Calibration tools for Johnson-Cook and Bammann-Chiesa-Johnson (BCJ) Models

Summer 2023

Daniel Kenney

SMART Internship Project

Material Models: Johnson-Cook

- 5 parameters to describe plasticity:

$$\sigma_e = \left[A + B\left(\varepsilon_e^p\right)^n\right]\left[1 + C\ln\left(\frac{\dot{\varepsilon}_e^p}{\dot{\varepsilon}_0}\right)\right]\left[1 - \hat{T}^m\right]$$

$$\hat{T} = \frac{T - T_r}{T_m - T_r}$$

- 5 additional parameters to predict failure:

$$\varepsilon_f = \left[D_1 + D_2 e^{D_3\sigma^*}\right]\left[1 + D_4\ln(\dot{\varepsilon}^*)\right]\left[1 + D_5\hat{T}\right]$$

$$\sigma^* = \frac{\sigma_m}{\bar{\sigma}}$$

Model Parameters	
A	Yield coefficient
B	Strain hardening coefficient
n	Strain hardening exponent
C	Strain rate coefficient
m	Temperature sensitivity exponent

Material Models: BCJ

Model Variables	
$Y(T)$	Strain-rate independent yield stress
$V(T)$	Strain-rate sensitive yield component
$f(T)$	Strain-rate yielding sensitivity factor
κ	Isotropic hardening (ISV)
$\underline{\alpha}$	Kinematic hardening (ISV)
ϕ	Damage (ISV)

- Internal state variable (ISV) model with 3 ISVs and 19 constants
- Characteristic yield function:

$$\Psi = |\underline{\sigma}' - \underline{\alpha}| \underbrace{- \kappa -}_{\text{Hardening}} \underbrace{(1-\phi)}_{\text{Damage}} \underbrace{\left(Y - \mathrm{V} \operatorname{arcsinh}\left(\frac{\dot{\varepsilon}}{\mathrm{f}}\right)\right)}_{\text{Initial Yield}}$$

BCJ: Strain Hardening

- Isotropic hardening:

$$\dot{\kappa} = H\left|\underline{D}^{in}\right| - \left(R_d\left|\underline{D}^{in}\right| + \mathrm{R_s}\right)\kappa^2$$

- Kinematic hardening:

$$\dot{\underline{\alpha}} = h\underline{D}^{in} - \left(r_d\left|\underline{D}^{in}\right| + r_s\right)|\underline{\alpha}|\underline{\alpha}$$

Calibration Parameters	
H	Kinematic hardening term
R_d	Dynamic recovery of isotropic hardening
R_s	Strain-rate sensitive yield component
h	Isotropic hardening term
r_d	Strain-rate yielding sensitivity factor
r_s	Isotropic hardening (ISV)
$\underline{D}^{in}$	Inelastic rate of deformation

BCJ: Damage Accumulation

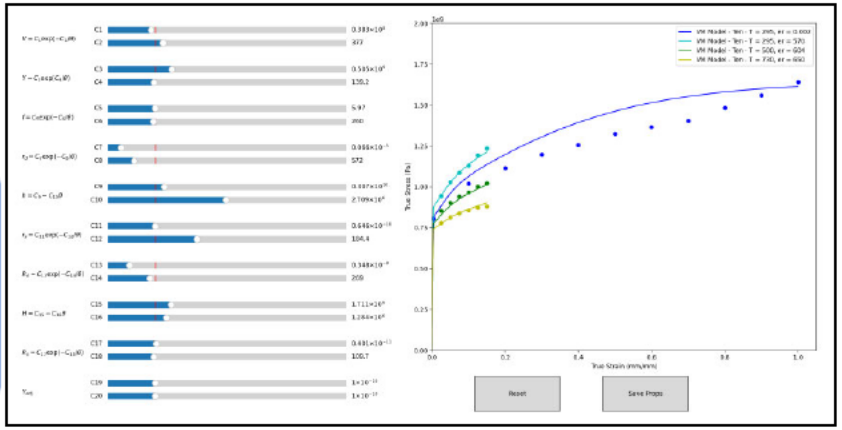
- Damage dependent on
- Damage:

$$\dot{\phi} = \chi \left[\frac{1}{(1-\phi)^m} - (1-\phi) \right] \left| \underline{D}^{in} \right|$$

$$\chi = \sinh\left(\frac{2(2m-1)p}{(2m+1)\bar{\sigma}} \right)$$

Calibration Parameters	
H	Kinematic hardening term
R_d	Dynamic recovery of isotropic hardening
R_s	Strain-rate sensitive yield component
h	Isotropic hardening term
r_d	Strain-rate yielding sensitivity factor
r_s	Isotropic hardening (ISV)
$\underline{D}^{in}$	Inelastic rate of deformation

