JAIST Repository
https://dspace.jaist.ac.jp/

Title	超並列計算機におけるハードウェア高速化によるシステム性能の評価
Author(s)	井口, 寧; 黒川, 原佳; 松澤, 照男
Citation	Research report (School of Information Science, Japan Advanced Institute of Science and Technology), IS-RR-2001-027: 1-13
Issue Date	2001-12-12
Type	Technical Report
Text version	publisher
URL	http://hdl.handle.net/10119/8395
Rights	
Description	リサーチレポート（北陸先端科学技術大学院大学情報科学研究科）

JAIST
JAPAN
ADVANCED INSTITUTE OF
SCIENCE AND TECHNOLOGY

Japan Advanced Institute of Science and Technology

超並列計算機におけるハードウェア高速化によるシステム性能の評価

井口 寧，黒川 原佳，松澤 照男

2001 年 12 月 12 日

IS-RR-2001-027

北陸先端科学技術大学院大学

情報科学センター

〒 923-1292 石川県能美郡辰口町旭台 1-1

inoguchi@jaist.ac.jp

©Yasushi Inoguchi, 2001

ISSN 0918-7553

要旨

本論文では，CRAY-T3E および CRAY-T3E/1200E における，超並列システムの CPU のクロック速度とプロセッサ間通信速度の向上によるシステム全体の性能向上について検討する．システム全体の性能は，CPU クロックの他，相互結合網の通信速度やメモリなど他の要素が影響するため，CPU クロック速度の向上に対して線型的な性能向上は望めない．また，通信速度の向上は，並列処理を行なう場合にはじめて効果が現われるため，アプリケーションプログラムを実行する上での性能向上に対する複合的な要因を評価する必要がある．検討に用いるベンチマークプログラムは，様々な評価が可能な NAS Parallel Benchmark (NPB) を用いた．その結果，CPU クロック速度が 2 倍，通信速度が 1.35 倍向上したシステムで，平均 40 % 程度の性能向上が得られることが分かった．

1 はじめに

近年，多数のプロセッシング要素 (PE) を結合した大規模な超並列システムが盛んに研究され，実用システムとしても構築されている．これらのシステムで用いられているRISCベースのCPUは，実装プロセスの進歩にともなって，システムクロックの高速化が容易に可能であるため，システムの性能は急速に向上している．一方超並列システムの性能は，CPUの処理性能だけでなく，PE間通信速度やメモリシステムの速度にも大きく影響される．このため，CPUのクロック速度やPE間通信速度の向上比から，超並列システムとしての処理性能の向上率を予測することは容易ではない．

より高速なハードウェア部品と交換することにより，システム全体の性能向上を目指す試みは，PCクラスタ等のようなコモディティな部品を用いる場合や，商用並列システムにおけるより高速なCPUの搭載として，数多く行なわれている．このため，超並列システムの構成部品のコモディティ化は，今後急速に進むと考えられる．しかしながら，超並列システムの一部を高速化した場合は性能が据え置かれた部分がボトルネックとなり，期待した性能が得られない場合がある．超並列システムのハードウェアをバージョンアップした場合，アプリケーションごとにどのような性能向上やボトルネックが存在するかを的確に知る必要がある．

超並列計算機の性能評価に用いるベンチマークプログラムは，プログラムの性質や調べたい性能に合わせて様々なプログラムが存在する．中でもよく用いられるのは，TOP500 の評価に用いられている Linpack や NAS Parallel Benchmark(NPB)[1, 2, 3] である．Linpack は，連立一次方程式の直接解法を基本としたものであるが，Linpack は超並列計算機のある一面を評価していると考えられる．NPB は，Computational Fluid Dynamics(CFD) の研究分野において必要な数値計算を含む，複数のベンチマークプログラムの集合体である．行列計算や乱数生成などの他分野でも扱う数値計算の多くを内包し，広い分野に適用できるベンチマークプログラムである．

本論文では，CRAY-T3E および CRAY-T3E/1200E を例として，CPU のクロック速度と PE 間通信性能の向上が超並列システム全体の処理速度の向上にどのように寄与するか，様々な角度から評価を行なう．ベンチマークプログラムとして，ハードウェア機能の速度向上による様々な影響を検討するため，より詳細にシステムの性能を評価できる NPB Version 2.3 を用いて詳細に検討する．本論文の構成は次の通りである．第二章で測定条件としてのハードウェア性能と，NPB について述べる．これらを用いた評価結果と考察について第三章で論じる．第四章はまと

めである.

