CMSI計算科学技術 特論B

第4回
アプリケーションの性能最適化の実例1

2014年5月8日

独立行政法人理化学研究所
計算科学研究機構 運用技術部門
ソフトウェア技術チーム チームヘッド

南 一生
minami_kaz@riken.jp

京
K computer

講義の概要

スーパーコンピュータとアプリケーションの性能
アプリケーションの性能最適化１（高並列性能最適化）
- アプリケーションの性能最適化２（高並列性能最適化）
アプリケーションの性能最適化の実例１
- アプリケーションの性能最適化の実例２

内容

理研で進めた性能最適化
RSDFTの性能最適化
PHASEの性能最適化

- 本資料は，理化学研究所AICS運用技術部門ソフトウェア技術チーム，長谷川幸弘氏，黒田明義氏の発表データを使用して作成しています．

理研で進めた性能最適化

理研で進めた性能最適化

6本のターゲットアプリ

プログラム名	分野	アプリケーション概要	期待される成果	手法
NICAM	地球科学	全球雲解像大気大循環モデル	大気大循環のエンジンとなる熱帯積雲対流活動を精緻に表現することでシミュレーションを飛躍的に進化させ，現時点では再現が難しい大気現象の解明が可能となる．(開発　東京大学,JAMSTEC,RIKEN AICS)	FDM (大気)
Seism3D	地球科学	地震波伝播・強震動シミュレーション	既存の計算機では不可能な短い周期の地震波動の解析・予測が可能となり，木造建築およびコンクリート構造物の耐震評価などに応用できる．(開発　東京大学地震研究所)	FDM (波動)
PHASE	ナノ	平面波展開第一原理電子状態解析	第一原理計算により，ポスト35nm世代ナノデバイス，非シリコン系デバイスの探索を行う．(開発　物質・材料研究機構)	平面波 DFT
FrontFlow/Blue	工学	Large Eddy Simulation (LES)に基づく非定常流体解析	LES解析により，エンジニアリング上重要な乱流境界層の挙動予測を含めた高精度な流れの予測が実現できる．(開発　東京大学生産技術研究所)	FEM (流体)
RSDFT	ナノ	実空間第一原理電子状態解析	大規模第一原理計算により，10nm以下の基本ナノ素子(量子細線，分子，電極，ゲート，基盤など)の特性解析およびデバイス開発を行う．(開発　東京大学)	実空間 DFT
LatticeQCD	物理	格子QCDシミュレーションによる素粒子・原子核研究	モンテカルロ法およびCG法により，物質と宇宙の起源を解明する．(開発　筑波大)	QCD

理研で進めた性能最適化

コラボレーション

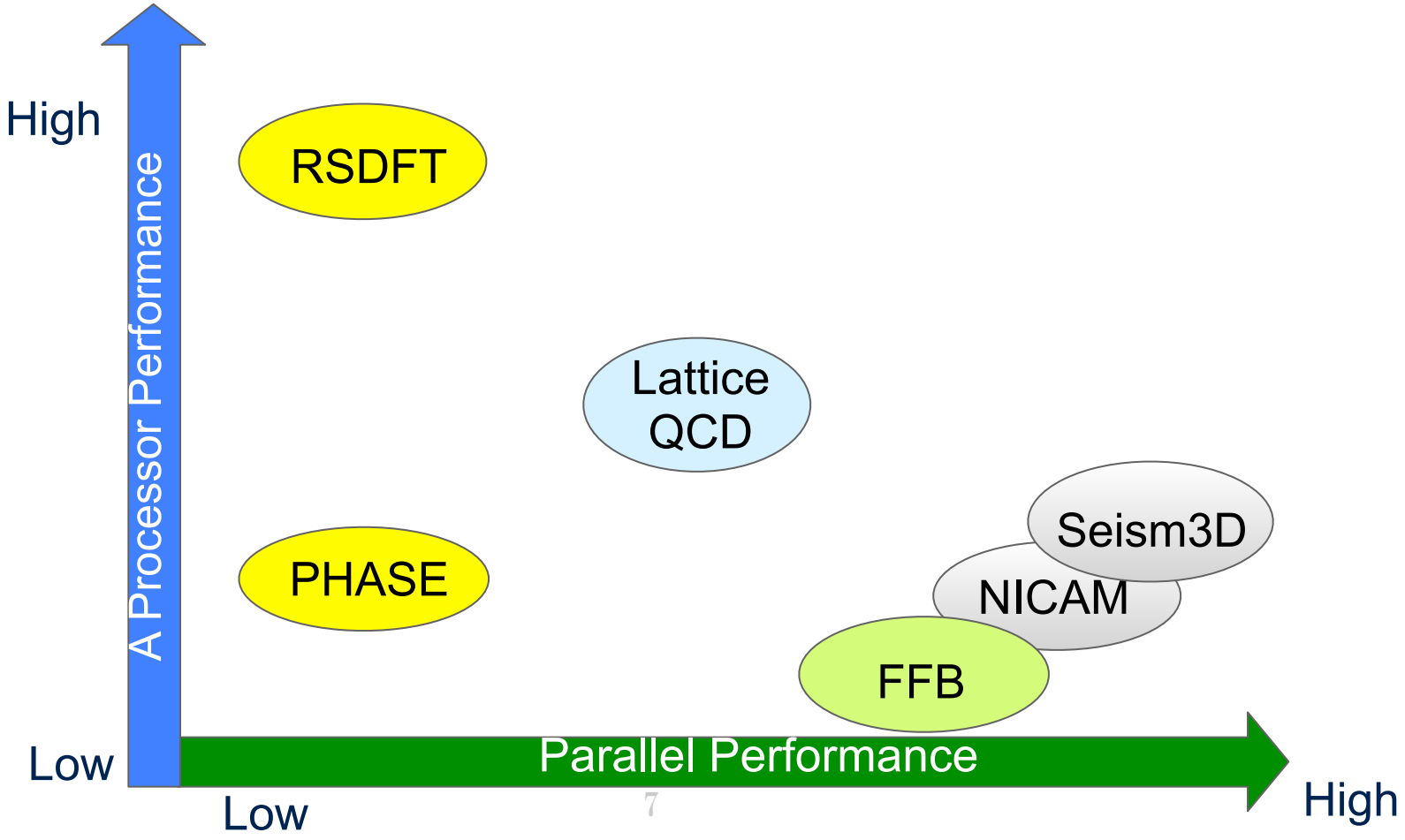
計算科学
（コード開発者）

計算機科学
（理研）

東京大学,JAMSTEC
東京大学地震研究所
物質・材料研究機構
東京大学生産技術研究所
筑波大　RIKEN AICS

理研で進めた性能最適化

6本のターゲットアプリの計算機科学的な位置づけ

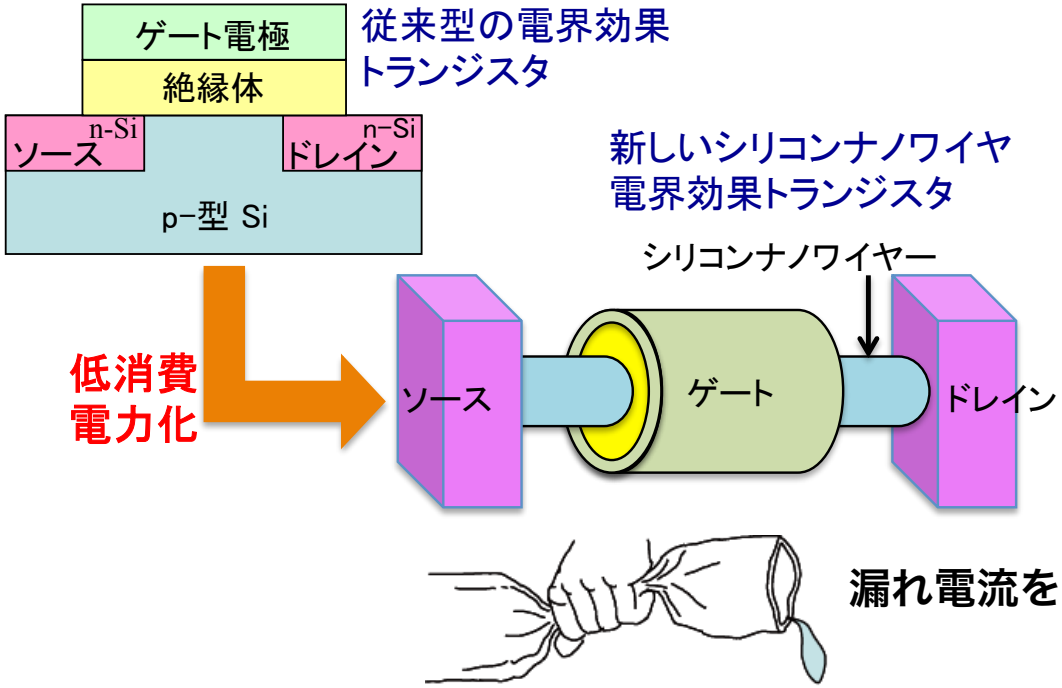


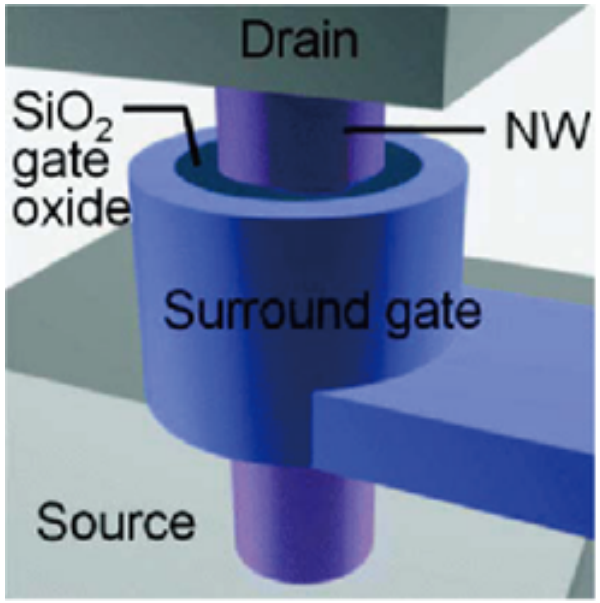
RSDFTの性能最適化

RSDFTとは

- ナノスケールでの量子論的諸現象を第一原理に立脚して解明し新機能を有するナノ物質・構造を予測
- 例えば・・・

漏れ電流が問題





漏れ電流を押さえる

RSDFTの原理

Kohn-Sham方程式

電子密度 $n(\mathbf{r}) = \sum_i |\varphi_i(r)|^2$

$$\left[-\frac{1}{2}\nabla^2 + v_{\text{nucl}}(\mathbf{r}) + \int \frac{n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} d\mathbf{r}' + \frac{\delta E_{\text{XC}}[n]}{\delta n(\mathbf{r})} \right] \varphi_i(\mathbf{r}) = \varepsilon_i \varphi_i(\mathbf{r})$$

