Vol.2012-ARC-202 No.2
Vol.2012-HPC-137 No.2
2012/12/13

レイテンシコアの高度化・高効率化による将来の HPCI システムに関する調査研究のための アプリケーションと性能評価

片桐孝洋† 大島聡史† 中島研吾† 米村崇†† 熊洞宏樹††
樋口清隆†† 橋本昌人†† 高山 恒一†3 藤堂眞治†4，岩田 潤一†5
内田和之†5，佐藤正樹†6，羽角博康†6，黒木聖夫†7

本報告では，レイテンシコアの高度化・高効率化による将来の HPCI システムに関する調査研究におけるターゲットアプリケーションの特徴について，演算パターンと通信パターンの観点からの分類法を提案する．東京大学情報基盤センターに設定された富士通 PRIMEHPC FX10 を用いたプロファイル結果を示し，同計算機でのハードウェア性能からの特徴について紹介する．

1. はじめに

本報告では，数値計算レイテンシコアの高度化・高効率化による将来の HPCI システムに関する調査研究（以降，単に調査研究とよぶ）におけるターゲットアプリケーションにおいて，コンピュータサイエンスの観点からの特徴を示し，アプリケーションの分類法を提案することを目的とする．

アプリケーション特性は，一般的には，シミュレーション分野や，支配方程式の数理特徴から行うことが多い．しかしながら，支配方程式の数理的特徴が同じであっても，計算機上への実装が異なれば，異なる性能挙動を示す．

そこでまず，コンピュータサイエンスでの立場から，アプリケーションで支配的な演算からなる**演算カーネル**と，支配的な通信からなる**通信カーネル**を同定する．次に，演算カーネルでは，配列へのアクセスパターンから分類を行う．通信カーネルでは，通信パターンから分類を行う．

上記の演算カーネルと通信カーネルの分類を用いて，本調査研究で行われるエクサスケールコンピューティングに向けた計算機ハードウェア設計に，アプリケーション特性を反映する．このことで，計算科学とコンピュータサイエンスとの協調設計(Co-design)を実現する．

本報告で提案される分類は支配方程式に依存しないため，本調査研究で取り扱わないアプリケーションの性能予測に活用できる可能性がある．

本報告の構成は以下のとおりである．2 章で調査研究の目的とターゲットアプリケーションを説明する．3 章では，ターゲットアプリケーションの分類と特徴を説明する．4 章では，演算カーネルと通信カーネルの実例を紹介する．5 章は，富士通 PRIMEHPC FX10（以降，FX10）を用いた予備評価の紹介である．最後に，本原稿で得られた知見を述べる．

† 東京大学 情報基盤センター スーパーコンピューティング研究部門
†† 日立製作所 情報・通信システム社
†3 日立製作所 中央研究所
†4 東京大学 物性研究所
†5 東京大学 大学院工学系研究科
†6 東京大学 大気海洋研究所
†7 海洋研究開発機構

2. 調査研究の目的とターゲットアプリケーション

2.1 将来の HPCI システムのあり方の調査研究

本調査研究は，第 4 期科学術基本計画（平成 23 年 8 月 19 日閣議決定）で掲げられた国家存立の基盤としての世界最高水準のハイパフォーマンス・コンピューティング技術の強化，及び科学術基盤の充実強化に向けた重要な取り組みの一つとして，HPC 技術等の HPCI システムの高度化に必要な技術的知見を獲得することを目的とし，平成 24 年度，平成 25 年度に調査研究を実施するものである[1].

2.2 レイテンシコアの高度化・高効率化による将来の HPCI システムに関する調査研究

本調査研究は，東京大学を中心とし，九州大学，富士通，日立製作所，日本電気による調査研究である[2]. 2018 年頃設置可能な並列システムを，汎用型プロセッサからのアプローチでフィージビリティ・スタディ（FS）を行う．アプリケーション，システムソフトウェア，アーキテクチャの co-design を行う．システムソフトウェアスタック共通化(From PC cluster to high-end machines)を行う．

本報告はこのうち，アプリケーション性能予測に関する検討事項に相当する．

2.3 本調査研究における進め方

「今後の HPCI 技術開発に関する報告書」[3][4]を尊重し，京および FX10 におけるアプリケーション並列性能および I/O 性能，耐故障性および運用・保守の観点で課題を精査し，概念設計に反映する．進め方の概要を図 1 に示す．

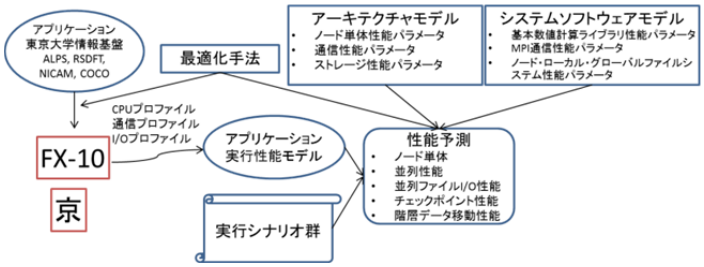

図 1　本調査研究における調査研究の進め方

本報告は，図 1 における性能予測のための手法と最適化方式の調査研究に関連する．

2.4 利用シナリオ

利用シナリオ（図 1 における「実行シナリオ群」）とは，各アプリケーションの実行において，特徴的なジョブの実行形態のことである．主に以下のアンサンブル型を想定し，入出力ファイルなどの I/O 性能を含め，システム全体の設計に反映させる．

- **アンサンブル型：**
  全系の 1/10～1/100 の資源を利用する 1 ジョブに対し，複数ジョブを同時実行して全資源を使い切る形態．この形態では，複数同時のファイル入力，および複数同時のファイル出力が起こる．

