TzSimple1Gen OpenSees Command

By: Scott Brandenberg
Date: June 28th, 2004

TzSimple1Gen $File1 $File2 $File3 $File4 $File5 <$File6>

The TzSimple1Gen command constructs TzSimple1 materials (Boulanger, 2003) for pre-defined zeroLength elements. The command requires five arguments, and supports an optional sixth argument, all of which are file names. The first file contains soil and pile properties required to define the TzSimple1 materials. The second file contains information about the nodes that define the mesh. The third file contains information about the zeroLength elements that are to be assigned TzSimple1 materials (hereafter called tz elements). The fourth file contains information about the beam column elements that are attached to tz elements. The fifth file is the output file to which the TzSimple1 materials are written. The sixth file is the output file to which the applied patterns are written.

The command has been structured such that File2, File3, File4, File5 and File6 can be sourced directly from within a master tcl file. Hence File2, File3 and File4 serve two purposes:

1. They provide information to TzSimple1Gen to create the TzSimple1 materials.
2. They can be sourced directly in a master tcl file to define the nodes, zeroLength elements for tz materials, and pile elements, respectively.

Furthermore, File5 and File 6 serve the following purpose:

1. They can be sourced directly in a master tcl file to define the TzSimple1 materials and the applied patterns.

The dual use of the files is demonstrated in an example problem in the Appendix.

File1

The first input file, File1, contains soil and pile properties that are required to calculate the material properties for the TzSimple1 materials, and optional information about t-multipliers, and applied patterns (either loads on the pile nodes, or displacements on the free ends of the t-z elements). Optional information is placed inside angle brackets (i.e. < Optional Information >). The format of File1 is as follows:

matType(1) z_t(1) z_b(1) γ' t(1) γ' b(1) p_t(1) p_b(1) Additional Arguments(1)
.
.
.
matType(N) z_t(N) z_b(N) γ' _t(N) γ' _b(N) p_t(N) p_b(N) Additional Arguments(N)
<sp or load zPattern_t(1) zPattern_b(1) PatternVal_t(1) PatternVal_b(1)
.
.
.
sp or load zPattern_t(N) zPattern_b(N) PatternVal_t(N) PatternVal_b(N)
mt zMt_t(1) zMt_b(1) MtVal_t(1) MtVal_b(1)
.
.
.
mt zMt(N) zMt_b(N) MtVal_t(N) MtVal_b(N)>

where,

matType = tz1 Approximates Reese and O'Neill (1987) soft clay t-z relation.
matType = tz2 Approximates Mosher (1984) t-z relation.

z_t = z-coordinate of top of sub-layer (meters).
z_b = z-coordinate of bottom of sub-layer (meters).
γ'_t = buoyant unit weight at top of sub-layer (kN/m^3).
γ'_b = buoyant unit weight at bottom of sub-layer (kN/m^3).
p_t = pile perimeter at coordinate z_t (meters).
p_b = pile perimeter at coordinate z_b (meters).

Additional arguments are required for each different material (i.e. each different matType), as summarized below:

matType = tz1 Approximates Reese and O'Neill (1987) relation.

AdditionalArguments = ca_t ca_b <c_t c_b>

ca_t = interface adhesion strength (kPa) at depth z_t.
ca_b = interface adhesion strength (kPa) at depth z_b.
c_t = viscous damping term (optional) on the far-field (elastic) component of the displacement rate (velocity) at depth z_t (0.0 default).
c_b = viscous damping term (optional) on the far-field (elastic) component of the displacement rate (velocity) at depth z_b (0.0 default).

matType = tz2 Approximates Mosher (1984) relation.

AdditionalArguments = delta_t delta_b <c_t c_b>

delta_t = friction angle (degrees) at depth z_t.
delta_b = friction angle (degrees) at depth z_b.
c_t = viscous damping term (optional) on the far-field (elastic) component of the displacement rate (velocity) at depth z_t (0.0 default).
c_b = viscous damping term (optional) on the far-field (elastic) component of the displacement rate (velocity) at depth z_b (0.0 default).

matType = tz3 Liquefied sand with normalized residual interface adhesion strength ratio t-z relation.

