雲マイクロ物理解明のための大規模数値計算手法の開発

名古屋大学HPC 計算科学連携研究プロジェクト成果報告シンポジウム
名古屋大学高等研究院

2011年　5月10日

後藤俊幸、渡邊威、尾形修司、小林亮
（名工大大学院　創成シミュレーション工学専攻）

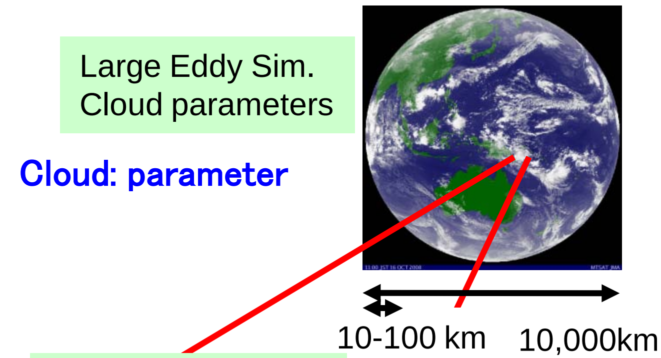
草野完也
（名古屋大学　太陽地球環境研究所 ）

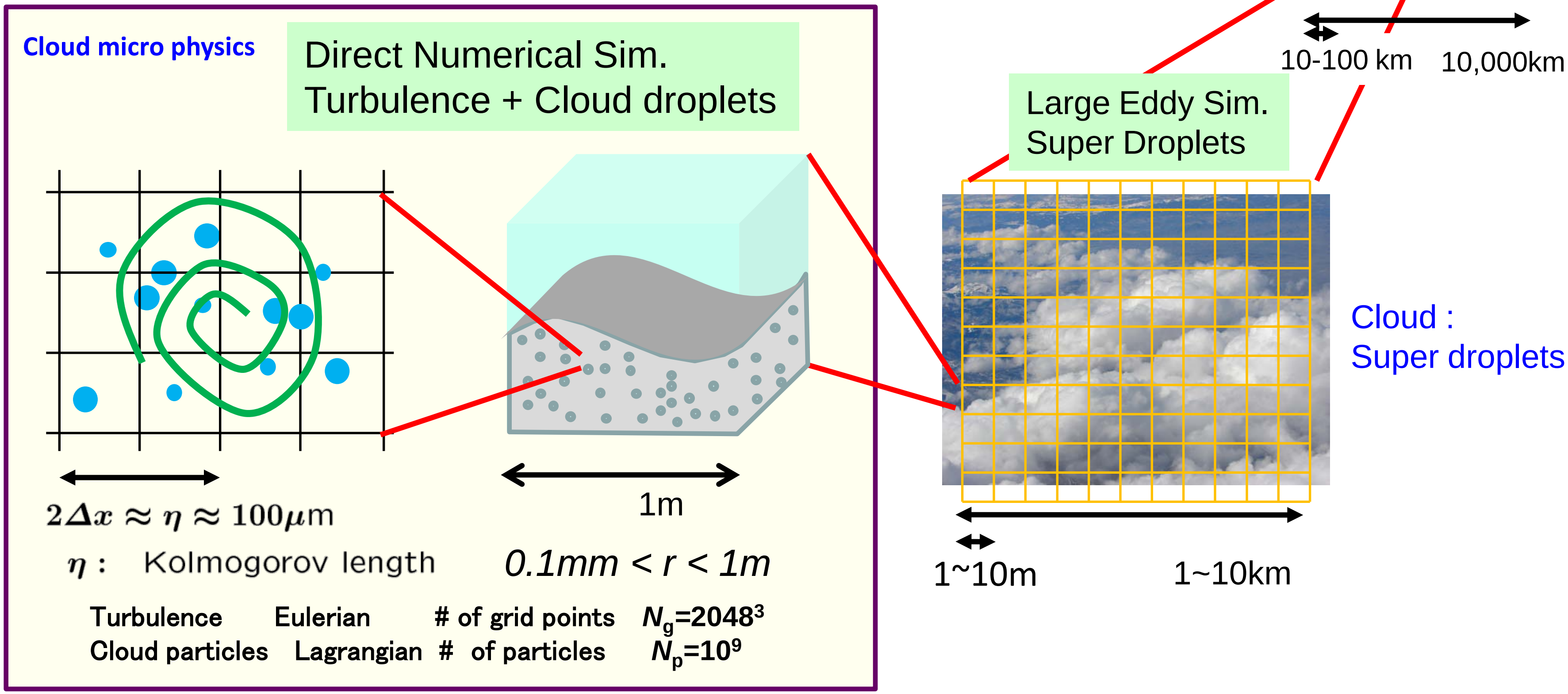
坪木和久
（名古屋大学　地球水循環研究センター）

石井克哉
（名古屋大学情報基盤センター）

Targets

Turbulence and cloud droplets in stratocumulus
Mixing of dry and moisture air
Nucleation, growth and dynamics of cloud droplets,etc.





Purpose

Development of highly efficient parallel code for the turbulence and cloud interaction

Basic Equations

Turbulence (Eulerian) **Boussinesq approximation**

$$\frac{\partial u}{\partial t} + u \cdot \nabla u = -\nabla p + \nu \nabla^2 u + e_z B + f, \qquad \nabla \cdot u = 0$$

buoyancy external force

$$\frac{\partial T}{\partial t} + u \cdot \nabla T = \kappa \nabla^2 T + \frac{L}{c_p} C_d$$

$$\frac{\partial q_v}{\partial t} + u \cdot \nabla q_v = \kappa \nabla^2 q_v - C_d$$

condensation, evaporation

$$B = g \left(\frac{T - T_0}{T_0} + \epsilon (q_v - q_{v0}) - q_c \right)$$

High Reynolds number turbulence : Spectral method

Scalar transport : Spectral or hybrid method

Cloud droplets (Lagrangian)

$$\frac{dX_j}{dt} = V_j(t)$$

$$\frac{dV_j}{dt} = \frac{1}{\tau_j(t)}\left(u(X_j(t),t) - V_j(t)\right) + ge_3$$

Stokes approximation

$$R_j(t)\frac{dR_j(t)}{dt} = KS(X_j(t),t)), \qquad R_j = \text{droplet radius}$$

$$C_d(x,t) \equiv \frac{1}{m_{air}}\frac{dm_l(x,t)}{dt} = \frac{4\pi r_l D}{\rho_0(\Delta x)^3}\sum_{k=1}^{N_\Delta} R(X_j,t)S(X_j(t),t)$$

Condensation rate

$$S = \frac{q_v}{q_{vs}(T)} - 1, \qquad \text{supersaturation rate}$$

$$K^{-1} = \frac{q_l R_v T}{D_v e_{sat}(T)} + \frac{\rho_l L}{\rho_a c_p \kappa_a T}\left(\frac{L}{R_v T} - 1\right)$$

Interpolation of velocity and scalar fields at particle position

Redistribution of cloud properties onto grid points

Simplification for development of the Eulerian codes (year 2010)

$$\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} = -\nabla p + \nu \nabla^2 \boldsymbol{u} + \nabla \boldsymbol{\tau}$$

For 3D FFT

$$\nabla \cdot \boldsymbol{u} = 0 \longrightarrow \Delta p = -\nabla \boldsymbol{u} : \nabla \boldsymbol{u}$$

$$\frac{\partial T}{\partial t} + \boldsymbol{u} \cdot \nabla T = \kappa \nabla^2 T + f_\theta$$

