Poseidon Manual

Nick Roberts

January 2, 2024
Poseidon v?
Manual v0.1

Contents

Chapter 1

Poseidon Workflow

For the CFA metric problem, Poseidon follows the workflow shown in figure 1.1.

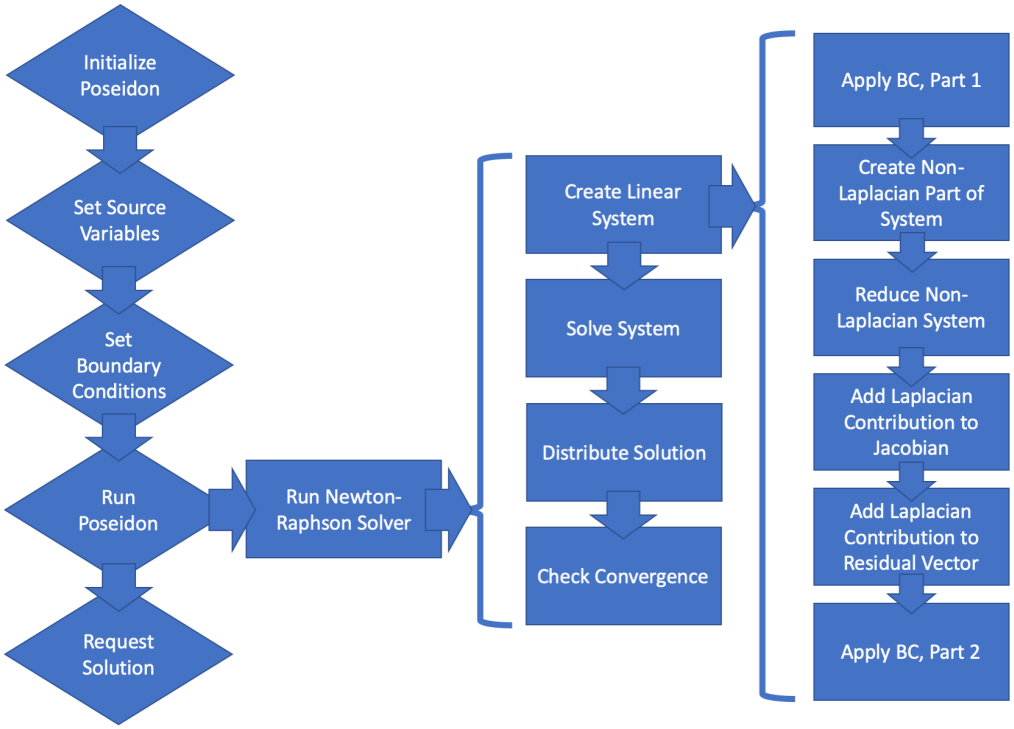

Figure 1.1: Workflow diagram for Poseidon's CFA solve. The steps represented by diamonds correspond to routines called by a user, while the steps in squares are those called internally.

The first step is to initialize Poseidon by calling the subroutine

Chapter 2

Poseidon Input Parameters

Currently, there are three files in the Params directory of the Poseidon code. The first two "Poseidon_Params.d" and "Driver_Params.d" are the only ones currently used. The file "CFA_Coeffs.c" is part of an in-development feature and should be ignored for now.

These two parameter files allow for run-time setting of many parameters that can effect how the Poseidon code or a test problem run.

2.1 Poseidon Parameters

The Poseidon parameters dictate the limits on the solver methods as well as how Poseidon decomposes its domain. Many of the Poseidon's domain decomposition parameters are dependent on the domain decomposition parameters set by the calling program. For example NBTROW and NTEPB, the number of block theta rows, and the number of theta elements per row rescpectively, are related to the total number of theta elements in the calling programs domain decomposition by the equation, $N_{te} = NBTROW * NTEPB$.

The parameters in "Params/Poseidon_Params.d" are

I. DIM - This parameter sets the number of dimensions to be used in the Poseidon solver.

   A. *** This should remain 3 for the time being ***

      i. Running a 1-D problem is done by setting LLIMIT below to 0.

      ii. Running a 2-D problem is done in 3-D. Future, 2-D Optimization will utilize this setting.

