National Science Foundation

University of California at San Diego
NHERI
Natural Hazards Engineering Research Infrastructure

UC San Diego
JACOBS SCHOOL OF ENGINEERING
Structural Engineering

Example Use of Research Tools

Adam Zsarnóczay, Postdoctoral Researcher

Stanford University, SimCenter

adamzs@stanford.edu

Joint Researcher Workshop
UC San Diego, Lehigh & SimCenter

December 16-17, 2019
University of California, San Diego

LEHIGH NHERI
Real-Time Multi-Directional Testing Facility

University of California at San Diego
NHERI
Natural Hazards Engineering Research Infrastructure

SIMCENTER
COMPUTATIONAL MODELING AND SIMULATION CENTER

Objectives

- Demonstrate use-cases of our research tools
- Get feedback on existing features
- Collect requests for new features

Research Tools

- **quoFEM** **Uncertainty Quantification and Optimization**

- **EE-UQ** **Structural Response Estimation under Earthquakes**
- WE-UQ Structural Response Estimation under Wind
- CWE CFD Analysis for a Building

- **PBE** **Damage and Loss Assessment for a Building**
- rWHALE Damage and Loss Assessment for an Urban Region

Case study

example application: Buckling Restrained Braced Frames

1. Design an experiment	quoFEM
2. Calibrate a numerical component model	quoFEM
3. Simulate structural response	EE-UQ
4. Estimate damage and losses	PBE

Case study

example application: Buckling Restrained Braced Frames

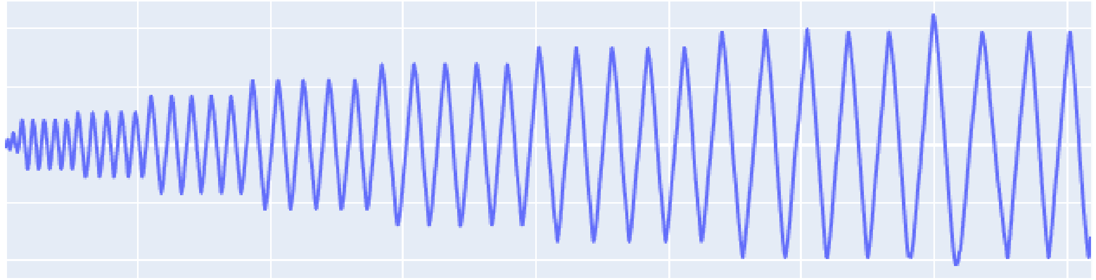
1. Design an experiment	**quoFEM**
2. Calibrate a numerical component model	quoFEM
3. Simulate structural response	EE-UQ
4. Estimate damage and losses	PBE

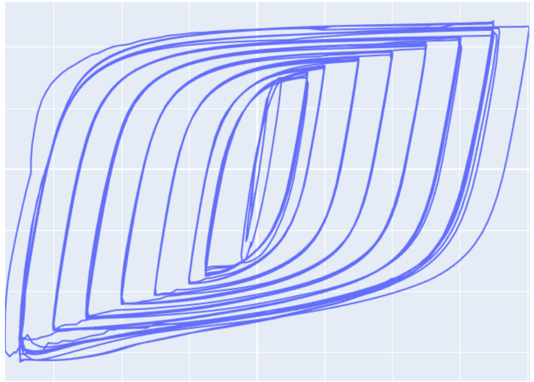
Design an experiment

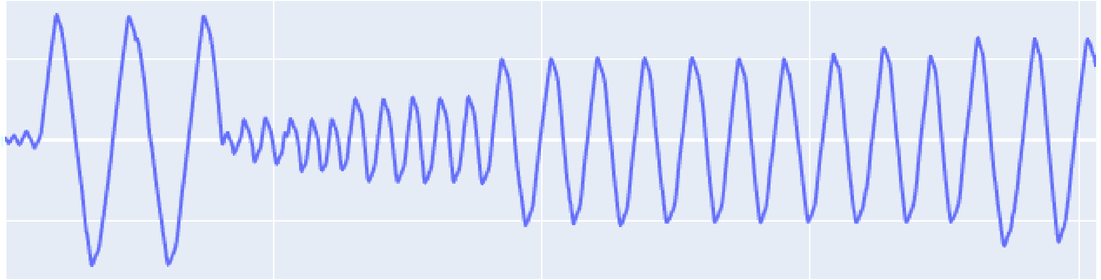
setting: uniaxial cyclic load test of a Buckling Restrained Brace

objective: estimate maximum tension/compression during test

problem: some attributes of the specimen are not known
new load protocol



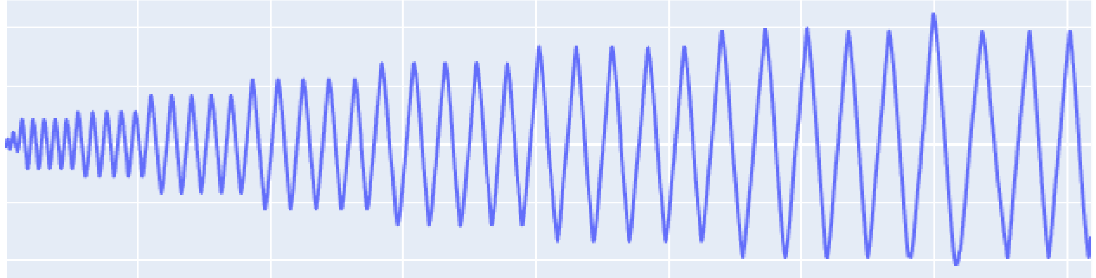


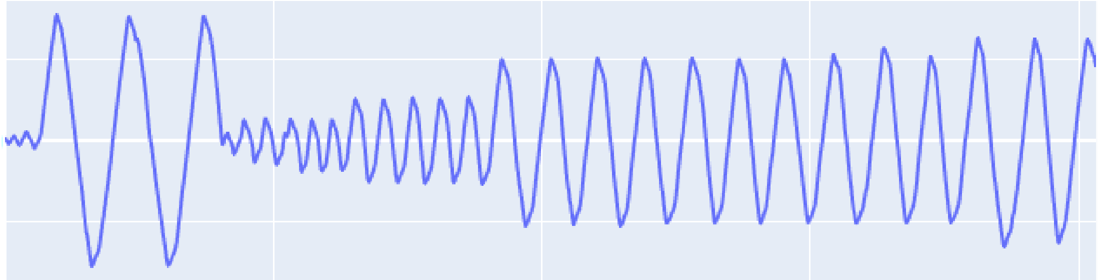


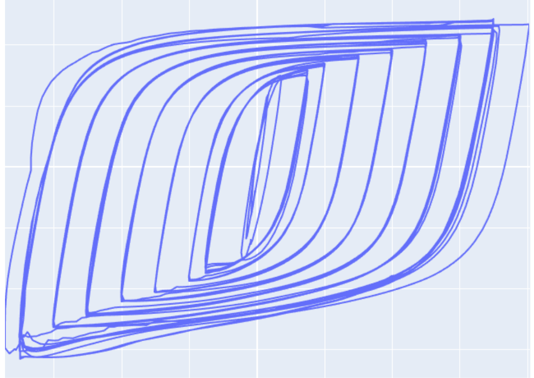
?

Design an experiment

**quoFEM: sample the joint distribution of uncertain attributes
simulate the experiment; estimate max loads**







?

Design an experiment

First, through literature review get:

- conservative bounds for attributes
- simplified BRB model in OpenSees

Then, prepare:

- new load protocol
- script to simulate experiment
- script to extract max forces

Adam Zsarnóczay adamzs@stanford.edu

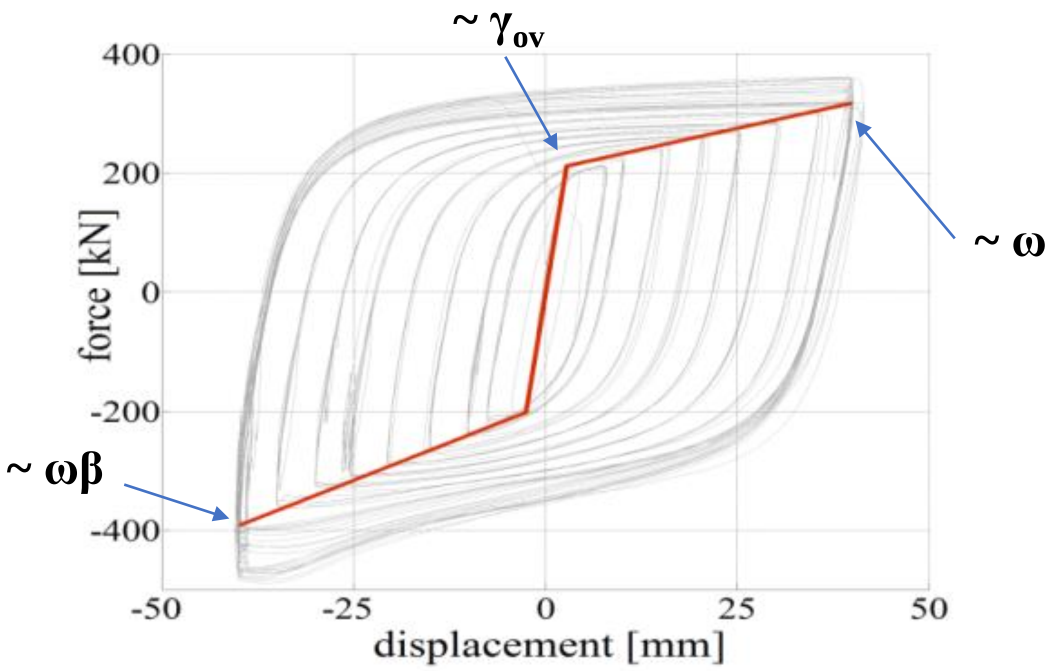
Design an experiment

First, through literature review get:

- **conservative bounds for attributes**
- simplified BRB model in OpenSees

Then, prepare:

- new load protocol
- script to simulate experiment
- script to extract max forces

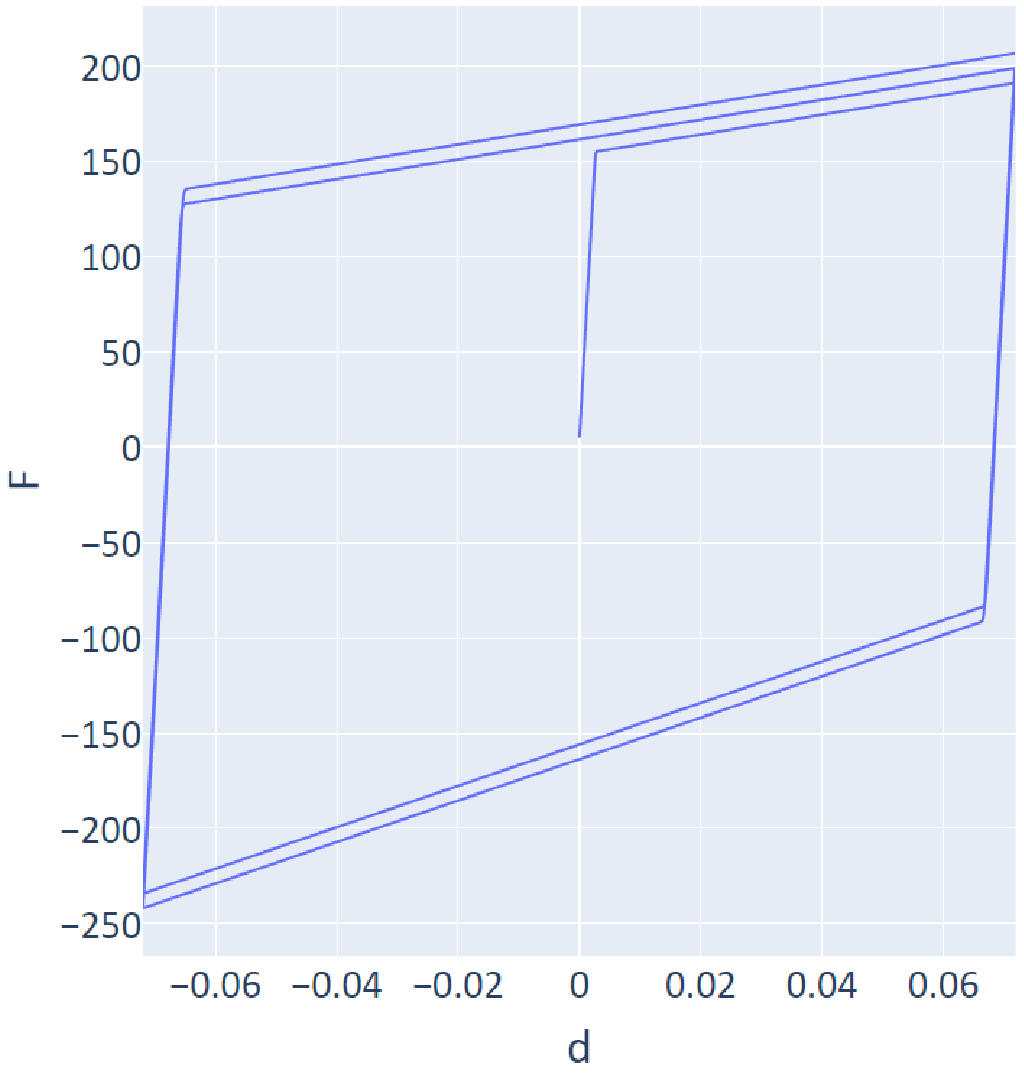


γ_{ov}	1.10 – 1.20
ω	1.35 – 1.75
β	1.01 – 1.30
ω_{iso}	0.05 – 0.30

Design an experiment

First, through literature review get:
- conservative bounds for attributes
- **simplified BRB model in OpenSees**

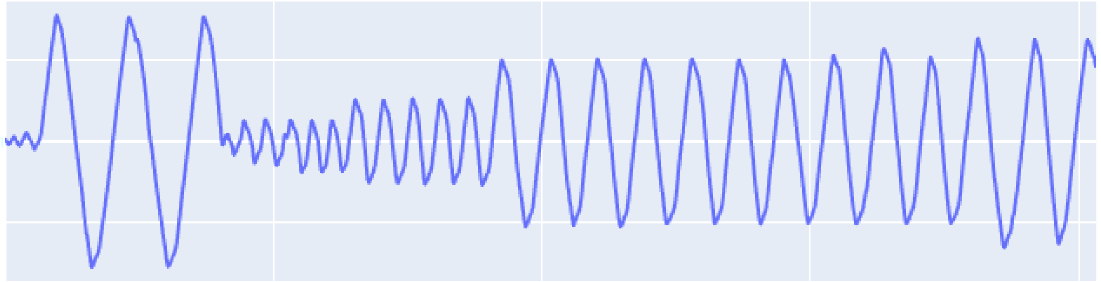
Then, prepare:
- new load protocol
- script to simulate experiment
- script to extract max forces



Design an experiment

First, through literature review get:

- conservative bounds for attributes
- simplified BRB model in OpenSees



Then, prepare:

- **new load protocol**
- script to simulate experiment
- script to extract max forces

```
set disp_protocol { 0.125 -0.125 0.25 -0.25
    3.0 -3.0 3.0 -3.0 3.0 -0.125 0.25 -0.25
    0.5 -0.5 0.5 -0.5 0.5 -0.5 0.5 -0.5 0.5
    -0.5 1.0 -1.0 1.0 -1.0 1.0 -1.0 1.0 -1.0
     1.0 -1.0 2.0 -2.0 2.0 -2.0 2.0 -2.0 2.0
     -2.0 2.0 -2.0 2.0 -2.0 2.0 -2.0 2.0 -
    2.0 2.0 -2.0 2.0 -2.0 }
```

disp_protocol.tcl

Design an experiment

First, through literature review get:

- conservative bounds for attributes
- simplified BRB model in OpenSees

Then, prepare:

- new load protocol
- **script to simulate experiment**
- script to extract max forces

```
model BasicBuilder -ndm 1 -ndf 1

set l 2.500
set A_y 600
set f_DM 1.39
set f_SM 1.13

# hyperparameters
pset gammaOv 1.1
pset omega 1.55
pset omegaIso 0.4;
pset beta 1.15

set l [expr $l*$m]
set A_y [expr $A_y*$mm2]

set f_yk [expr 235.0 * $MPa]
set E_s [expr 210.0 * $GPa]

# calculate material props
set E_0  [expr $f_SM*$E_s];
set f_y  [expr $gammaOv*$f_yk];
set eps_y [expr $f_y/$E_0]
```

BRB_response.tcl

Design an experiment

First, through literature review get:

- conservative bounds for attributes
- simplified BRB model in OpenSees

Then, prepare:

- new load protocol
- **script to simulate experiment**
- script to extract max forces

```
set matBRB 51
uniaxialMaterial Steel4 $matBRB $f_y $E_0 \
                -asym \
                -kin $b_k $R_0 $r_1 $r_2 $b_kc $R_
                -iso $b_i $rho_i $b_l $R_i $l_yp $
                -ult $f_u $R_u $f_u $R_u

element corotTruss 0 0 100 $A_y $matBRB

recorder Node -file "force_disp.out" -time -node 1

pattern Plain 1 Linear {
  load 100 [expr 1.0*$kN]
}

set IDctrlNode 100
set IDctrlDOF 1

constraints Plain
numberer RCM
system BandGeneral
test NormDispIncr 1.e-10 100
algorithm NewtonLineSearch -maxIter 100
```

BRB_response.tcl

Design an experiment

First, through literature review get:

- conservative bounds for attributes
- simplified BRB model in OpenSees

Then, prepare:

- new load protocol
- script to simulate experiment
- **script to extract max forces**

```python
#!/usr/bin/python
# written: adamzs 09/19

import numpy as np
import pandas as pd

def process_results(response):

    res = pd.read_csv('force_disp.out',
        sep=' ', header=None, names=['F', 'd'])

    F_c = np.abs(res['F'].min())
    F_t = res['F'].max()

    with open('results.out', 'w') as f:
        f.write('{} {}'.format(F_c, F_t))
```

postprocess.py

```
-262.325 227.717
```

results.out

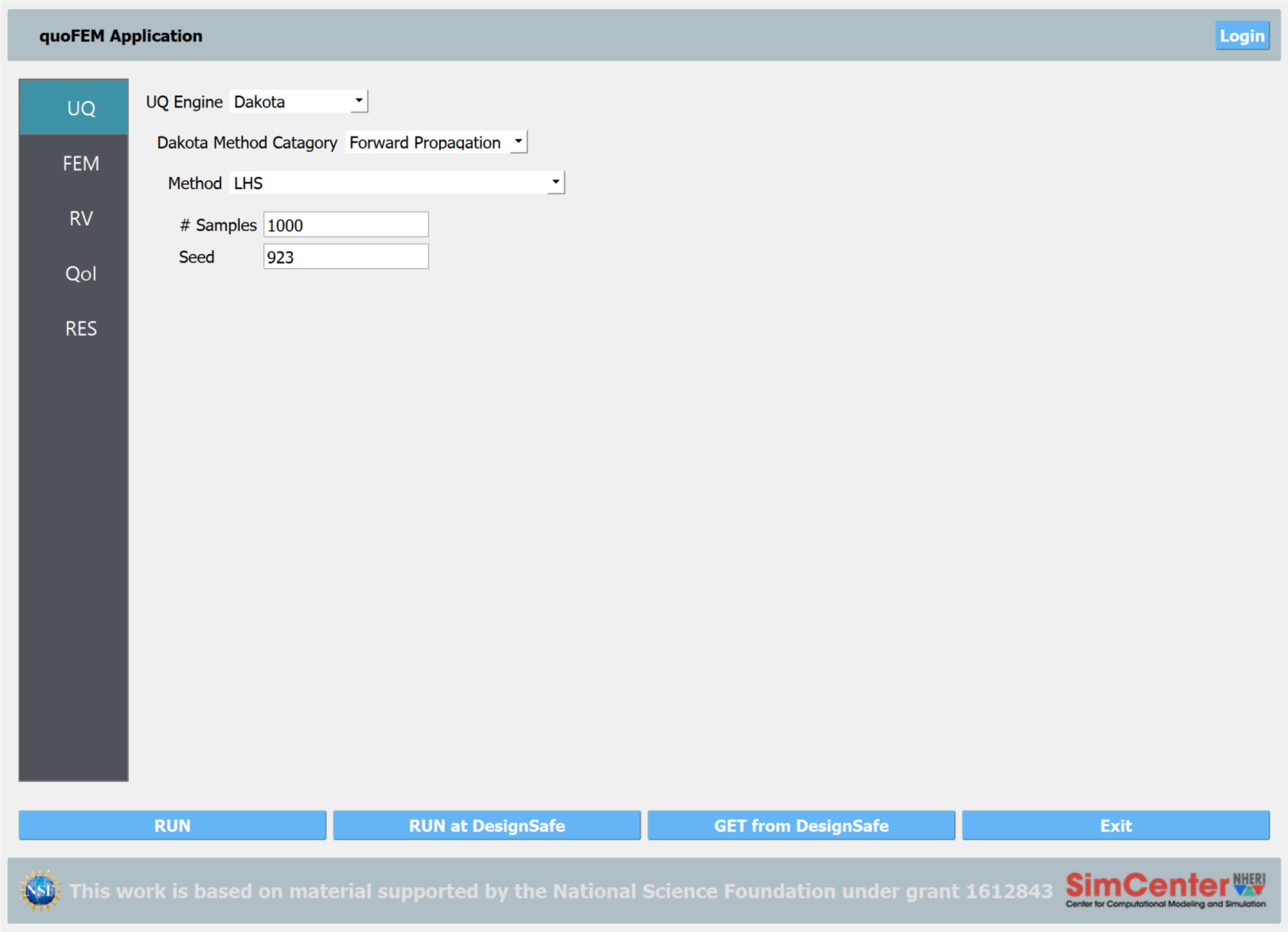
Design an experiment

select the type of problem

-> **forward propagation**

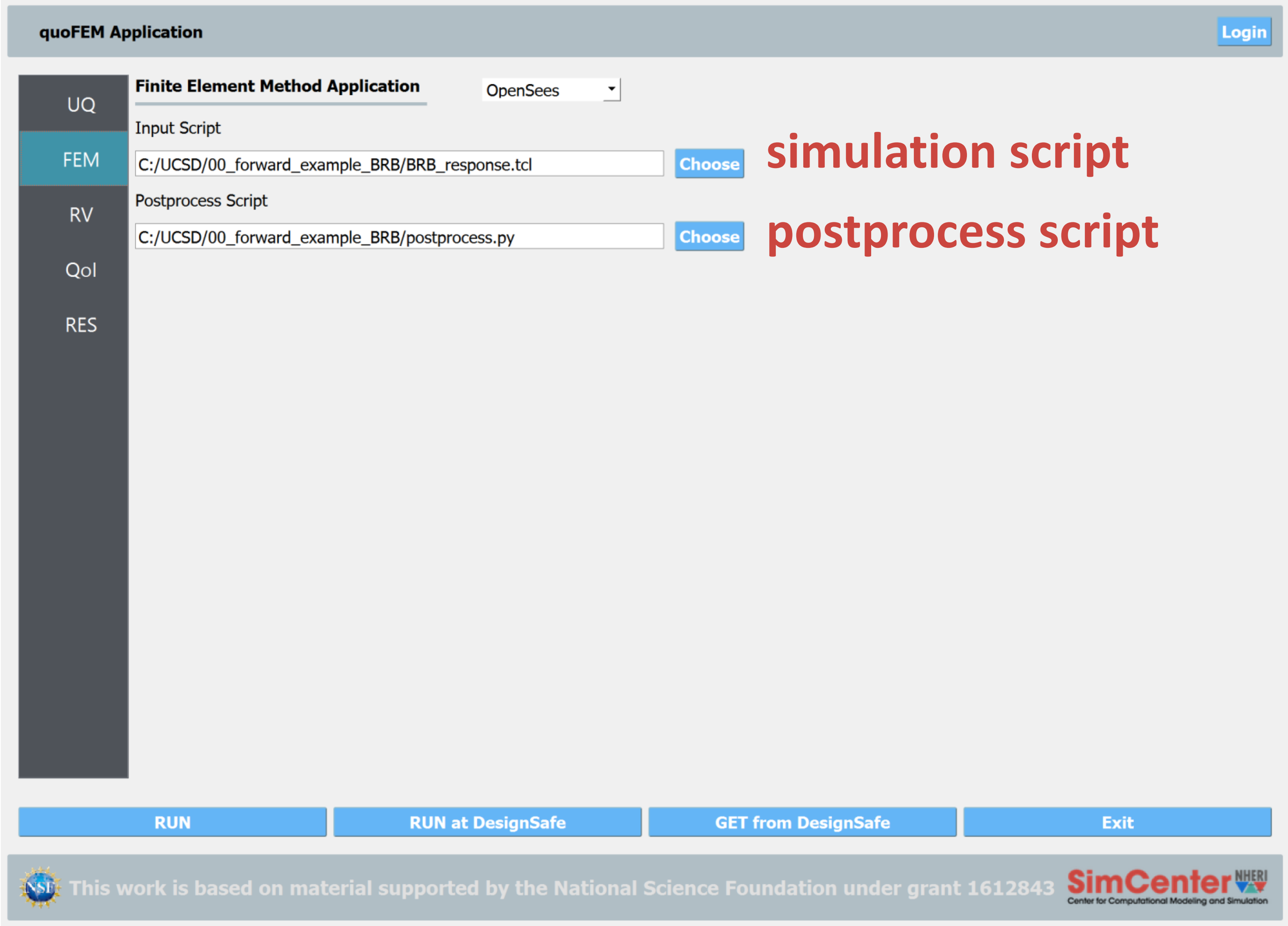
available methods:

