Gurpal Singh
May 4, 2017
ME 510
Final Project

Abstract:

In this study, a one-dimensional uniaxial tension model for modeling the stress response of Polymethylmethacrylate is explored. The model is called the DSGZ model and is based off four previous constitutive models. It is a model that is strain, strain-rate, and temperature dependent. It is also able to capture the strain hardening and strain softening behavior of glassy polymers. To supplement the project a 3-dimensional constitutive model is also briefly explored. The DSGZ model is then compared against experimental data. The material coefficients are found using Matlab and a sensitivity study is also done. The results show that the DSGZ model can predict the stress response of PMMA at different temperatures as well as strain rates. However, it is only a one-dimensional model for glassy polymers and can only be used to model tension.

Stage 1

Material Name: Plexiglas also known as Polymethylmethacrylate (PMMA)

Primary/Relevant Applications: Used for optical Lenses, lenses of exterior lights of automobiles, viewing windows in submarines and aircrafts, lenses in cataract surgery, transplants and prosthetics, and basketball backboards

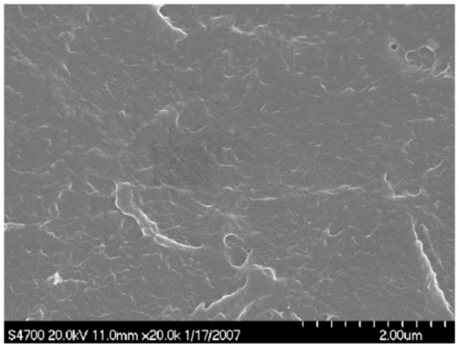
Reason for Interest: As kids, my younger brother and I broke a lot of basketball backboards so it would be interesting to see what was going on in the material and what caused it to break.

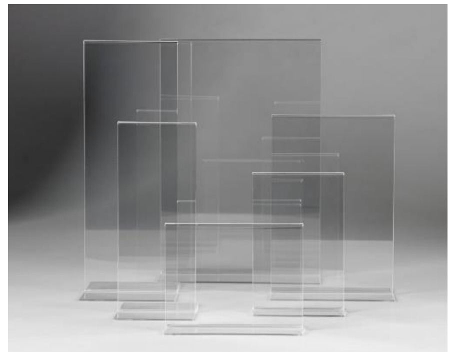
Structure:

Classification: Poylmer

Composition: $(C_5O_2H_8)_n$

Microstructure and macrostructure of pure PMMA:





Organization(symmetry): PMMA has a uniform (isotropic) cellular structure

Function:

Qualitative characteristics: Time dependent, nonlinear, temperature dependent

Quantitative characteristics:

Poisson's ratio: ranges from 0.35 - 0.4

Young's Modulus: 2.4 - 3.4 GPa

Summary of Two Manuscripts of experimental data:

In the first paper, "Experimental study of tensile properties of PMMA at intermediate strain rate" tensile tests of PMMA were conducted with quasi-static loading [2]. Arruda and his peers used an Instron machine and a split Hopkinson pressure bar to measure the compressive and tensile properties of PMMA. The tests were conducted at intermediate strain rates of 2.92×10^{-1}, 6.54×10^{-1}, 2.81, and 18.6 s^{-1} on a self-developed testing apparatus and at strain rates of 2.31×10^{-5}, 2.38×10^{-4}, 2.01×10^{-3}, and 2.00×10^{-2} s^{-1} on MTS810 under quasistatic loading. The deformation was measured by a self-designed optical extensometer. The results for the different strain rates are shown below. It should be noted that at certain strain rates the material behaves in an elastic manner.

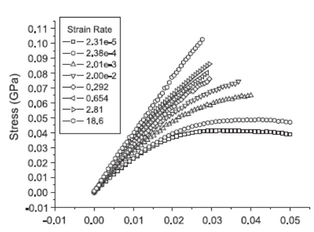


Fig. 3. Stress–strain curves of PMMA at different strain rates.

