

数値流体解析の並列効率とそのGPUによる 高速化の試み

清水建設(株) 技術研究所
PHAM VAN PHUC
(ファム バン フック)

流体計算時間短縮とGPUの活用の試み

■ 現CPUとの比較によりGPU活用の可能性

■ 現CPUの最大利用

- ノード内の最大計算資源の利用
 - すべてCPUコアの利用
 - 適切なアルゴリズムの利用



■ GPUの利用の試み

- 流体計算コードの一部のGPU化
 - 圧力方程式のGPU化
 - 有効な計算方法の抽出

CPUコア性能の何倍？



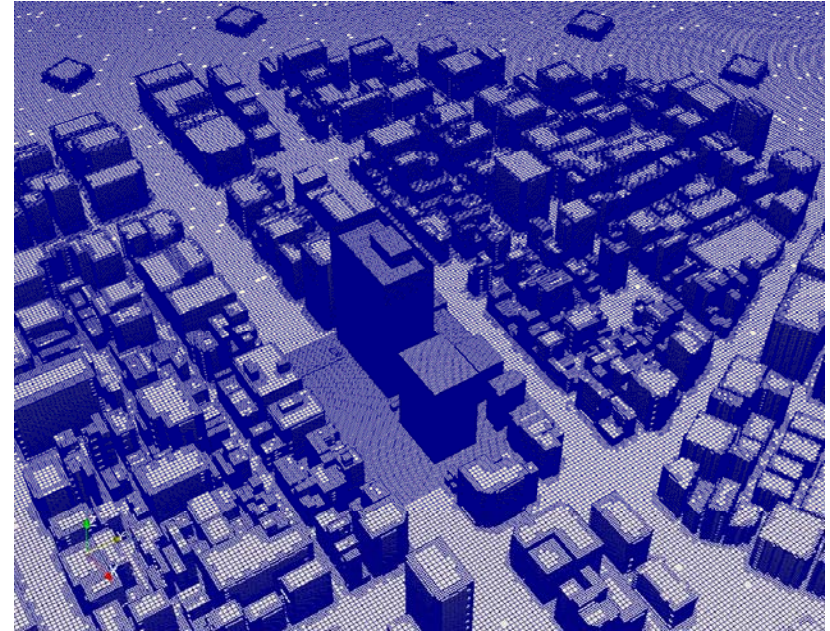
難しい度と性能？



数値流体計算

対象流体コードと流体問題

- 計算コード:
 - OpenFOAM 1.6.x
 - コンパイラ
 - Gcc 4.3.3, OpenMpi
 - Cuda2.3 (Tesla)、
 - Cuda3.2(Fermi)
 - 倍精度演算
- 実務流体問題: 都市モデル
 - 乱流モデル : RANS, LES
 - 格子種 : 非構造格
 - 並列計算法 : 領域分割法
 - 格子数: 200万と2,000万
 - 解析ソルバ:
 - GS法、CG法、AMG法...



計算格子の一例

CPUの最大利用

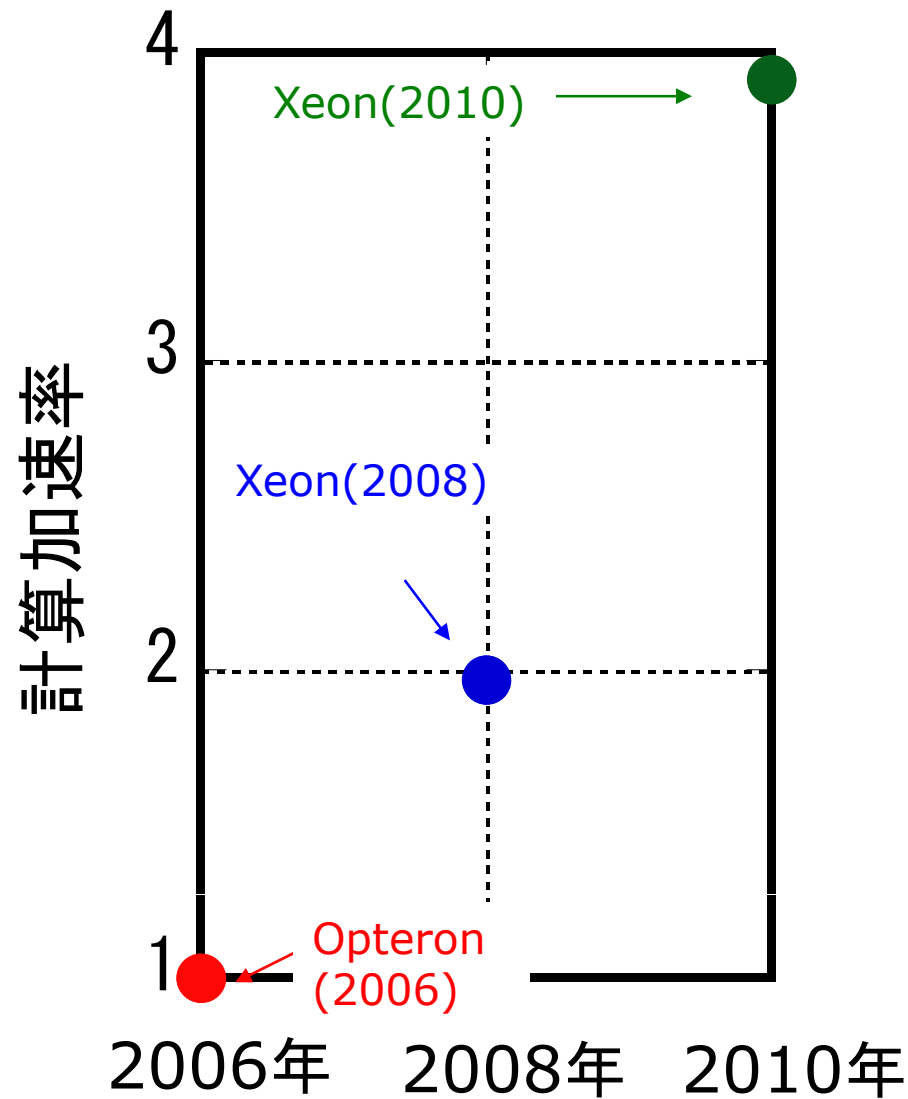


対象とする主なCPU計算機

| CPU計算機 | Opteron (2006) | Xeon (2008) | Xeon (2010) |
|----------|-------------------------------------|------------------------------------|------------------------------------|
| 年代 | 2006 | 2008 | 2010 |
| CPU | AMD Opteron Dual Core X 4CPUs | Intel Xeon Dual Core X 2CPUs | Intel Xeon Quad Core X 2CPUs |
| コア/ノード | 8 | 4 | 8 |
| CPU クロック | 2.6GHz | 3.4GHz | 3.33GHz |

