



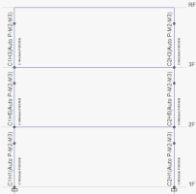
MULTI-CUT REBAR(10)

Advisor : Prof. K.C.Chang

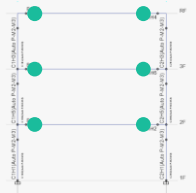
Presenters : You-Ran Nai

Nonlinear Hinge

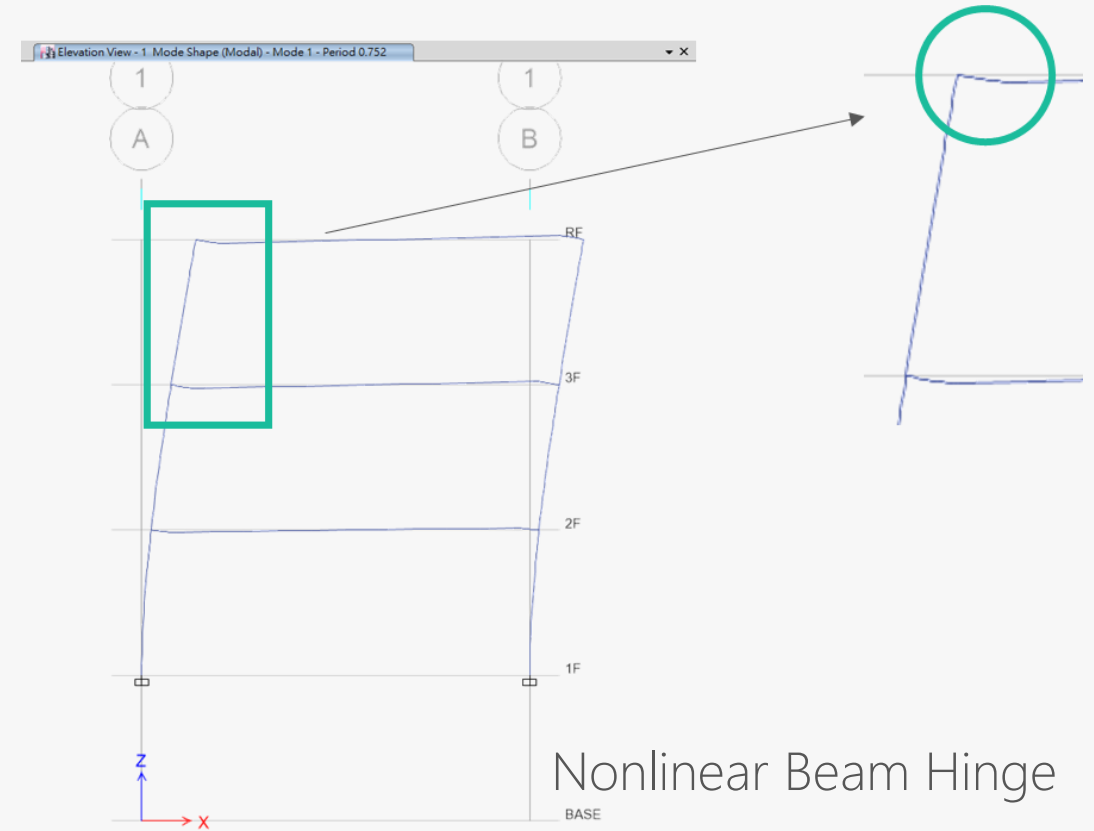
ETABS 2016



No Nonlinear Beam Hinge \longrightarrow $T=0.345$



Nonlinear Beam Hinge \longrightarrow $T=0.752$



Nonlinear Hinge

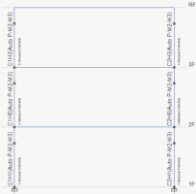
Static Nonlinear

Fast Nonlinear
Analysis (FNA)
Modal Time
History Analysis

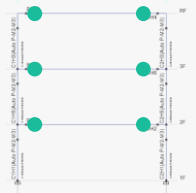
Nonlinear Direct
Integration Time
History

P-Delta ETABS 2016

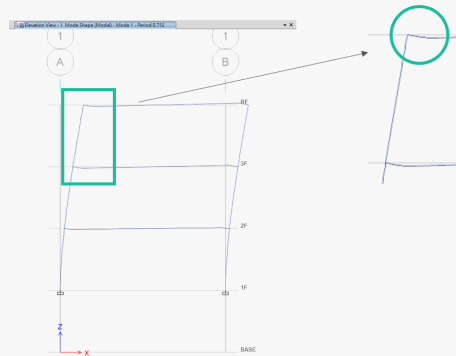
P-Delta



No Nonlinear Beam Hinge \longrightarrow $T=0.345$



Nonlinear Beam Hinge \longrightarrow $T=0.752$



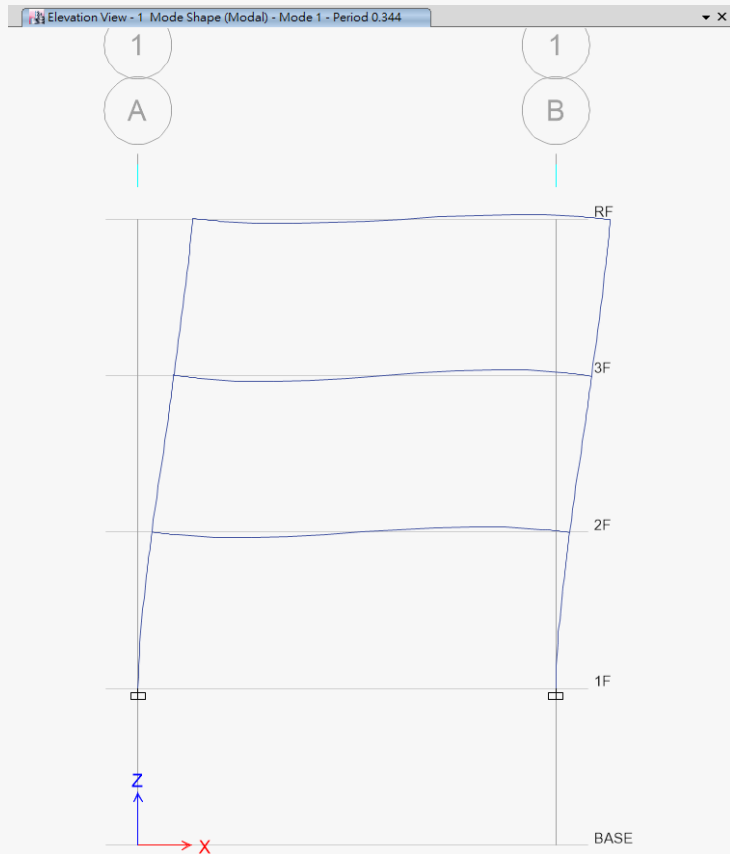
No P-Delta

No Nonlinear Beam Hinge

Nonlinear Beam Hinge

$T=0.344$

P-Delta



1. **Non-iterative Based on Mass**, in which load is automatically computed from the mass at each level. This providing for faster computation. P-Delta is considered by treating the structure as a simplified stick model. Local buckling is not captured as effectively.

The benefit of this non-iterative method is that P-Delta may be considered in load cases which do not support the Iterative Based on Load Cases method.

2. **Iterative Based on Load Cases**, in which load is computed from a specified combination of static load cases which considers P-Delta on an element-by-element basis. Local buckling is captured more effectively. A fraction of a live load case.

