TOHOKU
UNIVERSITY

ISSN 0286-7419

東北大学
サイバーサイエンスセンター

大規模科学計算システム広報

SENAC

Vol.44 No.1 2011—1

SENAC 2011—1

Supercomputing System
Cyberscience Center
Tohoku University
www.ss.isc.tohoku.ac.jp

大規模科学計算システム関連案内

＜大規模科学計算システム関連業務は、サイバーサイエンスセンター本館内の情報部情報基盤課が担当しています。＞

http://www.ss.isc.tohoku.ac.jp/

階	係・室名	電話番号(内線) e-mail	主なサービス内容	サービス時間 月～金
一階	共同利用支援係 （受　付）	022-795-3406(3406) FAX:022-795-6099 uketuke@isc.tohoku.ac.jp	各種申請書、講習会、利用相談、広報、センターへの質問や要望の受付等	8:30～17:15
一階	利用相談室	022-795-6153(6153) sodan05@isc.tohoku.ac.jp 相談員不在時 022-795-3406(3406)	計算機利用全般に関する相談 大判プリンタ、利用者端末等の利用	10:00～16:00 8:30～21:00
一階	利用者談話室	(3444)	各センター広報の閲覧	8:30～21:00
一階	展示室（分散コンピュータ博物館）		歴代の大型計算機等の展示	9:00～17:00
三階	庶務係	022-795-3407(3407) syomu@isc.tohoku.ac.jp	庶務に関すること	8:30～17:15
三階	会計係	022-795-3405(3405) kaikei@isc.tohoku.ac.jp	会計に関すること、負担金の請求に関すること	8:30～17:15
三階	共同研究支援係	022-795-6252(6252) rs-sec@isc.tohoku.ac.jp	共同研究、計算機システムに関すること	8:30～17:15
三階	共同利用支援係	022-795-6251(6251) uketuke@isc.tohoku.ac.jp	ライブラリ、アプリケーションに関すること	8:30～17:15
三階	ネットワーク係	022-795-6253(6253) net-sec@isc.tohoku.ac.jp	ネットワークに関すること	8:30～17:15
四階	研究開発部	022-795-6095(6095)		
五階	端末機室	(3445)	PC端末機(X端末)	8:30～21:00

(　) 内は東北大学内のみの内線電話番号です。青葉山・川内地区以外からは頭に 92 を加えます。

本誌の名前「SENAC」の由来

昭和33年に東北地区の最初の電子計算機として、東北大学電気通信研究所において完成されたパラメトロン式計算機の名前でSENAC-1(SENdai Automatic Computer-1)からとって命名された。

［巻頭言］

「持続可能な社会の構築」のためのスパコン

新年あけましておめでとうございます。サイバーサイエンスセンター長の小林先生から機会を頂きましたので、スパコンユーザーとしてご挨拶申し上げます。

私の専門は数値流体力学（CFD）で、特に航空機周りの流れをいかに精度良く再現するか、より効率良く飛ぶ空力形状を得るにはどうするか、という研究を進めて来ました。ちょうどタイミング良く三菱航空機(株)のジェット旅客機 MRJ の開発が始まり、その空力設計に研究成果が役立ったことをうれしく思っています。

過去、CFD はスパコン開発の牽引役の一つでした。80 年代にはベクトル型スパコンの性能が飛躍的に上がり、年配の著名な流体力学の先生が、「CFD が怒濤のごとく押し寄せてきた」、と述べられたのが印象的です。航空宇宙技術研究所（現 JAXA）のある CFD 研究者は、「いずれは風洞試験が要らなくなる」、と発言して実験に携わる人々を不安がらせたこともありました。

それから四半世紀、既に翼の空力設計の主役は風洞から CFD に替わっています。例えば 80 年代に開発されたボーイング 767 型機では 77 個の翼モデルが風洞で調べられましたが、最新の 787 型機開発では 5 個で済まされました。MRJ 開発では更に CFD が多用され、我が国で不足している大型風洞設備と開発経験をスパコンが補ってくれています。

これまでの我が国のスパコン開発には、国産旅客機の実現や 100 年先の地球環境予測、といった分かり易い目的が掲げられてきました。そのスパコンも最近では様々な分野で使われるようになっています。その分、スパコン開発の目的を一言で説明できなくなり、一昨年の事業仕分けでは、“なぜ世界一が必要？”と聞かれて返答に困ったのでしょう。高度・広範な目的に使われていることはスパコン利用の発展の証ではあるのですが、一般の人への説明も難しくなってきています。

誰にも理解の得られるスパコンの目的となると、なかなか良い文言が浮かんでこないのですが、いま流行の言葉を借りれば、「持続可能な社会の構築」のためのスパコン、が適切なのかと思います。スパコンは 100 年先の地球環境を見せてくれましたが、その未来に向けて私たちが安全・安心に暮らせる社会を実現しなくてはなりません。その構築のための道具としてスパコンは不可欠です。

とはいうものの、「持続可能な社会の構築」も漠とした言葉です。やはり航空機に話を戻した方が良いでしょう。

航空分野でも持続可能な社会構築のための議論が盛んになっています。全世界の CO_2 排出の中で航空機によるものは現在は 2%程ですが、今の勢いで航空交通が増えると楽観視できません。飛行機の利用者が増えると安全面も更に高める必要があります。しかし、飛行機を専門としている私でも晴天乱流で揺れたりすると少し不安になります。

飛行機は空気のお陰で揚力を得ているのですが、一方では複雑な動きをする空気と戦っています。飛行機の燃費に直結する空気抵抗も翼表面の乱流現象が増大させています。空港騒音もジェ

ット噴流や風切り音の乱流現象が主因です。その乱流をスパコンで再現し制御することで燃費の良い静かな飛行機を実現しようとしていますが、現象が微細で非定常であるためにスパコン性能がまだまだ足りません。

10年ほど前にNASAの部長級の人が、「航空ルネッサンス」という言葉を講演の中で使いました。“過去50年、旅客機の性能は大幅に改善されたものの外形は変わってない。いま要求されている環境に優しい且つ落ちない飛行機を実現するとなると革新的な形の飛行機、つまり航空ルネッサンスが必要だ”、といったような話でした。

既に米国では海を泳ぐエイのような形の未来型旅客機の研究も進んでいますが、そのような開発には過去のデータが役立たないため、飛行試験も事前にスパコン内でやろうとの研究も始まっています。巡航飛行だけでなく、旋回や離着陸などの総ての飛行時の流れをスパコン内で再現し、舵の効きも調べて危険な飛行試験を減らそうとの目的です。その実現には先に書いた手強い乱流の再現も必要ですが、スパコン性能の改善が今の勢いで続けばいずれは可能となるでしょう。

80年代、航空宇宙技術研究所にスパコンが導入された際にある部長から、“研究所の全予算の一割以上を10人足らずの研究者で使うのもおかしい”、と言われたことがあります。CFDが風洞試験にはまだまだ敵わなかった当時として、その価値を自信を持って説明はできませんでした。パソコンの性能が向上して研究のツールにもなっている現在、大学でスパコンを持つことについても同じ意見が出るかも知れません。

日本は「ものづくり」の国だとのことで、テレビでは職人芸的なトピックがよく取り上げられます。しかし、ものづくりの本当に大事なことはヤスリ掛けの技術よりも、最先端技術を常に生み出しつづける“智”にあります。大学は新たな“智”を生み出すとともに、それを担う人を育てるところです。その大学に“智”を生み出すためのツールとしてのスパコンは不可欠であり、未来への投資です。

幸い、東北大学のサイバーサイエンスセンターが提供して下さっている計算機環境は世界的にもトップレベルのものです。その恵まれた環境のユーザーとして、スパコンでなければ不可能な問題に常にチャレンジして行かなくてはなりません。それが私にとっては、「航空ルネッサンス」を目指すもので、「持続可能な社会の構築」のためのものづくりに繋がると信じています。この場を借りてセンターにあえてお願いするとすれば、チャレンジングな大規模計算には計算機使用料も多額になるため、未来指向型研究に対するサポート制度を今以上に拡充していただければと思います。

東北大学大学院工学研究科
教授　中橋和博

[大規模科学計算システム]

並列コンピュータ Express5800/A1080a-D の並列処理

吉田　正浩
日本電気株式会社　HPC 事業部

１．はじめに

HPC の分野では、複数の CPU を使って並列実行し高速化を図ってきました。その中でも、SX-9 のように超高速な CPU（もしくはコア）により、逐次実行での性能を確保したうえで、さらに並列計算も行いより一層の性能向上を図るというアプローチと、多くのコアで並列計算を行うことで処理速度を向上させるというアプローチがあります。サイバーサイエンスセンターで運用している並列コンピュータ Express5800/A1080a-D は、１つの計算ノードに 8 個のコアを搭載した CPU を 4 台搭載することで 32 コアが利用可能となっており、後者のアプローチに適した計算機だと言えます。

Express5800/A1080a-D は共有メモリ型の計算機であり、そのアーキテクチャは図 1 のように表されます。メモリはメモリノードという各 CPU に直結されたブロックに分かれており、共有メモリ型計算機の中でも特に NUMA(Non-Uniform Memory Access)と呼ばれるアーキテクチャとなっています[1]。

この資料では並列計算の入門として、Express5800/A1080a-D で利用可能な並列計算の手法のうち、自動並列化と OpenMP[2]による並列化をご紹介します。

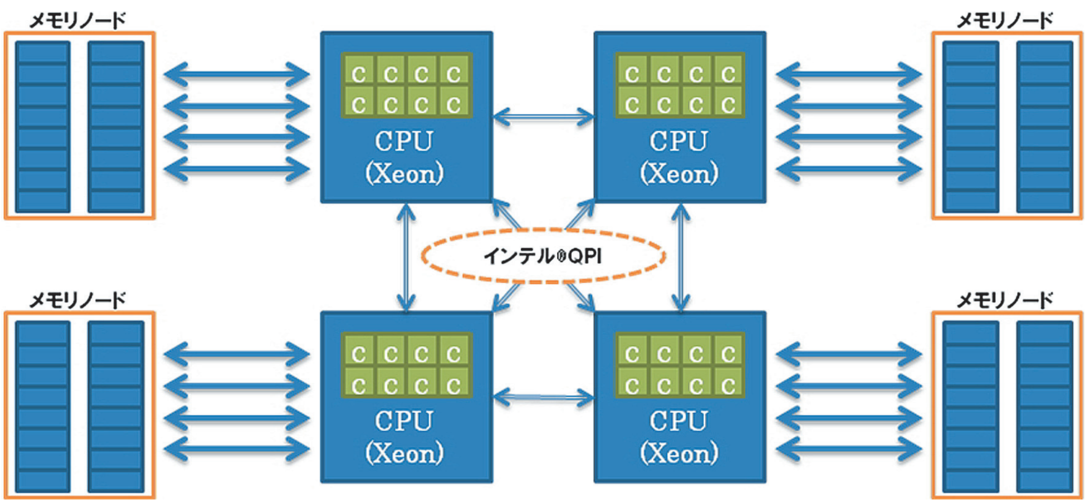


図 1 Express5800/A1080a-D のアーキテクチャ

２．並列処理の概要

一つのジョブを複数の CPU やコアで分散して処理するために、様々な手法・規格が定められています。これらを分類すると、共有メモリ型の計算機での並列実行を念頭においた「スレッドレベルでの並列化」と分散メモリ型の計算機で並列実行を行うための「プロセ

スレベルの並列化」の二種類に分けられます。

スレッドレベルの並列化では、並列に実行が可能な do ループ等の処理を、並列に実行可能な単位である「スレッド」に分割し、複数のスレッドを各コアに割り当てる事で並列実行を実現しています。プロセスレベルの並列化では、複数の「プロセス」を起動し、各プロセスが割り当てられた処理を並列に実行します。並列実行の制御とデータの転送のために、プロセス間の通信を行う必要があります。

図 2 に逐次実行（非並列実行）、スレッドレベルの並列実行、プロセスレベルの並列実行の概念図を示します。スレッドレベル、プロセスレベルのどちらの並列実行においても、複数のコアを使う点は同じですが、スレッドレベルが単一計算ノード内の複数のコアを使用する点に限られるのに対し、プロセスレベル並列化ではノード間にまたがった並列化が可能です。

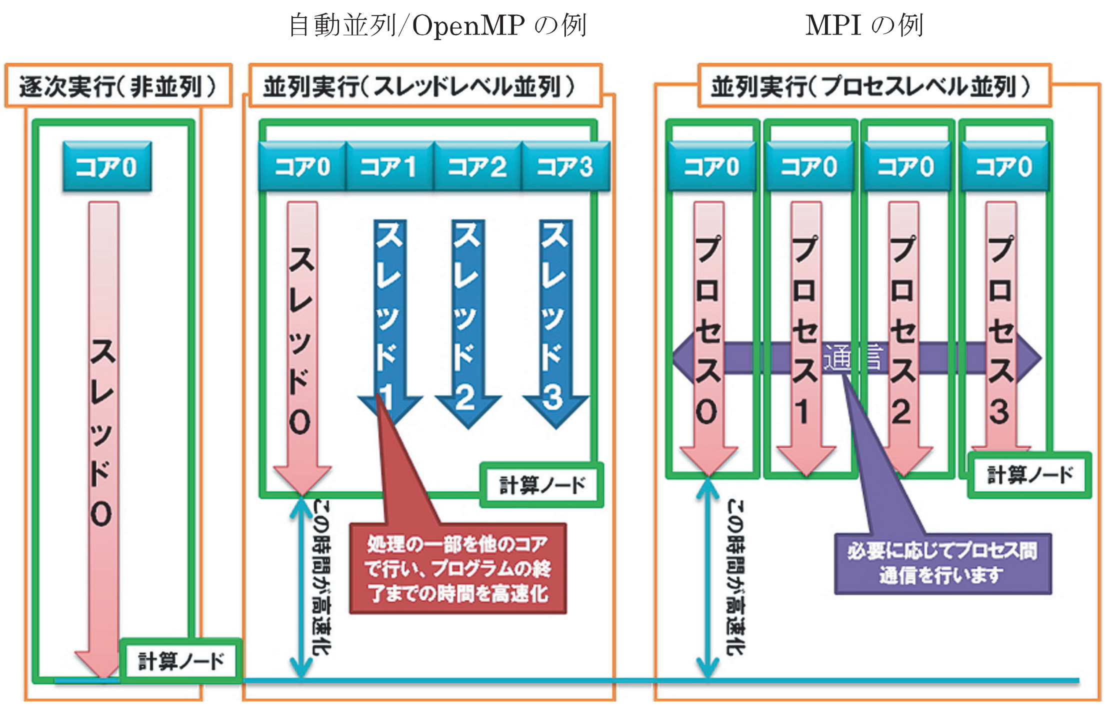


図 2 並列実行モデル

現在、広く利用されている並列化の手法としては、自動並列、OpenMP、MPI などがあげられ、自動並列化と OpenMP はスレッドレベルの並列化を意図した手法であり､MPI のみがプロセスレベルの並列化が可能な手法です。

自動並列化は、コンパイラが並列化可能な処理を認識し、自動的にスレッドに分割可能とする手法です。コンパイラオプションを指定するだけで、複数のコアを使った並列計算が可能となります。ただし、プログラムの構造が複雑な場合には、期待通りに並列化が行われず、性能向上がみられない場合もあります。一方、OpenMP はプログラマが並列実行する部分をコンパイラ指示行で指定することで、スレッドレベルの並列化を行います。

OpenMP は指示行による並列化を実現する API（Application Program Interface）で、プログラム中に指定する指示行の意味や、並列化をサポートする専用の関数などが定義されています。

以降では Fortran のプログラムリストを基に解説を行います。一部 C 言語のプログラムリストも併記しています。

３．自動並列化

自動並列化は、コンパイラが“安全”に並列化可能だと判断したループや処理のうち、高速化が期待できる部分を自動的にスレッドに分割する並列化方法です。“安全”というのは、並列に実行しても浮動小数点演算による丸め誤差を除いて、計算結果に影響を与えないという意味となります。リスト 1のループは、異なる i の値に対しての処理が独立しているために、i の値毎にスレッドに分割しても安全だと判断できます。一方、リスト 2では、i の処理に i-1 の値を使っており、異なるループの処理内容が他のループの処理結果に影響を与える「依存関係」が存在しています。この場合、コンパイラは安全では無いと判断し、自動並列化による並列化を行いません。

```
do i=1,1000
   a(i) = b(i) * c(i)
enddo
```

リスト 1

```
do i=2,1000
   a(i) = b(i) * a(i-1)
enddo
```

リスト 2

センターで提供している、Intel Fortran,C/C++コンパイラでは、“-parallel”をコンパイラオプションに指定することで自動並列化が利用可能です。コンパイル方法は図 3の通りです。この際に、“-par-report”オプションも同時に指定することで、プログラム中のどのループが並列化されるかを表示できます。さらに詳細情報を出力させる“-par-report=3”を指定すると、並列化されていないループが、何故並列化されなかったかも出力されます。図 3のコンパイル例では、プログラム中 12 行目のループは並列化されているが、15 行目のループは依存関係があるために並列化されていないというメッセージが出力されています。

並列処理を実行するには、通常プログラムの実行時に環境変数 OMP_NUM_THREADS で並列数を指定しますが、センターで提供しているジョブ管理機能では、ジョブクラスによって自動的に並列数が設定され、ジョブクラス a8, a16, a32 ではそれぞれ 8, 16, 32 の並列数で実行可能です。図 4の並列数の指定例では 4 並列で実行されます。詳細は並列コンピュータ Express5800 の利用法[4]を参照ください。

ループの自動並列化を阻害する主な要因として次のような場合があります。このような場合には、プログラムを書き換えていただくか、指示行を挿入することでコンパイラに並列化可能であることを示すことで、自動並列化が可能となる場合もあります。

・ループ中に依存関係が存在する場合
・ループの繰り返し回数が少なく、自動並列化の性能向上が見込まれない場合
・ループの繰り返し回数がループ開始時に予め分からない場合（while ループなど）
・ループ内部へのジャンプや、ループ外部へのジャンプがある場合（goto 文など）
・サブルーチンや関数の呼び出しがある場合（write 文、read 文などの IO も含む）
・依存関係を判断できない間接参照（またはポインタの参照）が存在する場合

```
[Fortran]
f95 –parallel –par–report=3 program.f90
program.f90(12): (col. 3) remark: LOOP WAS AUTO-PARALLELIZED.
program.f90(15): (col. 3) remark: loop was not parallelized:
                                  existence of parallel dependence.
```
コンパイル時のメッセージ

```
[C 言語/C++]　コンパイル方法
cc –parallel –par–report=3 program.c
```

図 3 コンパイル方法とコンパイルの様子

```
[csh の場合の例]
setenv OMP_NUM_THREADS 4
./a.out
[bash の場合の例]
export OMP_NUM_THREADS=4
./a.out
```

図 4 並列数の指定と実行

リスト 3 に示すプログラムで、自動並列化の動作を検証します。このプログラムを自動並列化オプション付きでコンパイルし、4 つのスレッドで実行すると、図 5 のようなイメージで処理が分割され実行します。スレッドレベル並列化の際に、それぞれのスレッドは固有のスレッド ID を持ちます。自動並列化の場合は特に意識する必要はありませんが、図 5 では、以降の説明のためにスレッド ID の例も記載しています。

並列処理の様子を具体的に考察します。初めに、スレッド 0（マスタースレッドとも呼ばれます）が単独でプログラムの実行を開始します。自動並列化によって並列化されたループに到達すると、スレッド 0 によって、3 つのスレッドが生成されます。スレッド 0 を含めた各スレッドには、ループのインデックス i の担当範囲が定められており、それぞれが割り

当てられた i の担当範囲の処理を開始します。今、スレッド 0 が a(1) = b(1) * c(1)という処理を始めるとすると、ほぼ同時にスレッド 3 が a(751) = b(751) * c(751)の処理を行っていると考えられます。スレッド 0 は a(1)に関する処理を完了すると a(2)=b(2)*c(2)を、スレッド 3 は a(751)の処理が完了すると次に a(752)=b(752)*c(752)を…という順序でそれぞれのスレッドが独立して、処理を進めます。各スレッドは、自身に割り当てられた処理が完了すると、他のスレッドの処理完了を待ち、全てのスレッドがそろった時点で同期をとり並列処理を終了します。

```
subroutine auto_parallel(a, b, c)
  integer :: i
  real(8),dimension(1000) :: a, b, c
  do i=1,1000
     b(i) = i
     c(i) = 2*i
  enddo
!乗算
  do i=1,1000
    a(i) = b(i) * c(i)
  enddo
end subroutine auto_parallel
```

Fortran

```
void loop_parallel(double *a, double
*b, double *c){
  int i;
  for(i=0;i<1000;i++){
     b[i] = i;
     c[i]= 2*i;
  }
//乗算
  for(i=0;i<1000;i++){
    a[i] = b[i] * c[i];
  }
return;
```

C

リスト 3

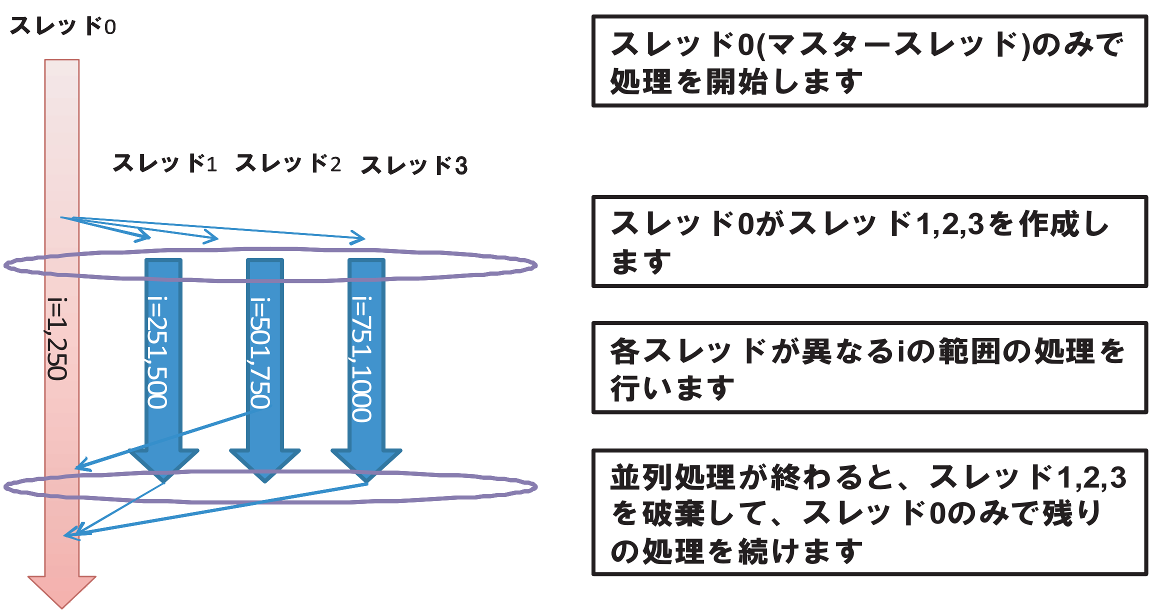


図 5 自動並列化による並列実行イメージ

4．OpenMP

OpenMP は、プログラマが指示行を挿入することで並列化を行う手法です。プログラマの責任で並列処理を指示するため、自動並列化ではコンパイラが“安全”に並列化できると判定できないループや処理についても並列化が可能です。逆に、**アルゴリズム上並列化できないような処理に関しても、誤って OpenMP の指示行を追加すると強制的に並列化されてしまい、結果不正を引き起こすことがあるので注意が必要です。**

4.1 OpenMP による並列化の例

初めに、リスト 4 に示す簡単な OpenMP のプログラムの実行を通して、OpenMP 使用の流れを解説します。このプログラムは、並列実行時に自らのスレッド ID を出力するという動作を行います。

Fortran

```fortran
program write_myid
implicit none
integer::myid
integer::omp_get_thread_num
myid=0

!$omp parallel private(myid)
   myid = omp_get_thread_num()
   write(*,*) "My thread id is ",myid
!$omp end parallel
end program write_myid
```

C

```c
#include <stdio.h>
#include<omp.h>
int main(){

int myid=0;
#pragma omp parallel private(myid)
{
   myid = omp_get_thread_num();
   printf(“My thread id is %d¥n”, myid);
}
}
```

リスト 4

OpenMP の指示行は、Fortran の固定形式では「Comp」、Fortran の自由形式では「!$omp」、C 言語では「#pragma omp」から始まる行で指定されます。このプログラムは、指示行により!$omp parallel から !$omp end parallel で囲まれた部分を並列実行可能と指定しています。この部分を並列領域と呼び並列領域に含まれる処理が、実行時に指定されたスレッド数で並列に実行されます。このプログラムは並列領域で、omp_get_thread_num 関数を呼び出しています。この関数は OpenMP の規格で定義された補助関数であり、スレッドにユニークに割り当てられた番号、すなわち、スレッド ID を返します。各スレッドが取得したスレッド ID を次の行で出力し、並列領域を終了しています。

このプログラムは図 6 のようにコンパイラオプションに“-openmp”を指定してコンパイルします。OpenMP によって、並列化されたプログラムも、自動並列化と同様に OMP_NUM_THREADS 環境変数で並列数を指定します。実行は図 7 の通り、通常のプロ

グラムと同様に./a.out という形で行います。このプログラムの実行イメージは、図 8 のようになっています。

```
[Fortran]
f95 –openmp   program .f90
[C 言語/C++]
cc   –openmp program.c
```

図 6 コンパイル方法(OpenMP)

```
$ f95 -openmp write_myid.f90
$ setenv OMP_NUM_THREADS 4
$ ./a.out
 My thread id is            0
 My thread id is            3
 My thread id is            1
 My thread id is            2
```

Fortran

```
$ cc -openmp write_myid.c
$ setenv OMP_NUM_THREADS 4
$ ./a.out
 My thread id is            0
 My thread id is            2
 My thread id is            1
 My thread id is            3
```

C

図 7 実行方法と実行結果

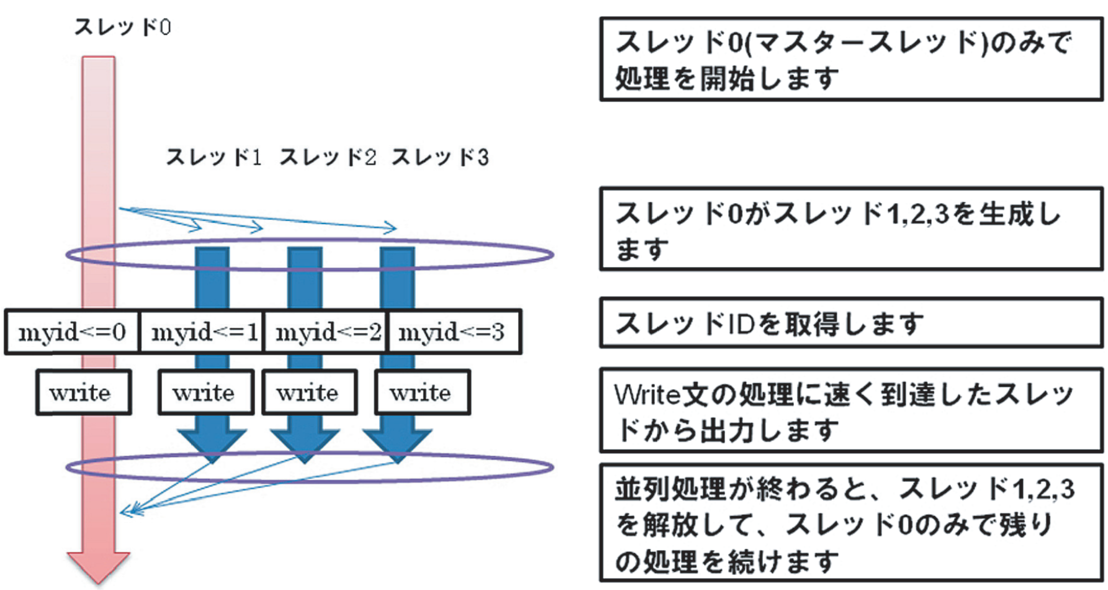


図 8 プログラムの実行イメージ

結果を確認します。図 7 において、4 行の出力が行われていることより、4 並列で動作している事が確認できます。また、ID が 0,1,2,3 の順ではなく、0,3,1,2 と出力されており、各スレッドの処理される順序がスレッド ID 順ではない事が確認できます。これは、(何らかの同期を行わない限り) スレッドの実行順序は保証されず、先に write に辿りついたスレッドから ID を出力するためであり、このプログラムでは、実行する度に出力される順序が変わります。OpenMP で並列化した場合は、各スレッドに割り当てられた処理を独立して実行するため、実際に処理が行われる順序についての保証がなく、結果が非並列実行と異なる場合があります。OpenMP で並列化を行う際はこの点に注意が必要です。(スレッド内の処理は順に実行完了します。)

4.2 ループの並列化

ここでは、ループの並列化方法を述べます。ループの並列処理を行う場合は!$omp parallel do 指示行を用います。この指示行を挿入すると、直後の do ループが OMP_NUM_THREADS 等で指定されたスレッド数に応じて分割されます。

```
subroutine loop_parallel(a, b, c)
  integer :: i
  real,dimension(1000) :: a, b, c
!$omp parallel do
  do i=1,1000
    a(i) = b(i) * c(i)
  enddo

end subroutine loop_parallel
```

Fortran

```
void loop_parallel(double *a, double
*b, double *c, int N){
  int i;
#pragma omp parallel for
for(i=0;i<N;i++){
    a[i] = b[i] * c[i];
  }
return;
}
```

C

リスト 5

リスト 5 のプログラムをコンパイルし実行すると、図 5 で示した自動並列化の例と同様に動作します。!$omp parallel do 指示行は、スレッドの生成、解放とループの分割まで行うため、前節の!$omp parallel と!$omp end parallel の指定は不要です。また、parallel do 指示行は直後の do ループに対して有効であり、リスト 6 の(a)と(b)では、分割されるループが異なります。(a)では j のループが、(b)では i のループが分割されます。(b)のループでは、ループの分割処理を j のループ回数行う必要があり、一般に(a)よりもオーバーヘッドが大きくなります。

```
!$omp parallel do
  do j=1,1000
    do i=1,1000
      a(i,j) = b(j) * c(i)
    enddo
  enddo
```

