

UNITED STATES
DEPARTMENT OF THE INTERIOR
U.S. GEOLOGICAL SURVEY

**SMSIM — Fortran Programs for Simulating
Ground Motions from Earthquakes:
Version 2.3 — A Revision of OFR 96–80–A**

by

David M. Boore¹

A modified version of OFR 00–509, describing the program
as of 15 August, 2005 (version 2.30)

This report is preliminary and has not been reviewed for conformity with U.S. Geological Survey editorial standards or with the North American Stratigraphic Code. Any use of trade, product, or firm names is for descriptive purposes only and does not imply endorsement by the U.S. Government.

Although these programs have been used by the U.S. Geological Survey, no warranty, expressed or implied, is made by the USGS as to the accuracy and functioning of the programs and related program material, nor shall the fact of distribution constitute any such warranty, and no responsibility is assumed by the USGS in connection therewith.

¹ U.S. Geological Survey, MS 977, 345 Middlefield Rd., Menlo Park, CA 94025

**SMSIM — Fortran Programs for Simulating
Ground Motions from Earthquakes:
Version 2.3 — A Revision of OFR 96–80–A**

TABLE OF CONTENTS

INTRODUCTION	1
METHOD	2
PROGRAM OVERVIEW	5
COMPILATION AND MODIFICATION	6
SMSIM PROGRAMS	
ANNOTATED LIST OF SMSIM PROGRAMS	
Drivers	7
Subroutines Used by Random-Vibration Programs	9
Subroutines Used by Time-Domain Programs	10
Subroutine Modules Common to the <i>RV</i> , <i>TD</i> , and <i>FAS</i> Programs	12
INPUT AND OUTPUT OF SMSIM AND FAS PROGRAMS	
Input	15
Output	23
SITE-AMPLIFICATION PROGRAMS	
ANNOTATED LIST OF SITE-AMPLIFICATION PROGRAMS 25	
INPUT AND OUTPUT OF SITE-AMPLIFICATION PROGRAMS	
SITE_AMP	26
F4NRATTLE	28
NRATTLE	29
PWR2LYR	29
LAYR2PLT	29
EXAMPLE OF USING SITE_AMP AND NRATTLE	29
ACKNOWLEDGMENTS	30
REFERENCES	30
FIGURES:	
1. Motions computed using various rms-to-peak relations	34
2. Sample input file for the SMSIM programs	35
3. The specification of geometrical spreading	36
4. The specification of Q	37
5. The specification of path duration	38
6. The specification of site amplification	39
7. Parameters used to define the exponential window	40
8. Dependence on type of window: M 4, $r = 10$	41

9. Dependence on type of window: M 4, $r = 200$	42
10. Dependence on type of window: M 7, $r = 10$	43
11. Dependence on type of window: M 7, $r = 200$	44
12. Dependence on number of simulations: M 4, $r = 10$	45
13. Dependence on number of simulations: M 7, $r = 10$	46
14. Sample output summary file: GM_RV_DRVR program	47
15. Sample output column file: GM_RV_DRVR program	48
16. Sample output file with p _{gd} , p _{gv} , p _{ga}	49
17. sample output time series file: GM_TD_DRVR program	50
18. Time series produced by GM_TD_DRVR program	51
19. Sample input file for the SITE_AMP program	52
20. Output file for the SITE_AMP program	53
21. Velocity vs. depth for various models for which amplifications were computed	54
22. Amplifications computed using SITE_AMP and NRATTLE	55

SMSIM — Fortran Programs for Simulating Ground Motions from Earthquakes: Version 2.3 — A Revision of OFR 96–80–A

by

David M. Boore

INTRODUCTION

A simple and powerful method for simulating ground motions is based on the assumption that the amplitude of ground motion at a site can be specified in a deterministic way, with a random phase spectrum modified such that the motion is distributed over a duration related to the earthquake magnitude and to distance from the source. This method of simulating ground motions often goes by the name “the stochastic method.” It is particularly useful for simulating the higher-frequency ground motions of most interest to engineers, and it is widely used to predict ground motions for regions of the world in which recordings of motion from damaging earthquakes are not available. This simple method has been successful in matching a variety of ground-motion measures for earthquakes with seismic moments spanning more than 12 orders of magnitude. One of the essential characteristics of the method is that it distills what is known about the various factors affecting ground motions (source, path, and site) into simple functional forms that can be used to predict ground motions. **SMSIM** is a set of programs for simulating ground motions based on the stochastic method.

This Open-File Report is a revision of an earlier report (Boore, 1996) describing a set of programs for simulating ground motions from earthquakes. The programs are based on modifications I have made to the stochastic method first introduced by Hanks and McGuire (1981). The report contains source codes, written in Fortran, and executables that can be used on a PC. Programs are included both for time-domain and for random-vibration simulations. In addition, programs are included to produce Fourier amplitude spectra for the models used in the simulations and to convert shear velocity vs. depth into frequency-dependent amplification. The revision to the previous report is needed because the input and output files have changed significantly, and a number of new programs have been included in the set.

The programs do not include extended-fault models, nor do they account for path and site effects by direct computations of wave propagation in layered media (but such path and site effects can be captured in the program by piecewise-continuous frequency- or

distance-dependent functions specified by the user). Furthermore, the random-vibration calculations do not make use of the many advancements in random-vibration theory subsequent to the early work of Cartwright and Longuet-Higgins (1956).

The point-source method as programmed in **SMSIM** is at the heart of a number of extended-fault stochastic method ground-motion simulation algorithms, including **FINSIM** by Beresnev and Atkinson (1998) and **EXSIM** by Motazedian and Atkinson (2005).

The programs are distributed on an “as is” basis, with no warranty of support from me. I would, however, appreciate being informed about bugs, and I always welcome suggestions for improvements to the codes. Please note that I have made little effort to optimize the coding of the programs or to include a user-friendly interface. Speed of execution has been sacrificed in favor of a code that is intended to be easy to understand, although on modern computers speed of execution is rarely a problem (large numbers of runs may be needed in inversion studies, in which case speed optimization of the **SMSIM** codes may be useful, for example as has been done by Scherbaum *et al.*, 2005). I will be pleased if users incorporate portions of my programs into their own applications.

Other stochastic-method codes are available and in common use. In particular, the reader is directed to the programs in Volume VIII of Herrmann (1996) and **RASCAL** by Silva and Lee (1987). I have not made a detailed comparison of my codes to these other codes and therefore cannot make any statements about the relative strengths and weaknesses of the various codes.

The programs can be obtained from my web site (<http://quake.usgs.gov/~boore>). The programs are contained in 4 files. The source code, executables, sample input, and sample output have been compressed into two binary files with the names **SMSIMxxx.ZIP** and **SITEAxxx.ZIP**, where “xxx” is the version number. The manual is contained in file **smsim_manual.pdf**. A file named **README.TXT** contains information about installing the programs, as well as an abbreviated history of changes in the program. I recommend unzipping the files into a folder with the name “smsim”, because Fortran **include** statements used in the various routines expect to find files in a folder named “smsim”.

METHOD

A description of the method is given in Boore (1983), Boore and Joyner (1984), Boore (1986), Joyner and Boore (1988), and Boore (2003) and will not be repeated here, other than to say that the radiation from a fault is assumed to be distributed randomly over a time interval whose duration is related to the source size and possibly the distance from

the source to the site. The detailed parameters used to characterize the source, path, and site effects are described later in this report.

The ground motion can be obtained via time-domain (*TD*) simulation, from which peak parameters such as peak acceleration and response spectra can be obtained (mean values of the parameters require a Monte Carlo simulation with many realizations for a given set of input parameters). The peak parameters also can be obtained directly using random-vibration (*RV*) theory[†]. This is a much quicker way of obtaining the peak parameters, but it is not useful if time series are needed in the analysis. In addition, there are assumptions in the random-vibration theory that are not present in the time-domain simulations. For this reason, the time-domain simulations can be considered “truth”; many simulations are needed (on the order of 50 or more), however, in order for the square-root-of-*n* reduction of noise to provide accurate estimates of the peak parameters. In general, I have found the random-vibration simulations to be good estimates of the ground motions in almost all cases, at greatly reduced computer time.

The code contains an improvement to my pre-1996 implementation of random-vibration theory. That an improvement was needed is shown in Figure 1, which compares the response spectrum computed using *RV* and *TD* simulations. The heavy line is the *TD* simulation, and the solid dots are the result of what used to be the preferred *RV* method. The results from the two methods track one another very well, except for certain period ranges where the *RV* results show discontinuous changes in value. When I first noticed these differences, I had no explanation for them. I now understand why they occur and have found a way to prevent them (the improved results are given by the circles). The explanation has to do with how I treated the following integral from Cartwright and Longuet-Higgins (1956; their equation (6.8)):

$$\frac{1}{\sqrt{2}} \int_0^\infty \{1 - [1 - (1 - \epsilon^2)^{1/2} e^{-\Theta}]^N\} \Theta^{-1/2} d\Theta, \quad (1)$$

where ϵ is computed from the spectral moments and is a measure of the bandwidth of the spectrum, and N is the number of extrema, proportional to the square root of the ratio of the fourth and second spectral moments. This integral is the ratio of the peak and *rms* motions, which for our purposes is the fundamental piece of information provided by random-vibration theory. (Cartwright and Longuet-Higgins’ equation (6.8) is an approximation to their equation (6.4); the code for computing equation (6.8) is

[†] Some would prefer the term “random-process theory”; I have used “random-vibration theory” because many of the applications are to the vibrations of harmonic oscillators and because the term is more familiar to engineers.

simpler than that for equation (6.4), and judging from the comparisons with time-domain calculations in this report and other comparisons that I have made, it is an excellent approximation for the ranges of magnitudes, distances, and oscillator periods of interest in earthquake engineering. Using equation (6.4) would also require redoing the analysis of Boore and Joyner (1984) for determining the duration used to compute the *rms*— the Boore and Joyner results are based on equation (6.8)). As described in my first paper on the stochastic method (Boore, 1983), I expanded the term in square brackets using the binomial series and integrated term-by-term. This gave equation (21) in Boore (1983). The expansion assumes that N is an integer, but N is computed from spectral moments and in general is not an integer. In the calculations shown by the solid dots in Figure 1, however, the real number N was converted to an integer to determine how many terms of the series to include in the sum (equation (21) in Boore, 1983). For small N (e.g., for long-period oscillator response for short-duration earthquakes), changes by one integer lead to the discontinuous offsets seen in Figure 1. The solution to this is simple: calculate the integral in equation (1) numerically. The integral has an integrable singularity that is easily removed by the variable transformation

$$z = \Theta^{1/2},$$

and the integrand is well behaved, having a simple shape and decaying rapidly with increasing z . Using routines from Press *et al.* (1992), the integration is very rapid. Doing the integration numerically has another advantage: before, I devised an *ad hoc* scheme for switching from what I called the “exact” solution (the summation given by equation (21) in Boore, 1983, which was used to compute the solid dots in Figure 1) to the asymptotic expansion of the integral. This scheme is discussed on p. 82 of Joyner and Boore (1988). Now there is no need to switch from one approximation of the integral to another— one simply computes the integral numerically at all times. Speaking of the asymptotic expansion (which is commonly used in applications of random-vibration theory), the two-term approximation is shown by the light line in Figure 1; it is clearly inadequate at long periods.

One of the most important messages from the comparisons shown in Figure 1 is how well the *RV* method works, even for excitations much shorter than the oscillator period (the $M = 4.0$ source, with a stress parameter of 200 bars, has a duration of 0.24 sec). For small earthquakes, basic assumptions in the *RV* method, such as the independence of peaks and the stationarity of the motion, are violated, and corrections to the *RV* method are needed (e.g., Boore and Joyner, 1984, and Liu and Pezeshk, 1999). Further comparisons are shown in Figures 8 through 13, referred to in the discussion of input parameters.

PROGRAM OVERVIEW:

The set of programs are collectively called **SMSIM** (**S**tochastic **M**odel **S**IMulation or **S**trong **M**otion **S**IMulation, take your pick). Separate programs are included for random-vibration and time-domain simulations, but an effort has been made to make the input and output parameter files the same for both applications. The programs include a number of drivers that call subroutines and modules of subroutines.

The drivers provided in this report produce peak acceleration, peak velocity, peak displacement, Arias intensity, and response spectra for a range of oscillator periods. The time-domain drivers include the option of writing to a file a specified subset of the suite of time series used to obtain peak motions. Some of the drivers obtain a single magnitude and distance interactively from the user; others read the magnitudes and distances from a control file.

The modules were designed so that the drivers can be easily modified to produce the ground-motion parameters for other combinations of magnitude, distance, or input parameters.

Programs are also given to compute Fourier spectral amplitudes corresponding to the model specified by the input-parameter file, either directly in the frequency domain or from Fourier transformation of the simulated time series (to provide an independent check of the calculations).

Also included in the package are a set of programs for computing site response for *SH* waves. These include site-amplifications using the square-root of the effective seismic impedance (sometimes known as the “quarter-wavelength approximation”— see Boore, 2003, for more detail), as well as amplifications based on the complete solution of plane-wave propagation in a stack of constant-velocity layers. The square-root impedance amplifications are a useful first-order approximation to the complete amplification, and the program `SITE_AMP` provides a useful way to digitize a velocity model made up of a series of linear velocity gradients into a consistent set of constant-velocity layers, which can then be used in the complete wave-propagation program `NRATTLE.FOR` (written by C. Mueller with modifications by R. Herrmann, included in **SMSIM** with their permission).

The programs do NOT include a graphics user interface; they are intended to be run within a command prompt window. The input is from ASCII files or from the screen. The outputs of the programs are ASCII files, some of which have arranged the results in columnar format for easy importation into spreadsheets or graphics programs.

Included in the package are the executable and source codes, as well as a number of sample input and output files, including the input files used by Atkinson and Boore (1995) and Frankel *et al.* (1996) to simulate ground motions in eastern North America.

COMPILATION AND MODIFICATION:

I used the Lahey/Fujitsu LF95 Version 5.6 compiler from Lahey Computer Systems for all programs. The Lahey compiler was used in order to handle the large arrays in the time-domain programs. In this version of the program I have taken advantage of the Fortran 90 extension that allows the use of dynamically allocatable arrays in the time-domain programs. By so doing I no longer have to guess how large to dimension the time series arrays, and the executable files are much smaller. The relevant sections of code are flagged with “* ALLOCATABLE” comment lines on either side of the sections of code. The comments also indicate how to disable the dynamic allocations if desired. The command to compile and link the program is as follows:

```
lf95 <program_stem_name>.for
```

where `<program_stem_name>` is the stem name of one of the drivers (main programs that call the subroutines).

The compiler binds a run-time DOS-extender into the executable file, so that the Lahey programs are not needed to run the program. The programs run under Windows 9x, Windows 2000, and Windows XP operating systems.

File names are no longer restricted to an eight character stem followed by a three character extension, as they were when the first versions of the programs were compiled using a Fortran 77 compiler.

The compilers are fast enough that I have made liberal use of the Fortran `INCLUDE` statement at the end of drivers to bring in the necessary subroutines and subroutine modules at compile time. It would be more efficient, of course, to produce a library module and link this module with the application-specific programs.

Modifications are easy to make in the routines. As indicated below, changes in source shape and source scaling can be included by modifying the appropriate routines in `RV_TD_SUBS.FOR`; allowance has been made for an input parameter to choose any added source shaping or scaling without changing the original meanings of the input parameter. A more likely modification would be to write new drivers to produce ground motion for specific combinations of magnitude, distance, stress parameter, or oscillator periods.

The emphasis in the output is on various peak measures of ground shaking, but for the convenience of the user, a routine (`A_TS_DRVR.FOR`) is included that will save a subset of acceleration time series as individual files.

The purpose of the programs and subroutines is noted in the following annotated list. This list may not be comprehensive, because other programs may have been added since this manual was written. A short description is given of the purpose of each program; for details, see the program source code. The user may find that some of the subroutines are useful in other applications.

SMSIM PROGRAMS

ANNOTATED LIST OF SMSIM PROGRAMS:

Drivers:

- `GM_RV_DRVR`, `GM_TD_DRVR`: Front-end programs for computing ground motions (“GM”) using random-vibration (“RV”) and time-domain calculations (“TD”). They interactively obtain input and output file names, whether or not response spectra are to be computed, and information needed to set the damping and periods for the response spectra. The periods can be either individual periods or a set of periods between specified limits. The programs obtain input parameters from a file, and pass these parameters to the subsequent subroutine modules through common blocks (dimension statements, variable declarations, and common statements are contained in `SMSIM.FI` and are inserted into `GM_RV_DRVR` and `GM_TD_DRVR` at compile time by the use of the Fortran `INCLUDE` statement.) The input parameters can, of course, be overridden in customizations of the driver program. This would occur if, for example, the motion is required for many values of the stress parameter rather than the one value included in the input parameter file. The programs also obtain interactively the magnitude and distance for the simulation. The `GM_TD_DRVR` program asks if a single realization of simulated acceleration, velocity, and displacement time series is to be saved in a file. The programs obtain input parameters from a file (e.g., `OFR.DAT`). The programs compute peak velocity, peak acceleration, Arias intensity, and, if specified, response-spectral amplitudes, with separate calls to the main subroutine `GM_RV` for the random-vibration calculations and one call to the main subroutine `GM_TD` for the time-domain calculations (the reason for the difference is that I decided that most applications would require simulations for a number of oscillator periods, which for time-domain programs are most efficiently computed by passing a simulated acceleration time series to a subroutine that computes response spectra). After writing

the results to an output file in columnar format, the program loops back for another magnitude and distance, if desired.