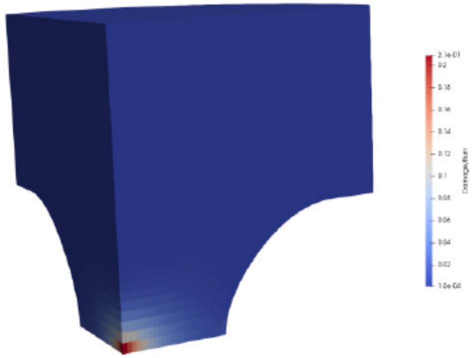
Workflow



Calibrate BCJ model to experimental data with BCJ GUI

Run Single Element verification to ensure calibrated parameters work in EPIC

Run EPIC simulations with notched specimens to calibrate damage parameter n



Simulate Taylor bar tests to validate model

Calibration Tools – Python GUIs

JC Props File

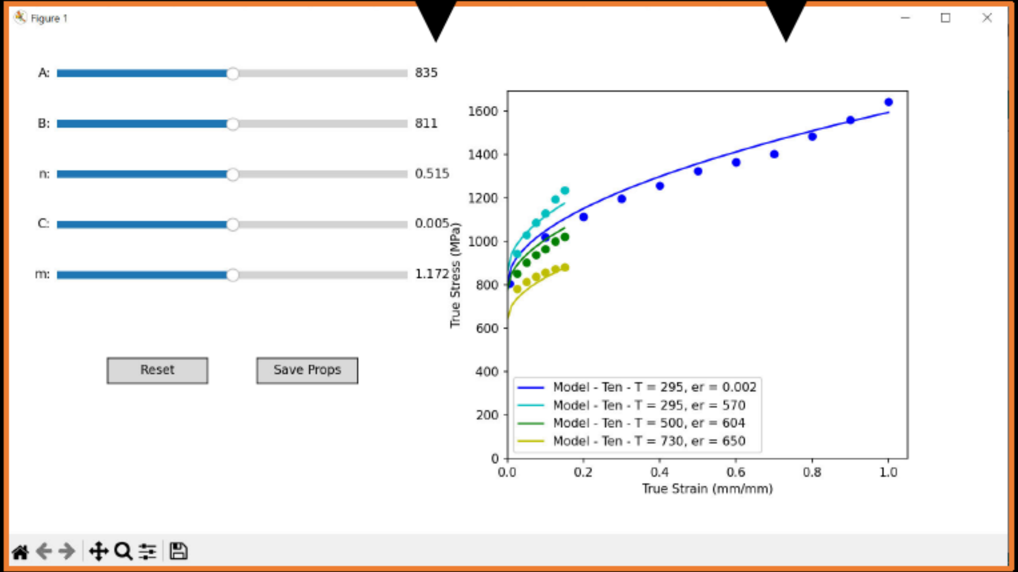
- Material Properties
- Initial Constants

