Using CVM-H for Code Validation Benchmarks

December 2, 2015

This is a quick-start guide for installing and running the SCEC Community Velocity Model CVM-H. It provides just the minimum information you need so that you can use CVM-H for code validation benchmarks.

If you want to know more about CVM-H, refer to the CVM-H User Guide.

Contents

Part 1: Installing CVM-H

CVM-H must be installed on a computer running Linux. Note that CVM-H will not run on a Mac, so you need to use Linux. The installation is done from a Linux shell prompt.

Step 1: Create Directories

Create two directories on your Linux computer:

- The directory where CVM-H will be installed. We will call this the “install directory.” It can be anywhere you like.

- The directory where CVM-H will be downloaded. We will call this the “download directory.” It can be anywhere you like. After you download, unpack, configure, build, test, and install the software, you can delete this directory.

Step 2: Download the Software

Change to the download directory. Then run the following two commands:

```
wget  http://scecdata.usc.edu/cvws/download/cvmh/cvmh-15.1.0.tar.gz

wget  http://scecdata.usc.edu/cvws/download/cvmh/cvmh-15.1.0.md5
```

The first command downloads a 530 MB file, so it might take a while, depending on the speed of your Internet connection. The second command downloads a small file.

After both files are downloaded, run the following command:

```
md5sum  -c  cvmh-15.1.0.md5
```

You should see the following message:

```
cvmh-15.1.0.tar.gz:  OK
```

The “OK” means that the files were downloaded without error. If you get “FAILED” instead, it means that the downloaded files are corrupted, in which case you should start over and download the files again.

Step 3: Unpack and Configure the Software

To unpack the software, run the following command:

```
tar  zxvf  cvmh-15.1.0.tar.gz
```

This command creates a directory called cvmh-15.1.0, which contains many files and subdirectories. Next, change to the directory you just created, with the following command:

```
cd  cvmh-15.1.0
```

Now you configure the software by giving a command like the following, where “${INSTALL_DIR}” is the installation directory:

```
./configure  --prefix=${INSTALL_DIR}
```

For example, if your installation directory is /usr/share/cvmh, then you would configure the software with this command:

```
./configure  --prefix=/usr/share/cvmh
```

Notice that there are two dashes before the word “prefix.” You should insert the actual name of your installation directory after the equal sign.

Note: At this point, you may see an error message similar to this:

```
configure: error: cannot find install-sh, install.sh, or shtool in ./aux-config
```

If you see this error message, then refer to Appendix 1 to find out what to do.

Step 4: Build and Install the Software

Once the configure operation completes without errors, you are ready to build the software. Give this command:

```
make
```

It should take about a minute to build the software. When it’s done, test the software using the following command:

```
make  check
```

It will take a few minutes to execute a series of tests. You should see the word “PASS” displayed after each test. Finally, you install the software with this command:

```
make  install
```

At this point, CVM-H is ready to be used.

Part 2: Running CVM-H

Running CVM-H is pretty simple: You give it a list of points, and it responds with V_P, V_S, and density at each point. The program that does this is called vx_lite. Points are specified with (X, Y, Z) coordinates as follows:

X = Easting, in UTM coordinates, in meters.
Y = Northing, in UTM coordinates, in meters.
Z = Depth in meters, positive underground.

The UTM (Universal Transverse Mercator) coordinate system used in CVM-H is UTM zone 11, datum NAD 1927, geoid Clarke 1866.

How to Run vx_lite

Put your list of points into an ASCII text file, one point on each line. The file has three columns as follows:

Column 1 = Easting X, in meters.
Column 2 = Northing Y, in meters.
Column 3 = Depth Z, in meters (positive underground).

Then, run the vx_lite program with the following command:

```
${INSTALL_DIR}/bin/vx_lite  -s  -z  dep  -m  ${INSTALL _DIR}/model  < infile  > outfile
```

where “${INSTALL_DIR}” is the installation directory. For example, if you installed CVM-H into /usr/share/cvmh, then the command would be:

```
/usr/local/cvmh/bin/vx_lite  -s  -z  dep  -m  /usr/local/cvmh/model  < infile  > outfile
```

In this command, “infile” is the input file that contains the (X, Y, Z) coordinates, and” outfile” is the output file that contains the velocities and densities. The output file is an ASCII text file with 19 columns. Each line in the output file has data for one point. For our purposes, we only care about the last three columns of the output file, which are:

Column 17: Compressional wave velocity V_P in m/s.
Column 18: Shear wave velocity V_S in m/s.
Column 19: Density ρ in kg/m^3.

If vx_lite cannot supply data for a point (for example, if the point is in the air), then the returned velocities and density are -99999.00. The program vx_lite should be able to supply data for any point with $Z > 0$. If the point is located outside the Southern California region covered by the 3D velocity model, then vx_lite uses a simple 1D velocity model.

