Gyrokinetic simulations of multi-scale turbulence on the supercomputer Fugaku

Shinya Maeyama

Nagoya University

IFERC Workshop on GPU Programming 2021, virtual, 14 June 2021

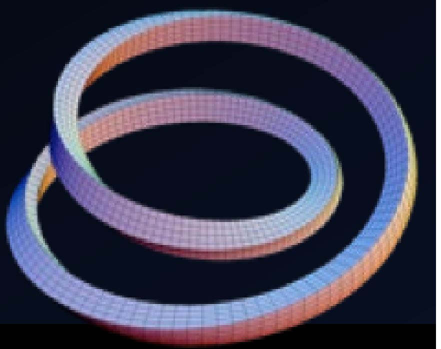
Outline

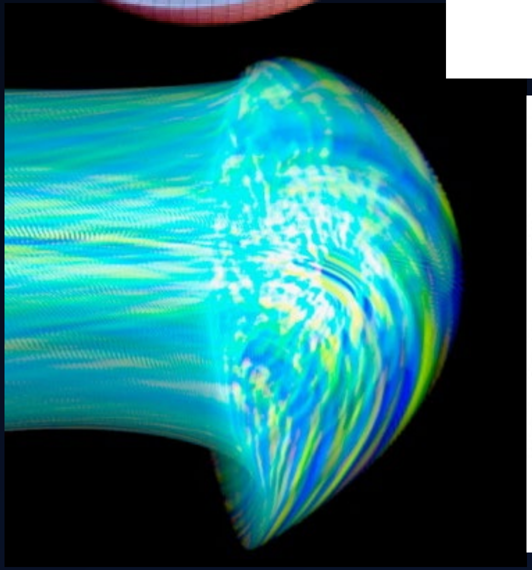
- Optimization of GKV code on the supercomputer Fugaku - 10 min
- Multi-scale turbulence simulations toward burning plasma - 10 min

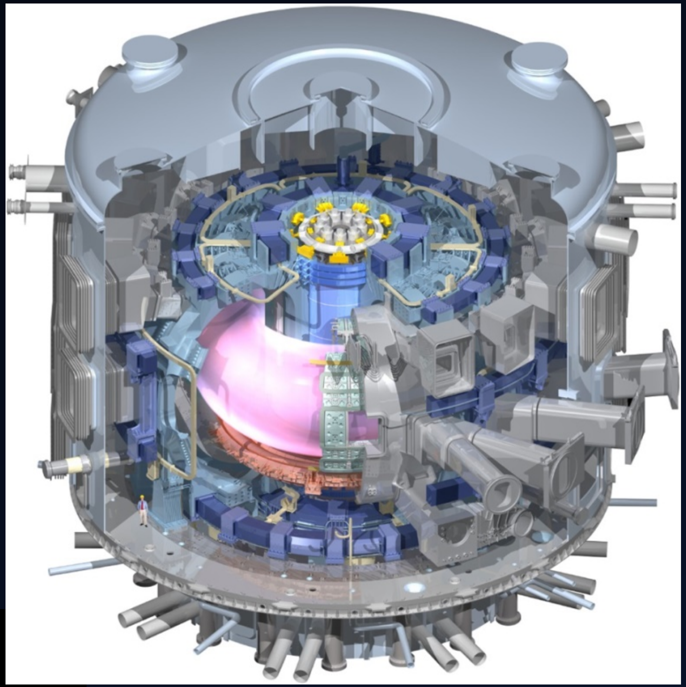
GyroKinetic Vlasov code GKV

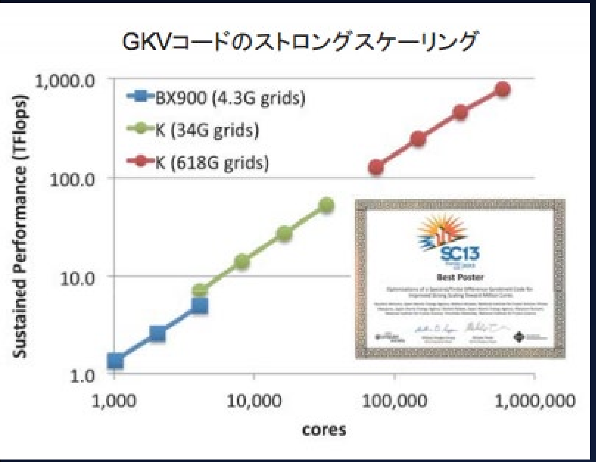
- Plasma turbulence simulation based on delta-f gyrokinetic model
- 5D Eulerian solver
    - Fourier spectral method(x, y)+Finite difference$(z, v_{\parallel}, \mu)$
    - Explicit + implicit (collision) time integration
    - MPI/OpenMP hybrid parallelization
- High resolutions in local flux tube geometry
    - Multi-scale simulation from ion to electron scales
    - Multi-species collision
- Free download
    - https://p.phys.nagoya-u.ac.jp/gkv/
    - https://github.com/GKV-developers/gkvp

Flux tube

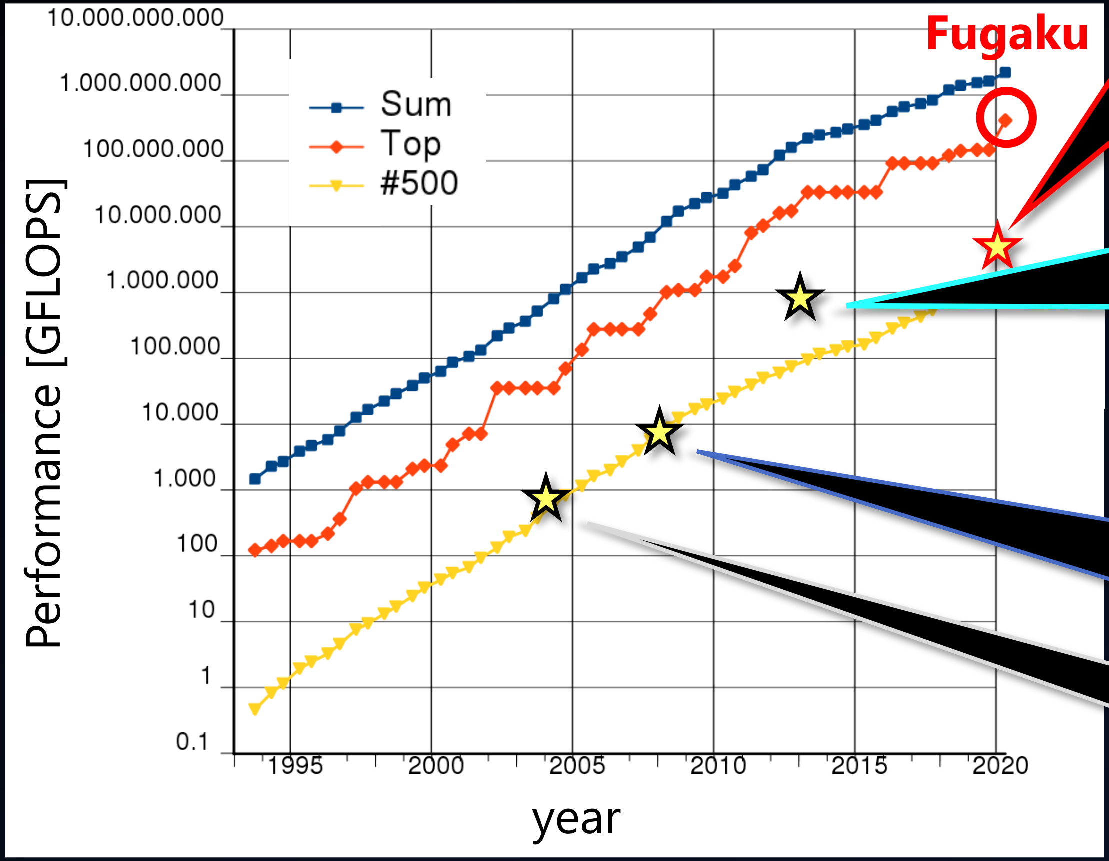






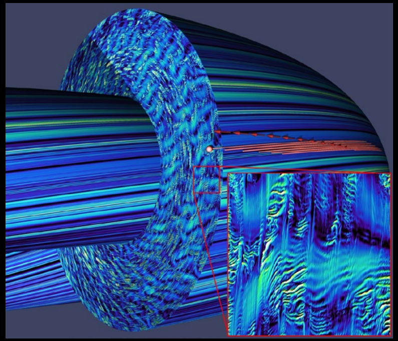


GKV as an HPC application



Physics model of GKV is extended along with HPC.

