

シミュレーションが 未来をひらく



固有値ソルバの現状と ポストペタスケール環境に向けた展望

○廣田 悠輔, 深谷 猛, 今村 俊幸
(理研AICS, JST CREST)

2015年3月20日

Cyber HPC Symposium

大阪大学 銀杏会館

(主催:大阪大学 サイバーメディアセンター)



研究チームの紹介

理研AICS 大規模並列数値計算技術研究チーム

- メンバー
 - チームリーダー: 今村俊幸
 - 特別研究員: 大井祥栄, 深谷猛, 廣田悠輔, 椋木大地
 - 客員研究員: 高橋大介, Franz FRANCHETTI, 岡本吉央
- チームミッション: 「大規模・高並列・高性能な数値計算ソフトウェアライブラリの研究開発」
- 主な研究対象
 - 高性能固有値ソルバ(標準固有値, 一般化固有値)
 - 高性能FFTライブラリ
 - 対故障性を考慮した数値計算アルゴリズム
 - 高精度数値計算ライブラリ
 - etc.

CREST課題(ポストペタ領域)の紹介

CREST プロジェクト(ポストペタ領域):

「ポストペタスケールに対応した階層モデルによる超並列固有値解析エンジンの開発」
(代表: 櫻井鉄也)

- ポストペタスケール時代の階層的並列構造に対応した超並列固有値ソルバを開発.
- 従来の固有値ソルバの弱点であるスケーラビリティ, 対故障性に対応するため, 新しいアルゴリズムに基づいて開発する.
- 疎行列・密行列の両方のアルゴリズム, 疎密ハイブリッドアルゴリズム, 対故障性技術, メニーコア向け実装技術などの研究を行う.



今村グループ: 「密行列向け固有値計算アルゴリズム」を共同で研究

- ペタフロップス環境およびポストペタ環境向けのプログラム開発
- 再帰的アルゴリズムに基づく行列変換法の研究
- ペタフロップス環境での試験実装と実アプリを用いた評価
- GPU・メニーコア向け実装技術研究

- 固有値計算の概観
- 固有値ソルバ EigenExa
 - アルゴリズムと実装の概要
 - 京コンピュータでの性能評価
 - 詳細な性能分析 (on FX10)
- ベクトル計算機への展開
- おわりに

固有値計算の概観

取り扱う固有値問題

$$Ax_i = \lambda_i x_i, \quad A \in \mathbb{R}^{N \times N}$$

問題の性質や求める固有値の数などによる分類

- 行列 A : 対称(エルミート)性, 実行列/複素行列, 密行列/疎行列, 次数 etc.
- 求める固有値・固有ベクトルの数: すべての固有対, 最大/最小から数個 etc.

(※ 固有対: 固有値と対応する固有ベクトルの組)



分類に応じたアルゴリズムを選択

(例)

- 実対称密行列の全固有対を計算: 行列の直交変換＋分割統治法 etc.
- 非対称疎行列の一部固有対を計算: Arnoldi 法, Sakurai-Sugiura 法 etc.

固有値ソルバ開発のニーズ

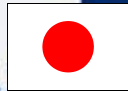
- 計算科学における重要性:
固有値ソルバは計算科学ソフトウェアにおける主要な部品の一つ
- 従来の固有値ソルバの問題:
大規模並列計算機におけるデファクトスタンダード ScaLAPACK
は1990年代の計算機向けの設計
(シングルコア, 少ないノード数, 相対的に高いメモリバンド幅)
→ 近年の計算機では性能を発揮できない

**現行およびポストペタスケール時代の並列計算機に適した
大規模固有値ソルバ開発が重要**

世界における密行列向け固有値ソルバ開発

ELPA

- 1step vs 2stepの議論: 1stepが高速の場合が多い
- 三重対角 $\Rightarrow$ 帯の変換部分は未だよい実装できず
...実装レベルでは困難か?
- B/Qでアセンブラチューニングの方向 & GPU化へ



DPLASMA

- PLASMA, MAGMAでの2stepスキーム
- 1nodeはスケーラブルに動作
- DAGタスクスケジューリング



EigenExa

- 新1stepスキーム採用
- 京において 2^{16} コアまでの動作確認・通信コストの認識
- 動作 & 通信モデル構築
- 階層化アルゴリズム、自動チューニングの取り込み
- GPU版も準備へ

ScaLAPACK ver 2.0.2

- 枠組みに大きな変化なし
- MPIがBLACSの標準に
- ルーチンの強化:
 - ✓ 非対称行列ソルバー(PDHSEQR)
 - ✓ 新ルーチンMRRR(PDSYEVR)

固有値ソルバ EigenExa

EigenExa

- 「京」向けの性能チューニングが施された大規模固有値計算ライブラリ
※ 「京」以外（x86 クラスタ, SX-ACE 等）でも利用できます！
- 2つの固有値計算ルーチン
 - 従来型の計算スキームを採用したルーチン `eigen_s`
 - 「京」向けに新規開発されたスキーム採用のルーチン `eigne_sx`

プロジェクトの歴史

- 初代地球シミュレータ版ソルバを開発。
 以後、約10年にわたり開発を継続（Eigen-K）。
 （一部は矢川CRESTの一環で開発）



Sorry photo is ES2



T2K
PC cluster



Exa-scale
System??

