

シームレス高生産・高性能 プログラミング環境

東京大学

石川裕、片桐孝洋

筑波大学

佐藤三久

京都大学

中島浩

T2K Open Supercomputer Alliance

目次

- 全体概要と研究開発体制
- 進捗状況
- 目標達成状況
- 中間評価内容と回答 & その後の取り組み
- 成果
- 成果の利活用
- 独創性・優位性
- 人材育成
- 成果普及活動
- 今後の展望
- まとめ

全体概要と研究開発体制

PCクラスタから大学情報基盤センター等に設置されているスパコンまで、ユーザに対するシームレスなプログラミング環境を提供

- 高性能並列プログラミング言語処理系
 - 逐次プログラムからシームレスに並列化および高性能化を支援する並列実行モデルの確立とそれに基づく並列言語コンパイラの開発
- 高生産並列スクリプト言語
 - 最適パラメータ探索など粗粒度の大規模な階層的並列処理を、簡便かつ柔軟に記述可能で処理効率に優れたスクリプト言語とその処理系の開発
- 高生産・高可搬性ライブラリの開発
 - 自動チューニング(AT)機構を含む数値計算ライブラリの開発
 - PCクラスタでも基盤センタースパコン(1万規模CPU)でも単一実行時環境を提供するSingle Runtime Environment Image環境の提供

高性能並列プログラミング言語処理系の開発

筑波大学

東京大学

次世代プログラミング言語仕様検討会
(主査: 佐藤@筑波大)
NEC, 富士通, 日立
JAXA, JAMSTEC,
核融合研, 筑波大,
東大, 京大, 九大

高生産並列スクリプト言語の開発

京都大学

普及活動
PCクラスタ
コンソーシアム
(会長: 石川@東大)
富士通, 日立, NEC
など27社・機関

高効率・高可搬性ライブラリの開発

東京大学

富士通研究所

日立中央研究所

目標：自動チューニング機構を含む数値計算ライブラリを開発し、数値計算コード部分のチューニングに関わる開発時間をなくす

- 汎用的なAT機能をまとめたライブラリであるOpenATLibを開発
- OpenATLibを利用したAT機能付き数値計算ライブラリXabclibを開発
- 新しいAT機能として、実行速度、メモリ量、および、演算精度のチューニングを行うことができる「数値計算ポリシ」機能を実現
- 本ライブラリの利用により、数値計算コード部分のチューニング時間削減が達成
- 実用ソフトウェアの有効性評価のため、プラズマシミュレーションで使われる非線形MHDシミュレーションコードにXabclibを適用し性能評価を行った。その結果、従来ライブラリでは実現できない高速化が達成可能なことを確かめた。

目標達成状況：単一実行時環境に関する研究

目標：PCクラスタでも基盤センタースパコン(1万規模CPU)でも単一実行時環境を提供するSingle Runtime Environment Image環境を提供すること

- ファイルシステム、通信ライブラリ、バッチジョブスクリプトの3つのサブ課題を設定し研究開発
- Xruntimeと呼ばれるソフトウェアパッケージとしてまとめた
- 筑波大、東大、京大の3大学が有するスパコン上で、同一バイナリかつジョブスクリプトを再利用できる環境が達成
- Xruntimeを使うことにより、計算機アーキテクチャが同じでも通信ライブラリやバッチジョブシステムが異なっても、同一バイナリかつジョブスクリプトを再利用できるようになった。XruntimeはPCクラスタでも稼働する。

- 目標：大規模並列システムでのプログラミングを助け生産性を向上させるために、既存言語を指示文により拡張した並列プログラミング言語を開発し、標準化を目指す
- 大学、メーカ、研究所からのメンバからなる言語仕様検討委員会を組織し、並列プログラミング言語XcalableMPの仕様を策定
- XcalableMPの指示文を使えるC言語およびFortran言語のコンパイラを開発・公開
- MPIを使った実行時システムを用いるため、筑波大、東大、京大の3大学が有するスパコンを含む多様なスパコンで動作可能
- ベンチマークプログラムでの評価を行い、HPCの著名なSC国際会議で行われるHPC言語に関するベンチマークHPCC Class2部門で、SC09, SC10に渡ってHonorable Mention賞を受賞
- マルチコアへの対応、並列IO、MPIライブラリのインタフェースなどの拡張を実施

- 目標：最適パラメータ探索など粗粒度の大規模な階層的並列処理を、簡便かつ柔軟に記述可能で処理効率に優れたスクリプト言語とその処理系を開発
- スクリプト言語Perlを、パラメータスイープ等の粗粒度の並列処理を簡便に書けるように拡張した並列スクリプト言語Xcryptを開発
- 二分法、モンテカルロ法、樹状探索などによる並列最適パラメータ探索を簡単に実行するためのXcryptライブラリを開発
- dry run機能や実行状態保存機能を開発することで、スクリプトのデバッグや故障時からの復帰を容易にした
- 筑波大、東大、京大の3大学が有するスパコンを含む多様なスパコンで同一のスクリプトを動かすことが可能
- プラズマシミュレーションの最適パラメータ探索や制約充足問題、電磁界シミュレーションプログラムの性能チューニング等の実アプリケーションにXcryptを適用することで、これらの機能の有用性の実証も行った

中間評価内容と回答 & その後の取り組み(1/4)

- 成果の利活用拡大のためには、各アプリケーションに対応したチューニングを行うなど、ユーザーニーズを研究開発に反映する枠組みが期待される。
 - プログラミングコンテストの開催
 - 開発した自動チューニング付き数値計算ライブラリXabclib、並列プログラミング言語XcalableMP、高生産並列スクリプト言語Xcryptを用いたプログラミングコンテストを開催した。XcalableMPの課題はLinpackであったが、分散機能の利用を工夫したXcalableMPプログラムがあり、興味ぶかい試みが見られた。Xcryptのコンテストにおいては、実用レベルに近いHigh Performance Linpackの性能チューニングツールが学部学生から提出され、Xcryptの実用性、記述容易性が実証された。
 - アプリケーション開発者との共同研究
 - AT機能付き数値計算ライブラリXabclibは、学際大規模情報基盤共同利用・共同研究拠点(JHPCN)の採択課題において、H22年～H23年度、核融合科学研究所のプラズマシミュレーション研究に利用された。その結果、ATしない機能、簡素な行列設定ルーチン、チューニング効果をまとめたログ生成機能の要求があった。最終版にそれらの機能を組み込んだ。また、H24年度課題として、防衛大との共同研究で利用される予定である。東京大学および北海道大学の情報基盤センターにおいて、H24年度からスパコンユーザに公開される予定であり、本課題終了後も引き続きユーザーニーズの反映を行う。またXcryptもJHPCNの採択課題「原子衝突による材料科学のための大規模シミュレーション基盤」において、シリコン(Si)の集束イオンビーム加工シミュレーションのポストプロセス処理(透過型電子顕微鏡イメージ生成)を、多数のシミュレーションインスタンスに対して簡便に実施するツールとして活用された。

中間評価内容と回答 & その後の取り組み(2/4)

- また、各サブテーマ間の連携の明確化や、次世代スーパーコンピュータと密に連携した取り組みも重要である。
 - 当初より、各サブテーマは独立できるようにテーマ設定されている。4年間という短い研究開発期間に実用化、人材育成を行うには、サブテーマ間で依存関係を作ることは危険だからである。当初より、開発したソフトウェア群をひとつにまとめ、PCクラスタコンソーシアムと連携して、配布 & 利用促進のためのチュートリアル開催などを行ってきている。
 - 次世代スーパーコンピュータ「京」との連携については、本課題代表者ならびに筑波大分担研究代表者は理研AICSのチームリーダーとして「京」の高度化研究開発に取り組みだした。
 - 東京大学情報基盤センターは「京」の商用版であるFX-10が2012年4月から稼働している。本課題で開発された成果は「京」およびその商用版FX-10に移植改良されながら一般に公開していく。

中間評価内容と回答 & その後の取り組み(3/4)

- 情報基盤センターを中心とした人材育成が行われているものの、メーカーや他機関との交流、講習会やセミナー等を通じた若手研究者の育成に一層の取組が必要である。
 - 平成22年度、平成23年度と講習会やプログラミングコンテストを開催してきている。また、最先端研究基盤事業における「e-サイエンス実現のためのシステム統合・連携ソフトウェアの高度利用促進」のなかで若手・女性研究者支援を行いワークショップも開催してきた。
 - また、平成23年度よりPCクラスタコンソーシアムにXMP規格部会およびシステムソフトウェア技術部会を設立し本プロジェクト終了後も引き続き開発したプログラミング言語の規格および普及を進める体制を構築した。

中間評価内容と回答 & その後の取り組み(4/4)

