Study of energetic particle interaction with MHD using the M3D-C1 code

Chang Liu

Princeton Plasma Physics Laboratory

NSTX-U / Magnetic Fusion Science meeting

Oct 5 2020

Highlights

- M3D-C1-K is newly-developed kinetic module for M3D-C1 code, to study the interaction between energetic ions and MHD activities (Alfvén waves, kink/tearing modes etc). The goal is to have the same capability of M3D-K code, with better performance on modern computers.
- Linear and nonlinear benchmarks have been done to test the correctness of the code.
- We use a semi linear method to study the frequency chirping of Alfvén eigenmodes and the evolution of mode structure.

Overview

1. Implementation of particle pushing and MHD coupling
2. GPU acceleration of particle pushing
3. Simulation results and benchmark with other codes
4. Study of Alfvén eigenmode frequency chirping
5. Summary

Outline

1. Implementation of particle pushing and MHD coupling

2. GPU acceleration of particle pushing

3. Simulation results and benchmark with other codes

4. Study of Alfvén eigenmode frequency chirping

5. Summary

Particle pushing based on guiding-center model

$$\frac{d\mathbf{X}}{dt} = \frac{1}{B^*}\left(v_{\|}\mathbf{B}^* - \mathbf{b}\times\mathbf{E}^*\right)$$

$$m\frac{d\mathbf{v}_{\|}}{dt} = \frac{q}{B^*}\mathbf{B}^*\cdot\mathbf{E}^*$$

$$\mathbf{B}^* = \mathbf{B} + \frac{mv_{\|}}{e}\nabla\times\mathbf{b}, \qquad B^* = \mathbf{B}^*\cdot\mathbf{b}$$

$$\mathbf{E}^* = \mathbf{E} - \frac{mv_{\|}}{e}\frac{\partial\mathbf{b}}{\partial t} - \frac{\mu}{q}\nabla B$$

- Particle markers are advanced using 4th order Runge-Kutta.
- In guiding center mode, the fields are evaluated at the guiding center. In gyrokinetic mode, the fields are calculated using 4-point averaging along the gyro orbit.
- In the linear run, markers follow drift kinetic equations with equilibrium B fields only.

Slow manifold + Boris provides a structure preserving algorithm for particle motion

- Particle pushing based on Runge-Kutta method suffers from accumulation of numerical error and can lead to nonphysical results for long-time simulation.
- For multi-timescale system, slow manifold characterizes the equilibrium point of fast motion, which can be used to reduce the dimension of dynamical system. For particle motion in the magnetic field, slow manifold is the set of special particle trajectories where gyro motion is absent.
- For these trajectories, there is only one timescale. One can use the full-orbit particle pushing algorithm to calculate the trajectory with large time-step.
  - Boris algorithm is a good candidate that is stable (implicit) and structure preserving.
- The effect of gyro motion on the slow motion can be calculated by including an effective electric force $-\mu\nabla B$.

H. Qin and X. Guan, Phys. Rev. Lett. 100, 035006 (2008).

C.L. Ellison, J.M. Finn, J.W. Burby, M. Kraus, H. Qin, and W.M. Tang, Phys. Plasmas 25, 052502 (2018).

J.W. Burby, Journal of Mathematical Physics 61, 012703 (2020).

J. Xiao and H. Qin, ArXiv:2006.03818 (2020).

Implementing Boris slow-manifold algorithm

- Boris algorithm is similar to a leap-frog method where x and v are evolved at interleaved time points.
  - We should use a Cartesian coordinate to avoid coordinate transformation in dx/dt.
  - Particles should be initialized carefully to stay on the slow manifold, which includes all the drift terms.
  - When calculating the moments, x and v at the same time should be used.

$$x_{l+1} = x_l + v_{l+1/2}\Delta t$$

$$v_{l+3/2} = v_{l+1/2} + \left(E^{\dagger}_{l+1} + \frac{v_{l+3/2} + v_{l+1/2}}{2} \times B_{l+1} \right) \Delta t$$

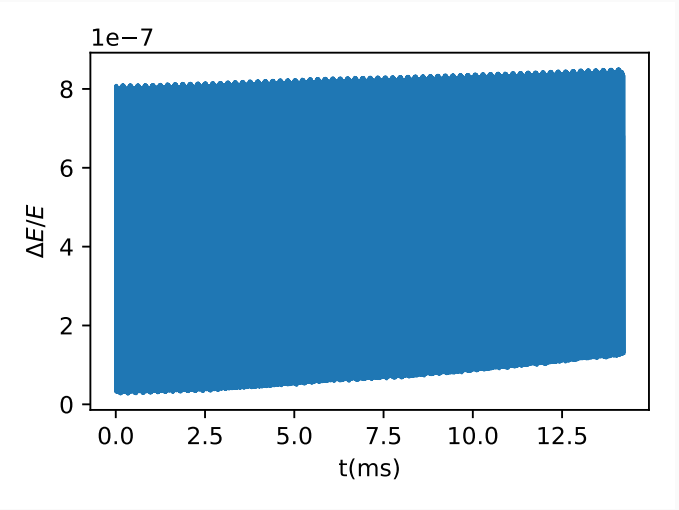
$$E^{\dagger}_{l+1} = E_{l+1} - \mu \nabla B_{l+1}$$

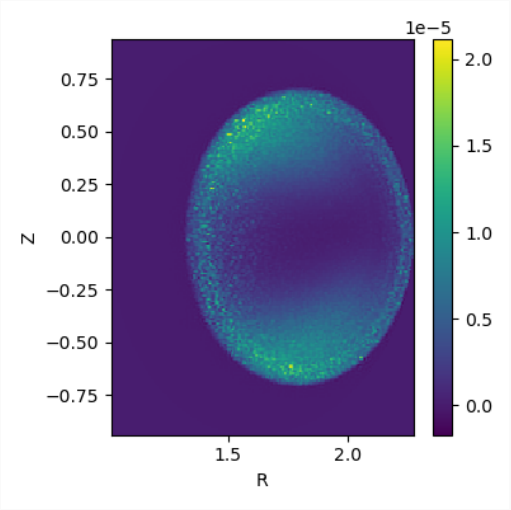
$$v_{l+1} = (v_{l+3/2} + v_{l+1/2})/2$$ (Needed by moment calculation at the end of particle pushing)

Conservation of kinetic energy

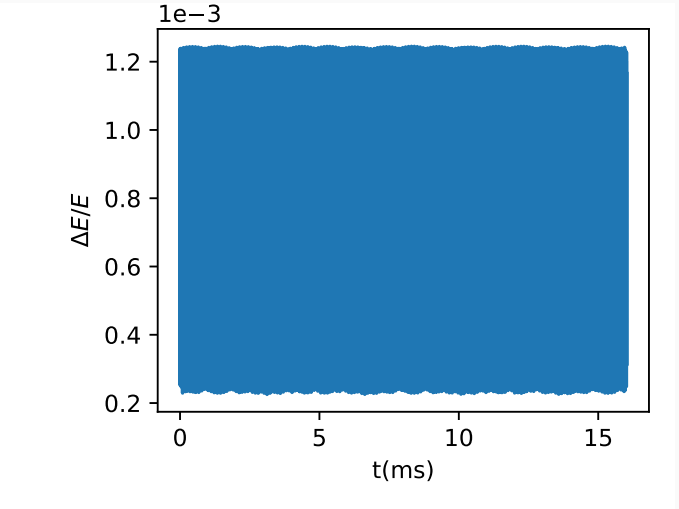
$B = 2\text{T}$, $E = 130\text{keV}$ ($v = 5 \times 10^6 \text{m/s}$), $dt = 6.5 \times 10^{-8}\text{s} = 2\pi/\Omega$

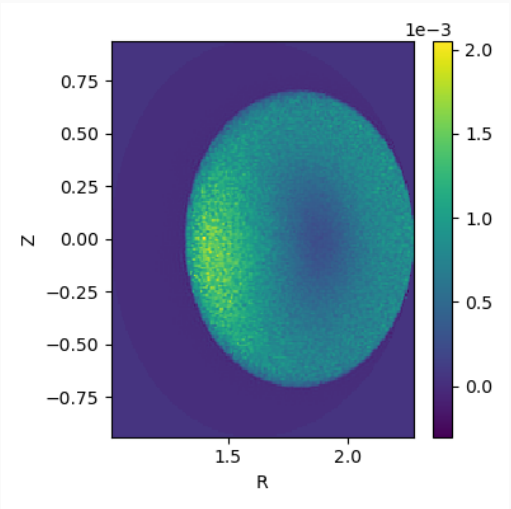
Guiding center RK4:





