複数GPUによる大規模LES計算

小野寺　直幸[*1]

青木　尊之[*1]，小林　宏充[*2]

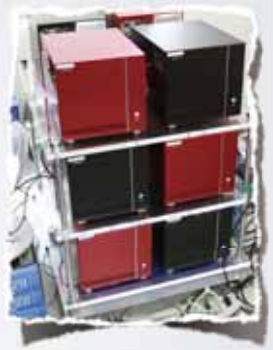
*1 東京工業大学

*2 慶應義塾大学

CompView　AOKI Lab.

発表概要

- 研究背景と研究目的
- 乱流モデル
- 平行平板乱流計算
- 複数GPUによる高速計算
- GPUを用いた実用計算
- まとめ



研究背景と研究目的

研究背景

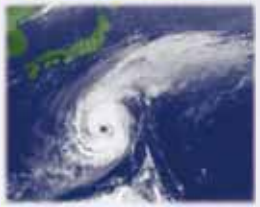
乱流の数値計算は工学的に非常に重要であるが、直接計算を行うのは非常に困難であり、LESにおいても多くの計算を必要とする。

＊Navier-Stokes方程式

$$\frac{\partial u_i}{\partial t} + \boxed{\frac{\partial u_i u_j}{\partial x_j}} = \frac{1}{Re}\frac{\partial^2 u_i}{\partial x_j^2} - \frac{\partial p}{\partial x_i}$$

→ 非線形項により小スケールの渦が生成

乱流の渦のスケールのイメージ



large scale ⇔ small scale

100km~ ⇔ ~1m

＊渦のスケール

エネルギー保有領域　l

エネルギー散逸領域　η

$$\boxed{\frac{l}{\eta} = Re_l^{\frac{3}{4}}}$$

＊必要な格子点数　$n \propto Re_l^{\frac{9}{4}}$

＊時間スケール　$t \propto Re_l^{\frac{1}{2}}$

＊ 情報爆発 !!!

→ 実際の現象に対して、乱流の直接計算(DNS)を行うことは不可能

研究背景と研究目的

乱流のモデル化

＊乱流モデルを適用したNavier-Stokes方程式

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial \bar{u}_i \bar{u}_j}{\partial x_j} = -\frac{\partial \bar{p}}{\partial x_i} + \frac{1}{Re}\frac{\partial^2 \bar{u}_i}{\partial x_j^2} - \boxed{\frac{\partial \tau_{ij}}{\partial x_j}}$$

← モデル化

小スケール

$$\boxed{\begin{aligned}\tau_{ij} &= \overline{u_i u_j} - \bar{u}_i \bar{u}_j \\ &= -2\nu_{SGS} S_{ij}\end{aligned}}$$

$$S_{ij} = \frac{1}{2}\left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j}\right)$$

→ 分子粘性項と同様に定式化

小スケールの渦をモデル化することでDNSに比べて計算負荷を削減

⇩

現実問題で取り扱う様な高レイノルズ数の流れを扱うことが可能

研究目的

GPUを用いたLarge eddy simulation (LES)コードの開発を行う

乱流モデルとLESコードの実装

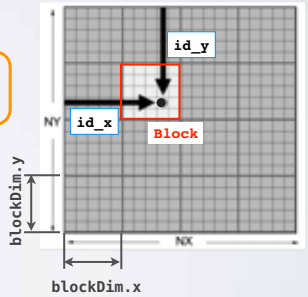
$$\bar{u}_i^* = \bar{u}_i^n + \left(-\frac{\partial \bar{u}_i^n \bar{u}_j^n}{\partial x_j} + \frac{1}{Re}\frac{\partial^2 \bar{u}_i^n}{\partial x_j^2} - \frac{\partial \tau_{ij}}{\partial x_j}\right)\Delta t$$

$$\tau_{ij} = -2\nu_{SGS} S_{ij}$$

$$\nu_{SGS} = C\Delta^2 |S|$$

$$\bar{u}_i^{n+1} = \bar{u}_i^* - \frac{\partial \bar{p}^{n+1}}{\partial x_i}\Delta t$$

$$\frac{\partial^2 \bar{p}^{n+1}}{\partial x_i^2} = \frac{\partial \bar{u}_i^*}{\partial x_i}$$



LESのCUDA実装

支配方程式の半離散化式

微分演算子

$$\bar{u}_i^* = \bar{u}_i^n + \left(-\frac{\partial \bar{u}_i^n \bar{u}_j^n}{\partial x_j} + \frac{1}{Re}\frac{\partial^2 \bar{u}_i^n}{\partial x_j^2} - \frac{\partial \tau_{ij}}{\partial x_j}\right)\Delta t \longrightarrow u_x,\ u_{xx},\ \tau_x$$

$$\bar{u}_i^{n+1} = \bar{u}_i^* - \frac{\partial \bar{p}^{n+1}}{\partial x_i}\Delta t \qquad \frac{\partial^2 \bar{p}^{n+1}}{\partial x_i^2} = \frac{\partial \bar{u}_i^*}{\partial x_i} \longrightarrow p_x,\ p_{xx}$$

＊ 参照ステンシル

u τ ：速度・応力テンソル　$O(\Delta x^4)$

⇒５点参照

u_{i-2} u_{i-1} u_i u_{i+1} u_{i+2}

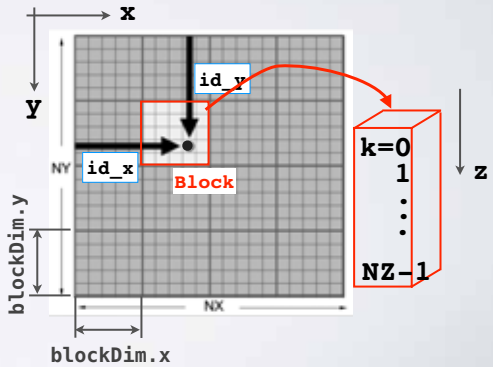
τ_{i-2} τ_{i-1} τ_i τ_{i+1} τ_{i+2}

p ：圧力　$O(\Delta x^2)$

⇒３点参照

p_{i-1} p_i p_{i+1}

＊ CUDAへのBLOCKの割当



乱流モデル

RANS (レイノルズ平均モデル)