2.5 性能予測手法

図 1 を進めるに当たり，ターゲットアプリケーションにおける概念設計中の計算機性能の実行時間を予測するため，以下の手法をとる．

1. **ホットスポット同定：**基本プロファイラ（主要な関数やループの実行時間が取得可能な性能プロファイラ）を用いて，複数のホットスポット（ループレベル）を同定する．その後，全体性能の予測をホットスポットのみで行う．
   - ホットスポットの部品化を行う．できるだけ，採用されている数理アルゴリズム（支配方程式，離散化方法）とホットスポットの対応がわかるようにする．
2. **ホットスポット分離：**計算部分，通信部分，I/O 部分のホットスポットを基のソースコードから分離する．
   - 計算部分：演算カーネルと呼ぶ．
   - 通信部分：通信カーネルと呼ぶ．
   - I/O 部分： I/O カーネルと呼ぶ．
3. **通信パターン確認：**プロファイラによる可視化ツールや対象コードを解析することで，通信バターンを確認する．
4. **詳細プロファイルと分析：**詳細プロファイラ（対象のループにおけるハードウェア上の性能情報が取得可能な性能プロファイラ）を用い，ホットスポットごとにハードウェア性能情報を取得して分析する．
   - 演算カーネルにおける，演算効率／命令発行量／キャッシュ利用効率，など．
   - 通信カーネルの，通信回数／量／通信待ち時間，など．
   - I/O カーネルの，データ読み書き，量／頻度，など．
5. **ベンチマーク化：**ホットスポットのみで動作するようにコードを再構成する．
   - マシン特化の書き方，および，汎用的な書き方，の２種を区別する．
   - 演算カーネル，通信カーネル，I/O カーネルの分類をする．
6. **詳細モデル化：**ハードウェア因子による実行時間の予測ができるようにする．

本調査研究では，詳細モデル化を行うに当たり，FX10 で提供される性能プロファイラを用いる．このことで，演算および通信カーネルを抽出できる．また，現存する計算機でのハードウェア因子について，プロファイラを通じて取得ができる．この情報を基に，現在設計中の計算機での性能予測が可能となる．

2.6 ターゲットアプリケーションの特徴

以下にターゲットアプリケーションの計算機システムに対する性能要求をまとめる．なお，メモリ帯域，FLOPS，Byte/Flops (B/F) 値は，「計算科学ロードマップ白書」（2012 年 3 月）[5]における見積値，もしくは，本調査研究でアプリケーション開発者自身が行った見積値である．

(1) ALPS (Algorithms and Libraries for Physics Simulations)

新機能を持った強相関・磁性材料の物性予測・解明のシミュレーションである．虚時間経路積分にもとづく量子モンテカルロ法と厳密対角化を利用している．

- 総メモリ：10～100PB.
- 整数演算，低レイテンシ，高次元のネットワーク．
- 利用シナリオ:1 ジョブ 24 時間, 生成ファイル 10GB. 同時実行 1000 ジョブ，総合生成ファイル：10TB.

(2) RSDFT (Real-Space Density-Functional Theory)

Si ナノワイヤ等，次世代デバイスの根幹材料の量子力学的第一原理シミュレーションである．実空間差分法を利用している．

- 総メモリ：1PB.
- 演算性能：1EFLOPS （B/F = 0.1 以上）.
- 利用シナリオ:1 ジョブ 10 時間, 生成ファイル 500TB. 同時実行 10 ジョブ，総合生成ファイル 5PB.

(3) **NICAM (Nonhydrostatic ICosahedral Atmospheric Model)**

長期天気予報の実現，温暖化時の台風・豪雨等の予測のシミュレーションである.正 20 面体分割格子非静力学大気モデルを採用し，水平格子数 km で全球を覆い，積雲群の挙動までを直接シミュレーションする.

- 総メモリ：1PB，メモリ帯域：300PB/sec.
- 演算性能：100PFLOPS（公称値 B/F = 3).
- 利用シナリオ:１ジョブ 240 時間,生成ファイル:8PB. 同時実行 10 ジョブ，総合生成ファイル 80PB.

(4) **COCO (CCSR Ocean COmponent Model)**

海況変動予測，水産環境予測のシミュレーションである.外洋から沿岸域までの海洋現象を高精度に再現し，気候変動下での海洋変動を詳細にシミュレーションする.

- 総メモリ：320TB，メモリ帯域：150PB/sec.
- 演算性能：50PFLOPS（B/F = 3).
- 利用シナリオ:１ジョブ 720 時間,生成ファイル 10TB. 同時実行 100 ジョブ，各生成ファイル 1PB.

3. ターゲットアプリケーションの特徴と分類

ここでは，ターゲットアプリケーションを分類するに当たり，実行時間のうち優位に大きな時間を占める演算ループ（**演算カーネル**）と，主要な通信のパターンを形成するループ（**通信カーネル**）を抽出することで，処理の抽象化を行うことを目的とする.

3.1 演算カーネルにおける配列アクセスパターン

演算カーネルにおける配列アクセスパターンにより，メモリに対する性能要求が変わるので，それで分類すべきである.たとえば,（1）連続アクセスで，かつキャッシュにデータを搭載できるようにループを細切れに分割可能なアクセス;（2）連続アクセス;（3）ストライド・アクセス;（4）ランダム・アクセス;が考えられる.

3.2 通信カーネルにおける通信パターン

通信パターンにより，通信網に対する要求性能が変わるので,それで分類すべきである.たとえば,（1）隣接通信;（2）クラスタ内は密通信だがクラスタ間は隣接通信;（3）規則通信（バタフライ形状，など),（4）複数同時の集団通信;（5）全体全通信;が考えられる.

3.3 演算・通信の分類

以上の演算・通信のパターンを考慮すると，以下の分類が考えられる.