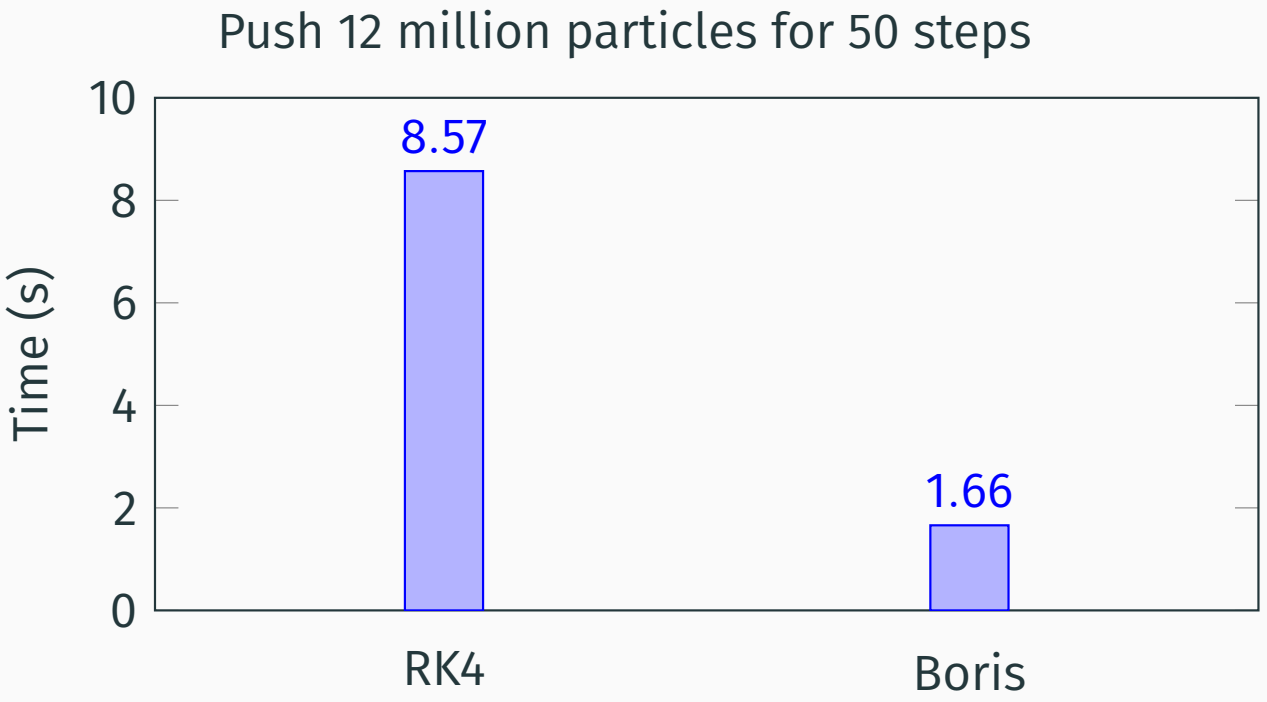
Boris:





Advantages of slow manifold Boris algorithm

- The conservation properties of both RK4 and Boris are both good enough for 10ms simulation.
- However, the benefits of Boris is that the calculation is much simpler than RK4 for each timestep.
  - One only needs to do one time of field calculation instead of 4 times.
  - There is no need to calculate curvature term ($\nabla \times \mathbf{b}$), and the mirror force ($-\mu \nabla B$) can be treated as a gradient of scalar.
- The speedup can be more attractive by using a larger timestep for Boris.



Weight evolution

$$\delta f \text{ method} \qquad \frac{d\delta f}{dt} = -\delta\dot{\mathbf{z}} \cdot \frac{\partial f_0}{\partial \mathbf{z}}$$

$$\frac{dw}{dt} = \frac{\delta\dot{f}}{f} = \frac{1-w}{f_0}\left(-\delta\mathbf{v}\cdot\nabla f_0 - \dot{\epsilon}\partial_\epsilon f_0\right)$$

- Here we use energy derivative ($\dot{\epsilon}$) to calculate weight evolution, which is not consistent with the guiding center equation ($\dot{v_{||}}$) but easier to implement. Will change to $\dot{v_{||}}$ in future.
- The change of Jacobian ($B^*_{||}$) can be taken into account by introducing a new weight $d = w + (1-w)\delta B^*_{||}/B^*_{||0}$, like in Belova (1997).

E.V. Belova, R.E. Denton, and A.A. Chan, J. Comp. Phys. 136, 324 (1997).

Particle deposition

- Parallel and perpendicular pressure are calculated from particles using δ-function deposition

$$\int \nu P_{\parallel} g d\mathbf{x} = \sum_i m v_{i,\parallel}^2 \nu(\mathbf{x}_i)$$

$$\int \nu P_{\perp} g d\mathbf{x} = \sum_i \mu_i B(\mathbf{x}_i) \nu(\mathbf{x}_i)$$

  - We can add a small diffusion to the obtained $P_{\parallel}$ and $P_{\perp}$ to reduce noise, but it will break the energy conservation of the coupling scheme.
- The calculated $P_{\parallel}$ and $P_{\perp}$ can be used for both pressure and current coupling to MHD equations.
  - Pressure coupling

$$\nabla \cdot \mathbf{P} = \nabla P_{\perp} + \nabla \cdot \left(P_{\parallel} - P_{\perp} \right) \mathbf{bb}$$

  - Current coupling

$$\mathbf{J}_{hot} \times \mathbf{B} = \frac{P_{\parallel}}{B^2} \mathbf{b} \times \nabla \times \mathbf{b} - \frac{P_{\perp}}{B^2} \nabla_{\perp} \ln B - \nabla \times \left(\frac{P_{\perp}}{B} \mathbf{b} \right) \times \mathbf{B}$$

Y. Todo and T. Sato, Physics of Plasmas 5, 1321 (1998).

Outline

1. Implementation of particle pushing and MHD coupling

2. GPU acceleration of particle pushing

3. Simulation results and benchmark with other codes

4. Study of Alfvén eigenmode frequency chirping

5. Summary

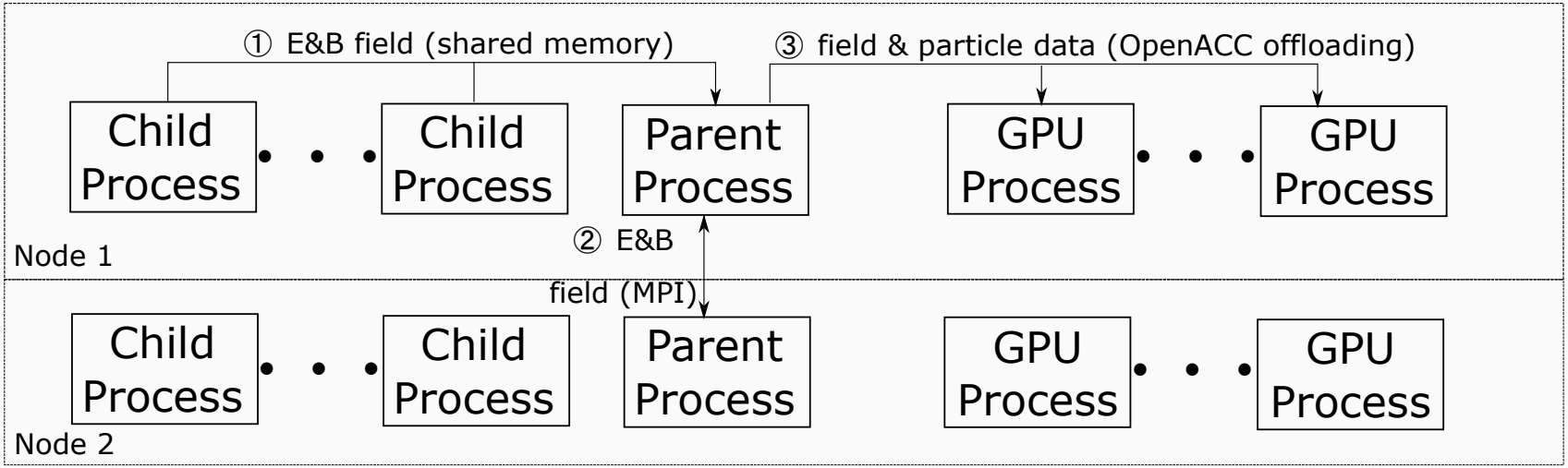
The importance of migrating code to GPU

- Many of the newly built supercomputers utilize GPU to reach high computation power.
  - In Traverse, a new cluster built by PPPL and Princeton University, 97% of computing power comes from GPU.
- GPU can be regarded as a co-processor with many cores and a shared memory.
  - Computation on a single GPU core is slower than on a single CPU core, especially for logical operations.
  - GPU should be used to do strongly parallel jobs with each job very simple, and particle pushing is indeed a suitable job.
- With the help of new API like OpenMP4 or OpenACC, it is now easier to migrate the existing code to run on GPUs.
  - Most of the migration work is related to communications between GPU and CPU (offloading), since they have separate memory.
  - Existing MPI structure of the code can complicate the work.

