The Unified Community Velocity Model Software Framework

Patrick Small[a,1,*], David Gil[b,**], Phil J. Maechling[b], Ricardo Taborda[c,**], Thomas H. Jordan[b,d], Geoffrey P. Ely[e]

[a]*Department of Computer Science, University of Southern California, Los Angeles, CA 90089, USA*
[b]*Southern California Earthquake Center, 3651 Trousdale Parkway, Suite 169, Los Angeles, CA 90089, USA*
[c]*Center for Earthquake Research and Information, and Department of Civil Engineering, University of Memphis, Memphis, TN 38152, USA*
[d]*Department of Earth Sciences, University of Southern California, Los Angeles, CA 90089, USA*
[e]*Leadership Computing Facility, Argonne National Laboratory, Argonne, IL 60439, USA*

Abstract

This paper presents the Unified Community Velocity Model (UCVM) software framework developed at the Southern California Earthquake Center. UCVM is a collection of software tools and application programming interfaces designed to provide standard access to multiple seismic velocity models used in seismology and geophysics research. Seismic velocity models are key components of current research efforts dedicated to advancing our knowledge of the Earth's crustal structure and its influence on the ground response during earthquakes, including both the regional deep geology and local effects due to the geometry, spatial distribution, and material composition of sediments in basins and valleys. UCVM provides a framework to facilitate research activities that involve the use of seismic velocity models and has particular direct applications in physics-based earthquake ground-motion simulation and seismic hazard analysis in large regions prone to earthquakes, such as Southern California. We describe the background that led to the development of the UCVM and its various software components, including advanced high-performance computer tools available within UCVM, its performance, and product deliverables. We also present examples of recent applications that use UCVM tools or output datasets.

Keywords:
seismic velocity models, earthquake simulation, high-performance computing, meshing

1. Introduction

The quantitative understanding of the physical world is an essential goal of geoscience research. We use mathematical abstractions to represent the behavior of systems in static and dynamic states; and properties such as density and elastic moduli to characterize the response of materials to absorbe or transmit forces. In seismology and geophysics, our understanding of earthquakes and their impacts, depends in part on our knowledge and representation of the geometry and properties of seismically-active faults and of the Earth's crust itself. On the one hand, we use fault and stress models to describe the geometry and conditions that facilitate the initiation and dominate the propagation of an earthquake's rupture process; and, on the other hand, we use seismic velocity and attenuation models to describe how the geologic structure of the crust, and the mechanical properties of the materials in geologic structures, basins and sedimentary deposits controls the characteristics of seismic waves and the resulting ground motion generated by earthquakes.

We are interested in how seismic velocity models are managed by and distributed to scientists, with emphasis on simulation applications. Source models and seismic velocity models are the basic inputs to physics-based earthquake simulation. Physics-based earthquake simulation uses numerical methods and models that incorporate, explicitly, the solution of wave propagation problems through crustal structures, with specific applications in the assessment of the expected levels of ground motion in a region prone to earthquakes for use in seismic hazard analysis. During the last two decades, various seismic velocity models have been made available to the geophysics and seismology research community (e.g., Brocher, 2008) (add more refs.). This gave way to the conception of community velocity models (CVMs). CVMs are seismic velocity models that have been developed, maintained, and advanced by a community of interested investigators who provide open-access to models and the information available therein (e.g., Magistrale et al., 2000) (add more refs).

CVMs have been typically distributed in the form of static datasets or computer programs that dynamically operate on those datasets to provide information about the geometry and material properties of the crust in a particular region. However, these datasets and computer programs have not always been thought carefully from a computational perspective. In addition, recent advances in earthquake simulations, powered by the increasing capability of supercomputers (e.g., add ref), have increased significantly the computational demand placed

---

[*]Principal corresponding author
[**]Corresponding author
Email addresses: `patrices@usc.edu` (Patrick Small),
`davidgil@usc.edu` (David Gil), `ricardo.taborda@memphis.edu`
(Ricardo Taborda)
[1]This is only a tentative author line-up. Final line-up to be agreed upon by the whole group

on CVMs as input to these simulations.

This paper presents the Unified Community Velocity Model (UCVM), a software framework designed to provide standard and computationally efficient access to seismic velocity models. UCVM was developed at the Southern California Earthquake Center (SCEC). It is a collection of software tools and application programming interfaces (APIs) that facilitate the access to the material properties stored in CVMs, with specific applications in physics-based earthquake ground-motion simulation and regional seismic hazard assessments. Here, we describe the development of the UCVM and its various software components, including features for dedicated use in high-performance parallel computers, and present examples of recent applications of UCVM tools in earthquake research.