Automation Method

☐ None

☐ Non-iterative - Based on Mass

☒ Iterative - Based on Loads

Iterative P-Delta Load Case

Load Pattern	Scale Factor
DL	1
DL	1

Add
Modify
Delete

Relative Convergence Tolerance 0.0001

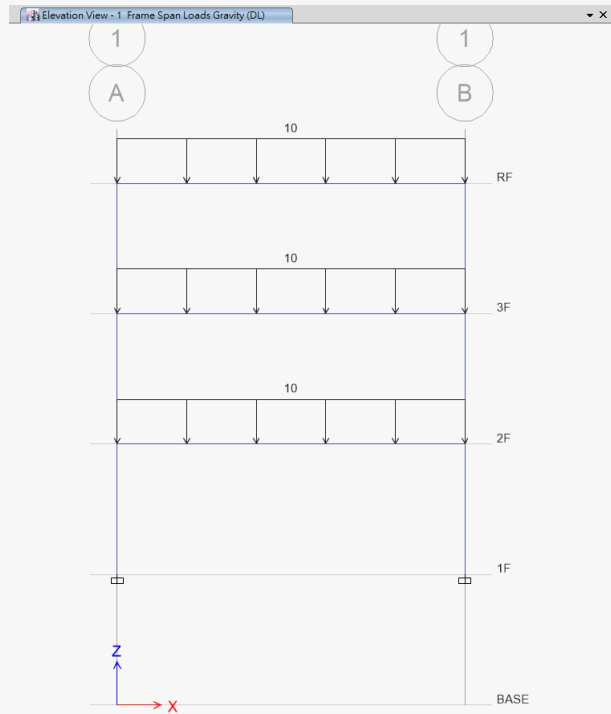
OK Cancel

Local Buckling

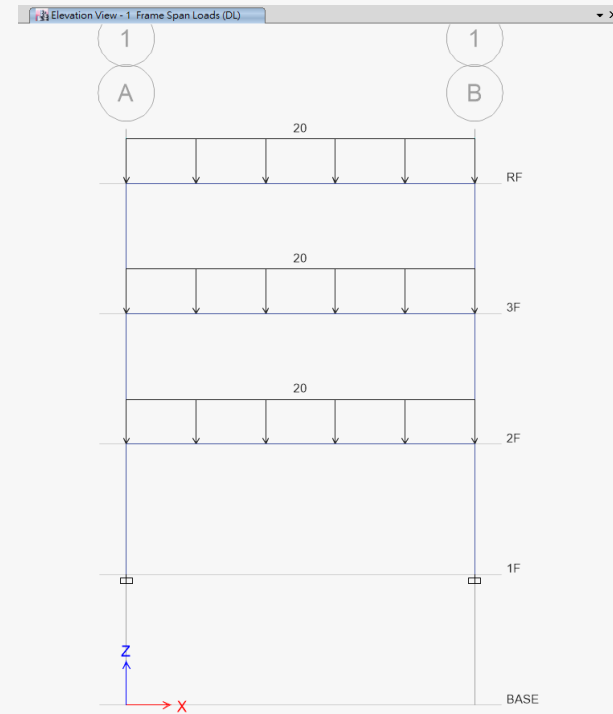
RC → SS

P-Delta_{Steel Structure}

10 ton/m \longrightarrow Same Period



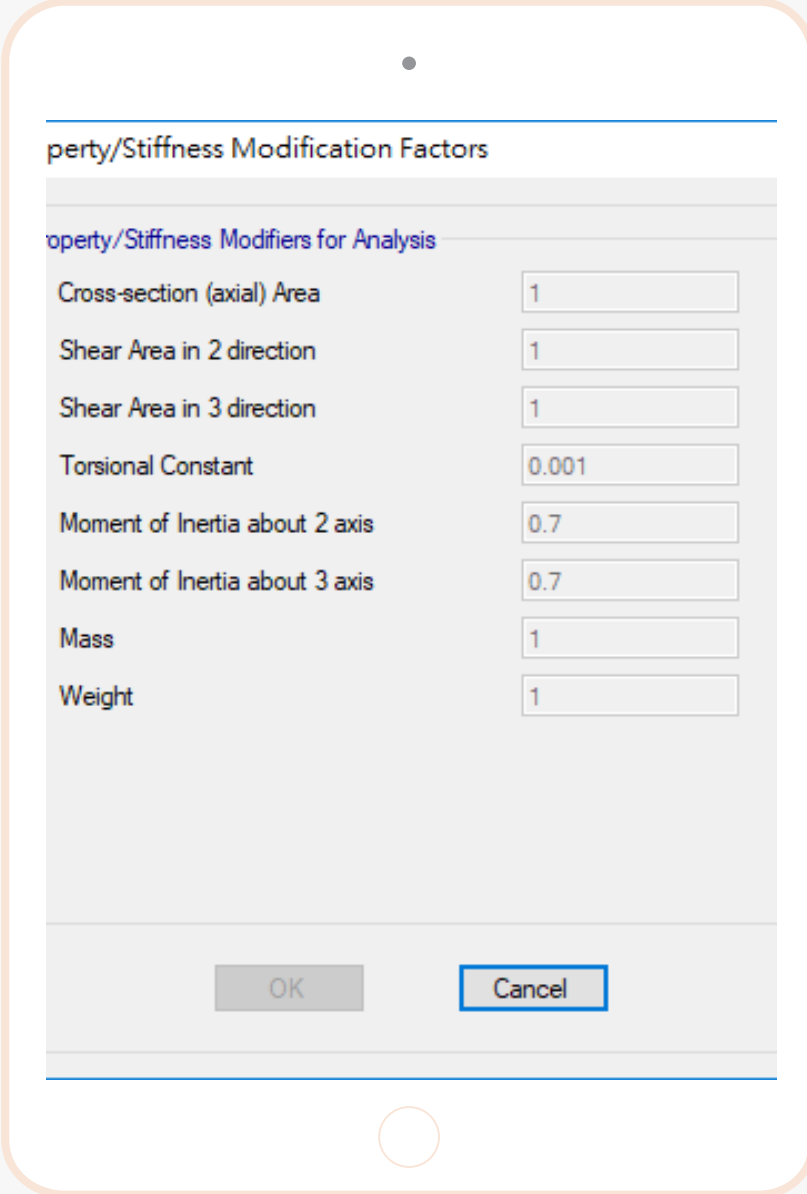
20 ton/m \longrightarrow Different Period



0.5, 0.7 Moment of Inertia



Same Problem



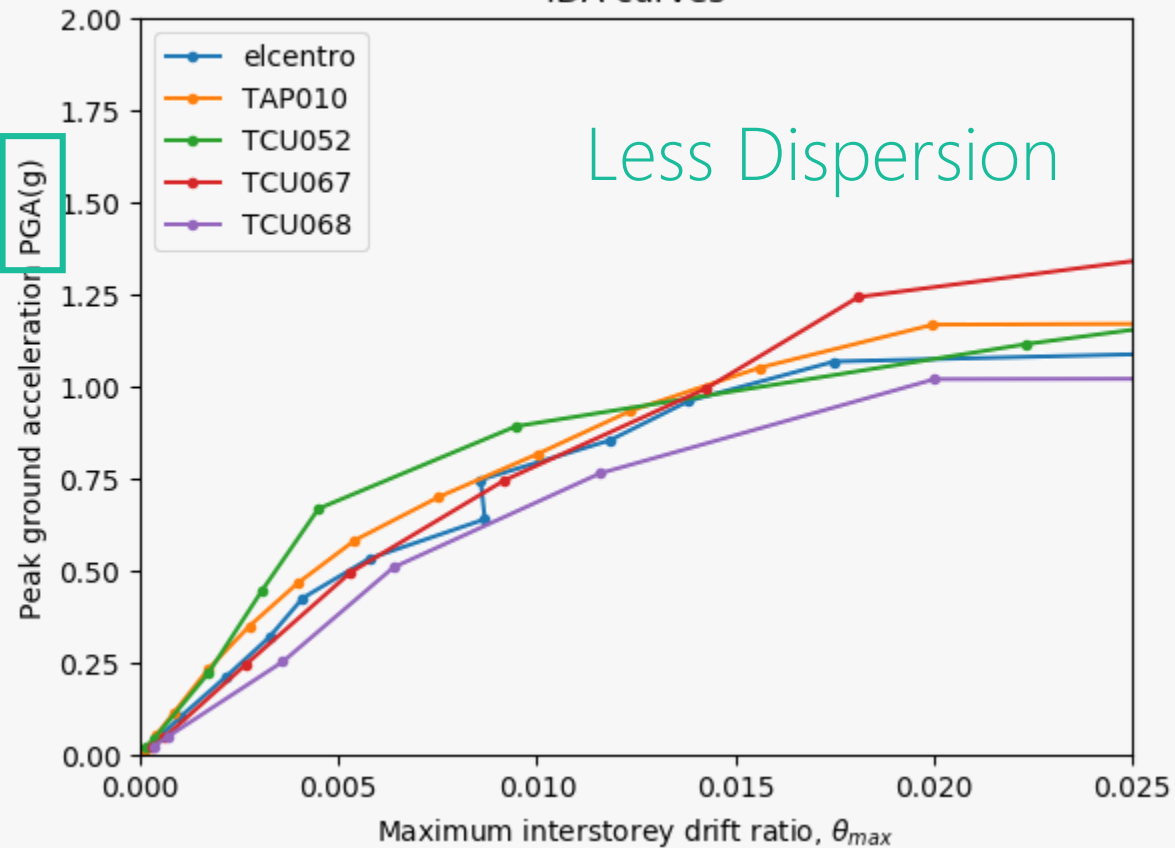
The image shows a software dialog box titled "Property/Stiffness Modification Factors". It contains a section titled "Property/Stiffness Modifiers for Analysis" with several input fields. The values entered in the fields are: Cross-section (axial) Area: 1, Shear Area in 2 direction: 1, Shear Area in 3 direction: 1, Torsional Constant: 0.001, Moment of Inertia about 2 axis: 0.7, Moment of Inertia about 3 axis: 0.7, Mass: 1, and Weight: 1. At the bottom of the dialog are "OK" and "Cancel" buttons. The dialog is presented on a tablet-like device with a home button at the bottom.

Property/Stiffness Modification Factors	
Property/Stiffness Modifiers for Analysis	
Cross-section (axial) Area	1
Shear Area in 2 direction	1
Shear Area in 3 direction	1
Torsional Constant	0.001
Moment of Inertia about 2 axis	0.7
Moment of Inertia about 3 axis	0.7
Mass	1
Weight	1