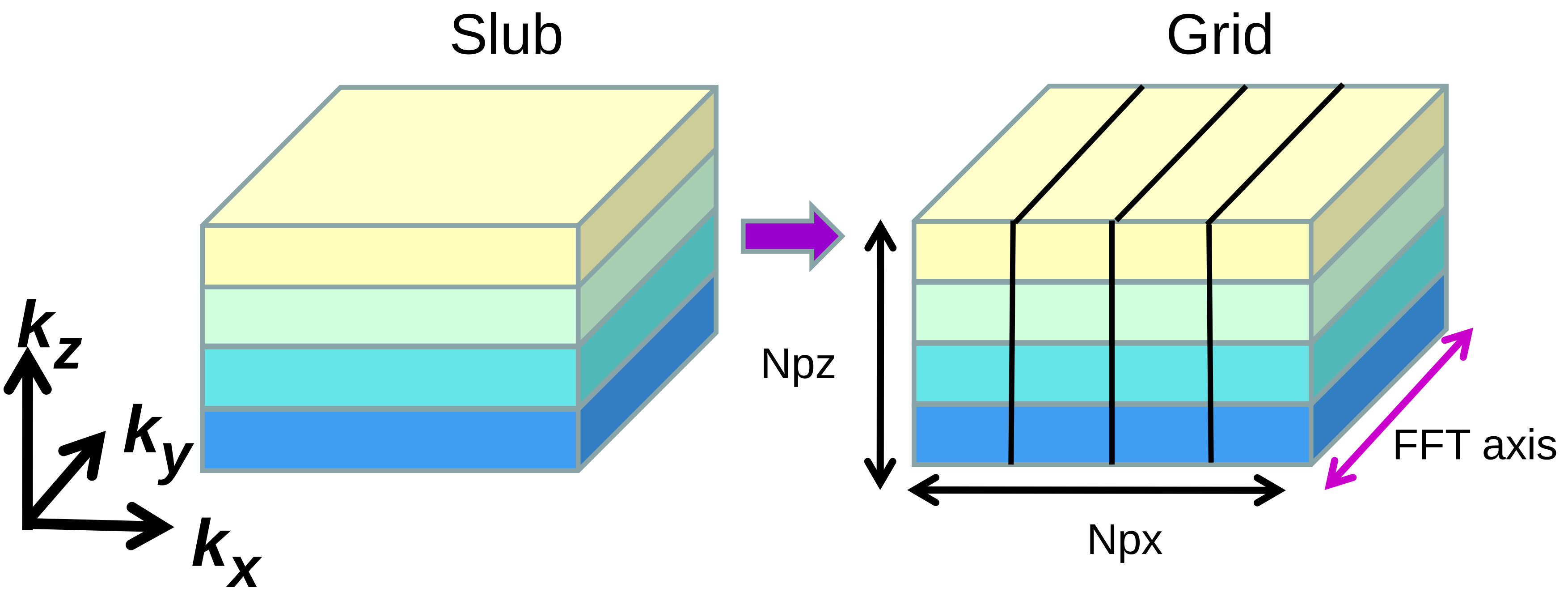
For 3D combined compact scheme

$$\frac{d\boldsymbol{X}_j}{dt} = \boldsymbol{u}(\boldsymbol{X}_j(t), t) + \boldsymbol{W}_j^+(t)$$