速度場を平均部分と変動部分に分解する。

$U_i = \bar{U}_i + U_i'$ モデル化

層流について解くため、計算負荷が少ない。モデルの影響が大きい。

LES (ラージエディ・シミュレーション)

速度場を格子上の成分と格子以下の成分に分ける。

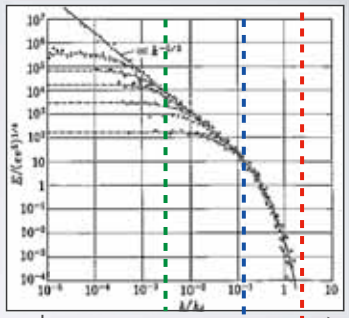
$u_i = \bar{u}_i + u_i'$ モデル化

非定常流れについて計算を行うため高精度な解析が可能となり、DNSと比べて計算負荷が少ない。

DNS (直接計算)

すべての変動部分に対し直接計算を行う。非常に多くの格子を必要であり適用できる問題が限られる。

エネルギースペクトル



RANS

LES

DNS

大スケール

小スケール

乱流モデル

RANS (レイノルズ平均モデル)

速度場を平均部分と変動部分に分解する。

$U_i = \bar{U}_i + U_i'$ モデル化

層流について解くため、計算負荷が少ない。モデルの影響が大きい。

LES (ラージエディ・シミュレーション)

速度場を格子上の成分と格子以下の成分に分ける。

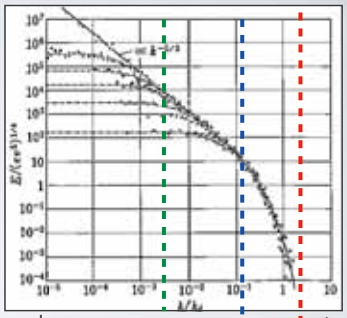
$u_i = \bar{u}_i + u_i'$ モデル化

非定常流れについて計算を行うため高精度な解析が可能となり、DNSと比べて計算負荷が少ない。

DNS (直接計算)

すべての変動部分に対し直接計算を行う。非常に多くの格子を必要であり適用できる問題が限られる。

エネルギースペクトル



RANS

LES

DNS

大スケール

小スケール

Large Eddy Simulation -SGS モデル-

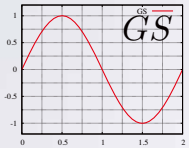
フィルター化された Navier-Stokes 方程式

$$u = \bar{u} + u'$$

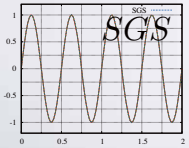
エネルギー散逸

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial \bar{u}_i \bar{u}_j}{\partial x_j} = -\frac{\partial \bar{p}}{\partial x_i} + \frac{1}{Re}\frac{\partial^2 \bar{u}_i}{\partial x_j^2} - \boxed{\frac{\partial \tau_{ij}}{\partial x_j}}$$

$$\boxed{\tau_{ij} = -2\nu_{SGS} S_{ij}}$$

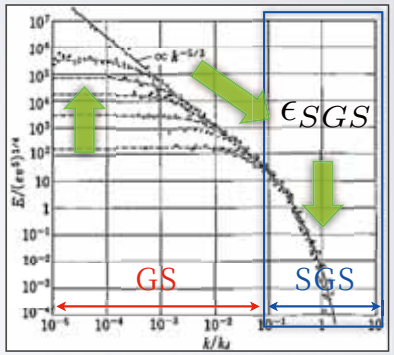


格子上の成分
↓
Navier-Stokes方程式



格子解像度以下の成分
↓
SGSモデルにより表現

エネルギースペクトル



Large Eddy Simulation -SGS モデル-

Smagorinsky モデル

$$\tau_{ij} = -2\nu_{SGS} S_{ij}$$

$$\nu_{SGS} = C\Delta^2|S| \qquad \boxed{C : const}$$

○ 安定
△ 壁境界
△ モデル係数が定数なため、経験的な知識が必要

Dynamic Smagorinsky モデル

$$\nu_{SGS} = C\Delta^2|S|$$

$$\boxed{C = \frac{< L_{ij}L_{ij} >}{< M_{ij}M_{ij} >}} \qquad L_{ij} = \widehat{\bar{u}_i\bar{u}_j} - \hat{\bar{u}}_i\hat{\bar{u}}_j$$

$$M_{ij} = 2\hat{\Delta}^2|\hat{\bar{S}}|\hat{\bar{S}}_{ij} - 2\hat{\Delta}^2|\hat{\bar{S}}|\hat{\bar{S}}_{ij}$$

○ 壁境界
△ 計算が煩雑
△ 全計算領域での平均化が必要
→並列計算に向いていない

Coherent-structure モデル

*H.Kobayashi, Phys. Fluids.17, (2005).

$$\nu_{SGS} = C\Delta^2|S|$$

→モデル係数は、速度勾配テンソルの第２不変量より求める

$$\boxed{C = C_1|F_{CS}|^{3/2}}$$

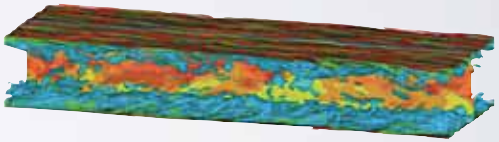
$$\boxed{F_{CS} = \frac{Q}{E}} \qquad (-1 < F_{CS} < 1)$$