In the second paper, “Experimental investigation on yield behavior of PMMA under combined shear-compression loading,” a compression shear test was used to explore the yield behavior of PMMA [3]. The samples of Plexiglas G PMMA were tested at angles of 0°, 15°, 30°, 45°, and 60° and at 3 different loading rates of 5 x 10^{-3} mm s^{-1}, 5 x 10^{-2} mm s^{-1}, and 5 x 10^{-1} mm s^{-1}. It was found that the yield stress has a positive correlation with strain rate. At low strain, there may be heat transfer occurring at a low strain rate which produces a softening effect. There is also data of shear strain vs shear strain at different angles and strain rates but only two plots are shown below.

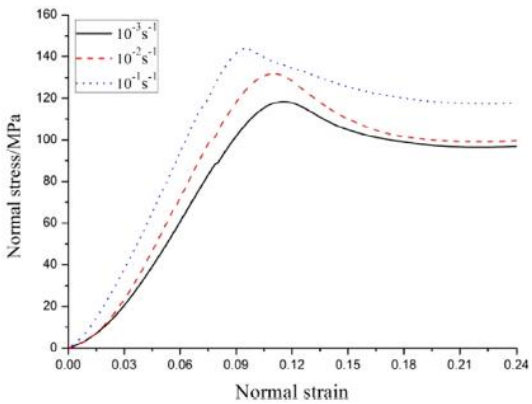


Fig. 2. Normal stress–strain curves at different strain rates.

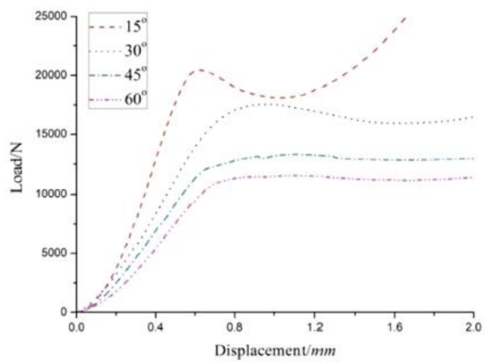


Fig. 3. Load–displacement curves at $\dot{\varepsilon} = 10^{-1}$ s^{-1}.

Stage 2

<u>*A quick note:*</u>

The model that I will be using for Plexiglass (PMMA) is a uniaxial 1-Dimensional model called the DSGZ model for modeling the compressive and tensile behavior in PMMA. To fulfill the requirements of the project a simpler 3-dimensional model called the Gent model will be used to show coordinate independence and why it cannot be used to model PMMA.

<u>Equation:</u>

Name: Gent Model

Important Features: 3-dimensional, hyperelastic model

Formulation:

Symbolic Notation:

$$\underset{\approx}{\sigma} = \frac{-\mu J_M}{I_1 - 3 - J_M} \underset{\approx}{B} - p \underset{\approx}{I}$$

Index Notation:

$$\sigma_{ij} = \frac{-\mu J_M}{I_1 - 3 - J_M} B_{ij} - p\delta_{ij}$$

<u>Objectivity:</u>

Proving coordinate system independence:

To show that this is a proper function it must coordinate transform like a second order tensor. I_1 is the first invariant of ***B*** so it must be coordinate system independent from the definition of an invariant.

In system A:

$$\sigma_{ij}^A = \frac{-\mu J_M}{I_1 - 3 - J_M} B_{ij}^A - p\delta_{ij}^A$$

Where,

$$B_{ij}^A = B_{pq}^B Q_{pi} Q_{qj} \qquad \delta_{ij}^A = \delta_{pq}^B Q_{pi} Q_{qj}$$

Making the substitution:

$$\sigma_{ij}^{A} = \frac{-\mu J_M}{I_1 - 3 - J_M} B_{pq}^{B} Q_{pi} Q_{qj} - p\delta_{pq}^{B} Q_{pi} Q_{qj}$$

Factoring out Q_{pi} and Q_{qj}:

$$\sigma_{ij}^{A} = Q_{pi} \left[\frac{-\mu J_M}{I_1 - 3 - J_M} B_{pq}^{B} - p\delta_{pq}^{B} \right] Q_{qj}$$

Thus this is a proper function since it coordinate transforms like a second order tensor:

$$\underset{\approx}{T}^{A} = Q^{T} \underset{\approx}{T}^{B} Q$$

Coefficients:

The Gent model involves two coefficients:

μ: Shear modulus of the material

J_M: A dimensionless parameter with the condition $J_m = I_M - 3$ where I_M is a limiting value of the first invariant.