- 一部性能評価や大規模並列計算とGPU開発の利用等に当たり
 - 東京工業大学の共同利用（産業利用）および先端研究施設共用促進事業『みんなのスパコン』TSUBAMEによるペタスケールへの飛翔トライアルユースによりTSUBAME 1.2を利用。

CPU計算機の計算性能

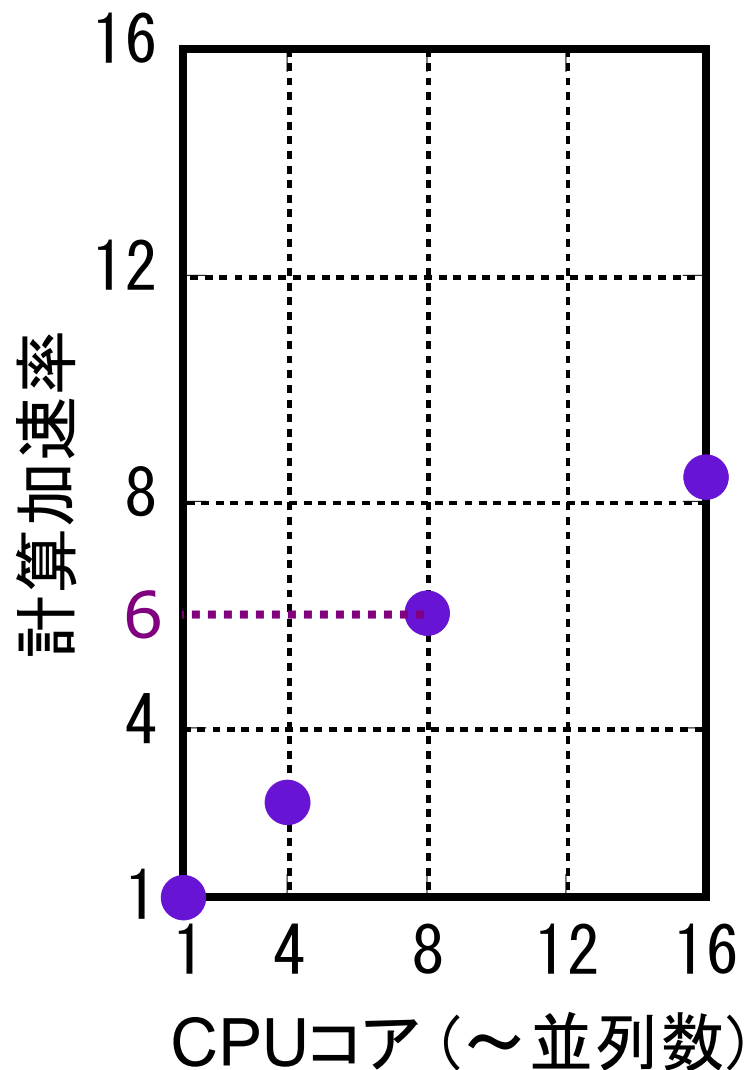


2000万格子の都市モデル
OpenFOAMの利用
1ノード内の同8CPUコアの評価

CPU計算機の利用:
2X/2years

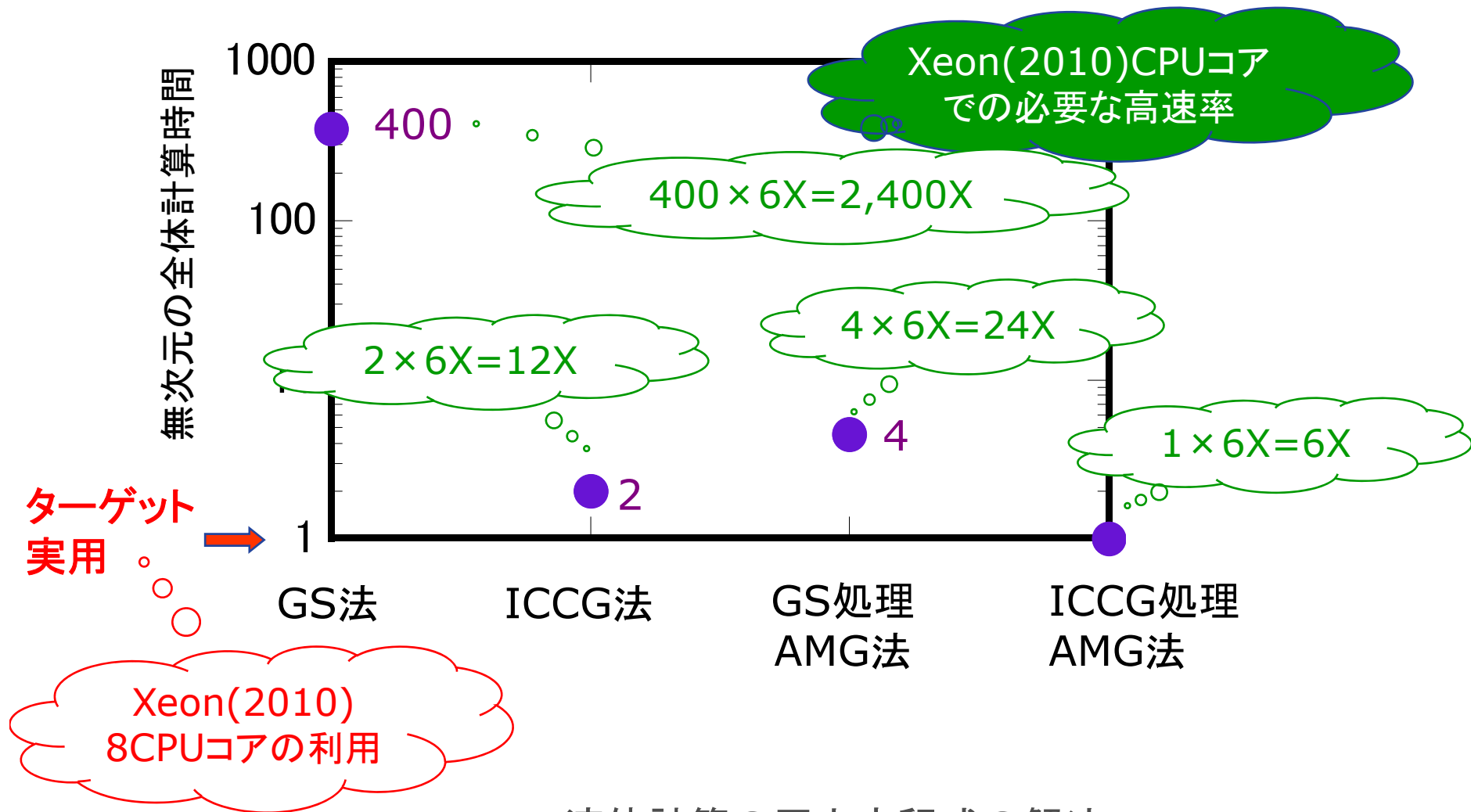
CPU計算機の計算並列効率

1ノード内のCPUコアの評価



- Xeonノード(2010)
- 8コア : 6X (CPUコア性能)

