Cosmic ray propagation code based on magnetohydrodynamic simulation

Tohoku University

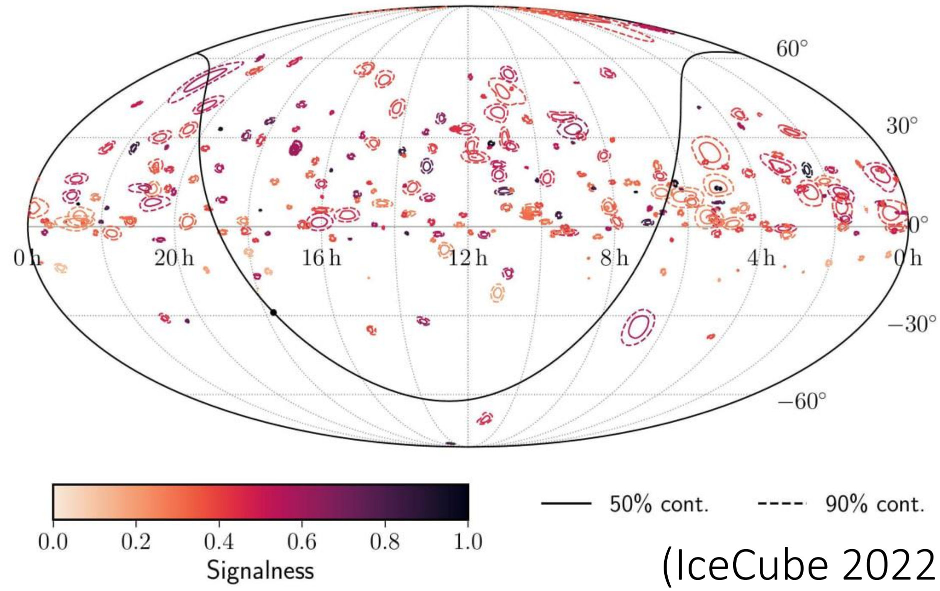
Wataru Ishizaki

Collaborate with

Shigeo Kimura & Kazumi Kashiyama

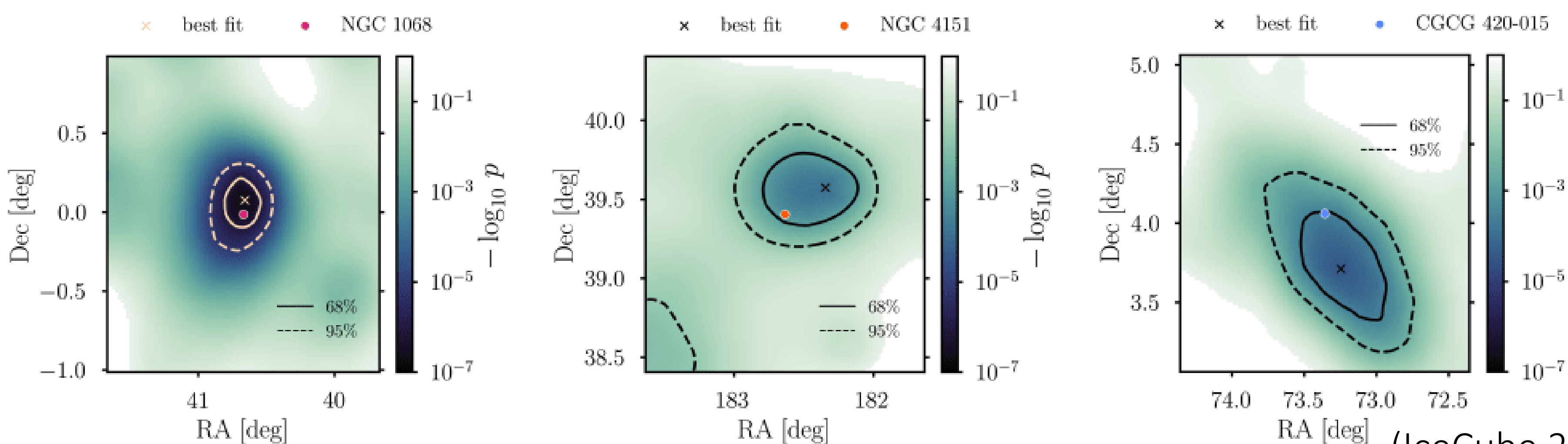
Astrophysical Neutrinos

- Astrophysical neutrinos
  - IceCube has been detected astrophysical neutrinos
  - Not only background diffuse emission, but also candidates of point sources
  - Signs of neutrino signals from several Seyfert galaxies have been claimed!



(IceCube 2022)

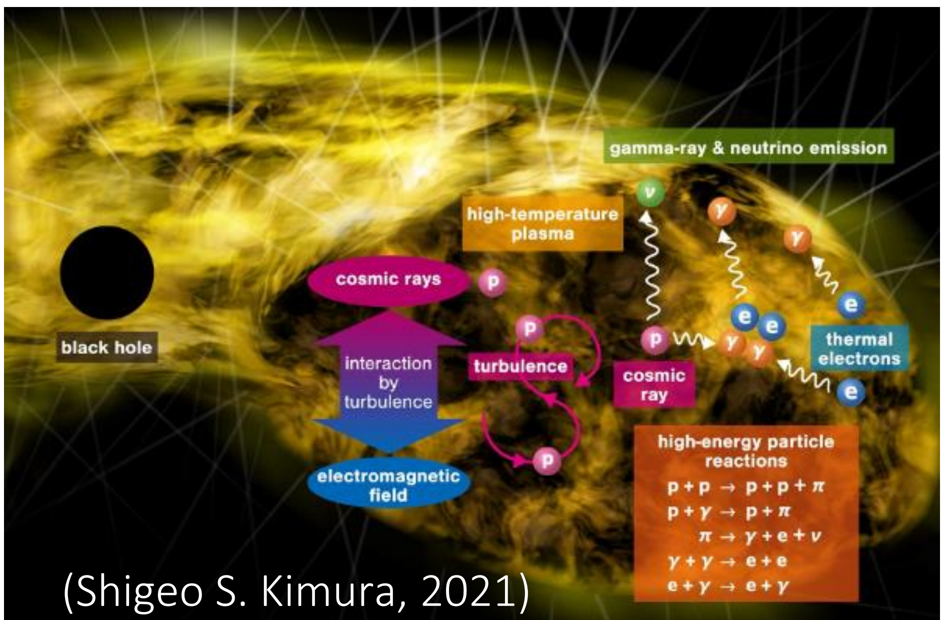
→Astrophysical (1.0)



(IceCube 2022)

What we want to find out

- Promising source?
  - Should be not very bright in gamma-rays but make a lot of cosmic rays
  - Low-density accretion disk with magnetic turbulence around an AGN
  - Particle acceleration and associated neutrino radiation via photo-meson processes



(Shigeo S. Kimura, 2021)

What we want to do and to find out

"Beyond one-zone"

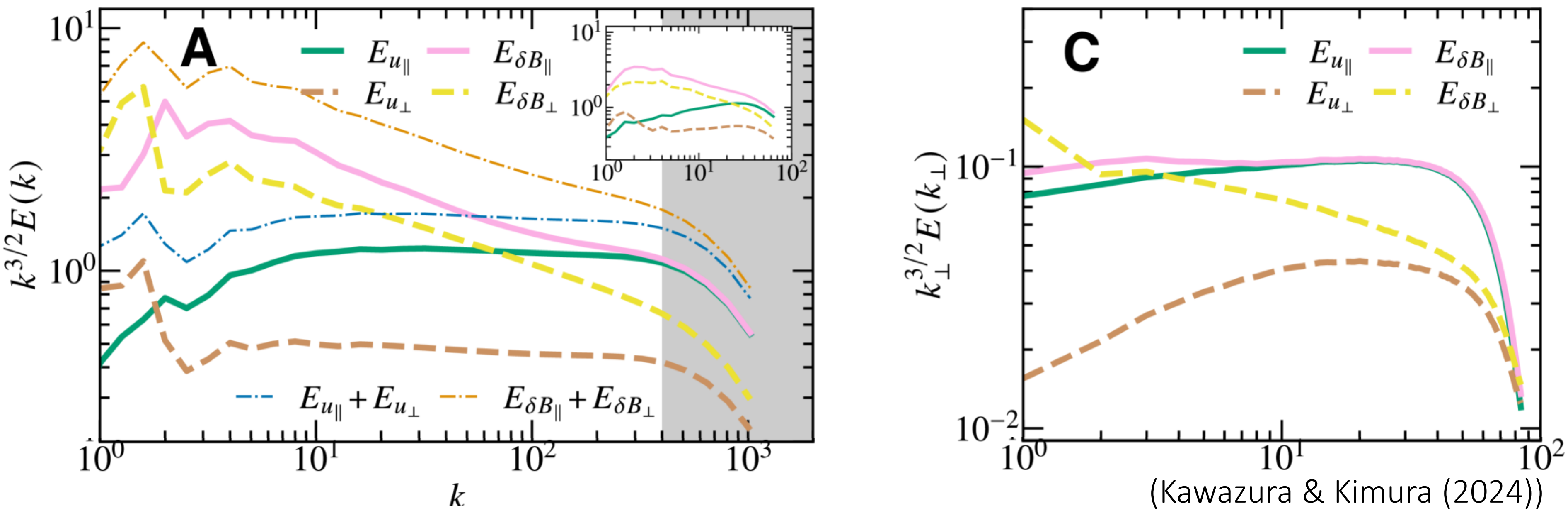
- Where are injected particles accelerated?
- Where do particles accelerated?
- What properties of the accretion disk do cosmic ray/neutrino spectra reflect?

⇨ **Calculate the acceleration and propagation of cosmic rays based on the structure of the disk obtained from MHD calculations!**

What we want to find out ("Ambition" part)

- Spectrum of Magneto-Rotational Instability (MRI) turbulence
    - MRI in accretion disks has a broad injection scale → inertial range has not resolved
    - Kawazura & Kimura (2024): First time ever to resolve from MHD scale to inertial range
    - At much smaller scales, properties are revealed by reduced MHD (Kawazura et al. 2022)
    - Ready to model turbulence from dissipation to MHD scales with a consistent theory!

Aim to solve "acceleration from supra-thermal to ultra-high energy cosmic rays" in accretion disks!