波動関数

ハミルトニアン

φ_i: 電子軌道（=波動関数）

i：電子準位（=エネルギーバンド）

r：空間離散点（=空間格子）

$$H\varphi_i(r) = \varepsilon_i \varphi_i(r)$$

固有値方程式

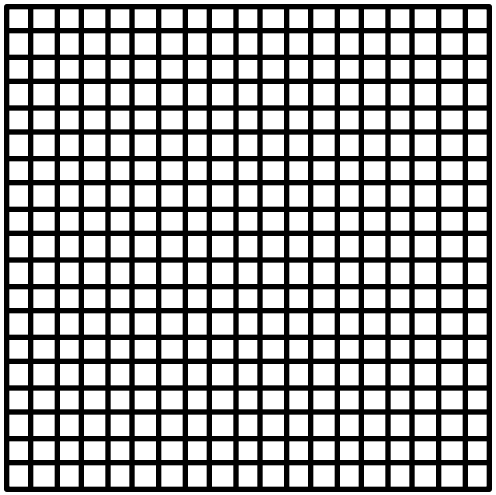
RSDFTの原理

実空間法

$$H\varphi_i(r) = \varepsilon_i\varphi_i(r)$$ 固有値方程式

Kohn-Sham方程式を3次元格子上に
離散化し差分方程式として解く

ML2



ユニットセル（実際は3次元）

各次元方向をML1,ML2,ML3等分して格子を生成

ML(=ML1×ML2×ML3）次元のエルミート行列の固有値問題

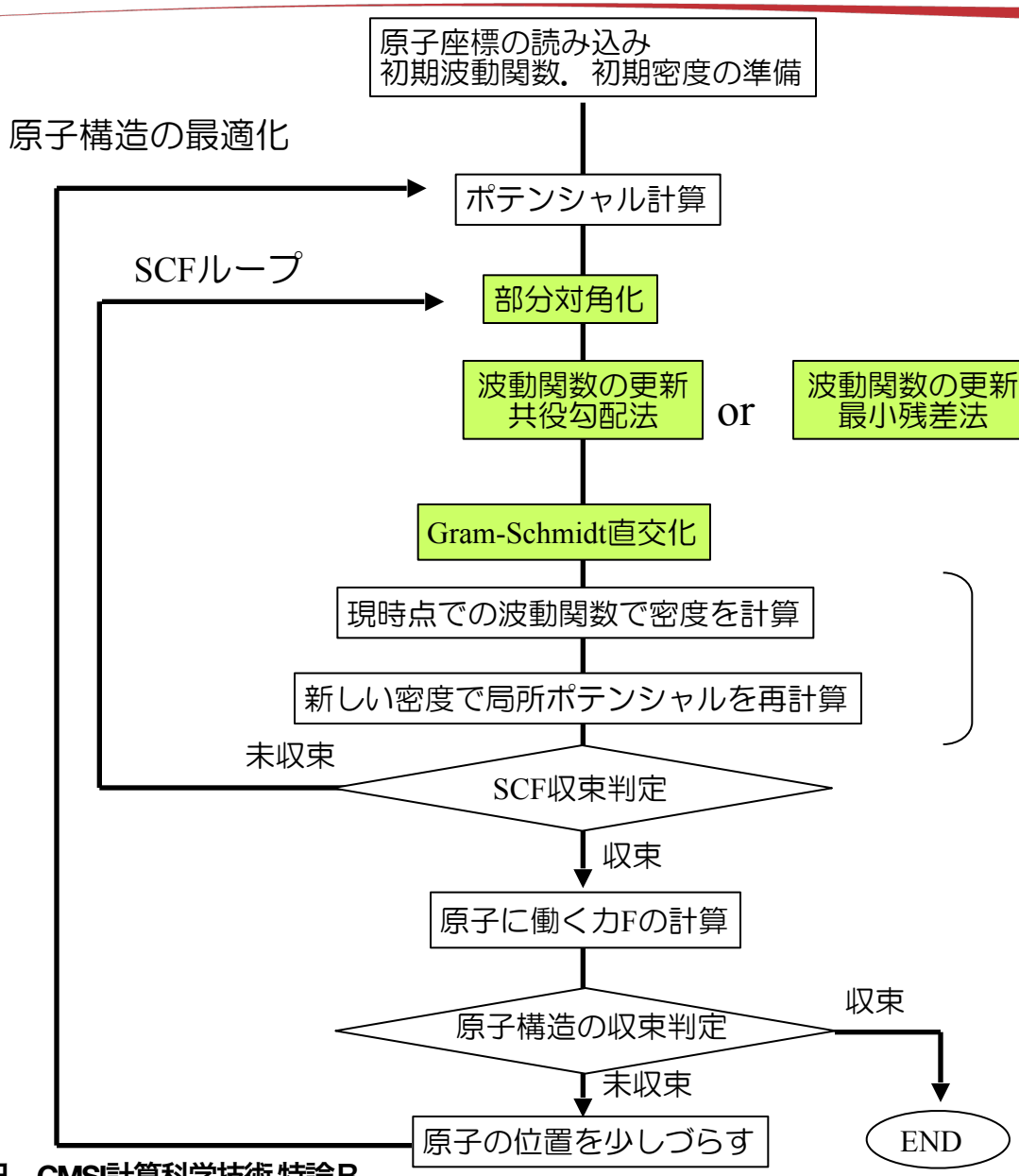
RSDFTの計算フロー

Self-Consistent Field procedure

1. (CG) 共役勾配法
2. (GS) Gram-Schmidt規格直交化
3. 密度とポテンシャルの更新
4. (SD) 部分対角化

SCF計算

RSDFTの計算フロー

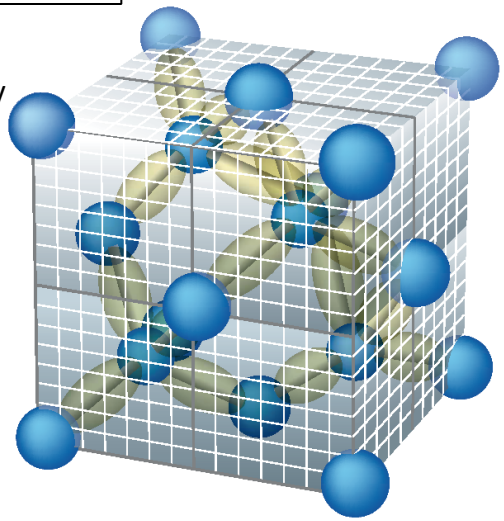


RSDFTの並列化

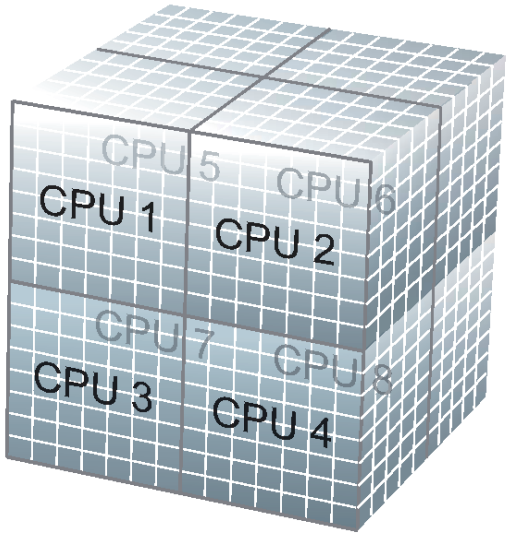
J.-I. Iwata *et al*., J. Comp. Phys. (2010)

Real space

Blue : Si atom
Yellow: electron density



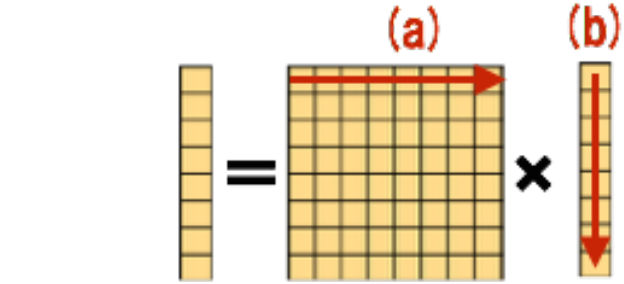
CPU space



RSDFTのCPU単体性能の向上

スレッド並列化
キャッシュの有効利用

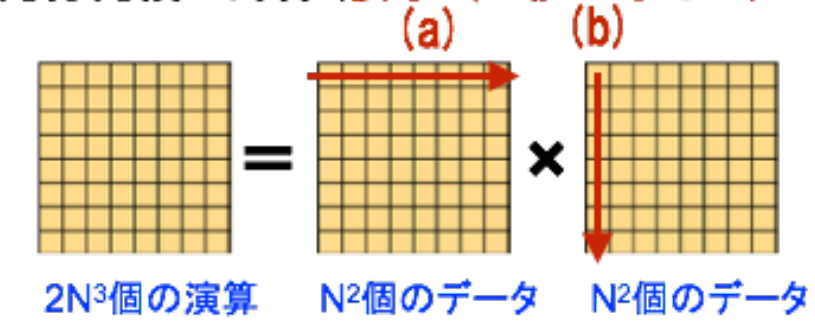
行列ベクトル積の計算（要求B/F値が大きい）



B/F値
=移動量(Byte)/演算量(Flop)
=$(N^2+N)/2N^2$
≒1/2

原理的には1/Nより大きな値

行列行列積の計算（要求B/F値が小さい）



B/F値
=移動量(Byte)/演算量(Flop)
=$2N^2/2N^3$
=1/N

原理的にはNが大きい程小さな値

RSDFTのCPU単体性能の向上

RSDFT

- 実空間差分法
- 空間並列

計算コアの最適化

- 行列積化

ターゲット計算機：PACS-CS, T2K-Tsukuba

スレッド並列の実装

ターゲット計算機：PACS-CS, T2K-Tsukuba

RSDFTのCPU単体性能の向上

GramSchmidt直交化の行列積化

$$\psi_1' = \psi_1$$

$$\psi_2' = \psi_2 - \langle\psi_1'|\psi_2\rangle|\psi_1'\rangle$$

$$\psi_3' = \psi_3 - \langle\psi_1'|\psi_3\rangle|\psi_1'\rangle - \langle\psi_2'|\psi_3\rangle|\psi_2'\rangle$$

$$\psi_4' = \psi_4 - \langle\psi_1'|\psi_4\rangle|\psi_1'\rangle - \langle\psi_2'|\psi_4\rangle|\psi_2'\rangle - \langle\psi_3'|\psi_4\rangle|\psi_3'\rangle$$

$$\psi_5' = \psi_5 - \langle\psi_1'|\psi_5\rangle|\psi_1'\rangle - \langle\psi_2'|\psi_5\rangle|\psi_2'\rangle - \langle\psi_3'|\psi_5\rangle|\psi_3'\rangle - \langle\psi_4'|\psi_5\rangle|\psi_4'\rangle$$

$$\psi_6' = \psi_6 - \langle\psi_1'|\psi_6\rangle|\psi_1'\rangle - \langle\psi_2'|\psi_6\rangle|\psi_2'\rangle - \langle\psi_3'|\psi_6\rangle|\psi_3'\rangle - \langle\psi_4'|\psi_6\rangle|\psi_4'\rangle - \langle\psi_5'|\psi_6\rangle|\psi_5'\rangle$$

$$\psi_7' = \psi_7 - \langle\psi_1'|\psi_7\rangle|\psi_1'\rangle - \langle\psi_2'|\psi_7\rangle|\psi_2'\rangle - \langle\psi_3'|\psi_7\rangle|\psi_3'\rangle - \langle\psi_4'|\psi_7\rangle|\psi_4'\rangle - \langle\psi_5'|\psi_7\rangle|\psi_5'\rangle - \langle\psi_6'|\psi_7\rangle|\psi_6'\rangle$$

$$\psi_8' = \psi_8 - \langle\psi_1'|\psi_8\rangle|\psi_1'\rangle - \langle\psi_2'|\psi_8\rangle|\psi_2'\rangle - \langle\psi_3'|\psi_8\rangle|\psi_3'\rangle - \langle\psi_4'|\psi_8\rangle|\psi_4'\rangle - \langle\psi_5'|\psi_8\rangle|\psi_5'\rangle - \langle\psi_6'|\psi_8\rangle|\psi_6'\rangle - \langle\psi_7'|\psi_8\rangle|\psi_7'\rangle$$

$$\psi_9' = \psi_9 - \langle\psi_1'|\psi_9\rangle|\psi_1'\rangle - \langle\psi_2'|\psi_9\rangle|\psi_2'\rangle - \langle\psi_3'|\psi_9\rangle|\psi_3'\rangle - \langle\psi_4'|\psi_9\rangle|\psi_4'\rangle - \langle\psi_5'|\psi_9\rangle|\psi_5'\rangle - \langle\psi_6'|\psi_9\rangle|\psi_6'\rangle - \langle\psi_7'|\psi_9\rangle|\psi_7'\rangle - \langle\psi_8'|\psi_9\rangle|\psi_8'\rangle$$

$$\vdots$$

オリジナルは行列ベクトル積

RSDFTのCPU単体性能の向上

GramSchmidt直交化の行列積化

ベクトル積を行列積に変換

再帰分割法

三角部(DGEMV)

四角部(DGEMM)

$$\varphi_1 = \psi_1$$

$$\varphi_2 = \psi_2 - \varphi_1\langle\varphi_1{}^*|\psi_2\rangle$$

$$\varphi_3 = \psi_3 - \varphi_1\langle\varphi_1{}^*|\psi_3\rangle - \varphi_2\langle\varphi_2{}^*|\psi_3\rangle$$

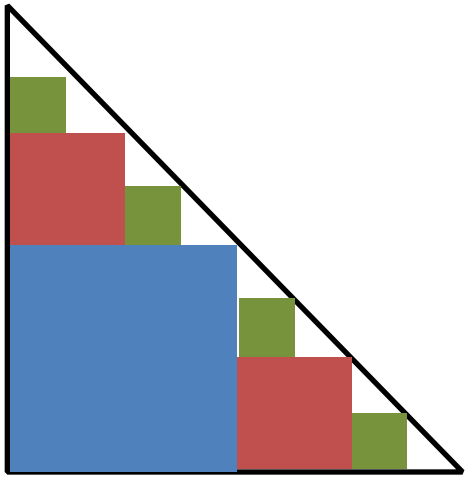
$$\varphi_4 = \psi_4 - \varphi_1\langle\varphi_1{}^*|\psi_4\rangle - \varphi_2\langle\varphi_2{}^*|\psi_4\rangle - \varphi_3\langle\varphi_3{}^*|\psi_4\rangle$$

$$\varphi_5 = \psi_5 - \varphi_1\langle\varphi_1{}^*|\psi_5\rangle - \varphi_2\langle\varphi_2{}^*|\psi_5\rangle - \varphi_3\langle\varphi_3{}^*|\psi_5\rangle - \varphi_4\langle\varphi_4{}^*|\psi_5\rangle$$

$$\varphi_6 = \psi_6 - \varphi_1\langle\varphi_1{}^*|\psi_6\rangle - \varphi_2\langle\varphi_2{}^*|\psi_6\rangle - \varphi_3\langle\varphi_3{}^*|\psi_6\rangle - \varphi_4\langle\varphi_4{}^*|\psi_6\rangle - \varphi_5\langle\varphi_5{}^*|\psi_6\rangle$$

$$\varphi_7 = \psi_7 - \varphi_1\langle\varphi_1{}^*|\psi_7\rangle - \varphi_2\langle\varphi_2{}^*|\psi_7\rangle - \varphi_3\langle\varphi_3{}^*|\psi_7\rangle - \varphi_4\langle\varphi_4{}^*|\psi_7\rangle - \varphi_5\langle\varphi_5{}^*|\psi_7\rangle - \varphi_6\langle\varphi_6{}^*|\psi_7\rangle$$

$$\varphi_8 = \psi_8 - \varphi_1\langle\varphi_1{}^*|\psi_8\rangle - \varphi_2\langle\varphi_2{}^*|\psi_8\rangle - \varphi_3\langle\varphi_3{}^*|\psi_8\rangle - \varphi_4\langle\varphi_4{}^*|\psi_8\rangle - \varphi_5\langle\varphi_5{}^*|\psi_8\rangle - \varphi_6\langle\varphi_6{}^*|\psi_8\rangle - \varphi_7\langle\varphi_7{}^*|\psi_8\rangle$$



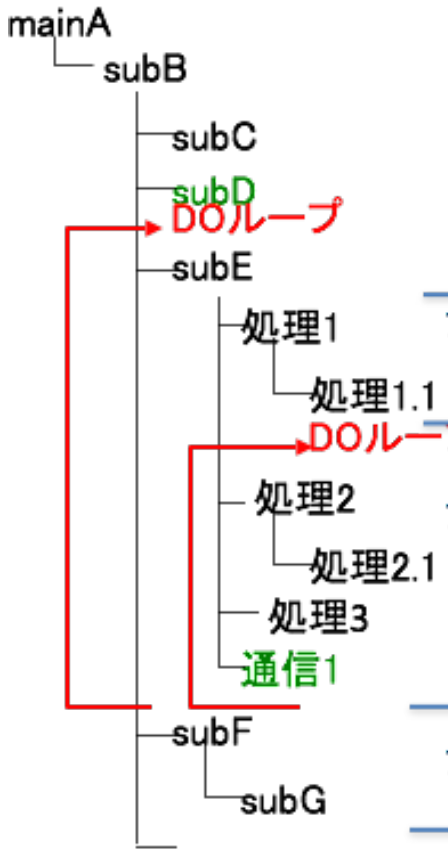
- 依存関係のある三角部とない四角部にブロック化して計算
- 再帰的にブロック化することで四角部を多く確保

※ＳＤも同様に行列積化が可能

RSDFTの並列特性分析

1. 並列特性分析
(処理構造分析・ブロック特性分析)

(1)コードの構造を分析し物理に沿った処理ブロック(計算/通信)に分割
(2)コードの実行時間の実測
(3)プログラムソースコードの調査
(4)処理ブロックの物理的処理内容を把握
(5)計算ブロック毎の計算特性把握(非並列/完全並列/部分並列、Nに比例/N**2に比例等)
(6)通信ブロック毎の通信特性把握 (グローバル通信・隣接通信、隣接面に比例&隣接通信/体積に比例、等)



	実行時間・スケーラビリティ	物理的処理内容	演算・通信特性	演算・通信見積り	カーネル
ブロック1 (計算)			部分並列	Nに比例	
ブロック2 (計算)			完全並列	N**3に比例	○
(通信)			隣接通信	隣接面に比例	○
ブロック3					

2. カーネル評価

(1)計算・通信ブロックについて物理的処理内容・コーディングの評価を行い同種の計算・通信ブロックを評価し異なる種類の計算・通信ブロックをカーネルの候補として洗い出す
(2)並列特性分析の結果から得た問題規模に対する依存性の情報を元にターゲット問題実行時に、また高並列実行時にカーネルとなる計算・通信カーネルを洗い出す

RSDFTの並列特性分析（処理・演算量）

ML:格子数，MB:バンド数

ルーチン	処理内容		演算量	高並列化性能	単体性能
DTCG	ML×ML対称行列の固有値，固有ベクトルを共役勾配法で固有値の小さいものから順にMB本求める.	レイリー商 $\frac{\langle \psi_m \lvert H_{KS} \rvert \psi_n \rangle}{\langle \psi_n \lvert \psi_n \rangle}$ → minimize	O(ML×ML)→O(N2) **O(N2)**		
GramSchmidt	規格直交化	$H_{m,n} = \langle \psi_m \lvert H_{KS} \rvert \psi_n \rangle$	O(MLxMB2)→O(N3) **O(N3)**		
DIAG	ML次元の部分空間に限ってハミルトニアンの対角化をする.				
	行列要素生成（MatE）	$\psi'_n = \psi_n - \sum_{m=1}^{n-1} \psi_m \langle \psi_m \lvert \psi_n \rangle$	O(MLxMB2)→O(N3) **O(N3)**		
	固有値求解（pdsyevd）	$\left(H_{N\times N} \right) \left(\vec{c}_n \right) = \varepsilon \left(\vec{c}_n \right)$	O(MB3)→O(N3) **O(N3)**		
	回転（RotV）	$\psi'_n(r) = \sum_{m=1}^{N} c_{n,m} \psi_m(r)$	O(MLxMB2)→O(N3) **O(N3)**		