CPUでの流体計算の解析ソルバの性能



流体計算の圧力方程式の解法

- ・GS法 : Gauss-Seidel法
- ・ICCG法 : 不完全コレスキー分解共役勾配法
- ・AMG法 : 代数的マルチグリッド法 (Algebraic Multi-Grid)

まとめ: CPUの現実用

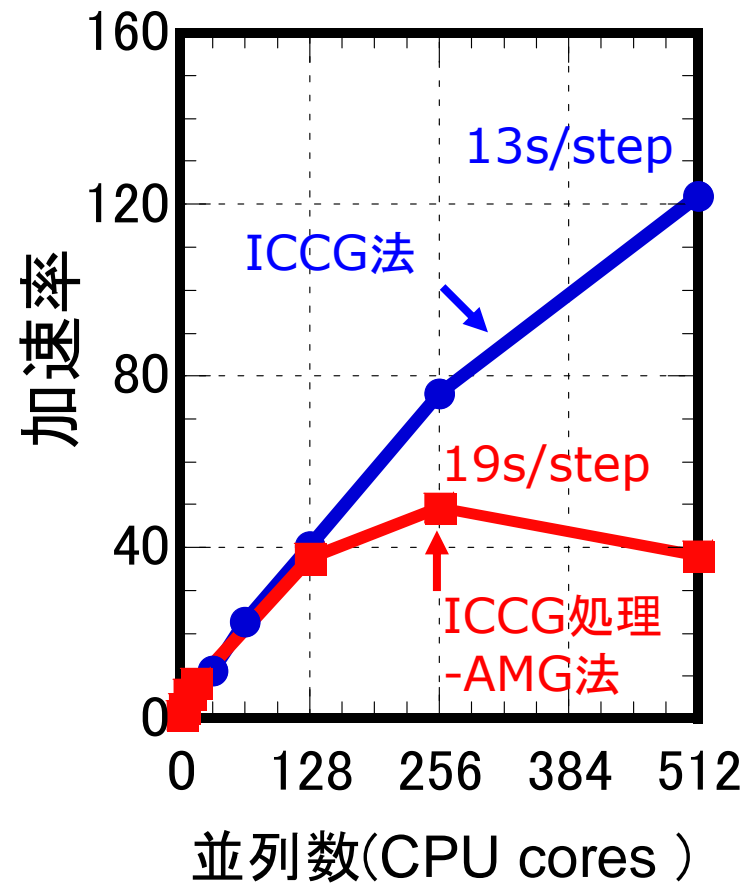
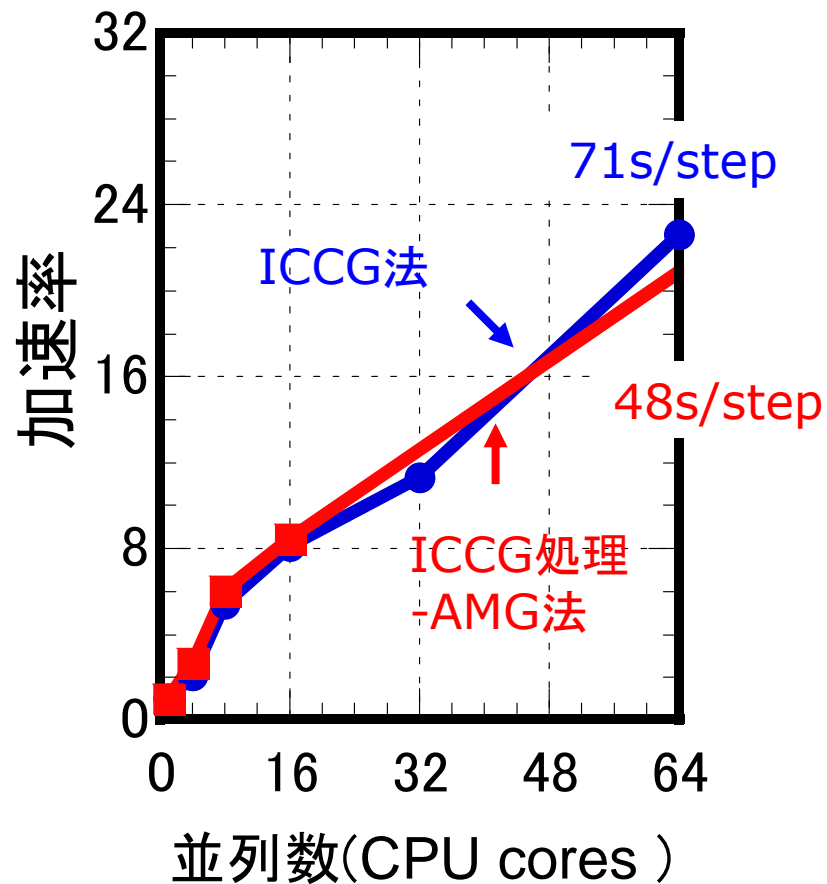
- Xeon(2010)の8コア・適切な解析ソルバの利用
(単CPUコアの性能)
 - GS法 : 2,400X
 - GS処理-AMG法 : 24X
 - ICCG法 : 12X
 - ICCG処理-AMG法: 6X

同アルゴリズムの解析ソルバで6Xの高速化が必要。

多数並列計算(GPU化?)での性能としては?

