User's Guide on Equivalent Linear (EQL)
in SW4 Version 3.0

Feiruo (Flora) Xia

Auspices Statement This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344

Contents

Chapter 1

Equivalent Linear Method (EQL)

The equivalent linear method approximates the nonlinear stress-strain relationship of soils using a linear solution scheme. This makes it well suited for use in SW4, which simulates linear wave propagation.

1.1 Shear Modulus Reduction and Damping Curves

The motivation behind developing a nonlinear method is that the stress-strain response of soils is known to exhibit nonlinear behavior for larger strains. This nonlinear behavior is valuable to capture, because the large magnitude earthquakes that pose the greatest risk to heavily populated metropolitan zones built on basins of soft sediment (i.e. the San Francisco Bay Area and Los Angeles) carry enough energy to produce large strains associated with nonlinear soil behavior on the near-surface.

Figure 1.1 has theoretical stress-strain and damping ratio (attenuation) curves for a generic soil. We define the secant shear modulus (commonly represented by G or μ) to be the slope of the line connecting the origin to any location along the stress-strain curve. Written in a Hooke's Law of Elasticity form, this is

$$\tau = G\gamma.$$

The small strain shear modulus G_{max} is the largest shear modulus value for any one material, and is the one generally associated with linear behavior. It provides a constant, scalar measure of the elastic shear rigidity of a soil. Under the linear assumption, the shear modulus does not change as a function of time or as a function of the magnitude of shear strain.

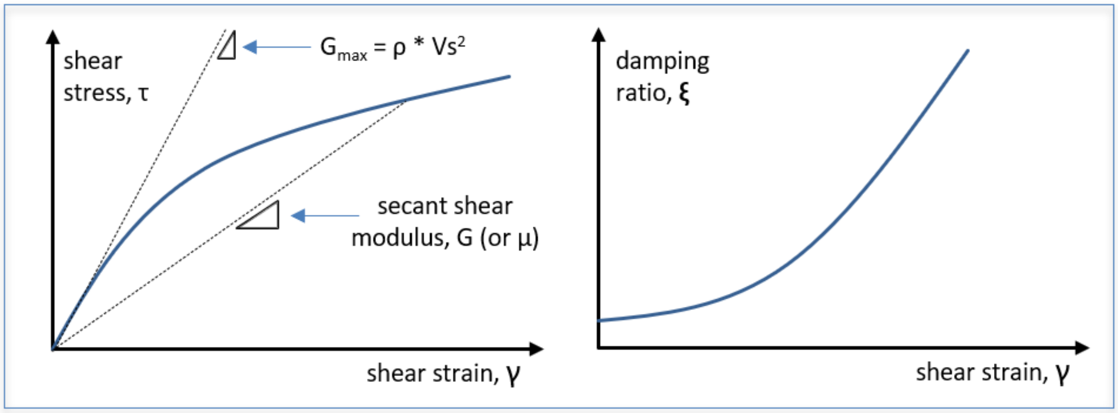

Figure 1.1: (left) Example soil stress-strain curve showing definition of secant shear modulus. (right) Example soil damping ratio curve, showing increase in damping with increased strain.

However, this assumption is not representative of the true relationship between stress and strain in soils for large strain scenarios. With an increase in strain, the true shear modulus is expected to decrease while the attenuation is conversely expected to increase. This may result in significant discrepancies between the true shear stress and the shear stress calculated from the small strain shear modulus.

1.1.1 Darendeli Curves

A primary challenge in developing nonlinear soil models is the question of how to parameterize the shear modulus and attenuation curves given limited information about the soil material properties. It is not feasible to go out and physically test every inch of soil in a regional scale model, so the Darendeli curves are used to instead generate the shear modulus reduction and damping curves given known material properties.

The Darendeli curves are empirical curves represented by a set of simple equations that were generated from first-order, second-moment Bayesian method statistical analysis of a decade of data collected at the University of Texas, Austin. Given soil density, depth of a grid point, and the shear wave velocity, the Darendeli equations are able to produce a set of shear modulus reduction and damping curves. For more information, see [2].

1.2 EQL Procedure

The goal of the equivalent linear method is to perform a linear wave propagation simulation where the material properties at every grid point in the domain of interest have been modified to correlate with approximately 65% of the maximum shear strain that each grid point is expected to experience over the course of a specific seismic event.

In other words, if γ_{max} is the largest shear strain a material will experience for a given earthquake, then we define our new equivalent linear shear modulus G_{eql} and damping ratio ξ_{eql} to be the values corresponding to $\gamma_{eff} = 0.65 * \gamma_{max}$ based on the shear modulus and attenuation curves for that material. These new equivalent linear material properties are the ones that are then used in a linear wave propagation simulation (where the material properties are held constant over the course of the simulation) to produce the equivalent linear solution.

