Modeling and Simulation of Earthquake Soil Structure Interaction for Buildings, Dams, Bridges Nuclear Installations and Tunnels: Real ESSI Simulator Core Functionality

Yuan Feng, Han Yang, Hexiang Wang

and

Boris Jeremić

University of California, Davis, CA, USA

Version: 13Feb2026, 15:41

http://real-essi.us/

This document is an excerpt from: http://sokocalo.engr.ucdavis.edu/~jeremic/LectureNotes/

please use google-chrome to view this PDF so that hyperlinks work

Contents

Chapter 1

Modeling and Simulation of Earthquake-Soil-Structure Interaction for Nuclear Installations, Dams, Buildings, Bridges: Core Functionality (2018-)

1.1 Core Functionality for ESSI Analysis of Nuclear Installations

Presented here are models that represent core functionality for elastic and inelastic analysis of infrastructure objects, including nuclear installations. There exist a number of other models, with different sophistication levels, that can be used, depending on the amount of data available, about the soil, rock, concrete, contacts/interfaces and seismic motions (Jeremić et al., 1989-2025). However, in order to begin to use of inelastic/nonlinear analysis, and assess inelastic/nonlinear effects on a dynamic response of soil structure systems, a set of initial models and analysis parameters are needed. Provided below is a set of models and materials parameters that are recommended for initial use of inelastic/nonlinear analysis of soil structure systems, using the Real-ESSI Simulator system. (`http://real-essi.info/`). It is noted that a detailed description of examples, commands and the Real-ESSI Simulator system is provided by Jeremić et al. (1989-2025, 1988-2025) and is also available at the Real-ESSI Simulator web site `http://real-essi.info/`. In addition, preprocessing, model development and postprocessing, results visualization for the Real-ESSI Simulator system is also described in detail pre and post processing documents that are available at `http://real-essi.info/`.

1.2 Model Setup

Each model has to be named:

`model name "model_name_string";`

In addition to that, there are a number of other considerations to be aware of:

- Each command line has to end with a semicolon ";"
- Comment on a line begins with either "//" or "!" and last until the end of current line.
- Units are required (see more below) for all quantities and variables.
- Include statements allow splitting source into several files
- All variables are double precision (i.e. floats) with a unit attached.
- All standard arithmetic operations are implemented, and are unit sensitive.
- Internally, all units are represented in the base SI units (m - s - kg).
- The syntax ignores extra white spaces, tabulations and newlines. Wherever they appear, they are there for code readability only. (This is why all commands need to end with a semicolon).

1.3 Linear Elastic Modeling

for single stage linear elastic modeling, one stage of loading has to be defined

```
new loading stage "self weight loading stage";
```

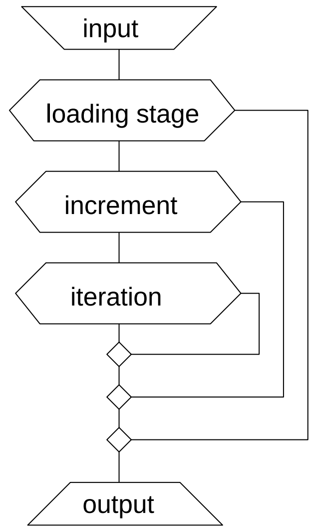

Figure 1.1:

1.4 Nonlinear/Inelastic Modeling

For inelastic modeling, stages of loading have to be defined in proper sequence.

```
new loading stage "self weight loading stage";
```

. . .

```
new loading stage "Seismic Loading";
```

. . .

1.5 Model Domain

Finite element model is developed by defining the finite element mesh which is made of nodes, finite elements, the material, and the loads.

1.5.1 Nodes

For example:

```
 add node No 1 at (1.0*m, 2.5*m, 3.33*m) with 3 dofs;
```

adds a node number 1 at coordinates $x = 1.0m$, $y = 2.5m$ and $z = 3.33m$ with 3 dofs. The nodes can be of 3dofs $[u_x, u_y, u_z]$, 4dofs $[]u_x, u_y, u_z, p]$ (u-p elements) , 6dofs $[u_x, u_y, u_z, r_x, r_y, r_z]$ (beams and shells) and 7 dofs $[u_x, u_y, u_z, p, U_x, U_y, U_z]$ (upU element) types.