AdditionalArguments = p_t p_b phi_t phi_b Sa/σ_v'_t Sa/σ_v'_b ru_t ru_b <c_t c_b>

p_t = pile diameter (m) at depth z_t.
p_b = pile diameter (m) at depth z_b.
phi_t = friction angle (degrees) at depth z_t (used only to calculate y_{50}).
phi_b = friction angle (degrees) at depth z_b (used only to calculate y_{50}).
Sa/σ_v'_t = residual interface adhesion strength ratio at depth z_t at $r_u = 1.0$.
Sa/σ_v'_b = residual interface adhesion strength ratio at depth z_b at $r_u = 1.0$.
ru_t = peak excess pore pressure ratio at coordinate z_t.
ru_b = peak excess pore pressure ratio at coordinate z_b.
c_t = viscous damping term (optional) on the far-field (elastic) component of the displacement rate (velocity) at depth z_t (0.0 default).
c_b = viscous damping term (optional) on the far-field (elastic) component of the displacement rate (velocity) at depth z_b (0.0 default).

matType = tz4 User-specified t_{ult} and z_{ult}.

AdditionalArguments = type t_{ult}_t t_{ult}_b z_{50}_t z_{50}_b <c_t c_b>

type = 1 for approximation of Reese and O'Neill's (1987) t-z relation
type = 2 for approximation of Mosher (1984) t-z relation

t_{ult}_t = ultimate capacity of t-z element (kN/m) at depth z_t.
t_{ult}_b = ultimate capacity of t-z element (kN/m) at depth z_b.
z_{50}_t = relative displacement (soil displacement minus pile cap displacement) at which 50% of ultimate resistance is reached in a monotonic virgin loading cycle at depth z_t (meters).
z_{50}_b = relative displacement (soil displacement minus pile cap displacement) at which 50% of ultimate resistance is reached in a monotonic virgin loading cycle at depth z_b (meters).
c_t = viscous damping term (optional) on the far-field (elastic) component of the displacement rate (velocity) at depth z_t (0.0 default).
c_b = viscous damping term (optional) on the far-field (elastic) component of the displacement rate (velocity) at depth z_b (0.0 default).

Applied Patterns

sp is a character tag identifying the subsequent fields on the line as defining a displacement pattern assigned to the free ends of the t-z elements.
load is a character string identifying the subsequent fields on the line as defining a load pattern assigned to the pile nodes.
zPattern_t = z-coordinate of top of applied displacement or load (meters).
zPattern_b = z-coordinate of bottom of applied displacement or load (meters).
PatternVal_t = applied incremental displacement (meters) or load pattern (kN/m) at coordinate zPattern_t.
PatternVal_b = applied incremental displacement (meters) or load pattern (kN/m) at coordinate zPattern_b.

T-Multipliers

mt is a character string identifying the subsequent fields on the line as defining a t-multiplier.
zMt_t = z-coordinate of top of t-multiplier distribution (meters).
zMt_b = z-coordinate of bottom of t-multiplier distribution (meters).
MtVal_t = t-multiplier at coordinate zMp_t.
MtVal_b = t-multiplier at coordinate zMp_b.

File2

The second input file, File2, contains information about the nodes that define the pile elements and the zeroLength elements that are to be assigned TzSimple1 materials. The format of File2 is as follows:

```
node   nodenum(1)   y(1)   z(1)
.
.
.
node   nodenum(N)   y(N)   z(N)
```

where,

nodenum = number of node
y = y-coordinate of node
z = z-coordinate of node

File3

The third input file, File3, contains information about the zeroLength elements that are to be assigned TzSimple1 materials. The format of File3 is as follows:

element zeroLength elenum(1) node1(1) node2(1) –mat matTag (1) <ExtraInput...........>
.
.
.
element zeroLength elenum(N) node1(N) node2(N) –mat matTag (N) <ExtraInput.........>

where,

elenum = element number
node1 = a node defining the zeroLength element
node2 = a node defining the zeroLength element
matTag = material tag to be associated with a TzSimple1 material
ExtraInput.......... is an optional text string that comes after matTag. The reason for allowing "ExtraInput" is to facilitate dual use of file 2 in both the TzSimple1Gen command, and in a master tcl file as demonstrated in the Appendix.