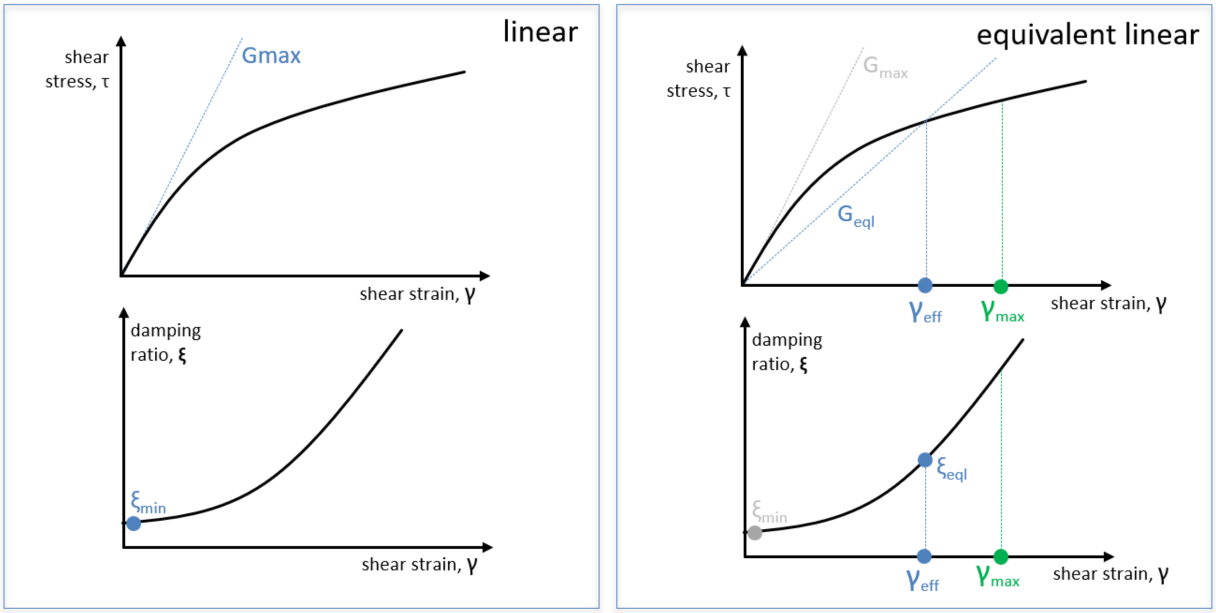

Figure 1.2: On the left, a visualization of the small strain (linear) shear modulus and damping ratio in blue with respect to nonlinear stress-strain and attenuation curves in black. On the right, a visualization of how the equivalent linear material properties compare to the linear ones.

Although this equivalent linear procedure is still considered an approximate method, as it does not capture the time variance of nonlinear soil behavior due to fluctuating strain magnitudes, it is a better approximation of nonlinear behavior than strictly linear methods. Furthermore, the method has been widely implemented, validated, and accepted by seismologists and engineers alike.

An important caveat of the equivalent linear method is that the maximum shear strain γ_{max} at every point in the computational domain for a specific seismic event is not known until the simulation has been performed. This means that the equivalent linear method necessarily is an iterative process, where the maximum shear strain and material properties are iteratively updated until some measure of convergence has been achieved. The following table and Figure 1.3 break down the different steps of the iterative process.

	Breakdown of Steps in Equivalent Linear Method
1	Initial linear simulation with linear material properties.
2	After step 1, compute effective shear strain at grid points from max shear strains.
3	Update material properties everywhere in model based on effective shear strain.
4	Run another linear site response analysis with new material properties.
5	Repeat steps 2-5 until percentage change in material properties is below threshold.
6	Run final iteration with finalized strain-compatible material properties.

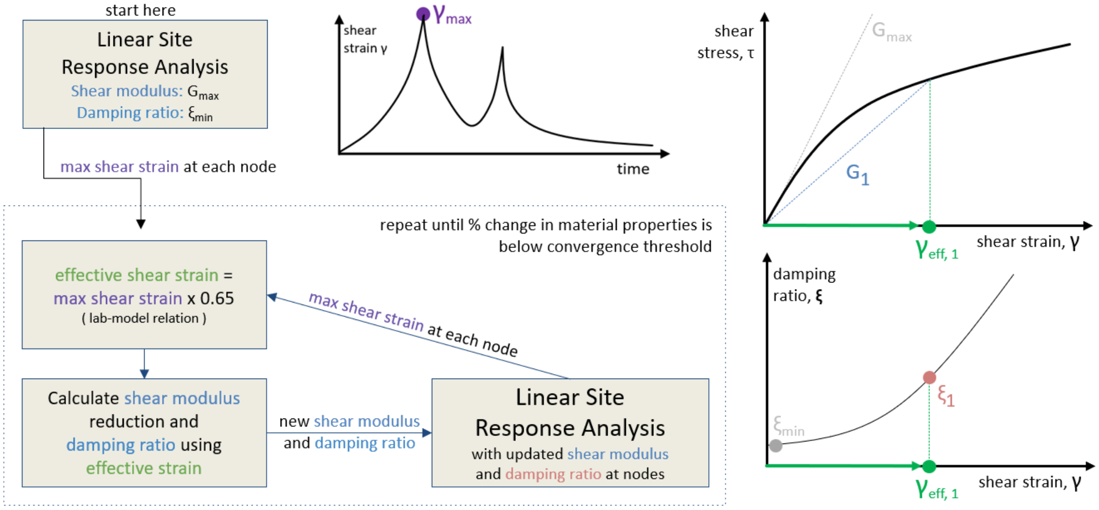
start here
Linear Site Response Analysis
Shear modulus: G_{max}
Damping ratio: ξ_{min}
max shear strain at each node
repeat until % change in material properties is below convergence threshold
effective shear strain = max shear strain x 0.65
(lab-model relation)
max shear strain at each node
Calculate shear modulus reduction and damping ratio using effective strain
new shear modulus and damping ratio
Linear Site Response Analysis
with updated shear modulus and damping ratio at nodes
shear strain γ
γ_{max}
time
shear stress, τ
G_{max}
G_1
$\gamma_{eff,1}$
shear strain, γ
damping ratio, ξ
ξ_{min}
ξ_1
$\gamma_{eff,1}$
shear strain, γ

Figure 1.3: The flowchart on the left shows the iterative process of the equivalent linear method. The two plots on the right represent how the shear modulus and damping for a material may change from G_{max} to G_1 and from ξ_{min} to ξ_1 respectively based on the effective shear strain $\gamma_{eff,1}$.

Chapter 2

Implementation of EQL in SW4

This chapter discusses some technical details behind the implementation of the equivalent linear method in SW4.

2.1 Computing Strains

2.1.1 Efficient Scalar Measure of Maximum Shear Strain

In 1D, the equivalent linear method uses the inherently scalar shear strain value to compute modifications to the shear modulus and damping. Since sw4 simulates 3D wave propagation and shear strain at any grid point is a tensor with six unique components, there is a need for an equivalent scalar measure of the shear strain.