- 大学院生の取り込みなど、若い研究者を一層育てるため取組を組織的に行うことが必要である。
 - 本研究課題は、3大学の3つの研究室で行なわれている。国立大学法人系の研究室に来る大学院生の数は毎年数人程度であり、そもそも人数が少ない。また、中間評価後の残された2年間で、新たに学生らに本課題の研究テーマを直接割り当てることは、実用化のことを考えると難しい。上記、プログラミングコンテストやワークショップを開催するなどして、間接的に若い研究者を育てるアプローチを取った。
- 独創性や優位性に関する海外比較や性能向上についての数値的指標を提示するなど、研究開発成果の効果的なアピールのための一層の工夫が望まれる。
 - 全てのサブテーマにおいて性能評価が実施され、国内外会議で成果発表を行った。

成果：単一実行時環境Xruntime

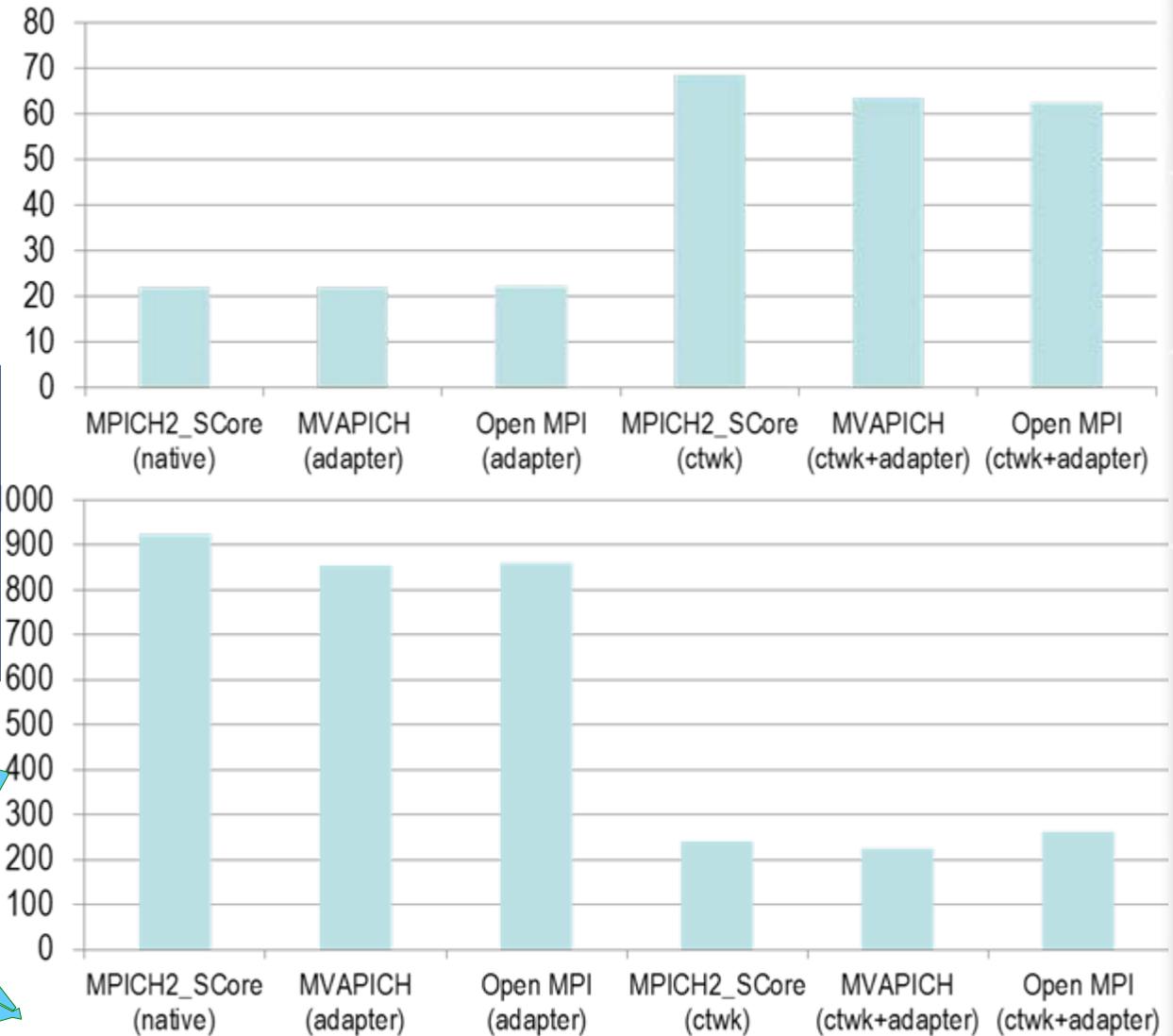
目標：PCクラスタから基盤センターで運用されているスパコン(1万CPU規模)まで、シームレスにユーザが利用できる単一実行時環境を実現する。

- ファイルシステムおよびステー징
 - PCクラスタから基盤センターマシンまで、ファイルI/Oを効率よくアクセス可能とするミドルウェア、CatwalkおよびSTG、ファイルキャッシュシステムpdCacheを開発
 - Catwalkは、アルゴンヌ国立研究所で開発保守されているMPICH2に統合される予定
- 通信ライブラリの研究
 - MPI通信ライブラリを利用したアプリケーションプログラムのバイナリ可搬性を提供するMPI-Adapterと呼ばれるライブラリを開発
- バッチジョブスクリプト
 - 汎用バッチジョブシステムXBSを開発
- これらソフトウェア群をXruntimeと呼ぶ
- Xruntimeは、PCクラスタコンソーシアムが配布しているSCoreパッケージに同梱し公開
- Xruntimeの有効性を実証(次ページ)

成果：Xruntimeの有効性実証

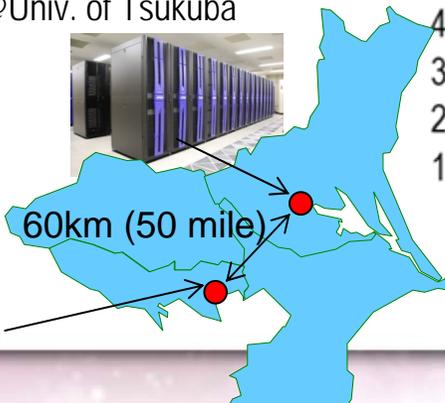
- 1) 東大PCクラスタ上でアプリケーションをコンパイル
- 2) コンパイルされたアプリケーションプログラム(バイナリ)とPCクラスタ上のバッチジョブスクリプトを使いユーザの開発環境上から筑波大のT2Kオープンスパコンにジョブを投入する
- 3) 筑波大のスパコンで当該アプリケーションが実行され、ユーザが利用している開発環境上に実行結果のファイルが生成される。

An Evaluation Result using NAS parallel benchmark BT-IO Class B



従来システムでは、通信ライブラリ実装およびバッチジョブシステムが違う環境では同一バイナリが実行できず、また、一つのバッチジョブスクリプトも利用できなかった。さらに、従来システムでは、ファイルシステムの運用の違いによりプログラムを書きなおす必要も生じていた。Xruntimeはこれら問題を解決した

T2K open supercomputer @Univ. of Tsukuba



Univ. of Tokyo

成果：自動チューニング付き数値計算ライブラリ

目標：汎用的な自動チューニング機能のためのインタフェースを実現する。
並列計算機上で自動チューニング機能の有効性を実証する。

課題

- 汎用的な自動チューニング
インタフェースの開発
 - 実行時に得られる情報を利用した自動チューニングの方式研究
 - 汎用性のある自動チューニングのインターフェース開発
- 自動チューニング機能付き
数値計算ライブラリ開発
 - 独自開発の汎用インターフェースを用いた、自動チューニング機能付き数値計算ライブラリ開発
 - 並列計算機において、自動チューニングの効果を検証

• 成果

1. **新しい自動チューニング機能の開発**
 - 数値計算ポリシー(速度、演算精度、メモリ量)
 - 疎行列-ベクトル積方式Branchless Segmented Scan (BSS)
 - 前処理方式と数値解法の自動選択機能
2. **新しいソルバ実装の開発**
 - スレッドセーフ機能
 - BiCGStab法(連立一次方程式の解法)
: 伊藤の前処理実装
 - Arnordi法(固有値問題の解法): 複素ベクトルのリスタート時の実数ベクトル化
3. **実用アプリケーションへの適用評価**
 - 核融合研究所のプラズマ解析MHDコード
 - 国内・国外のライブラリとの性能評価
 - **20倍以上の高速化例、高い収束率を達成**

完全陰解法を用いた非線形MHDシミュレーションコードへの適用例

佐藤雅彦氏（核融合科学研究所）との共同研究

MHD方程式

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho \mathbf{v})$$

$$\rho \frac{\partial \mathbf{v}}{\partial t} = -\rho \boldsymbol{\omega} \times \mathbf{v} - \rho \nabla \left(\frac{v^2}{2} \right) - \nabla p + \mathbf{j} \times \mathbf{B}$$

$$+ \frac{4}{3} \nabla [\nu \rho (\nabla \cdot \mathbf{v})] - \nabla \times (\nu \rho \boldsymbol{\omega})$$

$$\frac{\partial \mathbf{B}}{\partial t} = -\nabla \times \mathbf{E}$$

$$\frac{\partial p}{\partial t} = -\nabla \cdot (p \mathbf{v}) - (\gamma - 1) p \nabla \cdot \mathbf{v}$$

$$+ (\gamma - 1) [\nu \rho \boldsymbol{\omega}^2 + \frac{4}{3} \nu \rho (\nabla \cdot \mathbf{v})^2]$$