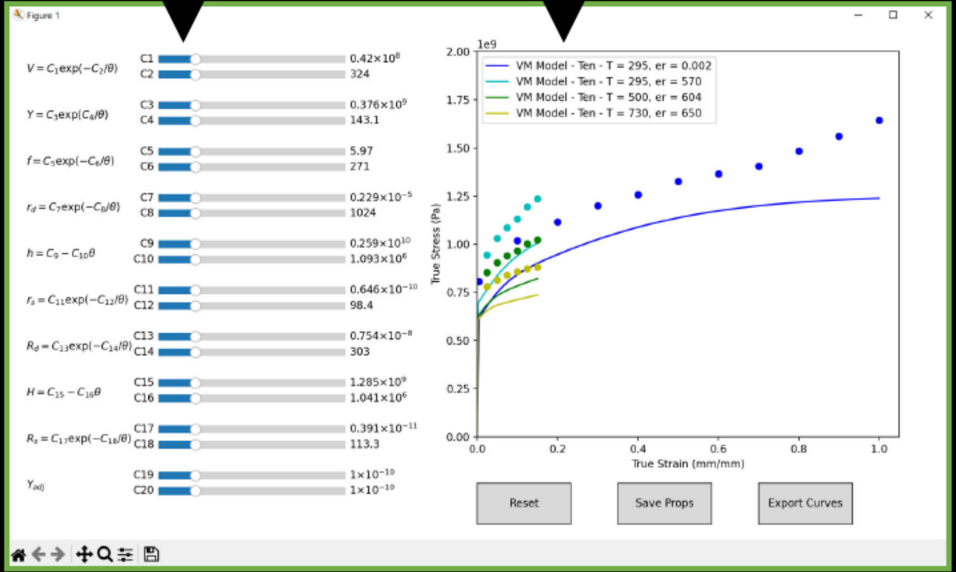
Data File(s)

- Experimental Data
- Experimental Conditions (T, $\dot{\varepsilon}$)

