Center for Tokamak Transient Simulations
FY2019 Progress Report
S. C. Jardin, PI

Table of Contents

0.0 Project Overview

The Center for Tokamak Transient Simulations is primarily focused on the further development and application of two of the world's most capable extended MHD codes, NIMROD and M3D-C1. These are time-dependent 3D MHD codes that can accurately describe the time evolution of transient event in tokamaks. Some of these transient events such as the seeding of a neoclassical tearing mode (NTM), and its subsequent locking and growing, can lead to a plasma disruption, a potentially catastrophic event for a burning plasma. For this reason, the emphasis of our Center is on better understanding the causes of disruptions, how to avoid them, and how to mitigate them. As part of the mitigation studies, we also seek to predict the consequences of a worst case non-mitigated disruption.

Section 1 describes our efforts in NTMs with the long term goal of better understanding how and when they are formed, under what conditions they will lock and grow, and how this leads to a disruption and how best to prevent it. Since NIMROD and M3D-C1 are fluid codes, and some intrinsic kinetic effects are necessary to fully describe the evolution of a NTM, we have a parallel effort, primarily by Utah State U., to include neoclassical kinetic effects into the fluid codes.

One of the most damaging types of disruption is the vertical displacement event (VDE), discussed in Section 2, where vertical position control is lost and the plasma column firsts moves vertically and then disrupts. We are the first team to attempt to predict the consequences of these VDEs in ITER by using a fully 3D plasma model and a realistic model for the ITER vessel. We are proceeding through steps of first 2D and then 3D modeling and code verification and validation to gain confidence in our results. Again, the goal is to better understand the consequences of VDEs in order to effectively mitigate them.

Section 3 describes our efforts in modeling disruption mitigation by impurity pellet injection. We are approaching this using two methods: stand-alone modeling with NIMROD and/or M3D-C1, and code-coupling modeling where we couple one of these global codes to a code, either Frontier or LPC, that calculates in detail the local ablation physics of the pellet. These are being applied to both solid pellets, and to shattered pellets where you have hundreds of pellet fragments entering the plasma.

It is well known that high energy "runaway" electrons are generated during a disruption. Our efforts to describe these and to couple them to the bulk MHD physics are described in Section 4. We now have relatively crude models of runaway electrons implemented in NIMROD and M3D-C1 but have plans to improve the fidelity of these models by coupling with intrinsic kinetic models.

Our efforts in experimental validation are described in Section 5. This is of utmost importance to test the validity and limitations of our models. None of the projections of our models to ITER will be taken seriously unless we can show that they can describe similar phenomena in today's experiments.

In section 6 we describe our efforts in improving the performance of our codes on some of the largest, most powerful computer in the world. This improvement comes through using improved algorithms, improved data structures, and in finding ways to increase the parallelism of the algorithms that we use. Many of the improvements made by the Computer Science side of our Center will also benefit other applications outside of fusion that have sparse matrix problems similar in form to ours.

1.0 Neoclassical Tearing Modes

A survey of disruptions in JET [de Vries, et al., Nucl. Fusion 51 053018] found that the main root cause of JET disruptions was due to neoclassical tearing modes that locked. Our goal is develop improved predictive capability that allow us to predict when a NTM will form and lock, and how to prevent it from causing a disruption.

1.1 NTM Studies with NIMROD

TechX

Typically NTMs are "born" rotating in the plasma fluid frame of reference. Disruptions occur after the NTM braks and "locks" (magnetic field penetration through the conductor) to the wall. As such, it is of interest to establish both the physics basis of this process on present devices, such as DIII-D, and the scaling of the torques responsible for this locking to planned larger devices such as ITER. We are working with the group at Utah State University to use the drift-kinetic closures to simulate the neoclassical aspect of NTMs. We anticipate that these 5D computations will be computationally expensive and as such we have implemented the heuristic fluid closures [Gianakon et al., Phys. Plasmas 2002] into NIMROD to allow for parameter space exploration.

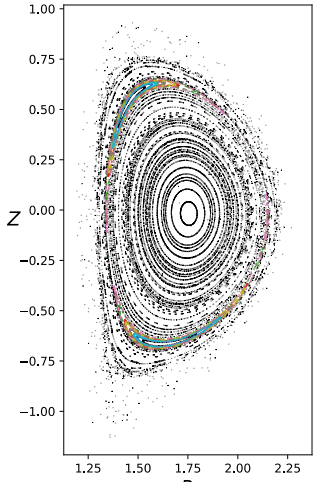

Figure 1 Example 2/1 island generated by applying a magnetic perturbation pulse at the computational boundary.

We've identified DIII-D shot 174446 as a good discharge from modeling. Around 3390ms into this discharge a edge localized mode triggers an NTM which quickly grows to large amplitude, locks to the wall, and triggers a disruption. This discharge was part of a run campaign to study NTM seeding. It has good diagnostic measurements that are focused around the location of the NTM. The discharge does not have any feedback control on the NTM. These characteristics make the case a good case for validation. Work is underway to explore the dynamics of the NTM seeding on DIIID shot 174446 in collaboration with Jim Callen, Rob LaHaye and Bob Wilcox.

Neoclassical tearing modes are metastable states, they are linearly stable and a small mode will decay, but a sufficiently large island is nonlinearly unstable and will grow. The generation of the seed island was an unexpected computational challenge. To solve this problem, we've developed a method to generate the seed island by applying a magnetic perturbation pulse at the computational boundary. This pulse generates an island via forced magnetic reconnection and we've demonstrated that we can generate large 2/1 islands using this technique (see Figure 1).

In order to build the capability to run simulations with boundary perturbations we are collaborating with Dmitiri Orlov, Rick Moyer and Todd Evans on simulating the width of the magnetic footprint during DIII-D discharges with resonant magnetic field perturbations (RMP). In particular it is hypothesized that nonlinear modeling is required to resolve discrepancies between linear modeling and experimental observations in terms of the magnetic footprint width. They have provided us with a reconstruction from DIII-D shot 166439 and with a calculated field perturbation on the NIMROD computational domain

boundary. Running a nonlinear MHD simulation with a static perturbation is a first step towards simulations that include both nonlinear NTM dynamics and field-error perturbations.

Previously, work to implement a thin resistive wall model in NIMROD by coupling to a boundary element response matrix computed from the GRIN code showed poor convergence with toroidal configurations [Becerra, master's thesis 2015]. The same numerical implementation works well in a cylindrical case where the response matrix may be specified analytically. As such, the accuracy of the response matrix calculation is suspected to be the cause of poor convergence. Through a subcontract to Tibbar Tech, a new boundary element solver, NIMbnd, was developed to circumvent these issues. This solver uses a Nystrom (collocation) method described by Young & Martinsson [arXiv:1002.2001v1] to obtain highly accurate solutions. Comparison of solutions obtained with specified normal magnetic field with a manufactured solution show relative rms errors in the range of 10^-10. Convergence studies of toroidal NIMROD calculation demonstrate high-order convergence with NIMbnd (while showing GRIN computations to not possess this property). Ultimately the resistive-wall effort and the DIII-D NTM and RMP studies will be coupled into a single focused effort which permits study of torque scaling on future devices. Each component has been treated separately in the past. In preparation for integrated modeling using a resistive wall with static error fields we've refactored the static error field implementation in NIMROD. The prior implementation prevented using simultaneous use of both boundary conditions.

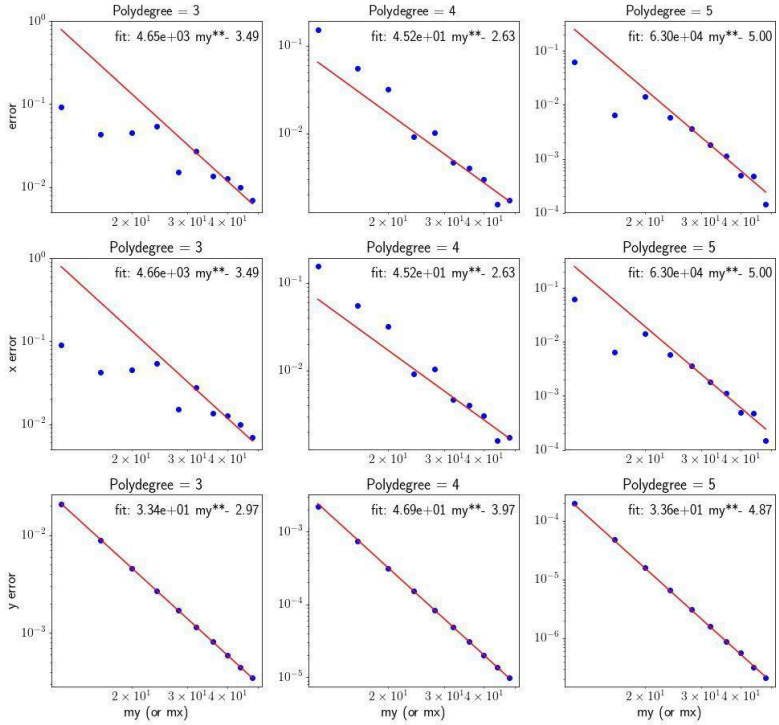

Figure 2 NIMbnd convergence studies show high-order convergence that increases with the polynomial degree of the finite element trial functions. The rate of convergence is limited by the error in the x (radial direction). This is a result of the model equilibrium which has a sharp gradient in the radial direction.

1.2 Mode Locking

U. Wisconsin

Ping Zhu's group is investigating the locked mode threshold using an error field model for resonant magnetic perturbation (RMP) interaction with magnetic islands in cylindrical geometry. They find the locked mode threshold for the level of toroidal rotation in the China Fusion Engineering Test Reactor (CFETR) design to be consistent, in order of magnitude, with empirical and theoretical scaling law predictions. Assessment of the ECCD power required from the control of TM/NTM has recently been started using NIMROD. Previous simulations have found NTM stabilization, either by a simplified ECCD model or by an ECCD distribution from GENRAY calculations. Tests of this ECCD module are in progress, and it is expected to be applied to CFETR analysis in the near future. In collaboration with L.-J. Zheng, Zhu has also contributed to a study of resistive wall mode (RWM) stability for CFETR.

1.3 Simultaneous advance of temperature and drift kinetic equation in NIMROD for use in NTM simulations

Utah State U.

One method of including kinetic effects in NIMROD is the Chapman-Enskog-Like (CEL) method . This is a continuum kinetic approach that partitions the distribution function into the form

$$f(x,v,t) = F(x,v,t) + f^M(x,v,t),$$

which self-consistently separates the Maxwellian part, f^M, parameterized by the density, n, macroscopic flow velocity, V, and temperature, T, from the kinetic distortion, F. The fluid quantities evolve according to the fluid equations originally implemented in NIMROD with closures provided by moments of the kinetic distortion, which evolves according to the CEL kinetic equation. That is to say, the kinetic distortion is governed by a form of the kinetic equation in which the Maxwellian moments have been removed. This ensures that during the evolution of the distribution function, it is perpetually partitioned into the Maxwellian and non-Maxwellian parts. As a result, each part can have an independent numerical treatment.

As a first step to using the CEL method to study kinetic effects on neoclassical tearing modes, we constructed a simplified model of anisotropic thermal diffusion. In this model, we use a static magnetic island in slab geometry. The plasma is assumed to have a uniform number density and the flow velocity and traceless pressure tensor are neglected. The parallel heat flux is computed from the parallel energy flow moment of the kinetic distortion. This model can be used to simulate the rapid temperature flattening across a slowly evolving magnetic island. The temperature equation for this model is

$$\frac{3}{2}n\frac{\partial T}{\partial t} = \kappa_\perp \nabla \cdot [(I - bb) \cdot \nabla T] - \nabla \cdot q_\parallel$$

where $\kappa_\perp$ is Braginskii's perpendicular thermal conductivity and the parallel heat flux is given by

$$q_\parallel = \frac{m}{2}\int dv v^2 v_\parallel F.$$

The kinetic distortion is governed by the CEL drift-kinetic equation CEL-DKE

$$\frac{\partial F}{\partial t} + v \cdot \nabla F - \frac{1-\xi^2}{2\xi} v_\parallel \cdot \nabla lnB \frac{\partial F}{\partial \xi} - \frac{s}{2}\left(v_\parallel \cdot \nabla + \frac{\partial}{\partial t}\right) lnT \frac{\partial F}{\partial s}$$
$$C + \left(\frac{5}{2} - s^2\right) v_\parallel \cdot \nabla lnT f^M + \frac{2}{3nT}\left(s^2 - \frac{3}{2}\right)(\nabla \cdot q_\parallel) f^M$$

Here we introduce normalized speed and pitch-angle-like coordinates,

$$s = |v|/v_T$$

$$\xi = v \cdot B/|v|\,|B|$$

These normalizations are a reasonable way to ensure the velocity domain is large enough to support the entire kinetic distortion. The first three terms on the left side of the CEL-DKE are the time derivative, free streaming and bounce terms, respectively. The last term on the left side is needed to scale the velocity domain as the temperature changes due to the speed coordinate being normalized by the evolving thermal speed. On the right side, we have the full, linearized Coulomb collision operator, and Maxwellian drives which result from projecting out the temperature moment from the kinetic equation. For comparison, Braginskii's closure for the parallel heat flux is

$$q_\parallel = -\kappa_\parallel (b \cdot \nabla T) b$$

where $\kappa_\parallel$ is Braginskii's parallel thermal conductivity.

The time scale of NTM dynamics preceding a disruption can be on the order of hundreds of milliseconds, whereas the electron transit time is on the order of a microsecond. Therefore, we seek an implicit time discretization scheme that allows large time steps to be taken. However, the CEL-DKE shows a strong nonlinear coupling of the temperature and kinetic distortion. This arises not only through the collision operator and parallel temperature gradient drive term, but also is due to the thermal speed normalization of the velocity space. Nonlinear coupling also enters through the heat flux, again from the velocity normalization. Therefore, significant coding was needed to advance the temperature and kinetic distortion simultaneously and implicitly. We will now describe how we implemented an implicit simultaneous advance of the temperature and kinetic distortion. Following this, we will describe our method of solving the nonlinear system.

The CEL-DKE is solved at a set of collocation points in speed, $F_i(x, \xi, t) = F(x, s_i, \xi, t)$. These points are chosen to correspond to a set of quadrature nodes used to construct the moments of the kinetic distortion used in the Maxwellian drives and to close the fluid equations. We are able to use either a spectral method or the Finite Element Method (FEM) to solve for the pitch-angle dependence of the kinetic distortion. The spectral method uses Legendre polynomials, eigenfunctions of the pitch-angle scattering operator, as a set of basis functions for our solution space. The FEM uses Lagrange or Gauss-Legendre-Lobatto (GLL) polynomials. In either case we can expand the kinetic distortion in terms of the pitch-angle basis functions, $P_j(\xi)$, as

$$F_i(x,\xi,t) = \sum_{j=0}^{dof} F_{i,j}\,(x,t)P_j(\xi)$$

where dof is the number of degrees of freedom in pitch-angle. The kinetic distortion's spatial dependent coefficients and temperature can be expanded in terms of NIMROD's usual 2D finite elements and Fourier modes ,

$$F_{i,j}(R,Z,\phi,t) = \sum_{k=0}^{N}\left\{F_{i,j,k,l=0}(t) + 2\Re e\left[\sum_{l>0} F_{i,j,k,l}\,(t)e^{il\phi}\right]\right\}\alpha_k(R,Z)$$

and

$$T(R,Z,\phi,t) = \sum_{k=1}^{N}\left\{T_{k,l=0}(t) + 2\Re e\left[\sum_{l>0} T_{k,l}\,(t)e^{il\phi}\right]\right\}\alpha_k(R,Z)$$

where ϕ is the toroidal angle, and $\alpha_k(R,Z)$ are N finite element basis functions over the poloidal plane.