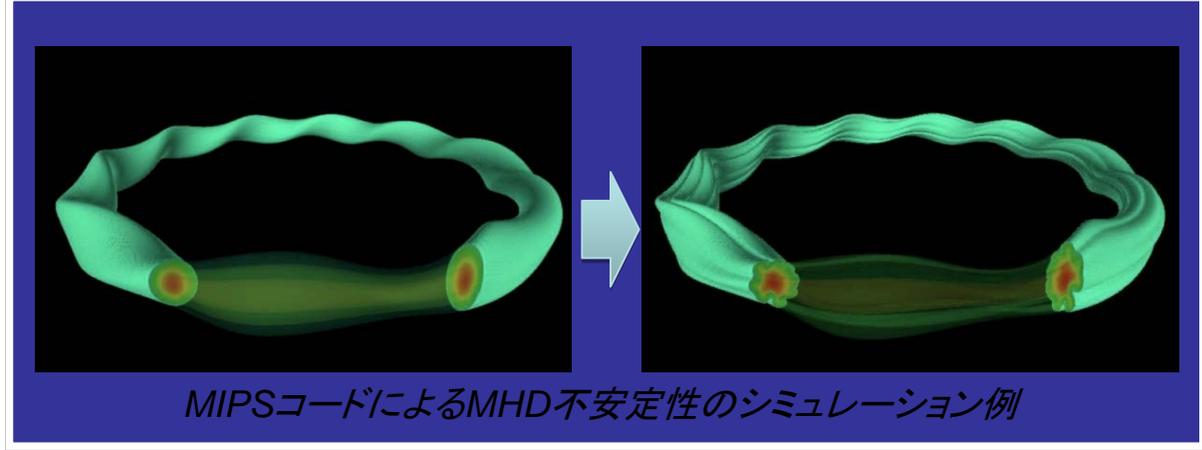
$$+ \eta \mathbf{j} \cdot (\mathbf{j} - \mathbf{j}_{eq})$$

$$\mathbf{E} = -\nabla \phi - \mathbf{v} \times \mathbf{B} + \eta (\mathbf{j} - \mathbf{j}_{eq})$$

$$\mathbf{j} = \frac{1}{\mu_0} \nabla \times \mathbf{B}$$

$$\boldsymbol{\omega} = \nabla \times \mathbf{v}$$

ρ : density, \mathbf{v} : velocity
 \mathbf{B} : magnetic field, p : pressure



MIPSコードによるMHD不安定性のシミュレーション例

現状のMIPSコードでは陽的な時間積分法を用いているため、クーラン条件により時間刻み幅が制限される。このため、現実的なパラメータでのシミュレーションを行うと非常に多くの計算時間を要する。

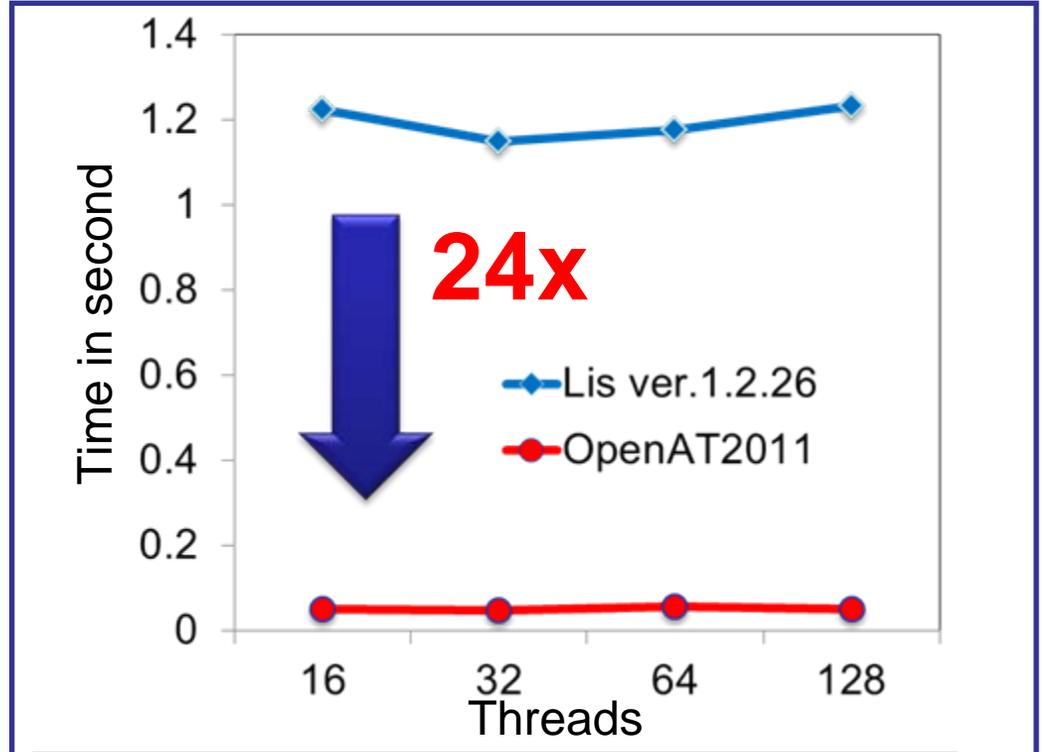
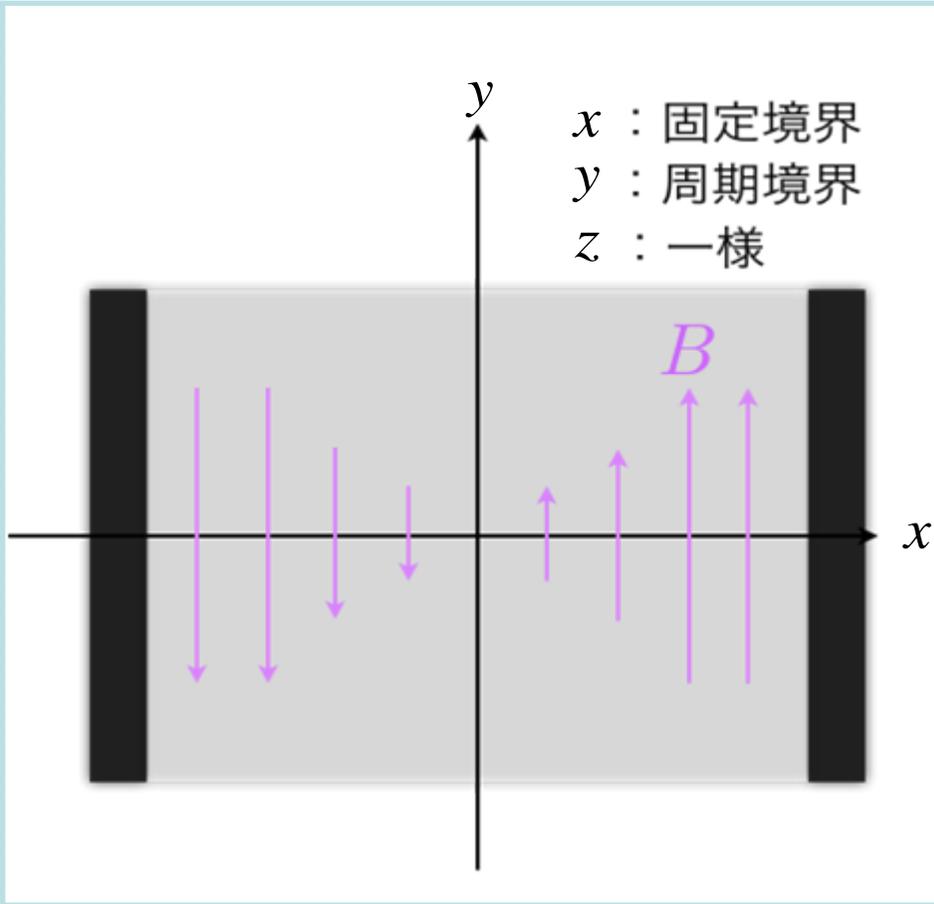


陰解法化によりクーラン条件を緩和して、時間刻み幅を大きく取ることによって計算時間の短縮を図る。



陰解法コードの高速化では、大規模連立一次方程式の高速解法の開発が不可欠であり、自動チューニング機能付きライブラリXabclibにより高速化を行う。

2次元モデルでの検証 (HITACHI SR16000/VL1)



Lisと比較して約24倍の高速化

- ソルバ: GMRES(m)
- リスタート周期: デフォルト: 40(Lis), AT(xabclib)
- 前処理: なし
- ソルバ内の反復回数: 10~11回(Lis), 6回(OpenATLib)
- 要求誤差: 1.0d-12
- N= 153,984 NNZ=3,648,064

目標：既存言語を指示文により拡張し、これからの大規模並列システム（分散メモリシステムと共有メモリノード）でのプログラミングを助け、生産性を向上させる並列プログラミング言語を設計・開発する