BCJ Props File

- Material Properties
- Initial Constants





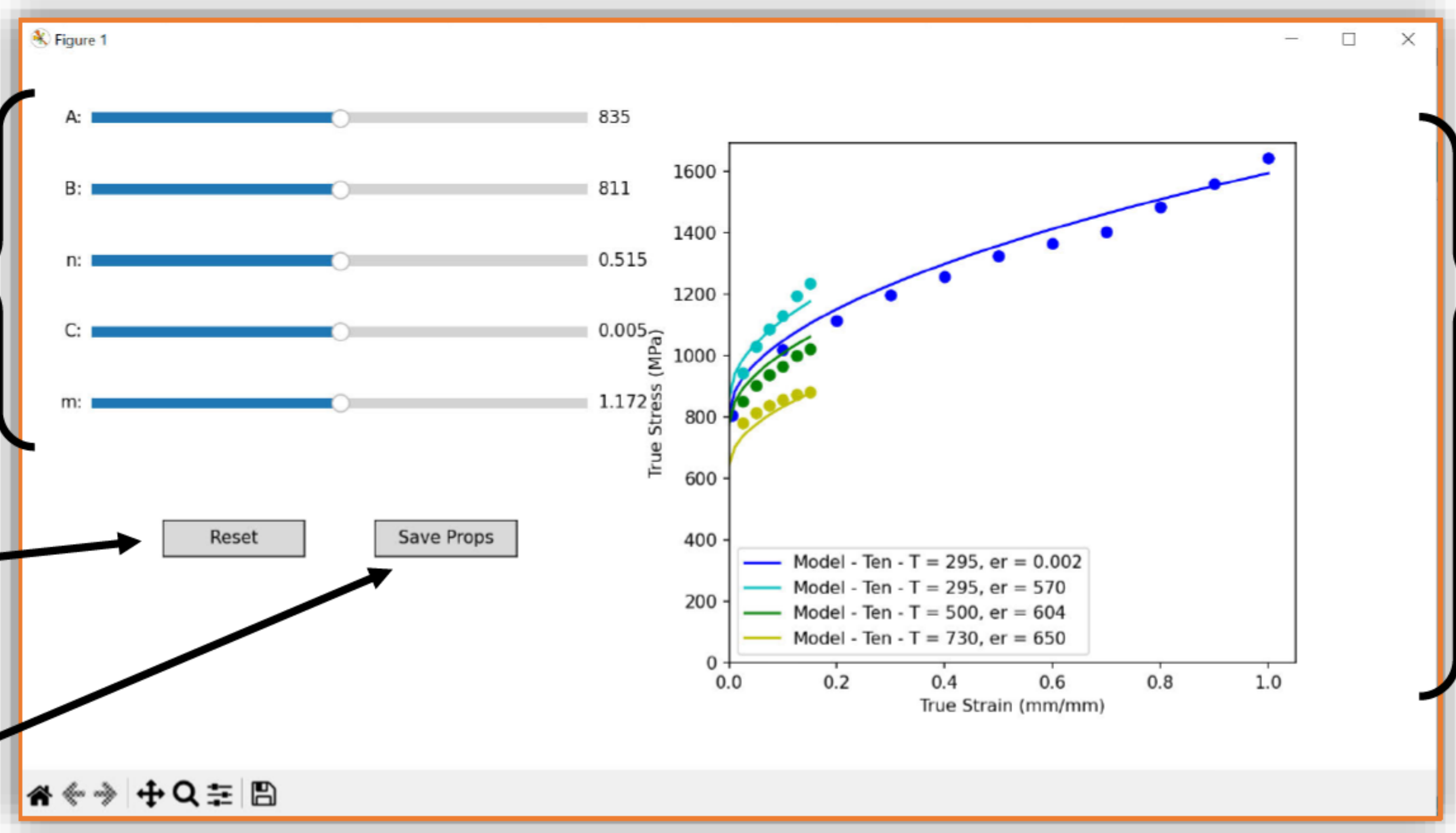
JC GUI Layout

Neither GUI incorporates damage!

Calibration sliders

Return sliders to their original values

Export constants to a new props file



Matplotlib plot

BCJ GUI Layout

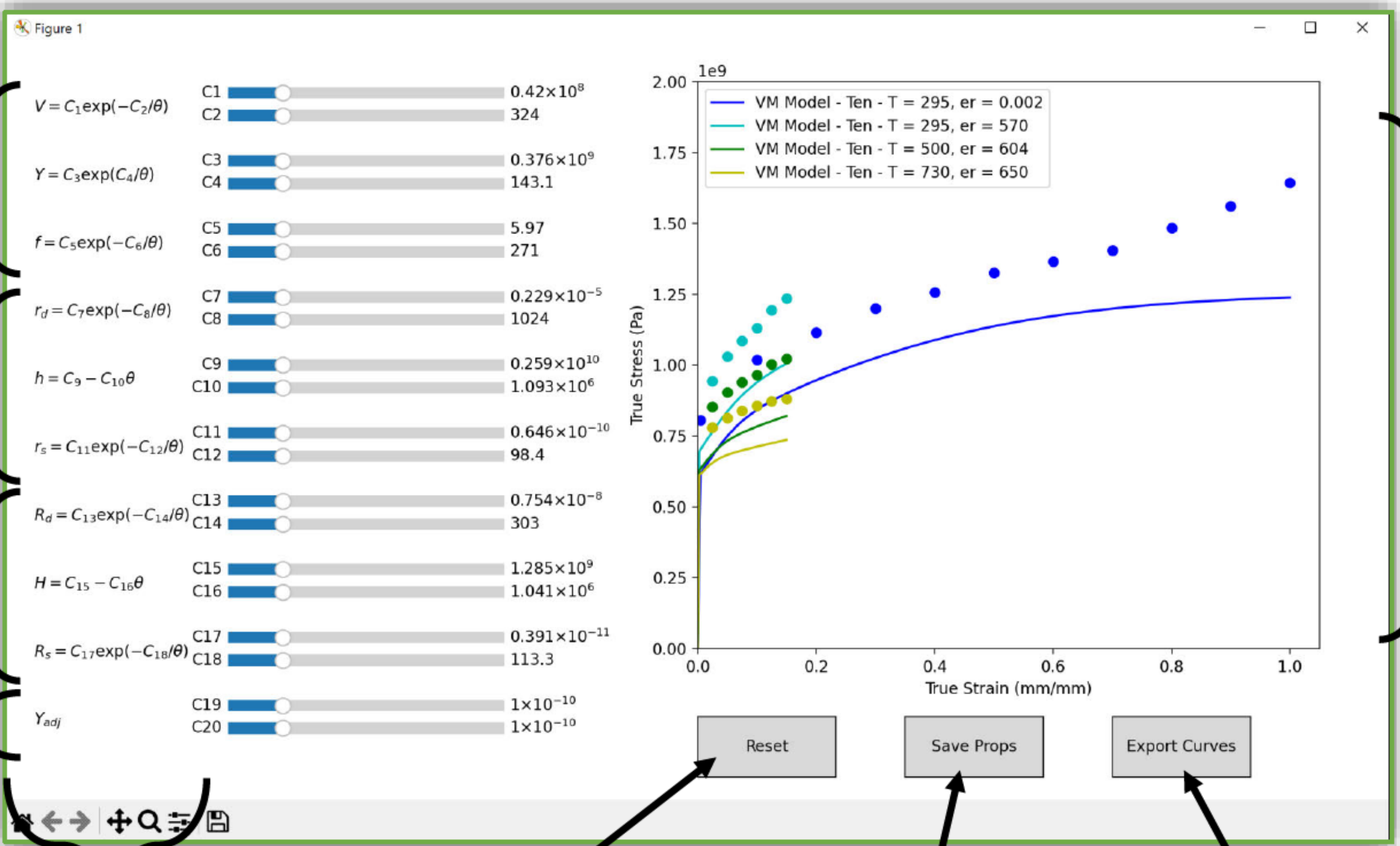
Neither GUI incorporates damage!