Since the number of degrees of freedom in pitch-angle is almost always chosen to be greater than the number of fluid quantities evolved in NIMROD, we combine fluid quantities with the kinetic distortion conveniently by extending the kinetic distortion solution array by one virtual speed point, s_0, and allowing the fluid quantities to occupy some of the corresponding pitch-angle degrees of freedom. For example, the temperature occupies the first degree of freedom of the virtual speed point, $T(R,Z,\phi,t) = F_{0,0}(R,Z,\phi,t)$. The remaining degrees of freedom for the virtual speed point are occupied by solutions to a set of trivial equations that are easily solved but have no physical meaning. We have pursued a distributed memory parallelism scheme distributing the speed collocation points using copies of the parallelism normally used in NIMROD to distribute the spatial domain. Combining the temperature with the kinetic distortion by adding a virtual collocation point has been met with the challenge of effectively preconditioning the linear system complicated by the fact the temperature now couples to the rest of the distribution function through the heat flux, the Maxwellian drives, and the couplings due to our speed normalization. Effective preconditioning is a matter of ongoing research, but preliminary results indicate that effective preconditioning can be achieved without inverting a large matrix that couples all the speed collocation points.

1.3.1 Nonlinear solution

In order to solve the highly nonlinear system of equations, T evolution coupled to the CEL-DKE, we have implemented Newton's method. For time discretization we use a theta-centered semi-implicit finite difference scheme. This involves the solution of nonlinear algebraic equations. If we use a superscript, n, to denote the time step, t_n, then the solution vector is the set of unknown coefficients, $F_{i,j,k,l}^{n+1} \equiv F_{i,j,k,l}(t_{n+1})$, where the $i = 0$ and $j = 0$ coefficients correspond to the coefficients in the temperature expansion. If we use vector notation to represent the solution vector, F^{n+1}, whose components are $F_{i,j,k,l}^{n+1}$, then the coupled system, discretized using the finite element/Fourier/collocation method previously described, lead to a set of nonlinear algebraic equations that can be written in the form

$$A(F^{n+1}, F^n) = b(F^n)$$

where A and b are nonlinear operators. Newton's method is an iterative method, and we label iterates with a second superscript, $F^{n+1,m}$. The converged solution is $F^{n+1,\infty}$. Newton's method applied to our set of nonlinear differential equations solves the following system of linear equations,

$$\left.\frac{\partial A(F, F^{n,\infty})}{\partial F}\right|_{F=F^{n+1,m}} \cdot (F^{n+1,m+1} - F^{n+1,m}) = b(F^{n,\infty}) - A(F^{n+1,m}, F^{n,\infty})$$

where $\partial A\,(F, F^{n,\infty})/\partial\, F|_{F=F^{n+1,m}}$ is the Jacobian matrix of A. The left side of Eq. ([eq:Newton]) is the linearized form of the nonlinear operator, A. This linear system is solved using the matrix-free Generalized Minimum Residuals method normally used by NIMROD. This means the full Jacobian matrix is never explicitly formed. All that is needed is that *action* of the Jacobian matrix on the vector of unknowns. The iterative loop envelopes the calls to the linear solver. Within the iterative loop and outside the linear solver are calls to subroutines that update the right side of Eq. ([eq:Newton]) and pre-computed quantities needed to form the action of the Jacobian matrix, such as collision operator matrices that couple degrees of freedom within the velocity space representation for the kinetic distortion. A preconditioning matrix is computed at most once per time step, but can often be reused over multiple time steps.

For research purposes, to find a sufficient preconditioner, a subroutine that forms the full Jacobian matrix was created. This was not intended for production runs, but is used to see which elements of the matrix can be dropped and still produce an effective preconditioner. Using this technique we have preliminary results that indicate it may be sufficient to compute submatrices that couple the temperature and kinetic distortion together, while ignoring the coupling between speed collocation points that come from the collision operator and speed derivative in the CEL-DKE. This allows us to utilize the parallelism over the speed collocation points in computing the preconditioner.

The table below compares the previously implemented preconditioning strategy, block Jacobi, with one constructed by looking at the effect of dropping specific terms from the full Jacobian matrix.

Total GMRES iterations on 1st Newton iteration

Block Jacobi	2328
New Approach	590

Both preconditioning strategies rely on breaking the full matrix into blocks. Each block is specified by the speed collocation points that are coupled by it. The block Jacobi method drops all terms in the preconditioner that would couple different speed collocation points, leaving only the blocks diagonal-in-speed to be computed and factored. The new strategy drops these terms in the CEL-DKE, but retains the coupling between temperature and kinetic distortion that comes through the heat flux in the temperature equation. This method requires the processors assigned to a single speed collocation point to calculate two blocks instead of one, yet only factors the diagonal-in-speed block.

Implementing the simultaneous advance of temperature and kinetic distortion required some unfamiliar linearization of the operator A. For example, temperature derivatives must be taken of the kinetic equation and kinetic distortion derivatives must also be taken of the heat flux in the temperature equation. Without these linearizations Newton's method fails to converge because the resulting linear system is not a true linearization of the nonlinear system. As an example of the more complex linearizations, consider the temperature derivative of the collision operator. The temperature derivative of the test particle collision operator, $C[F, f^M]$, is done simply by taking a temperature derivative of the collision frequency, which is a coefficient of the collision operator. This was easily done analytically. However, the temperature derivative of the field operator, in particular the collisions of the background electron Maxwellian off the ion kinetic distortion, $C[f_e^M, F_i]$, is more involved due to temperature ratios that appear in evaluating the electron distribution function at speeds normalized by the ion thermal speed, $s_e = v_{Ti}\, s_i / v_{Te}$. To avoid these complications and potential bugs, the temperature derivative of the field operator is performed numerically rather than analytically. This was very easy to implement, and as a check we were able to verify its implementation by comparing numerical derivatives of the test particle operator to analytic derivatives. Other linearizations still remain for terms, like the collisional friction, that will be introduced in the future as we include more physical effects and thermodynamic drives. We plan to adopt numerical derivatives for these terms as well.

13.2 Improved computation of the Rosenbluth potentials

Although we were able to obtain preliminary results for the anisotropic thermal conduction problem described above, we noticed that improvements were needed for the previous implementation of the Rosenbluth potentials in the field portion of the collision operator, $C[f^M, F]$. The original method, which computed the Rosenbluth potentials at the nodes of the finite element basis in pitch-angle, was only modestly accurate and did not preserve the symmetries that would be present in a least-squares formulation. Errors were noticed by examining the symmetry. We have made an effort to improve accuracy and preserve symmetry by numerically integrating over singularities in the Green's functions of the velocity space Poisson equation. We are currently making progress on two approaches to increase the accuracy of the Rosenbluth potentials: 1) a modified quadrature scheme that is adapted to the singularities, and 2) a numerical splitting technique taken from the astrophysics community where gravitational potential from sources with symmetry are computed. The second method treats the Rosenbluth potentials in an analogous fashion to mass distributions in astrophysical systems. In order to obtain trusted values, we have computed test integrals using a third party adaptive numerical integrator . This has facilitated the rapid development of more accurate alternative approaches.

2.0 Vertical Displacement Events

A Vertical Displacement Event (VDE) is an off-normal occurrence in a tokamak in which position control of the discharge is lost, and the tokamak plasma moves rapidly upward or downward until it makes contact with the vacuum vessel. The discharge current in ITER will be up to 15 MA. When a plasma with this current makes contact with the vessel, it will induce large currents into the metallic vessel, and these currents will cause large forces. Previous studies commissioned by ITER to calculate these forces

assumed that the plasma remained axisymmetric during the VDE to simplify the calculation. However, it is known that the plasma column will deform and produce "sideways forces" in ITER that could potentially damage the machine. Our two flagship MHD codes, NIMROD and M3D-C1 now have the capability of modeling a fully 3D plasma interacting with a conducting structure. We are using this capability to realistically model a full 3D VDE in ITER and to calculate the expected forces.

2.1 Benchmarking Activity

PPPL (lead), U. Wisconsin

A benchmark exercise for the modeling of vertical displacement events (VDEs) was applied to the 3D nonlinear magneto-hydrodynamic codes M3D-C1, JOREK and NIMROD. The simulations are based on a vertically unstable NSTX equilibrium enclosed by an axisymmetric resistive wall with rectangular cross section. A linear dependence of the linear VDE growth rates on the resistivity of the wall is recovered for sufficiently large wall

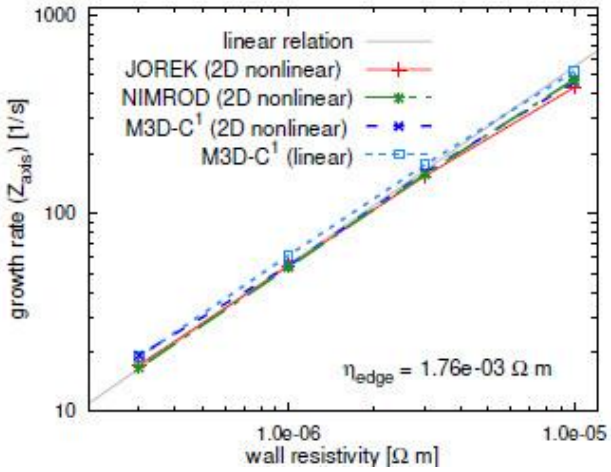

Figure 3 Comparison of VDE growth rates from the linear phase of 2D nonlinear M3D-C1, NIMROD and JOREK simulations. They deviate between 3% and 15%. Also shown are the results of linear M3D-C1 calculations.

conductivity and small temperatures in the open field line region. The benchmark results show good agreement between the VDE growth rates obtained from linear NIMROD and M3D-C1 simulations as well as from the linear phase of axisymmetric nonlinear JOREK, NIMROD and M3D-C1 simulations. Axisymmetric nonlinear simulations of a full VDE performed with the three codes are compared and excellent agreement is found regarding plasma location and plasma currents as well as eddy and halo currents in the wall. We show in Figures 3 (linear) and 4 (nonlinear) the results of the benchmark problem for the 3 codes. This study has been submitted for publication in Physics of Plasmas.

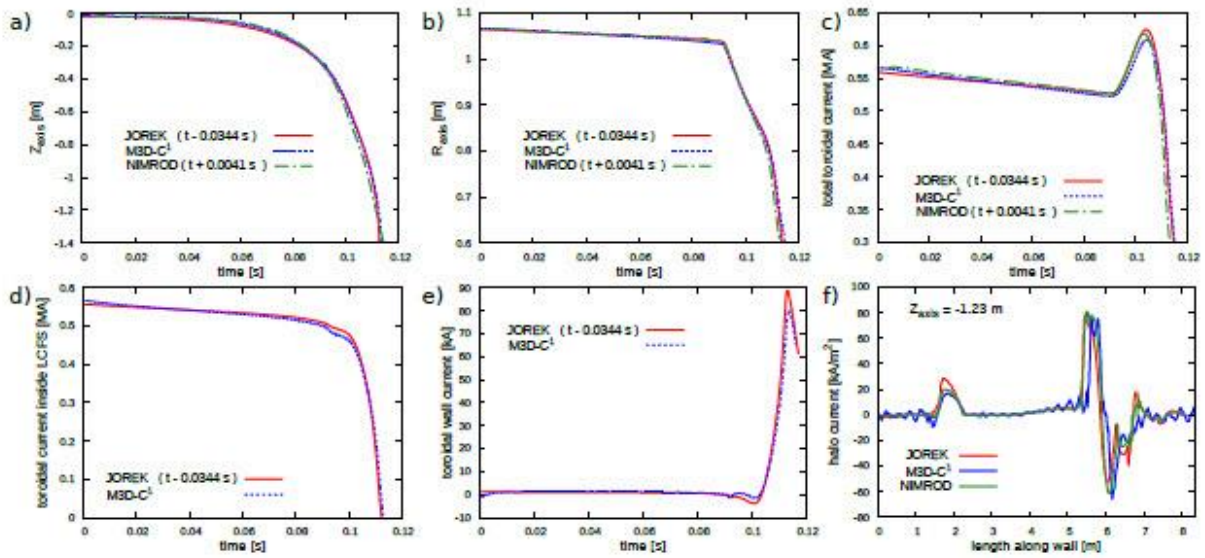

Figure 4 Comparison of time traces from a 2D nonlinear simulation performed with JOREK, NIMROD and M3D-C1: a) vertical position of magnetic axis, b) radial position of magnetic axis, c) toroidal current inside the LCFS and the open field line region, d) toroidal current inside the LCFS, e) net toroidal wall current. JOREK and NIMROD time traces are shifted so that the points in time of the first plasma-wall contact coincide. f) shows the component of the current density that is normal to the wall traced along the length along the wall at the point in time when Zaxis = -1.23 m. The trace starts at the low-field side mid-plane and continues counterclockwise.

2.2 Pre-sheath boundary conditions

U. Wisconsin

The boundary-condition development is work done by graduate student Kyle Bunkers. He adapted the magnetized sheath boundary conditions from Loizu, et al., [Phys. Plasmas 19, 122307 (2012)] for non-reduced equations, as used by NIMROD. The lowest order modeling has flow velocity at the problem-domain surface, which is considered the entrance to the magnetic presheath, being parallel to magnetic field and at the Bohm speed. Separate electron-temperature evolution with thermally insulated electrons is also part of the model. Bunkers attempted to use an edge-local nonlinear resistivity to limit current density to the ion saturation current, but this approach has not yet proven satisfactory. His axisymmetric computations with magnetic presheath boundaries reveal sensitivity to and synergy with the modeling of temperature. If the temperature modeling is simplified to fixed cold surface temperature-values, instead of insulating the electrons, then the computations are not sensitive to the conditions on flow. That is because the Chodura-Bohm speed is small at low temperature, which throttles outflow to the same degree as setting edge flow to zero. However, having insulating conditions on electrons allows parallel thermal conduction to heat open-field electrons, and as shown in Fig. 5, the Chodura-Bohm outflow increases the rate of both the thermal quench (TQ) and the current quench (CQ). Comparing computations with Braginskii thermal conduction to those with fixed anisotropic conductivities (Fig. 6) shows that the larger perpendicular conduction with the Braginskii

model broadens the electron temperature and electrical conductivity profiles and lengthens the CQ. More details are available in Bunkers' PhD dissertation.