1.5.2 Boundary Conditions

Example fix translation x and y for node $\#3$ `fix node # 3 dofs ux uy;`

Example fix all appropriate DOFs for node #7. `fix node # 7 dofs all;`

1.5.3 Static Acceleration Field

Example adding acceleration induced loading field for (some) elements

```
add acceleration field # 1
ax =      0*m/s^2
ay =      0*m/s^2
az = -9.81*m/s^2;
```

1.5.4 Dynamic Acceleration Field, Earthquake

One Example of add DRM load from wave fields:

```
add load # 1 type DRM from wave field
  # 1 in direction ux
  # 2 in direction uy
  soil_surface at z = 60.0*m
  hdf5_file = "input.hdf5" ;
```

1.5.5 Super Element

Super element is defined by providing mass and stiffness matrix, together with nodes and DOF numbering. It is assumed that the Super Element is a linear elastic element that is made up of a number of other finite elements. Super Element represents a part of model (structure, solid) that is linear elastic, and that has stiffness and mass matrix already defined using other finite element programs. Other finite element programs export stiffness and mass matrix. In addition, information about Super Element node numbers and degrees of freedom (DOFs) needs to be supplied as well.

1.6 Structural Modeling

Presented in this section are models that are used for modeling and simulation of structural behavior. Following the usually made assumption that structural components will remain linear elastic, only linear elastic material is used for structural modeling. It is noted, that fully nonlinear (inelastic, elastic-damage-plastic) models are also available for modeling of structural components (Jeremić et al., 1989-2025, 1988-2025). However, for the purpose of presenting core functionality features, those models are not covered here.

It is noted that a complete structural model can be replaced with one linear elastic super element, as described in section 1.5.5 on page 7.

1.6.1 Truss

Truss element represents a 3D two node linear geometry truss member. Real-ESSI command for truss element is given in detail in section **??**.

```
add element # 1 type truss
  with nodes (1,2)
  use material # 1
  cross_section = 1*m^2
  mass_density  = 2000*kg/m^3;
```

1.6.2 Beam

Beam finite element represents a 3D linear geometry, two node Bernoulli beam member, with 6 DOFs per node. Real-ESSI command for beam element is given in detail in section **??**.

```
add element # 1 type beam_elastic
    with nodes (1, 2)
    cross_section   = 1*m^2
    elastic_modulus = 2e8*Pa
    shear_modulus   = 1e8*Pa
    torsion_Jx      = 0.33*m^4
    bending_Iy      = 1.0/12*m^4
    bending_Iz      = 1.0/12*m^4
    mass_density    = 2000*kg/m^3
    xz_plane_vector = (1, 0, 1 )
    joint_1_offset  = (0*m, 0*m, 0*m )
    joint_2_offset  = (0*m, 0*m, 0*m );
```

1.6.3 Shell

Shell finite element represents a 3D linear elastic geometry, 4 node ANDES shell member with 6DOFs per node, including drilling DOFs (in plane twist). Real-ESSI command for shell element is given in detail in section **??**.

```
add element # 1 type 4NodeShell_ANDES
    with nodes (1,2,3,4)
    use material # 1
    thickness = 1*m ;
```

1.7 Solid Modeling

Presented in this section are models that are used for modeling and simulation of soils, using solid and contact/interface elements for interface of foundations and soil. Models for soil can be linear elastic, while they can also be nonlinear/inelastic, mimicking simple G/G_{max} behavior. Models for contact/interface can represent bonded contact, where no slip or gapping is allowed, and also a frictional slip and gapping contact/interface.

It is noted, that a number of more or less sophisticated material models for soil and for contact/interface are also available (Jeremić et al., 1989-2025, 1988-2025). However, for the purpose of presenting core functionality features, those models are not covered here.

