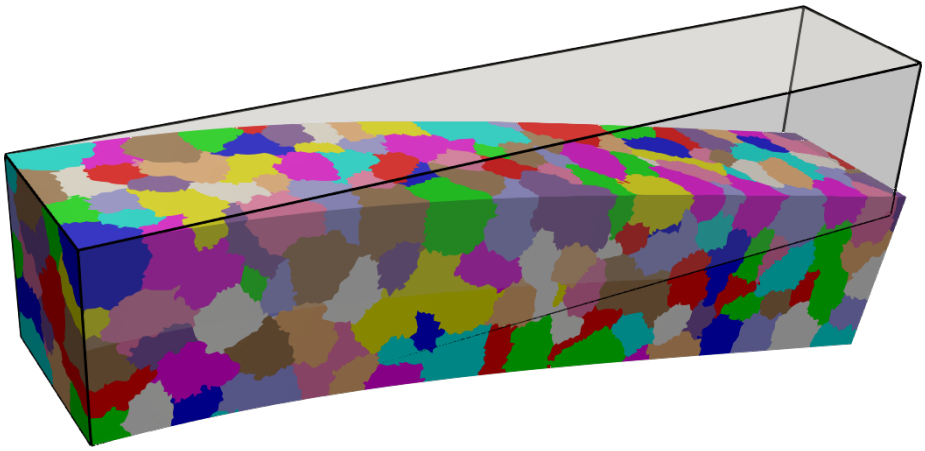
PSD: a Parallel Structural Dynamics solver



PSD is a finite elements-based solid mechanics solver with capabilities of performing High Performance Computing (HPC) simulations with billions of unknowns. The kernel of PSD is wrapped around FreeFEM for finite element discretization, and PETSc for linear algebra/Preconditioning. PSD solver contains straightforward supports for *static* or *dynamic* simulations with *linear* and *nonlinear* solid mechanics problems. Besides these *hybrid-phase field fracture mechanics* models have also been incorporated within PSD. For dynamics the *genralized-α model* for time discretization is used, this models enable straightforward use of Newmark-β, central difference, or HHT as time discretization. PSD uses sate-of-the art domain-decomposition paradigm via *vectorial finite elements* for parallel computing and all solvers are proven to scale quasi-optimally. PSD has proven scalabilty uptill 24,000 cores with largest problem solved containing over 5 Billion unknowns.

Besides the parallel suite, PSD also includes a sequential solver which does not require PETSc.

PSD works for two and three dimensional problems only. Unstructured meshes (triangular for 2D and tetrahedral for 3D) are supported in MEDIT's `.mesh` format or Gmsh's `.msh` format. PSD post processing is done via `.vtk` and `.vtu` files of the ParaView platform.

Installation procedure

PSD is built to work with Linux platforms. (Windows comping up soon)

Dependencies

- automake

- FreeFEM
- PETSc (optional)
- Gmsh
- gnuplot (optional)
- C++
- git

Now that I have all the dependencies what next

- Go ahead and grab the latest copy of PSD. The code is hosted on CEA's internal git repository.

  ```
  git clone https://codev-tuleap.intra.cea.fr/plugins/git/hpcseism/freefem.git PSD-Source
  ```

- Autoconf PSD within the cloned folder

  ```
  autoreconf -i
  ```

- Configure PSD within the cloned folder

  ```
  ./configure
  ```

  Note: `./configure` will install PSD in `/usr/local/bin` and you would need sudo rights to perform installation, for non sudo users or for local install consider chaning directory of installation. To change this directory use `--prefix=Your/Own/Path` with `./configure`. Remember to add `Your/Own/Path` to your `$PATH` variable, you can do so by `export PATH=$PATH:Your/Own/Path`

Note: `./configure` will try to look for installation of FreeFEM and Gmsh in `usr/bin/` or `usr/local/bin/` directories. If you have these packages installed in some other directory this should be specified during `./configure` by using flags `--with-FreeFEM=` and `--with-Gmsh=`. For example, if FreeFEM is installed at `home/FreeFem/bin` and Gmsh in `home/Gmsh/bin` then one should use

```
./configure --with-FreeFEM=home/FreeFem/bin  --with-Gmsh=home/Gmsh/bin
```

- Make PSD directives

```
make
```

- Install PSD

```
sudo make install
```

Note : You should not use `sudo` if you have used `--prefix` during the `./configure`

Now you should have the PSD solvers installed.