- **A_TS_DRVVR**: This driver computes and saves acceleration time series. Unlike **GM_TD_DRVVR**, the routine allows more than one time series to be saved, but it does not compute the various measures of ground motion such as peak acceleration, response spectra, etc. The time series are saved as ASCII files in the “**smc**” format used by the *National Strong Motion Program* of the *U.S. Geological Survey*. This format is described in the file **smcfmt.txt**, included with the software distribution; the latest version of this document can be obtained from <http://nsmp.wr.usgs.gov/smcfmt.txt>. For the convenience of the user, I have included the executable and control files for several of my utility programs for working with **smc** files (a more complete set can be found using the online software link on my web site <http://quake.usgs.gov/~boore>). These programs include **smctsplt** for making a Postscript file containing a plot of many time series on a single page, **smc2vd** to compute velocity and displacement time series from a list of acceleration time series, **smc2rs2** to compute response spectra for a list of acceleration time series, and **smc2asc** to create a single ascii file whose columns are time series, Fourier spectral amplitudes, or response spectra (the ascii file can then be imported into a graphics program for customized plotting). In using these programs I find it very convenient and fast to create a list of **smc** files by entering the command “**dir/on/b *.smc > smc.lst**” in the command prompt window, and then using a text editor to paste the relevant portions of the file **smc.lst** into the various control files for the **smc** utility programs.
- **TMR_RV_DRVVR**, **TMR_TD_DRVVR**: These programs compute motions for magnitudes, distances, and oscillator periods specified in a control file (e.g., **TMR_RV_DRVVR.CTL**). The only thing obtained interactively is the name of the control file.
- **TBL_RV_DRVVR**, **RATIO_TBL**: **TBL_RV_DRVVR**, which has only a random-vibration version, is similar to the preceding programs, except that the magnitudes and distances are specified in increments over specified ranges, and the output is in the form of tables, one table per oscillator period. An example of the control file is **TBL_RV_DRVVR.CTL**. **RATIO_TBL** is a utility program that reads in motions from two tables produced by **TBL_RV_DRVVR** and writes a table containing the ratio of the motions in the two input tables. Its input is specified by a control file (e.g., **RATIO_TBL.CTL**).
- **FAS_DRVVR**, **TBL_FAS**, **FAS_TD_DRVVR**: These programs compute Fourier amplitude spectra. **FAS_DRVVR** and **TBL_FAS** call the basic subroutines used to evaluate equations for

the spectral amplitudes, whereas `FAS_TD_DRVR` obtains the Fourier spectral amplitudes by generating a specified number of time series, computing the Fourier spectra of each, and averaging the spectra. `FAS_TD_DRVR` writes out the target spectral amplitude as well as 1) the FAS of the last time series out of *nsims* time series, 2) the average FAS obtained by averaging the FAS of each time series, and 3) the average FAS computed as the square root of the average of the squared FAS of each time series. The program `FAS_DRVR` closely follows the other drivers in obtaining input and output file names. After writing the results to an output file, the program loops back for another magnitude and distance, if desired. The program will compute Fourier amplitude spectra of ground displacement, velocity, and acceleration; it will also compute the Fourier amplitude spectra of the response of up to 10 oscillators.

Subroutines Used by Random-Vibration Programs:

The subroutines are collected into one file named `RV_SUBS.FOR`. The subroutines in this file are described below:

- `GM_RV`: The main subroutine for the random-vibration calculations, called separately for peak velocity, peak acceleration, and response-spectral output. The routine calls subroutines to set some frequency-independent spectral parameters and calls `GET_MOTION`, which does the actual simulations.
- `GET_MOTION`: Computes the necessary spectral moments and uses these moments in the numerical integration that provides the simulated amplitudes. The routine also computes the values based on the one- and two-term asymptotic expansions, in case the user wants to compare them to the results from the direct integration of equation (1). These estimates, *pk-cl-1* and *pk-cl-2* (standing for peak motion from Cartwright and Longuet-Higgins formulation using 1- and 2-term asymptotic expansions), are available through a common block included in `SMSIM.FI`.
- `CL68_NUMRCL_INT`: Calls routines to compute the integral in equation (1) (equation (6.8) of Cartwright and Longuet-Higgins, thus “*CL68*”).
- `CL68_INTEGRAND`: A function defining the integrand in equation (1).
- `AMOM_RV`: A function that returns a spectral moment, computed by adaptive integration. Unlike the straightforward integration of equation (1), I recommend adaptive integration (whose step sizes vary according to the requirements of the integrand) for the spectral moments; the spectral moments can have spike-like integrands, partic-

ularly for lightly-damped oscillators. The problem of fixed-increment integration is particularly critical for long-period oscillators, for which care must be taken that the frequency increment is not too coarse to approximate adequately the spectral moment.

- **DERIVS**: Subroutine needed in the adaptive-integration routine **ODEINT** described in the **RECIPES** section.

Subroutines Used by Time-Domain Programs:

I found it convenient to separate these subroutines into two stand-alone subroutines in individual files (**GM_TD.FOR** and **ACC_TS.FOR**) and a collection of subroutines contained in the file named **TD_SUBS.FOR**. The subroutines in these files are described below:

- **GM_TD.FOR**: Loops over the specified number of simulations, obtaining an acceleration time series by calling **ACC_TS** for each simulation, integrating to produce velocity and displacement time series, from which peak velocities and displacements are obtained, and in addition, if specified, computing response spectral amplitudes from the acceleration time series. The program computes the arithmetic average over the suite of simulations (NOT the log average) of the various measures of ground motion.
- **ACC_TS.FOR**: The main subroutine for the time-domain calculations, returning an acceleration time series. This time series is computed by generating a noise times series, windowing this time series, transforming to the frequency domain, normalizing the spectra such that its mean square amplitude between dc and the Nyquist frequency equals 1.0, applying the various filters to account for source, path, and site (the seismological part of the method), and then transforming back to the time domain. The normalization of the spectra differs somewhat from that of G.M. Atkinson, which was used in Atkinson and Boore (1995). In her implementation, the scaling was in terms of the average spectral amplitude rather than the average of the squared spectral amplitude. The difference between scaling leads to a systematic difference in the results, such that the motions in the tables in the Appendix of Atkinson and Boore (1995) would be reduced by about a factor of 0.89 if my scaling had been used.
- **TD_SUBS.FOR**: This is a collection of subroutines used in the time-domain calculations, as described below:
 - **GET_NPTS**: Obtains integers specifying the start- and stop-times, as well as the total number of points in the time series (a power of 2); this routine provides the maximum dimension used in allocating the arrays.

- **WIND_BOX**: Returns values of a window using a raised cosine-taper at each end.
- **WIND_EXP**: Returns values of an exponential window. The equation for the window has been revised slightly from that given in Boore (1983); details are in the following section on input and output.
- **FORK**: Computes the complex-to-complex discrete Fourier transform; used in program **FAS_TD_DRVR**. For efficiency, the *Numerical Recipes* real-to-complex routine **REALFT** is used in computing the acceleration time series. The only reason I did not use this routine in **FAS_TD_DRVR** is that I am more familiar with **FORK** and was in a hurry to write **FAS_TD_DRVR**.
- **MEAN**: Computes the mean of an array.
- **RMVTREND**: A subroutine that is called within **ACC2VD** if a linear trend is to be removed from the time series; this option is not used in **SMSIM**.
- **DCDT**: A routine written by C.S. Mueller that detrends a time series.
- **MNMAX**: Returns the minimum and maximum of an array.
- **MNMAXIDX**: Returns the minimum and maximum of an array, and the indices of the minimum and maximum values.
- **ACCSQINT**: Computes the integral of the squared acceleration; used in the computation of Arias Intensity.
- **AVGSQ_REALFT**: Returns the average of the squared spectral amplitudes computed by the *Numerical Recipes* program **REALFT**, not including the values at zero frequency and the Nyquist frequency.
- **ACC2VD**: Returns the velocity and displacement times series. The integration assumes that the acceleration trace is made up of piecewise continuous straight lines (this is the same assumption made in computing the response spectral amplitudes).
- **RDCALCDP**: Computes the relative displacement of the response of an oscillator to a specified motion; see the program listing for authorship.
- **SMCREAD**: Reads an ASCII file in *SMC* format and returns arrays of the the headers

and time series.

- **SMCWRITE:** Writes a time series as an ASCII file in *SMC* format.
- **FS_TD:** The main routine for computing the Fourier acceleration spectra using time-domain calculations. This routine is identical to `gm_td`, but it also computes and writes to a file the Fourier amplitude spectra computed from the time-domain simulations.

Subroutine Modules Common to the *RV*, *TD*, and *FAS* Programs:

Many of the subroutines are common to the *RV*, *TD*, and *FAS* programs, and they have been collected into three files, as listed below. In addition, a file with declaration and common statements is used by all of the programs.

- **SMSIM.FI:** This is the file with the declaration, dimension, and common statements.
- **RV_TD_SUBS.FOR:** This includes the following routines:
 - **BANNER:** This writes the current version number when called by programs writing output files.
 - **GET_PARAMS:** Reads the input parameters from a file and computes the upper limit of integration for *RV* calculations and the number of points for the FFT calculation in *TD* simulations.
 - **WRITE_PARAMS:** Writes the input parameters to a file.
 - **SPECT_AMP:** A frequency-dependent function that computes the Fourier spectral amplitudes.
 - **CONST_AMO_GSPRD:** Computes the frequency-independent part of the Fourier spectrum, including the geometrical spreading factor.
 - **GSPRD:** A function that computes the geometrical spreading factor.
 - **BUTTRLCF:** A function that returns the response of a bidirectional (acausal) high-pass Butterworth filter.
 - **SPECT_SHAPE:** A frequency-dependent function that computes the displacement source spectrum, normalized to unity at zero frequency. Several spectral shapes

are built-in, and the routine can be customized to include any arbitrary shape.

- **SPECT_SCALE:** Returns parameters that control the scaling of the source spectrum with source size. These include seismic moment and corner frequencies. The scalings include single-corner frequency and the Joyner (1984) and Atkinson (1993) two-corner-frequency scalings. The routine can be customized to include other spectral scalings. See the file for other source models (as of 17 August 2005, the routine defines ten models).
- **SITE_AMP_FACTOR:** A frequency-dependent function that computes the site amplification factor.
- **DIMIN:** A frequency-dependent function that returns the spectral diminution factors, including kappa, fmax, and whole path Q.
- **KAPPA_F:** A function that returns a possibly magnitude-dependent value for the diminution parameter κ .
- **Q:** A frequency-dependent function that computes the whole-path Q.
- **DURSOURCE:** A function that returns the source duration for an earthquake with specified corner frequencies.
- **DURPATH:** A function that computes the part of the duration that depends on distance rather than earthquake size.
- **HARMOSCF:** A frequency-dependent function that evaluates the amplitude response of a harmonic oscillator. It is used in the *RV* calculations of response spectra.
- **SMSIMFAS:** A subroutine that returns the Fourier amplitude spectrum of the model, based on the various functions that define the spectrum.
- **BJFV1V2F:** Returns the correction factor to apply to response spectra computed using one value of the average velocity V_{30} to obtain the response spectra for another value of V_{30} . The calculations are based on the empirical equations of Boore *et al.* (1997). The function is used in drivers TBL_RV_DRVR and TBL_FAS.
- **SKIP:** A simple routine that skips over a specified number of lines while reading a file.

- **UPSTR**: Changes a character string to uppercase (written by Larry Baker).
- **GET_LUN**: Obtains an available number for a logical unit (and thus avoids conflicts if logical units are defined in various subprograms).
- **TRIM_C**: Trims leading and trailing blanks from a character string.
- **SKIPCMNT**: Skips lines in an input file preceded by the character “!” (thus allowing an arbitrary number of comments before a line of data).
- **DATEIME.FOR**: This includes the following routines:
 - **GET_DATE**: Returns the system date. The routine uses Lahey Fortran system calls, but the modifications needed by the Microsoft Fortran compiler are indicated in the source code.
 - **GET_TIME**: Returns the system time. The routine uses Lahey Fortran system calls, but the modifications needed by the Microsoft Fortran compiler are indicated in the source code.
 - **TIME_DIFF**: Computes the difference in time obtained with two calls to *GET_TIME*; it is not compiler-dependent.
- **RECIPES.FOR**: Except for a few minor modifications indicated in the file **RECIPES.FOR**, the routines in this module are taken directly from the second edition of *Numerical Recipes* (Press *et al.*, 1992). I was not able to obtain permission to distribute the source code, either via hard-copy or via diskette, without paying a license fee. For this reason I have swapped out the actual routines for placeholders and descriptions of modifications. The routines have been linked into executables of the various drivers included in the distribution, and I do have permission to use them in this way. Users of anything but these executables must obtain their own *Numerical Recipes* routines, which is excellent advice in any case. Information regarding *Numerical Recipes*, including purchase information, can be found at <http://www.nr.com/>. See **RECIPES.FOR** for the required subroutines and the few modifications to these routines.

INPUT AND OUTPUT OF SMSIM AND FAS PROGRAMS:

INPUT: Some of the drivers obtain much of the input interactively from the screen. These programs ask questions of the user regarding input file name, file name for summary listing, and file name of file containing results in columnar form, suitable for import into graphics programs. In addition, the user is asked to provide information regarding whether or not response spectra are to be computed and the damping and the periods for which the spectra will be computed. The periods can be entered individually or can be computed by the program for a specified range and number of periods (logarithmically spaced). Alternatively, the spectra can be computed at the standard set of 91 periods used by the USGS and CSMIP in their routine processing of strong-motion data. In addition, the time-domain program `GM_TD_DRVR` asks if a sample of the acceleration and velocity time series should be saved, and if so, which sample.

After the results are written to various output files, the program asks if computations are to be made for another distance and magnitude, and if so, asks for the name of the file to which the output will be written in columnar format.

The numbers entered in response to a program query need not contain decimal points. For example, a period of 2.0 secs can be entered as “2.0” or “2”. In addition, if the user forgets to enter the parameter on one line, the program will expect the input on the next line.

As indicated in the annotated list of programs, some of the programs obtain input information from control files.

All of the drivers make use of an input file giving values for basic parameters that specify the particular model for which motions will be simulated. An example of such a file (`OFR.DAT`) is given in Figure 2. The parameters in this file do not represent any particular model that I have used in applications to either western North America (e.g., Boore, 1983, 1986; Boore *et al.*, 1992) or eastern North America (e.g., Boore and Atkinson, 1987; Boore and Joyner, 1991; Atkinson and Boore, 1995). The parameters have been chosen to illustrate the input parameters needed by the programs. **THE PARAMETERS ARE NOT TO BE USED BLINDLY FOR A PARTICULAR APPLICATION.** I am reluctant to give input files with the parameters used in some of my papers because my ideas concerning the appropriate parameters are evolving. For the convenience of the reader, however, I have included in the self-extracting zip file the input-parameter files for several models, including the Atkinson and Boore (1995) model (`AB95.DAT`), the Frankel *et al.* (1996) model used in the U.S. National Hazard maps (`E_BCK01L.DAT`), and a single-corner-

frequency coastal California model (WR032496.DAT)(I use the term “coastal California” to reflect more accurately the source of the data used in determining the parameters than the commonly used phrases “western United States” or “western North America”; this parameter file must not be used for distances beyond 100 km.) To emphasize the evolving nature of the coastal California model, I have used the date of the latest modification of the file as part of the file name.

The input-parameter file is made up of lines of text and lines containing the input parameters. The lines of text are for the convenience of the user; the programs skip over them. It is very important that the number of text lines remain the same, however, for otherwise the program will attempt to read a text line as a parameter line. In other words, only change the parameter lines! In addition, list-directed input is used. This means that all parameters must be included, even if some are not used.

The *RV* and *FAS* drivers do not need the parameters in the last several lines of input file, but these lines should be included nevertheless.

Content of Model Specification File (e.g., OFR.DAT):

1. *rho*, *beta*, *prttn*, *radpat*, *fs*: These parameters are the density (in gm/cc) and shear-wave velocity (in km/s) in the vicinity of the source, the partition factor (to partition the S wave energy into two horizontal components, usually given by $1/\sqrt{2}$; it should be consistent with the next parameter), the radiation pattern, averaged over some portion of the focal sphere (this should refer to the radiation factor of the total S-wave radiation if the partition factor is taken to be $1/\sqrt{2}$; see Boore and Boatwright, 1984, for tables of values), and the free surface factor (usually equal to 2). Note that *rho*, *beta* were 2.7, 3.2 and 2.8, 3.8 in the applications of Boore (1983) and Atkinson and Boore (1995), respectively. I now suggest 2.8, 3.5 for WNA and 2.8, 3.6 for ENA.
2. *source number*, *pf*, *pd*: These parameters control the shape of the spectrum. As explained in the text lines in the input-parameter file, the *source number* specifies whether the spectral shape is a single-corner spectrum (*source number* = 1), one of several possible double-corner spectra (e.g., *source number* = 2 for Joyner (1984), as modified in Boore and Joyner (1991); *source number* = 3 for Atkinson (1993); *source number* = 9 for Atkinson and Silva, 2000). Other source models have been hardwired in; see SPECT_SHAPE or SPECT_SCALE in RV_TD_SUBS.FOR for the current list of built-in source models. The parameters *pf* and *pd* are used if *source number* = 1 and control the spectral shape given in the following equation:

$$S(f) = 1/(1 + (f/f_c)^{pf})^{pd}, \quad (2)$$

where f_c is the corner frequency. For the usual single-corner model, $pf = 2$ and $pd = 1$. A sharper corner, preferred by some, is given by $pf = 4$ and $pd = 0.5$. In all cases, an omega-square model requires that $pf \times pd = 2$.

3. *stressc*, *dlsdm*, *fbdfa*, *amagc*: These parameters control the scaling of the spectral amplitudes with source size, primarily by specifying the dependence of the corner frequencies on magnitude. The parameters are not used for many of the source numbers, whose shapes and scaling are specified by relations between corner frequencies and seismic moment (see Atkinson and Boore, 1998, for a convenient formulation of many models), but in this case dummy parameters must be included in the input-parameter file. The parameters *fbdfa* and *amagc* are used if *source number* = 2, in which case *fbdfa* is the corner frequency f_b divided by f_a and *amagc* is the critical moment magnitude beyond which the scaling is no longer self-similar. If *source number* = 1, then the stress parameter is given by:

$$\Delta\sigma = stressc \times 10^{dlsdm \times (M - amagc)}. \quad (3)$$

The usual case of magnitude-independent stress is given by setting $dlsdm = 0.0$ (note that “ $dlsdm$ ” stands for derivative of log sigma with respect to magnitude). $stressc$ has units of bars.

4. $r_ref, nsegs, (r_{low}(i), a_s(i), b_s(i), m_s(i)), i = 1, nsegs)$: These parameters control the geometrical spreading, as represented by $nsegs$ segments, each segment starting at r_{low} with a distance-dependence of r^{slope} for distances beyond r_{low} . The slope of the j th line segment($slope(j)$) can be magnitude dependent, as given by the following equation:

$$slope(j) = a_s(j) + b_s(j)(M - m_s(j)). \quad (4)$$

The magnitude dependence was introduced following Silva *et al.* (2002), who use it to capture a finite-fault effect while still using a point-source approximation (W. Silva, oral commun., 2002). r_ref is the reference distance, in kilometers, for which the geometrical spreading term = 1.0; this variable has been added for generality, but in almost all cases it should be set as follows: $r_ref = 1.0$. As shown in Figure 3, in the sample input-parameter file, the geometrical spreading is r^{-1} from 1.0 (actually, any distance less than 70 km) to 70 km, r^0 from 70 to 130 km, and $r^{-0.5}$ beyond 130 km (this is the dependence used by Atkinson and Boore, 1995).