- 現在は「京」向けに開発（EigenExa）（JST CREST 2011/4—2016/03）
 ✓ 新しい計算スキームを採用したルーチンを開発

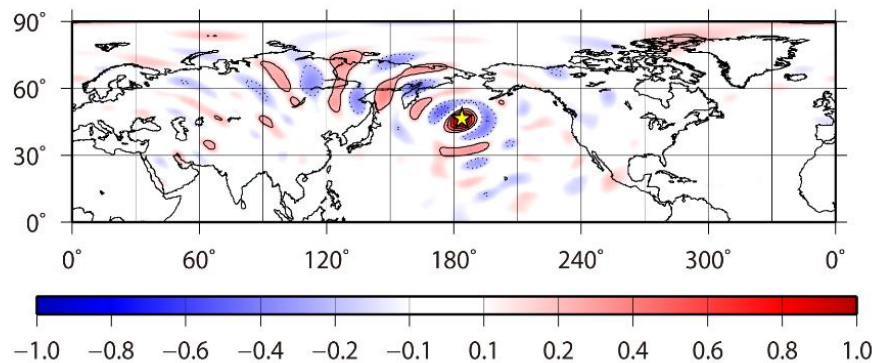
(再配布不可ページ)

応用例2: データ同化研究における活用

■ 「LETKF」による世界最大規模の全球大気のアンサンブルデータ同化

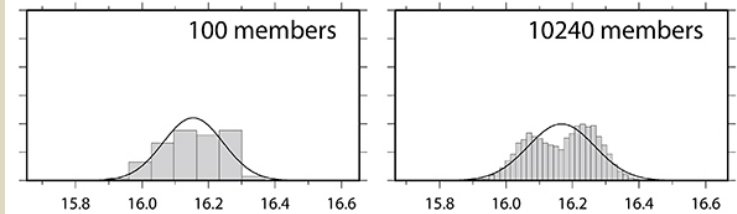
- アンサンブルデータ同化システム「LETKF」に、EigenExaを組み込むことで、1回の計算を125分から15分へ、**約8倍高速化**。
- さらに、EigenExaとの相乗効果により、「京」4,608ノード(約 590 TFLOPS)を使い、263 TFLOPS (理論ピーク性能比 で44%超)という**極めて高い実行効率を達成**。
- これにより、10,240個のアンサンブルで3週間分という**世界最大規模の全球大気のアンサンブルデータ同化に成功**。

10240 members w/o localization



アンサンブルデータ同化による水蒸気量の相関マップ

黄色い星の場所の水蒸気量の観測データに対する各地点の水蒸気量の相関係数を示す。アンサンブル数が増えることで統計ノイズが抑えられ、観測データが1万km遠方に及ぼす影響まで確認できた。



アンサンブルデータ同化による水蒸気量の誤差を表すヒストグラム

実線はガウス分布(正規分布)関数。10,240個のアンサンブル(右)により、100個のアンサンブル(左)では確認できなかった大気状態のばらつきの非ガウス性を初めて直接確認。

計算科学ソフトウェアへの導入

- Platypus QM/MM: 酵素反応機構などの生体高分子の精密解析ソフトウェア
- RSDFT: 結晶・界面・分子などの広範な物理系に対する密度汎関数法による電子状態計算ソフトウェア
- PHASE: 密度汎関数理論に基づく物質中の電子状態の計算ソフトウェア
- ELSEs: 大規模系の電子構造計算、分子動力学計算による様々なプロセスのシミュレーションソフトウェア
- NTChem: 数千原子分子系の第一原理電子状態計算や数百原子分子系の化学反応過程追跡計算ソフトウェア
- Rokko: 並列ライブラリ等を利用する大規模密行列、大規模疎行列の対角化を統一的に利用するためのライブラリ
- LETKF: 並列計算性能を重視したアンサンブルデータ同化手法
- POD: 多次元データから低次元成分を抽出する解析法

密行列向け固有値ソルバ EigenExa

◆ 実対称密行列の固有値問題

$$Ax_i = \lambda_i x_i, \quad A = A^\top \in \mathbb{R}^{N \times N}$$

全固有値 λ_i と全固有ベクトル x_i ($i = 1, \dots, N$) を計算

◆ 計算の手順

step 1) 順変換

$$C := Q^\top A Q, \quad Q^\top Q = I$$

step 2) 分割統治法

$$\text{solve } Cy_i = \lambda_i y_i$$

step 3) 逆変換

$$x_i := Q y_i \quad (i = 1, \dots, N)$$

複数のアプローチが存在

様々なアプローチ

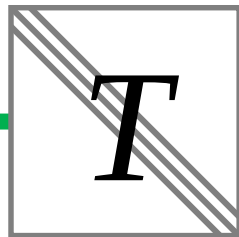
(従来の)1ステップ型

- ScaLAPACK
- eigen_s (EigenExa)

(実対称行列)



(三重対角行列)

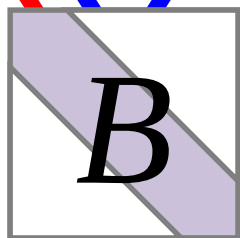


T の
固有対

A の
固有対

2ステップ型

- ELPA
- (DPLASMA)



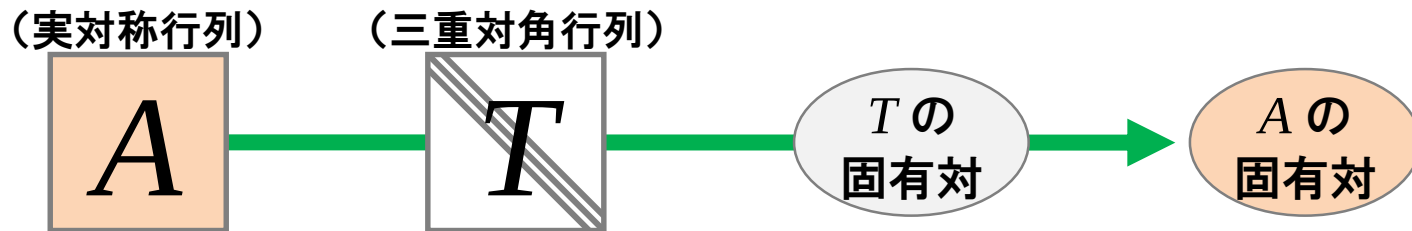
(帯行列)

B の
固有対

- eigen_sx (EigenExa)

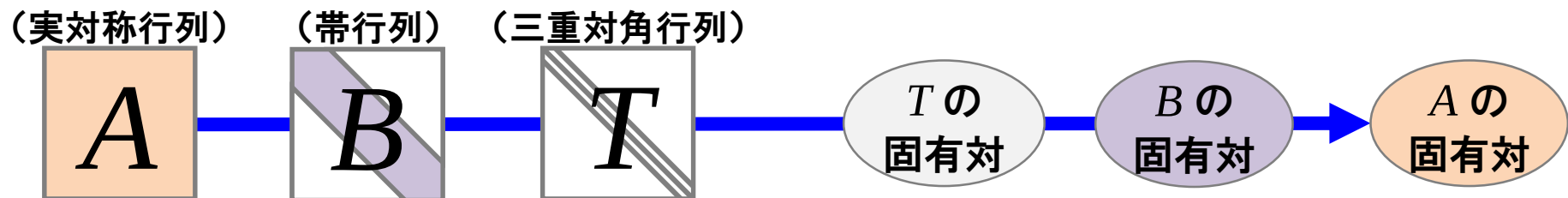
(新しい)1ステップ型

従来の1ステップ型(ScaLAPACK等)