◆ 並列言語 XcalableMPの策定・開発

- ✓ XcalableMP仕様策定ワーキンググループを組織し、言語仕様を策定
- ✓ ベース言語(C, Fortran95)を指示文で並列拡張
- ✓ グローバルビューでは、データ並列プログラミングモデルとワークシェアによって、典型的な並列化をサポート
- ✓ ローカルビューとしてPGAS (Partitioned Global Address Space)を提供
- ✓ 通信モデルを明確化して、performance-awareな並列化を促進

◆ 成果

- ✓ SC09, SC10のHPCC Class2ベンチマークで Honorable Mention賞を受賞
- ✓ XcalableMP Specification ver 1.0を公表(2011.11)
- ✓ XcalableMP コンパイラ(レファレンス実装)を公開中

```
int array[YMAX][XMAX];
```

```
#pragma xmp nodes p(4)
#pragma xmp template t(YMAX)
#pragma xmp distribute t(block) on p
#pragma xmp align array[i][*] to t(i)
```

data distribution

```
main(){
  int i, j, res;
  res = 0;
```

add to the serial code : incremental parallelization

```
#pragma xmp loop on t(i) reduction(+:res)
for(i = 0; i < 10; i++){
  for(j = 0; j < 10; j++){
    array[i][j] = func(i, j);
    res += array[i][j];
  }
}
```

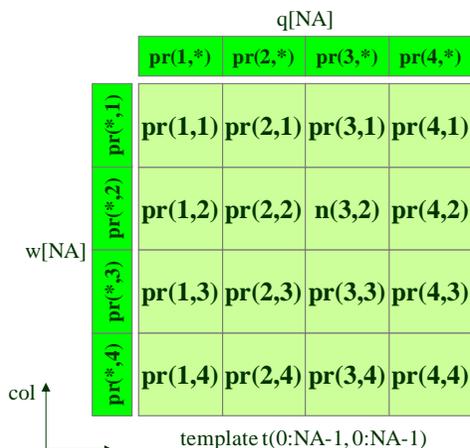
work sharing and data synchronization



成果 : XcalableMPの性能と生産性の検証

● NPB CGの例

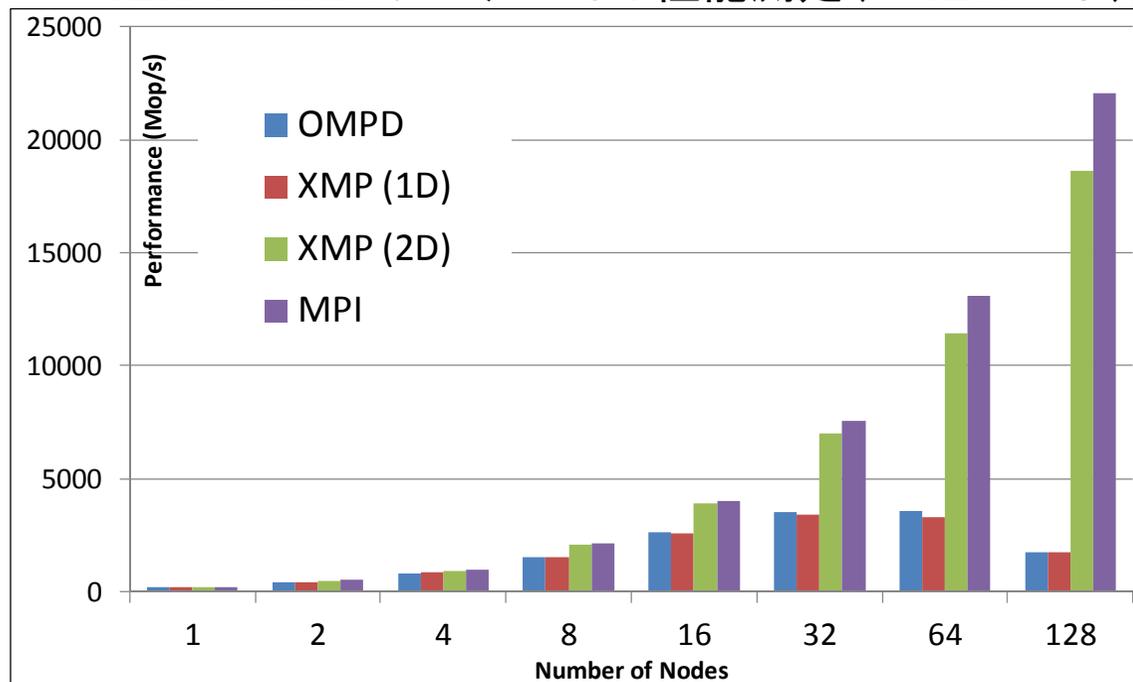
- XMPにより、1次元分割、2次元分割による並列化を行い、性能と生産性(行数)を比較
- 2次元分割をすることにより、MPI(2次元分割)と同等な性能を達成し、かつ、行数は逐次版に25%程度の並列記述で済む



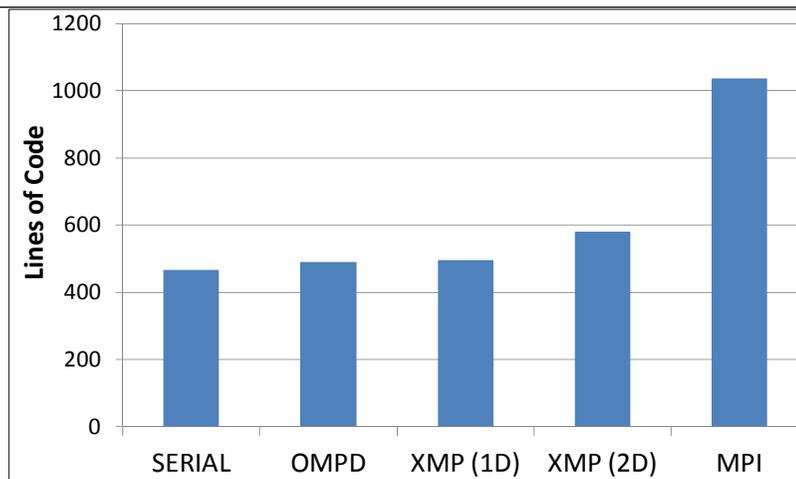
OMPDは、XMPの元になった言語のひとつ

```
#pragma xmp nodes on n(NPCOLS, NPROWS)
#pragma xmp template t(0:NA-1, 0:NA-1)
#pragma xmp distribute t(BLOCK, BLOCK) onto p
#pragma xmp align [i] with t(i,*) :: x,z,p,q,r
#pragma xmp align [i] with t(*,i) :: w
```

● T2K-Tsukubaシステムでの性能測定(～128ノード)