File4

The fourth input file, File4, contains information about the pile elements to which the t-z elements connect. This file is required to calculate the tributary length to assign to each TzSimple1 material. The format of File4 is as follows:

element elementType elenum(1) node1(1) node2(1) <ExtraInput....................>
.
.
.
element elementType elenum(N) node1(N) node2(N) <ExtraInput..................>

elenum = element number

node1 = a node defining the pile element

node2 = a node defining the pile element (different from node1).

ExtraInput.......... is optional information that comes after matTag. The reason for allowing "ExtraInput" is to facilitate dual use of file 2 in both the TzSimple1Gen command, and in a master tcl file as demonstrated in the Appendix.

File5

The output file, File5, contains the TzSimple1 materials. The format of File5 is as follows:

```
########################################################################
## Start TzSimple1 Materials

uniaxialMaterial   TzSimple1   matTag(1)   tzType(1)   Tult(1)   z50(1)   c(1)
.
.
.
uniaxialMaterial   TzSimple1   matTag(N)   tzType(N)   Tult(N)   z50(N)   c(N)

## End TzSimple1 Materials
#########################################################################
```

where,

matTag = material tag associated with the corresponding zeroLength element.
tzType = 1 Backbone of t-z curve approximates Reese and O'Neill (1987) relation.
tzType = 2 Backbone of t-z curve approximates Mosher (1984) relation.
Tult = capacity of TzSimple1 material (kN).
z_{50} = relative displacement at 0.50*tult (m).
c = viscous damping term (optional) on the far-field (elastic) component of the displacement rate (velocity).

File 6

File 6 contains the Pattern that either applies loads to the pile nodes, or displacements to the free ends of the tz elements. The format of File6 for applied loads (i.e. character string "load") is as follows:

```
########################################################################
## Begin Pattern File
load nodenum 0.0 ForceValue 0.0
.
.
.
load nodenum 0.0 ForceValue 0.0
sp nodenum 2 DisplacementValue
.
.
.
sp nodenum 2 DisplacementValue
## End Pattern File
########################################################################
```

Note that tz elements are assumed to be oriented in the 2-direction, so the patterns are applied in the 2-direction.

File Format

The format of File1 must be strictly followed to prevent I/O error. Metadata and comment lines are not permitted for File 1. For File2, data is read for each line that begins with “node” and other lines are ignored. For File3 and File4, data is read for each line that begins with “element” and other lines are ignored. Hence, extra data and blank rows are permitted for File2, File3, and File4. The output files, File5 and File6, contain a header that explains that the TzSimple1Gen program was used to create the file.

The TzSimple1Gen command does not contain the functionality of tcl. The programming features of tcl allow nodes to be defined using a loop, as demonstrated in the following example:

```
# Create pile nodes, with double nodes for adding zeroLength soil springs
for {set i 0}{$i<=30}{incr i 1}{
set yDim1 [expr $i*$dy1-660]
node [expr $i*2+1]    0.           $yDim1
node [expr $i*2+2]    0.           $yDim1
node [expr $i*2+63]  -$dx          $yDim1
node [expr $i*2+64]  -$dx          $yDim1
node [expr $i*2+125]       $dx           $yDim1
node [expr $i*2+126]       dx            $yDim1
}
```

However, TzSimple1Gen cannot recognize such loops. The node numbers and nodal coordinates must be numbers for TzSimple1Gen; expressions are not permitted.

Linear Interpolation

At a given node, soil properties, t-multipliers and displacement patterns are linearly interpolated based on the location of the node and the location and associated soil properties and pattern values defined in File1. Distributed loads along the pile are integrated over the tributary length to obtain nodal loads. A node will be assigned a load pattern if any part of its tributary area overlaps with any part of the applied distributed load, even if the node itself lies outside of the region in which distributed loads were defined. T-z elements lying outside of the region of defined t-multipliers will be assigned a t-multiplier of 1.0.

APPENDIX 1: EXAMPLE PROBLEM

An example problem has been constructed to show how the structure of the input and output files relates to a simple soil-pile model. The purpose of the example problem is to illustrate the functionality of the TzSimple1Gen command, and not to accurately model the response the pile. The small number of t-z elements in the example problem is likely insufficient to result in an accurate solution, but permits a clear means of demonstrating the function of the TzSimple1Gen command.