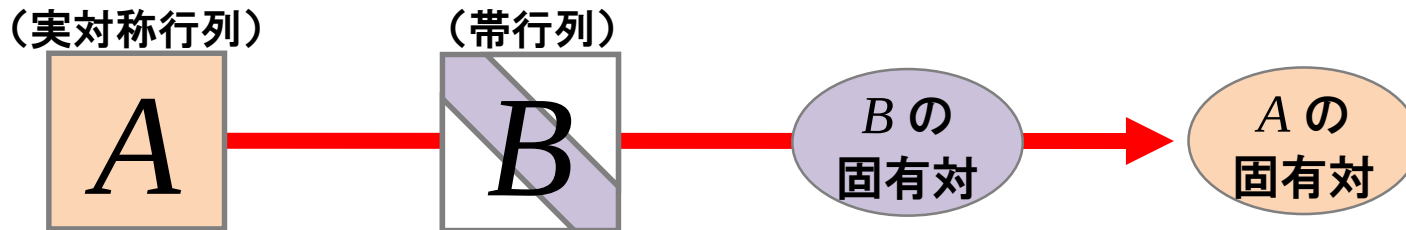
- 数値計算の教科書の記載されているアルゴリズム
- ハウスホルダー変換による三重対角化がボトルネック
 - メモリバンド幅に律速
 - 素朴に実装すると, ほぼ全てが行列ベクトル積相当の演算
 - 工夫を施しても, 半分は行列ベクトル積が残る
 - 逐次計算で**集団通信が頻発**し, 隠蔽が困難

2ステップ型(ELPA, DPLASMA)



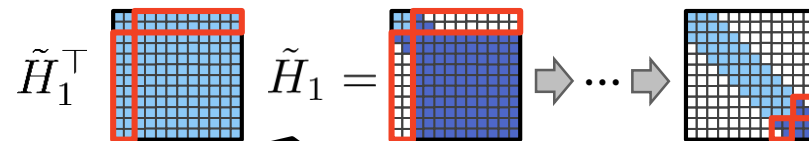
- 2ステップの三重対角化を採用
 - 帯行列への変換は行列積が中心(byte/flop値の削減)
 - 帯行列から三重対角行列への変換のコストは非常に小さい
 - 2段階の逆変換が必要
 - 逆変換の演算量は従来の2倍に増加
 - 帯から三重対角への変換の逆変換の高性能実装が困難
- ⇒ 求める固有ベクトル数が多い場合はボトルネックに

新しい1ステップ型(EigenExa)



- 帯行列の固有値と固有ベクトルを直接計算
 - 帯行列化を採用することで, byte/flop値を削減
 - 逆変換のコストの増加を回避
 - EigenExa の `eigen_sx` では五重対角行列を採用
- 帯行列向けの分割統治法の実装が課題
 - 三重対角行列経由の場合よりコスト増加
 - 理論的には研究されているが, 実装例は知らない

五重対角化 (eigen_sx)



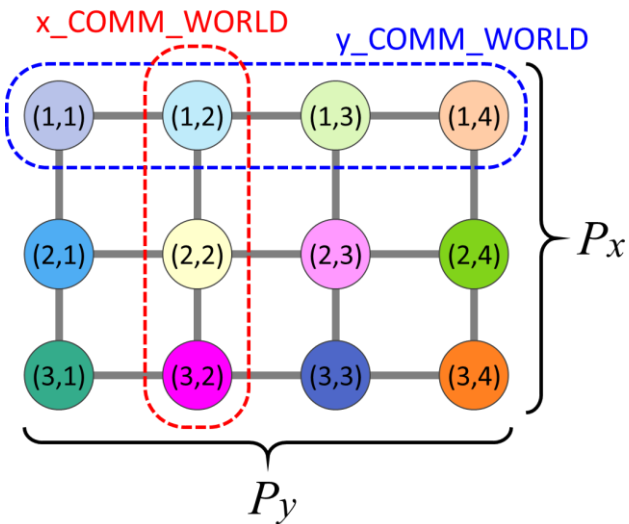
$$\tilde{H} = I - \begin{bmatrix} \text{col} & \text{row} \end{bmatrix} (= I - \tilde{u}\tilde{\beta}\tilde{u}^\top)$$

(ブロックハウスホルダー変換)

- 向上
-
- 各要素は
1回のみ利用
- 各要素を
2回利用可能
- ※一列単位で処理
 - ※二列単位で処理
 - ※1回の通信データ量が増加

EigenExaにおける並列実装の概要

EigenExa のプロセス形状



プロセス内のデータ構造

(1, 1)	(1, 2)	(1, 3)	(1, 4)
11 15 19 45 49 79	12 16 1a 46 4a 7a aa	13 17 1b 47 4b 77 7b ab	14 18 1c 44 48 4c 78 7c ac
(2, 1)	(2, 2)	(2, 3)	(2, 4)
25 29 55 59 89	22 26 2a 56 5a 8a	23 27 2b 57 5b 8b bb	24 28 2c 58 5c 88 8c bc
(3, 1)	(3, 2)	(3, 3)	(3, 4)
35 39 69 99	36 3a 66 6a 9a	33 37 3b 67 6b 9b	34 38 3c 68 6c 9c cc

行列の要素とプロセスの関係

11	12	13	14	15	16	17	18	19	1a	1b	1c
	22	23	24	25	26	27	28	29	2a	2b	2c
		33	34	35	36	37	38	39	3a	3b	3c
			44	45	46	47	48	49	4a	4b	4c
				55	56	57	58	59	5a	5b	5c
					66	67	68	69	6a	6b	6c
						77	78	79	7a	7b	7c
							88	89	8a	8b	8c
								99	9a	9b	9c
									aa	ab	ac
										bb	bc
											cc