(a)

```
  do j=1,1000
!$omp parallel do
    do i=1,1000
      a(i,j) = b(j) * c(i)
    enddo
  enddo
```

(b)

リスト 6

4.3 private 変数と shared 変数

OpenMP では、変数の属性を指定できます。属性の指定は!$omp parallel や!$omp parallel do に続けて、“属性名(変数名)”という形で指定します。リスト 4 の例では!$omp parallel do private(myid)と myid を private 変数に指定していました。

よく使用される属性は private と shared の二つの属性です。private 属性を持つ変数（これを private 変数と呼びます）は、スレッドごとに独立した値を保持できる変数で、一時的な変数などに用いられます。private 変数の初期値、終了値は不定となります。よって、並列処理部内で値の代入を行う必要があります。

また、shared 属性を持つ変数(shared 変数と呼びます)は、すべてのスレッドからアクセス可能な変数です。すべてのスレッドが参照・更新できてしまうため、更新は他のスレッドと排他的に行う必要があります。

OpenMP の並列領域内の変数はデフォルトで shared 変数になります。ただし、!$omp do で並列化したループの制御変数と、Fortran の do ループの制御変数は自動的に private 変数となります。注意が必要なのは、C 言語で 2 重ループの外側ループで並列化する場合で、内側ループの制御変数を明示的に private 変数と指定しなければ、制御変数が shared 変数とみなされ、期待した実行結果が得られなくなります。

リスト 7 に、tmp という変数を private 変数として使用する例を示します。このプログラムでは、変数 tmp はスレッドごとに保持されます。スレッド 0 の tmp とスレッド 1 の tmp は全く別の変数として扱われ、独立した値を保持します。他のスレッドのプライベート変数に直接アクセスすることはできません。もしも、tmp が shared 変数として扱われた場合、複数のスレッドが同時に更新しようとするため、更新後の値が最後に書き込んだスレッドの値となり、次の参照時に意図しない値を読み込むことになります。

```fortran
!$omp parallel do private(tmp)
do i=1,n
   tmp = 0.0d0
   do j=1,m
      tmp = tmp + a(j,i)
   enddo
   b(i) = tmp
enddo
```

Fortran

```c
#pragma omp parallel for private(tmp,j)
for(i=0;i<n;i++){
   tmp = 0.0;
   for(j=0;j<n;j++){
      tmp = tmp + a[i][j];
   }
   b[i] = tmp;
}
```

C

リスト 7

5. ファーストタッチ

OpenMP の紹介の最後に、NUMA アーキテクチャの計算機向けに考慮が必要となるファーストタッチと呼ばれる手法をご紹介します。先に述べたとおり、Express5800/A1080a-D は NUMA アーキテクチャの共有メモリ計算機です。NUMA アーキテクチャでは、ある CPU から見て直結されたメモリノードをローカルメモリ、他の CPU 配下のメモリノードをリモートメモリと呼びます。リモートメモリへのアクセスを行うと、ローカルメモリへのアクセスに比べて、メモリレイテンシ（メモリアクセス時の遅延時間）が伸び、さらに、CPU 間を接続している QPI(QuickPath Interconnect)がボトルネックになる恐れがあります。また、図 9 のように、特定のメモリノード間にアクセスが集中すると、アクセスが集中したメモリノードと CPU 間のデータ転送がボトルネックとなり、並列性能が低下する原因となります。

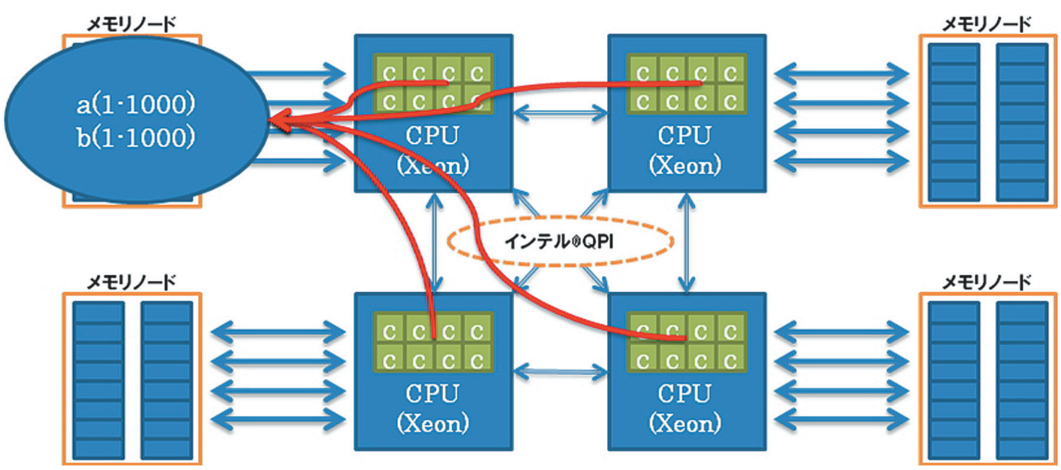


図 9 メモリアクセスの集中

そこで、プログラム中で使用する配列がどのメモリノードに配置されるかを制御する必

要が生じます。Express5800/A1080a-D をはじめ、ほとんどの NUMA マシンでは、変数・配列を初期化した（初めに値を書き込んだ）コアのローカルメモリに優先して配置するよう制御されます。この仕様を利用して、配列の初期化部分を並列化する事で、図 10 のように配列の配置を各メモリノードに分散可能です。この手法により、リモートメモリへのアクセスを防ぐと同時に、一つのメモリノードへのアクセス集中を回避することができます。配列の初期化を並列に行い、メモリアクセスをローカルメモリ中心となるように制御する手法をファーストタッチと呼びます。

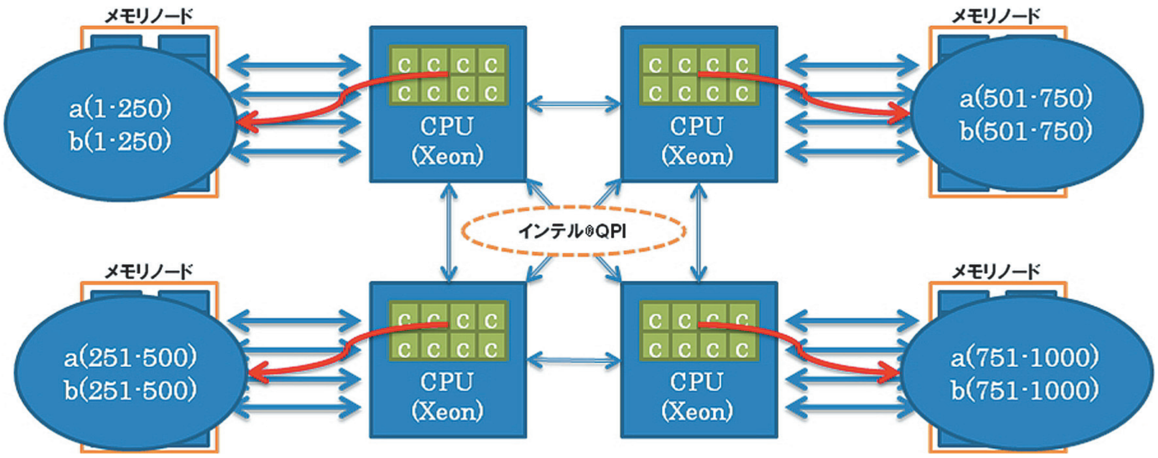


図 10 ファーストタッチ後のメモリアクセス

ファーストタッチの例として、次のリスト 8 のプログラムで考察します(なお、このプログラムは stream ベンチマーク[3]を模しています)。このプログラムでは、サブルーチン Triad の j のループで並列化でき、処理時間のほとんどは Triad によって占められます。それゆえ、メモリの配置を考えなければ、Triad の他の部分を並列化する必要はありません。しかし、ファーストタッチを行わず、Triad のみを並列化した場合では、表 1 のように 32 並列で 2.9 倍の性能向上にとどまっており、効率的な並列化が出来ているとは言えません。

表 1 ファーストタッチの効果

	ファーストタッチ無し		ファーストタッチ有り
並列数	1	32	32
サブルーチン部分(秒)	114.2	39.3	6.8
プログラム全体(秒)	114.8	40.0	7.1
サブルーチン部分性能向上率 (1 並列を 1.0 とした場合)	1.0	2.9	16.8

```fortran
program stream
implicit none
integer:: N,NTIMES,j,k
parameter (N=20000000,NTIMES=1000)
real(8)::a(N),b(N),c(N),scalar
real(8)::omp_get_wtime,time,time1,time2
   do j=1,N
     a(j)=1.0
     b(j)=2.0
     c(j)=3.0
   enddo
   scalar=1.0d0
   time=0.0d0
   do k=1,NTIMES
     time1 = omp_get_wtime()
     call Triad(scalar,a,b,c,N)
     time2 = omp_get_wtime()
     time = time + (time2 - time1)
   enddo
   do j=1,N
     scalar = scalar + a(j)
   enddo
   write(*,*) scalar,time
end program stream

subroutine Triad(scalar,a,b,c,N)
integer:: N,j
real(8)::a(N),b(N),c(N),scalar
!$omp parallel do
   do j=1,N
      a(j) = b(j)   + scalar*c(j)
   enddo
end subroutine Triad
```

Fortran

```c
#include <stdio.h>
#include <omp.h>
#define N 20000000
#define NTIMES 1000
void Triad(double scalar);
static double a[N],b[N],c[N],scalar;
int main(){
        int k,j;
        double time,time1,time2;
        for (j=0; j<N; j++) {
                a[j] = 1.0;
                b[j] = 2.0;
                c[j] = 3.0;
                }
        scalar=1.0;
        time=0.0;
        for (k=0; k<NTIMES; k++) {
                time1=omp_get_wtime();
                Triad(scalar);
                time2=omp_get_wtime();
                time=time+(time2-time1);
                }
      for (j=0; j<N; j++) {
                      scalar=scalar + a[j];
                }
      printf("%f %f¥n",scalar,time);
}
void Triad(double scalar){
                int j;
#pragma omp parallel for
                for (j=0; j<N; j++)
                        a[j] = b[j]+scalar*c[j];
}
```

C

リスト 8

そこで、リスト 9 では初期化部分も並列化し、ファーストタッチを行います。配列 a,b,c ともに、初めてのアクセスは初期化部で行われ、各スレッドが Triad 内で処理を担当するインデックス"j"の範囲の値を初期化しています。これにより a,b,c の各要素は図 10 のように、初期化を担当したスレッドが割りつけられたコアのローカルメモリに配置され、アクセスの集中とリモートメモリへの頻繁なアクセスを回避できます。ファーストタッチの性能は表 1 に示した通りであり、32 並列での性能向上が確認できています。

```fortran
(宣言部省略)
!$omp parallel do
   do j=1,N
      a(j)=1.0
      b(j)=2.0
      c(j)=3.0
   enddo
(中略)
subroutine Triad(scalar,a,b,c,N)
(中略)
!$omp parallel do
   do j=1,N
      a(j) = b(j)   + scalar*c(j)
   enddo
end subroutine Triad
```

Fortran

```c
(宣言部省略)
#pragma omp parallel for
      for (j=0; j<N; j++) {
            a[j] = 1.0;
            b[j] = 2.0;
            c[j] = 3.0;
 (中略)
void Triad(double scalar){
 (中略)
#pragma omp parallel for
            for (j=0; j<N; j++)
                  a[j] = b[j]+scalar*c[j];
}
```

C

リスト 9

ファーストタッチを行う際に、演算部分の並列ループと初期化時の並列ループの“メモリアクセスパターン”を可能な限り近づける必要があります。リスト 10 では、配列 a(i,j) の初期化を j のループで行っていますが、演算部分では内側の i のループで並列化し a(i,j) にアクセスします。このとき、a(i,j)は j で分割されメモリに配置されているため、演算部では図 11 のようにリモートメモリへのアクセスが頻繁に行われます。このようなケースでは、初期化をリスト 11 のように演算ループと同じく i のループで並列化することで、演算部分でのリモートメモリへのアクセスを大幅に削減が可能となります。

```
program memorytest
implicit none
integer:: N,i,j,k
parameter (N=2000)
real(8)::a(N,N),b(N),c(N,N),scalar
!$omp parallel do
  do j=1,N
   do i=1,N
    a(i,j)=2.0d0
    c(i,j)=1.0d0
   enddo
  enddo
!$omp parallel do
  do j=1,N
    b(j)=1.0d0
  enddo
  call mem_test(a,b,c,N)
  scalar = 0.0d0
  do j=1,N
     scalar = scalar + c(N,j)
  enddo
  write(*,*) scalar
end program memorytest

subroutine mem_test (a,b,c,N)
integer:: N,i,j
real(8)::a(N,N),b(N),c(N,N)
  do j=2,N
    b(j) = c(1,j-1)
!$omp parallel do
    do i=1,N
      c(i,j) = c(i,j) + a(i,j) / b(i)
    enddo
  enddo
end subroutine mem_test
```

リスト 10

```
program memorytest
implicit none
integer:: N,i,j,k
parameter (N=2000)
real(8)::a(N,N),b(N),c(N,N),scalar
  do j=1,N
!$omp parallel do
   do i=1,N
    a(i,j)=2.0d0
    c(i,j)=1.0d0
   enddo
  enddo
!$omp parallel do
  do j=1,N
    b(j)=1.0d0
  enddo
  call mem_test(a,b,c,N)
  scalar = 0.0d0
  do j=1,N
     scalar = scalar + c(N,j)
  enddo
  write(*,*) scalar
end program memorytest

subroutine mem_test (a,b,c,N)
integer:: N,i,j
real(8)::a(N,N),b(N),c(N,N)
  do j=2,N
    b(j) = c(1,j-1)
!$omp parallel do
    do i=1,N
      c(i,j) = c(i,j) + a(i,j) / b(i)
    enddo
  enddo
end subroutine mem_test
```

リスト 11

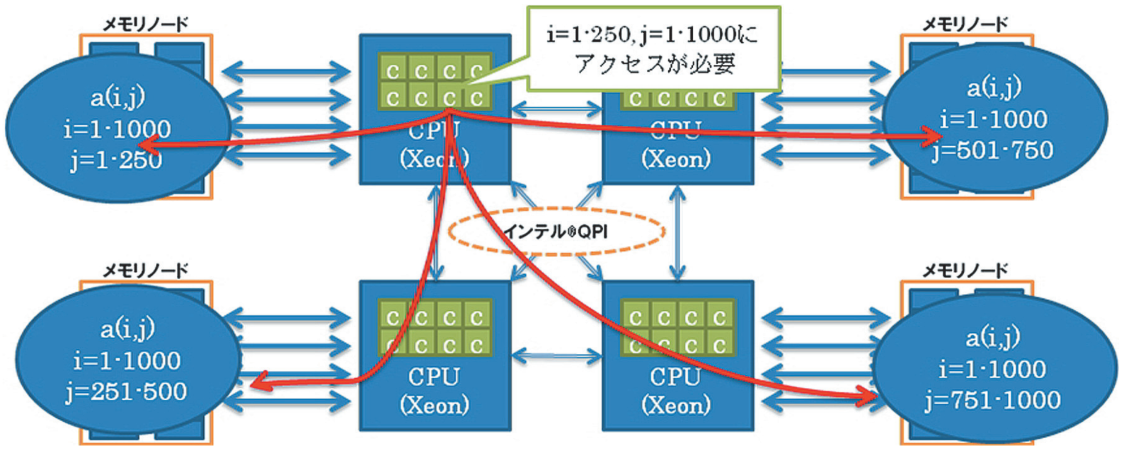


図 11 初期化と異なるループで並列化を行った場合のメモリアクセスパターン

6. おわりに

並列計算の入門として、自動並列化や OpenMP をご紹介いたしました。自動並列化、OpenMP ともに、紹介しきれなかった機能や、技法が存在します。より高度な機能については Intel コンパイラのマニュアルを参照ください。

並列計算はハードルが高いと敬遠される方もいらっしゃいますが、うまく利用できると数十倍の高速化を達成できる場合もあります。本稿が、並列化を行うきっかけや、並列計算を行う手助けになれば幸いです。

参考文献

[1] 那須 康之, 鈴木 健一, 谷岡 隆浩 “Express5800/A1080a-D のハードウェア”, SENAC Vol.43, No. 3(2010.7)

[2]OpenMP Architecture Review Board “OpenMP Application Program Interface Version 3.0 May 2008”, http://www.openmp.org/mp-documents/spec30.pdf

[3] John D. McCalpin, “STREAM:Sustainable Memory Bandwidth in High Performance Computers”, http://www.cs.virginia.edu/stream/

[4] 情報部情報基盤課共同研究支援係, 共同利用支援係, サイバーサイエンスセンター スーパーコンピューティング研究部 “並列コンピュータ Express5800 の利用法”, SENAC Vol.43, No. 2(2010.4) ,http://www.ss.isc.tohoku.ac.jp/service/gen/Express-Ver3.pdf

白

［解　説］

東北大学におけるキャンパス無線 LAN サービスについて

東北大学サイバーサイエンスセンター
後藤英昭, 水木敬明, 曽根秀昭

東北大学情報部 情報基盤課
七尾晶士, 澤田勝己, 北澤秀倫, 森 倫子

1.　はじめに

東北大学総合情報ネットワークシステム (TAINS) では、これまで、キャンパス無線 LAN サービスとして下記の二種類のローミング(相互利用)システムを提案して、各部局における無線 LAN システムの整備と相互利用をサポートしてきました[1,2]。アクセスポイントの設置は基本的には部局の手に委ねられていました。

- 「どこでも TAINS」
  学内ネットワークのみで利用できるローミングシステム.
- 「eduroam (エデュローム)」
  国内および海外の参加機関で利用できる無線 LAN ローミング基盤[3].

しかしながら、部局による整備では、特に会議室やホール等におけるアクセスポイントの設置が難しく、無線 LAN が本来必要とされるような共用エリアで利用できないという問題がありました。

平成 21 年度補正予算が認められたことを受けて、サイバーサイエンスセンターではユビキタスネットワークシステムの導入・整備を行い、「TAINS 無線 LAN システム」のサービスを平成 22 年 12 月より開始しました。本稿では、TAINS 無線 LAN システムの概要と、部局整備の無線 LAN システムとの関係、および、東北大学におけるキャンパス無線 LAN サービスの現状について解説します。2 章は利用者および管理者向けの内容で、3 章は主に部局管理者向けの内容となっています。

2.　TAINS 無線 LAN システム

2.1　概要

TAINS 無線 LAN システムは、全学で約 100 基の Wi-Fi 対応無線 LAN アクセスポイントと、それらを収容するネットワーク機器で構成されており、学内外の教職員・研究員・学生にキャンパス無線 LAN サービスを提供します。また、3 章で後述するように、部局所有のアクセスポイントを収容するためのネットワークも提供します。

当システムはアクセスポイントおよびアクセスネットワークの機能のみを提供するもので、無線 LAN の利用にあたっては、次節で説明するように利用者ごとのアカウントが必要になります。アクセスポイントは、全学の委員会等が開かれる頻度の高い会議室等の共用エリアに設置されています。平成 23 年 1 月現在のアクセスポイント設置場所を表 1 に示します。TAINS 無線 LAN システムの詳細と最新情報については、TAINS のウェブサイト[4]をご覧ください。

TAINS 無線 LAN システムでは、「どこでも TAINS」と「eduroam」の二種類が利用できます。「どこでも TAINS」は VPN(仮想プライベートネットワーク)を利用した無線 LAN ローミングシステムで、偽アクセスポイントや盗聴に対して高い安全性を有し、接続も比較的容易で、本学で長年に渡って利用実績のある方式です[2]。国際無線 LAN ローミング基盤 eduroam は、IEEE802.1X による安全なユーザ認証が可能で、国内外の参加機関でも無線 LAN 利用が可能な、グローバル・スタンダードな方式です[1]。特に eduroam は商用サービスとの連携も進められており、平成 23 年 1 月現在、ライブドア社が関東地区で展開している公衆無線 LAN サービスのアクセスポイントも大学発行のアカウントで利用できます[3]。

表 1. アクセスポイント設置場所

キャンパス	設置場所
片平	金属材料研究所 2 号館 1 階 講堂・会議室・ロビー
	電気通信研究所 ナノスピン実験施設 4 階 カンファレンスルーム 同上 1 号館 1 階 談話室 同上 2 号館 4 階 中会議室・大会議室
	エクステンション教育研究棟 1 階 部局長会議室
	さくらホール 1 階 ホール 同上 2 階 会議室
	片平本館 2 階 情報推進課会議室
川内	附属図書館 本館 1 号館 同上 2 号館
	マルチメディア教育研究棟 2 階 教室 同上 3 階 教室 同上 4 階 教室 同上 6 階 大ホール
	国際交流センター 1 階
	川内北キャンパス厚生会館 文系食堂
星陵	附属図書館 医学分館 本館 同上 別棟
雨宮	附属図書館 農学分館
	雨宮厚生会館
青葉山	附属図書館 北青葉山分館 同上 工学分館 旧館 同上 新館
	サイクロトロン・ラジオアイソトープセンター サイクロ棟 2 階　会議室
	未来科学技術共同研究センター 本館 2 階 会議室 同上 ハッチェリースクエア 3 階 会議室
	学際科学国際高等研究センター 1 階 大セミナー室
	サイバーサイエンスセンター 1 階 利用相談室 同上 5 階 大会議室

2.2　無線 LAN の利用

無線 LAN システムの利用にあたっては、利用者の身分に応じて、以下に示すアカウントが必要になります。アカウントの取得方法と詳細については、それぞれのシステムの説明(ウェブサイト)を参照してください。

[教職員]

「どこでも TAINS」方式で無線 LAN を利用するには、学内 LAN に接続されたいずれかの VPN サーバで有効なアカウントが必要です。VPN サーバとしては、TAINS の「リモートアクセスサービス」で提供されているものが利用できます。東北大学統合電子認証システム[5, 6]にログインし、利用者がサブ ID/パスワードを設定することによって、これらが上記 VPN サーバのアカウントになります。

「どこでも TAINS」方式では、部局や研究室に設置された VPN サーバを利用することも可能

です。
「eduroam」を利用する場合は、eduroam 用のアカウントが必要になります。利用者は、東北大学統合電子認証システムでサブ ID/パスワードを設定します。サブ ID の後ろにレルム名と呼ばれる @eduroam.tohoku.ac.jp を付加したものが、eduroam の ID になります。接続方法などの詳細については、TAINS の「eduroam アカウントサービス」のウェブサイトを参照してください。

[学生]
「どこでも TAINS」方式で無線 LAN を利用するには、学内 LAN に接続されたいずれかの VPN サーバで有効なアカウントが必要です。教育情報基盤センターの「キャンパス無線 LAN 接続サービス」を利用することで、無線 LAN を利用できるようになります。部局や研究室の VPN サーバを利用できる学生は、それを使うこともできます。
本稿の執筆の時点で、学生は eduroam を利用できません。(学生の eduroam 利用の可否については学内で調整・検討中です。)
なお、研究室に所属している学部(4 年次)・大学院学生は、サイバーサイエンスセンター・CSI 研究室が実施している「eduroam トライアル[7]」に参加することで、研究室の連帯責任の下で eduroam が利用できる場合があります。

[来訪者]
来訪者が eduroam のアカウントを持っている場合は、eduroam 方式で TAINS 無線 LAN システムが利用できます。ただし、本学以外のアカウントを利用している場合は、利用者の端末がゲスト用のネットワークに接続されるため、本学が購読している電子ジャーナルや学内専用のサービスにアクセスすることはできません。また、学外のアカウントで「どこでも TAINS」は利用できません。

3.　部局整備の無線 LAN システムとの関係

TAINS 無線 LAN システムを「どこでも TAINS」の枠組みの中で見ると、アクセスポイントを提供する部局が新たに一つ増えたものとみなすことができます。従来と違うのは、サイバーサイエンスセンターが、自部局だけではなく、全学のキャンパスにアクセスポイントを設置しているという点です。利用者の視点では、「どこでも TAINS」の利用エリアが増えたように見えるだけで、使い方は従来と変わりありません。
eduroam に関しては若干事情が異なります。これまで、部局整備の無線 LAN システムで eduroam のサービスを提供する場合には、ウェブサイトなどへのアクセスをネットワーク側でブロック(禁止)して、一部の VPN プロトコルだけに制限する必要がありました[3]。もし外部利用者(来客)の端末を学内の端末と同じネットワークに収容すると、外部利用者が電子ジャーナルや学内リソースにアクセスできてしまうという問題が生じます。電子ジャーナルの場合は契約違反になりますし、学内のウェブサイトに容易にアクセスできるようでは情報漏えいの危険性が生じます。
TAINS 無線 LAN サービスでは、認証 VLAN の技術を用いて、内部利用者と外部利用者のネットワークを分離しています(図 1)。外部利用者の端末が収容されるネットワークの出口のアドレスは、通常の TAINS のサブネット(130.34.0.0/16 や 192.168.0.0/16)ではなく、新たに設けたゲスト用のネットワークです。内部利用者は従来どおり学内のアドレスが利用できるので、電子ジャーナルや学内各種サーバへのアクセスは容易にできます。このような仕組みにより、利便性を確保しつつ、学内ネットワークの安全性を維持しています。
各部局には一般にゲスト用のネットワークが用意されていませんが、前述のように VPN 専用のアクセスでは利便性が非常に悪く、特に海外からの来客は遠隔地の VPN サーバに安定に接続できないことが多いために、実質的にインターネットが利用できないという問題が生じます。そこで、TAINS 無線 LAN サービスでは、部局所有のアクセスポイントを収容するための専用ネットワー

クも提供することにしました。具体的には、TAINS の VLAN 機能を利用して、部局の無線 LAN システムを収容します(図 1)。このアクセスポイント収容ネットワークを利用することによって、部局の無線 LAN システムの安全性向上と、利用環境の共通化による無線 LAN の利便性向上などが期待できます。

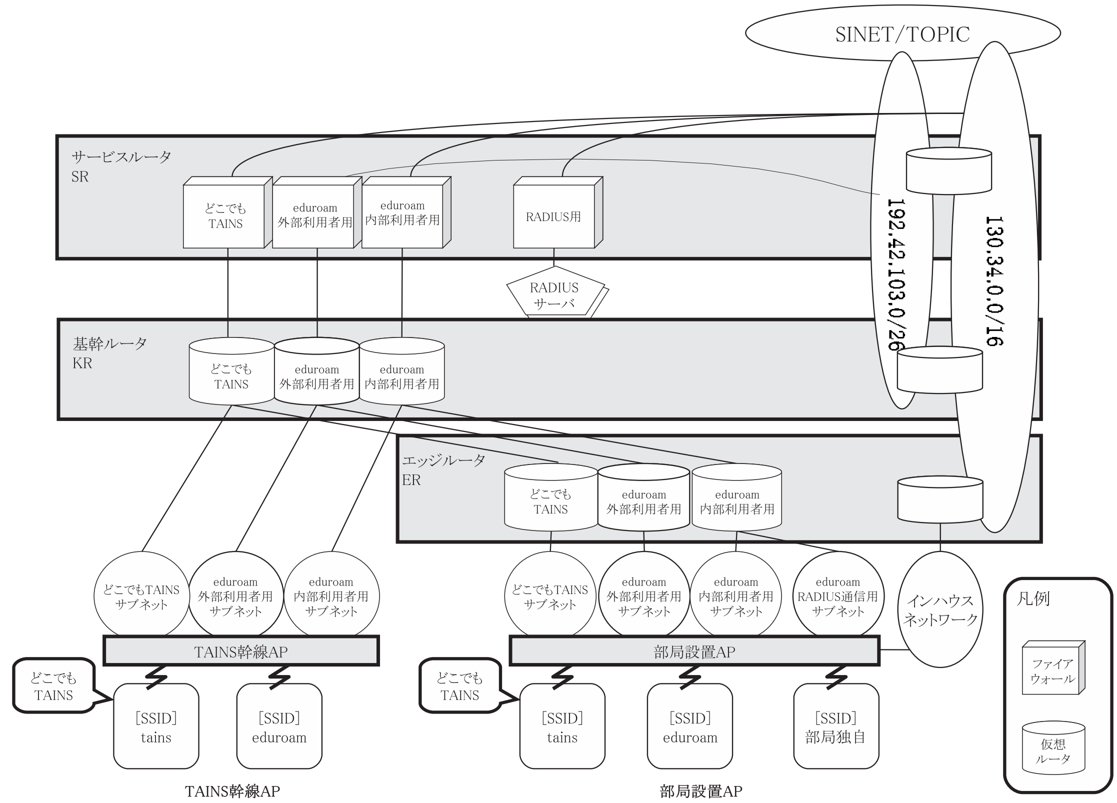


図 1. TAINS 無線 LAN システムのネットワーク構成

4.　むすび

TAINS 無線 LAN システムの概要と、部局整備のシステムを含めた東北大学におけるキャンパス無線 LAN サービスの現状について解説しました。TAINS 無線 LAN システムのサービス開始により、本学のキャンパス無線 LAN の利便性が大幅に向上するものと期待しています。

SENAC で既報のように[1]、「どこでも TAINS」方式と eduroam 方式にはそれぞれ一長一短があります。サイバーサイエンスセンターでは、当面の間、両者を並行して展開していく予定です。

参考文献

[1] 後藤英昭, 今井哲郎, 曽根秀昭, “eduroam とキャンパスユビキタスネットワーク,” 大規模科学計算システム広報 SENAC, Vol.40, No.1, pp.84-87, 2007.

[2] 後藤英昭, 水木敬明, 曽根秀昭, “無線・有線 LAN ローミングシステム「どこでも TAINS 2」,” TAINS ニュース 2008.3.21, No.35, pp.5-7, 2008.

[3] eduroam JP:　http://www.eduroam.jp/

[4] 東北大学総合情報ネットワークシステム TAINS:　http://www.tains.tohoku.ac.jp/

[5] 木下哲男, 伊藤清顕, 早川美徳, 寺澤篤史, “統合電子認証システムが始まりました,” TAINS ニュース 2010.3.15, No.38, 2010.