Why the Gent Model cannot be used to model PMMA:

The Gent model is a useful model to predict the mechanical behavior of rubbers and some polymeric materials. However, it would be a very poor choice to use the Gent constitutive model to try to model the behavior of PMMA. This is due to several reasons. The Gent model does not consider time or temperature dependencies of a material. PMMA has both a time and temperature dependency which makes it a more complicated material to model. Furthermore, PMMA displays traits of viscoplasticity, and viscoelasticity as well. The behavior of PMMA tensile and compression tests show that PMMA experiences both strain softening and strain hardening. The Gent model would not be able to capture the yield peak and the strain softening behavior. Thus, it would be a poor choice to use the Gent model to model PMMA and instead, the 1-dimensional uniaxial DSGZ model is used.

DSGZ Model:

Equation:

Name: DSGZ Model

Important Features: Time and temperature dependent. 1-dimensional, non-linear

Formulation:
(where, σ and ε are the principal components i.e. 11, 22, 33)

$$\sigma(\epsilon,\dot{\epsilon},T) = K\{f(\epsilon) + \left[\frac{\epsilon * e^{\left(1-\frac{\epsilon}{C_3 * h(\dot{\epsilon},T))}\right)}}{C_3 * h(\dot{\epsilon},T))} - f(\epsilon)\right] * e^{[ln(g(\dot{\epsilon},T))-C_4]*\epsilon}\} * h(\dot{\epsilon},T))$$

Where,

$$f(\epsilon) = (e^{-C_1*\epsilon} + \epsilon^{C_2})(1 - e^{-\alpha*\epsilon})$$

$$h(\dot{\epsilon},T) = (\dot{\epsilon})^m * e^{\frac{a}{T}} \qquad\qquad g(\dot{\epsilon},T) = e^{\frac{a}{T}}$$

Objectivity:

Since this is a 1-dimensional model, objectivity was demonstrated using the Gent constitutive model.

Coefficients:

The effects of the coefficients were explored by achieving a close curve fit and then changing one variable at a time.

The DSGZ model has 8 material coefficients as follows:

K (Pa s^m):
K determines the amplitude of the curve and the yield peak. In other words, it is a scaling factor of the entire curve.

C_1

C1 effects the curvature after the yield peak. A negative value gives a more of a concave curve with a curve similar to that of an exponential function. For positive values, around 1 – 20 it appears to take on the shape of that of polymeric materials.

C_2

C2 has an effect on the amplitude of the yield peak. As the values become negative the yield peak rises dramatically. For higher positive values of C2 the curvature of the strain hardening part of the curve becomes more concave.

C_3 (s^m)

C3 has an effect on the initial rise from zero to the yield peak. From playing around with values in matlab, it seems like C3 must take on a small positive value around 0.005 to fit the behavior of PMMA. A large value for C3 will ignore the yield peak. Really small negative values do not work well as they produce a disoriented curve. Larger negative values show the same effect as large positive values.

C_4

C4 effects the entirety of the curve. Smaller values than 10 and negative values will not work since they produce a curve that flat and then falls downwards. Values around 10 – 15 are fairly close to the shape expected for PMMA. Larger values than 20 begin to flatten out the yield peak.

a (K)

a effects the overall height of the curve. Values lower than 1000 will not work for PMMA since the curve is way below all the data points. Around a value of 1200 for a, a decent fit is achieved.

m

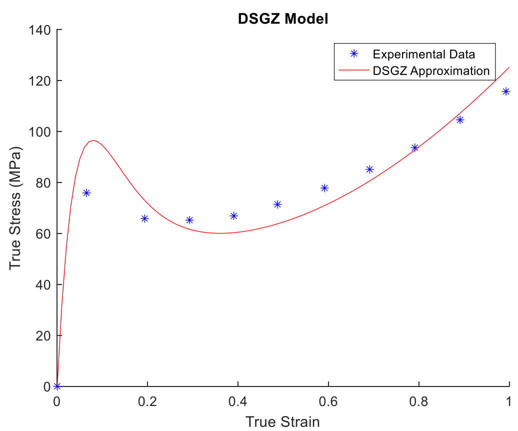
As the value for m is increased, the curve falls dramatically below the data points. As m is decreased, the curve keeps its shape but is lifted above the data points. With negative values, the curve does not follow the general pattern of the data at all.