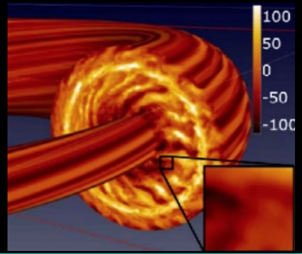
Multi-scale turbulence simulations for burning plasma (e, D, T, He)



NOTE: Performance of GKV on Fugaku 12288 nodes (March 2021)

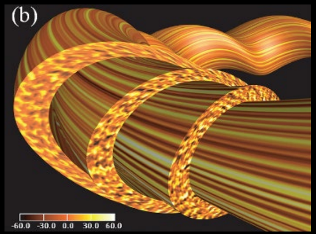
Multi-scale turbulence simulations for hydrogen-electron plasma

cf. Maeyama, et al., Phys. Rev. Lett. (2015)



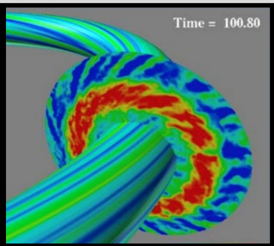
Ion-scale turbulence simulations in complex LHD plasma

cf. Nunami, et al., Phys. Plasmas (2012)



Ion-scale turbulence simulations in simple Tokamak plasma

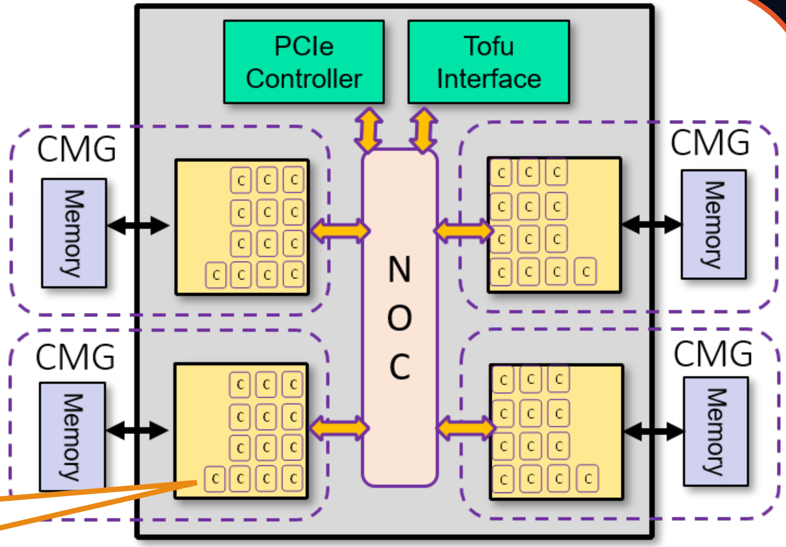
cf. Watanabe, et al., Nucl. Fusion (2006)



Supercomputer Fugaku

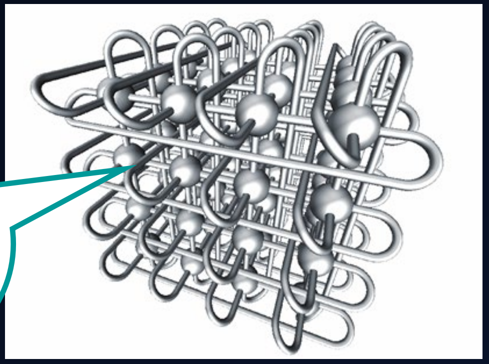
158,976 nodes, 537 PFLOPS

12 cores/CMG,   4 CMG/CPU



- CMG: Core Memory Group (NUMA Node)
- NOC: Network on Chip

1 CPU/node



6D Tofu interconnect (3D torus network from a user view)

32 FLOP / (core・cycle)
2.2 GHz

- 512-bit SIMD (SVE)
- 2FMA

From K to Fugaku

	K	Fugaku
System perf. [DP PFLOPS]	10.6	537
# node/system	82,944	158,976
# core/node	8	48
Node perf. [TFLOPS]	0.128	3.38
Core perf. [GFLOPS]	16	70.4
Memory size [GB]	16	32
Memory BW [GB/s]	64	1024
Memory Byte/FLOP	**0.5**	**0.30**
Interconnect BW [GB/s]	20 (= 5.0 x 4)	40.8 (= 6.8 x 6)
Interconnect Byte/FLOP	**0.156**	**0.012**

Memory-BW-limited code can still be efficient.

Commun./Comput. cost ratio becomes much severe.
→ Need to increase arithmetic intensity.

Typical computation/communication of GKV

$$\frac{\partial \tilde{f}_s}{\partial t} + \left(v_\parallel \frac{\boldsymbol{B} + \tilde{\boldsymbol{B}}_\perp}{B} + \boldsymbol{v}_{sG} + \boldsymbol{v}_{sC} + \tilde{\boldsymbol{v}}_E \right) \cdot \nabla \tilde{f}_s + \frac{dv_\parallel}{dt} \frac{\partial \tilde{f}_s}{\partial v_\parallel} = S_s + C_s,$$

$$\nabla_\perp^2 \tilde{\phi} = -\frac{1}{\varepsilon_0} \sum_s e_s (\tilde{n}_s + \tilde{n}_{s,\mathrm{pol}}),$$

$$\nabla_\perp^2 \tilde{A}_\parallel = -\mu_0 \sum_s e_s \tilde{u}_{\parallel s},$$

- Domain decomposition of 5D phase space and plasma species $\left(n_x, \frac{n_y}{P_w}, \frac{n_z}{P_z}, \frac{n_v}{P_v}, \frac{n_\mu}{P_\mu}, \frac{n_s}{P_s}\right)$
- Density/current evaluation: MPI_allreduce among (v,m,s)
- Finite difference (z,v): MPI_isend/irecv to (z,v)
- 2D FFT (x,y): MPI_alltoall $\left(n_x, n_y, \frac{n_v}{P_v}, \frac{n_z n_\mu}{P_z P_\mu P_w}, \frac{n_s}{P_s}\right)$ → **Improved implementation** [Asahi'19CCPE]
- Implicit collision: MPI_alltoall $\left(n_v, n_\mu, n_s, \frac{n_z}{P_z}, \frac{n_x n_y}{P_w P_v P_\mu P_s}\right)$

→ **MPI-free iterative solver** [Maeyama'19CPC]

Improved parallel spectral calculation

Spectral calculation of nonlinear advection $N = v_x \partial_x f + v_y \partial_y f$ by transpose-split method.

