2023 PyLith Hackathon Report

Brad Aagaard, U.S. Geological Survey

Table of Contents

Summary

The 3rd Pylith Hackathon was held June 12–17, 2023, at the Colorado School of Mines in Golden, Colorado with funding from the Computational Infrastructure for Geodynamics (CIG). The hackathon involved 17 participants working on 5 different projects to implement new features and create new examples for the PyLith crustal deformation modeling software. The projects included (1) spontaneous rupture using fault friction, (2) extending the poroelasticity implementation, (3) developing two-dimensional (2D) and three-dimensional (3D) examples involving strike-slip faults, (4) integrating PyLith with the cascading adaptive transitional metropolis in parallel (CATMIP) Bayesian inversion framework for use in studies inverting for static fault slip, and (5) adding self-gravitation using the current multiphysics formulation in PyLith. Participants learned how to navigate the PyLith code base (https://github.com/geodynamics/pylith), implement point-wise functions for governing equations and bulk and fault rheologies using the finite-element method, extend the code using the modular, object-oriented design, write examples that demonstrate how to use the new features in PyLith simulations, and implement method of manufactured solutions tests and full-scale tests. The PyLith development team benefitted from discussions with the other participants (contributors) about the technical aspects of the various projects as well as general discussions about PyLith design. The in-person format and 6-day duration allowed the groups to make substantial progress. Participants appreciated the project-based organization of the hackathon and recommended that future hackathons include online meetings of the various projects before the in-person gathering to self-organize and prepare. Sarah Minson (remote) provided technical advice on the use of the CATMIP Bayesian inversion framework, and this type of participation could be expanded to allow additional technical presentations and advice on various topics in future hackathons.

Participants

Brad Aagaard (U.S. Geological Survey, PyLith developer)
Kali Allison (University of California, Davis)
Alexander Berne (Caltech)

Grant Block (University of New Mexico)
Daniel Douglas (New Mexico Tech)
River He (University of Rhode Island)
Matthew Knepley (University at Buffalo, PyLith Developer)
Eric Lindsey (University of New Mexico)
Rishav Malick (Caltech)
Evan Marschall (University of California, Riverside)
Kathryn Materna (U.S. Geological Survey)
Anna Pearson (University of Washington)
Rezga Shakeri (University of Colorado, Boulder)
Matthew Swarr (University of Montana)
Robert Walker (Kegman Incorporated)
Zechao Zhuo (Michigan State University)
Charles Williams (GNS Science, PyLith developer)

Format

The hackathon was held in-person at the Colorado School of Mines in Golden, Colorado. The participants were organized into five groups around a project of mutual interest. Each group worked independently but participated in plenary presentations and discussions on various topics, such as how to use Git, PyLith code layout, PyLith's formulation for governing equations using finite-elements, and debugging strategies. Each day included a morning planning discussion and a wrap-up discussion to monitor the progress of each project, identify issues on which groups needed guidance, and plan discussions for subsequent days.

Tools

We used a shared Google document for meeting notes, including sharing links and creating task lists, as well as two whiteboards for inter- and intra-project discussions.

Participants used Visual Studio Code attached to a Docker container for an integrated software development environment. External dependencies were provided by the Docker container; participants built CIG-related code (pythia, https://github.com/geodynamics/pythia; spatialdata, https://github.com/geodynamics/spatialdata; PETSc, https://petsc.org/; and PyLith) in a persistent Docker storage volume. This made it easy for participants to update the dependencies and environment separately from PETSc and PyLith. The integrated development environment facilitated managing code using Git, navigating the code base, compiling code, and running and debugging tests.

Projects

Spontaneous Rupture

Kali Allison and River He

Kali Allison and River He focused on re-implementation of spontaneous rupture (fault friction) using the PyLith multiphysics formulation. They began by implementing a simple method of manufactured solutions test for a block subjected to shear deformation and a locked fault running through the middle of the block. They based this approach on a pre-existing test, replacing the kinematic fault (prescribed slip) with a dynamic fault (spontaneous rupture). Then they started implementing the dynamic fault and fault friction, beginning with static friction as a simple implementation of a fault rheology. After setting up the C++ classes, they split up the work with Kali focused on implementing the fault objects while River concentrated on setting up the friction models.

Kali completed about 90% of the implementation of the kernels for the residual. The remaining portion for the residual kernels involves setting up and initializing the context containing the parameters. The next step will be implementing the kernels for the Jacobian.

River focused on implementing the kernels for rate and state friction. This choice was strategic, as the insights and methodologies gained from this friction model can be effectively transferred to other friction models later. Updating the state in the rate and state friction model can be challenging, as it involves incorporating the reference parameters and the time step. The current implementation mimics the setup used in updating the viscous strain state variable in the isotropic linear Maxwell bulk rheology provided by PyLith.