RSDFTの並列特性分析（コスト）

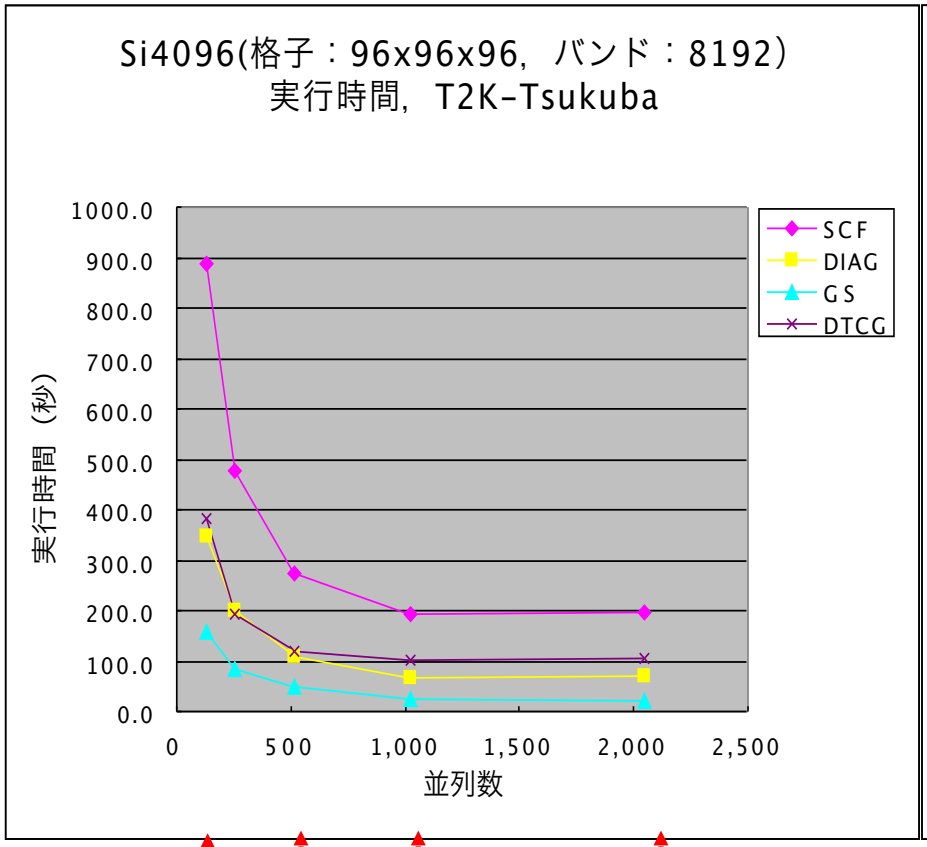
計算機　　：RICC
8,000原子：格子数120x120x120，バンド数16,000
並列数　　：8x8x8（空間方向のみ）
SCFループ1回実行の実測データからSCFループ100回として実行時間を推定

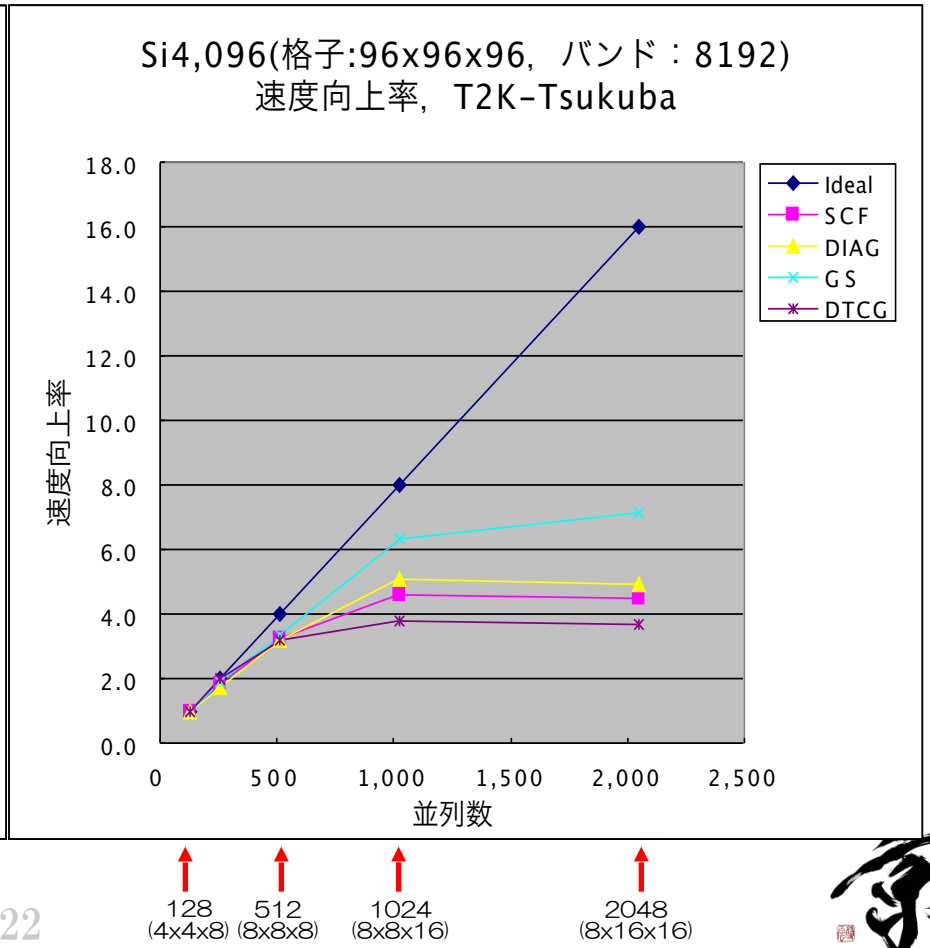
処理内容	コスト	演算	プロセス間通信
初期化 パラメータの読込み 全プロセスへの転送	0.4%		Bcast，Isend/Irecv
SCF部	99.6%	$O(N^3)$	
DO SCFループ（100回と仮定）			
DIAG	30.5%	$O(N^3)$ DGEMM中心	行列生成部: Reduce, Isend/Irecv（HPSI） 固有値ソルバー部：PDSYEVD内(Bcast) ローテーション:部分Bcast，部分Reduce
DTCG	27.4%	$O(N^2)$ 演算＜ロード	スカラー値のallreduce中心 Isend/Irecv(ノンローカル項/HPSI) Isend/Irecv(境界データ交換/BCSET)
GramSchmidt	38.6%	$O(N^3)$ DGEMM中心	Allreduce （内積，規格化変数）
Mixing, 途中結果の出力	3.1%		途中結果出力は毎SCFではないのでコストはもっと少
ENDDO			

演算時間の90%以上をBLASルーチン(DGEMM,DGEMV）が占めている

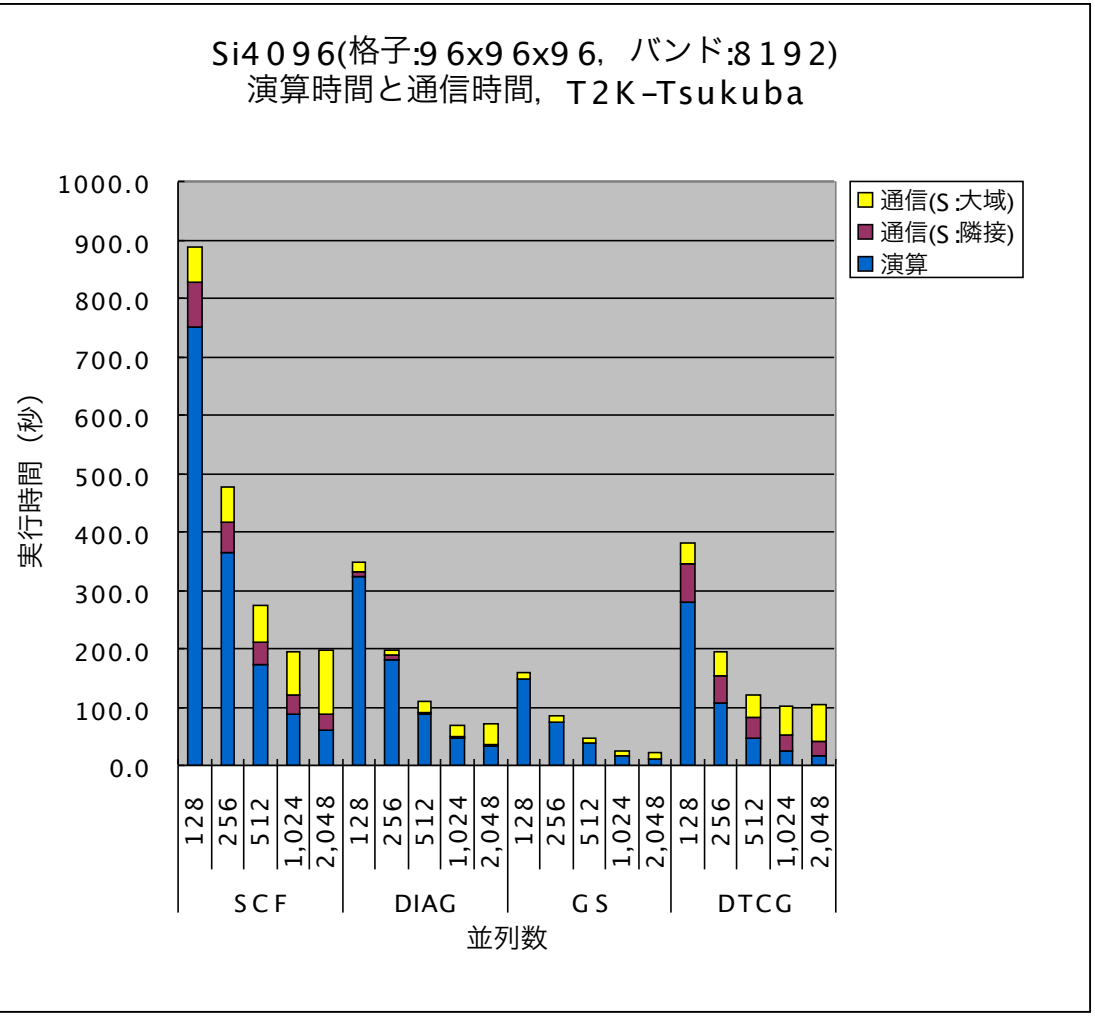
RSDFTの並列特性分析（ブロック毎のスケーラビリティ）

計算機：T2K-Tsukuba
コンパイラ&ライブラリ：PGI + mvapich2-medium
S方向の分割数:128, 256, 512, 1024, 2048





RSDFTの並列特性分析（ブロック毎のスケーラビリティ）



※PDSYEVDの通信は演算部に含まれている

単位：秒

	並列数	演算	通信(S:隣接)	通信(S:大域)
SCF	128	749.472	77.435	60.057
	256	363.281	52.715	59.732
	512	172.218	38.797	64.133
	1,024	87.206	33.345	73.494
	2,048	59.728	28.637	108.404
DIAG	128	322.212	10.005	14.931
	256	181.852	6.831	9.690
	512	86.939	4.492	16.882
	1,024	46.392	4.061	17.659
	2,048	31.597	4.229	34.866
GS	128	148.354	0.000	9.629
	256	73.964	0.000	10.070
	512	37.479	0.000	9.969
	1,024	16.900	0.000	8.008
	2,048	11.451	0.000	10.687
DTCG	128	278.906	67.429	35.497
	256	107.465	45.884	39.972
	512	47.800	34.305	37.283
	1,024	23.914	29.284	47.827
	2,048	16.679	24.408	62.851

通信時間の増加が問題

RSDFTの並列特性分析（並列・単体性能）

ML:格子数，MB:バンド数

ルーチン	処理内容		演算量	高並列化性能	単体性能
DTCG	ML×ML対称行列の固有値，固有ベクトルを共役勾配法で固有値の小さいものから順にMB本求める.	レイリー商 $\frac{\langle\psi_m\|H_{KS}\|\psi_n\rangle}{\langle\psi_n\|\psi_n\rangle}$ → minimize	O(ML×ML)→O(N²) **O(N²)**	通信時間増大 演算時間と逆転 並列度の不足	行列ベクトル積 性能は悪い
GramSchmidt	規格直交化	$H_{m,n}=\langle\psi_m\|H_{KS}\|\psi_n\rangle$	O(MLxMB²)→O(N³) **O(N³)**	通信時間減少せず 演算時間と同程度 並列度の不足	行列積化で良好
DIAG	ML次元の部分空間に限ってハミルトニアンの対角化をする.				
	行列要素生成（MatE）	$\psi_n'=\psi_n-\sum_{m=1}^{n-1}\psi_m\langle\psi_m\|\psi_n\rangle$	O(MLxMB²)→O(N³) **O(N³)**	通信時間増大 演算時間と同程度 並列度の不足 Scalapackのスケーラビリティが悪い	行列積化で良好
	固有値求解（pdsyevd）	$\left(H_{N\times N}\right)\left(\vec{c}_n\right)=\varepsilon\left(\vec{c}_n\right)$	O(MB³)→O(N³) **O(N³)**		Scalapackの性能が悪い
	回転（RotV）	$\psi_n'(r)=\sum_{m=1}^{N}c_{n,m}\psi_m(r)$	O(MLxMB²)→O(N³) **O(N³)**		行列積化で良好

RSDFTの高並列化

RSDFT

- 実空間差分法
- ベクトルの内積計算が基本
- 空間並列

※高速固有値ライブラリ

Imamura el al. SNA+MC2010 (2010)