(Kawazura & Kimura (2024))

Basic equation and methodology

- **Fokker-Planck equation**: describing cosmic ray propagation and acceleration

$$\frac{\partial f}{\partial t} = \nabla \cdot (\boldsymbol{\kappa} \cdot \nabla f - \mathbf{V} f) + \frac{1}{p^2}\frac{\partial}{\partial p}\left[p^2 \left(\frac{1}{3}(\nabla \cdot \mathbf{V})p + \Lambda_{\mathrm{loss}}\right) f\right] + \frac{1}{p^2}\frac{\partial}{\partial p}\left(D_{pp} p^2 \frac{\partial f}{\partial p}\right)$$

This work→ Development of a post-process code to solve this equation based on MHD simulation in 3D (spatial) + 1D (energy)

Basic equation and methodology

- **Fokker-Planck equation**: describing cosmic ray propagation and acceleration

Advection: velocity field (MHD sim.)

Adiabatic heating/loss: velocity field (MHD sim.)

$$\frac{\partial f}{\partial t} = \nabla \cdot (\boldsymbol{\kappa} \cdot \nabla f - \mathbf{V} f) + \frac{1}{p^2}\frac{\partial}{\partial p}\left[p^2\left(\frac{1}{3}(\nabla \cdot \mathbf{V})p + \Lambda_{\rm loss}\right) f\right] + \frac{1}{p^2}\frac{\partial}{\partial p}\left(D_{pp} p^2 \frac{\partial f}{\partial p}\right)$$

3D MHD simulation → **Fokker-Planck Solver** → **Observable (γ, neutrino, …)**

This work → Development of a post-process code to solve this equation based on MHD simulation in 3D (spatial) + 1D (energy)

Basic equation and methodology

- **Fokker-Planck equation**: describing cosmic ray propagation and acceleration

Advection: velocity field (MHD sim.)

Adiabatic heating/loss

$$\frac{\partial f}{\partial t} = \nabla \cdot (\boldsymbol{\kappa} \cdot \nabla f - \mathbf{V} f) + \frac{1}{p^2}\frac{\partial}{\partial p}\left[p^2 \left(\frac{1}{3}(\nabla \cdot \mathbf{V})p + \Lambda_{\rm loss}\right) f\right] + \frac{1}{p^2}\frac{\partial}{\partial p}\left(D_{pp} p^2 \frac{\partial f}{\partial p}\right)$$

Cooling: appropriate processes
(e.g., photomeson production)

3D MHD simulation → **Fokker-Planck Solver** → **Observable (γ, neutrino, …)**

This work → Development of a post-process code to solve this equation based on MHD simulation in 3D (spatial) + 1D (energy)

Basic equation and methodology

- **Fokker-Planck equation**: describing cosmic ray propagation and acceleration

Advection: velocity field (MHD sim.)

Adiabatic heating/loss

Momentum Diffusion: modeled from MHD simulation

$$\frac{\partial f}{\partial t} = \nabla \cdot (\boldsymbol{\kappa} \cdot \nabla f - \mathbf{V} f) + \frac{1}{p^2}\frac{\partial}{\partial p}\left[p^2\left(\frac{1}{3}(\nabla \cdot \mathbf{V})p + \Lambda_{\rm loss}\right)f\right] + \frac{1}{p^2}\frac{\partial}{\partial p}\left(D_{pp}p^2\frac{\partial f}{\partial p}\right)$$

Diffusion: modeled from MHD simulation

Cooling: appropriate processes (e.g., photomeson production)

3D MHD simulation → **Fokker-Planck Solver** → **Observable (γ, neutrino, …)**

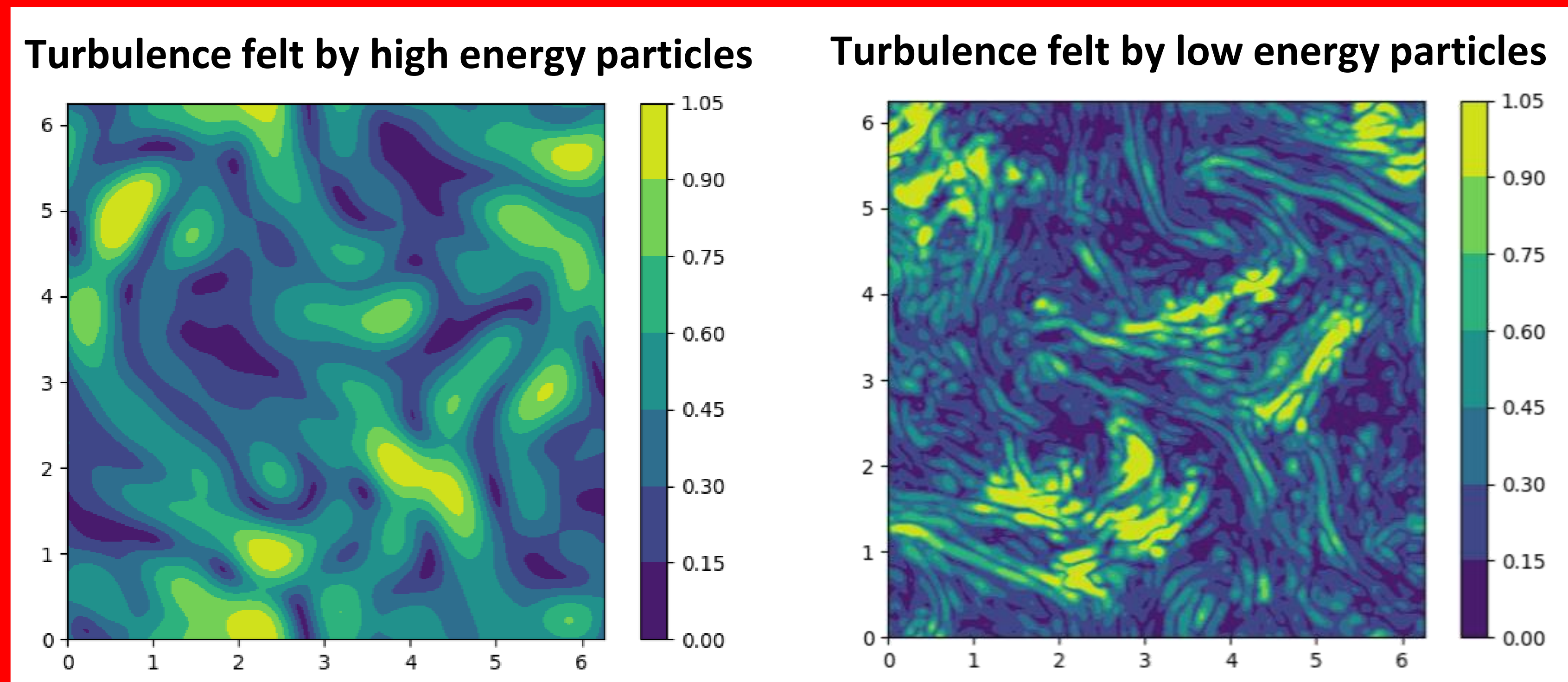
This work → Development of a post-process code to solve this equation based on MHD simulation in 3D (spatial) + 1D (energy)

Basic equation and methodology

$$r_c = \frac{E}{eB}$$

- **Fokker-Planck equation**: describing cosmic ray propagation and acceleration

Momentum Diffusion: modeled from MHD simulation

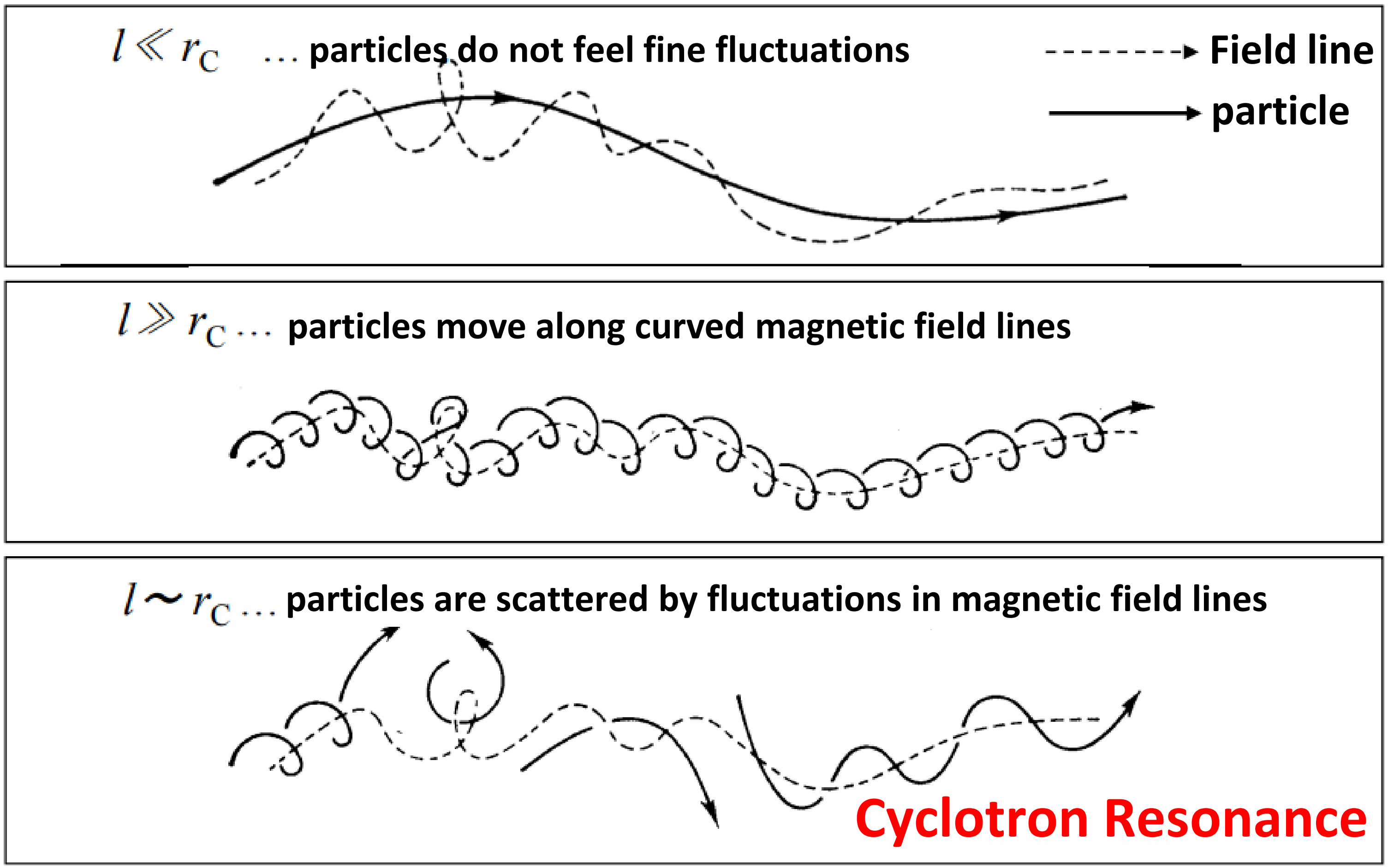


Y. Kawazura (see also Kawazura & Kimura (2024))

This work→ Development of a post-process code to solve this equation based on MHD simulation in 3D (spatial) + 1D (energy)

Interaction between charged particles and turbulence

r_c (gyro radius) ⟷ l : wavelengths of magnetic field disturbance



Terasawa-san's slide translated and modified by WI (2011/7/29　タウンミーティング「地下素粒子実験・宇宙観測」)

Interaction between charged particles and turbulence

- Efficient scattering of particles by waves (cyclotron resonance)
- Wave-Particle interaction: Energy transfer between waves and particles

⇒ causes both spatial diffusion and <u>momentum diffusion</u> of particles!