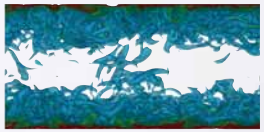
$$Q = -\frac{1}{2}\frac{\partial \bar{u}_j}{\partial x_i}\frac{\partial \bar{u}_i}{\partial x_j}$$

$$E = -\frac{1}{2}\left(\frac{\partial \bar{u}_j}{\partial x_i}\right)^2$$

○ 壁境界
○ モデル係数の計算が局所的
→GPUなどの並列計算に向いている

平行平板乱流計算　統計量



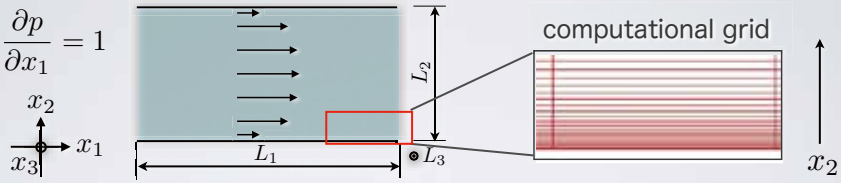


平行平板乱流

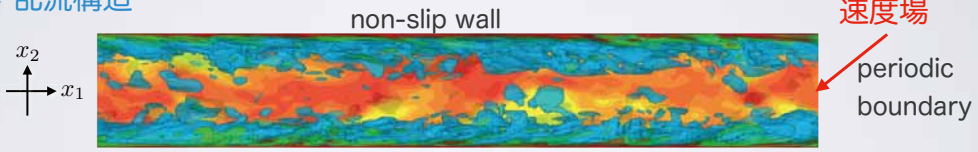
計算条件

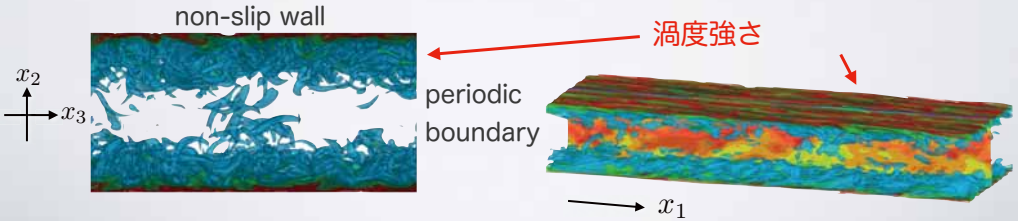
$Re_\tau = 180$　　$(L_1, L_2, L_3) = (4\pi\delta, 2\delta, 1.3\pi\delta)$　　$x_i^+ = Re_\tau \Delta x_i$

$$\frac{\partial p}{\partial x_1} = 1$$



乱流構造



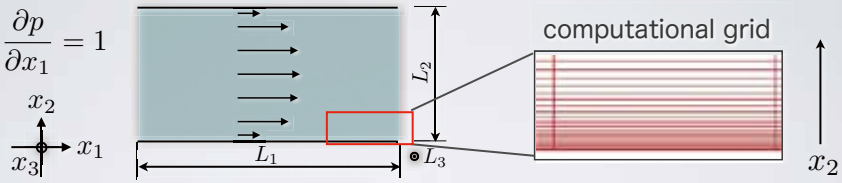


チャネル乱流計算

計算条件

$Re_\tau = 180$ $(L_1, L_2, L_3) = (4\pi\delta, 2\delta, 1.3\pi\delta)$ $x_i^+ = Re_\tau \Delta x_i$

$$\frac{\partial p}{\partial x_1} = 1$$

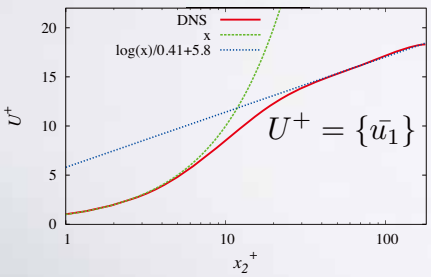


乱流統計量

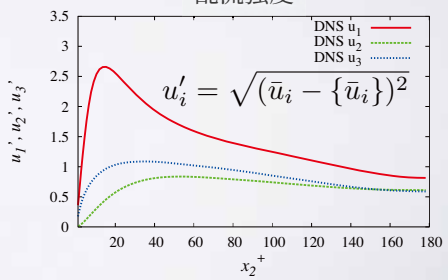
* Direct numerical simulation (DNS). R.Moser 1999

DNSとの統計量の比較を行う

平均流速



乱流強度

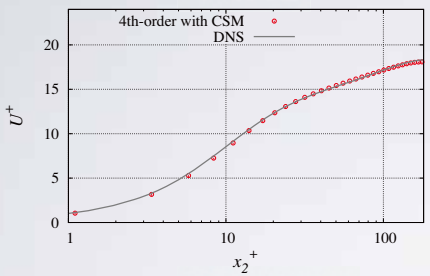


乱流統計量

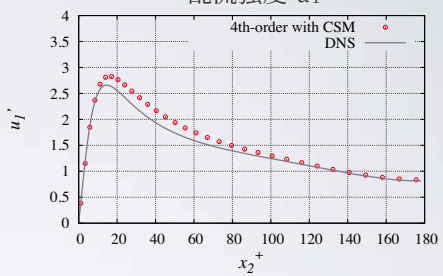
4次精度中心差分 with SGSモデル

$Re_\tau = 180$ $(N_1, N_2, N_3) = (64, 64, 64)$ $(x_1^+,\ x_2^+,\ x_3^+) = (35,\ 2.1 \sim,\ 11.4)$