[6] 東北大学統合電子認証システム:　http://www.bureau.tohoku.ac.jp/auth/index.html

[7] eduroam トライアル:　http://www.rd.isc.tohoku.ac.jp/csi/eduroam-trial/

[共同研究成果]

Reproduction of diurnal variability of Meiyu precipitation in the Yangtze River Valley using non-hydrostatic model

Guixing Chen, Weiming Sha and Toshiki Iwasaki
Department of Geophysics, Graduate School of Science, Tohoku University

Meiyu precipitation in the Yangtze River Valley (YRV) tends to reach its maximum in the morning. The middle and lower reaches of YRV are identified as two preferred regions of morning heavy rainfalls. To clarify the underlying physics responsible for this diurnal variability, a series of high-resolution simulations are conducted to examine nocturnal convection systems for a 5-day period of repeatedly occurring heavy rainfalls. Both the rainfall amount and its diurnal cycle in the vicinity of Meiyu front are well reproduced; so are the associated low-level processes that vary diurnally and regionally. It is found that the formation and evolution of nocturnal convections exhibit strong association with the nighttime intensification of the low-level jet (LLJ) over south China and the arrival of warm/moist air pool to the Meiyu frontal zone in the YRV. Overnight they strengthen the Meiyu front from its south sector. To the south of the Meiyu front and the north terminus of LLJ, the moisture convergence and convective instability are distinctly enhanced, triggering the nocturnal growth of convection and producing the strong morning rainfall.

1. Introduction

During the warm season, East Asia is featured distinctly by the march of the summer rainband, which is well known as the Meiyu in China and the Baiu in Japan. It brings plenty of rainfall for the East Asian countries; it sometimes causes heavy rainfalls and severe disasters. Along with this seasonal change, diurnal cycle in cloudiness and precipitation is found to be another key aspect of the weather and climate. Diurnal variability of Meiyu/Baiu precipitation is studied to expand our understanding and predicting on the precipitation systems and their underlying physical processes.

Observation and numerical analyses on Meiyu precipitation systems have been conducted for decades [1]. The rain-bearing systems in Yangtze River Valley (YRV) come from the mesoscale ones that are generally embedded in the synoptic/large-scale Meiyu frontal zone. Recent works make use of the high-resolution rainfall data and reveal that the Meiyu precipitation tends to reach its maximum during the late night and morning [2, 3]. This morning signature of precipitation is attributed to the long-lived organized convection such the mesoscale convection systems [4-6]. Furthermore, the diurnal cycle varies in regions of the YRV. The precipitation in the middle reach of YRV (MYRV, 105–112°E, 25–30°N) is attributed to the migrating episodes that initial at the east slope of the plateau since midnight and move eastward during late night and morning. However, the precipitation in the lower reach of YRV (LYRV, 112−120°E, 25−30°N) results from the local eruption of moist convections driven by the enhanced monsoon flow during the early morning.

There have been some difficulties in reproducing the diurnal cycle of Meiyu precipitation in the YRV. The Meiyu precipitation results mainly from the heavy rainfall events whose location and timing are difficult to predict in numerical models. On the other hand, the high-resolution model is necessary for resolving the mesoscale features of rain-bearing systems, while a large domain of simulation is needed for capturing the associated synoptic or large-scale environment and topographical impacts of the Tibetan Plateau.

The need of the high spatio-temporal resolution over a large model domain highlights the importance of the huge computation resource and parallel computation efficiency. In this brief report, we take the advantage of supercomputer systems to conduct a series of simulation to reproduce the diurnal variability of Meiyu precipitation in the YRV. Technical supports for speeding up of the simulation are given through the collaborative work with Cyberscience Center, Tohoku University.

2. Model setting and numerical experiment

The model used in this study is a non-hydrostatic model developed by the Japan Meteorological Agency and Meteorological Research Institute (JMA-NHM). A detailed description of JMA-NHM is provided by Saito et al. [7]. The basic equations in the model are fully compressible equations. The semi-implicit time-integration scheme [horizontally implicit–vertically implicit (HI-VI) scheme] is employed for treating the gravity wave. The advection scheme (4-order flux type horizontally and 2-order center difference vertically), atmospheric radiation schemes, mixed-phased cloud physics, etc., are also incorporated into the model. The modified Kain–Fritsch convection scheme and the improved Mellor–Yamada level-3 closure model are chosen for the parameterization of atmospheric processes.

The simulation targets are the heavy rainfall events that occurred repeatedly during 11−16 June 1998; their diurnal cycles are highly similar to the climatological ones. The model domain contains 504×364 horizontal grid points with 10-km grid, which covers East Asia and the Tibetan Plateau (Fig. 1). 38 vertical levels with variable grid intervals from $\Delta z = 40$ to $\Delta z = 1060\,\mathrm{m}$ are used, where the lowest level and the model top are located at 20 and 19270 m. To reproduce the timing and locations of heavy rainfalls more reliable, five integrations rather than a successive one are conducted. Each initiates at every 00UTC of 11−15 June for daytime spin-up and lasts for 36 h; time step is 30 s. Diurnal cycles of 9-33 h are examined with emphasis on the nocturnal evolution of precipitation systems.

The elevation in the model is given by GTOPO30, which is derived by the U.S. Geological Survey's Center. Initial and lateral boundary conditions for horizontal wind, pressure, temperature, and water vapor are given by 6-hourly 0.5° GEWEX Asian Monsoon Experiment (GAME) reanalysis data. The initial conditions of cloud microphysical quantities are provided by the corresponding 6-h forecasts of JMA-NHM from 18UTC. The 3-hourly rainfall data of the Tropical Rainfall Measuring Mission (TRMM) are used to validate the simulation of rainfall amount and diurnal cycle.

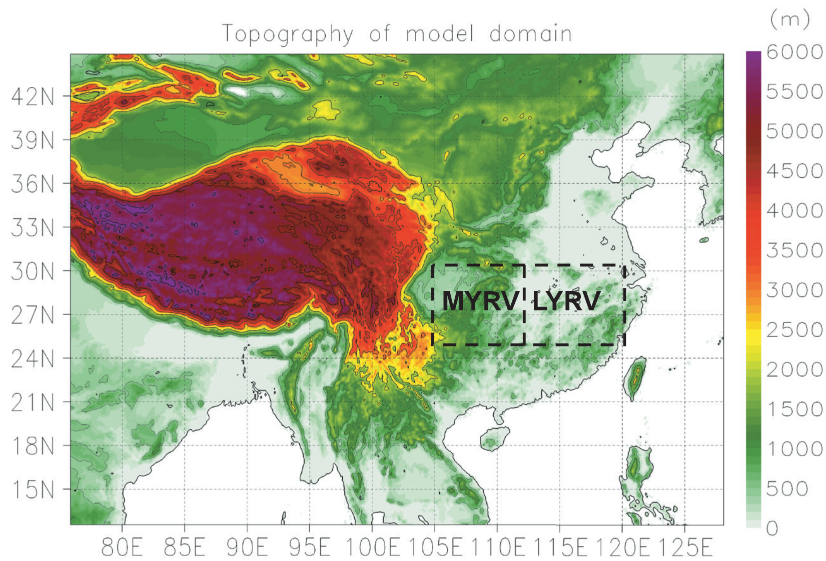


Figure 1. The Model domain and smoothed elevation of GTOPO30 (shaded). Dashed rectangles mark the MYRV and LYRV.

3. Rainfall validation

For rainfall validation, the rainfall amount during 11–16 June is compared between the TRMM-3B42 observation (fig. 2a) and the simulated one (fig. 2c). Both the observations and simulation show that heavy rainfalls appear in the MYRV along the east periphery of the Yun-gui Plateau and in the west-east oriented frontal zone in the LYRV, where the flood took place. Along the Meiyu rainband at 27–30°N, there are a number of precipitation centers at 110°E, 114°E, 116–118°E and 125°E, respectively. The extreme amount in the LYRV exceeds 300 mm in 5-day period. The simulated Meiyu rainband is narrower and more intense than that in TRMM measurements, partially due to its high resolution. Overall, the simulations reproduce well the overall distribution, mesoscale feature, and magnitude of the Meiyu heavy rainfalls in the YRV.

Time-longitude diagrams of hourly rainrate along the latitude of 25–30°N are used to evaluate the modelling on the individual event of heavy rainfall and mean diurnal cycle. Figures 3a and 3b show that three intense events in the MYRV at the mornings of 13, 14 and 16 June are captured by simulation. They are associated with the eastward propagating convections that develop at 105–106°E at late night and strengthen at ~110°E at morning. The rainfall becomes evanescent at the daytime of 14 June, partly returns at the morning of 15

June, and fully recovers at the morning of 16 June. The simulations reproduce the relative weak morning rainfalls on 12 and 15 June, although the rainy area is slightly overestimated. In the LYRV, both the observation and simulation show that the extensive rainfall takes place in the morning hours of 12−16 June. Note that, the eastward migrating signal from the MYRV becomes obscure when arriving at the LYRV. More likely, the Meiyu precipitation systems develop explosively within hours at one same morning in both the MYRV and LYRV. For example, strong rainfall takes place 105−111°E from the late night of 15 June to next morning; meanwhile intense rainfall appears at 114−115°E and 118−120°E. It seems clear that the diurnal cycles in the MYRV and LYRV are independent to each other.

Figures 3c and 3d show the 5-day composite diurnal cycle of rainfall. Both the observation and simulation show that the diurnal cycle of rainfall has two preferred regions: MYRV and LYRV. A gap is readily identified at 112°E that divides these two regimes. Rainfall tends to develop at the east periphery of the Yun-gui Plateau at 15UTC, and then moves to 110°E where it attains its maximum at 21−03UTC. This eastward propagation has a slow phase speed of 9−13 m/s, i.e., traveling across ~4° longitude within 8−12 h. This migrating signal is thus confined in the MYRV. It accounts only for the local eastward-shifted diurnal phase. Instead, another kind of diurnal cycle can be distinguished in the LYRV. Precipitation systems have a rapid local development at 18UTC and produce the extensive rainfall in the morning hours. These regional features in the LYRV and MYRV are highly consistent to the climatological pattern of diurnal cycle, underscoring the fact that the heavy rainfall is a determining factor of the diurnal rainfall cycle and regional difference.

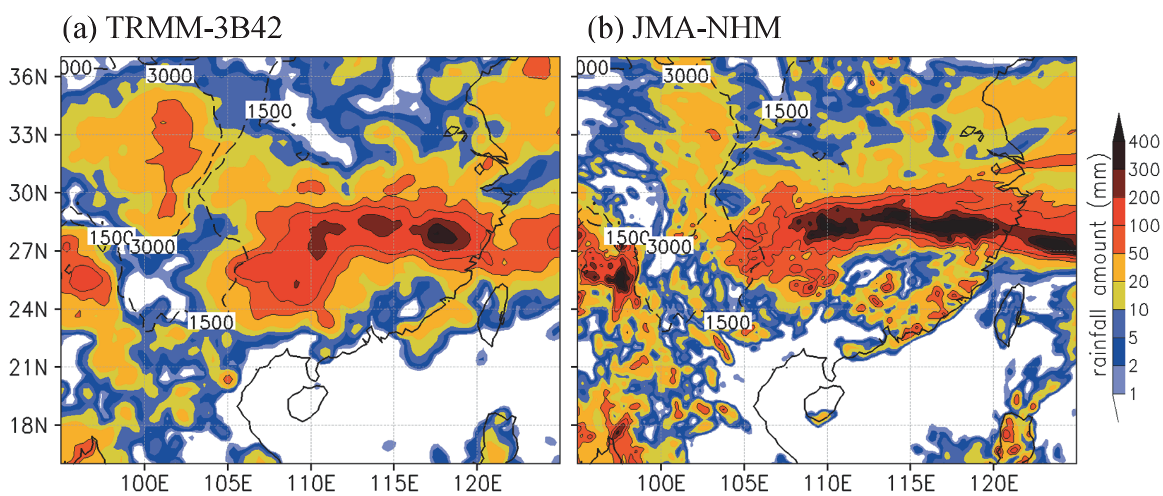


Figure 2. Accumulated rainfall amount from 09UTC June 11 to 09UTC June 16, 1998. (a) TRMM-3B42 observation; (b) JMA-NHM simulation.

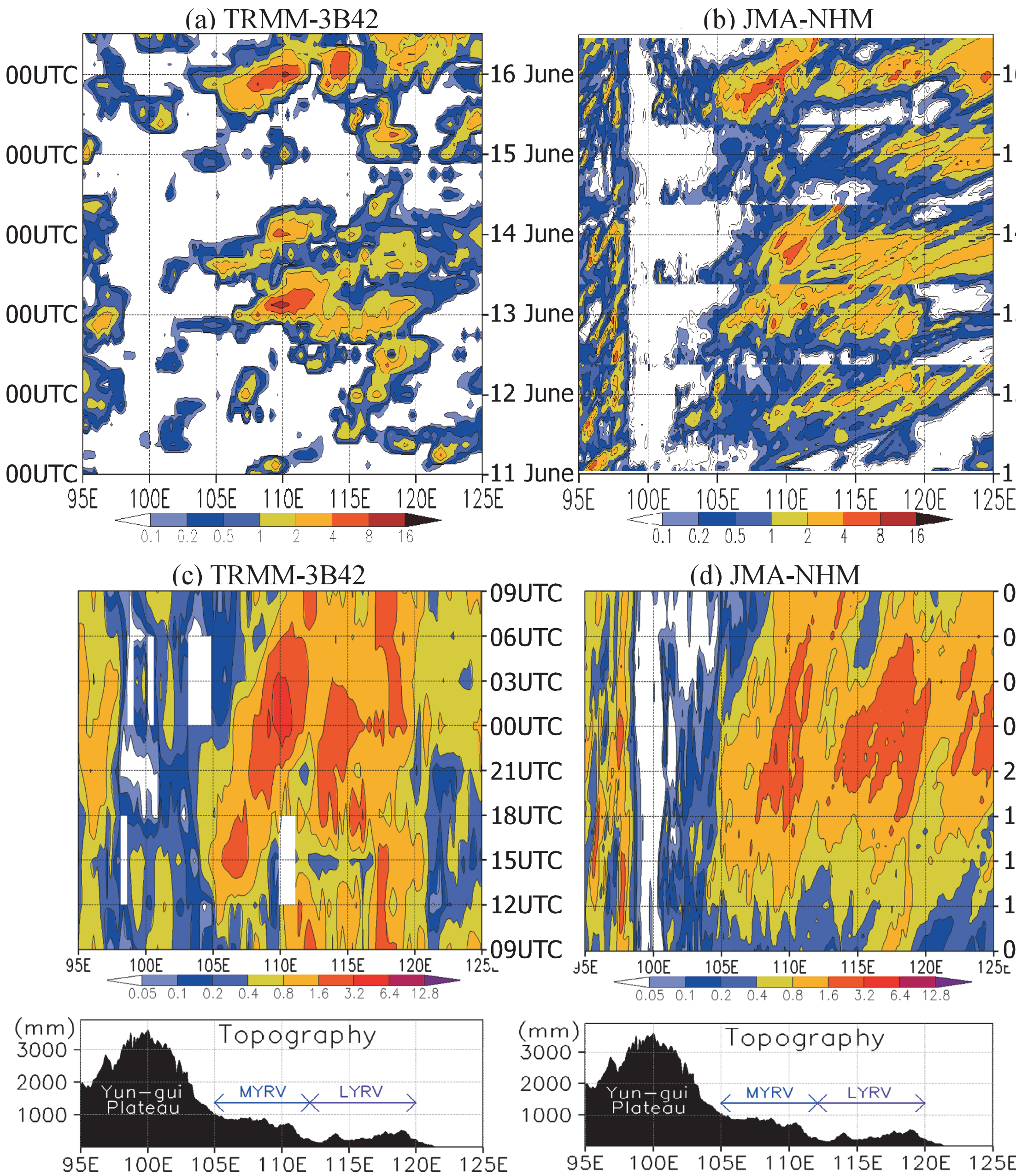


Figure 3. Time-longitude variation of rainfall averaged in the zone of 25~30°N by (a) TRMM-3B42 observation and (b) JMA-NHM simulation. (c) and (d) show the 5-day composite of diurnal cycle for the observation and simulation, respectively. The terrain elevation of 25~30°N is shown in the bottom panel.

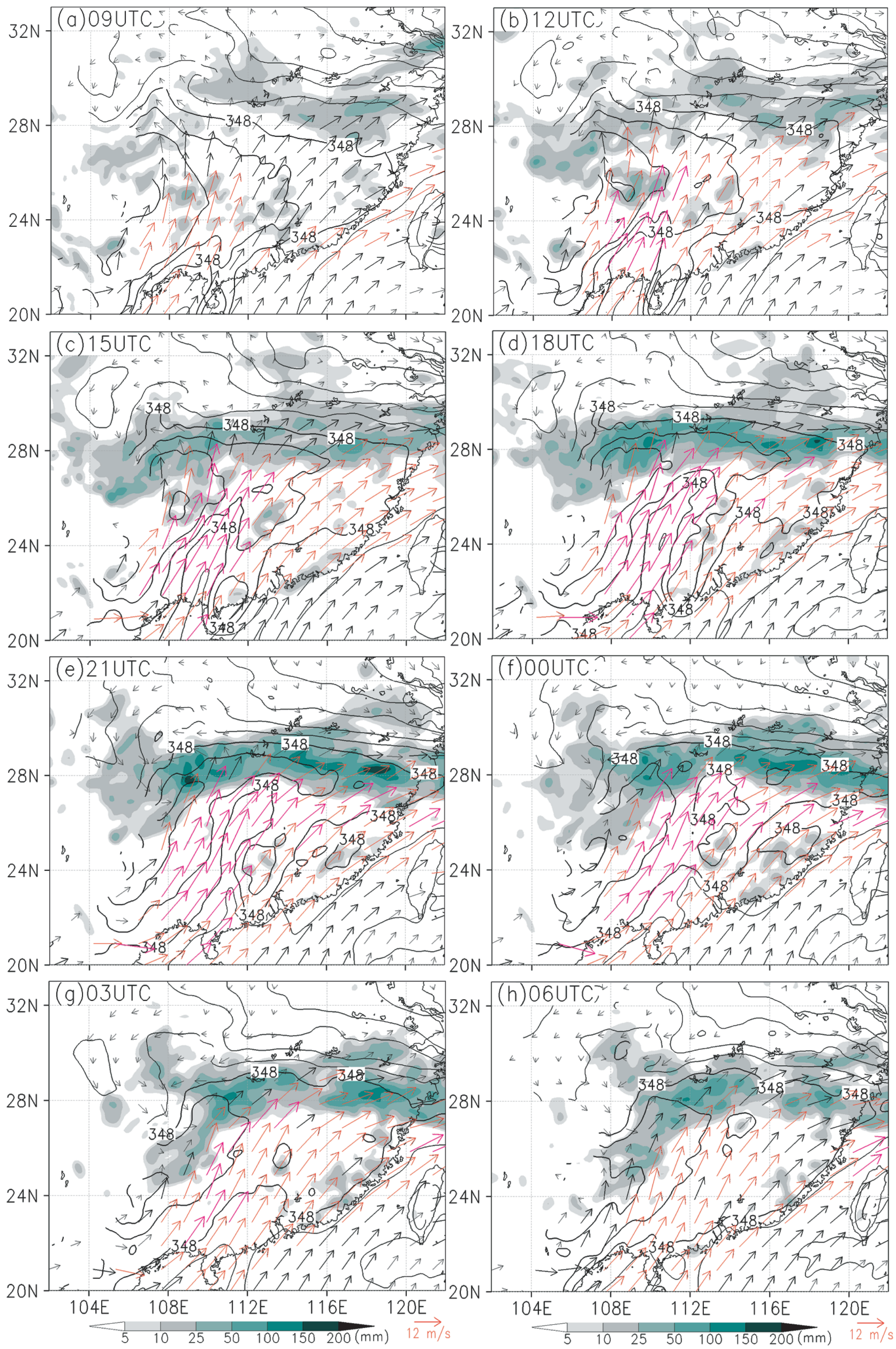


Figure 4. The 3-hourly simulated rainfall (shaded) 850 hPa horizontal winds (vector) and θe (contour) averaged during 11−16 June, 1998. The red vector shows the wind speed exceeding 12 m/s.

4. Diurnal varying low-level processes associated with nocturnal precipitation

Many studies have recognized that the lower-tropospheric conditions are crucial in allowing the organized convection to persist overnight [8, 9]. In the southeastern China, the organized convection can be reinforced by the nocturnal enhancement of monsoon flow in meso-α or synoptic scale [6]. However, the coarse observation data have limitation resolving the diurnal variation of mesoscale systems. Here, the high-resolution simulations are used to facilitate our studies on mesoscale/regional aspects of the thermodynamic and flow structure on a diurnal scale.

Figure 4 portrays the 3-hourly evolutions of simulated rainfall, 850 hPa horizontal wind and equivalent potential temperature (θ_e), averaged for a 5-day period. Through the period, the low-level anticyclone is situated at the northeastern part of South China Sea. The southwesterlies prevail over south China especially along the east periphery of the Yun-gui Plateau. Figures 4a shows that southwesterlies are relatively weak, ranging from 10 to 12 m/s at the afternoon. The warm/moist air with $\theta_e \geq 348$ K dominates the inland south China and develops particularly in the west. It may result from the daytime surface flux over the heated landmass, convective mixing and subsequent shallow convections [9]. The simulated rainfall is visible but weak in the YRV.

At evening and midnight (12–15UTC), the wind speed of southwesterlies increases to 16–20 m/s in the west of south China and 13–16 m/s in the east (figs. 4 b-c). A remarkable acceleration of 6–8 m/s (3–4 m/s) can be estimated for the west (east) region of south China. This nocturnal enhancement of low-level wind is in a good agreement with the sounding observation over south China (not shown). It highlights that the model has a good performance reproducing the diurnal variation of low-level processes as well. The intensified low-level wind exhibits as a converging southerly flow along the east periphery of the plateau, where convections become more vigorous. The high-θ_e air moves fast northward from south China to the YRV, while the low-θ_e air replaces the coastal region of south China.

Figures 4 d-f show that the southwesterly LLJ attains the strongest in strength (~20 m/s) through the late night and morning (18–00UTC). The wind speed maximum tends to extend northward or northeastward. This downstream extension of LLJ leads to the wind speed increasing by 4–5 m/s to the south of 28°N. The pool of warm/moist air ($\theta_e \geq 352$ K) arrives at the YRV. Consequently, the meridional gradient of θ_e value experiences a rapid increase, especially at the south sector of Meiyu front (28–30°N). It implies that the nighttime frontogenesis of Meiyu front is driven largely by the low-level forcing from the south. The arrival of warm/moist air pool also increases the convective instability to the south of the front. On the other hand, the confluent flow of LLJ enhances moisture convergence and mesoscale lifting. All these favorable conditions lead to a rapid growth of moist convections and hence the heavy rainfalls.

Figures 4 d-f also show that the southerlies are intensified over the west region of south China with an earlier and stronger phase than that in the east. It is found that this acceleration of low-level wind is driven by the large imbalance between the pressure gradient force and Coriolis force since evening when turbulent stress becomes decayed. This is responsible for the earlier initiation of convection in the MYRV (figs. 3 c-d). Meanwhile, the enhanced southerlies tend to veer to southwesterlies that steer away from the plateau; northerlies appear over the Sichuan Basin, a signature of mesoscale cyclone or vortex. Over there, the intense rainfall moves eastward or southeastward.

Figures 4 g-h show that southwesterlies over south China return to a suppressed phase at late morning and early afternoon; the wind speed declines to ~12 m/s. The warm/moist air pool in the YRV has been consumed and the value of θ_e becomes less than 348 K, while it begins to recover over south China. The heavy rainfalls to the south of Meiyu front begin to decay gradually as they continue migrating eastward.

5. Concluding remarks

To clarify the mechanism for the morning precipitation in the Meiyu regime, a series of high-resolution simulations are conducted to reproduce successfully the heavy rainfall events in the YRV. It is found that the two preferred regions of heavy rainfall in the MYRV and LYRV are attributed to the nocturnal intensification of monsoon flow and associated low-level moist processes. As a response, the low-level frontogenesis, moisture convergence, convective instability and mesoscale lifting are distinctly enhanced to the south of the Meiyu front, where the nighttime formation of convection and morning heavy rainfalls take place. Therefore, the timing of convection initiation is consistent with the arrival of warm/moist air pool, which is transported by the enhanced nocturnal LLJ. It highlights the importance of the diurnal-scale interactions among the monsoon flow, Meiyu front, boundary processes and topographic effects on the formation and evolution of Meiyu heavy rainfalls. We consider that the further investigation on the mechanism in regional and diurnal scales in the future is helpful to improve the predicting skill on the Meiyu precipitation system. The high-resolution simulations using the supercomputer systems will promote our understanding on the weather and climate over the East Asian countries.

Acknowledgements

Parts of the experimental results in this research were obtained using supercomputing resources at Cyberscience Center, Tohoku University. Technical supports were given through the collaborative work with Cyberscience Center.

References

[1] Ding, Y. H., 1992: Summer monsoon rainfalls in China, *J. Meteorol. Soc. Jpn.*, *70*(1B), 373-396.

[2] Yu, R., T. Zhou, A. Xiong, Y. Zhu, and J. Li, 2007a: Diurnal variations of summer precipitation over contiguous China, *Geophys. Res. Lett.*, *34*, L01704, doi:10.1029/2006GL028129.

[3] Chen, G., W. Sha, and T. Iwasaki, 2009a: Diurnal variation of precipitation over southeastern China: Spatial distribution and its seasonality, *J. Geophys. Res.*, 114, D13103, doi:10.1029/2008JD011103.

[4] Yu, R., Y. Xu, T. Zhou, and J. Li, 2007b: Relation between rainfall duration and diurnal variation in the warm season precipitation over central eastern China, *Geophys. Res. Lett.*, *34*, L13703, doi:10.1029/2007GL030315.

[5] Li Z., T. Takeda, K. Tsuboki, K. Kato, M. Kawashima, and Y. Fujiyoshi, 2007: Nocturnal evolution of cloud clusters over eastern China during the intensive observation periods of GAME/HUBEX in 1998 and 1999, *J. Meteorol. Soc. Jpn.*, *85*(1), 25-45.

[6] Chen, G., W. Sha, and T. Iwasaki, 2009b: Diurnal variation of precipitation over southeastern China: 2. Impact of the diurnal monsoon variability, *J. Geophys. Res.*, 114, D21105, doi:10.1029/2009JD012181.

[7] Saito, Kazuo, and Coauthors, 2006: The Operational JMA Nonhydrostatic Mesoscale Model. Mon. Wea. Rev., 134, 1266-1298.doi: 10.1175/MWR3120.1

[8] Trier, S. B., C. A. Davis, D. A. Ahijevych, M. L. Weisman, G. H. Bryan, 2006: Mechanisms supporting long-lived episodes of propagating nocturnal convection within a 7-Day WRF model simulation. *J. Atmos. Sci.*, 63, 2437-2461, doi: 10.1175/JAS3768.1

[9] Yamada, H., B. Geng, H. Uyeda, and K. Tsuboki, 2007: Role of the heated landmass on the evolution and duration of a heavy rain episode over a Meiyu-Baiu frontal zone, *J. Meteorol. Soc. Jpn.*, 85(5), 687-709, doi:10.2151/jmsj.85.687.

白

［共同研究成果］

Building-Cube 法による非定常流体解析と空力音推算

佐々木大輔，恩田博，中橋和博
東北大学大学院工学研究科航空宇宙工学専攻

1.　緒言

近年，世界各国の空港で航空機の離発着時の騒音規制が強化される方向にあり，今後騒音の大きい機体の運航は制約が増えていくことが予想される．日本においても，騒音による空港運用時間の制限や機体の騒音レベルに応じた離発着料が徴収されている．そのため，航空機メーカーは離発着時に生じる騒音の低減を目指して様々な開発を進めてきた．その結果，主要な騒音源であったエンジン騒音（ファン騒音・ジェット騒音）の低減が可能となってきた．エンジン騒音の低下に伴い，従来あまり検討されてこなかった機体から生じる空気力学的な騒音（風切り音）も無視できなくなりつつある．そのため，代表的な機体騒音源である高揚力装置や降着装置の騒音削減を目的とした研究が盛んに行われている[1-8].