5. $fr1, Qr1, s1, ft1, ft2, fr2, Qr2, s2, c_q$: The whole-path attenuation is given by

$$\exp(-\pi fr/Q(f)c_q), \quad (5)$$

in which the function $Q(f)$ is described by the parameters in this entry of the input-parameter file. As shown in Figure 4, $Q(f)$ is given by a piecewise continuous set of three straight lines in $\log Q$ and $\log f$ space. The first and third lines have slopes of $s1$ and $s2$ and values of $Qr1$ and $Qr2$ at reference frequencies $fr1$ and $fr2$, respectively. The first and third lines apply for $f \leq ft1$ and $f \geq ft2$, respectively, with a straight line in $\log Q, \log f$ space connecting the values of Q at the transition frequencies $ft1$ and $ft2$ (in other words, $Q(f) = Qr1(f/fr1)^{s1}$ for $f \leq ft1$ and $Q(f) = Qr2(f/fr2)^{s2}$ for $f \geq ft2$, with a connecting line between these two). I decided on this representation after much experimentation; it is the simplest way of representing a complicated $Q(f)$ function with terms that are familiar to most users. The values in the sample input-parameter file have been chosen to represent closely the $Q(f)$ function given in Boore (1984) and in my WNA applications. (As in all of the input parameters for specific applications, this function should be confirmed or modified based on special studies; in particular, intermediate- and long-period motions at large distances can be sensitive to the location of the low-frequency branch of the $Q(f)$ function, which is not well determined from data.) Note that because the decision of which line segment to

use depends solely on the transition frequencies, the relative size of the reference frequencies does not matter (i.e., $fr2$ could be less than $fr1$). Two special cases should be mentioned: $Q = Q_0$ (a constant) and $Q = Q_r(f/f_r)^s$. The constant- Q case is given by specifying $s1 = 0$, $s2 = 0$, $Qr1 = Q_0$, $Qr2 = Q_0$, and any non-zero values for $fr1$, $fr2$, $ft1$, and $ft2$. The special case of a single power-law dependence is specified by $Qr1 = Q_r$, $Qr2 = Q_r$, $fr1 = f_r$, and $fr2 = f_r$, and any non-zero values for $ft1$ and $ft2$. All frequencies should have units of Hz. The velocity c_q may be different than the velocity β . For consistency c_q should equal the velocity used in deriving the Q function, although in most applications to date it has been implicitly assumed (even if incorrectly) that c_q and β are the same (c_q was not included as a variable in version 1.0 of the **SMSIM** package).

6. w_fa , w_fb : The source duration used in the calculations is given by

$$dursource = w_fa/fa + w_fb/fb, \quad (6)$$

where fa and fb are the source corner frequencies. For the single corner-frequency model (*source number* = 1), $fa = fb$, so any combination of weights w_fa and w_fb can be used, as long as they add up to the desired weight. In my WNA applications, I used $w_fa = 1.0$ and $w_fb = 0.0$. In Atkinson and Boore (1995), $w_fa = 0.5$ and $w_fb = 0.0$.

7. $nknots$, ($rdur(i)$, $dur(i)$, $i = 1, nknots$), *slope of last segment*: These parameters are used in the specification of the path duration (*durpath*) by a series of straight-line segments with parameters $nknots$, $rdur$, dur , *slope of last segment*. where $nknots$ is the number of intersections between line segments. The meaning of these parameters is indicated in Figure 5. The values given in the figure correspond to those in the input-parameter file, which in turn were chosen to represent the duration used by Atkinson and Boore (1995). For my WNA applications I used one segment with a slope of 0.05 (i.e., $nknots = 1$, $rdur = 0.0$, $dur = 0.0$, *slope* = 0.05), but this had little empirical basis. In contrast, the path duration in the application by Atkinson and Boore (1995) for ground motion in eastern North America was based on empirical results. As an aside, Atkinson (1995) finds a very different path-duration relation for western Canada than used by Atkinson and Boore (1995) in eastern North America; similar studies should be done for other regions to determine the path durations.
8. $namps$, ($famp(i)$, $amp(i)$, $i = 1, namps$): The site amplification is approximated by a series of straight-line segments in log amplification, log frequency space, connecting the values $famp$, amp . The amplification for $f \leq famp(1)$ and $f \geq famp(namps)$ is given by $amp(1)$ and $amp(namps)$, respectively. This is shown in Figure 6 (which uses

the parameters in the input-parameter file). **THE NUMBERS IN THE DATA FILE WERE INVENTED FOR THE SAKE OF ILLUSTRATION AND SHOULD NOT BE USED IN APPLICATIONS.** Some values for WNA, based on Boore and Joyner (1997), are given in file WR032496.DAT. For no amplification, set $namps = 1$, $amp = 1.0$, and $famp$ equal to any number.

9. fm , $kappa$, $dkappadmag$, $amagkref$: These parameters are used in the following filter:

$$\exp(-\pi\kappa f)/\sqrt{1+(f/fm)^8}. \quad (7)$$

where

$$\kappa = kappa + dkappadmag \times (\mathbf{M} - amagkref). \quad (8)$$

In many applications only one or the other of the two parameters fm and κ are desired; this is easy to implement with appropriate choices of the parameters. For example, to use only fm , specify $kappa = 0.0$ and $dkappadmag = 0.0$; to use only $kappa$, specify a large number for fm . The units of fm , $kappa$ should be consistent with that of f : Hz and 1/Hz. In most applications $dkappadmag = 0.0$, and $amagkref$ in this case can be anything (it is not used).

10. $fcut$, $norder$: In some cases it may be desired to include a low-cut filter in the simulations. This might be the case, for example, for simulations of processed strong-motion data. The parameters $fcut$ and $norder$ control the low-cut filter and are used in BUTTRLCF. The filter is given by the following function:

$$1.0/(1.0 + (fcut/f)^{2.0 \times norder}) \quad (9)$$

(this is the response of a bidirectional filter made up of two Butterworth filters, each of order $norder$). Set $fcut = 0.0$ for no filter.

11. zup , eps_int , amp_cutoff , osc_crrctn : The first parameter specifies the upper limit in the integral of equation (1). This parameter is actually a legacy from an earlier version of the program and has been kept so that the older input parameter files will still work. The parameter is adjusted downward in the program until the integrand is no longer identically zero. The summary output of the random vibration driver RV_DRV includes the value used in the computations; if the adjusted value equals the input value, the input value should be increased until the adjusted value is less than the input value (I doubt that this will ever happen if zup is set to the example value of 10.0). The second parameter specifies the error in the adaptive integration routine ODEINT, and the third is used as the basis for computing the upper frequency limit (fup) used in the random-vibration calculations (this is computed in subroutine GET_PARAMS and

also in RV_DRVR). f_{up} is determined such that the filter in equation (7) has a value of amp_cutoff . I have found the values in the sample input-parameter file to give good results, but the user should experiment to make sure that they are appropriate values. I have included these three parameters for generality, even though I do not anticipate that they will be changed. The last parameter, $osc_corrctn$, controls which method is used to modify the peak-to-rms relation for oscillator response: $osc_corrctn = 1$ for the modification in Boore and Joyner (1984) and $osc_corrctn = 2$ for the modification in Liu and Pezeshk (1999). The Liu and Pezeshk procedure seems to give somewhat better agreement with response spectra computed from time domain calculations than does the Boore and Joyner modification, but the basis for this conclusion is based on limited comparisons; it may be model dependent.

12. $indxwind$, $taper$, eps_w , eta_w , f_tb2te , f_te_xtnd : These parameters control the shape of the window applied to the time series composed of random numbers in the time-domain simulations. Either a box ($indxwind = 0$) or an exponential window ($indxwind = 1$) can be used. $taper$ is used only in the box window; it is the fraction of the duration of motion for which a raised-cosine taper will be applied to the front and to the back of the box window (i.e., the extent of the taper will be $taper \times (dursource + durpath)$ in both the front and the back of the window). $taper$ is not used for the exponential window. The parameters eps_w , eta_w , f_tb2te , and f_te_xtnd are only used for the exponential window. The exponential window is given by:

$$w(t; t_\eta) = a(t/t_\eta)^b \exp(-c(t/t_\eta)), \quad (10)$$

where the parameters a , b , and c are determined such that $w(t)$ has a peak with value of unity when $t = eps_w \times t_\eta$ and $w(t) = eta_w$ when $t = t_\eta$ (see Figure 7). (As I discuss in Boore, 1983, a can also be chosen such that the integral of the square of $w(t)$ equals unity; this is appropriate if the spectrum of the windowed noise is not normalized so that it has a mean square amplitude of unity.) The equations for a , b , and c are given in Boore (1983). The time t_η is given by $t_\eta = f_tb2te \times (dursource + durpath)$; the window is evaluated out to the time $f_te_xtnd \times t_\eta$ (“ f_tb2te ” means “Factor to convert the Box duration to the Exponential duration”, and “ f_te_xtnd ” means “Factor to eXTeND the time t_η ”). The parameters of the window shown in Figure 7 are not those recommended for general use; I recommend $eps_w = 0.2$, $eta_w = 0.05$, and $f_tb2te = 2.0$. The choice of f_tb2te is somewhat arbitrary; the choice is based on good agreement of response spectra computed from accelerations simulated with the exponential window $f_tb2te = 2.0$ and those computed using a box window. The simulations were done using these values and $f_te_xtnd = 1.0$ (the results are not sensitive to the choice of f_te_xtnd , as long as it equals or exceeds unity). Sample output for the box and exponential windows is given in Figures 8 through 11 for

$M = 4$ and 7 , and $R = 10$ and 200 km. The results in the figures indicate that, in general, the response spectral amplitudes from the exponential window are closer to the random vibration results than are those from the box window. This is not surprising, because the correction factor for oscillator response proposed by Boore and Joyner (1984) and used in the SMSIM programs was derived empirically from calculations made using the exponential window.

13. *dur_fctr*, *dt*, *tshift*, *seed*, *nsims*, *iran_type*: These parameters deal with time-domain details. *dt* is the time spacing and *tshift* produces a time shift in the start of time series; this can be useful to accommodate pre-arrival tails due to the noncausal filters used in the analysis. Note that in the time-domain calculations velocity and displacement are computed by integrating the acceleration trace twice. If a low-cut filter is used to simulate processed observed accelerograms, it is important to allow a long enough *tshift* and minimum duration to capture the filter response (the filtering is only done on the acceleration spectrum); otherwise the displacement will probably exhibit an erroneous long-term trend that will dominate the signal (the filter is acausal, so allowance for the filter response before *tshift* is important). It is a good idea to save the time series for one of the suite of simulations to check that the various time-domain parameters have been chosen so that numerical artifacts such as long-period drifts are not present. To help ensure that adequate pre- and post-signal pads are included, an algorithm to determine the number of points used in the FFT is used, using the parameter *dur_fctr* as follows. The algorithm overrides *tshift* if necessary to allow for filter response. For example, if $f_{cut} = 0.1$, $n_{order} = 2$, pads of 15 sec are needed before and after the signal (the pads are given by $1.5 * (n_{order}/2) / f_{cut}$, according to Converse, 1992, p. 2-3). If *tshift* were set to 5 sec, this would be replaced with 15 sec. If, on the other hand, *tshift* were greater than 15 sec then *tshift* would not be overwritten. *dur_fctr* is used to give some control on the total duration used in the FFT. The number of points in the FFT is calculated in the program to be a power of 2 such that

$$npts * dt > \max(filterpad, tshift) + dur_fctr \times (dursource + durpath) + filterpad. \quad (10)$$

seed and *nsims* are the initial seed of the random-number generator and the number of time-domain simulations, respectively. If peak motions are desired, *nsims* should be large enough to reduce the uncertainty in the computed means of the peak motions determined from each realization. Examples of results computed for several *nsims* are given in Figures 12 and 13. In general, the uncertainty in the mean decreases as $1/\sqrt{nsims}$. Note in these figures that the time-domain and random-vibration results

show some systematic disagreements for the larger earthquake. This difference is a maximum of a factor of about 1.12, and is probably related to the assumption in the random-vibration theory that the amplitudes of successive peaks are independent of one another. This is certainly not true for a long-period oscillator response. Corrections schemes for “clumping” might yield a better comparison. I tried one such scheme (due to Toro, 1985), but the comparison was not improved. In view of the aleatory uncertainty in ground-motion data and the epistemic uncertainty in the input parameters, I am willing to live with uncertainties that in general are less than 10 percent in order to take advantage of the greatly increased speed of the random-vibration calculations compared to the time-domain calculations. The last parameter, *iran_type* controls which whether the random noise has a Gaussian (*iran_type* = 0) or a uniform (*iran_type* = 1) distribution. This parameter was introduced to see if the distribution made any difference in the calculated motions. It did not in the test cases used in the study, but the user may want to do his/her own experiment. For normal applications, I suggest *iran_type* = 0.

14. *remove dc from random series?*: This parameter, used in older versions, was removed in version 2.19, because it led to distortions in the long-period spectral content (see Readme.txt for version 2.19 for more comments).

OUTPUT: Several files are produced by the drivers GM_RV_DRVR and GM_TD_DRVR:

- A summary of the input and the results. The summary file from GM_RV_DRVR includes estimates of dominant frequency, as measured from the frequency of zero crossings (equation 27 in Boore, 1983). In addition, the output includes the parameter *eps* that measures the bandwidth of the motion ($eps = \sqrt{1 - \xi^2}$, where ξ is given by equation (22) in Boore, 1983; see Cartwright and Longuet-Higgins, 1956, p. 216–217, for more discussion). Finally, the random-vibration output also includes estimates of the number of extrema (*nx*) and the number of zero crossings (*nz*). The bandwidth parameter ξ and *nx* and *nz* are related by $\xi = nz/nx$.
- A file with columns containing the oscillator period and frequency and response spectra amplitudes (relative displacement, *SD*, pseudo relative velocity, *PSV*, and pseudo absolute acceleration, *PSA*; note that the terms “*PRV*” and “*PAA*” are more logical than “*PSV*” and “*PSA*” for pseudo relative velocity and pseudo absolute acceleration, but I have bowed to convention in the names of the ground-motion measures). The file produced by GM_RV_DRVR also contains columns with normalized spectral shape (*psa/pgs*) and dominant frequency. The file produced by GM_TD_DRVR contains a column of the standard deviation of the spectral amplitudes, normalized by *psv*.

This standard deviation is NOT the same as observed standard deviation of ground motion, but rather is a measure of the variation of the spectra from the individual time series simulations. This file is in a convenient format to be imported into the graphics program that I use (CoPlot, published by CoHort Software, 1-800-728-9878, <http://www.cohort.com>).

- If requested at run time, a file with peak ground displacement, velocity, and acceleration, multiplied by various powers of radial frequency ($\omega = 2\pi f$). This file can be used to plot asymptote lines on a log-log “tripartite” plot for comparison with the response spectra.
- If requested at run time in program GM_TD_DRVR, a file with acceleration, velocity, and displacement time series computed by the time-domain simulation (one out of *nsims* time series).
- Program A_TS_DRVR produces files containing acceleration time series; see the earlier discussion of the program for comments about the format of the output files.

Sample output files are given in Figures 14, 15, and 16 (these are reproductions of files SAMPLE_RV_SUM.SUM, SAMPLE_RS_COL.COL, and SAMPLE_AD_COL.COL, respectively); a sample of the time series file is given in Figure 17, and the whole time series is plotted in Figure 18.

The response of a single-degree-of-freedom oscillator with gain of V and specified natural period (T_0) and damping (η) can be obtained by multiplying the *PSV* output for the specified natural period and damping by the factor $VT_0/2\pi$. This scheme can be used to simulate the response of a Wood-Anderson instrument and thereby to obtain estimates of local magnitude M_L corresponding to the ground motion. According to Uhrhammer and Collins (1990), for a Wood-Anderson instrument $V = 2080$, $T_0 = 0.8\text{s}$, and $\eta = 0.69$. Time series corresponding to the oscillator output are not returned by the programs, although they can be produced by a simple modification to the subprogram RDCALCDP in the module TD_SUBS.FOR; the modification is indicated by a comment in RDCALCDP.

The FAS program creates a summary file and a file with columns of frequency, period, and Fourier spectral amplitude for the ground displacement, velocity, acceleration, and oscillator response. The output is similar to that in Figures 14 and 15.

With the units as given in the discussion of input, the output ground motion will be in cgs units.

SITE-AMPLIFICATION PROGRAMS

ANNOTATED LIST OF SITE-AMPLIFICATION PROGRAMS:

Unlike the **SMSIM** programs, the site-amplification programs are largely self-contained, with the necessary subroutines bundled with the source code of the main programs.

SITE_AMP: Converts a velocity and density model into site amplifications, using the square-root of the impedance (see Boore, 2003, for details). The amplification is relative to the surface motion that would exist if the material were replaced with uniform material whose velocity and density equals those at the source; for this reason, the amplification approaches unity for low frequencies if the source velocity and density equal those in the halfspace of the input model. The velocity model can be a combination of constant velocity layers and linear gradients. The program will compute amplifications at specified frequencies or at frequencies corresponding to the “breakpoints” in the velocity model. The calculation of amplification at specified frequencies allows the program to generate automatically a staircase model for portions of a velocity model represented by a gradient (program **PWR2LYR** can be used to generate a layered model for velocity vs. depth described by a single power law). This is very useful in preparing velocity models for use in layered-matrix site-response calculations (e.g., C. Mueller’s **NRATTLE**; see **F4NRATTLE**). The previous version of the program (Boore, 1996), for which frequencies were determined by the travel time to each breakpoint in the velocity model, required specification of velocities at many intermediate breakpoints. I found that a spreadsheet was useful to prepare the input file, giving depth and velocity values at a number of depths spanned by linear velocity gradients or to break up thick constant velocity layers into a number of thinner layers. This is no longer required. Q or $1/Q$ can be specified, and the cumulative κ through the velocity model will be computed (starting from the top). This value of κ or a specified κ can be used to compute the diminution due to κ .

F4NRATTLE: This program reads a file made by **SITE_AMP** and writes a file in the proper format for use by **NRATTLE**, C. Mueller’s program (as modified by R. Herrmann and myself) for computing the response of a stack of layers to *SH* waves. The header lines in the source code should be consulted for important comments regarding usage. The main use of this program is to take advantage of the ability of **SITE_AMP** to construct constant-velocity layered models approximations to models with linear velocity gradients.

NRATTLE: C. Mueller’s program (as modified by R. Herrmann and myself) for computing the response of a stack of layers to *SH* waves.

PWR2LYR: This utility program generates a constant-velocity layered model for a velocity described by a power law $V = V_0(z/z_0)^\eta$.

LAYR2PLT: This utility program converts a layered velocity and density model into a piecewise continuous velocity function for plotting purposes. As input, it can use the output file made by **SITE_AMP**.