1.7.1 Solid Brick

Solid brick finite element with 8 nodes, linear interpolation of displacements between nodes, and three DOFs per node is available. This element is very good for modeling soil volume close to and far away from the structural. Real-ESSI command for 8 node solid brick is given in detail in section **??**.

```
add element # 1 type 8NodeBrick
    using 2 Gauss points each direction
    with nodes (1, 2, 3, 4, 5, 6, 7, 8)
    use material # 1;
```

1.7.2 Contact, Interfaces, Joints

```
add element # 1 type StressBasedSoftContact_NonLinHardShear
    with nodes (1, 2)
    initial_axial_stiffness        = 5*MPa
    stiffening_rate                = 100
    max_axial_stiffness            = 800*MPa
    initial_shear_stiffness        = 800*kPa
    axial_viscous_damping          = 50*Pa*s
    shear_viscous_damping          = 50*Pa*s
    residual_friction_coefficient = 0.68
    shear_zone_thickness           = 5e-3*m
    contact_plane_vector           = (0, 0, 1 );
```

1.8 Core Material Modeling Parameters for Soil, Rock, Concrete, and Steel

1.8.1 Linear and Nonlinear Elastic Soil, Rock, Concrete, and Steel Modeling

1.8.2 Inelastic/Nonlinear Soil Modeling

Simple modeling of soil can be done using the so called stiffness degradation curves, or G/G_{max} curves, and damping curves, as developed by Seed and Idriss (1970).

As an example, an elastic plastic material model based on von Mises yield surface with isotropic hardening or softening and Armstrong Frederick nonlinear kinematic hardening can be used to develop such curves. Model parameters are given below:

```
add material # 1 type vonMisesArmstrongFrederick
    mass_density             = 2500*kg/m^3
    elastic_modulus          = 30 * MPa
    poisson_ratio            = 0.3
    von_mises_radius         = 300 * Pa
    armstrong_frederick_ha   = 150 * MPa
    armstrong_frederick_cr   = 25000
    isotropic_hardening_rate = 0*Pa;
```

while the corresponding G/G_{max} and damping curves are given in Figure 1.2.

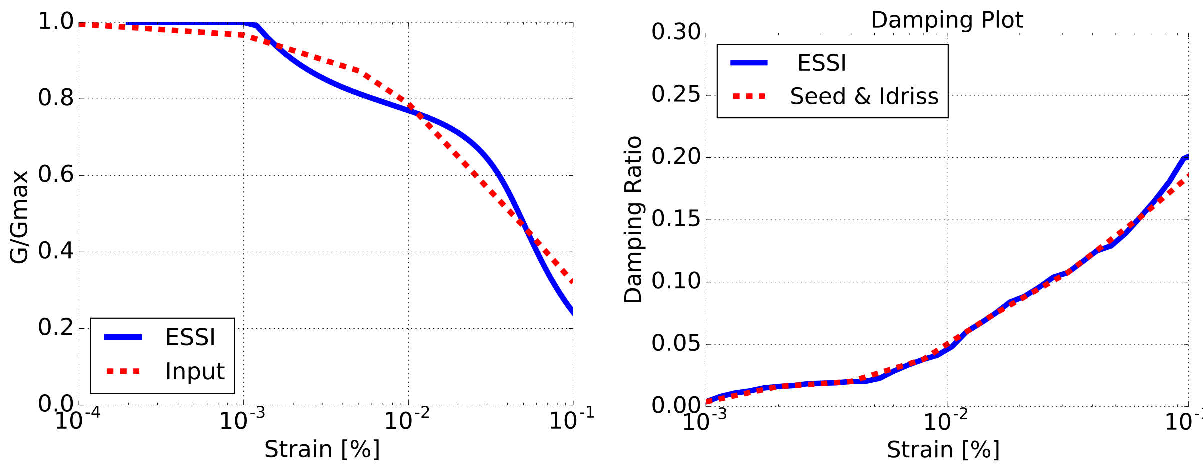

Figure 1.2: Stiffness degradation (G/G_{max}) and damping curves developed using von Mises Armstrong-Frederick Nonlinear Kinematic Hardening material model.

It is noted that von Mises Armstrong-Frederick Nonlinear Kinematic Hardening material model is a full 3D elastic plastilc material model, that is capable of modeling G/G_{max} and damping behavior, defined in 1D shear testing, fairly well in full 3D.

