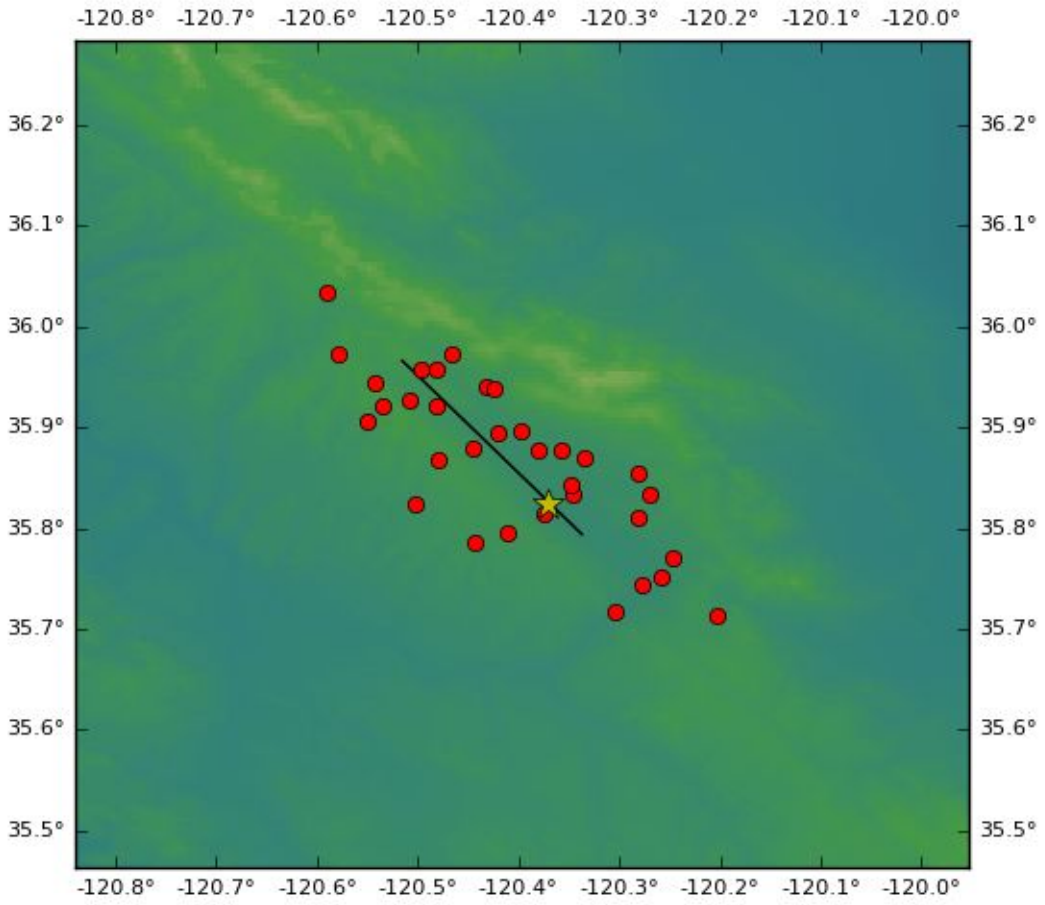
SCEC Open-Source Software and Data Access Workshop



Software Tutorial and Development Team

Philip Maechling	SCEC Software Director	maechlin@usc.edu
Tran Huynh	SCEC Operations Director	huynht@usc.edu
Christine Goulet	SCEC Special Projects Director	cgoulet@usc.edu
Kevin Milner	OpenSHA Developer	milner@usc.edu
Fabio Silva	Broadband Platform Developer	fsilva@usc.edu
Mei-Hui Su	SCEC Community Model Developer	mei@usc.edu
Silvia Mazzoni	DesignSafe Data Access Developer	
Scott Callaghan	CyberShake Developer	scottcal@usc.edu
Edric Pauk	SCEC Web Developer	pauk@usc.edu

Conventions Used in Document:

Description text is written in size 11 pt. Text typed on a computer is written in 9pt. A leading dollar sign $ indicate a cursor on a bash terminal command line, on a Linux system like this:

```
$ pwd

/home/scec

$ ls
anaconda2 Desktop Downloads opensha Public ucvm_bin Videos
bbp Documents Music Pictures Templates ucvm_src
```

The user can type these command in a terminal window on a Linux system with a properly configured version of SCEC's open source scientific software including the Unified Community Velocity Model (UCVM) (Sept 2018 Version), the SCEC Broadband Platform (BBP) (Sept 2018 Version), and OpenSHA (Sept 2018 Version).

Table of Contents

1. Introduction

2. Unified Community Velocity Model (UCVM)

3. SCEC Broadband Platform (BBP)

4. SCEC NHERI DesignSafe-CI Ground Motion Data Portal

5. SCEC OpenSHA Seismic Hazard Analysis Tool

6. SCEC Project Specific Data Management — The UGMS MCER Case

7. The SCEC Software Clinic

1. Introduction

This is a user's guide to the SCEC Open Source Software and Data Access Workshop. This guide describes the software demonstrations that will presented at the Workshop. The detailed descriptions of the software demonstration in this guide may help users follow-along with the software demonstration as they are given at the workshop. This guide should provide enough information so that users can run the software demonstrations shown at the workshop on their own computers, using their own version of the SCEC software tools.

2. Unified Community Velocity Model (UCVM)

The Unified Community Velocity Model (UCVM) software framework is a collection of open-source scientific software tools designed to support earth scientists, civil engineers, and other groups interested in detailed information about earth properties. The seismic velocity models currently available through UCVM are primarily California high resolution regional scales seismic velocity models that provide Vp, Vs, and density for a defined geographic volume for a region. One important use of UCVM is by researchers performing high resolution 3D wave propagation simulations of California and other regions.

2.1. Laptop - UCVM Demo 1 - Interactive Query Point of CVM and Query CVM Points in File :

2.1.1. Initial Computing Environment

1. A Linux computer (CentOS 7.5) with UCVM dev branch I
2. Environment Variables set including:
   a. LD_LIBRARY_PATH
   b. PYTHONPATH
3. User cd into a Working Directory
   Users selects working directory and then use path to installation of UCVM binaries
4. GTL set to off for CCA and CVM-H models
5. Lat/lon specification are lat,lon with no spaces
6. Input files latlons.txt exists on system
7. Default values are used for query by depth, rather than query by elevation

2.1.2. Science Use Case

Determine Vp, Vs, and density for one, or more, lat/lon/depth point in southern California.