解析ソルバの並列計算の効率

TSUBAME 1.2計算ノード利用
LES乱流モデル






並列規模により解析ソルバの選定が重要

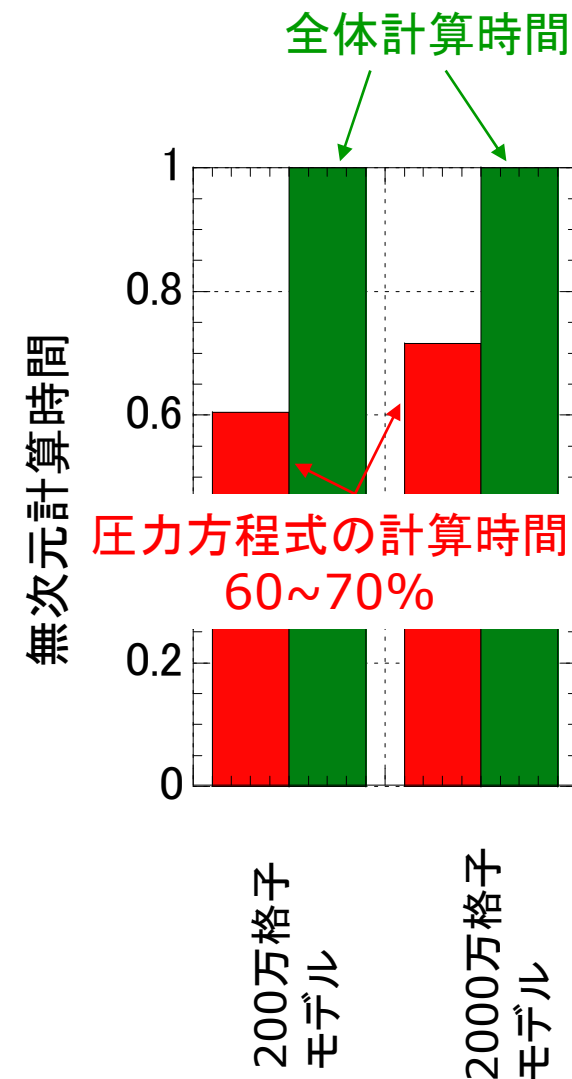
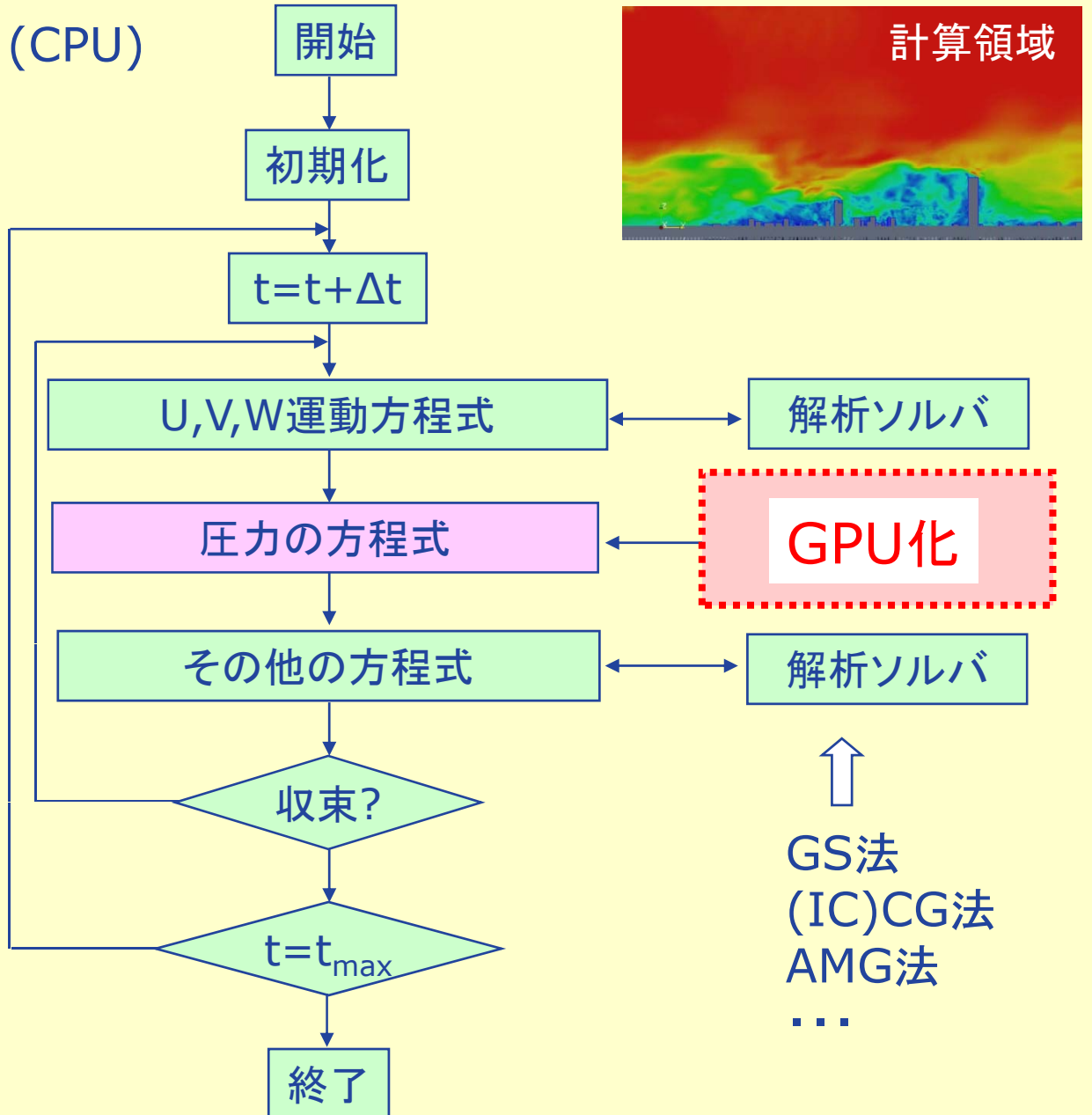
GPU活用の試み