= "Fermi acceleration"

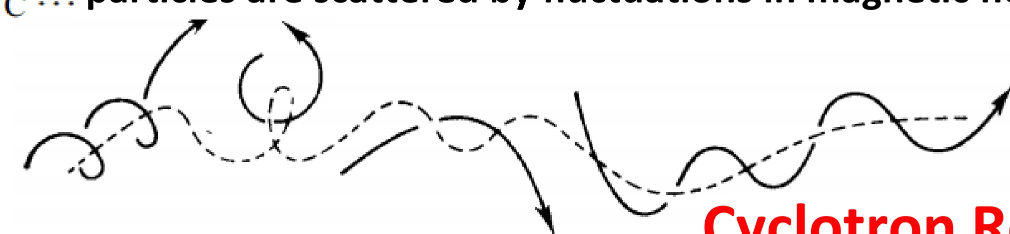
To calculate the acceleration of charged particles, we need…

to model a turbulence → Based on MHD simulations!

to calculate propagation of particles consistent with turbulence model

→ Solve Fokker-Planck equation w/ MHD turbulence in 3+1D

$l \sim r_C$ … **particles are scattered by fluctuations in magnetic field lines**



Cyclotron Resonance

Terasawa-san's slide translated and modified by WI (2011/7/29 タウンミーティング「地下素粒子実験・宇宙観測」)

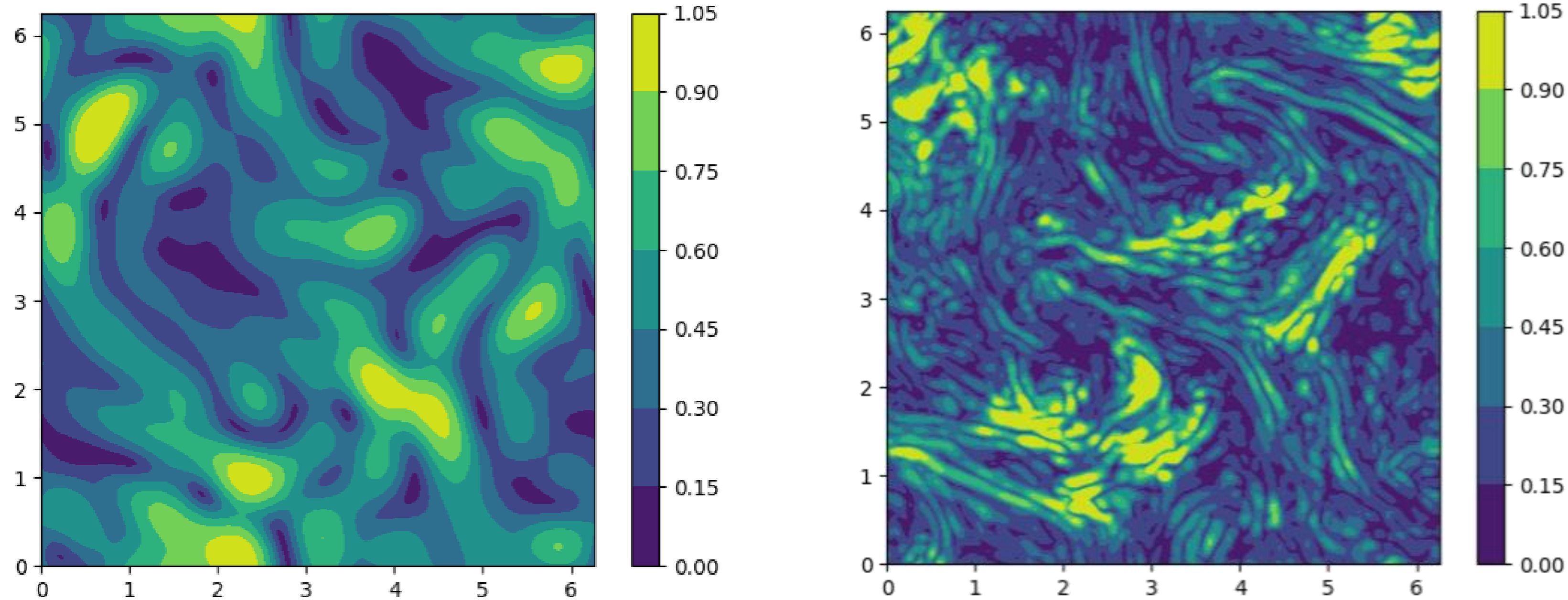
Basic equation and methodology

$$r_c = \frac{E}{eB}$$

- **Fokker-Planck equation**: describing cosmic ray propagation and acceleration

Momentum Diffusion: modeled from MHD simulation

Turbulence felt by high energy particles | **Turbulence felt by low energy particles**



Y. Kawazura (see also Kawazura & Kimura (2024))

This work→ Development of a post-process code to solve this equation based on MHD simulation in 3D (spatial) + 1D (energy)

Basic equation and methodology

- **Fokker-Planck equation**: describing cosmic ray propagation and acceleration

Advection: velocity field (MHD sim.)

Momentum Diffusion: modeled from MHD simulation

Adiabatic heating/loss

$$\frac{\partial f}{\partial t} = \nabla \cdot (\boldsymbol{\kappa} \cdot \nabla f - \mathbf{V} f) + \frac{1}{p^2}\frac{\partial}{\partial p}\left[p^2\left(\frac{1}{3}(\nabla \cdot \mathbf{V})p + \Lambda_{\mathrm{loss}}\right)f\right] + \frac{1}{p^2}\frac{\partial}{\partial p}\left(D_{pp}p^2\frac{\partial f}{\partial p}\right)$$

Diffusion: modeled from MHD simulation

Cooling: appropriate processes
(e.g., photomeson production)

3D MHD simulation → **Fokker-Planck Solver** → **Observable (γ, neutrino, ...)**

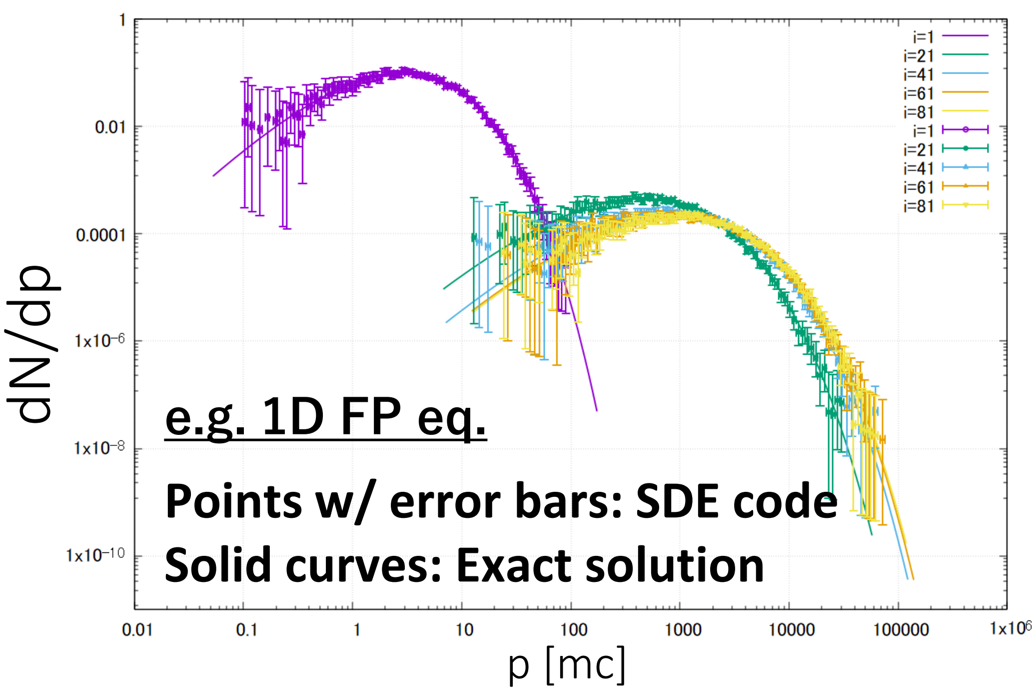
This work→ Development of a post-process code to solve this equation based on MHD simulation in 3D (spatial) + 1D (energy)

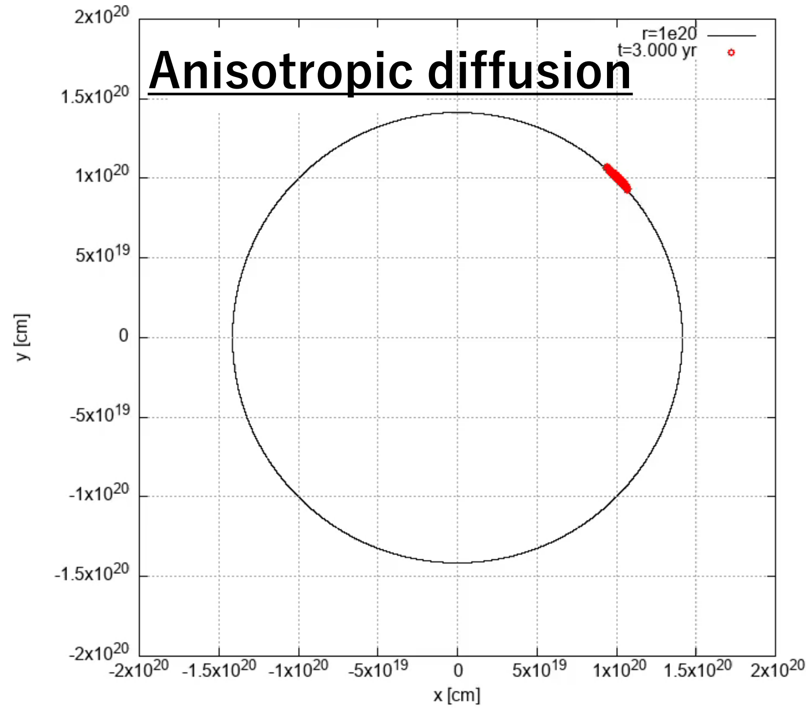
Method & Current status

- Stochastic Differential Equation method (SDE method)
  - Convert partial differential eq. into “many ordinary differential eqs. w/ stochastic terms”
  - Parallelization ◎, Multi-D/species ◎, Stability ◎, Accuracy △

    (∵ Accuracy is determined by statistics→ Can be covered by an efficient computation!)