II. DEGREE - This parameter sets the degree of the Finite Element expansion in the radial direction.

III. LLIMIT - This parameter sets the truncation limit on L for the spherical harmonic expansion.

IV. PPROC - This sets the number of processes Poseidon will use.

V. NSHELL - This sets the number of radial shells that Poseidon will break the domain into.

VI. NSSHEL - This sets the number of subshells within each radial shell. This is currently used to allow more processes per shell into the PETSc solve.

   A. The value of this parameter must be less than or equal to PPROC/NSHELL.

VII. NBPSHL - This sets the number of blocks that each radial shell is broken into.

VIII. NBTROW - This sets the number of blocks that make up the rows in the theta dimension of each shell.

IX. NBPCOL - This sets the number of blocks that make up the columns in the phi dimension of each shell.

   A. NBPSHL = NBTROW*NBPCOL

X. NREPS - This sets the number of radial elements per shell.

XI. NREPSS - This sets the number of radial elements per subshell

   A. NREPS = NSSHEL*NREPSS

XII. NTEPB - This sets the number of theta elements per block

   A. Total Number of Theta Elements = NBTROW*NTEPB. This must must equal the number of theta elements set in Poseidon_Initialize.

XIII. NPEPB - This sets the number of phi elements per block

   A. Total Number of Phi Elements = NBPCOL*NPEPB. This must must equal the number of phi elements set in Poseidon_Initialize.

XIV. PRQ - This sets the number of radial quadrature points used in all integrations.

   A. Should be set ¿= DEGREE+1, otherwise errors due to imprecise integration will be present.

XV. PTQ - This sets the number of radial quadrature points used in all integrations.

   A. Currently needs to be ¿= 3, even for 1-D cases. Future code changes will eliminate this need.

XVI. PPQ - This sets the number of radial quadrature points used in all integrations.

XVII. MI - This sets the maximum number of iterations for the Newton-Raphson solver.

XVIII. CC - This sets the convergence criteria for the Newton-Raphson solver.

XIX. WRTTT - This controls if a timetable is printed in each iteration, and run report.

XX. WRTIR - This controls if a report is printed after each iteration.

XXI. IRNS - This controls the number of sample points used in each iteration report.

XXII. WRTRS - This controls if a set of results files are printed.

A. The files that are output if this setting is one are,

i. Base_Sources.out - The indirect sources variables, density and velocity, for the test run.

ii. Sources.out - The direct source variables, total energy density, momentum density, and trace of the spatial stress-energy tensor, for the test run.

iii. R_Mesh.out - The radial grid point locations used.

iv. Solution.out - The analytic results for the test run.

v. Results.out - The solver results for the test run.

2.2 Test Problem Parameters

The test problem parameters are currently set in "Params/Driver_Params.d". These variables set aspects of the domain decomposition, as well as test problem parameters.

The parameters in "Params/Driver_Parameters.d" are

I. RE - This sets the number of radial elements in the domain decomposition.

II. CE - This sets the number of radial elements that are considered in the core.

A. This is solely used if the Split mesh type is selected below.

III. TE - This sets the number of theta elements in the domain decomposition.

IV. PE - This sets the number of phi elements in the domain decomposition.

V. RQ - This sets the number of source points in the radial direction per element.

VI. TQ - This sets the number of source points in the theta direction per element.

VII. PQ - This sets the number of source points in the phi direction per element.

VIII. DIM - This sets the dimension of the problem.

   A. Leave this as 3 for now.

IX. PROC - This sets the total number of processes used by the driver program.

   A. PROC = yPROC*zPROC

X. yPROC - This sets the number of processes in the theta direction.

XI. zPROC - This sets the number of processes in the phi direction.

XII. MT - This sets the mesh type.

   A. 1 = Uniform, 2 = Logarithmic, and 3 = Uniform spacing for the first CE radial elements, and logarithmic for the rest.

XIII. IR - This sets the inner radius of the problem.

XIV. CR - This sets the radius where the mesh changes from uniform to logarithmic is MT = 3.

XV. OR - This sets the outer radius of the problem.

XVI. SST - This sets the Yahil self-similar profile parameter, time.

XVII. SSK - This sets the Yahil self-similar profile parameter, kappa.

XVIII. SSG - This sets the Yahil self-similar profile parameter, gamma.

Chapter 3

Running a Poseidon Example Problem

3.1 The Yahil Self-Similar Collapse Test

3.1.1 The Yahil Self-Similar Profile

This test uses the self-similar stellar collapse profile created by Amos Yahil [**?**]. This profile models the self-similar collapse of a stellar core with a polytropic equation of state, $P = \kappa\rho^{\gamma}$. To use, one sets three parameters, κ, γ, and t, where κ is the polytropic constant, γ is an effective adiabatic index, and t is the time until the central density becomes infinite. In practice, κ and γ, are set and t is varied.