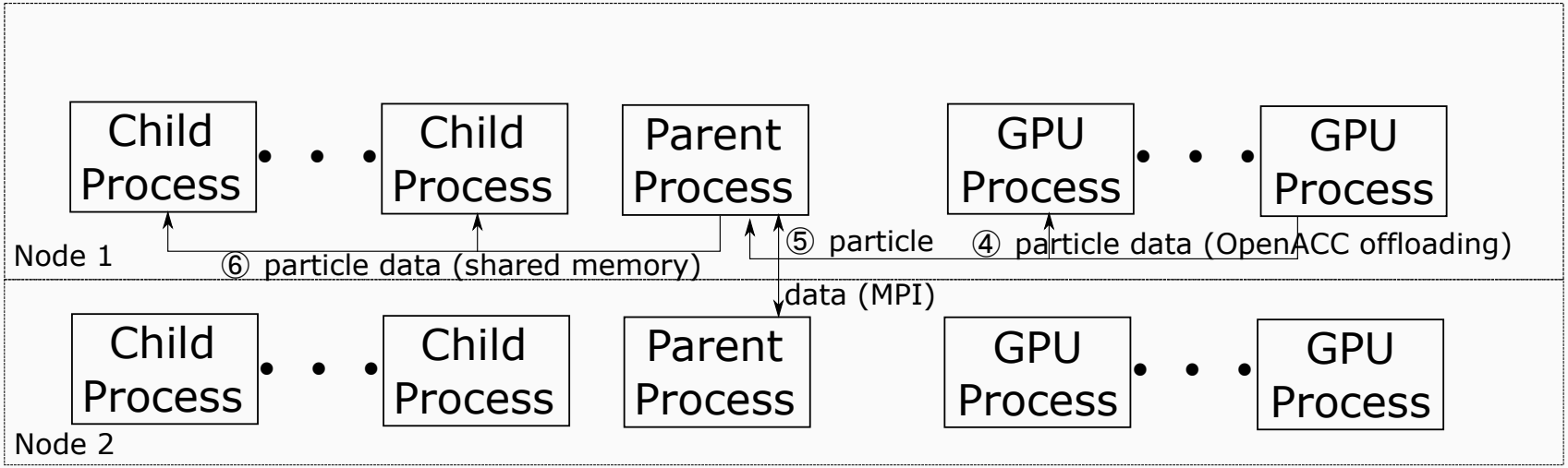
Combining distributed memory and shared memory

- Currently M3D-C1 use MPI processes for parallelization, with domain decomposition and distributed memory mode.
  - Each process only knows the information about a small subdomain of the whole 3D mesh.
- The most efficient way to push particles in GPU is to use particle-based data structure, and each particle is pushed independently.
  - This is more memory-consuming since every GPU needs the field information of the whole mesh. Fortunately for modern GPU with >16GB RAM, this is not a problem.
  - Previously we use a mesh-based data structure to store particle data. This leads a lot of communication due to particles particles moving from one mesh to the other.
- To incorporate the distributed-memory M3D-C1 and share-memory particle pushing, we exploit two methods for data sharing
  - Shared memory function (within one node) introduced in MPI-3.
  - MPI_Allgatherv between different nodes

Procedure and data flow in a particle pushing step



Procedure and data flow in a particle pushing step



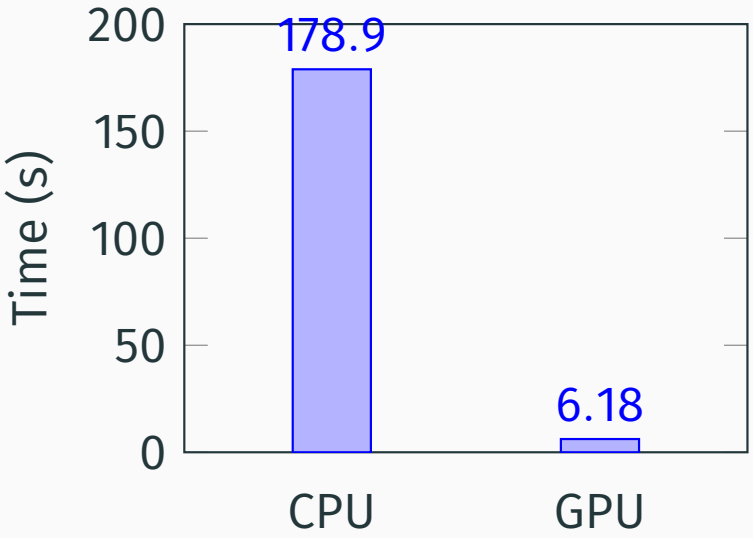
- Using GPU profiler, it is found that the data transfer time is < 0.1s for GPU offloading, which is ignorable compared to GPU computation time with subcycles.

Subcycles of particle pushing within MHD timestep

- Currently M3D-C1 can use timestep of tens of τ_A to simulate large time-scale physics, with the help of the advanced semi-implicit algorithm for velocity advance.
  - This is an improvement over the M3D code, which typically use much smaller timestep.
  - For particle pushing we use explicit RK4, thus the timestep is limited by particle speed.
- Here we use subcycles for particle pushing, which means that we push particle multiple times between two MHD timesteps.
  - GPU-CPU communications are only needed at the beginning and end of subcycles.
  - Fields are fixed during subcycles. This can be improved by utilizing information of time derivative of field.

Performance benchmark of particle pushing on GPU and CPU

- The benchmark was done on Traverse cluster.
- We compare the performance of the advancing 4 million particles for 50 timesteps in a mesh of 5679 elements, on 4 Nvidia Tesla GPUs using OpenACC, compared with that on 2 IBM Power9 CPUs using OpenMP.
- Most of the time for particle pushing is spent in calculating the value of basis functions at each particle location, which requires a calculation of 5th order polynomials.