- 指定されたプロセスを2次元プロセスグリッドに配置
(実行時に MPI コミュニケータを指定)
- 行列データは2次元サイクリック分割で指定したプロセスグリッドに分散
(各プロセスはローカル配列としてデータを格納)
- ScaLAPACK との同時利用ではデータ分散の変換が必要

性能評価 1

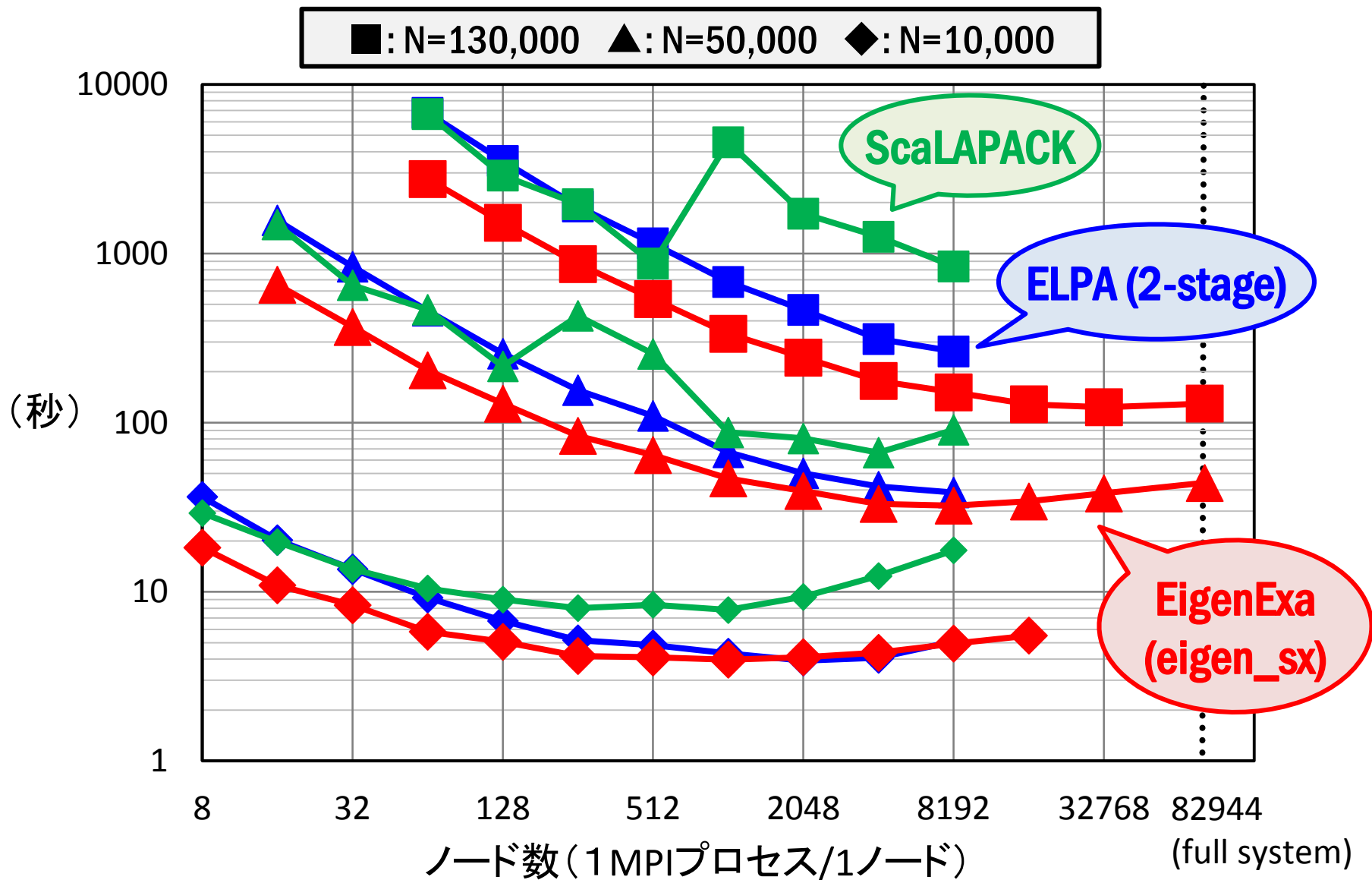
固有値ソルバの性能比較 on 京コンピュータ

- 評価対象ソルバ
 - EigenExa (五重対角行列経由ルーチン eigen_sx)
 - ELPA (※ 他のソルバと比べて京向けの最適化が不徹底)
 - ScaLAPACK
- 評価条件
 - 乱数行列の全固有対計算の実行時間を測定
 - MPI/OpenMPハイブリッド実行(ノードあたり1プロセス8スレッド)