2 測定条件

2.1 CRAY-T3E システム

評価に用いたCRAY-T3E と CRAY-T3E/1200E について概略を述べる. CRAY-T3E は, CRAY Inc.(旧 CRAY Research) が開発した超並列システムである [4]. これに対し, CRAY-T3E/1200E は, CRAY-T3E の CPU のクロック速度と PE 間通信速度を向上させており, システムの特徴は同一である. CRAY-T3E と CRAY-T3E/1200E に共通する基本的なシステムの特徴は, 以下の通りである. 各 PE は, 1 つの CPU, ローカルメモリ, およびネットワークインターフェースから構成される. CPU は, Alpha 21164(EV5) を用いており, これを CRAY-T3E は 300MHz, CRAY-T3E/1200E は 600MHz で駆動している. 一次および二次キャッシュを CPU 内に有しているが, CPU 外には三次キャッシュは持たない. 代わりにベクトルパイプランの概念を取り込んだ Stream Buffer と呼ばれる定ストライド (4 段) のパイプライン型 (6 本) のバッファを持ち, 二次キャッシュのキャッシュミス時に先読みによるデータ転送を行なう. グローバル・アドレス空間へのデータアクセスのために, E-Register と呼ばれるバッファを有しており, これを用いて全てのデータ (ローカル/リモート) にアクセスできる. 例えば, Gather や Scatter 等の操作では実際には通信を行なわず, 直接リモートメモリに対して操作が行なわれる. PE 間の相互結合網は, 三次元のトーラス構造のネットワークトポロジである.

表 1 に, CRAY-T3E および CRAY-T3E/1200E のシステムの性能を示す. 表中, CRAY-T3E と CRAY-T3E/1200E で性能が異なる部分を太字で示してある. また各 PE 内部の概念図を図 1 に示した. CPU のクロック速度向上によって演算性能および CPU の内部キャッシュ速度は 100%向上している. しかし, メインメモリへの帯域等, CPU のチップ外の性能は同一となっている. ネットワーク性能は, 通信帯域の増加によって約 35 % 向上した. また, CRAY-T3E システムの通信機構は, PE(CPU) の動作と密接に関連したシステムであるため, 通常の PC クラスタ等よりも通信遅延が非常に少ないシステムとなっている.

ソフトウェア環境としては, どちらも Programing Environment Version 3.4 を用いた. これに MPI, Fortran90 および C コンパイラが含まれている. コンパイル・オプションは, 全ての場合について "-O3 -dp", つまり最適化レベルを最高, デフォルトの精度を倍精度の指定とした.

表 1: CRAY-T3E と CRAY-T3E/1200E の性能

	CRAY-T3E	CRAY-T3E/1200E
CPU (Clock)	EV5 (**300 MHz**)	EV5 (**600 MHz**)
Primary cache size (Bandwidth)	4KB (**4.8 GB/s**)	4KB (**9.6 GB/s**)
Secondary cache size (Bandwidth)	96KB (**4.8 GB/s**)	96KB (**9.6 GB/s**)
Stream Buffer Bandwidth	600MB/s	600MB/s
Main Memory	64 MB	512 MB
Interconnect Bandwidth	**480MB/s**	**650MB/s**

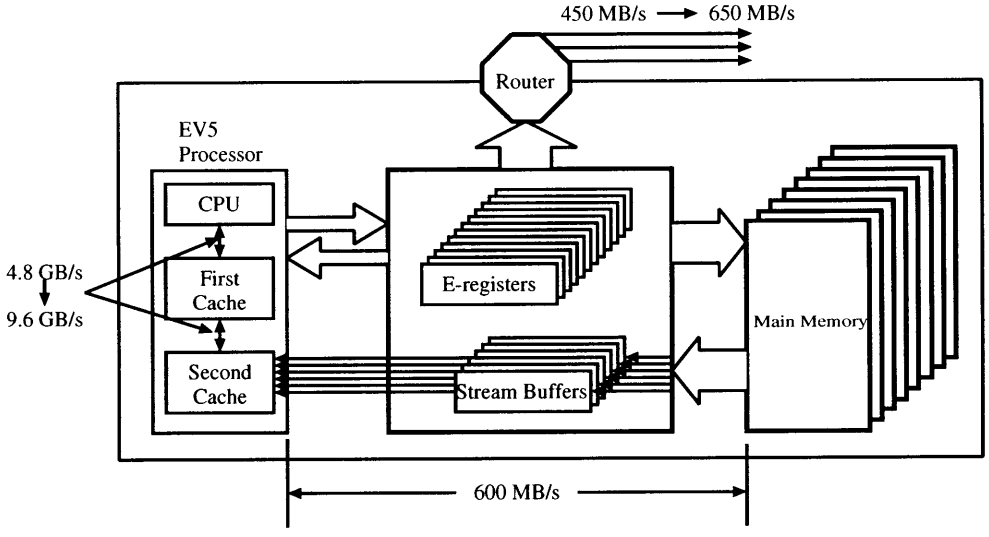

図 1: PE 内の概要および性能

2.2 NAS Parallel Benchmark

つぎに，NAS Parallel Benchmark (NPB) について概略を述べる．NPB は，NASA Ames Reseearch Center の NAS(Numerical Aerodynamics Simulation) Lab. で開発された熱流体関連の科学技術計算のベンチマークである．