Calibration sliders for **initial yielding**

Calibration sliders for **kinematic hardening ($\underline{\alpha}$)**

Calibration sliders for **isotropic hardening (κ)**

Yield adjustment constants (typically not used)



Matplotlib plot

Equations relate constants with their influence on the model

Return sliders to their original values

Export constants to a new props file

Export model curves to .csv

Data file

- Each data set is contained in its own file
- Data and test conditions (T and $\dot{\varepsilon}$) are contained
- The data set's handling tag is specified here

Test Data

Test Conditions

Name to be used for plot legend and export curves

Headers specifying data locations

	A	B	C	D	E	F
1	Strain	Stress	Temperatu	Strain Rate	Name	
2	0.025	886.5974	300	0.002	T = 295, er = 0.002	
3	0.05	949.667				
4	0.075	976.8202				
5	0.1	1035.151				
6	0.125	1055.378				
7	0.15	1077.272				
8						
9						

.csv data file contents

JC Props file

- .csv file
- Constants must correspond to the first column, but they can be given in any order

First row specifies variable position

	A	B	C	D
1	Comment	Johnson & Cook 1985		
2	A	700		
3	B	510		
4	n	0.26		
5	C	0.014		
6	m	1.03		
7	Tr	295		
8	Tm	1793		
9	er0	1		
10				
11				

Reference strain rate should be 1 for EPIC/ABAQUS simulations

.csv JC props file contents

BCJ Props file

- .csv file
- Constants must correspond to the first column, but they can be given in any order

First column specifies variable position

	A	B	C	D
1	Comment	For Calibration with vumat		
2	C01	41986950		
3	C02	323.9334		
4	C03	3.76E+08		
5	C04	143.0838		
6	C05	5.966219		
7	C06	271.0246		
8	C07	2.29E-06		
9	C08	1023.628		
10	C09	2.59E+09		
11	C10	1092518		
12	C11	6.46E-11		
13	C12	98.35671		
14	C13	7.54E-09		
15	C14	302.7175		
16	C15	1.29E+09		
17	C16	1040875		
18	C17	3.91E-12		
19	C18	113.258		
20	C19	1.00E-10		
21	C20	1.00E-10		
22	Bulk Mod	1.59E+11		
23	Shear Mod	7.7E+10		
24				
25				

.csv BCJ props file contents

BCJ Python Calibration Tool: BCJ_GUI_v2.py

- Calls BCJ_**Basic**_v2.py for calculations
  - BCJ_Basic_v2.py should not require any modifications, unless adjusting the model equations
- BCJ_**GUI**_v2.py <u>will</u> require modifications in most cases, for best use

- Material model can be calibrated in MPa or Pa, but careful consistency must be used with data, calibrated parameters, and FE model implementation