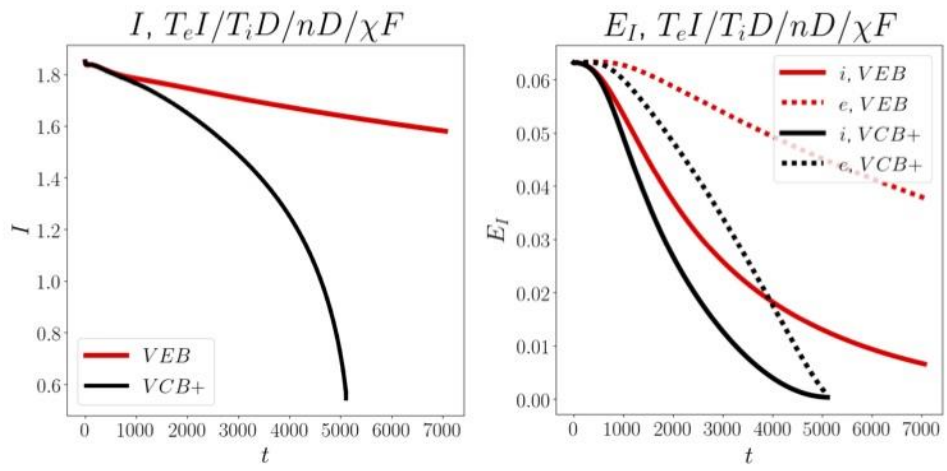

Figure 5. Evolution of plasma current (left) and internal energy (right) in normalized units from axisymmetric two-temperature computations with Chodura-Bohm (VCB+) and ExB (VEB) outflow conditions. Both computations have fixed anisotropic thermal conductivities. [K. J. Bunkers, "The influence of boundary conditions on vertical displacement events," PhD dissertation, UW-Madison (2019)]

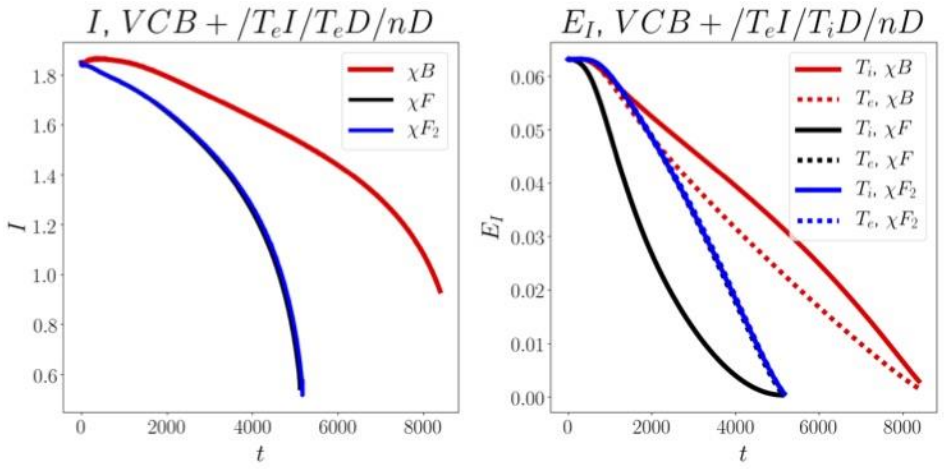

Figure 6. Evolution of plasma current (left) and internal energy (right) in normalized units from axisymmetric two-temperature computations with Braginskii (χB) and fixed (χF, χF_2) thermal conductivity. All three computations have Chodura-Bohm boundary conditions. [K. J. Bunkers, "The influence of boundary conditions on vertical displacement events," PhD dissertation, UW-Madison (2019)]

2.3 Application to ITER

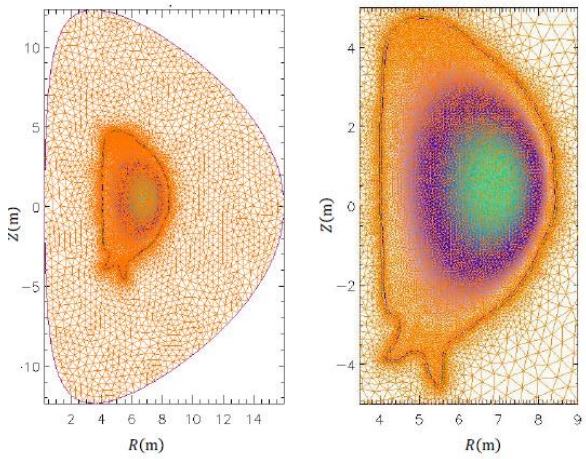

Figure 7: *Mesh and domain used in the ITER simulations.*

To better understand the potential magnitude of the associated forces and the role of the so called "halo currents" on them, we have used the M3D-C1 code to simulate potential VDEs in ITER. We used actual values for the vessel resistivity and pre- quench temperatures and, unlike most of the previous

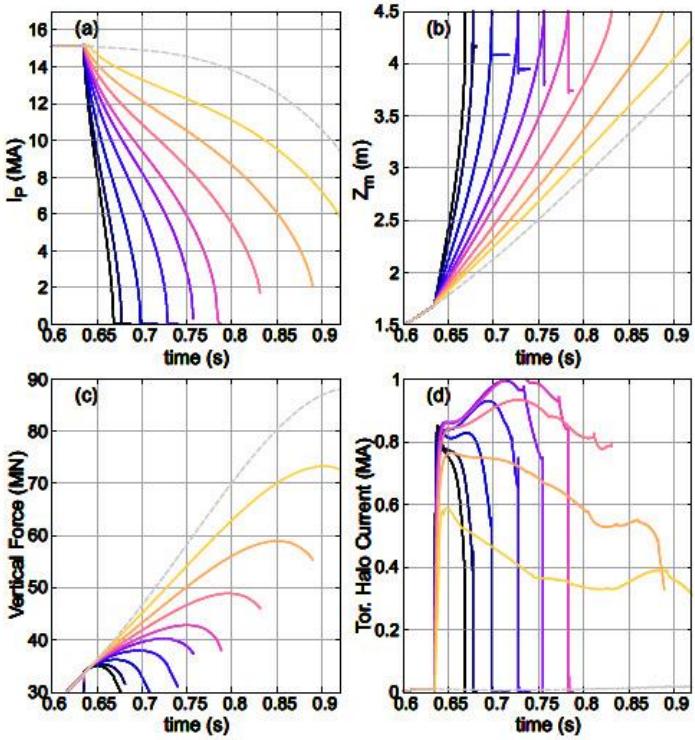

Figure 8 (a) Plasma current, (b) magnetic drift, (c) total vertical force on the wall and (d) toroidal halo current as a function of time, for different post-TQ thermal conductivity values. For temperature higher than 70 eV, the halo current decreases. The grey curve corresponds to a case without a TQ.

studies, the halo region is naturally formed by triggering the thermal quench with an increase in the plasma thermal conductivity. We used the 2D non-linear version of the code and vary the post-thermal quench thermal conductivity profile as well as the boundary temperature in order to generate a wide range of possible cases that could occur in the experiment. We also show that, for a similar condition, increasing the halo current does not increase the total force on the wall since it is offset by a decrease in the toroidal contribution.

We show in Figure 7 the mesh used in the M3D-C1 calculation and in Figure 8 the results of several calculations where different values of thermal conductivity during the thermal quench were assumed. This work has been accepted for publication in [Clauser, et al, Nuclear Fusion (2019)]

3.0 Disruption Mitigation

Disruptions are best avoided, but if this is impossible, it is essential to have a disruption mitigation system that can be fired when a disruption is imminent. The present plan is for ITER to have a shattered pellet injection (SPI) system using a noble gas such as Neon or Argon. However, it is not certain that such a system will be effective and our goal is to perform detailed modeling of SPI mitigation, show that we can get agreement with experiments on today's machines, and project to ITER. If SPI cannot be shown to be effective in ITER, there are other proposals such as Electromagnetic Pellet Injection (EPI) that can be considered.

3.1 NIMROD SPI Disruption Mitigation Development

General Atomics

We continue to improve the NIMROD particle-based SPI model and apply it to refine NIMROD DIII-D SPI simulations of the thermal quench. Higher resolution simulations with more toroidal modes have allowed more local deposition of the ablated neutrals. The Spitzer resistivity and Braginskii temperature-dependent thermal conduction have been scaled to more accurately capture the experimental range at lower temperatures (T<500eV), providing greater physics fidelity to the NIMROD SPI thermal quench simulations.

As we approach greater physics fidelity, we have identified a primary cause of termination in the NIMROD SPI simulations as related to nonlinear resistive instabilities of large amplitude, resulting in complex 3D fields that become difficult to resolve for the anisotropic thermal conduction. Enhanced resistivity models (e.g. hyper resistivity) and numeric techniques (dissipation and increased resolution) are being applied to overcome this issue.

The NIMROD particle SPI model was implemented with the following approximations:

- fragments are discrete and small in number
- fragments do not interact with each other
- details of fragment geometry can be neglected
- details of the ablation cloud can be neglected

An invited presentation was given at the APS DPP Annual Meeting last fall on shattered pellet injection simulations with NIMROD. The accompanied paper was published in the April, 2019 issue of Physics of Plasmas and was selected as an Editor's Pick

3.2 M3D-C1 Disruption Mitigation

General Atomics (lead), PPPL

A new pellet-ablation model (based on work by P.B. Parks) for mixed D2-Ne pellets was added to M3D-C1, allowing for deposition of both impurities and main ions. In addition, improvements were made to the pellet model within M3D-C1, allowing for ballistic motion in Cartesian coordinates and for proper deposition of ablated material when the ablation rate changes rapidly. An initial benchmark of these new pellet capabilities was carried out between M3D-C1 and the PELLET code, showing excellent agreement in the pellet-ablation rate. Furthermore, M3D-C1 was upgraded to allow for multiple pellet sources. An arbitrary number of pellets can now be defined, each with its own location, velocity, size, and composition. This will allow for more accurate shattered-pellet-injection modeling (with multiple pellet fragments) and for the simulation of multiple toroidal injections.

3D, nonlinear M3D-C1 simulations were performed based upon the case used for the successful axisymmetric benchmark between M3D-C1 and NIMROD described in the next section. With an axisymmetric, on-axis impurity source, the plasma remains stable throughout the entire thermal quench. Eventually, a current sheet forms that goes unstable to 3D modes, broadening the current profile and causing a current spike. This broadening appears to be related to the high resistivity in the core (due to the low temperature and large number of impurities there), which may explain why current spikes of this magnitude have not been seen in past 3D MHD disruption modeling. A journal letter will be written based on these results.

Initial, 3D, nonlinear M3D-C1 simulations with nonaxisymmetric impurity sources have also been performed. Such sources rapidly induce localized 3D instabilities, which appear to mix the impurities radially. This causes a faster thermal quench as compared to simulations with axisymmetric impurity deposition (for fixed number of ions injected.

3.3 M3D-C1/NIMROD Code validation and Benchmarking

General Atomics (lead), PPPL

An axisymmetric benchmark of M3D-C1 and NIMROD impurity dynamics was completed. Beginning with an equilibrium reconstruction of DIII-D shot 137611, both codes modeled the constant injection of neutral impurities on-axis and evolved the impurity charge states along with the macroscopic MHD. The resulting dynamics of the thermal quench, radiation, and impurity transport showed excellent agreement between the two codes for both argon and neon impurities, along with constant or Spitzer-like resistivity as shown in Figs 9 and 10. This work has been published in Lyons, et al, Phys Plasma (2019).

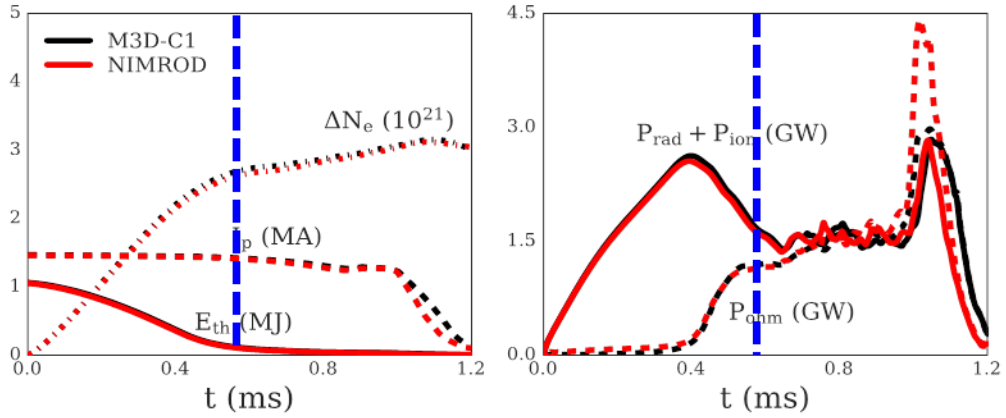

Figure 9: For argon injection with Spitzer resistivity, time histories of important 0D plasma quantities. Left: thermal energy, plasma current, and change in electron number. Right: KPRAD loss power(radiation plus ionization) and ohmic-heating power.

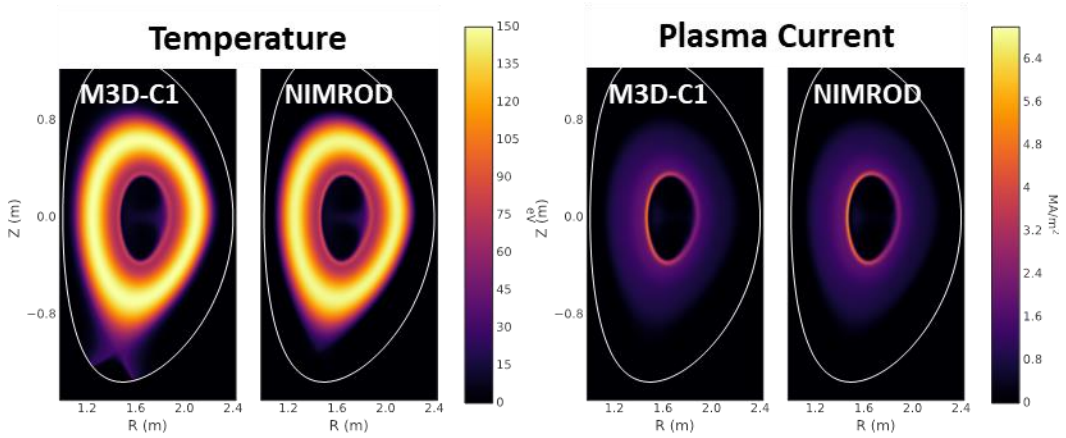

Figure 10: (left Electron temperature contours and (right) current density profiles at the time indicated with the blue dashed line in Figure 9 (t = 55 ms)

3.4 Development of Analytic Kinetic Heat Flow Models for use with NIMROD and M3D-C1

General Atomics

In the case of elastic scattering of energetic plasma electrons above ~ 10 eV with the neutral atoms in the ablation cloud, the detailed atomic structure of the bound electrons in the atom is relatively unimportant. One can simply use the Rutherford differential scattering cross section for the Coulomb field of a bare (unshielded) nucleus. The atomic structure can be approximately accounted for by simply modifying the Coulomb logarithm to reflect the minimum scattering angle permitted in the collision. Below 10 eV, and especially for Z > 6, the shielding of the nuclear charge Z by the atomic electrons

requires a modification of the electron-ion Rutherford expression. For deflections caused by inelastic collision with the Z bound electrons, the Rutherford expression is modified using an incoherent scattering function. Published tables for all elements were used to construct more accurate forms for the kinetic-based parallel heat flux of the degraded primary electrons.

