A01高橋班
大規模並列環境における
数値計算アルゴリズム

研究代表者：高橋大介
筑波大学システム情報系

研究組織

- 研究代表者
  - 高橋大介（筑波大学・准教授）：
    研究統括および高速アルゴリズム
- 研究分担者
  - 今村俊幸（電気通信大学・准教授）：性能チューニング
  - 多田野寛人（筑波大学・助教）：大規模線形計算
- 連携研究者
  - 佐藤三久（筑波大学・教授）：並列システムの性能評価
  - 朴泰祐（筑波大学・教授）：演算および通信性能の最適化
  - 櫻井鉄也（筑波大学・教授）：数値アルゴリズム

平成23年度の研究内容

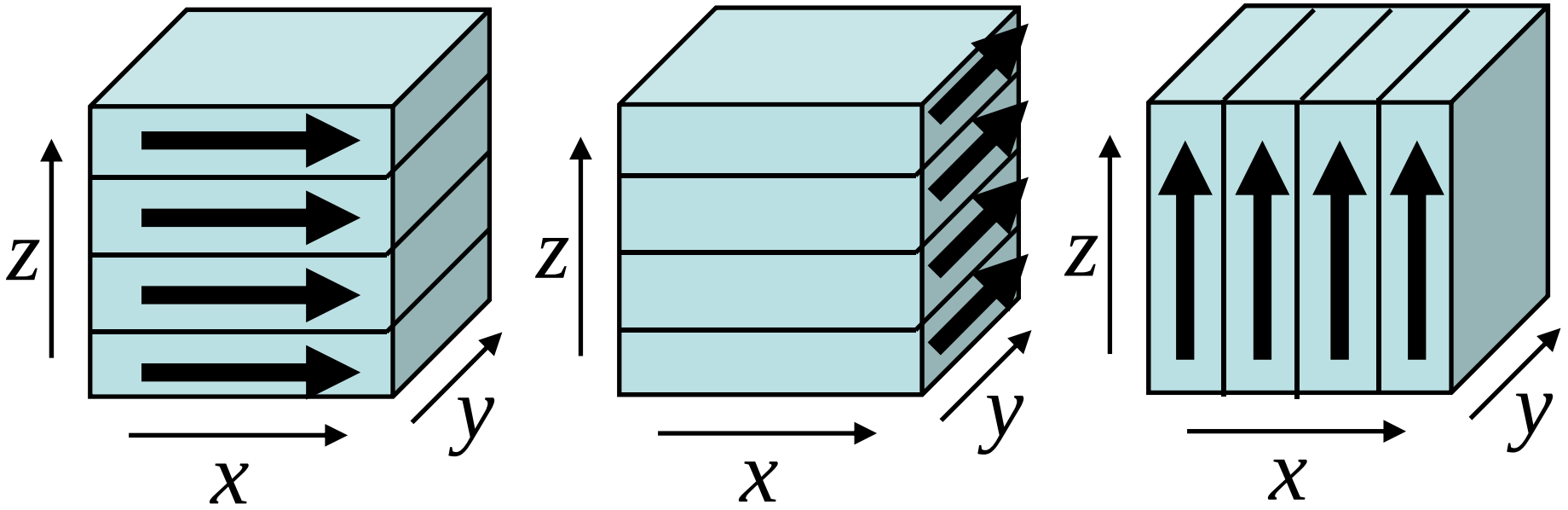
- 研究代表者：高橋大介（筑波大学）：
  - ペタスケール計算環境におけるFFTライブラリ
  - GPUによる3倍・4倍精度BLASの実装と評価
  - GPUにおける格納形式自動選択による疎行列ベクトル積の高速化
- 研究分担者：今村俊幸（電気通信大学）：
  - GPGPU技術の線形計算への展開
- 研究分担者：多田野寛人（筑波大学）：
  - 複数右辺ベクトルをもつ連立一次方程式に対する高速・高精度Block Krylovアルゴリズムの開発

ペタスケール計算環境における FFTライブラリ

- 計算科学アプリケーションプログラムのいくつかにおいては，三次元FFTが律速になっている.
- x，y，zのうちz方向のみに一次元分割した場合，超並列化は不可能.
  - 1,024×1,024×1,024点FFTを2,048プロセスで分割できない（1,024プロセスまでは分割可能）
- y，zの二次元分割で対応する.
  - 1,024×1,024×1,024点FFTが1,048,576（=1,024×1,024）プロセスまで分割可能になる.