BCJ_GUI_v2.py Editing

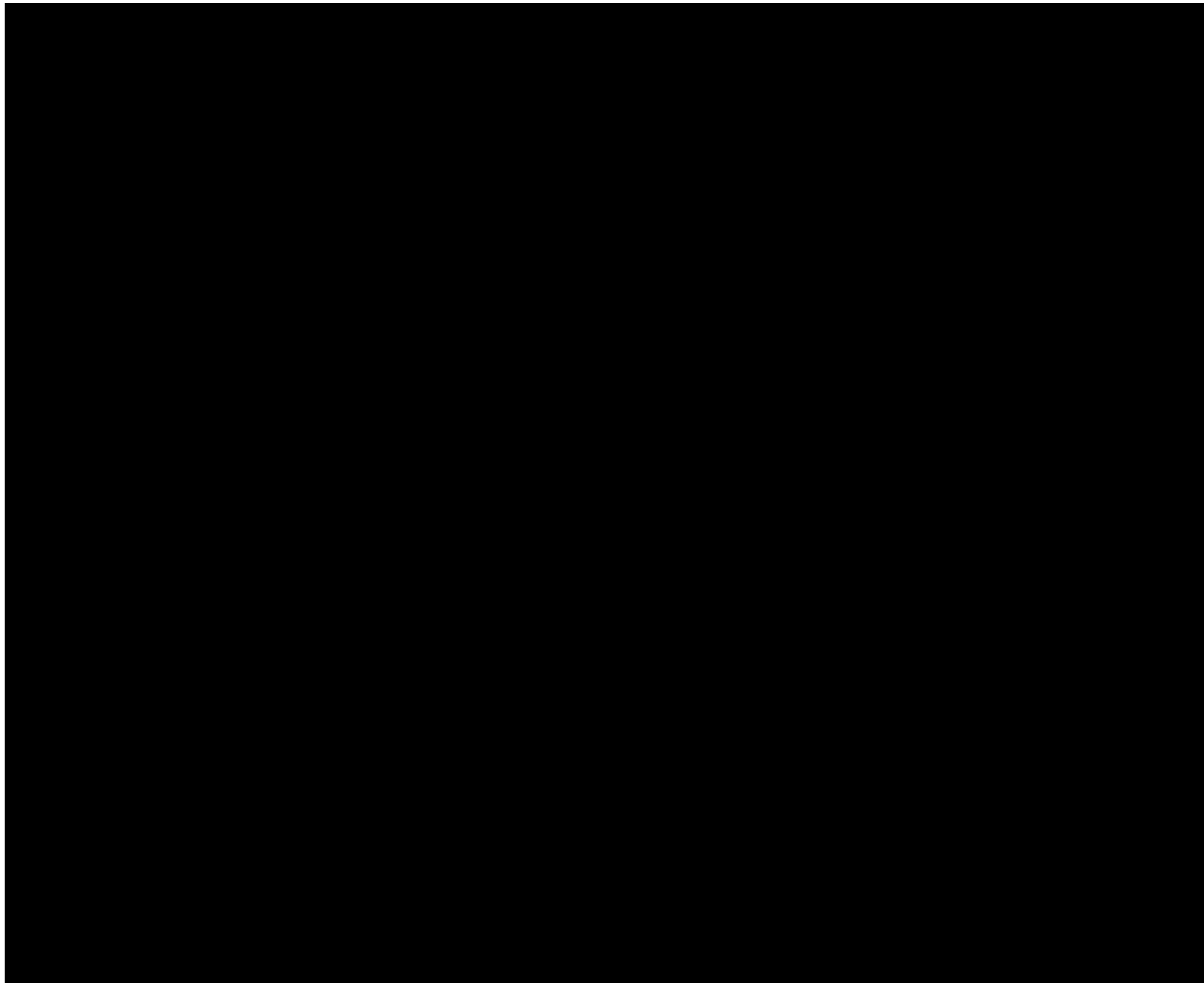
incnum = number of increments

Istate = tension/compression type for data

- *This should be modified to be included in test data info*

Ask_Files if user wants to specify their files

- If Ask_Files is true, the location of the files must be specified in the body of the code



BCJ_GUI_v2.py Editing (cont.)

Material useful for switching between several materials

Plot_ISV to plot $\underline{\alpha}$ and κ on the σ-ε plot

Scale_Mpa used to scale data files (for data values in MPa)

Max_stress used to format chart to size of data

Material useful for switching between several materials

Plot_ISV to plot $\underline{\alpha}$ and κ on the σ plot

Scale_Mpa used to scale data file (for data values in MPa)

Max_stress used to format chart size of data

BCJ_GUI_v2.py Editing (cont.)

Example of where to put file path locations in code

If user is prompted to select the files from a folder, the filename and path will be printed in terminal. This can be pasted into the code for easier repeated use.