2. Community Velocity Models

This section will present how CVMs work and the various CVMs available to the community today. It will basically explain that CVMs provide the triplets of Vs, Vp and density, and, as an example, we can expand on a description of CVM-S and CVM-H, including their variations CVM-SI and CVM-H+GTL.

3. The UCVM Software Framework

3.1. Overview

The most significant functionality provided by UCVM is the ability to query a wide array of community velocity models for material properties in a standard way. Once a velocity model has been successfully integrated with UCVM, the framework can be utilized to query for the primary and shear seismic wave velocities (Vp, Vs) as well as the soil/rock density (ρ) at any geographic point within that model. The original model's local coordinate system and map projection are concealed behind a generic interface that instead allows queries by geographic latitude and longitude along with a vertical z-coordinate. The z-axis may be defined as either depth below the free surface or elevation relative to mean sea level.

To support this flexible query mechanism consistently across all models, UCVM includes a high-resolution digital elevation model (DEM). The DEM is synthesized from the USGS National Elevation Dataset (TODO: cite USGS NED) and the ETOPO1 Global Relief Model (TODO: cite ETOPO1). It currently spans the State of California along with portions of surrounding States.

The framework further extends the standardized interface by allowing multiple velocity models to be aggregated into a single composite model. Composition is accomplished by tiling two or more velocity models in three dimensions according to a user-specified priority ordering. Under this scheme, a query point is submitted sequentially to each velocity model within the ordered list that comprises the composite. The first model to return valid velocity data for the point is considered to have fulfilled the data request and subsequent models are not queried. Thus, overlap among the individual models is acceptable and
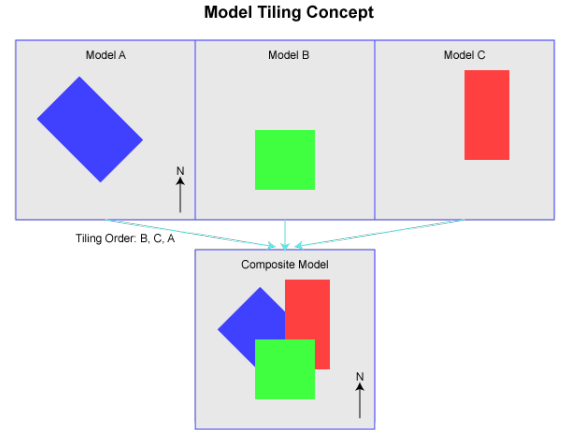

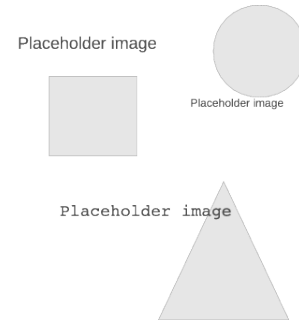Figure 1: Tiling of velocity models (TODO: Redo/cleanup this figure).

Figure 2: Interpolation of GTL and crustal model.

their relative priority ordering arbitrates which one satisfies any given query. Generally, no smoothing is performed at the interfaces between models (an exception is interpolation between a geotechnical layer and a crustal model as discussed later in this paper). This tiling concept is illustrated in Figure 1.

Community velocity models vary widely in their area of coverage, depth extent, and resolution. UCVM is flexible in its support for such variability. However, in order to better accommodate high frequency ground motion simulations, it categorizes models into two general groups: crustal models, and geotechnical layers (GTLs). Crustal models provide subsurface seismic wave velocities associated with basin, crust, and mantle structures. These models may potentially extend to many tens of kilometers below the Earth's surface yet do so at coarse resolutions (TODO: cite CVM-H, CVM-S). Geotechical layers, in contrast, provide velocities for only the near-surface (typically a few hundred meters) at very high resolution (TODO: cite Ely Vs30 GTL). Ground motion simulations, in particular, rely on high-resolution near-surface velocities and therefore a GTL serves to supplement the coarser data provided by crustal models.