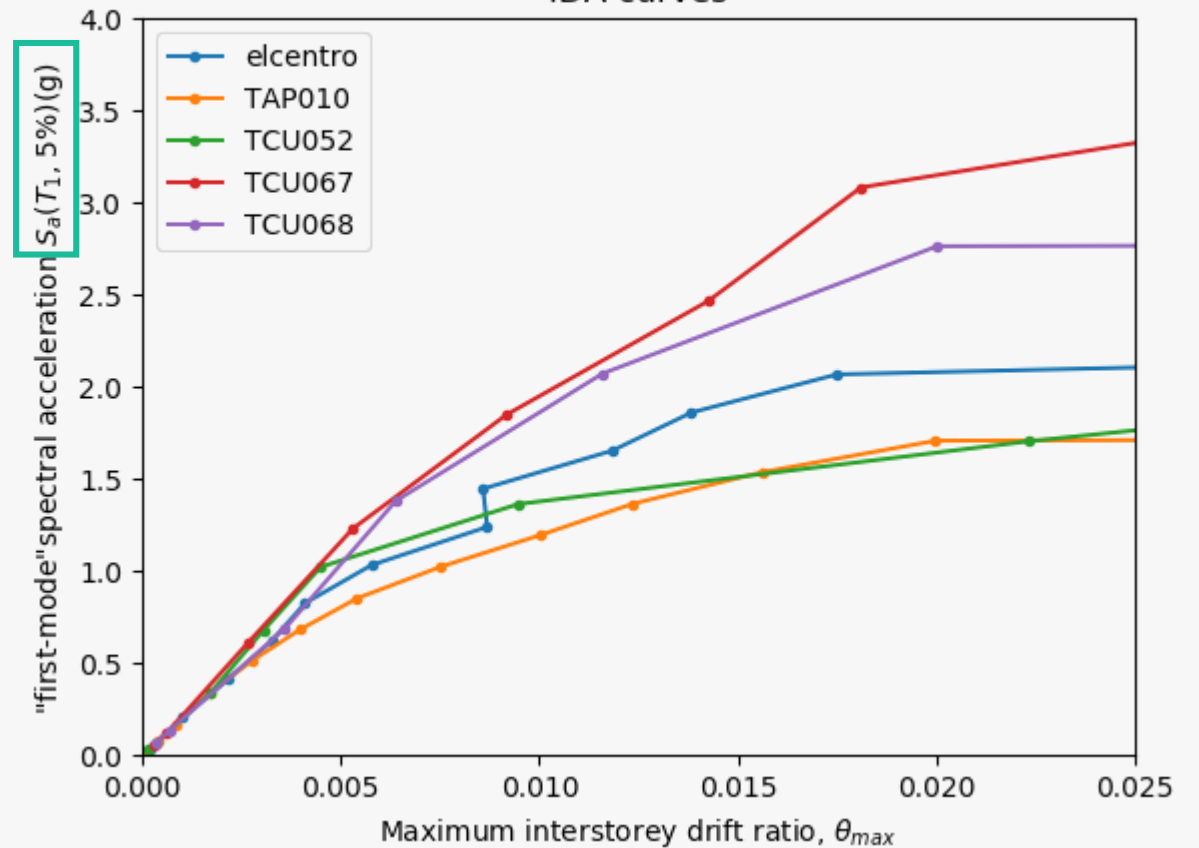
PGA versus $S_a(T_1, 5\%)$



IDA curves

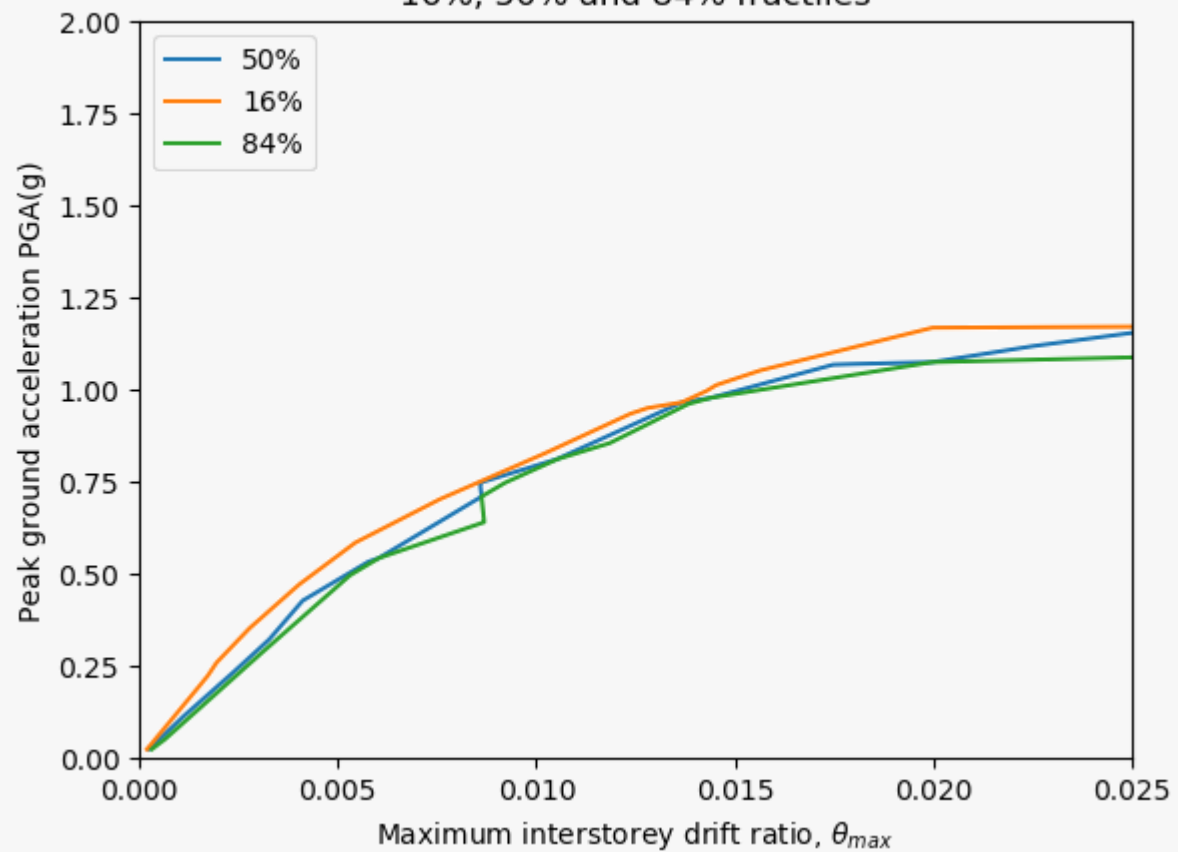


IDA curves



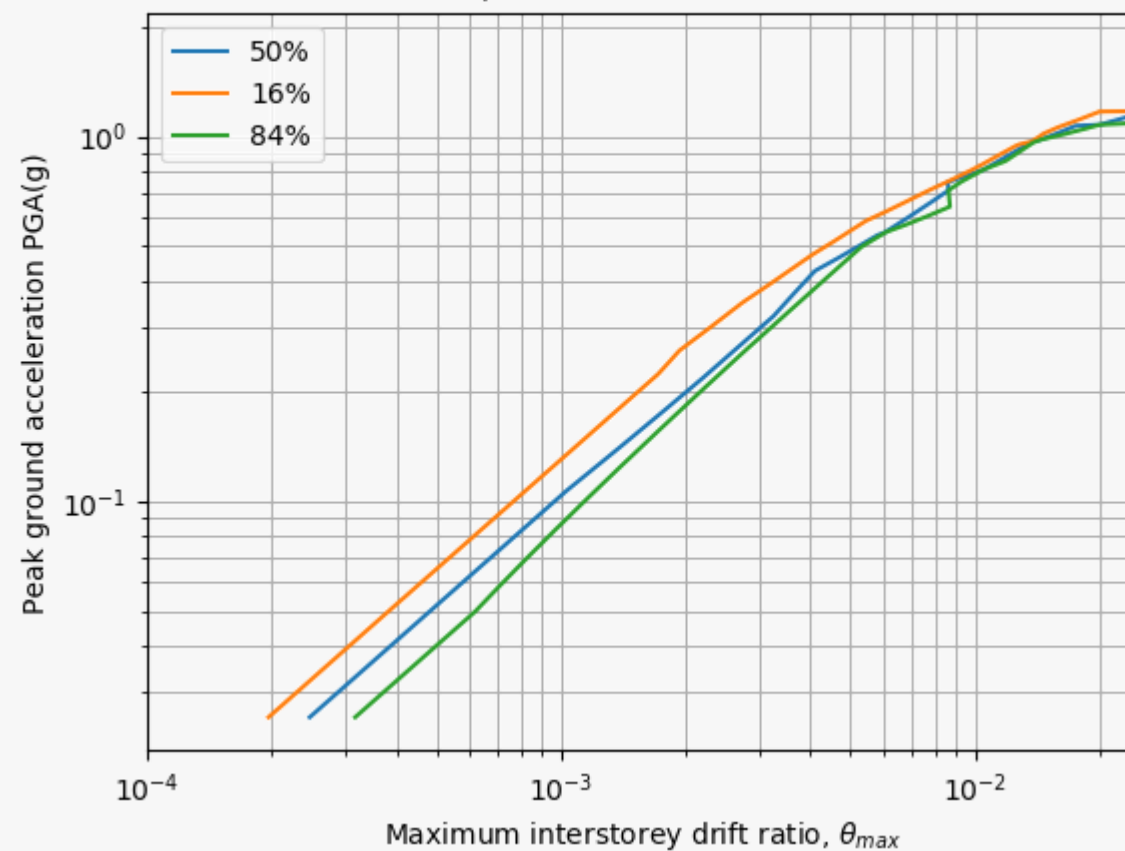
Linear

16%, 50% and 84% fractiles

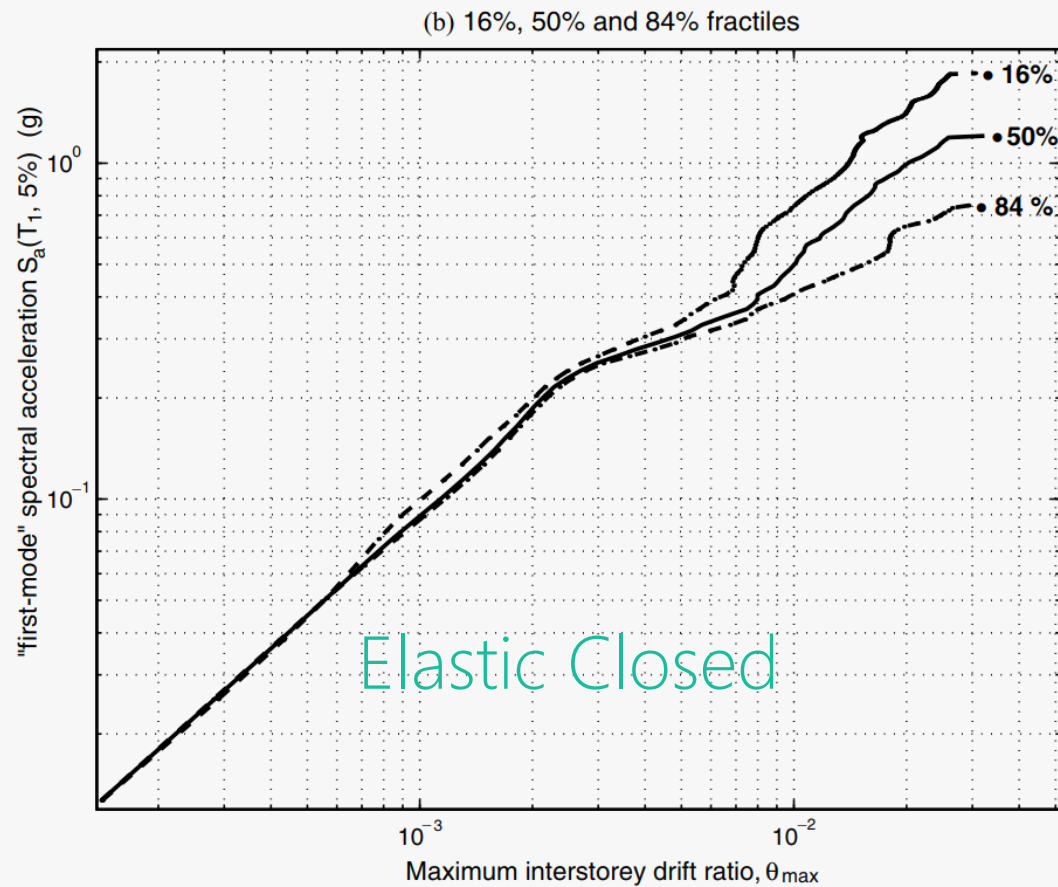


Log