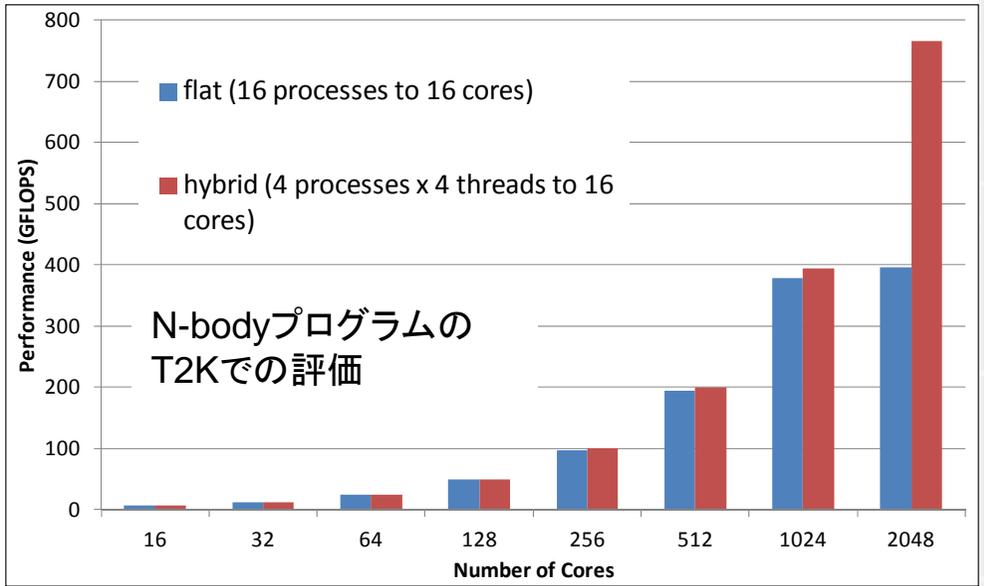
● CLOCによる行数の測定: 指示文を1行としてカウント



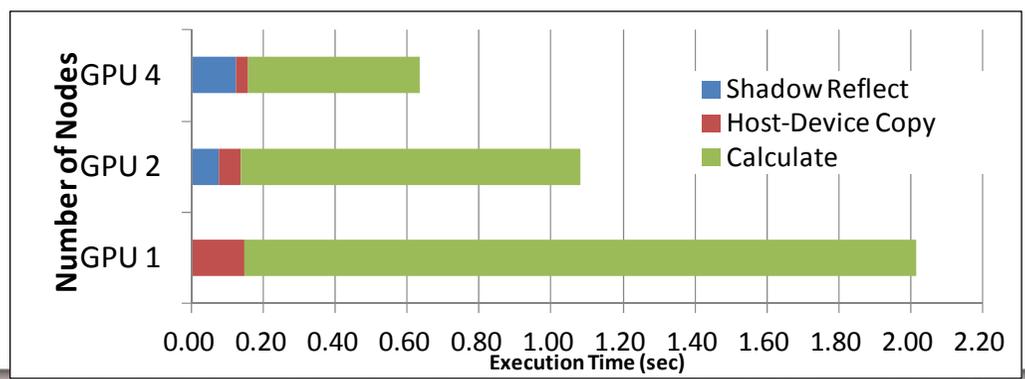
成果 : XcalableMPの拡張機能

- マルチコアノード向けの拡張: 並列ループ記述にノード内の並列化も記述
 - N-bodyプログラムでの評価: ノード数が大きいときには、flatに比べて、大幅な効率化が可能
- MPIなどで記述した並列ライブラリを呼び出すための並列ライブラリ・インターフェース
- XMP-IO: 分散配列の並列入出力機能
- XMP-dev: GPUなどの演算加速機構向けXMP拡張
 - GPU上のメモリに分散配列を定義し、袖領域の通信により、簡便に複雑な通信を記述が可能
 - Laplaceソルバーでの評価: 逐次プログラムに比べて、4GPU(Tesla C2050)で21.6倍の性能

```
#pragma xmp loop on t(i) threads firstprivate (x) num_threads (4)
for (int i = 0; i < N; i++) { out[i] = x * in[i]; }
```



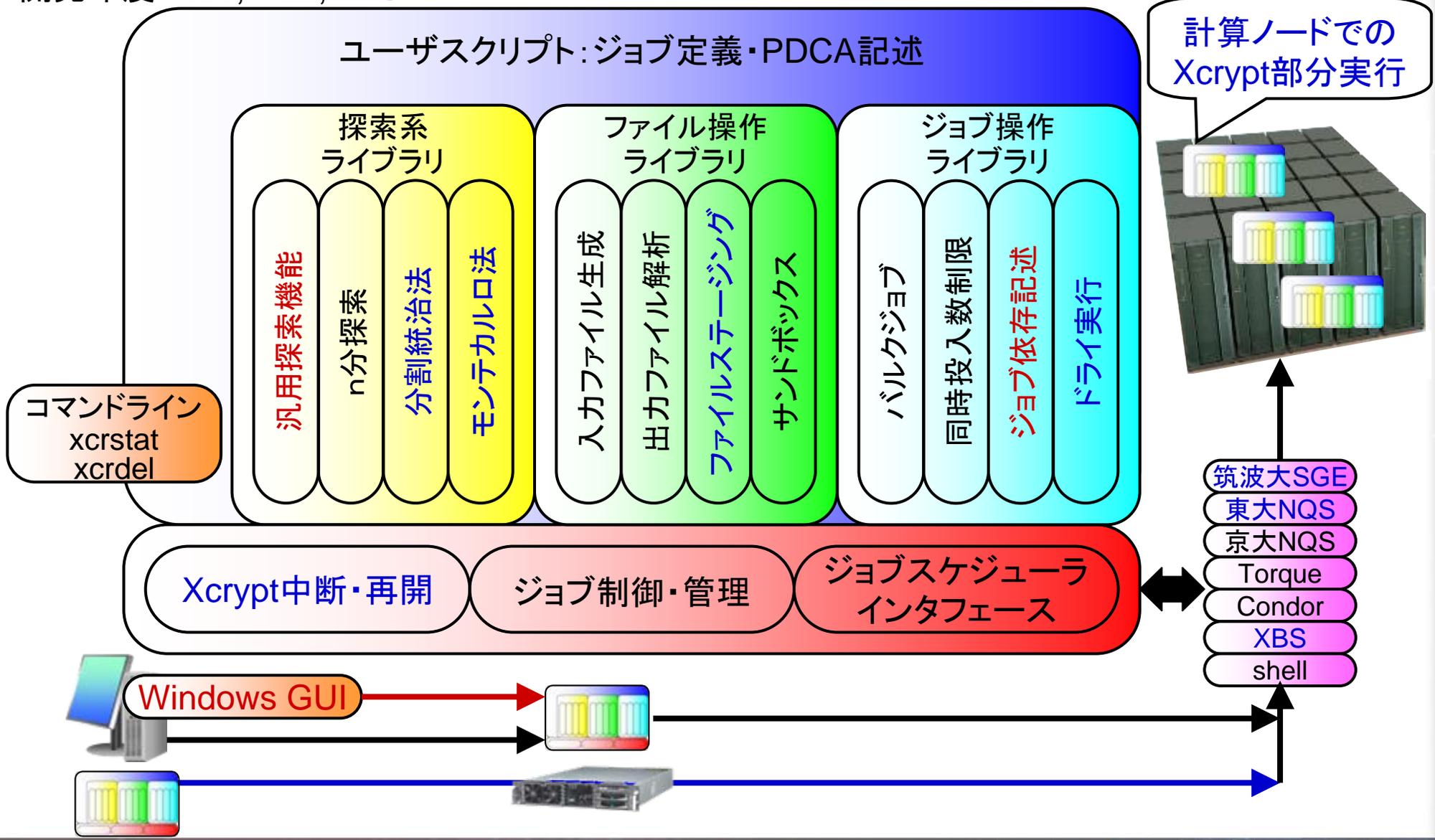
LaplaceソルバーのXMP-devのプログラムと実行時間(4K*4K)の内訳



```
#pragma xmp device replicate(u, uu)
{
  #pragma xmp device replicate_sync in (u)
  for (k = 0; k < ITER; k++) {
    #pragma xmp device reflect (u)
    #pragma xmp device loop (x, y) on t(x, y) threads
    for (y = 1; y < N-1; y++)
      for (x = 1; x < N-1; x++)
        uu[y][x] = (u[y-1][x] + u[y+1][x] +
                    u[y][x-1] + u[y][x+1]) / 4.0;
    #pragma xmp device loop (x, y) on t(x, y) threads
    for (y = 1; y < N-1; y++)
      for (x = 1; x < N-1; x++)
        u[y][x] = uu[y][x];
  }
  #pragma xmp device replicate_sync out (u)
} // #pragma xmp device replicate
```

成果：高生産並列スクリプト言語:Xcryptシステム

開発年度：H21, H22, H23



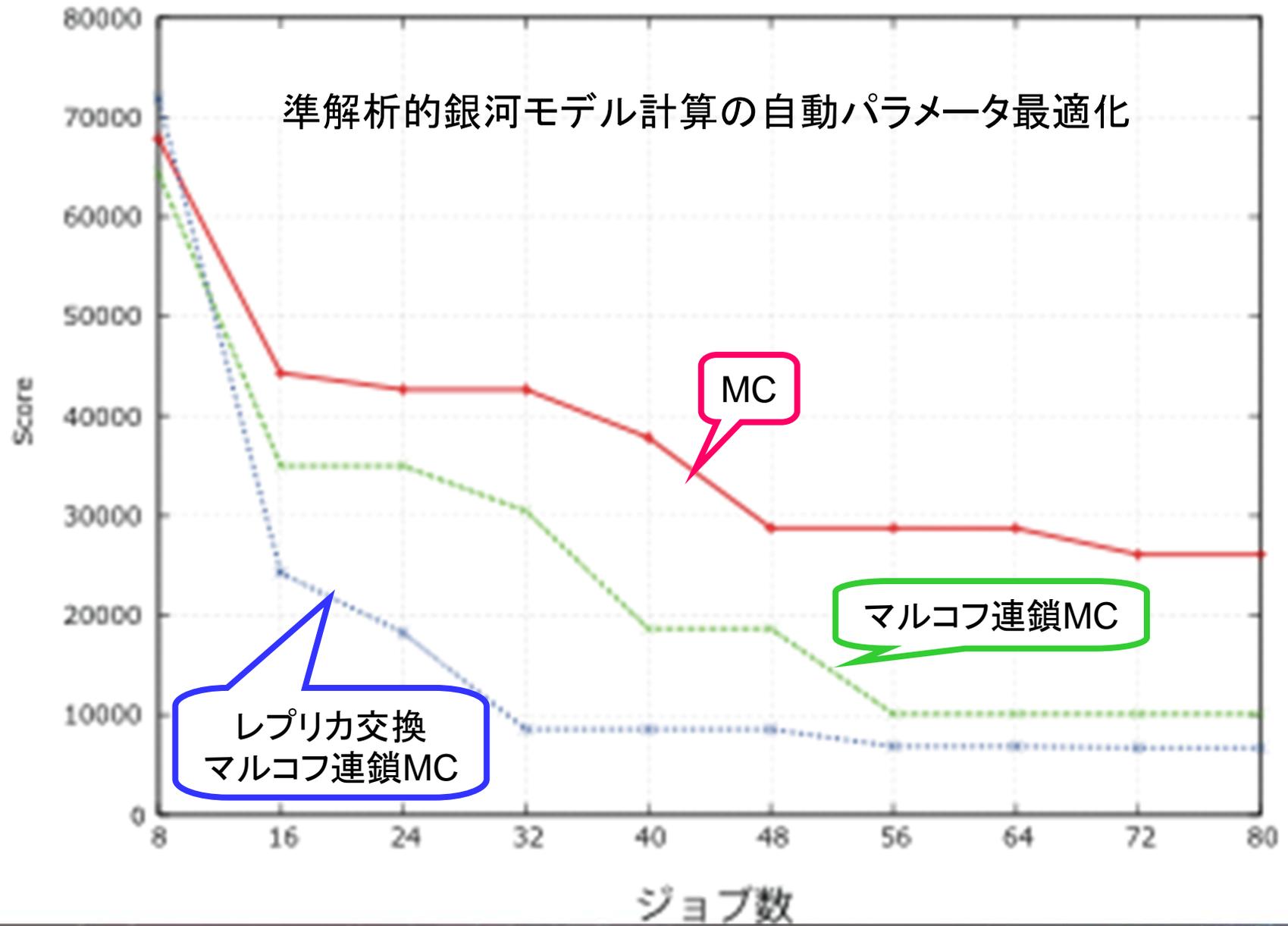
成果 : Xcrypt応用事例 (1/3)

