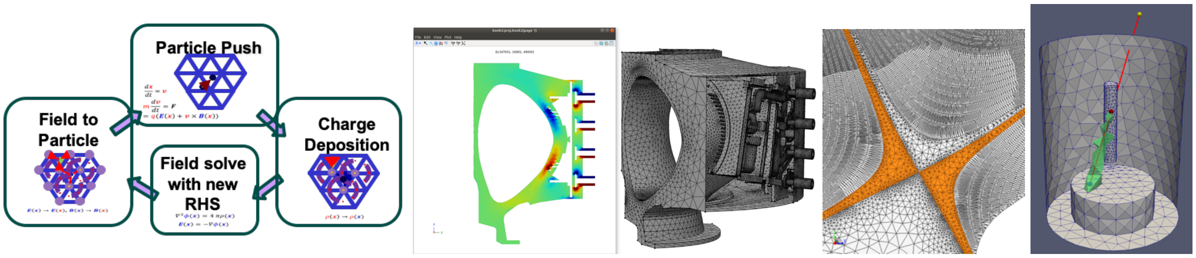
# Unstructured Mesh Technologies for Fusion Simulations

Cameron W. Smith
Gerrett Diamond
Gopan Perumpilly
Chonglin Zhang
Agnieszka Truszkowska
Morteza Hakimi
Onkar Sahni
Mark S. Shephard
smithc11@rpi.edu
Rensselear Polytechnic Institute
Troy, NY

Eisung Yoon
esyoon@unist.ac.kr
Ulsan National Institute of Science and Technology
Ulsan, South Korea

Daniel A. Ibanez
daibane@sandia.gov
Sandia National Laboratories
Albuquerque, NM



(a) (b) (c) (d)

Figure 1: (a) Unstructured mesh particle in cell simulation steps, (b) PetraM radio frequency simulation field and mesh, (c) XGCm unstructured mesh near the X-point, and (d) GITRm particle track in PISCES mesh.

## ABSTRACT

Multiple unstructured mesh technologies are needed to define and execute plasma physics simulations. The domains of interest combine model features defined from physical fields within 3D CAD of the tokamak vessel with an antenna assembly, and 2D cross sections of the tokamak vessel. Mesh generation technologies must satisfy these geometric constraints and additional constraints imposed by the numerical models. Likewise, fusion simulations over these domains study a range of timescales and physical phenomena within a tokamak. XGCm studies the development of plasma turbulence in the reactor vessel, GITRm studies impurity transport, and PetraM simulations model RF wave propagation in scrape off layer plasmas. GITRm and XGCm developments are using the PUMIpic infrastructure to manage the storage and access of non-uniform particle distributions in unstructured meshes on GPUs. PetraM combines PUMI adaptive unstructured mesh control with MFEM using CAD models and meshes defined with Simmetrix tools.

## CCS CONCEPTS

- **Computing methodologies → Parallel algorithms**; **Molecular simulation**; **Multiscale systems**; **Massively parallel and high-performance simulations**.

---

*SC19, November 2019, Denver, CO, USA*
.

## KEYWORDS

fusion, plasma physics, particle-in-cell, unstructured mesh

**ACM Reference Format:**

Cameron W. Smith, Gerrett Diamond, Gopan Perumpilly, Chonglin Zhang, Agnieszka Truszkowska, Morteza Hakimi, Onkar Sahni, Mark S. Shephard, Eisung Yoon, and Daniel A. Ibanez. . Unstructured Mesh Technologies for Fusion Simulations. In *Proceedings of The International Conference for High Performance Computing, Networking, Storage, and Analysis (SC19).* ACM, New York, NY, USA, 3 pages.

## 1 INTRODUCTION

Unstructured mesh fusion simulations require technologies for geometric model creation, mesh generation, adaptation, and particle-mesh interactions. A PetraM [10] radio frequency wave simulation begins with creation of a geometric model from a combination of complex CAD and physical field data. The resulting geometric model is automatically meshed given user provided geometry-based mesh controls. The mesh, model, and analysis attributes are then fed into an adaptive finite element analysis. XGCm tokamak plasma turbulence simulations [6, 8] require use of field following meshes of poloidal planes along with particle-in-cell (PIC) methods. The PIC method is dependent on a particle data structure that efficiently stores particles in groups by their bounding mesh element. Likewise, GITRm, a 3D implementation of GITR [15], impurity transport simulations use the same particle data structure with different particle distribution characteristics and mesh interactions.