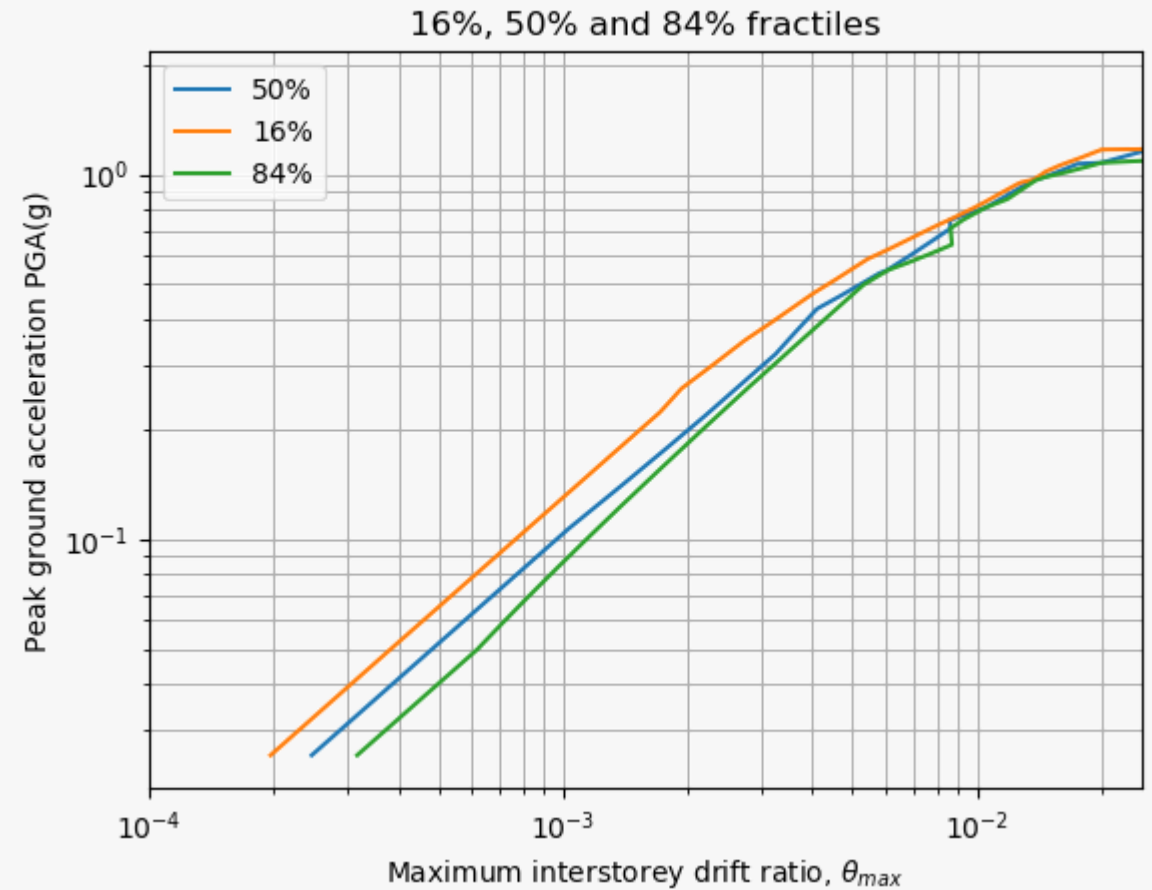
16%, 50% and 84% fractiles

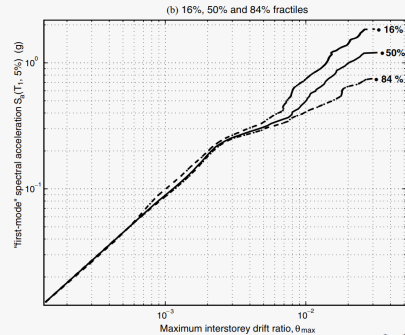


5-storey steel braced frame

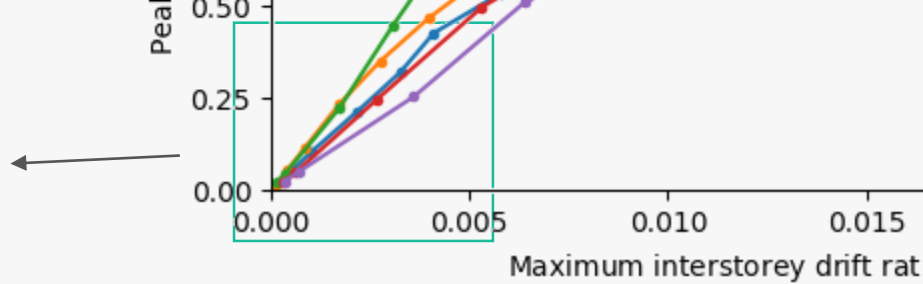
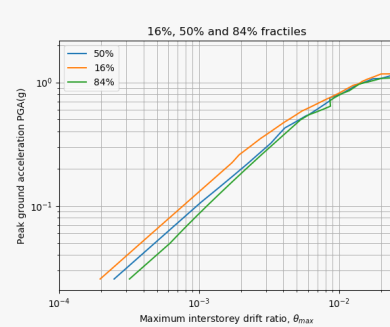
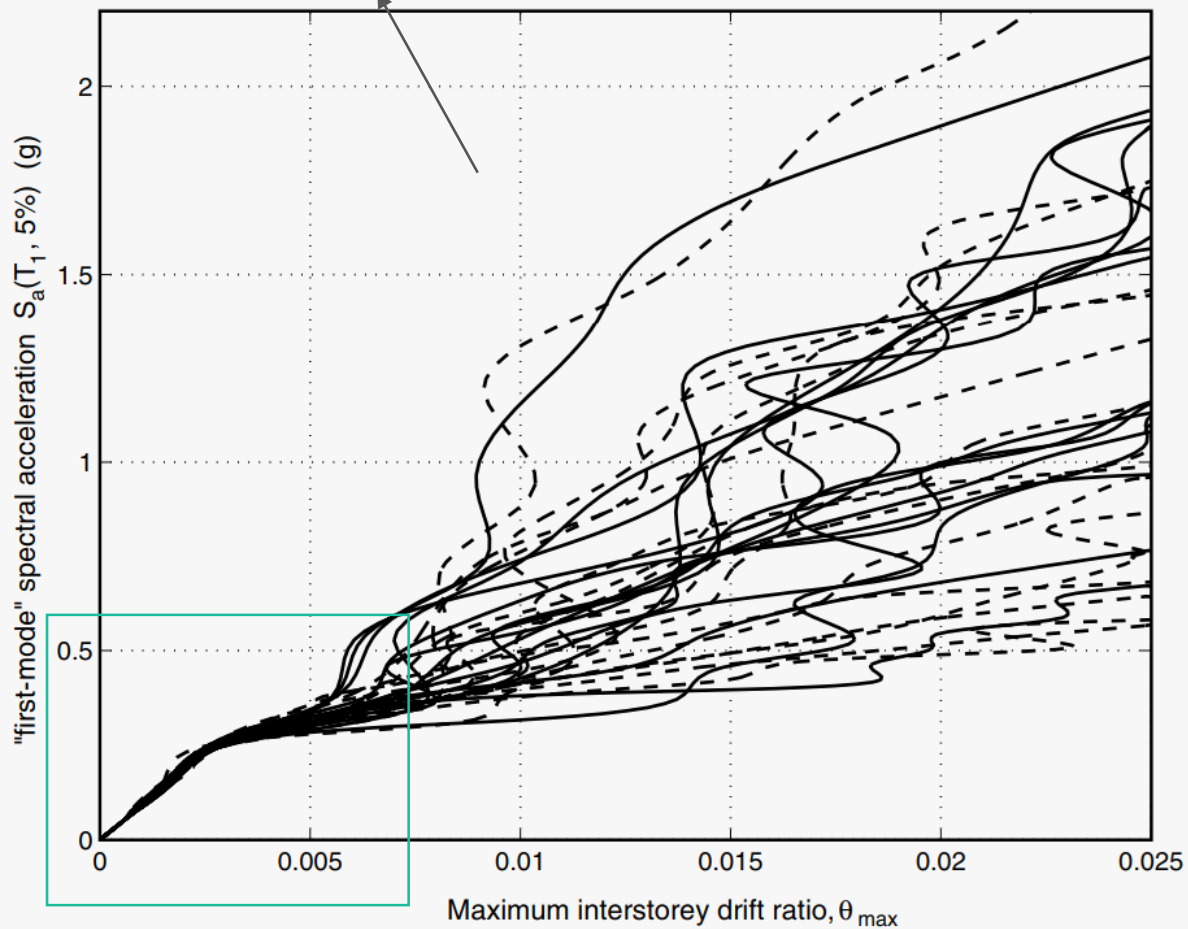