(1) **演算パターンの分類**

- 主演算の種類
  - **整数演算，浮動小数点演算**
- 配列アクセスパターンの種類
  - **密行列-行列積系**
    - ほとんどが密行列 - 行列積で構成される演算．たとえば，密行列に対する LU 分解.
    - 演算の割合が高い.
    - 連続アクセス,キャッシュブロック化可能.
    - 行列サイズが反復ごとに縮小する場合と，行列サイズはほとんど減少しない場合がある.
  - **ステンシル演算系**
    - 演算パターンは規則的（ステンシル演算）.配列アクセスも規則的.
    - メモリアクセスの割合が高い.
    - 連続アクセス，等間隔アクセス，の 2 種がある.
  - **疎行列 - ベクトル積系**
    - 間接参照，非連続アクセス.
    - メモリアクセスの割合が高い.
    - 扱う問題の性質に依存し，オーダリングにより，メモリアクセスの局所化が可能．この場合は，一部連続アクセス化ができる.
    - アクセス局所化不可能の場合は，ランダム・アクセスになる.
  - **探索系**
    - 整数演算.
    - ランダム・アクセス.
    - IF 文成立が予測不可能.

(2) **通信パターンの分類**

- **隣接通信系**
  - 論理的に隣接する計算領域を持つプロセスのみ通信を行う.
  - メッシュ形状（離散化方式），計算領域分割形状に依存し，送るべき相手の数の違いが生じる.
- **規則通信系**
  - 規則的な通信パターン.
  - 同時，もしくは一連の通信処理をまとめたフェーズ（**通信フェーズ**）において，全プロセスにデータを送ることがない.
  - 通信フェーズは，複数の段数に分割されていることがある.
  - バタフライ形状，リング形状，ツリー形状，などが代表的.
- **同時通信系**
  - 2 次元，もしくは，3 次元以上の次元で MPI プロセスを分割したグリッド（**プロセス・グリッド**）において，プロセス・グリッドを考慮したプロ

セス群の分割がある．

- 上記のプロセス・グリッドに対して，同時に集団通信（放送やリダクション）を発行する．
- たとえば，2 次元プロセス・グリッド分割時に，（1）対角要素のみ，（2）行方向のみ，（3）列方向のみ，など，同時に通信する方向に対する任意性がある．

- **全対全通信系**
  - 各プロセスが通信フェーズにおいて，自分以外のすべてのプロセスに通信を発行する．

3.4 分類法に関する議論

演算パターンおよび通信パターンの分類は，処理の抽象化であり，パターンが単純であるほど効果を奏する．反面，パターンが単純すぎると，処理の現状を反映しなくなる．

パターンの単純性と分類による抽象化の精度は，トレードオフであり，中庸となる分類法の確立自体が学術的な挑戦である．

3.3 節の演算カーネルの分類では，ループの構造，たとえば，最内ループに IF 文がある／ないなどの性能劣化させる要因の考慮が無い．著者が知る限り，ステンシル演算で IF 文などのループ構造に着目した分類がない．また，B/F 値を上昇させる要因となる，同時に参照する配列数も考慮する必要がある．

3.3 節の通信カーネルの分類では，通信の頻度（回数）と 1 回当たりの通信量（何バイト）が考慮されていない．したがって，示されたパターンのみだけでは正確な抽象化とならないことがある．

一方，3.3 節の分類の利点は，扱う支配方程式や，問題の数理的性質に依存せず，パターンの種別のみで対象となるアプリケーション性能について，設計中の計算機上での性能を概観することができる．これは，コンピュータサイエンスの視点から重要な事項である．

3.5 ターゲットアプリケーションの分類

前節の内容を考慮し，本調査研究のターゲットアプリケーションの分類を，以下の表 1 と表 2 にまとめる．

表 1　演算カーネルの分類

アプリ名	演算種類	配列アクセスパターン	特徴
ALPS/ Looper	整数	探索系	IF 文成立事前予測不可能
RSDFT	浮動小数点	密行列-行列積系	固有値ソルバー，GS 直交化
NICAM	浮動小数点	ステンシル演算系	力学過程（最内 IF 文あり／なし），物理過程（ループ内演算多数）
COCO	浮動小数点	ステンシル演算系	最内 IF 文，同時参照配列多数

表 2　通信カーネルの分類

アプリ名	通信パターン	特徴
ALPS/ Looper	規則通信系	バタフライ形状，オーバラップ
RSDFT	同時通信系	行／列方向の別，Bcast と Allreduce
NICAM	隣接通信系	主に 6 方向．最大で 15 方向．
COCO	隣接通信系	最大で 4 方向

4. 実例

4.1 ALPS/Looper

(1) 概要

ALPS/Looper の主要な演算は，状態を表現する構造（世界線）のループ認識とループ更新となる．そのため，ループのクラスタリング処理のアルゴリズムの実行時間が多い．ALPS/Looper では，ループのクラスタリング処理に union-find アルゴリズムを使用しているため，この処理が主要な演算となる．

並列化は，世界線の領域分割を基本としている．ループ構造の認識のため，領域間で通信が必要になる．この通信は，バタフライ形状(binary-tree algorithm)で実装されている．

(2) 演算カーネル

主な演算はループの認識と更新のために行う探索処理であり，探索処理特有の演算カーネルをなす．また主演算は整数演算である．

演算カーネル中の IF 文の成立は入力データ依存である．IF 文成立の予測できれば探索の必要がないため，予測困難な成立の IF 文が含まれている．

(3) 通信カーネル

通信カーネルはバタフライ形状の通信を行っている．通信フェーズでの通信は，同時に行われる実装になっている．

4.2 RSDFT

(1) 概要

RSDFT の主演算は，実数対称標準固有値問題における多数の固有値と固有ベクトルをもとめる演算（固有値ソルバー）と，固有ベクトルの直交化のための演算（直交化ルーチン）からなる．また，自己無撞着計算のために，CG 法を利用している．

固有値ソルバーと直交化ルーチンは，キャッシュブロック化アルゴリズムを採用しているため，多くはBLAS (Basic Linear Algebra Subprograms) のレベル 3 演算（密行列の行列 - 行列積）で構成されている．固有値ソルバーおよび直交化における処理の一部は，BLAS レベル 2 演算（密行列の行列 - ベクトル積）がある．