Particle tracking

3DFFT for Peta scale machine

Domain decomposition

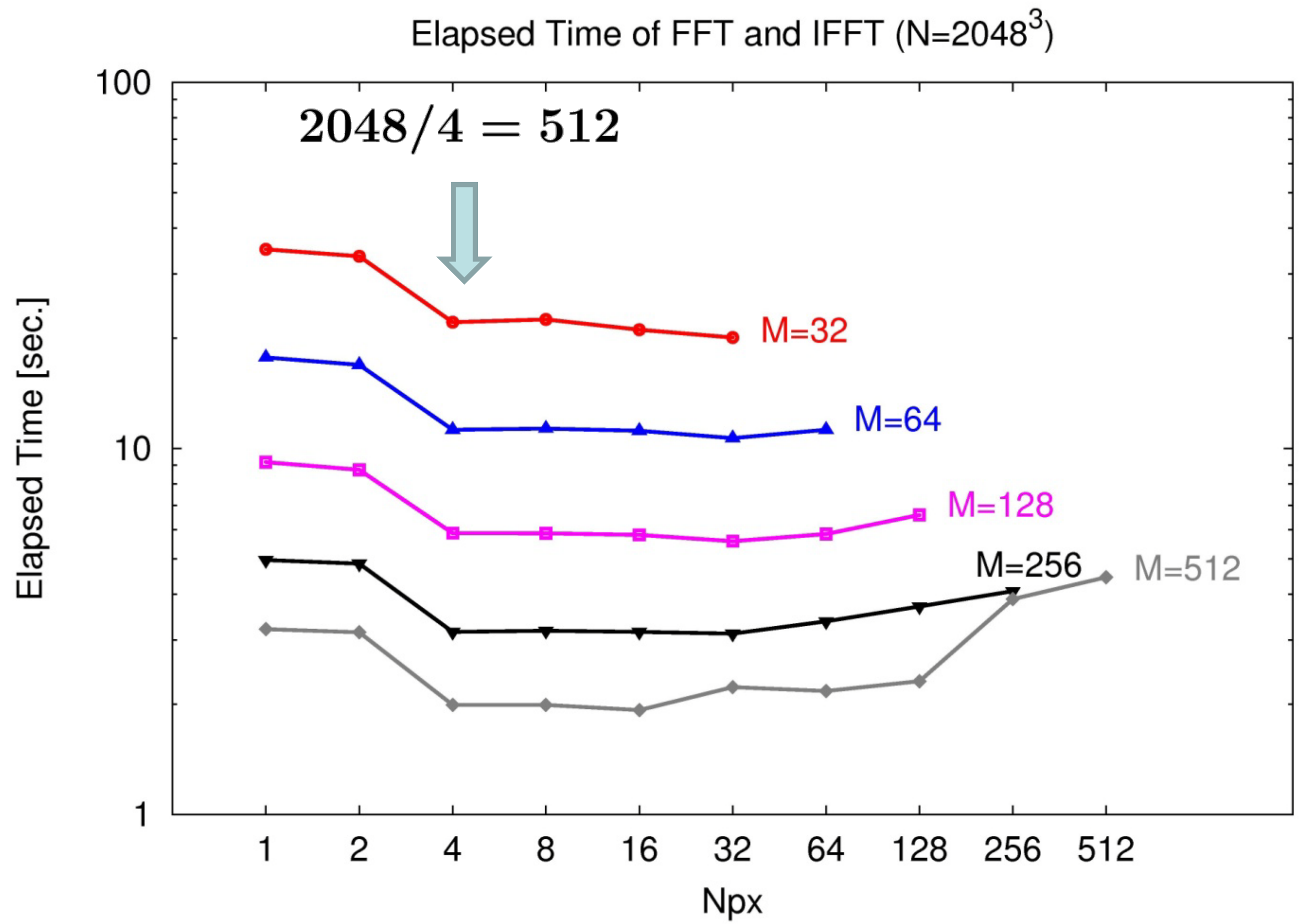


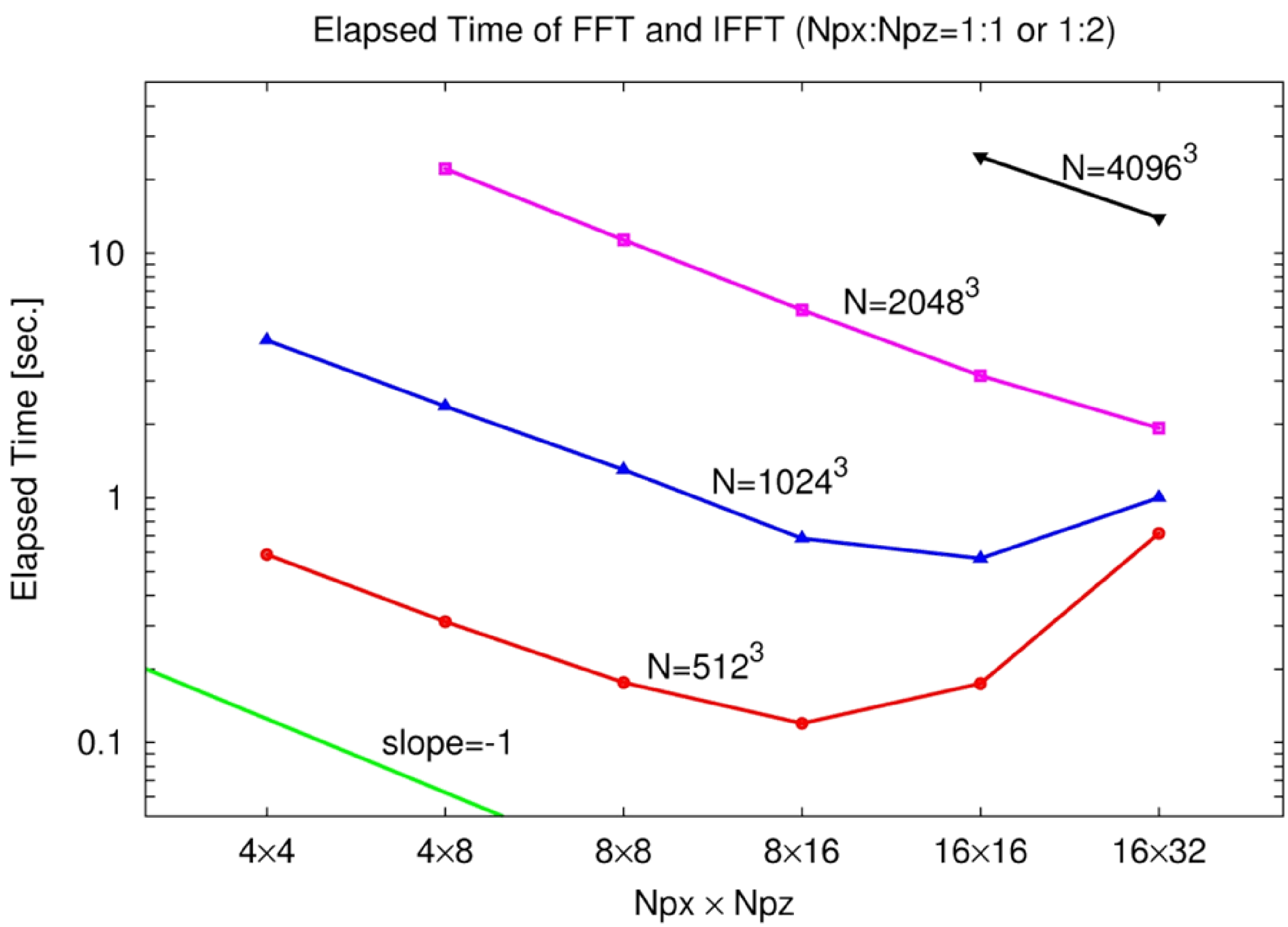
a(kx,ky,kz)

a(y,z,x)

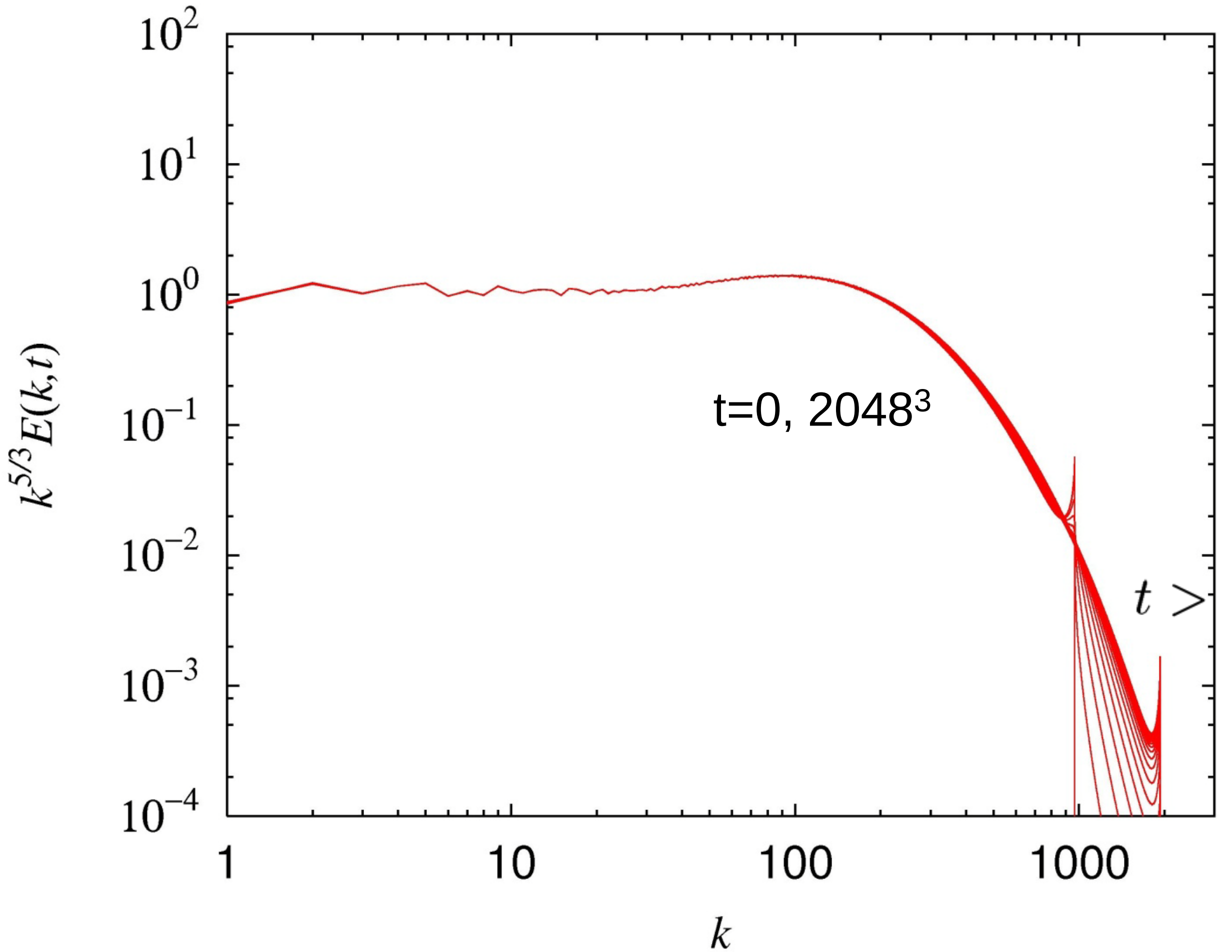
Fujitsu FX1

a(kx,ky,kz)=a(2048,2048,1024)




Elapsed Time of FFT and IFFT (Npx:Npz=1:1 or 1:2)
Elapsed Time [sec.]
10
1
0.1
N=4096^3
N=2048^3
N=1024^3
N=512^3
slope=-1
4×4
4×8
8×8
8×16
16×16
16×32
Npx × Npz

N=4096^3 Computation of turbulence (velocity)



Fujitsu FX1

2048 flat MPI proc.

$t > 0 \quad N = 4096^3$

$\approx 6.4 \times 10^{10}$

$R_\lambda \approx 820$

11TB

cf, on Jaguar (XT5 with hex-core AMD processors) at ORNL, 32k cores.