3-storey rc moment resist frame

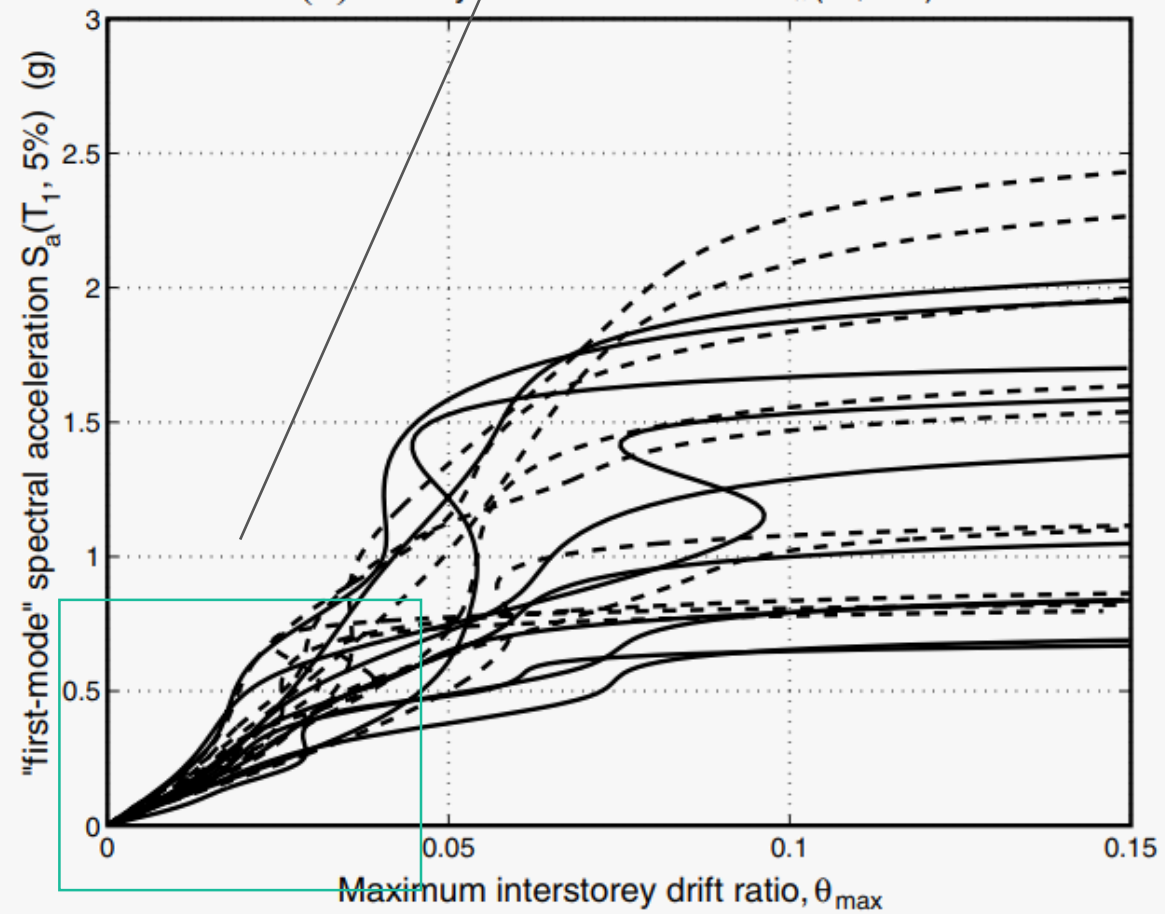




(a) Thirty IDA curves



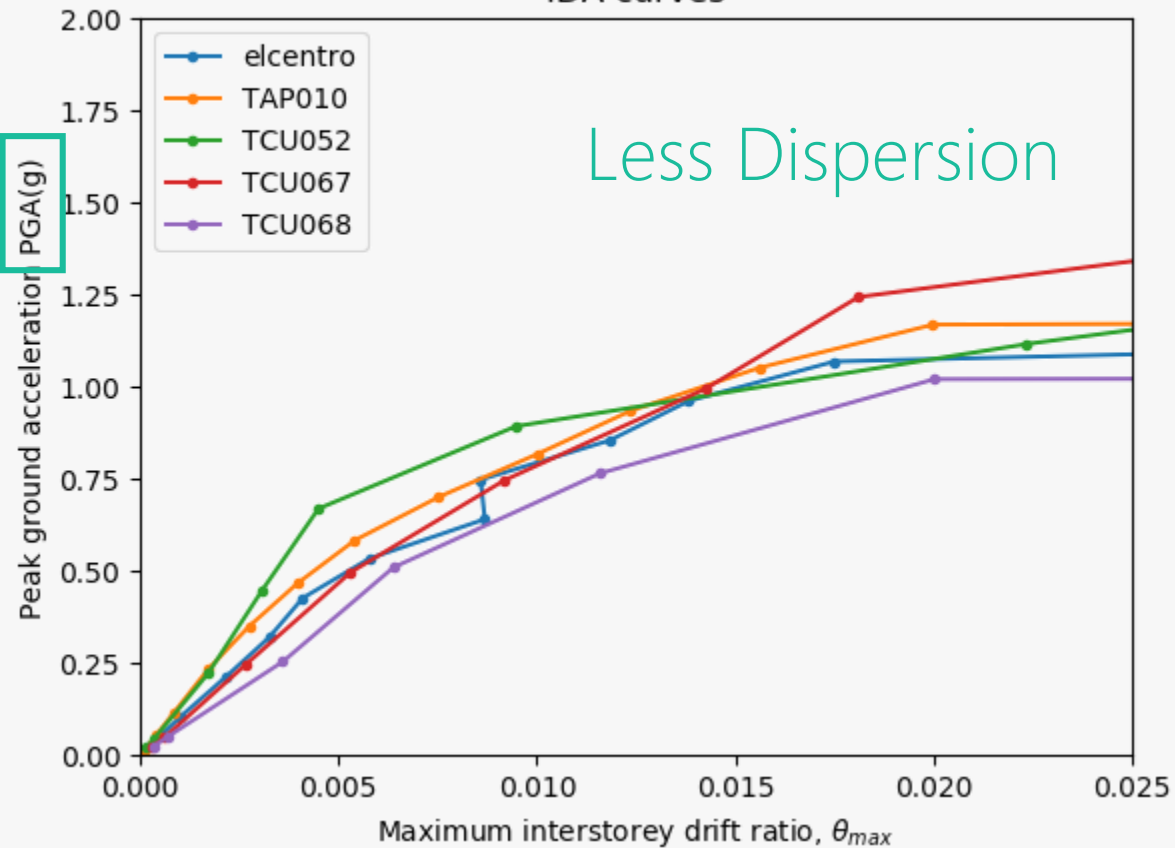
(b) Twenty IDA curves versus $S_a(T_1, 5\%)$