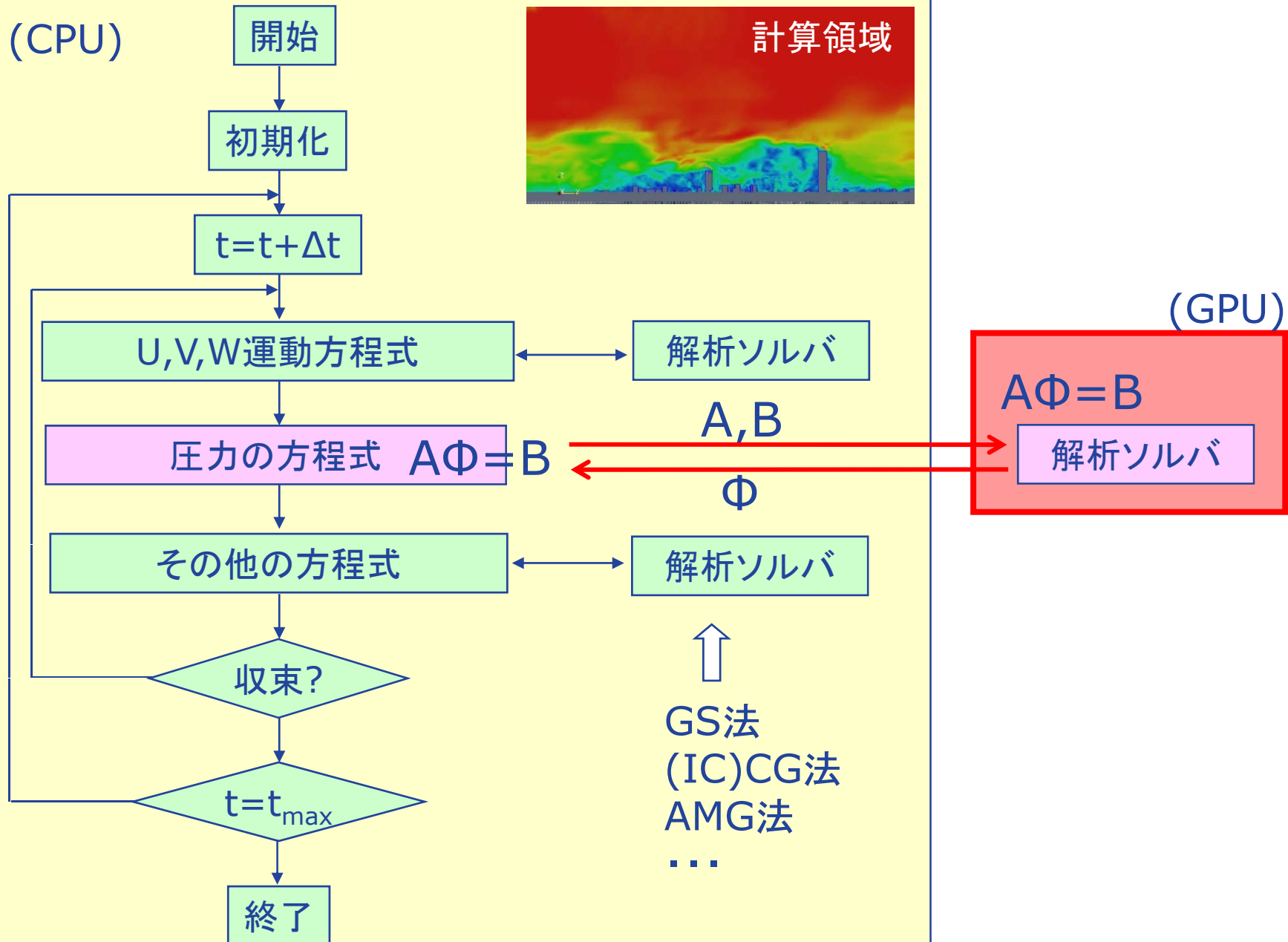
GPU計算機のテスト環境

| | | | |
|-----|--|---|--|
| CPU | Intel Xeon Quad Core × 2 CPUs (8cores) | Intel Xeon Quad Core × 2 CPUs (8cores) | Intel Xeon Quad Core × 2 CPUs (8cores) |
| GPU | — | C1060 × 4 GPU (Tesla) | C2070 × 2 GPU (Fermi) |
| 価格 | ~100万円 | ~100万円 +60万円 2X | ~100万円 +80万円 2X |
| |  |  |  |