Outline

1. Implementation of particle pushing and MHD coupling

2. GPU acceleration of particle pushing

3. Simulation results and benchmark with other codes

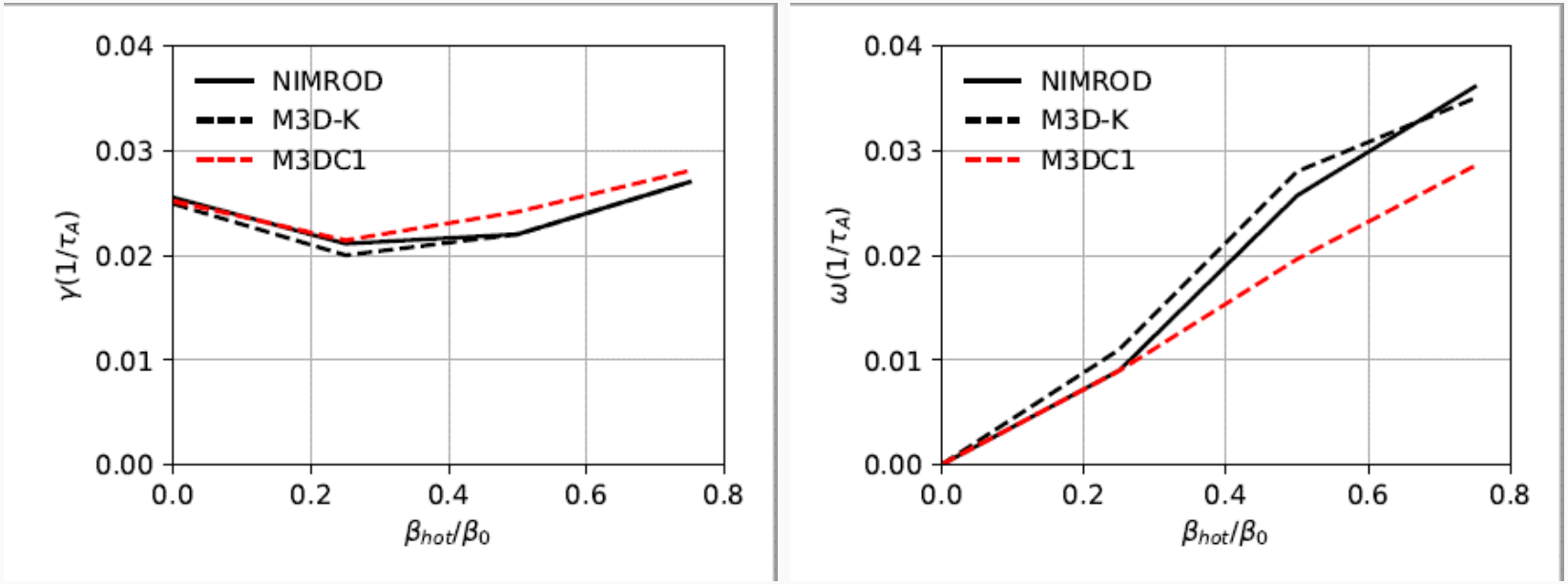
4. Study of Alfvén eigenmode frequency chirping

5. Summary

Fishbone simulation result agrees with M3D-K and NIMROD

$R/a = 2.8, \quad \beta_{total} = 0.08, \quad q_0 = 0.6, \quad q_a = 2.5$

$\hat{\rho}_h = v_0/(\Omega_h a) = 0.0125, v_0/v_A = 4$



- These results are obtained using pressure coupling scheme used in M3D-K and NIMROD. With current coupling, the growth rate increase significantly with β_h

G.Y. Fu, W. Park, H.R. Strauss, J. Breslau, J. Chen, S. Jardin, and L.E. Sugiyama, Phys. Plasmas 13, 052517 (2006).

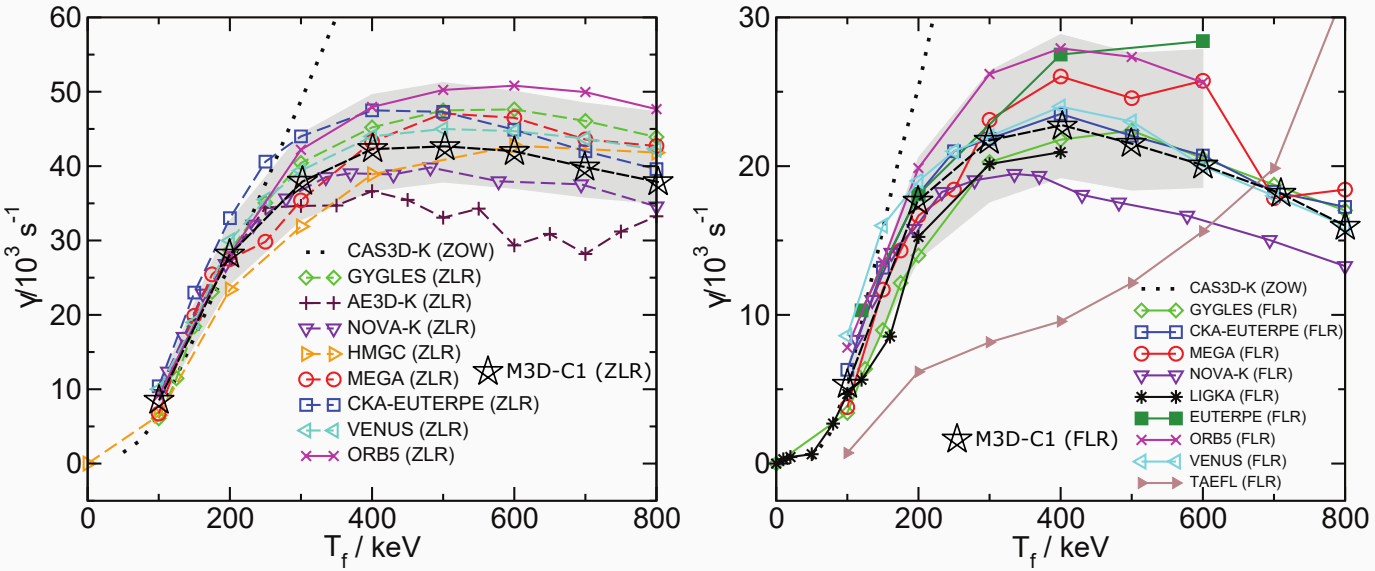
C.C. Kim and the NIMROD Team, Phys. Plasmas 15, 072507 (2008).

TAE linear simulation without and with FLR effects

- This is an ITPA collaborative effort to compare different codes and physical model. Several hybrid MHD, gyrokinetic and gyrofluid codes are benchmarked.

  $R/a = 10, \quad \beta \approx 0.2\%, \quad q = 1.71 + 0.16(r/a)^2$

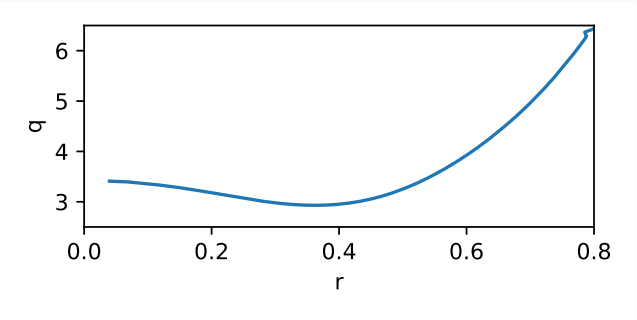
  $n_f = c_3 \exp\left(-\frac{c_2}{c_1} \tanh \frac{\sqrt{s}-0.5}{c_2}\right)$

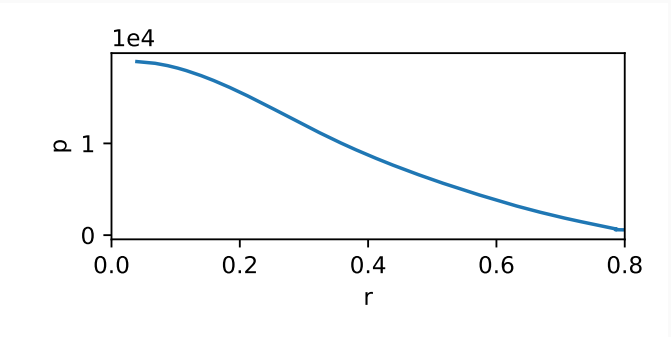


A. Könies, et al., Nucl. Fusion 58, 126027 (2018).
Yawei Hou, et al., Physics of Plasmas 25, 012501 (2018)

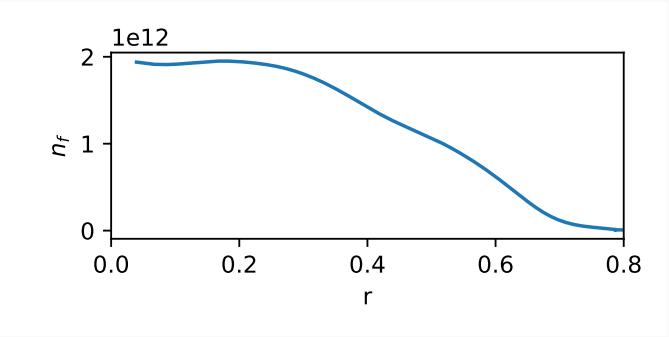
Use a DIII-D equilibrium to study excitation of RSAE

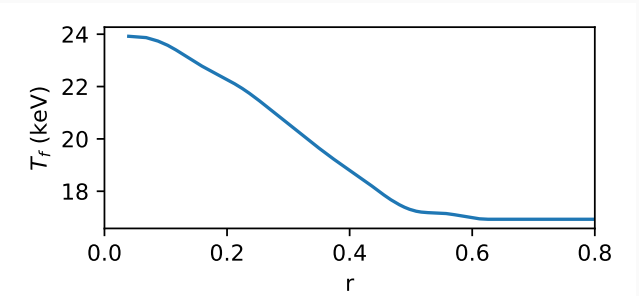
- Recently several MHD and gyrokinetic codes are employed to study the linear growth of reversed shear Alfvén eigenmode (RSAE) using DIII-D experimental parameters.
- $B_0 = 2\text{T}$, $R = 1.6435\text{m}$, $a = 0.627\text{m}$
- q profile has a minimum at with $q_{min} = 2.93$ at $r = 0.36\text{m}$





- EP follows a Maxwellian distribution in momentum space.