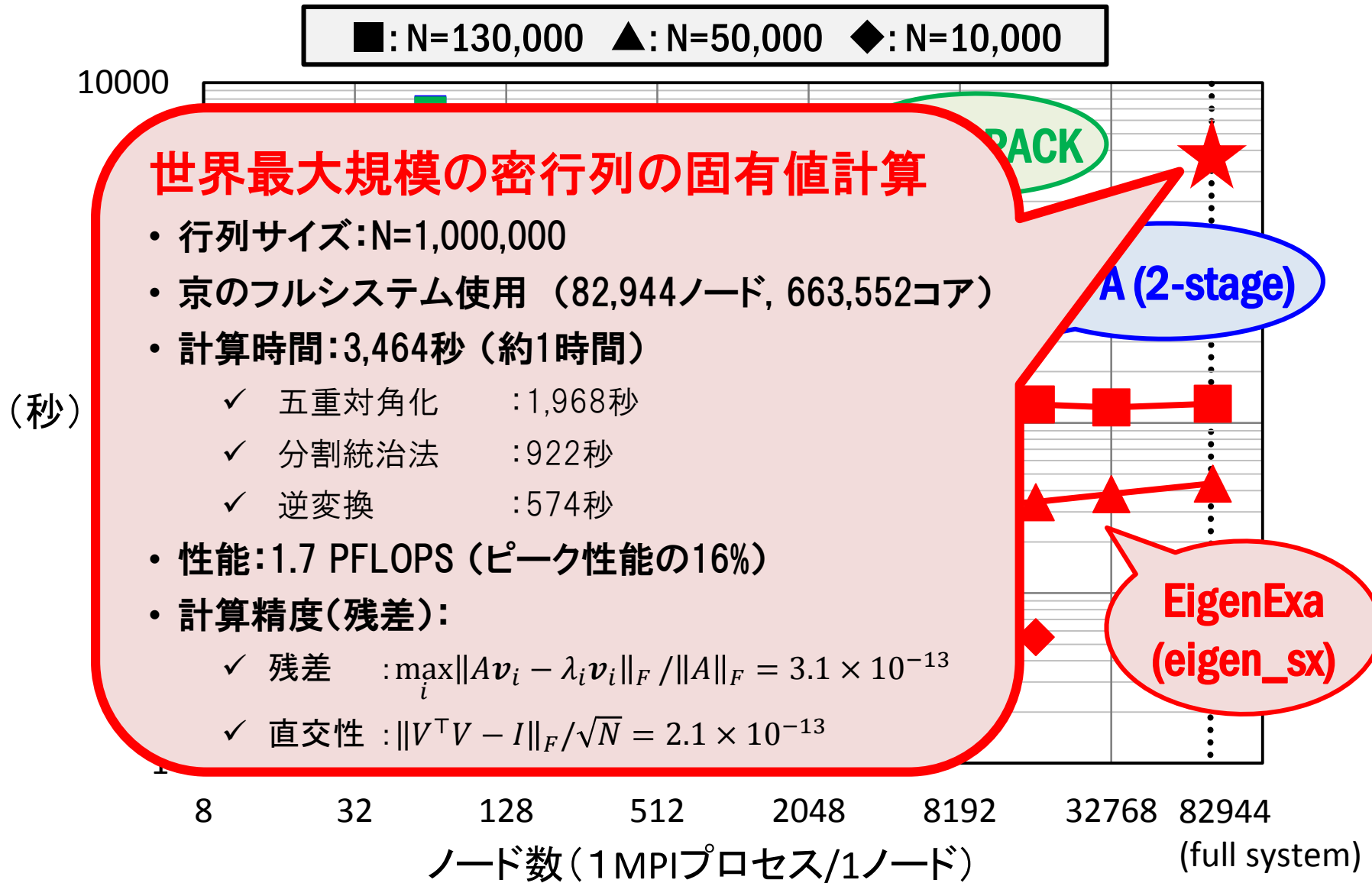
京コンピュータの仕様

- SPARC64 VIIIfx (8コア, 計 128 GFLOPS) × 1 / node
- 82944 ノード (計 10.62 PFLOPS)
- 6Dメッシュ/トーラスネットワーク(5 GB/s/link)

固有値計算全体の時間@京

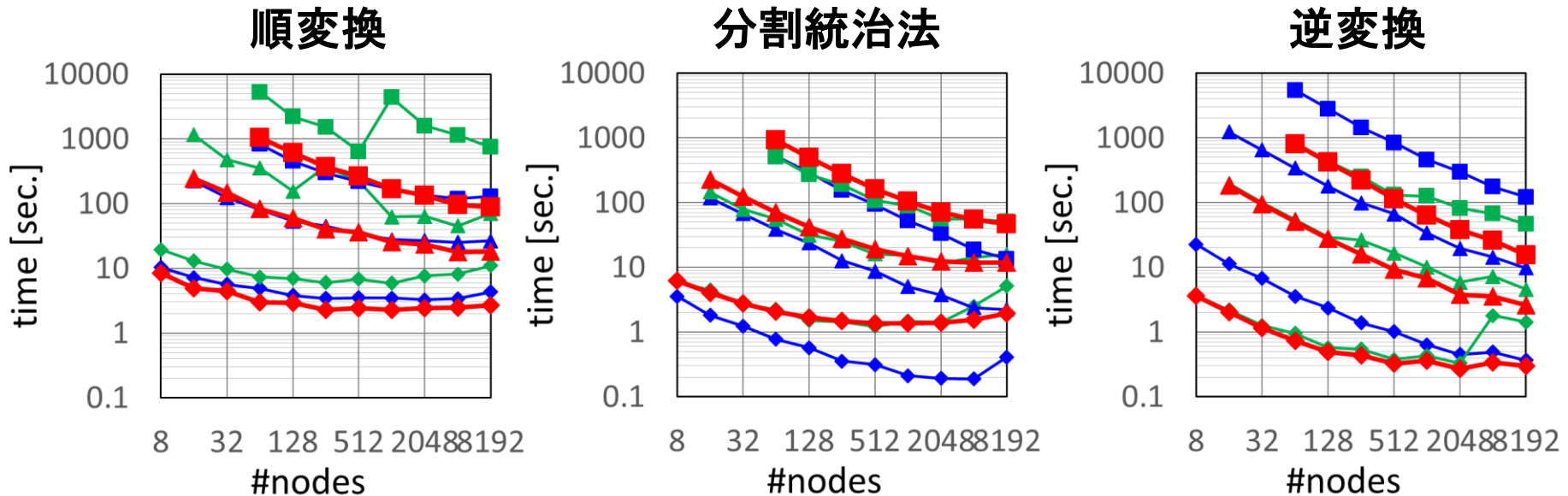


世界最大規模(100万次元)の計算



各ステップの計算時間@京

■ EigenExa
 ■ ELPA
 ■ ScaLAPACK



- 順変換および逆変換では高いスケーラビリティ
- 分割統治法が性能ボトルネック
 - ELPA や ScaLAPACK より多い計算量
 - 実装上の問題(ScaLAPACK を元にした実装)