- **Latin Hypercube Sampling**
- Monte Carlo Sampling
- Importance Sampling
- Gaussian Process Regression
- Polynomial Chaos Expansion



Design an experiment

load the pre-defined scripts



simulation script

postprocess script

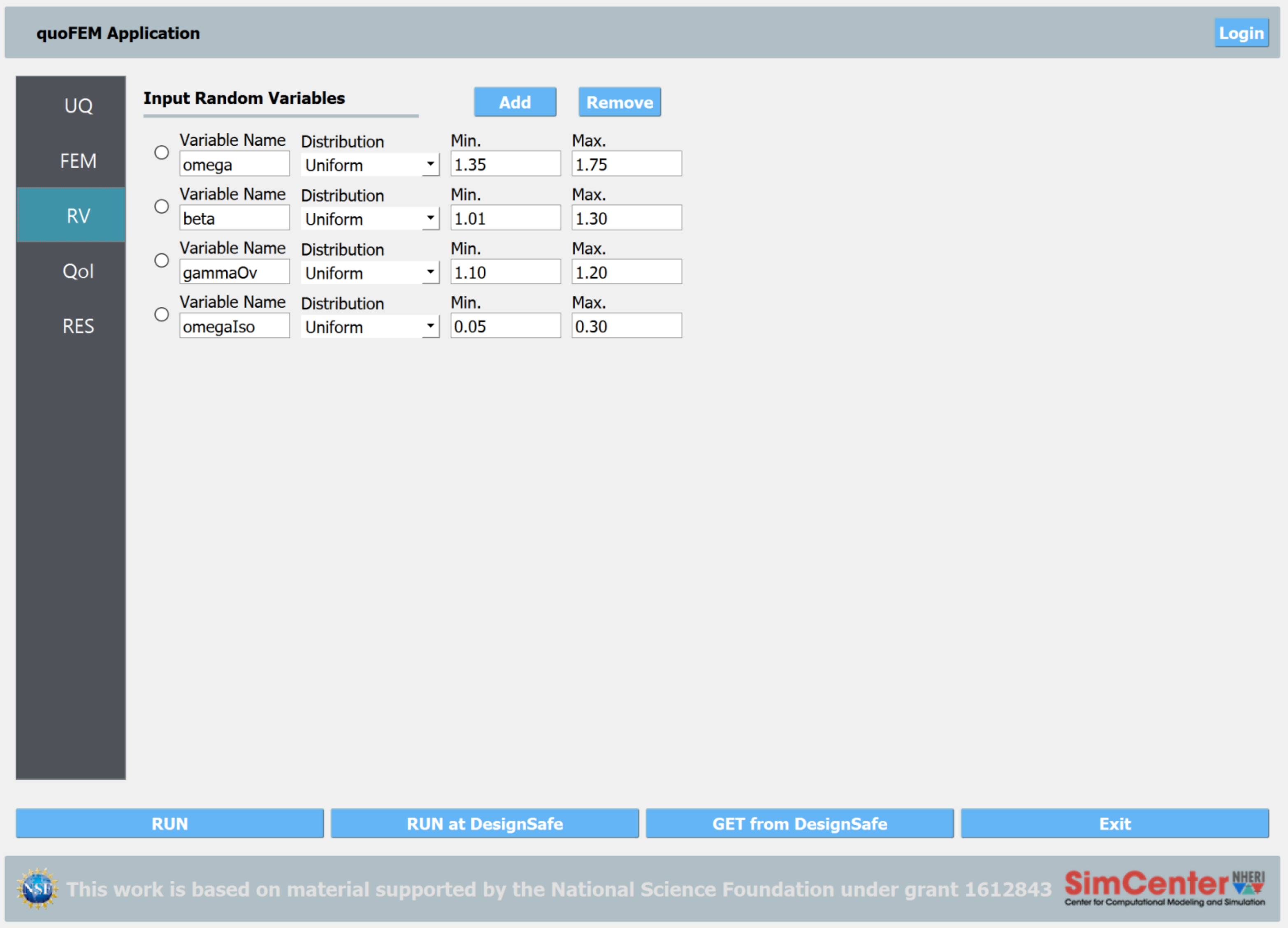
Design an experiment

BRB_response.tcl

```
# hyperparameters
pset gammaOv 1.1
pset omega 1.55
pset omegaIso 0.4;
pset beta 1.15
```

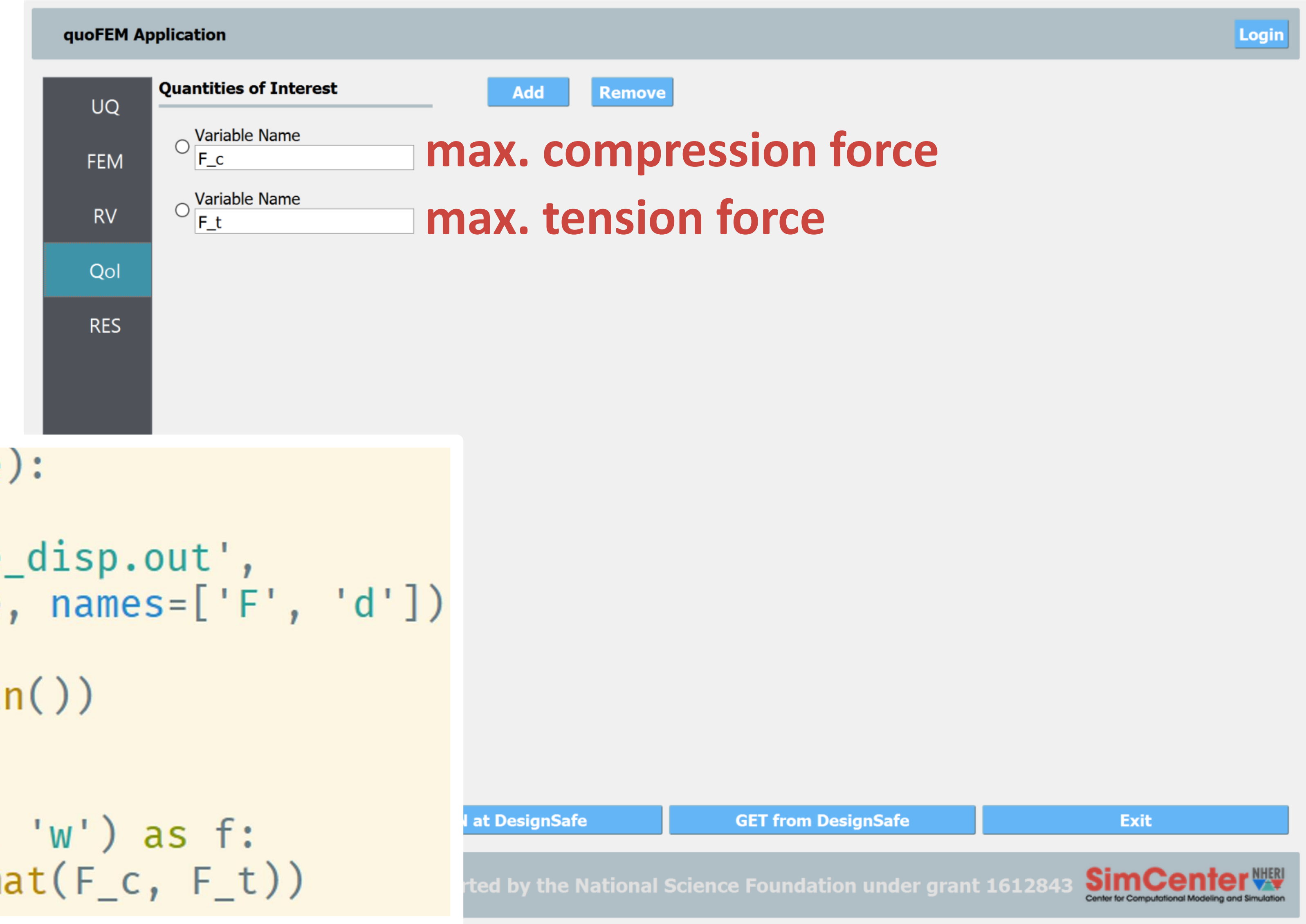
supported distributions:

- Normal, Lognormal
- Uniform
- Beta
- Weibull
- Gumbel



Design an experiment

specify the
Quantities of Interest



postprocess.py:

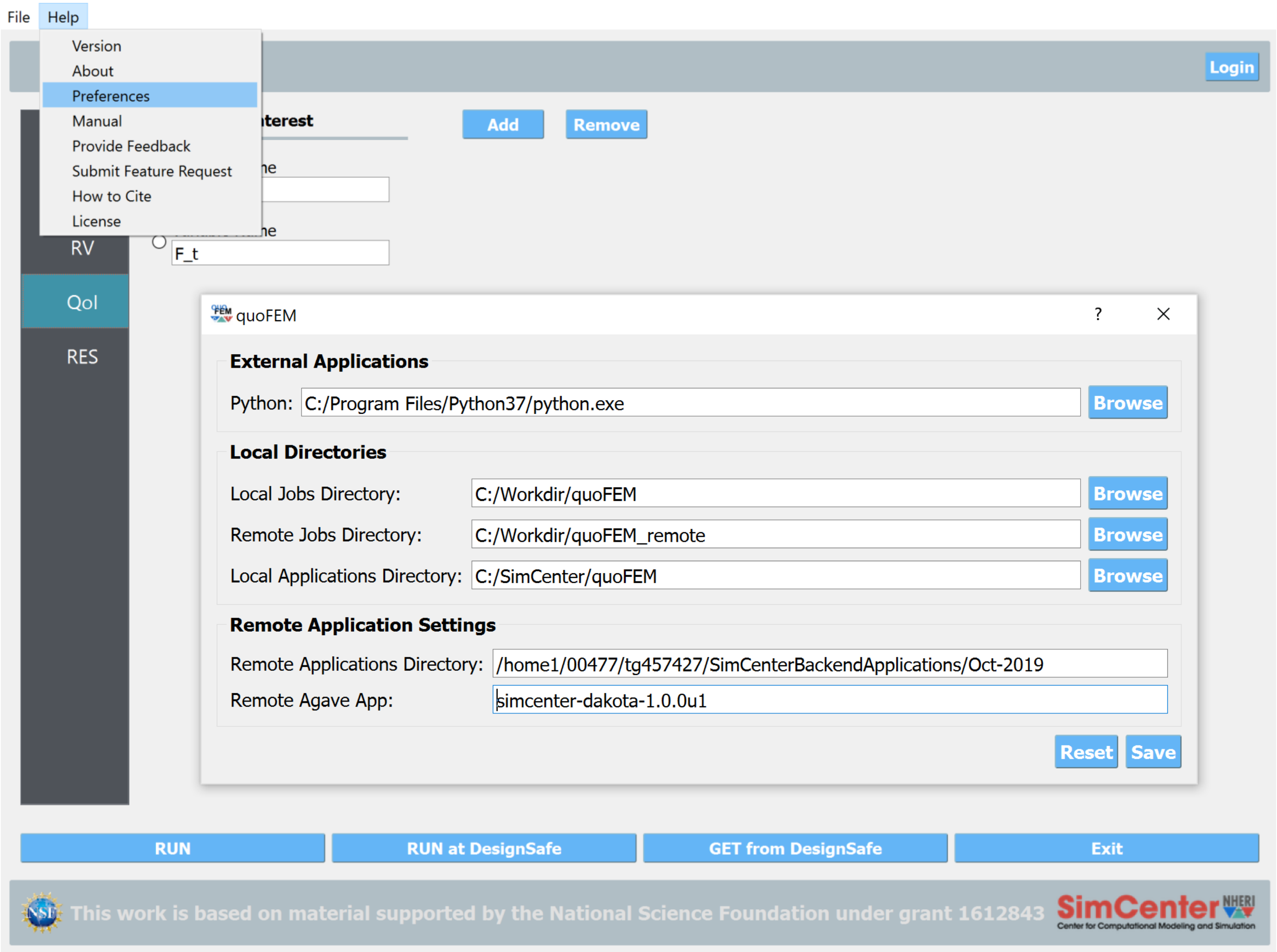
```python
def process_results(response):

    res = pd.read_csv('force_disp.out',
        sep=' ', header=None, names=['F', 'd'])

    F_c = np.abs(res['F'].min())
    F_t = res['F'].max()

    with open('results.out', 'w') as f:
        f.write('{} {}'.format(F_c, F_t))
```

Design an experiment

quoFEM

advanced users can

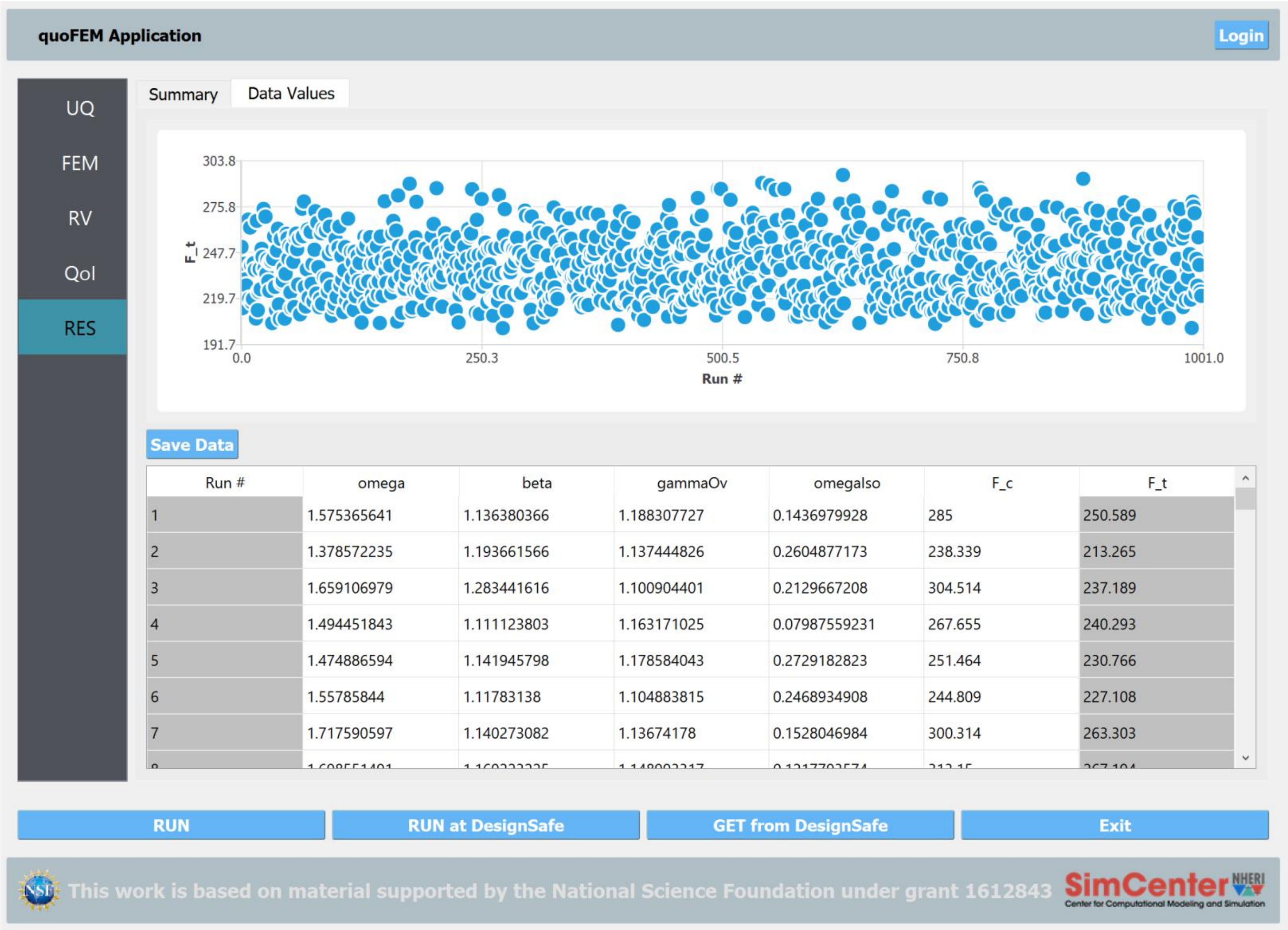
- specify their python interpreter
- edit working directories



Design an experiment

result visualization

scatterplots show sampled inputs and corresponding outputs

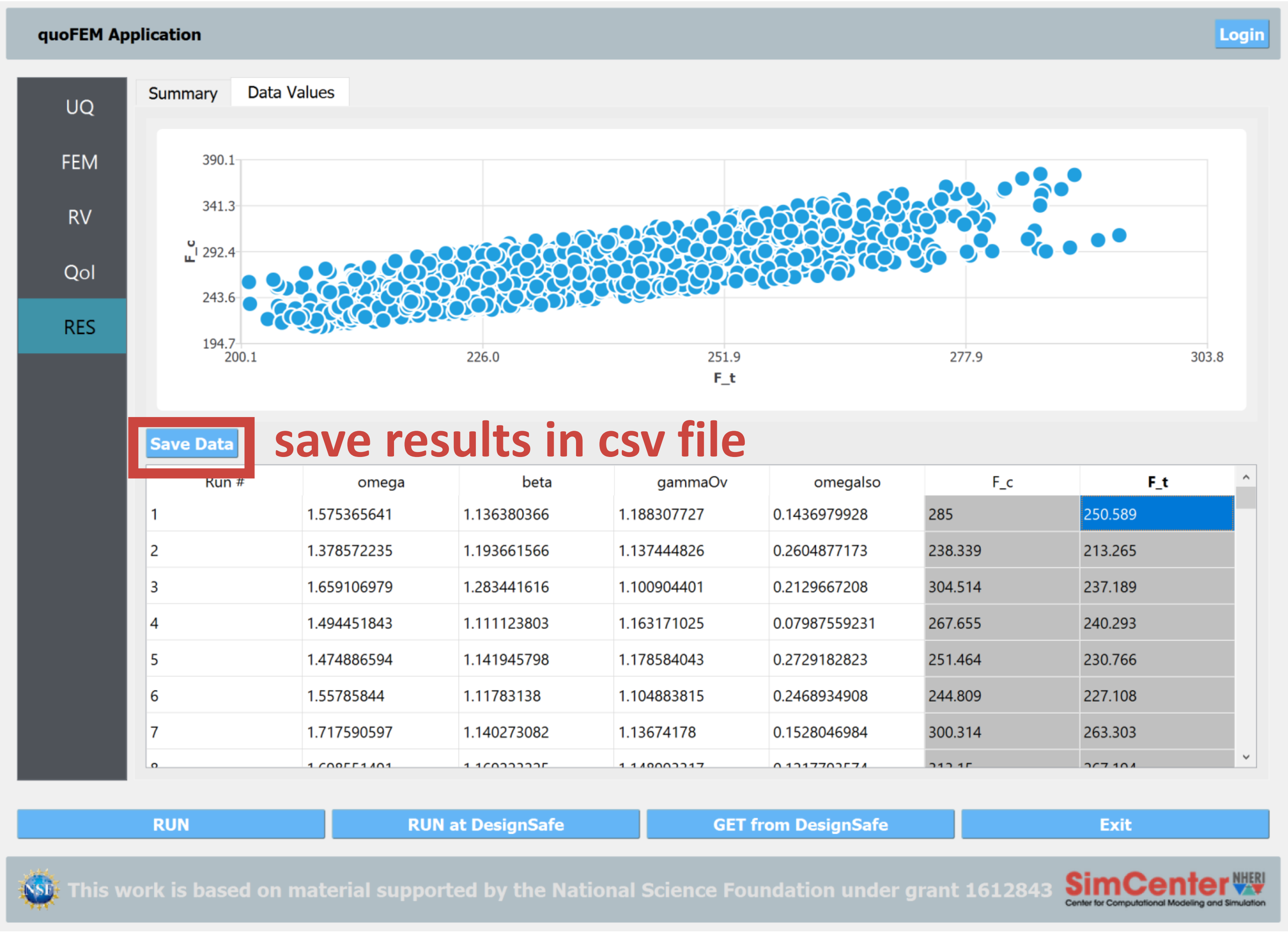


Design an experiment

result visualization

scatterplots show sampled inputs and corresponding outputs

and **joint distributions**



Run #	omega	beta	gammaOv	omegaIso	F_c	F_t
1	1.575365641	1.136380366	1.188307727	0.1436979928	285	250.589
2	1.378572235	1.193661566	1.137444826	0.2604877173	238.339	213.265
3	1.659106979	1.283441616	1.100904401	0.2129667208	304.514	237.189
4	1.494451843	1.111123803	1.163171025	0.07987559231	267.655	240.293
5	1.474886594	1.141945798	1.178584043	0.2729182823	251.464	230.766
6	1.55785844	1.11783138	1.104883815	0.2468934908	244.809	227.108
7	1.717590597	1.140273082	1.13674178	0.1528046984	300.314	263.303

Design an experiment

advanced visualization and data processing

easy to load csv in Python, MatLAB, or Excel

```
import pandas as pd
import plotly.express as px
res = pd.read_csv("C:/UCSD/00_forward_example_BRB/output_UCSD.csv",
                  sep=',\t', dtype=float)
res.head(10)
```