A common measure used in other 3D simulation codes such as SPEED [12] is the largest shear strain in the coordinate system rotated 45 degrees from the principal strain axes. In other words, given principal strains $\epsilon_3 \geq \epsilon_2 \geq \epsilon_1$,

$$\gamma_{max}(\mathbf{x}, t) = \epsilon_3(\mathbf{x}, t) - \epsilon_1(\mathbf{x}, t). \tag{2.1}$$

Although this is an intuitive strain measure to use, it has the drawback of requiring an eigenvalue problem to be solved for the principal strains. For a calculation that is repeated at every grid point in a computational domain, for every time step – up to potentially hundreds of billions of times in one simulation – this motivates the need for alternative approaches that do not require the principal strains to be computed.

The scalar measure of maximum shear strain used in sw4-eql makes use of the 2nd invariant of the strain deviator tensor, often denoted by J_2, to approximate $\gamma_{max}(\mathbf{x}, t)$ in 2.1. Because J_2 is an invariant value, it can be defined in terms of the trace of the strain tensor $\epsilon(\mathbf{x}, t)$ in any coordinate frame, which is a fast computation to make:

$$J_2(\mathbf{x}, t) = 1/3 \cdot \mathrm{tr}(\epsilon(\gamma_{max}(\mathbf{x}, t))^2 - 1/2 \cdot [\mathrm{tr}(\epsilon(\mathbf{x}, t))^2 - \mathrm{tr}(\epsilon(\mathbf{x}, t)^2)]. \tag{2.2}$$

The approximation of $\gamma_{max}(\mathbf{x}, t)$ is defined using J_2 as

$$\gamma_{max} \approx \sqrt{4J_2(\mathbf{x}, t)}. \tag{2.3}$$

The error between this approximation and the true value is well constrained. We can explore this by first relating J_2 to the principal strains,

$$J_2 = \frac{1}{6}[(\epsilon_3 - \epsilon_1)^2 + (\epsilon_1 - \epsilon_2)^2 + (\epsilon_2 - \epsilon_3)^2]. \tag{2.4}$$

Substituting 2.4 in for J_2 in 2.3, we get

$$\gamma_{max} \approx \sqrt{\frac{2}{3}[(\epsilon_3 - \epsilon_1)^2 + (\epsilon_1 - \epsilon_2)^2 + (\epsilon_2 - \epsilon_3)^2]}. \tag{2.5}$$

At the limits of 2.5, the smallest value it can ever take on is exactly $\gamma_{max} = \epsilon_3 - \epsilon_1$, and the largest is $\sqrt{\frac{4}{3}} \cdot \gamma_{max} \approx 1.16\gamma_{max}$. Therefore,

$$\gamma_{max} \leq \sqrt{4J_2(\mathbf{x}, t)} \leq 1.16\gamma_{max}. \tag{2.6}$$

This error range is within the range of uncertainty for material properties in many 3D seismic velocity models, and computing 2.3 can be up to five times faster than 2.1, making this approximation well suited for use in sw4-eql.

2.1.2 Strain for Different Source Types

The solution **u** calculated by sw4 corresponds to the type of moment history time function used [8]. For instance, if the moment history time function corresponds to velocity, then the solution **u** will also be in units of velocity. However, if **u** is in units of velocity, then the strain tensor that sw4 computes using **u** is actually a strain *rate* tensor. Therefore, the implementation of the equivalent linear method requires a user to specify what format the source is in (displacement, velocity, acceleration), so the correct strain tensor can be calculated.

If the source is in units of displacement, then no additional actions need to be taken. If the source is in units of velocity, then the corresponding displacement u(t) at each new time step is additionally calculated from the velocity solution v(t) using forward Euler,

$$\mathrm{u(t{+}dt) = u(t) + v(t) \cdot dt} \tag{2.7}$$

If the source is acceleration, then displacement is calculated from the acceleration solution a(t) using a second order Taylor Series expansion,

$$\mathrm{u(t{+}dt) = u(t) + v(t) \cdot dt + \frac{1}{2}a(t) \cdot dt^2} \tag{2.8}$$

where

$$\mathrm{v(t) = v(t\text{-}dt) + a(t\text{-}dt) \cdot dt.} \tag{2.9}$$

In most cases, if the rupture command is invoked to import a Source Rupture File (SRF) for the simulation, the solution will be in units of velocity.

2.2 Avoiding Sources

An unintended byproduct of the necessary way moment tensor sources in sw4 are designed to impart sufficient energy into a numerical simulation is that strains computed too close to the location of a source will be unphysically large. This results in severely weakened and heavily attenuated soils close to sources, which produces distorted wave propagation behavior and unintentionally large amplitude waves in the rest of the simulation. Since the primary focus is not on nonlinear behavior close to sources, but rather on nonlinear behavior in the far field and near the surface, the simplest way to handle this issue is to have the equivalent linear method avoid modifying material properties too close to any source location; in other words, we keep the material properties linear in a user-defined radius around each source location.

2.3 Shorter Intermediate Iterations

Although the equivalent linear method requires iteration to arrive at the material properties compatible with the maximum shear strain invoked at every grid point due to a specific seismic event, there is no requirement that each intermediate iteration must continue running after all of the points in the computational domain have logged their maximum shear strain value for that iteration. In particular, for simulations that have a long tail, there is potential for significant time savings if the intermediate iterations can be stopped early.

In sw4-eql, the percentage of points in the equivalent linear domain that have logged an updated maximum shear strain value is used as a proxy for the amount of energy in the simulation. Once the percentage reaches less than 0.05%, the iteration is stopped as any further updates to the max strain in such a small percentage of points in the model has minimal impact on the results.

2.4 Convergence

The convergence statistics are grouped into four bins based on shear wave velocity – the separations between the bins are at 500m/s, 1000m/s, and 1500m/s.

There are two convergence criteria used at the end of each iteration to determine whether another intermediate iteration should be run, or if the final iteration should be run. The first criteria checks the largest percentage change in Vs in each Vs bin following the updating of material properties at the end of an intermediate iteration – when invoking the **eql** command, the user is able to define a desired threshold percentage for this criteria. The second criteria checks what percentage of grid points in each Vs bin meet the first criteria. As long as at least 99% of grid points in each Vs bin meet the first criteria, then the material properties are considered strain compatible and the final full iteration is started. The set of two criteria ensure that additional iterations are not run for less than one percent of the model.