INPUT AND OUTPUT OF SITE-AMPLIFICATION PROGRAMS:

As in the previous programs, input comes from the screen and from a parameter file.

SITE_AMP:

INPUT: The program first asks the user for the names of the input file. It then asks several questions regarding the attenuation parameters to be used in the calculations (these can be included as values of Q or $1/Q$ for each layer, as specified in the input file, or can be the parameter κ , read in from the screen). This is followed by asking the user for the name of the output file. The program then asks for information regarding a linear relation between density and velocity that is used to assign densities for layers for which density is not specified in the input file (the program asks for the values specifying the end points of the line, and the densities are given by the end-point values for velocities outside of the specified range). As explained below, the program looks at the parameter file and queries the user about whether the velocities are a piecewise continuous function of depth or represent a layered model. The program then asks for the velocity and the density in the source region, and the angle of incidence for the incoming waves. Finally, the program asks the user to specify the frequencies at which the amplifications are to be computed. A number of options are available. If no frequencies are specified, the program computes the quarter-wavelength frequencies for each breakpoint in the velocity model. If frequencies are specified, they can be from a file, or can be computed as being linearly or log spaced between endpoints specified interactively. In this case, depths are computed corresponding to each frequency, and a constant-velocity layered model is constructed so that the travel time across each layer equals that in the input model for the same depth range. This is a very convenient way to construct a layered model from a model with linear velocity gradients. Such a layered model can then be used in a layered-matrix wave propagation to compute amplifications, even if the amplifications from **SITE_AMP** are ignored.

A sample input file is given in Figure 19 (this is file **SMPLSAMP.DAT**). The parameters in this file have been made up to illustrate the input; **THEY DO NOT REPRESENT A REAL APPLICATION.**

1. The first column is labeled “Depth”, but it could also contain layer thickness. The units can be anything. If the first entry is “0.0”, the program assumes that the entries will be depths and that the velocities and densities are the values for the specified depth; it asks the user to confirm this. If the first column is depth, then the program simply assumes straight-line connections between the entries. In this way a mix of linearly increasing, constant, and step changes can be included in the model. A constant parameter (velocity or density) over a depth range is entered by the depths that bound the layer, but with the same velocity or density for the two consecutive entries (e.g., as between depths 0.040 – 0.100 and 0.300–1.000 for the velocity parameter in the sample input file); similarly, a layered model can be included easily by entering the velocities and densities on either side of the interface, but with the same depth for the two consecutive entries (e.g., at depths of 0.040 and 0.300 in the sample input file). If the velocity is made up only of constant velocity layers, then the first column of the input file should contain layer thickness.
2. The second column contains the velocities, in any units that match the depth units.
3. The third column contains the densities. An entry of “0.0” will flag the program to use the *DENSITY* function. To compute a density from the velocity, using the relation specified interactively or the default relation built into the program (with end points of 2.5 gm/cm^3 at 0.3 km/s and 2.8 gm/cm^3 at 3.5 km/s , values that I have assumed for generic rock in WNA (see Boore and Joyner, 1997)). A nonzero entry will override the value that would have been computed by the *DENSITY* function (as has been done for a few depths in the sample input file).
4. The fourth column contains attenuation values. If the values are greater than 1.0, then the value is assumed to be Q , otherwise it is assumed to be $1/Q$. Interactive input from the screen determines if the user wants the program to ignore the values in column 4 (in effect, assume that they are all equal to 0.0, with an overall possible an overall value of κ used for the attenuation).

Regarding units: velocity and depth units can be anything as long as they are consistent with one another (i.e., depth in meters should be matched with velocity in meters per second). Note that if the densities are to be computed from the default end points, the velocity units must be kilometers per second. The density units do not have to be consistent with those of depth and velocity. I strongly advise that the density units be given in grams per cubic centimeters.

OUTPUT: One output file is created with columns containing the input model, the

equivalent constant-velocity layered model, the average velocities and densities, and the frequency, amplification, effective κ , and the combined effect of amplification and attenuation. A sample is given in Figure 20. Columns 1–4 repeat the input parameters, with unspecified values of density filled in from the density–velocity relation (column 4 is $1/Q$). Columns 5 and 6 give the depths and travel time from the surface to each depth corresponding to the specified frequencies — the travel time is fundamental for computing the average velocities and thus the square-root impedance amplifications. Columns 7 through 10 contain a constant-velocity approximation to the input velocity model (it is constructed to give the same travel time for each layer as the continuous model over the same depth range) and are included as a convenience in case the approximate amplifications are to be checked using a wave-propagation program that requires a stack of constant-velocity layers. Columns 11 and 12 are the velocities and densities averaged from the surface to the specified depth. Columns 13 and 14 are the specified frequency and the computed amplification. Column 15 contains the cumulative attenuation through the stack of layers, and column 16 gives the combination of the amplification and the attenuation ($A \times \exp(-\pi\kappa f)$). Note that for identification purposes, the stem name of the input name has been used to label columns 13 through 16 (e.g., “ofr_amp” in the example given in Figure 20).

One way in which I use the program `SITE_AMP` is to import the output file into my graphics program and plot the amplification vs. frequency (using log–log axes). I then pick off a set of amplifications and frequencies that will be used to specify the site-amplification filter incorporated into the input file used by the `SMSIM` programs.

F4NRATTLE:

INPUT: Most of the input is contained in the control file, whose name is specified interactively. This program uses a file made by `SITE_AMP` as input. The name of this file, as well as other parameters, are contained in a control file (e.g., `F4NRATTLE.CTL`). An example of the control file has not been reproduced here, but the sample file should be self-explanatory.

OUTPUT: The output is a file in the format required by `NRATTLE`; see file `NRATTLE.CTL` for an example.

NRATTLE:

INPUT: The input parameters are contained in a control file (e.g., NRATTLE.CTL). An example of the control file has not been reproduced here, but the sample file should be self-explanatory. For convenience, F4NRATTLE makes the required file, but the file can also be made independently of F4NRATTLE.

OUTPUT: There are two output files, one giving the velocity model and the complex spectra of the amplification function (NRATTLE.OUT is an example), and the other giving just the absolute value of the amplification as a function of frequency (see NRATTLE.ASC for an example).

PWR2LYR:

INPUT: The input is obtained exclusively from the screen and should be self-explanatory. Options are available for specifying the parameters of the power law and of layering of the output model. It is possible to instruct the program to construct a model with layering such that the travel time across each layer is a constant; in this case, the time across each layer can be specified, or the number of layers to be included between the top and bottom depths can be specified and the constant travel time across each layer will be calculated.

OUTPUT: The output file has a name specified when running the program; the output is self-explanatory.

LAYR2PLT:

INPUT: The program asks the user for the name of a file containing a velocity model; this file can either be an output file from SITE_AMP or can be a file having columns of thickness, velocity, and density (see the first few lines of the source for an example).

OUTPUT: The output file has a name specified when running the program; the output is self-explanatory.

EXAMPLE OF USING SITE_AMP AND NRATTLE:

Examples of using the site amplification programs are given in Figures 21 and 22. Figure 21 is a plot of the velocity model with linear gradients specified in the example file OFR_AMP.DAT, as well as several constant-velocity layered models obtained by running SITE_AMP with various choices regarding the specification of frequencies, and then using LAYR2PLT to convert the layered model into a continuous model (a “stair-step” model) for

plotting. Figure 22 contains amplifications from SITE_AMP (the heavy line) and NRATTLE, where the control files for NRATTLE were made using F4NRATTLE.

ACKNOWLEDGMENTS

I thank Bill Joyner for advice and encouragement over the years and Bob Herrmann for cross-checking of output from our programs and for reminding me that Cartwright and Longuet-Higgins' equation (6.8) is an approximation to their equation (6.4). In addition, I am grateful to Gail Atkinson for her version of my original time-domain simulation program, to Walt Silva for helpful discussions, and to Basil Margaris for provoking me into writing the first version of this report. Chuck Mueller reviewed the manuscript. I also thank Stavros Anagnostopoulos and Jose Roesset for permission to use their program for computing response spectra and Chuck Mueller and Larry Baker for permission to distribute their programs.

This work was partially supported by the Nuclear Regulatory Commission.

REFERENCES

- Atkinson, G.M. (1993). Earthquake source spectra in eastern North America, *Bull. Seism. Soc. Am.* **83**, 1778–1798.
- Atkinson, G.M. (1995). Attenuation and source parameters of earthquakes in the Cascadia region, *Bull. Seism. Soc. Am.* **85**, 1327–1342.
- Atkinson, G.M. and D.M. Boore (1995). Ground motion relations for eastern North America, *Bull. Seism. Soc. Am.* **85**, 17–30.
- Atkinson, G.M. and D.M. Boore (1998). Evaluation of models for earthquake source spectra in eastern North America, *Bull. Seism. Soc. Am.* **88**, 917–934.
- Atkinson, G.M. and W. Silva (2000). Stochastic modeling of California ground motions, *Bull. Seism. Soc. Am.* **90**, 255–274.
- Beresnev, I.A. and G.M. Atkinson (1998). FINSIM— a FORTRAN program for simulating stochastic acceleration time histories from finite faults, *Seism. Res. Lett.* **69**, 27–32.
- Boore, D.M. (1983). Stochastic simulation of high-frequency ground motions based on seismological models of the radiated spectra, *Bull. Seism. Soc. Am.* **73**, 1865–1894.

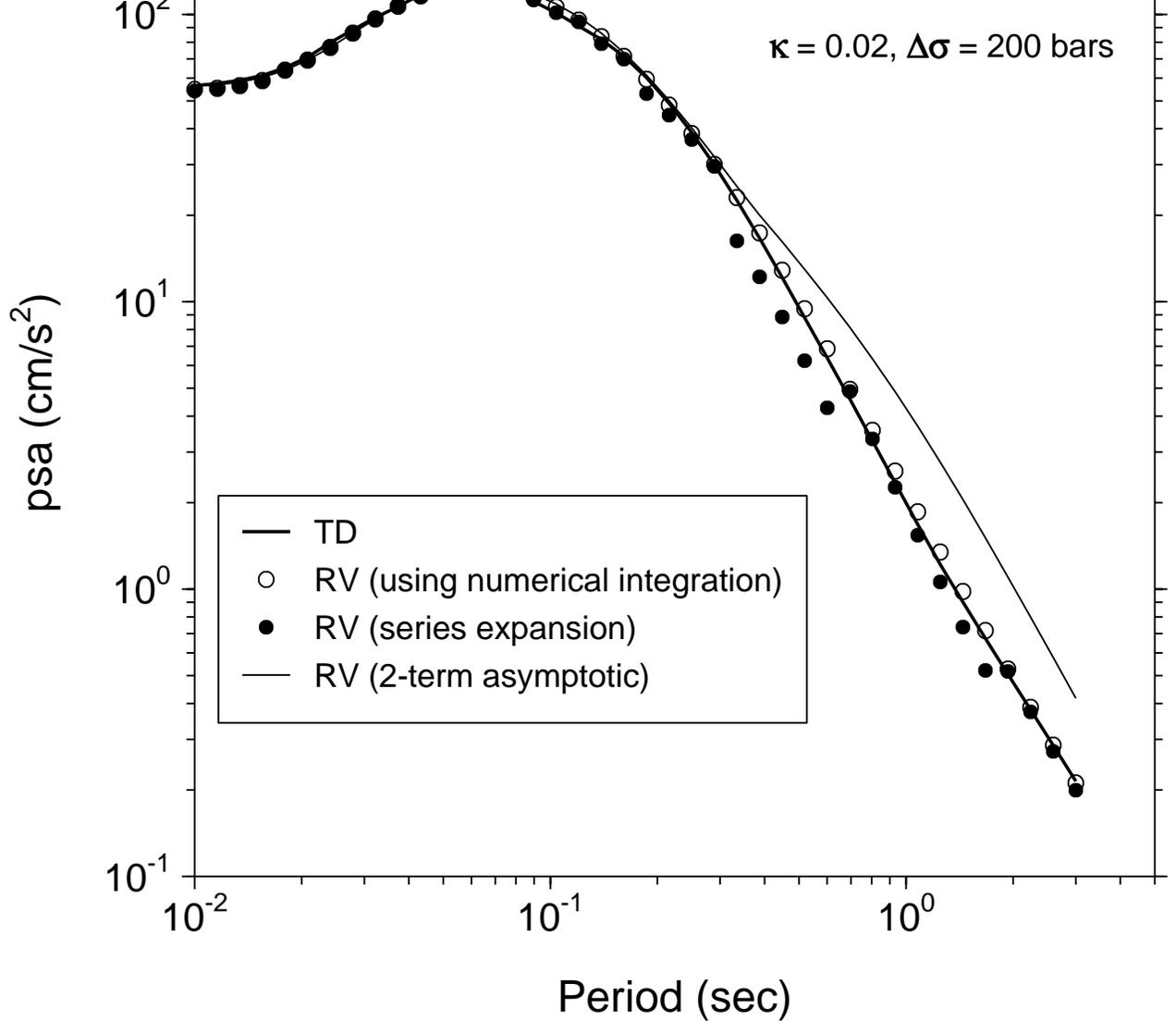
- Boore, D.M. (1984). Use of seismoscope records to determine M_L and peak velocities, *Bull. Seism. Soc. Am.* **74**, 315–324.
- Boore, D.M. (1986). Short-period P - and S -wave radiation from large earthquakes: implications for spectral scaling relations, *Bull. Seism. Soc. Am.* **76**, 43–64.
- Boore, D.M. (1996). SMSIM – Fortran programs for simulating ground motions from earthquakes: version 1.0, *U.S. Geol. Surv. Open-File Rept. 96-80-A and 96-80-B*, 73 pp.
- Boore, D.M. (2003). Prediction of ground motion using the stochastic method, *Pure and Applied Geophy.* **160**, 635–676.
- Boore, D.M. and G.M. Atkinson (1987). Stochastic prediction of ground motion and spectral response parameters at hard-rock sites in eastern North America, *Bull. Seism. Soc. Am.* **77**, 440–467.
- Boore, D.M. and J. Boatwright (1984). Average body-wave radiation coefficients, *Bull. Seism. Soc. Am.* **74**, 1615–1621.
- Boore, D.M. and W.B. Joyner (1984). A note on the use of random vibration theory to predict peak amplitudes of transient signals, *Bull. Seism. Soc. Am.* **74**, 2035–2039.
- Boore, D.M. and W.B. Joyner (1991). Estimation of ground motion at deep-soil sites in eastern North America, *Bull. Seism. Soc. Am.* **81**, 2167–2185.
- Boore, D.M. and W.B. Joyner (1997). Site-amplifications for generic rock sites, *Bull. Seism. Soc. Am.* **87**, 327–341.
- Boore, D.M., W.B. Joyner, and L. Wennerberg (1992). Fitting the Stochastic ω^{-2} Source Model to Observed Response Spectra in Western North America: Trade-offs Between $\Delta\sigma$ and κ , *Bull. Seism. Soc. Am.* **82**, 1956–1963.
- Boore, D.M., W.B. Joyner, and T.E. Fumal (1997). Equations for estimating horizontal response spectra and peak acceleration from western North American earthquakes: A summary of recent work, *Seism. Res. Lett.* **68**, 128–153.
- Cartwright, D.E. and M.S. Longuet-Higgins (1956). The statistical distribution of the maxima of a random function, *Proc. R. Soc. London* **237**, 212–232.

- Converse, A.M. (1992). BAP — Basic strong-motion accelerogram processing software; Version 1.0, *U.S. Geol. Surv. Open-File Rept. 92-296A*, .
- Frankel, A., C. Mueller, T. Barnhard, D. Perkins, E. Leyendecker, N. Dickman, S. Hanson and M. Hopper (1996). National seismic hazard maps: Documentation June 1996. *U.S. Geol. Surv. Open-File Rept. 96-532*, 69 pp.
- Hanks, T.C. and R.K. McGuire (1981). The character of high-frequency strong ground motion, *Bull. Seism. Soc. Am.* **71**, 2071–2095.
- Herrmann, R.B. (1996). *Computer Programs in Seismology*, Dept. of Earth and Atmospheric Sciences, St. Louis University, St. Louis, Missouri.
- Joyner, W.B. (1984). A scaling law for the spectra of large earthquakes, *Bull. Seism. Soc. Am.* **74**, 1167–1188.
- Joyner, W.B. and D.M. Boore (1988). Measurement, characterization, and prediction of strong ground motion, in *Earthquake Engineering and Soil Dynamics II, Proc. Am. Soc. Civil Eng. Geotech. Eng. Div. Specialty Conf.*, June 27–30, 1988, Park City, Utah, 43–102.
- Liu, L. and S. Pezeshk (1999). An improvement on the estimation of pseudoresponse spectral velocity using RVT method, *Bull. Seism. Soc. Am.* **89**, 1384–1389.
- Motazedian, D. and G. M. Atkinson (2005). Stochastic finite-fault modeling based on a dynamic corner frequency, *Bull. Seism. Soc. Am.* **95**, 995-1010.
- Press, W.H., S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery (1992). *Numerical Recipes in FORTRAN: The Art of Scientific Computing*, Cambridge University Press, Cambridge, England, 963 pp.
- Scherbaum, F., F. Cotton, and H. Staedtke (2006). The estimation of stochastic host region models for host-to-target-region conversions, *Bull. Seism. Soc. Am.* **96**, (in press).
- Silva, W.J. and Lee, K. (1987). WES RASCAL code for synthesizing earthquake ground motions, *State-of-the-Art for Assessing Earthquake Hazards in the United States, Report 24*, U.S. Army Engineers Waterways Experiment Station, *Misc. Paper S-73-1*.
- Silva, W., N. Gregor, and R. Darragh (2002). Development of regional of hard rock

attenuation relations for central and eastern North America, unpublished manuscript available from <http://www.pacificengineering.org>.

Toro, G.R. (1985). Stochastic model estimates of strong ground motion, Section 3 of *Seismic Hazard Methodology for Nuclear Facilities in the Eastern United States*, Report Prepared for EPRI, Project Number P101-29.

Uhrhammer, R.A. and E.R. Collins (1990). Synthesis of Wood-Anderson seismograms from broadband digital records, *Bull. Seism. Soc. Am.* **80**, 702–716.