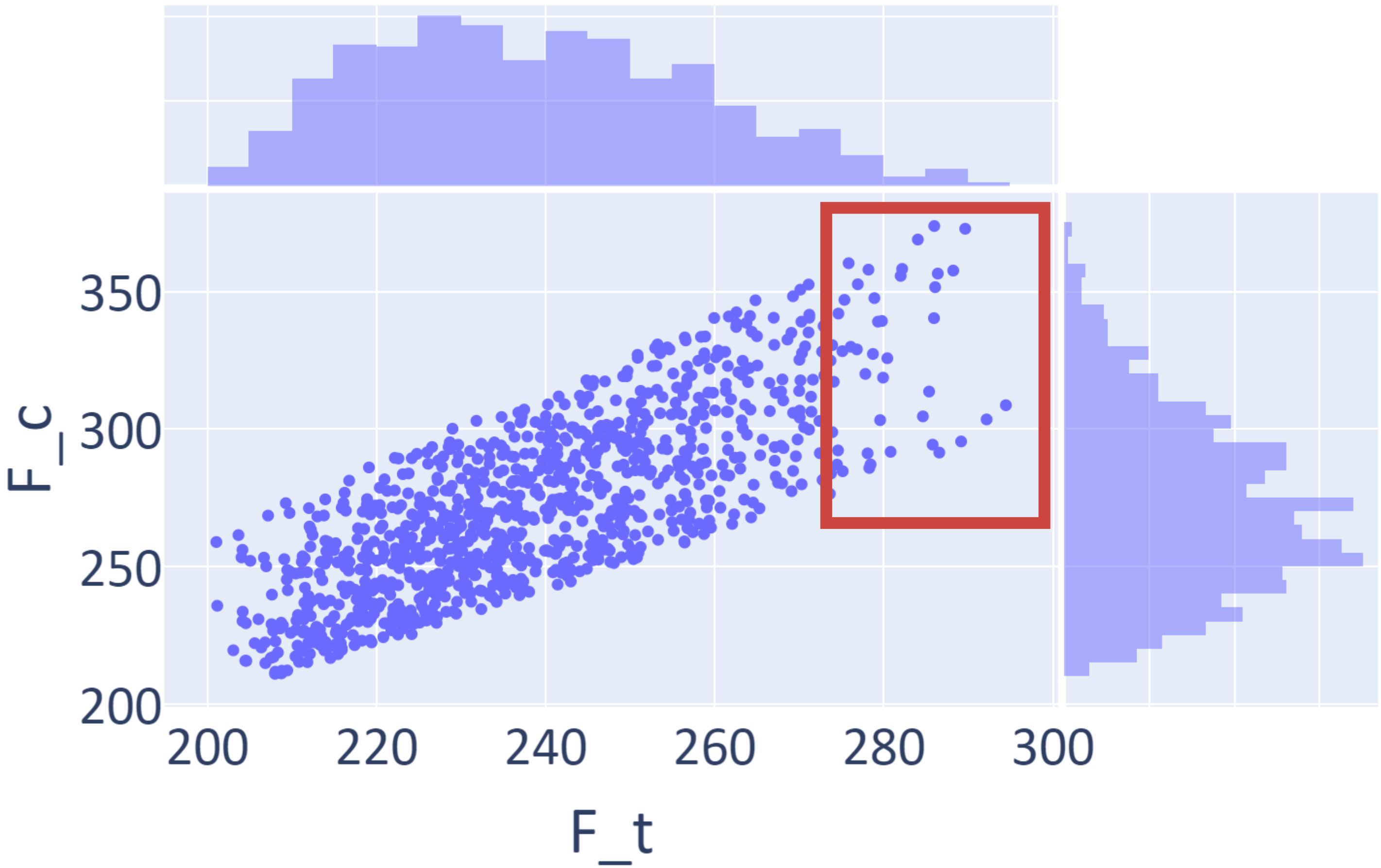
	Run #	omega	beta	gammaOv	omegaIso	F_c	F_t
0	1.0	1.57537	1.13638	1.18831	0.143698	285.000	250.589
1	2.0	1.37857	1.19366	1.13744	0.260488	238.339	213.265
2	3.0	1.65911	1.28344	1.10090	0.212967	304.514	237.189
3	4.0	1.49445	1.11112	1.16317	0.079876	267.655	240.293
4	5.0	1.47489	1.14195	1.17858	0.272918	251.464	230.766
5	6.0	1.55786	1.11783	1.10488	0.246893	244.809	227.108
6	7.0	1.71759	1.14027	1.13674	0.152805	300.314	263.303
7	8.0	1.69855	1.16922	1.14899	0.121779	313.150	267.194
8	9.0	1.40822	1.20869	1.17317	0.176460	257.507	218.961
9	10.0	1.73973	1.10822	1.18051	0.253128	289.577	262.384

Design an experiment

advanced visualization and data processing

```
fig = px.scatter(res, x='F_t', y='F_c',
                 marginal_x='histogram', marginal_y='histogram',
                 height=500, width=800)
fig['layout'].update(font=dict(family="Calibri", size=28))
fig.show()
```

joint distribution with marginals

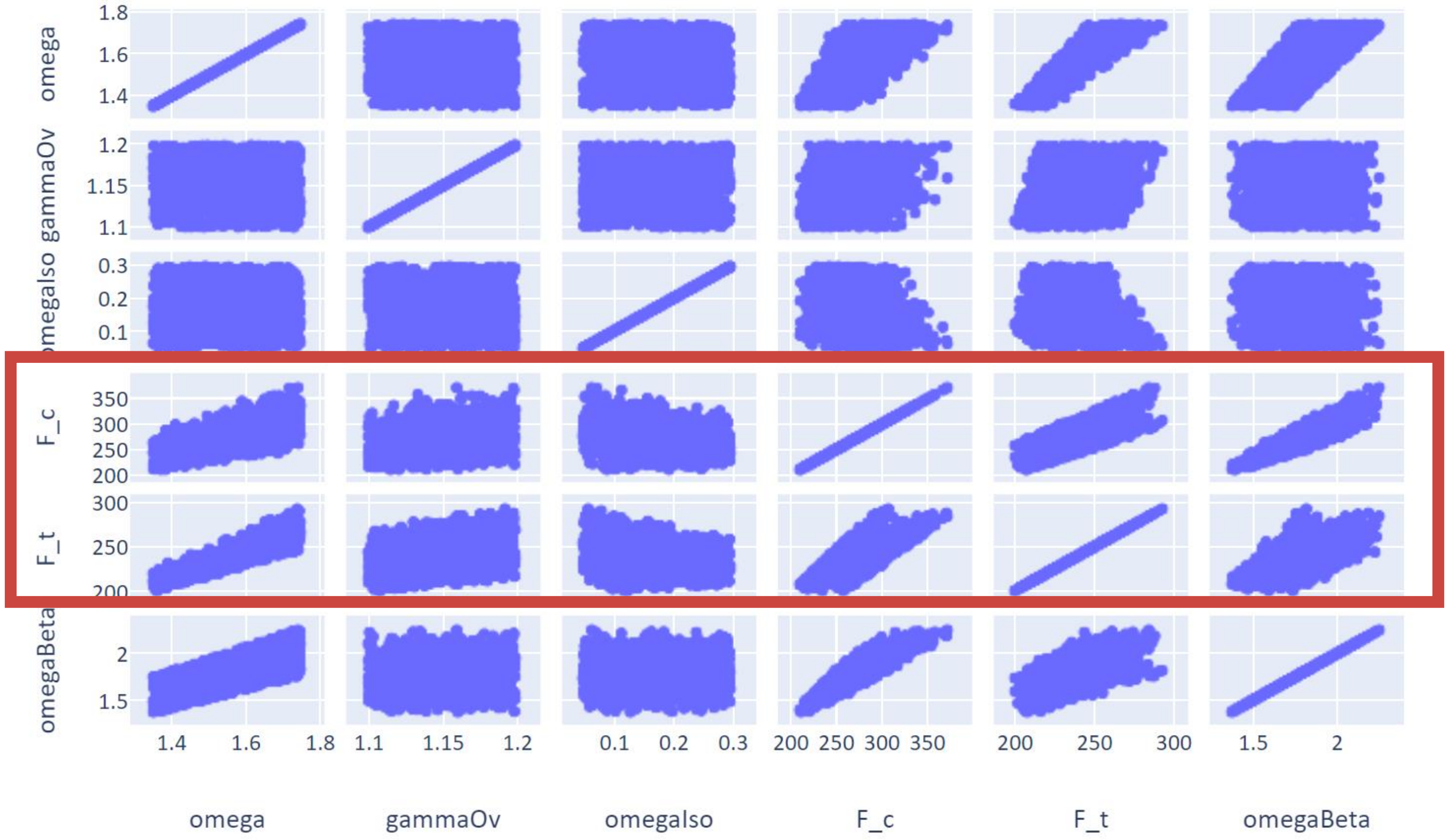


Design an experiment

advanced visualization and data processing

use scatter matrix to identify dependencies

```
fig = px.scatter_matrix(res,
                        dimensions=['omega','omegaBeta','omegaIso',
                                    'gammaOv','F_c','F_t'],
                        height=500, width=800)
fig['layout'].update(font=dict(family="Calibri", size=15))
fig.show()
```

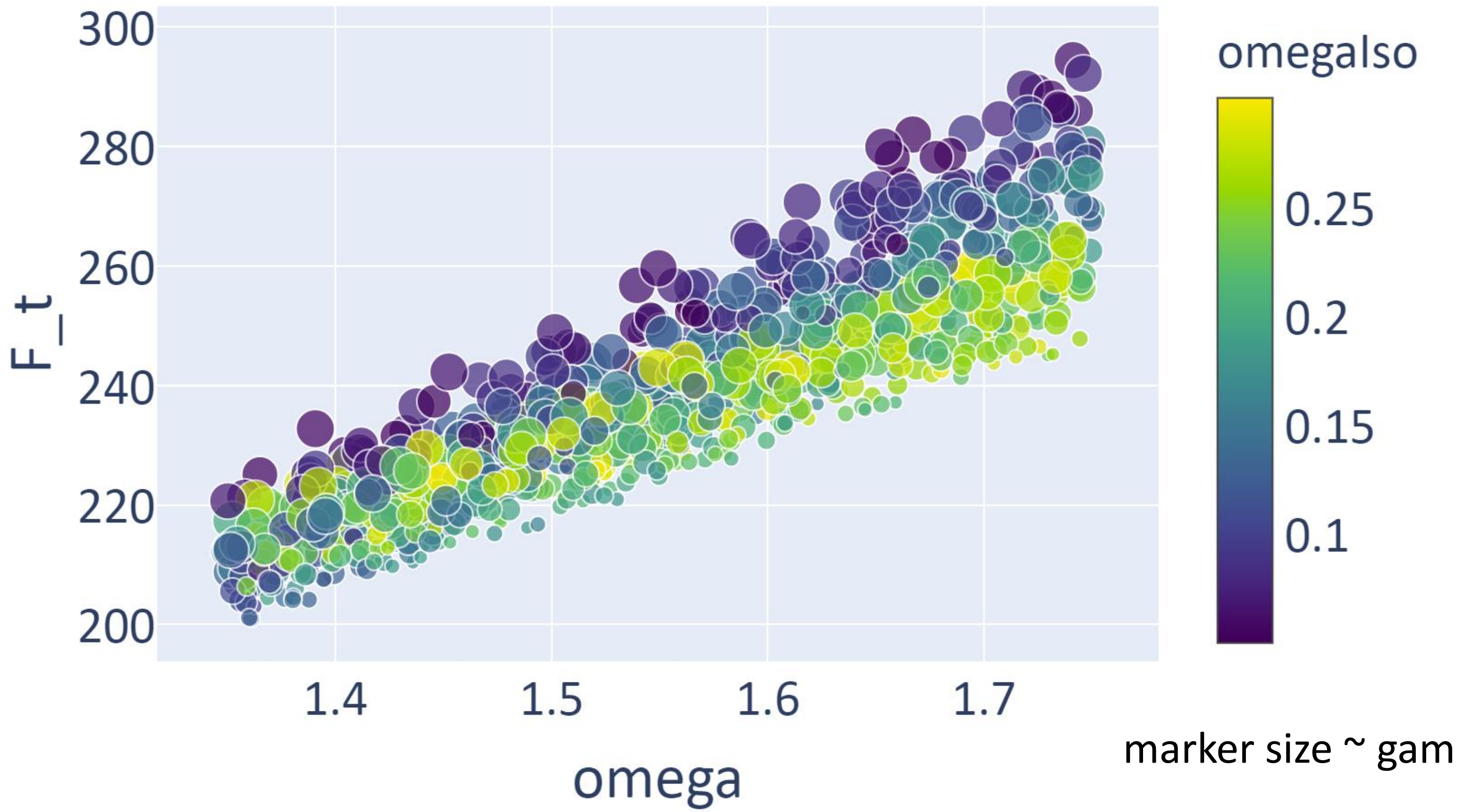


Design an experiment

advanced visualization and data processing

```
res['gammaOv_viz'] = np.log(res['gammaOv']*10.0-9.9)
fig2 = px.scatter(res, x='omega', y='F_t', color='omegaIso',
                  color_continuous_scale=px.colors.sequential.Viridis,
                  size='gammaOv_viz', size_max=20, height=500, width=800)
fig2['layout'].update(font=dict(family="Calibri", size=28))
fig2.show()
```

use color and marker size to visualize dependency on 3 parameters



marker size ~ gammaOv

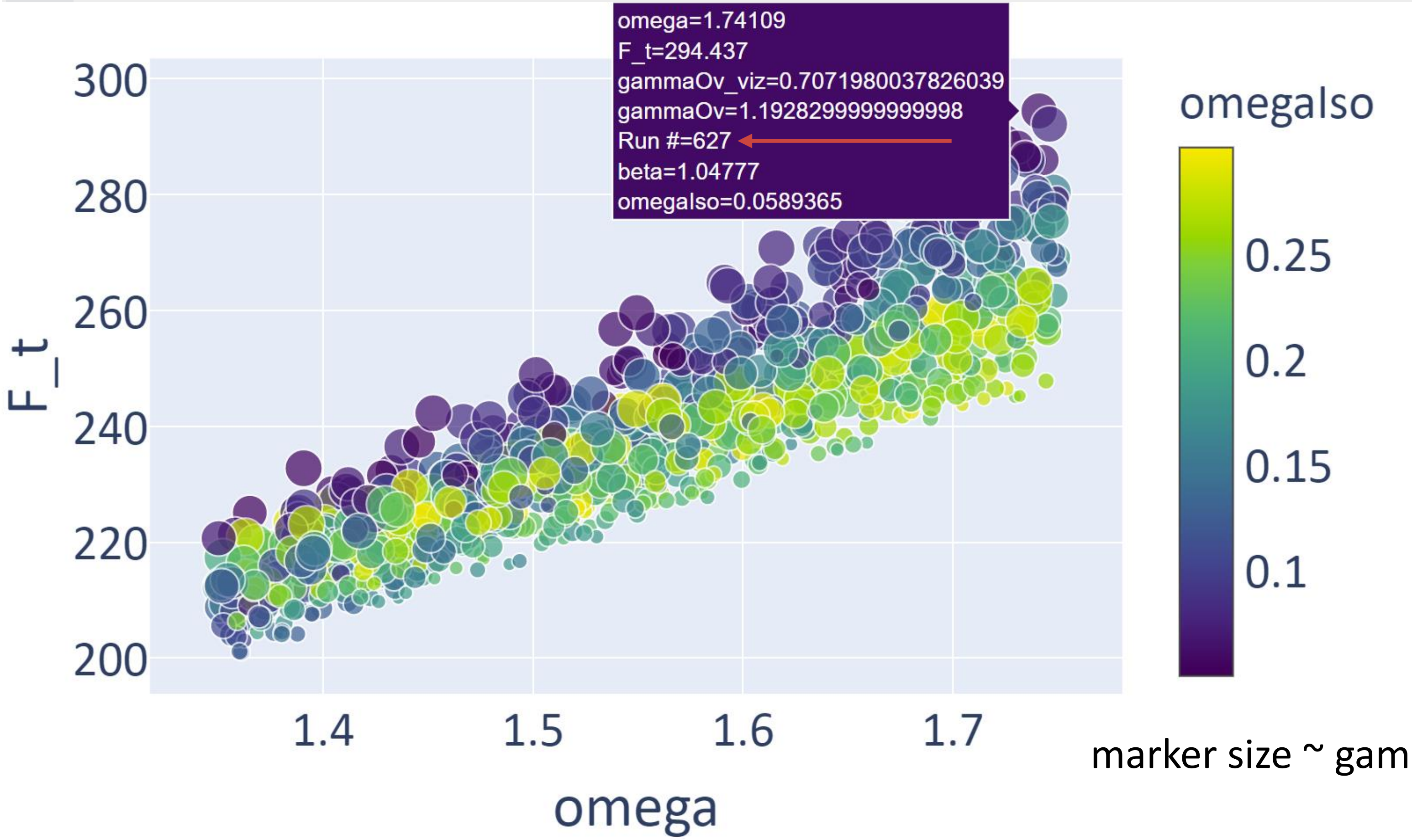
Design an experiment

quoFEM

advanced visualization and data processing

raw data is available for every simulation

```
res['gammaOv_viz'] = np.log(res['gammaOv']*10.0-9.9)
fig2 = px.scatter(res, x='omega', y='F_t', color='omegaIso',
                  color_continuous_scale=px.colors.sequential.Viridis,
                  size='gammaOv_viz', size_max=20, height=500, width=800)
fig2['layout'].update(font=dict(family="Calibri", size=28))
fig2.show()
```



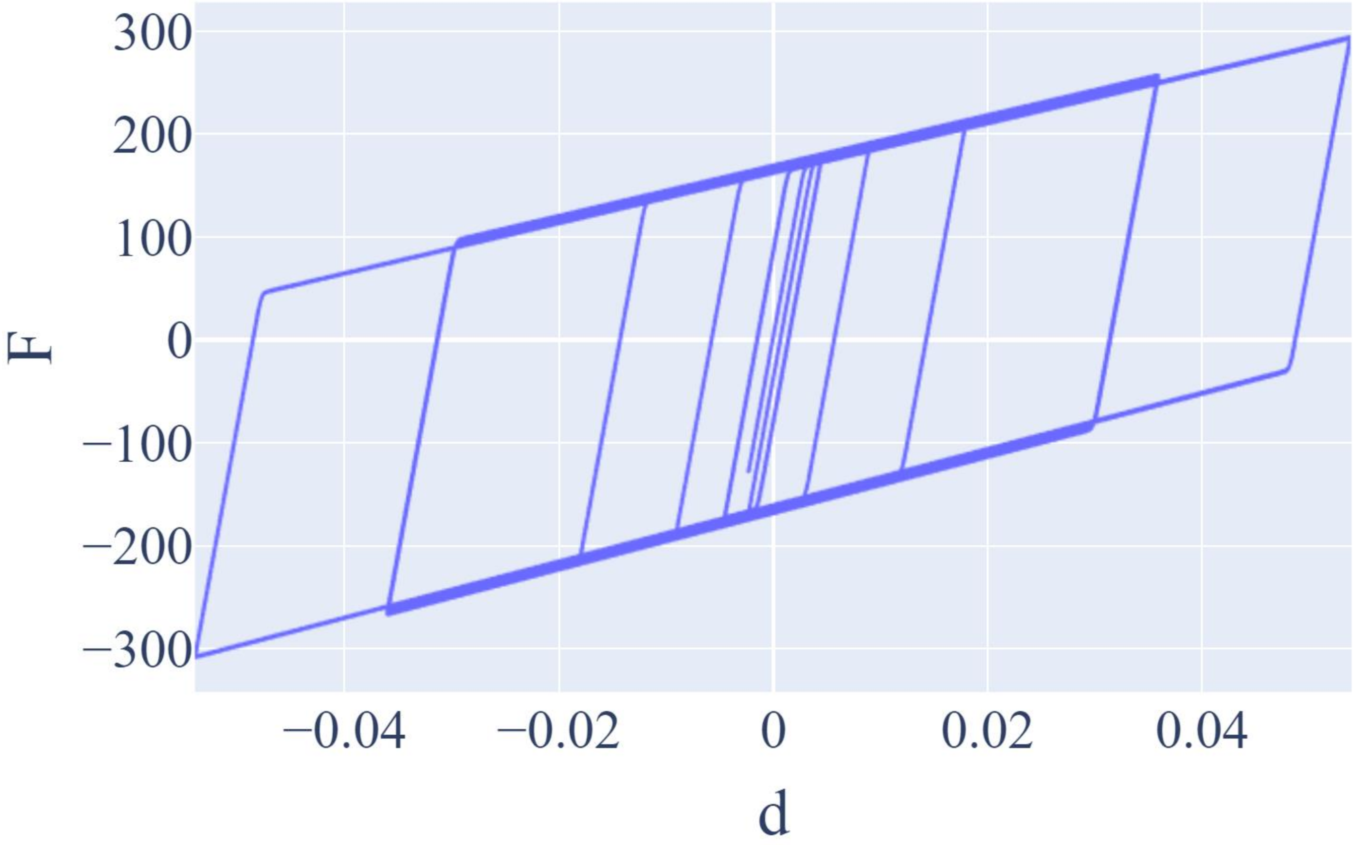
marker size ~ gammaOv

Design an experiment

advanced visualization and data processing

```
output_file = "C:/Workdir/quoFEM/tmp.SimCenter/workdir.627/force_disp.out"
res = pd.read_csv(output_file, sep=' ', header=None, names=['F', 'd'])
fig = px.line(res, x='d', y='F', height=500, width=800)
fig['layout'].update(font=dict(family="Calibri", size=28))
fig.show()
```

raw data is available for every simulation



Case study

example application: Buckling Restrained Braced Frames

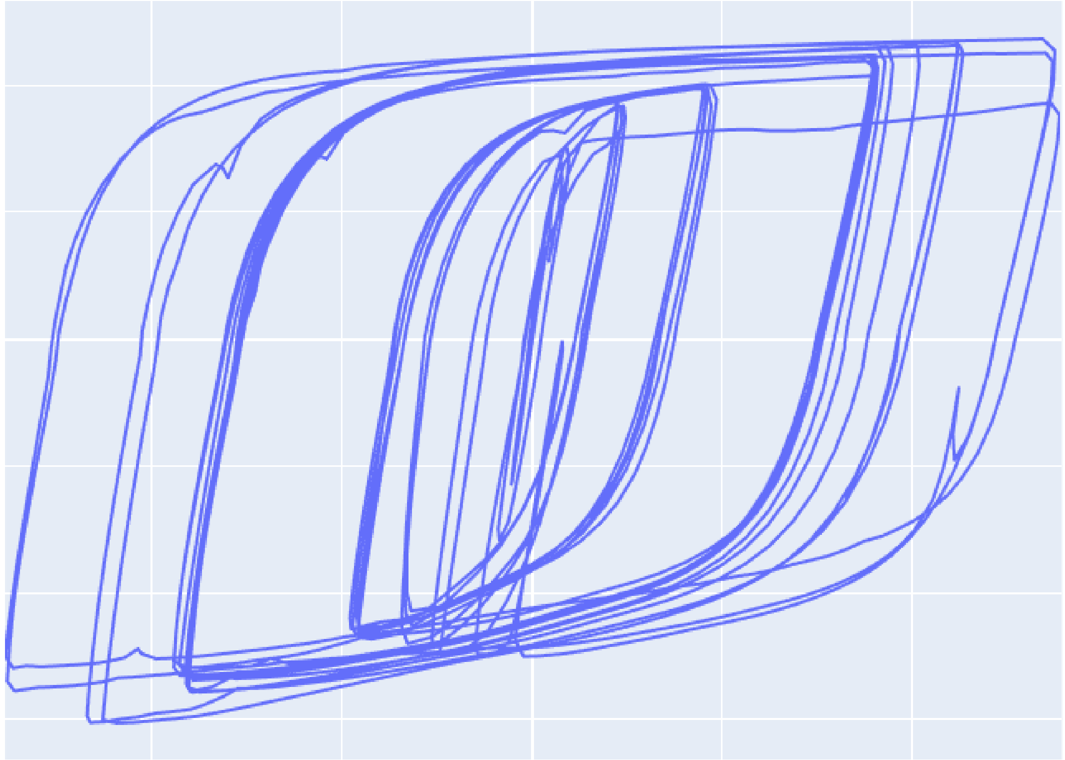
1. Design an experiment — quoFEM
2. **Calibrate a numerical component model — quoFEM**
3. Simulate structural response — EE-UQ
4. Estimate damage and losses — PBE