計算コアの最適化

- 行列積化

ターゲット計算機：PACS-CS, T2K-Tsukuba

スレッド並列の実装

ターゲット計算機：PACS-CS, T2K-Tsukuba

超並列向けの実装

- **バンド並列の拡張**
- **EIGENライブラリ※の適用**

ターゲット計算機：K computer

RSDFTの高並列化

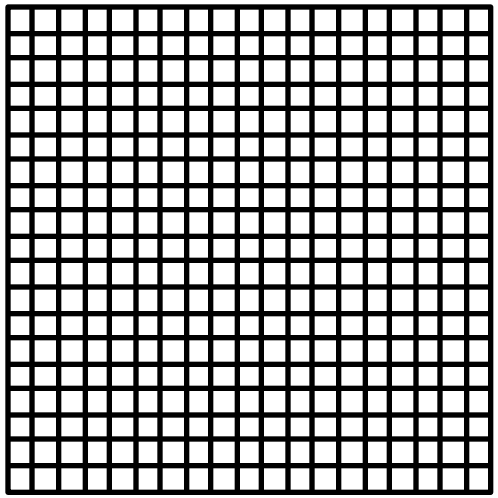
固有値方程式

$$H\varphi_i(r) = \varepsilon_i\varphi_i(r)$$

φ_i: 電子軌道（=波動関数）

i：電子準位（=エネルギーバンド）

r：空間離散点（=空間格子）



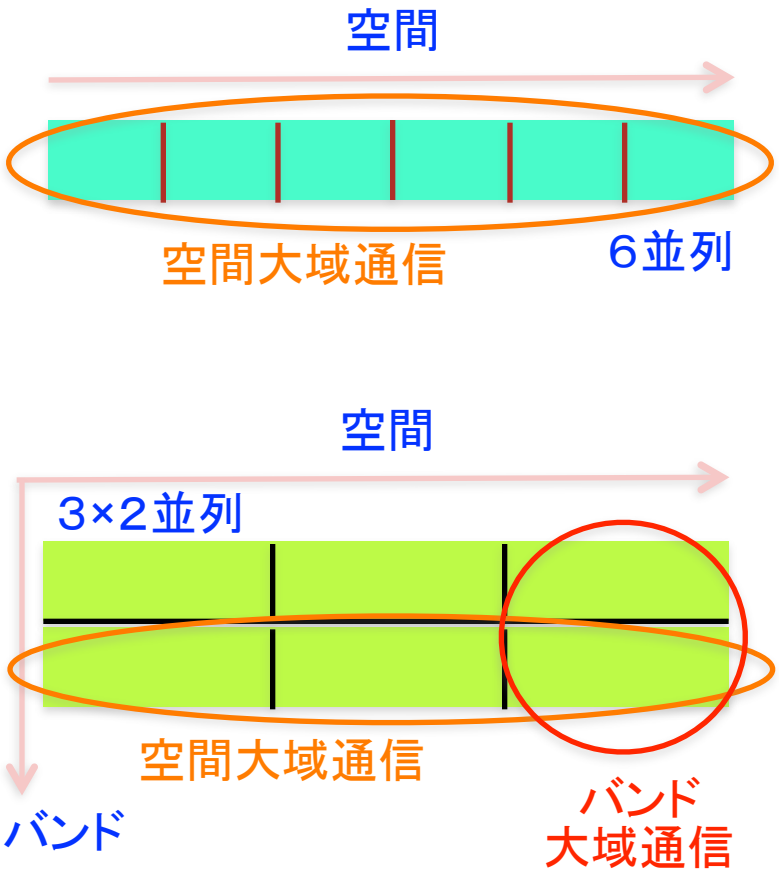
ユニットセル（実際は3次元）

- i はエネルギーバンド量子数
- i についての依存関係はない
- **空間(S)に加えエネルギーバンド(B)の並列を実装**
- 万を超える並列度を確保

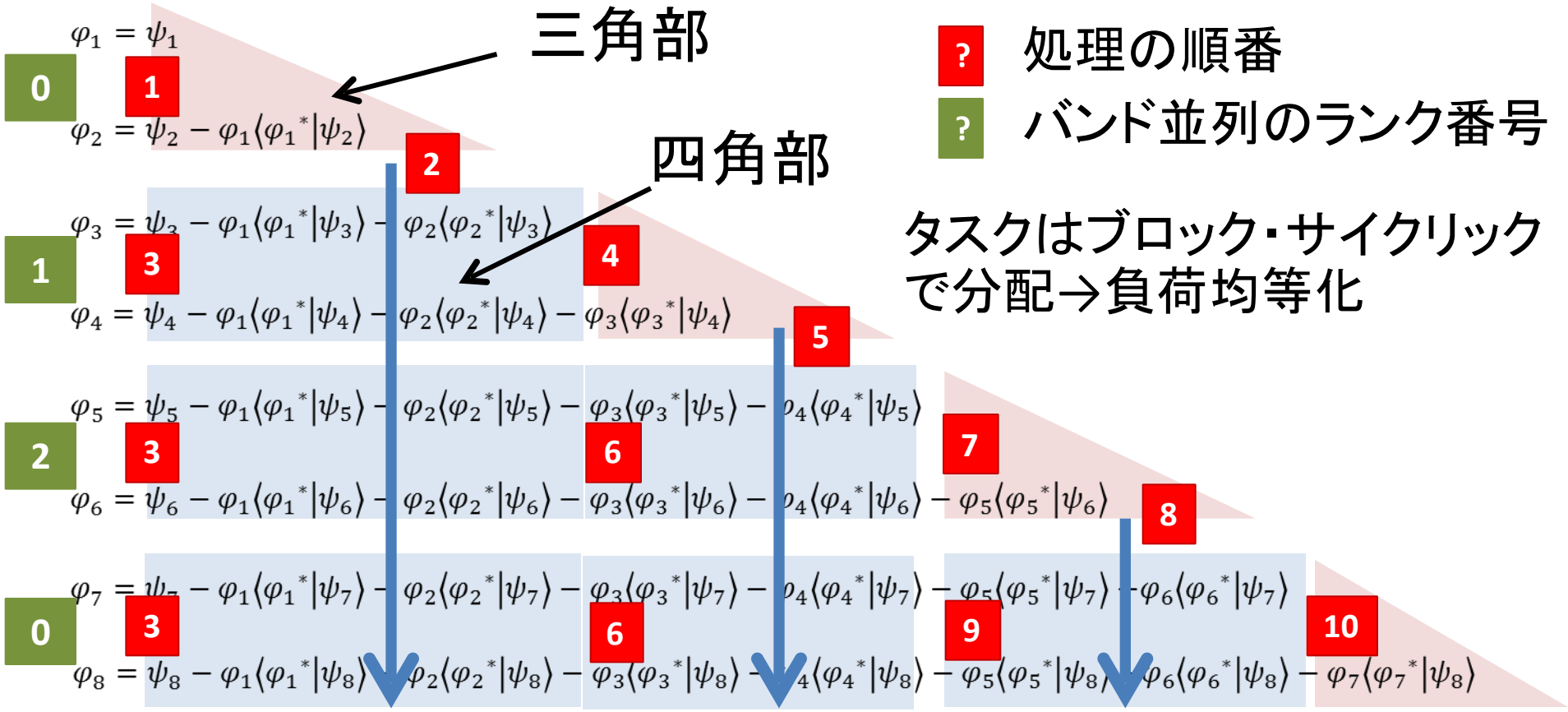
RSDFTの高並列化

並列軸拡張の効果

- 並列軸を増やす事で空間の分割粒度を増やすことが出来る
- 10万並列レベルに対応可能
- 空間並列のみの場合は全プロセッサ間の大域通信が必要
- 通信時間の増大を招く
- 2軸並列への書換で空間に対する大域通信が一部のプロセッサ間での通信とできる
- バンドに対する大域通信も同様
- 大域通信の効率化が実現可



RSDFTの高並列化- Gram-Schmidtの実装 -



(1) 三角部の計算

(2) 計算した値を四角部に転送（バンド方向の各プロセッサに分配）

(3) 四角部を並列に計算

RSDFTの高並列化 - 通信の見積りと効果の予測 -

空間＋バンド並列版（S+B並列版）

下線はバンド並列で追加された通信

■グローバル通信
- ✓ALLREDUCE
  - ➢GramSchmidt：内積配列，規格化変数
  - ➢DTCG：スカラー変数
- ✓REDUCE：
  - ➢DIAG(MatE)
- ✓BCAST:
  - ➢DIAG(RotV)
  - ➢GramSchmidt :三角部で更新した波動関数を配送
- ✓ALLGATHERV
  - ➢DIAG
  - ➢GramSchmidt

■隣接通信
- ✓境界データの交換:BCSET
- ✓ノンローカル項計算：HPSI
- ✓対称ブロックデータの交換：DIAG(MatE)

RSDFTの高並列化 - 通信の見積りと効果の予測 -

ルーチン	通信パターン	型	通信サイズ	通信回数
GramSchmidt ：バンド方向の通信	mpi_allgatherv	mpi_real8	MB/バンド並列数	1
	mpi_allreduce	mpi_real8	NBLK*NBLK～ (NBLK1+1)*(NBLK1+1)	MB/NBLK*MB/NBLK/バンド並列数 + Int(log(NBLK/NBLK1)*(MB/NBLK/バンド並列数)
	mpi_allreduce	mpi_real8	NBLK1～1	NBLK1*(MB/NBLK/バンド並列数)
	mpi_allreduce	mpi_real8	1	MB/NBLK*MB/NBLK/バンド並列数 + Int(log(NBLK/NBLK1)*(MB/NBLK/バンド並列数) +NBLK1*(MB/NBLK/バンド並列数)
	mpi_bcast	mpi_real8	ML0*NBLK	MB/NBLK/バンド並列数
DIAG	allgatherv	mpi_real8	MB/バンド並列数	1
	mpi_reduce	mpi_real8	MBLK*MBLK	(MB/MBLK * MB/MBLK)/バンド並列数
	Isend/irecv	mpi_real8	MBLK*MBLK	1
	Scalapack(pdsyevd)内の通信は省略			
	mpic_bcast	mpi_real8	MBSIZE*NBSIZE	(MB/MBSIZE * MB/NBSIZE)/バンド並列数
HPSI	mpi_isend	mpi_real8	lma_nsend(irank)*MBLK	6*各方向の深さ*MB/MBLK/バンド並列数
	mpi_irecv	mpi_real8	lma_nsend(irank)*MBLK	6*各方向の深さ*MB/MBLK/バンド並列数
	mpi_waitall	-	-	MB/MBLK/バンド並列数
BCSET	mpi_isend	mpi_real8	Md*MBLK	6*MB/MBLK/バンド並列数
	mpi_irecv	mpi_real8	Md*MBLK	6*MB/MBLK/バンド並列数
	mpi_waitall	-	-	MB/MBLK/バンド並列数

MB:バンド数，NBLK:行列ｘ行列で処理する最大サイズ，NBLK1:行列ｘベクトルで処理する最小サイズ，MBSIZE:MBxMB行列の行方向のブロックサイズ，
MBSIZE:MBxMB行列の列方向のブロックサイズ，MBLK:min(MBSIZE,NBSIZE)， Md:高次差分の次数， lma_nsend:ノンローカル項の数

RSDFTの高並列化 - 通信の見積りと効果の予測 -

ルーチン			通信パターン	型	通信サイズ	通信回数
DTCG			mpi_allreduce	mpi_real8	MB_d	MB/MB_d/バンド並列数
			mpi_allreduce	mpi_real8	MB_d	MB/MB_d/バンド並列数
			mpi_allreduce	mpi_real8	MB_d	MB/MB_d*Mcg/バンド並列数
			mpi_allreduce	mpi_real8	MB_d*6	MB/MB_d*Mcg/バンド並列数
			mpi_allreduce	mpi_real8	MB_d	MB/MB_d*Mcg/バンド並列数
			mpi_allreduce	mpi_real8	MB	2
	precond_cg		mpi_allreduce	mpi_real8	MB_d	MB/MB_d*Mcg/バンド並列数*3
		BCSET	mpi_isend	mpi_real8	Md*MB_d	6*MB/MB_d*Mcg/バンド並列数
			mpi_irecv	mpi_real8	Md*MB_d	6*MB/MB_d*Mcg/バンド並列数
			mpi_waitall	-	-	MB/MB_d*Mcg/バンド並列数
	HPSI		mpi_isend	mpi_real8	lma_nsend(irank)*MB_d	6*各方向の深さ*MB/MB_d*(Mcg+1)/バンド並列数
			mpi_irecv	mpi_real8	lma_nsend(irank)*MB_d	6*各方向の深さ*MB/MB_d*(Mcg+1)/バンド並列数
			mpi_waitall	-	-	MB/MB_d*(Mcg+1)/バンド並列数
		BCSET	mpi_isend	mpi_real8	Md*MB_d	6*MB/MB_d*(Mcg+1)/バンド並列数
			mpi_irecv	mpi_real8	Md*MB_d	6*MB/MB_d*(Mcg+1)/バンド並列数
			mpi_waitall	-	-	MB/MB_d*(Mcg+1)/バンド並列数

MB:バンド数，MB_dバンドまとめ処理数，Md:高次差分の次数， lma_nsend:ノンローカル項の数

RSDFTの高並列化 - 効果の確認 -

Weak Scaling 測定

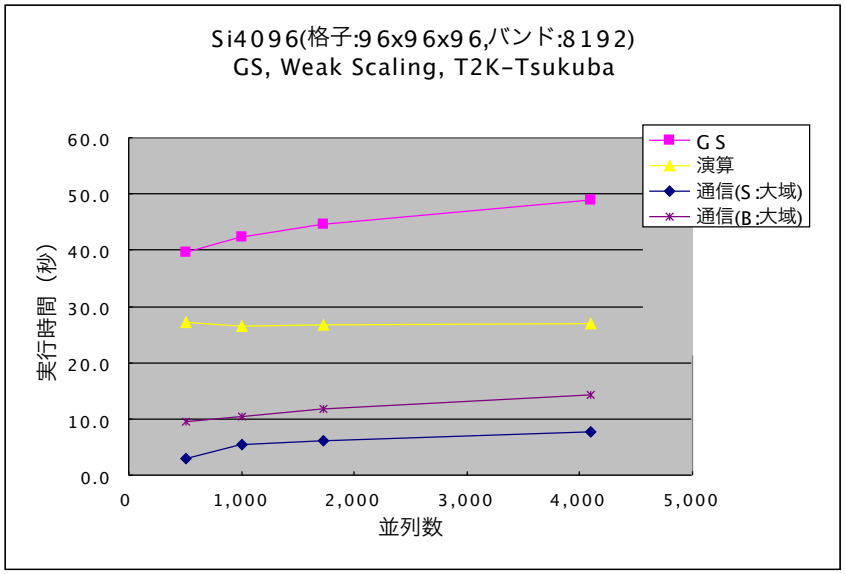
タスクサイズ/プロセスを固定する.
　　格子サイズ：12x12x12，バンドサイズ：2,400
バンド方向の並列数は8で固定.
空間方向を並列数に比例して増加させる.

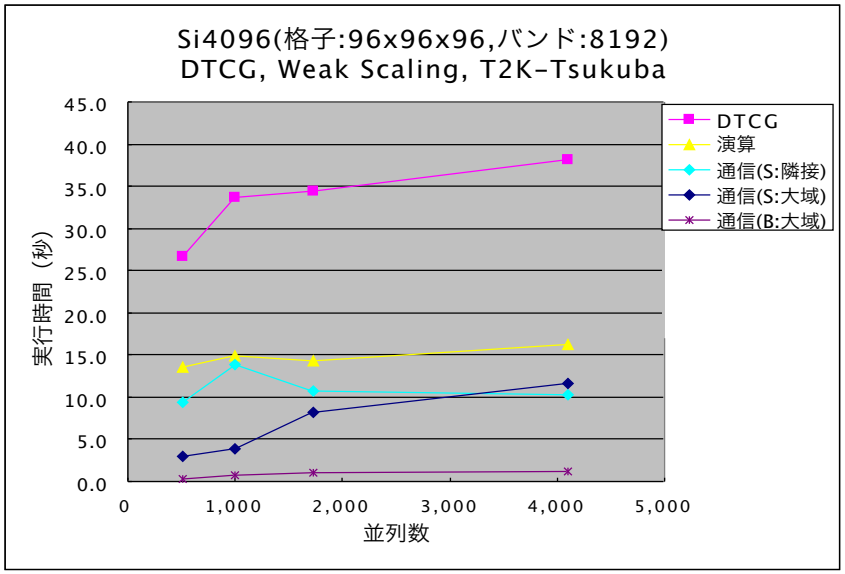
	原子数	格子数	バンド数	並列数
パターン1	512	48x48x48	19,200	512 (4x4x4x8)
パターン2	1,000	60x60x60	19,200	1,000(5x5x5x8)
パターン3	1,728	72x72x72	19,200	1,728(6x6x6x8)
パターン4	4,096	96x96x96	19,200	4,096(8x8x8x8)
パターン5	8,000	~~120x120x120~~	~~19,200~~	8,000(10x10x10x8)