Jun 7, 2000 5:08:21 pm
D:\SMSIMRVOLDNEW.GRA
D:\SMSIMRVOLDNEW.DT

Figure 1. Response spectra computed with time-domain simulations and random-vibration simulations with various relations between the peak and rms values. The series expansion uses equation (21) in Boore (1983), and for the model parameters used for this figure, the series expansion produces response spectra (solid dots) that are a discontinuous function of period. The 2-term asymptotic expansion uses equation (24) in Boore (1983), and the random-vibration results computed with numerical integration use equation (1) in this report.

```

Sample data file **** NOT FOR A PARTICULAR APPLICATION **
rho, beta, prttn, radpat, fs:
  2.8 3.6 0.71 0.55 2.0
spectral shape: source number (1=1-corner;2=Joyner;3=A93;...;8=Jena;9=AS00;10=A05),
  pf, pd (1-corner spectrum = 1/(1+(f/fc)**pf)**pd; 0.0 otherwise)
  (usual model: pf=2.0,pd=1.0; Butterworth: pf=4.0,pd=0.5)
  (Note: power of high freq decay --> pf*pd)
  1 2.0 1.0
spectral scaling: stressc, dlsgm, fbdfa, amagc
  (stress=stressc*10.0**(dlsgm*(amag-amagc))
  (fbdfa, amagc for Joyner model, usually 4.0, 7.0)
  (not used for source 3, but placeholders still needed)
  80.0 0.0 4.0 7.0
gsprd: r_ref, nsegs, (r_low(i), a_s, b_s, m_s(i)) (Usually set r_ref = 1.0 km)
  1.0
  3
  1.0 -1.0 0.0 6.5
  70.0 0.0 0.0 6.5
  130.0 -0.5 0.0 6.5
q: frl, Qr1, sl, ft1, ft2, fr2, qr2, s2, c_q
  0.1 275 -2.0 0.2 0.6 1.0 88.0 0.9 3.6
source duration: weights of 1/fa, 1/fb
  1.0 0.0
path duration: nknots, (rdur(i), dur(i)), slope of last segment
  4
  0.0 0.0
  10.0 0.0
  70.0 9.6
  130.0 7.8
  0.04
site amplification: namps, (famp(i), amp(i))
  5
  0.1 1.0
  1.0 1.5
  2.0 2.0
  5.0 2.5
  10.0 3.0
site diminution parameters: fm, akappa, dkappadmag, amagkref
  25.0 0.03 0.0 6.0
low-cut filter parameters: fcut, norder
  0.0 2
rv params: zup, eps_int (int acc), amp_cutoff (for fup), osc_crrctn(1=b&j;2=l&p)
  10.0 0.00001 0.001 1
window params: idxwnd(0=box,1=exp), tapr(<1), eps_w, eta_w, f_tb2te, f_te_xtnd
  1 0.05 0.2 0.05 2.0 1.0
timing stuff: dur_fctr, dt, tshift, seed, nsims, iran_type (0=normal;1=uniform)
  1.3 0.005 20.0 640.0 640 0

```

Figure 2. Sample input file for the SMSIM programs. The file has been constructed for illustrative purposes and does not correspond to a real application.

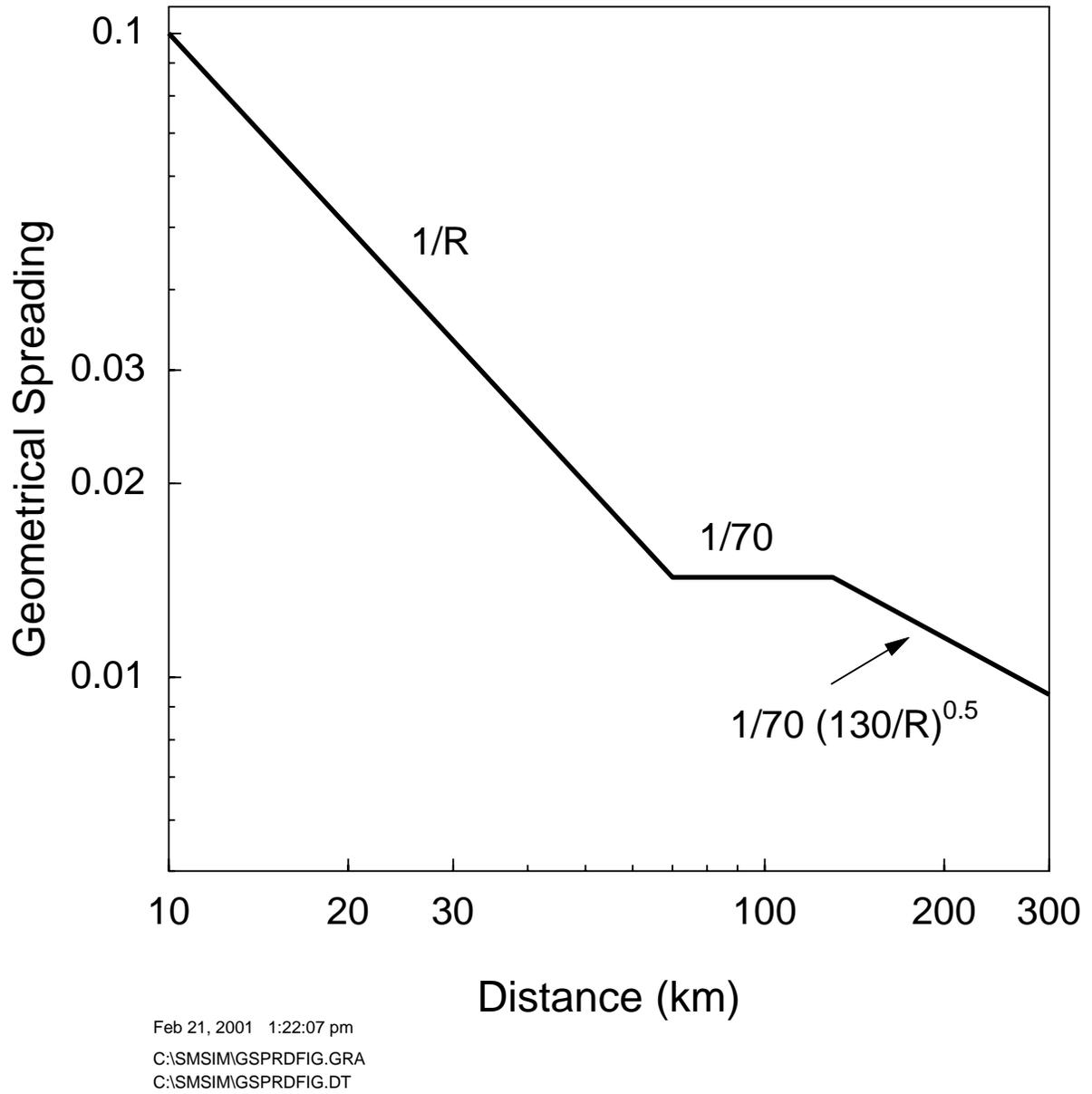
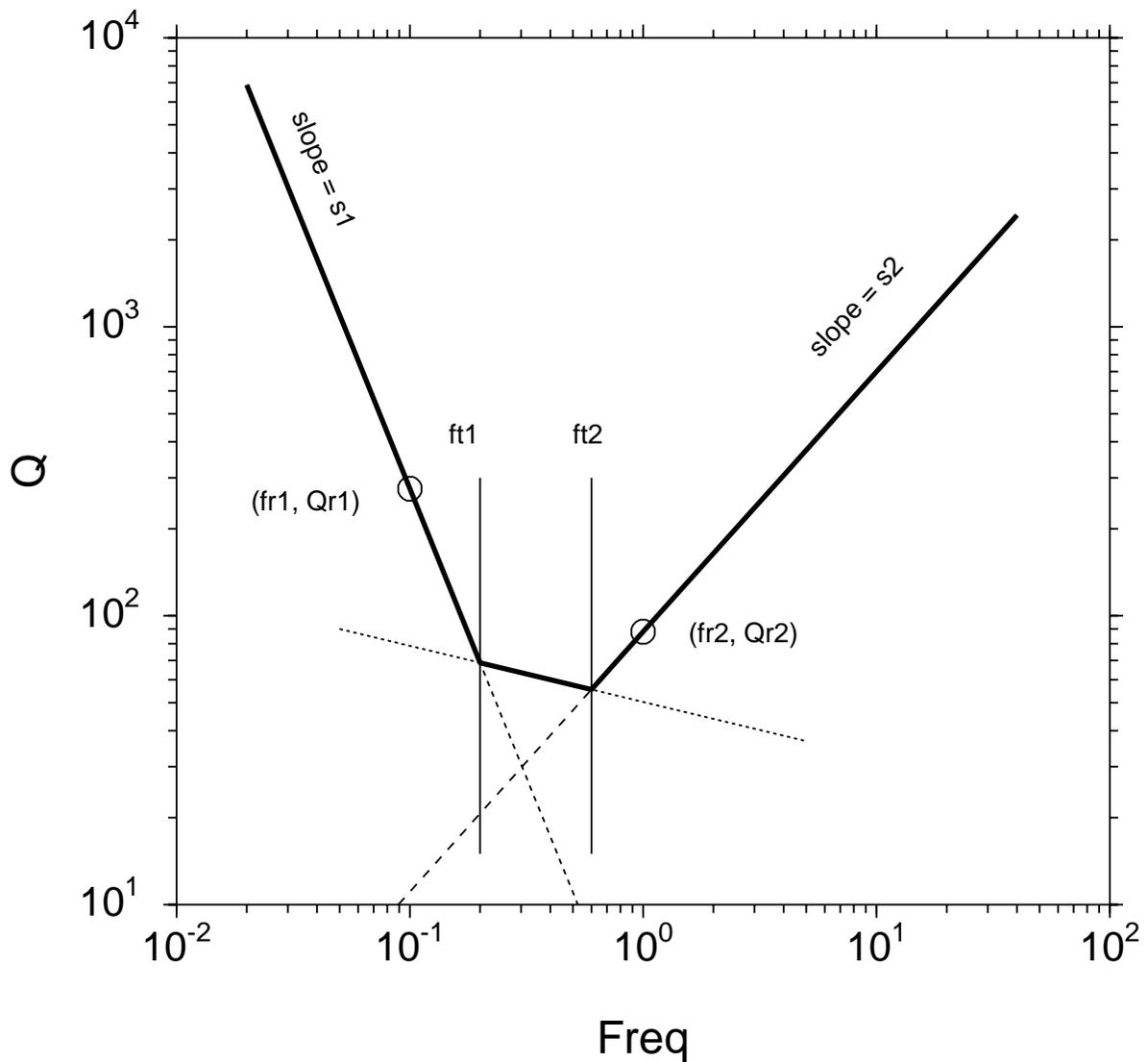
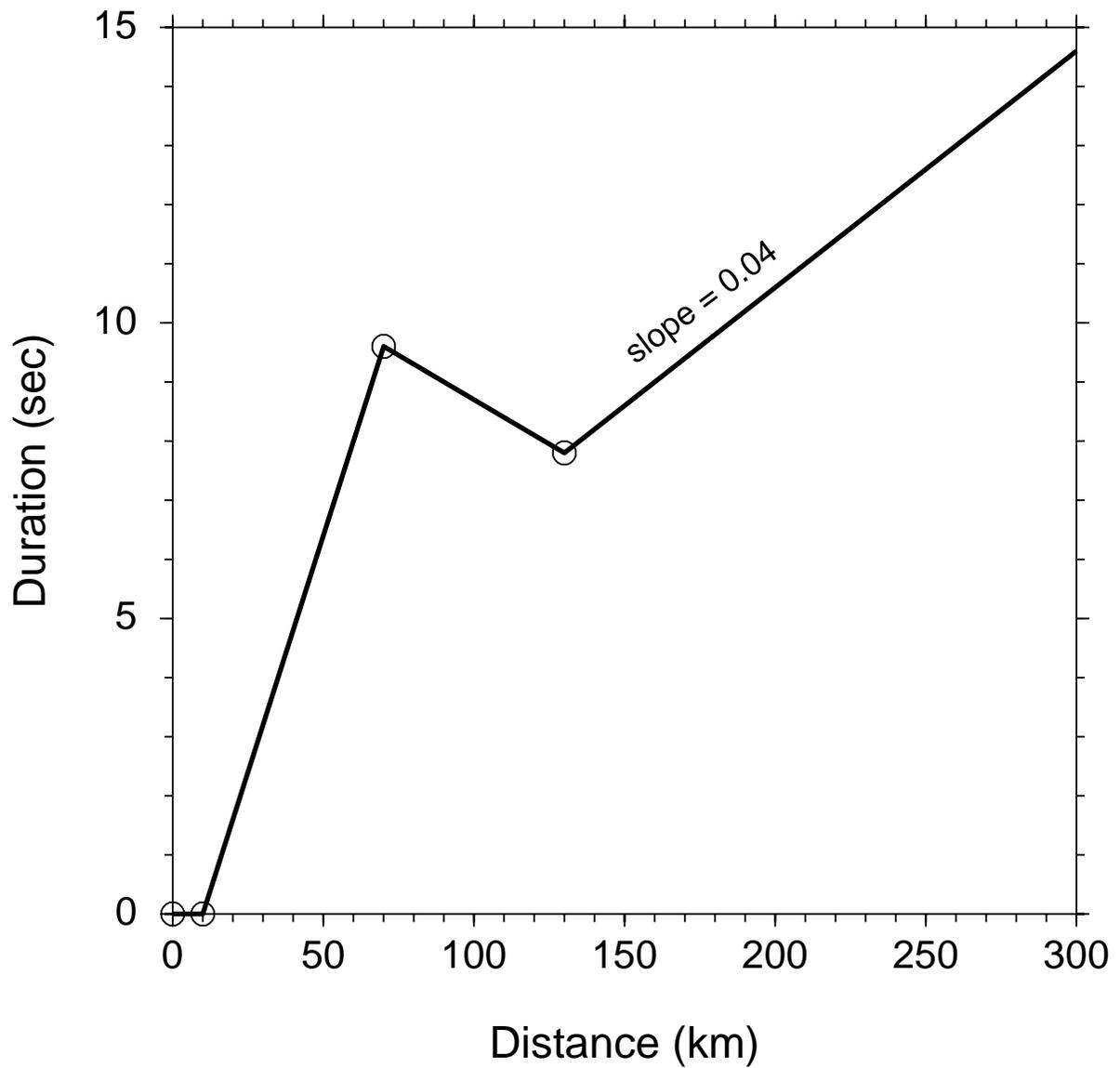


Figure 3. The geometrical spreading function for the sample input file.



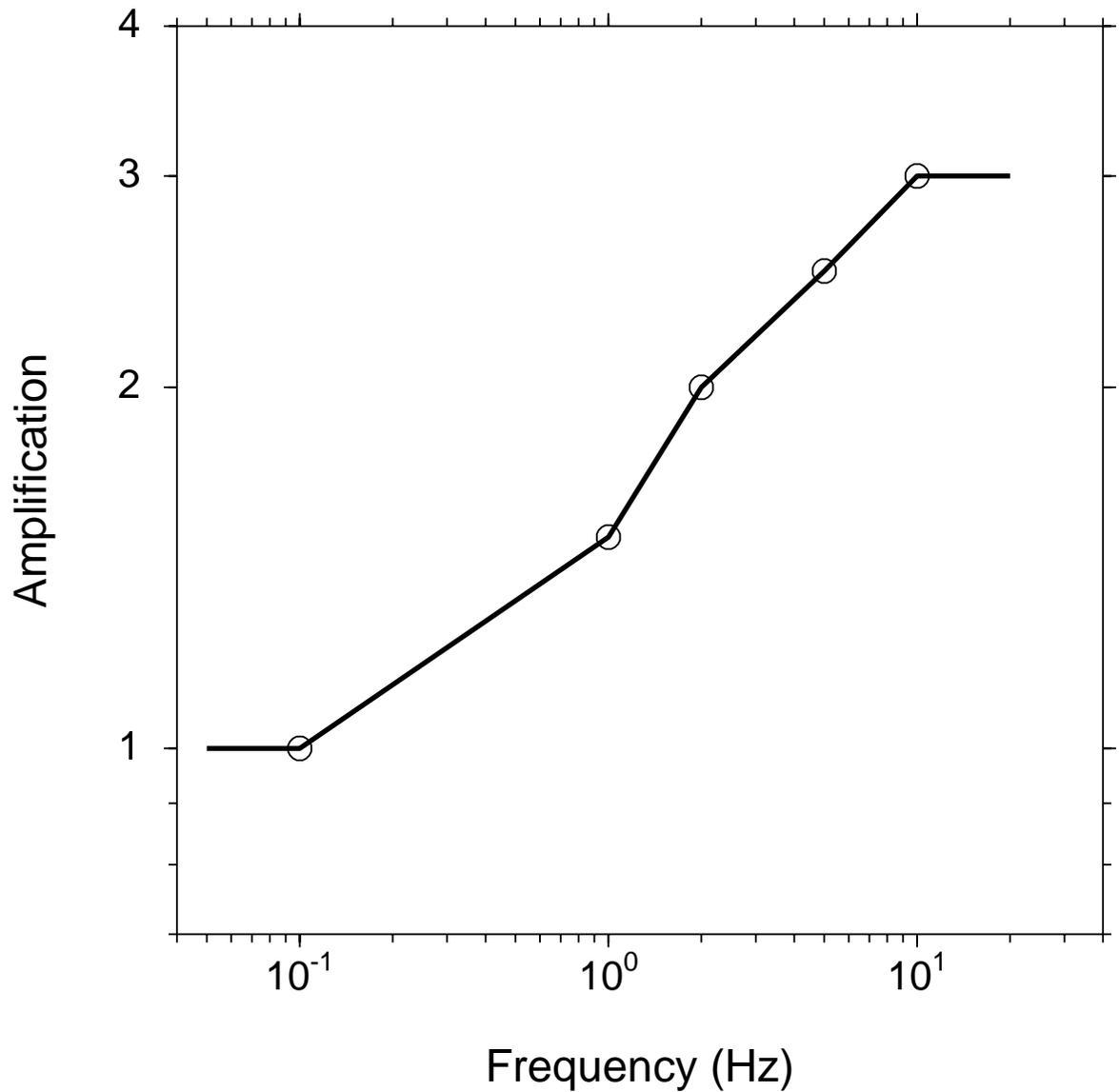
Mar 16, 2000 11:56:40 am
 D:\SMSIMQ_FIG.GRA
 D:\SMSIMQ_FIG.DT

Figure 4. Illustration of the specification of $Q(f)$: it is made up of three lines in log-log space. The lines shown are those for the parameters in the sample input file, which is an approximation of the $Q(f)$ function in Boore (1984).