Calibrate component model

objective: calibrate Buckling Restrained Brace model parameters

problem: complex behavior, large number of material parameters

quoFEM: minimize error between simulation and reference result



Calibrate component model

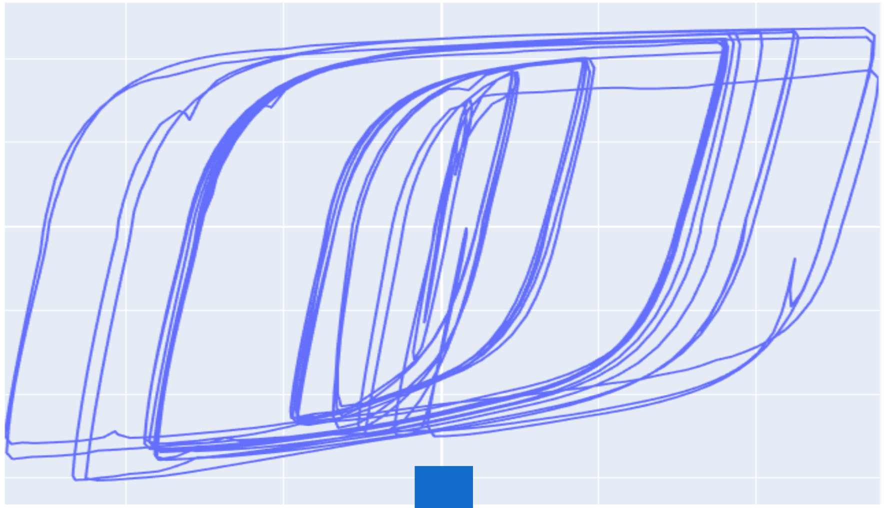
Prepare:

- reference data from test results
- complex BRB model in OpenSees
- updated simulation script
- script with the objective function

Calibrate component model

Prepare:

- **reference data from test results**
- complex BRB model in OpenSees
- updated simulation script
- script with the objective function



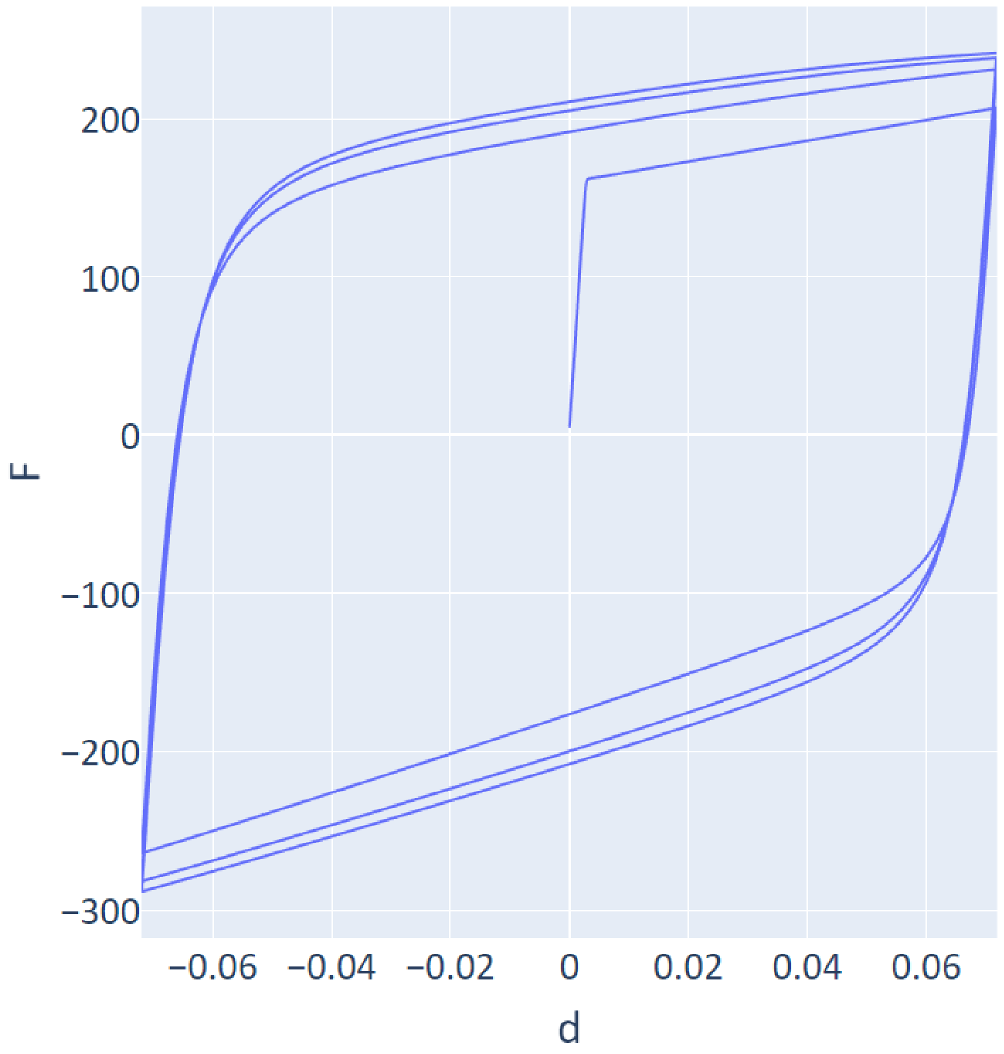
```
78.96559395918378,1[illegible]215204081632668
52.411307755102094,0.9329030612244907
26.90855167346942,0.5383724489795925
4.500137142857147,0.1784795918367349
-18.123096000000018,-0.21307142857142883
-38.138502857142896,-0.5226428571428579
-59.64269926530619,-0.8938724489795928
-74.79477551020416,-1.2276428571428586
-73.4783206530613,-1.2275918367346954
-43.2296948571429,-0.7959540816326539
-23.08952375510207,-0.4911122448979598
```

reference.txt

Calibrate component model

Prepare:

- reference data from test results
- **complex BRB model in OpenSees**
- updated simulation script
- script with the objective function



Calibrate component model

Prepare:

- reference data from test results
- complex BRB model in OpenSees
- **updated simulation script**
- script with the objective function

```
model BasicBuilder -ndm 1 -ndf 1

set l 2.500
set A_y 600
set f_DM 1.39
set f_SM 1.13

# hyperparameters
pset gammaOv 1.15
pset omega 1.4
pset omegaIso 0.25;
pset beta 1.15
pset rhoxi 0.3
pset rx1 0.925
pset fxult 1.55
pset fxultc 2.5

set l [expr $l*$m]
set A_y [expr $A_y*$mm2]

set f_yk [expr 235.0 * $MPa]
set E_s [expr 210.0 * $GPa]

# calculate material props
set E_0  [expr $f_SM*$E_s];
set f_y  [expr $gammaOv*$f_yk];
set eps_y [expr $f_y/$E_0]
```

BRB_response.tcl

Calibrate component model

Prepare:

- reference data from test results
- complex BRB model in OpenSees
- updated simulation script
- **script with the objective function**

```python
#!/usr/bin/python
# written: adamzs

import numpy as np
import pandas as pd

def process_results(response):

    data = pd.read_csv('reference.txt',
        header=None, names=['F_ref','d'])
    # convert displacements to [m]
    data['d'] = data['d']/1000.

    sim = pd.read_csv('force_disp.out',
        sep=' ', header=None, names=['F','d'])
    data['F_sim'] = sim['F']

    eps_s = (data['F_sim'] - data['F_ref'])**2.
    eps_srss = np.sqrt(np.sum(eps_s)/len(data.index))

    with open('results.out', 'w') as f:
        f.write("{:.6f}".format(eps_srss))
```

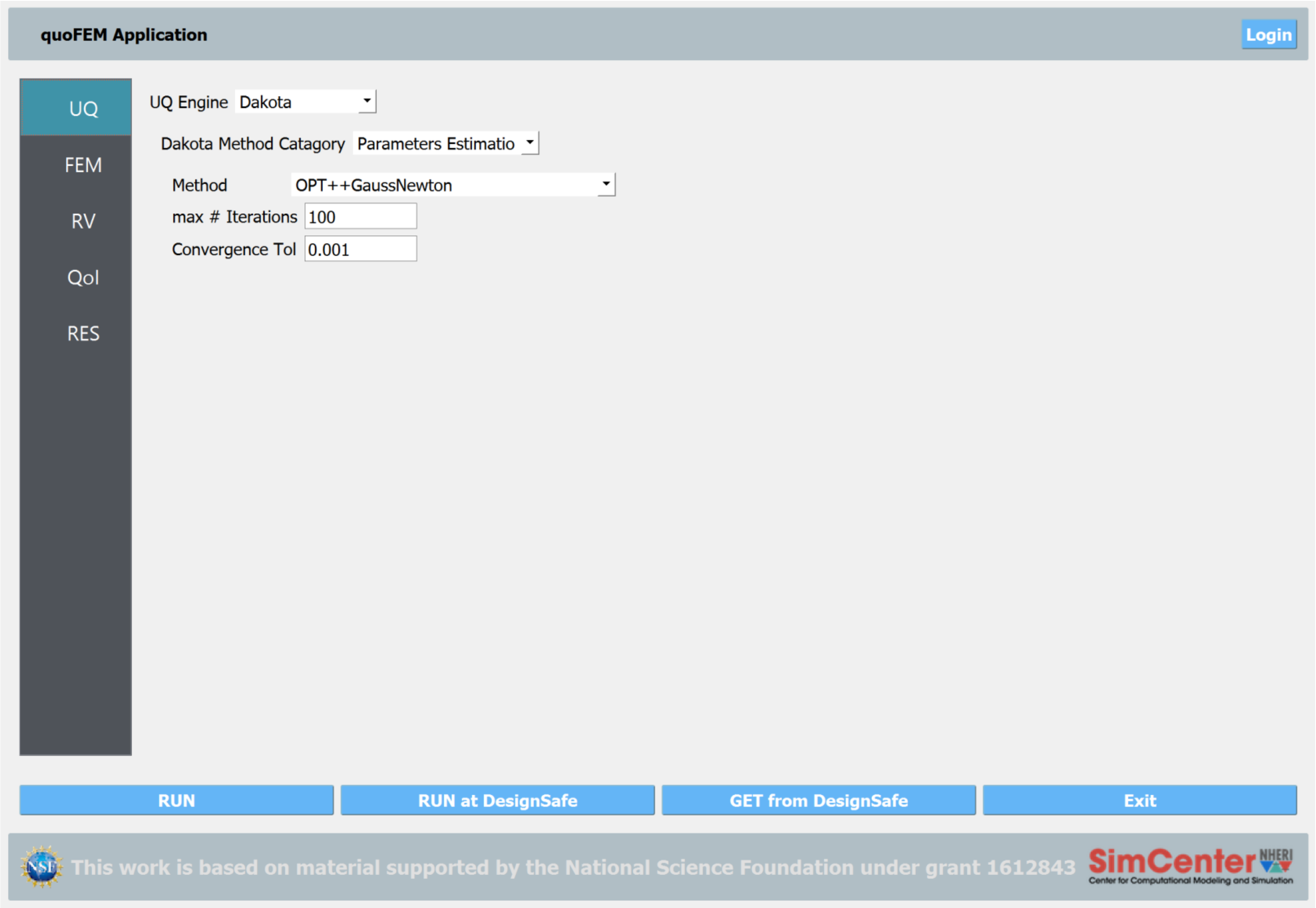
postprocess.py

Calibrate component model

select the type of problem
-> **parameter estimation**

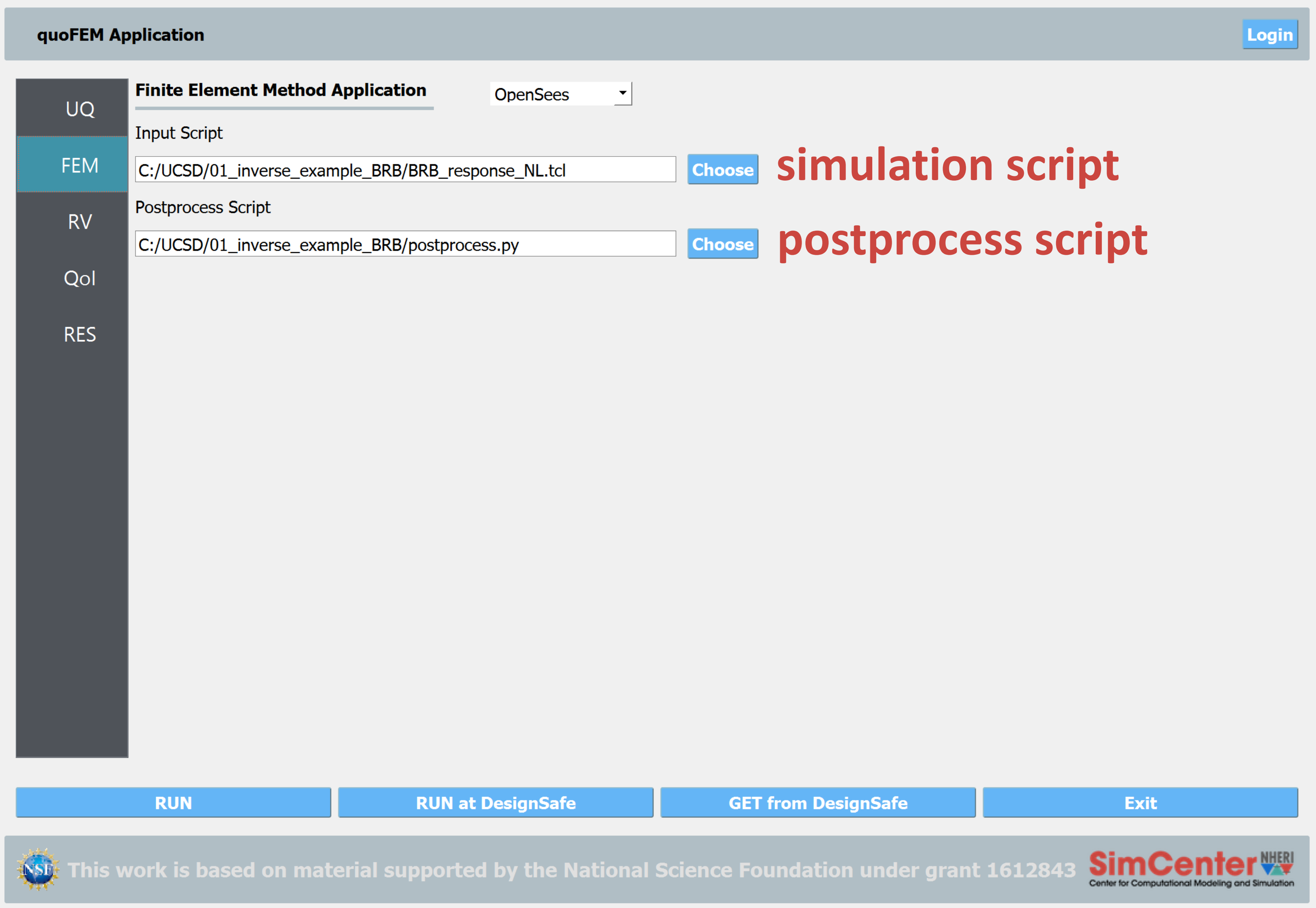
Gauss-Newton is a gradient-based optimization method

we also have Bayesian methods for solving the inverse problem



Calibrate component model

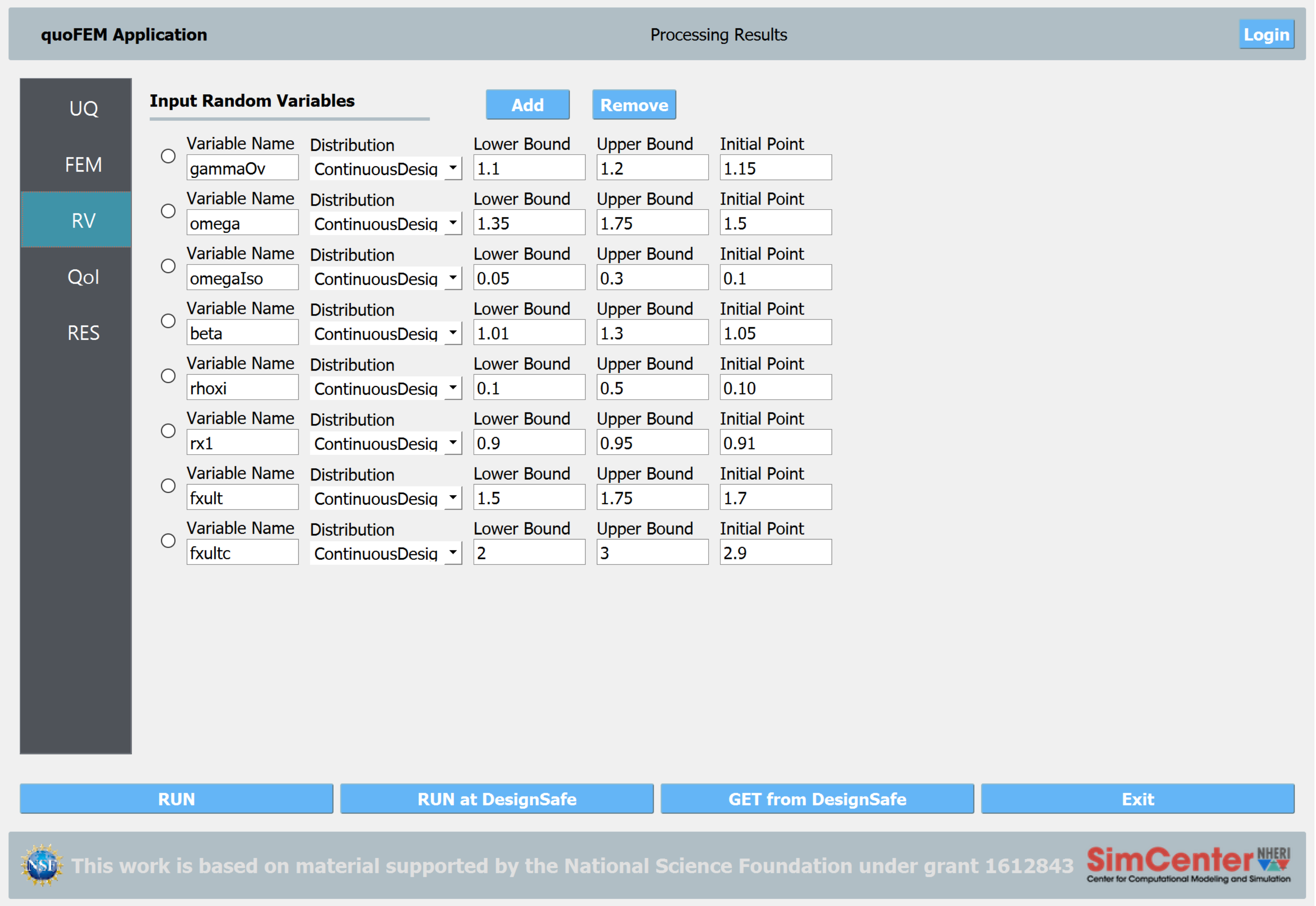
load the pre-defined scripts



Calibrate component model

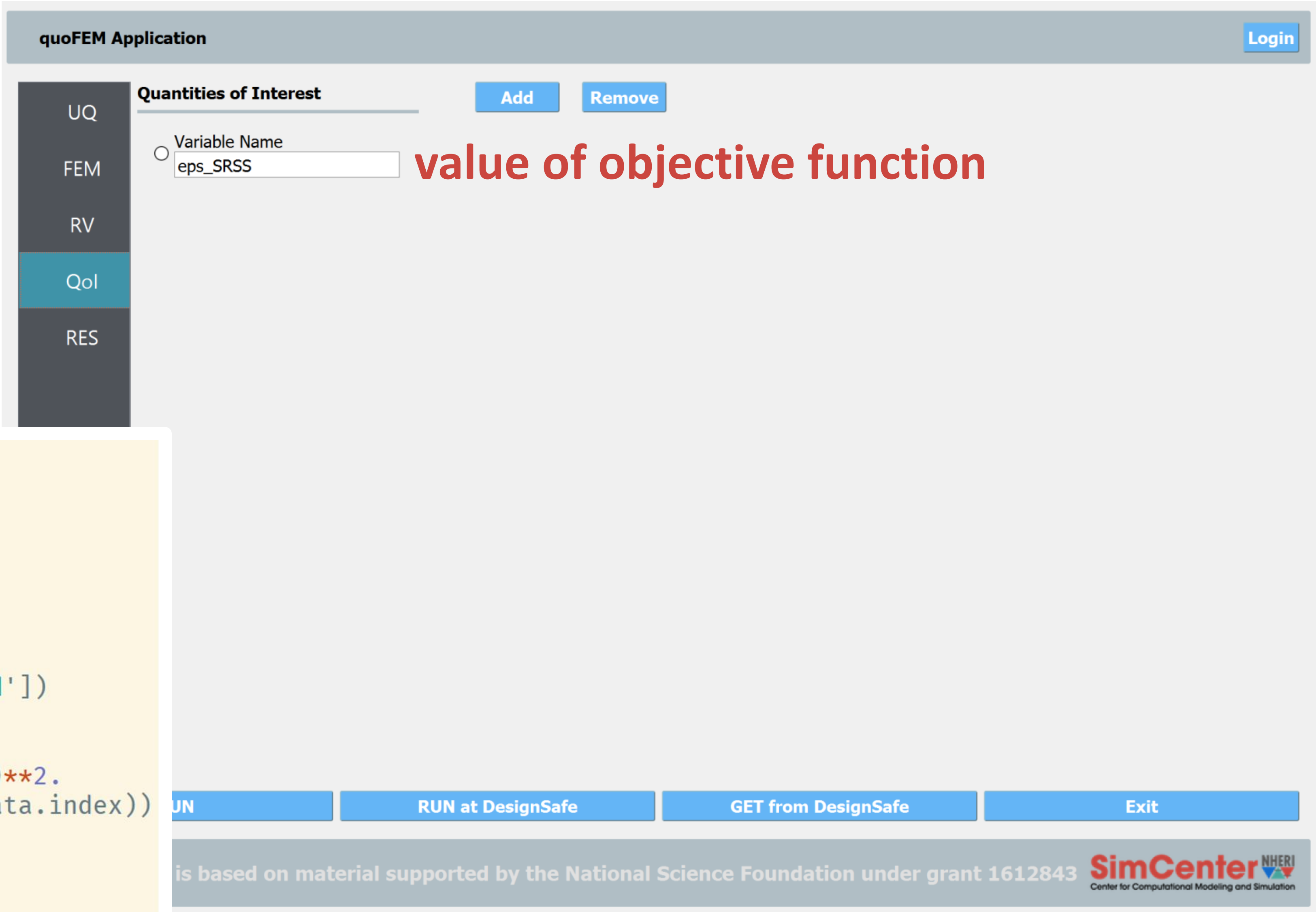
quoFEM

```
# hyperparameters
pset gammaOv 1.15
pset omega 1.4
pset omegaIso 0.25;
pset beta 1.15
pset rhoxi 0.3
pset rx1 0.925
pset fxult 1.55
pset fxultc 2.5
```