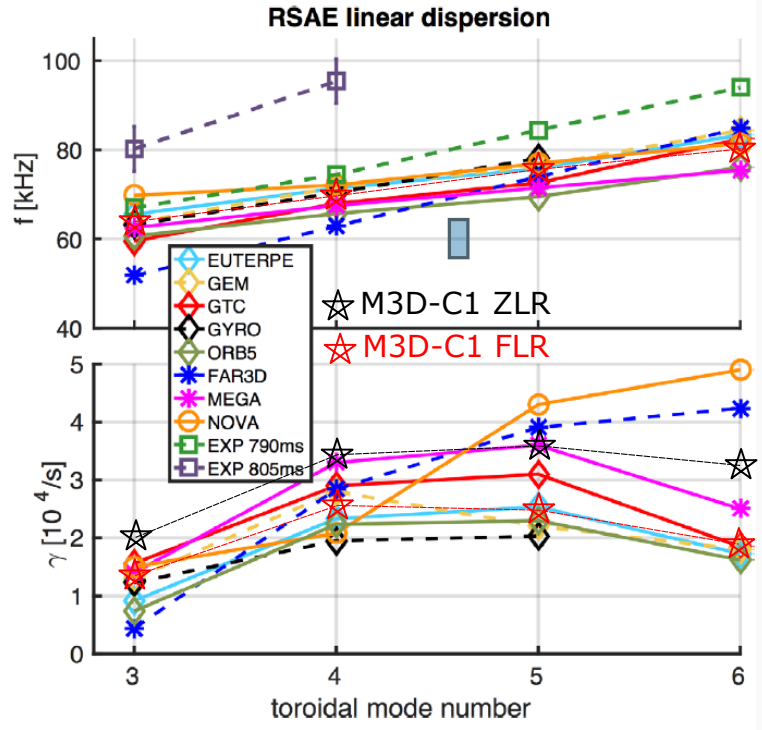




S. Taimourzadeh et al., Nucl. Fusion 59, 066006 (2019)

Linear simulation of RSAE driven in DIII-D tokamak

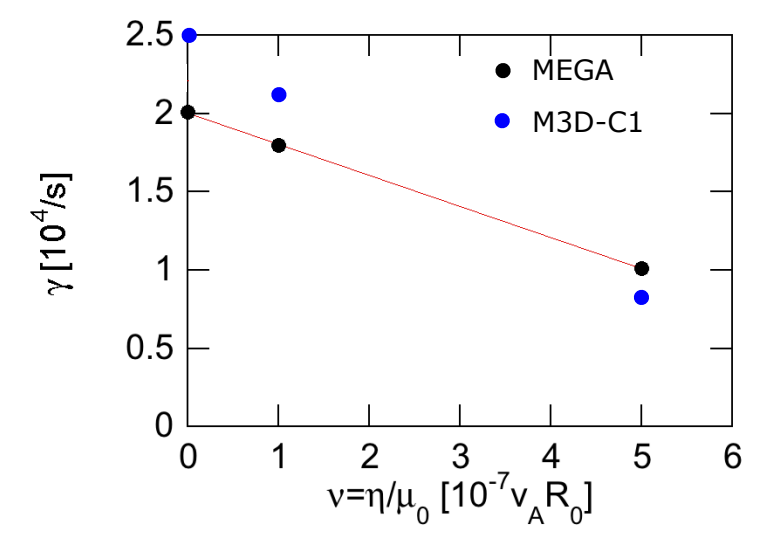
- Including FLR effects leads to smaller mode growth rate, especially for high-k modes.
- We got almost the same results using pressure coupling or current coupling, meaning that the parallel dynamics are not important.
- Compressional effects ($\delta B_{||}$) are not important.

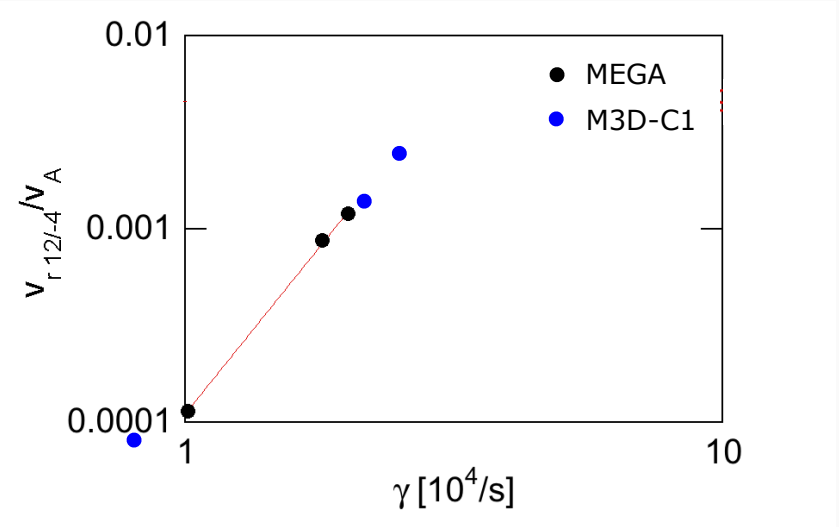


S. Taimourzadeh et al., Nucl. Fusion 59, 066006 (2019).

Nonlinear simulation of RSAE with different values of dissipation

- Based on the linear simulation, we further calculate the growth rate of $n = 4$ RSAE with a finite dissipation (resistivity, viscosity), and the saturation level of the mode.
  - The growth rate shows a linear dependence on dissipation, with a slope larger than MEGA results.
  - Saturation levels also shows stronger dependence, with a similar power of $V_{\mathrm{sat}} \sim \gamma$ relation as MEGA results.

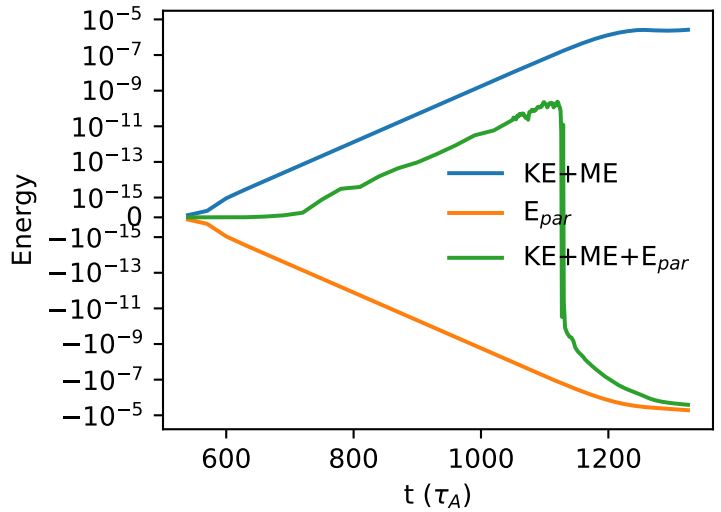




Energy conservation test

- We study the energy conservation in the nonlinear simulation of RSAE in DIII-D.
- Both the kinetic and magnetic energy are calculated using the perturbed field only, to reduce the noise coming from the equilibrium fields.

- Pressure is chosen to be very small.
- In energetic particle energy calculation, the contribution from full-f current is subtracted to reduce noise (Belova (1997))



- The total energy change is within 10% of MHD energy increase during linear growing stage, but in the saturation stage this error is significantly larger.
  - We think this is caused by the phase mixing related to continuum damping of RSAE, since high-k modes can be excited.

E.V. Belova, R.E. Denton, and A.A. Chan, J. Comp. Phys. 136, 324 (1997).