2.1.3. Use Specified Simulation Input Variables:

cmd: ucvm_query

ucvm_config_file: -f ucvm.conf

ucvm_model: -m cvmh

Input lat/lon/depth: -116.54, 33.82, 0

2.1.4 cmdline:

$ cd /home/scec/ucvm/work

$ ../bin/ucvm_query -f ../conf/ucvm.conf -m cvmh

Using Geo Depth coordinates as default mode.

-118.0 34.0 0.0 <CTRL -D>

or

$ ../bin/ucvm_query -f ../conf/ucvm.conf -m cvms < ../tests/test_latlons.txt > ucvm_test_results.txt

2.1.5. Expected Runtime:

Runtime varies by CVM, number of query points, and order of query points. Individual points may take 10-20 seconds to return values.

2.1.6. Simulation Output Results:

Returns a list of input parameters returned with additional columns of information, including Vp, Vs, and rho. Output file from above contains:

cd /home/scec/ucvm
./bin/ucvm_query -f ./conf/ucvm.conf -m cvms < ./tests/test_latlons.txt
You will then see the following output:
Using Geo Depth coordinates as default mode.
-118.0000 34.0000 0.000 280.896 390.000 cvms 696.491 213.000 1974.976 none 0.000 0.000 0.000 crust 696.491 213.000 1974.976
-118.0000 34.0000 50.000 280.896 390.000 cvms 1669.540 548.000 2128.620 none 0.000 0.000 0.000 crust 1669.540 548.000 2128.620
-118.0000 34.0000 100.000 280.896 390.000 cvms 1683.174 603.470 2130.773 none 0.000 0.000 0.000 crust 1683.174 603.470 2130.773
-118.0000 34.0000 500.000 280.896 390.000 cvms 3097.562 1656.495 2354.105 none 0.000 0.000 0.000 crust 3097.562 1656.495 2354.105
-118.0000 34.0000 1000.000 280.896 390.000 cvms 3660.809 2056.628 2443.042 none 0.000 0.000 0.000 crust 3660.809 2056.628 2443.042

Column Number	ucvm_query Parameter	Description	Input/Output Input 1	Input 2	Input 3
	Longitude	Coordinates provided as input parameters to the ucvm_query program	−118.286	118.286	−122.000
	Latitude	Coordinates provided as input parameters to the ucvm_query program	34.033	34.033	34.033
	Depth	Coordinates provided as input parameters to the ucvm_query program	100.0	1000.0	10,000.0
			Output 1	**Output 2**	**Output 3**
1	Longitude	Input longitude in decimal degrees	−118.2860	−118.2860	122.0000
2	Latitude	Input latitude in decimal degree	34.0330	34.0330	34.0330
3	Z	Input depth (or elevation) in meters	100.000	1000.000	10,000.000
4	Elevation	Returned elevation (m) from UCVM digital elevation model	60.386	60.386	−3,791.911
5	V_{S30}	Returned V_{S30} (m/s) from a Wills–Wald site type to V_{S30} relationship (for points in California), or from Wald topography relation (outside of California)	280.000	280.000	180.000
6	CVM Abbreviation	CVM abbreviation identifying which CVM from the input CVM list provided the returned material properties	cvmh	cvmh	cvms
7	V_P	Returned *P*-wave velocity (m/s) from the CVM in column 6	1935.708	2533.139	6,300.000
8	V_S	Returned *S*-wave (m/s) from the CVM in column 6	496.300	1011.327	3,637.307
9	ρ	Returned density (kg/m^3) from the CVM in column 6	1875.844	2103.371	2,859.770
10	GTL name	GTL algorithm used to modify V_P, V_S, and ρ values in the GTL depth range	elygtl	None	None
11	GTL V_P	Returned GTL V_P (m/s) at input point	0.000	0.000	0.000
12	GTL V_S	Returned GTL V_S (m/s) at input point	280.000	0.000	0.000
13	GTL ρ	Returned GTL ρ (kg/m^3) at input point	0.000	0.000	0.000
14	Combining algorithm	Name of algorithm used to combine base model V_P, V_S, and ρ with returned GTL values for V_P, V_S, and ρ	ely	crust	crust
15	Combined V_P	Returned V_P based on combined base model and GTL values	1987.721	2533.139	6,300.000
16	Combined V_P	Returned V_S based on combined base model and GTL values	432.608	1011.327	3,637.307
17	Combined ρ	Returned ρ based on combined base model and GTL values	1899.856	2103.371	2,859.770

2.1.7. Usage Notes:

Multiple query points can be listed, as inputs

Users must interpret the UCVM return parameters, and identify which columns of information they need for their application.

2.1.8. Use Case Variations:

- Query with different point
- Query with multiple points
- Query with different CVM
- Query with tiled CVMs
- Query with points in file
- Query output results to file

2.2 Laptop - UCVM Software Demo 2 - Plot Vs Map at 100m depth using CVM-H:

2.2.1. Initial Computing Environment

1. Plotting utilities must be run from ucvm/utilities directory, so users must cd into ucvm_bin/utilities directory.

2.2.1. Science Use Case

Create horizontal map of Vs for a region without a background model (e.g. CVM-H). Output map to a file with smooth color scale.