Calibrate component model

specify the
Quantities of Interest

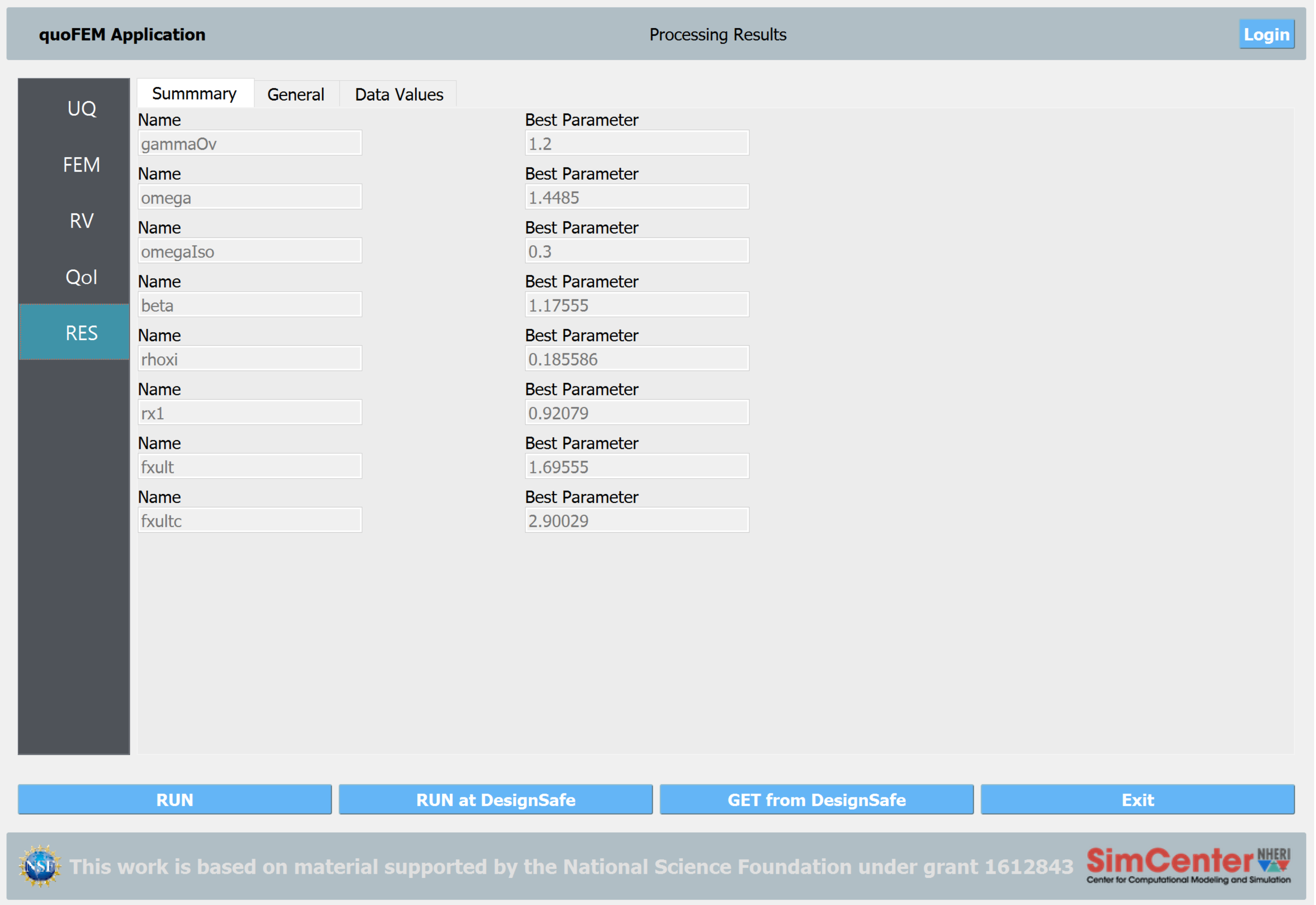


value of objective function

postprocess.py:

```python
def process_results(response):

    data = pd.read_csv('reference.txt',
        header=None, names=['F_ref','d'])
    # convert displacements to [m]
    data['d'] = data['d']/1000.

    sim = pd.read_csv('force_disp.out',
        sep=' ', header=None, names=['F','d'])
    data['F_sim'] = sim['F']

    eps_s = (data['F_sim'] - data['F_ref'])**2.
    eps_srss = np.sqrt(np.sum(eps_s)/len(data.index))

    with open('results.out', 'w') as f:
        f.write("{:.6f}".format(eps_srss))
```

Calibrate component model

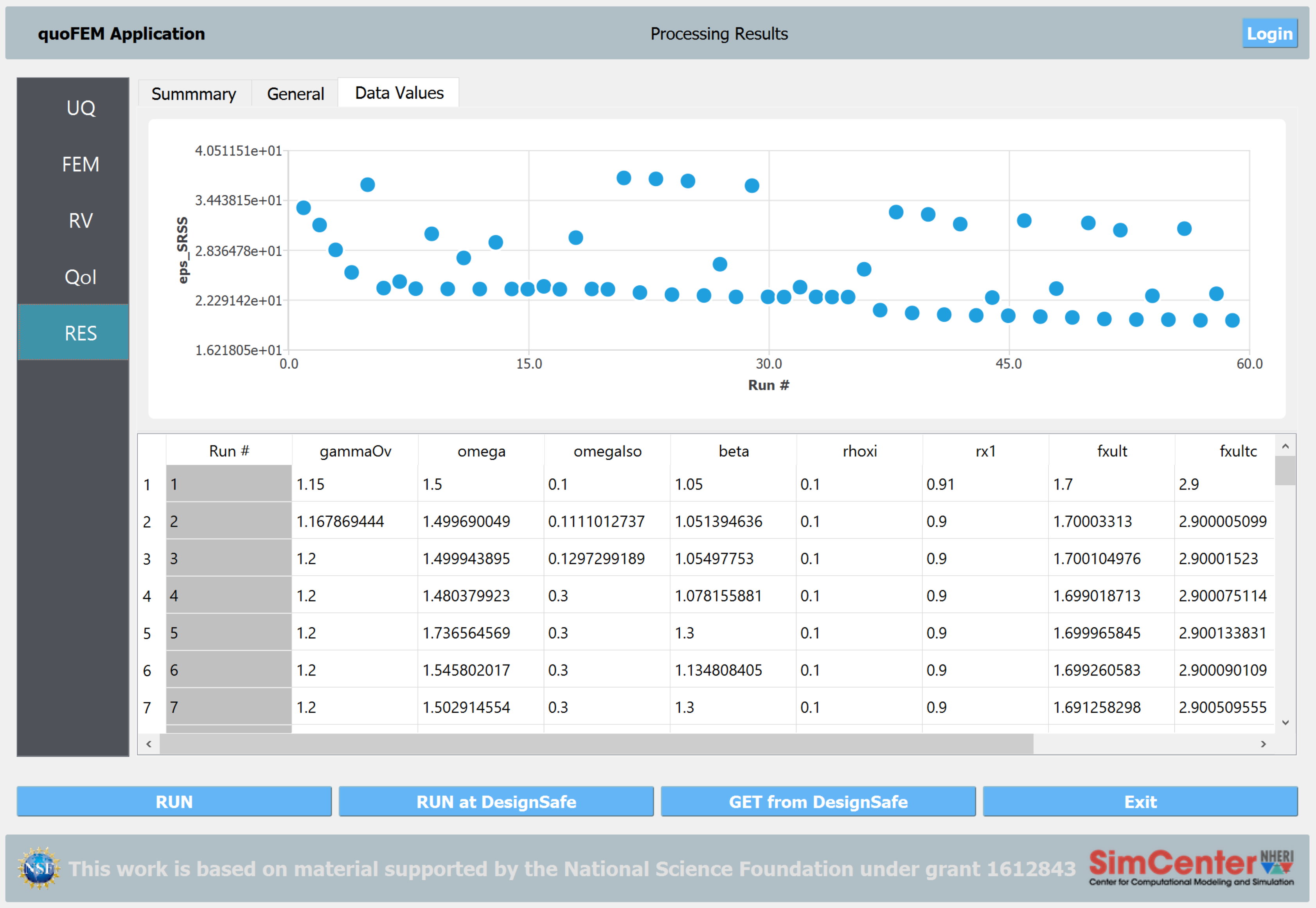


Calibrate component model

quoFEM

result visualization

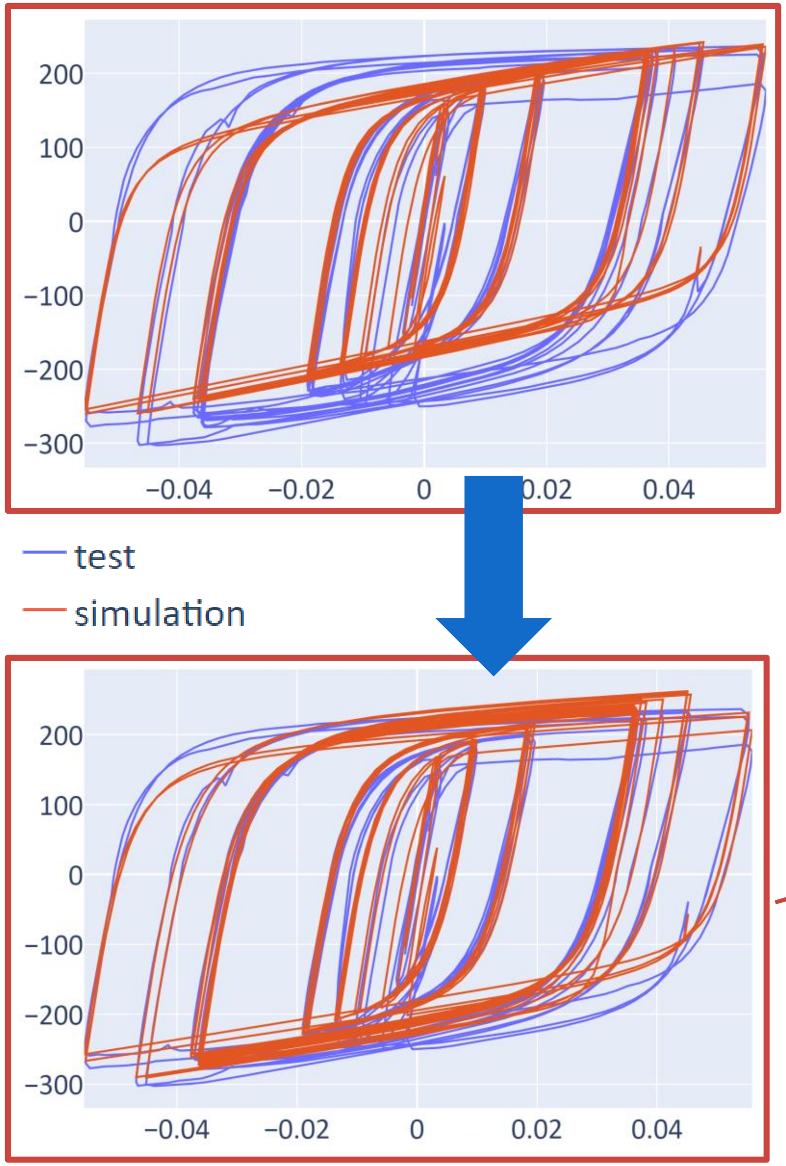
scatterplot shows convergence



	Run #	gammaOv	omega	omegaIso	beta	rhoxi	rx1	fxult	fxultc
1	1	1.15	1.5	0.1	1.05	0.1	0.91	1.7	2.9
2	2	1.167869444	1.499690049	0.1111012737	1.051394636	0.1	0.9	1.70003313	2.900005099
3	3	1.2	1.499943895	0.1297299189	1.05497753	0.1	0.9	1.700104976	2.90001523
4	4	1.2	1.480379923	0.3	1.078155881	0.1	0.9	1.699018713	2.900075114
5	5	1.2	1.736564569	0.3	1.3	0.1	0.9	1.699965845	2.900133831
6	6	1.2	1.545802017	0.3	1.134808405	0.1	0.9	1.699260583	2.900090109
7	7	1.2	1.502914554	0.3	1.3	0.1	0.9	1.691258298	2.900509555

Calibrate component model

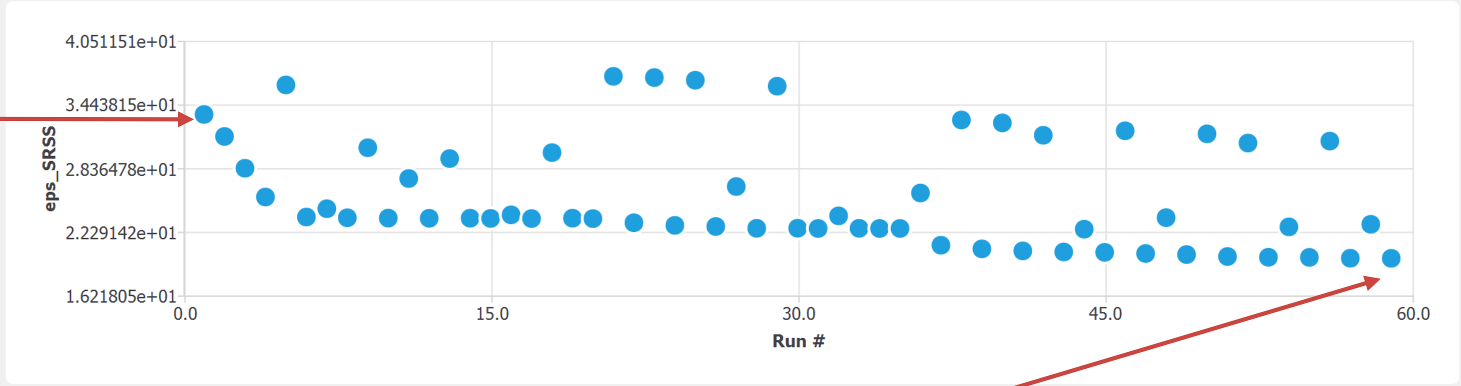
quoFEM



quoFEM Application | Processing Results | Login

UQ
FEM
RV
QoI
RES

Summmary | General | Data Values



	Run #	gammaOv	omega	omegaIso	beta	rhoxi	rx1	fxult	fxultc
1	1	1.15	1.5	0.1	1.05	0.1	0.91	1.7	2.9
2	2	1.167869444	1.499690049	0.1111012737	1.051394636	0.1	0.9	1.70003313	2.900005099
3	3	1.2	1.499943895	0.1297299189	1.05497753	0.1	0.9	1.700104976	2.90001523
4	4	1.2	1.480379923	0.3	1.078155881	0.1	0.9	1.699018713	2.900075114
5	5	1.2	1.736564569	0.3	1.3	0.1	0.9	1.699965845	2.900133831
6	6	1.2	1.545802017	0.3	1.134808405	0.1	0.9	1.699260583	2.900090109
7	7	1.2	1.502914554	0.3	1.3	0.1	0.9	1.691258298	2.900509555

RUN | RUN at DesignSafe | GET from DesignSafe | Exit

This work is based on material supported by the National Science Foundation under grant 1612843

SimCenter NHERI Center for Computational Modeling and Simulation

Case study

example application: Buckling Restrained Braced Frames

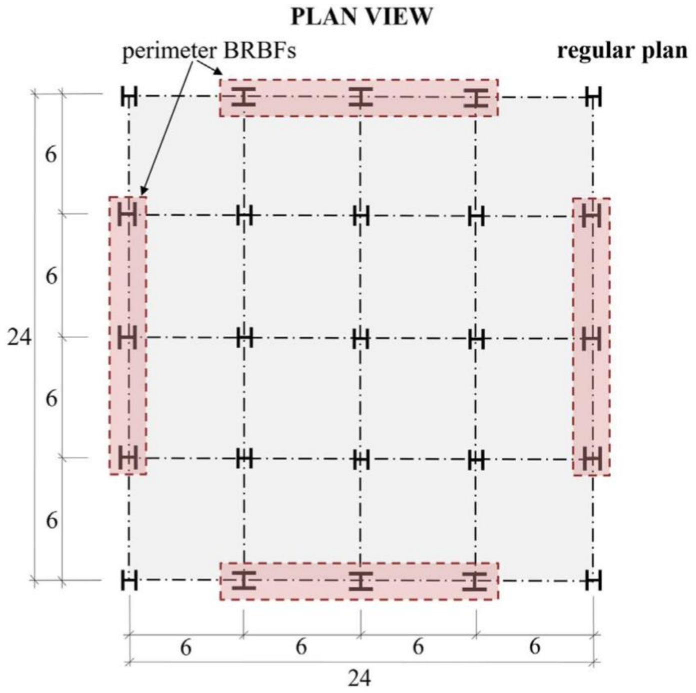
1. Design an experiment — quoFEM
2. Calibrate a numerical component model — quoFEM
3. **Simulate structural response — EE-UQ**
4. Estimate damage and losses — PBE

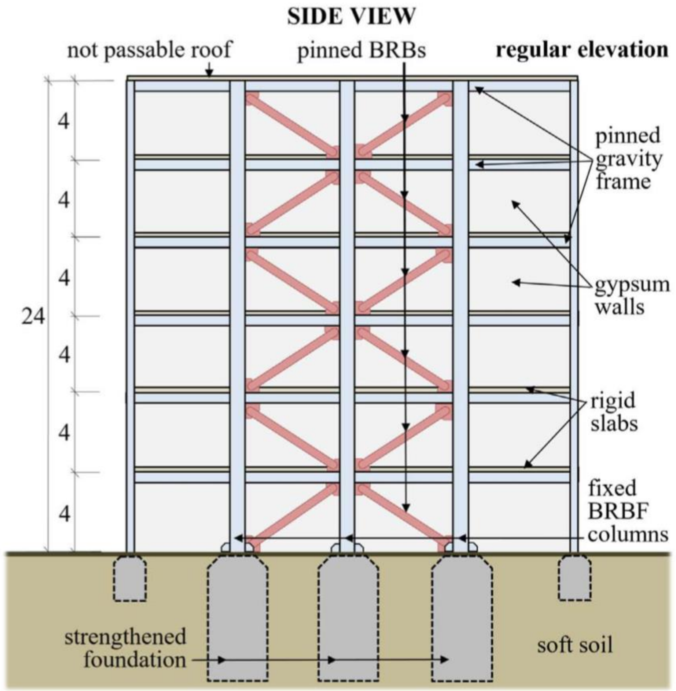
Simulate structural response

objective: estimate interstory drifts and floor accelerations in an earthquake scenario

problem: complex, computationally expensive calculations

EE-UQ: conveniently run simulations through DesignSafe





Simulate structural response

Prepare:

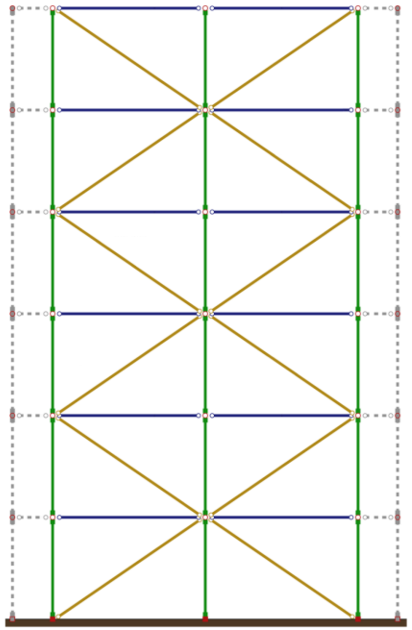
- BRBF model in OpenSees
- ground motion records from PEER NGA2

Adam Zsarnóczay adamzs@stanford.edu

Simulate structural response

Prepare:

- **BRBF model in OpenSees**
- ground motion records from PEER NGA2



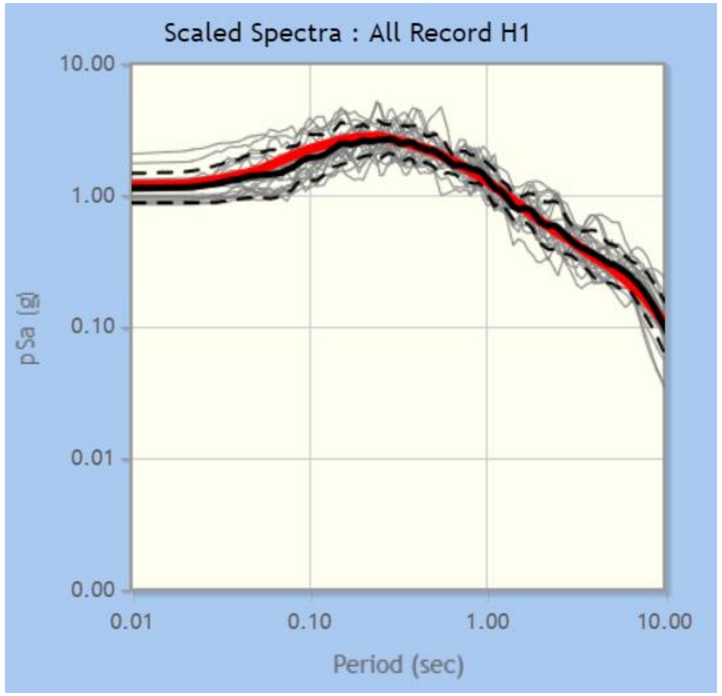
```
# load the core methods
source core/setAnalysis.tcl
source core/setRecorders.tcl
source core/Basics.tcl

# load the model info
source BRBF/Zsarnoczay_Vigh_2017/6AHHD.tcl

# specify the story height and the nodes
# for drift monitoring
set h {4.0 4.0 4.0 4.0 4.0 4.0}
set driftNodes {0 100 200 300 400 500 600}

# build the model
set pushover 0
source core/ModelBuilder.tcl
```

BRBF.tcl

Simulate structural response

Prepare:

- BRBF model in OpenSees
- **ground motion records from PEER NGA2**

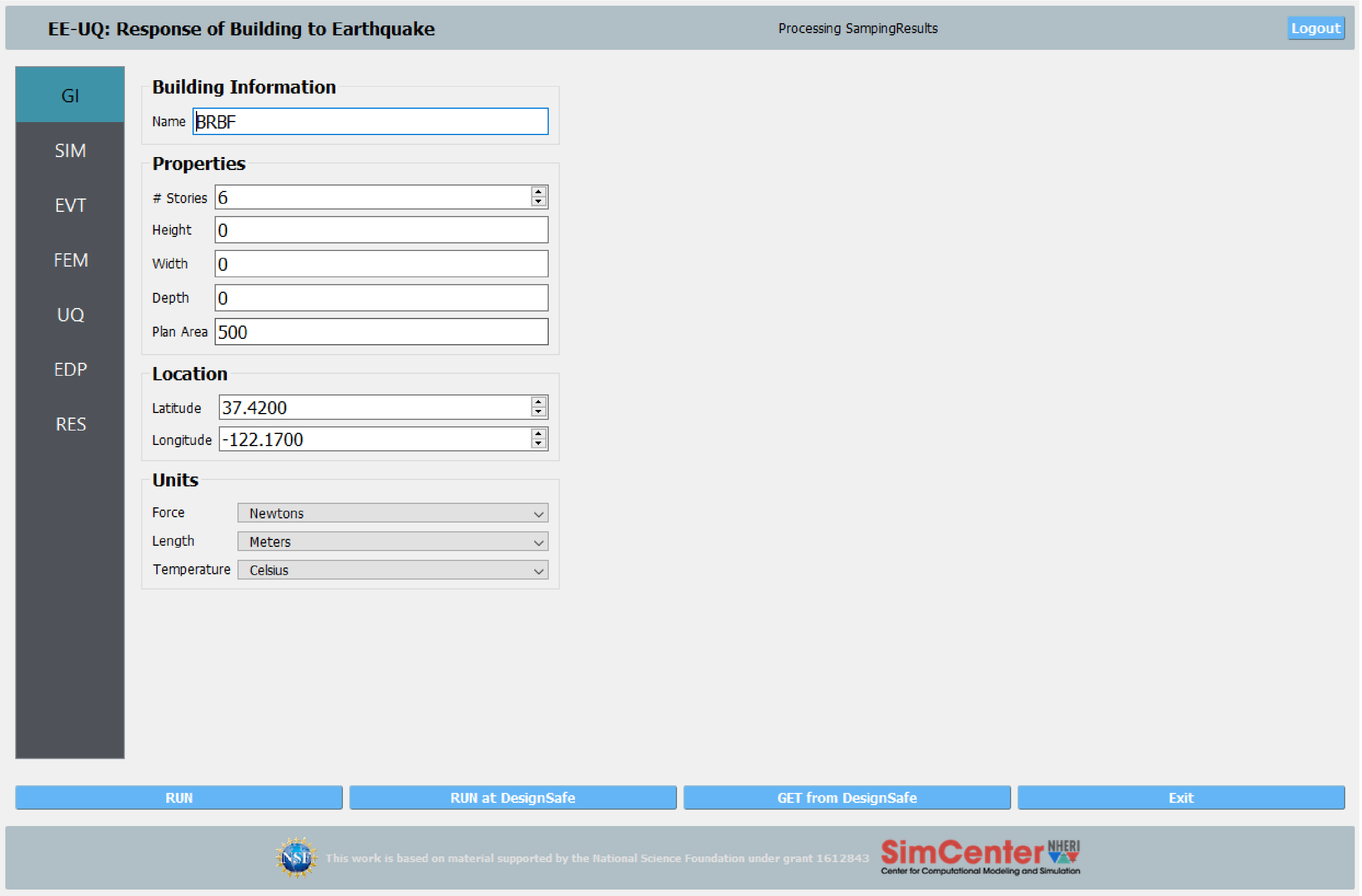



```
RSN143_TABAS_TAB-L1.AT2,1.4637
RSN1193_CHICHI_CHY024-E.AT2,3.1806
RSN1489_CHICHI_TCU049-E.AT2,3.9083
RSN1511_CHICHI_TCU076-E.AT2,3.1206
RSN1521_CHICHI_TCU089-E.AT2,3.5561
RSN1524_CHICHI_TCU095-E.AT2,3.2244
RSN1546_CHICHI_TCU122-E.AT2,3.9546
RSN1549_CHICHI_TCU129-E.AT2,2.0144
RSN1605_DUZCE_DZC180.AT2,2.3368
RSN3750_CAPEMEND_LFS270.AT2,3.718
RSN5827_SIERRA.MEX_MDO000.AT2,2.4069
RSN5829_SIERRA.MEX_RII000.AT2,2.7287
RSN5975_SIERRA.MEX_CXO360.AT2,3.6221
RSN5991_SIERRA.MEX_E10320.AT2,3.1185
RSN6890_DARFIELD_CMHSN10E.AT2,3.7356
RSN6893_DARFIELD_DFHSS17E.AT2,3.6641
RSN6906_DARFIELD_GDLCN55W.AT2,1.5617
RSN6911_DARFIELD_HORCN18E.AT2,1.9323
RSN6923_DARFIELD_KPOCN15E.AT2,3.4101
RSN8161_SIERRA.MEX_E12360.AT2,2.8477
```

Records.txt

Simulate structural response

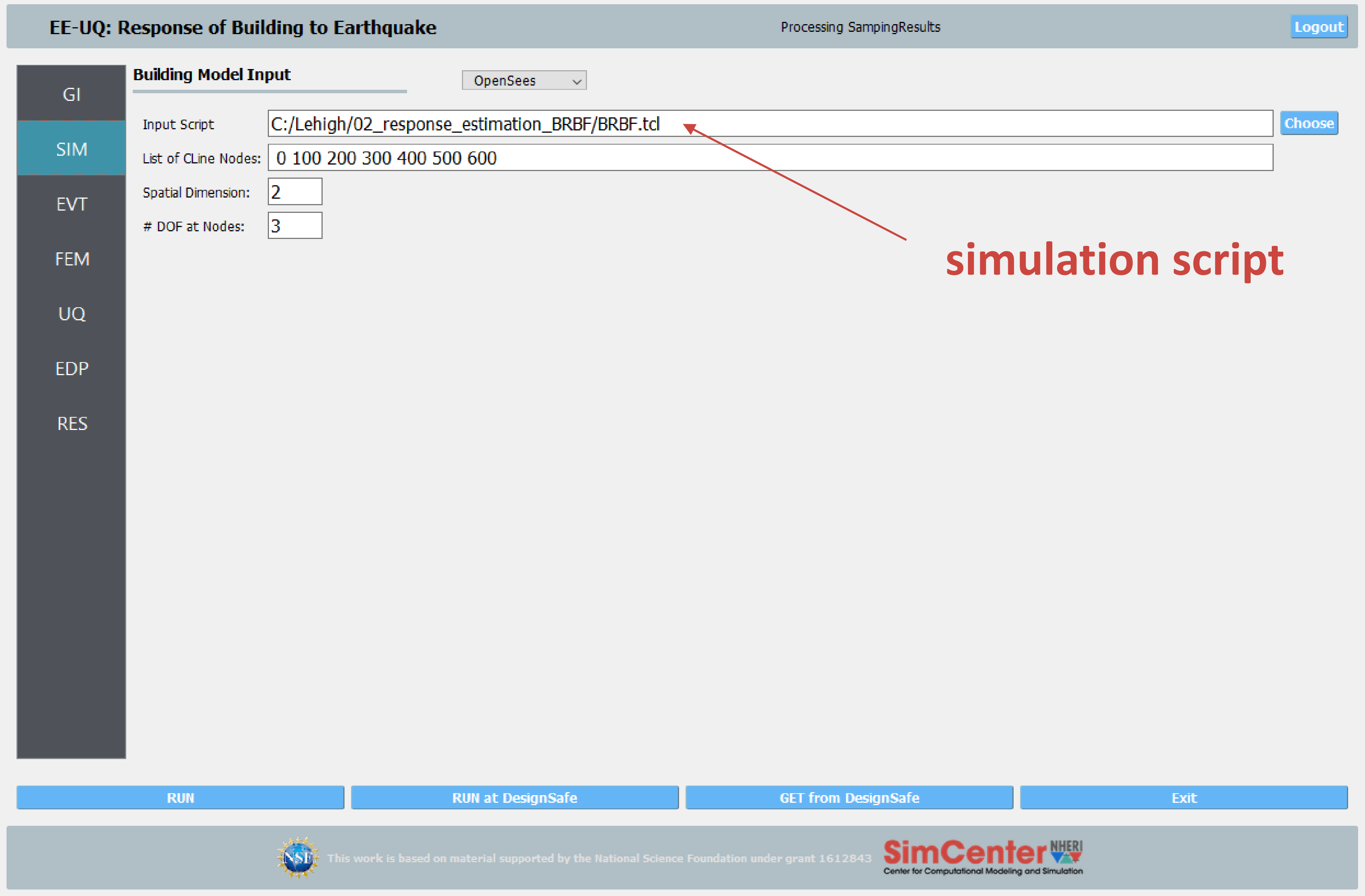
general information



Simulate structural response

EE-UQ

load the pre-defined simulation script



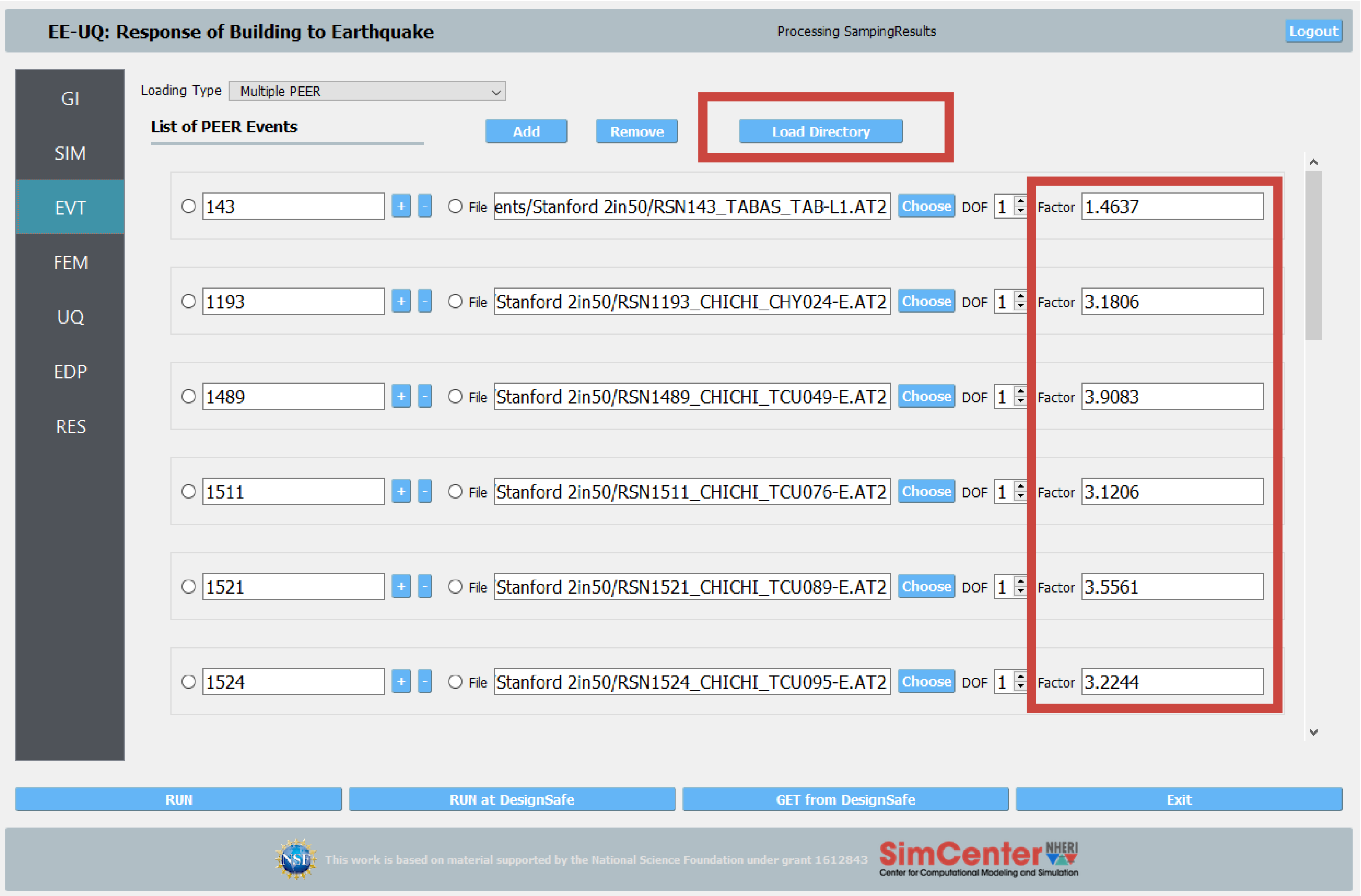
Simulate structural response

EE-UQ

load the selected ground motions

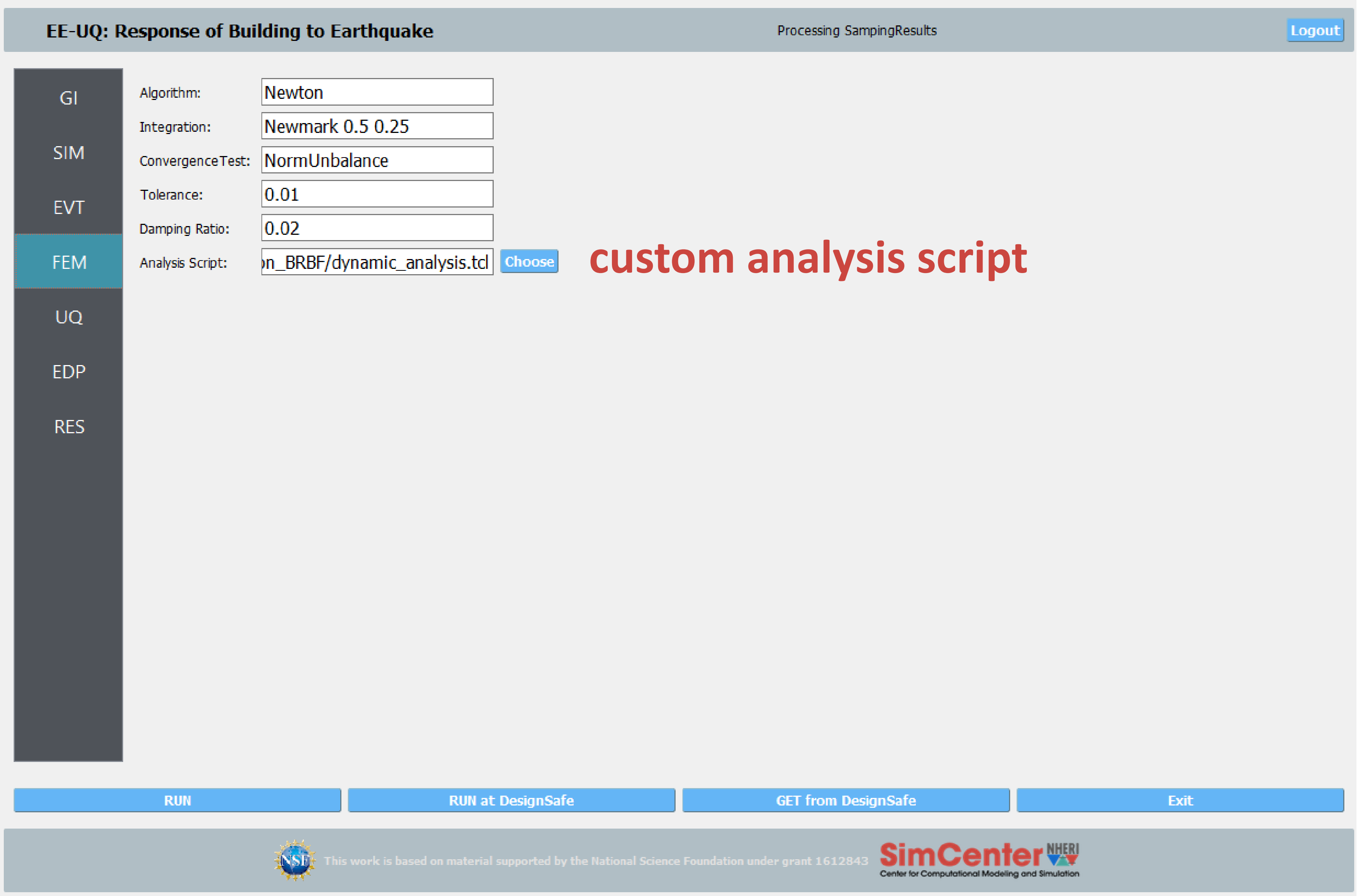
```
RSN143_TABAS_TAB-L1.AT2,1.4637
RSN1193_CHICHI_CHY024-E.AT2,3.1806
RSN1489_CHICHI_TCU049-E.AT2,3.9083
RSN1511_CHICHI_TCU076-E.AT2,3.1206
RSN1521_CHICHI_TCU089-E.AT2,3.5561
RSN1524_CHICHI_TCU095-E.AT2,3.2244
RSN1546_CHICHI_TCU122-E.AT2,3.9546
RSN1549_CHICHI_TCU129-E.AT2,2.0144
RSN1605_DUZCE_DZC180.AT2,2.3368
RSN3750_CAPEMEND_LFS270.AT2,3.718
RSN5827_SIERRA.MEX_MDO000.AT2,2.4069
RSN5829_SIERRA.MEX_RII000.AT2,2.7287
RSN5975_SIERRA.MEX_CXO360.AT2,3.6221
RSN5991_SIERRA.MEX_E10320.AT2,3.1185
RSN6890_DARFIELD_CMHSN10E.AT2,3.7356
RSN6893_DARFIELD_DFHSS17E.AT2,3.6641
RSN6906_DARFIELD_GDLCN55W.AT2,1.5617
RSN6911_DARFIELD_HORCN18E.AT2,1.9323
RSN6923_DARFIELD_KPOCN15E.AT2,3.4101
RSN8161_SIERRA.MEX_E12360.AT2,2.8477
```