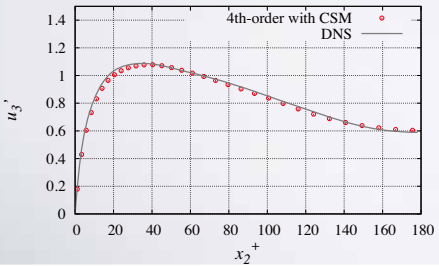
平均流速



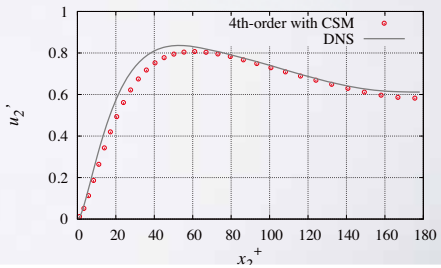
乱流強度 u1'



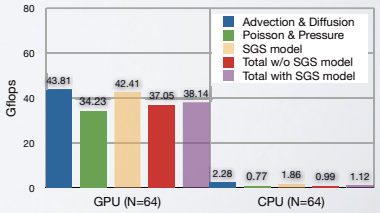
乱流強度 u2'



乱流強度 u3'



チャネル乱流計算　実行性能



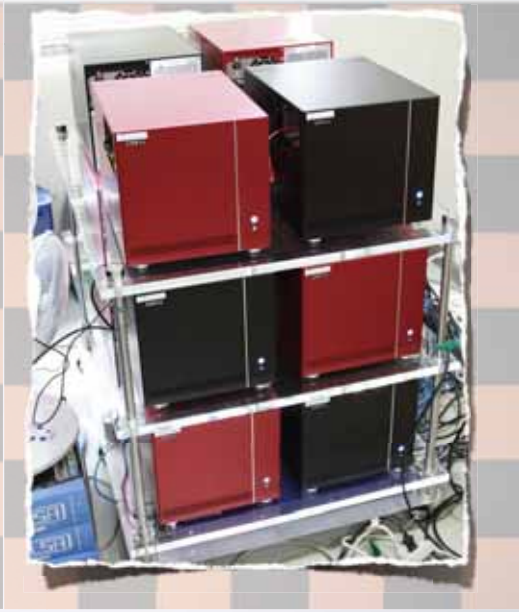
計算機環境

Red & Black クラスター @aoki_lab

CPU : Core i7
GPU : Tesla C2050 (3GB)
CUDA : 3.1
OS : CentOS 5.3
Mem : 6GB　　4 node

CPU : Core i7
GPU : GeForce GTX285 (2GB)
CUDA : 3.1
OS : CentOS 5.2
Mem : 6GB　　8 node

QDR Infiniband



GPUとCPUの演算性能

浮動小数点演算数

$$\bar{u}_i^* = \bar{u}_i^n + \left(-\frac{\partial \bar{u}_i^n \bar{u}_j^n}{\partial x_j} + \frac{1}{Re}\frac{\partial^2 \bar{u}_i^n}{\partial x_j^2} - \frac{\partial \tau_{ij}}{\partial x_j} \right) \Delta t$$

$$\tau_{ij} = -2\nu_{SGS} S_{ij}$$

$$\nu_{SGS} = C\Delta^2 |S|$$

$$\bar{u}_i^{n+1} = \bar{u}_i^* - \frac{\partial \bar{p}^{n+1}}{\partial x_i}\Delta t$$

$$\frac{\partial^2 \bar{p}^{n+1}}{\partial x_i^2} = \frac{\partial \bar{u}_i^*}{\partial x_i}$$

Flops / step

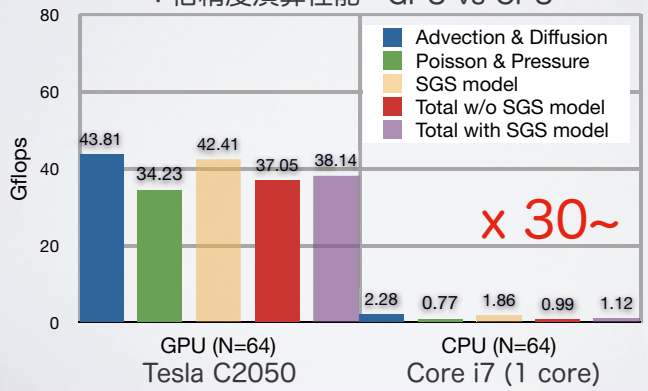
→Advection & Diffusion
870 Flop

→SGS model
875 Flop

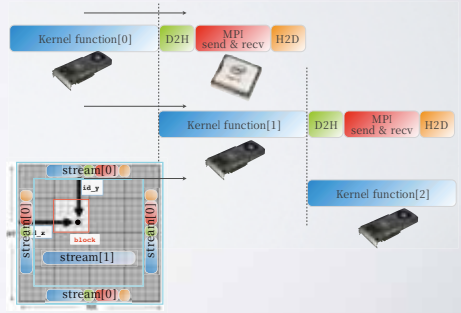
→Poisson & Pressure
1726 Flop (SOR 50 iteration)