並列化は，密行列におけるブロック・サイクリック分散で行われている．主要な通信は，2 次元プロセス・グリッドを前提とし，固有値ソルバー部分では，複数同時の放送処理が主体である．直交化処理部分では，複数同時のリダクション処理が主体である．

(2) 演算カーネル

RSDFT の演算量の観点での主演算は，固有値ソルバーと Gram-Schmidt 法による直交化の部分となる．双方の主演算は行列 - 行列積（BLAS3 演算，dgemm）である．

固有値ソルバー部分の全体サイズはブロックサイズごとの呼び出しで，反復ごとに縮小していく（Cf. LU 分解）が演算自体は，あるブロックサイズで何度も BLAS3 が呼ばれる実装となっている．

(3) 通信カーネル

2 次元のプロセス・グリッド配置において，行もしくは列方向に同時に集団通信が発行される．固有値ソルバーでは放送処理が発行されるが，直交化部分ではリダクション演算が発行される．

4.3 NICAM

(1) 概要

NICAM は，完全圧縮性非静力学方程式を支配方程式とし，離散化方法は有限体積法である．水平格子に修正型正 20 面体格子を採用し，鉛直格子にローレンツ格子を採用している．数値計算アルゴリズムは陽解法であるが，タイムステップによる陽解法の数値不安定性を取り除くため，遅いモードは陽解法である Runge-Kutta 法を採用し，速いモードでは，水平陽解法鉛直陰解法（HEVI 法）を採用している．

計算の特徴として，力学過程と物理過程がある．力学過程では Navier-Stokes 方程式を陽解法で解いており，頻繁に通信を行う．一方，物理過程では通信を含まず，解法としては HEVI 法による陽解法と陰解法の混合方式である．

(2) 演算カーネル

- **力学過程**

陽解法のステンシル演算が主演算となる．ただし，IF 文が内部に存在しないシンプルな実装と，IF 文が内部に存在する実装の 2 つが混在しており，最適化の観点から両者を区別する必要がある．同時に参照される配列数が 1 個のものでは，以下の 2 種のアクセスパターンがある：(i)ストライド・アクセス；(ii)連続アクセスで最内 IF 文があるもの．

その他は，4 配列更新，連続アクセス，かつ最内 IF 文あり，3 配列更新かつ連続アクセス，などの変種がある．

図 2 に，上記(i)のループ構成(zcode/mod_oprt.f90.DEF 中の主要ループ)を示す．また，図 3 に上記の(ii)のループ構成（zcode/mod_oprt.f90.DEF 中の主要ループ）を示す．

```
do l=1,ADM_lal
  do k=1,ADM_kall
    do n=nstart,nend
     ij=n; ip1j=ij+1; ip1jp1=ij+1+ADM_gall_1d;
     ijp1=ij+ADM_gall_1d; im1j=ij-1;
     im1jm1=ij-1-ADM_gall_1d; ijm1=ij-ADM_gall_1d;
     scl(n,k,l)=( cdiv(0,ij,l,1)*vx(ij     ,k,l)
+cdiv(1,ij,l,1)*vx(ip1j   ,k,l)+cdiv(2,ij,l,1)*vx(ip1jp1,k,l)
+cdiv(3,ij,l,1)*vx(ijp1    ,k,l) +cdiv(4,ij,l,1)*vx(im1j   ,k,l)
+cdiv(5,ij,l,1)*vx(im1jm1,k,l) +cdiv(6,ij,l,1)*vx(ijm1    ,k,l)
..)*fact
    …
```

図 2 ステンシル演算でストライド・アクセスのループ構成

```
do l=1,ADM_lal
  do k=1,ADM_kall
    do n=nstart,nend
     ....
      s_m1_k_min_n = min(s_in_min(n,k,l,1),
  s_in_min(n,k,l,2),
          s_in_min(n,k,l,3)...
      if (s_m1_k_min_n==CNST_MAX_REAL)  then
        s_m1_k_min_n = s(n,k,l) else … endif
      c_out_sum_n =
  (0.5D0+sign(0.5D0,c(n,k,l,1)))*c(n,k,l,1) +
  (0.5D0+sign(0.5D0,c(n,k,l,2)))*c(n,k,l,2) +
  (0.5D0+sign(0.5D0,c(n,k,l,3)))*c(n,k,l,3)   ...
  if ( abs(c_out_sum_n) < CNST_EPS_ZERO) then
  …
```

図 3 ステンシル演算で連続アクセス，かつ最内 IF 文があるループ構成

- **物理過程**

陰解法を含むが，連立一次方程式の解法は 3 重対角行列に特化された実装になっているため，連立一次方程式の求解の効率は良い．雲の存在する箇所の演算が多く，演算ロードバランスが悪い状況が生じる．

プロファイルにより同定した演算カーネルの特徴は，2 配列同時アクセス，かつ連続アクセスの構成である．スカ

ラーの演算が多数あり，最内に IF 文を含む実装となっている．そのため，実装方式とコンパイラ最適化に大きく影響を受けることが予想される．

図 4 に，上記のカーネル（nhm/physics/mod_mp_nsw6.f90 中の主要ループ）の構造を示す．

```
do k = kmin, kmax
  do ij = 1, ijdim
    temc = tem(ij,k)-CNST_TEM00
    多数の演算
    V_TERM(ij, k, I_QR) = - (cr * rho_fact * gam_br_dr_1
          / gam_br_1 * (olambdr_dr))
    多数の演算
    if( cnst_v_term_qi==cnst_undef ) then
      V_TERM(ij,k,I_QI) =
           - 3.29D0 * abs( rho_a*tmp ) ** 0.16D0
        else
      ...
  if ( temc > 0.0D0 ) then
      多数の演算
  else
          ....
```