NPB は，5つの主要アルゴリズムのカーネルと3つの数値流体計算コード (アプリケーションコード：圧縮性流体を擬似的に計算) からなる．それぞれ、並列計算コードと逐次計算コードからなり，並列計算コードは，Fortran 77 と MPI，逐次計算コードは，Fortran 77 で記述されている．また、問題サイズとして4種類用意されており，ベンチマークコードは，問題サイズ (メモリサイズ) の大きさによって CLASS 分けがなされている．評価で用いた CLASS は，問題サイズが小さい順に CLASS W，A，B，C である．

以下に各ベンチマークコードの概要を示す.

カーネル ベンチマーク

CG 正値対称大規模疎行列の最小固有値を共役勾配法により求めるプログラムである. 非構造格子を用いた流体アプリケーションで良く用いられる. 計算は, 多くのメモリ帯域を必要とする. 同一長のベクトルデータの通信と, 内積を得るための1データの通信が行なわれる.

EP 乗算合同法によって一様な正規乱数を生成するプログラムである. モンテカルロ法でよく用いられ, 並列処理では通信がほとんど発生しない. そのため, 浮動小数点演算性能のみを示す.

FT FFT を用いて三次元偏微分方程式を解くプログラムである. FFT を各次元毎に解いていくため配列を持ち替える. 特に FT は複素数型の配列を MPI_Alltoall[1] により通信するため, 通信負荷が非常に大きい.

IS 大規模な整数値ソートを行うプログラムである. 粒子法等でよく用いられる. 粒子を再び適切なセルに割り当てるためのソートを行なう. MPI_Alltoallv の負荷が このベンチマークのみ C 言語で記述されている.

MG 三次元 Poisson 方程式を Multigrid 法によって求めるプログラムである. 非圧縮流体計算中に現れる Poisson 方程式を解くために用いられる. メッセージ長が非一様な通信を行なうが, 計算負荷は高くない.

アプリケーション ベンチマーク

BT ブロック 3 重対角方程式を ADI 法を用いて解くプログラムである.

SP 5 重対角方程式をスカラー ADI 法を用いて解くプログラムである.

BT, SP ともに物理量等の格子面データの送受信を行ない, 一部のデータ通信に関して通信隠蔽が行なわれている. 演算量は, CLASS W を除いて, BT が多い.

LU 上下三角行列を対称 SOR 法を用いて解くプログラムである. 並列化にはパイプライン処理が用いられ, 通信処理の隠蔽が行なわれる. CLASS W では最も演算量が多い.

計測結果として, Mop/s(Mega OPeration per Second) 値が得られる. EP と IS 以外の Mop/s 値は, ほぼ MFlop/s と同等の値である. EP の Operation は, 乱

[1] 各 PE が配列要素の一部を持ち, これを全ての PE で交換する通信である.

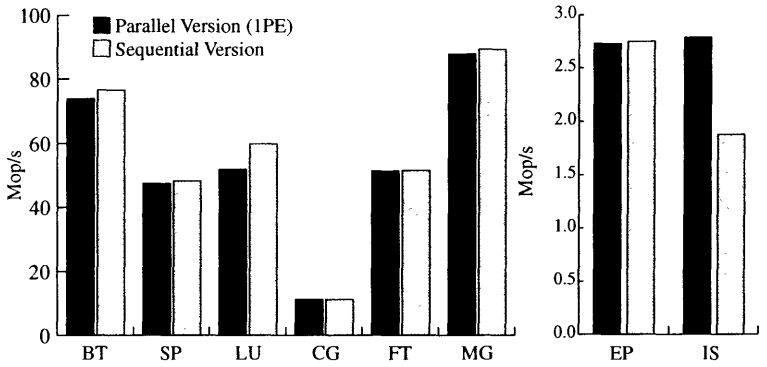

図 2: 逐次コードと並列コードの性能差

数生成数であり，IS の Operation は，整数演算の数であるため，プログラム中で実行される浮動小数点演算は極めて少ない．このことから，EP と IS に関しての Mop/s 値は， MFlop/s と異なる処理速度の指標として考える必要がある．