2.5 Keyword in the input file

2.5.1 eql

The **eql** command is used to apply the equivalent linear method of iteratively modifying material stiffness and attenuation at grid points in *SW4* as a function of max strain levels experienced over a specific seismic event.

Note that when using a rupture file, the default SW4 time-series output is usually velocity instead of displacement. This means that `srctype` in most cases will be **velocity**.

Syntax:

```
eql iter=...  srctype=...  srcdistlim=...  vslim=...
depthlim=...  zlim=...  conv=...
```

Required parameters:

None

eql command parameters

Option	Description	Type	Units	Default
iter	maximum number of iterations allowed before running final simulation	int	none	99
srctype	defines whether the source is of type **displacement**, **velocity**, or **acceleration**	string	none	velocity
conv	maximum percentage change between iterations for each vs bin to count as converged	float	percent	5
srcdistlim	distance to avoid source locations by	float	m	3000
vslim	maximum original vs to apply to	float	m/s	3500
depthlim (or zlim)	maximum depth below topography (or absolute z depth)	float	m	6400000

Chapter 3

Examples

The following examples demonstrate how to use the equivalent linear method in SW4. The first example illustrates the basic functionality of the equivalent linear method, and the second example shows an application to simulating a realistic fault rupture scenario. On LC, the compiled code for sw4-eql is located at

```
/usr/workspace/wave/sw4-eql/optimize_quartz_mp/sw4
```

and can be called in an sbatch file in the same way as for sw4-3.0.

```
set progs = /usr/workspace/wave/sw4-eql/optimize_quartz_mp
set scenario = example_one_eql.sw4input
srun -N $SLURM_NNODES -n $CPUS $progs/sw4 $scenario
```

The sw4 input files and sample sbatch files for running the two examples in this chapter can be found in `/usr/workspace/wave/sw4-eql/examples/eql`.

3.1 Gaussian Hill with Low Vs Inclusion

The computational domain of this basic example is a 8000m x 8000m x 8000m cube with a simple Gaussian hill topography and a rectangular low-Vs (soft soil) basin in the center. The surrounding material has a Vs of 3500 m/s, and the basin has a Vs of 1000 m/s. The Qs of the entire domain is 300, the Vp of the entire domain is 5000 m/s, and the Qp of the entire domain is 600. Figure 3.1 shows a visualization of the linear Vs and Qs on the y-z plane at x=4000m. The x-z plane at y=4000m looks the same, since the model is symmetric about the x-z and y-z planes.

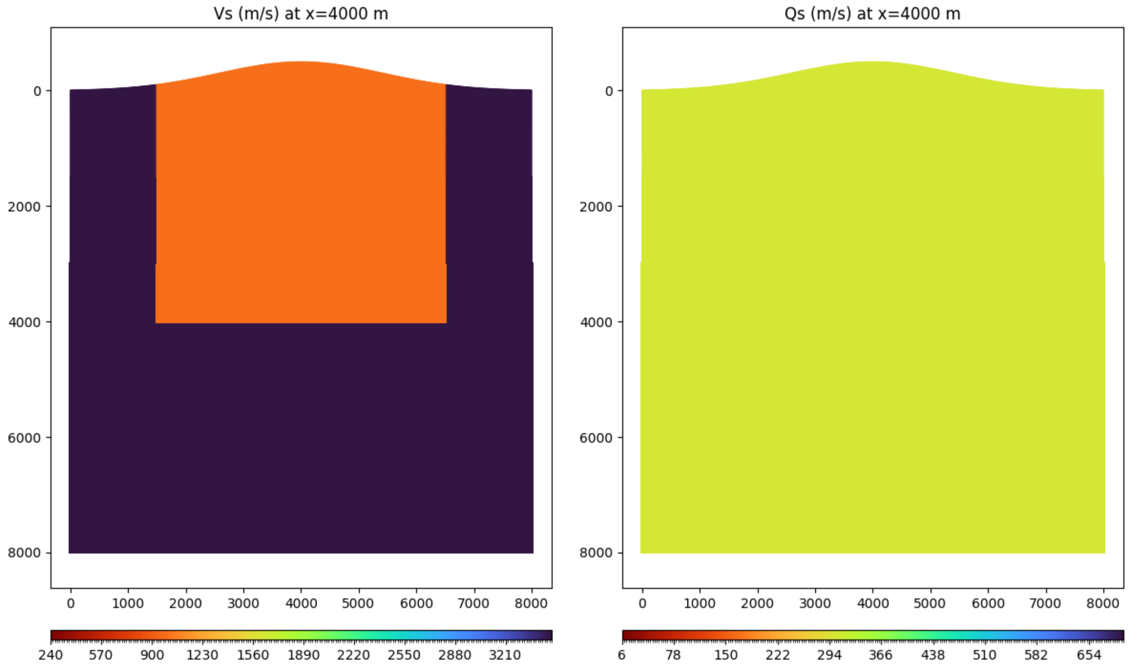

Figure 3.1: Vs and Qs on the y-z plane at x=4000m for the original linear model.

To use the equivalent linear method when simulating this model, a command such as the following is included in the sw4 input file:

```
eql iter=2 srctype=velocity srcdistlim=2500 vslim=3000 zlim=3500 conv=5.0
```

A breakdown of the different parts of the command can be found in the below table. See section 2.5 for more details on the keyword commands for eql.

Command/Param	**Explanation**
eql	command name
iter=2	Sets a maximum of two intermediate iterations before running the final iter. This means at most, three iterations will be run.
srctype=velocity	Tells the eql code that the solution **u** computed by SW4 is in velocity (important for correct strain computation).
srcdistlim=2500	Avoid changing material properties within 2500m of a source.
vslim=3000	Only modify material properties for grid points that have a Vs lower than 3000m/s.
zlim=3500	Only modify material properties for grid points with z coordinate less than 3500m.
conv=5.0	5% max change between iterations for convergence threshold.