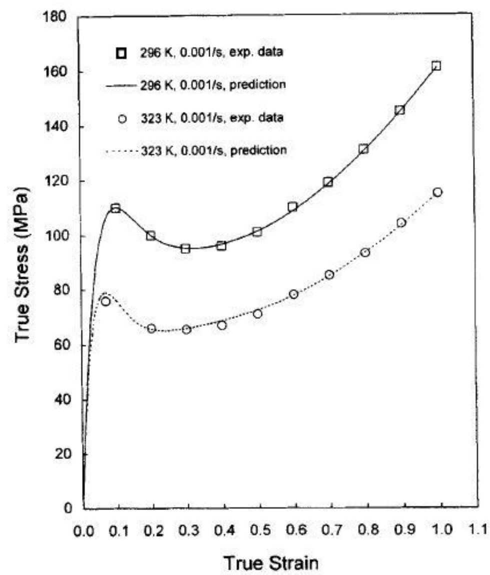
α

As α is increased the yield peak is raised but the data points after the yield peak still reasonably fit to this curve. As the value of alpha is decreased, the curve after the yield point begins to take on a declining shape.

Modeling:

To demonstrate that the DSGZ model can model tension and compression in PMMA, I coded up the model in Matlab and showed that the curve can follow the experimental data. Furthermore, a plot from the paper that used this model on experimental data is shown below [1]. The coefficients for Matlab are not yet optimized as they were only a rough approximation obtained from the curve fitting application in Matlab.





Source Summaries:

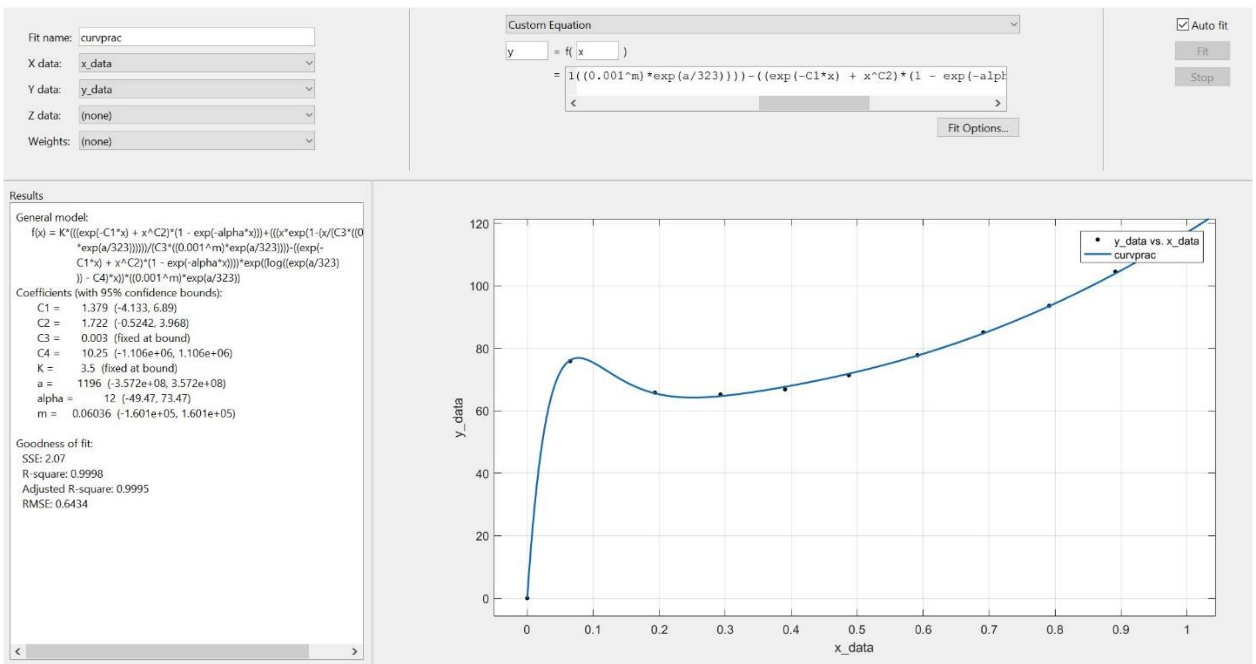
The first source that I found was a paper called "A Uniform Phenomenological Constitutive Model for Glassy and Semicrystalline Polymers," which proposed a 1-dimensional constitutive model for glassy polymers [1]. The equations of the model are listed above in the equations section. The proposed model successfully modeled glassy polymers in a uniaxial compression tests. The paper also provides experimental data for PMMA tested at different temperatures as well as different strain rates which is another plus. The results showed that the DSGZ model successfully modeled PMMA, polyamide, and polycarbonate.

