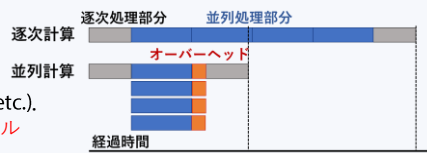


背景

コンピュータの性能向上のトレンド

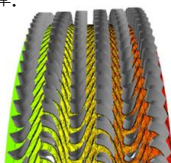
- ・コアあたりの演算性能の向上に限界に...
- ・コア数増加による計算処理能力の向上.
- ・クロックサイクル当たりの命令処理数を向上 (ベクトル演算 etc.).
- 演算性能を発揮するにはマルチコアによる並列計算とベクトル演算の活用が必須!



数値流体力学 (CFD: Computational Fluid Dynamics)

- ・工学分野から医療分野, 生物分野, 宇宙・気象など様々な流れの解析に利用.
- ・物理法則に基づく数学モデルを数値的に解き流体の挙動をシミュレーション.
- ・流れ場を格子点空間で表現, 格子点上の物理量を計算.

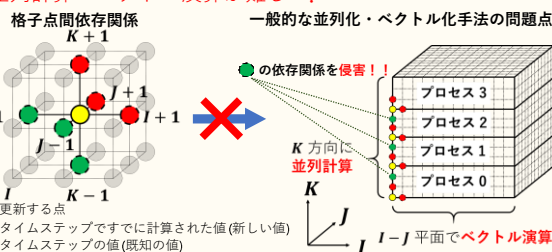
- CFD
- 陽解法 - ルンゲ・クッタ法
 - 解法・プログラムがシンプル
 - × 悪い収束性
 - 問題によっては発散
 - 収束まで多くの反復計算
 - 陰解法 - LU-SGS法
 - × 解法・プログラムが複雑
 - 良い収束性
 - 安定的に収束
 - 少ない反復計算で収束



蒸気タービン内の液滴質量分率分布のシミュレーション結果
(<http://www.caero.mech.tohoku.ac.jp/research/NumericalTurbine/NumericalTurbine.html>)

LU-SGSのデータ依存関係

- ・隣接する格子点間でデータ依存関係が存在し計算順序に制限.
- 並列計算・ベクトル演算が難しい.



目的と提案手法

研究の目的

- ・並列計算・ベクトル演算を活用したLU-SGSの高速化.
- ・近年のプロセッサの傾向から, 多数のコアとベクトル演算器を同時に活用することが重要!

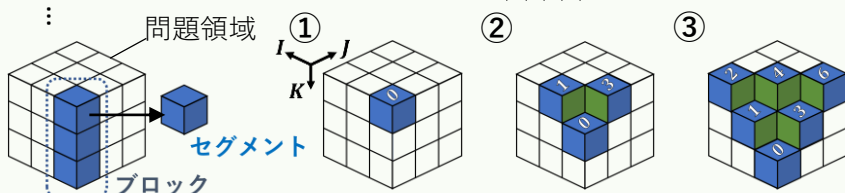
提案手法: 外側波面と内側波面を組み合わせた階層的な波面法

- ・粗粒度と細粒度の波面法を階層的に適用.
- ・波面: 同時に計算される要素 (格子点やセグメント) が形成する面.
- ・外側波面
 - ・内側波面
 - ・マルチコアを用いた並列計算.
 - ・単一命令で複数の格子点要素をベクトル演算.
- LU-SGSのデータ依存関係を侵害せず並列計算とベクトル演算を同時に適用可能.

外側波面を用いた並列化

- ・問題領域は $I-J$ 平面でブロック単位に分割.
- ・それぞれのコアは1つのブロックの計算を担当.
- ・さらに各ブロックは K 方向にセグメント単位に分割.
- ・並列計算はセグメント単位で実行.

- ① コア0が始端のセグメントを計算.
- ② コア1と3がコア0が計算したセグメントから境界データを受信し計算を開始.
コア0は同じブロックの2つ目のセグメントを計算.
- ③ ②で計算したセグメントの境界データを受信しコア0, 1, 2, 3, 4, 6が並列計算を開始.

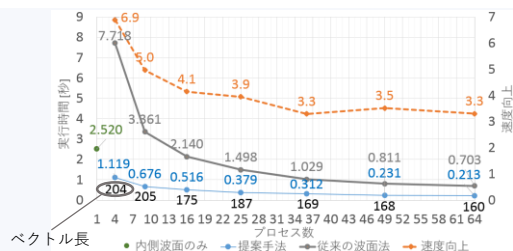


内側波面を用いたベクトル化

- ・内側波面:
 - 超平面とも呼ぶ. 格子点の I, J, K 軸座標の和が等しい格子点グループ.
- ・同じ波面内の格子点をベクトル演算.
- ・スウィープ方向で隣接格子点間の依存関係を回避.
 - ① セグメントの始点である青点を計算.
 - ② 青点の値を用いて水色点を計算.
 - ③ 水色点の値を用いて緑点を計算.



評価



評価条件

- ・プロセッサ: SX-Aurora TSUBASA Vector Engine Type 10B
- ・実行プログラム: 数値タービンLU-SGSカーネル
- ・問題サイズ: 300 × 300 × 400

- ・提案手法は従来の波面法(外側波面のための手法)に比べて3.3-6.9倍高速化.
- ・ベクトル化のための内側波面の手法と比較しても高速.
- ・階層的な波面法により並列化とベクトル化を活用.
- ・ベクトル長分の高速化が得られていない.
- ・内側波面の参照パターンによるメモリアクセス時間の増加, 実装のオーバーヘッド.
- ・並列数増加につれて並列実行効率が低下.
- ・セグメントサイズの縮小によりベクトル長が短縮.

研究計画

目標: アーキテクチャを考慮したCFDの高速化

1. 最適なメモリアクセスによるさらなる高速化.
 - ・参照パターンに適したデータレイアウトを適用
 - ・ボトルネックであるメモリアクセス時間を削減.
2. 速度向上率が得られるセグメント分割手法の確立.
 - ・ブロックを多くのセグメントに分割.
 - 並列効率↑ ベクトル長↓
 - ・セグメントサイズが大きくなるように分割.
 - 並列効率↓ ベクトル長↑
3. 本手法の利点をより活用できる問題の調査.
 - ・より大規模な問題に適用.
 - ・ウィークスケーリングの問題に適用.
 - 並列効率とベクトル長のトレードオフが緩和され, スケーラビリティの向上に期待.