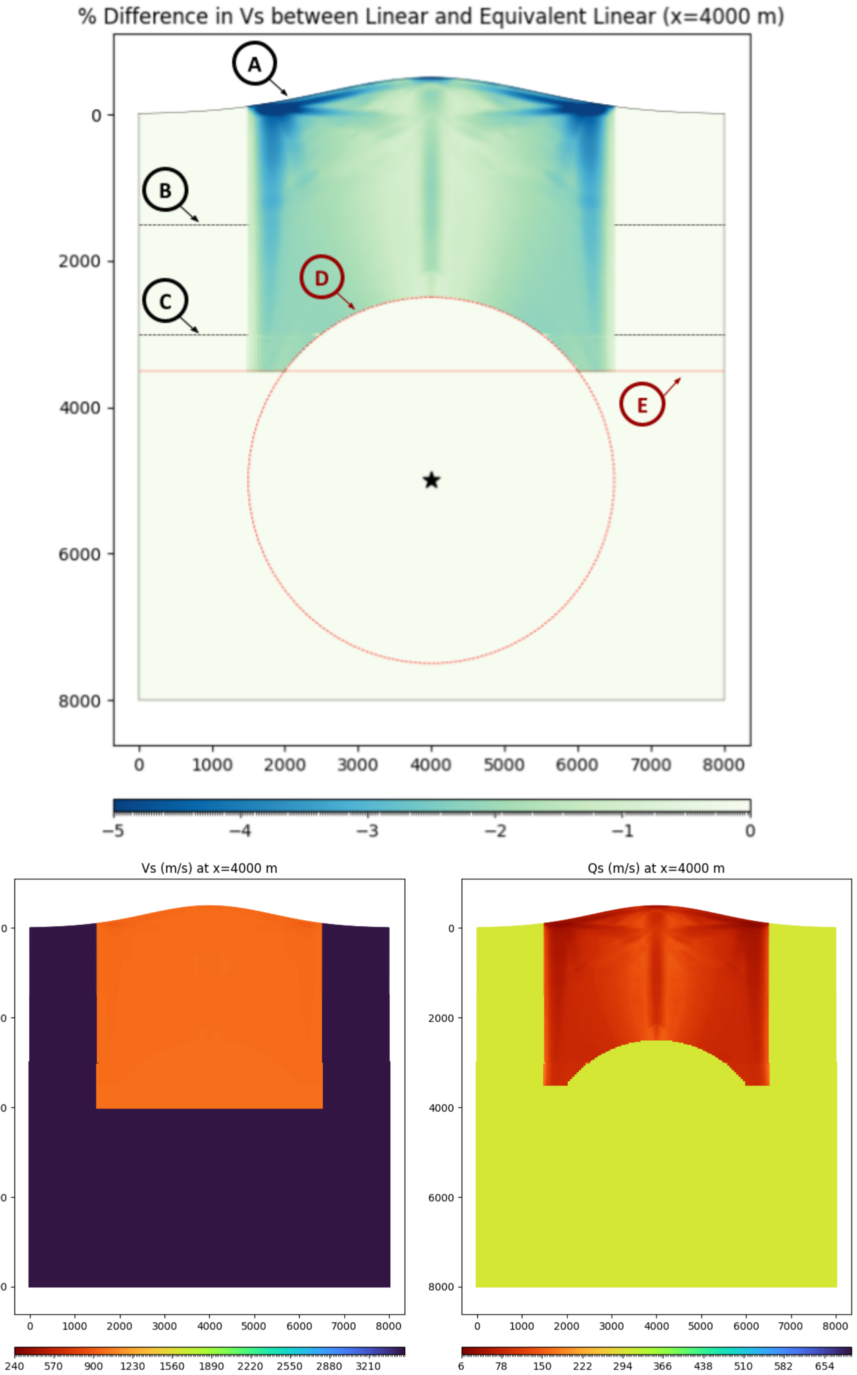

Figure 3.2: (top) Percentage difference between linear and equivalent linear for Vs.
(bottom left and right) Vs and Qs on the y-z plane at x=4000m for the Vs and Qs that have been modified after two intermediate iterations. In other words, these are the material properties used for the final solution iteration.

Figure 3.2 bottom left and right show a visualization of the resulting equivalent linear Vs and Qs on the y-z plane at x=4000m. Note that the color scales for Vs and Qs are different. The changes in Qs between linear and equivalent linear are relatively large because the equivalent linear method computes change in attenuation using damping ratio (ξ) curves, where generally $Qs = 1/2\xi$.

The changes in Vs are relatively small due to the small model and low magnitude of the single point source, so the percentage difference in Vs between linear and equivalent linear is plotted in the top image of Figure 3.2. The following table explains the markers drawn on the top image that show why the percentage difference region looks the way it does.

	Explanation of Circle-Letter Markers in Figure 3.2
A	Only the basin with a Vs of 1000m/s meets the vslim of <3000 and is modified.
B	Indicates location of curvilinear grid boundary (topography).
C	Indicates location of mesh refinement boundary.
D	Avoid changing material properties within 2500m of a source.
E	zlim=3500m means that the material properties below this line are not changed.

Finally, Figure 3.3 shows the X,Y,Z component time series and FAS for a receiver placed on the surface of the computational domain (at depth = 0).