3 評価結果と考察

3.1 性能評価の方針

超並列システムとしての性能を評価するために，次の手順でベンチマークの実行速度と速度向上率について評価した．最初に，並列化によるオーバーヘッドを明らかにするため，同じ測定条件において，逐次計算コードと並列計算コードの実行速度を測定する．次に，CPU クロックの高速化の影響を調べるため，1PE のみを用いて PE 間通信を行なわない場合の実行速度の測定を行なう．1PE 上での実行では，CPU クロックの高速化による寄与と，動作速度が同一であるメモリシステムの影響が考えられる．このため，メモリの帯域がベンチマークに与える影響を明らかにする．これらをもとに，システム全体の処理速度向上率から CPU クロック速度の向上による寄与を減算することにより，PE 間通信速度の向上による寄与を間接的に求める．

3.2 逐次計算コードと並列計算コード

並列計算コードのオーバーヘッドを評価するため，CRAY-T3E/1200E を用いて，CLASS A での逐次計算コードと並列計算コードの 1 PE での実行速度を図 2 に示した．図の縦軸は Mop/s，横軸は各ベンチマークとする．逐次計算コードと並列計算コードでは，プログラムコードが異なり，並列化のためのオーバーヘッドが存在する．このオーバーヘッドのため，一般には並列計算コードの方が逐次計算コー

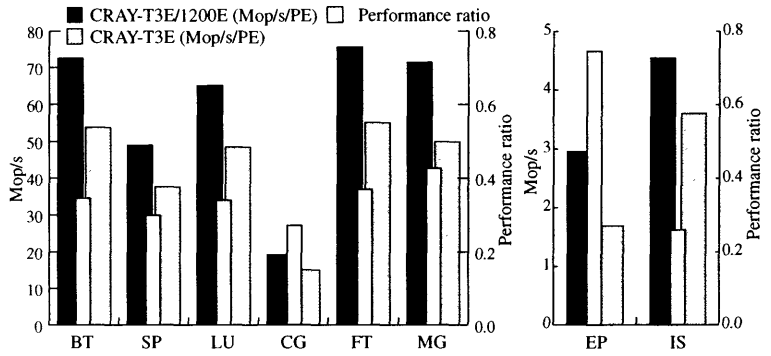

図 3: 1PE における性能 (CLASS W)

ドよりも性能が低下する．CRAY-T3E/1200E での実行結果においても，IS 以外については並列計算コードの性能が低い結果が得られた．CG と FT は，並列化のためのオーバーヘッドが小さく，逐次計算コードと並列計算コードの差が非常に少ない．IS は並列計算コードの方が高速であり，この理由は MPI ライブラリの最適化がなされており，メモリコピーに MPI_Alltoallv を用いる方が同等の機能を逐次計算コードによって実現するよりメモリのアクセス効率が高いためと推測される．いずれの場合も，逐次計算コードと並列計算コードで顕著な性能差は見られず，並列化のためのオーバーヘッドは少ないと言える．

3.3 CPU クロック周波数の向上

次に CPU のクロック速度向上によるシステム性能への寄与について検討する．CPU のクロック速度の差による性能向上を示すために，1 PE のみを用いて (つまり PE 間通信を行なわないで) CRAY-T3E および CRAY-T3E/1200E 上における並列計算コードの実行速度を評価した．評価の結果を図 3 に示す．横軸はベンチマークプログラムの種類，左縦軸は Mop/s，右縦軸は次式で定義される性能比 (Performance ratio) を示した．

$$\text{Performance ratio} = \frac{\text{Mop/s/PE(CRAY-T3E/1200E)}}{\text{Mop/s/PE(CRAY-T3E)}} - 1.0 \qquad (1)$$

CRAY-T3E/1200E は CRAY-T3E の 2 倍のクロック周波数なので，理想状態では，2 倍の性能比 (Performance ratio = 1.0) となるが，メモリなどの動作速度は向上していないため，これよりは小さい性能比となる．CPU における演算量が大きい場合や，キャッシュメモリが有効に動作する場合においては大きな性能比が得られ，メモリへのアクセスが多い場合は，性能比は小さくなることが予想される．