Development of 3D combined compact scheme

$$\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} = -\nabla p + \nu \nabla^2 \boldsymbol{u} + \nabla \boldsymbol{\tau}$$

$$\nabla \cdot \boldsymbol{u} = 0 \qquad \Delta p = -\nabla \boldsymbol{u} : \nabla \boldsymbol{u}$$

Non-Local in space

$$\frac{\partial T}{\partial t} + \boldsymbol{u} \cdot \nabla T = \kappa \nabla^2 T + f_\theta$$

Local in space

Finite difference

$$\left.\frac{\partial f}{\partial x}\right|_i \approx \frac{f_{i+1} - f_{i-1}}{2h}$$

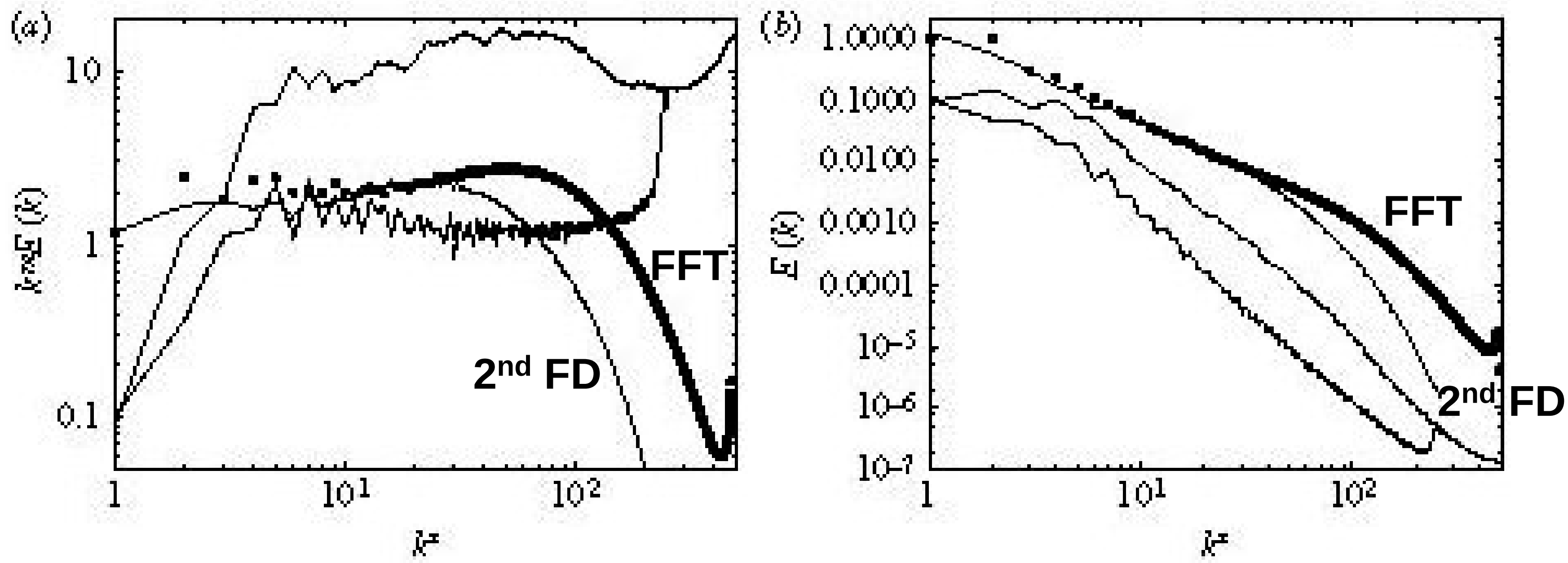


FIGURE 6. (*a*) Compensated spectra: ‒ ‒ ‒ ‒, Lamb dipoles; ·······, Taylor–Green ($m = 3$); ——, forced turbulence; ●, Gotoh *et al.* (2002) ($m = 5/3$); (*b*) Not-compensated spectra.

Orlandi JFM (2009)

Combined Compact scheme

$$a_1 f'_{i-1} + a_0 f'_i + a_2 f'_{i+1} + h\left(b_1 f''_{i-1} + b_0 f''_i + b_2 f''_{i+1}\right)$$
$$= \frac{1}{h}\left(c_1 f_{i-2} + c_2 f_{i-1} + c_0 f_i + c_3 f_{i+1} + c_4 f_{i+2}\right)$$

$a_0=1$

$$51 f'_{i-1} + 108 f'_i + 51 f'_{i+1} + 9h\left(f''_{i-1} - f''_{i+1}\right)$$
$$= \frac{107}{h}\left(f_{i+1} - f_{i-1}\right) - \frac{1}{h}\left(f_{i+2} - f_{i-2}\right)$$

$b_0=1$

$$138\left(f'_{i+1} - f'_{i-1}\right) - 18h\left(f''_{i-1} - 6f''_i + f''_{i+1}\right)$$
$$= -\frac{1}{h}\left(f_{i+2} + f_{i-2}\right) + \frac{352}{h}\left(f_{i+1} + f_{i-1}\right) - \frac{702}{h} f_i$$

Combined compact scheme (f’,f’’,f’’’)

$$
\begin{pmatrix}
\mathbf{B} & \mathbf{C} & \mathbf{O} & \cdots & \cdots & \mathbf{O} & \mathbf{A} \\
\mathbf{A} & \mathbf{B} & \mathbf{C} & \mathbf{O} & & & \mathbf{O} \\
\mathbf{O} & \mathbf{A} & \mathbf{B} & \mathbf{C} & \mathbf{O} & & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
\vdots & & \mathbf{O} & \mathbf{A} & \mathbf{B} & \mathbf{C} & \mathbf{O} \\
\mathbf{O} & & & \mathbf{O} & \mathbf{A} & \mathbf{B} & \mathbf{C} \\
\mathbf{C} & \mathbf{O} & \cdots & \cdots & \mathbf{O} & \mathbf{A} & \mathbf{B}
\end{pmatrix}
\begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \vdots \\ \mathbf{x}_{N-2} \\ \mathbf{x}_{N-1} \\ \mathbf{x}_N \end{pmatrix}
=
\begin{pmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \\ \mathbf{d}_3 \\ \vdots \\ \mathbf{d}_{N-2} \\ \mathbf{d}_{N-1} \\ \mathbf{d}_N \end{pmatrix},
$$

Nihei and Ishii, JCP (2003)

$$
\mathbf{A} = \begin{pmatrix} a_1 & -b_1\Delta x & c_1(\Delta x)^2 \\ -\frac{a_2}{\Delta x} & b_2 & -c_2\Delta x \\ \frac{a_3}{(\Delta x)^2} & -\frac{b_3}{\Delta x} & c_3 \end{pmatrix}, \quad
\mathbf{B} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},
$$

$$
\mathbf{C} = \begin{pmatrix} a_1 & b_1\Delta x & c_1(\Delta x)^2 \\ \frac{a_2}{\Delta x} & b_2 & c_2\Delta x \\ \frac{a_3}{(\Delta x)^2} & \frac{b_3}{\Delta x} & c_3 \end{pmatrix}, \quad
\mathbf{O} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix},
$$