3.5 Local Modeling of Pellet ablation and Coupling to MHD codes

Stony Brook U. (lead), GA, PPPL

The ablation pellets and shuttered pellet fragments in tokamaks is intrinsically a multiscale problem with spatial scales ranging from millimeters (dense clouds near cryogenic pellets) to 10x meters (expansion of ablated material along magnetic field lines), as well as multiple time scales. A two-level approach is adopted, illustrated schematically in Figure 11.

(a) Local (near-field) model resolves all relevant physics processes near the pellet such as pellet surface ablation, formation of a dense and cold ablation cloud, deposition of energy of hot electrons in the ablation cloud, heating, ionization and channeling of the ablated material along magnetic field lines by MHD forces, radiation losses, and the grad-B drift of the ablated material across magnetic fields lines. The local model describes the evolution of the process in great detail and compute pellet ablation rates. Two codes are used for near-field simulations of the pellet ablation. The 1st one is based on FronTier, a multiphysics code with explicit interface tracking. It is suitable for spherically symmetric and 2D cylindrically symmetric simulations. The second code is based on the Lagrangian particle method. It is a highly adaptive 3D code optimal for simulations of dense, cold ablated gas near the pellet and high-temperature plasma in the far field. It resolves intrinsically 3d effects such as the grad-B drift. The Lagrangian treatment of matter is also optimal for multiscale coupling with tokamak-scale MHD codes. Two codes are currently supported for the V&V purpose.

(b) Global studies have been performed using typical MHD codes (NIMROD and M3D-C1) for tokamak plasmas with the addition of analytic source terms. They need more refined model for the source terms, especially for SPI, when a large number of gas / plasma clouds, created by the ablation of pellet fragments, partially screen each other from the incoming electron heat flux. Multiscale coupling of the LP code to NIMROD and M3D-C1 has started.

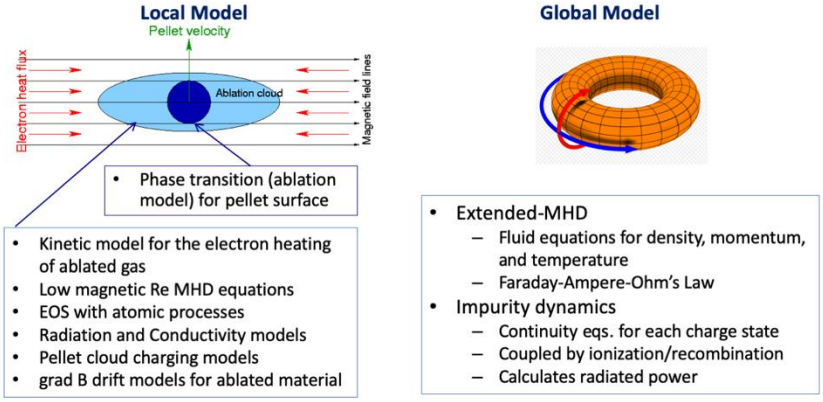

Figure 11: . Schematic of two-level approach to the ablation of pellets / SPI fragments

3.5.1 New physics models

Several new physics models have been implemented in the FT and LP codes during the past year. In particular, we implemented EOS libraries and for tabular data sets. The corresponding data sets for neon and deuterium were built using a SAHA LTE solver. We currently work on building data sets for neon-DT mixtures. We also plan to build non-LTE data sets for EOS using CRETIN calculations (with the help of Eric Hollman of GA), as non-LTE approximation is more accurate in the far field. Boundary conditions for the ablated material plasma interface were also improved. The most important new physics model recently added to the LP code is a model for the grad-B drift of the ablated material across magnetic field lines. Simulations that resolve grad B drift compute the pellet shielding length and the material deposition self-consistently. Grad-B drift is also an essential component of our multiscale coupling model.

3.5.2 Simulations of single pellets

Both FT and LP codes have been verified using theoretical predictions for scaling laws (Figure 12). Recent spherically symmetric simulations also explained the influence of atomic processes on pellet ablation rates at various plasma conditions.

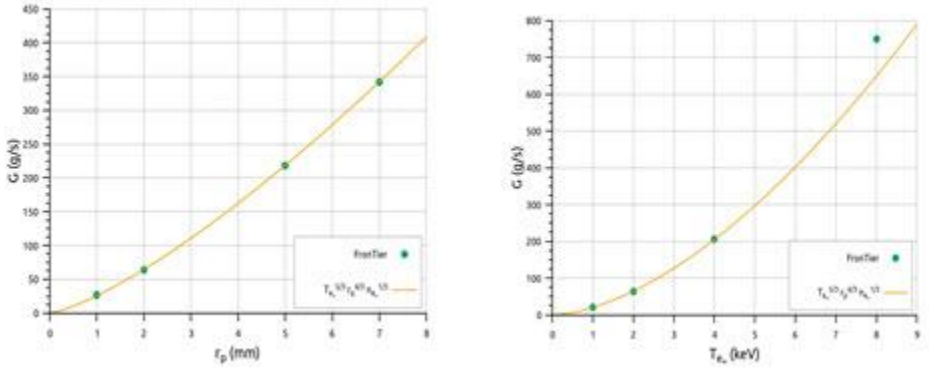

Figure 12: . Verification of theoretical scaling laws. FT simulations of the ablation rate dependence on the pellet radius (left) and plasma electron temperature (right) are shown.

A series of refined simulations have been performed to study the ablation rate of a single neon pellet in various tokamak background plasmas, with the magnetic field increasing up to 9 Tesla. Both codes showed a significant reduction of the ablation rate in magnetic fields because the magnetic field channels the ablation flow along magnetic field lines, increasing the pellet shielding. But FT code showed stronger reduction of the ablation rate compared to the LP code. We found that this was due to the fact that both codes treat the ablating pellet surface somewhat differently. Without magnetic field, FT code produces a lower-density, higher-velocity ablation flow, while the LP code obtains higher-density, lower-velocity ablation flows. As the result, the ablation rate (the mass flux) is in a very good agreement in both codes at zero magnetic field. But lower and higher density clouds react differently to high magnetic fields, resulting in some disagreements. This suggests that additional work must be performed on the surface ablation model. As we wait for experimental results operating with a single neon pellet, we started validation studies using the available data on deuterium fueling pellets.

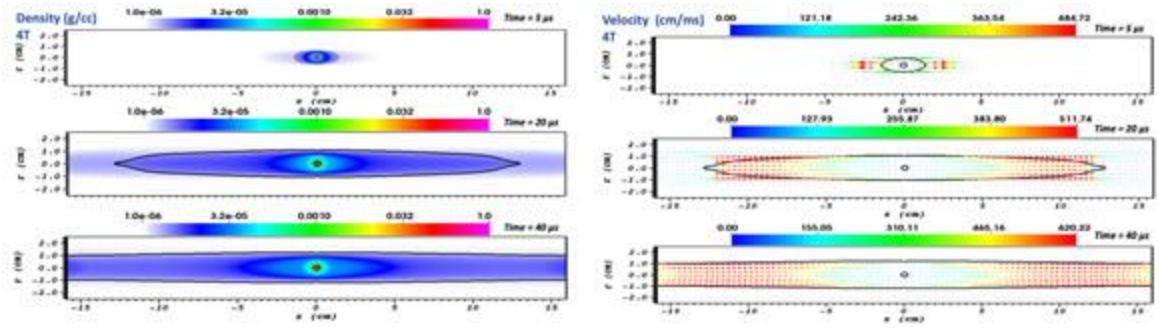

Figure 13: FronTier simulation of the ablation of neon pellet in 4T magnetic field demonstrating improved states when front tracking is used for the ablation cloud - plasma interface.

We have successfully tested the grad-B drift model in the LP code. Simulations that resolve grad B drift compute the pellet shielding length self-consistently. For 2mm neon pellet in 2T magnetic field of DIII-D geometry, the computed shielding length was 17.3 cm. This number is in a very good agreement with our previous theoretical estimates: in our previous simulations, the length was chosen in the range 16 – 18

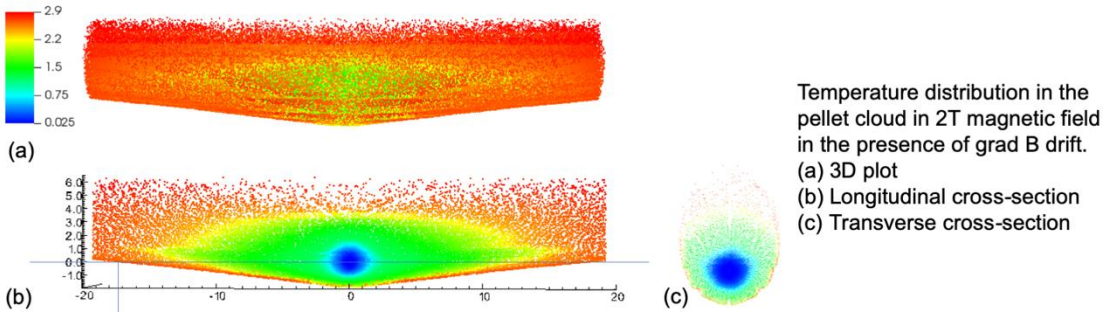

Figure 14: . Lagrangian particle simulation of the neon pellet ablation with grad-B drift in 2T magnetic field. The computed shielding length was 17.3 cm

3.5.3 Simulations of SPI Fragments

We have performed initial simulations of SPI fragments at conditions relevant to DIII-D experiments. To study the fragment interaction problem, we started simulations using only two fragments in two typical configurations: separated in the direction across the field and along a single magnetic field line. As small SPI fragments are separated by distances of order 4 cm in the transverse direction, simulations without grad-B drift did not observe any interaction of fragments. In the presence of grad B drift, ablation clouds of fragments interact with each other, as shown in Figure 15. The ablation rate of the top fragment is reduced by 9% compared to the bottom one. The effect will be much stronger for large number of fragments. Images on the right show SPI fragments located on the same magnetic field line. (c): no grad B drift, (d): grad B drift included. For .66 mm fragments, the top fragment ablation rate is reduced by ~ 9 % (a-b). (c): 18% reduction compared to single fragment, (d) 15% reduction compared to single fragment (grad B drift)

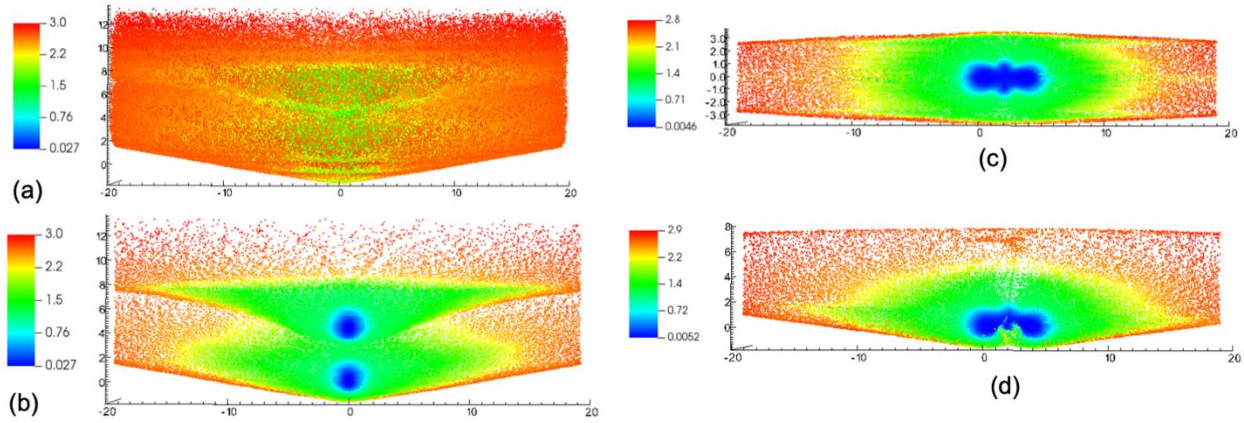

Figure15: Lagrangian particle simulation of the ablation of two SPI fragments with grad-B drift in 2T magnetic field.

3.5.4 Simulation Studies of Parallel Flow Problem

Understanding the dynamics of long-range propagation of ablated material along magnetic field lines is a high-priority task. 3D Lagrangian particle simulations of a plasma column expansion up to 10 m in length have been performed and the LP simulation results are in very good agreement with 1D PRL code simulations (obtained by L. Baylor) in terms of expansion distances and longitudinal profiles of thermodynamic states. This serves as an additional verification result for the LP code. Work in progress: a set of 1D equations that incorporate changing magnetic field was developed by P. Parks. These equations are being implemented in 1D version of the Lagrangian Particle code. 1D LP simulations with these modified equations will be compared with full 3D Lagrangian particle simulations of parallel flow in changing magnetic field.

3.5.5 Multiscale coupling of LP and tokamak MHD codes

Lagrangian particle approach is beneficial for coupling with global tokamak codes for the following main reasons: (a) no need for overlapping domain decomposition typical for grid-based codes, (b) Lagrangian treatment of ablated material leads to conservative extraction of ablation flow data. We approached the coupling problem in two-stages.

Stage 1: Loose coupling consists in pre-computing pellet / SPI ablation data and using them as source terms in global MHD codes. The current source terms incorporate information obtained from FT and LP simulations. Work on a detailed pellet ablation database G(B, Ne, T, rp) is undeway.

Stage 2: Strong coupling.

- Global MHD and LP Pellet / SPI codes will run in parallel on a supercomputer using different nodes / communicators

- Data exchange to be performed at the time step of the global MHD code (which is much larger compared to the LP code time step)

- Pellet code data is currently represented by particle states data files; in the future, it will be represented in terms of basis functions of the global code and the corresponding coefficients will be sent to the global MHD code

We have recently developed a coupling approach that has a well-defined, physics-based separation of scales:

- LP code evolves self-consistently the entire ablation cloud that provides pellet shielding

- grad B drift model in the LP code propagates ablated material across magnetic field lines, establishing the cloud shielding length. Ablated material that drifted beyond the main ablation cloud is transferred to the tokamak code, together with thermodynamic data and energy sinks. Particle representation ensures conservative mass transfer

- LP code obtains the magnetic field and electron density and temperature from the tokamak code

- LP data input has been successfully incorporated in NIMROD (Figure 16).