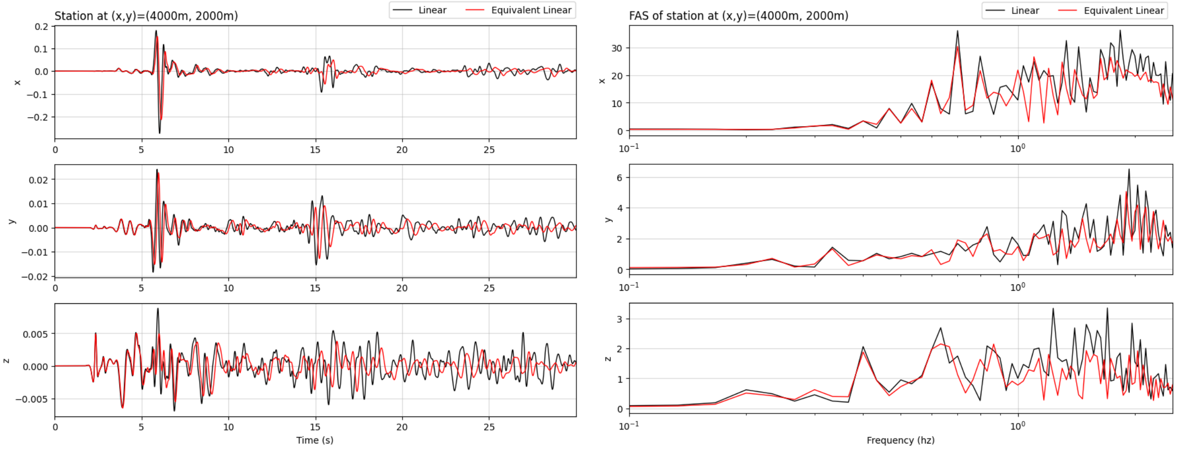

Figure 3.3: (left) X,Y,Z components of time series at (x,y) = (4000, 2000) on surface of model (at depth=0). (right) Corresponding FAS for time series on left. Black is linear, red is equivalent linear.

3.2 M6 San Francisco Bay Area Rupture Scenario

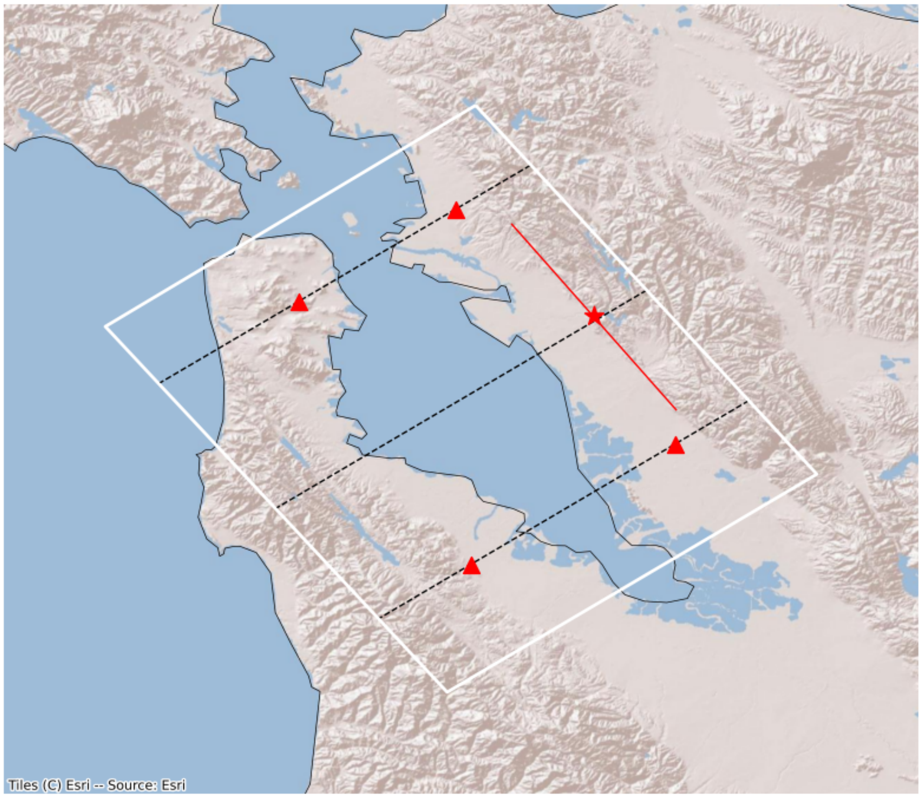

Figure 3.4: 50km x 40km computational domain in the SF Bay Area outlined in white. The red star and red line show the location of the fault and epicenter, the dashed black lines represent areas where 2D image slices of material properties are recorded, and the red triangles represent the locations of surface receivers.

This example models an approximately magnitude 6.0 rupture event along the Hayward fault in the San Francisco Bay Area, simulated on a computational domain that is 50km x 40km and 15km in depth. The Source Rupture File (SRF) containing the rupture information is

```
m6.00-25.0x8.0_s100-rvf0.72-Hayward_scor0.95_vr0.72_dh3.0.srf
```

which can be found in the same folder as the sw4 input file and sample batch file for this example.

The command added in the sw4 input file to use the equivalent linear method is as follows (without this command, a linear simulation is run):

```
eql iter=2 srctype=velocity srcdistlim=500 vslim=3500 depthlim=10000 conv=5
```

A breakdown of the different parts of the command can be found in the below table. See section 2.5 for more details on the keyword commands for eql.

Command/Param	Explanation
eql	command name
iter=2	Sets a maximum of two intermediate iterations before running the final iter. This means at most, three iterations will be run.
srctype=velocity	Tells the eql code that the solution **u** computed by SW4 is in velocity (important for correct strain computation).
srcdistlim=500	Avoid changing material properties within 500m of a source.
vslim=3500	Only modify material properties for grid points that have a Vs lower than 3500m/s.
depthlim=10000	Only modify material properties for grid points with depth below the surface less than 10km.
conv=5.0	5% max change between iterations for convergence threshold.

3.2.1 Explanation of Excerpts from Simulation Output

Start-up phase In the start-up phase, there are two major tasks that the equivalent linear method will carry out. The first is to compute the distance of each point from the source locations, to be able to prevent material properties too close to sources from being changed (see 2.2 for technical details). The second is to count the number of grid points that the equivalent linear method will modify, and to divide that count into Vs bins for calculating the convergence criteria between iterations (see 2.4 for technical details).

```
EQL: Calculating distance to sources...
EQL: Counting number of EQL grid points...

Processed dist. (min 500.00m) to src. for each grid point.
Points in bin 1:     98822833
Points in bin 2:    174997582
Points in bin 3:    237624790
Points in bin 4:   4841269644
  Total EQL pts:   5352714849

Execution time, start up phase 4.51125116e+01 seconds
```

In this example, grid points that are within 500 meters of a source are ignored, and there are a total of 5.3 billion grid points in the EQL domain.