2.2.2. User specified Simulation Input Variables

cmd: plot_horizontal_slice

ucvm_config_file: -f ucvm.conf

ucvm_models: -c cvmh

bottom left: -b 30.0,-120.0

upper right: -u 36.5,-113.0

spacing (degrees): -s 0.01

depth (meters): -e 500.0

data type to plot (vp, vs, rho, poisson) -d vs

color scale: (s -smooth, d - discrete): -a s

output_file: -o cvmh_vs_map.png

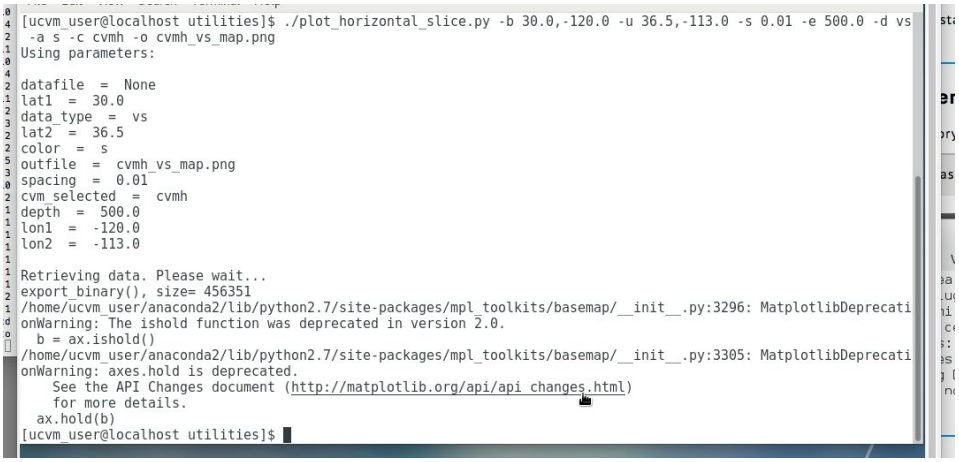
2.2.3 cmdline

User moves to the utilities directory to run the plotting scripts.

```
$ cd /home/scec/ucvm/utilities
$ ./plot_horizontal_slice.py -b 30.0,-120.0 -u 36.5,-113.0 -s 0.01 -e 1000.0 -d vs -a s -c cvmh -o cvmh_vs_map.png
```

2.2.3. Expected Runtime

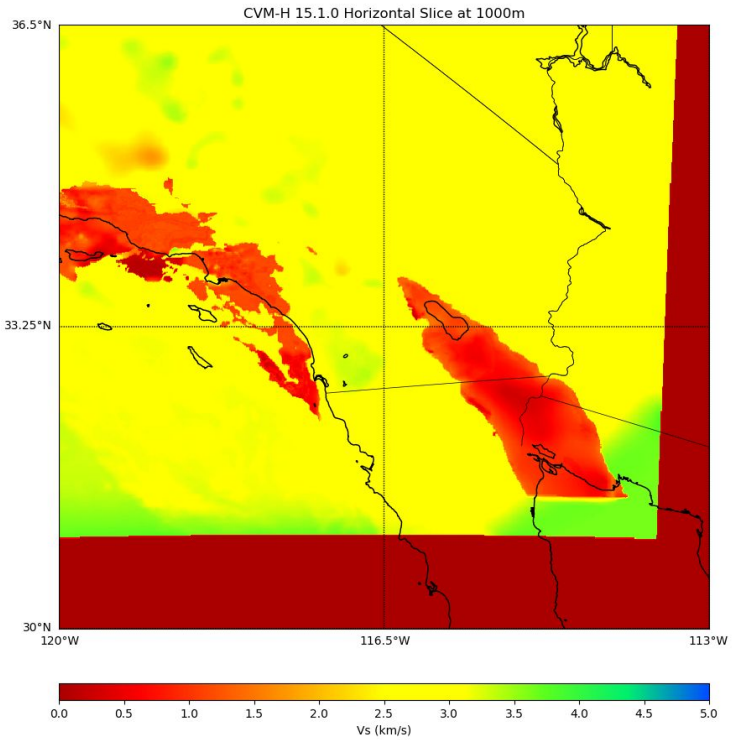
Regional-scale plots with spacing 0.01degree (~1km) resolution may take 1-2 minutes to return plot. Prints runtime warnings but no errors.



no errors.

2.2.4. Simulation Output Results

Produces a map png file named “cvmh_vs_map.png” in the UCVM utilities directory. It can be viewed with a image viewer on the linux system. Also outputs a .json file with the discrete points used, and a .bin file with the data values plotted at each point.



2.2.5. Use Case Variations

- Different velocity model (or models)
- Different depth

- Different region
- GTL setting

2.3 Laptop - UCVM Software Demo 3 - Plot Vs Cross Section 0 to 1000m depth using CVM-H

2.2.1. Initial Computing Environment

1. Plotting utilities must be run from ucvm/utilities directory, so users must cd into ucvm_bin/utilities directory.

2.2.1. Science Use Case

Create vertical cross section showing Vs for a region from surface to 500m depth for CVM-H. Output map to a file with discrete color scale.

2.2.2. User specified Simulation Input Variables

cmd: plot_cross_section
ucvm_config_file: -f ucvm.conf
ucvm_models: -c cvmh
bottom left: -b 34.0,-118.75
upper right: -u 34.0,-117.7
horizontal spacing (meters): -h 100.0
vertical spacing (meters): -v 20
starting depth (meters): -s 0.0
ending depth (meters): -e 500.0
data type to plot (vp, vs, rho, poisson) -d vs
color scale: (s -smooth, d - discrete): -a d
output_file: -o cvmh_cross_section.png

2.2.3 cmdline

User moves to the utilities directory to run the plotting scripts.

$ cd /home/scec/ucvm/utilities
$ ./plot_cross_section.py -b 34.0,-118.75 -u 34.0,-117.5 -h 100 -v 20 -d vs -c cvmh -s 0.0 -a d -e 500.0 -o cvmh_cross_section.png

2.2.3. Expected Runtime

Regional-scale plots with spacing 0.01degree (~1km) resolution may take 1-2 minutes to return plot. Prints runtime warnings but no errors.