The second source extends the DSGZ into a 3-dimensional model and demonstrates that it produces comparable results to abaqus [4] for polypropylene at different strain rates and temperatures. They compare the abaqus and DSGZ results to actual experimental data and validate that it is a working model for polymers. Their test The paper was fairly complex as the 3-dimensional model dove into the concepts of viscoplactiticy. The results validated the DSGZ model and shows that it can capture the behavior of glassy polymers.

Stage 3:

To find the 8 material coefficients, the curve fitting application in Matlab was used. By playing around with different coefficient values I could narrow down the bounds. These bounds were then inputted into the “Fit Options” for the curve fitting tool and a curve was created to fit the experimental data for PMMA. The material coefficients were found by fitting the data for a tensile test of PMMA at a temperature of 323 K and a strain rate of 0.001/s. The fit showed an R^2 value of 0.9998. A better fit could be obtained by playing with the tolerance levels and the coefficient bounds but it would be cumbersome process with the curve fitting too. To acquire an even better fit, an optimization technique is recommended.

Curve Fit to find Material Coefficients:

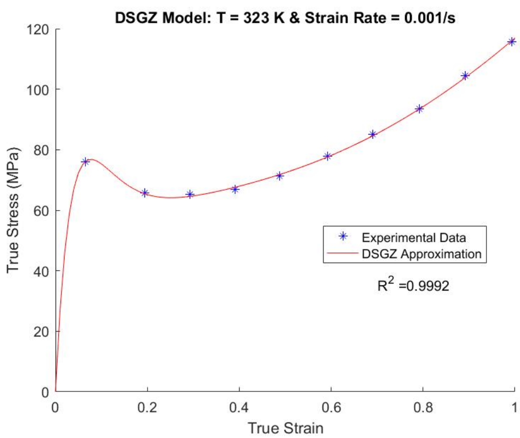


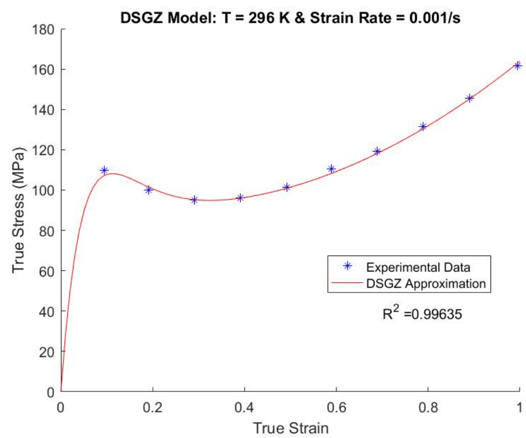
The optimized material coefficients from the fit were found to be:

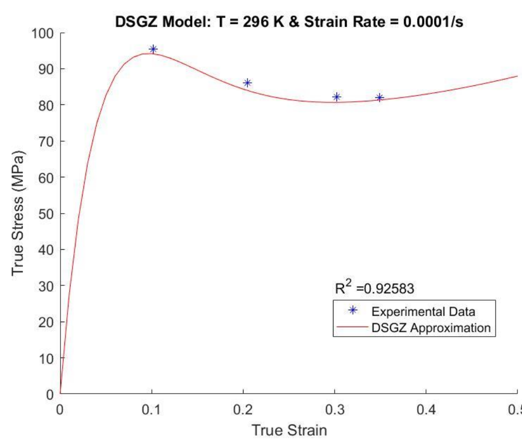
$K = 3.5$ (Pa s^m)	$C_1 = 1.379$	$C_2 = 1.722$	$C_3 = 0.003$ (s^m)
$C_4 = 10.25$	$a = 1196$ (K)	$\alpha = 12$	$m = 0.06036$