Intermediate Iterations At the start of each intermediate iteration, the equivalent linear method will print out a count of which iteration is being started (beginning from 0). During intermediate iterations, any **image** or **rec** commands will not record values, as the material properties have not been finalized.

```
Running sw4 on 6400 processors, using 3 threads/processor...
...
STARTING EQUIVALENT LINEAR ITERATION: 0
...
STARTING EQUIVALENT LINEAR ITERATION: 1
```

Time Stepping During time stepping for intermediate iterations, an additional line will be printed out whenever the time step information is printed. This additional printout reports the number of grid points that have had their maximum strain value updated, and the percentage out of all of the points in the equivalent linear domain. This can be taken to be an indirect measure of how much energy is in the system.

```
Begin time stepping...
Time step       1  t =   2.1118581e-03
Number of grid points with updated strain:       841366 (0.0157%)
Time step      26  t =   5.4908310e-02
Number of grid points with updated strain:    288806223 (5.3955%)
Time step      51  t =   1.0770476e-01
Number of grid points with updated strain:    431646365 (8.0641%)
...
```

When the percentage of grid points in the equivalent linear domain that are being updated at each time step dips below 0.05%, the iteration is stopped as the changes to max strain beyond that point are minute and inconsequential.

```
...
Number of grid points with updated strain:      2874705 (0.0537%)
Time step    9451  t =   1.9959171e+01
Number of grid points with updated strain:      2727384 (0.0510%)
Time step    9476  t =   2.0011967e+01
Number of grid points with updated strain:      2686476 (0.0502%)
Average update percent (0.0499) below threshold, ending time stepping...
  Time stepping finished...
```

```
Execution time, time stepping phase 1 hour  48 minutes 3.51209126e+01 seconds
```

It is assumed that once this percentage approaches zero, it will not rise again. This assumption is expected to hold true as long as the simulation is of only one seismic event, and there is no source that will impart extra energy into the system at a later time than the initial series of ruptures. For more information on this mechanism of stopping iterations early, see 2.3.

Checking for Convergence Between iterations, the equivalent linear method updates material properties, and computes convergence criteria to determine whether to run another short intermediate iteration, or to run the final full iteration.

```
Calculating updated equivalent linear material properties...

   --------- EQL material properties ranges -------------
   1.59158188e+03 kg/m^3 <=  Density <= 2.93997373e+03 kg/m^3
   6.77957387e+02 m/s    <=  Vp      <= 6.44278656e+03 m/s
   1.41217416e+00        <=  Vp/Vs   <= 6.92560481e+00
   -2.11912738e+06 Pa     <=  lambda  <= 3.77455416e+10 Pa

   9.94738676e+07 Pa     <=  mu      <= 3.94949048e+10 Pa
   2.50000000e+02 m/s    <=  Vs      <= 3.74749008e+03 m/s

   Using attenuation
   1.00000000e+01        <=  Qs      <= 4.52496865e+02
   2.00000000e+01        <=  Qp      <= 9.04993729e+02
   ------------------------------------------------------
omega[0]=6.283185e-01 omega[1]=6.283185e+00 omega[2]=6.283185e+01
omc[0]=6.283185e-01 omc[1]=1.986918e+00 omc[2]=6.283185e+00...
```

The primary criteria that determines whether the material properties have converged is the percentage change in Vs after it has been updated following an intermediate iteration. The equivalent linear method finds the largest percentage change in each Vs bin, and also checks what percentage of grid points in each bin meet the convergence criteria.

In the below excerpt, since less than 99% of the grid points in the first three bins meet the 5% max change in Vs between iterations criterion, another intermediate iteration is run (as seen by the printout at the bottom for starting another iteration). For more information on the technical details behind the convergence criteria, see 2.4.

```
Checking for equivalent linear convergence...

   ------------- EQL Convergence Statistics ----------------
   Convergence criteria: at most 5.00000000e+00%
                         max change in Vs across all nodes

   0.00000000e+00 <= Vs <  5.00000000e+02 : 3.68978494e+01 %
   5.00000000e+02 <= Vs <  1.00000000e+03 : 3.03930441e+01 %
   1.00000000e+03 <= Vs <  1.50000000e+03 : 1.74087550e+01 %
   1.50000000e+03 <= Vs                   : 6.83651589e+00 %

   ---------------------------------------------------------
   Convergence criteria: at least 99% of nodes
                         meeting convergence criteria

   0.00000000e+00 <= Vs <  5.00000000e+02 : 8.57980200e+01 %
   5.00000000e+02 <= Vs <  1.00000000e+03 : 9.69202923e+01 %
   1.00000000e+03 <= Vs <  1.50000000e+03 : 9.67009848e+01 %
   1.50000000e+03 <= Vs                   : 9.99968099e+01 %
   ---------------------------------------------------------

   STARTING EQUIVALENT LINEAR ITERATION: 1
```

Final Iteration Once the percentage change in material properties after an iteration is small enough to meet the convergence criteria, the material properties are considered finalized and the final iteration begins.

```
Checking for equivalent linear convergence...

   ------------- EQL Convergence Statistics ----------------
   Convergence criteria: at most 5.00000000e+00%
                         max change in Vs across all nodes

   0.00000000e+00 <= Vs <  5.00000000e+02 : 1.09325669e+01 %
   5.00000000e+02 <= Vs <  1.00000000e+03 : 1.11316941e+01 %
   1.00000000e+03 <= Vs <  1.50000000e+03 : 4.27635230e+00 %
   1.50000000e+03 <= Vs                   : 8.85311427e-01 %
```

In this case, we can see that although there are still some grid points that have changed by over 10% after the material properties were updated following this intermediate iteration, they account for less than 0.001% of all of the grid points in that Vs bin. Because one intermediate iteration can take over an hour to run, we want to avoid running additional iterations for such a small percentage of the model.

```
---------------------------------------------------------
Convergence criteria: at least 99% of nodes
                      meeting convergence criteria

0.00000000e+00 <= Vs <  5.00000000e+02 : 9.99899720e+01 %
5.00000000e+02 <= Vs <  1.00000000e+03 : 9.99961919e+01 %
1.00000000e+03 <= Vs <  1.50000000e+03 : 1.00000000e+02 %
1.50000000e+03 <= Vs                   : 1.00000000e+02 %
---------------------------------------------------------
```