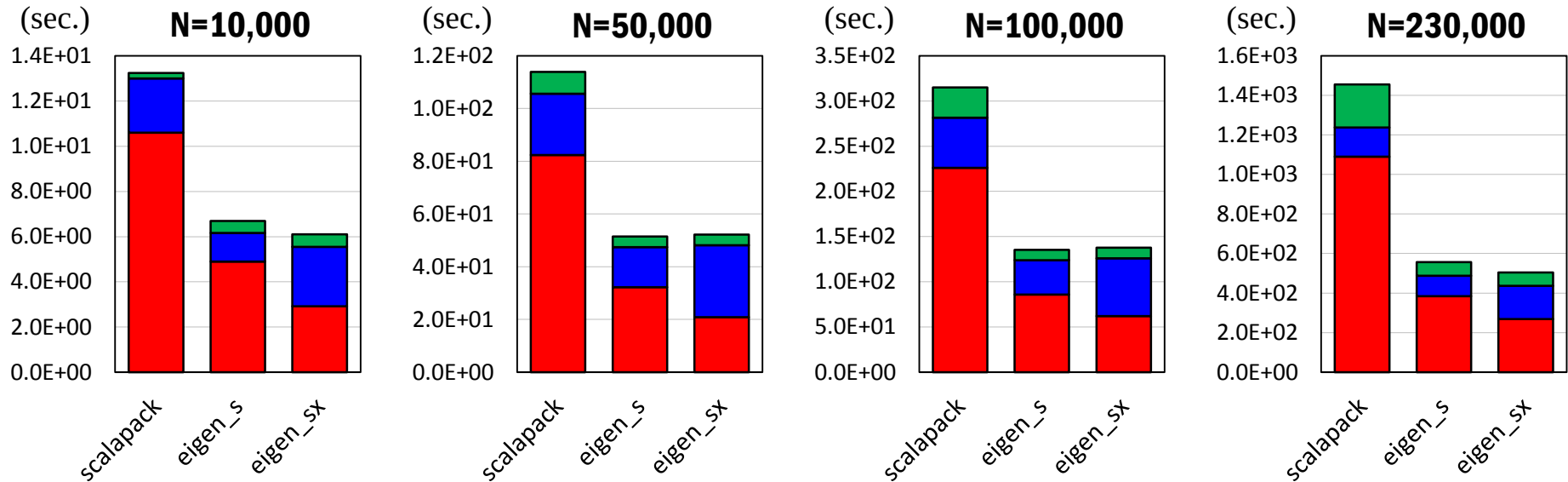
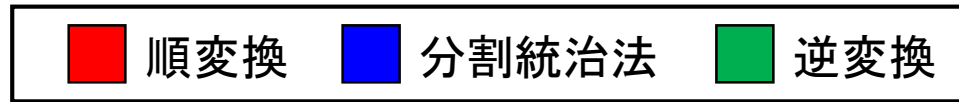
EigenExa の詳細な性能評価 @ 東大FX10

- 評価対象ルーチン
 - ScaLAPACK
 - EigenExa の eigen_s ルーチン(三重対角行列経由)
 - EigenExa の eigen_sx ルーチン(五重対角行列経由)
- 測定条件
 - 乱数行列の全固有対計算の実行時間を計算ステップごとに測定
 - MPI/OpenMPハイブリッド実行(ノードあたり1スレッド16スレッド)

東大 FX10 の性能

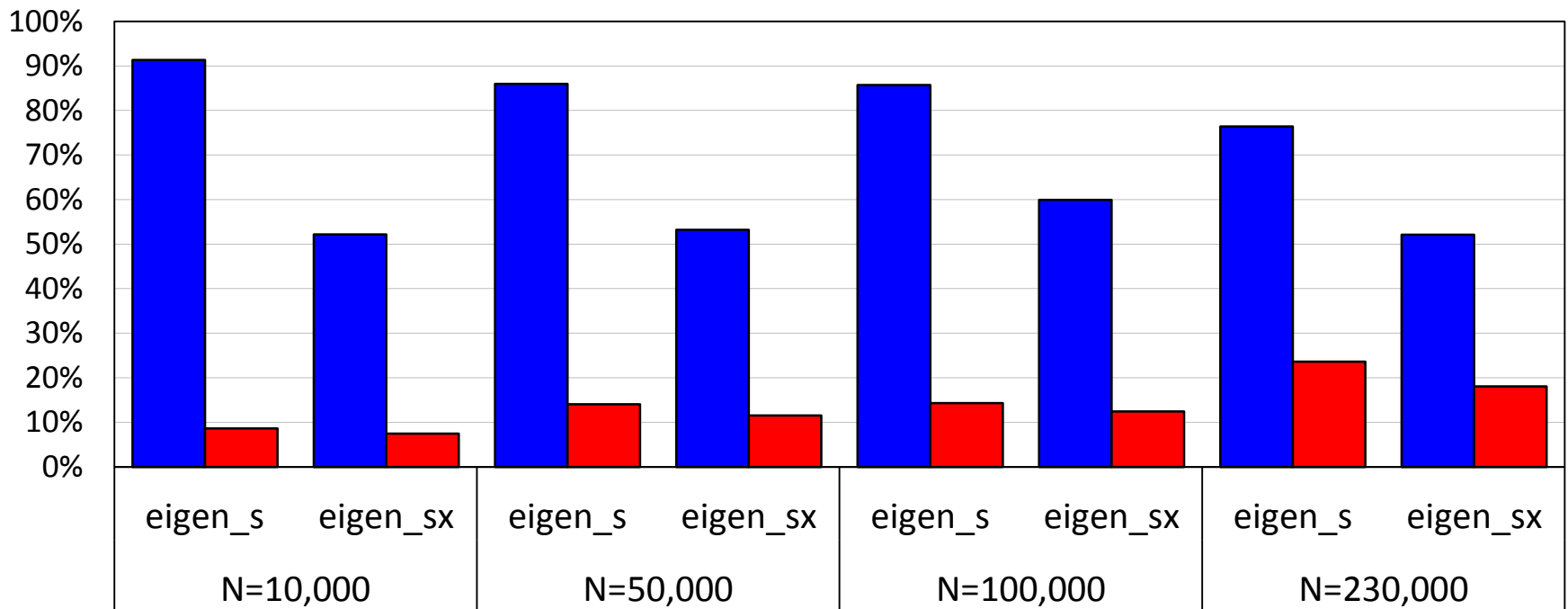
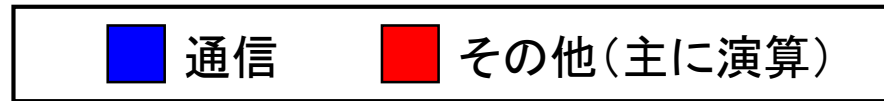
- SPARC64 IXfx (16コア, 計 236.5 GFLOPS) × 1 / node
- 4800 ノード (計 1.13 PFLOPS)
- 6Dメッシュ/トーラスネットワーク(5 GB/s/link, 双方向)

実行時間@東大FX10



- eigen_s, eigen_sx は ScaLAPACK に比べてはるかに高速
- eigen_sx は eigen_s 比で同程度か若干高速
- 順変換と分割統治法が主要な演算ステップ

三重(五重)対角化の内訳@東大FX10



- 通信時間が支配的.
- 五重対角アプローチ(eigen_sx)は計算の高速化よりも, 通信時間の削減の効果が大きい.