図 3 に示すように，EP は最も大きな性能比が得られた．EP は，メモリアクセスに対して極めて演算量が大きく，CPU のクロック速度向上の効果が現れやすい

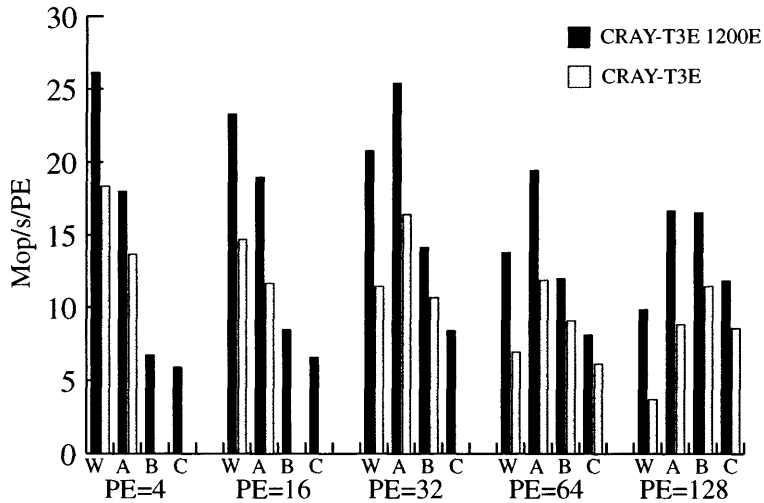

図 4: メモリ帯域による影響: カーネルコード CG

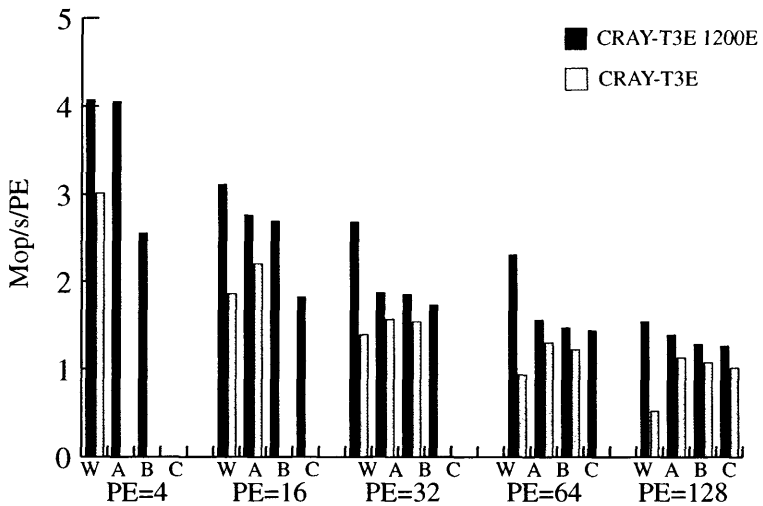

図 5: メモリ帯域による影響: カーネルコード IS

ベンチマークであると言える．このため，PE 単体の性能向上が高い結果となっている．一方，CG と IS は，他のベンチマークと較べて，性能向上が 30 % 以下となり，性能向上が低い．両ベンチマークとも，メインメモリへの広範なアクセスが行なわれる．CRAY-T3E/1200E のメモリ帯域の性能が上がっていないため，CPU のクロック速度向上による寄与が少ないと考えられる．これ以外の場合，FT と MG が 40 % 程度，BT，SP，LU が 35 % 程度の速度向上が得られた．EP を除いた平均的な PE の性能向上は，およそ 35 % である．

3.4 メインメモリの帯域による影響

CPU の演算性能が向上し，メインメモリの帯域が向上しない場合，データロードに対する影響が大きいベンチマークでは性能向上が難しいと考えられる．図 3 の結果からメモリ帯域の影響が比較的大きいと思われる CG と IS を用いてメモリ帯域の影響を検討する．

図 4 に CG，図 5 に IS の実行速度を示す．横軸は各 PE 数での問題の大きさ (CLASS)，縦軸は性能 (Mop/s/PE) である．少ない PE 数では，PE 当りに割り当てられる配列が大きくなる．CRAY-T3E の場合，メインメモリの容量が小さく，大きな問題サイズを少ない PE 数で実行することは不可能であるため，測定できていない問題サイズがある．いずれの場合も，PE 数が大きくなると並列化によるオーバーヘッドの影響が大きくなり，PE 当りの性能が低下する傾向が見られる．