The extended shaft shown in Figure 1 is composed of six beam column elements with nodes at or below the ground surface attached to a t-z element. A free-field soil displacement is imposed such that the surface of the clay displaces 0.2 m, and a distributed load of 5 kN/m is applied over the top 1.0 m of the pile. Material properties are shown in Figure 1.

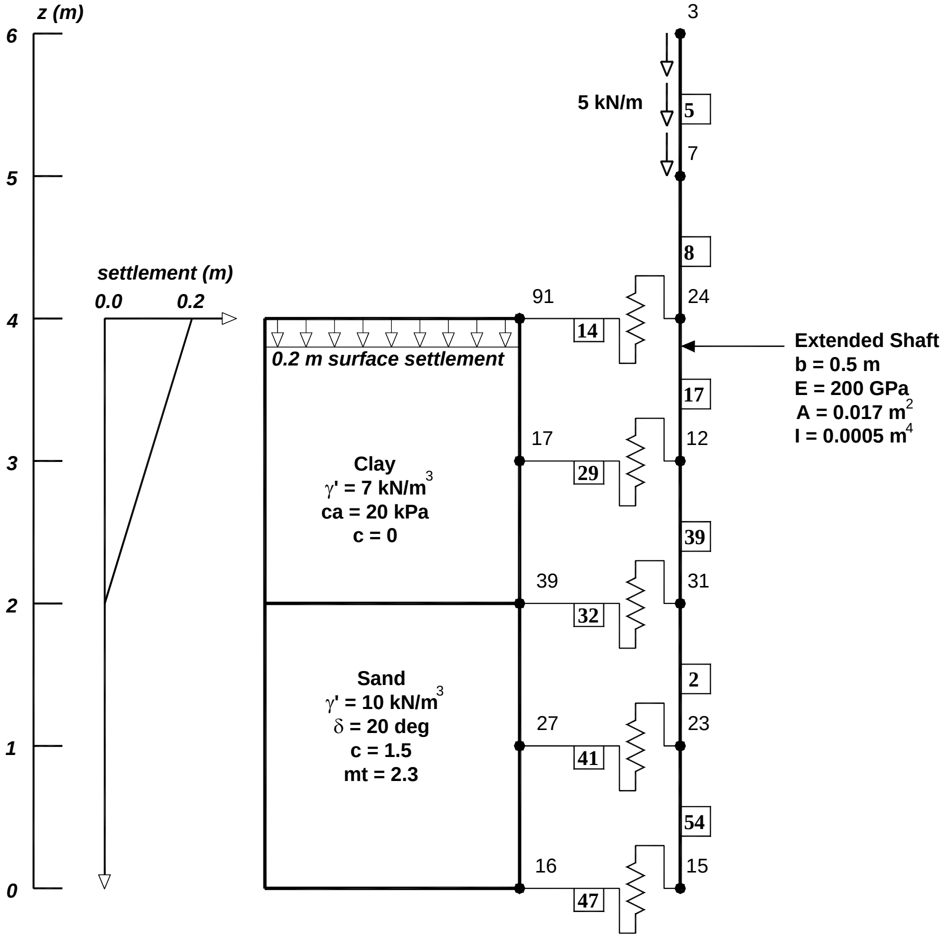

Figure 1: Example problem to illustrate TzSimple1Gen command.

File1: Soil Properties Input File “SoilProp.tcl”

```
tz1   4.0  2.0  7.0    7.0    1.6  1.6  20.0  20.0  0.0  0.0
tz2   2.0  0.0  10.0  10.0    1.6  1.6  20.0  20.0  0.0  0.0
sp    4.0  2.0  -0.002 0.0
sp    2.0  0.0  0.0   0.0
load  6.0  5.0  -0.05  -0.05
mt    2.0  0.0  2.3   2.3
```

File2: Nodes Input File “Nodes.tcl”

```
##############################################################
## Begin Nodes

node 15  0  0
node 16  0  0
node  23  0  1
node  27  0  1
node  31  0  2
node  39  0  2
node  12  0  3
node  17  0  3
node  24  0  4
node  91  0  4
node  7    0   5
node  3    0   6

## End Nodes
#####################################################################
```

File3: Tz Elements Input File “TzElements.tcl”

```
################################################################
## Begin tz elements

element   zeroLength   47   15   16   –mat   5  –dir   2
element   zeroLength   41   23   27   –mat   17  –dir   2
element   zeroLength   32   31   39   –mat   19  –dir   2
element   zeroLength   29   12   17   –mat   23  –dir   2
element   zeroLength   14   24   91   –mat   7  –dir   2

## End tz elements
####################################################################
```