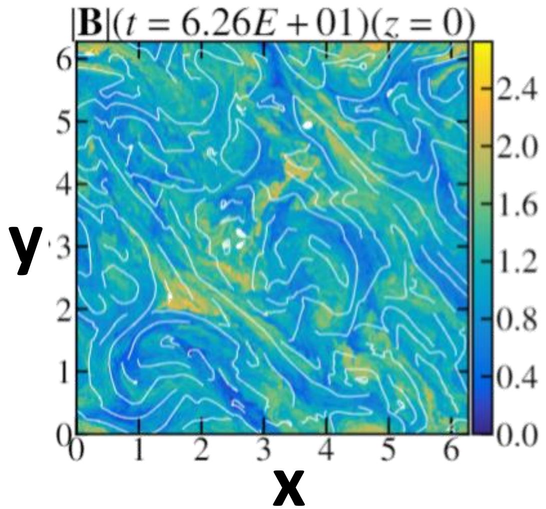
- Current status: Fokker-Planck equation solver part completed!

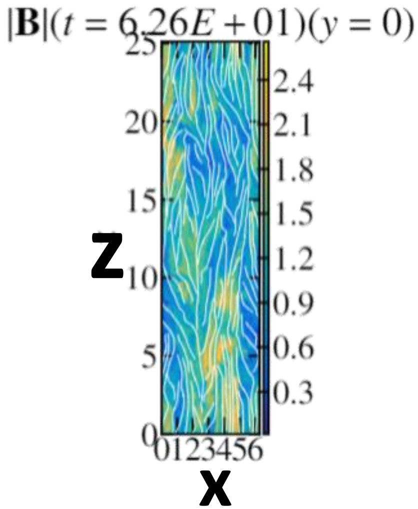




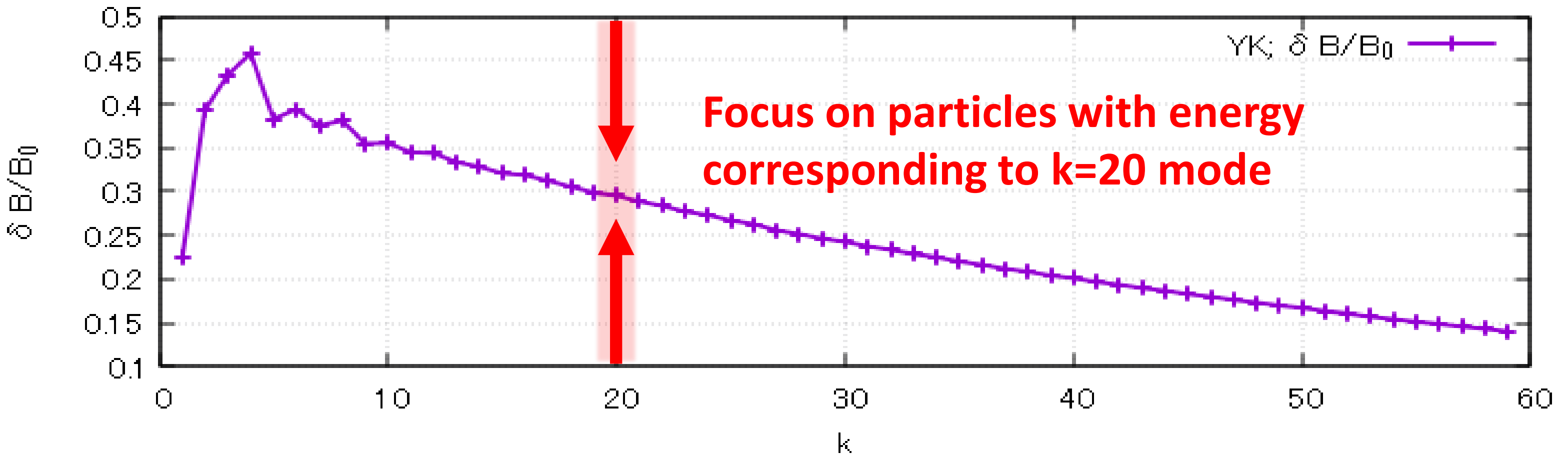
Test Calculation: MHD turbulence

- MHD turbulence (incompressive) (by Y. Kawazura, see also Kawazura & Kimura (2024))
  - Calculate spatial diffusion with MHD simulation data of artificially excited turbulence

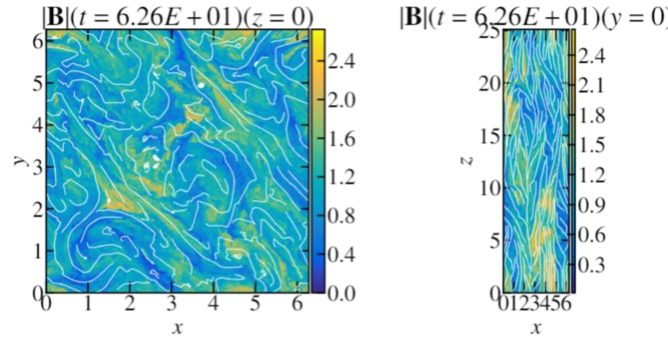




- 3D calc. (256x256x256)
- B-field is roughly in the z direction
- Periodic boundary
- Velocity field is ignored (for simplicity)



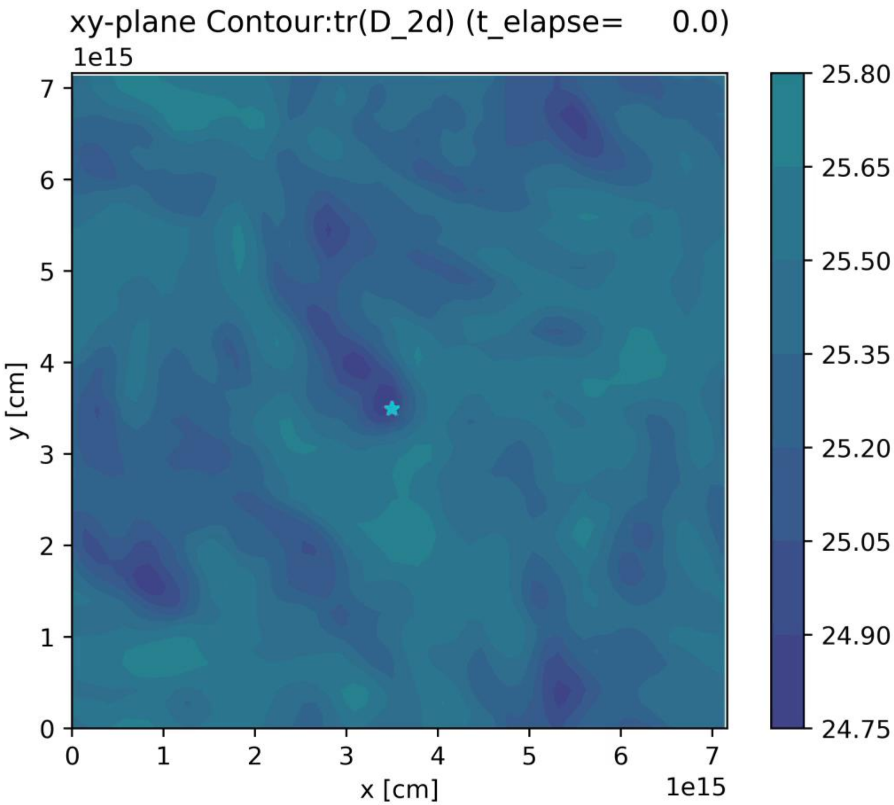
Test Calculation: MHD turbulence

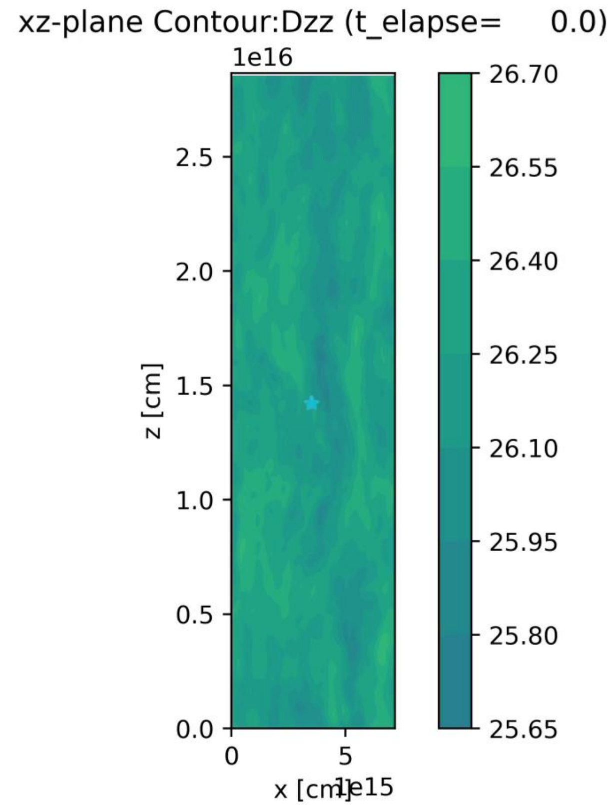


- Model:
  - Quasi-Linear Theory (QLT) + Alfvenic turbulence (e.g., Blandford & Eichler (1987))