Outline

1. Implementation of particle pushing and MHD coupling

2. GPU acceleration of particle pushing

3. Simulation results and benchmark with other codes

4. Study of Alfvén eigenmode frequency chirping

5. Summary

Frequency chirping of Alfvén eigenmodes

- Frequency chirping is widely observed in AEs excited by EPs in tokamaks and STs.
- Berk-Breizman theory gives a solid explanation about up and down frequency chirping through clump-hole formation.
- It is computationally expensive to do a full nonlinear simulation for both particles and MHD field for a long time (>10ms$\sim 10^4 \tau_A$)

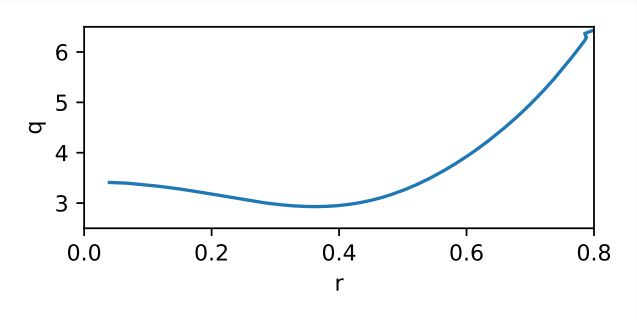
Semi-linear method for particle-MHD simulation

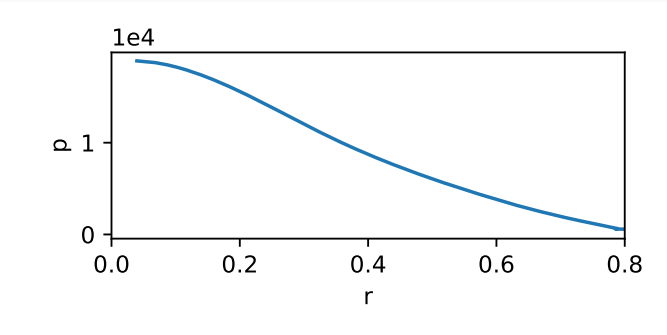
- Recently Roscoe White presented his study on mode frequency chirping using ORBIT, which utilized NOVA to calculate the eigenmode structure, and information from ORBIT particles to calculate mode amplitude and phase changes using δf method.
  - This study shows that the frequency chirping is caused by nonlinear effects in particles instead that in MHD modes.
- To study the frequency chirping, one can do a linear MHD simulation plus a nonlinear particle simulation.
  - There is only one mode get excited, and mode-mode interaction is not important.
  - The mode saturation and frequency chirping is due to the flattening of particle distribution function and clump-hole formation, which can be incorporated through a nonlinear δf method.
- For M3D-C1, linear MHD equations are much easier to simulate since there is no need to calculate and factorize MHD equation matrix at every timestep.
  - The particle simulation is also easier to do since the basis function is easier to calculation for 2D mesh than 3D.

Y. Todo, H.L. Berk, and B.N. Breizman, Nucl. Fusion 52, 094018 (2012).
Y. Chen, G.Y. Fu, C. Collins, S. Taimourzadeh, and S.E. Parker, Physics of Plasmas 25, 032304 (2018).

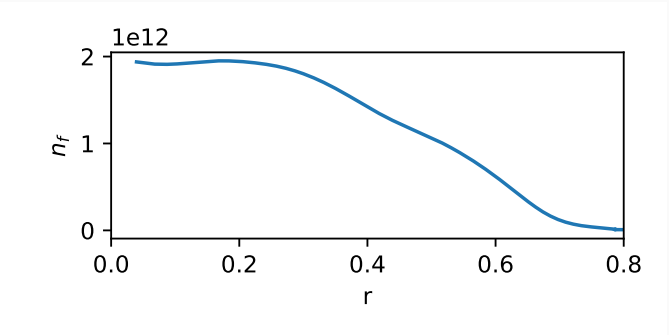
Use a DIII-D equilibrium to study excitation of RSAE

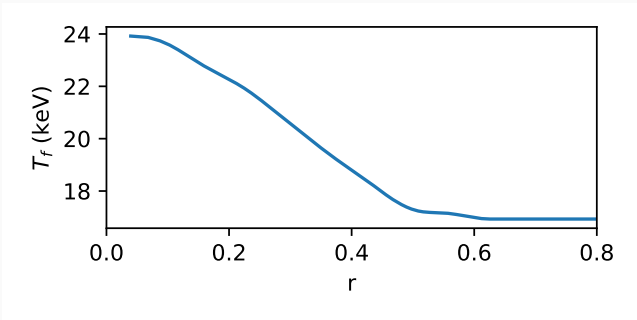
- $B_0 = 2\text{T}$, $R = 1.6435\text{m}$, $a = 0.627\text{m}$
- q profile has a minimum at with $q_{min} = 2.93$ at $r = 0.36\text{m}$





- EP follows a Maxwellian distribution in momentum space.



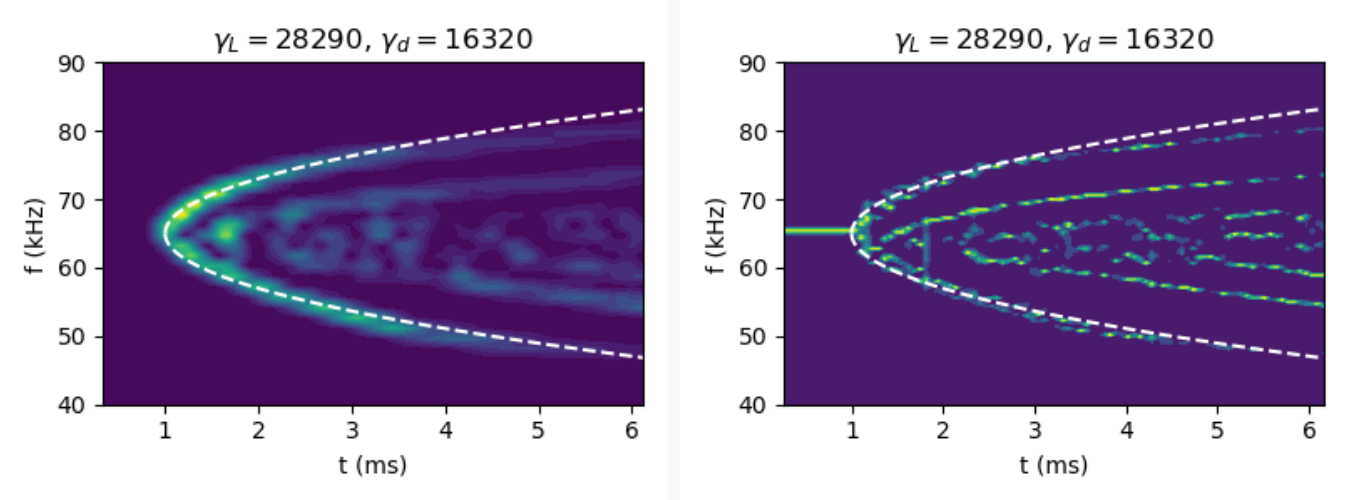


- Only the nonlinear evolution of $n = 4$ RSAE is studied.