LESコードの演算性能

＊倍精度演算性能　GPU vs CPU

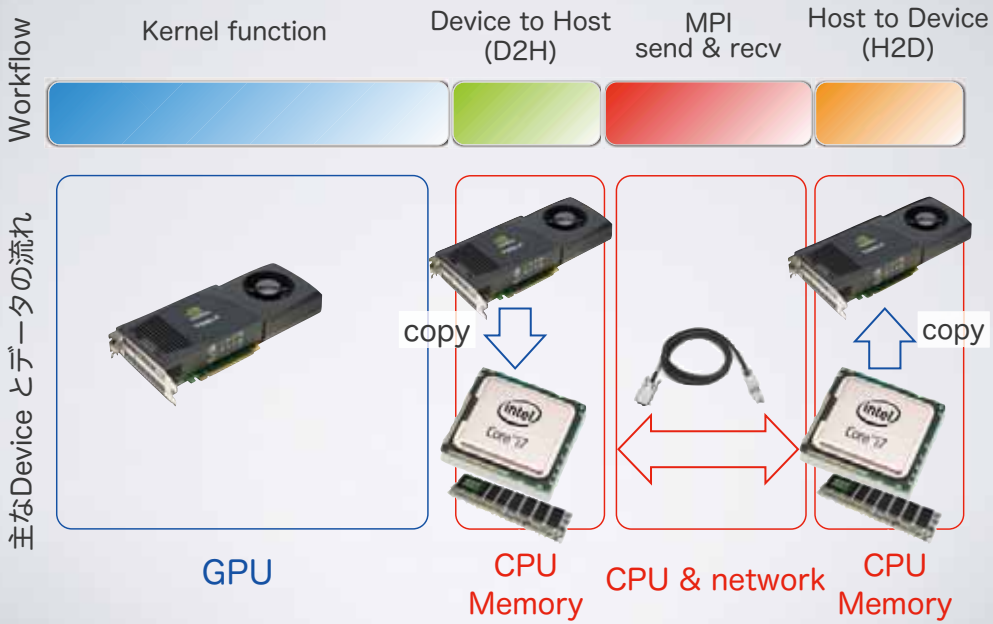


チャネル乱流計算　GPUを用いた並列化



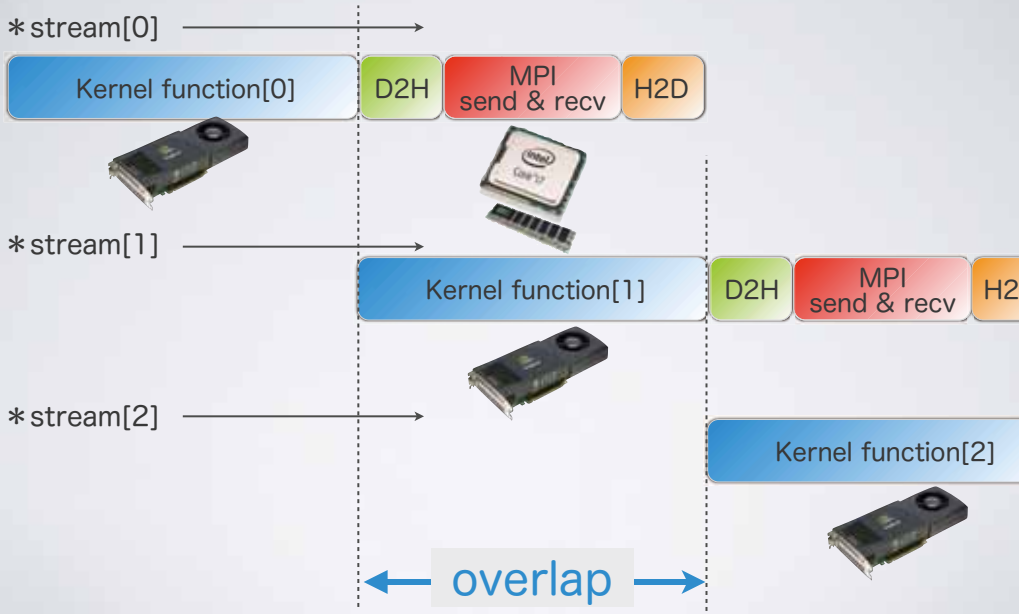
複数GPUによる並列計算

計算手順



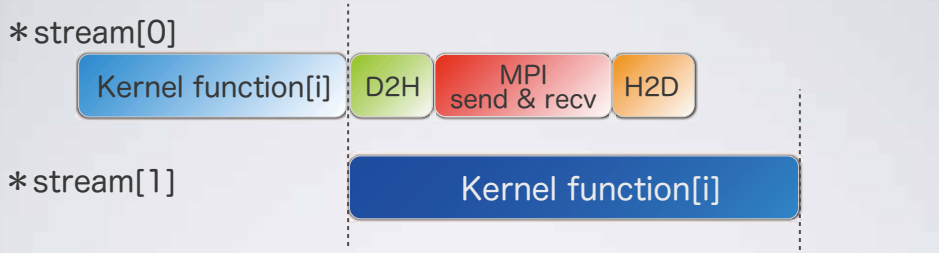
複数GPUの並列計算による通信の隠蔽

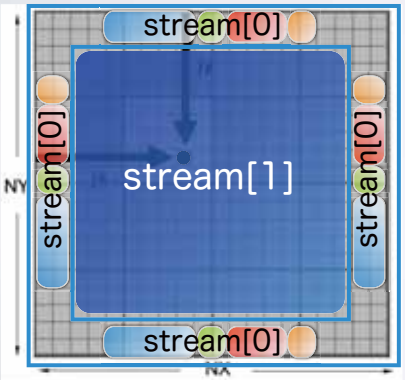
計算手順



複数GPUのLESでの通信の隠蔽

計算手順





[1] sgs model

$$\tau_{ij} = -2\nu_{SGS} S_{ij}$$

$$\nu_{SGS} = C\triangle^2 |S|$$

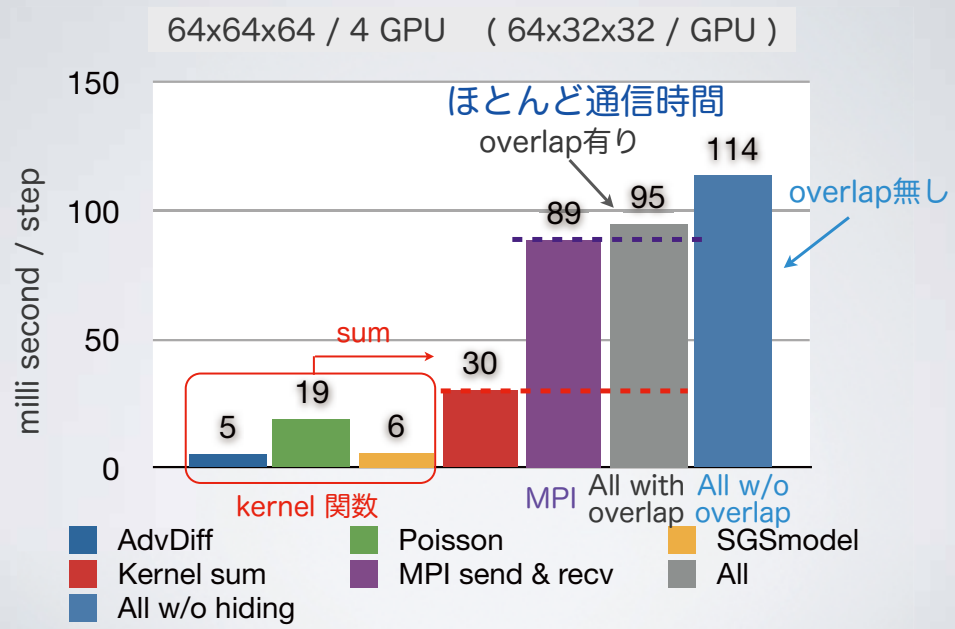
[2] advection, diffusion, sgs tensor

$$\bar{u}_i^* = \bar{u}_i^n + \left(-\frac{\partial \bar{u}_i^n \bar{u}_j^n}{\partial x_j} + \frac{1}{Re}\frac{\partial^2 \bar{u}_i^n}{\partial x_j^2} - \frac{\partial \tau_{ij}}{\partial x_j} \right) \Delta t$$