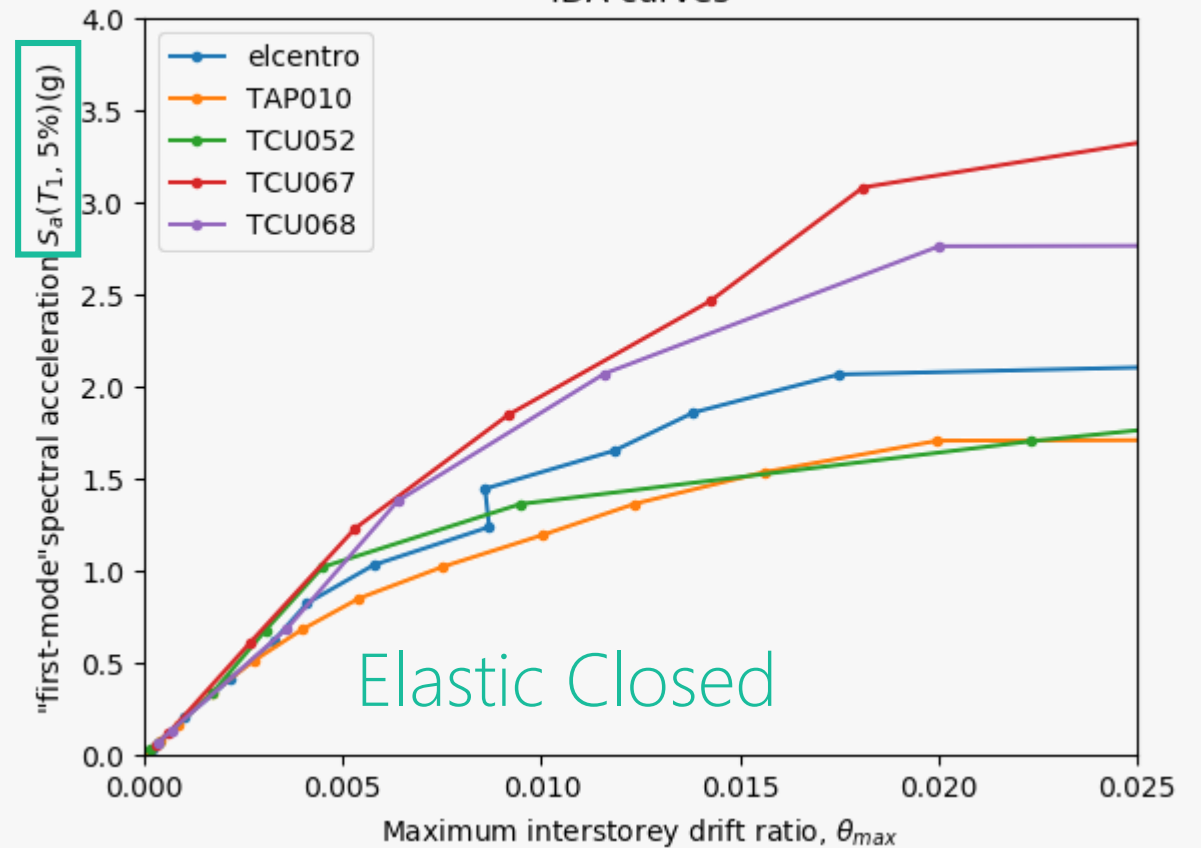
PGA versus $S_a(T_1, 5\%)$



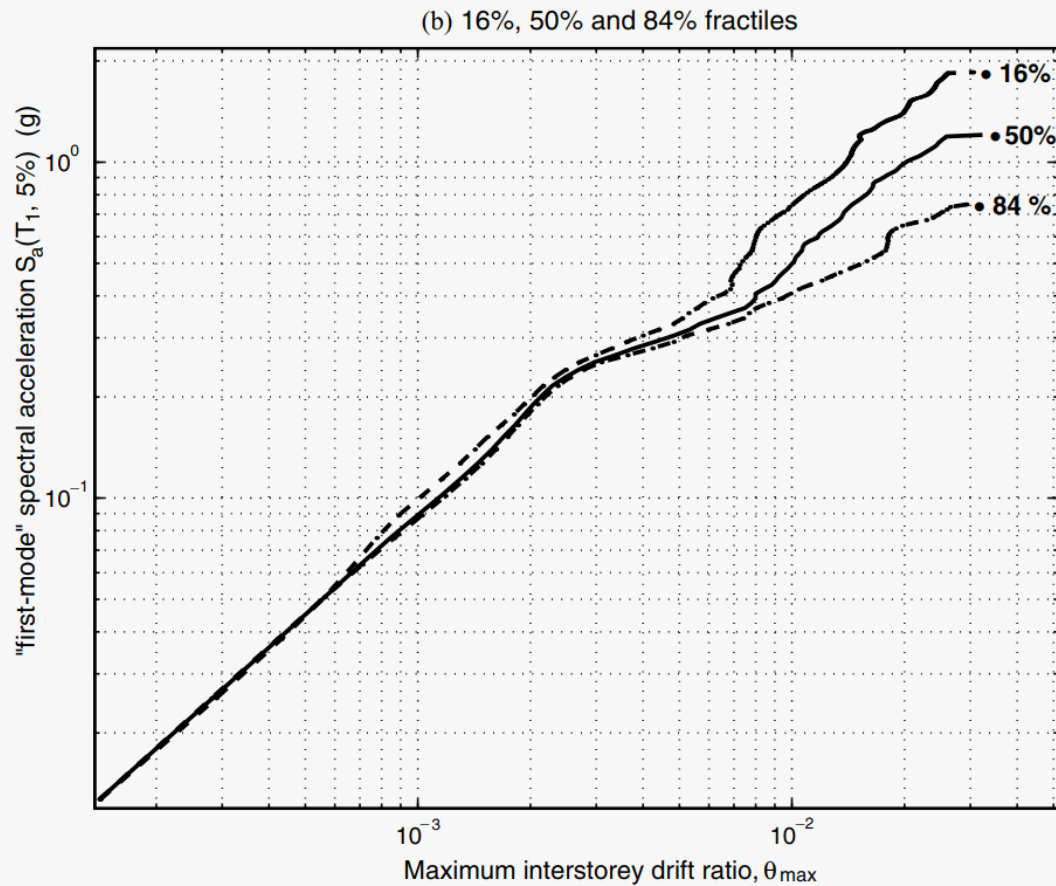
IDA curves



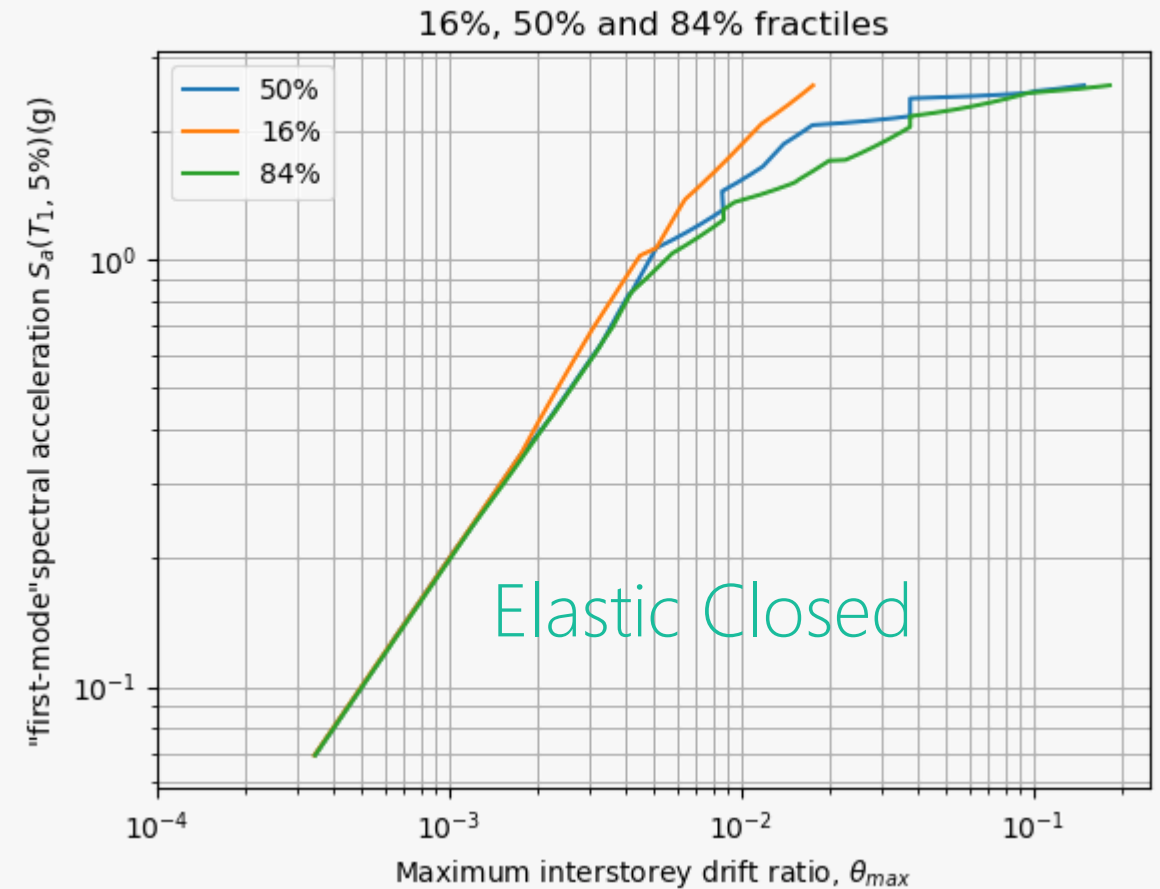
IDA curves



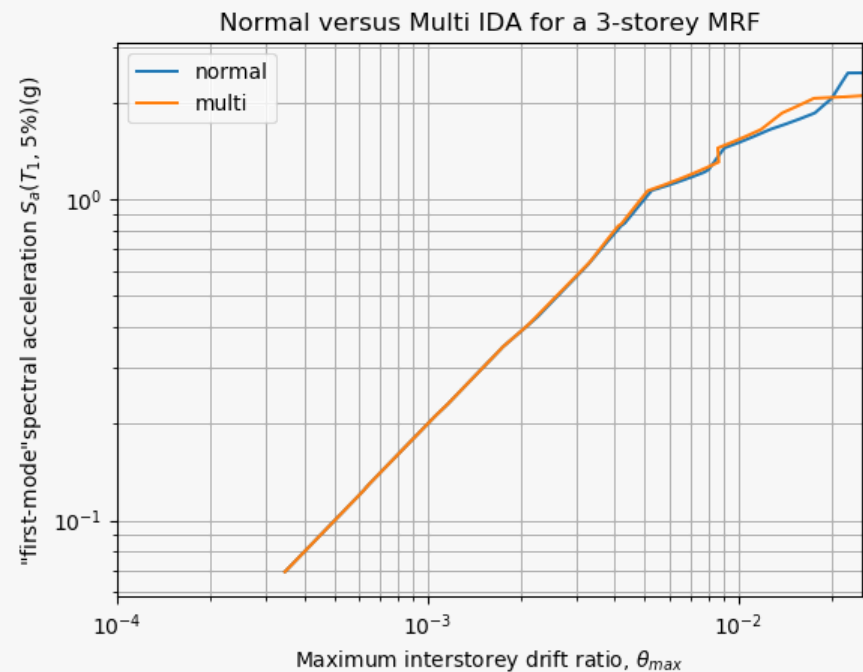
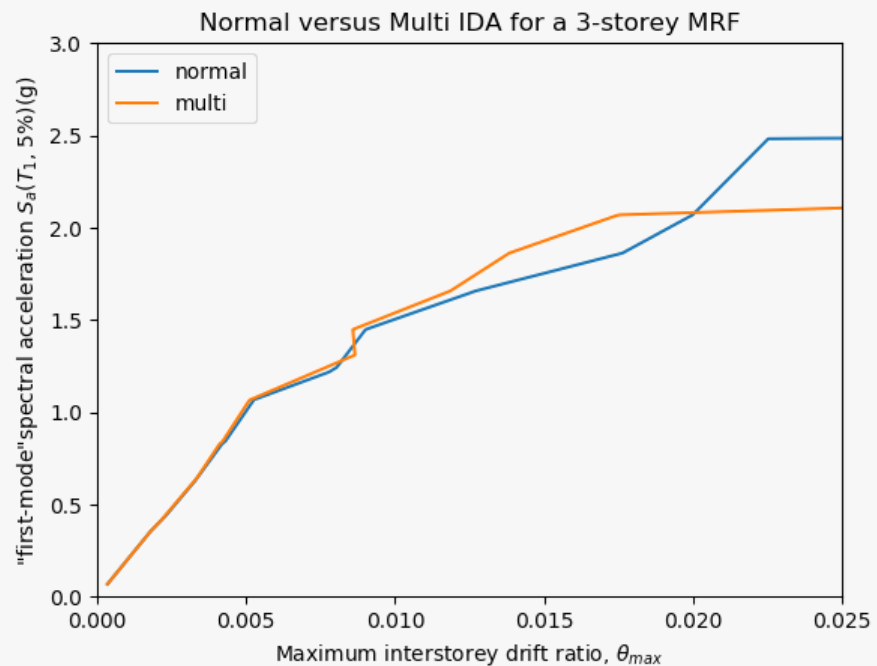
5-storey steel braced frame