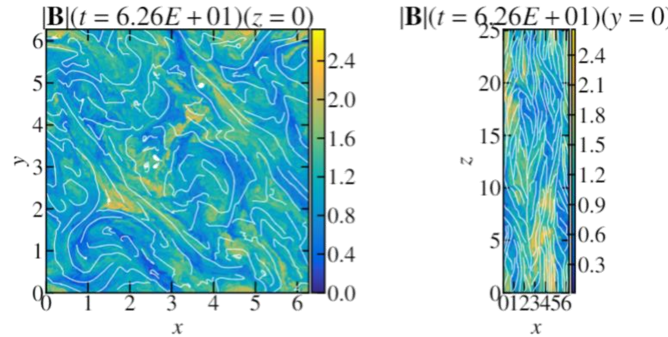
$$D_{\parallel} = \frac{1}{3} r_{\rm L} c \left(\frac{\delta B}{B_0} \right)^{-2}, \quad D_{\perp} = D_{\parallel} \left(\frac{\delta B}{B_0} \right)^{4}$$

Gyro radius: $r_{\rm L} = \frac{E}{eB_0}$





Test Calculation: MHD turbulence



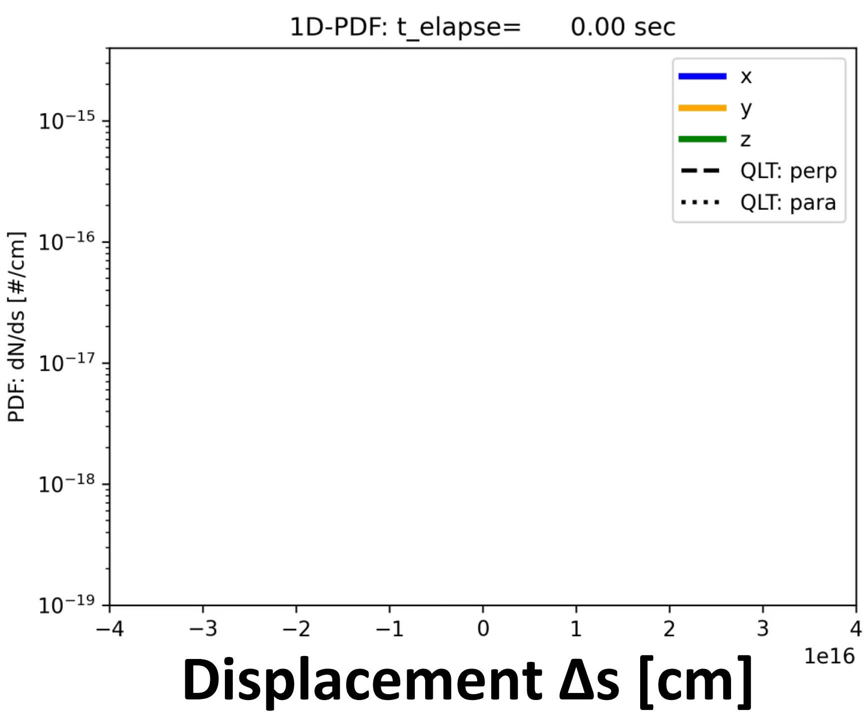
- Model:
  - Quasi-Linear Theory (QLT) + Alfvenic turbulence (e.g., Blandford & Eichler (1987))

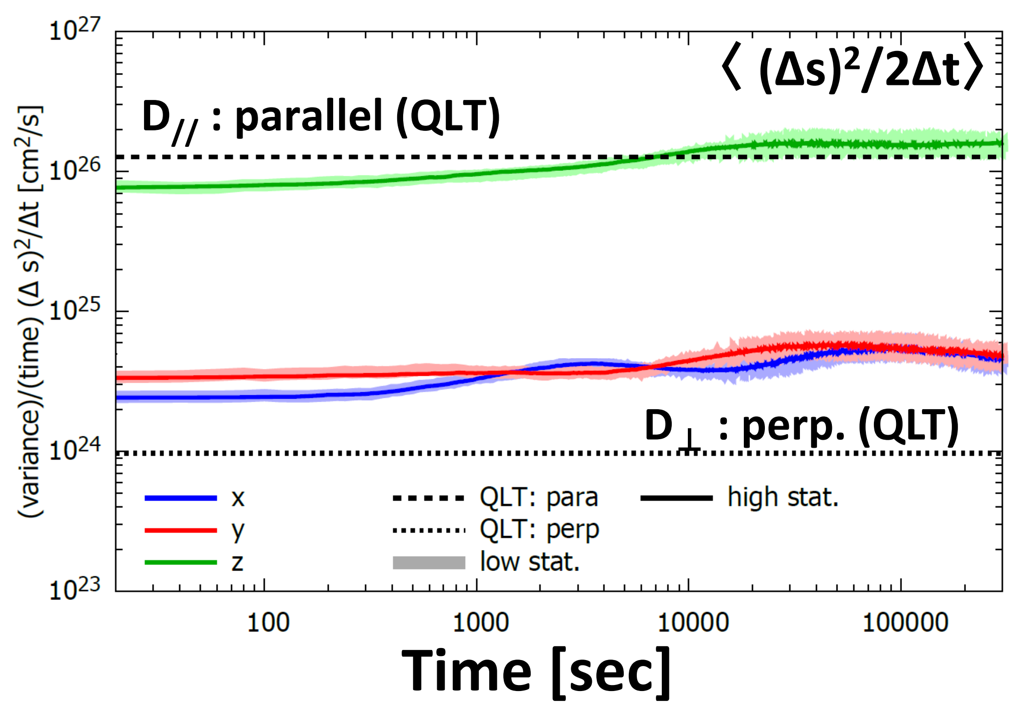
$$D_{\parallel} = \frac{1}{3} r_{\rm L} c \left(\frac{\delta B}{B_0}\right)^{-2}, \quad D_{\perp} = D_{\parallel}\left(\frac{\delta B}{B_0}\right)^{4}$$

Gyro radius: $r_{\rm L} = \frac{E}{eB_0}$

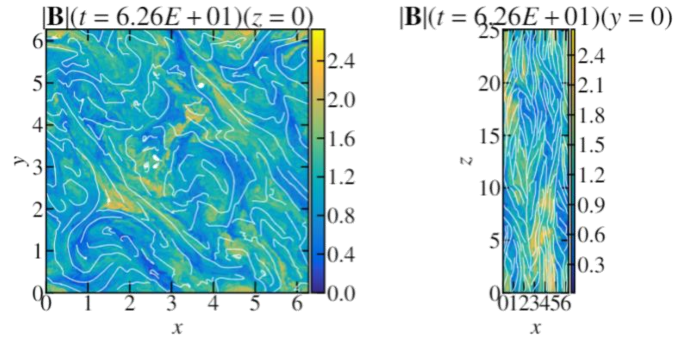
Parallel (z): consistent with QLT

Perpendicular (x,y): much faster than QLT…





Test Calculation: MHD turbulence



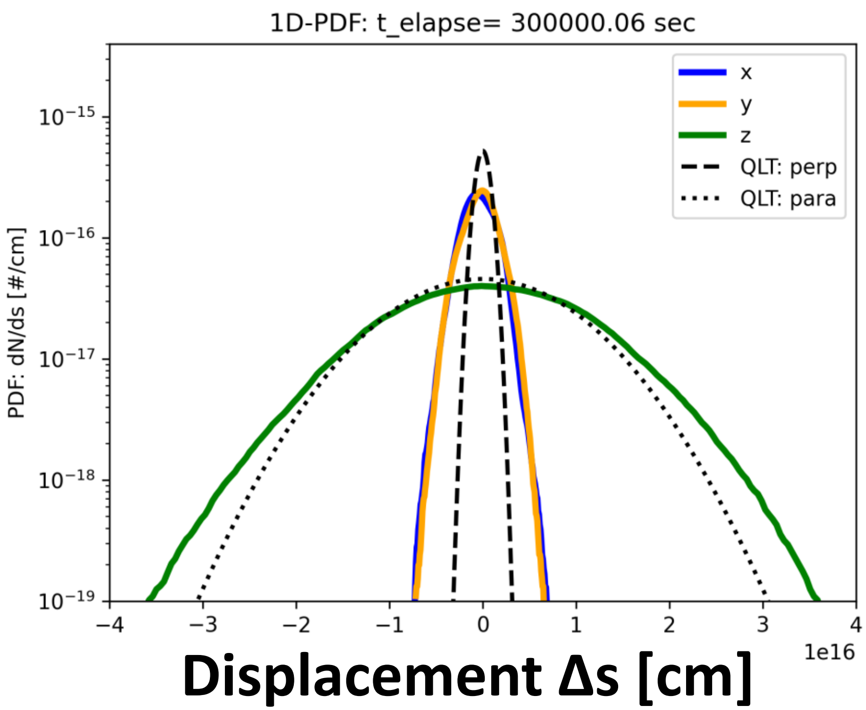
- Model:
  - Quasi-Linear Theory (QLT) + Alfvenic turbulence (e.g., Blandford & Eichler (1987))

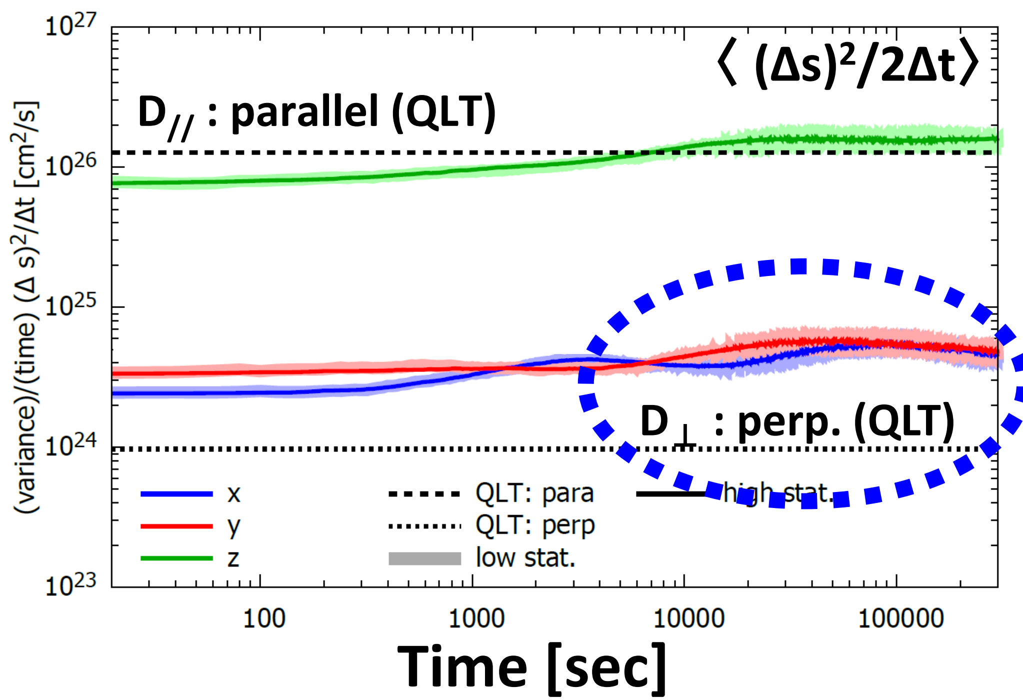
$$D_{\parallel} = \frac{1}{3} r_{\rm L} c \left(\frac{\delta B}{B_0}\right)^{-2}, \quad D_{\perp} = D_{\parallel} \left(\frac{\delta B}{B_0}\right)^{4}$$