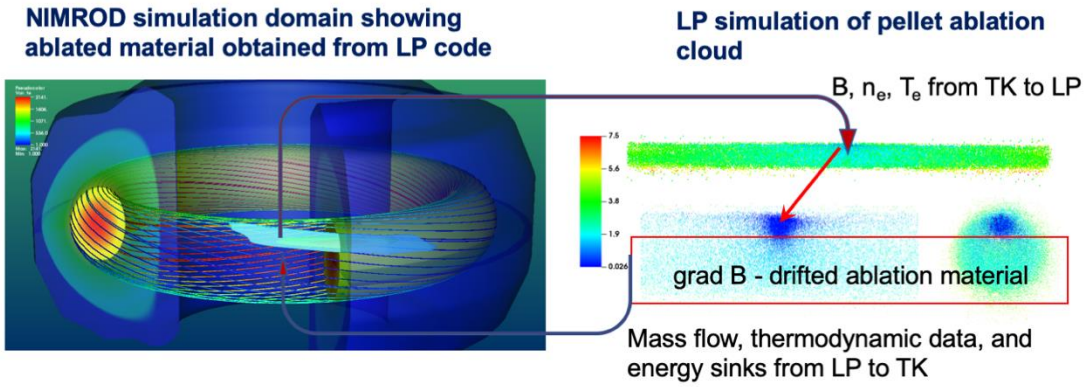

Figure 16: Multiscale coupling of LP simulation of a single neon pellet to NIMROD code

4.0 Runaway Electrons

High energy runaway electrons will be generated during a disruption. There is another SciDAC, SCREAM, which is primarily concerned with describing the generation of runaway electrons. However, our disruption modeling codes NIMROD and M3D-C1 need some model for the runaway electrons and how the effect the MHD evolution of the disrupting plasma. We describe relatively crude models of runaway electron dynamics that we have coupled with the MHD codes to start to understand their effects on the MHD. The longer term plans is to couple our codes with more complete kinetic models being developed within the SCREAM SciDAC.

4.1 Reduced Runaway Electron Model in NIMROD

U. Wisconsin

The generation of a runaway (relativistic) electron (RE) beam as a result of disruption is the subject of numerous theoretical investigations. Most focus on the kinetic physics of the RE population, yet the interaction with the overall discharge evolution and the influence on macroscopic instabilities is also important. Fully kinetic descriptions of REs for macroscopic evolution are desirable, but they will be computationally challenging, particularly for spatially 3D modeling. As a practical step, we are implementing a variant of a reduced model that has been used for previous macroscopic disruption studies by Helander, et al., [Phys. Plasmas 14, 122102 (2007)] and by Cai and Fu [Nucl. Fusion **55**, 022001 (2015)]. The REs are described by cold-beam drift kinetics, so that the parallel velocity is the speed of light and perpendicular flow is governed by drifts. They are then described by a fluid continuity equation, and their electrical current is coupled into the Ohm's law for macroscopic dynamics by making the RE population free of electrical resistivity. This is summarized by the following equations:

$$\frac{\partial n_{RE}}{\partial t} + c\hat{\mathbf{b}}\bullet\nabla n_{RE} + \mathbf{v}_{\mathbf{D}}\bullet\nabla n_{RE} = 0$$

$$\mathbf{v}_{RE} = \frac{\mathbf{E}\times\mathbf{B}}{B^2} + \frac{\gamma}{\omega_{ce}}\hat{\mathbf{b}}\times\left(\frac{\mu}{m_e}\nabla B + v_{||}^2\kappa + v_{||}\frac{\partial \mathbf{b}}{\partial t}\right)$$

$$\mathbf{E} == -\mathbf{v}\times\mathbf{B} + \eta\left(\mathbf{j} - \mathbf{j}_{RE}\right)$$

The rest of the system is the standard extended-MHD equations, except that the current density from the runaway electrons, JRE is not included in the Lorentz force density in the momentum equation. Postdoctoral associate Dr. Ge Wang has started implementing this model in NIMROD during this reporting period. He has implemented the RE continuity equation using the first order systems least-squares method to enhance numerical stability for the implicit parallel advection at the speed of light (or for a numerically reduced speed). The implementation of has been tested on propagation of a RE density distribution that varies over the azimuthal angle of a large aspect-ratio cylinder. The magnetic field has uniform q=1.25, and the effective major radius is 10. With the speed of light set to unity and no drifts, the density should rotate at a rate of 0.08 rad/time. Computed results after 20 and 200 units of time are shown in Fig. 17. The computation had fine azimuthal resolution of 96 cubic elements, and time-steps of unity exceed the explicit stability limit by more than a factor of 20. The full implementation will require iteration between the magnetic advance and the RE advance in order to couple their evolution, implicitly.

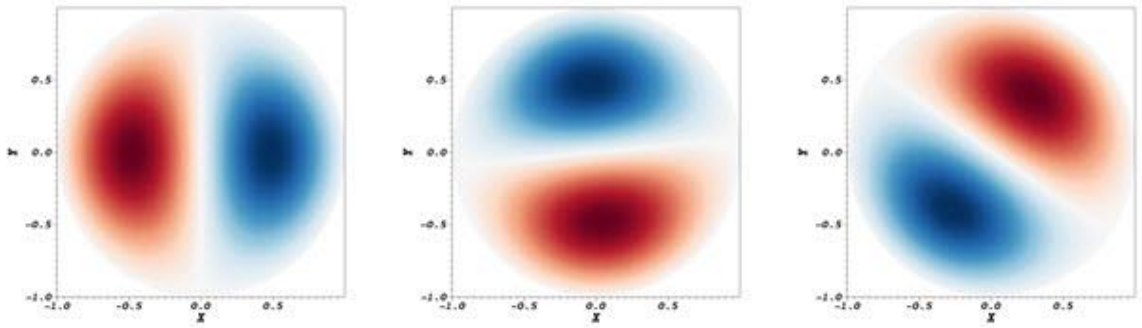

Figure 17: Contours of computed constant RE density from the test case at t = 0 (left), t = 20 (center), and t = 200 (right), where the expected rotation is 0, 1.6 rad, and 16 rad, respectively.1:

4.2 Reduced Runaway Electron Model in M3D-C1

A fluid runaway electron model has been implemented in M3D-C1 by C. Zhao and C. Liu. The M3D-C1 equations are modified as follows:

$$\frac{\partial n_{RE}}{\partial t} + \nabla \bullet (n_{RE} c \frac{\mathbf{B}}{B}) = S_{RE}$$

$$\mathbf{J}_{RE} = -e n_{RE} c \frac{\mathbf{B}}{B}$$

$$\mathbf{E} + \mathbf{V} \times \mathbf{B} = \frac{1}{ne}(\mathbf{R}_c - \mathbf{R}_{RE} - \nabla \bullet \Pi_e) + \mathbf{S}_{CD}$$

Here the terms in red are those that have been added. We are also investigating replacing the first equation with an anisotropic diffusion equations: $\partial n_{RE}/\partial t + \nabla \bullet (\mathbf{B} D_{RE} B^{-2} \mathbf{B} \bullet \nabla) = S_{RE}$ which may have better numerical stability and accuracy for long times.

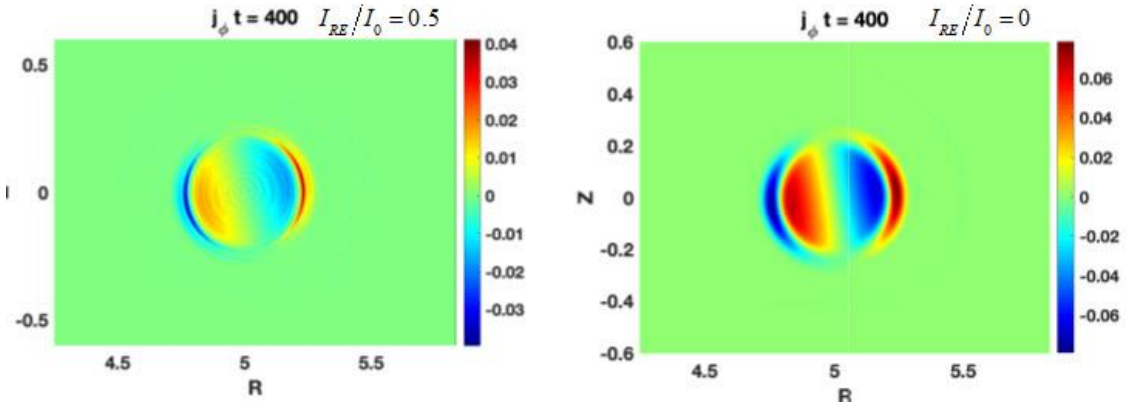

Figure 18: Linear eigenfunctions for a (1,1) instability with (left) and without (right) runaway electrons present in the equilibrium.

We show the eigenfunctions of a (1,1) instability with and without runaways present in Figure 18. We find that the perturbed runaway current dominates the perturbed toroidal current, even when only half

the equilibrium current is being carried by runaways. The presence of the runaways causes the current around the rational surface to become more peaked.

4.3 Improved Fluid Model for Runaway Electrons

Utah State U.

The major goal of this fluid model is to develop accurate closure relations which capture runaway electron effects. For an arbitrary relative flow velocity between electrons and ions, we need an accurate evaluation of the nonlinear collision terms in the relative flow velocities. Using our moment approach, we have developed a formalism to calculate exact collisional moments for arbitrary flow velocities.

For the project period, USU has developed a formalism to calculate the collisional moments. We have derived explicit formulas for the analytic collision coe-fficients which are expressed in terms of finite series of the electron/ion flow difference, V_{ei}. Since the moment expansion is performed in each fluid frame with a shifted Maxwellian distribution function, the formulas are exact for arbitrary $V_{ei.}$ The formulas involve only algebraic expressions and can be implemented in symbolic computation programs such as Mathematica and Maple. We have completed the Mathematica code for this purpose.

The exact calculation of the collisional moments is expected to significantly modify the Braginskii closure relations at finite values of . $V_{ei.}$ Figures 19-20 show the friction coefficients for a few lowest order moments. They show that the Braginskii closures are valid only when Vei is much smaller than the electron thermal speed. Note that the friction force is proportional to V_{ei} for V_{ei} at 20% of the thermal speed in Fig. 19 and the collision coefficients are constant only when V_{ei} is much less than the thermal speed in Figs. 20. For flow differences comparable to the thermal speed, the exact collision coefficients are significantly different from the linear coefficients and will modify the Braginskii closure coeffi-cients significantly. For V_{ei} much larger than the thermal speed, the friction is ignorable as expected.

The remaining task in the near future is to develop the closure theory using the exact collision coe-fficients for arbitrary V_{ei} . When V_{ei} is parallel to the magnetic field, which is the case for the lowest drift-kinetic order, the parallel closures can be calculated by the methods developed in previous work. Different from the linear collision calculation where the collisionality is determined by the collision frequency, the collisionality is determined additionally by the magnitude of V_{ei} in the exact calculation. For high collisionality it is straightforward to calculate the closure coeffi-cients by inverting collision matrices. For low collisionality the integral (nonlocal) closures can be obtained by existing methods. In the near future, the parallel closures will be explored for various V_{ei} values . When the perpendicular component of V_{ei} are not ignorable, the inner-product with the higher-order moments makes the algebraic structure of the moment equations complicated. We also plan to develop a method to compute closures for general orientation of V_{ei} with respect to the magnetic field.

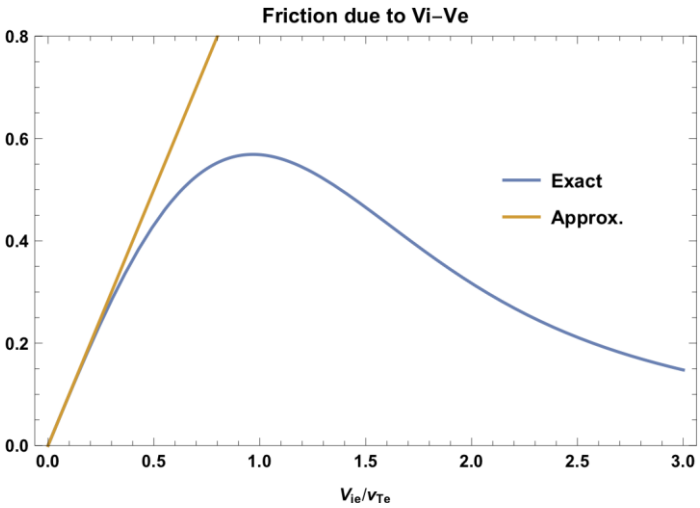

Figure 19: Friction due to Vi-Ve

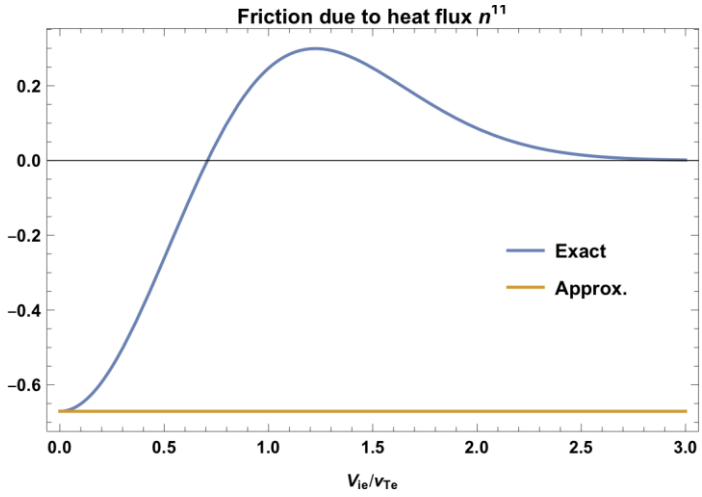

Figure 20: Friction due to heat flux

5.0 Experimental Validation

Validation of our models with existing experiments is an essential step in model development and application to ITER. We list three areas in which we have made progress in validating certain aspects of our models.

5.1 Validation of NIMROD Particle Based SPI Model against DIII-D

General Atomics

The NIMROD SPI validation comparison with DIII-D simulates three pellets: a pure neon pellet, a mixed deuterium/neon pellet (10 times more deuterium atoms than neon), and a pure deuterium pellet. These pellets are injected into a DIII-D equilibrium. Comparisons of the thermal quench times and radiated

fraction shows good agreement. The trends agree and quantitative agreement is reasonable. The agreement in thermal quench time reflects the appropriate choice of fragment velocity and the resulting time-of-flight. These results were presented in at the APS DPP Annual Meeting last fall and published in the April, 2019 issue of Physics of Plasmas.

5.2 Validation studies with EAST

U. Wisconsin

Prof. Ping Zhu's group has applied NIMROD to the nonlinear 3D simulation of vertical displacement in the Experimental Advanced Superconducting Tokamak (EAST) configuration. They find that the plasma position starts its displacement when the TQ phase ends and the CQ phase just begins. As time goes on, the toroidal asymmetry spreads into the plasma core from its initial location near the separatrix. Meanwhile, the asymmetry becomes more pronounced with time, and the vertical acceleration grows after the TQ starts. Zhu's group has also used the test particle model within NIMROD to investigate the RE loss mechanism during massive gas injection (MGI) triggered disruption in EAST. It is found that during TQ phase, a significant fraction of seed test REs is lost due to magnetic field stochasticity. Reappearance of the magnetic flux surfaces after the TQ phase hinders further RE losses. Different levels of helium gas injection affect the magnetic field stochasticity which can significantly modify the RE dynamics. In cases of larger He injection, the TQ occurs very fast, leading to strong magnetic field stochasticity and loss of REs at once. In cases of lower He injection, the complete RE loss occurs in three steps through changes in magnetic stochasticity: 1) edge, 2) healing, and 3) complete stochasticity/RE loss. The results suggest that the RE loss time can be drastically increased by reducing the He injection.