When the final full iteration begins, the printout will note that it is the final iteration. The final iteration with equivalent linear is functionally a regular linear simulation, but with modified material properties. The various pieces of equivalent linear code to track max strain and percentage of grid points are turned off, and any **image** or **rec** commands will work to save the requested outputs.

```
EQUIVALENT LINEAR FINAL ITERATION: 2
...
 Begin time stepping...
Time step       1  t =   2.1118581e-03
Time step      26  t =   5.4908310e-02
writing image plane on file .../surf.cycle=00047.z=0.uz.sw4img
writing image plane on file .../wave.cycle=00047.y=8000.ux.sw4img
...
  ==> Max wallclock time to write images is 3.21630377e+02 seconds.

   Execution time, time stepping phase 5 hours 6 minutes 2.81988993e+01 seconds
  ==> Max wallclock time to write time-series data is 4.84484673e-01 seconds.
```

Because the final iteration runs for the full simulation time, it will generally take longer than the intermediate iterations, and about as long as a linear simulation.

3.3 Running the Examples

It is recommended to copy the examples from

```
/usr/workspace/wave/sw4-eql/examples/eql
```

into a personal directory for execution. Each example comes with a sw4input file and a sbatch file with recommended options for submitting the job to a LC computer to run. There are two minor modifications that a user will need to make to the files before running the examples:

- In the batch files, the user will need to put in the bank they are using for computations where the file says `#SBATCH -A <ACCOUNT HERE>`
- In the sw4 input files, the user will need to put a desired target location for the simulation outputs to save to, where it says `fileio path=/target/path/here`

Additionally, for the M6 example it is recommended to do a short test run first, by changing `#SBATCH -t 16:00:00` to `#SBATCH -t 00:10:00`. This allocates about enough time (ten minutes) and resources to test the simulation setup and run a couple time steps, and should be able to get through the SLURM queue relatively quickly. The full simulation takes approximately 10 hours to run.

Appendix A

Quick Start Reference for LC Systems

On LC, the compiled code for sw4-eql is located at

```
/usr/workspace/wave/sw4-eql/optimize_quartz_mp/sw4
```

and can be called in an sbatch file in the same way as sw4-3.0, such as in the below code snippet example.

```
set progs = /usr/workspace/wave/sw4-eql/optimize_quartz_mp
set scenario = example_one_eql.sw4input
srun -N $SLURM_NNODES -n $CPUS $progs/sw4 $scenario
```

The sw4-eql code has all of the same functionality as sw4-3.0, and does not interfere with any of the existing features of sw4-3.0. To use the equivalent linear method in simulations, invoke the **eql** command in the sw4input file. If the **eql** command is not used, then a linear simulation will be run just as in sw4-3.0.

An example of an **eql** command that may go into a sw4 input file:

```
eql iter=2 srctype=velocity srcdistlim=500 vslim=3500 depthlim=10000 conv=5
```

See 2.5 for more information on the keyword parameters associated with the command.

Bibliography

[1] Volkan Akçelik, George Biros, Omar Ghattas, Judith Hill, David Keyes, and Bart van Bloemen Waanders. Parallel algorithms for pde-constrained optimization. In *Parallel processing for scientific computing*, pages 291–322. SIAM, 2006.

[2] Mehmet Baris Darendeli. *Development of a new family of normalized modulus reduction and material damping curves*. The university of Texas at Austin, 2001.

[3] Murat Dicleli and Srikanth Buddaram. Comprehensive evaluation of equivalent linear analysis method for seismic-isolated structures represented by sdof systems. *Engineering Structures*, 29(8):1653–1663, 2007.

[4] Izzat M Idriss and H Bolton Seed. Seismic response of horizontal soil layers. *Journal of the Soil Mechanics and Foundations Division*, 94(4):1003–1031, 1968.

[5] James Kaklamanos, Laurie G Baise, Eric M Thompson, and Luis Dorfmann. Comparison of 1d linear, equivalent-linear, and nonlinear site response models at six kik-net validation sites. *Soil Dynamics and Earthquake Engineering*, 69:207–219, 2015.

[6] David McCallen, Houjun Tang, Suiwen Wu, Eric Eckert, Junfei Huang, and N Anders Petersson. Coupling of regional geophysics and local soil-structure models in the eqsim fault-to-structure earthquake simulation framework. *The International Journal of High Performance Computing Applications*, 36(1):78–92, 2022.

[7] CN Meissner. Eqsim and raja: Enabling exascale predictions of earthquake effects on critical infrastructure. Technical report, Lawrence Liv-

ermore National Laboratory (LLNL), Livermore, CA (United States), 2023.

[8] N Anders Petersson and Bjorn Sjogreen. Sw4.

[9] N Anders Petersson and Björn Sjögreen. Wave propagation in anisotropic elastic materials and curvilinear coordinates using a summation-by-parts finite difference method. *Journal of Computational Physics*, 299:820–841, 2015.

[10] H Bolton Seed and Izzat M Idriss. Influence of soil conditions on ground motions during earthquakes. *Journal of the Soil Mechanics and Foundations Division*, 95(1):99–137, 1969.

[11] Björn Sjögreen and N Anders Petersson. A fourth order accurate finite difference scheme for the elastic wave equation in second order formulation. *Journal of Scientific Computing*, 52:17–48, 2012.

[12] Marco Stupazzini, Roberto Paolucci, and Heiner Igel. Near-fault earthquake ground-motion simulation in the grenoble valley by a high-performance spectral element code. *Bulletin of the Seismological Society of America*, 99(1):286–301, 2009.

[13] Lu Zhang, Siyang Wang, and N Anders Petersson. Elastic wave propagation in curvilinear coordinates with mesh refinement interfaces by a fourth order finite difference method. *SIAM Journal on Scientific Computing*, 43(2):A1472–A1496, 2021.