（独）宇宙航空研究開発機構（JAXA）では，図 1 に示す主脚騒音計測用モデル（LEG 形状，Landing gear noise Evaluation Geometry）により，実験的・数値解析的に脚から生じる空力騒音に関しての研究を行っている[6-8]．図より分かるように LEG モデルには多数の部品が存在している．これら部品の空力音発生への影響を CFD（数値流体力学）で調べるには，計算空間内で形状を忠実に再現することが不可欠であると共に，個々の部品を取り外す，あるいは位置なり形を修正するなどして様々な解析を行う必要がある．

航空機の翼等の CFD では，翼表面に発達する境界層を精度良く解像する必要性から計算格子としては壁に沿って格子線を配置する構造格子が古くから用いられてきた．その後，スーパーコンピュータの発達で 3 次元解析も頻繁に行われるようになってからは，複雑形状を扱うための非構造格子が開発された．今日では市販 CFD ソフトの多くも非構造格子を用いている．しかしながら，その非構造格子でも航空機の主脚（降着装置）のような複雑な形状になると格子生成に要する時間が大きく，各部品の影響を調べる場合では前処理にかかる時間を無視できない．

複雑形状を扱うという点で，格子線を座標方向にとる直交格子法が見直されている．この場合，物体は格子線との交わり点として表現されるため，いかなる複雑な形状でも格子生成は容易で短時間で行える．しかしながら直交格子の大きな難点は，滑らかな物体壁も基本的には階段状で表現することになり，境界層の解像が難しいことである．また，物体近くの細かな流れに合わせて格子を作ると，空間全体では膨大な格子点数が必要となることも大きな課題であった．

筆者らは，非構造格子 CFD ソルバ（TAS[9]）の開発で学んだ様々な課題およびスーパーコンピュータのさらなる性能向上に着目し，直交格子の利点を活かした Building-Cube 法（BCM）を開発してきた[10-13]．Building-Cube 法では，流れ場を様々なサイズの Cube と呼ばれる立方体領域に分割し，各 Cube 内には同数の等間隔直交格子 Cell を配置するというブロック型直交格子法である．物体壁を単純な階段状に表現しているため，物体境界の幾何学精度を保ち且つ境界層内の粘性領域を解くには物体近傍で非常に細かい格子が必要である．しかし，格子生成のような人的作業が不要になることの利点は，特に図 1 のような複雑形状問題では非常に大きい．また，圧倒的に細かな格子を用いることで計算結果の格子依存性や物理モデルへの依存性も除去できよう．従って，Building-Cube 法はスーパーコンピュータの性能の更なる向上を期待したアプローチである．

図 2 に，翼周りに生成した BCM 格子を示す．物体近傍では小さい Cube サイズを使用するため，十分に細かい格子を生成することになり，物体を詳細に表現することが可能となる．一方，翼から離れた領域では Cube サイズを大きくすることで，格子点数の削減を図っている．本アプローチにより，各 Cube における計算負荷は均一となり，大規模並列計算環境時でも十分な並列性能を上げることが期待できる．

本研究では，3 次元非圧縮性流体解析ソルバを JAXA の LEG 形状に適用する．図 3 に示すタイヤのみ存在する単純な形状（Simple 形状）と脚扉（Door）・サイドブレース（Side brace）・トルクリンク（Torque link）等が付属した複雑な形状（DST 形状）の二つの形状を用いて計算を行い，今村らの結果[6,8]との比較を通して本手法の有効性の検証を行うと共に，今後の空力音解析に向けての知見を得る．

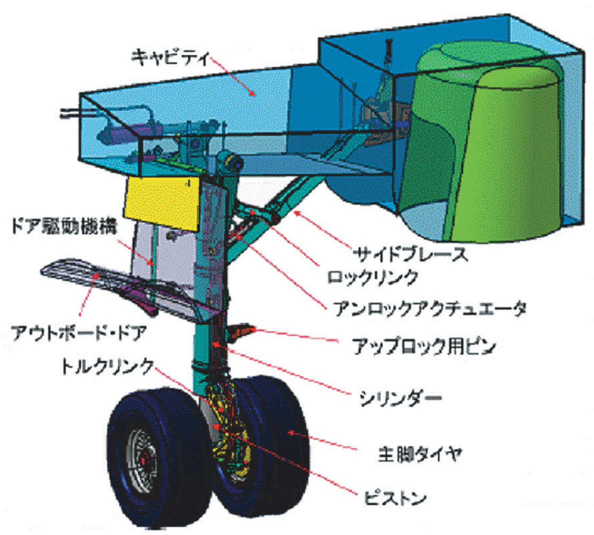
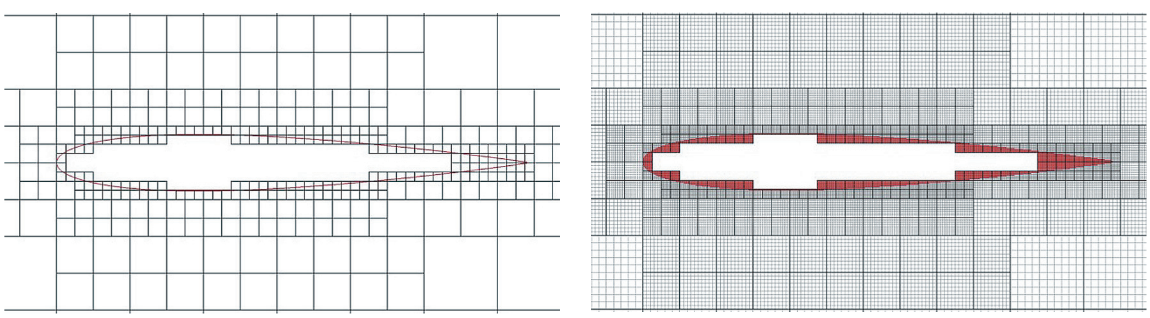


図 1　JAXA 主脚騒音計測用モデル [6-8]



(a) Cube 配置　　(b) Cell 配置

図 2　翼周りの BCM 格子生成

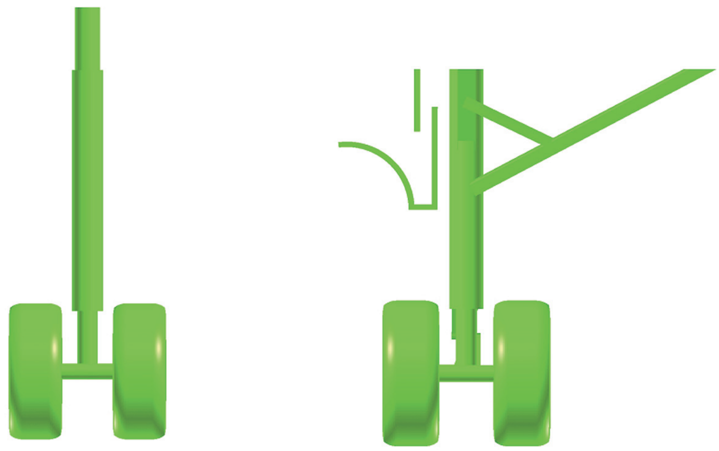


図 3　解析に用いる主脚形状（左：Simple 形状，右：DST 形状）

2.　空力音解析手法

2.1　解析方法の概要

ナビエ・ストークス方程式は音波の伝播も表すため，圧縮性も考慮したナビエ・ストークス方程式を連続の式およびエネルギー方程式と連立して解くことにより，理論的には空力音を求めることが可能である．このような手法は空力音の直接計算と呼ばれ，高速気流から生じる空力音の予測などに用いられている．しかし，低マッハ数流れの場合には音波の伝播は極めて微弱な圧力変動となるため，これを捉えるには膨大な数の空間メッシュが必要となり，現実的ではない．したがって低マッハ数の流れでは，音源となる流れ場を非定常乱流解析などにより求め，音の伝播を波動方程式により別々に計算する分離解法を用いる手法が一般的である．一般に，低マッハ数の流れ場においては音波の伝播に伴う密度変動は微小であり，音から流れ場へのフィードバックは無視できると考えられるため，ここでは流れ場の計算において流体の圧縮性を考慮しない．

2.2　分離解法の基礎式

空力音も流れの微小変動によるものであるため，流れの基礎方程式を変形することで静止流体中を伝播する音を表す Lighthill 方程式[14]が得られる．このとき，物体の寸法に比較して音波の波長が十分長く，かつ，観測点が音波の波長に対して十分遠方にある場合には波動方程式の簡略解として，観測点における圧力変動 Pa は以下の Curle の式[15]で与えられる．

$$P_a = \frac{1}{4\pi a}\frac{\mathbf{x}_i}{|\mathbf{x}|^2}\cdot\frac{\partial}{\partial t}\int_S n_i\, p(\mathbf{y}, t - |\mathbf{x}|/a)\,dS \tag{1}$$

ここで，a は音速，$\mathbf{x}$ は観測点位置，p は音源となる物体表面の圧力変動，n は物体表面 における外向き単位法線ベクトルをそれぞれ表す．本研究では BCM を用いた非圧縮性ソルバにより音源の流れ場を計算したのち，Curle の式を用いて遠方場における音圧を求める分離手法をとる．上記で得られた圧力変動に FFT を適用して周波数領域の変動に変換した音圧の実効値 p_e を用いて，観測点での音圧レベル（SPL: Sound Pressure Level）は最終的に以下の式で求められる．

$$SPL = 10\log_{10}\left(\frac{p_e^2}{p_0^2}\right)$$

ここで，p_0 は基準圧力（2.0×10^{-5}[Pa]）である．

2.3 BCM (Building-Cube Method)

BCM は格子生成，流体解析アルゴリズム，後処理の単純性を保つために等間隔直交格子を採用している高解像度の計算手法である．BCM では流れ場を多くの様々なサイズの”Cube”と呼ばれる立方体の領域に分割する．各 Cube サイズは物体の形状に合わせて決定され，各 Cube の内部には等間隔直交格子”Cell”を生成して計算が行われるが，全ての Cube が同数の Cell を持つことで高い並列化効率が得られる．また，直交格子である BCM では物体境界は階段状表現によって表現される．したがって物体境界の幾何学精度を保つため，また境界層内の粘性領域を解くために物体近傍では非常に細かい格子が必要である．図 4 に本論文の解析で使用する円柱周りの BCM 格子を示す．この図からわかるように物体表面は階段状に近似された非常に細かな Cell によって表現されている．

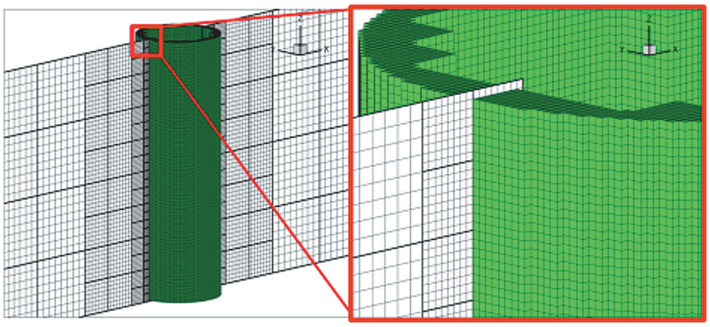


図 4　円柱まわりの BCM 格子（太線は Cube 境界を表し，細線は Cell を表す）

3.　円柱まわりの空力音解析

3.1　解析方法の概要

航空機主脚の計算を行う前に解析手法の妥当性を検証するために円柱による検証計算を行った．ここでは，文献[16]にある円柱と同様の条件で解析し結果を比較した．本研究では格子解像度の異なる 2 つの円柱（Coarse, Fine）を用意した．これら 2 つの円柱は Cube 数が等しく Cube 内部の Cell 数が異なる．最小格子幅は円柱直径 D を基準にして，Coarse が 1.2×10^{-2} D，Fine はその 1/2 の 6.1×10^{-3} D である．図 4 は Coarse Cylinder の格子を可視化したものである．また，空力音の観測位置は円柱の真横 1m の位置とする．表 1 に解析に用いた条件を記す．

表 1　円柱まわりの解析条件

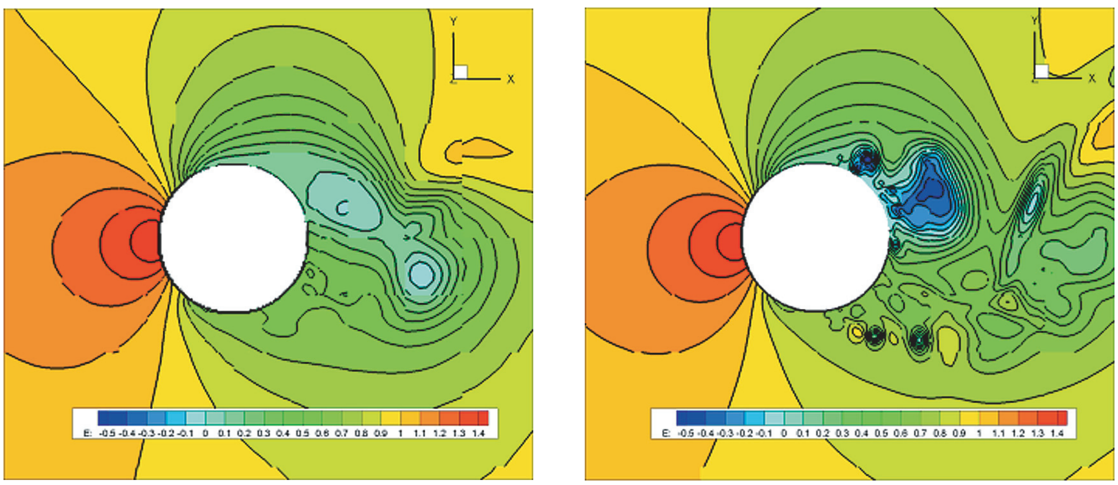
	Coarse	Fine
Cube 数	7,552	7,552
各 Cube 内 Cell 数	8^3	16^3
総 Cell 数	3,866,624	30,932,992
最小格子幅	1.2×10^{-2}D	6.1×10^{-3}D
円柱直径　D	1.0×10^{-2} m	
流速	15 m/s	
Reynolds 数	1.0×10^{4}	
サンプリング周波数	15,000 Hz	
サンプリング時間	0.033 sec	
解析領域	x: 50D, y: 50D, z: 4D	

3.2　計算結果

図5にCoarseとFineの圧力場を示す．どちらも円柱後方に渦が発生していることがわかるが，Fineの方がより細かな渦構造まで捉えられていることが確認できる．また，図6にそれぞれの格子における後流の圧力係数（Cp）等値面の様子を示す．これはCp=-1.8の等値面の可視化図であり，FineのほうがCoarseに比べて円柱近傍の細かな圧力変動を捉えられていることが分かる．

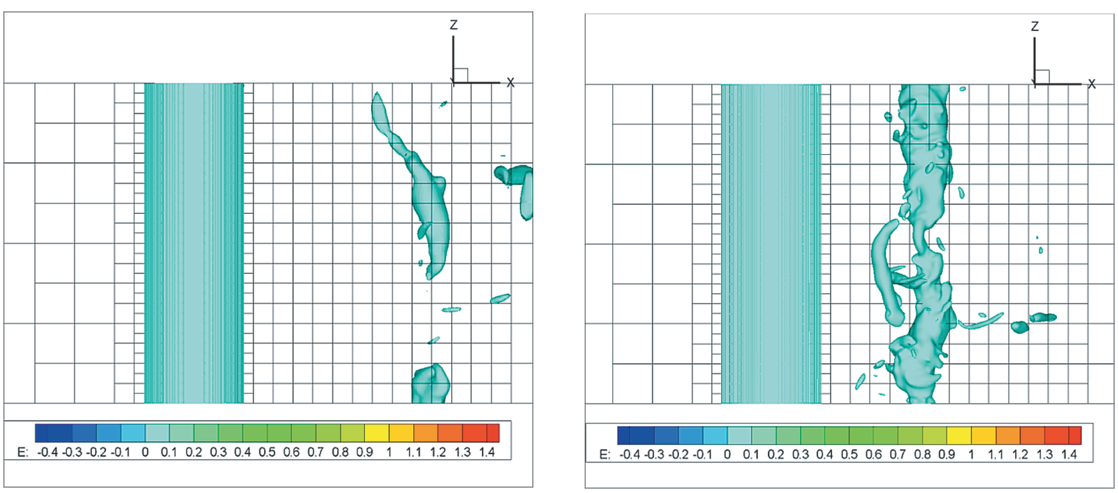
次に円柱の表面圧力変動から(1)式を用いて算出した音圧とSPL (Sound Pressure Level) を比較し，図7及び図8に示す．今回の流体解析の結果を，加藤らによる数値解析と実験の結果[16]と比較する．図7より，Coarseの周期は加藤らの解析結果と多少の差違が確認できるが，Fineの周期は計算結果と一致していることが分かる．また，図8より，音圧が最大となる周波数はCoarse,Fine共に実験結果とよく一致しているが，CoarseではSPLの値が30dB程度も小さく推定された．これはCoarseでは格子解像度が低いために，細かな渦構造が捉えられず円柱近傍の圧力変動が反映されずに値が小さくなっていると考えられる．一方，Fineでもピークの周波数は10dB程度小さく推定されているが，全体的に見ると実験値を良く予測できているといえる．

一般に，空力音の放出周波数はカルマン渦の放出周波数に等しいことから，円柱のストローハル数を元に計算するとその放出周波数は300[Hz]であると求められる．図8より実験値と同様に計算結果も300[Hz]にピークを持つため，本手法がピーク値をよく予測できているといえる．



(a) Coarse 格子　　(b) Fine 格子

図5　円柱近傍の圧力分布



(a) Coarse 格子　(b) Fine 格子

図6　円柱後流における圧力等値面の可視化

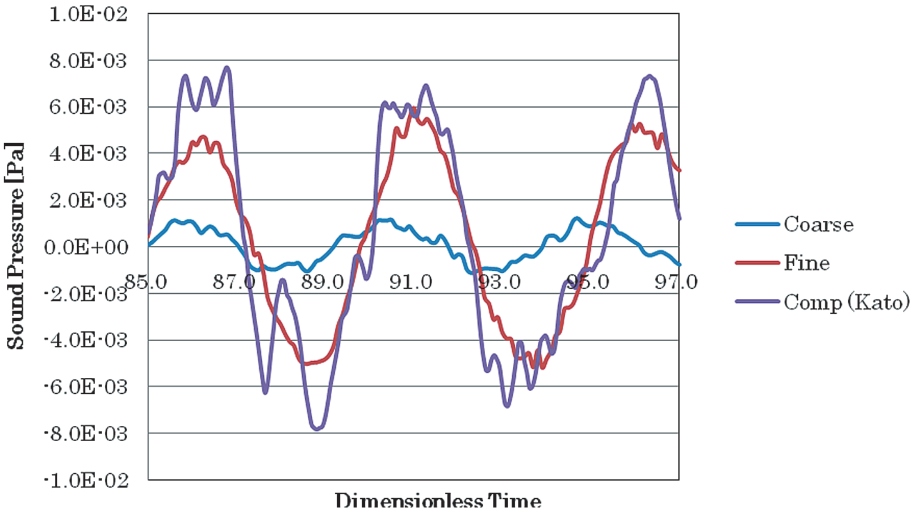


図 7　音圧の比較

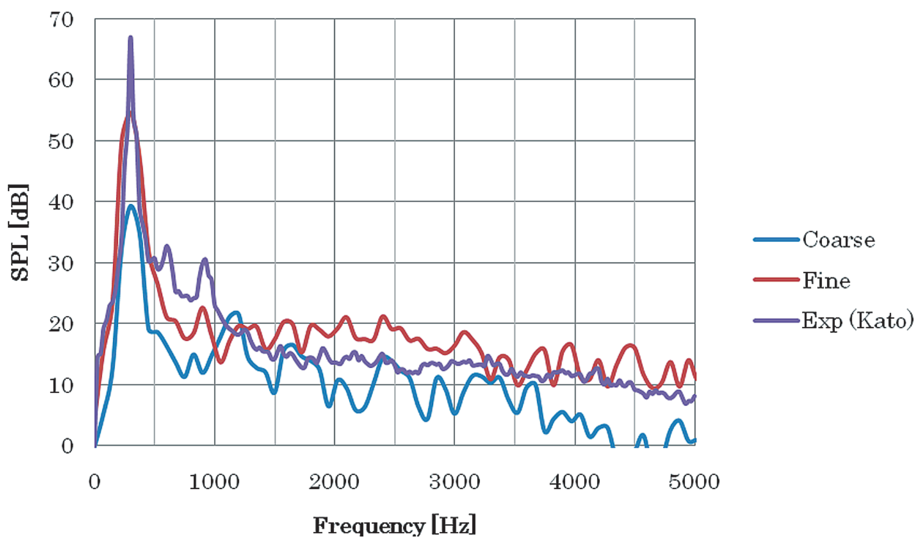


図 8　SPL の比較

4.　航空機主脚モデルまわりの解析

4. 1　解析方法の概要

計算に用いたのはタイヤのみの脚（Simple 形状）と，脚扉・サイドブレース・トルクリンク等が付属した脚（DST 形状）の 2 種類である．Reynolds 数はタイヤ直径を代表長として 1.8×10^6 である．計算空間はタイヤ直径を D として x, y 方向にそれぞれ 30D，z 方向に 15D である．境界条件は表 2 のように与えた．対流流出速度には流出面の速度の Cube 内平均値を用いている．また，脚の配置は図 9 に示すように z 軸負側に支柱，z 軸正側にタイヤが来るようにしている．z 軸負側の境界条件（支柱付け根）は実験を模擬するためにスリップ境界とした．

計算はそれぞれの脚につき Coarse，Fine の 2 種類の格子で実行した．それぞれの格子の諸元は表 3 に示す通りであり，多数の部品が存在している DST 形状の格子数は，Simple 形状の約 2 倍であり，約 1 億に達している．計算においては，Coarse, Fine ともに，まず非定常な流れ場を発達させるために無次元時間 10 だけ計算し，その後時間平均流れを得るために，Coarse は無次

元時間 50 だけ，Fine では無次元時間 10 だけ計算した．図 10 に示す Fine 形状の格子分布から分かるように，タイヤがよく再現されていることが確認できる．

表 2　境界条件

	velocity B.C.	pressure B.C.
X-	一様流	ノイマン
X+	対流流出	ディリクレ
Z-	スリップ	ノイマン
Z+	対流流出	ディリクレ
Y-	対流流出	ディリクレ
Y+	対流流出	ディリクレ

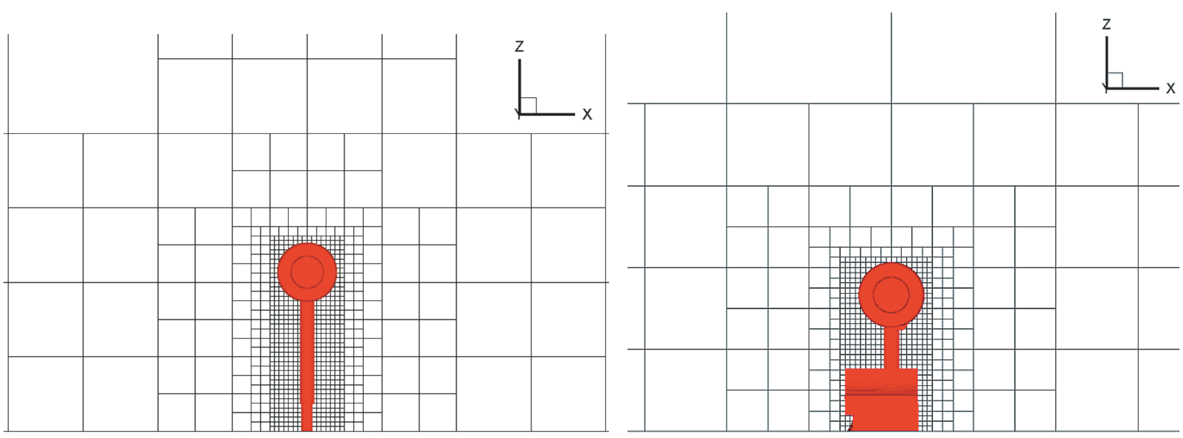
表 3 格子諸元

(a) Simple 形状

	Coarse	Fine
# of cubes	2,560	12,760
# of cells per cube	16^3	16^3
total # of cells	10,485,760	52,264,960
minimum grid spacing	9.7×10^{-3}	4.8×10^{-3}

(b) DST 形状

	Coarse	Fine
# of cubes	4489	23151
# of cells per cube	16^3	16^3
total # of cells	18,386,944	94,826,496
minimum grid spacing	9.7×10^{-3}	4.8×10^{-3}



(a) Simple形状　　(b) DST形状

図 9　y=0 断面における主脚近傍の Cube 境界

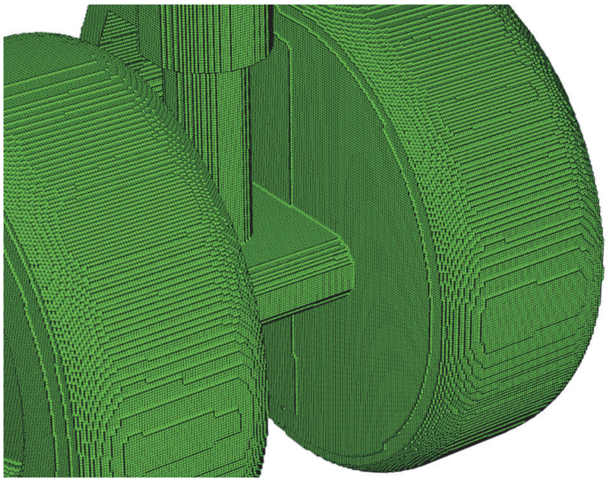


図 10　Fine 形状におけるタイヤ周りの格子

4.2　解析結果

4.2.1　タイヤ周方向圧力係数分布

Simple 形状の解析結果の時間平均を行い，図 11 に Coarse 格子と Fine 格子におけるタイヤ中心（y=0.363）における周方向圧力係数分布を示す．ここで，図 12 に示すように 0 度位置を後方淀み点，90 度位置をタイヤの設置点の反対側，180 度位置を淀み点，270 度位置をタイヤ設置点とした．この図 11 より，格子解像度が上がると Cp 最小値がより小さくなり，かつタイヤ後方での Cp 回復がより大きくなることが分かる．この結果は図 13 に示す今村らの論文[6]の計算結果（マルチブロック構造格子ソルバ UPACS による LES 計算，非構造格子ソルバ TAS による RANS 計算）や実験値と定性的に一致している．しかし，Cp の最小値が一致しておらず，これは今回用いた Fine 格子であっても境界層の格子解像度が不足していたため，剥離位置が実際よりも前方になってしまったためであると考えられる．

DST 形状においても，同じタイヤ位置における周方向の時間平均圧力係数分布を求めた．図 14 に示すように，Simple 形状と同様の傾向を示しており，Fine 格子での改善は見られるものの剥離が前方で始まっていることが分かる．

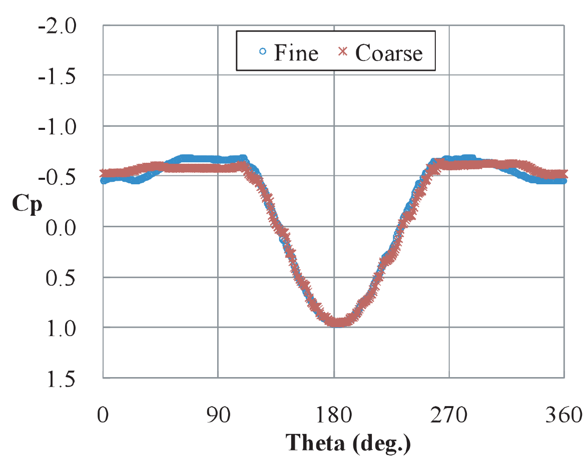


図 11　タイヤ周方向の時間平均圧力係数分布（Simple 形状）

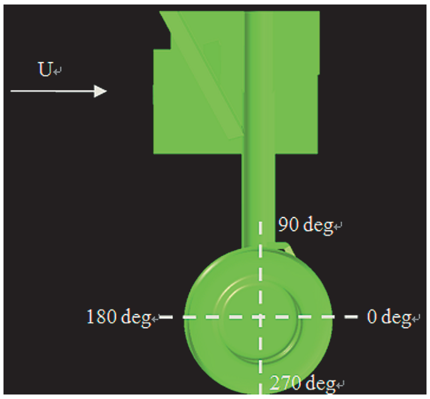


図 12　タイヤ周方向圧力係数分布の定義

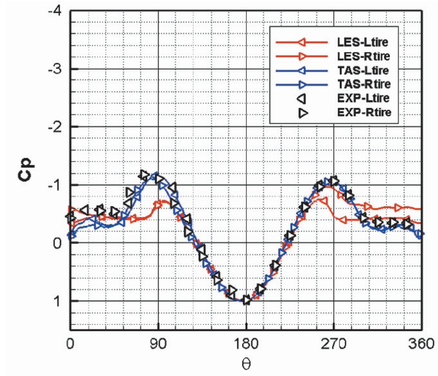


図 13　今村らによる計算及び実験における
タイヤ周方向圧力係数分布[8]

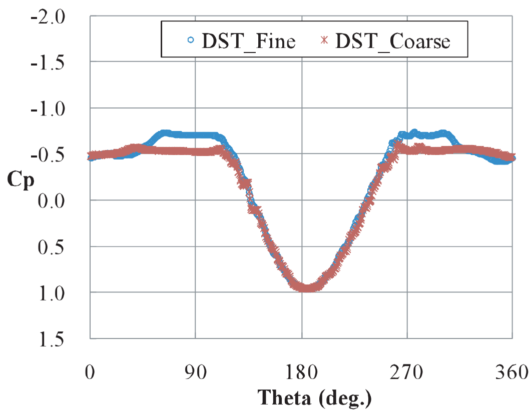


図 14　タイヤ周方向の時間平均圧力係数分布（DST 形状）

4.2.2　時間平均流線

Simple 形状と DST 形状の時間平均流線及び圧力分布を図 15，図 16 に示す．タイヤ外部の流線は大きく乱れており，流れが剥離している様子が確認できる．

脚扉の前方も全体的に流線が乱れており，角を回る部分で流れが剥離していることが確認できる．なお，DST 形状においてはタイヤ間に存在する部品の影響により，Simple 形状とは若干流れの様相が異なる．

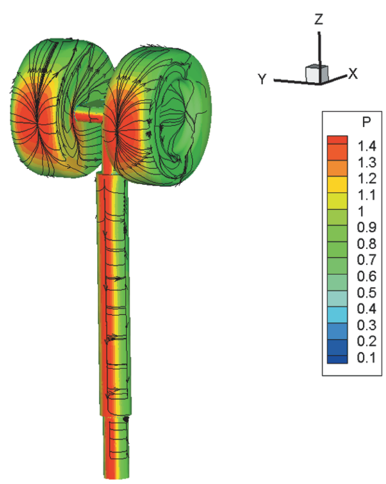


図 15　Simple 形状の時間平均流線と圧力分布

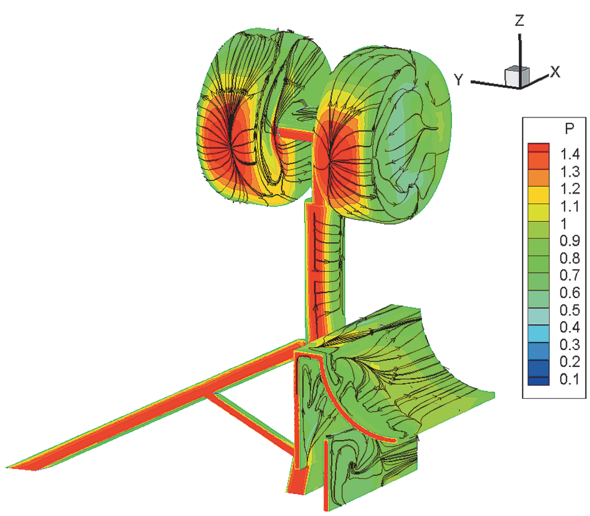


図 16　DST 形状の時間平均流線と圧力分布

4.2.3　断面速度分布

脚扉等の部品の存在するDST形状のFine格子における時間平均の地面に水平な断面上の速度分布を図17に示す．支柱付け根を0% Zmax，タイヤ先端を100% Zmaxとしてある．比較のため，図18に今村らの定常計算結果[8]である断面マッハ数分布を示す．今村らは非構造格子ソルバであるTASを用いてRANS計算を行っている．