図 4 物理過程における演算カーネルのループ構成

(3) 通信カーネル

力学過程の通信形態は隣接通信である．領域分割の形状から，原理的には 6 方向への通信となるが，最大で 15 方向の通信を含む．

4.4 COCO

(1) 概要

Tripolar 格子を用い，極部分を展開し長方形の直交格子に対する差分法（陽解法）を用いている．水平方向の 2 次元領域分割により並列化を行っている．分割領域の計算は，周囲の 2 格子分のデータ（袖領域）が必要であり，隣接領域と通信する．モデル計算における通信は，初期条件・境界条件の読み込みと結果出力の I/O のための gather，scatter（これは，MPI_GatherV，および，MPI_ScatterV で実装）以外の通信はない．

(2) 演算カーネル

COCO の演算カーネルは，トレーサー移流スキーム SOM のプログラム（src/option/tflxt.iso-som.F）の中にある 3 つのループ（flxomp2，flxomp3，flaxomp5）である．そのうちの 1 つ（flxomp5）については，キャッシュブロック化の実装が施されている．

演算カーネルの配列アクセスパターンは連続アクセスである．ループの構造は，この 3 演算カーネルすべてにおいて内部に IF を含む．また，同時にデータを更新する配列数が最大で 11 個も存在し，B/F 値を押し上げる要因になっている．

図 5 に，ブロック化されている flaxomp5 のループ構造を示す．

```
DO IJ1 = IJTSTR, IJTEND, IBLOCK
  DO K = KSTR, KEND
    DO IJ = IJ1, IJ2
      S0M = S0(IJ, K, N) – MIN( S0(IJ, KU, N) / SM(IJ, KU, N), …
      …
      IF ( ABS( SZ(IJ, K, N) ) < 1.5D0 * S0M ) THEN
        SZZ(IJ, K, N) = MIN( S0M + SXP,
            MAX( ABS( SZ(IJ, K, N) ) - S0M, SZZ(IJ, K, N) ) )
      ELSE
        SZZ(IJ, K, N) = MIN( S0M + SXP,
            MAX( S0M - SXP, SZZ(IJ, K, N) ) )
      END IF
    …
    ENDDO
  ENDDO
  …
ENDDO
```

図 5 flaxomp5 のループ構成

● 通信カーネル

通信カーネルは，隣接通信である．通信する相手の数は，領域分割の方法に依存する．ここでは，モデルの東西，および南北に 4 分割することを想定する．この場合，最大で隣接 4 方向（一部は 3 方向）の通信となる．

4.5 コード最適化に関する議論

前節の演算カーネルのコードについて，実装方式と数値計算アルゴリズムについて注意する必要がある．効率の悪い実装方式や数値計算アルゴリズムを採用すると，無駄に B/F 値が高くなる．その結果，設計中の計算機ハードウェアの本来の性能が，アプリケーションの観点から反映できなくなる．

実装方式であるが，採用する数値計算アルゴリズムが最適であっても，演算性能はコードの書き方とコンパイラ最適化の能力に強く依存する．コンパイラ最適化が十分にされていないコードをプロファイルすれば，設計中の計算機の能力を十分に引き出した評価ができない．また十分に性能最適化が出来るコードであると思っても，採用するコンパイラの最適化能力が不十分であれば，同様に効果を奏しない．ターゲットとなるハードウェアに向けた最適化が施されていることが，計算機評価の前提である．

これらの要求は計算機ハードウェア設計時のみの問題

ではなく，計算機開発後の運用時にも問題となる．

計算機ハードウェアやコンパイラ最適化能力などの計算機環境に依存せずコード最適化が行える能力は**性能可搬性 (Performance Portability)** と呼ばれる．性能可搬性は，ソフトウェア自動チューニング(Software Auto-tuning, (AT))の主要な課題として知られている[4][6]．ここでの主張は，ハードウェア設計時においてさえも，AT 技術の適用がなされる仮定をおくことは重要なことである．

一方，数値計算アルゴリズムの最適化はソフトウェア性能に劇的な改善を与える．ターゲットとなるハードウェアに適合しない数値計算アルゴリズムを採用した場合は，同じ計算を達成できる別のアルゴリズムへ変更することが必須となる．

以上を考慮し，設計時の計算機構成の変更と，ハードウェア性能を引き出す数値計算法の選択が，コンピュータサイエンス側の研究者と計算科学側の研究者による Co-design 時の重要な事項となる．

5. 予備評価

5.1 概要

本章では，FX10 を用いて各アプリケーションの性能について予備評価を行った結果（2012 年 10 月 30 日現在）を紹介する．なお，各アプリケーションの実行形式（ハイブリッド MPI 実行時の MPI プロセス数と各 MPI プロセスから起動されるスレッド数の組合せ）や，利用する問題サイズなどを，アプリケーション開発者の要求から変更し再評価する予定である．したがってこの予備評価結果は，必ずしもエクサスケール実行を想定した最終的なアプリケーション性能の解析結果ではない．

5.2 計算機環境

東京大学情報基盤センターに設置された FX10 を，48 ノード（768 コア）まで利用した．FX10 の性能は以下のとおりである．

システムの全体性能は 1.135 PFLOPS，総主記憶容量は 150 TB，総ノード数は 4,800 である．インターコネクトは，6 次元メッシュ / トーラス(TOFU ネットワーク)である．ノードの全体性能は，理論演算性能は 236.5 GFLOPS，プロセッサ数 (コア数)は 16，主記憶容量は 32 GB である．プロセッサは SPARC64 IXfx，周波数は 1.848 GHz，理論演算性能 (コア)は 14.78 GFLOPS である．

富士通社製コンパイラを利用し，プロファイルには富士通社の Technical Computing Suite V1.0 の性能プロファイラ（基本プロファイラ，詳細プロファイラ）を利用した．

アプリケーションによっては，入力データに依存し MPI プロセス，および MPI プロセスから派生するスレッド毎に計算量の不均衡が生じる．そこでプロセス 0 において，FLOPS 値およびメモリスループット値の最大値を代表値にした．