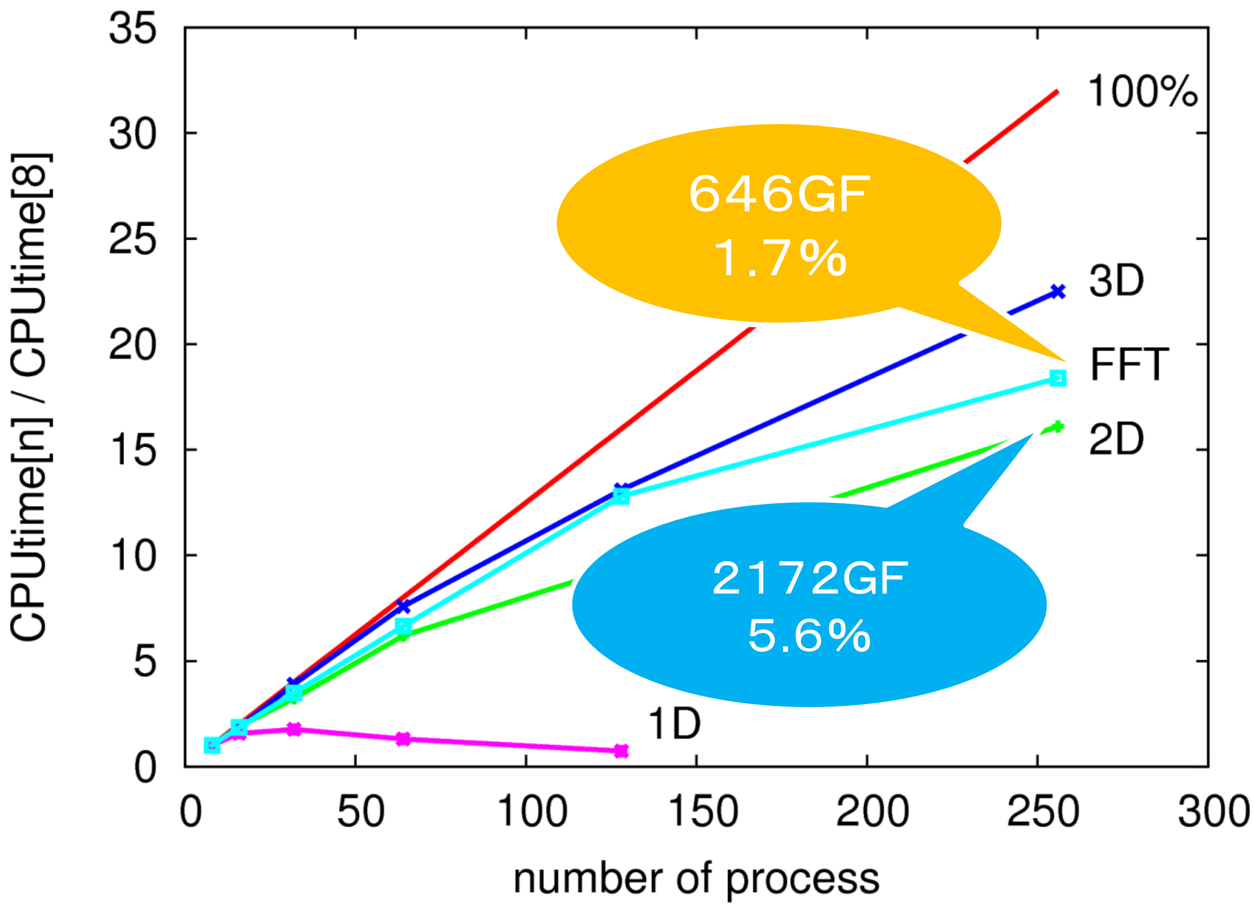
$$
\mathbf{x}_i = \begin{pmatrix} f_i' \\ f_i'' \\ f_i''' \end{pmatrix}, \quad \text{and} \quad
\mathbf{d}_i = \begin{pmatrix} \frac{d_1}{\Delta x}(f_{i+1} - f_{i-1}) \\ \frac{d_2}{(\Delta x)^2}(f_{i+1} - 2f_i + f_{i-1}) \\ \frac{d_3}{(\Delta x)^3}(f_{i+1} - f_{i-1}) \end{pmatrix}.
$$

Domain decomposition
in 1,2,3 dimensions

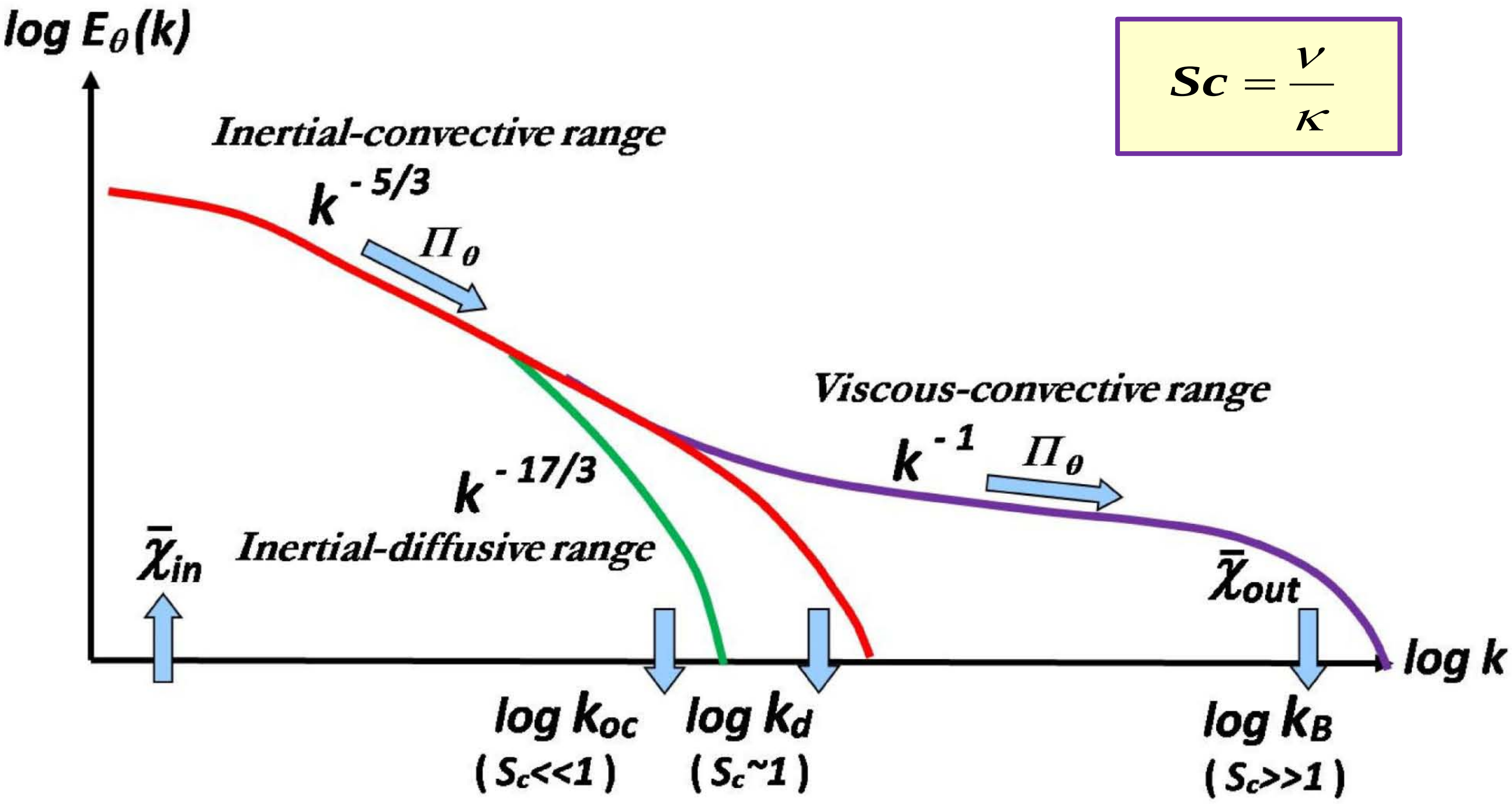
Simultaneous computation

MPI communication

MPI parallelization efficiency of combined compact scheme



Spectrum of scalar variance



$$Sc = \frac{\nu}{\kappa}$$

Development of hybrid code for passive scalar in turbulence

$$\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} = -\nabla p + \nu \nabla^2 \boldsymbol{u} + \nabla \boldsymbol{\tau}$$

$$\nabla \cdot \boldsymbol{u} = 0 \qquad \Delta \pi = -\nabla \boldsymbol{u} : \nabla \boldsymbol{u}$$

Non-Local in space

$$\frac{\partial T}{\partial t} + \boldsymbol{u} \cdot \nabla T = \kappa \nabla^2 T + f_\theta$$

Local in space

Hybrid code

Velocity : Spectral method

Scalar : Combined compact scheme

Comparison of Hybrid method to fully Spectral method

	$N_{velocity}$	N_{Scalar}	ν	Sc	CPU time (1step)
Run1(Full Spec)	1024	1024	8.0×10^{-4}	1	8.1(s) ↓ 29%
Run2(Hybrid)	1024	1024	8.0×10^{-4}	1	5.7(s)
Run3(Full Spec)	1024	1024	8.0×10^{-3}	50	8.1(s) ↓ 83%
Run4(Hybrid)	256	1024	8.0×10^{-3}	50	1.4(s)

CPU time

Run 1 vs Run 2

●Hybrid=0.7 ×Full Spectral (Scalar part : 0.4 ×Full spectral)