Section 2 discusses PetraM simulation definition and execution. Section 3 describes the mesh and particle data structures used to support XGCm and GITRm. These applications are respectively discussed in Sections 4 and 5. Section 6 summarizes the work and describes on-going efforts.

## 2 ADAPTIVE RF WAVE SIMULATION

Executing a PetraM simulation requires first obtaining and cleaning up the ‘as-built’ CAD model of the antenna. The left image in Figure 1b shows the mesh on the simplified CAD. Simplification is performed with the Simmetrix SimModeler GUI [11] by selecting geometric model entities and specifying the modifiction approach post entity removal. For example, the healing modifier will geometrically and topologically close the hole left in a surface after removal of a bolt that penetrates through it. After simplification the CAD model is unioned in SimModeler with the CAD of the reactor vessel, and a discrete surface from the EFIT software [1] that defines the interface between the core plasma and the vessel. Next, attributes are applied to the model [9] to control automated mesh generation procedures [14]. Likewise, analysis attributes are applied on the model to specify the boundary/initial conditions and material properties. The resulting attributed model, mesh, and input control parameters are passed to an MFEM [7] finite element analysis. During the analysis error estimators based on the electric field are computed to determine if PUMI [5] local mesh adaptation is required to maintain a specified level of error.

## 3 INFRASTRUCTURE FOR UNSTRUCTURED MESH PIC

Unstructured mesh PIC simulations execute the series of operations depicted in Figure 1a. The field to particle operation transforms information at mesh vertices to particles in topologically adjacent elements. Based on the updated particle information, new positions for each particle are computed, independently, and the new mesh element that the particle exists in is located; these two steps are referred to as ‘particle push’. With the new parent determined, mesh fields are updated via a particle to mesh operation. The last operation in the series is the solution of a PDE on the mesh using the finite element method.

The computationally dominant operation for an unstructured mesh PIC application with significantly more particles than mesh entities is the particle push. Critical to the push is efficiently accessing mesh field data for each particle within an element. Towards this, the PUMIPic [13] unstructured mesh PIC infrastructure provides a data structure for grouping particles by their parent mesh element and tuning the memory layout for optimal access patterns on CPU vector units and data parallel GPUs. Mesh field control, topological adjacency queries, and geometric model association queries are provided by Omega_h [3, 4]. All these mesh operations, and size field based mesh adaptation, are implemented to execute on CPUs or GPUs via Kokkos [2].

## 4 XGCM PLASMA PIC

Definition of an XGCm simulation begins with a customized version of the SimModeler GUI that generates field following 2D triangular meshes from EFIT data. Figure 1c depicts the X point near the diverter and multiple flux faces whose width is spanned by exactly one mesh element to satisfy numerical method requirements.

Given a field following mesh, XGCm executes the series of four operations depicted in Figure 1a. XGCm is a C++ version of XGC whole device gyrokinetic PIC being designed to execute on many-core CPUs or GPUs with a distributed mesh using PUMIPic. Initial $\delta f$ simulations have been executed on the NERSC Cori system on up to 1024 processes with 127 thousand mesh elements, eight poloidal planes, and 300 million particles. These tests show that the XGCm particle push operation is 25% faster than XGC.

## 5 GITRM IMPURITY TRANSPORT PIC

GITRm impurity transport simulations for tokamak reactors utilizes a revolved extrusion of 2D mesh similar to those used for XGCm. Of the series of PIC operations depicted in Figure 1a, only the field to particle and particle push operations are required in the current simulations as the spatial density of impurities is low enough to model their motion without exchanging information through the mesh.

Initial tests are reporting particle wall collisions that, with the current level of implemented physics, match the reference implementation using structured grids running on GPUs. Figure 1d depicts the PISCES model and the track of a single particle as it is pushed until it collides with the wall.

## 6 CLOSING REMARKS AND FUTURE WORK

Key software to support parallel unstructured mesh fusion simulations have been developed to support new and existing applications. Challenges in mesh generation and geometric model creation have been addressed by functionality developed with Simmetrix. Meshes and models from these tools drive simulations for RF wave progagation using MFEM high order finite elements, plasma turbulence in XGCm PIC, and impruity transport in GITRm PIC. Critical for the PIC codes was the development of methods that efficiently associated particles with mesh elements and their execution on GPUs.