Poroelasticity

Grant Block, Daniel Douglas, Rezgar Shakeri, and Robert Walker

This group worked on improving the poroelasticity implementation in PyLith.

Grant Block added the ability to output bulk density from PyLith simulations that use poroelasticity. Although the residual kernels computed bulk density, it was not stored or made available for output. Grant added the kernels for computing bulk density as a derived field so that it could be computed for output. Grant also worked on a simple 3D example case in which a high porosity magma body is pressurized by injecting fluid into a lower porosity mush zone, represented as two concentric spheres in an elastic domain. Grant created a mesh using Cubit (https://cubit.sandia.gov/, https://coreform.com/products/coreform-cubit/) and the corresponding parameter files but ran into a problem where initial pressure conditions were not being set in the poroelastic materials; setting initial pressure conditions for each material is not possible for meshes created using Cubit due to limitations in the mesh metadata, but it is possible for meshes created using Gmsh (https://gmsh.info/). Additional work is needed to update the example to use Gmsh to generate the mesh.

Daniel Douglas learned how to construct a mesh using Cubit and Gmsh and worked on an extension of the existing 2D subduction model example. He added an outer rise normal fault and was able to output porosity from PyLith simulations. He did encounter some difficulties setting up faults with prescribed slip for a simulation using poroelastic bulk rheologies. Daniel proceeded to incorporate data from the Slab2.0 database (https://doi.org/10.5066/F7PV6JNV) and generate a model of an outer rise fault rupturing along a cross section of the Hikurangi subduction zone in New Zealand.

Robert Walker focused on completing an initial implementation of point sources for both fluid injection and moment tensors. The implementation permits easy specification of multiple point sources with variation of the parameters via spatial databases. Additional development is needed to incorporate user-defined time series. Robert also continued to make progress on the poroelasticity fault implementation he has been working on over the past year and identified the need for a more general implementation of updating the fault slip for the kinematic fault.

Rezgar Shakeri worked on a method of manufactured solutions test for poroelastodynamics, which also involved adding acceleration of the solid phase in Darcy's law. The test exposed the need to use an implicit-explicit formulation (for which additional work is needed) rather than a pure explicit formulation, because some of the governing equations involve multiple terms with the time derivative of the solution.

2D and 3D Strike-Slip Faulting Examples

Evan Marschall and Zechao Zhuo

Evan Maschall and Zechao Zhuo developed new strike-slip 2D and 3D examples based on the 2019 Ridgecrest earthquake to help fill in the gap between simple examples and application of PyLith to research problems. The examples use georeferenced nonplanar fault geometry from a coarse representation of the mapped surface rupture. They created both Cubit and Gmsh files and common and mesh-specific parameter files. The examples increase in complexity (consistent with the existing examples) from slip on a single fault to spatially variable slip on two faults.

PyLith+CATMIP Integration

Eric Lindsey, Rishav Malick, Kathryn Materna, and Anne Pearson

This group focused on extending the PyLith-CATMIP integration code in Brad Aagaard's branch of CATMIP (https://doi.org/10.1093/gji/ggt180) to accept Green's functions that have two components of slip instead of just one and developing corresponding examples of inverting for static slip on a small 3D strike-slip fault. They added a nominal slip rake angle parameter and prior probability density functions for rake-parallel and rake-perpendicular components. They also implemented tests that exposed bugs and proceeded to fix them. Analogous to the existing 2D strike-slip example, they created a Tikhonov zero-order regularized linear inverse optimization in Python and a CATMIP inversion. The forward model uses a mesh from Gmsh and variable slip. The example also includes post-processing scripts to convert the CATMIP output from rake-parallel and rake-perpendicular components into left-lateral and reverse components, consistent with the Green's function calculation.

Self-Gravitation

Alexander Bearne and Matthew Swarr

Alexander Bearne and Matthew Swarr leveraged the multiphysics framework in PyLith to implement a new material for modeling a body force produced by self-gravitation, with intended application to planetary science and surface loading problems. They closely followed the implementation of the incompressible elastic material, using displacement and gravitational potential solution fields instead of displacement and pressure solution fields. As a first step, the displacement and gravitational potential fields are uncoupled; the method of manufactured solutions applies a body force to the sphere that is computed using the solution to Poisson's equation for the interior of a spherical body. Additional work is needed to couple the displacement and gravitational potential fields so that simulations of surface loading in PyLith can be benchmarked with existing semi-analytic codes.

Disclaimer