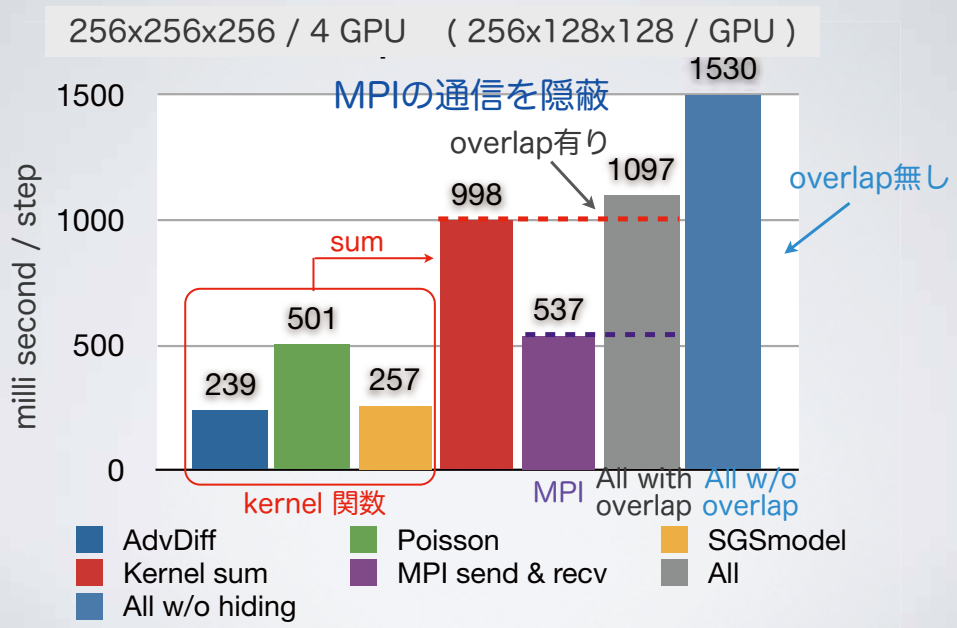
[3] Poisson equation, correction

$$\frac{\partial^2 \bar{p}^{n+1}}{\partial x_i^2} = \frac{\partial \bar{u}_i^*}{\partial x_i} \qquad \bar{u}_i^{n+1} = \bar{u}_i^* - \frac{\partial \bar{p}^{n+1}}{\partial x_i}\Delta t$$

複数GPUのLESでの計算時間



複数GPUのLESでの計算時間



LES乱流計算　応用

Tokyo Tech
TSUBAME

GPUを用いた格子乱流計算

支配方程式

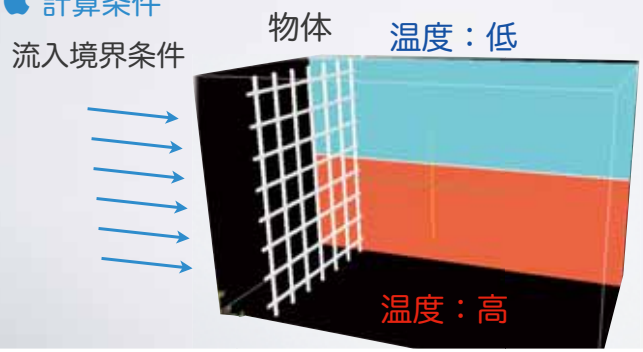
非圧縮性 Navier-Stokes 方程式

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial \bar{u}_i \bar{u}_j}{\partial x_j} = -\frac{\partial \bar{p}}{\partial x_i} + \frac{1}{Re}\frac{\partial^2 \bar{u}_i}{\partial x_j^2} - \frac{\partial \tau_{ij}}{\partial x_j} \qquad \frac{\partial \bar{u}_j}{\partial x_j} = 0$$