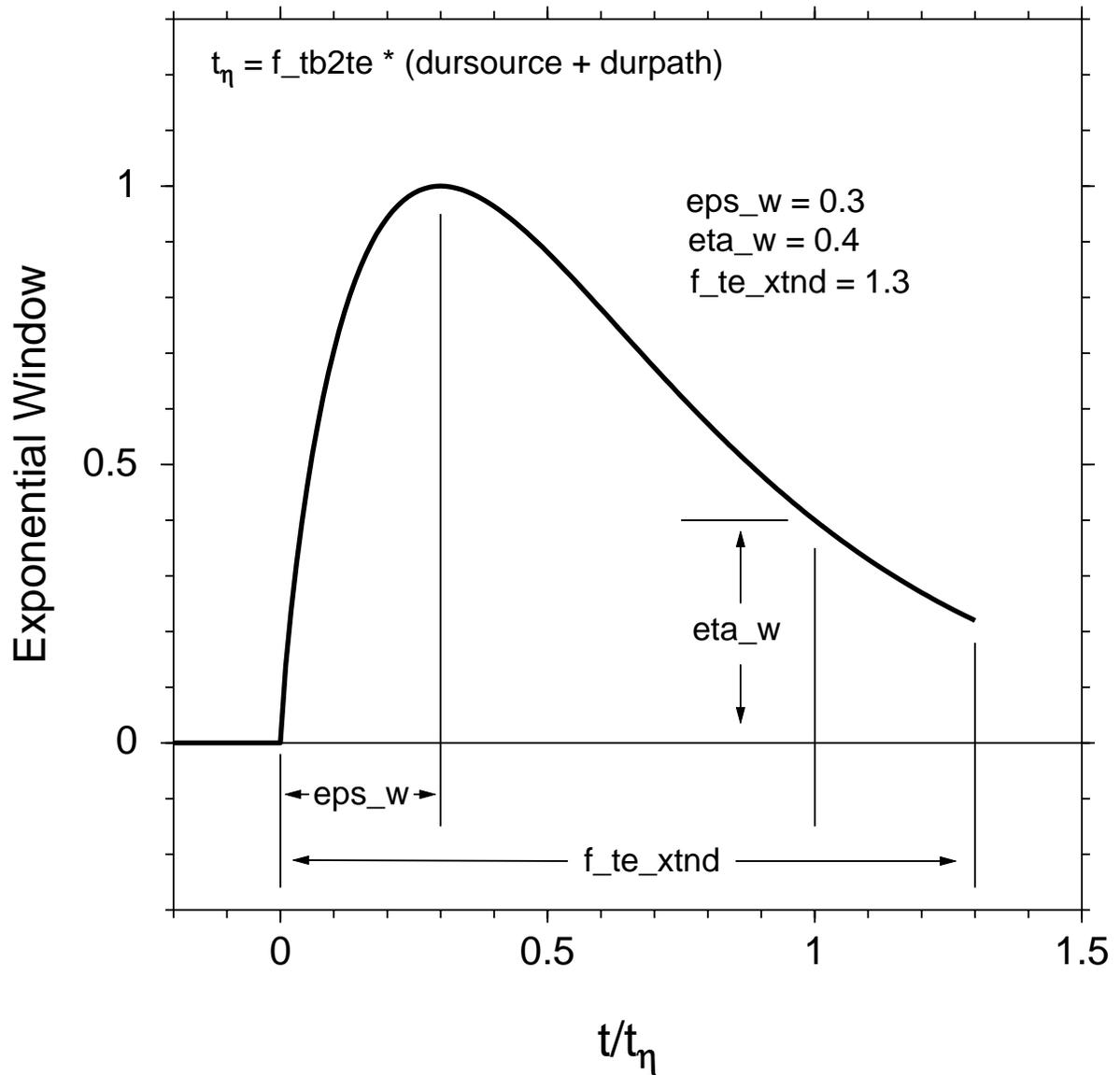
Mar 16, 2000 11:57:24 am
 D:\SMSIM\DURP_FIG.GRA
 D:\SMSIM\DURP_FIG.DT

Figure 5. The duration due to the path is made up of a series of straight lines specified by distance–duration pairs (circles) and the slope of the last line. The lines shown are those for the parameters in the sample input file, and correspond to the duration in Atkinson and Boore (1995).



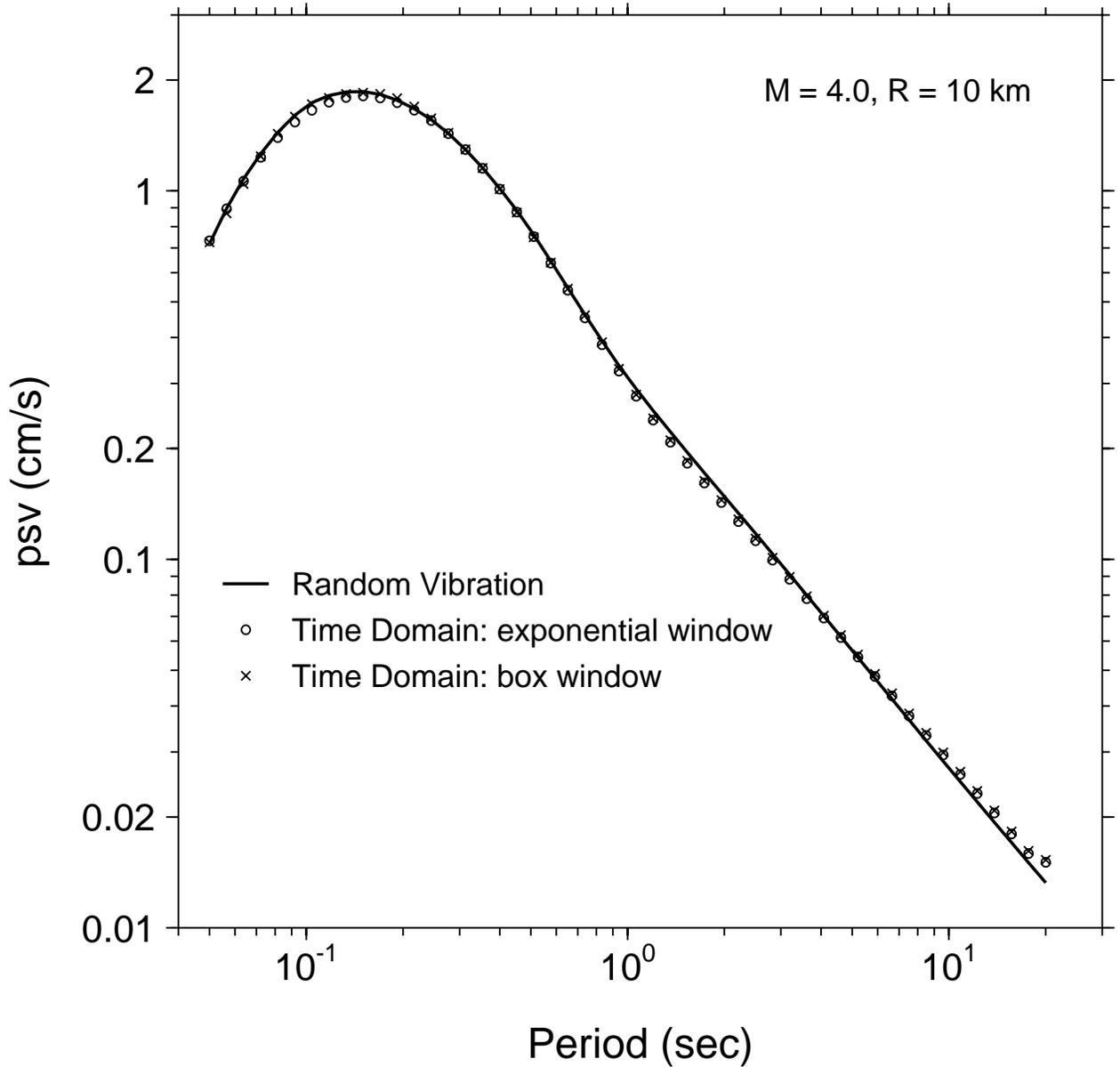
Mar 16, 2000 11:58:29 am
 D:\SMSIM\SAMP_FIG.GRA
 D:\SMSIM\SAMP_FIG.DT

Figure 6. The site-amplification is specified by a series of straight lines in log frequency, log amplification space. The lines shown are those for the parameters in the sample input file, and are made up; **THEY DO NOT CORRESPOND TO ANY OF MY PUBLISHED APPLICATIONS.**



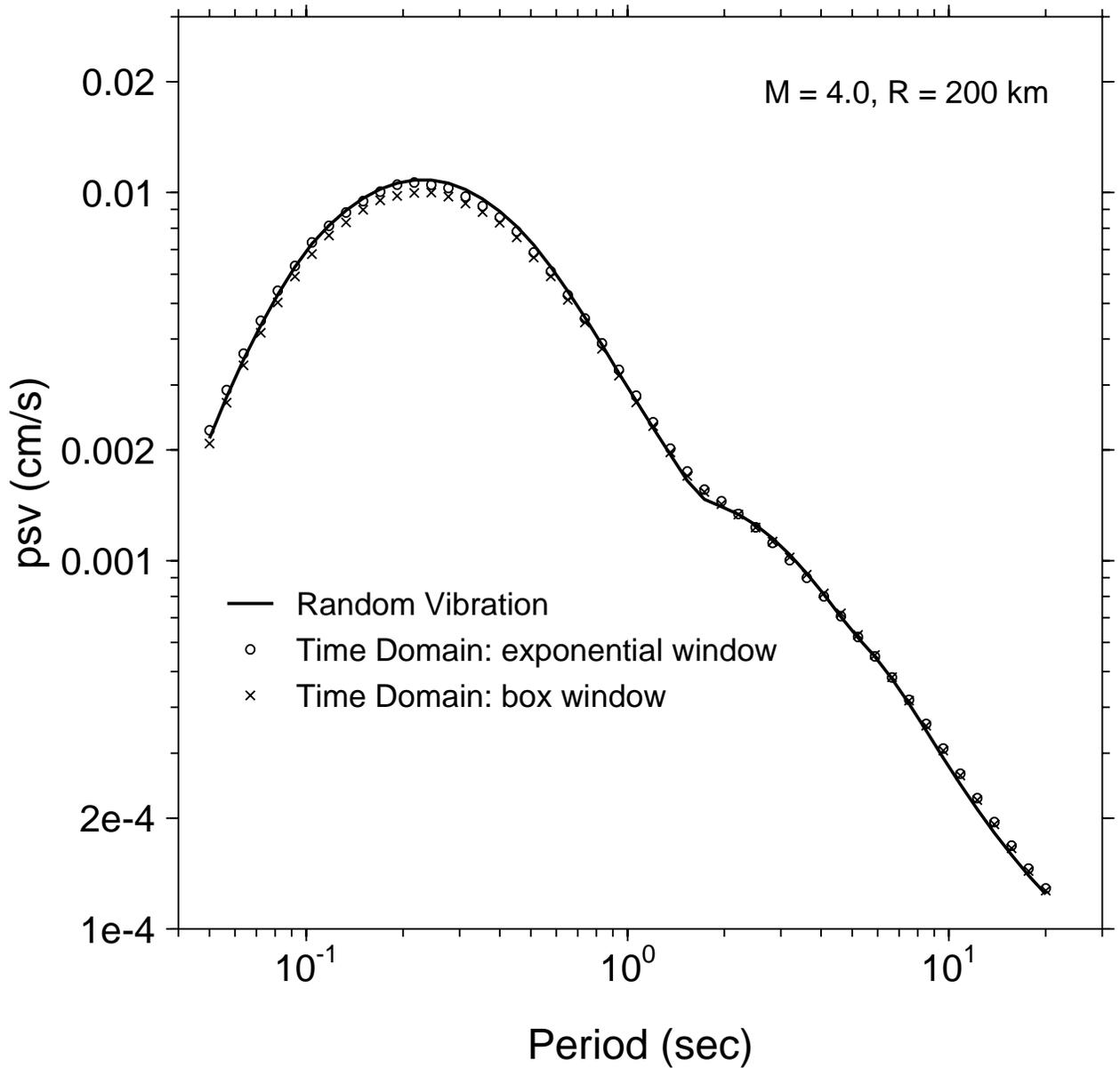
Feb 13, 2001 9:36:12 am
 C:\SMSIM\WEXP_FIG.GRA
 C:\SMSIM\WEXP_FIG.DT

Figure 7. The exponential window is specified by parameters whose meaning is shown here. The parameters have been chosen to illustrate their meaning and are **NOT THOSE IN THE INPUT FILE**.



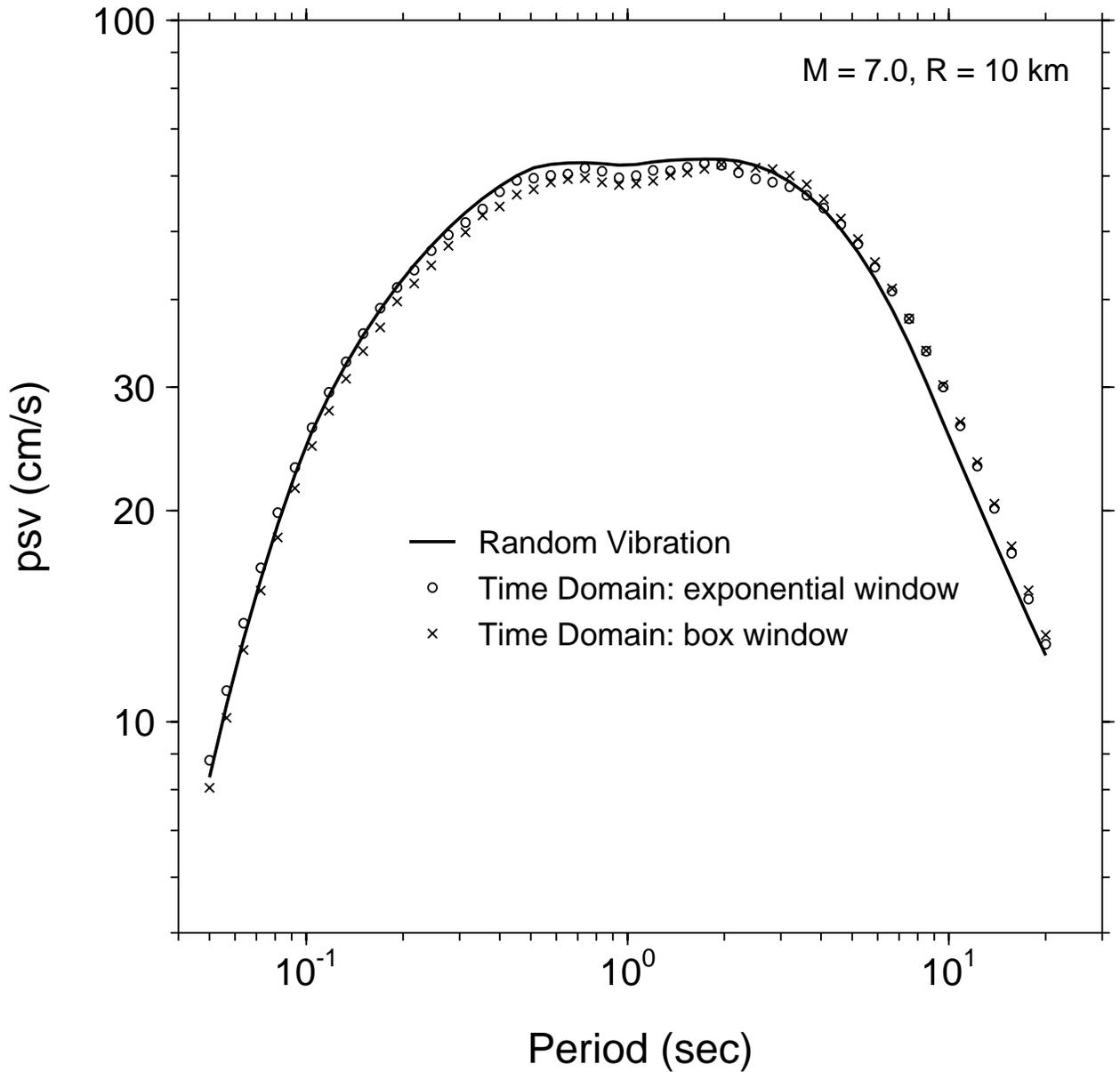
Feb 13, 2001 9:37:47 am
 C:\SMSIM\OFREBX40.GRA
 C:\SMSIMM40EXPBX.DT

Figure 8. Comparison of simulations using box and exponential windows, with the random-vibration calculations for magnitude 4 at 10 km, using the parameters in the input-parameter file (except for the time-domain simulations, where *indxwind* = 0 for the box window).



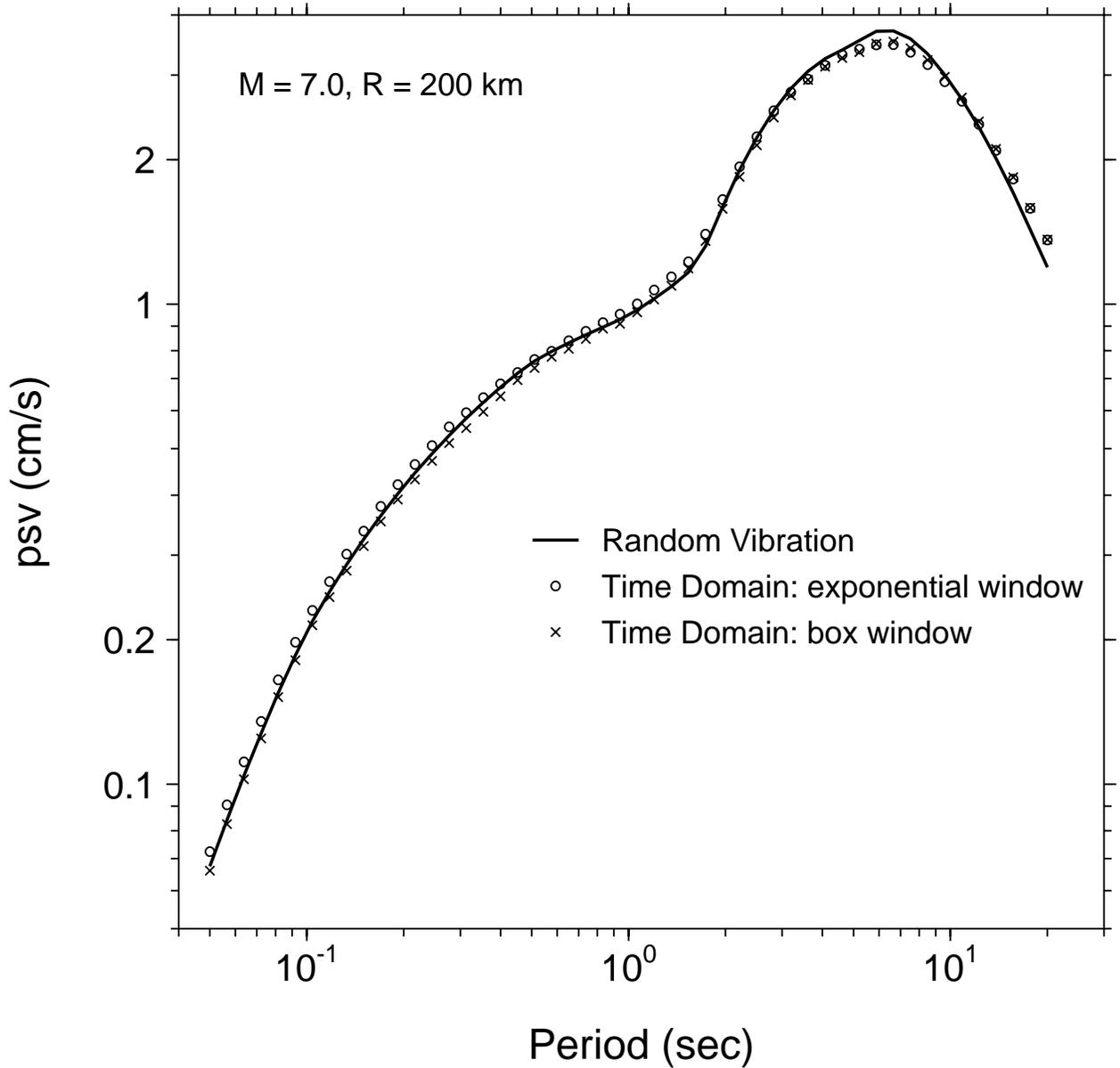
Feb 13, 2001 9:36:35 am
 C:\SMSIMM4R200.GRA
 C:\SMSIMM4R200.DT

Figure 9. Comparison of simulations using box and exponential windows, with the random-vibration calculations for magnitude 4 at 200 km, using the parameters in the input-parameter file (except for the time-domain simulations, where *indxwind* = 0 for the box window).