- As-is (**y2x**): $ik_x \hat{f}_{\boldsymbol{k}}, ik_y \hat{f}_{\boldsymbol{k}} \left(n_x, \frac{n_y}{P_w}, \frac{n_z}{P_z}, \frac{n_v}{P_v}, \frac{n_\mu}{P_\mu}, \frac{n_s}{P_s}\right)$ → 1D-FFT → Transpose → 1D-FFT → $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \left(\frac{n_x}{P_w}, n_y, \frac{n_z}{P_z}, \frac{n_v}{P_v}, \frac{n_\mu}{P_\mu}, \frac{n_s}{P_s}\right)$ → Nonlin. Adv. → FFT → Transpose → 1D-FFT
- New (**y2zm**): $\hat{f}_{\boldsymbol{k}} \left(n_x, \frac{n_y}{P_w}, \frac{n_z}{P_z}, \frac{n_v}{P_v}, \frac{n_\mu}{P_\mu}, \frac{n_s}{P_s}\right)$ → Transpose → $\hat{f}_{\boldsymbol{k}} \left(n_x, n_y, \frac{n_v}{P_v}, \frac{n_z n_\mu}{P_z P_\mu P_w}, \frac{n_s}{P_s}\right)$ → 2D-FFT → Nonlin. Adv. → 2D-FFT → Transpose

Pros: Data transpose is reduced. (**y2x**: 3 variables → **y2zm**: 2 variables with 3/2 dealiasing)
→ ~x1.8 speed up of nonlinear term (JFRS-1 192 node)

Cons: Wavenumber parallelization is limited up to $\frac{n_z n_\mu}{P_z P_\mu} > P_w$.
→ Switch **y2zm** to **y2x** for a large number of parallelization P_w.

* MPI_alltoall among $P_v P_\mu P_s$ in sub-communicators, not global commun.

Implicit collision solver [Maeyama'19CPC]

When considering Coulomb collision, velocity-dependent collision frequency ($\nu \propto 1/v^3$) severely restricts CFL.

<u>Our strategy</u>

Since collision is an integro-differential operator over $(v_\parallel, \mu, s)$,

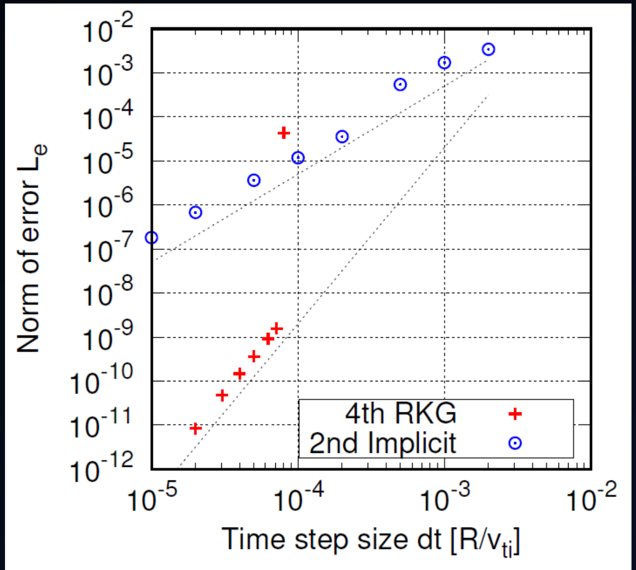
1. Data transpose by MPI_alltoall
$$f\left(n_x, \frac{n_y}{P_y}, \frac{n_z}{P_z}, \frac{n_v}{P_v}, \frac{n_\mu}{P_\mu}, \frac{n_s}{P_s}\right) = f\left(n_v, n_\mu, n_s, \frac{n_z}{P_z}, \frac{n_x n_y}{P_y P_v P_\mu P_s}\right)$$
2. Iterative implicit solver for $f(v_\parallel, \mu, s)$, independent to (x, y, z). (The iteration is MPI communication free!)
3. Transpose back again by MPI_alltoall

* MPI_alltoall among $P_v P_\mu P_s$ in sub-communicators, not global commun.

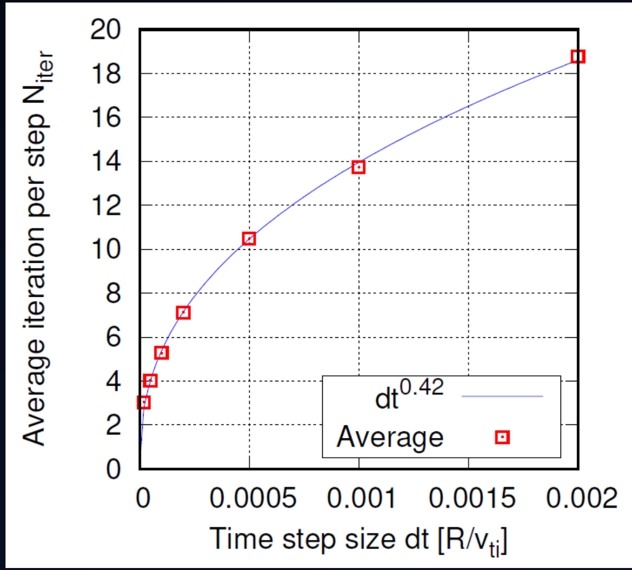
Implicit collision solver [Maeyama'19CPC]

- ✓ Implicit solver allows stable computation over larger time steps.
- ✓ Arithmetic intensity and computational performance are enhanced.
  – Promising for manycore processor.

Accuracy of the scheme



Average iteration num.



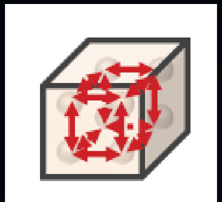
Performance for a nonlinear run on FX100 (1.8×10^{10} grids)

	4th Explicit	2nd Implicit
FLOPS (/PEAK)	8.67 TFLOPS (4.01%)	**25.78 TFLOPS (12.5%)**
Elapsed time per step	1.76 sec/step	3.13 sec/step
Time step size	$5 \times 10^{-5}\ R/v_{ti}$	$1 \times 10^{-3}\ R/v_{ti}$
Speed-up to solution	1	**10 times faster**

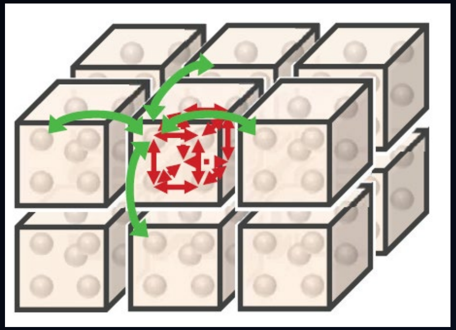
- Segmented MPI process mapping
- Pipelined computation-communication overlap

These techniques are available on Fugaku as well as K. [Maeyama’15PC]