Graphical Comparison of experimental data vs. model data:

Since the model is temperature, strain, and strain rate dependent, different temperatures and strain rates were used to compare the experimental data to the model prediction. The model was compared with experimental results for annealed PMMA at T = 323 K & strain rate = 0.001/s, T = 296 K & strain rate = 0.001/s, and finally T = 296 K & strain rate = 0.0001/s.



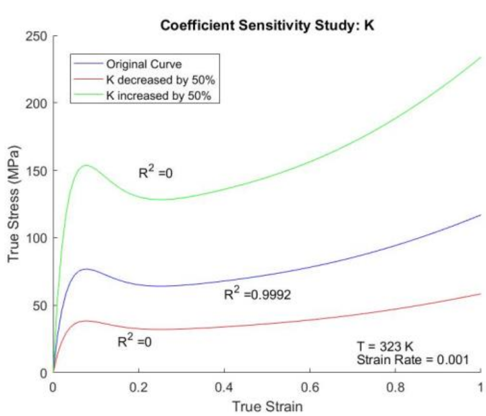


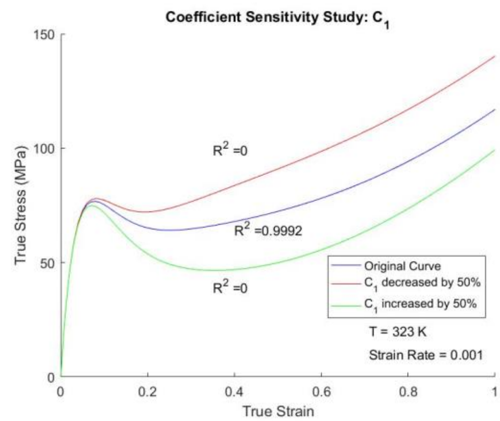


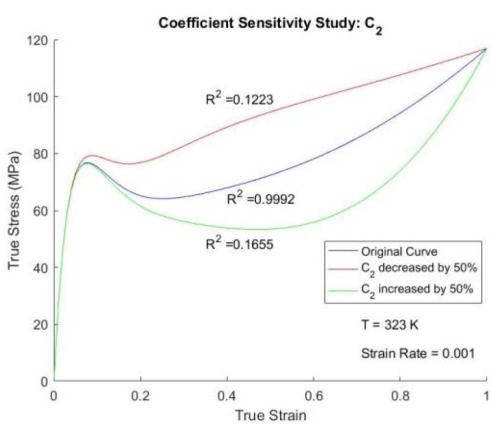
The plots show that the model accurately can model uniaxial tension for PMMA. The statistical measure of the goodness of a fit R^2 was calculated for each fit and it should be noted that for the first two fits the R^2 values are above 0.99. The third plot R^2 value is slightly lower since there were not enough experimental data points for that particular experiment but, as you can see the curve follows the general path of the data points. Furthermore, with the temperature drop, the yield peak drops. This is the expected result since in colder temperatures PMMA becomes a much more brittle material. PMMA also exhibits slight strain softening and then strain hardening which the DSGZ model could predict. Thus, the DSGZ is an appropriate model to use for PMMA because it accounts for the nonlinearities, temperature dependence, strain and strain-rate dependence.