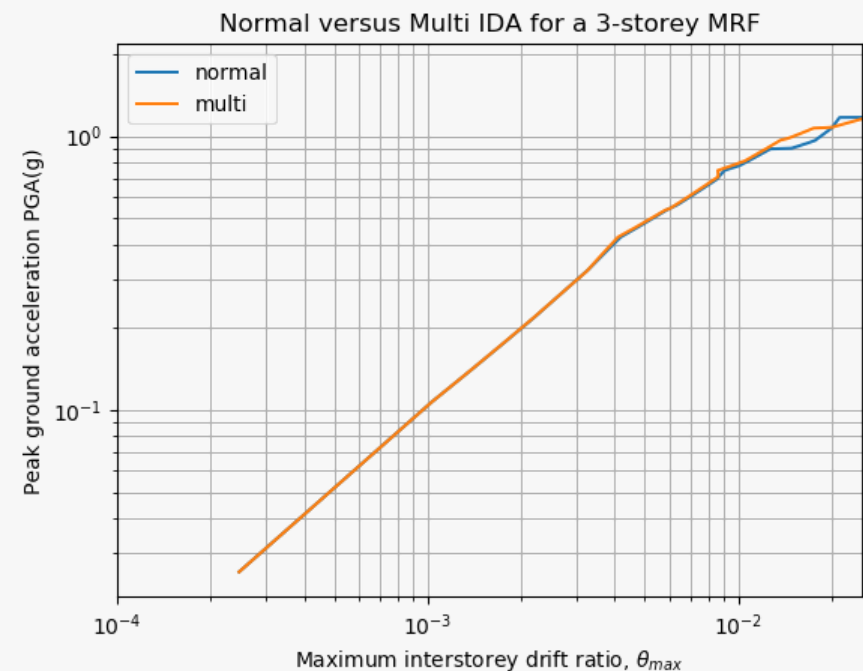
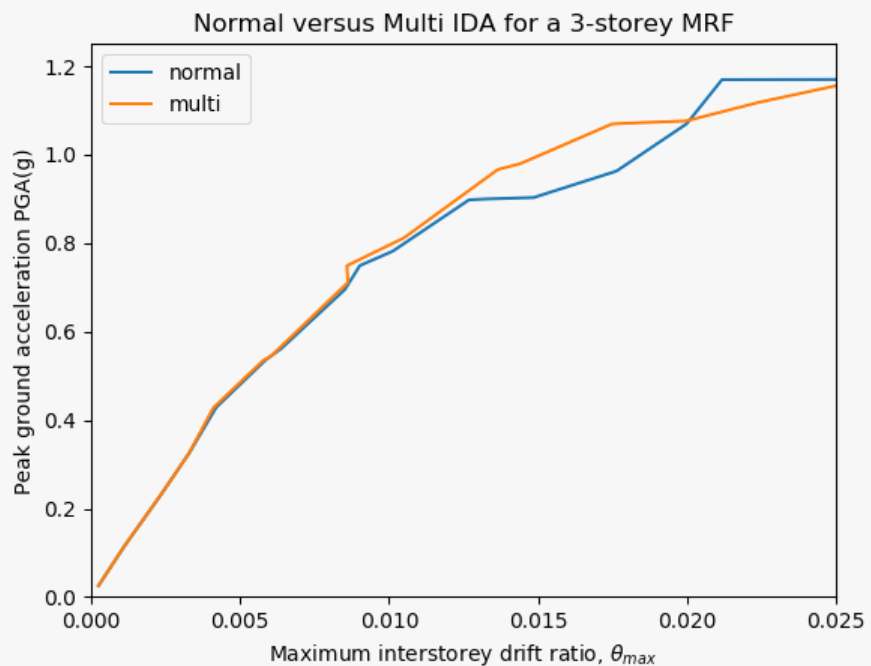
3-storey rc moment resist frame