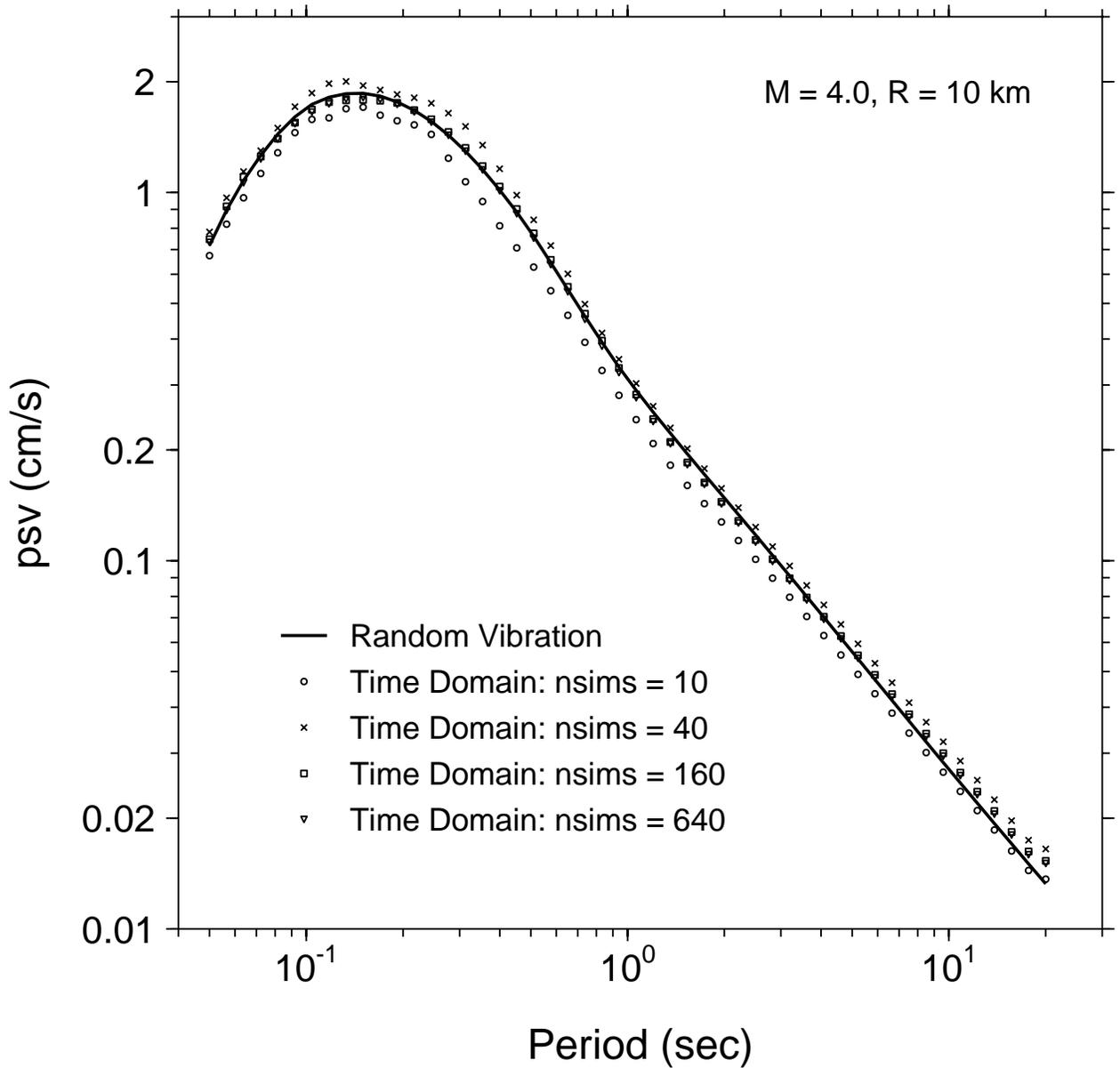
Feb 13, 2001 9:38:04 am
 C:\SMSIM\OFREBX70.GRA
 C:\SMSIMM70EXPBX.DT

Figure 10. Comparison of simulations using box and exponential windows, with the random-vibration calculations for magnitude 7 at 10 km, using the parameters in the input-parameter file (except for the time-domain simulations, where *indxwind* = 0 for the box window).



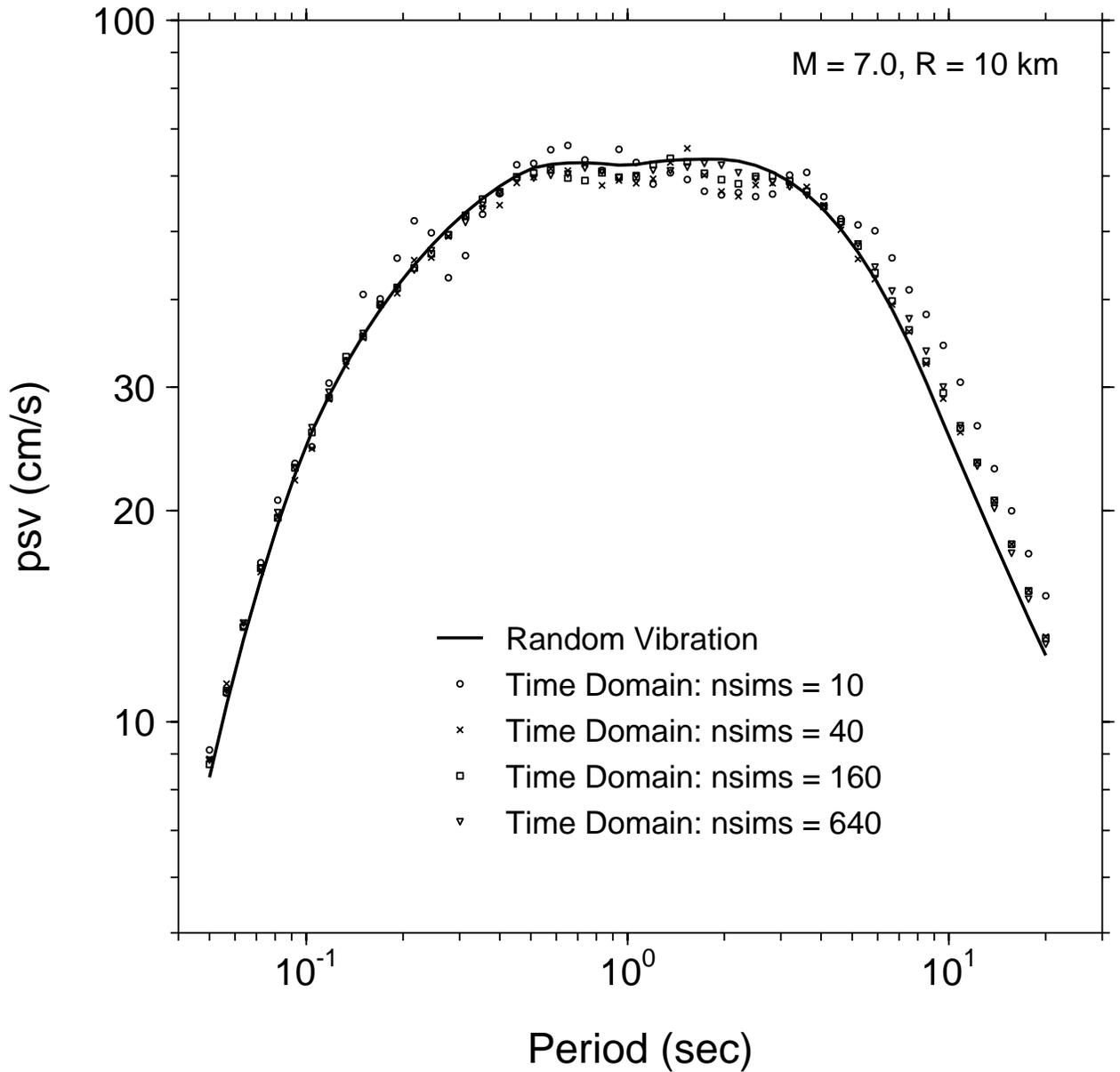
Feb 13, 2001 9:36:49 am
 C:\SMSIMM7R200.GRA
 C:\SMSIMM7R200.DT

Figure 11. Comparison of simulations using box and exponential windows, with the random-vibration calculations for magnitude 7 at 200 km, using the parameters in the input-parameter file (except for the time-domain simulations, where *indxwind* = 0 for the box window).



Oct 11, 2000 5:09:03 pm
 D:\SMSIM\OFRNRM40.GRA
 D:\SMSIMM40NRUNS.DT

Figure 12. Comparison of simulations using the time-domain calculations with various values for *nsims*, with *seed* = *nsims* for each suite of realizations. The random-vibration results are shown for comparison. The calculations are for magnitude 4 at 10 km, using the parameters in the input-parameter file.



Oct 11, 2000 5:10:06 pm
D:\SMSIM\OFRNRM70.GRA
D:\SMSIMM70NRUNS.DT

Figure 13. Comparison of simulations using the time-domain calculations with various values for *nsims*, with *seed = nsims* for each suite of realizations. The random-vibration results are shown for comparison. The calculations are for magnitude 7 at 10 km, using the parameters in the input-parameter file.

```

***** SMSIM, Version 2.300 *****

output file: sample_rv_sum.sum
*** Results computed using GM_RV_DRVR ***
Date: 08/17/05
Time: 11:44:36.76
file with parameters: ofr.dat
Title:
  Sample data file **** NOT FOR A PARTICULAR APPLICATION **
rho, beta, prttn, rtp, fs:
  2.80 3.60 0.710 0.55 2.00
spectral shape: source number (1=Single Corner;2=Joyner;3=A93;...;8=Jena;9=AS00;10=A05)
pf, pd (1-corner spectrum = 1/(1+(f/fc)**pf)**pd; 0.0 otherwise)
(usual model: pf=2.0,pd=1.0; Butterworth: pf=4.0,pd=0.5)
(Note: power of high freq decay --> pf*pd)
  1 2.00 1.00
spectral scaling: stresssc, dlsdm, fbdfa, amagc
(stress=stresssc*10.0**(dlsdm*(amag-amagc))
(fbdfa, amagc for Joyner model, usually 4.0, 7.0)
(not used for srce 3, but placeholders still needed)
  80.00 0.000E+00 4.00 7.00
gsprd: r_ref, nsegs, (rlow(i), a_s, b_s, m_s(i)) (Usually set r_ref = 1.0 km)
  1.00
  3
  1.00 -0.100E+01 0.000E+00 6.50
  70.00 0.000E+00 0.000E+00 6.50
  130.00 -0.500E+00 0.000E+00 6.50
q: frl, Qr1, sl, ft1, ft2, fr2, qr2, s2, c_q
  0.100 275.00 -2.000 0.200 0.600 1.000 88.00 0.900 3.60
source duration: weights of 1/fa, 1/fb
  1.00 0.00
path duration: nknots, (rdur(i), dur(i), slope of last segment
  4
  0.0 0.00
  10.0 0.00
  70.0 9.60
  130.0 7.80
  4.000E-02
site amplification: namps, (famp(i), amp(i))
  5
  0.100 1.000
  1.000 1.500
  2.000 2.000
  5.000 2.500
  10.000 3.000
site diminution parameters: fm, akappa, dkappadmag, amagkref
  25.000 3.000E-02 0.000E+00 6.00
low-cut filter parameters: fcut, norder
  0.000 2
parameters for rv calcs: zup, eps_int, amp_cutoff, osc_crrctn
  10.00 1.000E-05 1.000E-03 1
calculated fup = 7.329E+01
fup calculated in driver = 7.329E+01

***** NEW R AND M *****
r, amag, kappa = 2.000E+02 7.000E+00 3.000E-02
Time Start: 11:45:46.39
Column file: sample_rv_col.col
const= 4.757E-24
amag, stress, fa, fb, durex= 7.000 8.00E+01 1.075E-01 1.075E-01 1.99E+01
am0, am0b_m0fa= 3.548E+26 0.000E+00
          pga(cm/s2) domfreq eps nx nz pk_rms
          5.75E+00 6.12 0.8913 537.59 243.71 3.47
          pgv(cm/s) domfreq eps nx nz pk_rms
          1.96E+00 0.33 0.9985 243.70 13.23 2.47
          pgd(cm) domfreq eps nx nz pk_rms
          2.90E+00 0.07 0.9749 13.23 2.95 1.70
          Arias intensity (cm/s)
          8.76E-02
Fractional oscillator damping = 0.050
per(s) freq psv(cm/s) psa(cm/s2) domfreq eps nx nz pk_rms zupsd
0.020 50.000 1.87E-02 5.87E+00 6.56 0.8954 586.7 261.2 3.49 4.80
0.023 42.621 2.22E-02 5.93E+00 6.82 0.9012 626.2 271.4 3.50 4.80

```

Figure 14. Sample summary file from the random-vibration program GM_RV_DRVR. The time-domain summary output is similar, except that it does not include estimates of dominant frequency.

per	freq	sd:sample_r	psv:sample_r	psa:sample_r	psa/pgs:sample_r	domfreqpsv
0.0200	50.000	5.948E-05	1.868E-02	5.870E+00	1.021	6.5616
0.0235	42.621	8.273E-05	2.215E-02	5.933E+00	1.032	6.8180
0.0275	36.331	1.161E-04	2.649E-02	6.048E+00	1.052	7.3731
0.0323	30.969	1.662E-04	3.233E-02	6.292E+00	1.094	8.6393
0.0379	26.399	2.473E-04	4.102E-02	6.804E+00	1.184	10.7683
0.0444	22.503	3.839E-04	5.428E-02	7.674E+00	1.335	12.5910
0.0521	19.182	6.041E-04	7.281E-02	8.776E+00	1.527	13.0552
0.0612	16.351	9.413E-04	9.671E-02	9.935E+00	1.728	12.4764
0.0717	13.938	1.444E-03	1.264E-01	1.107E+01	1.926	11.4217
0.0842	11.881	2.175E-03	1.624E-01	1.212E+01	2.109	10.2132
0.0987	10.128	3.208E-03	2.042E-01	1.299E+01	2.260	9.0027
0.1158	8.633	4.560E-03	2.474E-01	1.342E+01	2.334	7.8368
0.1359	7.359	6.344E-03	2.933E-01	1.356E+01	2.359	6.7748
0.1594	6.273	8.686E-03	3.423E-01	1.349E+01	2.347	5.8367
0.1870	5.347	1.173E-02	3.941E-01	1.324E+01	2.303	5.0160
0.2194	4.558	1.567E-02	4.487E-01	1.285E+01	2.235	4.3039
0.2574	3.885	2.072E-02	5.059E-01	1.235E+01	2.148	3.6869
0.3019	3.312	2.714E-02	5.647E-01	1.175E+01	2.044	3.1555
0.3542	2.823	3.523E-02	6.249E-01	1.109E+01	1.928	2.6981
0.4155	2.407	4.536E-02	6.858E-01	1.037E+01	1.804	2.3051
0.4875	2.051	5.784E-02	7.455E-01	9.610E+00	1.672	1.9680
0.5719	1.749	7.229E-02	7.943E-01	8.727E+00	1.518	1.6764
0.6709	1.491	8.947E-02	8.379E-01	7.848E+00	1.365	1.4239
0.7870	1.271	1.103E-01	8.803E-01	7.028E+00	1.223	1.2061
0.9233	1.083	1.357E-01	9.235E-01	6.285E+00	1.093	1.0184
1.0831	0.923	1.690E-01	9.802E-01	5.686E+00	0.989	0.8581
1.2706	0.787	2.131E-01	1.054E+00	5.211E+00	0.907	0.7230
1.4906	0.671	2.714E-01	1.144E+00	4.823E+00	0.839	0.6078
1.7487	0.572	3.720E-01	1.336E+00	4.802E+00	0.835	0.5158
2.0514	0.487	5.575E-01	1.707E+00	5.230E+00	0.910	0.4496
2.4066	0.416	8.119E-01	2.120E+00	5.534E+00	0.963	0.3913
2.8232	0.354	1.135E+00	2.525E+00	5.620E+00	0.978	0.3391
3.3120	0.302	1.522E+00	2.888E+00	5.479E+00	0.953	0.2927
3.8854	0.257	1.964E+00	3.176E+00	5.135E+00	0.893	0.2521
4.5581	0.219	2.446E+00	3.372E+00	4.648E+00	0.809	0.2166
5.3472	0.187	3.049E+00	3.583E+00	4.210E+00	0.732	0.1859
6.2730	0.159	3.717E+00	3.723E+00	3.729E+00	0.649	0.1602
7.3590	0.136	4.219E+00	3.603E+00	3.076E+00	0.535	0.1382
8.6331	0.116	4.510E+00	3.283E+00	2.389E+00	0.416	0.1194
10.1277	0.099	4.607E+00	2.858E+00	1.773E+00	0.308	0.1033
11.8812	0.084	4.586E+00	2.425E+00	1.283E+00	0.223	0.0895
13.9381	0.072	4.429E+00	1.997E+00	9.001E-01	0.157	0.0779
16.3512	0.061	4.173E+00	1.604E+00	6.162E-01	0.107	0.0683
19.1821	0.052	3.875E+00	1.269E+00	4.158E-01	0.072	0.0604
22.5030	0.044	3.588E+00	1.002E+00	2.797E-01	0.049	0.0541
26.3990	0.038	3.351E+00	7.976E-01	1.898E-01	0.033	0.0491
30.9694	0.032	3.184E+00	6.460E-01	1.311E-01	0.023	0.0455
36.3311	0.028	3.090E+00	5.345E-01	9.243E-02	0.016	0.0430
42.6211	0.023	3.062E+00	4.514E-01	6.655E-02	0.012	0.0415
50.0000	0.020	3.084E+00	3.876E-01	4.870E-02	0.008	0.0409

Figure 15. Sample column file from the random-vibration program GM_RV_DRVR, for $M = 7$ and $R = 200\text{km}$. The time-domain summary output is similar, except that it does not include the spectral shape (psa/pgs) or estimates of dominant frequency. Unlike random vibration output, however, the time domain output contains a column of the standard deviation of the spectral amplitudes, normalized by psv . This standard deviation is NOT the same as observed standard deviation of ground motion, but rather is a measure of the variation of the spectra from the individual time series simulations.

t_pgd	pgd	w*pgd	w2*pgd	t_pgv	pgv/w	pgv	w*pgv	t_pga	pga/w2	pga/w	pga
7.071	2.904E+00	2.581E+00	2.293E+00	0.376	1.172E-01	1.958E+00	3.271E+01	0.020	5.824E-05	1.830E-02	5.748E+00
50.000	2.904E+00	3.650E-01	4.586E-02	2.659	8.285E-01	1.958E+00	4.625E+00	0.141	2.912E-03	1.294E-01	5.748E+00

Figure 16. Sample column file with peak ground displacement, velocity, and acceleration, multiplied by various powers of radial frequency ($\omega = 2\pi f$). This file can be used to plot asymptote lines on a log-log “tripartite” plot for comparison with the response spectra. This output is from GM_RV_DRVR.