5.3 ALPS/Looper のプロファイル結果と考察

計算規模は，L=524288，T=0.00083 である．実行形式は 48 ノード実行，48MPI プロセスを起動し，各 MPI プロセスは 16 スレッド実行を行うハイブリッド MPI 実行で全コアを使い切る．

ALPS/Looper は探索処理の性質上，浮動小数点演算がきわめて少なく整数演算が多い．そのため，実行性能の参考として GFLOPS 値ではなく，GIPS (Giga Instruction per second) 値を載せることにする．

表 3 に，基本プロファイラにより同定した上位の演算カーネルとその性能を載せる．表 4 に，基本プロファイラにより同定した通信カーネルにおける通信情報（1 回当たりのメッセージ量と，メッセージ量ごとの通信回数）の平均(AVE)，最大(MAX)，最小 (MIN) を載せる．

表 3 では，2 つの探索処理の演算カーネル(OMP_48, OMP_44)がある．両者は演算の性質が異なる．OMP_48 は，メモリスループットがピークに対して約 28%と高いが，その反面，GIPS 値が低い．一方，OMP_44 は，メモリスループットがピーク性能に対して約 9.4%と低いが，GIPS 値が高い．そのため，OMP_48 はメモリ束縛の演算カーネル，OMP_44 は演算束縛な演算カーネルと考えられる．

詳細プロファイラによる実行時間の分析では，OMP_44 においては，各スレッドで実行時間は一定で「整数ロードキャッシュアクセス待ち」が多いのに対し，OMP_48 では「ストア待ち」が多く，かつ各スレッドの実行時間がばらついている。

表 4 は，通信カーネルにおける通信情報である．ALPS/Looper では，同一メッセージ量をバタフライ形状で通信するため，メッセージ量は MIN，MAX で等しい．1 回当たりのメッセージ量が 4KB 以下の呼び出し回数が 1/3，1024KB 以上の呼び出し回数が 2/3 の割合で存在する．全体時間に対し MPI 通信時間の占める割合は約 1.3%である．

5.4 RSDFT のプロファイル結果と考察

空間を 3 次元分割した実行方式である．計算規模は 5832 原子，MB=12830 である．実行形式は 48 ノード，192MPI プロセスを起動し，各 MPI プロセスは 4 スレッド実行のハイブリッド MPI 実行で全コアを使い切る．空間の分割数は，4x6x8 である．

表 5 は，演算カーネルとその性能，表 6 は，通信カーネルにおける通信情報を載せる．

表 5 では，RSDFT における演算カーネルは，固有値ソルバーで使われている dgemm 演算部分 (diag_2d_dgemm_620) と，Gram-Schmidt 直交化で使われている dgemm 演算部分 (gram_schmidt_sub_dgemm_343) となるため，性能も dgemm

の性能そのものになる．B/F 値は 0.07 と低く，RSDFT 全体でみても B/F 値が 0.12 である.演算カーネル実行が全体時間に占める割合は双方で 23.5%である．しかし BLAS3 の演算カーネルが分散しており，その他の dgemm 実行時間と合わせると約 40%を占める．

表 6 では，主な通信は 2 次元プロセス・グリッド上で同時に発行される Bcast と Allreduce となる．Isend，Irecv は有限差分法計算で用いられているが，全体に占める割合は少ない．通信時間のうち約 30%が，固有値ソルバー部分で用いられるBcastで,1回当たりのメッセージ量は1024KB以上である．全体時間に対し MPI 通信時間の占める割合は 16.6%である．

5.5 NICAM のプロファイル結果と考察

計算規模は，g-level=9（水平格子：2621442），鉛直 40 層である．実行形式は 40 ノード，160MPI プロセスを起動し，各 MPI プロセスは 4 スレッド実行を行うハイブリッド MPI 実行で全コアを使い切る．

表 7 は，演算カーネルとその性能，表 8 は，通信カーネルにおける通信情報を載せる．

表 7 では，NICAM は演算カーネルが分散しており，全体時間に対して 5%程度の時間を占めるループが多数ある．そこここでは上位の 3 つについて，力学過程 2 つと，物理過程 1 つのものを紹介するに留める．今後，NICAM 開発者と理化学研究所の協力のもと，NICAM の主要演算カーネルを再度同定する予定である．

表 7 では，力学過程における典型的なステンシル演算カーネル（mod_oprt_01）は，演算の対ピーク性能が約 9.2%，B/F 値も 3 程度で，良好な性能特性を示している．一方，ステンシル演算だが最内に IF 文が存在する演算カーネル（mod_oprt_03）は，演算の対ピーク性能が約 3.7%，B/F 値も 4.4 程度になり，性能は悪化する．ただし，NICAM 全体の対ピーク性能と B/F 値は，最内に IF 文があるステンシルの mod_oprt_03 に近い値になっているのは興味深い．

一方，力学過程の演算カーネル（mod_mp_nsw6）は，最内部に IF 文があり,かつステンシル演算とは異なる多数のスカラー演算からなる．演算の対ピーク性能が約 8.3%，メモリスループットの対ピーク性能は 58%である．これは，メモリ束縛のカーネルである．

表 8 では，通信カーネルにおける 1 回あたりのメッセージ量の MAX，MIN がばらついている．これは各プロセスが担当する領域により,転送相手が 6 個から 15 個まで変動するためと予想される．また，メッセージ量ごとの回数もばらついている．全体時間に対し，MPI 通信時間の占める割合は約 5.3%である．

5.6 COCO のプロファイル結果と考察

計算規模は，1440x1320x66 で，160 ステップである．実行形式は 48 ノード，48MPI プロセスを起動し，各 MPI プロセスは 16 スレッド実行を行うハイブリッド MPI 実行で全コアを使い切る．緯度 8 分割×経度 6 分割である．