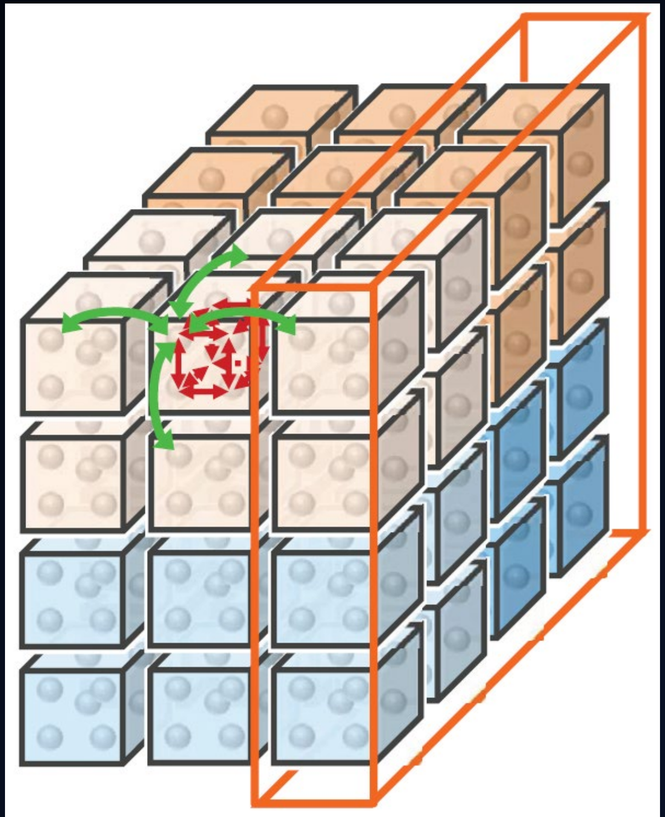
Process mapping on 3D torus network



① Arrange rank_w: Transpose is performed in a segment.

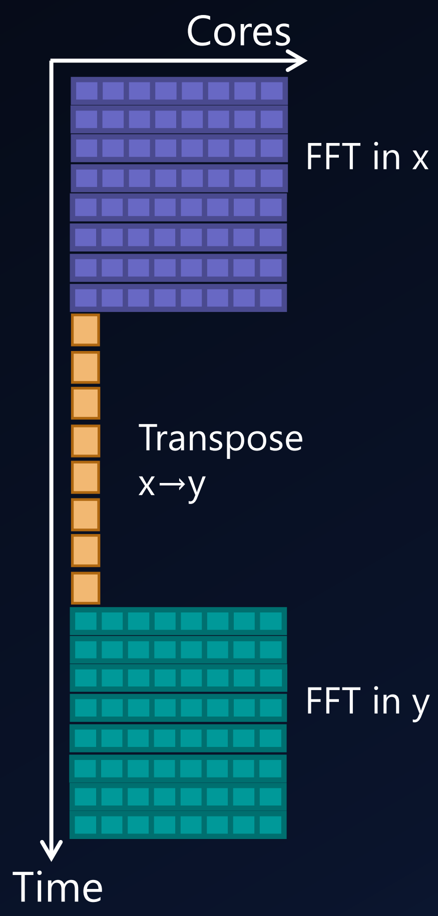


② Arrange rank_z,v,m: 1-to-1 comms. are performed between adjacent segments.

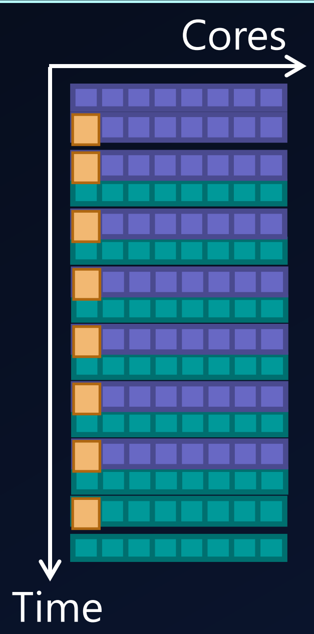


③ Arrange rank_s: Reduction is performed in a cross section.