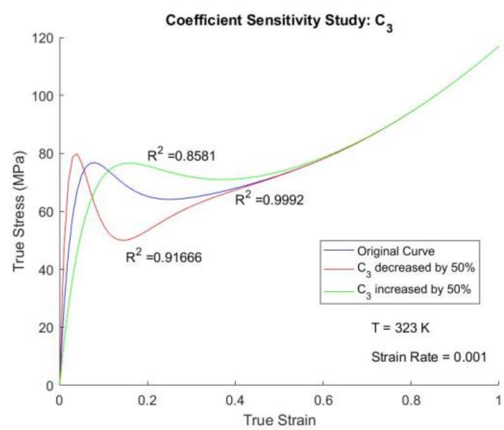
Material Coefficient Sensitivity Study:

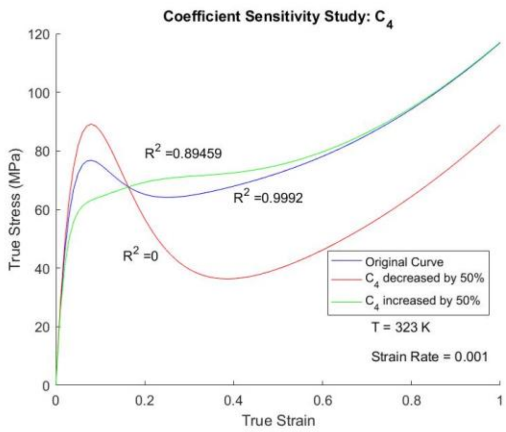
The sensitivity of the material coefficients was explored by increasing and decreasing each coefficient one at a time by 50%. Plots were then created to show the effect of the change and the sensitivity of the coefficient. The model was evaluated for a temperature of 323 K and a strain-rate of 0.001/s. R^2 values were also computed to show how the change in the coefficient effects the goodness of the fit. The blue curve represents the original curve with the optimized coefficients. The red curve is representative of a material coefficient decreased by 50% whereas the green curve is representative of a material coefficient increased by 50%.