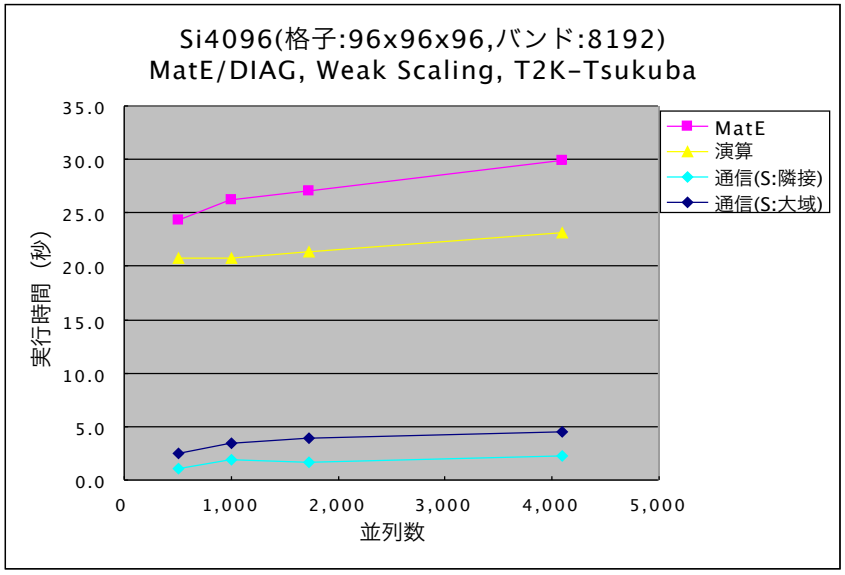
T2K-Tsukubaで測定

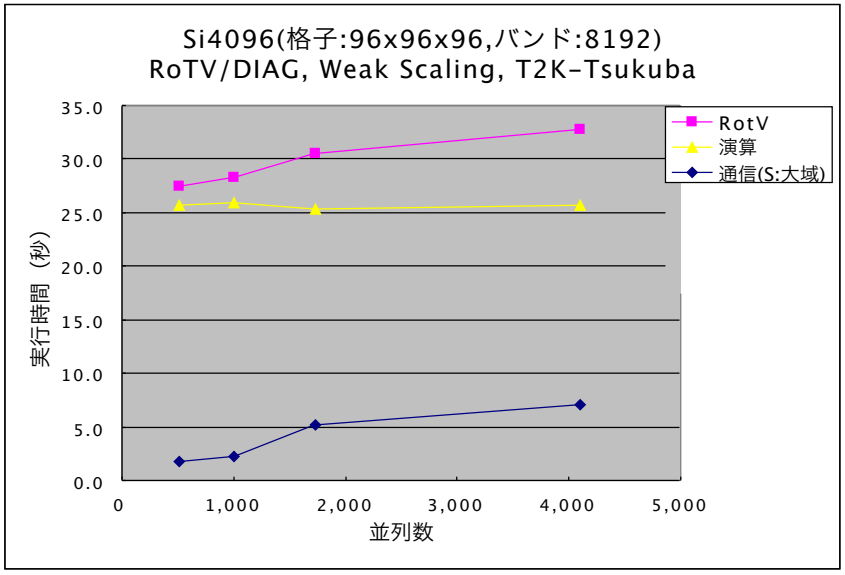
RSDFTの高並列化 - 効果の確認 -

Weak Scaling 測定





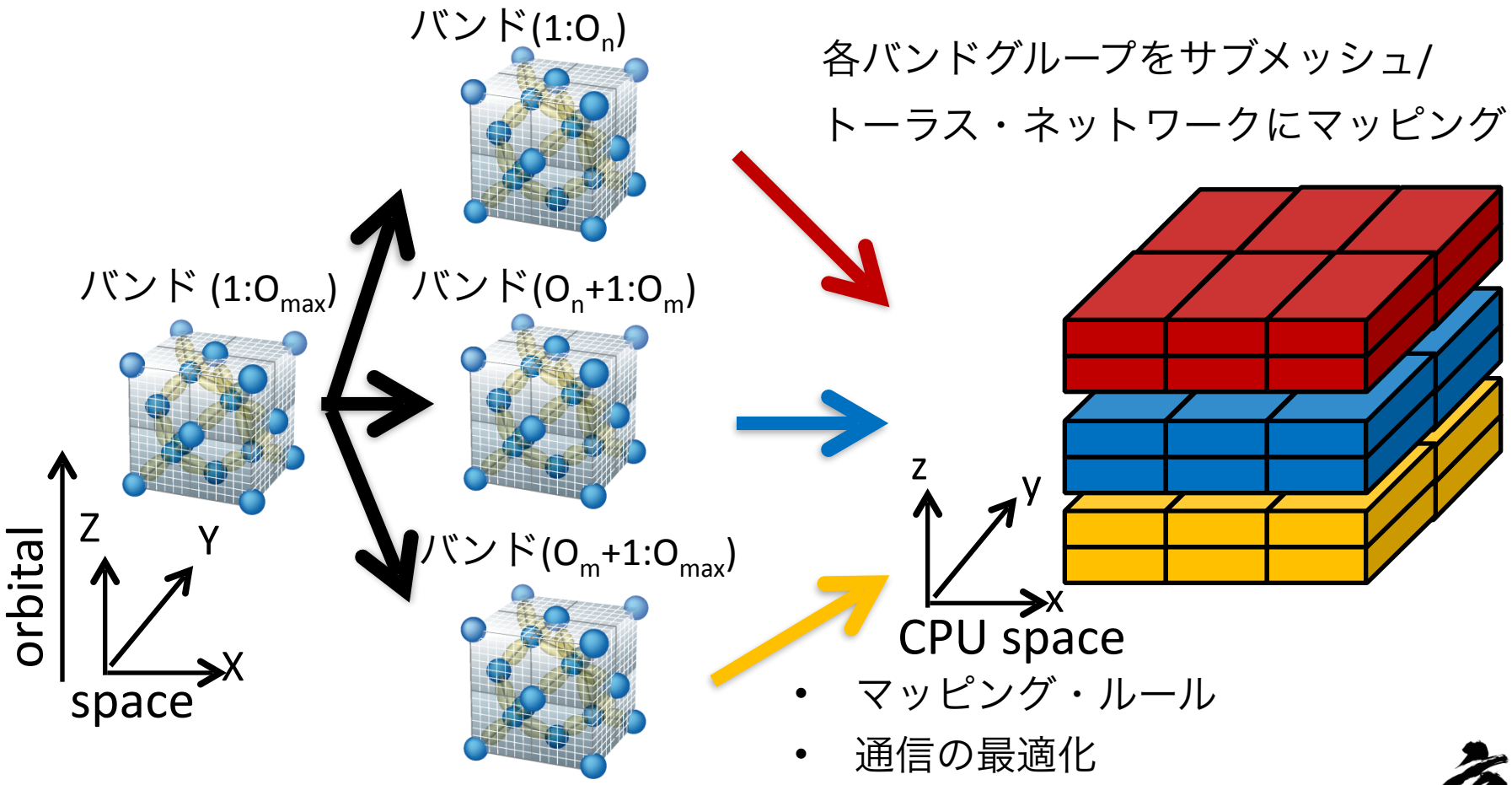




K computer

RSDFTの高並列化 -Tofuネットワークへのマッピング-

空間並列 | 空間並列＋バンド並列 | Tofuネットワークへのマッピング



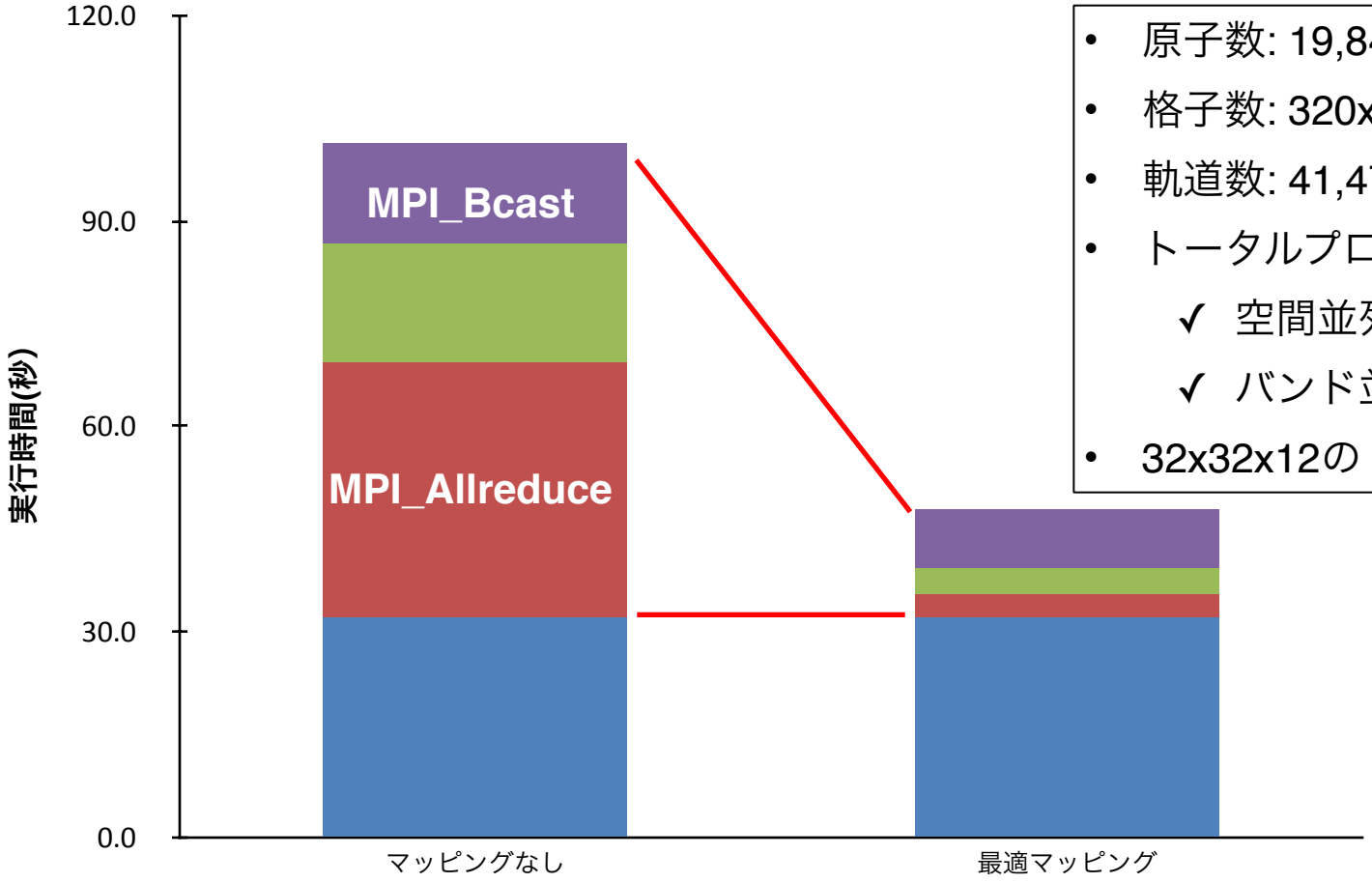
各バンドグループをサブメッシュ/トーラス・ネットワークにマッピング

- マッピング・ルール
- 通信の最適化

サブメッシュ/トーラス内で通信が閉じられる

RSDFTの高並列化 -Gram-Schmidtマッピングの効果-

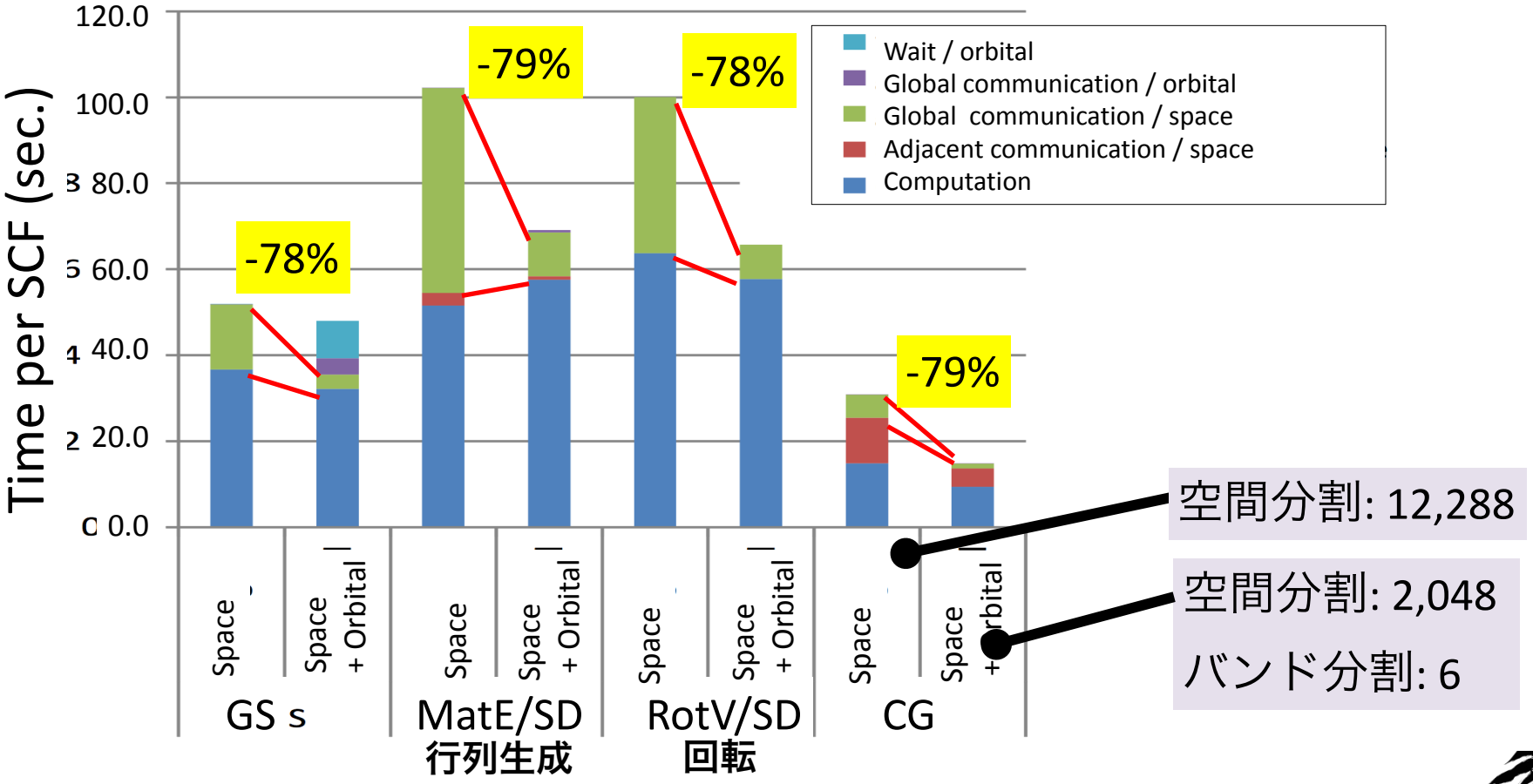
最適マッピング　→　サブコミュニケータ間のコンフリクトが発生しない
MPI通信でTofu向けアルゴリズムが選択される



- 原子数: 19,848
- 格子数: 320x320x120
- 軌道数: 41,472
- トータルプロセス数: 12,288
  - ✓ 空間並列: 2,048(32x32x2)
  - ✓ バンド並列: 6
- 32x32x12のトーラスにマッピング