2D-FFT w/o overlap



2D-FFT with overlap

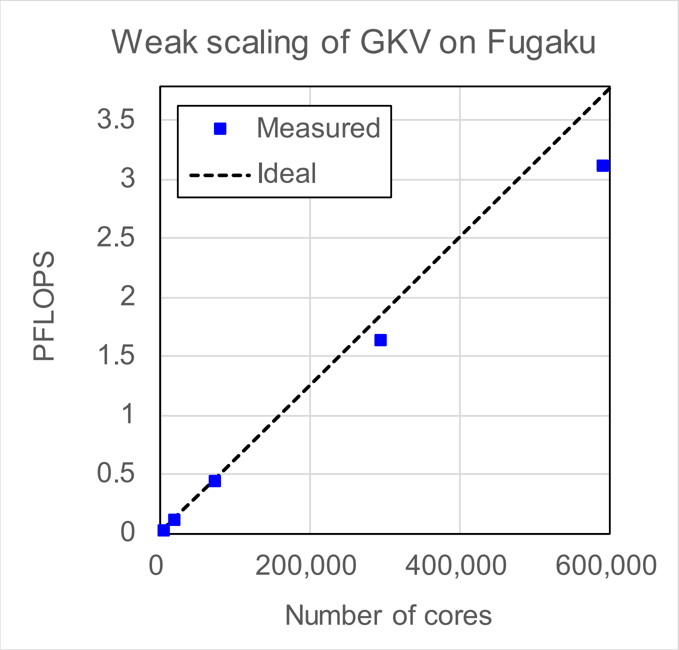


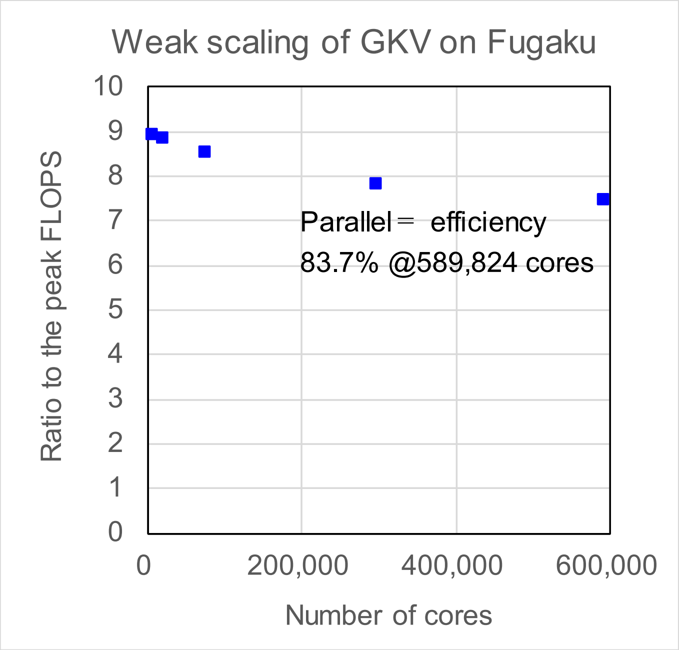
Weak scaling of GKV on Fugaku

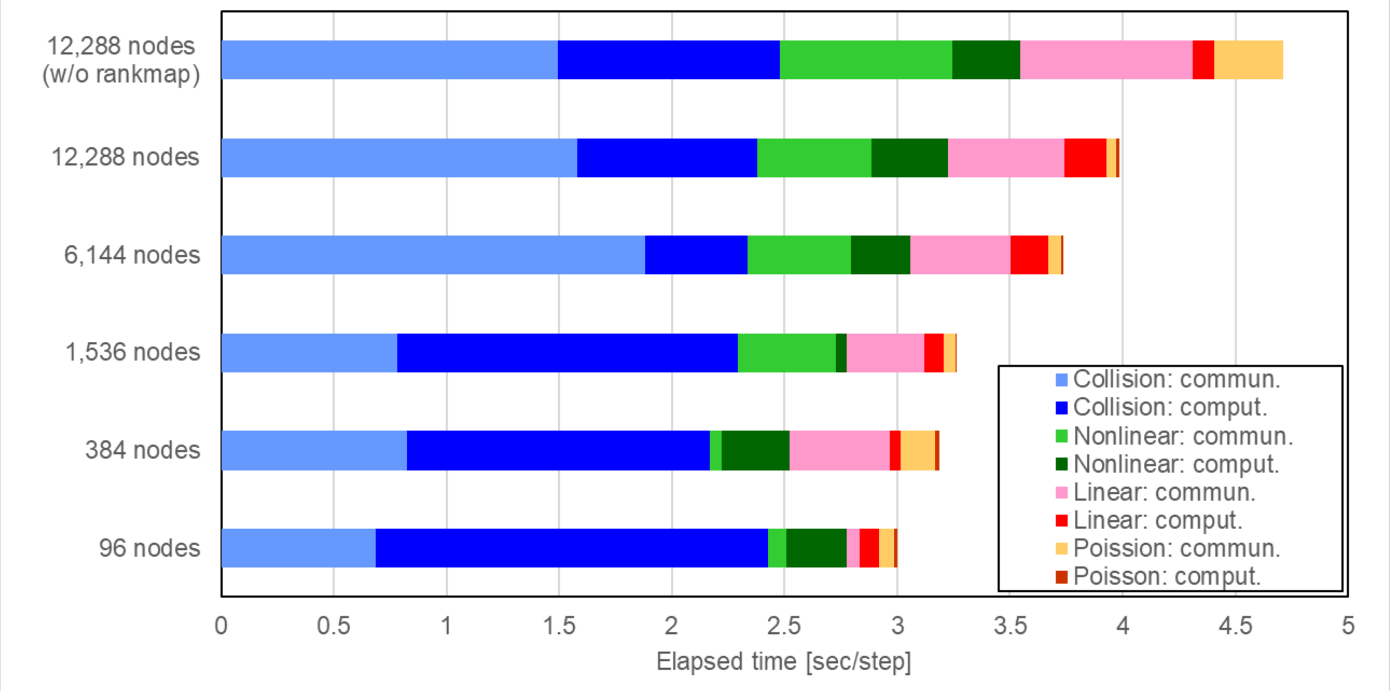
Excellent scaling up to target problem size ~1.24×10^{12} grids.

(Fugaku 12,288 nodes, 589,824 cores, 49,152MPI, 12OpenMP, rankmap 8x32x48)

- ✓ 3.1PFLOPS
- ✓ 7.5% to peak FLOPS
- ✓ Parallel efficiency 83.7%







<u>Bottleneck of performance degradation</u>

Communication costs of linear and nonlinear terms increase as grid and MPI number increases, which cannot be masked by related computations.

Outline

- Optimization of GKV code on the supercomputer Fugaku - 10 min
- **Multi-scale turbulence simulations toward burning plasma - 10 min**

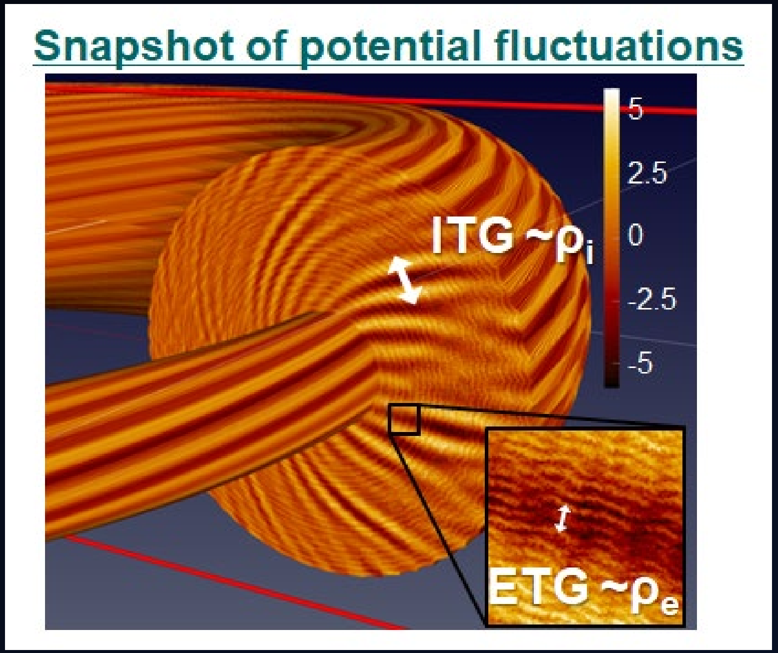
Background

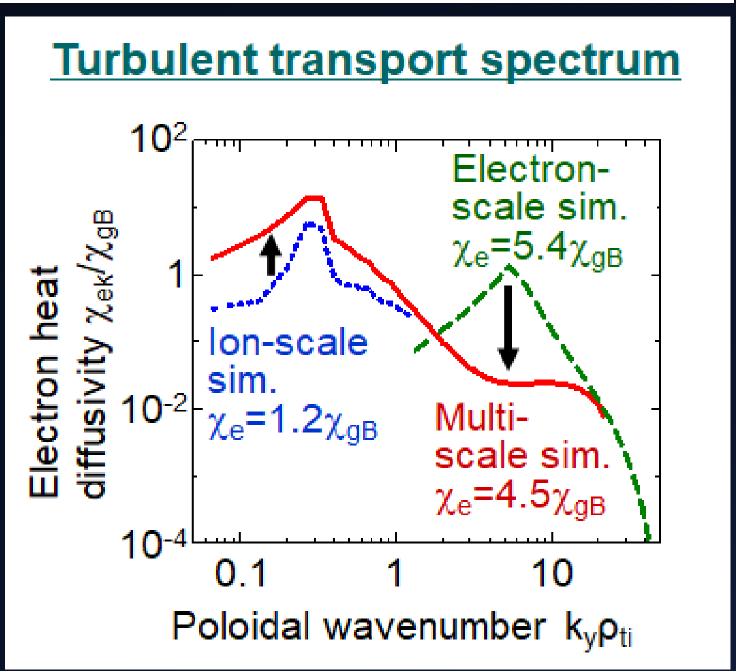
Recent gyrokinetic simulations reveal the importance of multi-scale interactions.

- Turbulent transport affected by electron scales [Maeyama’15PRL, Maeyama’17PRL]
- Multi-scale interactions are necessary to explain an experimental heat flux on Alcator C-Mod. [Howard’16NF]
- DIII-D results suggest the importance of multi-scale interactions in ITER [Holland’17NF]
- Recent multi-scale studies on JET [Bonanomi’18NF,Mantica’20PPCF,Mariani’21IAEA]

Our physical target on Fugaku: Extrapolation of multi-scale interactions toward burning plasma

- ✓ **High electron temperature (Te>Ti)**
- ✓ **Electron, Fuel (D,T), Ash (He) mixture**





TEM/ETG multi-scale turbulence simulation

- ✓ **Electron temperature gradient modes (ETG)** grow initially. After that **trapped electron modes (TEM)** appear.