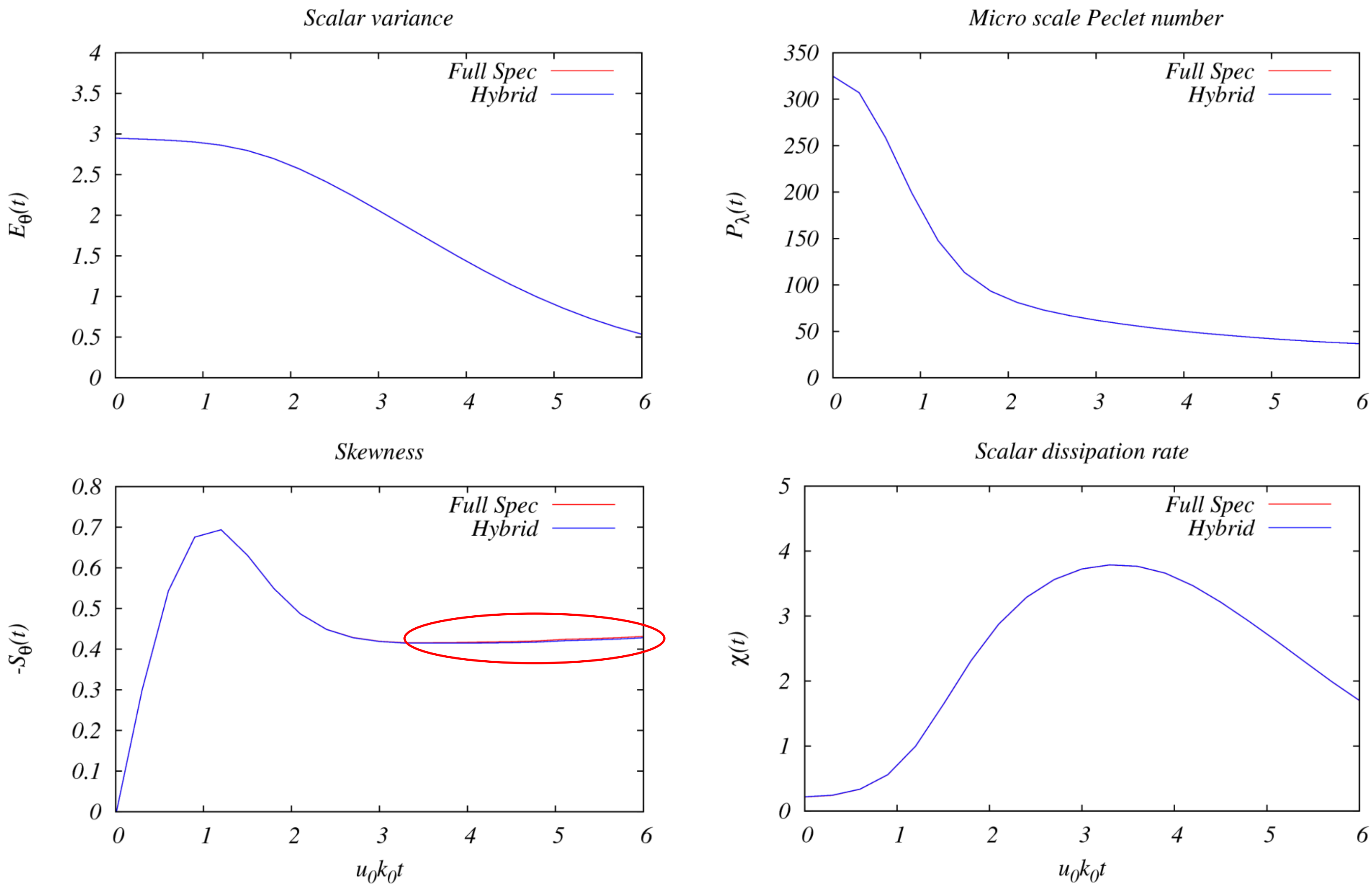
Run 3 vs Run 4

¼ number of grid points for the velocity because of Sc=50

●Hybrid=0.17 ×Full Spectral

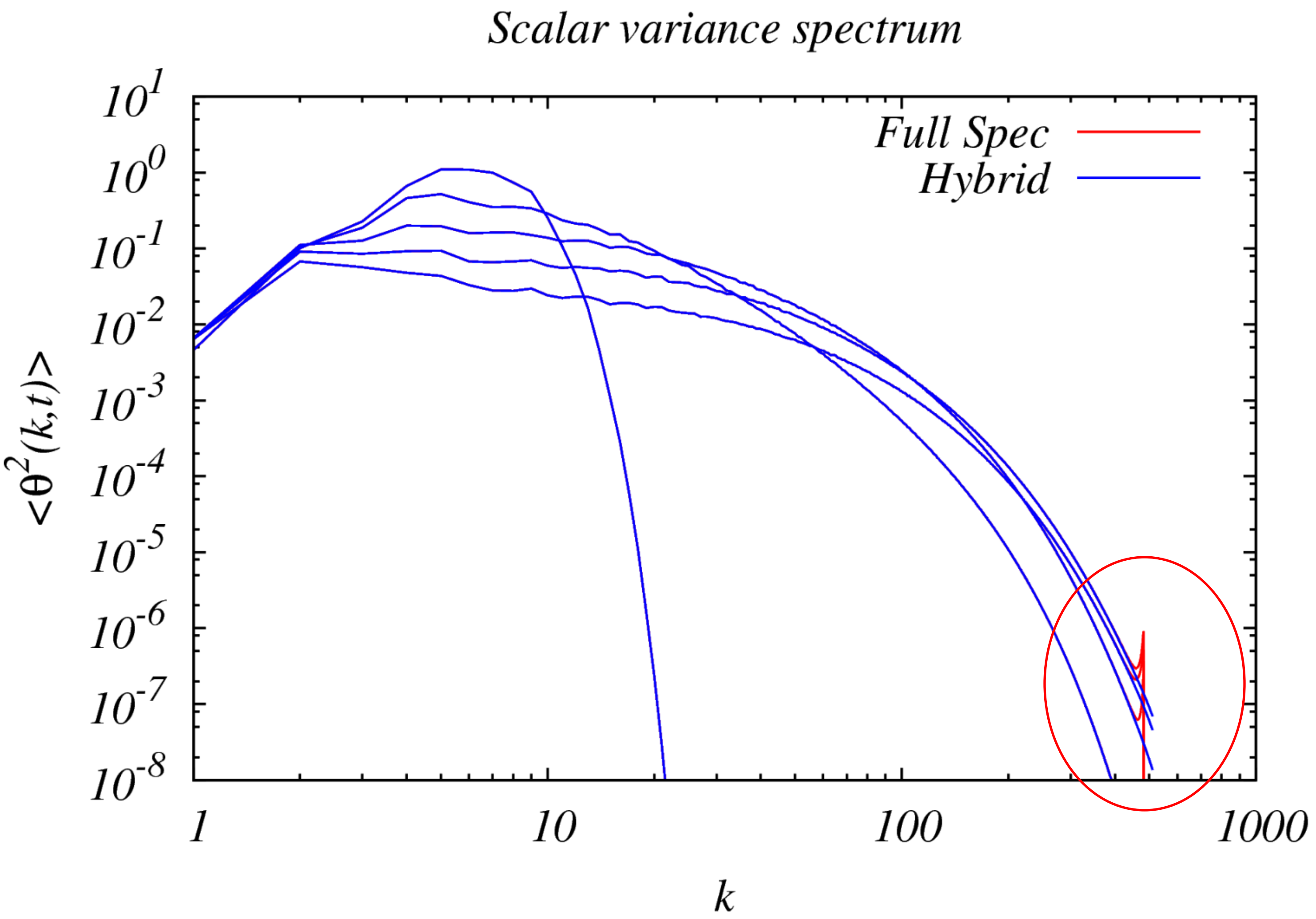
Comparison of Run 1 with Run 2

Sc=1



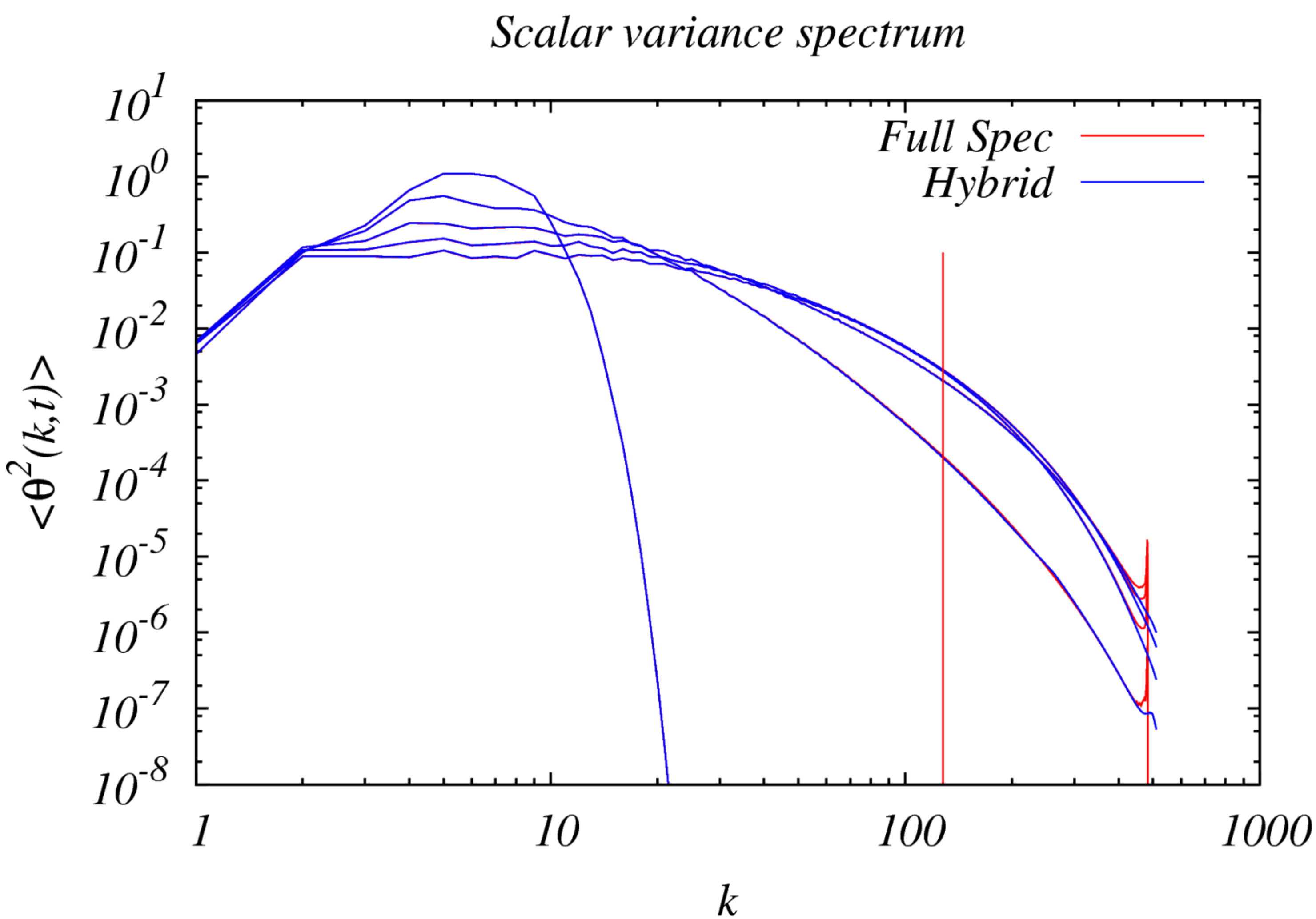
Comparison of Run 1 with Run 2

Sc=1



Comparison of Run 3 with Run 4