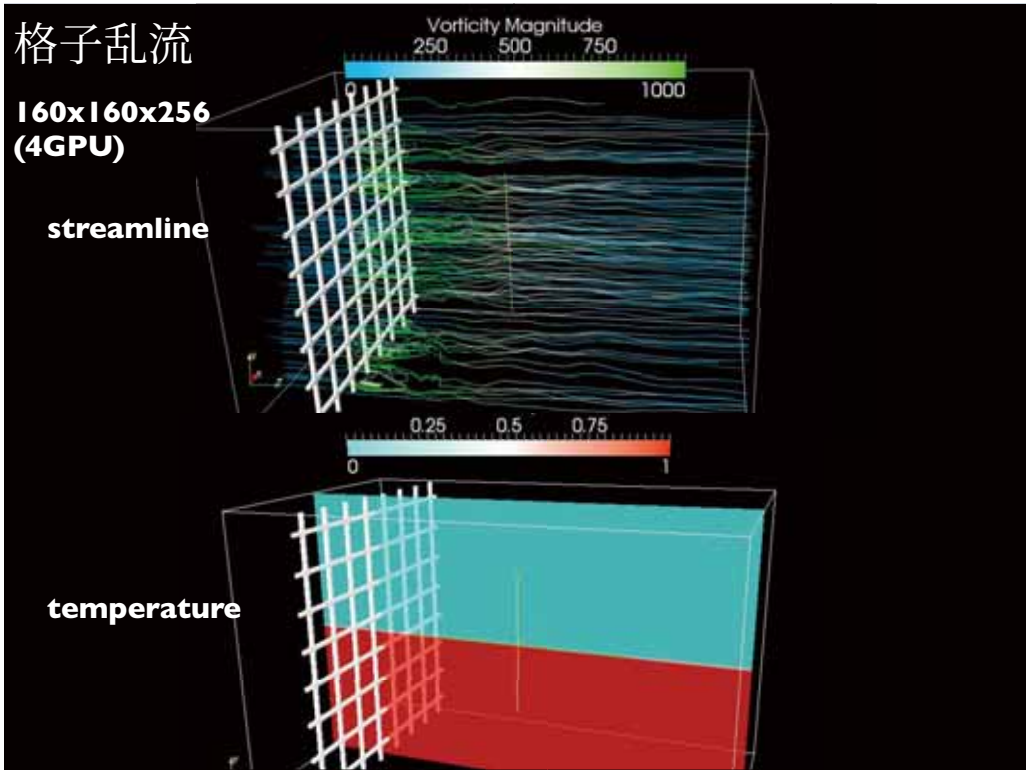
温度の対流の式 (マーカー量)

$$\frac{\partial \bar{T}}{\partial t} + \frac{\partial \bar{u}_j \bar{T}}{\partial x_j} = \frac{1}{Re \cdot Pr}\frac{\partial^2 \bar{T}}{\partial x_j^2}$$