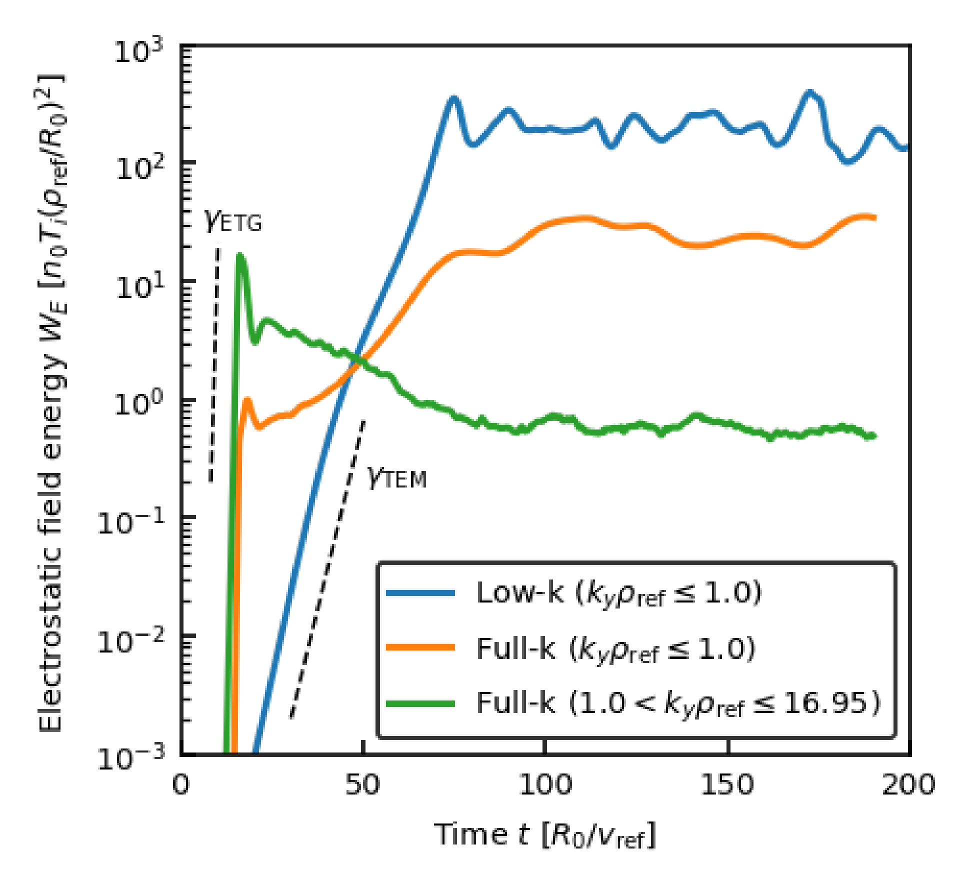
- ◆ TEM growth rate is reduced in the presence of ETG turbulence.

- ◆ TEM turbulence is suppressed in the presence of ETG turbulence.

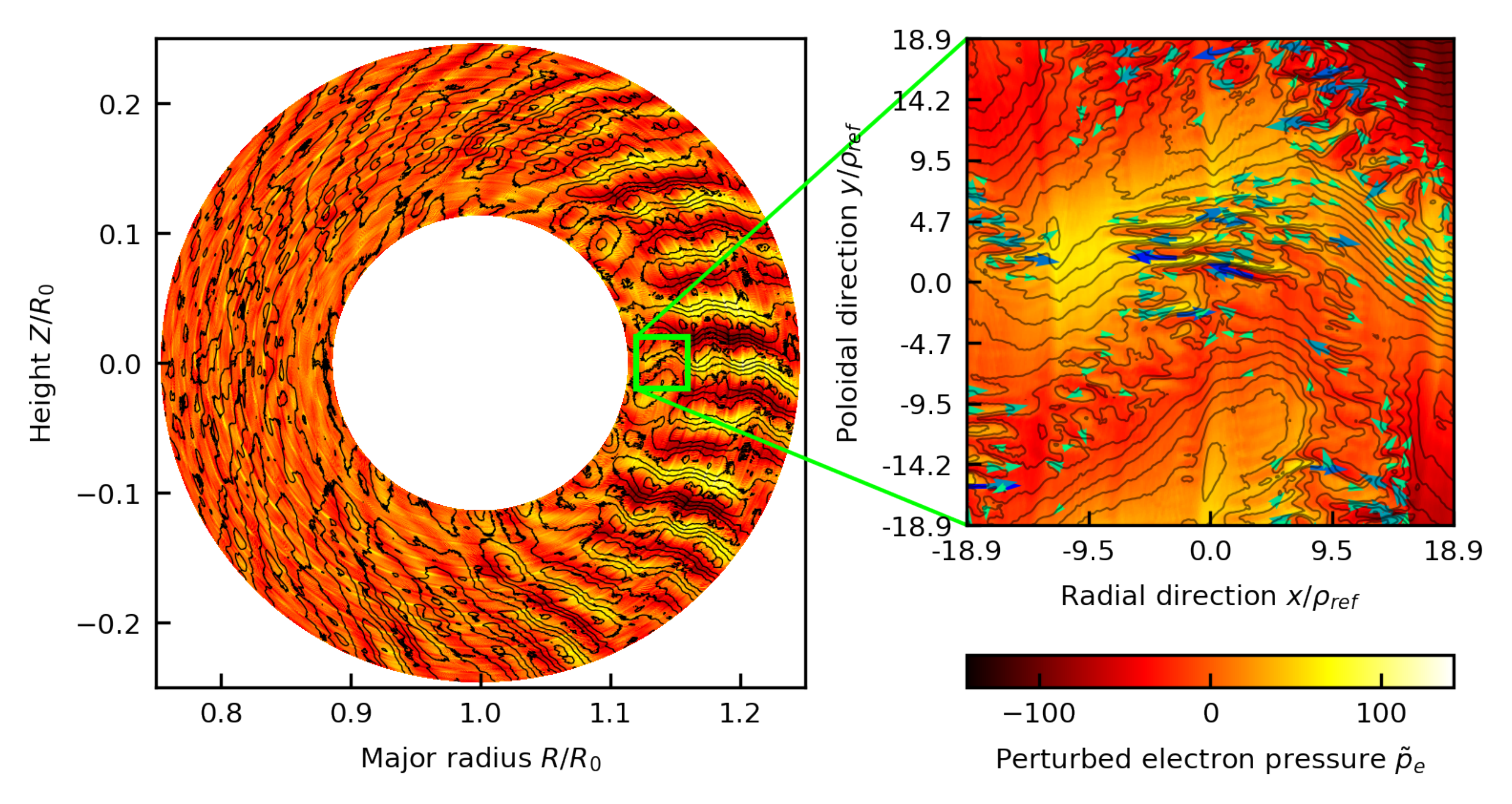
→ ETG stabilizes TEM.
→ This suggests the importance of multi-scale interactions even in burning plasma.

Time evolution of electrostatic energy

High-res. multi-scale simulation (Full-k) and low-res. ion-scale simulation (Low-k) are compared.



Color map of perturbed electron pressure and streamlines of turbulent ExB flows



ETGs coexist with TEMs.

→ Small-scale ETG turbulence dissipatively stabilizes large-scale TEM.