	T	A	V	D
0.000000	-2.6044E-04	0.0000E+00	0.0000E+00	
0.005000	-2.6091E-04	-1.3034E-06	-3.2575E-09	
0.010000	-2.6120E-04	-2.6087E-06	-1.3037E-08	
0.015000	-2.6120E-04	-3.9147E-06	-2.9345E-08	
0.020000	-2.6063E-04	-5.2193E-06	-5.2181E-08	
0.025000	-2.5989E-04	-6.5206E-06	-8.1532E-08	
0.030000	-2.5912E-04	-7.8181E-06	-1.1738E-07	
0.035000	-2.5776E-04	-9.1103E-06	-1.5970E-07	
0.040000	-2.5624E-04	-1.0395E-05	-2.0847E-07	
0.045000	-2.5461E-04	-1.1672E-05	-2.6364E-07	
0.050000	-2.5297E-04	-1.2941E-05	-3.2518E-07	
0.055000	-2.5105E-04	-1.4201E-05	-3.9304E-07	
0.060000	-2.4930E-04	-1.5452E-05	-4.6718E-07	
0.065000	-2.4761E-04	-1.6695E-05	-5.4755E-07	
0.070000	-2.4607E-04	-1.7929E-05	-6.3411E-07	
0.075000	-2.4446E-04	-1.9155E-05	-7.2683E-07	
0.080000	-2.4284E-04	-2.0373E-05	-8.2565E-07	
0.085000	-2.4137E-04	-2.1584E-05	-9.3055E-07	
0.090000	-2.3969E-04	-2.2787E-05	-1.0415E-06	
0.095000	-2.3816E-04	-2.3981E-05	-1.1584E-06	
0.100000	-2.3623E-04	-2.5167E-05	-1.2813E-06	
0.105000	-2.3425E-04	-2.6343E-05	-1.4101E-06	
0.110000	-2.3197E-04	-2.7509E-05	-1.5447E-06	
0.115000	-2.2929E-04	-2.8662E-05	-1.6851E-06	
0.120000	-2.2637E-04	-2.9801E-05	-1.8313E-06	
0.125000	-2.2323E-04	-3.0925E-05	-1.9831E-06	
0.130000	-2.1955E-04	-3.2032E-05	-2.1405E-06	
0.135000	-2.1598E-04	-3.3121E-05	-2.3034E-06	
0.140000	-2.1186E-04	-3.4191E-05	-2.4717E-06	
0.145000	-2.0772E-04	-3.5239E-05	-2.6453E-06	
0.150000	-2.0378E-04	-3.6268E-05	-2.8240E-06	
0.155000	-1.9976E-04	-3.7277E-05	-3.0079E-06	
0.160000	-1.9575E-04	-3.8266E-05	-3.1968E-06	

Figure 17. Sample of column file with acceleration and velocity time series, for $M = 7$ and $R = 200\text{km}$. This is simulation 1 of the time-domain calculations using the sample input-parameter file and program GM_TD_DRVR.

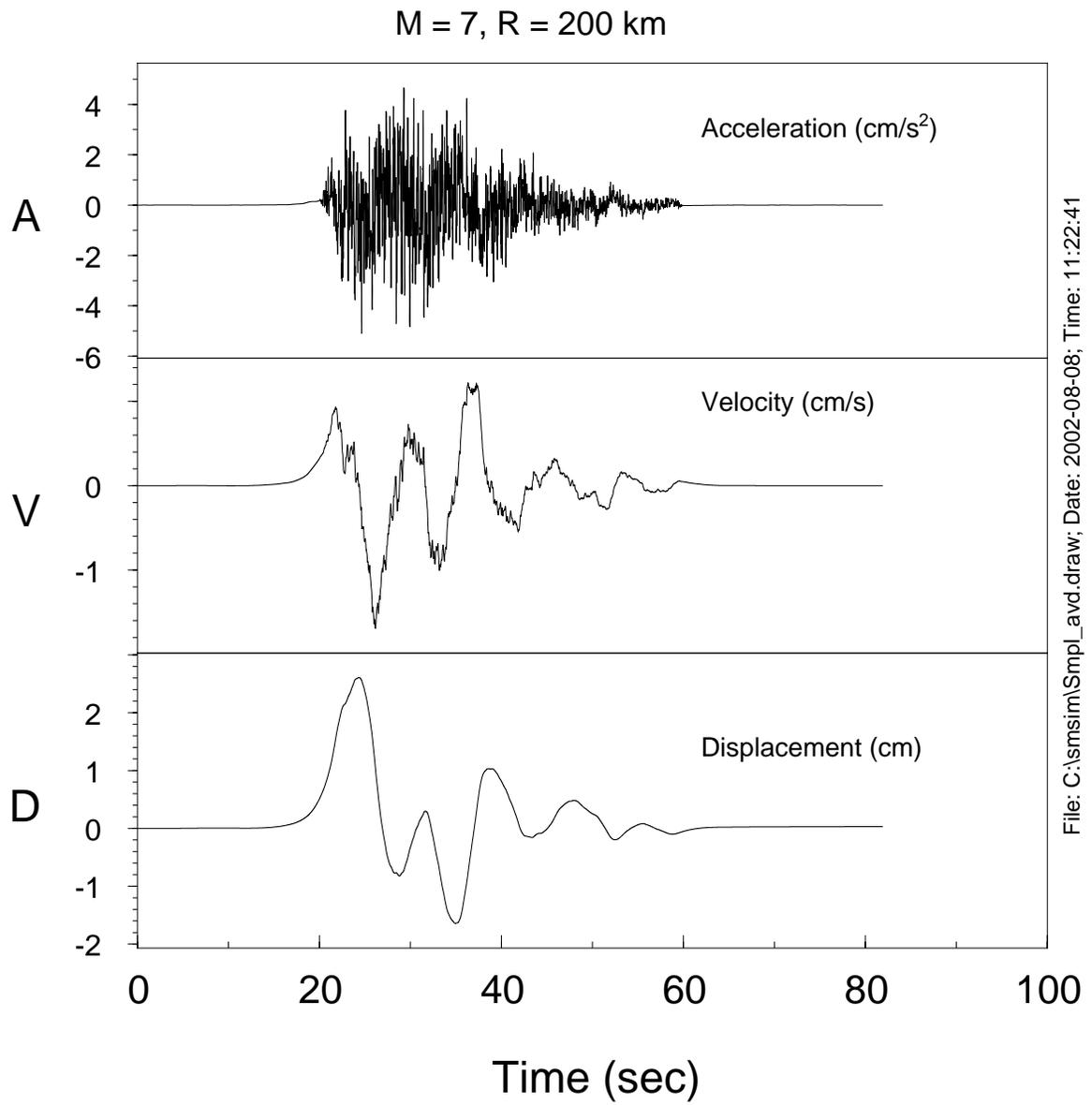
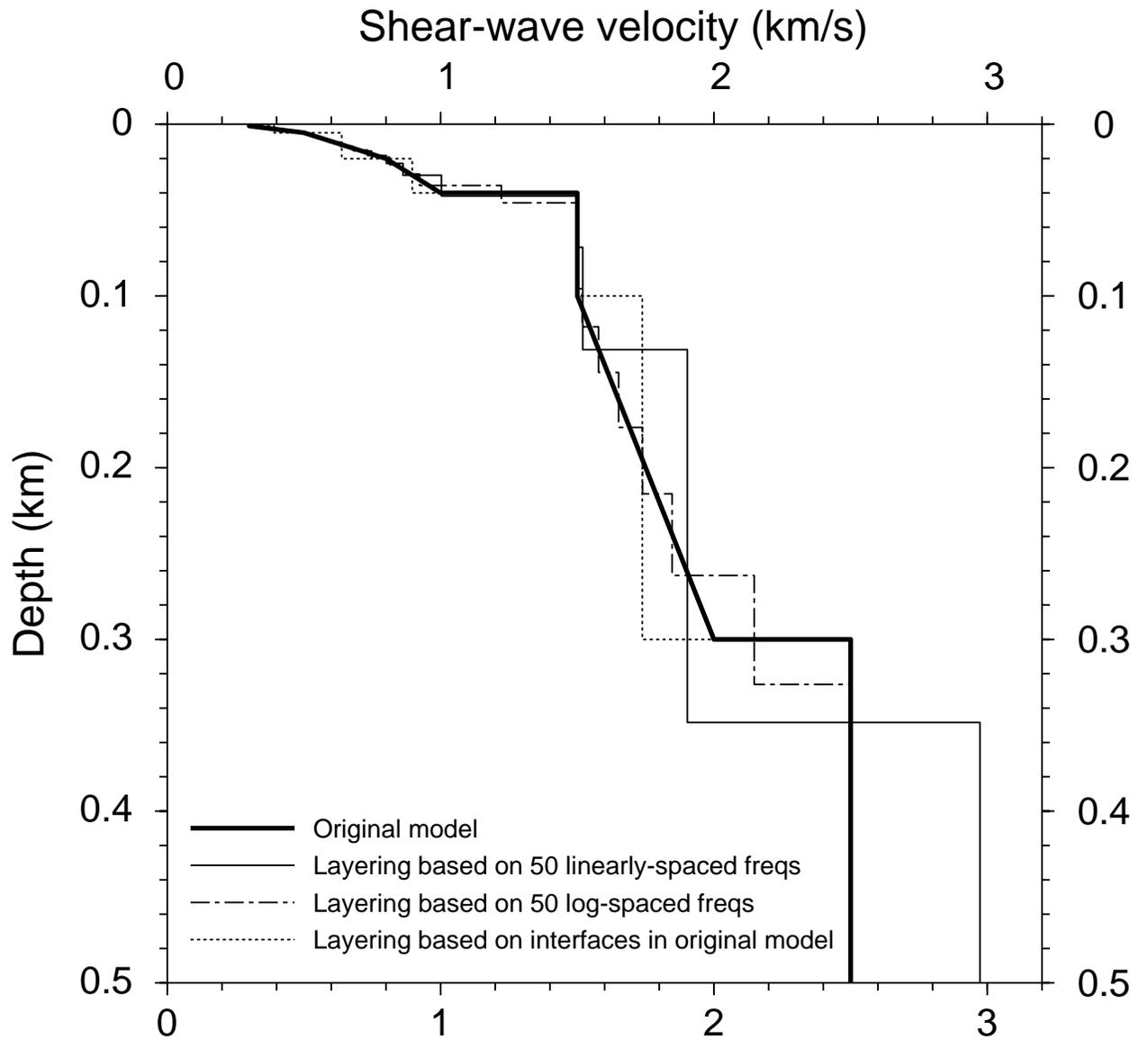


Figure 18. The time series from the column file partially shown in Figure 17.

Sample input file to site_amp

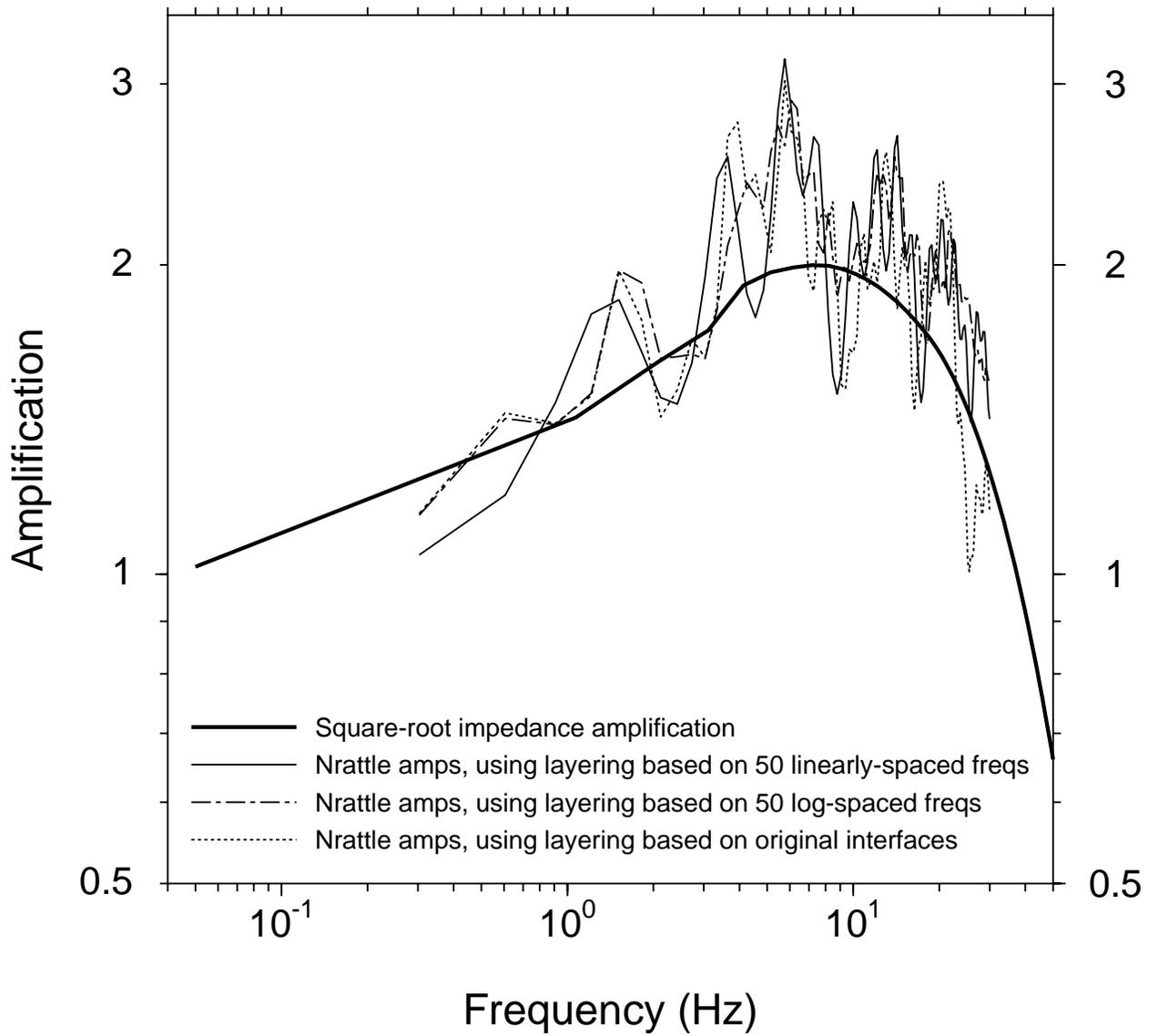
Depth	SVel	Dens	1/Q	
0	0.300	2.0	0.10	
0.001	0.300	2.0	0.10	! constant from 0.000 to 0.001
0.005	0.500	0.0	0.06	! linear gradient from 0.001 to 0.005
0.020	0.800	0.0	0.05	! linear gradient from 0.005 to 0.020
0.040	1.000	0.0	0.04	! linear gradient from 0.020 to 0.040
0.040	1.500	0.0	0.04	! step increase at 0.040
0.100	1.500	2.5	0.02	! constant vel from 0.040 to 0.100
0.300	2.000	0.0	0.02	! linear gradient from 0.100 to 0.300
0.300	2.500	0.0	0.02	! step increase at 0.300
0.500	2.500	2.6	0.01	! constant from 0.300 to 1.000,
1.000	2.500	2.6	0.001	! but density and 1/Q differ

Figure 19. Sample input-parameter file for the SITE_AMP program. **THE FILE HAS BEEN CONSTRUCTED FOR ILLUSTRATIVE PURPOSES AND DOES NOT CORRESPOND TO A REAL APPLICATION.** Note that the velocity model is made up of a combination of constant-velocity layers and velocity gradients. For the sake of illustration, the density has been assigned specific values for certain depths; these values override the densities assigned within the program when the input file contains 0.0 for the density. If the velocity model is a stack of constant-velocity layers, then layer thickness rather than depth could have been used, in which case the depths would not be repeated (i.e., there would be one entry per layer). A continuous velocity function should start with a depth of 0.0, as in the input-parameter file.



Oct 12, 2000 1:46:59 pm
 D:\SITE_AMP\OFR_AMPV.GRA
 D:\SITE_AMP\OFR_AMPV.DT

Figure 21. Velocity vs. depth for the model used in the SITE_AMP example (heavy solid line), as well as for layered approximations made by running SITE_AMP with various choices of the frequencies at which the output was calculated. Velocity is plotted only to 0.5 km depth in order to illustrate the matching of the layered models to the portions of the example model in regions characterized by velocity gradients. The layered approximations are derived so as to preserve travel time through the range of depths used in the example model.



Oct 12, 2000 1:43:05 pm
D:\SITE_AMP\OFR_AMPA.GRA
D:\SITE_AMP\OFR_AMPA.DT

Figure 22. Amplifications (including attenuation) for the `SITE_AMP` example (heavy line), as well as amplifications from the various layered approximations of the velocity model used in the example. The amplifications for the layered models were obtained using program `NRATTLE`.