ここまでのまとめ

- 並列計算機向け固有値ソルバの現状を紹介
- 固有値ソルバ EigenExa の概要を紹介
- EigenExa は「京」で高い性能を実現
 - ✓ 13万次元なら2分, 100万次元なら1時間で行列の全固有対を1時間で計算可能
 - ✓ 現在は分割統治法が性能ボトルネック
- EigenExa の詳細な性能を評価 (FX10 4800ノード)
 - ✓ 順変換と分割統治法が主要な計算ステップ
 - ✓ 順変換は通信時間が支配的.
五重対角化アプローチは, 通信時間削減に効果.

ポストペタスケール環境への展望

item	FX10	FX100	性能向上率
ノードあたりピーク演算性能 (GFLOPS)	236.5	1000～	x 4.7
ノードあたりピークメモリバンド幅 (GB/sec)	85	240	x 2.8
ネットワークバンド幅 (GB/sec)	5.0	12.5	x 2.5
ネットワークレイテンシ (μsec)	0.92	0.71	x 1.3

References:

- T. Yoshida, “SPARC64 Xlfx: Fujitsu’s next generation processor for HPC”, HC26, 2014.
- N. Shida, et al., “MPI library and low-Level communication on the k computer”, Open MPI Publications, 2012.
- Y. Ajima, et al., “The tofu interconnect 2”, Hot Interconnect 22, 2014.

◆ eigen_sx の展望

- メモリバンド幅，ネットワークレイテンシの性能向上率が相対的に低い
ため，順変換における優位性は拡大
- ピーク演算性能の向上により，分割統治法の演算量の多さに由来する
eigen_sx の不利は小さくなる

**ポストペタスケール環境において
eigen_sx は高い性能ポテンシャルをもつ**

ベクトル計算機への展開に対する展望

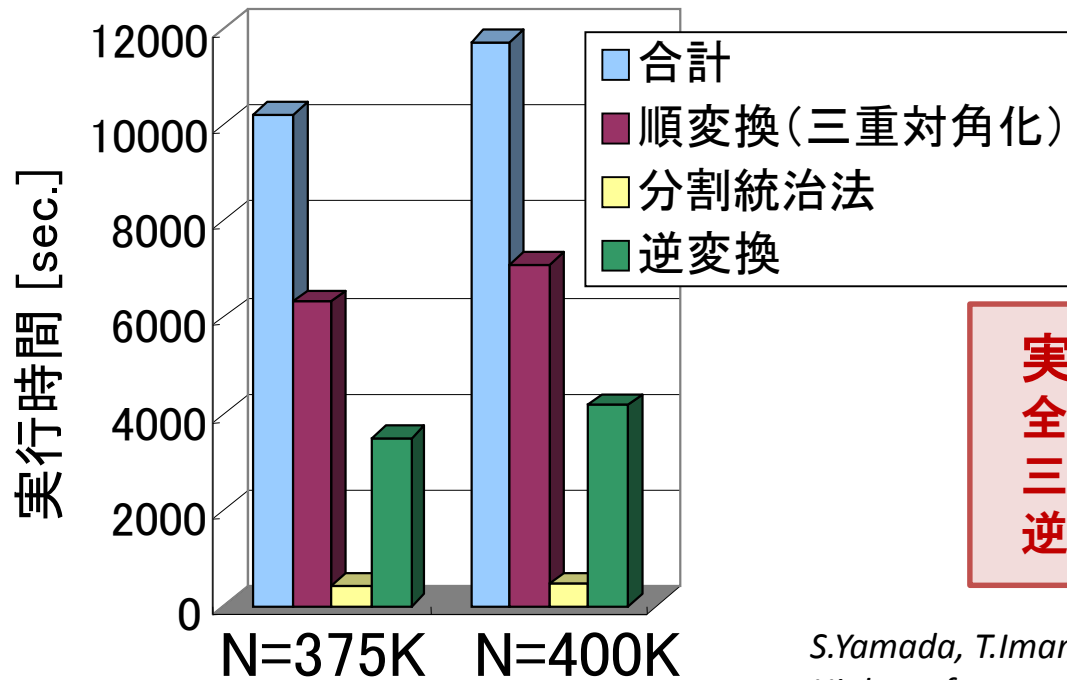
地球シミュレータ(初代)での性能

地球シミュレータ(初代)の仕様

	地球シミュレータ
Processor Element (PE) あたり演算性能	8 GFLOPS/PE
ノードあたり PE 数	8 PEs/node
メモリ帯域	256 GB/s
ネットワークバンド幅	12.3 GB/s
ノード数	640
総演算性能	40 TFLOPS

- 特徴: 相対的に高いメモリバンド幅, ネットワーク性能
- 性能評価(大規模問題)
 - 三重対角化に基づく固有値ソルバ (eigen_s 相当)
 - テスト行列はフランク行列
(分割統治法の実行時間は乱数行列の場合より短くなる)

地球シミュレータでの性能評価結果



使用 PE 数: 4992 (= 624 × 8)
使用メモリ: 3.8 TB

実行性能

全体: 18.9 TFLOPS (ピーク比 47%)

三重対角化: 12.0 TFLOPS (29 %)

逆変換: 30.7 TFLOPS (75 %)

(当時)世界最大規模

S.Yamada, T.Imamura, T.Kano and M.Machida,
High-performance Computing for Exact Numerical Approach
to Quantum Many-body Problems on the Earth Simulator, SC2006.