5.3 Validation studies with JET

HRS Fusion (lead), PPPL

Benchmarking was carried out of M3DC1, M3D, and JET data in asymmetric vertical displacement event (AVDE) disruptions. The main physics result is the wall force resulting from an AVDE is quenched when the current quench time is less than the magnetic wall penetration time. This was reported in [Strauss, et al. Phys. Plasma 24 102512 (2017)] and verified in JET experiments. This is an important result for ITER, which has been simulated with M3D [Strauss,Phys. Plasma 25 020702 (2018)] and will be verified later with M3D-C1.

M3D-C1 JET runs were initialized with data from shot 71985 which experienced an AVDE disruption. The resistive wall time τ_wall in the simulations was 0.0006s, while in the experiment it was 0.005s. The resistive wall time was shortened to speed up the simulations, but it is planned to use the experimental value in future simulations. The current in the simulations was driven with a time dependent electric field, to give control over the current quench time, while keeping the resistive wall penetration time fixed.

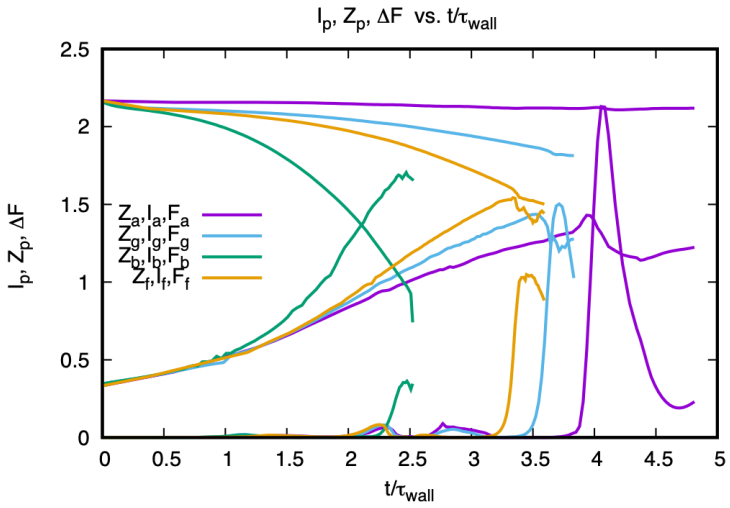

Figure 21: Time history of current I, vertical displacement Z, and asymmetric wall force for several cases as current quenches faster and the vertical displacement is faster, the wall force has a smaller peak amplitude.

Fig.21 shows time history of current I, vertical current centroid displacement Z, and asymmetric wall force F for three cases. As the current quenches and vertically displaces more rapidly, the peak wall force becomes smaller. The peak wall force as a function of current quench time divided by wall time, τ_CQ/τ_wall is shown in Fig. 23. Also shown are simulations done with M3D, and JET data. The JET data is from the ITER like wall (ILW) disruption database 2011-2016. The asymmetric or sideways force is labelled ΔF_x. Also shown is the Noll force ΔF_N=πBΔ(IZ), where Δ is the amplitude of the toroidal variation.

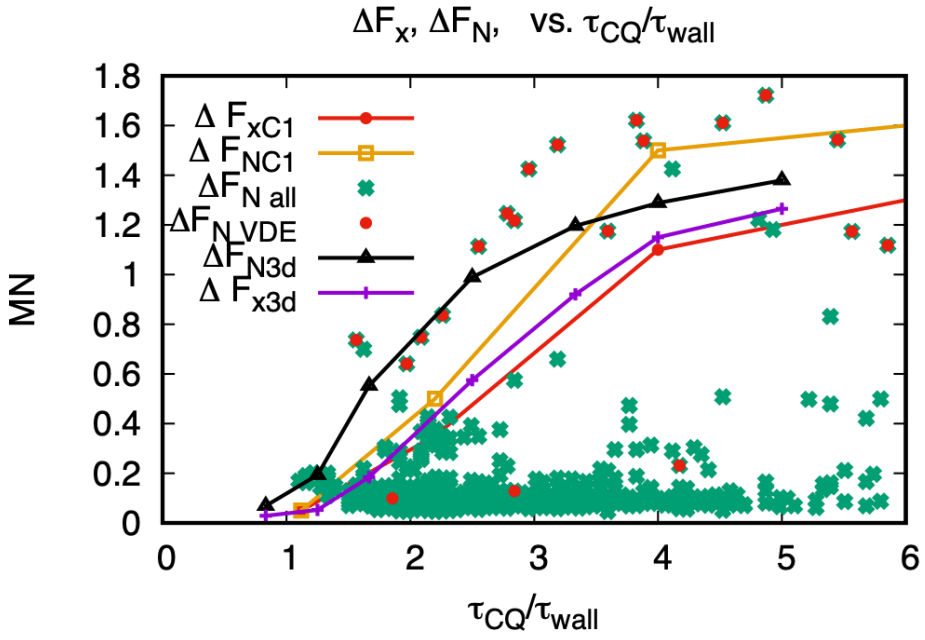

Figure 22 Subscript C1 indicates M3DC1 simulations, subscript m3d indicates M3D, subscript all are all the shots in the JET database, while subscript VDE are shots with VDEs.

Fig. 23 (left) shows the plasma current density as it approaches the wall, causing the peak sideways force. Fig. 23 (right top) shows, that the amplitude of the n=1 MHD mode coincides with the maximum sideways wall force. Fig. 23 (right bottom) is shown the q profile at several times. When the force and n=1 mode have maximum amplitudes, q is 1 at the plasma edge.

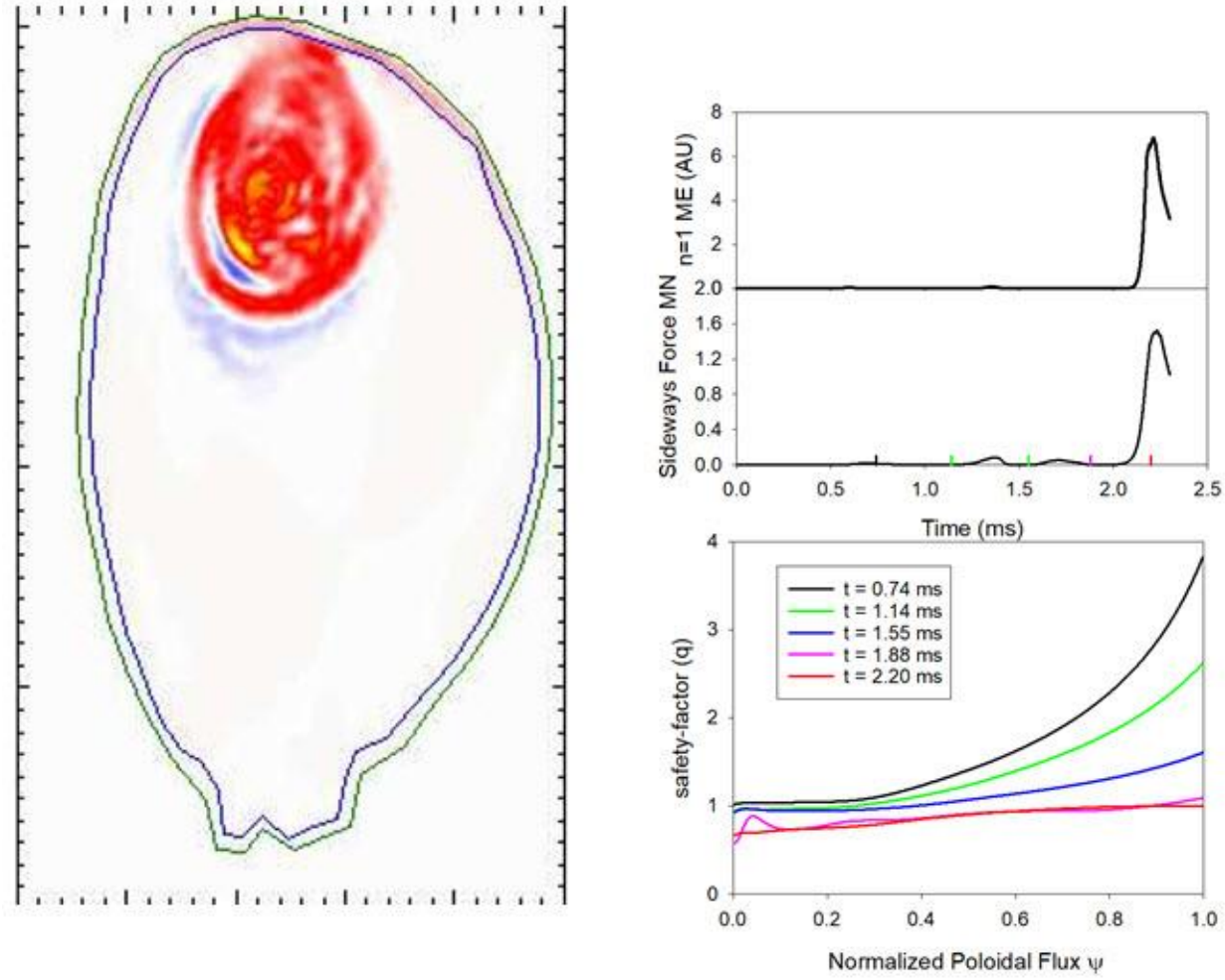

Figure 23: (left) Plasma drifts upward and is MHD unstable, producing wall force. (top right) Amplitude of n=1 mode , and wall force in MN. (bottom right) Safety factor profile q at different times. At late time, q=1 and (1,1) mode grows rapidly.

6.0 Computer Science and Code Optimization

The Fusion applications M3D-C1 and NIMROD both solve systems of equations using GMRES (M3D in real space and NIMROD in Fourier domain). In both cases, the system is ill-conditioned, and a preconditioner is required. Thus, in a naive implementation, prior to entering GMRES, a block (e.g. poloidal plane) of the system of equations is factored using SuperLU. Subsequently, in each GMRES

iteration, a pair of triangular solves (SpTRSV) are performed to act as the preconditioner. As factorization is expensive, it can be performed less frequently based on how much the operators have changed since the previous time step. Ultimately, in both codes, GMRES can be the dominant component with SpTRSV often dominating GMRES time (with LU factorization a close second). Thus, improving the performance and scalability of SpTRSV and SpLU is imperative if one wishes to improve M3D and NIMROD performance. To that end, we have explored four approaches in the last year. First, we have endeavored to improve the communication performance in SpTRSV. Second, in order to reduce off-node communication and total memory requirements we are exploring efficient threading techniques in SuperLU's SpTRSV and SpLU. Third, as GPUs dominate the DOE HPC landscape, we are extending our threading techniques to leverage GPUs for SpTRSV (GPUs for SpLU was already implemented a few years ago). Finally, in order to improve scaling, we are exploring the benefits of 3D matrix decompositions for SpLU and SpTRSV. Whereas 2D matrix decompositions are common practices for sparse direct solvers, 3D decompositions offer reduced off-node communication and vastly improved scalability.

In addition to these improvements in the communication and sparse matrix routines, Section 6.1 is an effort to reduce communication on the NIMROD side, Sections 6.2-6.4 describe improvments in the local ablation codes, and and Section 6.8 is an effort to make the M3D-C1 solver use fewer iterations.

6.1 NIMROD Refactoring

TechX

In terms of computational infrastructure, a refactoring of the code using the abstraction concepts from Fortran 2008/2013 is progressing. The original Fortran 90 code was written in an abstract way minimizing the need for a complete rewrite. The new code is more flexible and will enable such CTTS-related features such as better performance for continuum-kinetic solves by enabling static condensation and mixed OpenMP/MPI parallelism. The abstract NIMROD branch has been moved to a repository on gitlab which enables modern development practices suitable for a distributed team such as continuous integration testing, code review and merge requests. In January 2019 members of the NIMROD team visited Tech-X to discuss the future use of GPUs with NIMROD. Work is underway to understand the best programming paradigm. The use of OpenACC with NIMROD finite-element assembly kernels has shown the potential for good GPU acceleration. This work used simple data structures and in the future we plan to understand how to use OpenACC with the full NIMROD data structures. In particular we expect that the data layout needs to optimized to limit communication between GPUs during our iterative solves, and we expect that we will needed reorganize the date as illustrated in the Fig. 24. The code abstractions will facilitate testing of multiple data organization layouts.

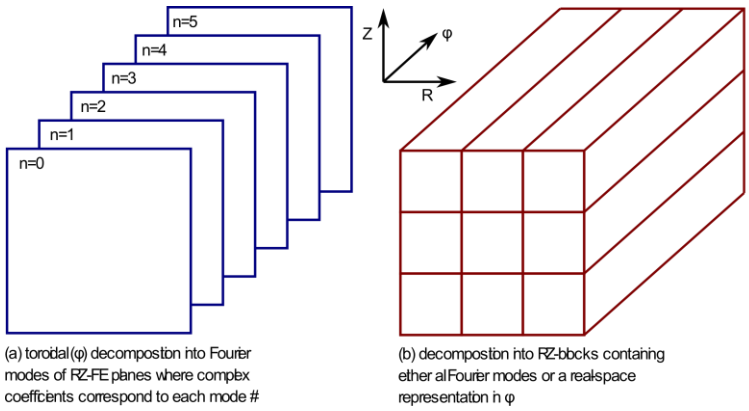

Figure 24: Proposed data reorganization designed to reduce communication in NIMROD's iterative solver. Reducing the communication between GPUs is essential for good performance

6.2 Massively Parallel K-tree Algorithms for Lagrangian Particle Code

Stony Brook U.

The most time and memory consuming algorithm in the Lagrangian Particle LP code constructs octrees for particle distribution and searches these octrees for finding neighbors of each individual particle for stencil operations of numerical PDE algorithms. In the current production code, this algorithm is parallelized using OpenMP and runs on a single multicore node, which limits the total number of particles in simulations. Because of adaptivity of the LP code, a typical multicore supercomputer node is sufficient for fully resolved simulation of a single pellet or a small number of pellet fragments.

In the SPI scenario, a large pellet is expected to shatter into hundreds of fragments. For example, it is estimated that a large pellet shatters into 250+ of fragments in DIII-D experiments. The corresponding simulations requires generating very large number particles (more than 10^7), making simulations on a single multicore node very difficult or impossible (unless performed on the most advanced supercomputer architecture). We have started working on MPI parallelization based on P4EST (Parallel Forest of Octrees) software library, developed in the past within an ASCR-funded project of Omar Ghattas. P4EST library enables a dynamic management of a collection of adaptive octrees on distributed memory supercomputers. P4EST scales to hundreds of thousands of processor cores. We have successfully implemented main particle data structures in the LP code compatible with P4EST and linked the codes. The new code has been tested over multiple MPI nodes with several millions of particles (it is capable of handling much larger numbers of particles). The code successfully performed particle initialization and transformations due to prescribed forces (see Figure 25). The physics routines and solvers have not been ported to the new massively parallel code yet, but we do not anticipate any essential difficulties in completing this task.

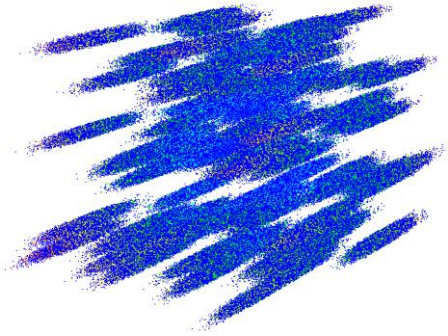

Figure 25: Distribution of ablated material in the parallel LP code coupled to the P4EST library.