図 4 より，CG は多くの場合で Mop/s/PE 値が W > A > B > C の順となり，問題サイズが大きくなるに従って性能が低下する傾向が得られた．これを詳しく見ると，問題サイズと PE 数によって性能にピークがあらわれ，その位置は，CRAY-T3E と CRAY-T3E/1200E で一致していることが分かる．性能ピークは，キャッシュの影響であると考えられる [5]．係数行列の配列データは，キャッシュに収まるデータサイズではない．しかし，例えば性能ピークの一つとなっている，CLASS A の 32 PE の場合では，コード中のワークベクトルの総量が二次キャッシュサイズに近いサイズとなるため，これ以下の問題サイズでは，キャッシュが有効に機能し，高い性能 (Mop/s/PE 値) が得られている．使用する PE 数がこれより少なくなると，1PE 当りのワークベクトル量が大きくなるため，キャッシュが機能する問題サイズが小さくなる．一方，使用する PE 数がこれより多い場合，1PE 当りのワークベクトル量が小さくなるため，キャッシュが機能する問題サイズの範囲が大きくなる．キャッシュが機能している間は，大きな問題サイズが大きい方が，より大きなベクトルデータとして連続アクセスできるため，実行性能が高くなる．

CRAY-T3E と CRAY-T3E/1200E の CPU 内キャッシュは，容量は変化しないが，速度は CPU のクロック周波数分向上している．ワークベクトルがキャッシュサイズより小さい場合には，CPU のクロック周波数向上が有効に作用したと考えられる．この結果，キャッシュが有効に機能している間は CRAY-T3E/1200E は大きな性能向上が得られた．一方，キャッシュの影響が少ない，問題サイズが大きい場合の性能向上率はほぼ一定しており，約 25 % である．

IS における実行速度を図 5 に示す．問題サイズおよび PE 数の増加に伴ない性能は低下している．また，CLASS W では，CG と同様にワークベクトルがキャッ

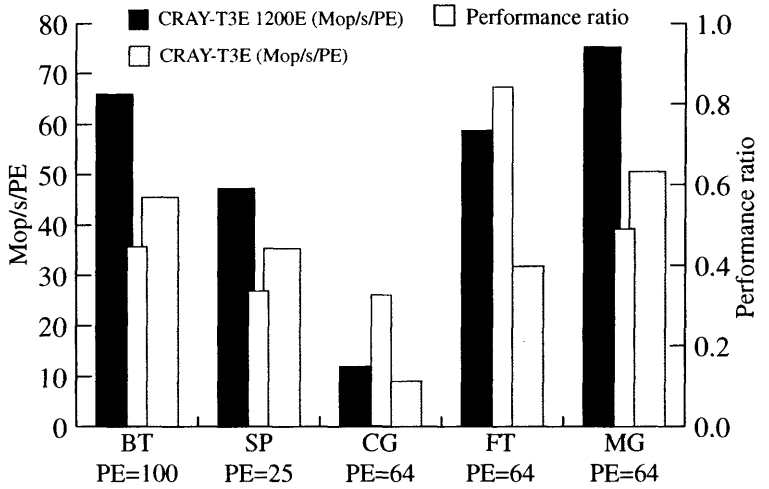

図 6: システム全体の性能比 (CLASS B)

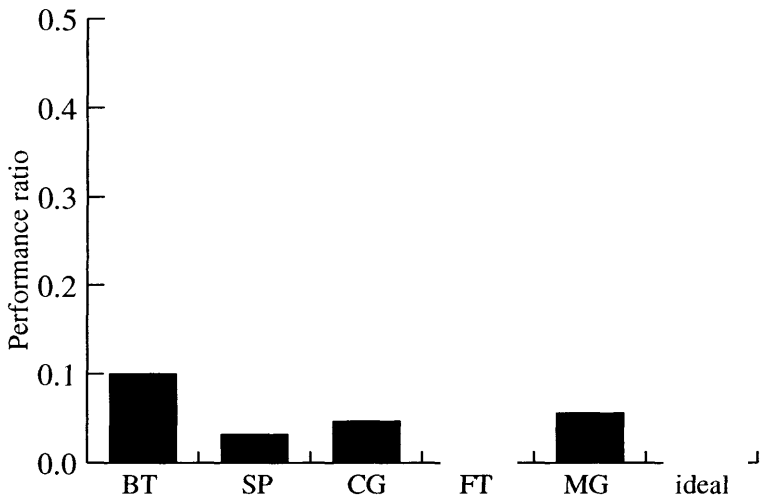

図 7: PE 間通信性能による性能向上率 (CLASS B)

シュサイズ以下に収まったと考えられ，CRAY-T3E/1200E において顕著な速度向上が得られている．CLASS W 以外の問題サイズでは，性能向上は約 20 % である．