近年のスカラ型並列計算機と比べて、分割統治法の実行時間が短く、逆変換の実行時間の占める割合が大きい。

- フランク行列の性質(固有値の近接)と、三重対角行列経由であるために、分割統治法の演算量が非常に少ない。
- メモリバンド幅、ネットワーク性能が高く、順変換が実行性能が高い。

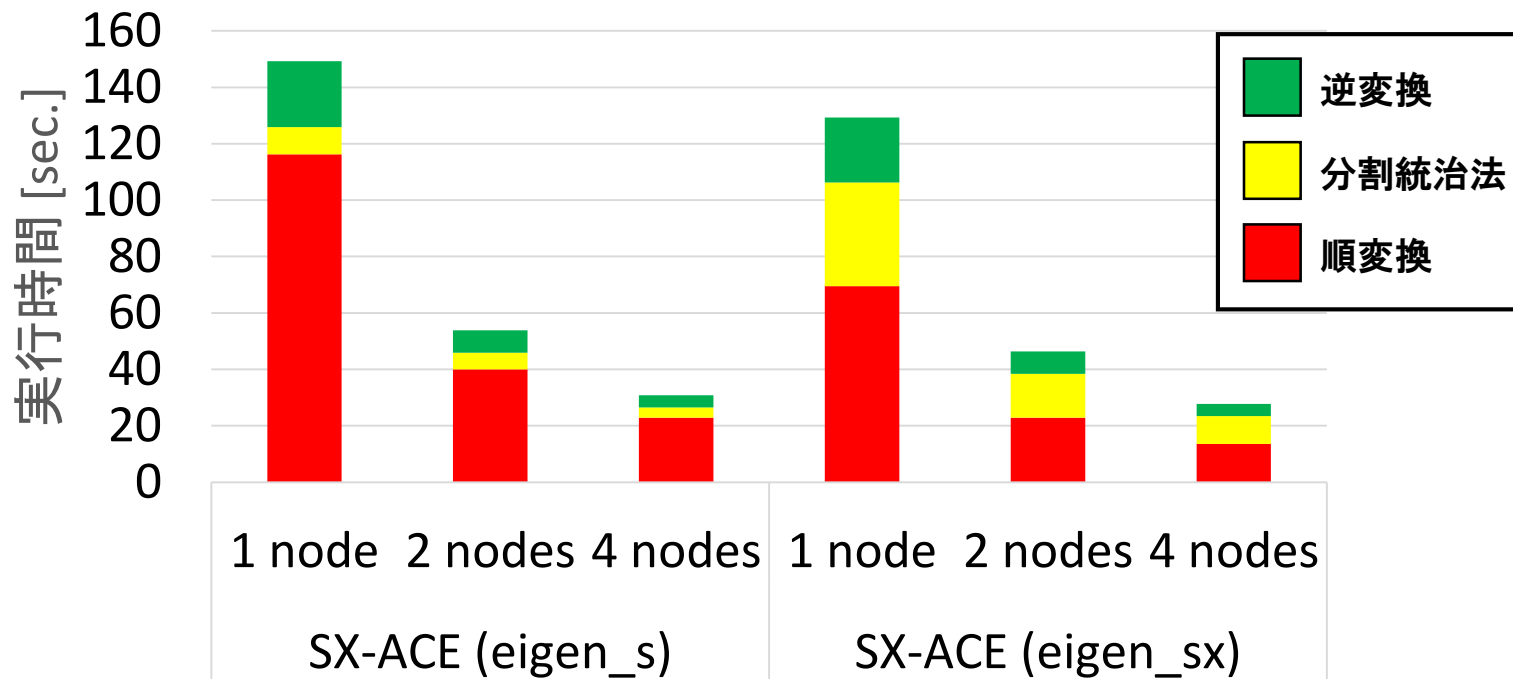
並列計算機の仕様

	SX-6 / ES1	SX-ACE	京
PE あたり演算性能	8 GFLOPS/PE	64 GFLOPS/PE	16 GFLOPS/core
PE数	8 PEs/node	4 PEs/node	8 cores/node
メモリ帯域	256 GB/s	256 GB/s	64 GB/s
B/F	4	1	0.5
ネットワークバンド幅	12.3 GB/s	4 GB/s	5 GB/s

SX-6/地球シミュレータからの変化

- ノードあたり演算性能(FLOPS)は4倍高速化
 - 演算性能に対するメモリバンド, ネットワークの相対的な性能は低下
- ⇒ 従来機と比べて「京」などに近い性質をもつと予想される

SX-ACE 上での性能(N = 12,000)



- SX-ACE の性能は ES1 よりも「京」や FX10 に類似
 - ✓ 順変換・分割統治法が実行時間の中心, 五重対角化アプローチの有効性
 - ⇒ 「京」向け性能向上手段が同様に有効である可能性
- ノード内の性能最適化は不十分(順 17%, 分 33%, SX-ACE 4ノード)
 - ⇒ ベクトル計算向け最適化は別途必要か

固有値計算からのベクトル計算展望

- ベクトル計算機向けアルゴリズム
 - 長いベクトル長
 - ベクトル化可能な同一操作
- 既存アルゴリズム(マルチコア向け)
 - キャッシュやレジスタブロックするための短ベクトル
 - 多段ループ構造
 - スレッドはそれぞればらばらの計算(分割統治法など)
- GPU(特にCUDA)
 - Warp単位のSIMD操作は比較的短いベクトル機と考えてよい
 - GPU 向けのアルゴリズムとの融合が一つの解か？
 - 複雑なメモリ階層は ADB 導入との類似点も？

おわりに

- 並列計算機向け固有値ソルバの現状を紹介
- 固有値ソルバ EigenExa の概要を紹介
- EigenExa は「京」で高い性能を実現
 - ✓ 13万次元なら2分, 100万次元なら1時間で全固有対
- EigenExa の SX-ACE 上での性能は「京」上での性能に類似
 - ✓ 「京」向け性能向上手段の五重対角アプローチは有効か
 - ✓ ベクトル計算機向けの最適化も別途必要

EigenExa 公開中

<http://www.aics.riken.jp/labs/lpnctr/EigenExa.html>