When querying a set of velocity models that includes at least one GTL, the tiling mechanism described previously is en-

Table 1: Supported Community Velociy Models

Model	Versions	Reference
CVM-H	TBD	TBD
CVM-S	TBD	TBD
TBD	TBD	TBD

hanced to perform limited interpolation. The framework independently composes the GTLs and the crustal models into a composite GTL and composite crustal model, respectively, using the tiling technique. An interpolation zone is then defined along the z-axis between the two composites. Query points that fall within the zone are assigned an interpolated velocity based on the composite GTL velocity at the top of the zone boundary and the composite crustal model velocity at the bottom. Both the zone dimensions and interpolation function is selectable by the user. Figure 2 illustrates how this interpolation is performed. The primary motivation for performing interpolation in this particular case is to eliminate near-surface discontinuities between the GTLs and the crustal models which would adversely affect ground motion simulations.

Table 1 summarizes the list of community velocity models that have been integrated into UCVM. A single geotechnical layer is currently supported, based on Ely's Vs30-derived algorithm (TODO: cite Ely Vs30-derived GTL). This algorithm procedurally estimates Vp, Vs, and ρ from the Vs30 velocity at the point of interest. To that end, two separate Vs30 maps are provided for use with the Ely GTL, and each covers the same geographic region as the DEM (namely, California). One map is based on the the Wills and Wald dataset (TODO: cite Wills and Wald), and the other is based on the Yong dataset (TODO: cite Yong).

3.2. Command-line Utilities

The UCVM software framework provides UNIX command-line utilities to interactively query velocity models and construct regular three-dimensional meshes of material properties. These programs are described in the following sections.

3.2.1. Querying

The primary command-line tool for extracting material properties is *ucvm_query*. The typical use case is to invoke the program while specifying the path to the UCVM configuration file and a prioritized list of models to query. Both the configuration file and the pool of available velocity models are setup during the framework installation process. The configuration file allows UCVM to find the necessary resources during runtime, while the model list is a subset of installed models to be used in the current query session.

Once executed, the program loads the necessary models, tiles them in the same order as they were specified on the command-line, and awaits query coordinates from the user. The default format for coordinates is longitude (decimal), latitude (decimal), and depth (meters). Alternatively, elevation may be used for the z coordinate with a command-line option. For each submitted query point, UCVM returns a text line composed of sev-

enteen fields. The first three fields of the output echo the original query coordinates. The next two fields list the surface elevation from the DEM and the Vs30 value from the active Vs30 map. Bilinear interpolation is used to calculate smoothed values at the point of interest since both of these maps are gridded at 250m resolution.

The remaining twelve fields represent three sets of material properties. Each group of material properties is comprised of four fields: the source model whence the material properties were drawn, Vp, Vs, and ρ. As UCVM is able to perform limited interpolation between models, both the inputs to the interpolation and the smoothed result are shown for each query point. Thus, the first two parameter groups correspond to the crustal model and GTL that satisfied the query, respectively, while the third parameter group displays the interpolated results. If no GTL is defined, or the query point falls outside of the interpolation zone, the third parameter group simply repeats the results of whichever model satisfied the request. In any case, the framework's final verdict is given in this last parameter group.

As extraction of iso-surfaces based on shear wave velocity is a common and time-consuming application of community velocity models, UCVM offers a specialized query program for this purpose. The iso-surface utility *basin_query* is invoked in a similar manner to *ucvm_query*, yet it also allows the user to specify a Vs threshold for the iso-surface and a step size for successive query points along the z-axis. Query points are specified in longitude and latitude coordinates, and the program iteratively queries the tiled velocity models from the surface to a maximum depth using the provided step-size. Four fields are returned for each query point; the first two echo the original geographic coordinates while the remaining two fields report the minimum and maximum depths where the Vs threshold is crossed.

3.2.2. Meshing

Modern seismic wave propagation simulations often require a three-dimensional input mesh which describes the wave velocities and soil density for the entire simulation region. Constructing a mesh can be a time-consuming and error-prone prerequisite to executing a simulation. The UCVM framework assists in this respect by providing command-line utilities for generating three-dimensional meshes and Etree databases (TODO: cite Etree library) which may be used with the simulation codes AWP-ODC (TODO: cite APW-ODC) and Hercules (TODO: cite Hercules), respectively.

The AWP-ODC wave propagation software utilizes a mesh format consisting of a regular grid of cells at uniform resolution. Nominally, each cell contains the material properties Vp, Vs, and ρ, which are extracted directly from one or more community velocity models. The framework utility *ucvm2mesh* performs this extraction and produces an output mesh of the correct format. The program accepts a configuration file at runtime which describes the extent of the Earth's surface and crust to be gridded, the grid spacing, and the list of velocity models to be used for extraction. The mesh grid is defined by first applying a two-dimensional map projection whose origin is a geographic

coordinate, and whose axes are the projection axes (which may be optionally rotated). As UCVM employs the Proj4 software library (TODO: cite Proj4), virtually every standard map projection is supported. The number of elements along each axis and the grid spacing in projection units fully define the maximum extents of the mesh. This two-dimensional grid is then replicated along the Z-axis at intervals of the grid spacing to yield the full mesh grid. The UCVM framework converts each point in this grid from map coordinates back to geographic coordinates, queries them by tiling all velocity models in the same manner as *ucvm_query*, and builds the mesh.