6.3 New Adaptive Algorithms for LP-based SPI code

Numerous algorithms in the LP SPI code (kinetic heating, grad-B drift model etc) need to compute integrals of certain properties of the ablated material along magnetic field lines. As the distribution of the ablated material is highly non-uniform, an adaptive selection of integration lines is very important for the numerical accuracy and the performance of the code. In the past year, we have developed and implemented such an adaptive algorithm. It is illustrated in Figure 26.

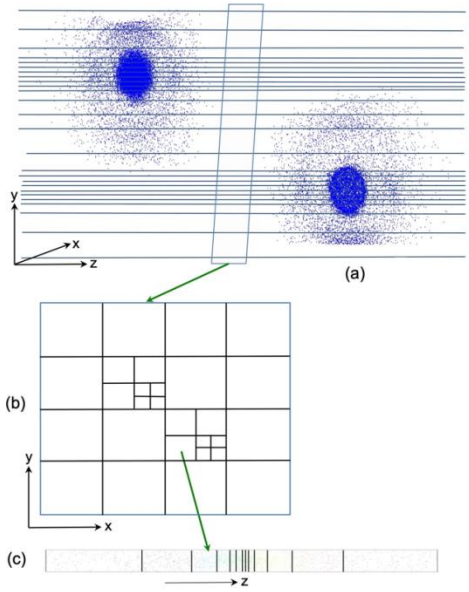

Figure 26: a) Schematic of 3D distribution of Lagrangian particles in SPI simulation. Horizontal lines schematically depict plasma density integral paths, adaptively refined near pellet fragments. b) Quadtree data structure, built using Lagrangian particles projected to a transverse plane. Each quadtree cell contains one path for the plasma density integral. c) Re-

distribution of Lagrangian particles in each quadtree cell to 3D using their saved longitudinal coordinate and line integration of density based on a binary tree in the longitudinal direction.

6.4 Explicit Tracking of Ablation Cloud-Plasma Interface in the FronTier Code

Stony Brook U.

Past simulations of the pellet ablation with the FronTier code used interface tracking for the pellet – ablated material interface, but the interface between the ablated material and the ambient plasma was not tracked. After we implemented additional physics models that account for the properties of the ambient plasma in both codes, it became important to explicitly track the interface between the ablated material and the ambient plasma. Such tracking was recently enabled in the FronTier code by developing new routines for the propagation of this interface, shown in Fig. 27. While new simulation results show a big improvement in the transverse profiles of physics states in the ablation cloud, in particular the distribution of the longitudinal velocity in the transverse direction, that was previously very diffused near the cloud boundary, the ablation rate change due to tracking was small (within several %, depending on simulation conditions).

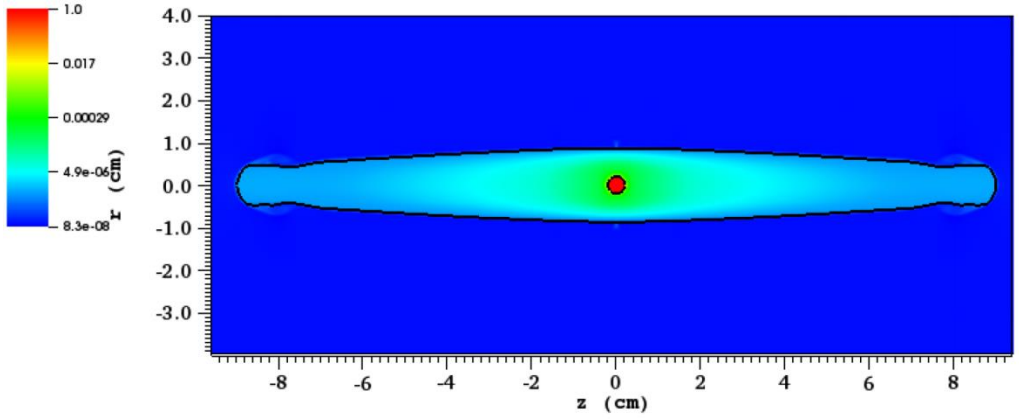

Figure 27: FronTier simulation of neon pellet in 2T magnetic field with explicit tracking of the interface between the ablated material and the ambient plasma

6.5 One-sided communication for SPTRSV

LBL

Distributed memory Sparse Triangular Solve (SpTRSV) solves a lower (or upper) triangular system of equations. Unlike iterative methods that often exhibit abundant regular parallelism with regimented communication patterns, SpTRSV exhibits irregular, fine-grained parallelism with asynchronous communication using small (e.g. 512-byte) messages. Whereas such communication and computation can be implemented in a scalable manner with two-sided MPI messaging (MPI_Isend/Irecv), such operations often incur extremely high messaging overhead relative to network bandwidths. As a result, average network bandwidth can suffer and the total time SpTRSV may spend in network operations can easily exceed 90% at 4K cores. In FY19, LBL explored alternate communication and tasking models to accelerate distributed memory SpTRSV performance.

MPI 3.0 introduced one-sided messaging (put or get instead of matched send and receive) wherein one process can simply write to another process's memory using RDMA. In theory, such an approach should enable more efficient inter-process communication and higher SpTRSV performance. In practice, such approaches often fail on account of two major limiting factors. First, most vendors (e.g. Cray) have heavily optimized their two-sided implementations, but only provide functional implementations for one-sided communication. As such, MPI_Put/Get overheads may be higher than MPI_Isend or MPI_Irecv. Second, by its very nature, MPI_Isend requires two messages be sent on the network: the data and a receipt acknowledgement. Although one-sided puts and gets do not require an acknowledgement, most algorithms require some form of synchronization to avoid overwriting a remote buffer before it has been read or reading a local buffer before data has been fully received. Thus, attempts at one-sided communication often degenerate into two-sided implementations. LBL solved these two challenges for SpTRSV by adopting ETH's foMPI implementation of MPI-3 optimized for the Cray Aries network [FOMPI] and privatizing buffers and encoding a checksum into one-sided messaging to pass data and synchronization in a single message.

We evaluate our implementation on both the Knights Landing and Haswell partitions of NERSC's Cori system (the primary target of the NIMROD and M3D-C1 teams) using a variety of standard matrices (plus the 'A30' exemplar from the M3D-C1) as we scale from 256 to 4096 cores on Knights Landing and 64 to 1024 cores on Haswell. The figures 28-29 below (top:KNL, bottom:HSW) show SpTRSV solve time as a function of concurrency, matrix, and implementation. Note, the two-sided implementation is the production SuperLU implementation that M3D-C1 and NIMROD rely upon while the fompi implementation is our new one-sided implementation. As one can see, the time in two-sided communication (light tan) varies substantially among matrices and can quickly dominate the total run time at high concurrency. Conversely, our one-sided implementation can substantially reduce communication time (light tan vs. light blue) and improve overall SpTRSV solve time at high concurrency by 2.4x. Note, as our optimizations focused solely on the dominant bottleneck (communication), computation time was unaffected (red vs. blue).

Our results were codified into a paper submitted to SIAM PP20 [PP20_SpTRSV] and the code is currently available for evaluation in a branch of the master SuperLU repo. We plan on fully integrating both the real (for M3D-C1) and complex (for NIMROD) implementations of the code into a future SuperLU release enabling both codes to seamlessly improve their preconditioner performance (the dominant piece of their GMRES solvers).

The nature of GPUs and other accelerated architectures are antithetical to the requirements for two-sided MPI. As a result, two-sided MPI will always perform less efficiently on a GPU than on a CPU. Thus, it is broadly accepted that any communication-sensitive application wishing to exploit a GPU must leverage one-sided communication. To that end, in the future, we will leverage NVIDIA's implementation of OpenSHMEM (NVSHMEM) to realize a truly one-sided and accelerated implementation of SpTRSV for the GPU-accelerated NERSC-9 Perlmutter system and will serve as a template for one-sided implementations for the Aurora and Frontier follow on accelerated systems.

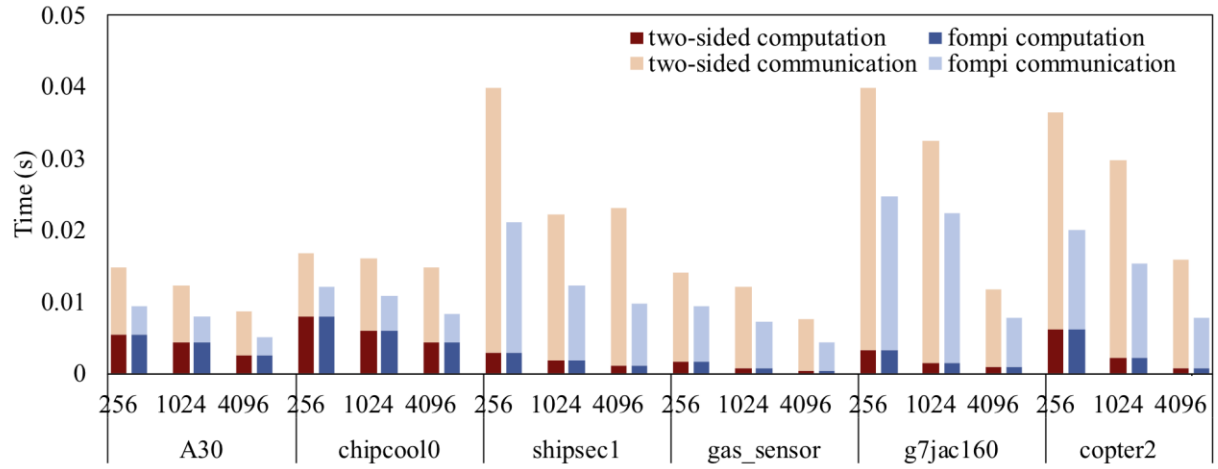

Figure 28 Testing of one-sided and two-sided communication onon Cori-KNL

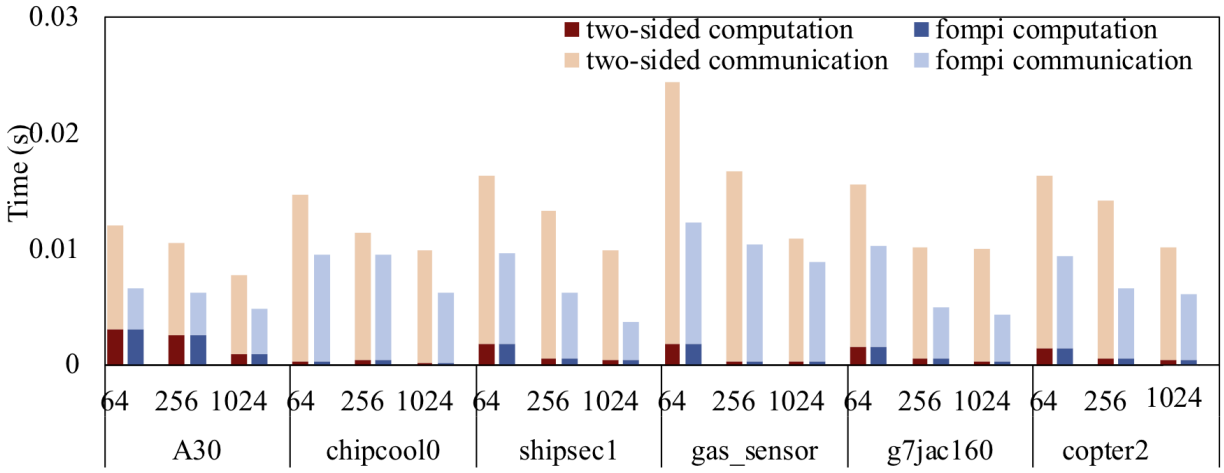

Figure 29: Testing of one-sided and two sided communication on on Cori-HSW

6.6 Threading in SpTRSV

LBL

SpTRSV performance can be significantly improved by leveraging on-node parallelism on top of a distributed-memory implementation. As mentioned above, distributed-memory SpTRSV quickly becomes communication-bound with a modest number of MPI ranks. On the other hand, modern computer architectures such as KNL and GPU systems offer ever-growing on-node compute capability. Therefore hybrid programming models with distributed-memory/shared-memory parallelism should be exploited for SpTRSV. In FY19, LBL added the MPI+OpenMP support in SpTRSV of SuperLU_DIST and

optimized the performance on the KNL and POWER9 partitions of Cori and Summit, respectively. In addition, LBL added support for one GPU per process in SuperLU's SpTRSV using CUDA.

Assuming one right-hand side (RHS), there are two types of shared-memory parallel algorithms for SpTRSV, namely level-set and blocking algorithms [THREAD]. The level-set method identifies sets of columns that can be solved simultaneously; the blocking method aggregates smaller sub-blocks into a larger one and performs the associated GEMV operation in parallel. Typically, the level-set method can be effective for very sparse triangular factors, while the blocking method is efficient for denser triangular factors. Indeed, the triangular factors resulting from the LU factorization usually exhibit increased density towards the lower right corner. Therefore good parallel SpTRSV algorithms should take into account both the level-set and blocking strategies.

In both our CPU and GPU implementations of shared-memory SpTRSV, we explored the two aforementioned parallelization strategies. For the CPU version, we use OpenMP tasks to assign each newly available column to one task and use OpenMP task loops to handle GEMV operations in each column. Note that the GEMVs are only parallelized when the number of rows in a column is large enough; otherwise it is more efficient to perform it with one thread. We keep private copies of the solution vector on each thread to avoid expensive atomic operations. For the CUDA-based GPU version, we assign one column to one thread block and use the entire thread block to handle the GEMV inside one thread block. Note that the CUDA kernel is optimized to avoid warp divergence and enable coalesced access. In contrast to the CPU version, here we utilize GPU atomics to update the solution vector.

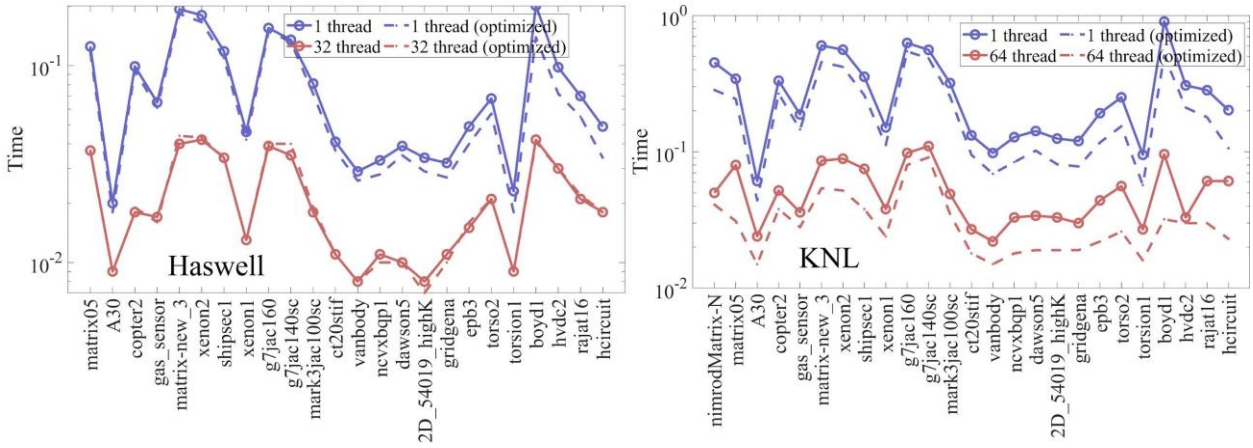

Figure 30: Comparison of run times with 1 and multiple threads on HSW (left) and Cori (right)

We tested and optimized the OpenMP-based SpTRSV on HSW and KNL partitions of the Cori system using several matrices including the M3D-C1 matrices A30 and matrix05, and the NIMROD matrix nimrodMatrix-N. The figure 30 below (left:HSW, right:KNL) show that we can achieve up to 5.5x and 10x speedups (comparing the two dashed lines representing 1 and multiple threads) using 32 and 64 threads on HSW and KNL, respectively. Note that the dashed lines represent the runtime after various code optimizations particularly targeted at KNL nodes.