- 宇宙プラズマシミュレーションのパラメータスイープ & 探索
 - Xcryptの機能検証に利用した最初期の応用事例
 - 2分探索・n分探索により人工衛星からの最適なプラズマ放出量を求解
- 分割統治型探索
 - 問題の分割法・結果の統合法の記述による並列分割統治探索の実現
 - 遺伝子相同性探索・論理式充足可能性問題に適用
- 準解析的銀河モデル計算の自動パラメータ最適化
 - 銀河形成モデルに基づく計算と観測銀河の近接度の尤度評価に基づく最適モデルパラメータ探索
 - モンテカルロ(MC)法・並列マルコフ連鎖MC法・レプリカ交換マルコフ連鎖MC法の適用
- HPLパラメータチューニング(プログラミングコンテスト課題)
 - HPLのパネルサイズ・プロセス配置をXcryptで自動チューニング
 - 学部学生により実用に近いレベルのチューニングツール開発

成果 : Xcrypt応用事例 (2/3)

- 材料科学シミュレーション(JHPCN研究課題)
 - シリコンの集束イオンビーム加工シミュレーション結果の電子顕微鏡像生成
 - 多数の原子衝突シミュレーションの並行&逐次実行(開発中)
- QCDシミュレーション自動化
 - 相転移が生じる温度パラメータ β をシミュレーション結果に基づき探索
 - 特定の β に対する複数の初期条件のシミュレーションを並行実行
- コンパイラオプション最適化
 - 所与のプログラムに適したコンパイラ最適化オプションを系統的に探索
 - ユーザ定義の探索空間・評価関数・候補生成等に基づく汎用探索機能で実装
- タイリングパラメータチューニング
 - FDTD法を空間方向・時間方向にタイリングする実装のタイルサイズチューニング
 - 問題の性質を利用して複雑な探索空間を効率的に走査

成果 : Xcrypt応用事例 (3/3)



- PCクラスタユーザに普及
 - 開発ソフトウェアはオープンソースとして公開し、PCクラスタコンソーシアムと連携
 - PCクラスタコンソーシアムは、2001年に発足した組織であり、富士通、日立、NECなどの日本のベンダなどから構成されている。
 - PCクラスタコンソーシアムメンバ企業が、開発したソフトウェアを基礎として先端的PCクラスタシステムを製品化することを念頭に開発段階から企業と連携
- センターユーザに普及
 - 最初の段階として、東大、筑波大、京大の各情報基盤センターのユーザに利用してもらうことを念頭。センターユーザは、センターが提供するソフトウェア環境の安定性に対する期待も高いため、現時点ではセンターユーザへの公開はしていないが、すでに共同研究を通じて複数のユーザグループの利用に供している。
 - 平成24年度初頭よりXabclibの一般公開を実施するほか、他のソフトウェアについても利用条件やサポート体制を整備しつつ、早期の公開を目指して活動を継続する。今後、ユーザとの共同研究を通して公開していく。
- PCクラスタから情報基盤センタースパコン、そして次世代スパコン「京」への橋渡しの環境として、さらに次世代スパコン「京」の高度化のための研究開発の基盤として使用していく

成果の利活用: 標準化

- 高効率・高可搬性ライブラリ: Xabclib, Xruntime
 - オープンソースとして提供し業界標準を目指す
- 並列プログラミング言語XcalableMP
 - 開発当初から、コミュニティの経験と意向をとりいれて仕様検討行う言語仕様検討委員会を組織し、仕様を策定。これにより、言語仕様の業界標準を目指す
 - XcalableMP処理系はオープンソースとして提供し、XcalableMPの普及を促進する
- 高生産並列スクリプト言語Xcrypt処理系
 - オープンソースとして提供し、階層型並列処理の業界標準とすることを旨す

独創性・優位性

- 自動チューニング機構に関する研究
 - － 実行時データに追従し最適な実装を選ぶAT機能により、従来の数値計算ライブラリでは達成できない高速化を実現することに独創性がある。
- 単一実行時環境に関する研究
 - － PCクラスタとセンタースパコンの実行時環境の違いを吸収し一つのバイナリでPCクラスタからセンタースパコン上までを実行できる環境を実現することに独創性がある。ソフトウェアベンダが欲していた環境であり、世界で初めて実現した。
- 高性能並列プログラミング言語処理系
 - － 既存言語(CとFortran95)を指示文により拡張することにより並列化機能を提供し、習得およびコード書き換えのコストを低減、生産性と実用性を向上。簡便な指示文で典型的なデータ並列の並列プログラムを可能とするとともに通信・同期などの事象をユーザが把握しやすい、明瞭な並列化モデルを提示することにより、性能改善のインタフェースを提供している。
- 高生産並列スクリプト言語
 - － 多数のバッチジョブの実行制御や入力生成・出力解析など階層型並列処理に特化した設計に独創性があり、既存のスクリプト言語やワークフローツールに対して記述容易性・柔軟性の面で優れている。

人材育成：継続的なるスパコン開発にむけた人材育成

- 背景
 - 2018～2019年に実現されるであろうエキサフロップス(1000ペタフロップス)級スパコンの基礎研究が国際的に開始
 - 8年後を見据えた研究は、現在の若手研究者が主導しなければならない
- 育てるべき人材像
 - 先端技術を研究開発できる能力
 - リーダシップ能力
 - 国際協力、産官学連携、計算科学や計算工学などの研究分野と連携
- 方策
 - 若手研究者が中心になり将来のスパコン実現のための研究課題整理ならびにロードマップを議論する場を提供すべく、平成22年東大、筑波大、京大、東工大が戦略的高性能計算システム開発に関するワークショップを立ち上げ、平成24年1月までに6回開催されている。文部科学省が取りまとめている「今後のHPC技術の研究開発を検討する作業部会」の計算機側の中核をなした。
- 東大
 - センター内予算で特任教員2名、本予算から特任助教1名、特任教授1名、大学院修士課程5名を雇用
 - 学生を米国アルゴンヌ国立研究所に2ヶ月間派遣(インターンシップ)
- 筑波大
 - 本予算から特定教員1名とRA1名を雇用
 - RA (Research Assistant) を米国Purdue大学に1ヶ月派遣
 - 並列プログラミング言語検討委員会には、若手の研究者を積極的に参加させ、標準化活動に参加させるとともに、先端研究者との交流を図った
- 京大
 - 本予算から特定教員(助教)2名と、学内定員・予算による助教1名の雇用
 - 若手主導による研究遂行

平成20年度

- 論文発表数: 18件
- T2Kシンポジウム開催: 10月24日(金)
- 国際会議 SC08研究展示: 11月17日(月)～20日(木)
- 情報処理学会主催HPCSシンポジウム ポスター発表: 1月22日(木)～23日(金)
- PCクラスタワークショップin大阪 プロジェクトにおける講演: 3月13日(金)
- 国際ワークショップWPSE 2009 (International Workshop on Peta-Scale Computing Programming Environment)開催:
3月25日(水)～26日(木)

平成21年度

- 論文発表数: 20件
- 国際会議 SC09研究展示
11月16日(月)～19日(木)
PCクラスタコンソーシアムからXruntime、Xabclib公開
- 国際ワークショップWPSE 2010 (International Workshop on Peta-Scale Computing Programming Environment)開催: 2月18日
- チュートリアル開催: 2月18日
PCクラスタコンソーシアムからXruntime (Catwalk, STG, MPI-Adapter)、Xabclib公開