Records.txt



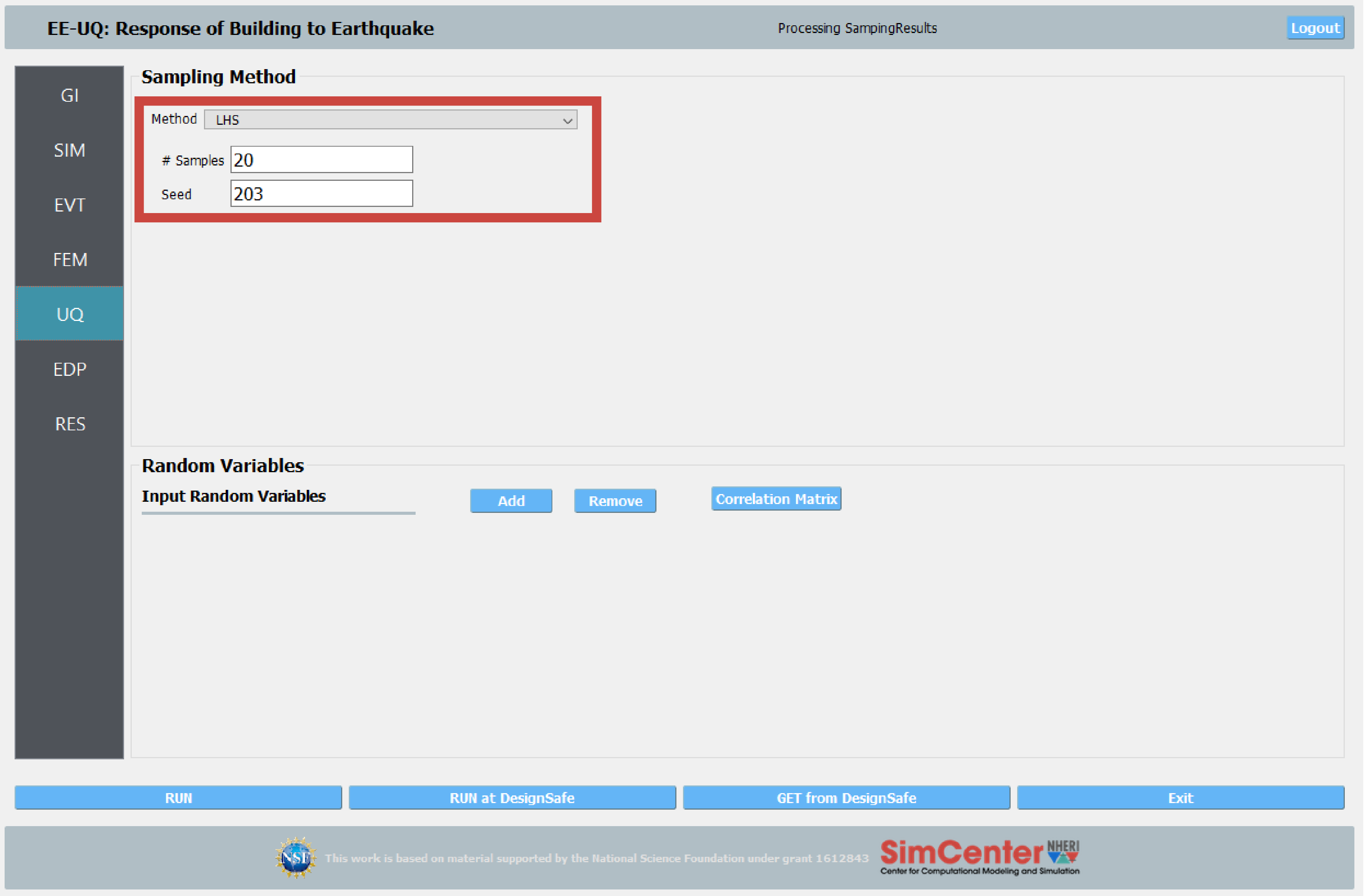
Simulate structural response

set up the analysis



Simulate structural response

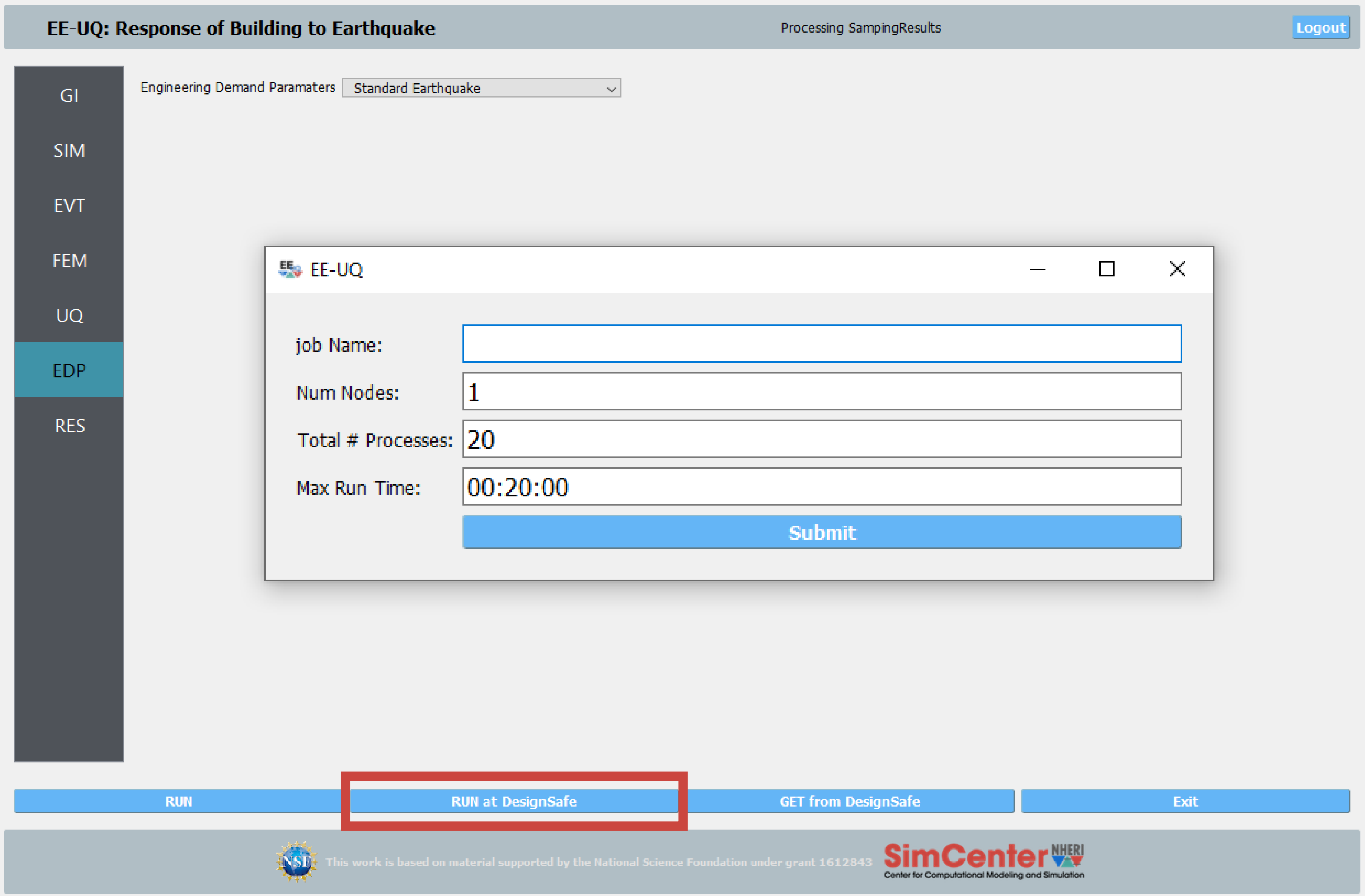
define random variables



Simulate structural response

choose EDPs

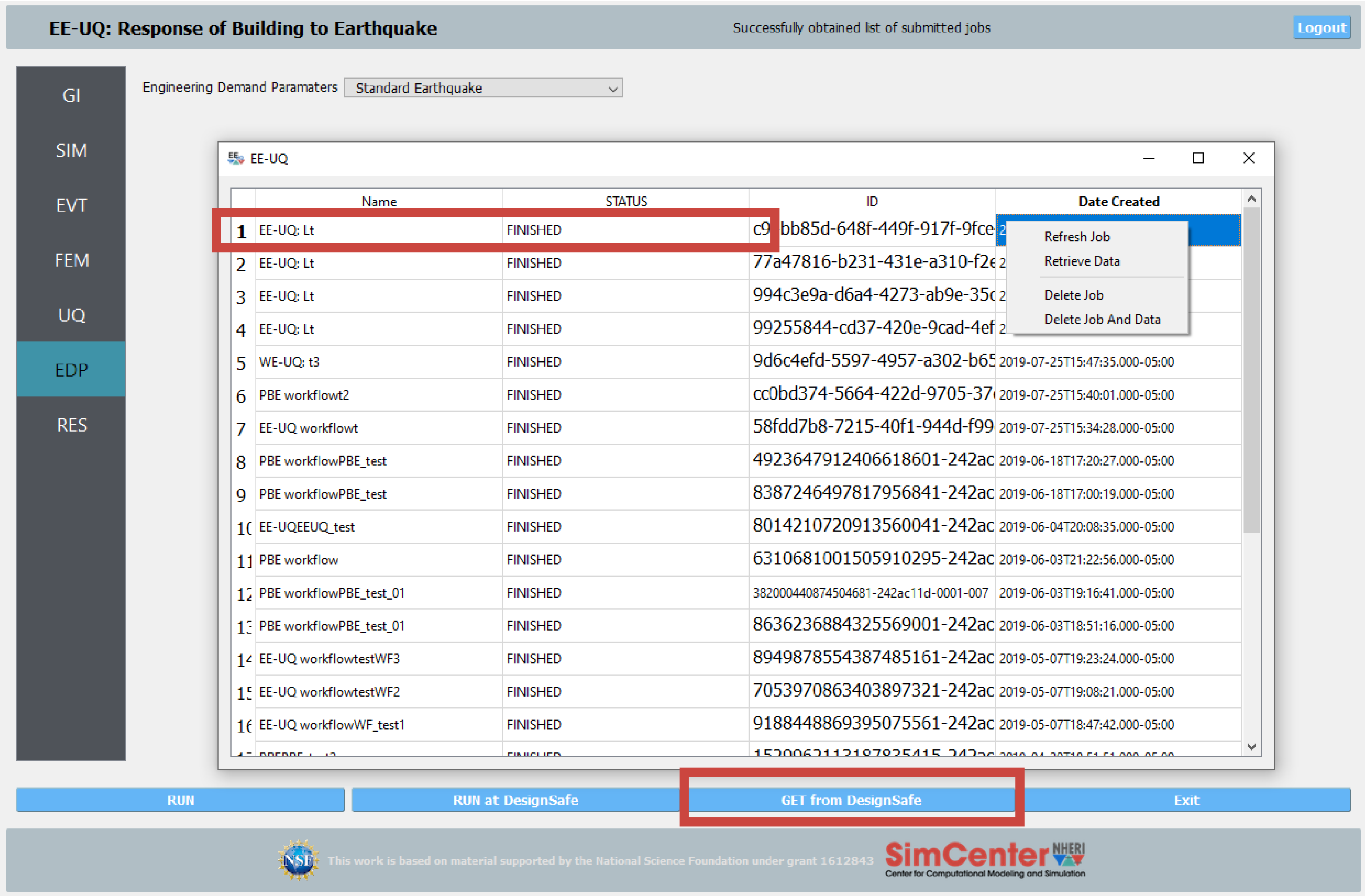
run remotely!



Simulate structural response

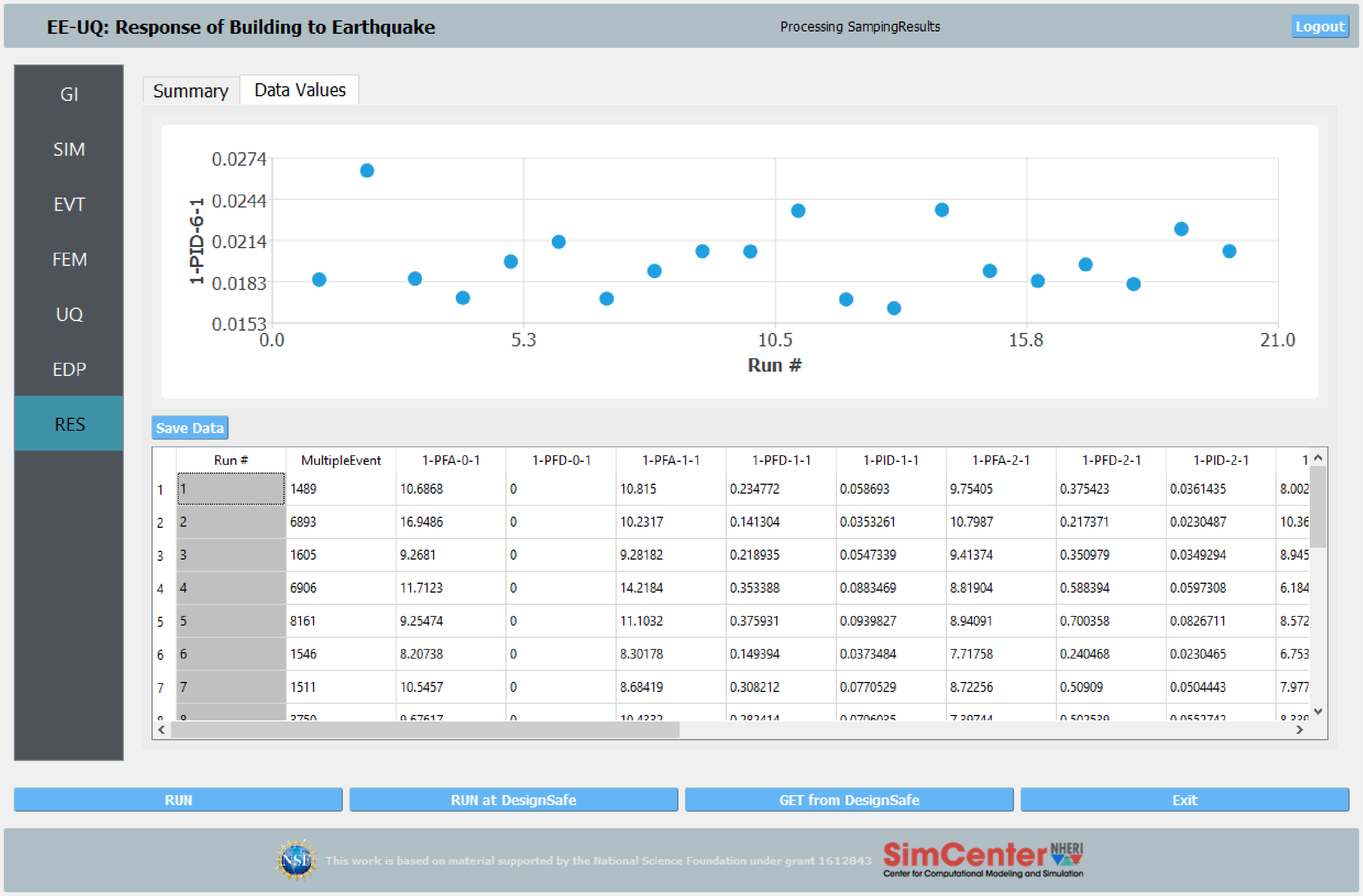
EE-UQ

collect results from DesignSafe



Simulate structural response

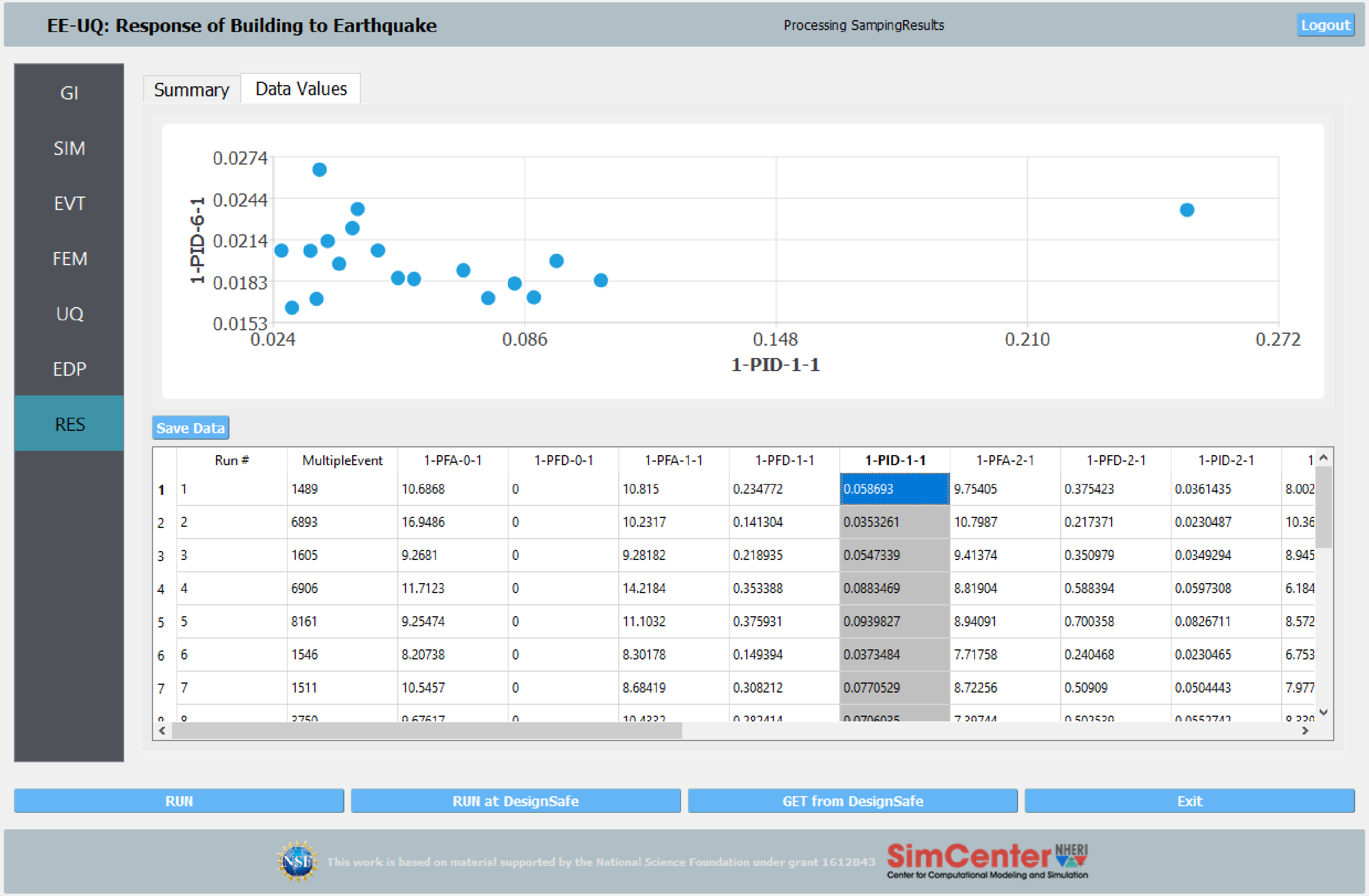
show EDPs for each analysis



Simulate structural response

EE-UQ

show joint
EDP distribution



	Run #	MultipleEvent	1-PFA-0-1	1-PFD-0-1	1-PFA-1-1	1-PFD-1-1	1-PID-1-1	1-PFA-2-1	1-PFD-2-1	1-PID-2-1	1
1	1	1489	10.6868	0	10.815	0.234772	0.058693	9.75405	0.375423	0.0361435	8.002
2	2	6893	16.9486	0	10.2317	0.141304	0.0353261	10.7987	0.217371	0.0230487	10.36
3	3	1605	9.2681	0	9.28182	0.218935	0.0547339	9.41374	0.350979	0.0349294	8.945
4	4	6906	11.7123	0	14.2184	0.353388	0.0883469	8.81904	0.588394	0.0597308	6.184
5	5	8161	9.25474	0	11.1032	0.375931	0.0939827	8.94091	0.700358	0.0826711	8.572
6	6	1546	8.20738	0	8.30178	0.149394	0.0373484	7.71758	0.240468	0.0230465	6.753
7	7	1511	10.5457	0	8.68419	0.308212	0.0770529	8.72256	0.50909	0.0504443	7.977
[illegible]	8	[illegible]	[illegible]	0	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Case study

example application: Buckling Restrained Braced Frames

1.	Design an experiment	quoFEM
2.	Calibrate a numerical component model	quoFEM
3.	Simulate structural response	EE-UQ
4.	**Estimate damage and losses**	**PBE**

Estimate damage and loss

setting: damage and loss assessment of a BRBF

objective: estimate damage and losses in an earthquake scenario

problem: connect response estimation with loss assessment
develop and use a custom damage and loss model for BRBF

PBE: conveniently run the whole workflow using pelicun

Estimate damage and loss

Prepare:

- custom components for BRBF loss model
- building information

Estimate damage and loss

PBE

Prepare:

- **custom components for BRBF loss model**
- building information


```
{
  "Name": "Steel Buckling Restrained Brace (BRB), Single
    Diagonal brace, Weight of brace > 41 plf and < 99 plf.",
  "QuantityUnit": [
    1,
    "EA"
  ],
  "Directional": true,
  "Correlated": false,
  "EDP": {
    "Type": "Story Drift Ratio",
    "Unit": [
      1,
      "ea"
    ],
    "Offset": 0
  },
  "GeneralInformation": {
    "ID": "B1033.111b",
    "Description": "None",
    "Author": "John Wallace",
    "Official": true,
    "DateCreated": "2012-10-12T16:47:42.0245683-07:00",
    "Approved": true,
    "Incomplete": false,
    "Notes": "None"
  },
```


B1033.111b.json

Estimate damage and loss

Prepare:

- **custom components for BRBF loss model**
- building information


```
"DSGroups": [
  {
    "MedianEDP": 0.02,
    "Beta": 0.4,
    "CurveType": "LogNormal",
    "DSGroupType": "Single",
    "DamageStates": [
      {
        "Weight": 1.0,
        "LongLeadTime": false,
        "Consequences": {
          "ReconstructionCost": {
            "Amount": [
              58395.7011,
              39709.0768
            ],
            "Quantity": [
              3.0,
              7.0
            ],
            "CurveType": "Normal",
            "Beta": 0.2986,
            "Bounds": [
              0,
              "None"
            ]
```


B1033.111b.json

Estimate damage and loss

Prepare:

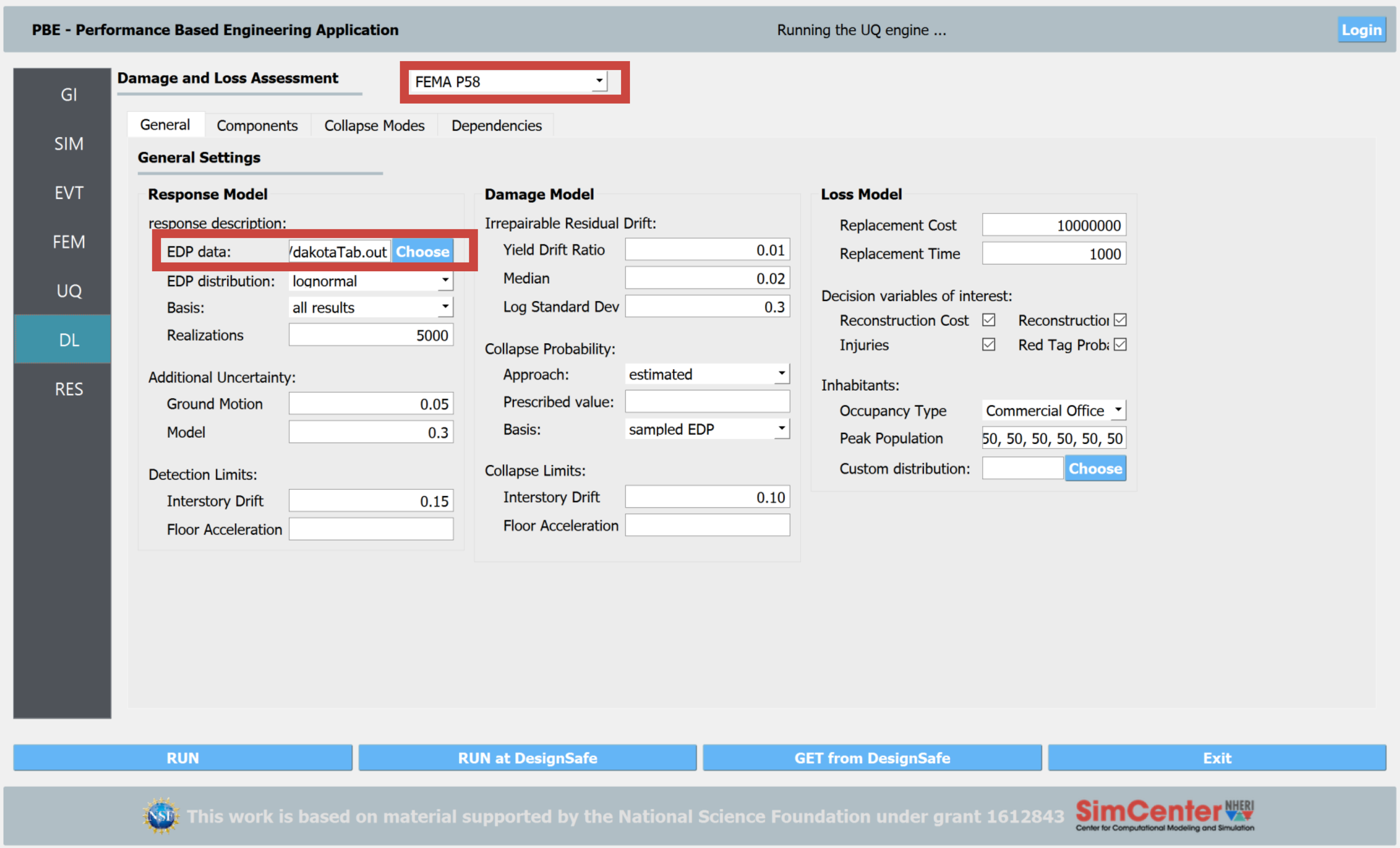
- custom components for BRBF loss model
- **building information**

replacement cost: $10,000,000
replacement time: 1000 days
population: 300
occupancy type: Commercial Office