Sc=50



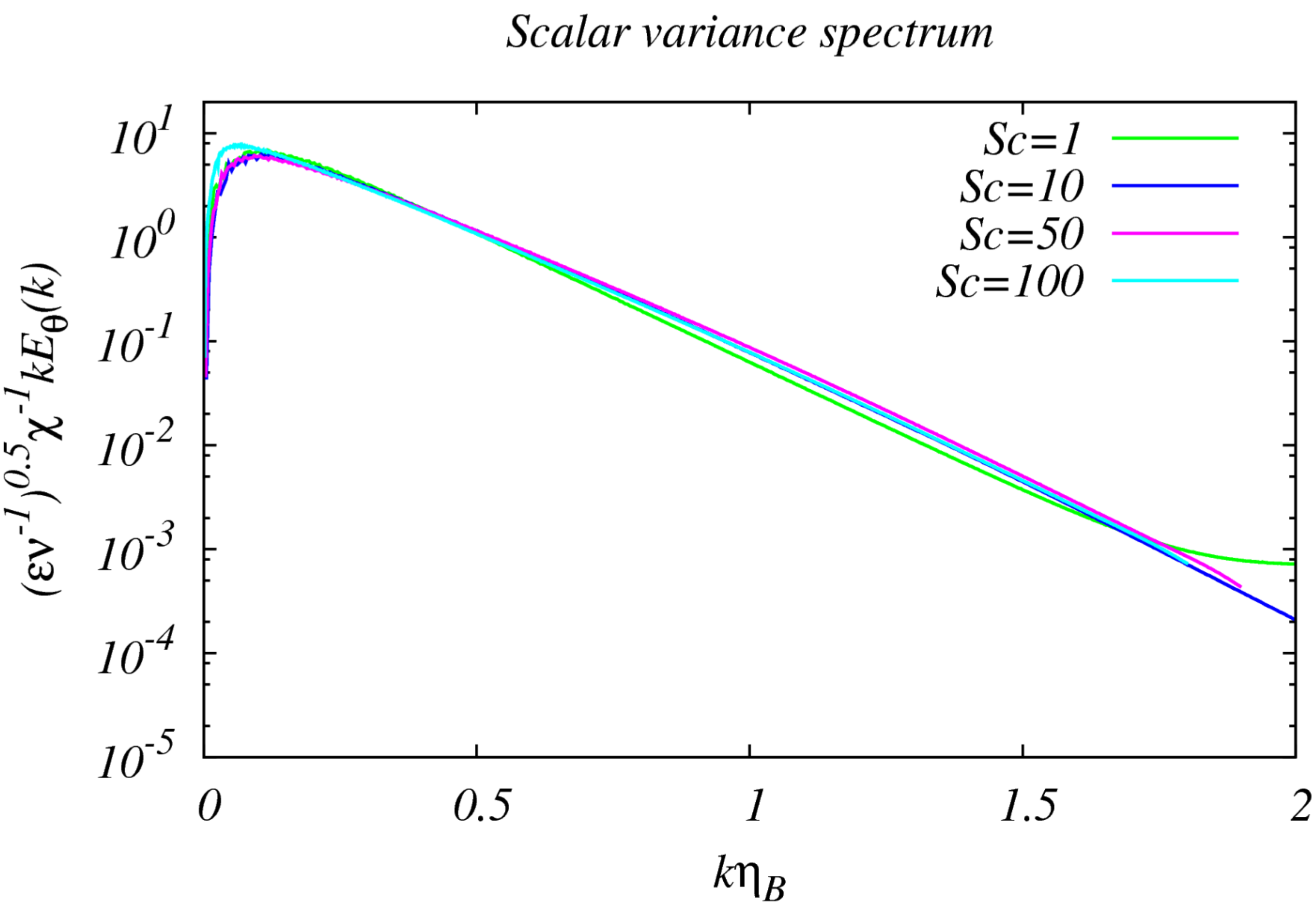
Scalar spectrum in Kraichnan form

	$N_{velocity}$	N_{Scalar}	ν	Sc	$K_{\max}^{velocity}\eta$	$K_{\max}^{Scalar}\eta$
Run5	512	1024	1.0×10^{-3}	1	1.2	2.4
Run6	256	1024	5.0×10^{-3}	10	1.9	2.4
Run7	256	1024	8.0×10^{-3}	50	2.6	1.5
Run8	512	2048	5.0×10^{-3}	100	3.8	1.5