[scec@localhost utilities]$ ./plot_cross_section.py -b 34.0,-118.75 -u 34.0,-117.5 -h 100 -v 20 -d vs -c cvmh -s 0.0 -a d -e 500.0 -o cvmh_cross_section.png

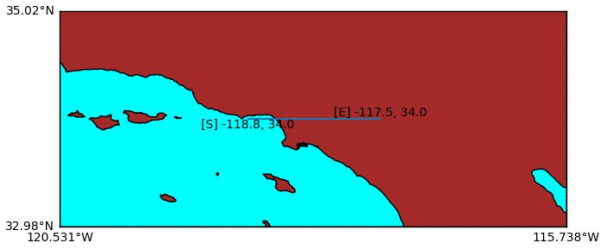
Using parameters:

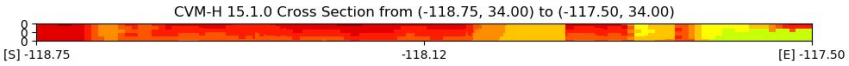
datafile = None
lat1 = 34.0
data_type = vs

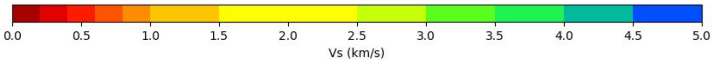
```
lat2 = 34.0
cvm_selected = cvmh
outfile = cvmh_cross_section.png
ending_depth = 500.0
horizontal_spacing = 100
color = d
gate = 2.5
lon1 = -118.75
vertical_spacing = 20
lon2 = -117.5
starting_depth = 0.0

Retrieving data. Please wait...
/home/scec/anaconda2/lib/python2.7/site-packages/mpl_toolkits/basemap/__init__.py:1709:
MatplotlibDeprecationWarning: The axesPatch function was deprecated in version 2.1. Use Axes.patch instead.
  limb = ax.axesPatch
/home/scec/anaconda2/lib/python2.7/site-packages/mpl_toolkits/basemap/__init__.py:1712:
MatplotlibDeprecationWarning: The axesPatch function was deprecated in version 2.1. Use Axes.patch instead.
  if limb is not ax.axesPatch:
export_binary(), size= 30030
```

2.2.4. Simulation Output Results

Produces a map png file named “cvmh_cross_section.png” in the UCVM utilities directory. It can be viewed with a image viewer on the linux system. Also outputs a .json file with the discrete points used, and a .bin file with the data values plotted at each point.







2.2.5. Use Case Variations

- Different velocity model (or models)
- Different depth
- Different region
- GTL setting

2.4 Cluster – UCVM Software Demo 4 - Discretize a Southern California Volume, and Create Simulation mesh with properties from CVM-H.

2.4.1. Initial Computing Environment

1. A Linux computer (CentOS 7.5) with UCVM dev branch I
2. Environment Variables set including:
   a. LD_LIBRARY_PATH
   b. PYTHONPATH
3. UCVM built using MPI libraries

2.4.1. Science Use Case

Discretize a region, and create a binary mesh file, for use in a wave propagation simulation.

2.4.2. User specified Variables Consideration

- User specifies output Mesh format (i,j,k,vp,vs,rho) (vp,vs,rho,qp,qs) (i,j,k,vp,vs,rho,qp,qs) with qp, and qs values based on this formula (qs = 50.0 * (vs / 1000.0)); (qp = 2.0 * qs;)

- Vertex setting sets the origin at exactly the given point, while Center setting moves the lat/lon point to the center of the cell and queries properties there. For example, given a origin at 34.00, -118,0, these two grid files are produced:

  Center result:
  0, 0 : -117.994639, 34.004553
  1, 0 : -117.983811, 34.004640
  2, 0 : -117.972984, 34.004726

  Vertex result:
  0, 0 : -118.000000, 34.000000
  1, 0 : -117.989173, 34.000088
  2, 0 : -117.978346, 34.000174

- If ucvm2mesh-mpi queries a point, and none of the velocity model had data for that point, ucvm2mesh-mpi exits with an error. For this reason, add a background model in your model list if you query points outside a CVM.

- When you specify a min Vs and min Vp, ucvm2mesh-mpi only checks the min Vs value. If it finds Vs lower than the min Vs, it sets both the Vs and Vp to the min values specified in the ucvm2mesh config file.

  ```
  if (vs < vs_min)
  {
          vs = vs_min;
          vp = vp_min;
  }
  ```

- Users specify the projection the want to use for the output material properties. Output projection specifications are based on the proj (https://proj4.org/) Python projection library. An example output projection string used by Proj4 software includes:

2.4.2. User specified Simulation Input Variables

User defines the mesh they want to construct using a UCVM mesh configuration file. This specified the geographical volume, the resolution of the mesh, mesh output format and

cluster computing specifications about where output files should be written, and how many processors to use.

- UCVM conf file
- Gridding cell centered or vertex
- Spacing of cells
- Projection and Mesh origin lat/lon
- Vs/Vp minimum
- Output grid (lat/lon depth pts) and Mesh (lat,lon,depth, properties) files,

User create a ucvm2mesh.conf file that describes the mesh they want, based on the scientific application of the simulation.

```
-bash-4.2$ more README.txt
Test Case 1:
Confirm that the ucvm2mesh and ucvm2mesh-mpi produce the same results. Thhen print the mesh point using dumpgrid.py and compare the
media files using diff.

Config: small region, no rotation, x=40,y=20,z=10 ,vertex center, 8000 pts, 1km spacing

The dumpgrid.py files are compared and there are no differences.
The *.media files are compared and there are no differences.

Further Tests:
- Repeat the tests with each output format
- Repeat the test with rotation
- Repeat the test with large region


-bash-4.2$ pwd
/home/scec-00/maechlin/ucvmc_mpi_testing/testcase1


-bash-4.2$ more ucvm2mesh.conf
# List of CVMs to query
ucvmlist=cvmh

# UCVM conf file
ucvmconf=/auto/scec-00/maechlin/ucvmc188/conf/ucvm.conf

# Gridding cell centered or vertex
gridtype=CENTER

# Spacing of cells (m)
spacing=100.0

# Projection
proj=+proj=utm +datum=WGS84 +zone=11
rot=0.0
x0=-118.0
y0=34.0
z0=0.0

# Number of cells along each dim
```

```
nx=400
ny=200
nz=100

# Partitioning of grid among processors
px=4
py=2
pz=1

# Vs/Vp minimum
vp_min=0
vs_min=0

# Mesh and grid files, format
meshfile=/auto/scec-00/maechlin/ucvmc_mpi_testing/testcase1/mesh_cvmsh_ijk12_1000m_mpi.media
gridfile=/auto/scec-00/maechlin/ucvmc_mpi_testing/testcase1/mesh_cvmsh_ijk12_1000m_mpi.grid
meshtype=IJK-12

# Location of scratch dir
scratch=/staging/scratch
```