主脚付け根近傍（Z=20%～30% Zmax位置）においては，サイドブレースと支柱の間の流れや脚扉と支柱の間に加速領域が生じており，今村らの結果と定性的に一致している．ただし，その加速は小さく，支柱や脚扉の先端から生じた剥離が影響していると考えられる．タイヤ断面位置（Z=70%～90% Zmax位置）においては，部品との干渉により位置毎に流れ場が大きく異なる．70% Zmax位置では，支柱の後流がトルクリンクに衝突している様子が確認でき，流れ場は今村らの結果と定性的に一致している．しかし，今村らの結果ほどの加速は観測されていない．90% Zmax位置ではタイヤ間の流れに関して今村らの結果と大きく異なっており，タイヤ後方で減速している様子が確認できる．これらの結果は，解像度不足によりタイヤ入口で剥離が生じ，タイヤ間に入る流れが減速してしまったことに起因すると思われる．今後，高Reynolds数の流れに対しての乱流モデル等のアプローチが必要である．

最後に，Simple形状のFine格子における時間平均の断面速度分布を図19に示す．70% Zmax位置では，支柱後流を遮るトルクリンクがないため，タイヤと支柱の間での加速が大きくなっていることが確認できる．80%や90% Zmax位置においては，流れ場はDST形状の断面速度分布とほぼ同じとなっている．

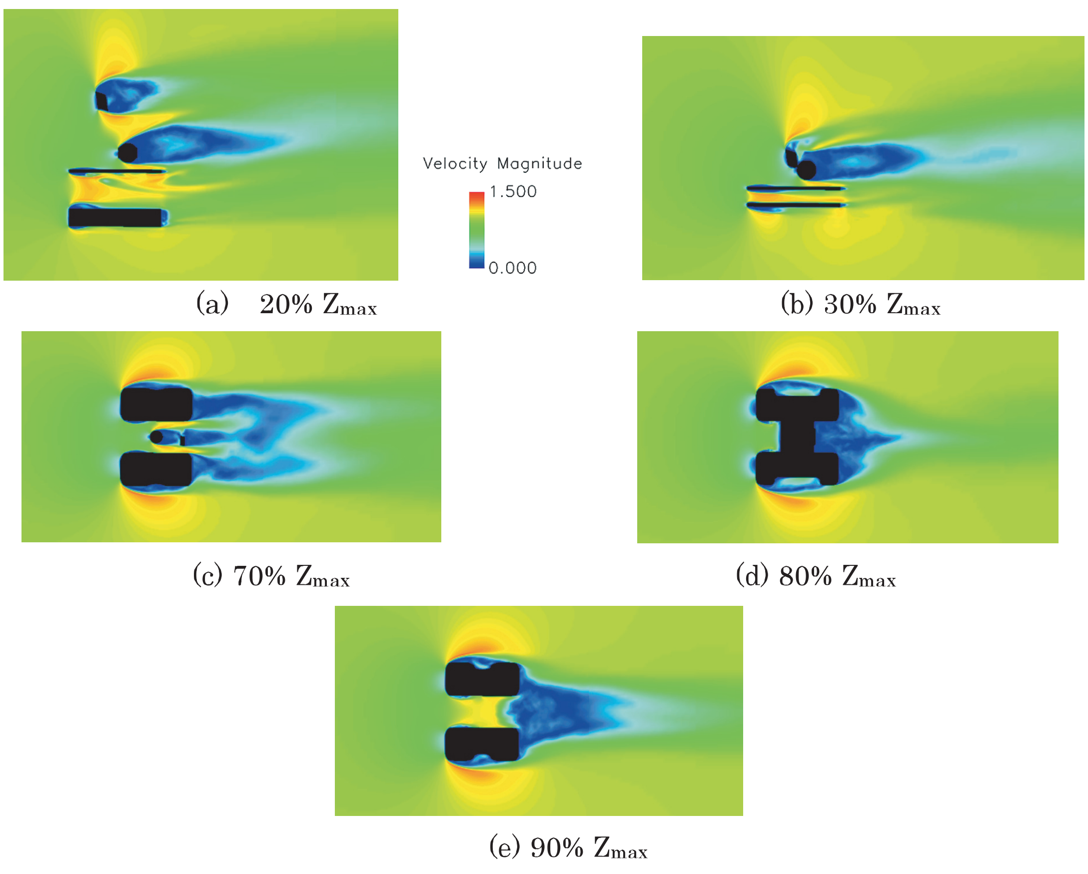


図17　DST形状の断面速度分布（Fine格子）

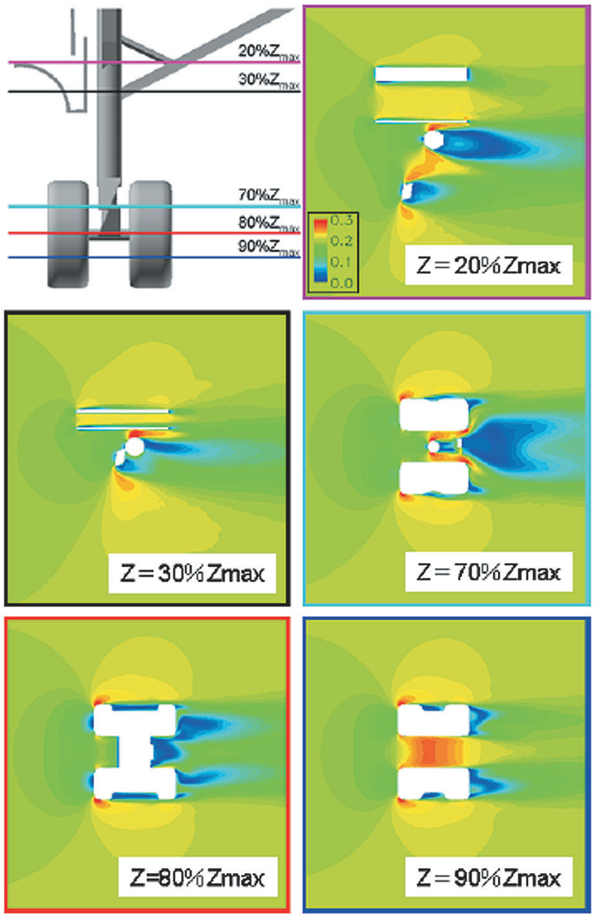
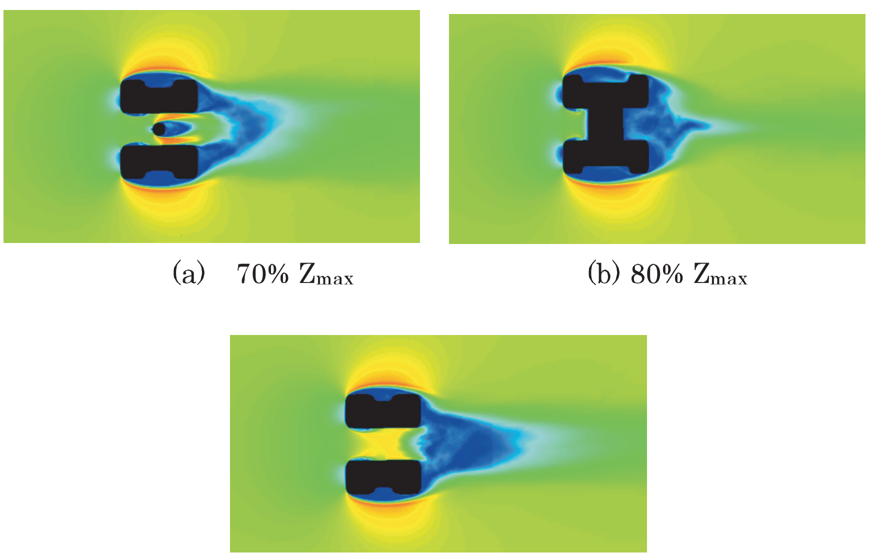


図 18　TAS による断面マッハ数分布[6]



(a)　70% Z_{max}　　(b) 80% Z_{max}

(c) 90% Z_{max}

図 19　Simple 形状の断面速度分布（Fine 格子）

4.2.4　空力音解析

Simple 形状の非定常流体解析結果を元に，Curle の式より観測点における音圧レベル SPL を推算した結果（Coarse, Fine）を図 20 に示す．図 21 には，比較のため，今村らによる TAS を用いた RANS 計算や実験の結果を示す. 本解析結果は, Coarse, Fine ともに 80～90dB で推移し，今村らの結果や実験値と SPL のオーダは一致している．ただし，Fine 格子であっても定性的には SPL 変動を捉えられているとはいえず, 空力音に関しては信頼性のある解析結果とは言えない. なお，実験データにある 600[Hz]付近のピークは，実験モデルに付属していた車軸上のピンの影響であり，計算モデルには付属していない．

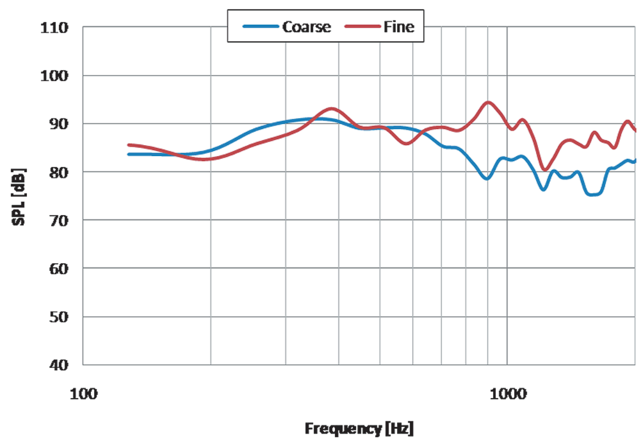


図 20　Simple 形状の SPL 値（Fine, Coarse）

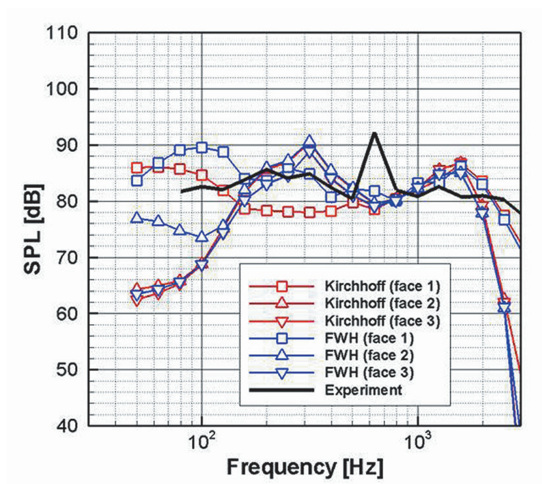


図 21　今村らによる解析と実験結果[6]

5.　結言

本稿では，BCM 非圧縮性流体解析ソルバと Curle の式による空力音推算法の開発を行い，円柱と航空機主脚モデルに適用した．円柱まわりの解析においては，空力音を定量的に推算することが可能であることを示した．一方，主脚モデルに関しては，タイヤのみの脚と脚扉やサイドブレース等各種部品の存在している脚の 2 種類について計算した．タイヤ周方向の表面圧力係数分布は，既存の結果と定性的に一致するものの，剥離が前方で始まっていることが分かった．z 軸の各断面での時間平均主流方向速度分布より，付属物が複雑な流れを誘起していることが確認できた．既存の結果と異なり脚扉の前面で剥離が生じたため，加速領域の大きさが制限され，実際の流れ場と異なっていることが分かった．

本手法では，直交格子を用いていることから複雑な主脚周りの解析を行うことは容易であり，各付属物の影響を評価する際には非常に有効である．境界層内の格子解像度不足等の課題も残されているが，今後のさらなる計算機性能の向上で Building-Cube Method の優位性が発揮出来るであろう．

謝辞

本研究は, 東北大学サイバーサイエンスセンターとの平成 21 年度共同研究課題「直交格子 CFD ソルバーの高速化に関する研究」を進める中で得られた成果であり，同センターのスーパーコンピュータを利用することで実現することができた．研究にあたっては同センター関係各位に有益なご指導とご協力をいただいた．また，本研究は科研費（21226018）の助成を受けたものである．

参考文献

[1] 今村太郎，浦弘樹，横川譲，山本一臣，“高揚力装置騒音計測用模型を用いたスラット騒音低減デバイスの研究”，第 40 回流体力学講演会／航空宇宙数値シミュレーション技術シンポジウム 2008 論文集，2008.

[2] Khorrami, M. R., Lockard, D. P., Humphreys, Jr., W. M., Choudhari, M. M., and Van de Ven, T., “Preliminary Analysis of Acoustic Measurements from the NASA-Gulfstream Airframe Noise Flight Test,” AIAA Paper 2008-2814, 14th AIAA/CEAS Aeroacoustics Conference, 2008.

[3] Neuhart, D. H., Khorrami, M. R., and Choudhari, M. M., “Aerodynamics of a Gulfstream G550 Nose Landing Gear Model,” AIAA Paper 2009-3152, 15th AIAA/CEAS Aeroacoustics Conference, 2009.

[4] Zawodny, N. S., Liu, F., Yardibi, T., Cattafesta, L., Khorrami, M. R., Neuhart, D. H., and Van de Ven, T., “A Comparative Study of a ¼-scale Gulfstream G550 Aircraft Nose Gear Model,” AIAA Paper 2009-3153, 15th AIAA/CEAS Aeroacoustics Conference, 2009.

[5] Keating, A., Dethioux, P., Satti, R., Noelting, S., Louis, J., Van de Ven, T., and Vieito, R., “Computational Aeroacoustics Validation and Analysis of a Nose Landing Gear,” AIAA Paper 2009-3154, 15th AIAA/CEAS Aeroacoustics Conference, 2009.

[6] 今村太郎，平井亨，横川譲，村山光宏，山本一臣，“主脚騒音計測用模型周り定常流れ場の数値解析”，*日本航空宇宙学会論文集*，Vol. 57, No. 671, pp. 493-498, 2009.

[7] 横川譲，今村太郎，浦弘樹，小林宙，内田洋，山本一臣，“JAXA 主脚騒音模型の遠方場特性”，第 41 回流体力学講演会／航空宇宙数値シミュレーション技術シンポジウム 2009 講演論文集，2009.

[8] 今村太郎，平井亨，雨宮和久，横川譲，榎本俊治，山本一臣，“簡素化した航空機主脚周りの非定常流体解析”，第 23 回数値流体力学シンポジウム CD-ROM 講演論文集，2009.

[9] Nakahashi, K., et al. “Some Challenges of Realistic Flow Simulations by Unstructured Grid CFD,” *International Journal for Numerical Method in Fluids*, Vol. 43, pp. 769-783, 2003.

[10] Nakahashi, K., “High-Density Mesh Flow Computations with Pre-/Post-Data Compressions,” AIAA Paper 2005-4876, 2005.

[11] Nakahashi, K., Kitoh, A. and Sakurai Y., “Three-Dimensional Flow Computations around an Airfoil by Building-Cube Method,” AIAA Paper 2006-1104, 2006.

[12] Takahashi, S., “Study of Large Scale Simulation for Unsteady Flows,” PhD dissertation, Department of Aerospace Engineering, Tohoku University, 2009.

[13] Ishida, T., Takahashi, S. and Nakahashi, K., “Efficient and Robust Cartesian Mesh Generation for Building-Cube Method,” *Journal of Computational Science and Technology*, Vol. 2, No. 4, pp. 435-446, 2008.

[14] Lighthill, M. J., “On Sound Generated Aerodynamically”, *Proceedings of Royal Society of London*, Series. A 211, pp. 564-587, 1952.

[15] Curle, N., “The Influence of Solid Boundaries upon Aerodynamics Sound”, *Proceedings of Royal Society of London*, Series. A 231, pp. 505-514, 1955.

[16] 加藤千幸，高野靖，飯田明由，藤田肇，池川昌弘，“LES による流体音の予測：第 1 報，二次元円柱からの放射音”，日本機械学会論文集 B, 60-569, 1994.

白

[共同研究成果]

第一原理に基づく大規模電子状態・量子輸送特性計算プログラムの開発

[1] 大阪大学大学院工学研究科，[2] 長崎大学先端計算研究センター
[1] 小野　倫也，[2] 江上　喜幸

量子力学の第一原理に基づいてナノ構造の量子輸送特性を計算する方法を開発している．従来のグリーン関数を用いて輸送特性を計算する方法は，グリーン関数を計算するためにほとんどの計算時間を費やしていた．本研究では，グリーン関数を計算することなく散乱の波動関数を計算する方法を開発し，計算速度が大幅に向上した．

1　はじめに

電子デバイスの微細化と高機能化が進むにつれ，ナノ構造体の量子輸送特性に対する興味が高まってきている．ナノ構造体のバリスティック伝導現象についての実験的証拠は，走査型トンネル顕微鏡やブレークジャンクション，リソグラフィーといった技術を用いて電極間にナノスケールの間隙を作成し，その間に挟まれた原子鎖や分子を流れる電流を測定することによって得られている．金属原子鎖系では，電極同士の接触点を引き伸ばしていくと電極間に数原子からなる原子鎖が形成される．この原子鎖を流れる電流は量子効果を受け，コンダクタンスが $G_0(=2e^2h{:}e$ は電気素量，h はプランク定数) の整数倍に量子化されることが金の原子鎖について最初に報告された．その後，多くの金属でこのようなコンダクタンスの量子化が確認された．また，分子系に関しても，ベンゼンチオールやDNA分子，自己組織化単分子膜1層のコンダクタンスの測定結果なども報告されている．分子系についての結果には，まだ検討の余地があると思われるが，新たな機能を持つ材料の可能性として興味深い．そこで，本研究では独自に開発した第一原理に基づいてナノ構造体の量子輸送特性を求めるOverbridging boundary-matching(OBM) 法 [1, 2] を開発した．この方法は，実空間手法 [3, 4] に基づいており，次世代スパコンなどを用いた超並列計算に適している．しかし，ナノ構造領域のグリーン関数を計算するために多くの計算時間を費やしており，改良が望まれていた．2008 年度の共同研究では，グリーン関数を計算することなく散乱の波動関数が満たす連立方程式を解くことで，計算量を大幅に削減することに成功した [5]．本稿では，ジェリウム電極を用いた場合において，さらなる高速化を可能にする方法 [6] について紹介する．ここでは，主に改良点のみに絞って説明するので，OBM 法の詳細については，参考文献 [1, 2] を参照されたい．

2　計算手法

ナノ構造体の伝導特性を計算するには，図1に示すようなモデルを用いる．電極面に平行な x, y 方向には周期境界条件を，電子の流れる z 方向は結晶が半無限に続く境界条件を課す．また，電子は左側電極から右側電極に向かって流れるものとする．散乱領域は，等間隔 $h_\mu = L_\mu/N_\mu$ のグリッドで分割されている．ここで，L_μ は散乱領域の長さ，N_μ は μ 方向のグリッド数 (μ は，x, y, または z) である．Kohn-Sham 方程式 [7] の二階微分項は，

$$\frac{\partial^2}{\partial \boldsymbol{r}^2}\psi(\boldsymbol{r}) \approx \sum_{n=-N_f}^{N_f}\Big\{c_{x,n}\psi(x_i+nh_x,y_j,z_k)+c_{y,n}\psi(x_i;y_j+nh_y,z_k)+c_{z,n}\psi(x_i,y_j,z_k+nh_z)\Big\}, \tag{1}$$

のように表される．ここで，N_f は二階微分を近似する差分のオーダーであり，係数は参考文献[8] による．グリッドにより離散化されたKohn-Sham 方程式は，次のように行列表示ができる．

$$-B_z^{\dagger}\Psi(z_{k-1}) + \left[E - H(z_k;\boldsymbol{\kappa}_{//})\right]\Psi(z_k) - B_z\Psi(z_{k+1}) = 0 \quad , \qquad (k = -\infty, ..., -1, 0, 1, ..., \infty) \tag{2}$$

ここで，$B_z = -\frac{1}{2h_z^2}I$ であり，I は N_{xy} 次元 $(N_{xy} = N_x \times N_y)$ の単位行列である．$H(z_k;\boldsymbol{\kappa}_{//})$ は，$z = z_k$ における x-y 面上のポテンシャルを含む N_{xy} 次元のブロック三重対角行列であり，$\boldsymbol{\kappa}_{//} = (\kappa_x, \kappa_y)$ は，電極面に平行な第一ブリルアン領域内のブロッホ波数である．また，$\Psi(z_k)$ は，$z = z_k$ における x-y 面上の波動関数 $\{\psi(x_i, y_j, z_k) : i = 1, \cdots, N_x;\ j = 1, \cdots, N_y\}$ を集めた列ベクトルである．

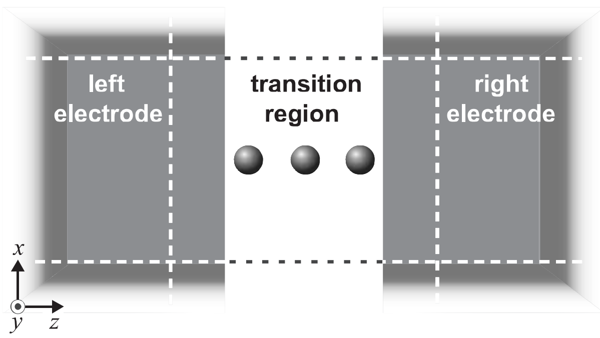


図 1: 輸送特性計算に使うモデル．散乱領域は，半無限に続くジェリウム電極に挟まれている．x, y 方向には周期境界条件を課している．破線は，Kohn-Sham 有効ポテンシャルを計算するときに用いるスーパーセルを表す．影付きの部分はジェリウム電極である．参考文献6 より転載．

簡単のため，以下では中心差分 $(N_f = 1)$ を用いるが，高次差分近似への拡張も容易である．行列表示を用いると，散乱領域 $(z_0 \le z \le z_{N_z+1})$ では (2) 式を次のように書き換えることができる．

$$\left[E - \hat{H}_T\right]\begin{bmatrix}\Psi(z_0)\\ \Psi(z_1)\\ \vdots\\ \Psi(z_{N_z})\\ \Psi(z_{N_z+1})\end{bmatrix} = \begin{bmatrix}B_z^{\dagger}\Psi(z_{-1})\\ 0\\ \vdots\\ 0\\ B_z\Psi(z_{N_z+2})\end{bmatrix}, \tag{3}$$

ここで，E はKohn-Sham エネルギー，$\hat{H}_T$ は散乱領域のハミルトニアンを $z = z_0$ と $z = z_{N_z+1}$ の面で切り出したもので，$N_x \times N_y \times (N_z + 2)$ 次元の三重対角行列である．具体的に $\hat{H}_T$ は次のように表される．

$$\hat{H}_T = \begin{bmatrix} H(z_0;\boldsymbol{\kappa}_{//}) & B_z & & & 0\\ B_z^{\dagger} & H(z_1;\boldsymbol{\kappa}_{//}) & B_z & & \\ & \ddots & \ddots & \ddots & \\ & & B_z^{\dagger} & H(z_{N_z};\boldsymbol{\kappa}_{//}) & B_z\\ 0 & & & B_z^{\dagger} & H(z_{N_z+1};\boldsymbol{\kappa}_{//}) \end{bmatrix}. \tag{4}$$

以下，標記の複雑さを避けるため，$\hat{H}_T$ のブロッホ波数を省略する．オリジナルの OBM 法では，散乱波を $[E-\hat{H}_T]$ の逆行列であるグリーン関数 $\hat{\mathcal{G}}_T(E)$ を用いて求めていた．$\hat{\mathcal{G}}_T(E)$ の (i,j) 番目のブロック行列要素を $\mathcal{G}_{i,j}$ と表すと，(3) 式は次のように表される．

$$\begin{bmatrix} \Psi(z_0) \\ \Psi(z_1) \\ \vdots \\ \Psi(z_{N_z+1}) \end{bmatrix} = \begin{bmatrix} B_z^\dagger \mathcal{G}_{0,0}\Psi(z_{-1}) & + & B_z\mathcal{G}_{0,N_z+1}\Psi(z_{N_z+2}) \\ B_z^\dagger \mathcal{G}_{1,0}\Psi(z_{-1}) & + & B_z\mathcal{G}_{1,N_z+1}\Psi(z_{N_z+2}) \\ & \vdots & \\ B_z^\dagger \mathcal{G}_{N_z+1,0}\Psi(z_{-1}) & + & B_z\mathcal{G}_{N_z+1,N_z+1}\Psi(z_{N_z+2}) \end{bmatrix}. \tag{5}$$

散乱波の計算には，すべてのグリーン関数の要素が必要なわけではなく，0 番目と (N_z+1) 番目のブロック行列 $[\mathcal{G}_{i,0}$ と $\mathcal{G}_{i,N_z+1}$ $(i=0,1,\cdots,N_z+1)]$ があれば十分である．しかし，共役勾配法を用いてグリーン関数を計算するには，$O(N_{xy}^2 \times N_z)$ の計算量が必要であるため，OBM 法を大規模計算に適用するには困難を伴う．

一方，Kong[9] によって提案された改良型 OBM 法では，各入射波 $\Phi_L^{in}(z_k)$ ごとに下の連立方程式を解くことによって散乱波を求める．

$$\left[E-\hat{H}_T-\tilde{H}\right]\begin{bmatrix} \Psi(z_0) \\ \Psi(z_1) \\ \vdots \\ \Psi(z_{N_z+1}) \end{bmatrix} = \begin{bmatrix} B_z^\dagger \Phi_L^{in}(z_{-1}) - \Sigma_L^r(z_0)\Phi_L^{in}(z_0) \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{6}$$

(6) 式は，下の散乱の境界条件を (3) 式に代入することによって得られる．

$$\Psi(z_k) = \begin{cases} \Phi_L^{in}(z_k) + \displaystyle\sum_{i=1}^{N_{xy}} r_i \Phi_i^{ref}(z_k) & (k \le 0) \\ \displaystyle\sum_{i=1}^{N_{xy}} t_i \Phi_i^{tra}(z_k) & (k \ge N_z+1) \end{cases}. \tag{7}$$

ここで，r_i は反射係数，t_i は透過係数，$\Phi_i^{ref}(z_k)$ は透過波に対する半無限に続く電極の一般化ブロッホ波，$\Phi_i^{tra}(z_k)$ は反射波に対する一般化ブロッホ波，

$$\tilde{H} = \begin{bmatrix} \Sigma_L^r(z_0) & 0 & \cdots & \cdots & 0 \\ 0 & 0 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 0 & 0 \\ 0 & \cdots & \cdots & 0 & \Sigma_R^r(z_{N_z+1}) \end{bmatrix}. \tag{8}$$

である．また，Σ_L^r は左側電極の自己エネルギー，Σ_R^r は右側電極の自己エネルギーである．これらは，次のように与えられる．

$$\Sigma_L^r(z_0) = B_z^\dagger Q^{ref}(z_{-1})[Q^{ref}(z_0)]^{-1}, \tag{9}$$

$$\Sigma_R^r(z_{N_z+1}) = B_z Q^{tra}(z_{N_z+2})[Q^{tra}(z_{N_z+1})]^{-1}. \tag{10}$$

ここで，$Q^A(z_k)(A = ref$ または $tra)$ は，$\{\Phi_i^A(z_k)\}$ からなる N_{xy} 次元の行列，すなわち $Q^A(z_k) = [\Phi_1^A(z_k), \cdots, \Phi_{N_{xy}}^A(z_k)]$ である．半無限電極の一般化ブロッホ波は，波数が実数の進行波と複素数のエヴァネッセント波から構成される．エヴァネッセント波は，指数関数的に振る舞い電極深部では急速に 0 に近づくので，入射波 Φ_L^{in} としては，右向き進行波のみを考えればよい．右側電極から電子が入射する場合も同様に考えればよい．

自己エネルギーは N_{xy} 次元の行列であるので，(6) 式を解く場合，$\Sigma_L^r(z_0)\times\Psi(z_0)$ と $\Sigma_R^r(z_{N_z+1})\times\Psi(z_{N_z+1})$ の計算コストが最も高く，計算のオーダーは最大で $O(N_{in}\times N_{xy}^2)$ である．ここで，N_{in} は入射波の数である．$[E-\hat{H}_T-\tilde{H}]$ はエルミット行列ではないが，N_{in} が N_z に比べて十分に小さいので，オリジナルの OBM 法に比べて計算コストを節約することができる．一方，N_{in} が大きくなると，一般的に非エルミット行列に対する共役勾配法の収束が良くないので，この方法はオリジナルの OBM 法に比べて計算時間がかかる．それゆえ，この方法は N_{in} を増加させる価電子の多い電極を含む場合やスーパーセルの x, y 方向の長さが長い場合には不利である．

我々は，ジェリウム電極の場合に限って (6) 式を高速に解く方法を開発した．ジェリウム電極は，結晶電極に比べ精度は劣るが短時間で計算でき，定性的に正しい解を与えるため [10–16]，大規模モデルを用いたシミュレーションに使われる．ジェリウム電極の自己エネルギー行列は z_k に無依存で，左側電極も右側電極も同じ $(\Sigma_L^r(z_k) = \Sigma_R^r(z_k) \equiv \Sigma^r)$ である．したがって，中心差分近似 $(N_f = 1)$ の場合における自己エネルギー行列 Σ^r の x-y 面上のグリッド点 $(\boldsymbol{\eta}_{//,\ell}, \boldsymbol{\eta}_{//,\ell'})$ における要素は，

$$\Sigma_{\ell,\ell'}^r = -\frac{1}{2h_z^2 N_{xy}} \sum_{\boldsymbol{G}_{//,\nu}} \exp\left[i(\boldsymbol{G}_{//,\nu}+\boldsymbol{\kappa}_{//})\cdot(\boldsymbol{r}_{//,\ell}-\boldsymbol{r}_{//,\ell'})\right]\cdot\exp(ik_{z,\nu}h_z), \tag{11}$$

である．ここで，$\boldsymbol{G}_{//,\nu} = (G_{\nu_x}, G_{\nu_y})$ は，ν_x と ν_y を整数として x-y 方向の波数ベクトル $\left(\frac{2\pi}{L_x}\nu_x, \frac{2\pi}{L_y}\nu_y\right)$，$k_{z,\nu}$ はブロッホ波の z 成分である．$k_{z,\nu}$ は，次のように求められる．

$$k_{z,\nu} = \begin{cases} \dfrac{\cos^{-1}(-\chi_\nu)}{h_z} & \text{進行波} \quad (|\chi_\nu| < 1) \\ \\ \dfrac{i\cosh^{-1}(-\chi_\nu)}{h_z} & \text{エヴァネッセント波} \quad (|\chi_\nu| > 1) \end{cases} \tag{12}$$