Gyro radius: $r_{\rm L} = \frac{E}{eB_0}$

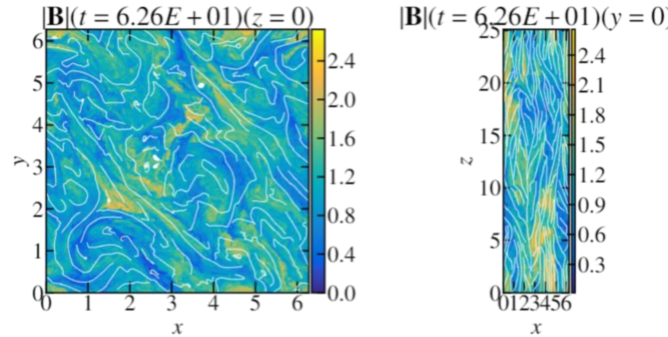
Parallel (z): consistent with QLT

Perpendicular (x,y): much faster than QLT…





Test Calculation: MHD turbulence



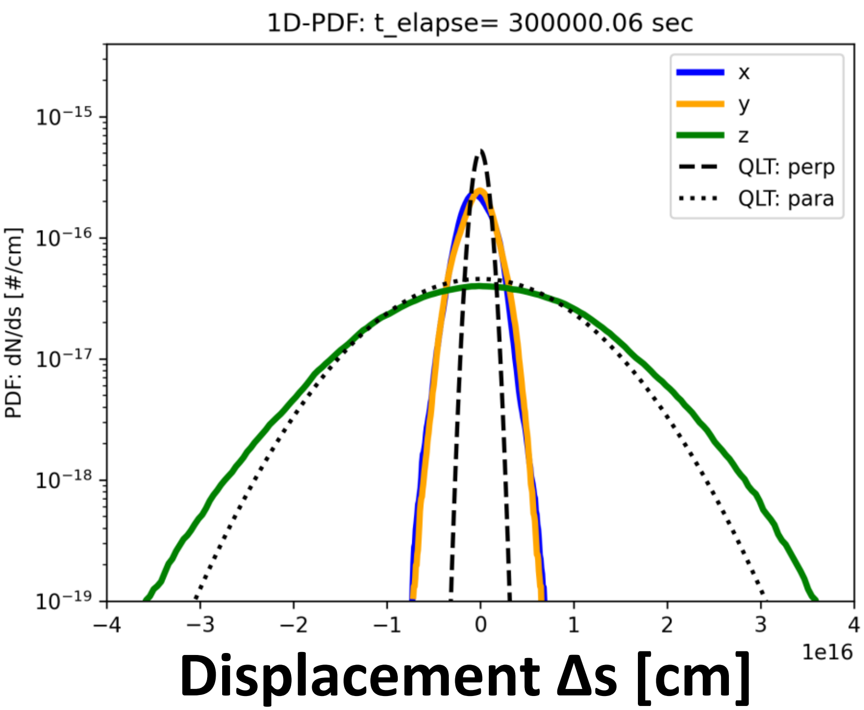
- Model:
  - Quasi-Linear Theory (QLT) + Alfvenic turbulence (e.g., Blandford & Eichler (1987))

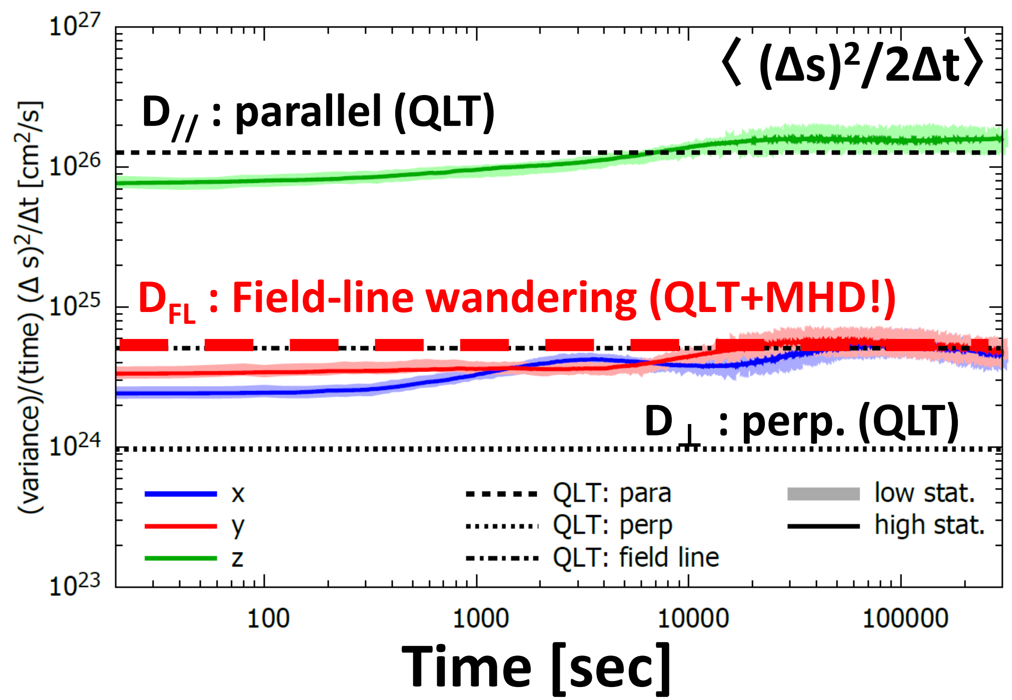
$$D_{\parallel} = \frac{1}{3} r_{\rm L} c \left(\frac{\delta B}{B_0}\right)^{-2}, \quad D_{\perp} = D_{\parallel} \left(\frac{\delta B}{B_0}\right)^{4}$$

Gyro radius: $r_{\rm L} = \dfrac{E}{eB_0}$

Perpendicular: consistent with the field-line wandering (MHD effect!)

(e.g., Jokipii & Parker (1968))





Summary / Next steps

- Summary
  - We are developing a code to solve for the acceleration and propagation of cosmic rays consistent with MHD simulations
  - Using a stochastic differential equation approach, a easily extendable and parallelization-efficient code can be designed
  - We have calculated for the spatial diffusion of particles on a box simulation of MHD turbulence
    - We adopted the quasi-linear theory (QLT) and solved the Fokker-Planck equation consistently with MHD simulation
    - We confirm that the code can capture the combined effect of QLT and MHD: field-line wandering
- Next step
  - Calculations for multi-energy case (since this work was a mono energy calculation)
  - Calculations including momentum space diffusion (acceleration)
  - Implementation of time evolution of turbulence field
  - Application to accretion disk systems

Relationship to other studies in C01

- Kimura-san said…

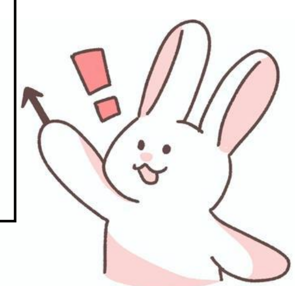
- MHD Simulation + Test Particle Simulation
  - Solve orbits of CR particles using MHD data sets
  - Enable us to obtain diffusion coefficients
  - limited to CRs with $r_L > \Delta x$

SSK et al. 2016, 2019, in prep

↓ Model for diffusion coefficient

- MHD Simulation + CR Transport simulation
  - Solve CR transport equation using MHD data sets
  - We need a model for diffusion coefficients
  - We can obtain useful info for CRs with $r_L < \Delta x$

Talk by Ishizaki-san; Poster by Kawashima-san



Stochastic Differential Equation (SDE) method

- (Ito-type) stochastic differential equation (SDE)
  - Ordinary differential equation w/ stochastic term
  - Ito-SDE has a following standard form:

$$\frac{d\hat{v}}{dt} = -a(\hat{v}) + b(\hat{v}) \cdot \hat{\xi}$$

$$\Leftrightarrow \; d\hat{v} = -a(\hat{v})dt + b(\hat{v}) \cdot d\hat{L} \Longleftrightarrow \hat{v}(t) = \hat{v}(0) - \int_0^t a(\hat{v}(s))ds + \int_0^t b(\hat{v}(s))d\hat{L}(s)$$

Where a(v) and b(v) are smooth functions, ξ is a stochastic variable generated by Gaussian process

- One-to-one correspondence between a SDE and a PDE (partial differential equation)
  - The ensemble of solutions to the SDE follows a PDE called the master equation
  - In particular, for the Ito-SDE, the master equation is the diffusion-advection equation

$$\frac{d\hat{v}}{dt} = -\underbrace{a(\hat{v})}_{\text{Drift}} + \underbrace{b(\hat{v})}_{\text{Random walk (mean free path)}} \cdot \hat{\xi} \Longleftrightarrow \frac{\partial P(v,t)}{\partial t} = \frac{\partial}{\partial v}(\underbrace{a(v)}_{\text{Advection}}P(v,t)) + \frac{\partial^2}{\partial v^2}\left(\underbrace{\frac{1}{2}b(v)^2}_{\text{Diffusion}}P(v,t)\right)$$

Drift

Random walk (mean free path)