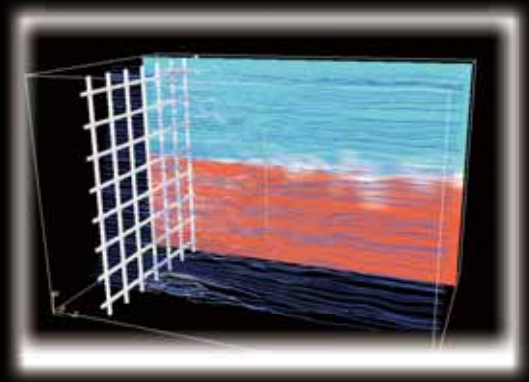
計算条件



格子乱流



物体周り流れの応用

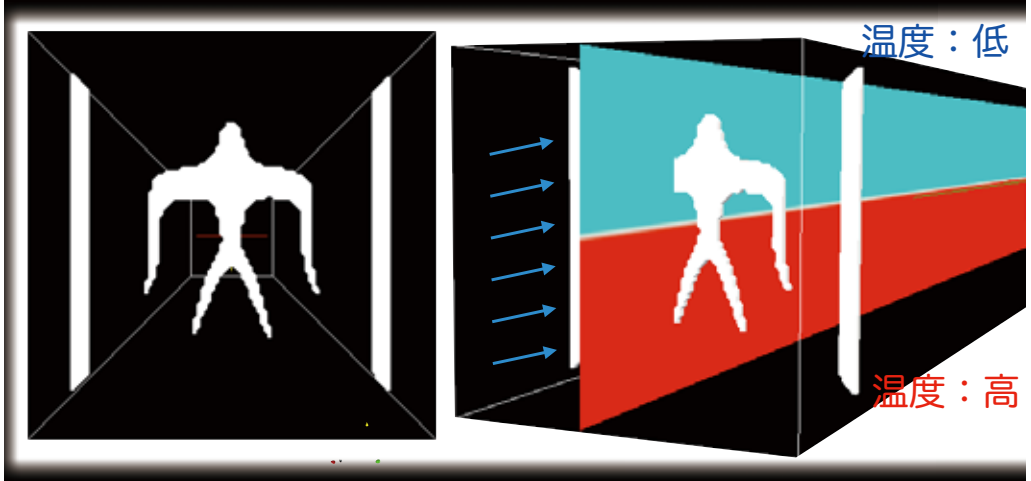


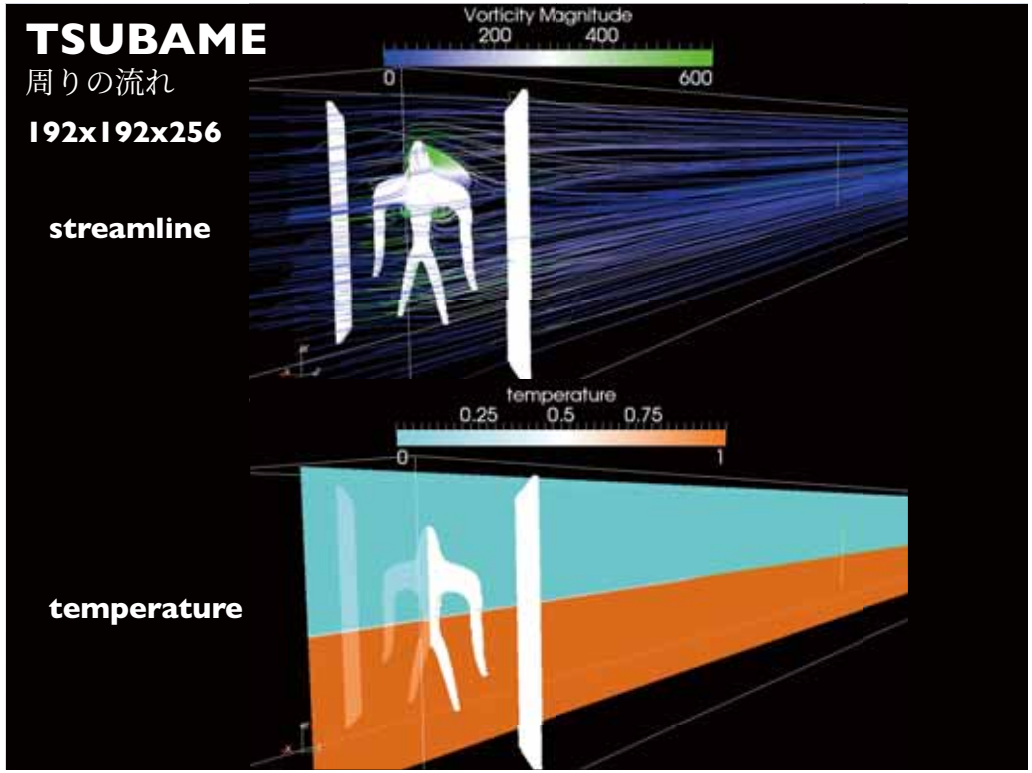
+

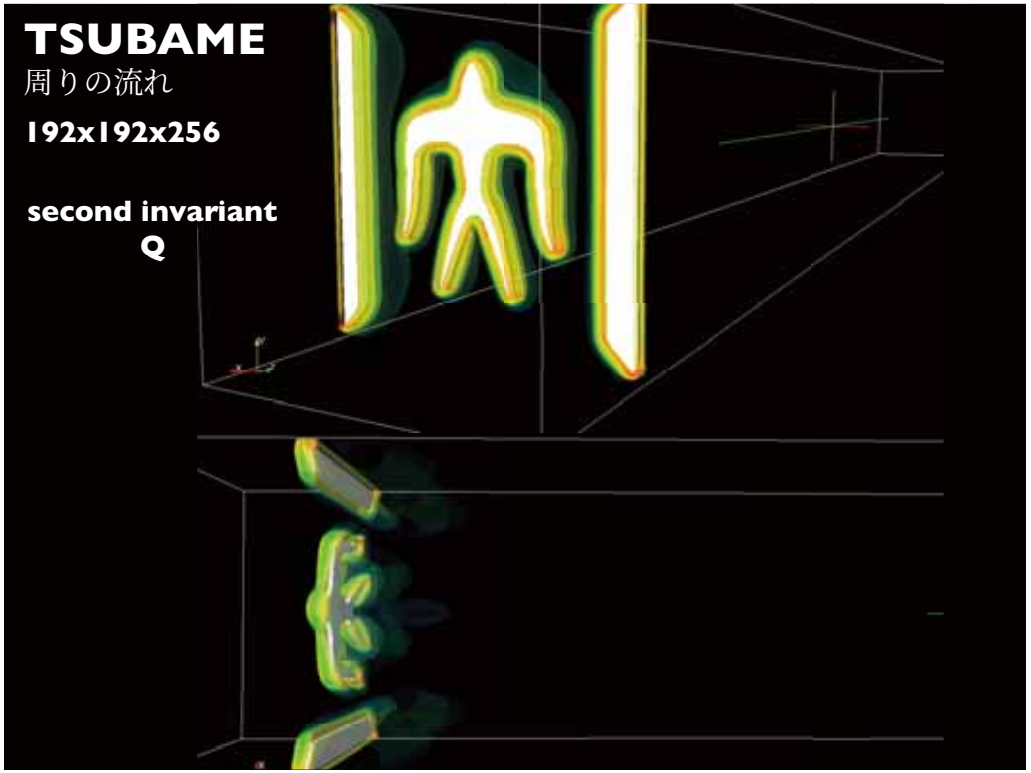
Tokyo Tech
TSUBAME

TSUBAME 周りの流れ

格子数：**192x192x256 (4GPU)**







まとめ

まとめ

1) GPUを用いてチャネル乱流のLESを行い、統計量の比較を行った。

2) LESの倍精度計算において、Tesla 2050を用いることでCore i7の30倍程度の高速化が達成された。

3) 複数GPUを用いた並列計算では、ノード間の通信を隠蔽する事が重要である。GPUのstreamを複数個定義する事により通信を隠蔽した。

4) LESを実現象で取り扱う様な物体周り流れに適用した。

今後の予定

1) TSUBAME 2.0を用いて複数GPUによる大規模計算を行う。