RSDFTの高並列化 -マッピングの効果-

SiNW, 19,848 原子, 格子数:320x320x120, バンド数:41,472
トータル並列プロセス数は12,288で固定



大域通信時間を大幅に削減

K computer

RSDFTの高並列化-スケーラビリティー

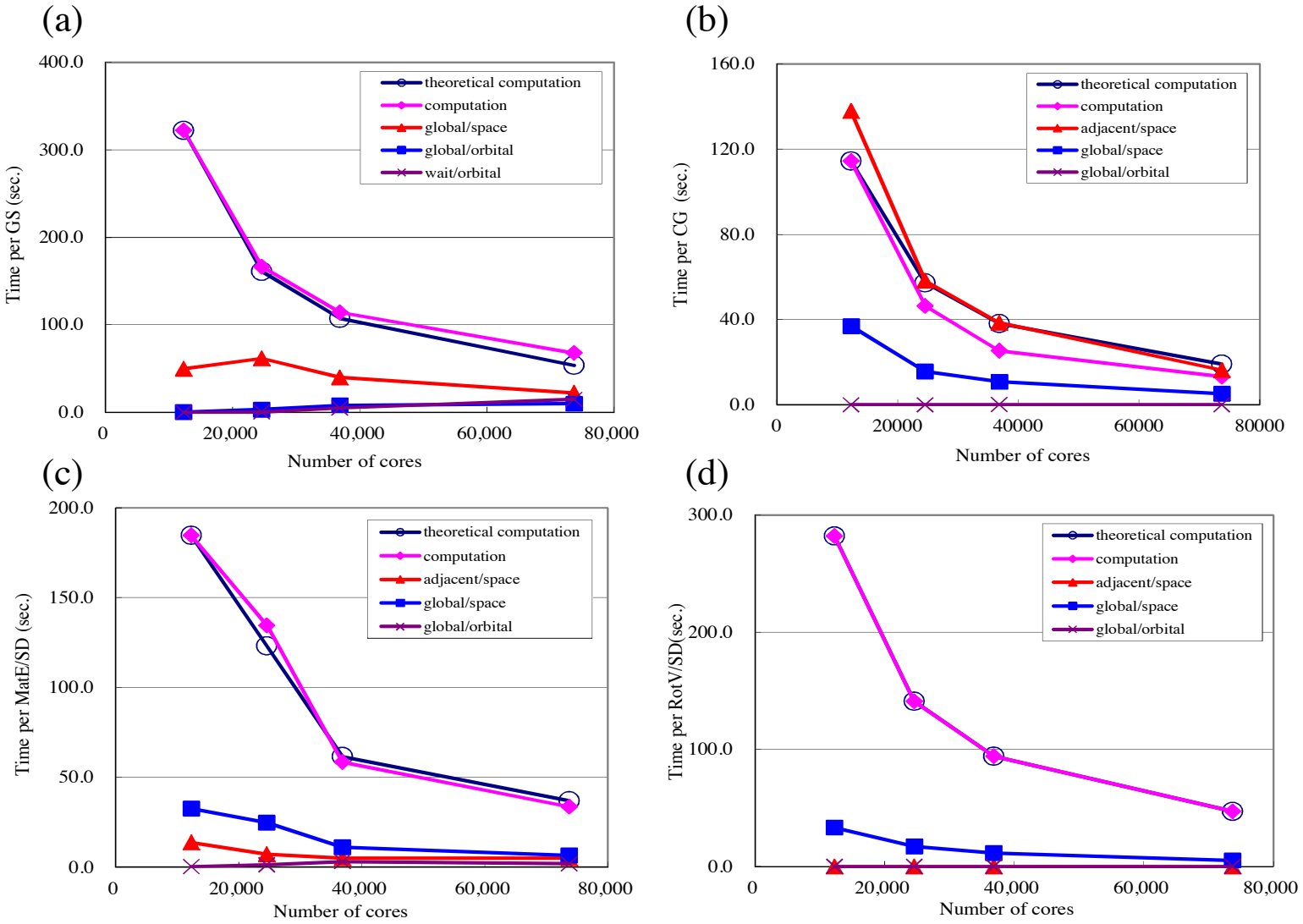


Figure 6. Computation and communication time of (a) GS, (b) CG, (c) MatE/SD and (d) RotV/SD for different numbers of cores.

RSDFTの高並列化

総合性能

Table 2. Distribution of computational costs for an iteration of the SCF calculation of the modified code.

Procedure block	Execution time (s)	Computation time (s)	Communication time (s)				Performance (PFLOPS/%)
			Adjacent/grids	Global/grids	Global/orbitals	Wait/orbitals	
SCF	2903.10	1993.89	61.73	823.02	12.57	11.89	5.48/51.67
SD	1796.97	1281.44	13.90	497.36	4.27	–	5.32/50.17
MatE/SD	525.33	363.18	13.90	143.98	4.27	–	6.15/57.93
EigenSolve/SD	492.56	240.66	–	251.90	–	–	0.01/1.03
RotV/SD	779.08	677.60	–	101.48	–	–	8.14/76.70
CG	159.97	43.28	47.83	68.85	0.01	–	0.06/0.60
GS	946.16	669.17	–	256.81	8.29	11.89	6.70/63.10

The test model was a SiNW with 107,292 atoms. The numbers of grids and orbitals were 576 × 576 × 180, and 230,400, respectively. The numbers of parallel tasks in grids and orbitals were 27,648 and three, respectively, using 82,944 compute nodes. Each parallel task had 2160 grids and 76,800 orbitals.

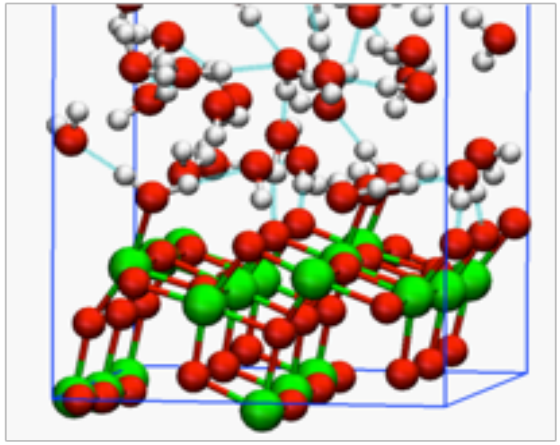
Performance evaluation of ultra-large-scale first-principles electronic structure calculation code on the K computer

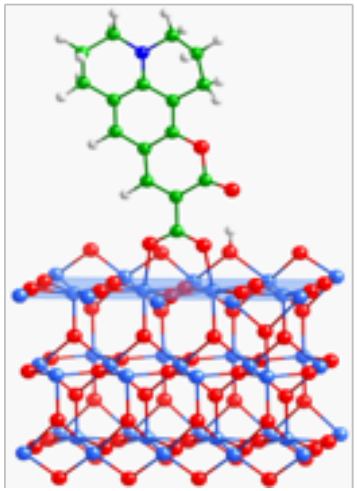
Yukihiro Hasegawa et al., *International Journal of High Performance Computing Applications* published online 17 October 2013

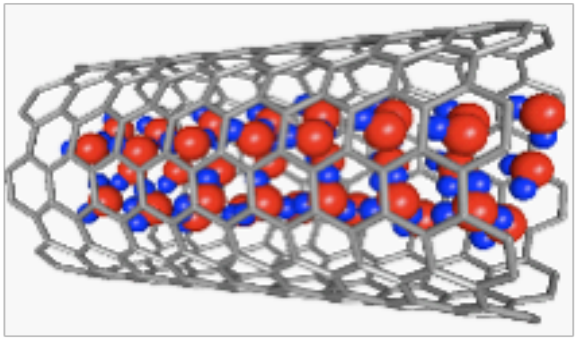
PHASEの性能最適化

PHASEとは

- ナノスケールでの量子論的諸現象を第一原理に立脚して解明し新機能を有するナノ物質・構造を予測.この点はRSDFTと同じ.
- 例えば以下のような用途に用いる.
- 繰り返し構造を持つ結晶等の解析が得意.







電子状態計算(デバイス特性，エネルギー問題，反応・拡散)，構造緩和

PHASEの原理

Kohn-Sham方程式

$$H\varphi_i(r) = \varepsilon_i \varphi_i(r)$$

波数：Gによる展開

求めたい波動関数は未知の関数のため，既知の関数の線形結合で記述する．PHASEでは平面波基底を用いる．

$$H\varphi_{ik}(G) = \varepsilon_i \varphi_{ik}(G)$$

φ_{ik}：電子軌道（=波動関数）

i：電子準位（=エネルギーバンド量子数）

G：波数格子

k：k点

PHASEの計算フロー

解くべき方程式

$H|\psi\rangle = \varepsilon_n|\psi\rangle$

繰り返しによる更新

$|\psi\rangle^i$：波動関数 の更新

$\Delta^{i+1} <= F(\mathbf{H|\psi\rangle^i} , |\psi\rangle^i)$

$|\psi\rangle^{i+1} <= |\psi\rangle^i + \Delta^{i+1}$

$|\psi\rangle^{i+1}$ の直交化

$\langle\psi|\psi\rangle^{i+1}$：電荷の更新

H　　：ポテンシャルの更新

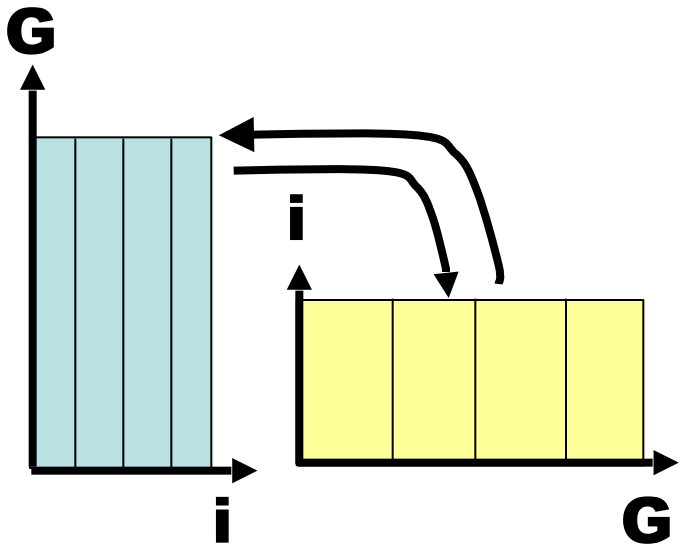
・H|ψ>と|ψ>より次のステップの|ψ>の修正量を決定する。
・決定した修正量を加算し次のステップの|ψ>を計算する。
・$|\psi\rangle^i$を直交化のために修正する。

・|ψ>がある条件を満たしたら収束。

PHASEの並列化

$$H\varphi_{ik}(G) = \varepsilon_i \varphi_{ik}(G)$$

- i はエネルギーバンド量子数
- 基本的にエネルギーバンドについて並列化されている
- 一部，波数：Gについて並列化されている
- G並列の前にエネルギーバンド並列されている波動関数をG並列可能なようにトランスバース転送が発生
- G並列後にG並列されている波動関数をエネルギーバンド並列に戻すためなトランスバース転送が発生
- このトランスバース転送のコストが大



PHASEの並列特性分析（処理・演算量）

- カーネルの抽出

種類	区間番号	
行列−行列積に書き換え可能	2,4,5,9,10	$O(N^3)$
FFTを含む	1,3,6,7,8,11	O(NlogN)
対角化	9	$O(N^3)$

抽出されたカーネルは以下の11区間.

区間1: V_{local}の逆FFT

区間2: $V_{nonlocal}$を波動関数ψ_iとβの内積fiに作用

区間3: V_{local}を波動関数ψ_iに作用，波動関数の修正値$H\psi_i$を計算

区間4: $f_{ijt}=\beta\cdot\psi$の計算

区間5: Gram−Schmidtの直交化

区間6: 固有値計算，波動関数ψ_iとf_iのバンド方向並べ替え

区間7: 電荷密度計算

区間8: V_{local}の逆FFT

区間9: 行列対角化計算，波動関数ψ_iの修正

区間10: $f_{ijt}=\beta\cdot\psi$の計算

区間11: 電荷密度，ポテンシャル，全エネルギー計算

以上のカーネルを計算特性別に分類すると3つに分類が可能である.

PHASEの並列特性分析（ブロック毎のスケーラビリティ）

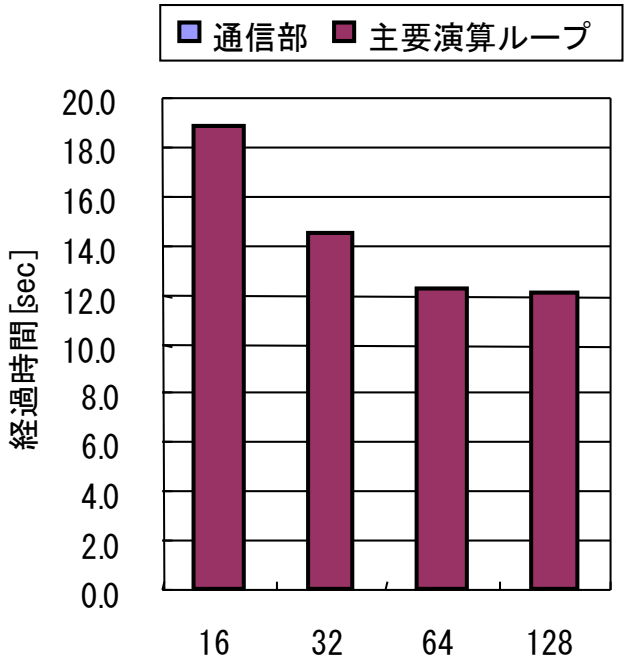
- 行列積カーネル

区間2: $V_{nonlocal}$を波動関数ψ_iとβの内積fiに作用

区間4: $f_{ijt}=\beta \cdot \psi$の計算

区間10: $f_{ijt}=\beta \cdot \psi$の計算

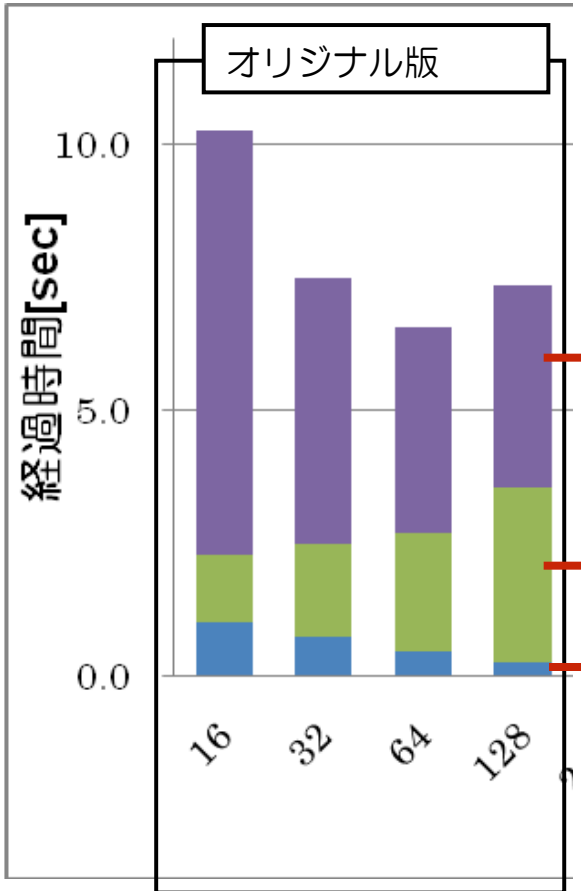
- PHASEの**処理ブロック:区間2**を例に示す．
- 低並列でストロングスケールで測定．
- $HfSiO_2$ 384原子アモルファス系を測定
- すでにこの並列度でスケールしていない．
- 原因は**非並列部の残存**．
- 区間4，10も同様




```
subroutine m_es_vnonlocal_w(ik,iksnl,ispin,switch_of_eko_part)
    +-call tstatc0_begin
    | loop_ntyp: do it = 1, ntyp
    |   loop_natm : do ia = 1, natm ----------原子数のループ
    |   +-call calc_phase
    |     T-do lmt2 = 1, ilmt(it)
    |     +-call vnonlocal_w_part_sum_over_lmt1
    |       +-call add_vnlph_l_without_eko_part
    |         subroutine add_vnlph_l_without_eko_part()
    |           T-if(kimg == 1) then
    |           | T-do ib = 1, np_e  ----------エネルギーバンド並列部
    |           |   T-do i = 1, iba(ik)
    |           |   V-end do
    |           | V-end do
    |           +-else
    |           | T-do ib = 1, np_e  ----------エネルギーバンド並列部
    |           |   T-do i = 1, iba(ik)
    |           |   V-end do
    |           | V-enddo
    |           V-end if
    |         end subroutine add_vnlph_l_without_eko_part
    |     V-end do
    |   V-end do loop_natm
    | V-end do loop_ntyp
end subroutine m_es_vnonlocal_w
```