The command is

```
add material # <.> type vonMisesArmstrongFrederick
    mass_density                = <M/L^3>
    elastic_modulus             = <F/L^2>
    poisson_ratio               = <.>
    von_mises_radius            = <.>
    armstrong_frederick_ha      = <F/L^2>
    armstrong_frederick_cr      = <.>
    isotropic_hardening_rate = <F/L^2> ;
```

1.8.3 Inelastic/Nonlinear Rock Modeling

1.8.4 Inelastic/Nonlinear Concrete Modeling

1.8.5 Inelastic/Nonlinear Steel Modeling

1.9 Core Material Modeling Parameters for Contacts, Interfaces and Joints

The command for stress based dry soft nonlinear hardening is:

```
add element # 1 type StressBasedSoftContact_NonLinHardShear
    with nodes (1, 2)
    initial_axial_stiffness       = 5*MPa
    stiffening_rate               = 100
    max_axial_stiffness           = 800*MPa
    initial_shear_stiffness       = 800*kPa
    axial_viscous_damping         = 50*Pa*s
    shear_viscous_damping         = 50*Pa*s
    residual_friction_coefficient = 0.68
    shear_zone_thickness          = 5e-3*m
    contact_plane_vector          = (0, 0, 1 );
```

1.9.1 Mass Concrete Against Silt, Sand, Gravel and Clay

A set of initial recommended material parameters for frictional contact/interface are given in Tables 1.1 for contact between mass concrete and sand/gravel. Frictional properties given below are recommended by NAVFAC (1986).

Table 1.1: Friction coefficients for contact/interface of dissimilar materials, mass concrete against soil.

Mass concrete on soil	Friction coefficient ($\tan\phi$)	Friction angle (ϕ)
Clean sound rock	0.70	35^o
Clean gravel, gravel sand mixture, coarse sand	$0.55 - 0.60$	$29^o - 31^o$
Clean fine to medium sand, silty medium to coarse sand	$0.45 - 0.55$	$24^o - 29^o$
Fine sandy silt, nonplastic silt	$0.35 - 0.45$	$19^o - 24^o$
Very stiff clay	$0.40 - 0.50$	$22^o - 27^o$

Example command for a contact/interface element for mass concrete against clean sand, silty sand-gravel mix, single size rock fill (friction coefficient 0.30) is given below:

```
add element # 1 type StressBasedSoftContact_NonLinHardShear
    with nodes ( 1, 2)
    initial_axial_stiffness       = 10 * MPa
    stiffening_rate               = 100
    max_axial_stiffness           = 50 * MPa
    initial_shear_stiffness       = 40 * kPa
    axial_viscous_damping         = 100 * Pa * s
    shear_viscous_damping         = 100 * Pa * s
    residual_friction_coefficient = 0.30
    shear_zone_thickness          = 5e-3*m
    contact_plane_vector          = (0, 0, 1);
```

Another example, for contact/interface element for mass concrete against clean gravel, gravel sand mixture, coarse sand (friction coefficient 0.55-0.60) is given below:

```
add element # 1 type StressBasedSoftContact_NonLinHardShear
```

```
    with nodes ( 1, 2)
    initial_axial_stiffness       = 20 * MPa
    stiffening_rate               = 100
    max_axial_stiffness           = 100 * MPa
    initial_shear_stiffness       = 80 * kPa
    axial_viscous_damping         = 200 * Pa * s
    shear_viscous_damping         = 200 * Pa * s
    residual_friction_coefficient = 0.55
    shear_zone_tkness             = 1e-2*m
    contact_plane_vector          = (0, 0, 1);
```

1.9.2 Steel Sheet Against Sand, Gravel and Rockfill

Recommended material parameters for frictional contact/interface of steel sheets against sand and gravel are given in Tables 1.2. Frictional properties given below are recommended by NAVFAC (1986).

Table 1.2: Friction coefficients for contact/interface of dissimilar materials, steel sheet piles against soil.

Steel sheets against soil	Friction coefficient ($\tan\phi$)	Friction angle (ϕ)
Clean gravel, sand-gravel mix, well graded rock fill	0.40	22^o
Clean sand, silty sand-gravel mix, single size rock fill	0.30	17^o
Fine sandy silt, nonplastic silt	0.20	11^o