ここで，

$$\chi_\nu = h_z^2 E - \frac{h_z^2}{h_x^2}(1-\cos(G_{\nu_x}+\kappa_x)h_x) - \frac{h_z^2}{h_y^2}(1-\cos(G_{\nu_y}+\kappa_y)h_y) - 1. \tag{13}$$

である．(12) 式の波動関数の成分 $\cos^{-1}(z)$ は $k_{z,\nu}$ が正数になるように，$\cosh^{-1}(z)$ は $-ik_{z,\nu}$ が正数になるように決める．(11) 式と (13) 式の導出は，参考文献 [6] を参考にされたい．

行列 Σ^r とベクトル $\Psi(z_k)$ の積 $\sum_{\ell'}\Sigma_{\ell,\ell'}^r\psi(\boldsymbol{\eta}_{//,\ell'}, z_k)$ は，2 次元の波数空間で計算すると計算量が最も少なくなる．フーリエ変換された自己エネルギー行列 $\tilde{\Sigma}_{\nu,\nu'}^r$ の要素は，

$$\tilde{\Sigma}_{\nu,\nu'}^r = \frac{1}{N_{xy}}\sum_{\boldsymbol{\eta}_{//,\ell}}\sum_{\boldsymbol{\eta}_{//,\ell'}} \exp\left[-i(\boldsymbol{G}_{//,\nu}+\boldsymbol{\kappa}_{//})\cdot\boldsymbol{\eta}_{//,\ell}\right]\exp\left[i(\boldsymbol{G}_{//,\nu'}+\boldsymbol{\kappa}_{//})\cdot\boldsymbol{\eta}_{//,\ell'}\right]\Sigma_{\ell,\ell'}^r. \tag{14}$$

と表される，平面波の直交性を利用すれば，下記のような対角化された自己エネルギー行列が得られる．

$$\tilde{\Sigma}^r_{\nu,\nu'} \quad = \quad -\frac{1}{2h_z^2}\delta_{\nu\nu'}\exp(ik_{z,\nu}h_z). \tag{15}$$

列ベクトル $\tilde{\Psi}(z_k)$ は，N_{xy} 個のフーリエ変換された波動関数 $\Psi(z_k)$

$$\tilde{\psi}_\nu(z_k) \quad = \quad \frac{1}{N_{xy}}\sum_{\boldsymbol{\eta}_{//,\ell}}\exp\left[-i(\boldsymbol{G}_{//,\nu}+\boldsymbol{\kappa}_{//})\cdot\boldsymbol{\eta}_{//,\ell}\right]\psi(\boldsymbol{\eta}_{//,\ell},z_k). \tag{16}$$

で表される．グリッド数と波数ベクトルの数は一致し，フーリエ変換された自己エネルギー行列 $\tilde{\Sigma}^r$ の非対角要素は 0 であるので，$\tilde{\Sigma}^r$ と $\tilde{\Psi}(z_k)$ の積に要する計算量のオーダーは，$O(N_{in}\times N_{xy})$ になる．$\Psi(z_k)$ のフーリエ変換と $\tilde{\Sigma}^r\times\tilde{\Psi}(z_k)$ の逆フーリエ変換に必要な計算コストは，高速フーリエ変換を用いた場合，$O(N_{in}\times N_{xy}\log N_{xy})$ まで減らすことができる．したがって，最も計算コストがかかっていた部分の計算量が $O(N_{in}\times N_{xy}^2)$ から $O(N_{in}\times N_{xy}\log N_{xy})$ に削減できた．

3　パフォーマンステスト

開発した改良型 OBM 法のパフォーマンスを確認するため，ナトリウム単原子鎖の電子輸送特性計算を行った (図 1 参照)．原子鎖は直接ジェリウム電極に接続され，原子鎖を構成する原子の数 N_{atom} を 3 個から 8 個まで変化させた．原子間距離 d は，$\sqrt{3}a_0/2$ とした．ここで，a_0 はナトリウム結晶の格子定数 (=8.11 bohr) である．原子鎖端の原子とジェリウム電極表面の距離は，[110] 原子鎖をモデル化して $\sqrt{2}a_0/4$ とした．ジェリウム電極のウィグナーザイツ半径は，$r_s = 3.99$ bohr とした．交換相関相互作用は局所密度近似 [19] を用い，原子核と電子の相互作用は Troullier-Martins 型の擬ポテンシャル [20, 21] で近似した．Kohn-Sham 方程式の有効ポテンシャルを決めるために，通常の z 方向にも周期的な境界条件を課したスーパーセル (図 1 長方形に対応する) を用いた．スーパーセルのサイズは，x, y 方向に $L_{x(y)} = 25.66$ bohr，z 方向に $L_z = (N_{atom}-1)\cdot d + 42.12$ bohr とした．x, y 方向のグリッド数は $N_{x(y)} = 36$，z 方向のグリッド数は $N_z = 60 + 10\times(N_{atom}-1)$ とした．この時，フェルミ準位における入射波の数は 13 である．また，$\Sigma^r\times\Psi(z_{0(N_z+1)})$ を計算する際に実空間で行うアルゴリズムと波数空間で行うアルゴリズムの計算コストの比 $\frac{N_{in}\times N_{xy}\log N_{xy}}{N_{in}\times N_{xy}^2}$ は，0.0024 である．温度ゼロ，バイアスゼロの極限におけるコンダクタンスは Landauer-Büttiker formula[22] の公式で決定した．テスト計算は，(a) オリジナルの OBM 法，(b) 改良型 OBM 法，(c) (b) の積を波数空間で行ったものの 3 通りを行い，計算には 3.25 GHz Intel® Xeon® processor を搭載した計算機を用いた．

図 2 に，原子鎖を構成する原子数に対する計算時間を示す．明らかに (c) の場合の計算時間が短く，開発された方法の有用性が分かる．

4　イリジウム原子鎖の電子輸送特性

2003 年に Smit ら [23] の実験によって，金以外にも白金やイリジウム原子鎖でもコンダクタンスが構成原子数によって偶奇振動することが発見された．後に de la Vega ら [24] が，タイトバインディング法を用いてこの現象を解析し，白金やイリジウムのコンダクタンスの振動周期が，Smit らの結果よりも長い周期で振動することを報告した．この長周期振動の振幅は，偶奇振動の振幅

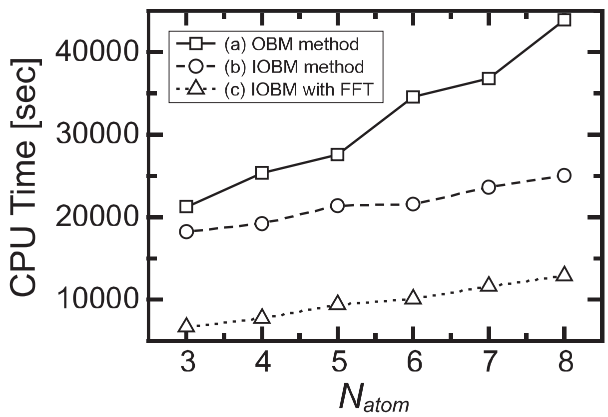


図 2: ナトリウム単原子鎖の構成原子数 N_{atom} に対する計算時間．(a) オリジナルの OBM 法，(b) 改良型 OBM 法，(c) (b) に対して波数空間での積分に置換したもの．参考文献 6 より転載．

よりも大きい．直感的に考えれば，実験観測において振幅の大きい振動の方が支配的と考えられるが，de la Vega らが発見したようなコンダクタンスの振動は，これまで実験的に観測されていない．2009 年に我々がオリジナルの OBM 法を用いて白金原子鎖に対して調べたところ，これらの長周期の振動は原子鎖の構造変化に非常に敏感であることを発見した．一方，偶奇振動のような短周期の振動は構造変化に影響されないため，実験が行われているような環境下では偶奇振動のみが観測されることを報告した [14]．イリジウム原子鎖については，まだ調べられていないが，同様のことが期待される．イリジウムは価電子数が多く入射波の数が多いので，改良型 OBM 法のテストモデルとしても重要なアプリケーションである．このような理由で，ここではイリジウム原子鎖の輸送特性を調べた．また，比較のため，金原子鎖の輸送特性についても調べた．

原子鎖の原子間距離 d は，$a_f/\sqrt{2}$ とした．ここで，a_f は面心立方晶の格子定数で，イリジウムの場合 $a_f = 7.25$ bohr，金の場合は 7.71 bohr である．グリッド幅は $d/16$ とし，原子鎖を構成する原子数 N_{atom} を 3 から 8 まで変化させた．原子鎖端の原子からジェリウム表面までの距離は $2a_f/\sqrt{6}$ とした．ジェリウム電極のウィグナーザイツ半径 r_s は，原子鎖を構成する元素と同じにするため，イリジウムの場合 1.36 bohr，金の場合 1.35 bohr とした．散乱領域の大きさは，x, y 方向にイリジウムの場合は $L_x = L_y = 15.07$ bohr，金の場合は $L_x = L_y = 15.38$ bohr とした．また，z 方向については，$L_z = (N_{atom} + 4) \cdot d$ とした．グリッド数は，x, y 方向にイリジウムの場合は $N_{x(y)} = 46$，金の場合は $N_{x(y)} = 48$ とし，z 方向には $N_z = 16 \times (N_{atom} + 4)$ とした．入射波の数 N_{in} は，どちらの場合も 37 であった．波数空間で積分をすることによる計算量の減少比 $\frac{N_{in} \times N_{xy} \log(N_{xy})}{N_{in} \times (N_{xy})^2}$ は，イリジウムの場合が 0.0016，金の場合が 0.0015 であり，波数空間での積分が計算量の向上に大きく貢献していることが分かる．計算には，東北大学サイバーサイエンスセンターの SX-9 を用いた．その他の計算条件および手順は，ナトリウムの場合と同様である．

図 3 に，原子数に対するコンダクタンスの変化を示す．金原子鎖の場合はコンダクタンスが偶奇振動を示しているものの，イリジウム原子鎖の場合は偶奇振動が観察されない．さらに詳細に解析するため，各透過チャネルの原子数に対する変化を表 1 に示す．この表において，各チャネルの量子数は，原子鎖を構成する原子の原子軌道の方位量子数に対応する．この表より，upper s-d_{z^2} チャネルは偶奇振動を示しているものの，他の d チャンネルは長周期で振幅の大きい振動をしていることが分かる．以前の我々の白金原子鎖に関する研究 [14] より，upper s-d_{z^2} チャネルは

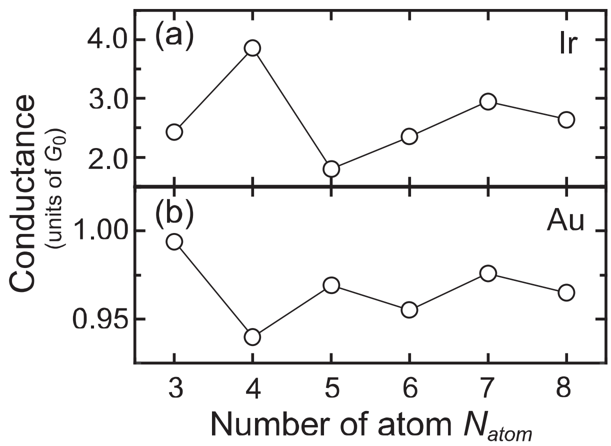


図 3: (a) イリジウム原子鎖と (b) 金原子鎖の構成原子数 N_{atom} に対するコンダクタンスの変化．参考文献 6 より転載．

表 1: イリジウム原子鎖のチャネル透過率．参考文献 6 より転載．

n	upper s-d_{z^2}	lower s-d_{z^2}	$d_{xz(yz)}$	$d_{x^2-y^2(xy)}$
3	0.854	0.016	0.763	0.016
4	0.973	0.011	0.935	0.497
5	0.727	0.018	0.510	0.019
6	1.000	0.020	0.656	0.010
7	0.838	0.056	0.979	0.048
8	0.975	0.116	0.569	0.206

原子構造の変化に対してほとんど無依存で，他の d チャネルは構造変化に対して敏感に変化することが分かっている．これらの結果より，Smit ら [23] の実験で観察されているイリジウム原子鎖のコンダクタンスの偶奇振動は，upper s-d_{z^2} チャネルのものであることが結論づけられる．

謝辞

本研究は，東北大学サイバーサイエンスセンターとの平成 21 年度共同研究課題「第一原理に基づく大規模電子状態・量子輸送特性計算プログラムの開発」を進める中で得られた成果である．

参考文献

[1] Y. Fujimoto, and K. Hirose, Phys. Rev. B **67**, 195315 (2003).

[2] K. Hirose, T. Ono, Y. Fujimoto, and S. Tsukamoto, *First-Principles Calculations in Real-Space Formalism, Electronic Configurations and Transport Properties of Nanostructures* (Imperial College Press, London, 2005).

[3] J.R. Chelikowsky, N. Troullier, and Y. Saad, Phys. Rev. Lett. **72**, 1240 (1994)

[4] T. Ono and K. Hirose, Phys. Rev. Lett. **82**, 5016 (1999).

[5] 小野倫也，江上喜幸，後藤英和，東北大学サイバーサイエンスセンター広報 SENAC, 42(4), 19-25, (2009).

[6] Y. Egami, K. Hirose, and T. Ono, Phys. Rev. E **82** 056706 (2010).

[7] W. Kohn and L.J. Sham, Phys. Rev. **140**, A1133 (1965).

[8] J.R. Chelikowsky, N. Troullier, K. Wu, and Y. Saad, Phys. Rev. B **50**, 11355 (1994).

[9] L. Kong, M.L. Tiago, and J.R. Chelikowsky, Phys. Rev. B **73**, 195118 (2006).

[10] N.D. Lang, Phys. Rev. Lett. **79**, 1357 (1997).

[11] S. Tsukamoto and K. Hirose, Phys. Rev. B **66**, 161402(R) (2002).

[12] M. Okamoto, T. Uda and K. Takayanagi, Phys. Rev. B **64**, 033303 (2001).

[13] S. Furuya, Y. Gohda and S. Watanabe, Jpn. J. Appl. Phys. **45**, 8991 (2006).

[14] T. Ono, J. Phys. Chem. C **113**, 6256 (2009).

[15] N. Kobayashi, M. Brandbyge, and M. Tsukada, Phys. Rev. B **62**, 8430 (2000); Surf. Sci. **433-435**, 854 (1999).

[16] M. Otani, T. Ono, and K. Hirose, Phys. Rev. B **69**, 121408(R) (2004); T. Ono and K. Hirose, Phys. Rev. Lett. **94**, 206806 (2005); T. Ono and K. Hirose, Phys. Rev. Lett. **98**, 026804 (2007) Y. Egami, S. Aiba, K. Hirose, and T. Ono, J. Phys.: Cond. Matt. **19**, 365201 (2007).

[17] Y. Egami, T. Sasaki, S. Tsukamoto, T. Ono, K. Inagaki, and K. Hirose, Mater. Trans., JIM **45**, 1433 (2004).

[18] Y. Egami, T. Ono, and K. Hirose, Phys. Rev. B **72**, 125318 (2005).

[19] J.P. Perdew and A. Zunger, Phys. Rev. B **23**, 5048 (1981).

[20] N. Troullier and J.L. Martins, Phys. Rev. B **43**, 1993 (1991).

[21] We used the norm-conserving pseudopotentials NCPS97 constructed by K. Kobayashi. See K. Kobayashi, Comput. Mater. Sci. **14**, 72 (1999).

[22] M. Büttiker, Y. Imry, R. Landauer, and S. Pinhas, Phys. Rev. B **31**, 6207 (1985).

[23] R.H.M. Smit, C. Untiedt, G. Rubio-Bollinger, R.C. Segers, and J.M. van Ruitenbeek, Phys. Rev. Lett. **91**, 076805 (2003).

[24] L. de la Vega, A. Martín-Rodero, A. Levy Yeyati, and A. Saúl, Phys. Rev. B **70**, 113107 (2004).

[先端的大規模計算利用サービス利用成果]

熱アシスト磁気記録の記録磁化過程シミュレーション

― Recording simulation of heat assisted magnetic recording ―

水野　友人[1], Simon Greaves[2], 村岡　裕明[2]
[1] TDK株式会社　浅間テクノ工場
[2] 東北大学　電気通信研究所

本稿では、熱アシスト磁気記録における記録磁化過程について、東北大学サイバーサイエンスセンターのSX-9を用いて得られたシミュレーション結果の報告を行う.

1.　はじめに

ハードディスク装置（HDD）は、情報化社会における情報記憶の中心技術であり、その産業規模は年間数億台の生産に達している. 記憶容量の需要を背景に、垂直磁気記録の実用化を始めとする技術革新による記録面密度の向上が続いている. 現在の記録密度は～500ギガビット/inch^{2}程度であるが、これをテラビット/inch^{2}級にまで大幅に高める記録・再生ヘッドの開発が強く求められている. しかし、1ビットを記憶するためのディスク上の領域（以下ビットサイズ）は数十ｎｍ以下の極めて微細なサイズであり、実験用デバイスの試作も容易ではなく、年々計算機シミュレーションによる研究開発への要求が高まっている. 本研究では、磁気ヘッド磁界に対する磁気ディスク媒体の磁化スピンの反転挙動をLLG方程式によりモデル化し、熱アシスト磁気記録における記録磁化過程の解析を行った. 計算には東北大学電気通信研究所 村岡研究室で開発されたシミュレーションプログラムを用いた.

2.　熱アシスト磁気記録

HDDの記録密度を増加するためにはビットサイズを減少させる必要があり、具体的には12nm以下、単位長さあたりの磁束反転数（Flux Changes per inch = FCI）2000kFCI以上に相当するビットサイズの実現が今後の記録密度増加に対して不可欠となっている. 一方、ビットサイズを小さくすると磁気記録層膜厚との積である1ビットの体積が減少し、熱安定性が損なわれる. これを回避するために、磁気記録媒体の磁気異方性(Ku)を増加し、媒体保磁力(Hc)を増加させる必要があるが、記録素子の微細化に伴う記録磁界の減少により、記録困難となる. 媒体磁気異方性を高めつつ、記録磁界を増加させずに容易に記録する方法の1つとして、熱アシスト記録が検討されている[1,2]. これは、記録したいビットのHcを加熱により減少させ、低い記録磁界で安定に記録を行う一方、記録後の冷却過程で磁気異方性が回復増大し、記録された磁化情報の熱安定性を保つことができる記録方法であり、今後のHDDの記録密度増大に有望な技術として考えられている(図1).

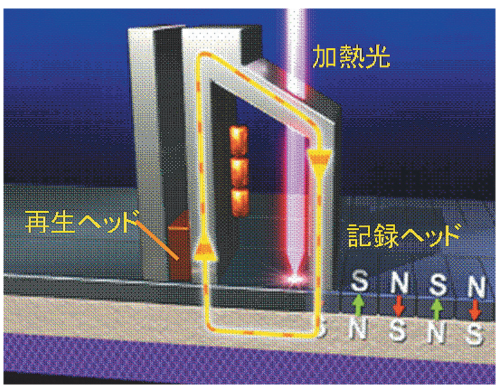


図1. 熱アシスト磁気記録のイメージ図. 磁気記録ヘッドは磁気信号読み取り再生ヘッドと、記録ヘッドから構成される. 熱アシスト磁気記録では、加熱光を導入することで記録を行う.

3.　計算手法

本節では、熱アシスト記録の物理現象を再現するために用いた計算手法について述べる．3.1 では磁気記録媒体の磁化の動力学方程式について説明し、3.2 では加熱で発生する媒体上の温度分布を得るための熱伝導方程式について説明する．最後に 3.3 において、近接場光の媒体吸収量の評価に用いた式について説明する．

3.1　マイクロマグネティクス

磁気記録媒体を構成する磁性微粒子の磁化挙動について、時間発展 t に関する磁化の動力学方程式(Landau-Lifshitz-Gilbert 方程式)に基づいた、マイクロマグネティクスモデルの計算を行う(式 1)．

$$\frac{d\vec{M}}{dt} = -\gamma\left(\vec{M}\left(\vec{H} - \frac{\alpha}{\gamma \cdot M_s}\cdot\frac{d\vec{M}}{dt}\right)\right) \quad \text{(式.1)}$$

$\vec{M}$ は磁性微粒子の磁化ベクトル(Ms は飽和磁化)、γはジャイロ磁気定数、$\vec{H}$ は印加磁界ベクトル、αは磁化ベクトルの磁界方向の緩和を表現するための無次元の摩擦係数であり、Gilbert のダンピング定数と呼ばれる．磁性微粒子は図 2 に示す Voronoi 多面体で表現し、各 Voronoi 粒子について式.1 を計算する．

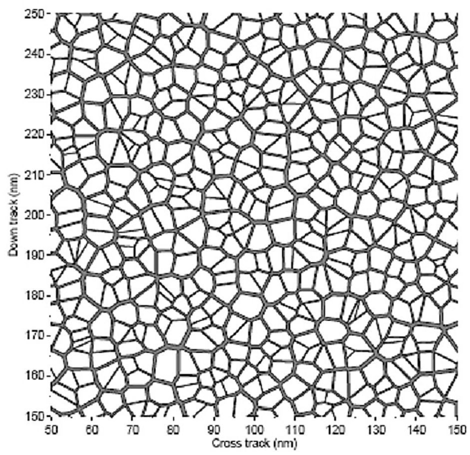


図.2 Voronoi 多面体による磁性微粒子の表現

3.2　熱伝導解析

熱アシスト磁気記録においては、磁性微粒子を加熱し、Hc を減少させた状態にヘッド磁界を印加して記録を行う．このため磁性微粒子の熱源による温度上昇を計算する必要があり、本研究では熱拡散方程式を用いて、熱源による媒体加熱をシミュレーションした．隣り合う i 番目と j 番目の磁性微粒子について、i 番目の磁性微粒子の温度の時間発展 t についての方程式を計算した(式 2)．

$$\frac{dT_i}{dt} = \frac{k}{CV}\frac{A\left(T_j - T_i\right)}{L} \quad \text{(式.2)}$$

i 番目の磁性微粒子について、T_i は温度、k は熱伝導率、C は比熱、V は体積である．A と L は、i 番目と j 番目の粒子との接触面積と中心間距離である．

3.3　熱源

媒体の加熱に用いるナノメートルオーダーの熱源として、近接場光を仮定した計算を行った．近接場光によって、通常の光ディスク光学系では不可能な回折限界以下の微小な光スポットが得られるため、高密度記録が期待できる[3]．近接場光による加熱エネルギーは、媒体による電磁場吸収により発生し、本研究では式.3 によるエネルギー吸収量 Q を、単位体積当たりの熱源として仮定した．

$$Q = \frac{1}{2}\varepsilon_0 \varepsilon_i \omega E^2 \quad (式.3)$$

ε_0 は真空の誘電率、ε_i は複素誘電率であり、屈折率と吸収係数を n と k とすると 2nk に等しい．ω は電磁場の振動数、E^2 は電界強度である．

4　熱アシスト記録シミュレーション

4.1 磁気記録媒体モデル

磁気記録媒体モデルとして、図.3 に示す 4 層から成る構成（ガラス基板/下地/中間層/磁気記録層）を検討した．磁気記録層については、10nm と 20nm の場合について記録状態の比較を行った．磁性粒子の大きさは平均値で 5nm とした．磁気記録層の磁気特性は表.1 に示す値を仮定している．温度 T における Ku と Ms は、絶対零度における Ku と Ms を仮定して、式.4 と式.5 により求めた．ここでは χ=1，n=5 を用いている(図.4)．

$$K_u(T) = \frac{\cosh(\chi) - \cosh\left(\chi\left(T/T_c\right)\right)}{\cosh(\chi) - \cosh\left(\chi\left(T/T_c\right)\right)} K_u(0) \quad (式.4)$$

$$M_s(T) = M_s(0)\left(\frac{Ku(T)}{Ku(0)}\right)^n \quad (式.5)$$

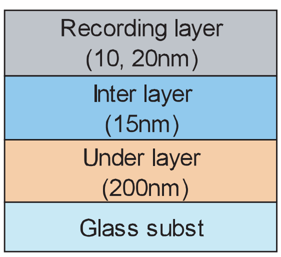


図.3 磁気記録媒体モデル

Ms (emu/cc)	Ku (erg/cc)	Hk (kOe)	Tc (K)	D (nm)
1000	2.40E+07	48	660	5

表.1 磁気記録媒体の磁気特性

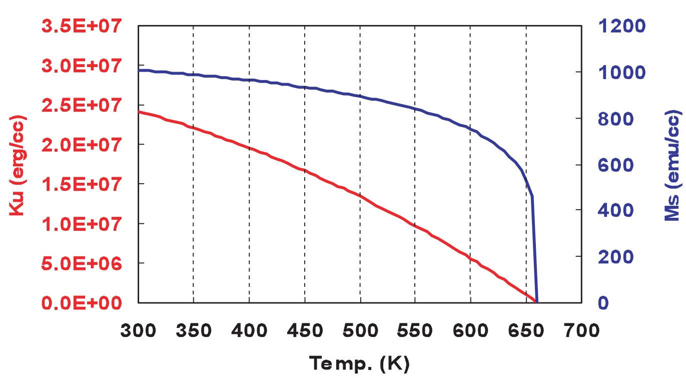


図.4 Ku と Ms の温度依存性

4.2 熱計算モデル

熱計算は、表.2に示す熱計算パラメータを仮定して計算を行った．また式.3による近接場光による発熱量の計算には、電界強度のFWHMが約20nm×20nm領域となる電界強度分布を仮定した（図.5）．（ここではガラス基板への熱伝導は考慮しない.）

Layer	比熱 (J/cc K)	熱伝導率 (W/cm K)	反射率 n	吸収係数 k
Recording	3.8	0.12	3.11	3.72
Inter layer	3.1	0.01	1.68	0.00
Under layer	3.1	0.90	2.91	3.09
Glass	1.6	0.00	1.50	0.00

表.2 磁気記録媒体の熱計算パラメータ

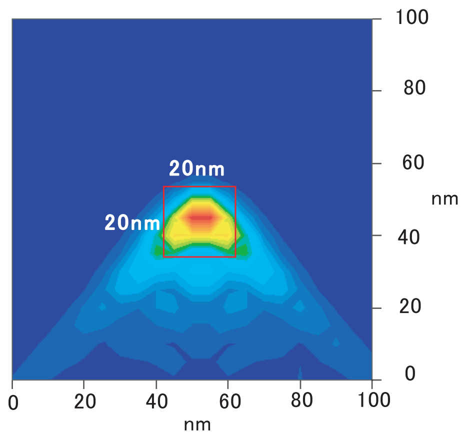


図.5 電界強度分布

4.3 記録ヘッドによる磁界強度分布

市販の磁界計算プログラムJMAG[4]を用いて、記録ヘッド形状、磁気特性および記録電流値を仮定した、Stoner-Wohlfarthモデルにおけるダウントラック方向の有効ヘッド磁界分布を求めた（図.6）．磁極端における最大有効磁界は13500(Oe)である．熱源として、磁極端から35nmの位置に、図.5の電界強度分布のピーク位置を配置した．媒体は熱源による加熱を受けた後、磁界強度が増加する方向（図.6の原点に近づく方向）に移動し、線速度24m/secを仮定した．

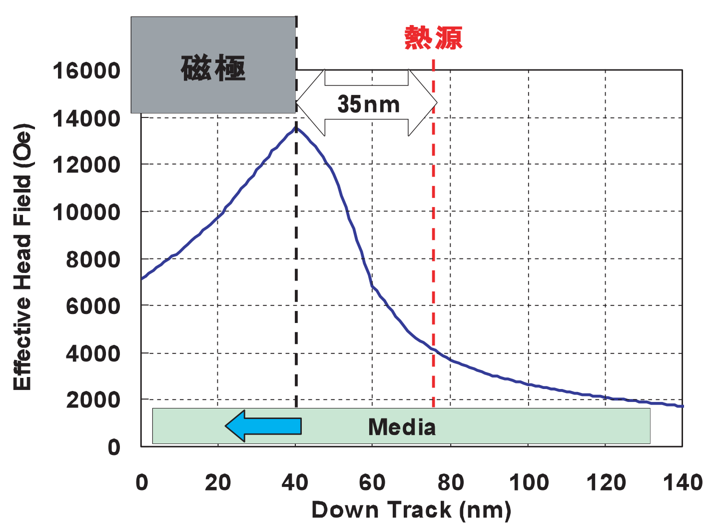


図.6 ダウントラック方向の有効ヘッド磁界分布

4.4 磁化遷移点と有効磁界勾配

図.7-1 と図.7-2 に、磁気記録層膜厚 10nm と 20nm の場合について、ダウントラック方向の有効ヘッド磁界分布、温度分布および Hc 分布を示す．最高到達温度は 800K とした．磁化遷移点はヘッド磁界分布と Hc 分布との交点となり、磁気記録層膜厚 10nm と 20nm の場合、図中の 472nm と 462nm の位置に対応する．磁気記録膜厚を 20nm にした場合に温度分布がダウントラック方向に広がり磁化遷移点が変化している．これは膜厚を 2 倍にした際､最高到達温度 800K を実現するために近接場光強度の増加が必要となった影響と考えられる．