Particularly, we tuned the memory access pattern and the criteria for activating threaded GEMV in each supernode column. As a result, the code optimization yields an up to a 3x better run time on KNL (the red solid and dashed lines in Fig. 30).

We tested our single GPU implementation on Summit using several matrices including the M3D-C1 matrix matrix05. We compared our implementation with the cuSPARSE library and our single-core CPU implementations. Fig. 31 (left: 1 RHS, right: 100 RHS) shows the runtime of the lower triangular solve for the three implementations excluding the setup, preprocessing and memory transfer time. Note that cuSPARSE requires costly conversion of the matrix format from the supernodal representation to the CSR format, which is not included in the figures. That said, our implementation can still be up to 5x and 3x faster compared to the cuSPARSE and CPU implementations for 1 RHS (which is the typical use scenario in M3D-C1 and NIMROD). For 100 RHSs, our implementation is 10x faster than the CPU implementation and comparable to the cuSPARSE library.

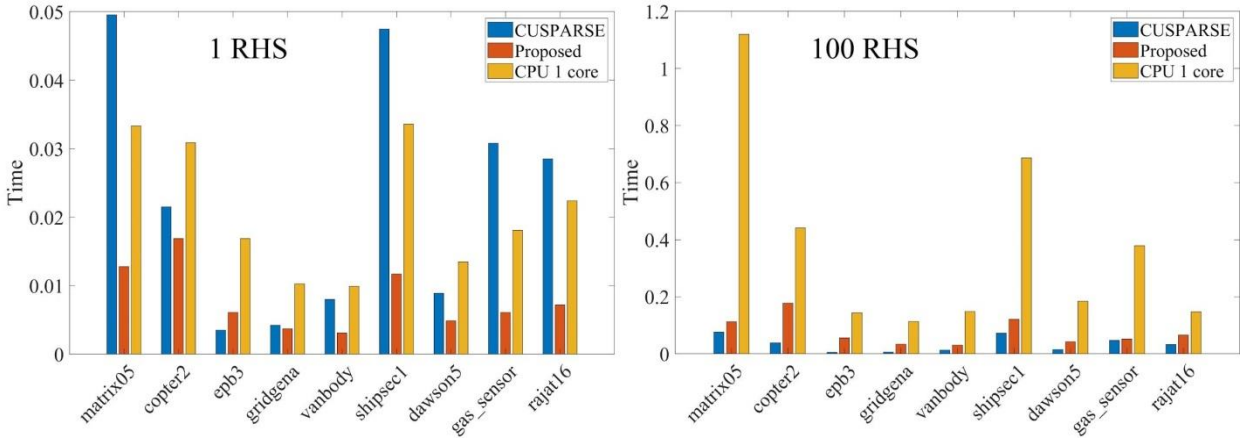

Figure 31: Comparison of run times with 1 RHS (left) and 100 RHS (right) between single CPU, single GPU using our implementation, and single GPU using cuSPARSE library.

When parallelized across multiple CPU/GPU nodes, although the available parallelism remains fixed, artifacts of MPI parallelization can reduce the amount of shared-memory parallelism within each process. This is primarily due to the need to communicate between nodes before a supernodal column becomes available for computation. Handling such communication with traditional two-sided MPI or even GPU-aware MPI functionalities will dramatically reduce the overall code performance. Therefore, we plan to leverage the one-sided foMPI and NVSHMEM implementations to alleviate this performance bottleneck in our future CPU and GPU-based SpTRSV implementations and SuperLU_DIST releases.

6.7 3D Sparse LU and triangular solves

LBL

In the previous 2D SpLU code, we found that more than 80% of the parallel run time is attributed to communication when using 1000's of MPI processes. We are therefore motivated to redesign algorithms to reduce communication, as the recent flurry of research on communication-avoiding

algorithms suggests. One critical strategy is to shrink the amount of data transferred through redundant computation, data replication, or both.

We extended the 2.5D dense communication-avoiding LU factorization algorithm to the sparse case. Here, we use a three-dimensional MPI process grid with d copies of the 2D grids along the 3rd dimension. Each 2D grid performs factorization of an independent subtree of the elimination tree, with restricted communication within subtrees. This reduced communication is at the expense of the extra memory to maintain d copies of the partial sums for the last portion of the Schur complement submatrix. In essence, we exploit more elimination tree parallelism, and trade off increased memory for reduced per-process communication. When using ever larger numbers of nodes, with sufficient memory, we can use a larger d to favor less communication. On a smaller machine, we can use a smaller d to favor less memory usage. We combined the new 3D SpLU code with the previously developed GPU 2D code. On 4096 nodes of a Cray XK7 (Titan at OLCF) with 32,768 CPU cores and 4096 NVIDIA K20x GPUs, the 3D algorithm achieves speedups up to 24x for planar graphs and 3.5x for non-planar graphs over the baseline 2D SuperLU with GPU. We also performed theoretical analysis of the asymptotic improvements for planar graphs (e.g., those arising from 2D grid or mesh discretizations) and certain non-planar graphs (specifically for 3D grids and meshes). For a planar graph with n vertices, our algorithm reduces communication volume asymptotically by a factor of O(sqrt(log n)) and latency by a factor of O(log n). For non-planar cases, our algorithm can reduce the per-process communication volume by 3x and latency by O(n^⅓). In all cases, the memory needed to achieve these gains is only larger by a constant factor [SpLU3D].

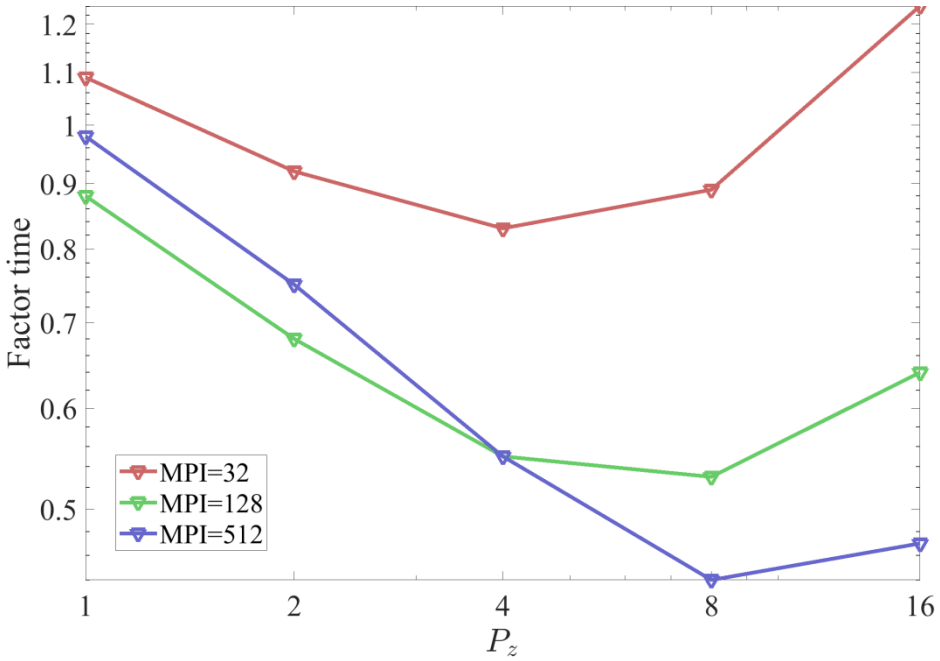

Figure 32: shows the parallel runtime of the 3D SpLU code for the matrix05 from M3D-C1. The 3D processes are arranged as Px x Py x Pz. Pz=1 is equivalent to the baseline 2D algorithm. The best runtime is achieved for 512 MPI processes (cores) with Pz set to 8. This corresponds to 2.2x speedup over the 2D code when Pz is set to 1.

For SpTRSV, we leverage the same 3D sparse LU data structure to develop a communication-avoiding algorithm, which yields asymptotic reductions in the latency and communication-volume costs of a conventional SpTRSV. Briefly, our new 3D SpTRSV maps independent subtrees of the elimination tree to each 2D process grid and replicates the common ancestors. It first solves independent subtrees on different 2D process grids, and then performs a reduction before solving the sub-problem in the common ancestor tree on a single 2D grid. This 3D triangular solve algorithm, when run on 12K cores of Cray XC30, outperforms the current state-of-the-art 2D algorithm by 7.2x for planar and 2.7x for the non-planar sparse matrices, respectively. Furthermore, we proved that our 3D SpTRSV can reduce the per-process communication volume asymptotically by a factor of O(n^¼) and O(n^⅙) for PDEs from planar graphs and non-planar PDEs, respectively [SpTRSV3D].

Our 3D SpLU code is already released in Version 7 of SuperLU_DIST. We plan to release the 3D SpTRSV code in the first or second quarter of FY20. Subsequently, we will incorporate the one-sided communication and GPU-accelerated SpTRSV in the independent 2D subtrees of the 3D framework. Finally, we will develop a GPU-accelerated SpTRSV code for the entire 3D scheme (instead of just the 2D parts).

6.8 New nodal DOF ordering procedures

RPI (lead), PPPL

During the execution of a single M3D-C1 time step, a global parallel sparse matrix is used to solve for the current nodal velocity, and first and second velocity derivative. This solution process represents the largest computational cost for each time-step. M3D-C1developers identified the possibility that a physically and numerically motivated restructuring of this matrix may allow better time-to-solution and provide time savings in executing M3D-C1 simulations.

In the M3D-C1 simulation, the fusion device toroidal domain is discretized by generating 2D unstructured meshes on poloidal planes. Prism elements are created by connecting triangular elements in adjacent poloidal planes. Shape functions are generated using a cross product of 2D quintic shape functions and cubic Hermitian shape functions for a fully 3D simulation. The matrix structure arising from the DOFs associated with the described unstructured mesh is block tridiagonal, where the off-diagonal blocks are associated with adjacent poloidal planes.

Standard DOF ordering results in such a block tridiagonal structure, where each DOF bearing mesh entity has a single contiguous set of rows in the matrix associated with its DOFs. In M3D-C1, each mesh vertex has 12 degrees of freedom, broken down into three groups of variables, each of which has four derivative terms. Our collaborators asked for a capability to instead group together each of these variables containing 4 terms for all mesh entities on the local plane. This results in the matrix retaining the same block tridiagonal structure, but the DOF rows for all entities on a poloidal plane – and thus for each block-row – will come first, followed by all the sets of derivative DOF rows. We dubbed this matrix structuring aggregate ordering as we are essentially aggregating associated terms into contiguous sections of rows in the matrix.

During our implementation of this functionality, based on the restriction that PETSc requires row-ownership of the matrix to a contiguous set of rows, we identified that this matrix structure would break the typical relationship in finite element analysis using parallel unstructured meshes wherein ownership of a mesh entity by the local part implies ownership of the matrix rows associated with DOFs on that mesh entity. This would result in a drastic increase of the communication of matrix nonzero contributions during the final parallel assembly of the matrix. Given this communication overhead, we determined it was desirable to develop the desired matrix restructuring in a general way allowing for experimentation to determine the optimal usage of the capability. Thus, rather than performing DOF aggregation over each poloidal plane, we instead developed a capability to perform the ordering procedure over any set of communicators which exactly partition the parallel execution space with no overlap, where each communicator aggregates the DOFs associated with mesh entities internally.

We have developed this functionality and implemented it within M3D-C1. An initial set of test confirms the system is better conditioned and thus solves faster, but the added communication overhead is substantial. Additional testing and consideration of potential alternatives is needed to determine the possible next steps.

7.0 Publications and Conference Proceedings

- I. Krebs, F. J. Artola, C. R. Sovinec, S. C. Jardin, K. J. Bunkers, M. Hoelzl, N. M. Ferraro, “Axisymmetric simulations of vertical displacement events in tokamaks: A benchmark of nonlinear MHD Codes”, Submitted to Phys. Plasmas (2019)
- C. F. Clauser, S. C. Jardin, N. M. Ferraro, “Vertical Forces during VDEs in an ITER plasma and the Role of Halo Currents”, *in press*, Nuclear Fusion (2019)
- B. Lyons, C. Kim, Y.l Liu, N. Ferraro, S. Jardin, J. McClenaghan, P. Parks and L. Lao, “Axisymmetric benchmarks of impurity dynamics in extended-magnetohydrodynamic simulations”, lasma Physics and Controlled Fusion 61 064001 (2019)
- S. Jardin, F. Villone, C. Clauser, N. Ferraro, N. Isernia, G. Rubinacci, S. Ventre,” ITER Disruption Simulations with Realistic Plasma and Conductors Modelling”, P5.1003, Proceedings of the European Physical Society (2019)
- C. Kim, Y. Liu, P. Parks, L, Lao, M. Lehnen, A. Loarte, “Shattered pellet injection simulations with NIMROD”, Phys. Plasmas **26** 042510 (2019)
- N. Bosviel, R. Samulyak, P. B. Parks, N. Naithlo, S. Yuan, “Numerical studies of Ne pellets ablation in tokamaks”, Proc. EPS-19, 46th Plasma Physics Conference of European Physical Society, Milan, Italy, July 8-12, 2019.
- H. Strauss, “Reduction of asymmetric wall force in ITER disruptions by current quench”, Physics of Plasmas **25** 020702 (2018).
- N. Ferraro, B. Lyons, C. Kim, Y. Liu, S. Jardin, “3D MHD modeling of fast thermal quenches due to impurities in tokamaks”, Nuclear Fusion **59** 016001 (2018)
- C. Sovinec and K. Bunkers, “Effects of asymmetries in computations of forced vertical displacement events”, Plasma Phys. Control. Fusion 61, 024003 (2019)

- S. Cheng, P. Zhu, D. Banerjee, et al.,"Dominant two-fluid MHD instabilities in CFETR upgrade phase I scenario in presence of perfect conducting wall" PPCF **61**, 045009 (2019)
- A. Ali and P. Zhu,"Effects of plasmoid formation on sawtooth process in a tokamak" Phys. Plasmas 26, 052518 (2019)
- Nan Ding, Samuel Williams, Yang Liu, Sherry Li, "Leveraging One-Sided Communication for Sparse Triangular Solvers", (submitted to) SIAM PP, 2020.
- P. Sao, R. Vuduc, X. Li, "A communication-avoiding 3D algorithm for sparse LU factorization on heterogeneous systems", J. Parallel and Distributed Computing (JPDC), 131 p. 218-234 (2019)