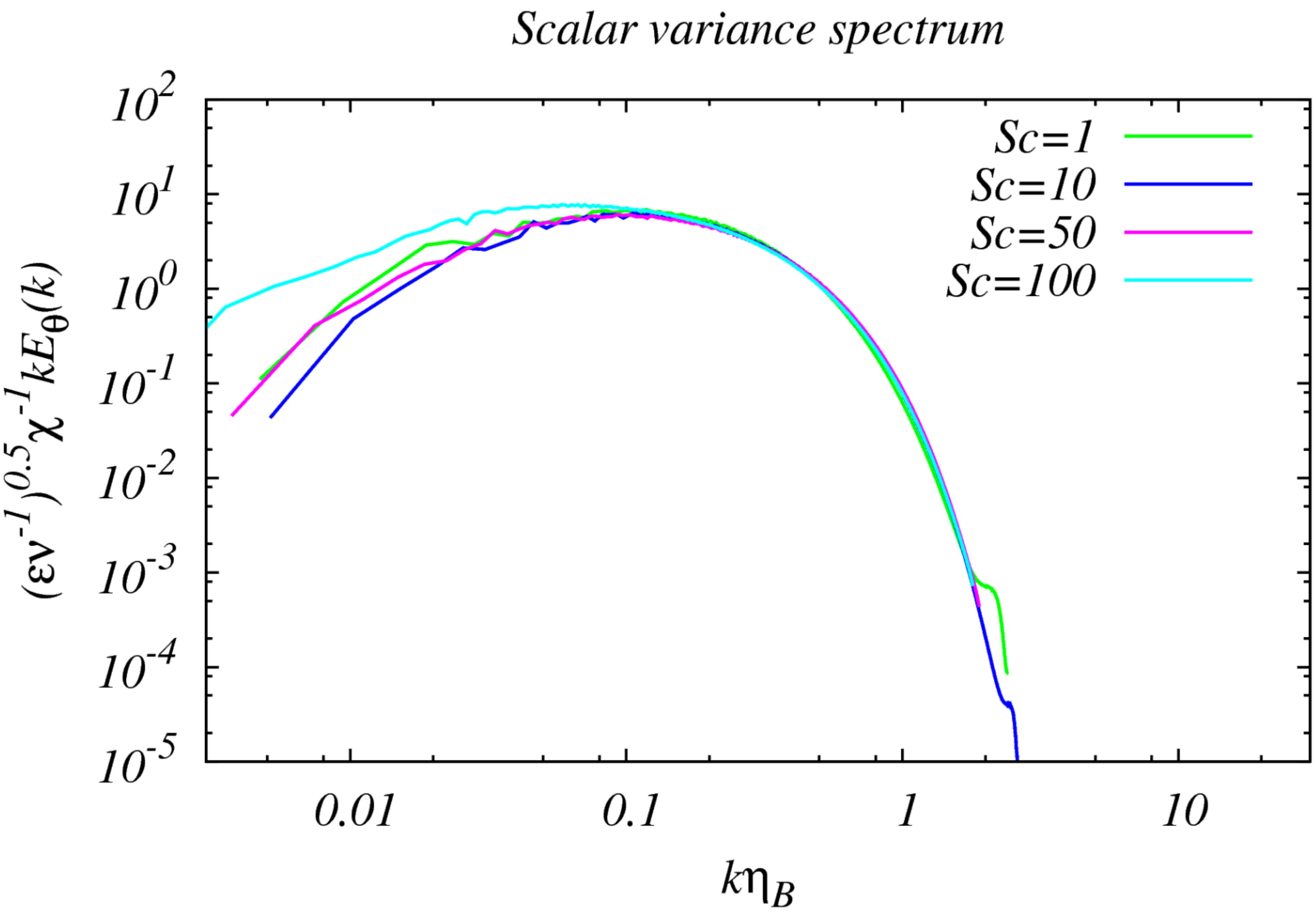
$$E_\theta(k) = C_B \overline{\chi} (\upsilon / \overline{\varepsilon})^{1/2} k^{-1} (1 + (6C_B)^{1/2}) \exp(-(6C_B)^{1/2} (k\overline{\eta}_B))$$

C_B : Batchelor constant

Summary

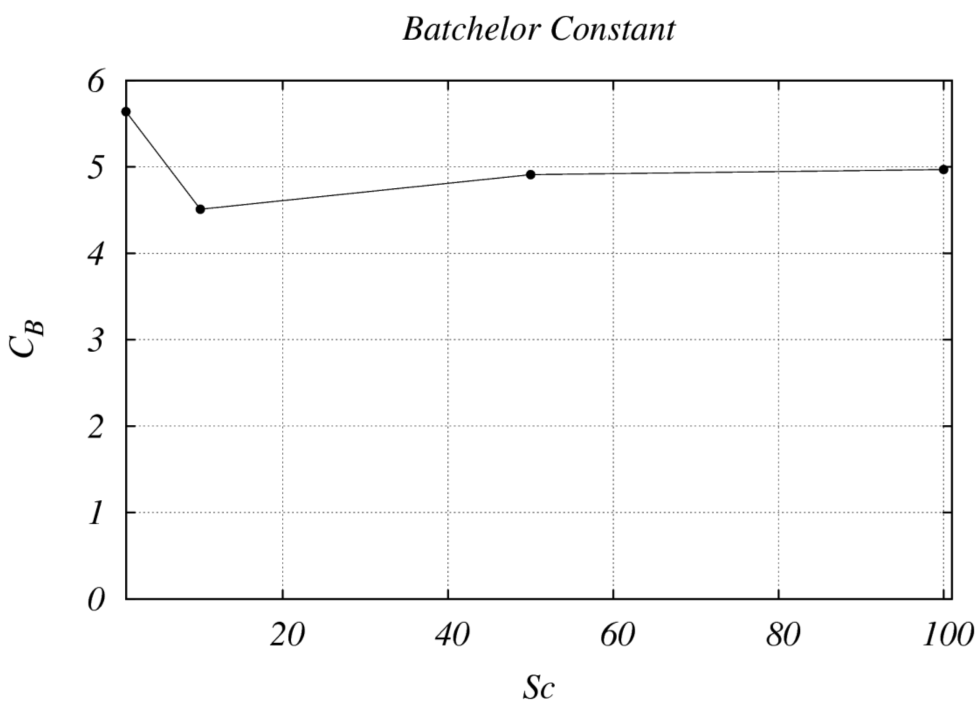

Scalar variance spectrum
$(\varepsilon\nu^{-1})^{0.5}\chi^{-1}kE_\theta(k)$
$k\eta_B$
10^{1}
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
0
0.5
1
1.5
2
Sc=1
Sc=10
Sc=50
Sc=100

Summary


Scalar variance spectrum
Sc=1
Sc=10
Sc=50
Sc=100
10^2
10^1
10^0
10^-1
10^-2
10^-3
10^-4
10^-5
(εν^-1)^0.5 χ^-1 kE_θ(k)
0.01
0.1
1
10
kη_B

Batchelor constant

	Sc	R_λ	Pe	C_B
Run5	1	88	40	5.64
Run6	10	28	66	4.30
Run7	50	17	118	4.91
Run8	100	23	235	4.97



まとめ

要素技術の開発

３D　FFTのMPI並列化：　２次元分割、高速化、NSソルバへの実装

NSソルバの高速化　：　２次元分割、高速化
4096^3 が2048 Flat　MPI　並列で稼働中

結合コンパクト差分MPI　並列化：　１，２，３次元分割　ハイブリッド並列

粒子数のラグランジュコード並列化：　少数プロセスのNSソルバ
＋多数プロセスの粒子ソルバ

今後

３D　FFTのMPI並列化：　さらに調整

NSソルバの高速化　：　さらに調整

結合コンパクト差分：　NSソルバとの結合、実装、調整

粒子数のラグランジュコード：　雲マイクロ物理の数理モデルの
プロトタイプの導入