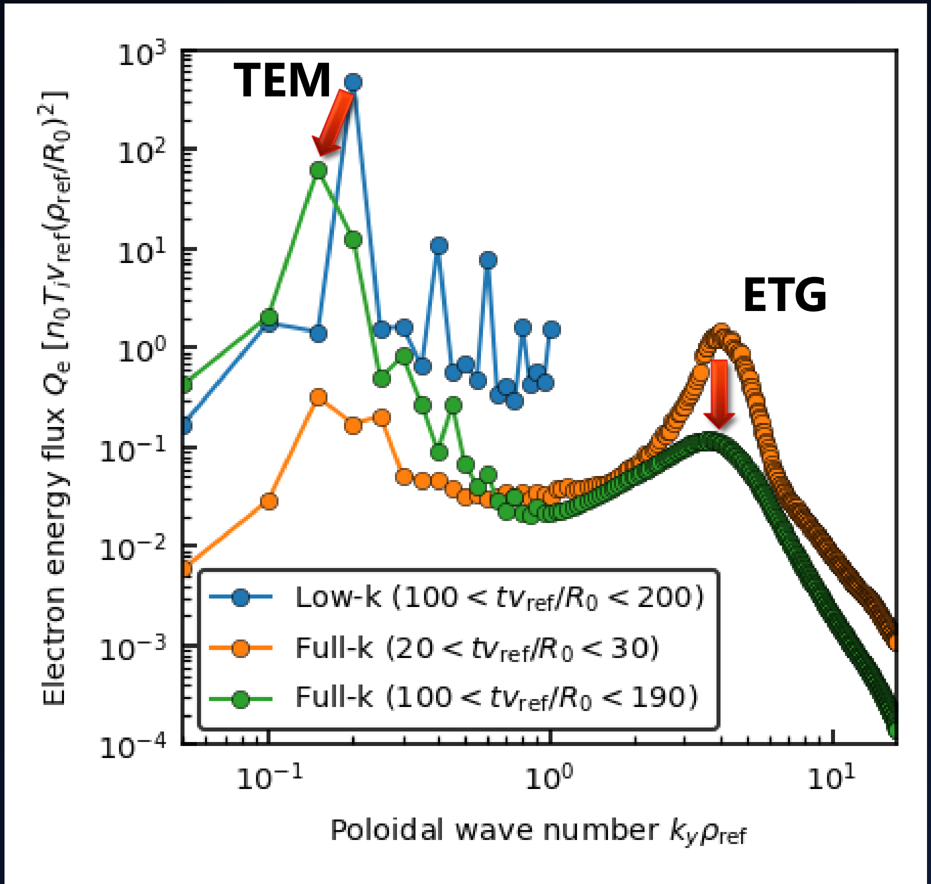
Impacts of turbulent transport

- ✓ **Large-to-small interactions: ETG peak is suppressed after TEM growth.**
- ✓ **Small-to-large interactions: TEM amplitude is also reduced in the presence of ETG.**

Comparison of heat flux [gyro-Bohm unit]
Electron : Q_e=524 (low-k) → 88 (Full-k)
Fuel : Q_D+Q_T=17 (Low-k) → 3.7 (Full-k)
Ash : Q_{He}=1.4 (low-k) → 0.3 (Full-k)
→ **This result firstly demonstrates the possibility of reduction of Qe by cross-scale interactions.**
→ It affects not only electrons but also fuel ions and helium ash.

Wavenumber spectra of electron energy flux

High-res. multi-scale simulation (Full-k) and low-res. ion-scale simulation (Low-k) are compared.



Velocity-space structures

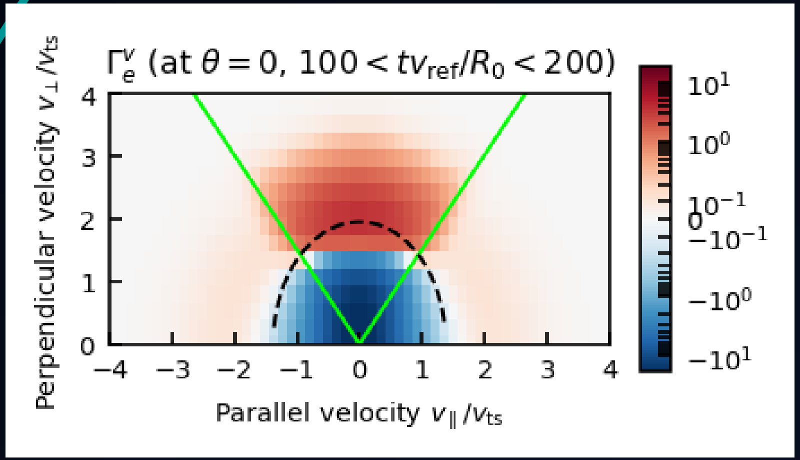
- ✓ Velocity-space dependent turbulent flux

$$\Gamma_e^v(z, v_\parallel, \mu) = \int_0^{L_x} \frac{dx}{L_x} \int_0^{L_y} \frac{dy}{L_y} \left(\frac{-\nabla J_0 \tilde{\phi} \times \boldsymbol{b}}{B_0} \cdot \nabla x \right) \tilde{f}_e$$

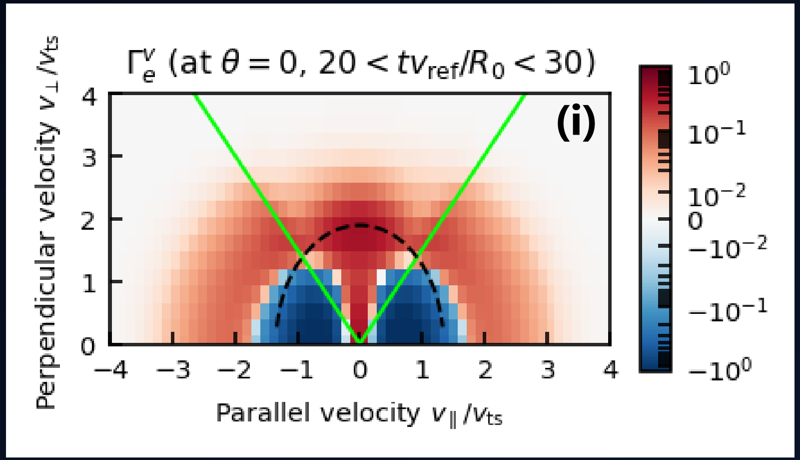
Cf.
Particle flux $\Gamma_e = \left\langle \int_{-\infty}^{\infty} dv_\parallel \int_0^{\infty} dv_\perp \, 2\pi v_\perp \Gamma_e^v(z, v_\parallel, \mu) \right\rangle$
Energy flux $Q_e = \left\langle \int_{-\infty}^{\infty} dv_\parallel \int_0^{\infty} dv_\perp \, 2\pi v_\perp \frac{m_e v^2}{2} \Gamma_e^v(z, v_\parallel, \mu) \right\rangle$

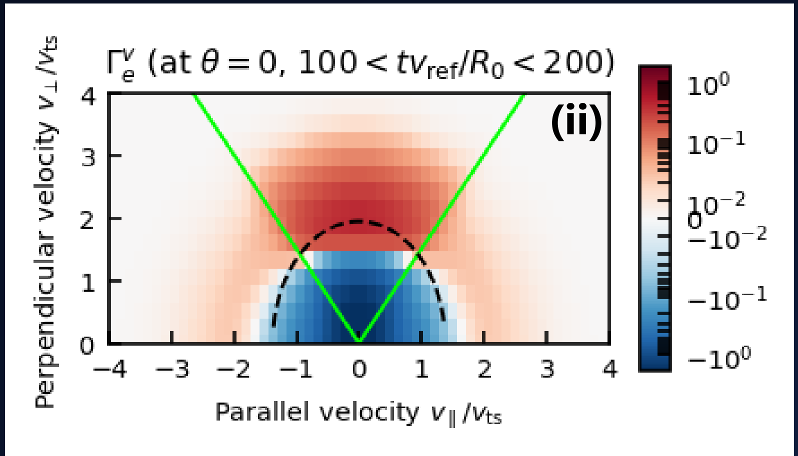
- ✓ Isotropic dependence by magnetic drift resonance of toroidal ETG. [black dotted line]
- ✓ Precession drift resonance by trapped electron [between green lines] dominates in TEM.

Low-res. ion-scale sim. (Low-k)



High-res. multi-scale sim. (Full-k) (i) initial ETG, (ii) TEM/ETG saturated.