Advection

Diffusion

$$\left(\langle f(v)\rangle = \int f(v)P(v,t)dv\right)$$

Stochastic Differential Equation (SDE) method

- (Ito-type) stochastic differential equation (SDE)
  - Ordinary differential equation w/ stochastic term
  - Ito-SDE has a following standard form:

$$\frac{d\hat{v}}{dt} = -a(\hat{v}) + b(\hat{v}) \cdot \hat{\xi}$$

$$\Leftrightarrow \; d\hat{v} = -a(\hat{v})dt + b(\hat{v}) \cdot d\hat{L} \Longleftrightarrow \hat{v}(t) = \hat{v}(0) - \int_0^t a(\hat{v}(s))ds + \int_0^t b(\hat{v}(s))d\hat{L}(s)$$

Where a(v) and b(v) are smooth functions, ξ is a stochastic variable generated by Gaussian process

- One-to-one correspondence between a SDE and a PDE (partial differential equation)

The advection-diffusion equation can be solved by solving a large number of Ito-SDEs and taking their ensemble!

$$\frac{d\hat{v}}{dt} = -a(\hat{v}) + b(\hat{v}) \cdot \hat{\xi} \Longleftrightarrow \frac{\partial P(v,t)}{\partial t} = \frac{\partial}{\partial v}(a(v)P(v,t)) + \frac{\partial^2}{\partial v^2}\left(\frac{1}{2}b(v)^2 P(v,t)\right)$$

Drift ($-a(\hat{v})$)

Random walk (mean free path) ($b(\hat{v}) \cdot \hat{\xi}$)

Advection ($a(v)$)

Diffusion ($\frac{1}{2}b(v)^2$)

$$\left(\langle f(v) \rangle = \int f(v)P(v,t)dv\right)$$

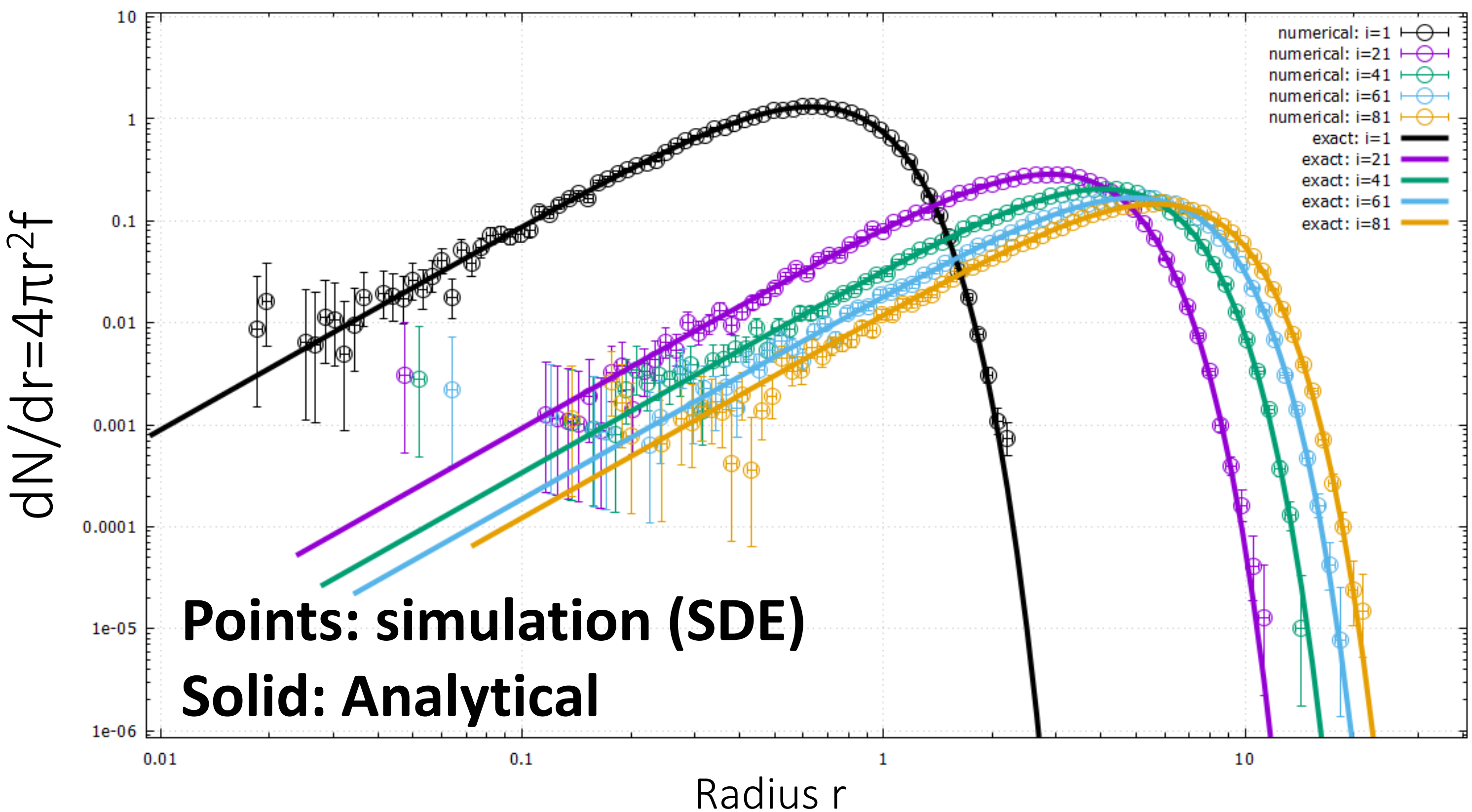
Advantages / Disadvantages (vs. grid-based method)

- Advantages
  - Easily expandable to higher dimensions and multi-particle species
  - High parallelization efficiency ~100% (∵ just solve many independent ODEs)
  - Computational stability is easily ensured because CFL conditions caused by grid size do not occur
  - Intuitive introduction of new effects, since only effects over single particle equations are considered
- Disadvantages
  - Difficult to set boundary conditions
    - But, in our field, we basically consider relatively simple boundary conditions (e.g., "0" at infinity)
  - Computational accuracy depends on particle number statistics
    - Can be compensated by high parallelization efficiency

Test-calculation: simple diffusion in 3D-space

- 3D diffusion: D=1.0, impulsive injection in t=0 (@r_0=0)
  - The calculation is performed in Cartesian coordinate (x,y,z)

$$\frac{\partial f}{\partial t} - D\Delta f = 0 \qquad G(r,t;r_0,t_0) = \frac{1}{\pi^{3/2}\left[4D\left(t-t_0\right)\right]^{3/2}} \exp\left(-\frac{\left|r-r_0\right|^2}{4D\left(t-t_0\right)}\right)$$



Test-Calculation 2: Stochastic acceleration

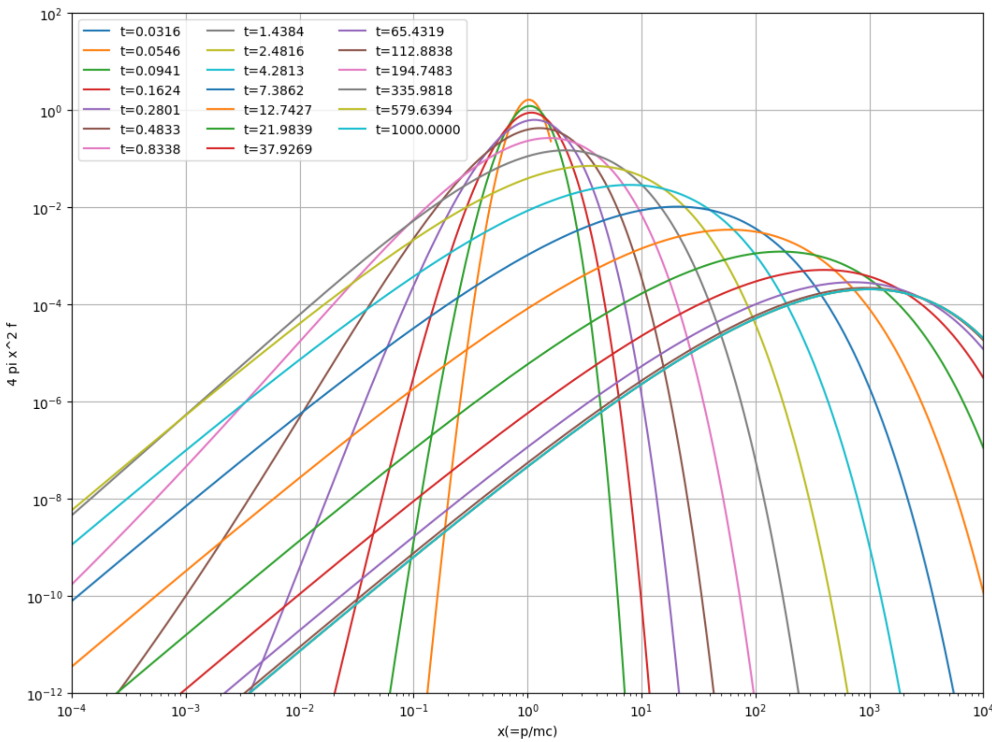
- Mertsch 2011; Green's function of the FP equation in momentum space

$$\frac{\partial f(p,t)}{\partial t} = -\frac{1}{p^2}\frac{\partial}{\partial p}\left(p^2\left(-D_{pp}(p,t)\frac{\partial f(p,t)}{\partial p} + A(p,t)f(p,t)\right)\right)$$

$$A(p,t) = mc\left(\frac{p}{mc}\right)a_0, \quad D_{pp}(p,t) = k_0(mc)^2\left(\frac{p}{mc}\right)^q$$

Injcted at $t=t_0$ and $x=x_0$ ($p=x_0mc$)

$$f = \frac{1}{(mc)^3 4\pi x_0^3}\exp\left(-\frac{3}{2}a_0(t-t_0)\right)\frac{a_0}{k_0}\frac{(xx_0)^{(2-q)/2}\sqrt{g(t)}}{1-g(t)}\exp\left(-\frac{a_0}{(2-q)k_0}\frac{x^{2-q}g(t)+x_0^{2-q}}{1-g(t)}\right)I_{\frac{1+q}{2-q}}\left[\frac{a_0}{(2-q)k_0}\frac{2(xx_0)^{(2-q)/2}\sqrt{g(t)}}{1-g(t)}\right]\left(\frac{x}{x_0}\right)^{-3/2}$$



$$g(t) = \exp\left[-(2-q)a_0(t-t_0)\right]$$

I_ν: modified Bessel function

$m=c=1$

$x_0=1$

$t_0=0$

$a_0=-0.1$ **(<0)**

$k_0=0.5$

$q=5/3$

Steady state at t~>100

Test-Calculation 2: Stochastic Acceleration

- Example of formulation in SDE
  - If $\phi=4\pi p^2 f$, we can rewrite the FP equation in the form of the master equation for an Ito-SDE

$$\frac{\partial f(p,t)}{\partial t} = -\frac{1}{p^2}\frac{\partial}{\partial p}\left(p^2\left(-D_{pp}(p,t)\frac{\partial f(p,t)}{\partial p} + A(p,t)f(p,t)\right)\right)$$

$$\Longrightarrow \frac{\partial \phi(p,t)}{\partial t} = -\frac{\partial}{\partial p}\left(\left(A(p,t) + \frac{2D_{pp}(p,t)}{p} + \frac{\partial D_{pp}(p,t)}{\partial p}\right)\phi(p,t)\right) + \frac{\partial^2}{\partial p^2}(D_{pp}(p,t)\phi(p,t))$$