$S_a(T_1, 5\%)$

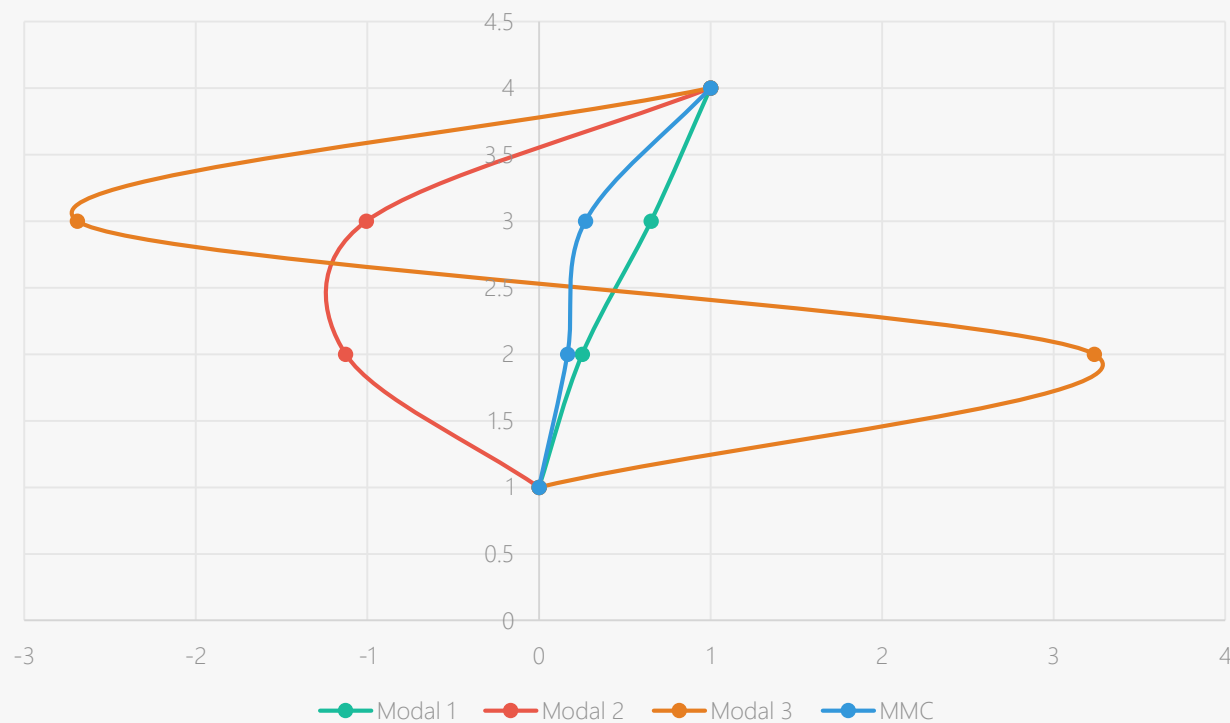


PGA



Pushover

Mode Shape

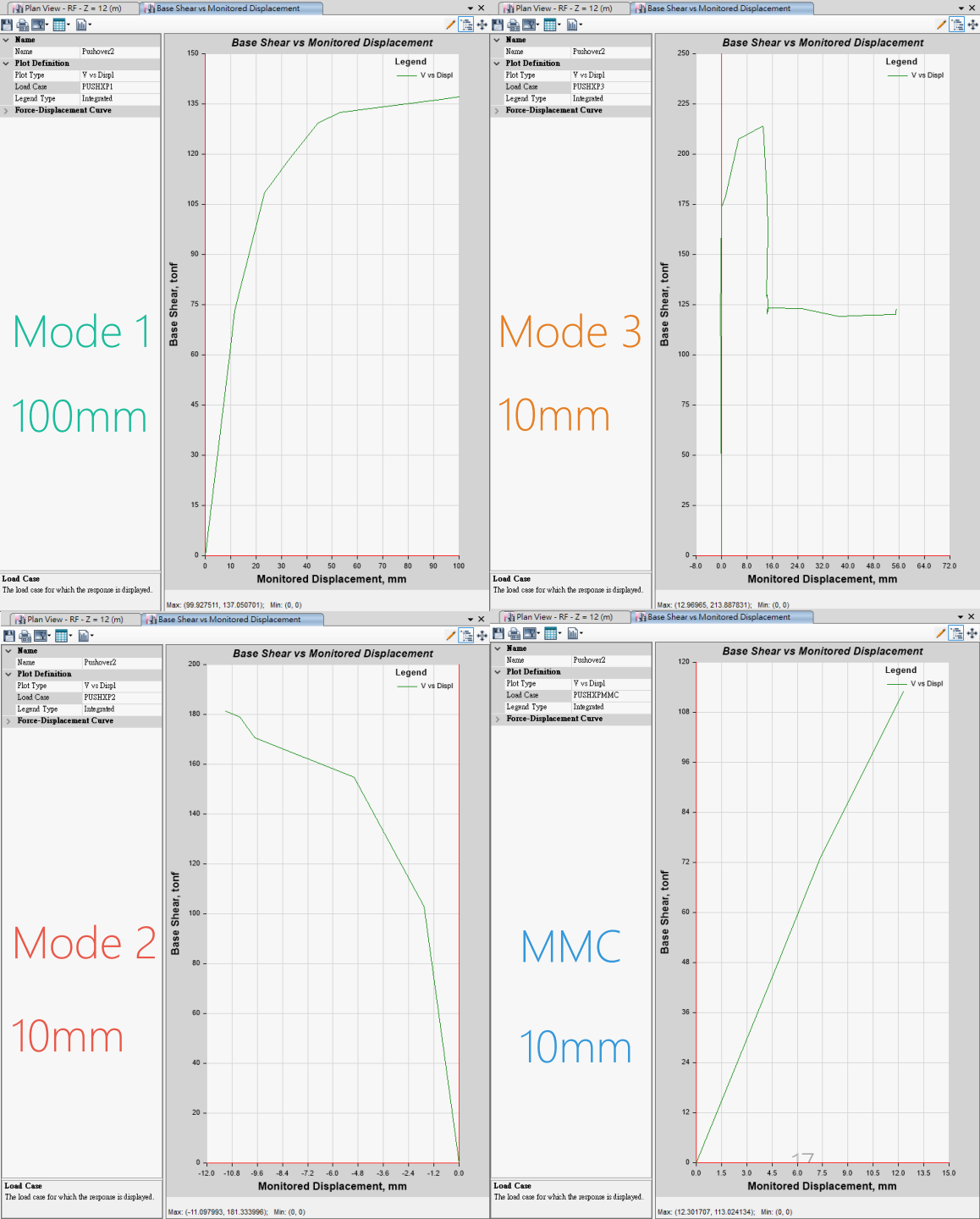


Mode 1
100mm

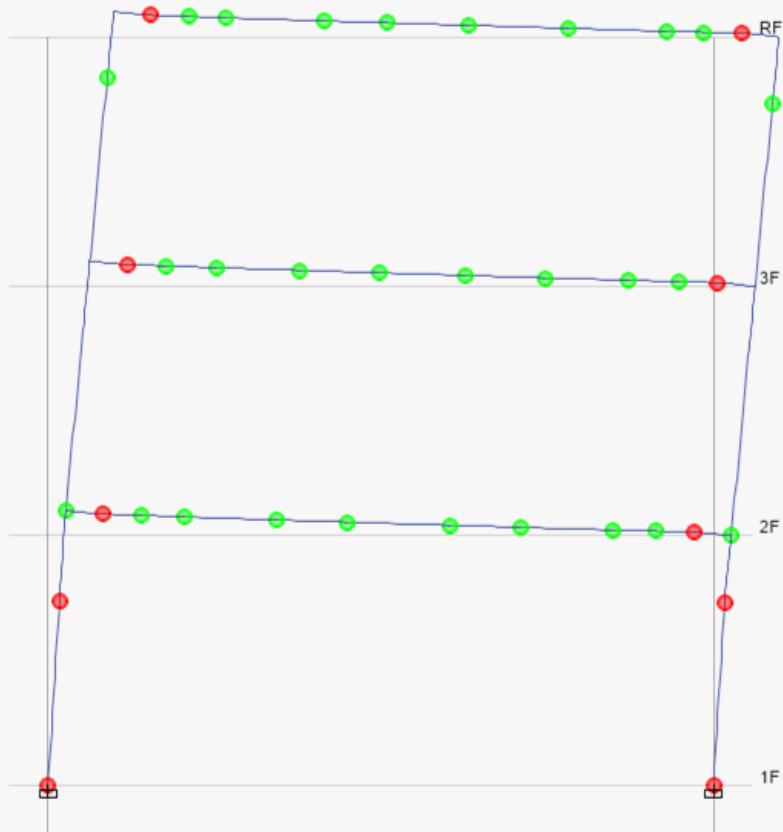
Mode 3
10mm

Mode 2
10mm

MMC
10mm



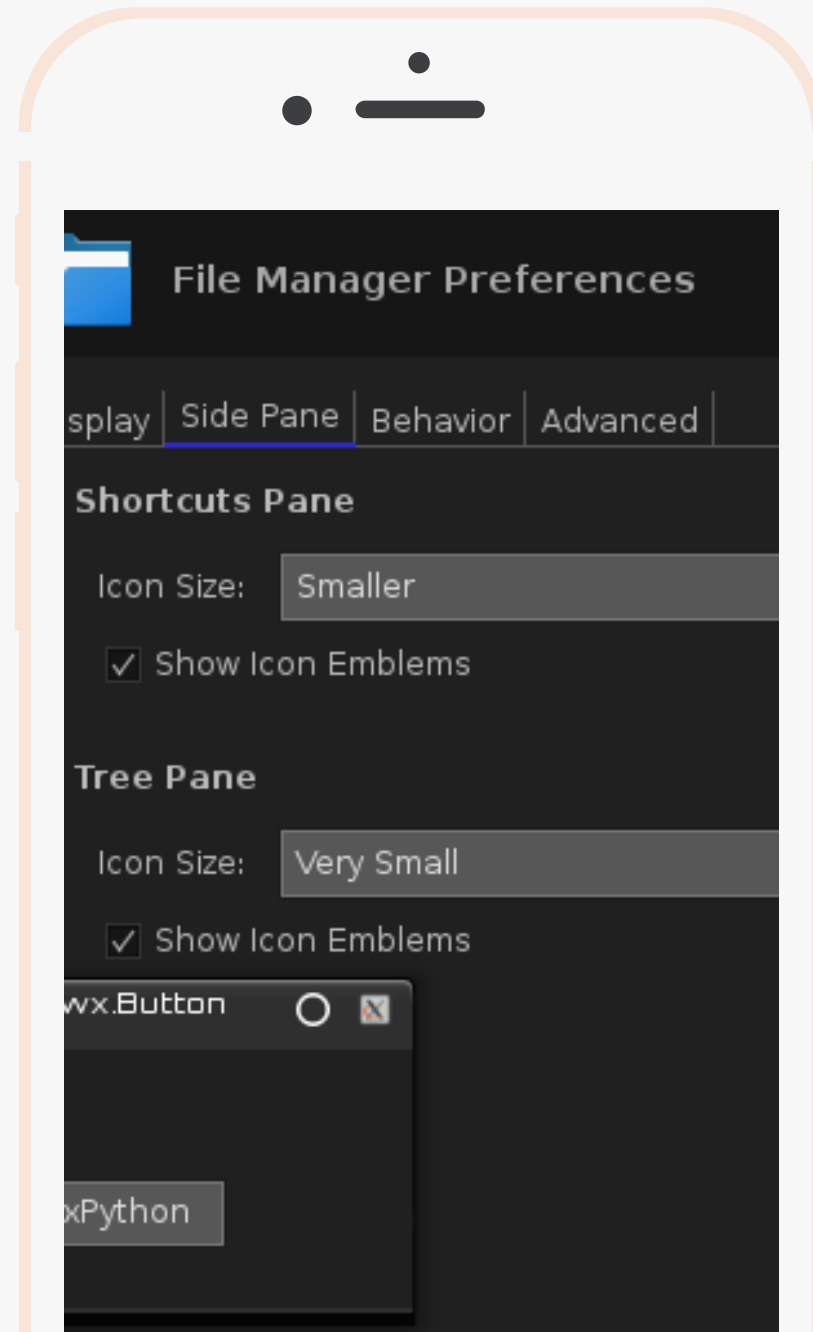
Time History



Pushover



LinearCut GUI



Attaching to a Manually Started Instance of ETABS

These instructions document the steps necessary to attach to instances of ETABS that were started manually. The example code is in VB.NET.

1. Start ETABS manually, e.g. by clicking on the program shortcut
2. Instead of creating a new `ETABSObject`, attach to the instance of the `ETABSObject` that was created and added to the Running Object Table by ETABS.

The code below creates an `ETABSObject` and starts the program:

```
VB
Dim myETABSObject As ETABS2016.cOAPI
Dim myHelper as ETABS2016.cHelper = New ETABS2016.Helper

myETABSObject = myHelper.CreateObject("C:\Program Files (x86)\Computers and Structures\ETABS 2016\ETABS.exe")

ret = myETABSObject.ApplicationStart()

Dim mySapModel As ETABS2016.cSapModel = myETABSObject.SapModel
```

Copy

Replace the code above with the code below to attach to the existing `ETABSObject`. Since the program is already started, there is no need to call `ETABSObject.ApplicationStart`:

```
VB
Dim myETABSObject As ETABS2016.cOAPI

myETABSObject = DirectCast(GetObject( , "CSI.ETABS.API.ETABSObject"), ETABS2016.cOAPI)

Dim mySapModel As ETABS2016.cSapModel = myETABSObject.SapModel
```

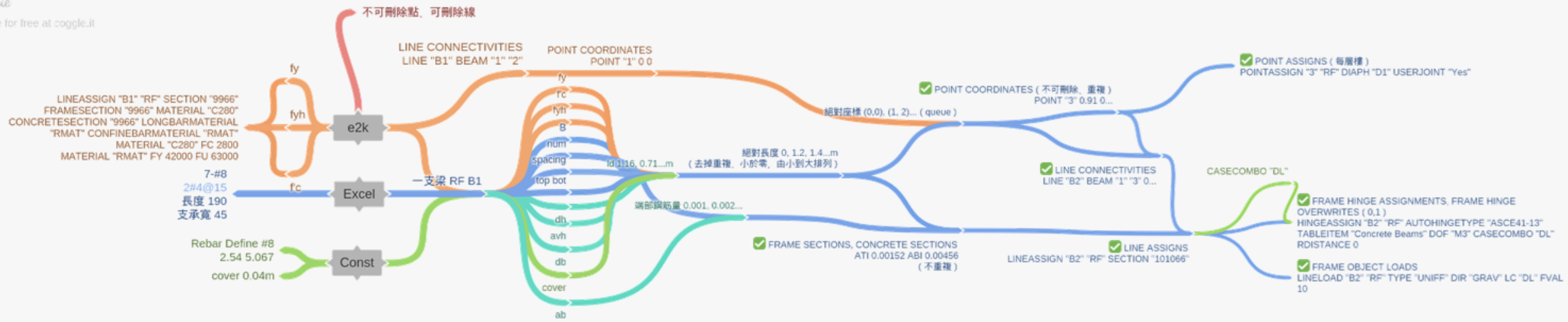
Copy

3. Other calls can then be made normally. The program can be ended manually or by calling `ETABSObject.ApplicationExit`
4. Please note that if multiple instances of ETABS are manually started, an API client can only attach to the instance that was started first.

E2K versus API

coggle

made for free at coggle.it



Roadmap

coggle

made for free at coggle.it