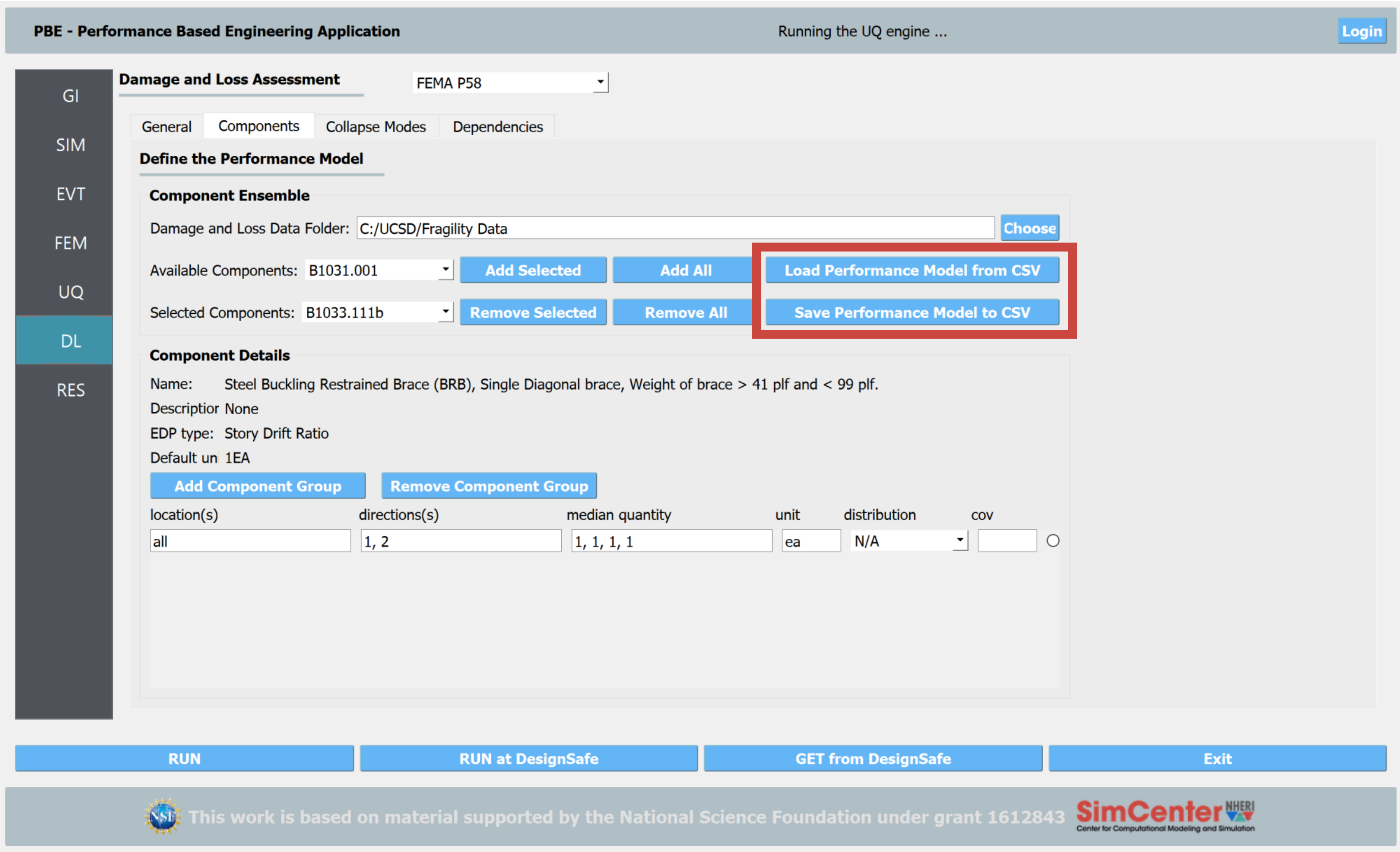
types, quantities, locations of structural components and non-structural components

collapse modes and consequences

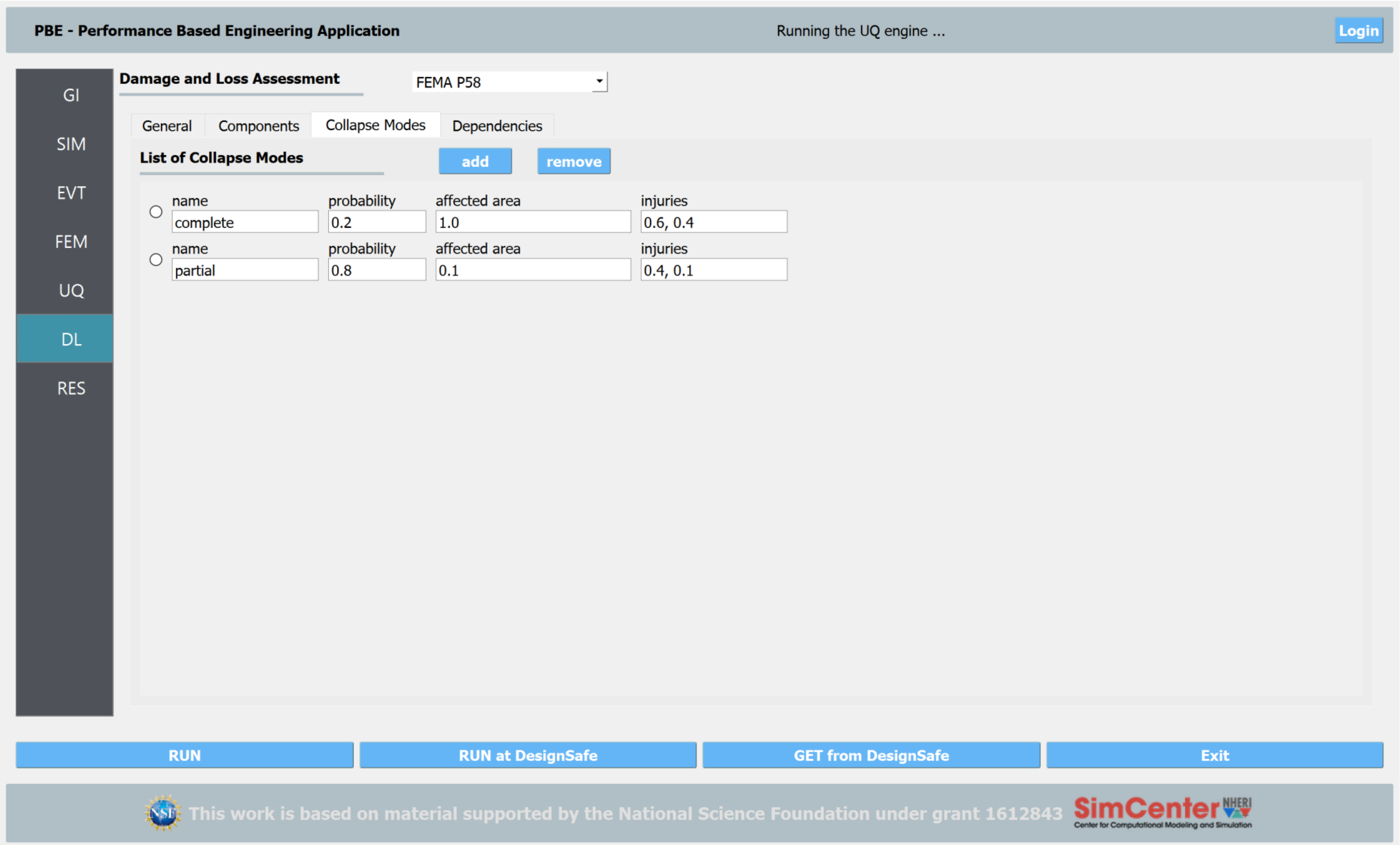
Estimate damage and loss



Estimate damage and loss



Estimate damage and loss



Estimate damage and loss

PBE - Performance Based Engineering Application   Running the UQ engine ...   Login

GI | SIM | EVT | FEM | UQ | DL | RES

Summary | Data Values

Decision Variable	Mean	Standard Dev.	Minimum	10^{th} Percentile	Median	90^{th} Percentile
event time: month	6.4364	3.48758	1	2	6	11
event time: weekday?	0.7084	0.454545	0	0	1	1
event time: hour	11.6982	6.86802	0	2	12	21
inhabitants:	88.8344	104.111	0	0	0	285
collapses: collapsed?	0.0998	0.299763	0	0	0	0
collapses: mode	0.785571	0.410837	0	0	1	1
red tagged?:	0.995557	0.0665184	0	1	1	1
reconstruction: irrepairable?	0.528549	0.49924	0	0	1	1
reconstruction: cost impractical?	0.11263	0.316214	0	0	0	1
reconstruction: cost	7.62358e+06	3.28379e+06	986410	2.39672e+06	1e+07	1e+07

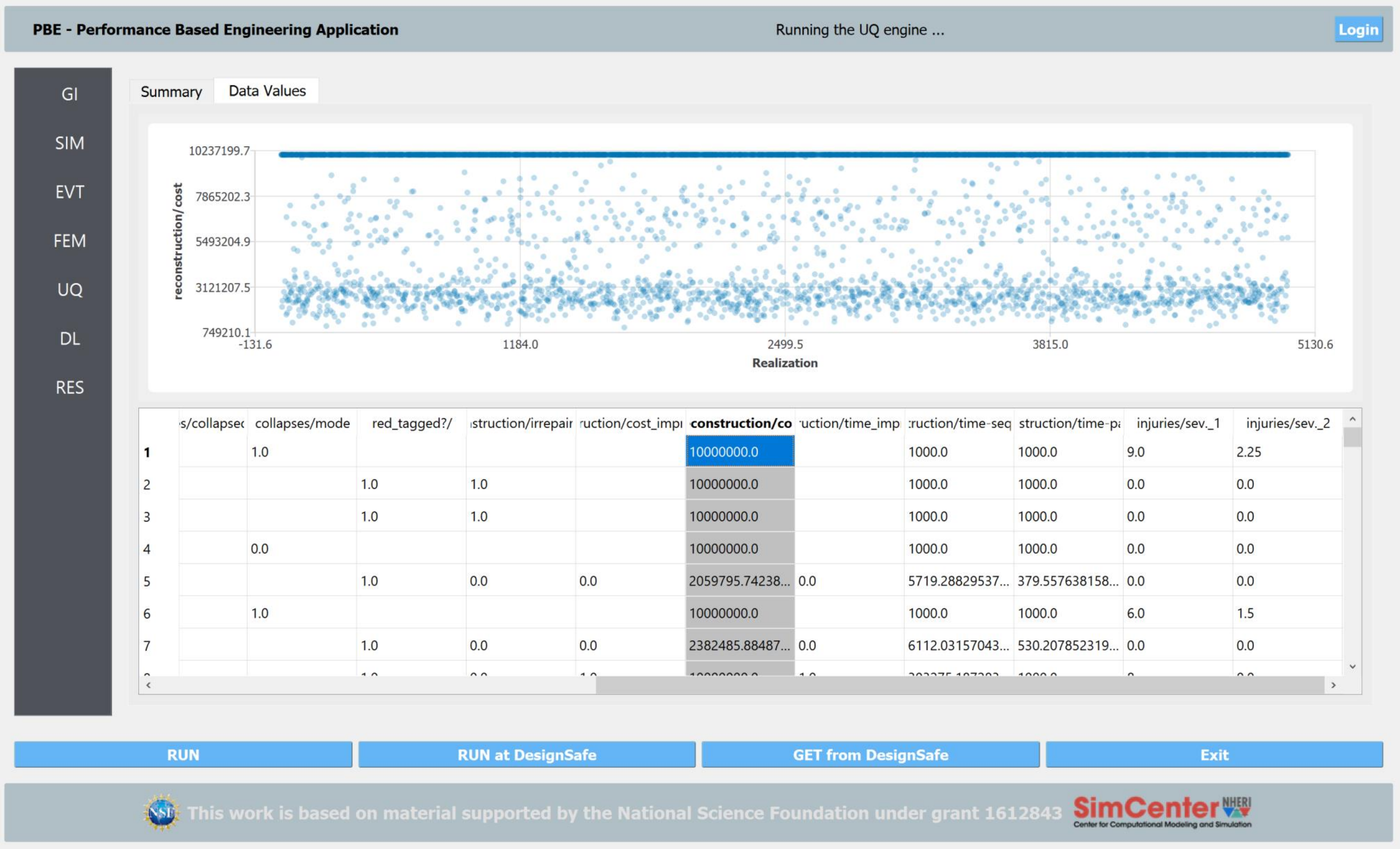
RUN | RUN at DesignSafe | GET from DesignSafe | Exit

This work is based on material supported by the National Science Foundation under grant 1612843

SimCenter NHERI Center for Computational Modeling and Simulation

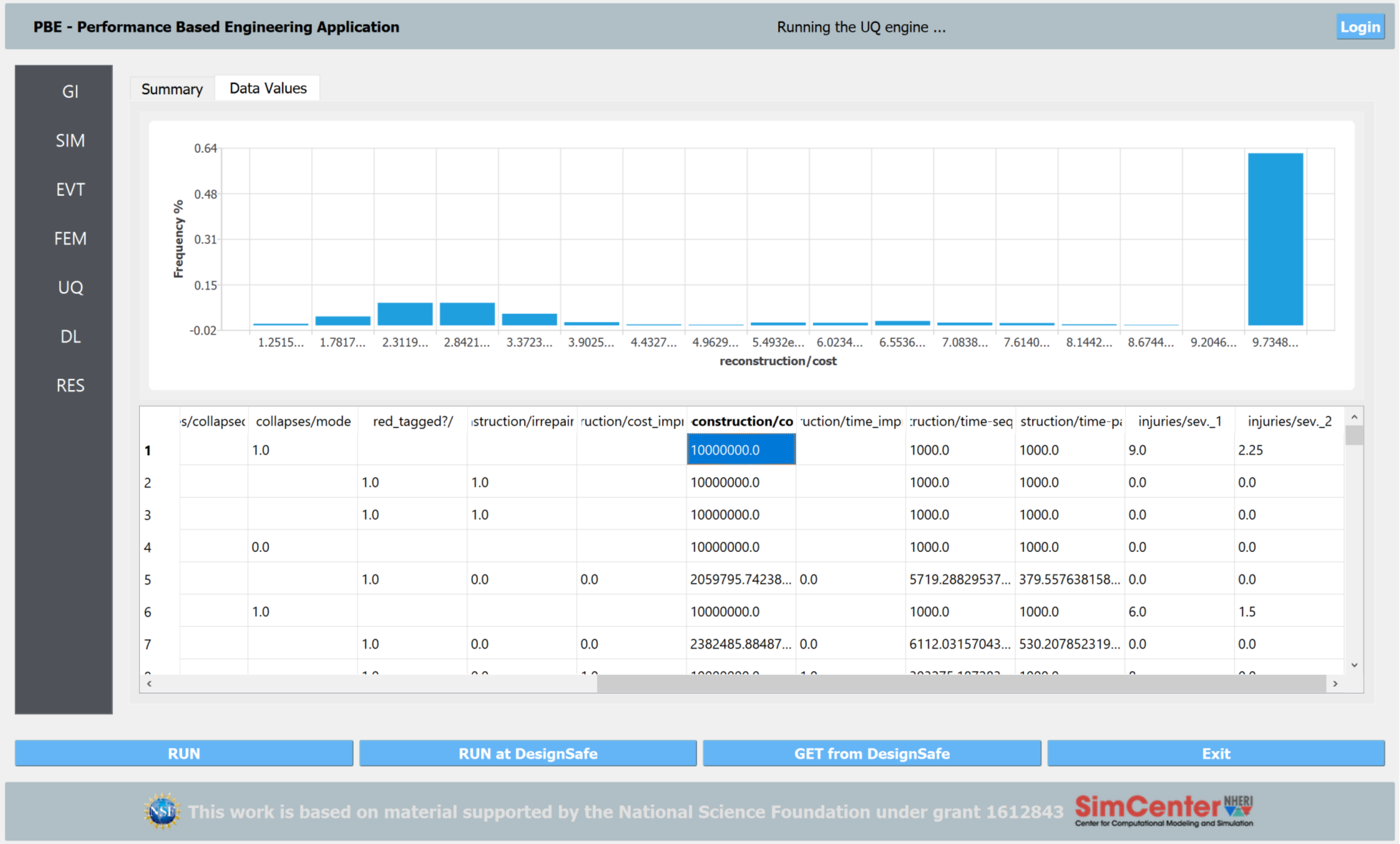
Estimate damage and loss

PBE



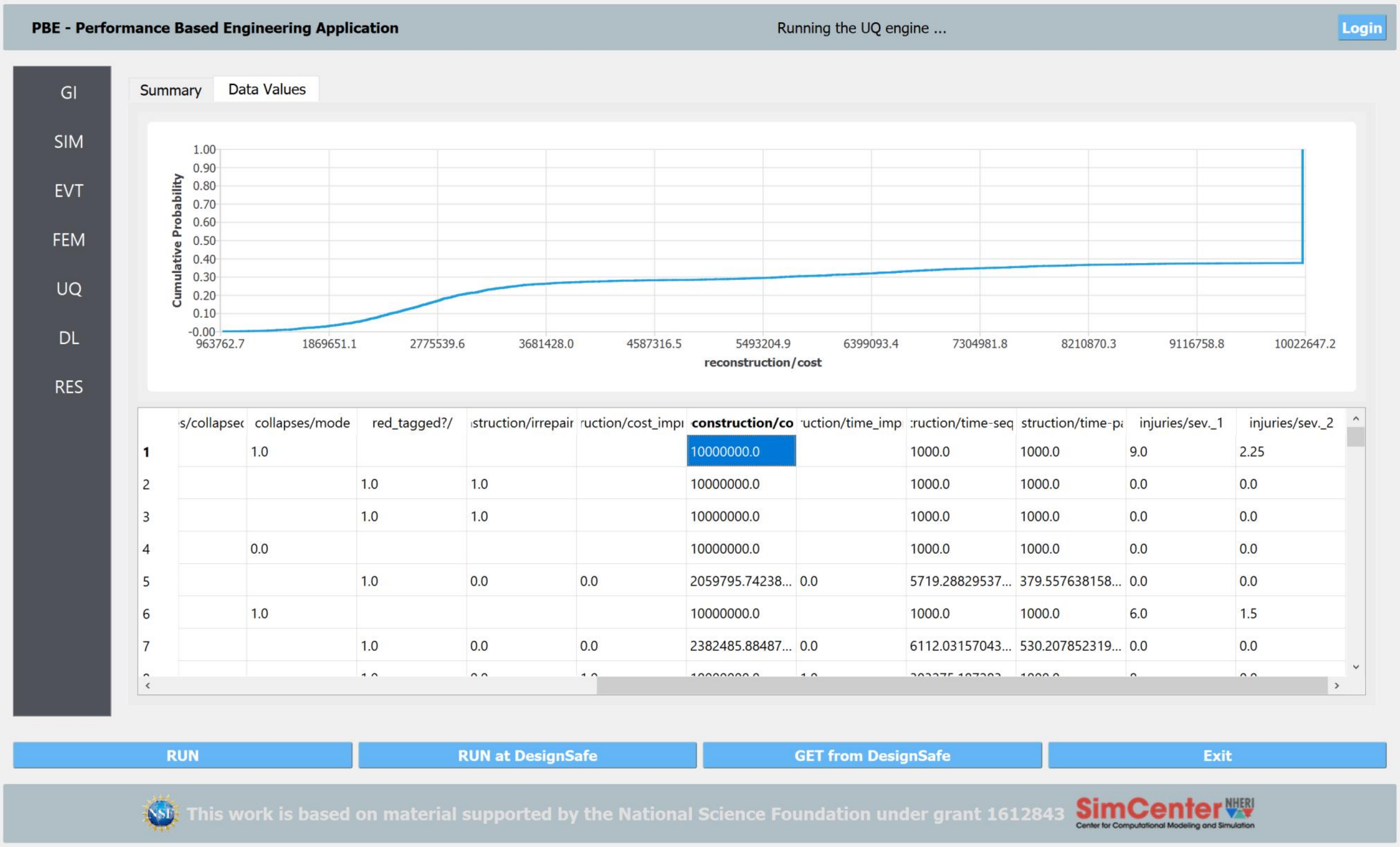
Estimate damage and loss

PBE



Estimate damage and loss

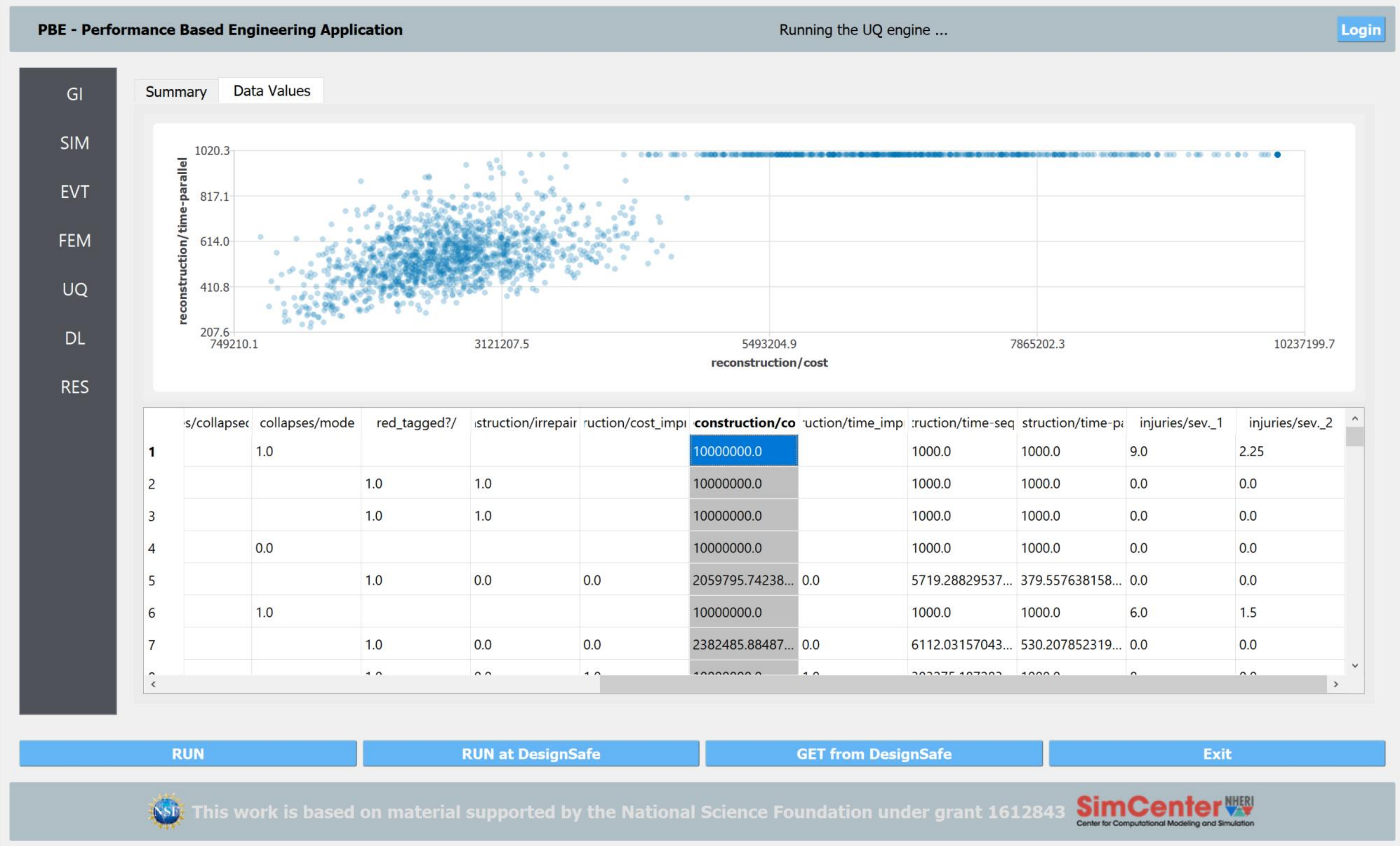
PBE



	s/collapsec	collapses/mode	red_tagged?/	istruction/irrepair	ruction/cost_impı	construction/co	uction/time_imp	ruction/time-seq	struction/time-pa	injuries/sev._1	injuries/sev._2
1		1.0				10000000.0		1000.0	1000.0	9.0	2.25
2			1.0	1.0		10000000.0		1000.0	1000.0	0.0	0.0
3			1.0	1.0		10000000.0		1000.0	1000.0	0.0	0.0
4		0.0				10000000.0		1000.0	1000.0	0.0	0.0
5			1.0	0.0	0.0	2059795.74238...	0.0	5719.28829537...	379.557638158...	0.0	0.0
6		1.0				10000000.0		1000.0	1000.0	6.0	1.5
7			1.0	0.0	0.0	2382485.88487...	0.0	6112.03157043...	530.207852319...	0.0	0.0

Estimate damage and loss

PBE



Research Tools

- **quoFEM** **Uncertainty Quantification and Optimization**

- **EE-UQ** **Structural Response Estimation under Earthquakes**
- WE-UQ Structural Response Estimation under Wind
- CWE CFD Analysis for a Building

- **PBE** **Damage and Loss Assessment for a Building**
- rWHALE Damage and Loss Assessment for an Urban Region

Objectives

- Demonstrate use-cases of our research tools
- **Get feedback on existing features**
- **Collect requests for new features**

National Science Foundation

University of California at San Diego NHERI Natural Hazards Engineering Research Infrastructure

UC San Diego JACOBS SCHOOL OF ENGINEERING Structural Engineering

Example Use of Research Tools

https://simcenter.designsafe-ci.org/research-tools

Adam Zsarnóczay, Postdoctoral Researcher

Stanford University, SimCenter

adamzs@stanford.edu

Joint Researcher Workshop

UC San Diego, Lehigh & SimCenter

December 16-17, 2019

University of California, San Diego

LEHIGH NHERI Real-Time Multi-Directional Testing Facility

University of California at San Diego NHERI Natural Hazards Engineering Research Infrastructure

SIMCENTER COMPUTATIONAL MODELING AND SIMULATION CENTER