Note the ucvm2mesh.conf file references a ucvm.conf file, which contains configuration settings (GTL, background model) for some models including CVM-H.

```
-bash-4.2$ cat /home/scec-00/maechlin/ucvmc188/conf/ucvm.conf
#
# UCVMC Config file
# Updated Jan 2017
#
# UCVM install path. Updated by install script to point to
# installation directory
#
ucvm_install_path=/home/scec-00/maechlin/ucvmc188
#
# Path to statewide topgraphy and Vs30 etree
#
ucvm_interface=map_etree
ucvm_mappath=/home/scec-00/maechlin/ucvmc188/model/ucvm/ucvm.e
#
# Standard California Velocity Models Registered into UCVM
#
# SCEC CVM-H v15.1 (aka CVM-H)
cvmh_modelpath=/home/scec-00/maechlin/ucvmc188/model/cvmh1510/model
#
# SCEC CVM-S4 (aka CVM-S)
cvms_modelpath=/home/scec-00/maechlin/ucvmc188/model/cvms/src
#
# SCEC CVM-S4.26.M01 (aka CVM-SI)
cvmsi_modelpath=/home/scec-00/maechlin/ucvmc188/model/cvms426/model/i26
#
# USGS Bay Area high-rez and extended etrees v08.3.0 (aka Cencal)
cencal_modelpath=/home/scec-00/maechlin/ucvmc188/model/cencal/model/USGSBayAreaVM-08.3.0.etree
cencal_extmodelpath=/home/scec-00/maechlin/ucvmc188/model/cencal/model/USGSBayAreaVMExt-08.3.0.e
tree
#
# UCVM defines two different 1D model
# 1d = modified (higher resolution, more layers) hadley-kanamori model
# bbp1d = 1D Los Angeles Basin model used in SCEC Broadband Platform
```

```
1d_modelpath=/home/scec-00/maechlin/ucvmc188/model/1d/1d.conf
bbp1d_modelpath=/home/scec-00/maechlin/ucvmc188/model/1d/bbp1d.conf
#
# The entries above are updated by the install script.
# The entries below are manually updated by the user, as needed.
#
# CVM-H model flags. The model defaults are:
# No 1D background model
# No GTL added
#
cvmh_param=USE_1D_BKG,False
cvmh_param=USE_GTL,False
#
```

Directions to the job scheduler (USC slurm system):

```
-bash-4.2$ more makemesh.slurm
#!/bin/bash
#SBATCH -p scec
#SBATCH --ntasks=8 #number of tasks with one per processor
#SBATCH --nodes=8 # Nodes requested
#SBATCH --mem 0 # Set to unlimited memory
#SBATCH --time=00:30:00
#SBATCH -o ucvm2mesh_mpi_%A.out
#SBATCH -e ucvm2mesh_mpi_%A.err
#SBATCH --export=NONE
#SBATCH --mail-user=maechlin@usc.edu
#SBATCH --mail-type=END

cd /auto/scec-00/maechlin/ucvmc_mpi_testing/testcase1
srun -v --mpi=pmi2 /auto/scec-00/maechlin/ucvmc/bin/ucvm2mesh-mpi -f
/auto/scec-00/maechlin/ucvmc_mpi_testing/testcase1/ucvm2mesh.conf
```

And this is how to submit the job to USC HPC cluster job scheduler:

-- Submit the job using Slurm Commands (sbatch)

This command will request the computer to run your job. Job scheduler are commonly used on shared computer resources, where many people are trying to use the computer. Run the sbatch command to submit your job to run on the computer nodes of the computer.

```
$ sbatch makemesh.slurm
```

Then check on the status of the job with:

```
$ squeue -u maechlin # check on status of job in queue
$ sinfo
```

A serial version can be run this way. Move to a working directory, called ucvm_user.

```
$ cd /home/scec/ucvm_work
```

Create or copy an ucvm2mesh.conf for the serial computer. Elements of the file the refer to MPI programs are ignored when invoking the serial version of the meshing tool.

```
$ more ucvm2mesh.conf
# List of CVMs to query
ucvmlist=cvmh

# UCVM conf file
ucvmconf=/home/ucvm_user/ucvm_bin/conf/ucvm.conf

# Gridding cell centered or vertex
gridtype=CENTER

# Spacing of cells (m)
spacing=100.0

# Projection
proj=+proj=utm +datum=WGS84 +zone=11
rot=0.0
x0=-118.0
y0=34.0
z0=0.0

# Number of cells along each dim
nx=400
ny=200
nz=100

# Partitioning of grid among processors
px=4
py=2
pz=1

# Vs/Vp minimum
vp_min=0
vs_min=0

# Mesh and grid files, format
meshfile=/home/ucvm_user/ucvm_user/mesh_cvmh_ijk12_100m_serial.media
gridfile=/home/ucvm_user/ucvm_user/mesh_cvmh_ijk12_100m_serial.grid
meshtype=IJK-12

# Location of scratch dir
scratch=/staging/scratch
```

```
$ pwd
/home/ucvm_user/ucvm_user

$ ls
ucvm2mesh.conf
$ /home/ucvm_user/ucvm_bin/bin/ucvm2mesh -f ./ucvm2mesh.conf
```

2.4.3. Expected Runtime

ucvm2mesh_mpi – 8 Million Pts, CVM-H model, 8 processors, 2 minutes (100m spacing)

ucvm2mesh – 8 Million Pts, CVM-H model, 8 processors, 2 minutes (100m spacing)