File4: Pile Elements Input File “PileElements.tcl”

```
######################################################################
## Begin beam-column elements

element   elasticBeamColumn   17   24  12  0.024   200000000.0   0.001   1
element   elasticBeamColumn   39  12   31  0.024   200000000.0   0.001   1
element   elasticBeamColumn   2    31   23   0.024  200000000.0   0.001   1
element   elasticBeamColumn   54  23   15  0.024   200000000.0   0.001   1
element   elasticBeamColumn   8    24   7    0.024   200000000.0   0.001   1
element   elasticBeamColumn   6    7     3    0.024   200000000.0   0.001   1

## End beam-column elements
###########################################################################
```

File5: Tz Materials Output File “TzMaterials.tcl”

```
############################################################################
## Material Properties for tz Elements

uniaxialMaterial TzSimple1 5 2 8.43829 0.000625 0
uniaxialMaterial TzSimple1 17 2 12.8583 0.000625 0
uniaxialMaterial TzSimple1 19 1 20.4201 0.000625 0
uniaxialMaterial TzSimple1 23 1 32 0.000625 0
uniaxialMaterial TzSimple1 7 1 16 0.000625 0

## End Material Properties for tz Elements
#############################################################################
```

File6: Pattern Output File “Pattern.tcl”

```
##############################################################################
## Begin Pattern File

sp 16 2 0
sp 27 2 0
sp 39 2 0
sp 17 2 -0.001
sp 91 2 -0.002
load 7 0.0 -0.025 0.0
load 3 0.0 -0.025 0.0

## End Pattern File
##############################################################################
```

Implementation in master tcl file

TzSimple1Gen has be structured to receive certain arguments that are not required to define the tz materials (i.e. “ExtraInput”), but are required to define the nodes and elements in OpenSees. So, as a matter of convenience, TzSimple1Gen reads fields that are not essential to its function so that the files can serve two purposes:

1. They can be used arguments for the TzSimple1Gen command.
2. They can be sourced directly from a master tcl script file to define nodes and elements.

For the example problem, assume that the input files (File1 through File4) required as arguments for TzSimple1Gen are contained in the same folder as the master tcl file, and their file names are as follows:

File1: “SoilProp.txt”

File2: “Nodes.tcl”

File3: “TzElements.tcl”

File4: “PileElements.tcl”

Furthermore, assume that the output files (File5 and File6) are named:

File5: “TzMaterials.tcl”

File6: “Pattern.tcl”

Using the preceding information, the following master tcl file utilizes the files in the TzSimple1Gen command, and also to define the nodes and elements in the domain.

```
######################################################################
# Master file for tz pushover analysis to illustrate TzSimple1Gen
# command
#
# Created by Scott Brandenberg, April 30, 2004.
######################################################################

wipe

############################################################
# BUILD MODEL
#
#create the ModelBuilder for the zeroLength elements
model basic -ndm 2 -ndf 3

# define the nodes
source Nodes.tcl

# create geometric transformation for pile elements
geomTransf Linear 1

# define pile elements
source PileElements.tcl

# generate TzSimple1 materials using TzSimple1Gen command
TzSimple1Gen "SoilProp.txt" "Nodes.tcl" "TzElements.tcl" "PileElements.tcl" "TzMaterials.tcl" "Pattern.tcl"

# define tz elements and materials.  Always define materials before
# elements to prevent input error.
source TzMaterials.tcl
source TzElements.tcl


# Fix free ends of tz elements
fix 91 1 0 1
fix 17 1 0 1
fix 39 1 0 1
fix 27 1 0 1
fix 16 1 0 1

# Fix horizontal deformation and rotation at pile tip
fix 15 1 0 1

############################################################
# NOW APPLY LOADING SEQUENCE AND ANALYZE (plastic)

# Create the pattern
pattern Plain 1 Linear {
source Pattern.tcl
}
```

```
#create the recorder
recorder Node -file SoilDisplacement.dat disp -time -node 16 17 27 39 91 -dof 1 2
recorder Node –file PileDisplacement.dat disp -time -node 3 7 12 15 23 24 31 -dof 1 2
recorder Element –file PileElementRecorder.dat –time –ele 2 5 8 17 39 54 force
recorder Element –file TzElementRecorder.dat –time –ele 14 29 32 41 47 force

#############################################################
# create the Analysis

constraints Penalty 1.e12 1.e12
test NormUnbalance 2e-3 100 0
numberer RCM
algorithm Newton
system ProfileSPD
integrator LoadControl 1.0
analysis Static

#############################################################
# analyze.

set NumSteps 100
analyze $NumSteps

wipe  #flush output stream
```