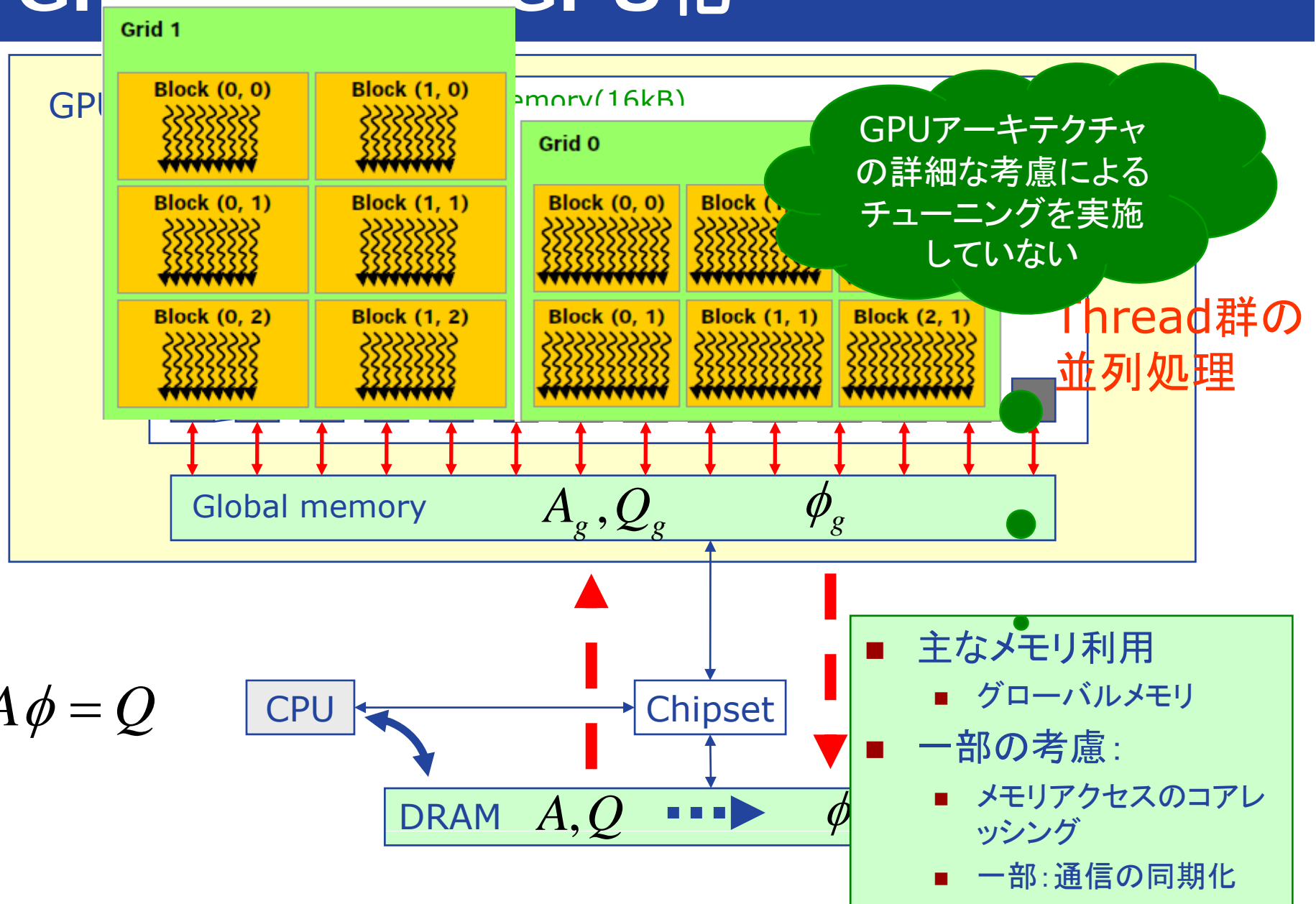
OpenFOAM流体計算とそのGPU化



OpenFOAM流体計算とそのGPU化

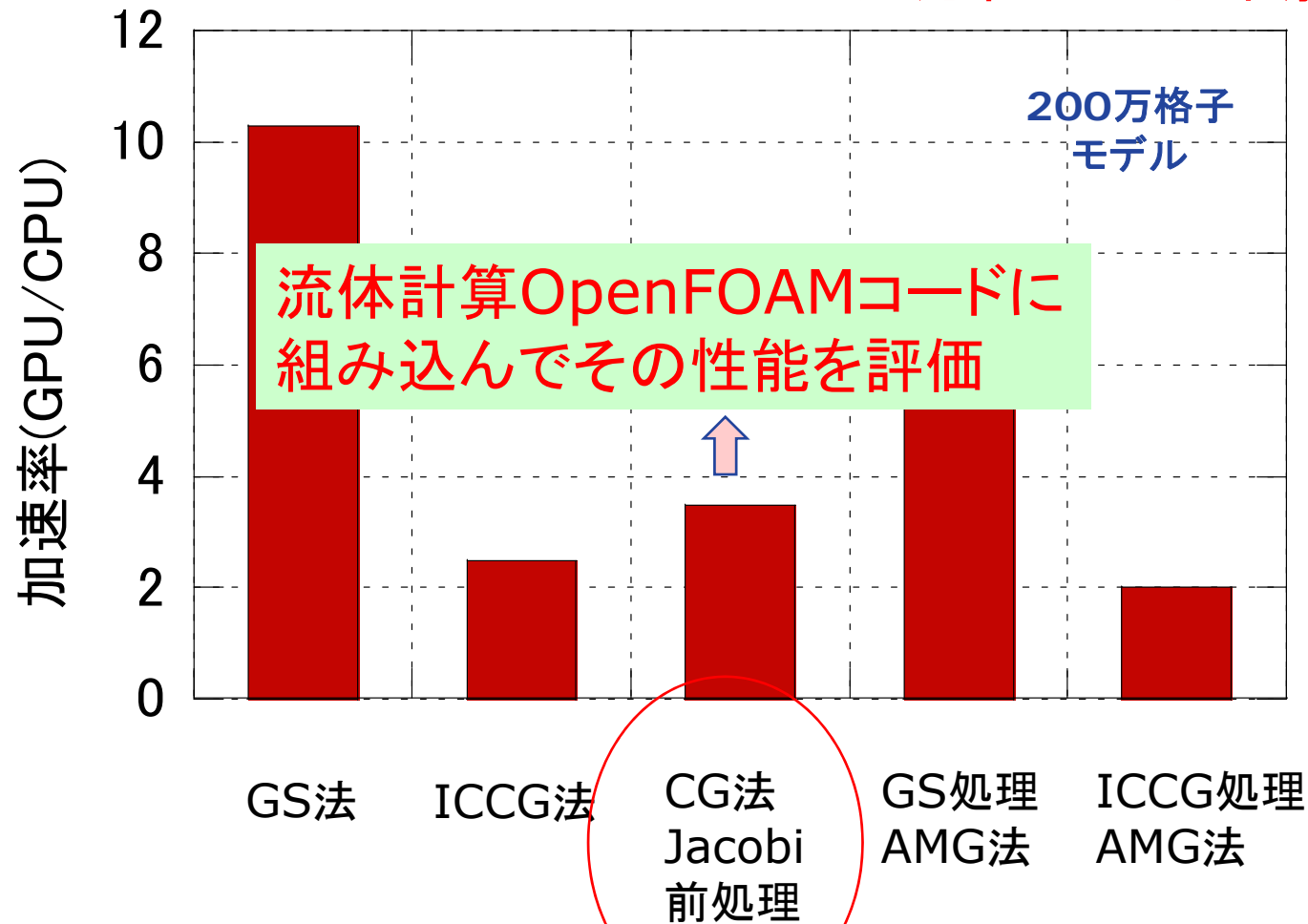


GPU構成とそのGPU化



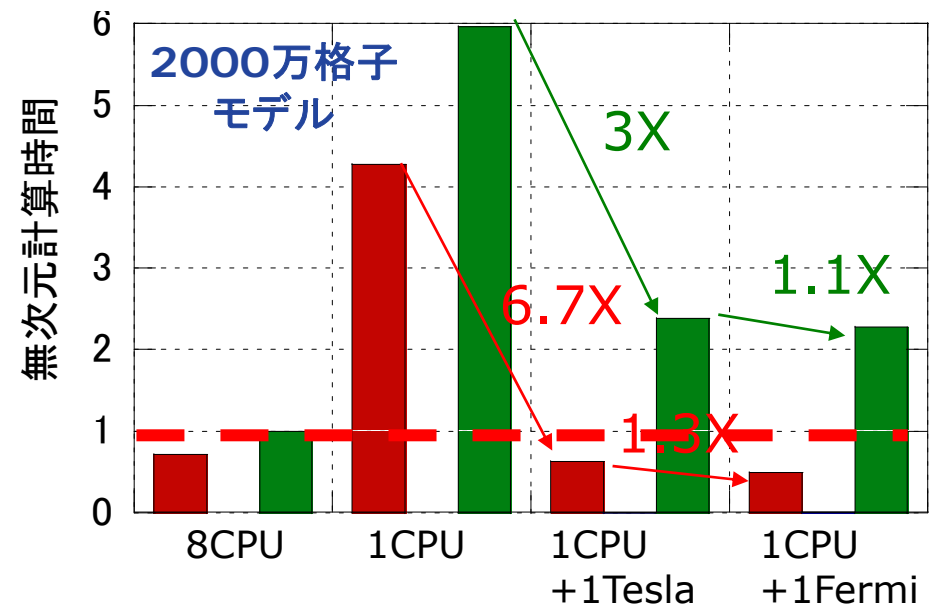
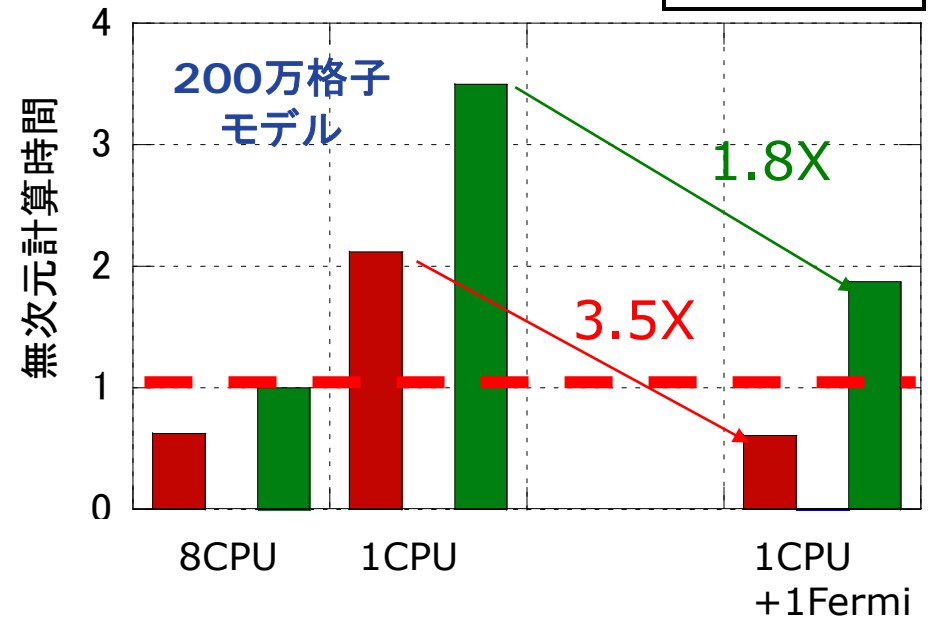
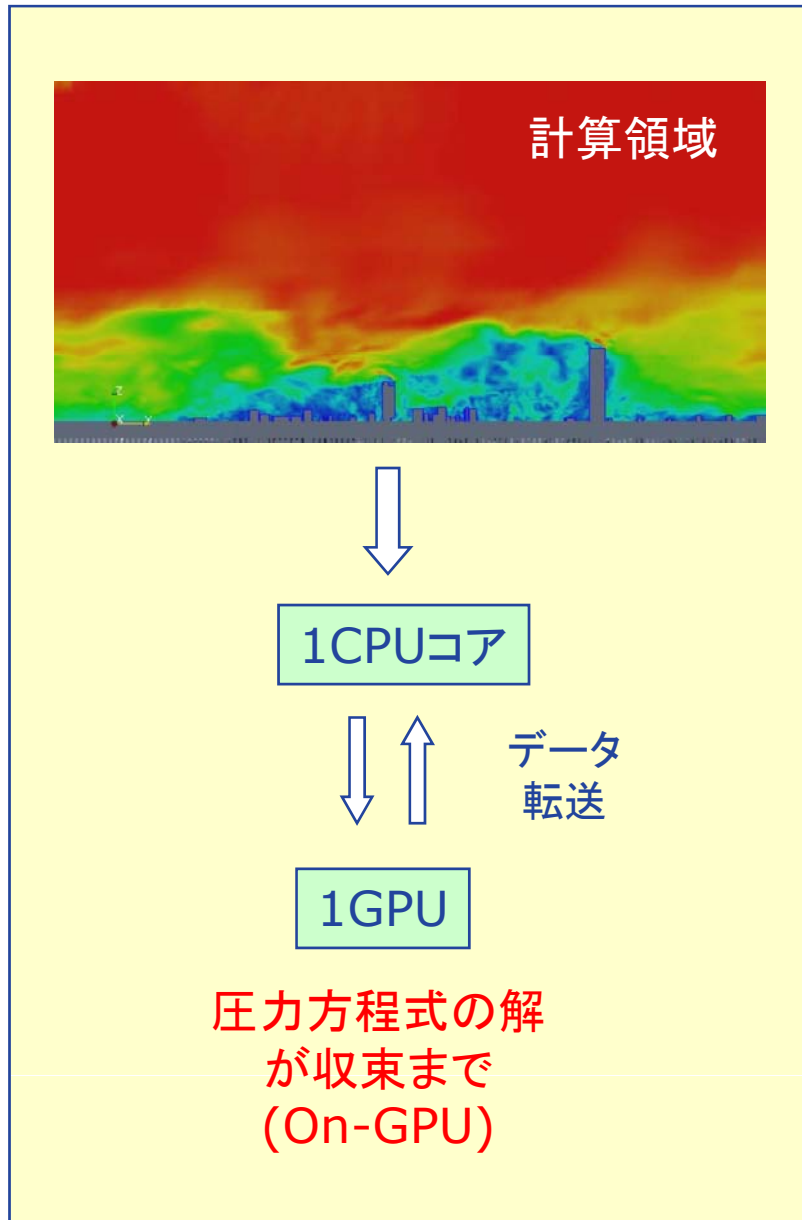
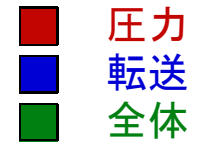
GPUによる解析ソルバの加速率