2.4.4. Simulation Output Results

Produces a grid file (ascii lat/lon/depth points), and Mesh file (i/j/k, and model properties)

```
------------------------------------------
Begin SLURM Prolog Mon 03 Sep 2018 08:19:21 PM PDT
Job ID:       1453023
Username:     maechlin
Accountname:  lc_pjm
Name:         makemesh.slurm
Partition:    scec
Nodes:        hpc[4191-4198]
TasksPerNode: 1(x8)
CPUSPerTask:  Default[1]
TMPDIR:       /tmp/1453023.scec
SCRATCHDIR:   /staging/scratch/1453023
Cluster:      uschpc
HSDA Account: false
Note:         Clearing SLURM_EXPORT_ENV=NONE before each job-step
End SLURM Prolog
------------------------------------------
[0] /auto/scec-00/maechlin/ucvmc/bin/ucvm2mesh-mpi Version: 17.1.0
[0] Running on 8 cores
[0] Using config file /auto/scec-00/maechlin/ucvmc_mpi_testing/testcase1/ucvm2mesh.conf
[0] Configuration:
        [0] UCVM Model List: cvmh
        [0] UCVM Conf file: /auto/scec-00/maechlin/ucvmc188/conf/ucvm.conf
        [0] Gridtype: 0
        [0] Spacing: 100.000000
        [0] Projection: +proj=utm +datum=WGS84 +zone=11
                [0] Rotation Angle: 0.000000
                [0] Origin x0,y0,z0: -118.000000, 34.000000, 0.000000
                [0] Dimensions: 400, 200, 100
        [0] Proc Dimensions: 4, 2, 1
        [0] Vp Min: 0.000000, Vs Min: 0.000000
        [0] Mesh File:
/auto/scec-00/maechlin/ucvmc_mpi_testing/testcase1/mesh_cvmsh_ijk12_1000m_mpi.media
        [0] Grid File:
/auto/scec-00/maechlin/ucvmc_mpi_testing/testcase1/mesh_cvmsh_ijk12_1000m_mpi.grid
        [0] Mesh Type: 1
        [0] Scratch Dir: /staging/scratch
[0] Initialization complete
[0] Converting grid to latlong
[0] Grid generation complete
[0] Configuring UCVM
[1] Allocating 10000 grid points
[1] Partition dimensions: 100 x 100 x 100
[1] I,J,K start: 100, 0, 0
[1] I,J,K end: 200, 100, 100

...

[0] Wrote slice 99 (10000 pnts) in 6.92 ms, 1444460.500000 pps
[0] Extracted 1000000 points
```

```
[5] Extracted 1000000 points
[1] Extracted 1000000 points
[7] Extracted 1000000 points
[6] Extracted 1000000 points
[3] Extracted 1000000 points
[4] Extracted 1000000 points
[2] Extracted 1000000 points
[0] Max Vp: 6392.830078 at
[0]         i,j,k : 79, 41, 99
[0] Max Vs: 3860.201172 at
[0]         i,j,k : 33, 115, 62
[0] Max Rho: 2806.543213 at
[0]         i,j,k : 79, 41, 99
[0] Min Vp: 1327.084961 at
[0]         i,j,k : 9, 151, 0
[0] Min Vs: 120.689651 at
[0]         i,j,k : 89, 131, 0
[0] Min Rho: 1517.037964 at
[0]         i,j,k : 9, 151, 0
[0] Min Ratio: 1.628962 at
[0]         i,j,k : 0, 0, 61
```

The output files are in the directory specified in the ucvm2mesh.conf file:

```
-bash-4.2$ pwd
/home/scec-00/maechlin/ucvmc_mpi_testing/testcase1
-bash-4.2$ ls -tlh *.grid *.media
-rw-r--r-- 1 maechlin scec  92M Sep  3 20:19 mesh_cvmsh_ijk12_1000m_mpi.media
-rw-r--r-- 1 maechlin scec 1.9M Sep  3 20:19 mesh_cvmsh_ijk12_1000m_mpi.grid
```

2.4.5. Use Case Variations

- Different velocity model (or models)
- Different region
- Different Min Vs
- Different GTL setting

2.5 Common UCVM Consideration and Issues

1. What is the UCVM abbreviation for the CVM I want to use?
2. Was UCVM Installed with/without MPI ?
3. What is the setting of GTL for selected CVM models ?
4. Which models support optional Ely-Jordan GTL ?
5. What mesh output formats are supported as outputs?
6. What property types can be specified in plots (vp, vs, rho, poisson)
7. What are the color scale options (and defaults)

3. SCEC Broadband Platform (BBP)

The SCEC Broadband Platform is a software system that can generate 0-100 Hz seismograms for historical and scenario earthquakes in California, Eastern North America, and Japan using several alternative computational methods.

3.1. Initial Computing Environment

1. A Linux computer (CentOS 7.5) with BBP dev branch I
2. Environment Variables set including:
   a. BBP_DIR
   b. BBP_VAL_DIR
   c. BBP_GF_DIR
   d. BBP_DATA_DIR
   e. PYTHONPATH

3.2. Science Use Case

Run BBP Platform to calculate seismograms at 3 station locations in Southern California using the Chino Hills Earthquake.

3.3. Use Specified Simulation Input Variables

Validation event to use: Chino Hills

Simulation method to use: GP

Custom source description (customize SRC file to see how it changes GoF)

Custom station list (trim station file to use only three stations)

3.4. Expected Runtime

Runtime varies by simulation method, number of stations used, and event magnitude. For the GP method with 3 stations the run time should be 3-5 minutes.

3.5. Simulation Output Results

Review a number of BBP output files, including station map, slip distribution, velocity and acceleration timeseries, comparison plots showing data versus simulated seismograms, GoF plot, map GoF plot.