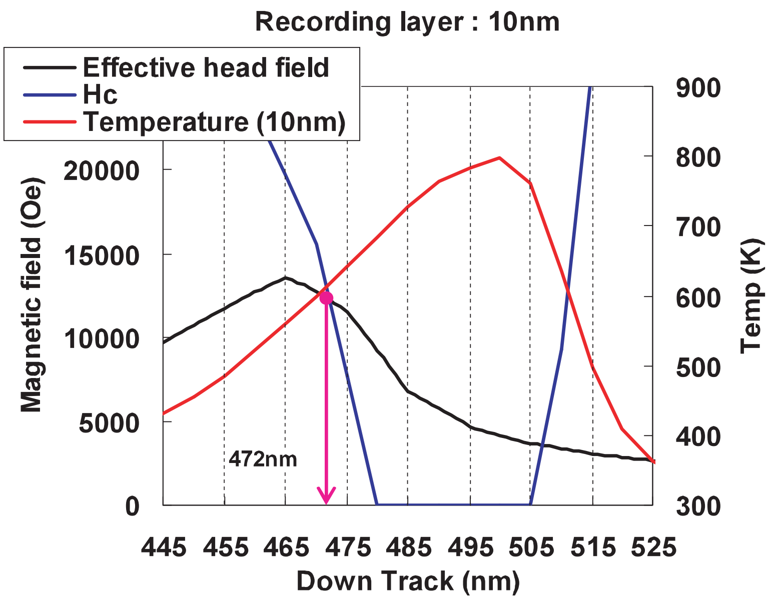


図.7-1 磁気記録層膜厚 10nm の場合の有効ヘッド磁界、Hc および温度分布

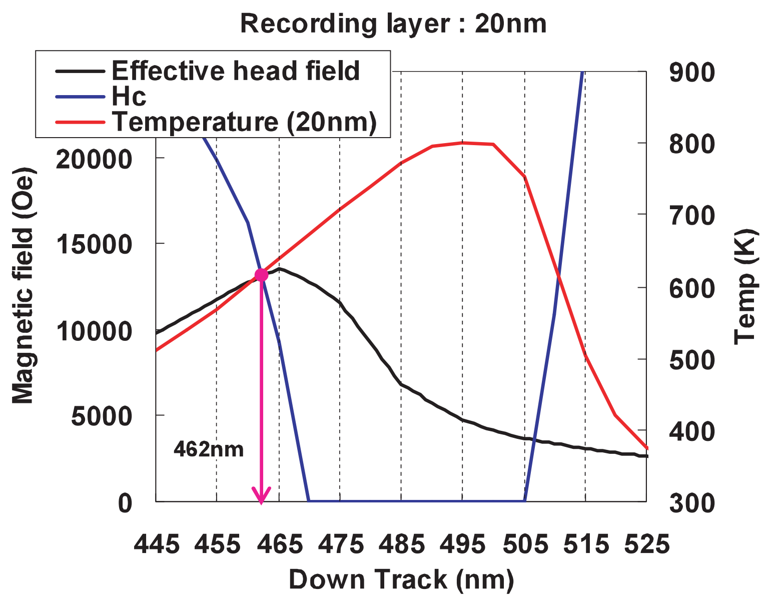


図.7-2 磁気記録層膜厚 20nm の場合の有効ヘッド磁界、Hc および温度分布

記録品質の指標となる信号-ノイズ強度比(SNR)は、磁化遷移点における有効磁界勾配が高い値であるほど大きな値になると考えられる．有効磁界勾配は式.6 で表現され[5]、磁気記録層膜厚が 10nm と 20nm の場合について、磁気記録点での有効磁界勾配を比較した(表.3)．また有効磁界勾配の計算には各温度での Hc の温度依存性が必要であり、各温度における磁化曲線の LLG シミュレーションから求めた動的 Hc(Dynamic-Hc)の温度依存性から算出した(図.8)．

$$\frac{dH_{total}}{dx} = \frac{dH_{head}}{dx} - \frac{dH_c}{dT}\frac{dT}{dx} \qquad (式.6)$$

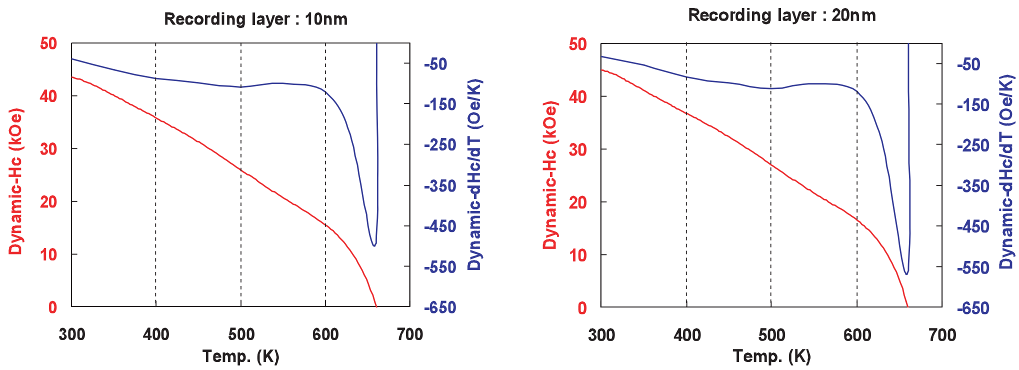


図.8 Dynamic-Hc の温度依存性

式.6 と図.7-1,7-2、および図.8 の結果から推定した有効磁界勾配は、磁気記録層膜厚 10nm と 20nm について、各々681(Oe/nm)と 1449(Oe/nm)の値を得た．また 2000kFCI における磁気記録状態を計算し、磁気記録層膜厚 10nm と 20nm 各々の場合について 7.8dB と 10.6dB の SNR 値を得た(図.9)．

磁気記録層膜厚 (nm)	磁化遷移点 (nm)	磁化遷移点の温度 (K)	ヘッド磁界 (Oe)	dH_{head}/dx (Oe/nm)	dHc/dT (Oe/K)	dT/dx (K/nm)	dHeff/dx (Oe/nm)	SNR (dB)
10	472	595	12253	-240	-113	8.2	681	7.8
20	462	620	12797	163	-185	7.0	1449	10.6

表.3 磁気記録層膜厚による有効磁界勾配の比較

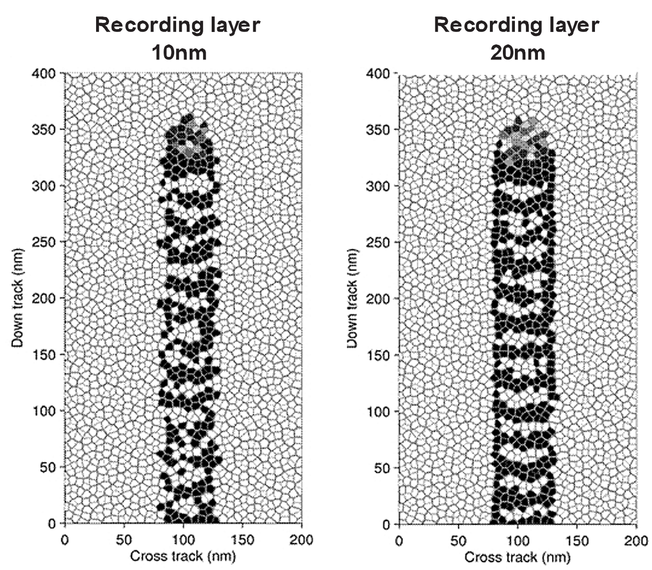


図.9 磁気記録層による記録状態の比較

磁気記録層膜厚 20nm の場合が 10nm の場合よりも高い SNR を示すことは、有効磁界勾配の差で理解できる．しかし磁気記録膜厚 10nm の場合は、ヘッド磁界勾配絶対値より 2.8 倍も高い有効磁界勾配を示しており、有効磁界勾配の値だけでは磁気記録層膜厚 10nm の低い SNR を説明することは難しい．磁気記録膜厚 10nm の場合について、時間ステップにおける記録磁化過程の解析を行い考察した．

4.5 磁気記録膜厚 10nm の場合における記録磁化過程の解析

図.10-1 は、磁気記録膜厚 10nm の場合について、赤でマークした磁性微粒子の記録磁化過程を示している．13.25(nsec)付近で磁化がヘッド磁界方向(-1.0)に磁化反転した後、次のビット形成磁界に引きずられ、13.58(nsec)付近で逆方向(+1.0)に再反転している．記録後に逆方向に再反転する粒子の存在はジッタの原因となり、記録信号の SNR が減少する．また、逆方向に再反転した時の磁界強度は約 6000(Oe)であり、温度は 575K であることがわかる(図.10-2)．

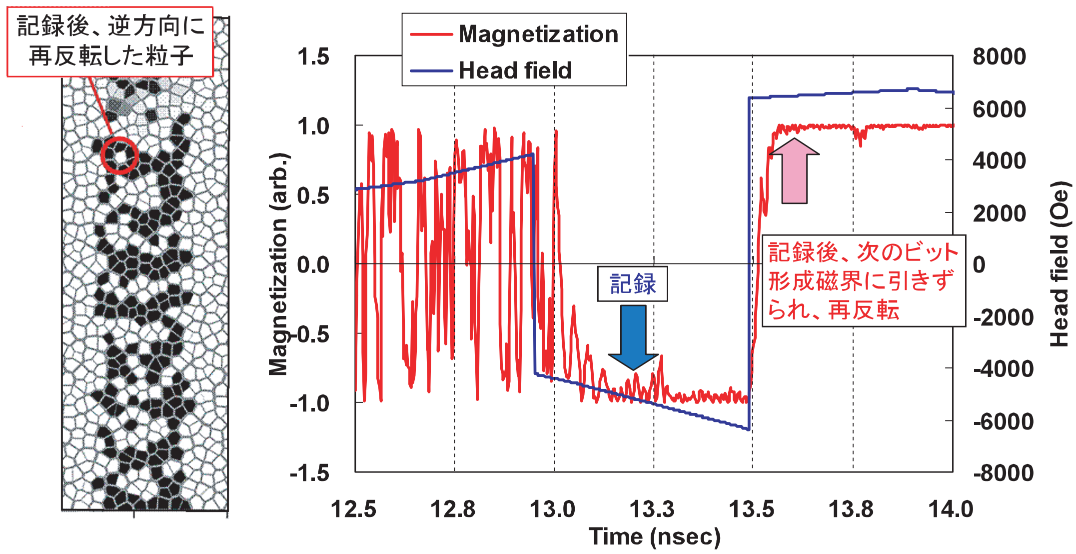


図.10-1 逆方向に再反転した粒子の磁化方向とヘッド磁界の変化

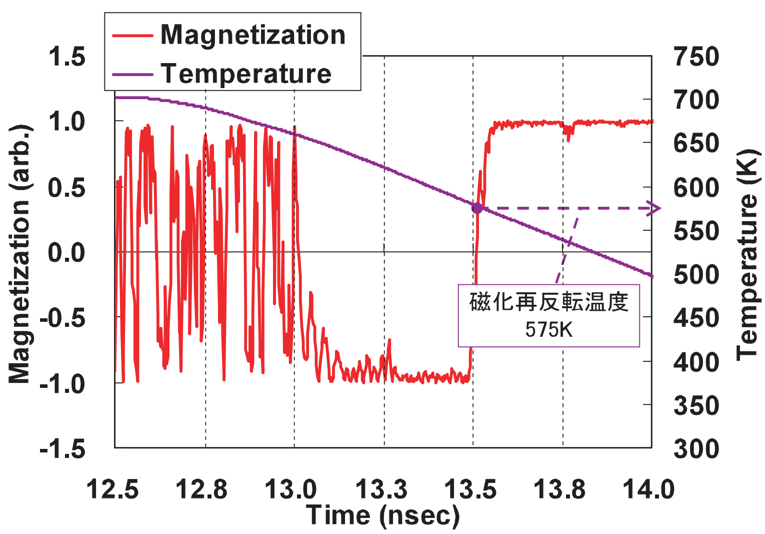


図.10-2 逆方向に再反転した粒子の磁化方向と温度の変化

磁気記録層膜厚の差を比較するため、磁気記録層膜厚 10nm と 20nm について逆方向に再反転した時と同じ環境条件（温度 577K，磁界 6000 Oe）を与え、記録状態の比較を行った．比較を容易にするため、計算機上で理想的な磁化パターンをプログラムした後に、577(K)と6000(Oe)の温度と磁界を与えた(図.11)．磁界方向は黒のパターンに対して逆方向(白のパターン方向）とした．

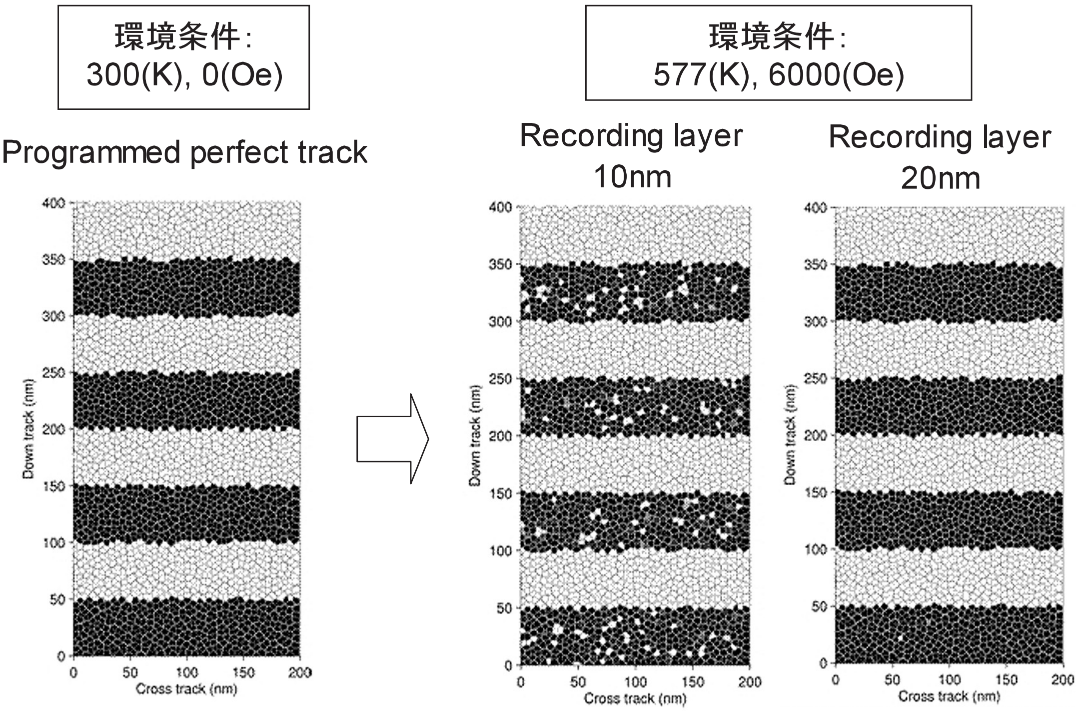


図.11 環境条件 577(K),6000(Oe)における、磁気記録層膜厚 10nm と 20nm の記録状態比較

図.11 から､磁気記録膜厚 10nm は 20nm に比べて再反転粒子が多いことから､577(K),6000(Oe)の環境条件では、記録後再反転しやすいことがわかる．熱アシスト磁気記録では、加熱によりHcを下げて記録する特徴があるが､加熱後の余熱のため再反転を引き起こす可能性があり､記録後も十分な熱安定性を確保する必要がある．図.4 の Ku の温度依存性から 577(K)の Ku を求めると､7.5e6(erg/cc)である．粒径 5nm について磁気記録層膜厚 10nm と 20nm の熱安定性指数(KuV/KbT)を求めると、各々18 と 37 と見積もられる．記録点付近での熱安定性指数の確保が重要であることを示唆している．

5 まとめと展望

熱アシスト磁気記録方式は今後の HDD の記録密度増大に有望な技術であり、本研究における LLG シミュレーションからも、2000kFCI の高線記録密度の可能性を示すことができた.一方で、ナノメートルオーダーの光による、微小磁性粒子の磁化反転制御に関連した物理現象（媒体中の近接場光分布の詳細や､Tc 付近の磁化挙動）を完全に反映しているとは言い難く、今後も第一原理計算による磁気記録過程の解析を進めることが必要である．

謝辞

本研究は､先端的大規模計算シミュレーションプログラム利用サービス 戦略分野利用推進において、東北大学サイバーサイエンスセンターのスーパーコンピュータ(SX-9)を利用することで実現することができました．この場を借りて関係各位に感謝いたします．

参考文献

[1] S. J. Greaves and H. Muraoka, "Effect of the temperature dependence of H_k in thermally assisted recording", J. Appl. Phys., Vol.101, 09H502(1-3), 2007.
[2] S. J. Greaves, T. Mizuno and H. Muraoka, "Transition Formation in Heat-Assisted Magnetic Recording", J. Magn. Soc. Jpn., Vol.34, No.2, pp. 49-52, 2010.
[3] T. Matsumoto, K. Nakamura, T. Nishida, H. Hieda, A. Kikitsu, K. Naito, and T. Koda, "Thermally assisted magnetic recording on a bit-patterned medium by using a near-field optical head with a beaked metallic plate", Appl. Phys. Lett., 93, 031108, 2008.
[4] JMAG-Studio, Commercial software, JSOL, Ltd,:http://www.jmag-international.com.
[5] R. Rottmayer et al., "Heat Assisted Magnetic Recording", IEEE Transcations on magnetics, vol.42, no.10, 2006.

白

[全国共同利用情報基盤センター研究開発論文集 No.32]より

大規模科学計算システムの紹介と性能評価

山下毅[†]　江川隆輔[‡]　小野敏[†]　伊藤英一[†]　岡部公起[‡]　小林広明[‡]

[†]東北大学情報部情報基盤課
[‡]東北大学サイバーサイエンスセンタースーパーコンピューティング研究部

1　はじめに

東北大学サイバーサイエンスセンター（以下、当センター）では，当センターのユーザの皆様へ，より強力かつ使い易いシステム提供を目指して，2010年4月にベクトル型スーパーコンピュータ SX-7C を SX-9 へ，また，並列型コンピュータシステム TX7/i9610 を，最新鋭のスカラ型サーバ Express5800/A1080a-D へとリプレースした．これにより，当センターの大規模科学計算システムは，ベクトル機能の強化ばかりでなく，汎用アプリケーション，もしくはベクトル型スーパーコンピュータでは効率的に処理できないアプリケーションの実行環境も大幅に強化した．本稿では，Express5800/A1080a-D システムのハードウェア，およびプログラミングを支えるソフトウェアの紹介をはじめ，当センターにおける並列計算機の運用，そして実アプリケーションを用いた性能評価について説明する．

2　大規模科学計算システムの紹介

2.1　システム更新概要

今回のリプレースでは，当センターの大規模科学計算システムのフロントエンドサーバである TX7/i9610 3ノードを，Express5800/A1080a-D [1]（日本電気（株）製）6ノードに，計算サーバであるベクトル型スーパーコンピュータ SX-7C の5ノードを，同じくベクトル型スーパーコンピュータ SX-9 [2]（日本電気（株）製）の2ノードに更新した．図1に更新された大規模科学計算システムの構成を示す．ファイルサーバは，100TB の記憶容量を持つ磁気ディスク装置である．Express5800/A1080a-D の各ノードは，高速なノード間接続装置 (Infiniband)で高速に相互接続されている．また SX-9 およびファイルサーバとはギガビットイーサネットスイッチおよびファイバチャネルスイッチにより接続されている．ギガビットイーサネットスイッチは学内 LAN(StarTAINS) と接続され，大規模科学計算システムは学内外のユーザにサービスされている．

旧システムにおけるベクトル型スーパーコンピュータ SX-7C の 5 ノード（ピーク性能 640Gflop/s, 640GB memory）は，2 ノードの SX-9(3.2Tflop/s, 2TB memory) にリプレース

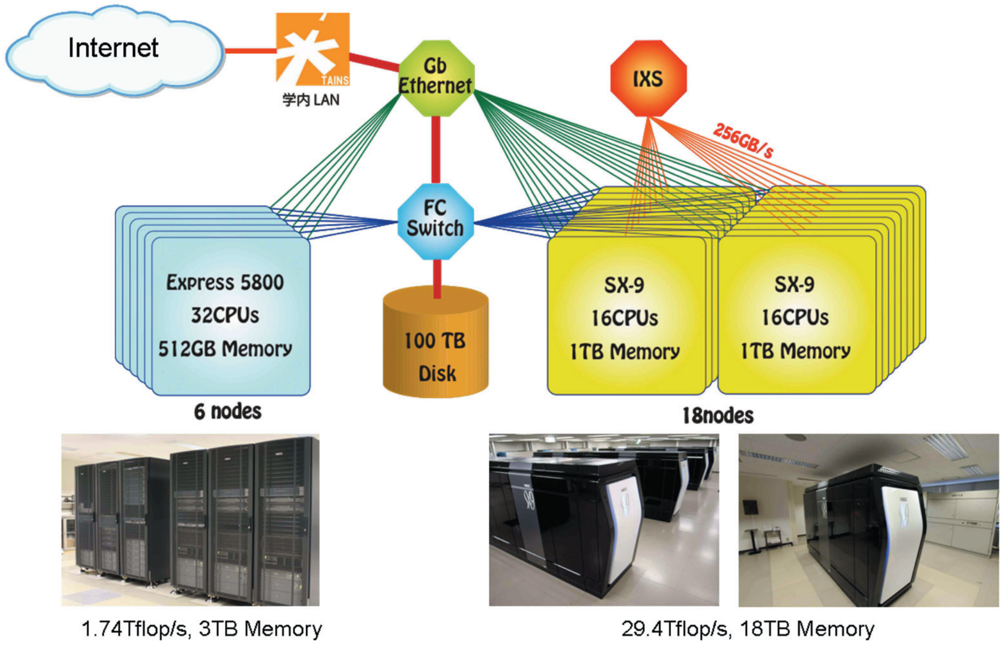


図１　大規模科学計算システムの構成

され，既に導入済みの 16 ノードの SX-9 と併せてトータル 29.4Tflop/s となり，全国の当センターユーザのベクトル型計算機に対する高いニーズに応えられるシステム構成とした．新システムであるスカラ並列コンピュータ Express5800/A1080a-D は 1.74Tflop/s の演算能力を有し，Gaussian, Marc, Marc Mentat, Mathematica, MATLAB, SAS 等の汎用アプリケーションの高速実行環境を提供するとともに，高いベクトル化率が達成困難なアプリケーション実行をサポートするためにリプレースされた．また，Express5800/A1080a-D は，当センターの主力計算機であるベクトル型システムのフロントエンドサーバとしても利用されており，並列ユーザばかりでなくベクトルユーザにも快適なプリプロセッシング環境の提供を可能にしている．今回のリプレースにより，ベクトル型システムおよびスカラ型システムを合わせた大規模科学計算システムの理論性能は 31.14Tflop/s に達する．

次に今回新たに導入した Express5800/A1080a-D システムの詳細について述べる．

2.2　Express5800/A1080a-D システム

Express5800/A1080a-D は，米国 Intel® 社の最新の 64bit 8 コアプロセッサである Intel Nehalem-EX プロセッサをノードあたり 4 台（32 コア）搭載している．当センターでは 6 ノード導入することにより，合計 192 コア，1.74Tflop/s の理論性能を実現している．メモリは各ノードあたり 512GB 搭載し，大規模演算を強力にサポートしている．各ノードは 40GB/sec の通信性能を有する Infiniband で接続され，高速なノード間通信を実現している．旧並列コンピュータシステムと新並列コンピュータシステムの諸元を表 1 に示す．システム全体では

表１　並列コンピュータ諸元比較

		TX7/i9610	Express5800/A1080a-D	向上比
1 コア	動作周波数	1.6GHz	2.26GHz	1.4 倍
	最大演算能力	6.4Gflop/s	9.06Gflop/s	1.4 倍
1CPU ソケット	コア数	2	8	4 倍
	最大演算能力	12.8Gflop/s	72.48Gflop/s	5.6 倍
	メモリバンド幅	8.3GB/s	34.1GB/s	4.1 倍
	L3 キャッシュ容量	24MB	24MB	1.0 倍
1SMP ノード	最大演算能力	409.6Gflop/s	289.9Gflop/s	0.7 倍
	メモリ容量	512GB	512GB	1.0 倍
システム全体	ノード間通信速度	800MB/s	4GB/s	5.0 倍
	最大演算能力	1.2Tflop/s	1.7Tflop/s	1.4 倍
	メモリ容量	1.5TB	3TB	2.0 倍

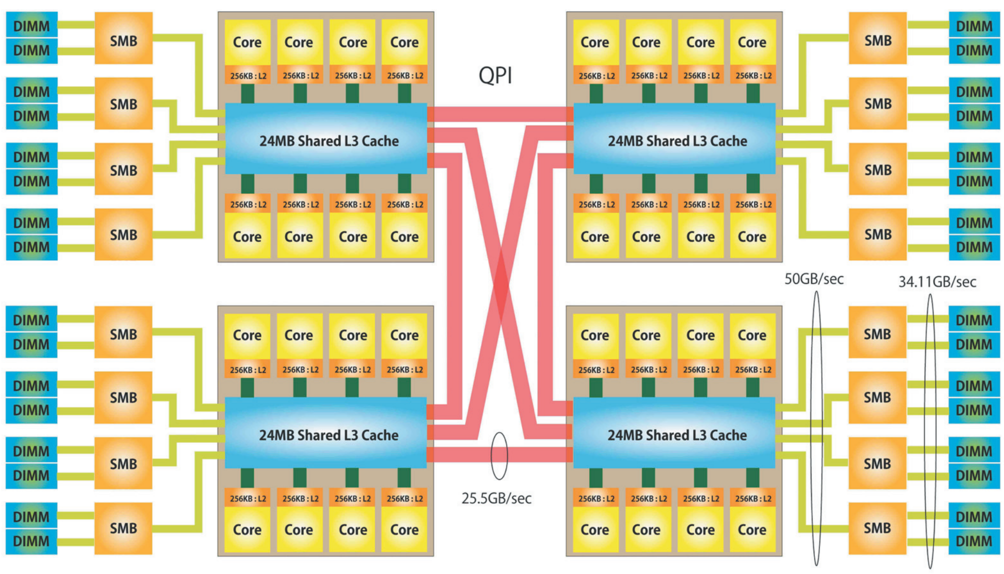


図２　ノード内構成図

Express5800/A1080a-D は TX7/i9610 と比較して，ノード間通信速度で 5.0 倍，最大演算能力で 1.4 倍, メモリ容量で 2.0 倍の性能を有している．

次に，図２に Express5800/A1080a-D のノード内部の構造を示す．Express5800/A1080a-D に搭載されている Intel Xeon プロセッサ 7500番台は，１クロックあたり，最大で４個の演算を同時に実行可能な Nehalemマイクロアーキテクチャと，単一 CPU ソケット内に８つのコアプロセッサを内蔵したマルチコア構造を採用している．これにより，ソケットあたりの最大性能は 72.3Gflop/s に達する．当センターに導入した Express5800/A1080a-D では，各ノードあたり４つの CPU を搭載しており，ノードあたりの性能は 289.3Gflop/s となる．これは旧シス

テムの TX7/i9610 と比較して，コアあたりで 1.4 倍，CPU ソケットあたりで 5.6 倍の計算能力を有する．また，各 CPU 間は Quick Path Interconnect (QPI) を介して，高いバンド幅で結合されている．メモリのスループットとレイテンシに関しても，Express5800/A1080a-D では，TX7/i9610 と比較して高い性能を実現している．TX7/i9610 では，2 つの CPU 毎に Front Side Bus (FSB) を共有した論理構造を取っていたのに対して，Express5800/A1080a-D では，プロセッサとメモリコントローラを単一の CPU ソケット内に実装するために，図 2 に示す様に各メモリは SMB(Scalable Memory Buffer) を介して，CPU に直接接続される論理構造を採用している．このため，プロセッサコアから同一ソケット内のメモリへアクセスする際には，DDR3 DIMM のスループットを十分に活用できる．Express5800/A1080a-D では，プロセッサコアあたり TX7/i9610 の約 1.4 倍のメモリスループットを実現している．これらの新たなアーキテクチャの導入により，さらなるユーザプログラムの高速化が期待される．

2.3　運用環境

プログラミング言語および科学技術計算用ライブラリとして，表 2 に示すものが利用出来る．コンパイラは自動並列化機能を有しているので，既存の逐次処理プログラムを修正することなく並列実行が可能である．その他 OpenMP によるノード内並列化および，MPI による複数ノードを使用した並列実行も可能である [4]．また，表 3 に示す汎用アプリケーションの実行環境が提供されている．特に分子軌道計算プログラムの Gaussian は最大 16 並列までの並列処理が可能で，実行時間の大幅な短縮が可能である．バッチ処理としては，スカラ型並列コンピュータおよび，ベクトル型スーパーコンピュータに NQS II(Network Queuing System)[3, 4] を導入することで，ジョブの一元管理が可能となり，利便性の高いジョブ投入環境となっている．スカラ型並列コンピュータで提供しているジョブクラスを表 4 に示す．

表 2　プログラミング言語及びライブラリ

Fortran95	ISO/IEC 1539-1:1997，自動並列化，OpenMP
C/C++	ISO/IEC 14882:1998，自動並列化，OpenMP
MPI	並列処理ライブラリ
ASL	Fortran95用科学技術計算ライブラリ
Math Kernel Library	数値演算ライブラリ

表 3　提供アプリケーション

Gaussian09,03	非経験的分子軌道計算プログラム
MSC.Marc / MSC.Marc Mentat	汎用構造解析プログラム
Mathematica	数式処理プログラム
MATLAB	科学技術計算言語
SAS	データ解析システム

表 4　ジョブクラス

ジョブクラス（キュー名）	利用可能コア数（並列数）	CPU 時間制限	メモリサイズ制限(GB)
as	1	無制限	16
am（Marc 専用）	1	無制限	16
am2（Marc 専用）	1	無制限	128
a8	8	無制限	128
a16	16	無制限	256
a32	32	無制限	512

3　Express5800/A1080a-D の性能評価

今回導入したシステムの性能を明らかにするために，当センターで実行されているユーザプログラムおよび，汎用アプリケーションである Gaussian03により性能評価を行った．新システムの Express5800/A1080a-Dでは表 3 の通り Gaussian09の提供も行っているが，旧システムの TX7/i9610 との性能比較を行うために Gaussian03での演算を行った．

3.1　プロセッサ性能

各プロセッサの諸元を表 5 に示す．ここで，B/Flop(Byte per flop) は演算命令あたりのメモリ転送能力を表す．始めに，当センターを利用するユーザが作成した実アプリケーションによる性能評価を行った．評価に用いたベンチマークは，Earthquake (地震解析)，Turbulent Flow (乱流)，Antenna (アンテナ)，Turbine (タービン)，Land Mine (地雷探索)，Plasma (プラズマシミュレーション）の 6 種類である．プロセッサの実行性能を比較した結果を図 3 に示す．上記の 6 つのアプリケーションのうち，4 つ（グラフでは左 4 つ）はプロセッサの演算能力が支配的なプログラムであり，2 つ（グラフでは右 2 つ）はメモリアクセスが支配的なアプリケーションで