Ongoing efforts for PetraM are extending the python-based interface to call C++ PUMI cuved mesh adaptation routines. When completed, the integration will provide a high-order, in-memory, adaptive capability for executing parallel PetraM RF simulations. XGCm efforts are focused on tuning of operations executed on CPUs while PUMIPic GPU support is matured. Particle data structure developments are being coordinated with the COPA Cabana [12] and Princeton Plasma Physics Laboratory teams. Work on GITRm is focused on implementating particle recombination and ionization with support from developers at the University of Tennessee and the Oak Ridge National Laboratory.

## ACKNOWLEDGMENTS

This research is supported by the National Science Foundation under Grant ACI1533581, and the U.S. Department of Energy, Office of Science, under awards DE-AC52-07-NA27344 (FASTMath SciDAC Institute) and DE-SC0018275 (“Unstructured Mesh Technologies for Fusion Simulation Codes”).

# REFERENCES

[1] B. Cornille, M. J. Lanctot, L. L. Lao, L. C. Appel, O. Meneghini, and C. T. Holcomb. 2013. Validation of EFIT++ MHD Equilibrium Reconstructions on DIII-D. In *APS Meeting Abstracts*. Article JP8.080.

[2] H. Carter Edwards, Christian R. Trott, and Daniel Sunderland. 2014. Kokkos: Enabling manycore performance portability through polymorphic memory access patterns. *J. Parallel and Distrib. Comput.* 74, 12 (2014), 3202 – 3216. https://doi.org/10.1016/j.jpdc.2014.07.003 Domain-Specific Languages and High-Level Frameworks for High-Performance Computing.

[3] Dan Ibanez. 2016. Omega_h GitHub repository. https://github.com/ibaned/omega_h

[4] Daniel Alejandro Ibanez. 2016. *Conformal mesh adaptation on heterogeneous supercomputers.* Rensselaer Polytechnic Institute, Troy, NY.

[5] Daniel A. Ibanez, E. Seegyoung Seol, Cameron W. Smith, and Mark S. Shephard. 2016. PUMI: Parallel Unstructured Mesh Infrastructure. *ACM Trans. Math. Softw.* 42, 3, Article 17 (May 2016), 28 pages. https://doi.org/10.1145/2814935

[6] S. Ku, R. Hager, C.S. Chang, J.M. Kwon, and S.E. Parker. 2016. A new hybrid-Lagrangian numerical scheme for gyrokinetic simulation of tokamak edge plasma. *J. Computational Physics* 315 (2016), 467–475. https://doi.org/10.1016/j.jcp.2016.03.062

[7] LLNL. 2016. MFEM: Modular finite element methods. http://mfem.org

[8] SCOREC RPI. 2019. XGCm GitHub Repo. https://github.com/SCOREC/xgc_scorec

[9] Mark S Shephard, M.W. Beall, R.M. O'Bara, and B.E. Webster. 2004. Toward simulation-based design. *Finite Elements in Anal. and Design* 40, 12 (July 2004), 1575–1598.

[10] Shiraiwa, S., Wright, J. C., Bonoli, P. T., Kolev, T., and Stowell, M. 2017. RF wave simulation for cold edge plasmas using the MFEM library. *EPJ Web Conf.* 157 (2017), 03048. https://doi.org/10.1051/epjconf/201715703048

[11] Simmetrix. 1997. Simmetrix: Enabling Simulation-Based Design. http://www.simmetrix.com/

[12] Stuart Slattery. 2019. Cabana GitHub Repo. https://github.com/ECP-copa/Cabana

[13] Cameron Smith and Gerrett Diamond. 2019. PUMIPic GitHub Repo. http://github.com/SCOREC/pumi-pic

[14] Saurabh Tendulkar, Mark Beall, Mark S Shephard, and KE Jansen. 2011. Parallel mesh generation and adaptation for CAD geometries. In *Proc. NAFEMS World Congr.* NAFEMS, 1–12.

[15] T.R. Younkin, D.L. Green, R.P. Doerner, D. Nishijima, J. Drobny, J.M. Canik, , and B.D. Wirth. 2017. GITR Simulation of Helium Exposed Tungsten Erosion and Redistribution in PISCES-A. In *59th Annual Meeting of the APS Division of Plasma Physics*. https://doi.org/link/BAPS.2017.DPP.UO4.2