3.5 PE 間通信性能

本節では，PE 間通信の性能向上を検討する．各ベンチマークの結果を並べるだけでは，システムの性能向上は CPU の性能向上と通信性能が両方含まれた状態としてあらわされる．CPU の性能向上は 3.3 節で明らかになっている．その結果を元に，システム全体の性能比 (P_{tmp}) から既知の CPU 性能向上の割合 (P_{CPU}) を相殺することにより，通信性能比 (P_{comm}) の向上を類推することができる．つまり，

次式を用いて間接的に推測する．

$$P_{comm} = P_{tmp} - P_{CPU} \tag{2}$$

P_{CPU} として，各ベンチマークを N 個の PE を用いて並列化すると，PE 当りの演算量が基本的には $1/N$ となる．比較のため，演算量が 3.3 節の CLASS W と同程度になる問題サイズと PE 数を用いる．例えば，BT の CLASS W，1 PE で実行する際の演算量は，BT を CLASS A，16 PE で実行する場合の演算量とほぼ同一である．この場合，図 3 に示した CLASS W，1 PE の CPU の性能向上比 P_{CPU} が分かっているので，CLASS A，16 PE による CRAY-T3E と CRAY-T3E/1200E の性能比 P_{tmp} を求め，ここから $P_{tmp} - P_{CPU}$ を計算し，間接的に通信性能比 P_{comm} を評価できる．

PE 間通信性能を評価するため，NPB のうちで通信処理の振る舞いが明確なベンチマークプログラムを用いる．LU は，通信隠蔽処理を行なっているため，理論的には通信時間が陽にあらわれない．また，IS も MPI_Alltoallv の挙動が明確でない．また，EP は，並列実行時にも通信が発生しない．よって，LU，IS，EP 以外のベンチマークプログラムを用いる．用いた問題サイズは CLASS B とし，PE 数は CLASS W の 1 PE のメモリワークサイズと同程度の量となるように，ベンチマークごとに調整する．用いた PE 数は，BT で 100 PE，SP で 25 PE，CG で 64 PE，FT で 64 PE，MG で 64 PE である．図 6 に，ベンチマークプログラムの各 PE 数での性能を示す．横軸に各ベンチマークプログラムと PE 数，左縦軸に Mop/s/PE，右縦軸に Performance ratio を示す．これに式 (2) を適用し，図 6 から図 3 の “1PE での性能比” を差し引いて推測された通信性能比 P_{comm} を図 7 に示す．

図 7 に示されるように，FT の通信性能向上率が最も高い．FT は複素型の大きな配列データを多く通信するため，通信帯域の向上が最も有効に働いたと考えられる．通信性能の向上率は，CRAY-T3E と CRAY-T3E/1200E の通信帯域の性能差に近い値となっている．その他の場合では，およそ 3 ～ 10 % 程度の通信性能向上が見られた．FT 以外では，通信回数が多いため，通信性能の多くは通信帯域ではなくネットワークのレイテンシーに依存するものと考えられる．SP, CG, および MG の結果より，平均的なアプリケーションプログラムを実行した場合の通信性能の向上は，およそ 5 % 前後と推測される．