平成22年度

- 論文発表数: 28件
- T2Kシンポジウム@東大: 7月12日
eScienceプロジェクト成果報告
パネルディスカッション
- チュートリアル開催: 10月22日, 10月29日
PCクラスタコンソーシアムからXruntime (Catwalk, STG)、Xabclib, Xcrypt公開
- プログラミングコンテスト開催: XMP, Xcrypt, Xabclib部門
- 国際会議 SC10研究展示
11月15日(月)～18日(木)
PCクラスタコンソーシアムからXruntime (Catwalk, STG)、Xabclib, Xcrypt公開

平成23年度

- 論文発表数: 17件
- T2Kシンポジウム@京都: 11月30日
eScienceプロジェクト成果報告
パネルディスカッション
- チュートリアル開催: 3月8日
XMP
- プログラミングコンテスト開催: XMP部門
- 国際会議 SC11研究展示
11月14日(月)～17日(木)
Xmp, Xcrypt, Xruntime, Xabclib公開 & T2Kブースにて配布
- 国際ワークショップWPSE 2012 (International Workshop on Peta-Scale Computing Programming Environment)開催:
2月27日(火)～28日(水)

今後の展望

- 成果物の展開
 - 3大学のスパコンならびに京コンピュータ上で使えるようにしていく
- ノウハウの利活用
 - 本課題で培ったノウハウを筑波大学および東京大学のT2Kスパコン後継機のための基本システムソフトウェア開発に反映する
- さらなる将来に向けての取り組み
 - さらにエクサスケールコンピューティング実現に向けた問題点を整理していく

まとめ

- 進捗状況
 - 計画通りに進行した
- 研究開発成果
 - 国内外の会議や論文誌も採択され、コミュニティからも注目されている
- 成果普及活動
 - 開発段階からPCクラスタコンソーシアムと連携し実用化・普及に向けた活動を行った
- 独創性・優位性
 - 自動チューニング機構、単一実行時環境、高性能並列プログラミング言語処理系、高生産並列スクリプト言語、それぞれに独創性・優位性を持つ
- 成果の利活用
 - PCクラスタユーザ、センターユーザに向けた普及を進めている
 - 言語仕様の標準化、開発したソフトウェアのオープンソース化による普及
 - PCクラスタコンソーシアムとの連携
- 人材育成
 - 先端技術を研究開発できる能力とリーダーシップ能力を兼ね備えた人材育成を目指して取り組んだ

補足資料

目標: PCクラスタから大学情報基盤センター等に設置されているスパコンまで、ユーザに対するシームレスなプログラミング環境を提供

- 高性能並列プログラミング言語処理系（筑波大学）
 - 逐次プログラムからシームレスに並列化・高性能化する並列言語コンパイラの開発
- 高生産並列スクリプト言語（京都大学）
 - 最適パラメータ探索など粗粒度大規模並列処理を、簡便に記述可能なスクリプト言語の開発
- 高生産・高可搬性ライブラリの開発（東京大学）
 - 自動チューニング(AT)機構を有する数値計算ライブラリの開発
 - PCクラスタでも基盤センタースパコン(1万規模CPU)でも単一実行時環境を提供するSingle Runtime Environment Image環境(バッチジョブ、ファイルシステム、MPI通信ライブラリ)の提供

成果: 対外発表、普及活動等

- 論文発表数: 18件(H20年度)、20件(H21年度)、28件(H22年度)、17件(H23年度)
- XcalableMP(XMP)、SC10, SC11 HPCベンチマーク Class2 (並列プログラミング言語部門)で、2年連続Honorable mention賞
- PCクラスタコンソーシアムから高生産・高可搬性ライブラリの一部を一般公開 & チュートリアル開催(2回)
- 国際会議SC08, SC09, SC10, SC11研究展示
- XcalableMP, Xcrypt, Xabclib/OpenATLibプログラミングコンテスト開催, XcalableMPプログラミングコンテスト開催 x 1
- PCクラスタコンソーシアム内にXcalableMP規格化部会設立(H23年6月)
- 「京」コンピュータでの展開開始

成果: 開発したプロトタイプソフトウェア

- Xcalable MP(XMP):** 並列プログラミング言語
- Xcrypt:** 並列スクリプト言語
- Xruntime:** 単一実行時環境
- Xabclib/OpenATLib:** 自動チューニング付ライブラリ

国際協力

- JST日仏戦略共同研究プログラムにてXMPおよびXabclibを利用した共同研究、JSPS多国間国際研究協力事業(G8RCI)にてXMPを利用した共同研究
- ファイルステージングCatwalk-ROMIOをアルゴンヌ国立研究所が開発維持しているMPICH2nに統合

Xabclib 測定条件

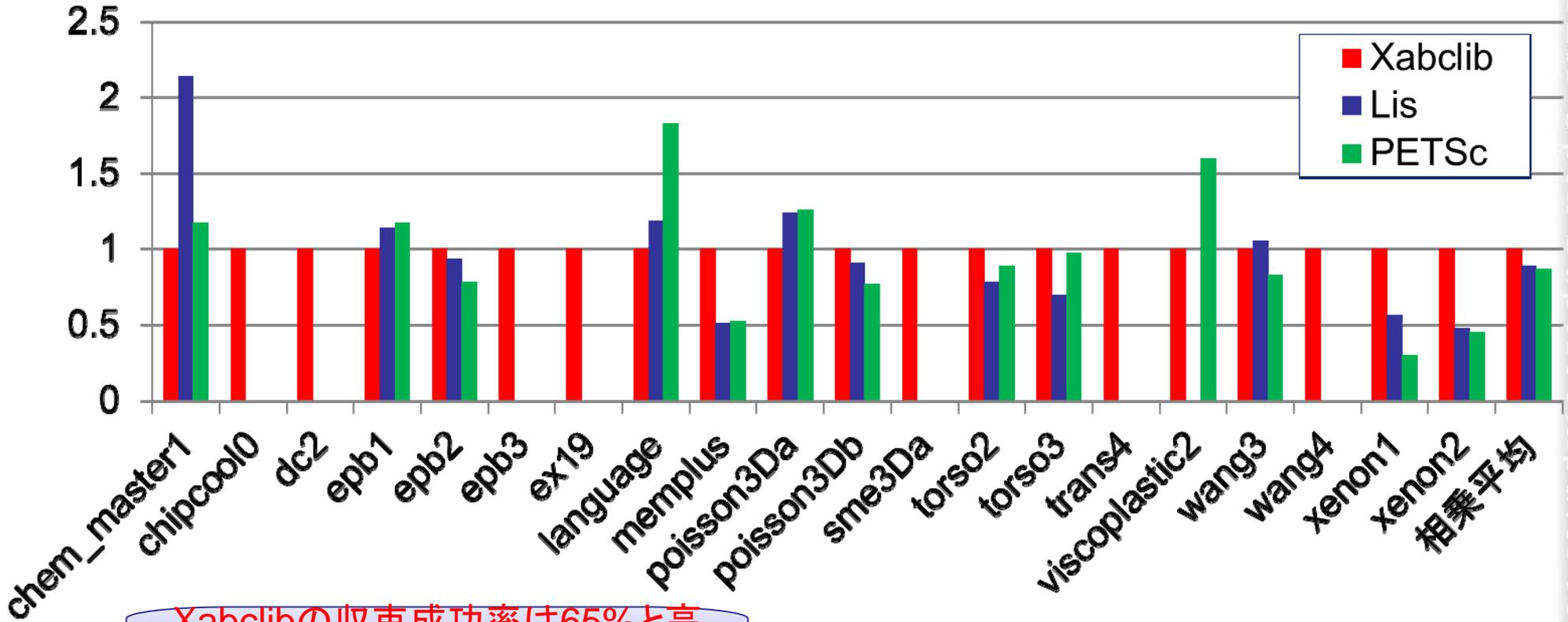
	Xabclib	Lis	PETSc
ポリシー	TIME (実行時間優先)		
解法	GMRES, BiCGSTAB	GMRES, BiCGSTAB	GMRES, BiCGSTAB
GMRES の リスタート周期	自動	40(デフォルト)	30(デフォルト)
前処理	無し, 自動	無し(デフォルト), ILU	無し, ILU(デフォルト)
許容残差	1.0D-8	1.0D-8	1.0D-8
CPU数	1, 16	1, 16	1
時間	600秒	600秒(強制打切り)	300秒(強制打切り)

GMRESのリスタート周期は各ライブラリのデフォルト値とした

PETScのマルチスレッド版は反復解法レベルでの対応は未完のため1CPUのみ

Xabclib GMRES法、1スレッド、前処理なし

Xabclibの速度を1をしたときの相対速度(大きいほど速い)