圧力方程式の収束解析



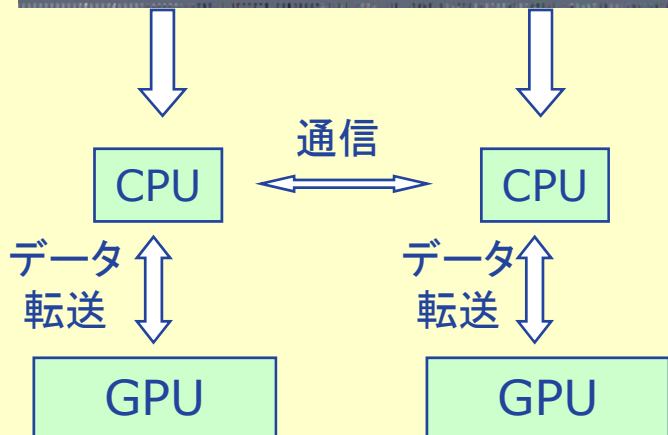
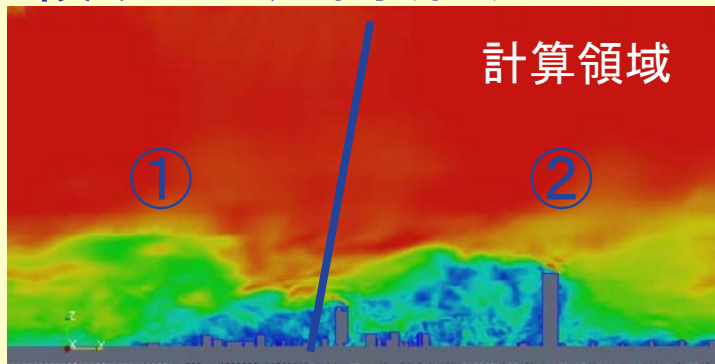
速い解析ソルバは加速率が小さい

流体計算：単CPU&GPUの処理法

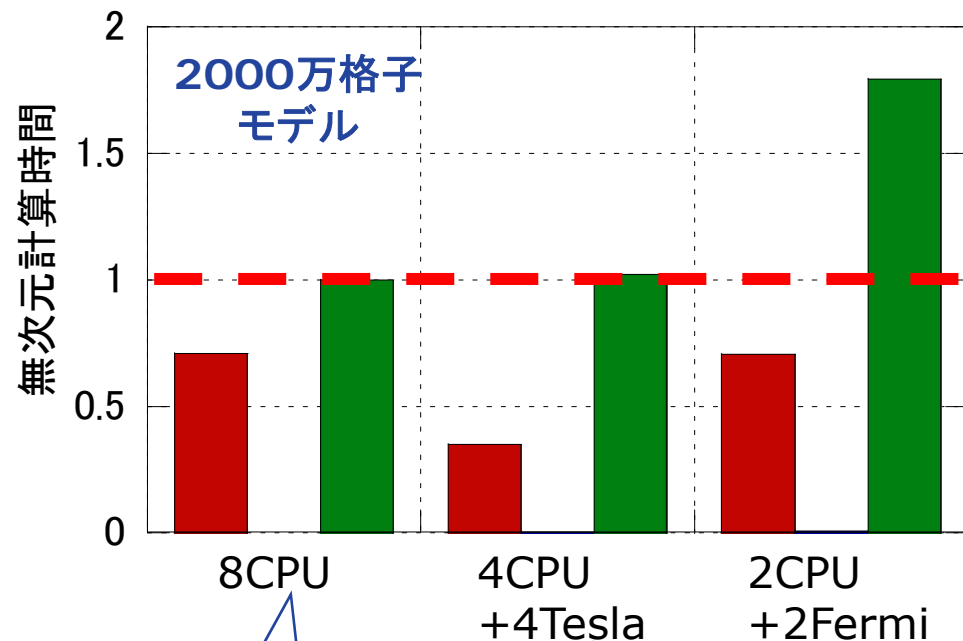
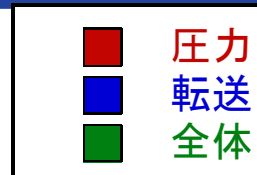