3.6. Usage Notes

Below is the output of the BBP showing how to set up the simulation scenario above.

```
$ run_bbp.py --expert
Welcome to the SCEC Broadband Platform version 17.3.0.
================================================================================

Please select the Broadband Platform mode of operation:
  * Validation - Simulates a historical event
  * Scenario   - Runs a user-defined hypothetical event

Do you want to perform a validation simulation (y/n)? y
================================================================================
```

```
Please select a validation event from the list below:

(1) Chino Hills
(2) LOMAP
(3) NR
(4) North Palm Springs
(5) Whittier
? 1
================================================================================

The Broadband Platform includes several scientific methods that can be used to calculate synthetic
seismograms.

Choose a Method to use in this Broadband validation simulation:
(1) GP (Graves & Pitarka)
(2) UCSB
(3) SDSU
(4) EXSIM
(5) CSM (Composite Source Model) - Beta Version
(6) Song
(7) Irikura Recipe Method 1 (Irikura1)
? 1
================================================================================

Each validation package includes a default source description (SRC) file for a historical event. Would you like to
provide a different file instead of the default file provided? Answer 'no' here if you would like to use the
standard source file for this event.

Do you want to provide a custom source file (y/n)? y
Do you want to
  (1) select a source description in /Users/fsilva/Work/bbp-scec/bbp_sims/run
      OR
  (2) enter the path of a source description file
? 1
Here are the source description files in the run directory.
Please select one:

(1) chino_hills_2008.src
(2) chino_hills_scec.src
(3) whittier_narrows_1987.src
(4) whittier_narrows_scec.src
? 2
================================================================================

When starting a simulation from a source description (SRC) file, the Broadband Platform workflow should
include a rupture generator. Answer 'yes' here unless providing a complex Standard Rupture Format (SRF) file.

Do you want to run the rupture generator (y/n)? y
================================================================================
```

```
Station Selection
=================
Would you like to:
  (1) generate seismograms for all stations in the validation package
    OR
  (2) provide a custom list with a subset of the stations
? 2
Do you want to
  (1) select a BBP station list in /Users/fsilva/Work/bbp-scec/bbp_sims/run
    OR
  (2) enter the path of a BBP station list file
? 1
Here are the BBP station list files in the run directory.
Please select one:

(1) ch_v14_2_2.stl
(2) chino_hills-3-stations.stl
(3) whittier-3-stations.stl
? 2
================================================================================
STL file: /Users/fsilva/Work/bbp-scec/bbp_sims/run/chino_hills-3-stations.stl
================================================================================

Site Response
=============
Running a site response module is an optional step while running a Broadband Platform simulation. It requires a
station list file containing the Vs30 values for each station location.

Do you want to run the site response module (y/n)? y
================================================================================
Do you want to generate velocity seismograms' plots (y/n)? y
================================================================================
Do you want to generate acceleration seismograms' plots (y/n)? y
================================================================================

The Broadband Platform can generate comparison plots of the validation data against GMPEs to show how
GMPEs match the recorded data for a certain event.

Do you want to generate a GMPE comparison plot (y/n)? n
================================================================================

Goodness-of-Fit Plot
====================
Running a goodness-of-fit (GoF) module is an optional  step while running a Broadband Platform simulation. It
creates a comparison plot showing how well the  calculated seismograms fit recorded data.

Do you want to run a goodness-of-fit module (y/n)? y
================================================================================
```

Users can optionally select a Goodness of Fit module to plot a comparison of how well the simulated seismograms match the recorded data in a historical event.

Choose a Goodness of Fit (GOF) Module:
(1) GP
(2) SDSU
(3) Both
? 1
==========

Additional Metrics
==========
Calculating additional metrics is an optional step on the Broadband Platform. It creates additional plots and data files that can be used to study the simulation.

Do you want to calculate additional metrics (y/n)? n

4. SCEC NHERI DesignSafe-CI Ground Motion Data Portal

4.0 Summary

The web-based SCEC NHERI DesignSafe-CI Ground Motion Portal is a specialized data access tool that provides civil engineers and other researchers with access to Simulated Ground Motions for GMSV project through the NHERI DesignSafe-CI Portal.

The NHERI Design Safe Portal is accessible at: https://www.designsafe-ci.org/
The critical step in accessing the portal is in obtaining access to the TACC system. You can request an account from the design-safe portal. The account approval is not automatic, and may take a couple of days.

4.1 Use Case

The tool provides access to simulated ground-motion seismograms via a query based on user-input magnitude, distance, and site class. The user inputs a tolerance range for these search parameters by providing minimum and maximum values for each. Optionally, the user may also constrain the intensity measure by selecting the spectral-intensity measure and specifying a tolerance range for this measure. The user selects the IM from pull-down menus of options by selecting the spectral-acceleration component (H1,H2,V, RotD50, RotD100) as well as the period at which this IM is measured (period range: 0.01-10 seconds).

4.2 How to use the SCEC BBP Ground-Motion Portal

1. Access Ground-Motion Portal
    a. https://www.designsafe-ci.org/
    b. Set up user account in DesignSafe CI/TACC.

    c. Log into DesignSafe
    d. Access SCEC Jupyter Notebook on DesignSafe CI
2. Perform Search:
    a. Select Simulation Dataset as database
    b. Submit Search Criteria
        i. Magnitude
        ii. Distance
        iii. Site Class: Vs30
        iv. Intensity Measure (Optional)
    c. Visualize & Evaluate Search Results
    d. Spectra Tables & Plots
    e. Metadata Tables
    f. Individual-Record Spectra Plots
    g. Individual-Record Seismogram Plots
3. Access Data
    a. Download Metadata + Seismogram Files
    b. Reference File Location in Design-Safe (Table)
    c. Work with data directly in Jupyter Notebook

5. SCEC OpenSHA Seismic Hazard Analysis Tool

Initial Laptop Computing Environment

- Java 8 or higher
- 8 GB+ of RAM
- Active internet connection, no firewall blocking access to opensha.usc.edu port 8080

Summary

Several apps to compute deterministic ground motions, probabilistic seismic hazard curves and uniform hazard spectra for one or more location, site parameters, earthquake rupture forecast, ground motion models (GMMs, aka ground motion prediction equations GMPEs). Allows disaggregation of hazard.

App list (* covered today):

Attenuation Relationship Plotter*
Hazard Curve Application*
Hazard Spectrum Application
Site Data Application*
Scenario ShakeMap Application

We will perform all the demo with PGA as the intensity measure

5.1 OpenSHA Software Demo 1: Attenuation Relationship Plotter

5.1.1. Science Use Case

You want to understand the impact of a basic parameter on ground motions, let's do that example using Vs30 Vs distance.

5.1.2. User specified simulation input variables

Select Boore and Atkinson (2008) as the GMM

Plot Choices options default, except for PGA

1. Adjustable parameters, select Vs30 =300 m/s, leave the rest as default, click add curve
2. Adjustable parameters, select Vs30 =1,000 m/s, leave the rest as default, click add curve

Change the x and y axes to log and linear to better assess differences.