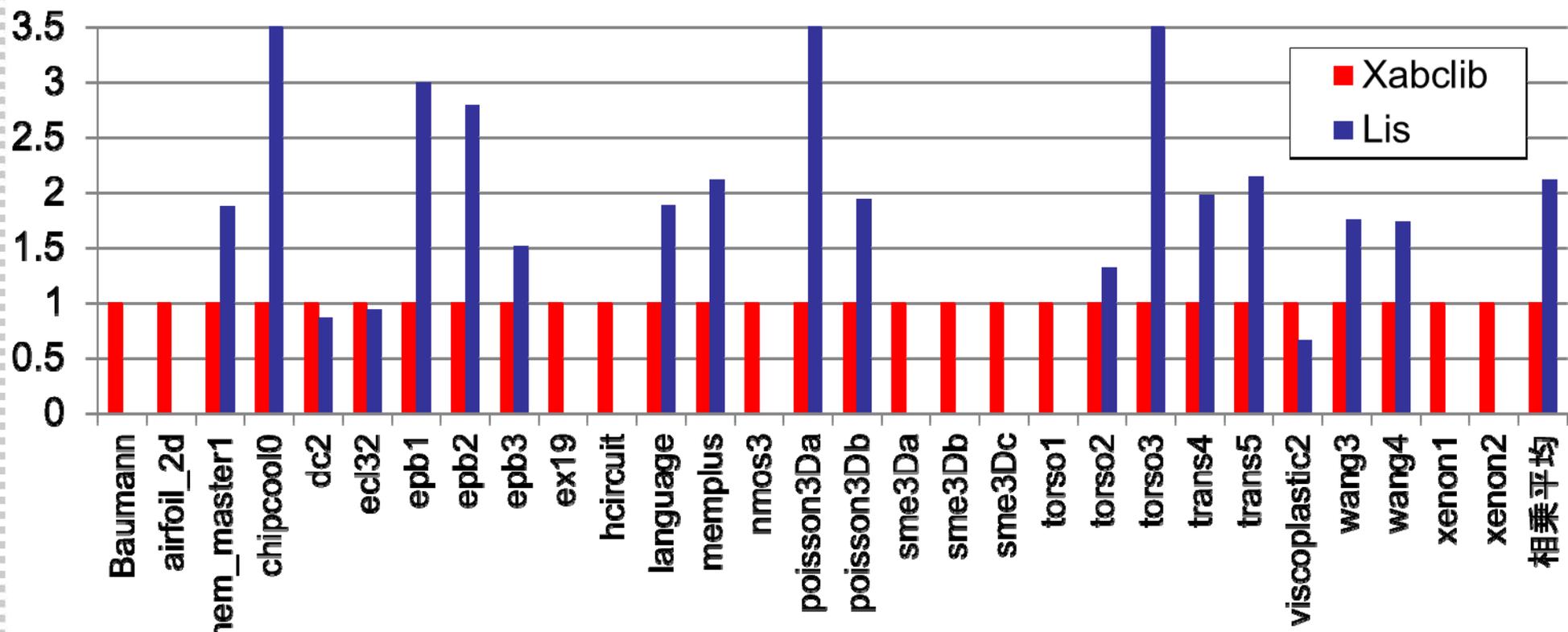
Xabclibの収束成功率は65%と高い

	成功数	失敗数
Xabclib	20	11
Lis	12	19
PETSc	13	18

Xabclibに対するLisの速度低下は平均0.89倍
 Xabclibに対するPETScの速度低下は平均0.87倍

Xabclib GMRES法、16スレッド、前処理あり

Xabclibの速度を1をしたときの相対速度(大きいほど速い)



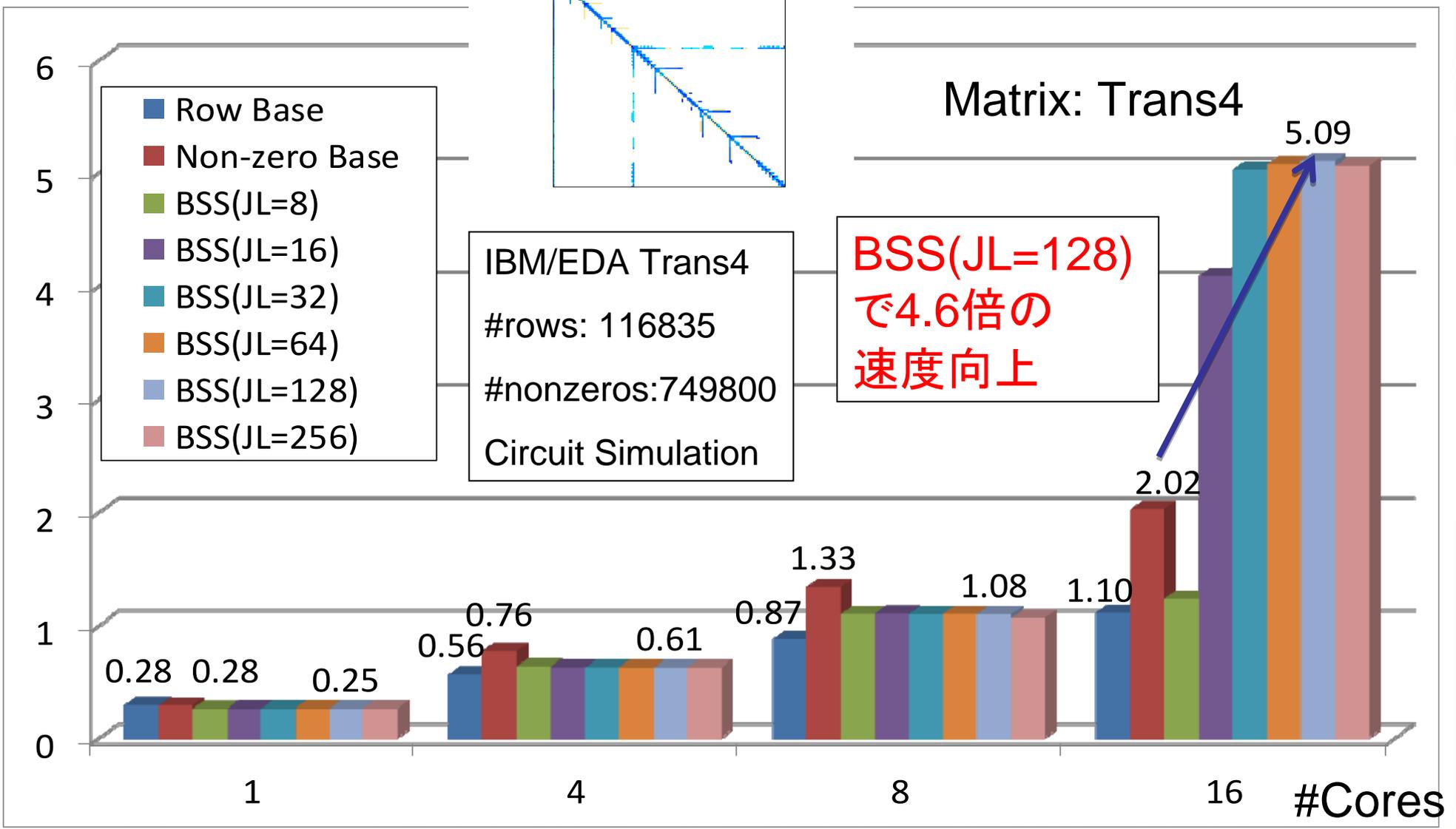
Xabclibの収束成功率は94%と高い

	成功数	失敗数
Xabclib	29	2
Lis	18	13

Xabclibに対するLisの速度向上は平均2.11倍

OpenATLib BSSの性能

[GFLOPS]



- 汎用探索ライブラリ
 - ユーザ定義情報
 - 探索パラメータ空間の定義、初期候補の生成
 - 評価対象ジョブの定義、ジョブ結果に基づく評価関数
 - 評価関数値に基づく次候補集合生成、並行実行する候補の選択
 - 最終結果の出力
 - ライブラリ機能
 - 探索木の逐次走査(深さ方向)・並行走査(幅方向)の管理
 - 評価対象ジョブの投入・終了検知
- Windows GUI
 - Xcrypt 実行環境設定
 - ワークフロー定義と xcr スクリプト自動生成、既存 xcr スクリプト編集
 - xcr スクリプト遠隔実行、実行状況・実行結果の確認

Xcrypt補足：H23機能拡張・強化

- Xcrypt 中断・再開
 - H22まで
 - Xcrypt の偶発的・意図的中断に備えてジョブ情報(実行済、未実行など)を保存
 - 再開時は保存したジョブ情報に基づき実行が必要なジョブだけを投入・実行
 - H23
 - 保存するジョブ情報にユーザ定義のジョブオブジェクトメンバー変数を追加
 - ジョブオブジェクトの状態に依存した実行シーケンスの再現が可能
- 出力ファイル解析高速化(一部開発中)
 - H22まで: 汎用的・高機能・耐巨大ファイルのパターン検索・抽出機能
 - H23: 比較的単純な検索を perl 機能等で直接実現して高速化
- 探索ライブラリ
 - H22 開発ライブラリ(分割統治法、モンテカルロ法)のAPI等を見直して再実装

Xcrypt補足：今後の展開

- 「京」への展開
 - 理研 AICS にて実施
 - 課題：ステージング機能の拡充(?), 「京」固有のジョブ実行パラメータ(?), ...
- 多言語対応
 - perl 以外のスクリプト(的)言語 (Ruby, Python, ...) からの Xcrypt 利用
 - サーバ(Xcrypt) / クライアント (Ruby, etc.) モデル + 言語間データ変換
 - Ruby 対応プロトタイプ開発中