Example commands for contact/interface element for steel sheets against clean sand, silty sand-gravel mix, single size rock fill (friction coefficient 0.30) is given below:

```
add element # 1 type StressBasedSoftContact_NonLinHardShear
    with nodes ( 1, 2)
    initial_axial_stiffness       = 1000 * MPa
    stiffening_rate               = 100
    max_axial_stiffness           = 5 * GPa
    initial_shear_stiffness       = 4 * MPa
    axial_viscous_damping         = 100 * Pa * s
    shear_viscous_damping         = 100 * Pa * s
    residual_friction_coefficient = 0.30
    shear_zone_thickness          = 5e-3*m
    contact_plane_vector          = (0, 0, 1);
```

and for steel sheets against clean gravel, sand-gravel mix, well graded rock fill, with friction coefficient 0.40, command is:

```
add element # 1 type StressBasedSoftContact_NonLinHardShear
    with nodes ( 1, 2)
    initial_axial_stiffness       = 2000 * MPa
    stiffening_rate               = 100
    max_axial_stiffness           = 10 * GPa
    initial_shear_stiffness       = 8 * MPa
    axial_viscous_damping         = 100 * Pa * s
    shear_viscous_damping         = 100 * Pa * s
```

```
    residual_friction_coefficient = 0.40
    shear_zone_thickness          = 5e-3*m
    contact_plane_vector          = (0, 0, 1);
```

1.9.3 Formed Concrete Against Sand, Gravel and Rockfill

Recommended material parameters for frictional contact/interface of formed concrete against sand and gravel are given in Tables 1.3. Frictional properties given below are recommended by NAVFAC (1986).

Table 1.3: Friction coefficients for contact/interface of dissimilar materials, formed concrete against soil.

Formed concrete against soil	Friction coefficient ($\tan\phi$)	Friction angle (ϕ)
Clean gravel, sand-gravel mix, well graded rock fill	$0.40 - 0.50$	$22^o - 27^o$
Clean sand, silty sand-gravel mix, single size rock fill	$0.30 - 0.40$	$17^o - 22^o$
Silty sand, gravel or sand mixed with silt and clay	0.30	17^o
Fine sandy silt, nonplastic silt	0.25	14^o

Example command for contact/interface element for formed concrete against clean gravel, sand-gravel mix, well graded rock fill (friction coefficient 0.40-0.50) is given below:

```
add element # 1 type StressBasedSoftContact_NonLinHardShear
    with nodes ( 1, 2)
    initial_axial_stiffness       = 30 * MPa
    stiffening_rate               = 100
    max_axial_stiffness           = 150 * MPa
    initial_shear_stiffness       = 120 * kPa
    axial_viscous_damping         = 100 * Pa * s
    shear_viscous_damping         = 100 * Pa * s
    residual_friction_coefficient = 0.40
    shear_zone_thickness          = 5e-3*m
    contact_plane_vector          = (0, 0, 1);
```

1.9.4 Rock or Concrete on Rock or Concrete

More recently, Lei and Barton (2022) presented a very nice set of experiments with data for proper choice of interface parameters for rock on rock interface, that can also be used for concrete as well.

1.10 Earthquake Motion Modeling

1.10.1 One Component (1C) Seismic Motions Defined at Surface or at Depth

DRM...

One can add DRM loading directly, where `input.hdf5` specifies the DRM motions to all DRM nodes.

```
add load # 1 type DRM
    hdf5_file = "input.hdf5"
```

```
    scale_factor = 1.0 ;
```

Since the direct specification of DRM motions to all DRM nodes is complicated, alternatively, user is able to specify DRM motion using a surface motion. Internally, wave deconvolution is conducted to specify the DRM motions to all DRM nodes.

```
add wave field # 1 with
    acceleration_filename = "acceleration.txt"
    unit_of_acceleration  = 1 * m/s^2
    displacement_filename = "displacement.txt"
    unit_of_displacement  = 1 * m
    add_compensation_time = 0.0 * s
    motion_depth          = 0 * m
    monitoring_location   = within_soil_layer
    soil_profile_filename = "soil_profile.txt"
    unit_of_Vs            = 1 * m/s
    unit_of_rho           = 1 * kg/m^3
    unit_of_damping       = absolute
    unit_of_thickness     = 1*m
    ;
```

```
add load # 1 type DRM from wave field # 1 in direction ux
    soil_surface at z = 0.0*m
    hdf5_file         = "input.hdf5" ;
```

where `input.hdf5` specifies the HDF5 file which contain the information about the DRM elements and DRM nodes.