z方向に一次元ブロック分割した場合の並列三次元FFT

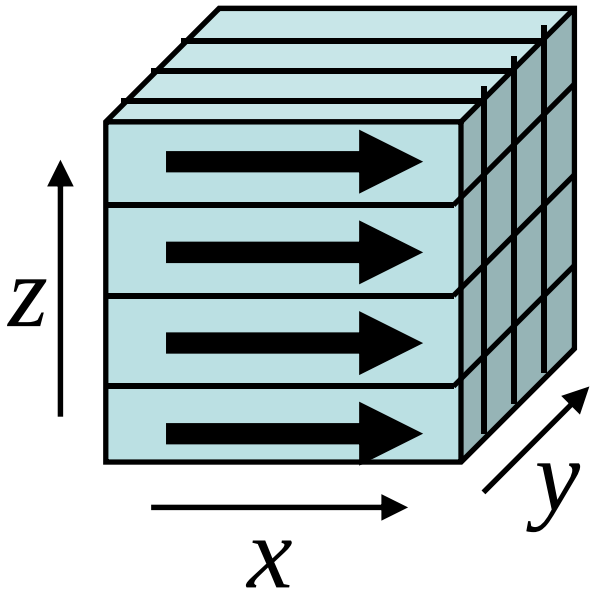
1. x方向FFT　　2. y方向FFT　　3. z方向FFT



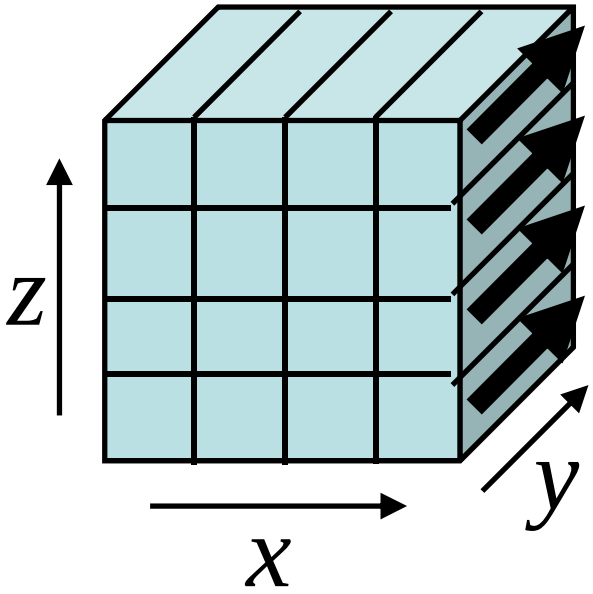
各ノードでslab形状に分割

y, z方向に二次元ブロック分割した場合の並列三次元FFT

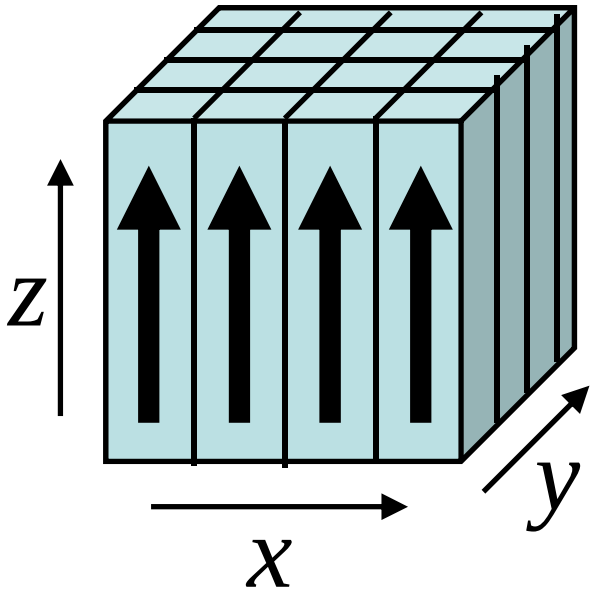
1. x方向FFT



2. y方向FFT



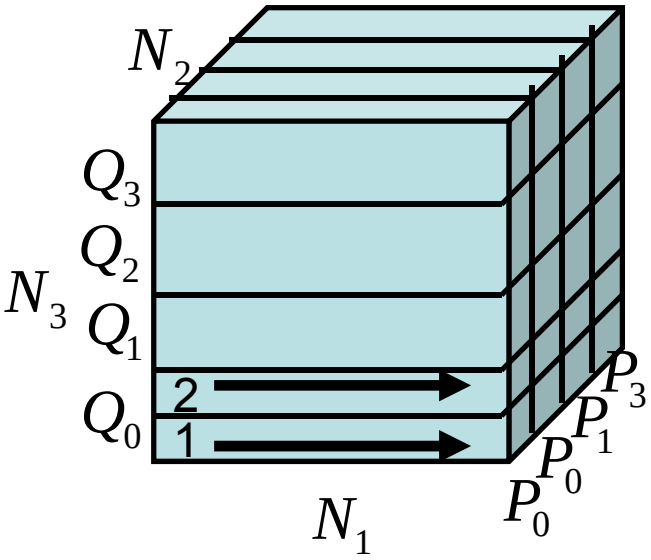
3. z方向FFT



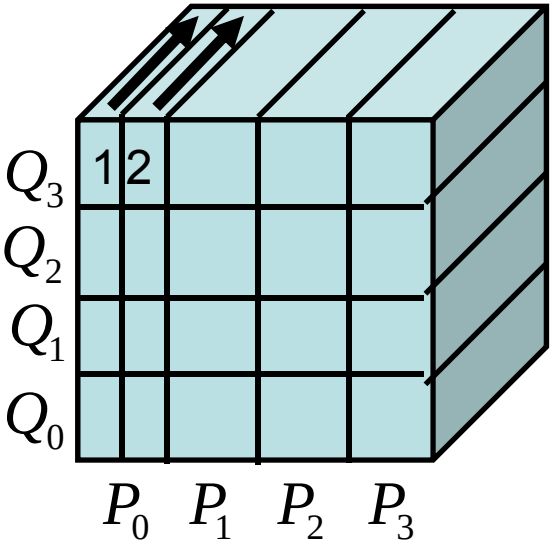
各ノードでpencil形状に分割

並列三次元FFTにおける演算と通信のオーバーラップ

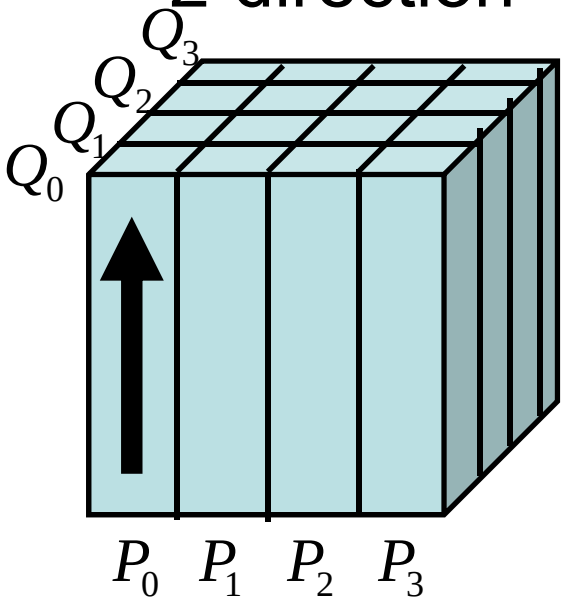
1. FFTs in x-direction

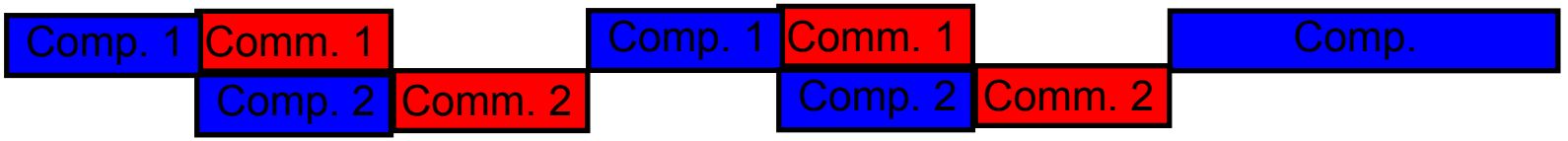


2. FFTs in y-direction

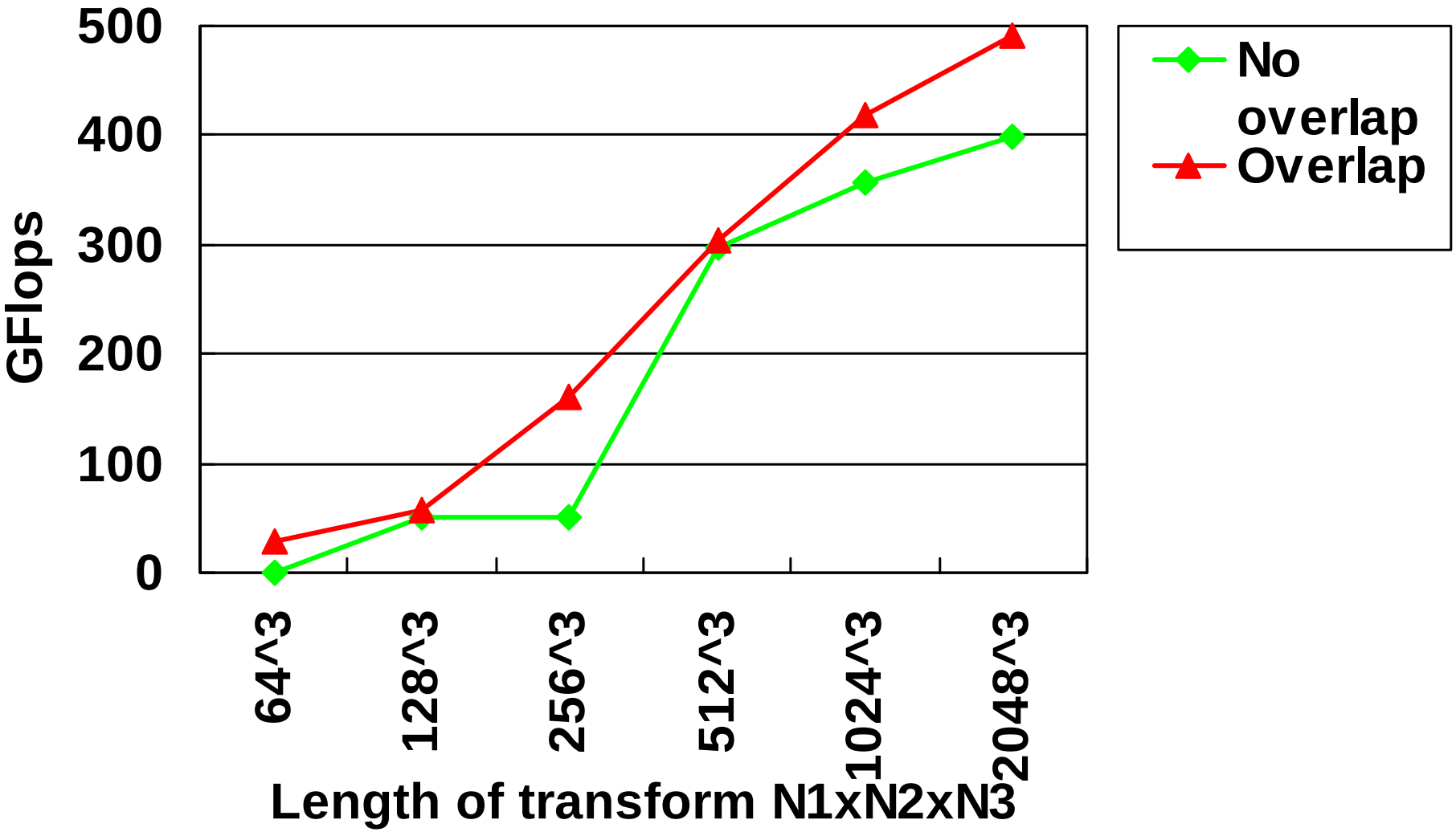