PHASEの並列特性分析 (ブロック毎のスケーラビリティ)

■行列積カーネル　区間5: Gram-Schmidtの直交化



- PHASEの**処理ブロック:区間5**を例に示す.
- Si512原子の結果.
- 低並列でストロングスケールで測定.
- すでにこの並列度でスケールしていない.
- 演算もスケールしないが通信が増大
- トランスバース転送が原因

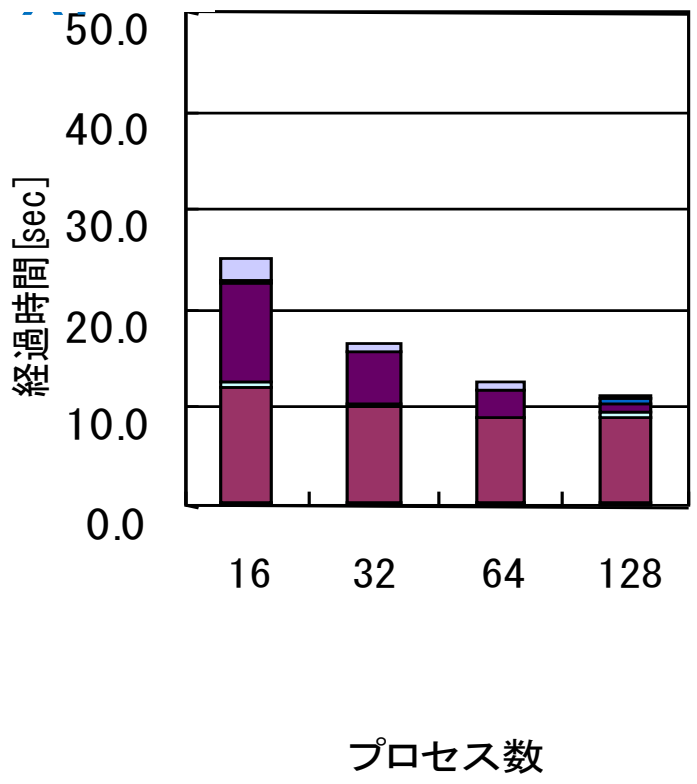
演算処理

通信処理

トランスバース処理
(通信以外)

PHASEの並列特性分析（ブロック毎のスケーラビリティ）

■FFTカーネル　区間8: V_{local}の逆FFT



- PHASEの**処理ブロック:区間8**を例に示す．
- 低並列でストロングスケールで測定．
- すでにこの並列度でスケールしていない．
- 384原子，800程度のエネルギーバンド数．
- エネルギーバンド並列のみでは128並列にも到達しない．

PHASEの高並列化・高性能化の結果

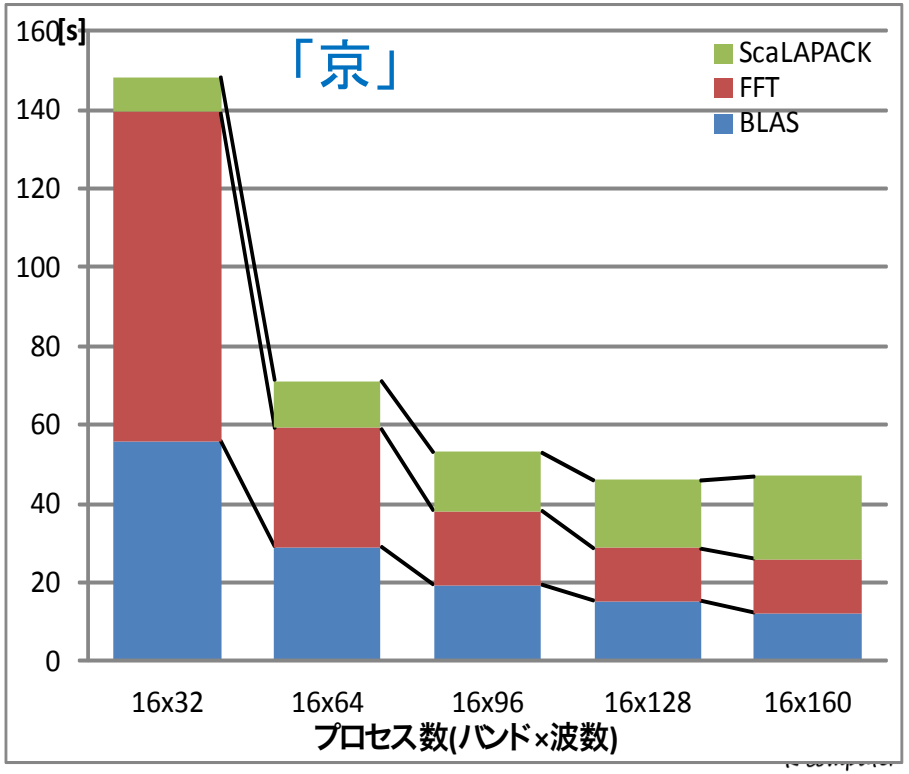
■対角化カーネル

- 対角化カーネル(区間９）
- $HfSiO_2$ 1,536原子，5,120元アモルファス系で測定.
- 500並列以上で並列オーバーヘッドが測定された.

「京」

ScaLAPACKを含むカーネルの並列特性.

カーネル	512 (16x32)	1024 (16x64)	1536 (16x96)	2048 (16x128)
区間９(秒)	11.4	13.8	16.5	18.8
通信	0.0	0.0	0.0	0.0
BLAS	2.1	1.1	0.7	0.5
ScaLAPACK	8.8	12.0	15.1	17.5
他	0.5	0.7	0.7	0.8



PHASEの高並列化・高性能化

固有値方程式

$$H\varphi_{ik}(G) = \varepsilon_i \varphi_{ik}(G)$$

φ_{ik}：電子軌道（=波動関数）

i：電子準位（=エネルギーバンド量子数）

G：波数格子

k：k点

- エネルギーバンド(B)に加え波数(G)の並列を実装
- 完全な２軸並列とする
- RSDFTと同様の行列行列積化も実装

PHASEの高並列化・高性能化

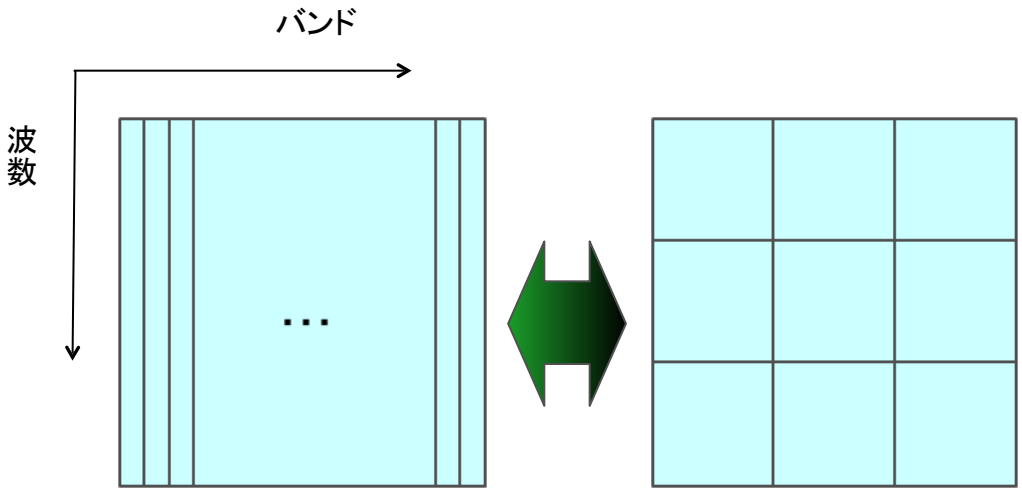
- 二軸並列化
  - 行列積カーネル
    - 従来の非並列部を並列化できる.
    - グラムシュミットの直交化処理のトランスバース転送が削減できる.

```
subroutine m_es_vnonlocal_w(ik,iksnl,ispin,switch_of_eko_part)
  +-call tstatc0_begin
  | loop_ntyp: do it = 1, ntyp
  |   loop_natm : do ia = 1, natm ---------原子数のループ
  |   +-call calc_phase
  |     T-do lmt2 = 1, ilmt(it)
  |     +-call vnonlocal_w_part_sum_over_lmt1
  |       +-call add_vnlph_1_without_eko_part
  |         subroutine add_vnlph_1_without_eko_part()
  |           T-if(kimg == 1) then
  |           | T-do ib = 1, np_e  ---------エネルギーバンド並列部
  |           |   T-do i = 1, iba(ik)
  |           |   V-end do
  |           | V-end do
  |           +-else
  |           | T-do ib = 1, np_e  ---------エネルギーバンド並列部
  |           |   T-do i = 1, iba(ik)
  |           |   V-end do
  |           | V-enddo
  |           V-end if
  |         end subroutine add_vnlph_1_without_eko_part
  |     V-end do
  |   V-end do loop_natm
  | V-end do loop_ntyp
end subroutine m_es_vnonlocal_w
```

非並列部が波数で並列化できる

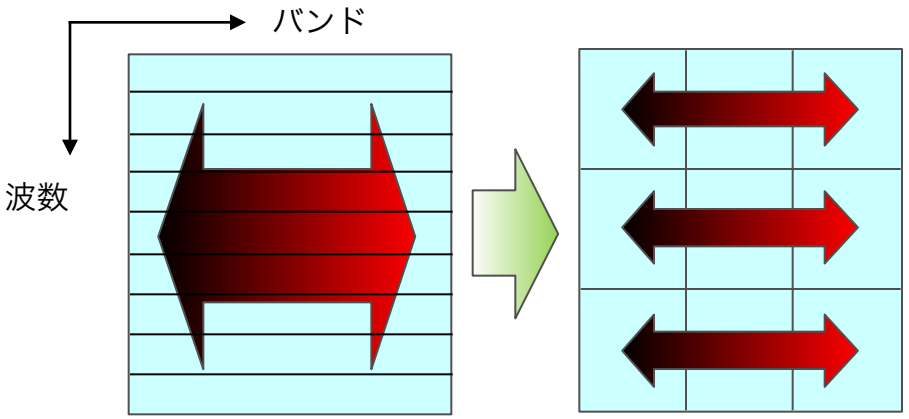
PHASEの高並列化・高性能化

- 二軸並列化
  - 行列積カーネル
    - 分割粒度が大きくなる.
    - ループの回転長が増えることで，並列性能が高まる.
    - cf. バンド方向のループ長が1/9から1/3と3倍に増える.



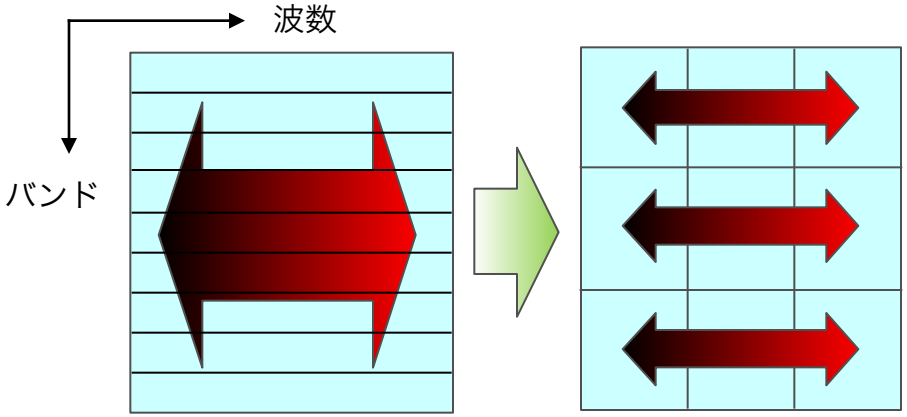
PHASEの高並列化・高性能化

- 二軸並列化
  - 行列積カーネル
    - バンド方向の大域通信が一部のプロセッサに閉じるため通信時間の短縮が図れる.



PHASEの高並列化・高性能化

- 二軸並列化
  - FFTカーネル
    - オリジナルのFFTカーネルは波数方向に並列化されていなかった.
    - そのためFFTに関する通信は発生していなかった.
    - 二軸並列化に伴いFFTに関する通信が発生する.
    - FFT通信の問題は全プロセッサ間の転置通信.
    - 二軸並列化では通信は全プロセッサでなく一部のプロセッサに閉じる.



PHASEの高並列化・高性能化

二軸並列化

- この分野では小規模問題を短時間で計算したいという科学的要求が高い.
- バンド計算(エネルギー準位など)：1万原子の1回SCF収束で良い～100SCF程度.
- 構造緩和(MD)や反応経路探索：外側に原子核の緩和に関するループ構造～100step程度.
- 10,000原子を10PFシステム(８0,000ノード), また10,00原子を10,000ノードで計算する事を目指せる.
- **ただし二軸並列はメリットとデメリットがあるため実施前に効果が期待できるか詳細な評価を実施した.**

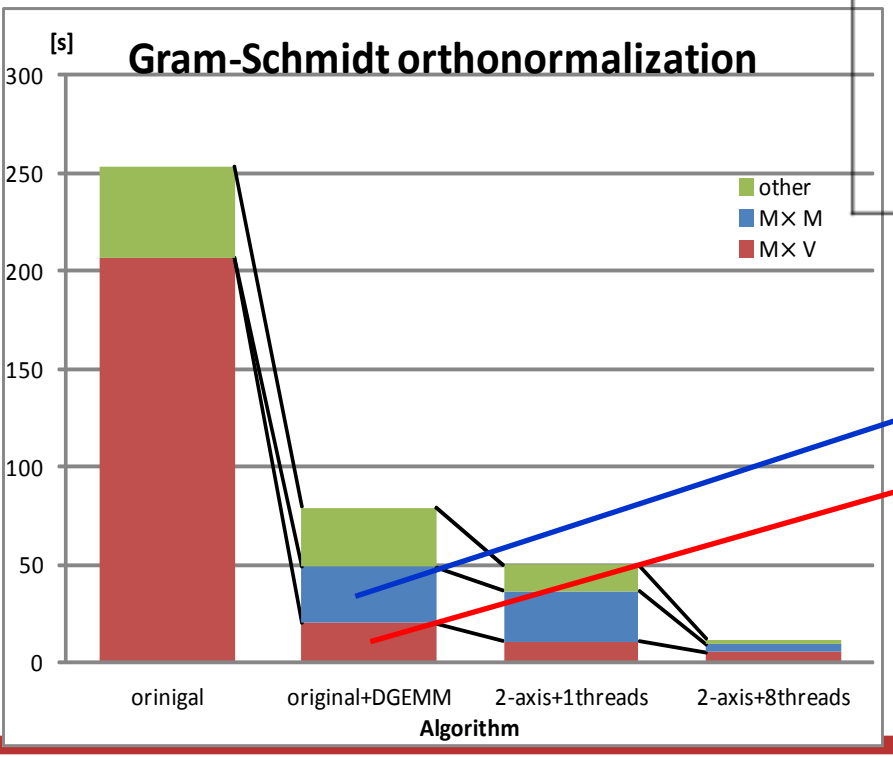
PHASEの高並列化・高性能化の結果

スレッド並列化
キャッシュの有効利用－行列積化

FX1　　「京」

サブルーチン	時間 [sec]	比率 [%]	演算効率 [%]	時間 [sec]	比率 [%]	演算効率 [%]
区間5	512.7	100.0	28.23	11.8	100.0	23.07
m_ES_F_transpose_r	105.3	20.5	0	0.0	0.1	0.00
m_ES_W_transpose_r	15.7	3.1	0	0.1	0.5	0.00
WSW_t	15.2	3.0	0.036	1.2	9.8	0.10
normalize_bp_and_psi_t	0.9	0.2	3.25	0.4	3.3	0.13
W1SW2_t_r	**49.2**	**9.6**	**5.46**	**3.7**	**31.8**	**2.30**
modify_bp_and_psi_t_r	**50.8**	**9.9**	**4.45**	**1.7**	**14.8**	**3.75**
W1SW2_t_r_block	**162.0**	**31.6**	**41.89**	**2.3**	**19.4**	**56.82**
modify_bp_and_psi_t_r_block	**96.2**	**18.8**	**74.65**	**2.1**	**17.6**	**60.90**
m_ES_W_transpose_back_r	14.1	2.8	0	0.0	0.3	0.00
m_ES_F_transpose_back_r	1.3	0.3	0	0.0	0.1	0.00