流体計算：複数CPU&GPUの処理法

領域の2分割(例)



圧力方程式の毎反復計算
データ通信の実施



計算領域の
8分割

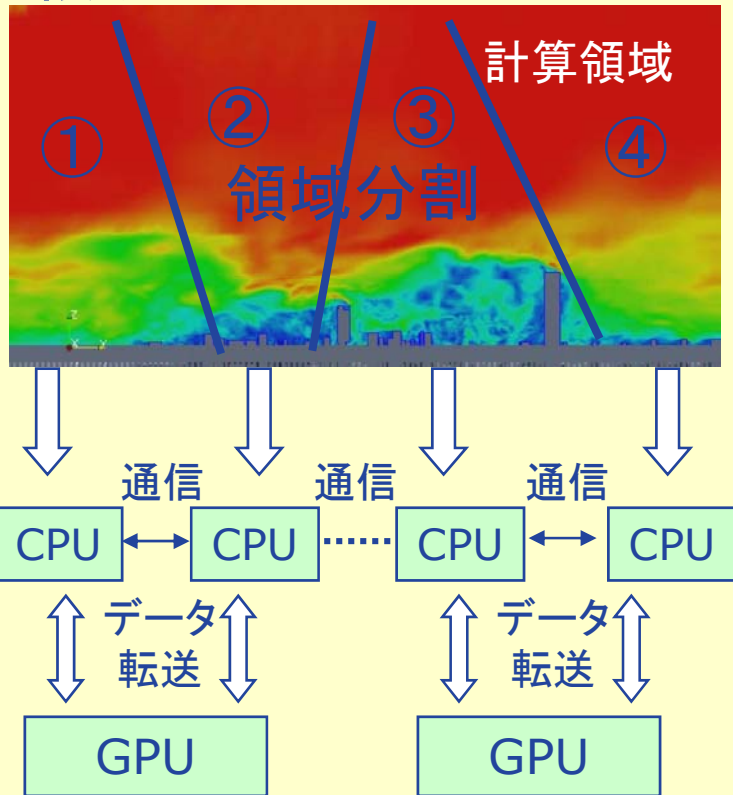
計算領域の
4分割

計算領域の
.2分割

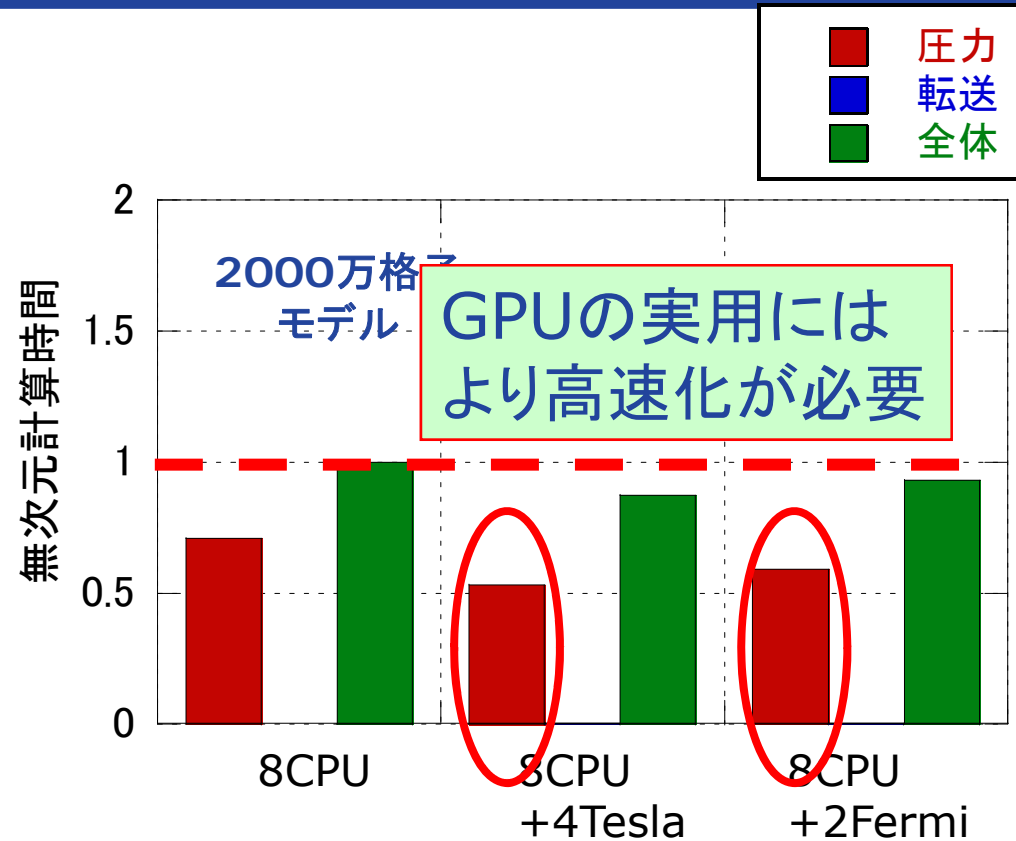
計算領域が大きく、
計算負荷が大。

流体計算：複数CPU&複数GPUの処理法

領域の4分割(例)



圧力方程式の毎反復計算
データ通信の実施



高速化にはCPU/GPUの
最大限の利用は重要

まとめ

■ CPUの利用

- 2006年~2010年:2X/2年
- 1ノード(8コア,8並列の実行)は単CPUコアに対して6X高速可能
- 計算高速化には適切な解析ソルバの利用が重要(数十倍~数百倍)
- 遅い解析ソルバは並列効率が高い傾向と見られる

■ GPUの利用の試み

- 計算規模により加速率(単CPUコアの性能)は変化
 - 200万格子規模 :圧力(3.5X),全体(1.8X)
 - 2000万格子規模:圧力(8.7X),全体(3.3X)
- GPU化にはノード内のCPUの最大限の利用は重要。
- GPUの実用化にはGPUのアーキテクチャの考慮やチューニングによる高速化が必要。

ご清聴ありがとうございました。

■ 謝辞

- 本検討の一部の成果は、東京工業大学の共同利用（産業利用）および先端研究施設共用促進事業『みんなのスパコン』TSUBAMEによるペタスケールへの飛翔 トライアルユースによりTSUBAME 1.2を利用した。ここに、記して謝意を表す。