- Ito-type SDEs and Master Equations (Restated)

$$\frac{d\hat{v}}{dt} = -a(\hat{v}) + b(\hat{v})\cdot\hat{\xi} \quad \leftrightarrow \quad \frac{\partial P(v,t)}{\partial t} = \frac{\partial}{\partial v}(a(v)P(v,t)) + \frac{\partial^2}{\partial v^2}\left(\frac{1}{2}b(v)^2 P(v,t)\right)$$

- By comparing the coefficients, the SDE corresponding to the FP equation is obtained as:

$$d\hat{p} = \left(A(p,t) + \frac{2D_{pp}(p,t)}{p} + \frac{\partial D_{pp}(p,t)}{\partial p}\right)dt + \sqrt{2D_{pp}(p,t)}d\hat{W}$$

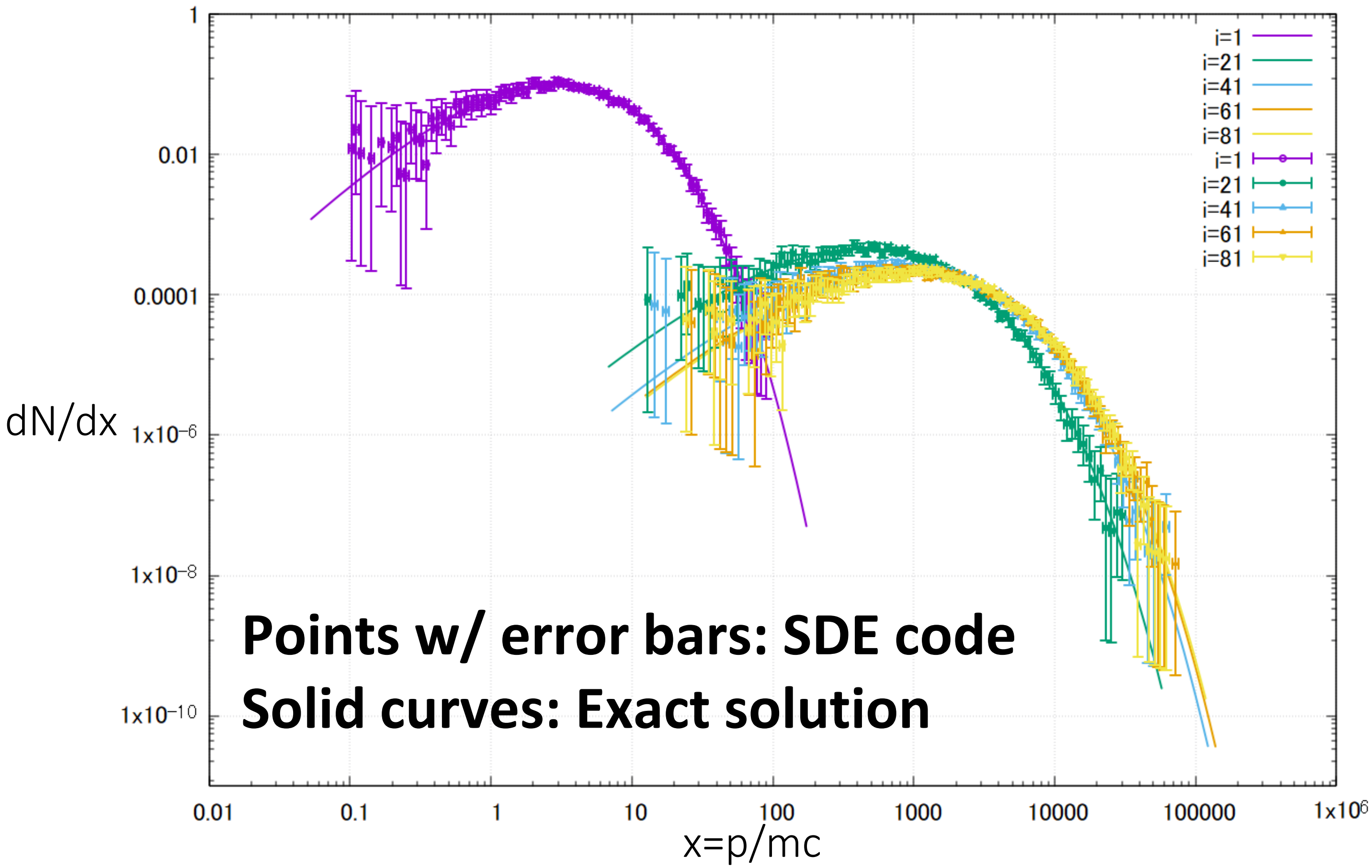
Test-Calculation 2: Stochastic Acceleration

- Comparison with exact solution

$$d\hat{p} = \left(A(p,t) + \frac{2D_{pp}(p,t)}{p} + \frac{\partial D_{pp}(p,t)}{\partial p} \right) dt + \sqrt{2D_{pp}(p,t)} d\hat{W}$$

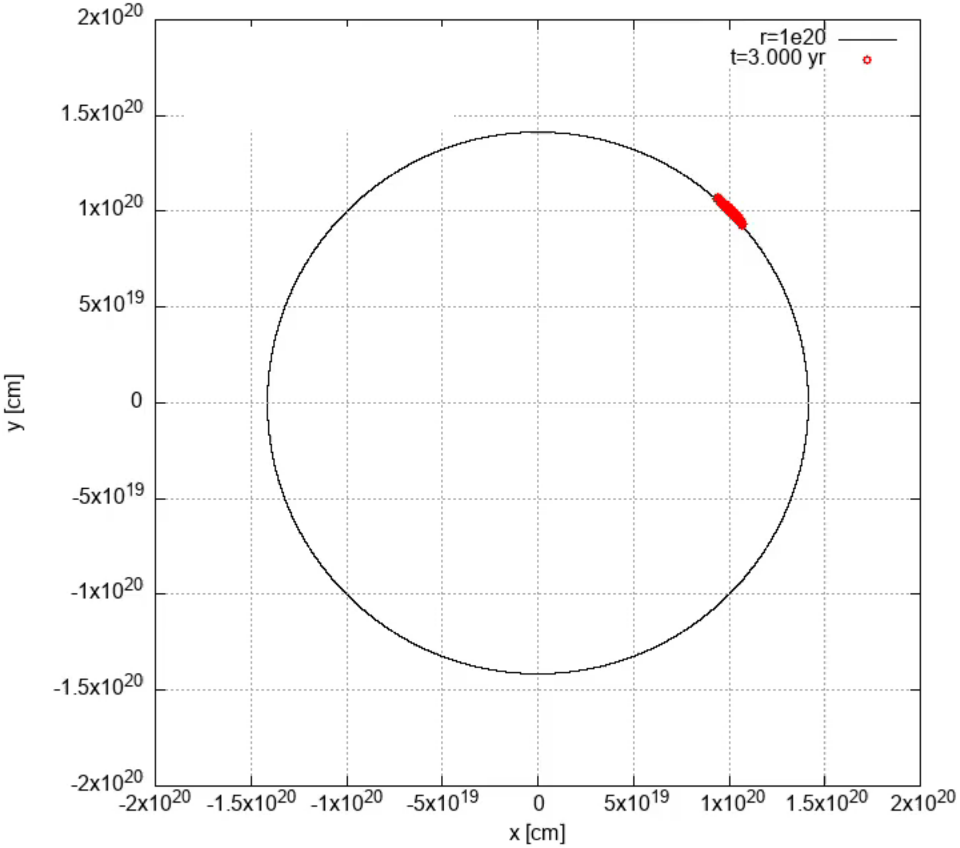
  - Points w/ error bars: simulation by SDE code, Solid line: analytical solution

Successfully solve the FP equation with this technique!



Test-Calculation 3: anisotropic diffusion

- Magnetic field: ϕ-direction
  - $\kappa_\perp = 10^{-6}\kappa_{//}$, $\kappa_{//} = 3.0\times10^{28}(E/10\ \mathrm{GeV})^{1/3}$
  - Setting: 1000-particles, 3000yr, Power-law injection to energy space



$$\overleftrightarrow{\kappa} = \begin{pmatrix} \kappa_\perp + (\kappa_\parallel - \kappa_\perp)\dfrac{y^2}{x^2+y^2} & -(\kappa_\parallel - \kappa_\perp)\dfrac{xy}{x^2+y^2} & 0 \\ -(\kappa_\parallel - \kappa_\perp)\dfrac{yx}{x^2+y^2} & \kappa_\perp + (\kappa_\parallel - \kappa_\perp)\dfrac{x^2}{x^2+y^2} & 0 \\ 0 & 0 & \kappa_\perp \end{pmatrix}$$

$$\nabla\cdot\overleftrightarrow{\kappa} = -\frac{\kappa_\parallel - \kappa_\perp}{\rho}\hat{\rho}$$

$$(\rho = \sqrt{x^2+y^2})$$

Anisotropic diffusion is solved well!

Now, Roughly all processes have been introduced

⇒Next, connection with MHD calculation

Test-Calculation 4: Pulsar Wind Nebulae

- MHD
  - By loading (mock) 3DMHD data, interpolation function of velocity and magnetic fields are generated
  - Comparison: Spherically symmetric steady-state diffusion model of PWNe (Ishizaki+2018; left)
  - Differences in calculation setup:
    - Boundary: SDE code injects particles multiple times at appropriate time intervals to reproduce fixed boundaries
    - Grid code solves the steady state eq., while SDE code solves the time-dependent eq. until it becomes steady.
  - Energy spectrum of particles at each radius (calculated in 3+1 dimensions in the SDE code)