「京」



FX1(左), 「京」(右)上でのGram-Schmidt直交化のBLAS Level3適用結果

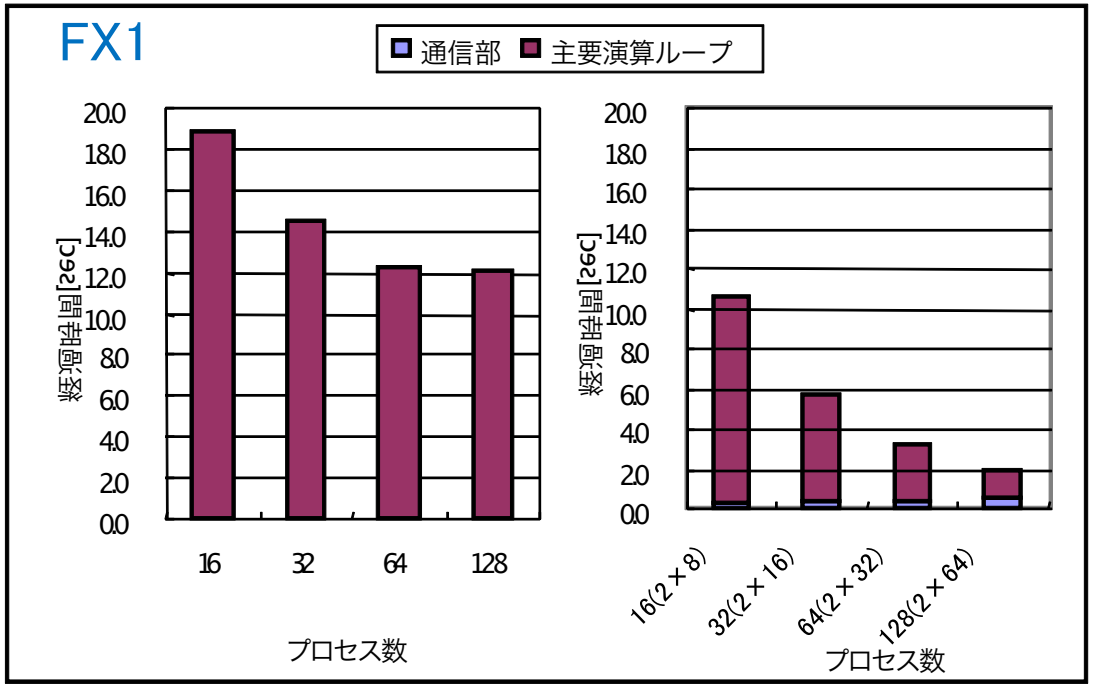
行列積

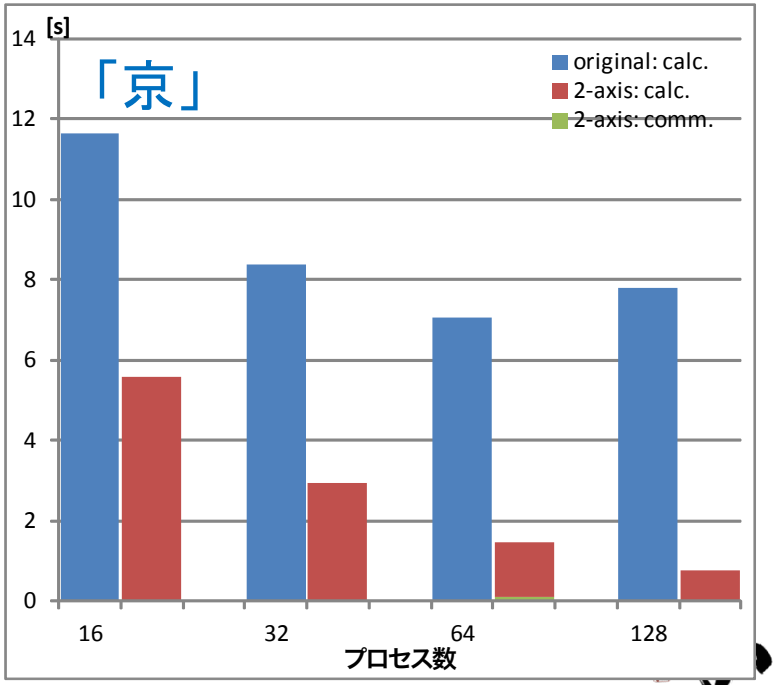
行列ベクトル積

K computer

PHASEの高並列化・高性能化の結果

- 二軸並列化
  - 行列積カーネル
    - 行列積化されたカーネルに(区間2)ついての結果.
    - $HfSiO_2$ 384原子アモルファス系のデータ.
    - 大幅な性能向上を達成.

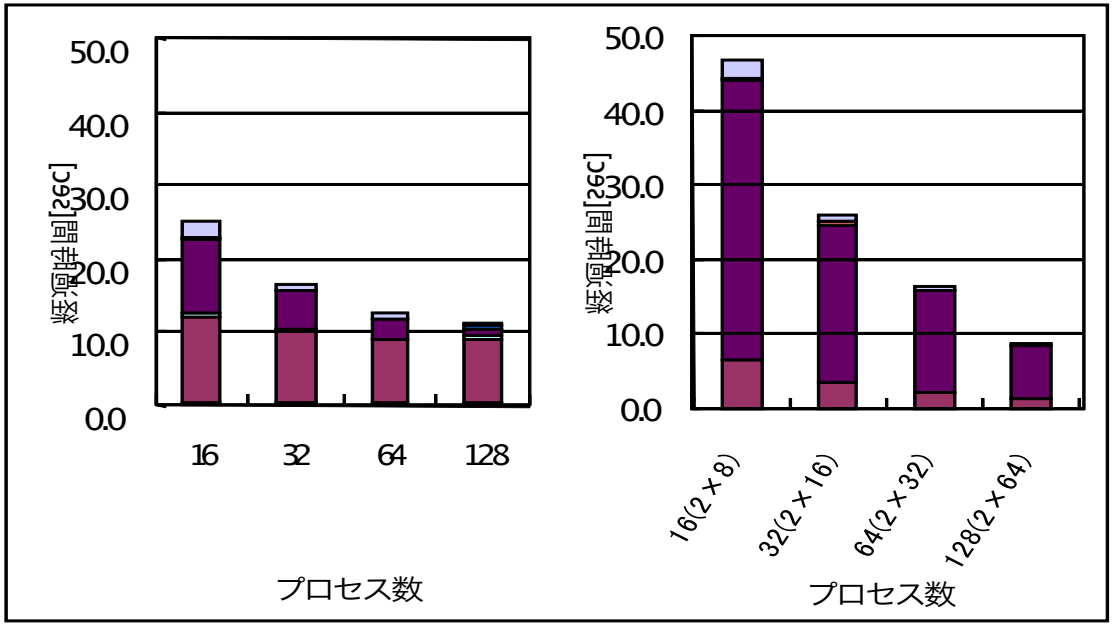


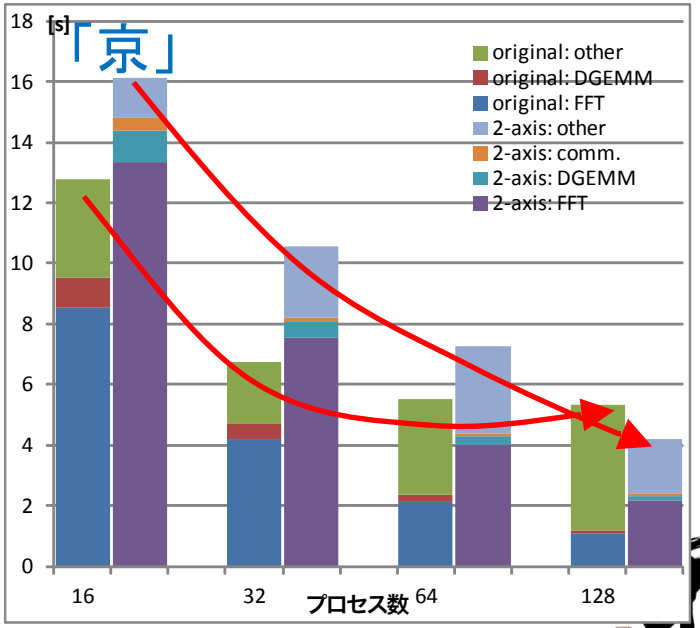


PHASEの高並列化・高性能化の結果

- 二軸並列化
  - FFTカーネル
    - FFTを含むカーネルに(区間8)ついての結果.
    - $HfSiO_2$ 384原子アモルファス系のデータ.
    - 性能向上を達成.

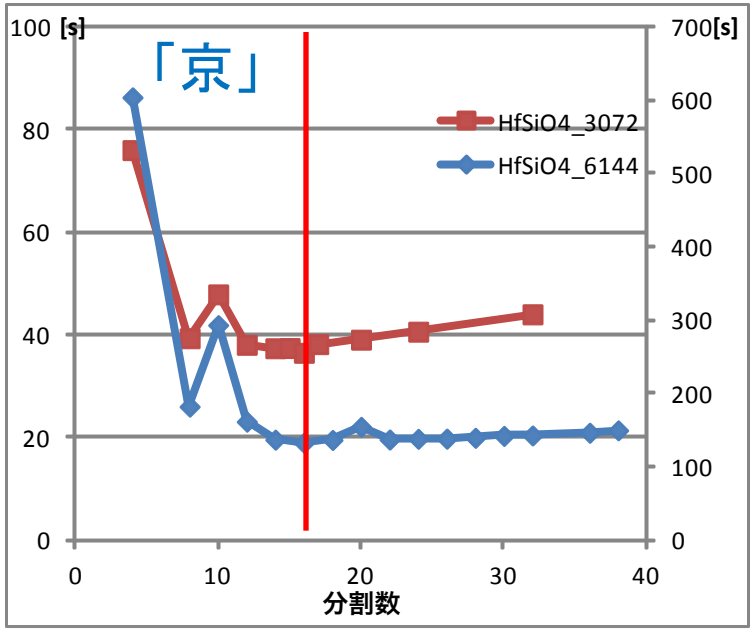
FX1



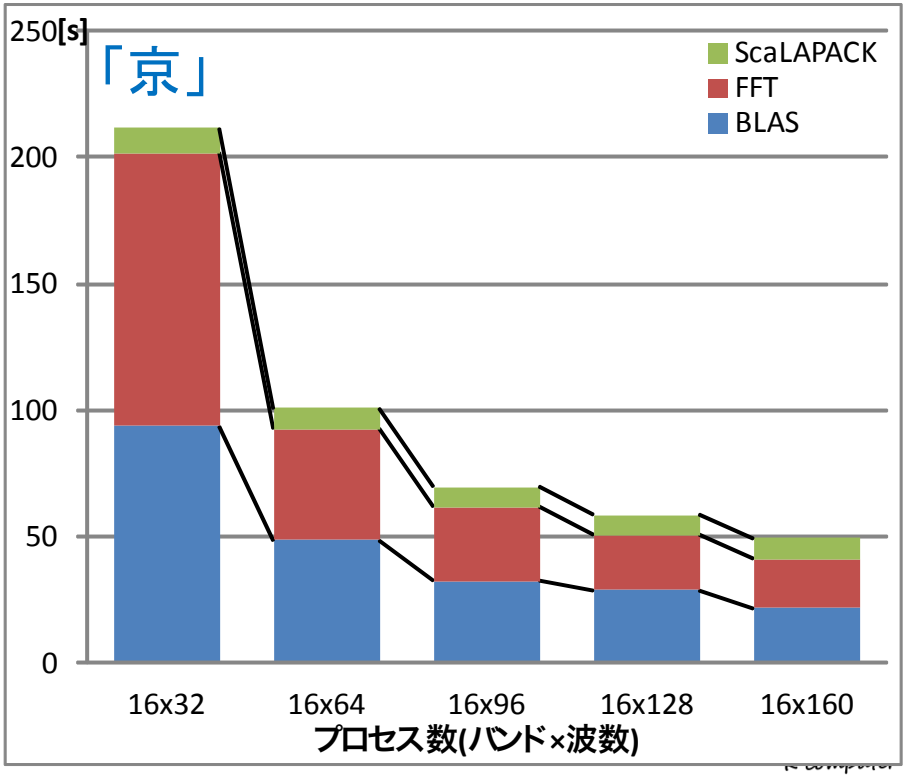


PHASEの高並列化・高性能化の結果

- Scalapack分割数の固定
  - 対角化カーネル
    - 対角化はエネルギーバンド数の元を持つ行列が対象
    - 行列の大きさに比べて分割数が多すぎる
    - 分割数を16×16=256に固定



区間9の実行時間と分割数の関係.



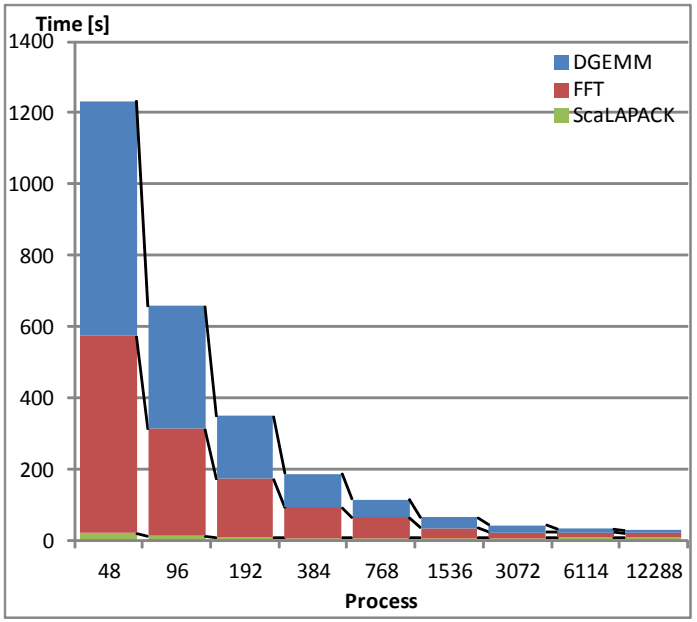
PHASEの高並列化・高性能化の結果

総合性能

- 「京」3,072並列にて，SiC 4,096原子計算にて，構造緩和 (263MD, 2days).
- 「京」82,944並列にて，SiC 20,440原子計算にて，MSDソルバー効率 20.2 % (2.1 PFLOPS) 達成.

Kernel		Time [sec]	Efficiency of theoretical Peak
SCF		39.78	20.11%
	DGEMM	13.19	49.73%
	FFT	14.46	7.86%
	ScaLAPACK	12.31	3.88%

「京」で測定した並列性能(SiC 3.800原子系)



まとめ

理研で進めた性能最適化
RSDFTの性能最適化
PHASEの性能最適化