で, この性能が一般的な並列 6 step FFT の性能であるが, 論理ピーク性能の半分ほどしか出ていない. パイプラインの本数を増やしていき, 通信と計算が重ね合わせられるようになると, 性能が向上していき, パイプラインを 4 本にした場合に論理ピーク性能の 80%弱まで性能が向上しているのがわかる.

図 8 と図 9 の結果を比較すると, スレッド並列のハイブリッド化による性能向上はそれほど大きくは無いが, 6-step FFT の再帰的な並列化によって, キャッシュメモリの利用が最適化され, 図 8 で大きな配列の FFT を行う場合に性能が大きく低下しているのに比べて, 性能が低下しなくなるという利点も得られた.

図 10 は, HPCC ベンチマークの G-FFT テストにおける, 本手法を実装した場合の実測値および論理ピーク性能(limit と表記)を示したグラフである.

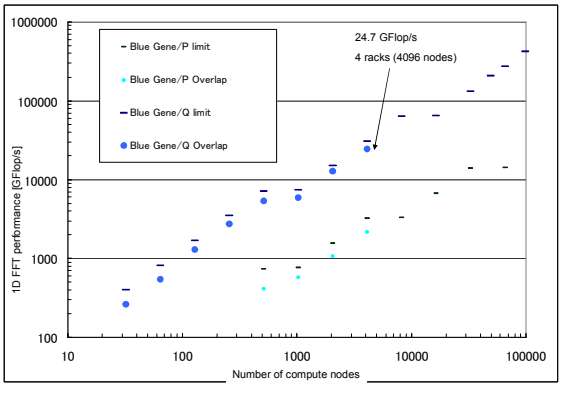


図 10 Blue Gene/Q と Blue Gene/P による HPCC ベンチマークの G-FFT テストの結果の比較

Blue Gene/P では, ノードあたり 4 コア (=4 スレッド) であり, パイプラインの本数は 4 本である. Blue Gene/Q では, ノードあたり 32 スレッドを使用し, パイプライン 4 本 x8 計算スレッドで測定した結果である.

4.2 多次元 FFT の性能評価

1 次元 FFT の場合と同様に, 1 ノードあたり 32 スレッドを使用し, パイプライン 4 本 x8 計算スレッドに設定し, 2 次元 FFT および, 3 次元 FFT の性能を測定した. このときの測定結果を図 11 と図 12 に示す.

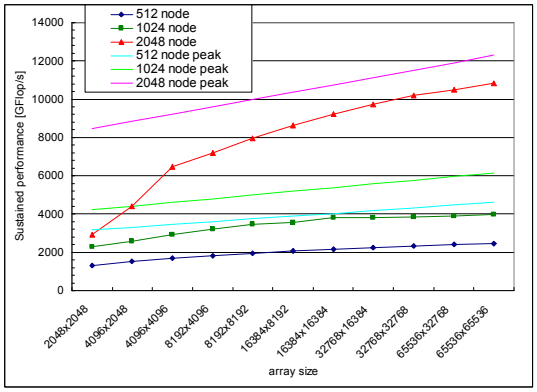


図 11 Blue Gene/Q における 2 次元 FFT の性能

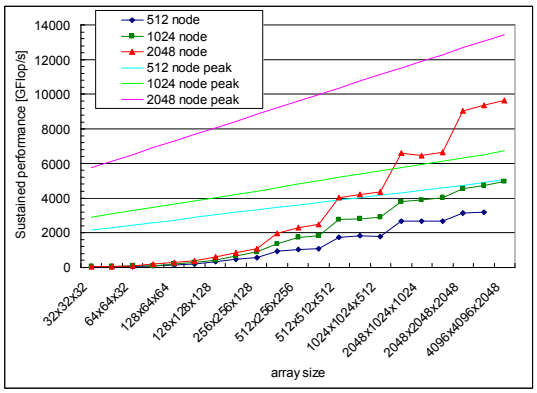


図 12 Blue Gene/Q における 3 次元 FFT の性能

ここで, 3 次元 FFT の論理ピーク性能は, 全対全通信 2 方向分を重ね合わせた場合を考え 4 回分の全対全通信の時間の合計を通信時間として計算したものである. よって, 同じサイズの配列についての論理ピーク性能は 2 次元 FFT と同じになる.

5. おわりに

本報告では以前 Blue Gene/P において我々の提案した並列FFT計算をパイプライン化して全対全通信と計算を重ね合わせる手法について，後継機種である Blue Gene/Q において実装し，性能を評価した．Blue Gene/Q ではハードウェアによるパケットルーティングのおかげで，全対全通信自体を最適化する必要が無くなった．一方，ノードあたりのコア数/スレッド数が増えたことによりスレッドのマッピング方法を工夫する必要が生じたが，パイプラインスレッドと計算スレッドの 2 段階のスレッドマッピングによって，最適なスレッド並列化を実現できた．これらの最適化により，Blue Gene/P 同様に，全対全通信から計算される論理ピーク性能に対して 80%前後という高い性能を実現できた．

今後は Blue Gene 以外のトーラスネットワークを持つ並列計算機や，他のネットワーク構造を持つ並列計算機においても本手法を適用した FFT の性能評価や，実アプリケーションにおける性能評価をしていきたいと考えている．

参考文献

1) D. Chen et al. The IBM Blue Gene/Q Interconnection Network and Message Unit, Supercomputing '11 Proceedings of the ACM/IEEE conference on Supercomputing SC11, 2011.
2) Y. Ajima et al., “Tofu: A 6D Mesh/Torus Interconnect for Exascale Computers,” *Computer*, Vol. 42, No. 11, Nov. 2009, pp. 36-40.
3) Y. Sabharwal et al., “Optimization of Fast Fourier Transforms on the Blue Gene/L Supercomputer,” *International Conference on High Performance Computing, HiPC 2008*, 2008, pp. 309-322.
4) J. Doi et al. Overlapping Methods of All-to-All Communication and FFT Algorithms for Torus-Connected Massively Parallel Supercomputers, Supercomputing '10 Proceedings of the ACM/IEEE conference on Supercomputing SC10, 2010.
5) D.H. Bailey, “FFTs in external or hierarchical memory,” *The Journal of Supercomputing*, Vol. 4, 1990, pp. 23–35.
6) D. Takahashi et al., “A Blocking Algorithm for Parallel 1-D FFT on Clusters of PCs,” *Proc. 8th International Euro-Par Conference (Euro-Par 2002), Lecture Notes in Computer Science*, No. 2400, 2002, pp. 691-700.
7) Introduction to Blue Gene/Q, http://www.ibm.com/common/ssi/cgibin/ssialias?subtype=WH&infotype=SA&appname=STGE_DC_DC_USEN&htmlfid=DCL12345USEN&attachment=DCL12345USEN.PDF.
8) The IBM Blue Gene Team. The Blue Gene/Q Compute Chip, Presented at the Hot Chips Conference 23, August 17-19, 2011.
9) J. Doi, Peta-Scale Lattice Quantum Chromodynamics on a Blue Gene/Q Supercomputer, Supercomputing '12 Proceedings of the ACM/IEEE conference on Supercomputing SC12, 2012.