表 9 は，演算カーネルとその性能，表 10 は，通信カーネルにおける通信情報を載せる．

表 10 では，ブロック化をしていない 2 つのカーネル（flxomp2，flxomp3）に対し，ブロック化を行っているカーネル flxomp5 は，B/F 値の観点で改善（5.9 に削減）がなされており，ブロック化の効果がある．なお，3 カーネルの実行時間の割合が合計でも約 27%と低いが，これは本予備評価の実行では，チェックポイント・リスタートのための Gather と Scatter を行う処理（単に通信だけでなく，通信のためのメッセージの梱包と解凍をする処理を含む）の頻度が高く，この処理時間が全体の実行時間に対し約 45%にもおよぶためである．大規模実行条件では，このチェックポイント・リスタートの時間間隔を減らすことができるので，表 10 の 3 つの演算カーネルの比重は多くなる．したがって，これらを演算カーネルといってよい．

表 11 から，COCO の主要な通信は Isend，Irecv の部分である.Isend の部分の 1 回当たりのメッセージ量は MAX,MIN の差が少ないので，通信量は各 MPI プロセスでほぼ一定と考えられる．ただし，メッセージ量ごとの通信回数をみると，1024KB 以下でばらついている．MPI 通信時間の占める割合は 4.6%である．

6. 関連研究

アプリケーションの特性を，コンピュータサイエンスの立場で分類した研究は多くない．著名なものとしては，Berkeley Motif[7][8]がある．

Berkeley Motif では，アプリケーションの特徴を，メモリ・アクセスパターンと通信パターンで分類している．その結果，HPC 分野で主要となる分類について 7 種類を定めている．また，HPC 分野から拡張した処理を含めた合計 13 種類を，アプリケーションの抽象化（Dwarf）として定義している．これらの抽象化を，計算機設計や計算機システム評価に生かそうとしている．

本報告における分類と，上記の Berkeley Motif とは方向は同じである．違いは，演算カーネルに IF 文のある／なしや，同時に参照する配列数など，特に演算カーネルのループ構成に注目したことである．演算カーネルのループ構成について考慮し，より有効となるアプリケーション分類として細分化を狙うのが，本報告で提案する分類である．

7. おわりに

本報告では，数値計算レイテンシコアの高度化・高効率化による将来の HPCI システムに関する調査研究における

計算機設計時に，コンピュータサイエンス研究者と計算科学研究者との協調設計(Co-design)で必要となるアプリケーション特性について，メモリ・アクセスパターンと通信パターンの観点から分類する方式を提案した．また，富士通 FX10 でのプロファイル結果について，予備評価の結果を紹介した．

本報告を通じて判明した問題点は，以下の通りである．

- **演算カーネルの観点**
  - 探索処理で整数演算を主体とする演算カーネルについて，性能評価指標をどう設定するか．
  - 陽解法やステンシル演算に分類される演算カーネルについて，実際のアプリケーションでは最内部に IF 文が存在する．この場合，性能を悪化させる要因となるので，単純なステンシル演算の演算カーネルとは別の分類，もしくは，細分化した分類とすべきである．
- **通信カーネルの観点**
  - 隣接通信のパターンでも，同時に通信する相手の数がプロセスごとに異なる．このことで，通信回数，通信量，1 回当たりの転送サイズがプロセスごとにばらつく．これらを考慮した通信モデルを構築する必要がある．
  - 集団通信については，集団通信（たとえば，MPI_Allreduce）が全ノードを対象に 1 回発行される事例ではなく，2 次元プロセス・グリッド上で，同時に多数の集団通信を行う事例があった．このような状況での通信時間のモデル化が必要である．

以上から，従来の分類法にとらわれることなく，実アプリケーションのループ構成を基にした，新しいアプリケーションの分類を考えていく必要がある．

今後の課題は山積している．まず，判明した演算カーネルのボトルネックの詳細解析と，ターゲットとなる計算機アーキテクチャに対するコード最適化を実施することで，計算機設計の根拠となる性能をより妥当なものにする必要がある．次に，実アプリケーション上でのファイル I/O のパターンを明らかにすることで，計算機システム設計へ反映する必要がある．最後に，得られた演算カーネルや通信カーネルのみで実行できる，カーネルレベルのベンチマークを作成することがある．このベンチマーク化で，ハードウェアからソフトウェアへ至る広範な計算機設計について，アプリケーション特性の反映を推進していく必要がある．

謝辞 本研究を行うに当たり，富士通 PRIMEHPC FX10 の性能プロファイラ情報など多数のご支援をいただいた富士通社の諸氏に感謝いたします．

本研究は，文部科学省「将来の HPCI システムのあり方の調査研究」（平成 24 年度～平成 25 年度）の支援による．

参考文献

1) http://www.mext.go.jp/b_menu/houdou/24/06/1322138.htm
2) http://www.mext.go.jp/b_menu/shingi/chousa/shinkou/028/shiryo/__icsFiles/afieldfile/2012/08/14/1324574_2_1.pdf
3) http://www.mext.go.jp/b_menu/shingi/chousa/shinkou/020/shiryo/__icsFiles/afieldfile/2012/04/12/1319671_03.pdf
4) HPCI 技術ロードマップ白書，2012 年 3 月．http://open-supercomputer.org/wp-content/uploads/2012/03/hpci-roadmap.pdf
5) 計算科学ロードマップ白書，2012 年 3 月．http://open-supercomputer.org/wp-content/uploads/2012/03/science-roadmap.pdf
6) 片桐孝洋：ソフトウエア自動チューニングー数値計算ソフトウエアへの適用とその可能性ー，慧文社，2004 年 12 月，ISBN4-905849-18-7.
7) Asanovic, K., Bodik, R., Demmel., J., Keaveny., T., Keutzer, K., Kubiatowics, J.D., Lee, E.A., Morgan, N., Necula, G., and Patterson, D.A.: The Parallel Computing Laboratory at U.C. Berkeley: A Research Agenda Based on the Berkeley View, Technical Report No. UCB/EECS-2008-23, March 21, 2008. http://www.eecs.berkeley.edu/Pubs/TechRpts/2008/EECS-2008-23.html
8) Asanovic, K., Bodik.,R., Catanzaro., B.C., Gebis, J.J., Husbands, P., Keutzer, K., Patterson, D.A., Plishker, W.L., Shalf, J., Williams, S.W., Yelick, K.A: The Landscape of Parallel Computing Research: A View from Berkeley, Technical Report No. UCB/EECS-2006-183, December 18, 2006. http://www.eecs.berkeley.edu/Pubs/TechRpts/2006/EECS-2006-183.html