Example Input and Output

Here is a sample input file. This file contains 8 points, which are located on the Imperial Fault, running from north to south, at depths ranging from 1 km to 15 km. In this example the coordinates are given in the form of integers, but it is OK to use floating-point numbers too.

```
635485  3643429  1000
639189  3638231  3000
642892  3633033  5000
646595  3627835  7000
650298  3622638  9000
654001  3617440  11000
657704  3612242  13000
661407  3607044  15000
```

With this input file, the program vx_lite produces the following output file. Here we show only the first three columns and the last three columns, omitting 13 columns in the middle.

```
635485.000000  3643429.000000   1000.00 ... 1925.55   487.55  1871.05
639189.000000  3638231.000000   3000.00 ... 3671.47  1992.65  2345.48
642892.000000  3633033.000000   5000.00 ... 5264.05  3004.80  2577.06
646595.000000  3627835.000000   7000.00 ... 5467.91  3273.81  2612.28
650298.000000  3622638.000000   9000.00 ... 6199.23  3874.95  2760.82
654001.000000  3617440.000000  11000.00 ... 6646.03  4154.92  2870.55
657704.000000  3612242.000000  13000.00 ... 6645.26  3948.52  2870.35
661407.000000  3607044.000000  15000.00 ... 6666.79  3927.69  2876.02
```

As described above, the last three columns are V_P, V_S, and density respectively. In general the velocity and density increase with depth, although as this example shows, there are exceptions.

Appendix 1: What To Do If Configure Fails

As mentioned above, when you configure the software, you may see an error message like this:

```
configure: error: cannot find install-sh, install.sh, or shtool in ./aux-config
```

During the configure operation, CVM-H looks for a program called Automake, which is normally available on Linux systems. The error message means that CVM-H was not able to find Automake. There are two possible reasons why this might occur:

- Automake is not installed on your system, or you have a version of Automake that is too old.
- Automake is installed, but its files are not in the expected location.

Step 1: Test if Automake is Installed

Enter the following command:

```
automake  --version
```

Notice that there are two dashes before the word "version." If Automake is installed, then its version number is displayed. If Automake is not installed then you will see an error message like "command not found."

CVM-H requires Automake version 1.11 or higher. If Automake is installed, and its version is 1.11 or higher, then skip ahead to step 2 below.

If Automake is not installed, or if its version is earlier than 1.11, then you will need to install Automake on your Linux computer. To do this, you will need to have administrative privileges. (If you don't have administrative privileges, then ask your system administrator to install Automake for you.)

First, you need to install a program called Autoconf, if it is not already installed on your system. Obtain Autoconf from here:

http://www.gnu.org/software/autoconf/autoconf.html

You should install the latest version of Autoconf, which is 2.69 as of this writing.

Then, install Automake, which you can obtain from here:

http://www.gnu.org/software/automake/automake.html

We recommend installing Automake version 1.11.6. More recent versions of Automake might also work, but we have not tested them.

Verify your installations of Autoconf and Automake by entering the following two commands, both of which should display the version numbers of the installed software:

```
autoconf  --version
automake  --version
```

After installing Automake, try again to configure CVM-H.

Step 2: Find Where the Automake Files are Located

If you have verified that Automake is installed on your computer, and its version number is 1.11 or higher, but the configure operation is still failing, then the Automake files are probably not in the expected location. CVM-H expects that the Automake files are located in the directory /usr/share/automake-1.11. In particular, CVM-H looks for the following three files:

```
/usr/share/automake-1.11/depcomp
/usr/share/automake-1.11/install-sh
/usr/share/automake-1.11/missing
```

Check if these files are present on your computer. If not, then search for a directory whose name begins with “automake” and contains the three files “depcomp,” “install-sh,” and “missing.”

The directory will probably be in some system location like /usr/share, or /usr/local, or /usr/local/share.

Once you find where the Automake files are located, you have to tell CVM-H where they are. First, change to the directory “aux-config” which is part of the CVM-H distribution:

```
cd  aux-config
```

Next, use three commands similar to the following to tell CVM-H where the three files are located. In this example, we assume that the files are located in a directory called /usr/local/share/automake-1.11. When you enter these three commands, replace /usr/local/share/automake-1.11 with the actual directory on your system where the files are located:

```
ln  -sf  /usr/local/share/automake-1.11/depcomp
ln  -sf  /usr/local/share/automake-1.11/install-sh
ln  -sf  /usr/local/share/automake-1.11/missing
```

Finally, leave the “aux-config” directory with this command:

```
cd  ..
```

Now try again to configure CVM-H.