The Hercules wave propagation software eliminates the need for an input mesh and instead utilizes an Etree database. The Etree database format is an octree of material properties that minimally spans the simulation region, yet is typically much larger. Since it is based on the octree data structure, the database supports variable resolution octants and an efficient storage format. The Etree library augments these advantages by also providing an efficent querying mechanism for extracting information. An Etree database is intended to be re-usable in different simulations, and each is rated for a maximum supported simulation frequency.

The UCM framework utility *ucvm2etree* is employed to generate an Etree database for use with Hercules. The program accepts a configuration file that defines the extent of the Earth's surface and crust to span, the desired maximum simulation frequency to support, and the list of velocity models to be used for extraction. Due to the differences between this database format and that of a regular mesh, the extents are specified in an alternative format to that used in the *ucvm2mesh* utility. A two-dimensional map projection is applied as before, but no grid is explicitly defined; instead, the desired total length along each dimension in projection units is used to derive a large bounding box for the Etree in projection coordinates. This three-dimensional bounding box is then divided into a two-dimensional logical array of rectangular cuboids (columns) of fixed size. This decomposition into columns assists in parallelization since the material properties for a particular column can be extracted independently from the others.

For each column of the bounding box, *ucvm2etree* queries the material properties Vp, Vs, and ρ from the tiled set of velocity models. The material properties are inserted into the database in a top-down fashion, starting from the surface and ending at the maximum depth. As the program moves down the Z-axis, the octant size is adaptively calculated by the following formula.

$$s_{octant} = \frac{l_{max}}{2^{\left(\left\lceil \log_2\left(\frac{l_{max}}{s_{max}}\right)\right\rceil\right)}}$$

The variable l_{max} is the length of the longest side of the Etree bounding box, and s_{max} is the maximum possible octant size that can support the local minimum shear wave velocity. This latter value is determined by the following formula.

$$s_{max} = \frac{v_{min}}{m f_{max}}$$

The variable v_{min} is the minimum shear wave velocity encountered in a sampled two-dimensional slice of the column at this depth. The values f_{max} and m represents the maximum frequency and points per wavelength, respectively, that were specified by the user. This calculation yields a valid octant size for the Etree and ensures that each column is extracted at a resolution which supports the maximum simulation frequency.

TODO: parallel meshing/etree extraction

misc utils: mesh2netcdf, ecoalesce, ecompact

3.3. Application Programming Interface

TODO: how to interface programmatically with UCVM

3.3.1. Architecture

TODO: describe architecture: proj4, etree library, layering/dependencies of framework

3.3.2. Extensibility

TODO: describe how to support new DEMs, Vs30 maps, and models

The DEM may be modified to cover any arbitrary region of the Earth's surface, provided adequate resolution elevation datasets exist.

Alternative Vs30 maps may be integrated as new datasets become available.

utils: grd_query, grd2etree

4. Computational Performance

It would be desirable to do some experiments in terms of performance, especially for the parallel applications utilities in UCVM. With some examples of how long the same thing would take if doing it differently. We will need to discuss this.

5. Recent Case Applications

This section will be dedicated to show case applications. Some ideas for potential subsections follow.

5.1. Visualization and Model Comparisons

5.2. Chino Hills Simulation Series

5.3. CyberShake

6. Summary and Conclusions

A couple of paragraphs with a summary of what is shown in the paper and a few key conclusions about the impact that we expect UCVM has already have and will have on earthquake research.

References

Brocher, T.M., 2008. Compressional and shear-wave velocity versus depth relations for common rock types in northern California 98, 950–968. URL: `http://www.bssaonline.org/cgi/content/abstract/98/2/950`, doi:10.1785/0120060403.

Magistrale, H., Day, S., Clayton, R.W., Graves, R., 2000. The SCEC southern California reference three-dimensional seismic velocity model version 2 90, S65–S76. URL: `http://www.bssaonline.org/cgi/content/abstract/90/6B/S65`, doi:10.1785/0120000510.