S. Taimourzadeh et al., Nucl. Fusion 59, 066006 (2019)

Spectrogram analysis using DMUSIC

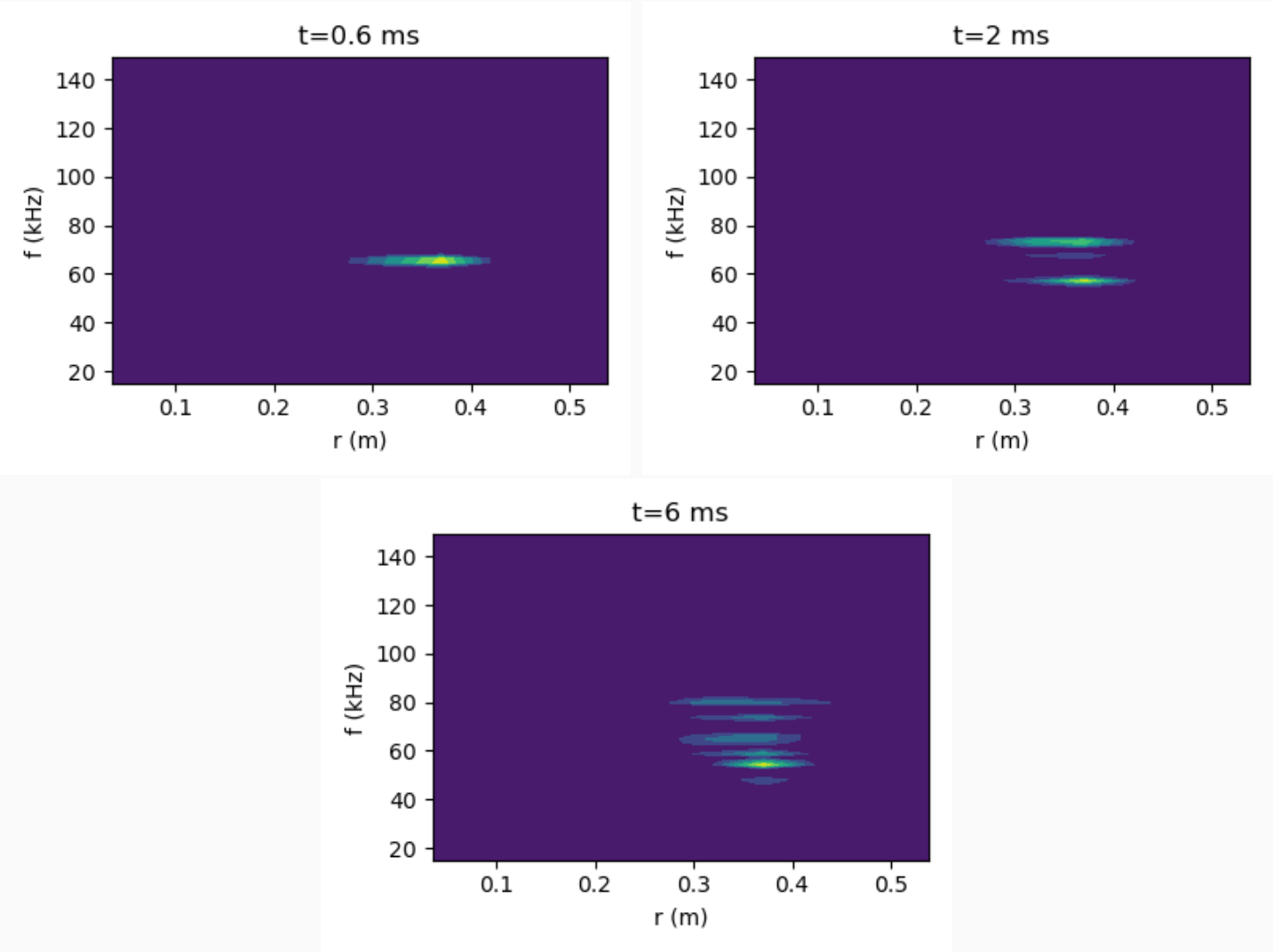
- DMUSIC is an algorithm used for frequency detection by performing an eigen decomposition of the covariance matrix of signal samples.
  - By choosing the N maximum eigenvalues for the correlation matrix, we can separate the signal subspace and the noise subspace, and then use the orthogonality between two subspaces to calculate the characteristic frequency and damping rate in the signal subspace.
  - The estimator function is strongly peaked at the signal frequency, which can give a sharper result than FFT spectrogram.



- The result shows that frequency chirping rate is consistent with the Berk-Breizman theory.

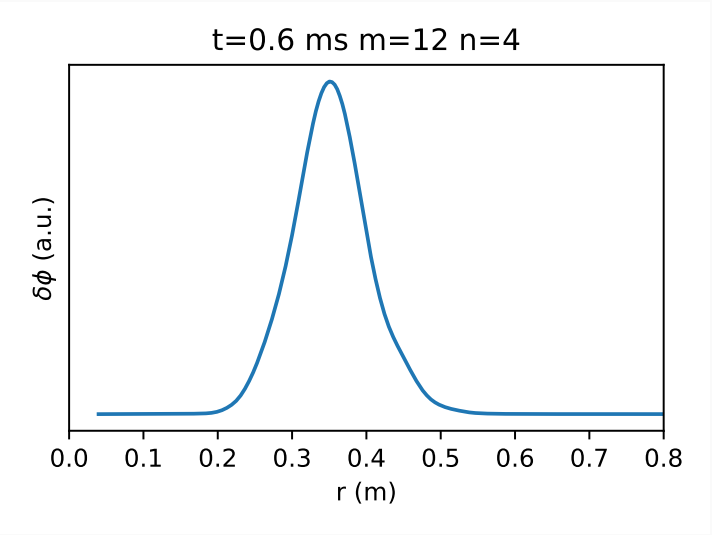
$$\delta f = \frac{16\sqrt{2}}{\pi^2 3\sqrt{3}} \gamma_L \sqrt{\gamma_d t}$$

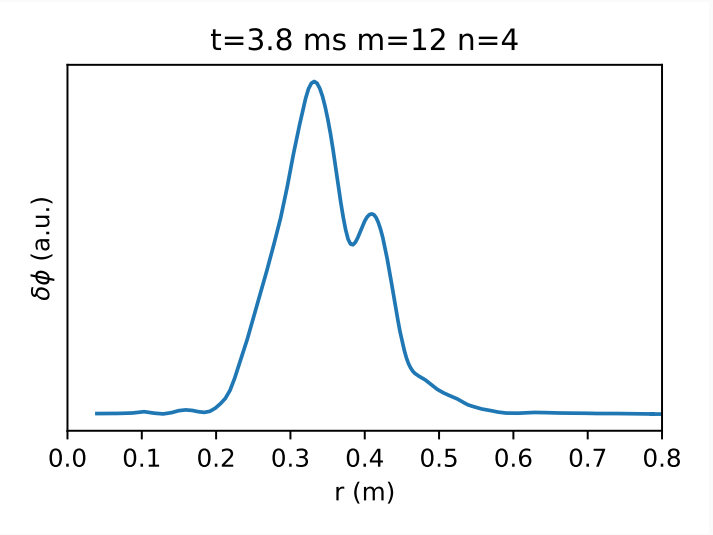
Mode structure

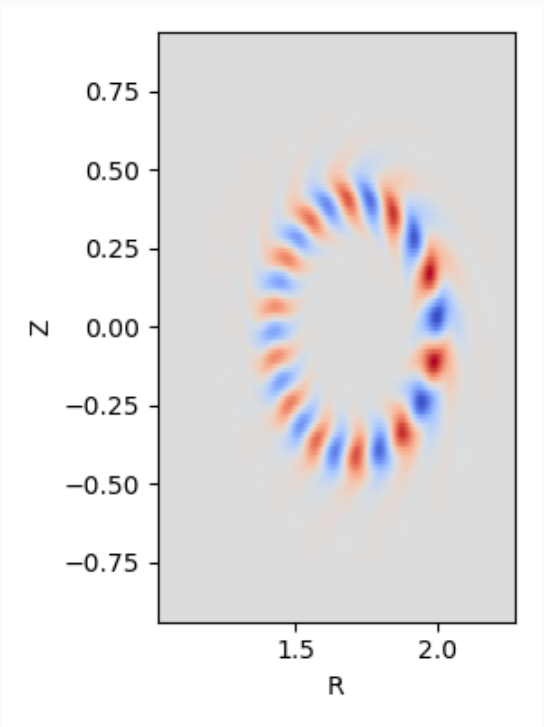


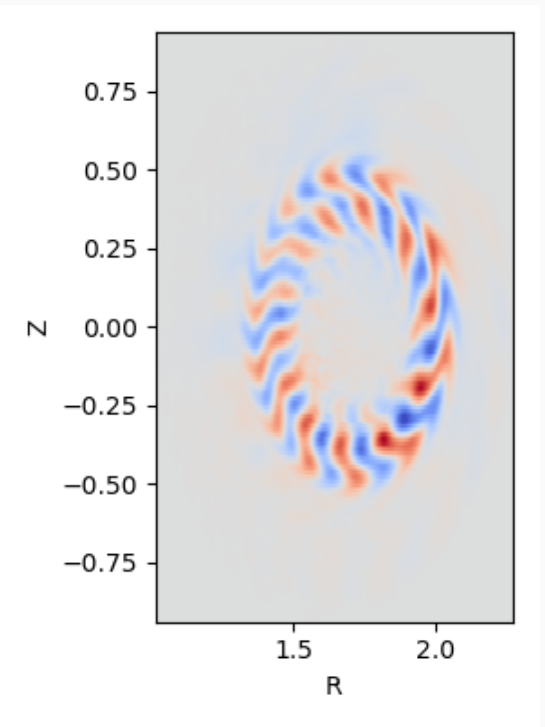
- The mode structure of the modes after splitting are not identical. The high frequency branch shifts inward and has a broader radial distribution.