3.1.2 Running the Test

Currently, this is the only test that Poseidon is set up to run. In the future, more tests will be added.

I. Edit the makefiles to suit the system you are running on.

   A. In "makefile", comment/uncomment lines to turn settings on or off.

      i. DIMENSION=3D, should be left as alone. Lesser dimensions are currently achieve by manually setting Poseidon parameters in a later step.

      ii. NPROCS is the number of processes you will run the driver with. This can be varied.

      iii. MACHINE_NAME should be changed to reflect the machine your are running with. See below for more details on this.

      iv. CMODE can be altered.

      v. MPI_MODE, PETSC_MODE, and HDF5_MODE all remain on.

B. You do not need to alter "makefile_objects" unless you are adding a new file to be compiled.

C. In "makefile_dictionary", Add or alter entires corresponding to MACHINE_NAME selected in "makefile" to properly link to a Fortran compiler, and the MPI, OpenMP, PETSc, and HDF5 libraries.

i. See subsection 3.1.3 later in this chapter for more details on the variables found here.

II. Compile the test with the command:

```
$ make yahil
```

III. Set the test parameters in the files, "Params/Poseidon_Params.d" and "Params/Driver_Params.d".

A. See chapter 2 for details on the various parameters.

B. I've included a directory labeled "Test_Parameters", each of the subdirectories contains parameters files that can be copied into the "Params" directory and used to run a few preset tests.

IV. Run the test driver with

```
$ make run_yahil
```

A. If you are doing a multiprocess run, make sure to alter NPROCS in makefile.

V. The results are in the "OUTPUT" directory.

3.1.3 makefile_dictionary

Makefile_dictionary uses the variable $\$MACHINE\_NAME$ to construct all the linkers and inputs needed for the compiler for a given system. A user selects a profile by uncommenting the desired system and commenting out all others in "makefile". Makefile_Dictionary then selects the profile settings associated with that machine. If a machine does not have a profile one can be created by adding

```
$MACHINE_NAME = New_Machine
```

to makefile. Then in makefile_dictionary create the entires

```
FORT_$(New_Machine) =
STD_DEBUG_$(New_Machine) =
STD_OPTIMIZE_$(New_Machine) =
OPENMP_$(New_Machine)_ON =
OPENMP_$(New_Machine)_OFF =
PETSC_$(New_Machine)_ON =
PETSC_$(New_Machine)_OFF =
```

```
HDF5_$(New_Machine)_ON =
HDF5_$(New_Machine)_OFF =
MPI_$(New_Machine)_Lib =
MPI_$(New_Machine)_Include =
LAPACK_$(New_Machine) =
```

$FORT\_\$(New\_Machine)$ points to the MPI capable Fortran compiler.

$STD\_DEBUG\_\$(New\_Machine)$, and $STD\_OPTIMIZE\_\$(New\_Machine)$ are the debug and optimization flags for the compiler, respectively.

$OPENMP\_\$(New\_Machine)\_ON$ should contain the flag(s) required to compile code with openmp, while $OPENMP\_\$(New\_Machine)\_OFF$ is usually left blank.

$PETSC\_\$(New\_Machine)\_ON$ should contain the flag(s) required to compile code with PETSc. _OFF is again left blank. If necessary, one can specify the PETSc ARCH, and DIR locations.

$HDF5\_\$(New\_Machine)\_ON$ should contain the flag(s) required to compile code with HDF5. _OFF is again left blank.

$MPI\_\$(New\_Machine)\_Lib$ and $MPI\_\$(New\_Machine)\_Include$ should contain the linkers to the MPI library and include directories.

$LAPACK\_\$(New\_Machine)$ should contain the flag(s) require to use the LAPACK library.

3.1.4 Results