表 5　Intel プロセッサ諸元比較

	クロック周波数 (GHz)	ピーク性能 (Gflop/s)	ピークメモリバンド幅 (GB/s)	コア数	B/Flop	キャッシュ容量
Express5800/A1080a-D Nehalem-EX	2.26	72.48	34.1	8	0.47	L2:256kB/core L3:24MB
Nehalem-EP	2.93	46.93	25.6	4	0.55	L2:256kB/core L3:8MB
TX7/i9610 Itanium II	1.60	12.80	8.5	2	0.66	L2:256kB/core L3:24MB

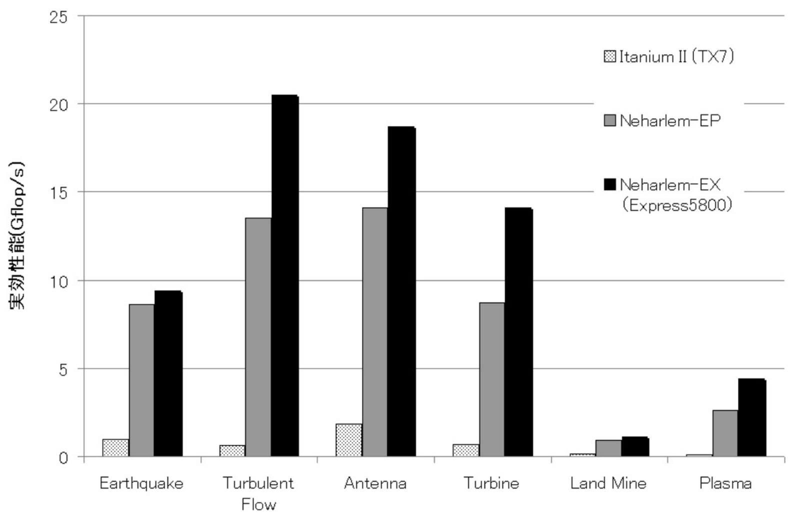


図 3　プロセッサ実効性能

ある．

グラフから分かるように，旧システムの TX7/i9610 と比較して大幅な性能向上を実現している．また参考として，Nehalem-EP (4 コアの Xeon) の値も併記している．ここで，Nehalem-EX の B/Flop の値が Nehalem-EPと比較して低いのは，コアあたりのメモリバンド幅が狭いことが原因として考えられる．高効率なジョブの実行のためにはコアあたりのメモリバンド幅が重要となる．

3.2　ノード性能

次に，次世代 CFD アルゴリズムである BCM(Building Cube Method) アプリケーションをノード内でシングル実行または並列実行し，コア数毎に評価した実効性能を図 4 に示す．新シス

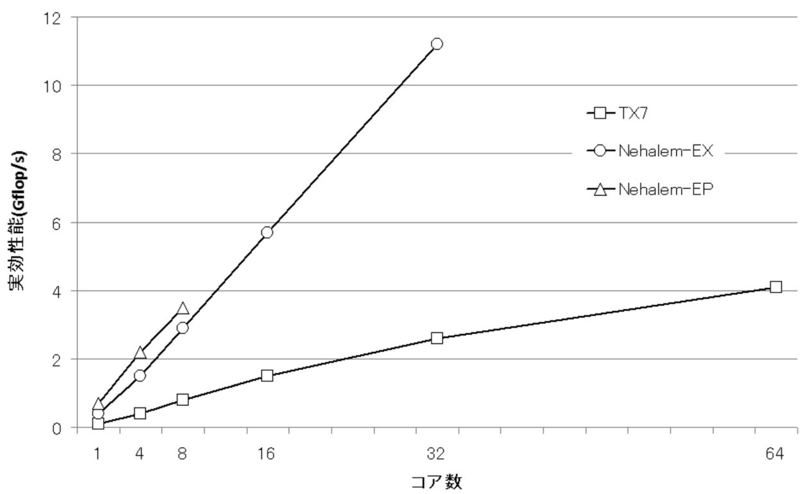


図 4　ノード実効性能

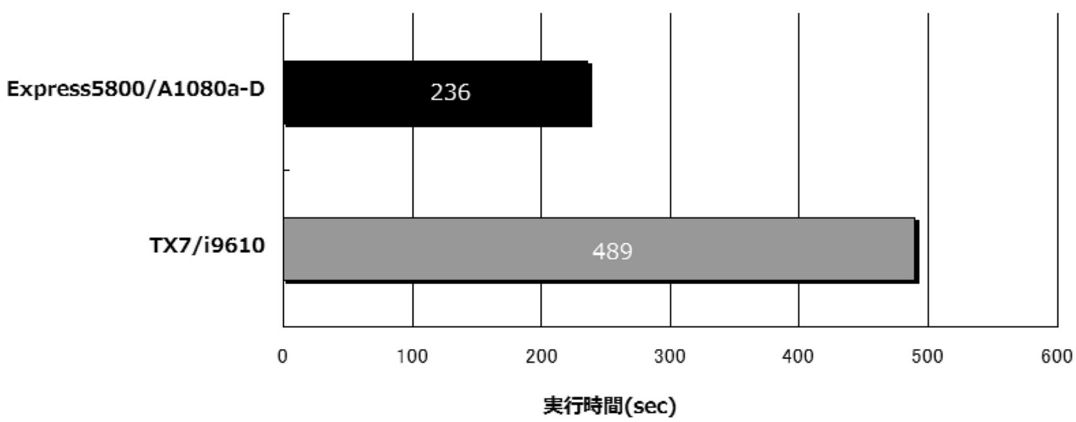


図 5　Gaussian03 32 並列テストプログラムの実行結果比較

テムは旧システムと比較して，同コア数での実効性能で大幅な性能向上を示すとともに，並列コア数に対してリニアに性能を向上させており，並列実行性能の高さも示している．また，旧システムと比較してノードあたりの総コア数は半減しているが，システム全体としても約 2.7 倍の性能向上を示している．

最後に，Gaussian03のテストプログラムを用いて性能評価を行った結果を図 5 に示す．旧システムの TX7/i9610 および新システム Express5800/A1080a-D 上で 32 並列テストプログラムの実行に要した時間を比較すると，新システムは従来システムと比較して約 2 倍の性能向上を実現している．これは各コア間を高速に繋ぐ QPI の性能と，約 1.4 倍高速になったメモリスループットの影響が大きいものと考えられる．

4　おわりに

本稿では，新並列コンピュータシステム Express5800/A1080a-D についてシステム構成，および演算性能評価について紹介した．実効性能，並列実効性能ともに期待通りの性能向上が見られた．当センターにおいては，システム更新によるハードウェアの性能向上を超える勢いで，利用者のジョブは大規模・長時間化してきている．このため，システムの増強と併せて 1997年から高速化推進活動を行い利用者プログラムの高速化および並列化率の向上を図っており，高度化する利用者のニーズに対応できるサービスの提供を目指している．

謝辞

本プログラム評価において快くプログラムをご提供いただいた，東北大学東北アジア研究センター佐藤研究室，東北大学大学院理学研究科地震・噴火予知研究観測センター長谷川研究室，東北大学工学研究科航空宇宙工学専攻升谷研究室，東北大学工学研究科電気・通信工学専攻澤谷研究室，東北大学工学研究科機械システムデザイン工学専攻太田研究室，東北大学工学研究科航空宇宙工学専攻中橋研究室に感謝いたします．

参考文献

[1] 日本電気株式会社　那須康之，鈴木健一，谷岡隆浩 Express5800/A1080a-D のハードウェア 東北大学サイバーサイエンスセンター大規模科学計算システム広報 SENAC Vol.43 No.3 2010-7

[2] 日本電気株式会社　稲坂純，萩原孝 スーパーコンピュータ SX-9 のハードウェア 東北大学サイバーサイエンスセンター大規模科学計算システム広報 SENAC Vol.41 No.3 2008-7

[3] 東北大学情報部情報基盤課 システム管理係，東北大学サイバーサイエンスセンタースーパーコンピューティング研究部 スーパーコンピュータシステム SX-9 利用ガイド 東北大学サイバーサイエンスセンター大規模科学計算システム広報 SENAC Vol.41 No.2 2008-4

[4] 東北大学情報部情報基盤課 共同研究支援係，共同利用支援係，東北大学サイバーサイエンスセンタースーパーコンピューティング研究部 並列コンピュータ Express5800 の利用法 東北大学サイバーサイエンスセンター大規模科学計算システム広報 SENAC Vol.43 No.2 2010-4

［報　告］

第13回テラフロップワークショップ報告

東北大学サイバーサイエンスセンター　小林広明

サイバーサイエンスセンターは、大学院情報科学研究科とドイツシュトゥットガルト大学高性能計算センターとの共同主催で、2010 年 10 月 21 日（木）－10 月 22 日(金)の両日、今年も「計算科学の可能性と次世代スーパーコンピュータ研究開発動向を探る」をテーマに、第 13 回テラフロップワークショップ（13th Teraflop Workshop）を開催しました。本ワークショップは、国際的に活躍している計算科学・計算機科学の研究者・技術者を招いて、高性能・高効率大規模科学計算に関する最新の研究成果の情報交換を行うとともに、今後のスーパーコンピュータ設計の方向性を議論することを目的として、2006年以降、毎年シュトゥットガルト大学と東北大学で交互に開催されてきました（詳しくは、http://www.teraflop-workbench.org/index.htm を参照)。

第 13 回テラフロップワークショップはシュトゥットガルト大学、アーヘン工科大、東京大学、理化学研究所、地球シミュレータセンター、日本原子力開発機構、北陸先端科学技術大学院大学、国立情報学研究所、NEC、東北大学からの 17 の招待講演で構成され、次世代スーパーコンピュータ「京」の開発状況や、プラズマ物理、材料、海洋・気象などの分野から最新の研究成果発表がありました。また、国立情報学研究所からは、我が国の基盤センター群の様々なスーパーコンピュータシステムを連携・共有させるグリッドミドルウェアの開発状況についての発表がありました。この技術は、現在我が国が検討を進めているスーパーコンピュータ利用基盤 HPCI の要素技術として注目を集めております。NEC からは、次期 SX 開発のためのコア技術の紹介と HPC 製品開発のロードマップが示されました。東北大学からは、新しいプログラミングフレームワークである OpenCL 経由で SX を利用するためのプログラミングモデル変換に関する研究成果の発表がありました。

2 日間のワークショップでは、延べ 162 名の参加者を得て、活発な議論が交わされました。第 14 回ワークショップは、2011 年 3 月 21 日～22 日にシュトゥットガルト大学で行われる予定です。

[Web 版大規模科学計算システムニュース]より

大規模科学計算システムニュースに掲載された記事の一部を転載しています。　http://www.ss.isc.tohoku.ac.jp/tayori/

平成23年度 学際大規模情報基盤共同利用・共同研究拠点
公募型共同研究課題募集のご案内（No.112）

東北大学サイバーサイエンスセンターは「学際大規模情報基盤共同利用・共同研究拠点」として文部科学大臣の認定を受け、活動を行っております。「学際大規模情報基盤共同利用・共同研究拠点」は、北海道大学、東北大学、東京大学、東京工業大学、名古屋大学、京都大学、大阪大学、九州大学にそれぞれ附置するスーパーコンピュータを持つ8つの共同利用の施設を構成拠点とし、各センターからなる「ネットワーク型」共同利用・共同研究拠点として、平成22年4月より本格的に活動を開始しました。

本ネットワーク型拠点の目的は、超大規模計算機と大容量のストレージおよびネットワークなどの情報基盤を用いて、地球環境、エネルギー、物質材料、ゲノム情報、Webデータ、学術情報、センサーネットワークからの時系列データ、映像データ、プログラム解析、その他情報処理一般の分野における、これまでに解決や解明が極めて困難とされてきた、いわゆるグランドチャレンジ的な問題について、学際的な共同利用・共同研究を実施することにより、我が国の学術・研究基盤の更なる高度化と恒常的な発展に資することにあります。本ネットワーク型拠点には上記の分野における多数の先導的研究者が在籍しており、これらの研究者との共同研究によって、研究テーマの一層の発展が期待できます。本公募型共同研究（平成23年度）は、平成23年4月～平成24年3月までの期間に実施します。

申し込み方法等詳細につきましてはウェブサイト
（ http://jhpcn-kyoten.itc.u-tokyo.ac.jp/ ）をご覧ください。

東北大学サイバーサイエンスセンター	名古屋大学情報基盤センター
北海道大学情報基盤センター	京都大学学術情報メディアセンター
東京大学情報基盤センター	大阪大学サイバーメディアセンター
東京工業大学学術国際情報センター	九州大学情報基盤研究開発センター

― 課題募集スケジュール ―

2011年	1月 7日	課題応募受付開始
	2月10日	課題応募受付締切　(17:00)
	3月中旬	採択通知
	4月 1日	共同研究開始
2012年	3月15日	共同研究課題終了

ご不明な点がありましたら、以下までお問い合わせください。

東北大学窓口	全体窓口
sys-sec@isc.tohoku.ac.jp	kyoten@itc.u-tokyo.ac.jp

（スーパーコンピューティング研究部，共同研究支援係，共同利用支援係）

研究成果リスト提出のお願い（No.112）

平素、本センターの大規模科学計算システムをご利用いただきありがとうございます。

本センターでは、学術研究を支える世界最高水準の大規模科学計算システムの導入と利用環境の整備・拡充を行い、研究の発展に資することを心掛けてまいりました。今後もシステムの整備を進めていくには、大規模科学計算システムが多くの研究分野で必要不可欠であり、かつ研究成果が得られていることを広くアピールしていく必要があります。

つきましては、本センター大規模科学計算システムを利用して得られた研究成果を、下記により提出くださるようお願い申し上げます。なお、提出していただく研究成果は、平成 22 年度中に発表されたものとします。

記

1．研究成果リスト：著者名、論文名、掲載誌（巻号頁）、発表年

2．提出方法　：電子メールでお願いします。
提出先メールアドレス　seika@isc.tohoku.ac.jp

3．締切り日　：平成 23 年 4 月 22 日（金）

4．問合せ先　：共同利用支援係（022-795-6251,uketuke@isc.tohoku.ac.jp）

論文等への利用の明記について

研究成果を論文等で発表する場合には、本センターを利用した旨を明記してください。

― 記入例 ―

「本研究の実験結果の一部は、東北大学サイバーサイエンスセンター大規模科学計算システムを利用して得られた。」

Part of the experimental results in this research were obtained using supercomputing resources at Cyberscience Center, Tohoku University.

（共同利用支援係）

科学研究費補助金等による負担金の支払いについて（No.112）

東北大学の支払責任者が、利用負担金を科学研究費補助金等（科学研究費補助金、21 世紀 COE プログラム、その他預り科学研究費補助金等）で支払う場合は、下記の shiharai コマンドによる指定が必要となります（**この指定がなければ科学研究費補助金等で支払うことができませんのでご注意ください**）。次回の利用負担金請求（平成 23 年 1 月 1 日から 2 月 15 日までの利用分）は 2 月下旬に行いますので､科学研究費補助金等で支払う場合は shiharai コマンドで 2 月 15 日(火) までに指定してください。

shiharai コマンドでは、支払順位、支払費目、科研費課題番号、支払予算額を指定します。次頁に、科研費(課題番号 87654321) で 10 万円、残りは科研費以外で支払う例を示しています。不明な点がある場合は、共同利用支援係（022-795-6251）へお問い合わせください。

【shiharai コマンドの使用例 】

ssh gen.isc.tohoku.ac.jp -l 利用者番号　　*並列コンピュータにログイン
利用者番号@gen.isc.tohoku.ac.jp's password: **パスワード**　　して下線部を入力します。

```
gen%  shiharai
        2 月 11 日 現在の利用負担金と支払の指定はつぎのとおりです
                  支払責任者                 :       秋田　太郎 (u23456)
                  4 月からの累計利用負担金額 :       200,000 円
                  請求済額                   :       150,000 円
                  今期使用金額               :        50,000 円
－－－－－－－－－－－－－   現在の支払指定   －－－－－－－－－－－－－－－－－－
支払順位    支払費目名    支払予算額    請求済額      残額      経理担当者
                         何も指定されていません
－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－

        1.支払指定            2.支払予算額の変更      3.支払順序の変更
        4.支払指定の削除      5.負担金の明細表示      8.help
        9.終了
何番の処理を選びますか？  1
支払順位を入力してください？  1
        1.大学運営資金             2.受託研究費                3.受託事業費
        4.寄附金                   5.科学研究費補助金等

支払費目を番号で入力してください？  5
科研費課題番号を入力してください：  87654321
支払予算額を入力してください（円）？  100000
登録してよろしいですか(yes/no)   y
```

（共同利用支援係, 会計係）

利用負担金（1月～3月分）について（No.112）

大規模科学計算システムの平成23年1月～3月分の利用負担金の請求を次のように行います(昨年と同じです）。

《学内支払責任者》

1月から2月中旬までの利用負担金を2月下旬に請求します。その後3月末までの利用負担金は、翌年度4月に請求します（平成23年度の大学運営資金で予算の調整を行います）。

《学外支払責任者》

1月から2月中旬までの実績額と3月末までの利用見込額を合算した利用負担金を2月下旬に請求します。利用見込額は以下のmikomiコマンドで2月15日(火)までに申告してください。

なお、見込額と実績が異なっても精算は行わないのでご注意ください。

【 mikomi コマンドの使用例 】

```
ssh gen.isc.tohoku.ac.jp -l 利用者番号                  *並列コンピュータにログイン
利用者番号@gen.isc.tohoku.ac.jp's password: パスワード    して下線部を入力します

gen%  mikomi

  2月5日　現在の見込み額は次のとおりです。
      支払責任者     :   〇〇　〇〇　(u23456)
      見込み額指定者 :
      見込み額       :         0円
1. 見込み額の指定   2. 削除   9. 終了
何番の処理を選びますか？1
見込み額を入力してください(円)？200000
登録してよろしいですか（yes/no）？yes
・・・・
gen%
```

（共同利用支援係,会計係）

平成23年度共同研究の募集について（No.113）

東北大学サイバーサイエンスセンターでは、研究者のより良いスーパーコンピューティング環境を構築するために、平成 23 年度のスーパーコンピューティングに関する共同研究を下記の要領で募集します。

この共同研究では、シミュレーションアルゴリズムの開発やプログラムの高速化に関する研究を本センタースタッフと共同で行うことが必須の条件となります。

1．共同研究の対象

本共同研究では、大規模な計算を大量に行う利用者の研究分野の研究課題を対象とします。利用者は、スーパーコンピュータで処理するプログラムのベクトル化や並列化、または並列コンピュータで処理するプログラムの並列化に関する研究を本センターと共同で行います。ただし、同一研究課題（内容）による申請は 3 年を限度とします。また、本年度は特にプログラムの MPI 化に積極的に取り組む課題を募集します。

2．応募者の資格

本センター大規模科学計算システムの利用有資格者

3．共同研究の要件

(1)研究期間は平成 23 年 4 月から平成 24 年 3 月までの間とします。
(2)プログラムのベクトル化や並列化の研究を、本センターのスタッフと共同で行います。
(3)共同研究を行うためにプログラムコード等を提供していただきます。
(4)共同研究の研究成果、あるいは研究成果であるプログラムにより得られた結果を論文等で公表する際、共同研究をふまえて本センターの貢献を明記してください（例えば、共著者、謝辞等)。また、共同研究の成果を本センター大規模科学計算システム広報誌 SENAC に投稿していただきます。この記事には、プログラムのベクトル化や並列化などの具体的な内容と効果、及び本センターとの共同した取り組みを記述していただく必要があります。
(5)年度末に共同研究実績報告書（所定の様式）を提出していただきます。この報告書にも本センターの貢献や謝辞等を必ず明記してください。
(6)研究成果を学会等へ報告した場合は、その別刷等を本センターに提出してください。

4．助成内容等（内容が従来と異なっています）

採択された課題の研究代表者に対し、50 万円を超えた分の演算負担金額の 2/3 を本センターが負担します。ただし、本センターの負担金額の上限は 500 万円とします。

5．応募締切

平成 23 年 2 月 18 日(金)必着

6．審査方法等

申請された課題は、ライブラリ・共同研究専門部会において本制度の趣旨に適合しているか否かを審査し、10 件程度を選考して結果の公表を行う予定です。

7．応募方法（従来の応募方法とは異なっています）

応募される方は、本センターのウェブサイト（http://www.ss.isc.tohoku.ac.jp/）の各種申請用紙(共同研究申請書)を利用して電子メールでお申し込みください。

送り先　E-mail ： sys-sec@isc.tohoku.ac.jp

８．問い合わせ先

共同研究支援係　TEL:(022)795-6252
送り先　E-mail ： sys-sec@isc.tohoku.ac.jp

（スーパーコンピューティング研究部，共同研究支援係）

平成23年度研究開発の公募について（No.113）

東北大学サイバーサイエンスセンターでは、大規模科学計算システムにおけるライブラリを充実し研究者の利便性を向上させるため、平成 23 年度のライブラリ研究開発を下記の要領で募集します。

本センターは利用者の新規ライブラリ研究開発に対して計算機利用時間の提供等の援助を行います。新規にライブラリを研究開発する研究者及び既に開発されてライブラリに登録したいプログラムをお持ちの方も応募できます。

∞∞∞∞∞∞　ライブラリ研究開発募集要領　∞∞∞∞∞∞

１．ライブラリ研究開発の目的及び援助

本センター大規模科学計算システムにおけるソフトウェア・ライブラリの充実及び利用手段の拡充等を目的とし、本センターは研究開発計画に対して計算時間の提供等や、既に開発されているプログラムをライブラリへ登録する場合の計算時間の提供等の援助を行います。

２．研究開発対象分野

研究開発対象分野としては、次のような分野で、当該ライブラリが他の利用者の利用に寄与するものに限ります。

(1)基礎および応用プログラムに関するもの
(2)計算機言語に関するもの
(3) その他計算機利用全般に関するもの

３．応募者の資格

本センター大規模科学計算システム利用有資格者。なお、大学院生は研究開発計画の開発協力者になることはできますが応募者になることはできません。

４．応募締切日

平成 23 年 2 月 18 日（金）

５．応募方法

応募される方は、本センターのウェブサイト（http://www.ss.isc.tohoku.ac.jp/）の各種申請用紙を利用して電子メールで共同研究支援係あて申し込みください。

６．研究開発の要件

(1)研究開発期間は当該年度内とし、継続する場合は翌年度に再び申請することができます（3 年まで）。
(2)既に研究開発されたものの登録について、計算時間の提供は当該年度とし、その中でライブラリへの登録等の諸作業も行っていただきます。
(3)研究開発の終了あるいは年度末には、研究開発報告書を提出してください。

７．研究開発課題の選考
　申請された研究開発課題は、本センターにおいて本制度の趣旨に適合しているか否かを審査し、平成 23 年 3 月末までに選考結果を公表する予定です。

８．申請・問い合わせ先
　ライブラリ研究開発に関するお問い合わせは以下へお願いします。

共同研究支援係　TEL ： 022-795-6252
E-mail ： sys-sec@isc.tohoku.ac.jp

（スーパーコンピューティング研究部, 共同研究支援係）

計算科学・計算機科学人材育成のための スーパーコンピュータ無償提供の試行について（No.113）

　東北大学サイバーサイエンスセンターでは、計算科学・計算機科学分野での教育貢献・人材育成を目的として、今年度に引き続き平成 23 年度も大学院・学部での講義実習等の教育目的での利用について、無料（ただし、利用状況によっては上限を設定する場合があります）でベクトル並列型スーパーコンピュータ SX-9（1 ノード，16CPU，1.6Tflop/s，1TB メモリ）システムをご利用いただける制度の試行をいたします。
　利用を希望される場合は、以下の情報を添えて、edu-prog@isc.tohoku.ac.jp までお申し込みください。

・講義担当者氏名
・同所属
・同連絡先（住所，電話，電子メール）
・講義名
・講義実施日時（１セメスターの中で実習を予定している回数）
・センターでの実習利用希望の有無（必要であれば予定日）
・講師派遣の有無
・講義シラバス
・講義ウェブ（もし用意されていれば）
・受講者数（予定）
・必要とする理由（利用目的：例えば、高速数値実験の研修を行うなど）
・期待できる教育効果
・その他（センターへの要望等）

　なお、講義終了後、報告書（広報誌 SENAC へ掲載）の提出をお願いいたします。たくさんのお申し込みをお待ちしております。不明な点は、edu-prog@isc.tohoku.ac.jp まで電子メールでお問い合わせください。

（スーパーコンピューティング研究部, 共同利用支援係）

― SENAC 執筆要項 ―

１．お寄せいただきたい投稿内容

次のような内容の投稿のうち、当センターで適当と判定したものを掲載します。その際に原稿の修正をお願いすることもありますのであらかじめご了承ください。

・一般利用者の方々が関心をもたれる事項に関する論説
・センターの計算機を利用して行った研究論文の概要
・プログラミングの実例と解説
・センターに対する意見、要望
・利用者相互の情報交換

２．執筆にあたってご注意いただく事項

(1)原稿は横書きです。
(2)術語以外は、「常用漢字」を用い、かなは「現代かなづかい」を用いるものとします。
(3)学術あるいは技術に関する原稿の場合､200 字～400 字程度のアブストラクトをつけてください。
(4)参考文献は通し番号を付し末尾に一括記載し、本文中の該当箇所に引用番号を記入ください。
　・雑誌：著者, タイトル, 雑誌名, 巻, 号, ページ, 発行年
　・書籍：著者, 書名, ページ, 発行所, 発行年

３．原稿の提出方法

原稿のファイル形式は Word を標準としますが、PDF での提出も可能です。サイズ*は以下を参照してください。ファイルは電子メールで提出してください。

－Word の場合－
　・用紙サイズ：A4
　・余白：上＝30mm　下＝25mm　左右＝25mm　綴じ代＝0
　・標準の文字数（45 文字 47 行）
＜文字サイズ等の目安＞
　・表題＝ゴシック体 14pt 中央　・副題＝明朝体 12pt 中央
　・氏名＝明朝体 10.5pt 中央
　・所属＝明朝体 10.5pt 中央
　・本文＝明朝体 10.5pt
　・章・見出し番号＝ゴシック体 11pt～12pt
　　*余白サイズ、文字数、文字サイズは目安とお考えください。

４．その他

(1)執筆者には、希望があれば別刷 50 部を進呈します。50 部を超える分については、著者の実費負担とします。別刷の希望部数等は投稿の際に申し出てください。
(2)投稿予定の原稿が 15 ページを超す場合は以下まで前もってご連絡ください。
(3)初回の校正は、執筆者が行って、誤植の防止をはかるものとします。
(4)原稿の提出先は次のとおりです。

東北大学サイバーサイエンスセンター内　情報部情報基盤課共同利用支援係
e-mail　uketuke@isc.tohoku.ac.jp
TEL　022-795-3406

索　　引

SENAC Vol.43 (No.1～No.4)　　2010.1～2010.10

編 集 後 記

新年を迎え、ようやく例年通りの”仙台らしい”気候が続いております。今年はスタッドレスタイヤに履き替えなくても良いかな等と考えていましたが、そうは問屋がおろしてくれませんでした。

さて、昨年は並列コンピュータのリプレース、ベクトル型スーパーコンピュータの増強等が行われました。本年も、これまで以上にユーザの皆様が使いやすい計算環境を提供していけるように、がんばって参りたいと思います。また、当センターの利用に関しましては、通常利用の他に、サイバーサイエンスセンター共同研究や、学際大規模情報基盤共同利用・共同研究拠点公募型研究など、ユーザの皆様とセンター教職員が協力しながら、皆様の研究を支援する取り組みも準備しております．当センターの利用に関して、ご不明な点、ご相談等ございましたら、お気軽にセンター教職員にお声がけください。最後になりましたが、本年もよろしくお願いいたします。(R.E)

2010年の夏が猛暑だったこともあって、秋以降の寒さをとても強く感じています。この冬、仙台の初雪は12月15日でした。平年より23日、観測史上3番目に遅い記録だそうですが、東北に住んでいると、雪が降れば本格的な冬になったという感じがします。今月号がお手元に届くころは、一年で最も寒さが厳しい時期ですので、体調にはお気を付けください。

本誌で何度か紹介させていただいておりますが、サイバーサイエンスセンター1階の展示室は情報処理学会から分散コンピュータ博物館として認定され、常時公開しております。東北地区では、当センターが唯一の認定機関で、貴重な歴史的資料が展示されていますので、是非足をお運びいただき、ご覧くださいますようお願いいたします。

公開は土日祝日を除く午前9時から午後5時まででです。(I.Y)

サイバーサイエンスセンター前
整備中の青葉山新キャンパス

SENAC 編集部会

小林広明　曽根秀昭　水木敬明　後藤英昭
江川隆輔　伊藤英一　山内　斎　小野　敏
高橋洋一　斉藤くみ子

平成23年1月発行
編集・発行　東北大学
サイバーサイエンスセンター
仙台市青葉区荒巻字青葉6-3
郵便番号　980-8578
印　刷　大成印刷株式会社

システム一覧

計算機システム	ホスト名	機　種
スーパーコンピュータ	super.isc.tohoku.ac.jp	SX-9
並列コンピュータ	gen.isc.tohoku.ac.jp	Express5800

サービス時間

利用システム名	利用時間帯
スーパーコンピュータ	連　続　運　転
並列コンピュータ	連　続　運　転
館 内 利 用	月曜日～金曜日は 8:30～21:00、 土・日・祝日は閉館

ジョブクラスと制限値

計算機システム	処　理	ジョブクラス	CPU時間	メモリ容量
スーパーコンピュータ	会話型	(4cpu)	1 時間	8GB
	バッチ処理	ss　(4cpu)	1 時間	256GB
		s　(4cpu)	無制限	32GB
		p8　(8cpu)	〃	512GB
		p16　(16cpu)	〃	1024GB
		p32　(32cpu)	〃	1024GB×2
		p64　(64cpu)	〃	1024GB×4
並列コンピュータ	会話型	(4 並列)	1 時間	8GB
	バッチ処理	as　(非並列)	無制限	16GB
		am　(Marc 専用)	〃	16GB
		am2　(Marc 専用)	〃	128GB
		a8　(8 並列)	〃	128GB
		a16　(16 並列)	〃	256GB
		a32　(32 並列)	〃	512GB

目次

東北大学サイバーサイエンスセンター
大規模科学計算システム広報 Vol.44 No.1 2011−1

PRINTED WITH SOY INK
再生紙と大豆の油を主成分とした
「ソイ・インキ」を使用しています。