3.6 総合性能

実際に科学技術計算のアプリケーションプログラムがどの程度の性能向上が得られるかを NPB によって示す．用いたベンチマークプログラムの種類は，得られる

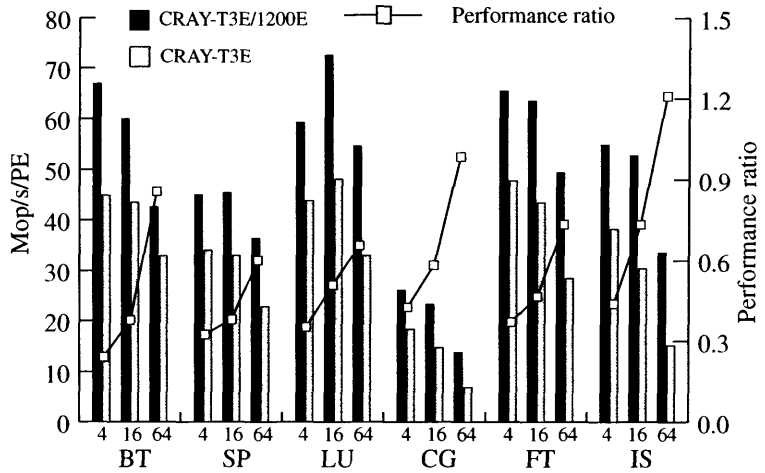

図 8: CLASS W におけるシステム全体性能

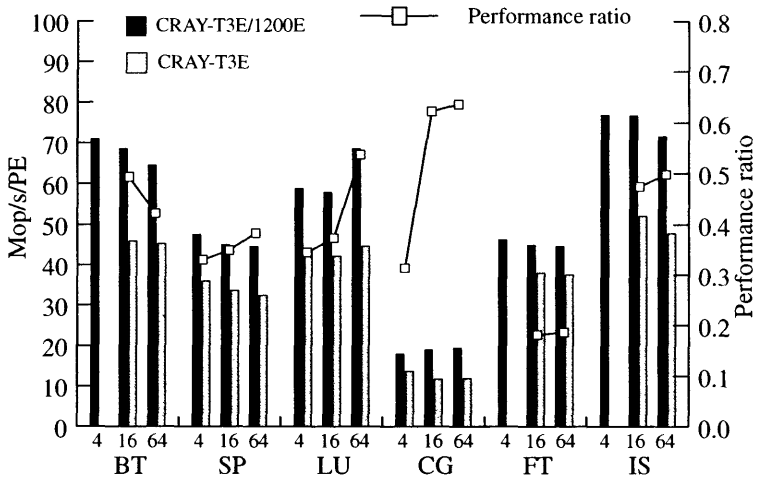

図 9: CLASS A におけるシステム全体性能

Mop/s 値の種類がFlops値とほぼ同一となるものを用いた．図 8 に CLASS W での結果を示し，図 9 に CLASS A での結果を示す．それぞれ横軸に各ベンチマークと PE 数を示し，左縦軸に Mop/s/PE，右縦軸に CRAY-T3E と CRAY-T3E/1200E の性能比を示した．

図 8 では，PE 数の増加に従って，性能比が右上がりに向上した．この理由として，3.4 節で述べたように，キャッシュの影響が大きくなっていることが挙げられる．PE 数が少ない場合で，30 ～ 40 %，PE 数が多い場合では，60%以上の性能向上が得られた．

より大きな問題サイズのクラス A の場合，図 9 に見られるように，PE 数の増加による性能比の向上は，CLASS W より少ない．CG，FT 以外の性能比は，PE 数にかかわらず，35 ～ 50 % 程度である．FT の性能比は他よりも低く，20 % 程度

である．CLASS A における FT では，ワークサイズが大きく，また通信依存性が大きくなるため，高い性能向上が難しいものと推測される．図 5 でも，問題量が大きい場合には同程度の性能比を示した．一方，CG は，16，64 PE において高い性能向上が得られた．これも 図 4 で見られたキャッシュの影響と考えられる．

前節までに PE の単体性能の性能向上と PE 間通信の性能向上を検討した．PE 単体の性能向上は，35 % 程度，通信の性能向上は 5 % 程度と推測された．NPB を用いたシステム全体の性能向上は，40 % 程度であると推測される．

4 おわりに

本論文では，並列処理システムの構成部品の高速化という事例について，その効果を NPB を用いて検討した．一般的に CPU クロックの高速化による性能向上は，メモリ帯域に大きく依存する．CRAY-T3E システムに比べて CRAY-T3E/1200E システムでは，PE の単体性能で多くのベンチマークコードで約 35% 程度の性能向上が得られた．このうち，メモリ帯域に依存するベンチマークコードでは，大きな性能向上は得られなかったが，ワークサイズがキャッシュの容量以下となる場合では，CPU クロックの高速化に見合う性能向上が得られた．通信性能の向上による影響は，通信データ量が非常に大きい場合には，通信性能の向上に比例した高速化率が得られた．しかしながら，通信データ量が少ない場合では，3 ～ 10% 程度の性能向上に留まった．

並列処理システム全体としての性能向上は，通信性能が大きく影響するアプリケーションでは，約 20 % の向上が得られ，NPB の各ベンチマークを平均すると，35 ～ 50 % 程度の性能向上が得られた．

参考文献

[1] http://www.nas.nasa.gov/Software/NPB/

[2] Browning,D.：“NAS Kernels Survey Report”，*Report RND-92-003* (1992)

[3] Bailey,D.，Barszcz,E.，Barton,J.，Browning,D.，Carter,R.，Dagum,L.，Fatoohi,R.，Fineberg,S.，Frederickson,P.，Lasinski,T.，Schreiber,R.，Simon,H.，Venkatakrishnaan,V.，Weeratunga,S. ：“THE NAS PARALLEL BENCHMARKS”，*RNR Technical Report RNR-94-007* (1994)

[4] http://www.cray.com/products/systems/crayt3e/

[5] 寒川 光: “数値計算プログラミングにおけるデータ移動制御のためのブロック化アルゴリズム”, 情処論 Vol. 33, No.10 pp.1183–1192, (1992)