APPENDIX 2: TECHNICAL INFORMATION

Calculating tult and z50

The ultimate resistances of the t-z materials were calculated based on adhesion strength for cohesive materials, and based on frictional strength for cohesionless materials. The equations used in TzSimple1Gen are included below:

For clay (tzType = 1)

$$t_{ult} = c_a \cdot p$$

where,

c_a = soil-pile interface adhesion strength (kPa)
p = pile perimeter (m)

For sand (tzType=2)

$$t_{ult} = K_o \cdot \sigma_v' \cdot \tan(\delta) \cdot p$$

where,

σ_v' = vertical effective stress
Ko = coefficient of earth pressure at rest (by default Ko = 0.4)
δ = soil-pile interface friction angle
p = pile perimeter

The relative displacement when t_{ult} is first mobilized is calculated as 0.5% of the pile diameter.

For liquefied sand (tzType = 3)

for r_u = 0 (assuming undrained capacity with r_u = 0 is the same as drained capacity)

$$tu_{ru0} = K_o \cdot \sigma_v' \cdot \tan(\delta) \cdot p$$

for $r_u = 1$

$$tu_{ru1} = Sa \cdot p$$

for $0 < r_u < 1$

$$tu = tu_{ru0} + r_u \cdot \left[tu_{ru1} - tu_{ru0} \right]$$

where,

Sa = interface adhesion strength of liquefied sand mobilized against pile
p = pile perimeter
r_u = peak excess pore pressure ratio

For user-specified (tzType = 4)

t_{ult} and z_{50} are user-specified.

Tributary Length

The t-z material properties must contain T_{ult} in units of force, not t_{ult} in units of force/length. T_{ult} represents the integral of t_{ult} over the tributary length. The integration was performed numerically in the following manner:

1. z_top (coordinate at the top of the tributary length) and z_bot (coordinate at the bottom of the tributary length) were calculated based on the length of pile above and below the t-z element in contact with soil.
2. The tributary length was divided into 10 sublayers, each with thickness, dz.
3. T_{ult} was calculated as the integral from z_bot to z_top of t_{ult}*dz where t_{ult} was calculated at the center of each sublayer.

In the case when a the tributary length of a t-z element spans across a soil layer boundary, t_{ult} will be based on properties of both soil layers due to numerical integration over the tributary length. The type of t-z element (i.e. sand or clay) and z_{50} will be based on the material properties at the tz element location. In the case when a t-z element lies at a boundary between two soil layers, the element type will be based on the upper soil layer. The numerical integration of t_{ult} over the tributary length will reduce errors associated with interface effects, however the spacing of the t-z elements near layer interfaces should be small.

Error Checking

1. The program checks that all five (or six) files can be opened. If not, the program will issue a warning and return without writing data to the output file(s).

2. The program checks that MatType = "tz1", "tz2", "tz3" or "tz4". If not, the program issues the following warning: "Invalid MatType in TzSimple1Gen.", and then exits without writing data.

3. The program checks that the depth of each node lies within the depths specified in the soil properties file (File1). If not, a warning is issued and vertical stress is set to zero.

APPENDIX 3: REFERENCES

Boulanger, R. W. (2003). *The TzSimple1 Material.* http://opensees.berkeley.edu.

Mosher, R. L., (1984) "Load Transfer Criteria for Numerical Analysis of Axially Loaded Piles in Sand," US Army Engineering Waterways Experimental Station, Automatic Data Processing Center, Vicksburg, Mississippi, January.

Reese, L.C., and O'Neill, M. W., (1987) "Drilled Shafts: Construction Procedures and Design Methods," Report No. FHWA-HI-88-042, U.S. Department of Transportation, Federal Highway Administration, Office of Implementation, McLean, Virginia.