表 3 ALPS/Looper の主要演算カーネルとその性能

カーネル名	スレッド数	GIPS (16 スレッド)	% to Peak	Memory Throughput (GB/sec)/chip	% to Peak	B/I	% to Total Time	備考
全体	16	8.15	6.89	14.2	16.8	1.75	100	-
OMP_48	16	3.38	2.86	24.1	28.4	7.13	21.6	探索処理
OMP_44	16	10.0	8.50	8.04	9.46	0.80	24.3	探索処理

表 4 ALPS/Looper の主要通信カーネルにおける通信

(a) mpi_isendirecv

Kind	Byte	Call	0-4K	4K-64K	64K-1024K	1024K-
AVG	13,981,050	384	128	0	0	256
MAX	13,981,050	384	128	0	0	256
MIN	13,981,050	384	128	0	0	256

(b) mpi_irecv / mpi_isend

Kind	Byte	Call	0-4K	4K-64K	64K-1024K	1024K-
AVG	6,990,525	192	64	0	0	128
MAX	6,990,525	192	64	0	0	128
MIN	6,990,525	192	64	0	0	128

表 5 RSDFT の主要演算カーネルとその性能

カーネル名	スレッド数	GFLOPS (4 スレッド)	% to Peak	Memory Throughput (GB/sec)/chip	% to Peak	B/F	% to Total Time	備考
全体	4	17.2	29.2	8.02	9.4	0.12	100	-
diag_2d_ dgemm_620	4	42.2	71.4	12.1	14.3	0.07	13.4	固有値ソルバー
gram_schmidt _sub_dgemm _343	4	41.5	70.2	10.8	12.7	0.07	10.0	直交化

表 6 RSDFT の主要通信カーネルにおける通信

(a) mpi_bcast (diag_2d_bcast_612)

Kind	Elapsed(s)	Wait(s)	Byte	Call	0-4K	4K-64K	64K-1024K	1024K-
AVG	129.7	64.1	5,172,432	512	0	0	0	512
MAX	130.3	65.2	5,185,832	512	0	0	0	512
MIN	128.8	62.7	5,165,732	512	0	0	0	512

(b) mpi_allreduce (precond_cg_allreduce_148)

Kind	Elapsed(s)	Wait(s)	Byte	Call	0-4K	4K-64K	64K-1024K	1024K-
AVG	78.4	59.9	16	307,968	307,968	0	0	0
MAX	88.0	69.8	16	307,968	307,968	0	0	0
MIN	71.9	52.7	16	307,968	307,968	0	0	0

(c) mpi_allreduce (precond_cg_allreduce_191)

Kind	Elapsed(s)	Wait(s)	Byte	Call	0-4K	4K-64K	64K-1024K	1024K-
AVG	77.5	59.2	16	307,968	307,968	0	0	0
MAX	87.8	70.1	16	307,968	307,968	0	0	0
MIN	69.5	50.5	16	307,968	307,968	0	0	0

表 7 NICAM の主要演算カーネルとその性能

計算種別	カーネル名	スレッド数	GFLOPS (4 スレッド)	% to Peak	Memory Throughput (GB/sec)/chip	% to Peak	B/F	% to Total Time	備考
―	全体	4	2.86	4.85	48.8	57.4	4.26	100	-
力学過程	mod_oprt_01	4	5.44	9.21	65.2	76.7	2.99	3.72	IF 文なし
	mod_oprt_03	4	2.20	3.73	39.3	46.2	4.46	3.56	IF 文あり
物理過程	mod_mp_nsw6	4	4.90	8.30	47.7	56.2	2.43	4.52	IF 文あり

表 8 NICAM の主要通信カーネルにおける通信

(a) mpi_irecv / mpi_isend

Kind	Byte	Call	0-4K	4K-64K	64K-1024K	1024K-
AVG	161,944	28,806	7,006	2,932	17,926	943
MAX	194,332	72,015	47,296	7,329	28,717	1,006
MIN	65,677	24,005	4,316	2,299	16,384	503

表 9 COCO の主要演算カーネルとその性能

カーネル名	スレッド数	GFLOPS (16 スレッド)	% to Peak	Memory Throughput (GB/sec)/chip	% to Peak	B/F	% to Total Time	備考
全体	16	3.66	1.55	13.0	15.3	3.55	100	-
flaxomp2	16	7.20	3.05	48.2	56.8	6.70	10.6	IF 文あり
flaxomp3	16	2.69	1.14	19.2	22.6	7.13	7.8	IF 文あり
flaxomp5	16	5.48	2.32	32.4	38.1	5.92	8.4	IF 文あり, ブロック化

表 10 COCO の主要通信カーネルにおける通信

(a) mpi_irecv / mpi_isend

Kind	Byte	Call	0-4K	4K-64K	64K-1024K	1024K-
AVG	59,666	201,020	72,527	101,259	27,234	0
MAX	61,064	224,292	94,162	102,896	27,234	0
MIN	53,548	196,366	68,200	100,932	27,234	0

(b) mpi_gatherv

Kind	Byte	Call	0-4K	4K-64K	64K-1024K	1024K-
AVG	10,283,820	130	65	0	28	37
MAX	251,953,500	130	65	0	29	65
MIN	5,141,908	130	65	0	0	36

(c) mpi_scatterv

Kind	Byte	Call	0-4K	4K-64K	64K-1024K	1024K-
AVG	3,282,987	270	135	0	109	26
MAX	80,433,170	270	135	0	111	135
MIN	1,641,493	270	135	0	0	24