Mode structure

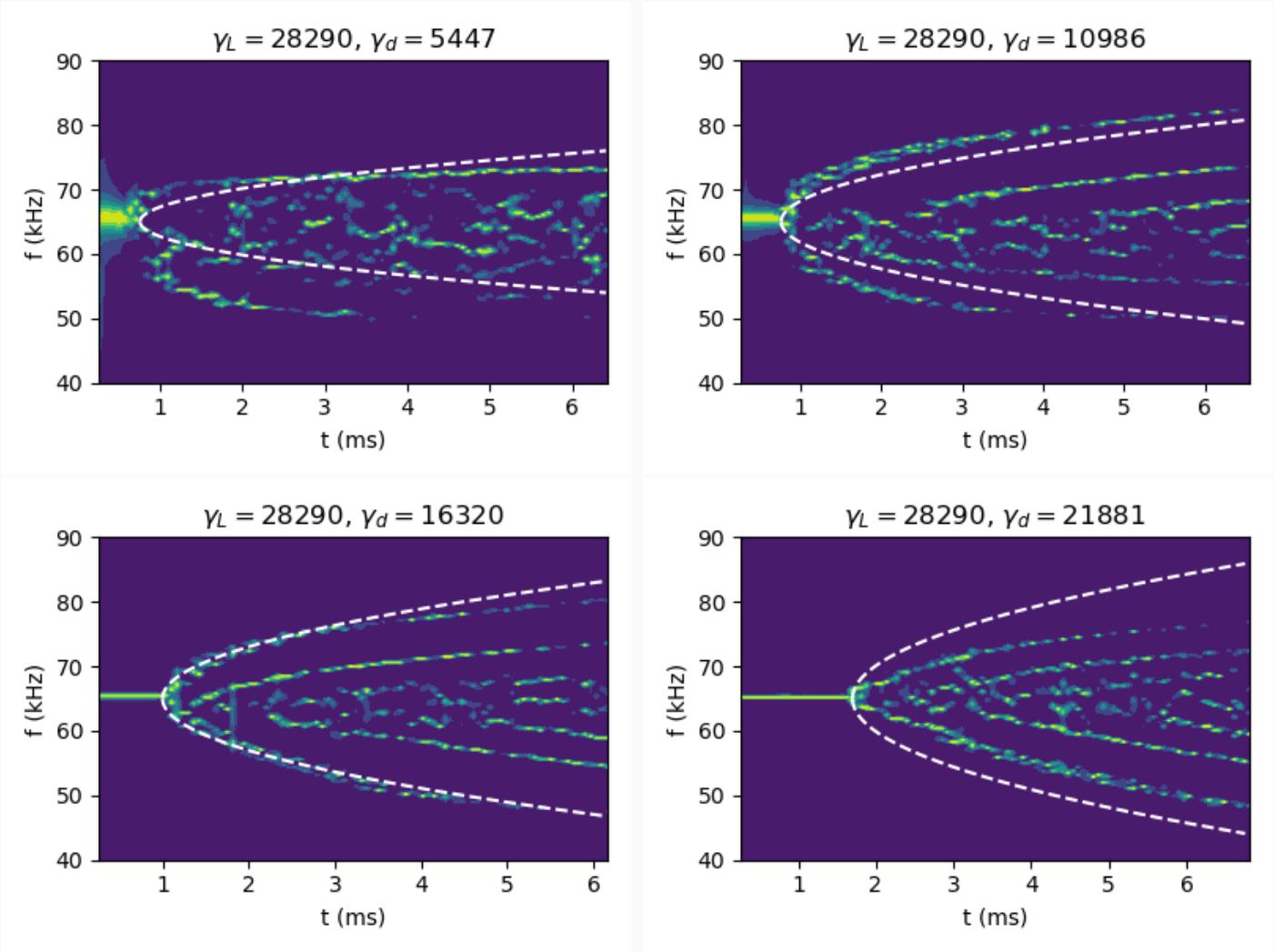






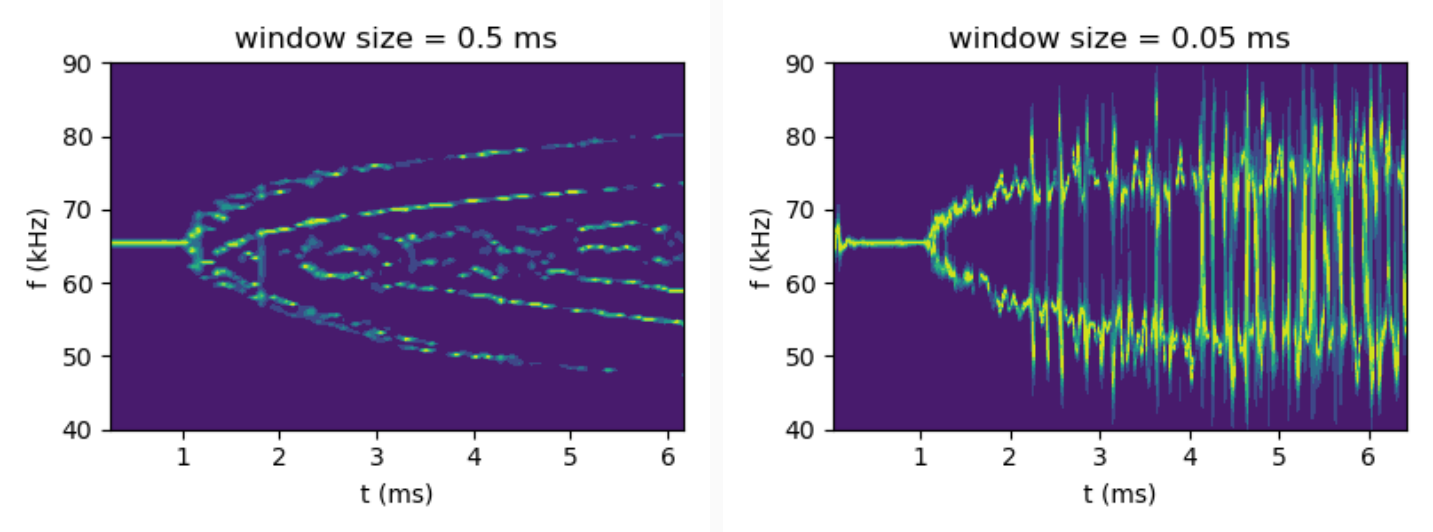


Frequency chirping for marginal and unmarginal cases



- The chirping rate is consistent with Berk-Breizman chirping rate for unmarginal case, but for marginal case ($\gamma_d \sim \gamma$), the chirping rate is smaller than the theory predicts.

Using different time window to analyze chirping at different timescales



- By using a smaller time window in DMUSIC analysis, we can analyze the fast chirping behavior for each frequency band.
  - This may be caused by particle motion which has a smaller timescale.

Outline

1. Implementation of particle pushing and MHD coupling

2. GPU acceleration of particle pushing

3. Simulation results and benchmark with other codes

4. Study of Alfvén eigenmode frequency chirping

5. Summary

Summary

- A new kinetic module coupled to M3D-C1 has been developed, which uses slow manifold method for particle pushing, and can do both pressure coupling and current coupling.
- Particle pushing code has been migrated to GPU.
- Linear and nonlinear benchmarks with other MHD and gyrokinetic codes are conducted, and good agreements are achieved.
  - The test of energy conservation is also successful in the linear stage.
- Frequency chirping of AEs can be simulated by combining a linear MHD simulation with a nonlinear particle simulation, which can save computation time and gives a reasonable result.
  - The structures of up and down chirping modes are different, which can lead to asymmetric chirping observed in experiments.

Future work

- Study linear and nonlinear evolution of kink and tearing mode and interaction with energetic particles in NSTX and DIII-D.
- Try to understand the discrepancy of chirping rate between theory and simulation for the marginal case.
- Optimize the code for long time nonlinear simulation for Alfvén mode avalanche.
- Use the same code base to study the runaway electrons interacting with MHD instabilities.