1.10.2 3 × 1C Seismic Motions Defined at Surface or at Depth

One Example of add DRM load from wave fields:

```
add load # 1 type DRM from wave field
  # 1 in direction ux
  # 2 in direction uy
  # 3 in direction uz
  soil_surface at z = 0.0*m
  hdf5_file = "input.hdf5" ;
```

1.10.3 Seismic Motions Imposed at Model Base

```
add load # 1 type imposed motion to node # 1 dof ux
    time_step                = 0.01*s
    displacement_scale_unit = 1*m
    displacement_file       = "displacement.txt"
    velocity_scale_unit     = 1*m/s
    velocity_file           = "velocity.txt"
    acceleration_scale_unit = 1*m/s^2
    acceleration_file       = "acceleration.txt";
```

1.10.4 Eigen Analysis

For structural model alone.

```
simulate using eigen algorithm
    number_of_modes = 3;
```

1.11 Core Modeling and Simulation Commands: Simulation Parameters

Developed model, using core functionality, as described above, numerically simulated using core functionality simulation controls.

Finite element system of equations can be solved in sequential processing mode, for smaller models, on sequential, single CPU computers (laptops, desktops, single CPU Amazon Web Services computers, etc.):

```
define solver sequential umfpack;
```

For larger models, parallel processing mode, on parallel computers ((multi CPU laptops, multi CPU desktops, clusters of PCs, Amazon Web Services parallel computers, Supercomputers, etc.):

Command Example for a direct solver:

```
define solver parallel petsc "-pc_type lu ↩
   -pc_factor_mat_solver_package mumps" ;
```

For selfweight phase of loading, static solution algorithm is used:

```
 simulate 100 steps using static algorithm;
```

For static loading, for example self weight as described above, load application and the siulation process is controlled trough load control:

```
define load factor increment 0.01;
```

For dynamic loads, simulation process is controlled using Newmark time integration method:

```
define dynamic integrator Newmark  with gamma = 0.6000 beta = 0.3025;
```

The dynamic simulation process is performed in a number of steps:

```
 simulate 2000 steps using transient algorithm time_step = 0.01*s;
```

For proper integration of constitutive equations on the integration point (Gauss point) level, within each finite element, constitutive algorithm needs to be defined:

```
define NDMaterial constitutive integration algorithm Forward_Euler;
```

For the finite element level, analysis of nonlinear systems require definition of nonlinear iteration algorithm:

```
define algorithm With_no_convergence_check;
```

Bibliography

B. Jeremić, G. Jie, Z. Cheng, N. Tafazzoli, P. Tasiopoulou, F. Pisanò, J. A. Abell, K. Watanabe, Y. Feng, S. K. Sinha, F. Behbehani, H. Yang, H. Wang, and K. D. Staszewska. *The Real-ESSI Simulator System*. University of California, Davis, 1988-2025. http://real-essi.us/.

B. Jeremić, Z. Yang, Z. Cheng, G. Jie, N. Tafazzoli, M. Preisig, P. Tasiopoulou, F. Pisanò, J. Abell, K. Watanabe, Y. Feng, S. K. Sinha, F. Behbehani, H. Yang, H. Wang, and K. D. Staszewska. *Nonlinear Finite Elements: Modeling and Simulation of Earthquakes, Soils, Structures and their Interaction*. Self-Published-Online, University of California, Davis, CA, USA, 1989-2025. ISBN 978-0-692-19875-9. URL: http://sokocalo.engr.ucdavis.edu/~jeremic/LectureNotes/.

Q. Lei and N. Barton. On the selection of joint constitutive models for geomechanics simulation of fractured rocks. *Computers and Geotechnics*, 145:104707, 2022. ISSN 0266-352X. doi: https://doi.org/10.1016/j.compgeo.2022.104707. URL https://www.sciencedirect.com/science/article/pii/S0266352X22000714.

NAVFAC. *Foundations & Earth Structures, DESIGN MANUAL 7.02*. Naval Facilities Engineering Command, 200 Stovall Street, Alexandria, Virginia 22332-2300, USA, revalidated by change 1 september 1986 edition, 1986.

H. B. Seed and I. M. Idriss. Soil moduli and damping factors for dynamic response analyses. report eerc 70-10. Technical report, University of California Berkeley, 1970.