Additional FreeFEM tweak for brittle fracture mechanics

Note that this procedure is only recommended if you are interested in using PSD for brittle fracture problems. In your FreeFEM source files (installation) go to `src/femlib/fem.cpp` , in this file replace the lines of code

```
R seuil=hm/splitmax/4.0;
```

by the following

```
R seuil=hm/splitmax/4.0/1000.0;
```

Using procedure

The parallel/sequential solvers can be run on Linux platforms and can be used with command line options to control the solver. In order to make your choice of physics, command line options need to be typed right into the bash.

To perform a simulation three steps are involved

Step 1: Setting up the solver

Now that the mesh has been loaded. Its time to set up the solver. Open the `terminal` window at the location of the solver, i.e., `$HOME/PSD/Solver.` Then run the following command in the `terminal`.

```
PSD_PreProcess [Options-PSD]
```

Via the command line options you will embed the physics within the solver. This step generates a bunch of `.edp` files which are native to FreeFEM and additionally prints out instructions on what to do next. You then need to open and edit couple of these files via your favorite text editor, which could be `vim`, `gedit` ,`Notepad++`, etc. To facilitate the edit process for your will have to go through the instructions printed on the terminal.

For example to generate a sequential 2D elasticity solver for a problem with body force and one Dirichlet border use

```
PSD_PreProcess -dimension 2 -bodyforceconditions 1 -dirichletconditions 1
```

Step 2: Launching the solver

Now you are all set to run your simulation. To do so you will need to do the run the following in the `terminal`:

if you complied a parallel PSD version

```
PSD_Solve -np $N Main.edp -v 0 -nw
```

if you complied a sequential PSD version

```
PSD_Solve_Seq Main.edp -v 0 -nw
```

- In the parallel command **$N** is an `int` value, i.e., number of processes that you want to use for performing the simulation in parallel.
- Additional flag `-wg` may be required while launching the solver, this is in case debug mode is on.

Sit back and be amazed at the speed of the solver.

[Options-PSD] explained

These are a set of commandline flags/options that control your simulation. You can think of it as a way to talk to the solver. Here is a table that lists out some of the options that are available (for full list see documentation).

Option	Type	Comment
`-help`	`[bool]`	To activate helping message on th
`-debug`	`[bool]`	To activate debug mode with a wait plot.
`-useGFP`	`[bool]`	To activate use of GoFastPlugins. A suite
`-useRCM`	`[bool]`	To activate mesh level renumbering via Rev
`-pipegnu`	`[bool]`	Use to activate realtime pipe plotting
`-timelog`	`[bool]`	To activate time logging the different pha
`-supercomp`	`[bool]`	Use when using a super computer withou
`-bodyforce`	`[bool]`	To activate volumetric source term (
`-vectorial`	`[bool]`	To use vectorial finite element r
`-pointprobe`	`[bool]`	To postprocess point field
`-sequential`	`[bool]`	To solve via a sequential sol

Option	Type	Comment
-energydecomp	[bool]	To activate energy decomposition, only
-doublecouple	[bool]	To activate double couple source for
-constrainHPF	[bool]	To use constrain condition in hybrid ph
-top2vol-meshing	[bool]	To activate top-ii-vol point source meshing
-getreactionforce	[bool]	To activate routine for extraction react
-plotreactionforce	[bool]	To activate realtime pipe plotting us
-crackdirichletcondition	[bool]	To activate pre-cracked surface I
-dirichletpointconditions	[int]	Number of Dirchlet point
-dirichletconditions	[int]	Number of Dirchlet bounda
-bodyforceconditions	[int]	Number of regions acted upon by
-tractionconditions	[int]	Number of Neumann/traction b
-parmetis_worker	[int]	Number of parallel workers used by ParMe
-lagrange	[int]	Lagrange order used for FE spaces. 1 fo
-dimension	[int]	Accepts values 2 or 3. Use 3 for 3D. and
-mesh	[sting]	Provide mesh to be solved by PS
-timediscretization	[sting]	Time discretization type. Use “generalized-alpha” or “newmark-b
-nonlinearmethod	[sting]	Nonlinear method type. Use “Picard” or “
-reactionforce	[sting]	Reaction force calculation method “stress-based
-doublecouple	[sting]	Soil dynamics double couple. Use “force-based”
-postprocess	[sting]	To communicate what to postprocess “u”, “v”, “a”, “uv”
-partitioner	[sting]	Mesh partitioner could be “metis” “parm
-problem	[sting]	Interested problem. Use “linear-elasticity”, “damage”, “el
-model	[sting]	Interested model. Use “hybrid-phase-fie

To report bugs, issues, feature-requests contact:

- **mohd-afeef.badri@cea.fr**
- **mohd-afeef.badri@hotmail.com**