5.1.3 Expected runtime

Seconds

5.1.4. Simulation output results

In "Show plot" view, click on the save PDF.
In "Show data" view, click on the save PDF.
What's the difference?

5.1.5. Use case variations

Redo the same test using a different GMM and compare between GMMs.

5.2 OpenSHA Software Demo 1: Hazard Curve Application

5.2.1. Science Use Case

Compute hazard curves for one or more location, site parameters, earthquake rupture forecast, GMPE.

5.2.2. User specified simulation input variables

Select Boore and Atkinson (2008) as the GMM. Use the default site lat/lon, but set Vs30 = 300 m/s. Leave the IMT as 1 second SA.

In the "ERF & Time Span" tab set the ERF to "WGCEP (2007) UCERF2 - Single Branch" as the earthquake rupture forecast (ERF). Set the duration to 50 years and the start year to 2018.

Under "Control Panel" select "Disaggregation" and select "Probability". Set the "Disaggregation Prob" to 0.02 which represents 2% in this 50 year forecast.

Hit compute.

5.2.3. Expected runtime

1 minute

5.2.4. Simulation output results
Numerical values in text format and data plots, specifics depend on app.

5.2.5. Use case variations
Change disaggregation probability to 0.5 (50%)

5.4 OpenSHA Software Demo 4: Site Data Application

5.4.1. Science Use Case
Query various available models for Vs30 and basin depth data for a single site, make a map, and batch download data.

5.4.2. User specified simulation input variables
Models of interest: disable all models except the following:

- CGS/Wills VS30 Map (2015)
- CGS/Wills Site Classification Map (2006)
- Global Vs30 from Topographic Slope (Wald & Allen 2008)
- SCEC Community Velocity Model Version 4, Iteration 26, Basin Depth (Depth to Vs = 2.5 and 1.0 km/sec)
- SCEC Community Velocity Model Version 4 Basin Depth (Depth to Vs = 2.5 and 1.0 km/sec)
- SCEC/Harvard Community Velocity Model Version 11.9.x Basin Depth (Depth to Vs = 2.5 and 1.0 km/sec)

Hit “View All Available Data”

Create and save a CSV file with multiple sites:

```
Latitude,Longitude,Name
35,-118,"Site 1"
34,-118,"Site 2"
35,-119,"Site 3"
```

Hit “Batch Download” and import this CSV file. Hit “Fetch All Data” then save as a new CSV File.

Switch to the “Site Data Maps” tab. Select “CGS/Wills VS30 Map (2015)” and hit “Set Region from Data”. Hit “Create map”

5.4.3. Expected runtime
0.1-3m per map depending on map grid spacing

5.4.4. Simulation output results

Site data values (text window), map, CSV file

5.4.5. Use case variations
Try different models, experiment with model order

6. SCEC Project Specific Data Management — The UGMS MCER Case

6.1. Laptop – UGMS MCER Web-based Data Access Tool Demo

URL: https://data2.scec.org/ugms-mcerGM-tool_v18.4/

6.1.1 Expected Laptop Computing Environment

- An internet connection
- Up-to-date web browser (Firefox, Safari, Chrome, etc.)
- JavaScript must be enabled

6.1.2. Science Use Case
Given a location and site-class or Vs30 value, determine the site-specific MCER response spectrum and site-specific design response spectrum for use by engineers in building design.

6.1.2. User specified Input variables

- Site Class or Vs30 (or unknown such that Vs30 will be estimated from Wills et al., 2015)
- Enter Latitude and Longitude
- Or choose position on map to auto-input latitude and longitude

6.1.3. Expected Runtime
5-7 seconds

6.1.4. Output Results
The site-specific MCER and Design Earthquake (=2/3 MCER) response spectrum and related data points in web page and PDF formats.

6.1.5. Use Case Variations

- Query with different location
- Query with different site class
- Query with user-specified Vs30

- Query with unknown site class (Vs30 estimated from Wills et al., 2015)

7. The SCEC Software Clinic

The software clinic will provide an opportunity to work with the SCEC software discussed, and to try the demonstration on your own computer, or on a limited number of computer provided at the workshop by SCEC.

During the software clinic, SCEC software developers will be available to answer questions and to support participants who would like to work with the SCEC software.

7.1. Workshop Access to SCEC Software

There are a number of ways that users can run the SCEC software during the clinic. We will try to support a variety of user computer types, and operating system types.

The OpenSHA software can be run on any computer with recent Java (Version 8 or higher), which is available for Linux, Mac, and Windows. Users that would like to run the OpenSHA software on their own computer, can download OpenSHA jar files and run the OpenSHA application on their computers.

Two of the software packages, UCVM and Broadband, are designed to run on Linux computers. SCEC will provide VirtualBox images of these software packages that workshop participants can run in a VirtualBox environment on their own computers. VirtualBox is a free software distribution (from Oracle) that must be installed before the software can be run.

The two SCEC data access products, the DesignSafe BBP Data Access system, and the MCER data access system are web-based, so participants can use any computer with a up-to-date web browser.

The DesignSafe BBP Data Access system, however, requires a TACC system login. Users must apply for this account before the workshop if they wish the run this software on their own computers. We will provide a web-browser equipped computer during the clinic, that has access to DesignSafe, if user want to try out the software without their own TACC account during the workshop.

7.2 Use of Virtual Box

SCEC provides a large (~15GB) VirtualBox "Appliance" (this is VirtualBox's term for an image file) that contains a CentOS system, and the three SCEC software application codes including OpenSHA, Broadband Platform, and UCVM.

This SCEC_Workshop.ova appliance file contains minimal installation of these three codes, in order to save space. For this reason, not all supported SCEC velocity models, or Broadband platform simulation regions, or OpenSHA applications are included. Additional capabilities are available if you do a complete installation.

It will take 20 minutes to transfer this file from a USB drive, and another 20 minutes to import this appliance into VirtualBox on a update-to-date Mac laptop.

As configured, the SCEC_Workshop.ova image does not share disk space with the host computer. For this reason, it is difficult to get files in and out of the VirtualBox. As a work-around, we use command line sftp commands to transfer files between computers.