3. FFTs in z-direction





T2K-Tsukuba（64ノード，1024コア，256MPIプロセス）における並列三次元FFTの性能



FFTの限界性能(1/2)

- 以下のようなシステムを仮定し, $N = 1024^3$点FFTを二次元分割で計算
  - ノード数:P=1,048,576
  - ノード当たりの理論ピーク演算性能:「∞」Flops
  - ノード間通信性能:W=「∞」byte/s
  - 通信レイテンシ: L=1μsec
- N点FFTの演算量: $5N\log_2 N$
- 全対全通信にRingアルゴリズムを使った場合の通信時間は,

$$T_{comm} = 2(\sqrt{P}-1)\left(L + \frac{16N}{P\sqrt{P}\cdot W}\right) = 2.046\,\mathrm{msec}$$

- $N = 1024^3$点FFTの限界性能は,

$$(5N\log_2 N)/T_{comm} \approx 76.9\,\mathrm{TFlops}$$

FFTの限界性能(2/2)

- 全対全通信にshuffle-exchangeアルゴリズムを使った場合の通信時間は,

$$T_{comm} = 2\log_2\sqrt{P}\left(L + \frac{16N}{2\cdot P\cdot W}\right) \approx 20\,\mu\mathrm{sec}$$

- $N = 1024^3$点FFTの限界性能は,

$$(5N\log_2 N)/T_{comm} \approx 8.1\,\mathrm{PFlops}$$

- Strong Scalingにおいては, 近い将来FFTの性能が限界に達する可能性が高い.
- 全ノードを用いてFFTを計算するアプリケーションは, 今後アルゴリズムや解法を変更する必要がある.

まとめ

- ペタスケール計算環境におけるFFTライブラリ
  - 並列三次元FFTにおいて，二次元分割により通信時間を削減すると共に，演算と通信をオーバーラップさせることで，さらに性能を改善することができた.
  - 「京」でのチューニングも実施中.
- 2011年11月までにFFTEライブラリの最新版（Ver. 5.0）をhttp://www.ffte.jp/で公開する予定.
  - 並列一次元FFT（一次元分割，複素数）
  - 並列二次元，三次元FFT（一次元分割，実数＋複素数）
  - 並列三次元FFT（二次元分割，複素数）