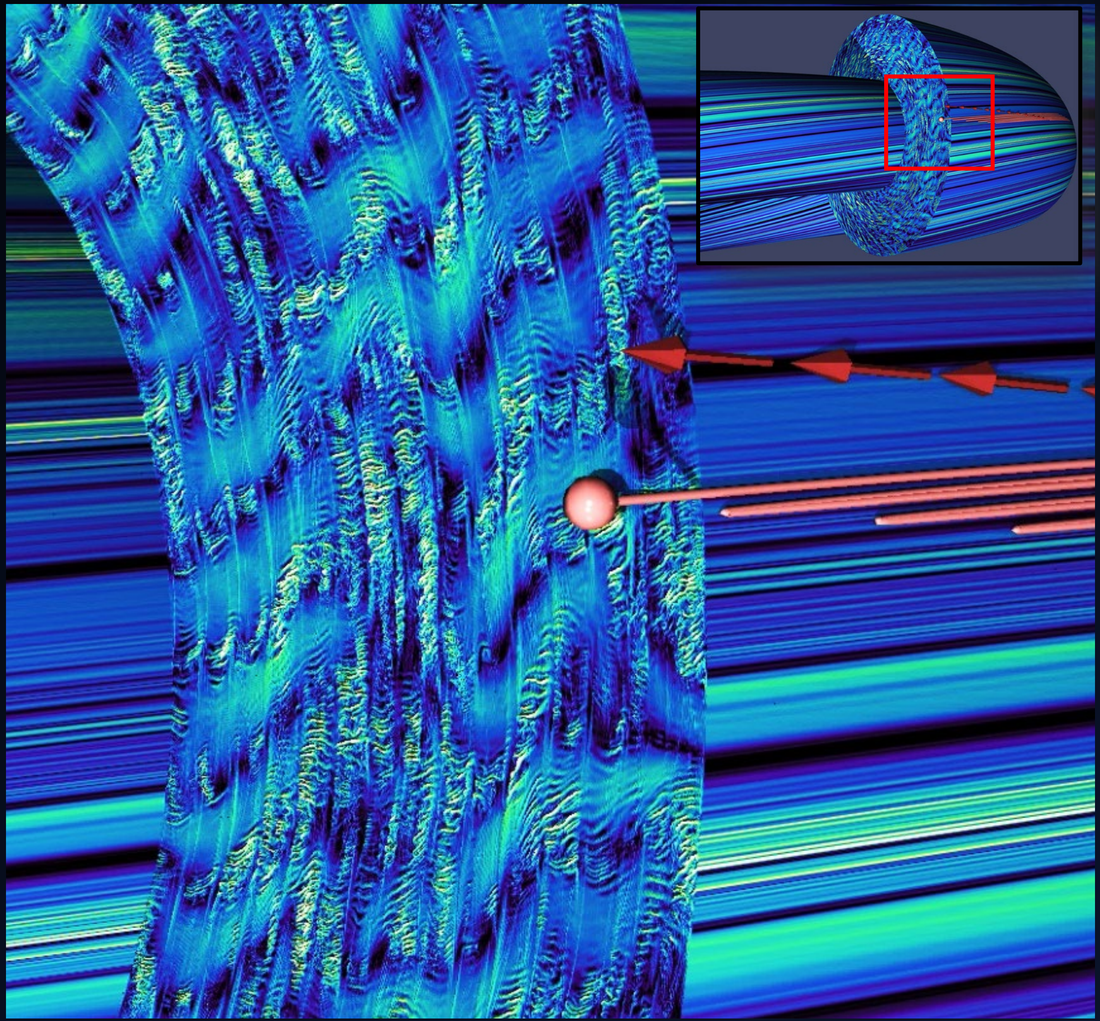
Schematic explanation of TEM/ETG cross-scale interactions

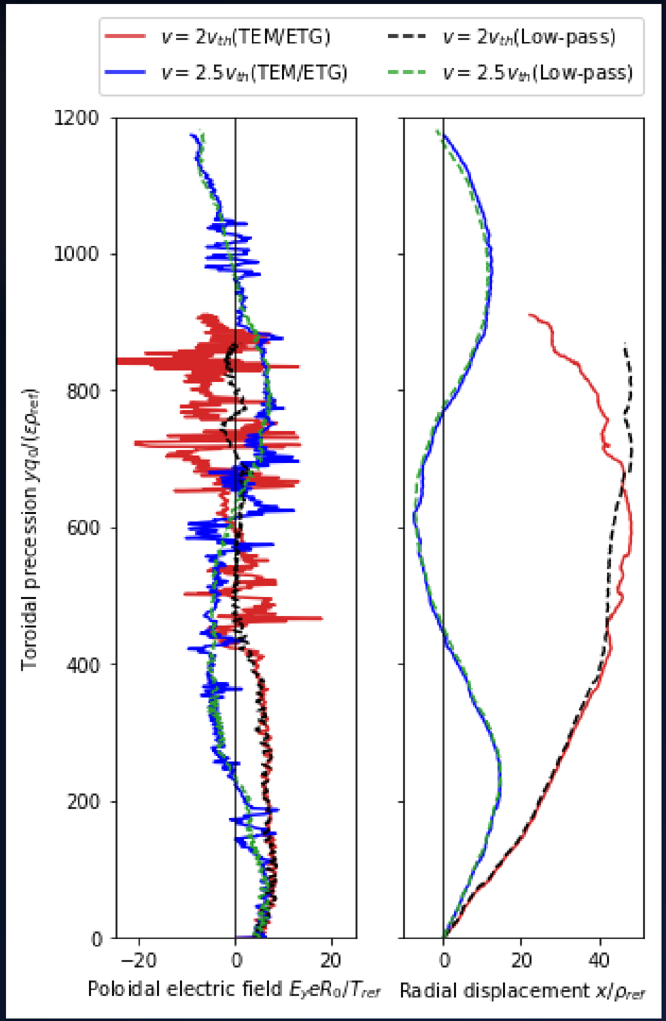
Small-to-large interactions:

✓ ETG turbulence distorts trapped electron trajectory, and reduces the precession drift resonance of TEM.

Large-to-small interactions:

✓ TEM turbulent eddies shears ETG streamers. (even when no strong zonal flows [Asahi'14PoP])





Summary

HPC

- ✓ Physical applicability of GKV is extended along with HPC.
- ✓ High Memory-Byte/FLOP=0.30 but low Interconnect Byte/FLOP=0.012 on Fugaku.
  → Reduce communication cost of GKV by mapping, overlap, improved implementation.
- ✓ GKV achieves excellent scaling and 3.1 PFLOPS (7.5% to peak) on Fugaku 12,288 nodes

Physics

- ✓ Multi-scale turbulence simulation toward burning plasma (High $T_e > T_i$, mixture of e,D,T,He)
- ✓ Cross-scale interactions changes turbulent spectrum, which affect turbulent transport levels of not only electron but also fuel D,T and He ash.
- ✓ The result firstly demonstrated the possibility of reduction of Q_e by cross-scale interactions.
- ✓ Large-to-small interactions: TEM turbulent eddies suppress ETG streamers
- ✓ Small-to-large interactions: ETG stabilizes TEM by disturbing precession drift resonance
  - Analogy 1: damping of short-wave-length zonal flow by ETG [Maeyama'15PRL; Maeyama'17NF]
  - Analogy 2: destruction of MTM current sheet by ETG [Maeyama'17PRL]
- ✓ Mutual exclusive nature between disparate-scale turbulence