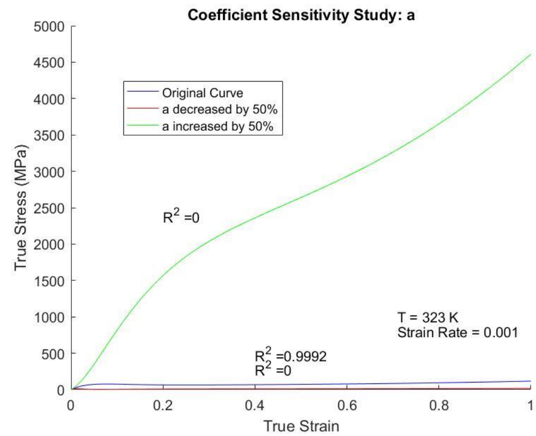


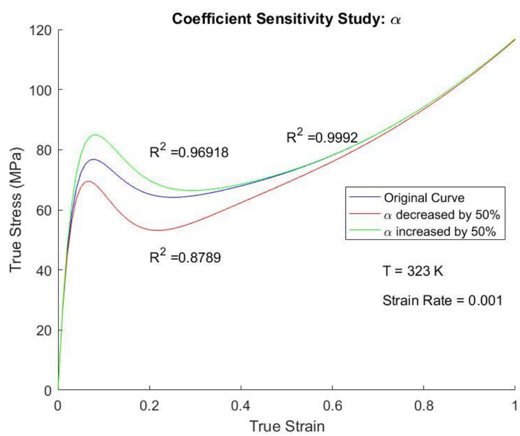


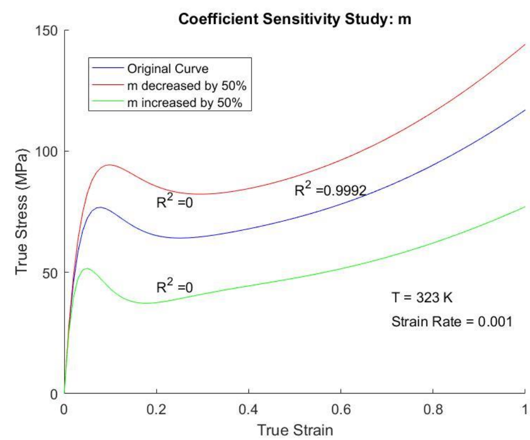






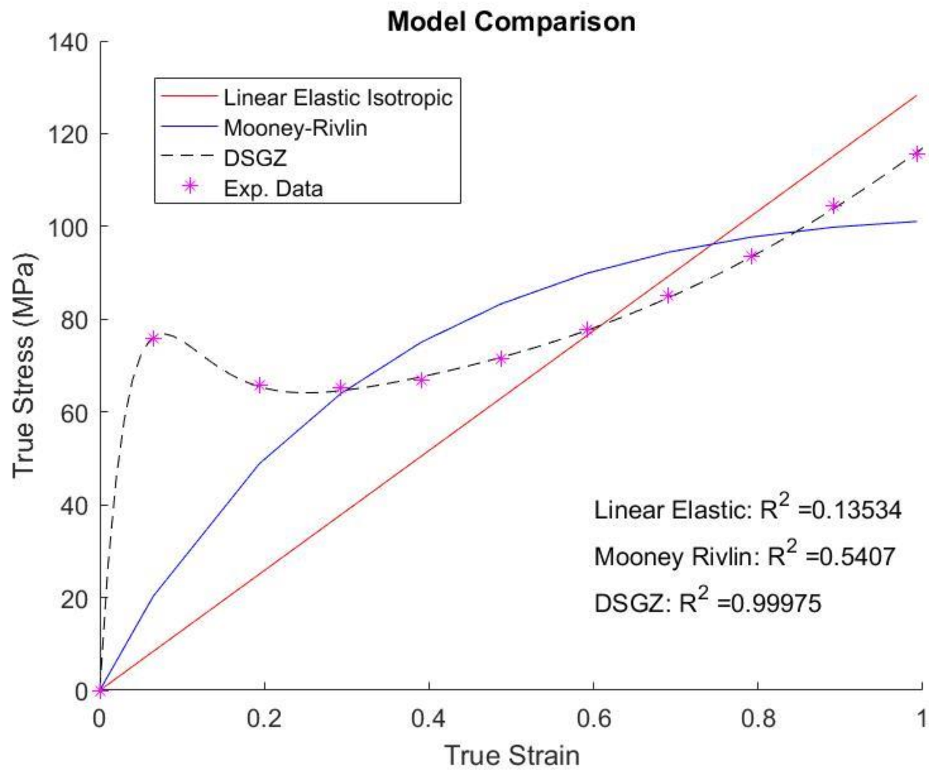






Comparison to Other Models:

The DSGZ model was compared to the linear elastic isotropic model, the hyperelastic Mooney-Rivlin model, as well as the viscoelastic Maxwell mode. The plots below show the comparison of the different models. The R^2 values were also computed to show the goodness of the fit that each model provides. The comparison was done for a temperature of 323 K and a strain rate of 0.001/s.



Linear Elastic Isotropic Model:

For this model stress and strain are linearly related. There is only one coefficient which determines the slope of the line. Matlab's curve fitting tool was used to optimize the coefficient for the experimental data. Since the data does not follow a linear pattern it should be obvious that a linear elastic isotropic model is a poor choice. Furthermore, this model does not account for temperature dependence or strain-rate dependence.

Optimized Material Coefficients: ***C = 129.2***

Hyperelastic Mooney-Rivlin Model:

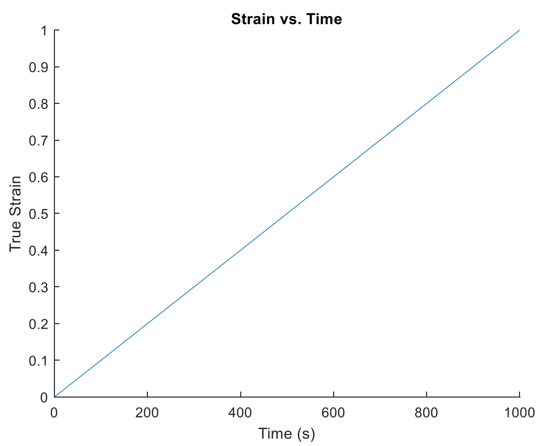
Although this model had a higher R^2 value than the linear elastic fit, it still is a poor fit to the data. The Mooney Rivlin model can account for some nonlinearity but as you can see from the data, PMMA exhibits extreme nonlinearity. Thus, the Mooney Rivlin material also does not work for PMMA. The Mooney Rivlin has two material coefficient which were also optimized using the curve fitting tool on Matlab.

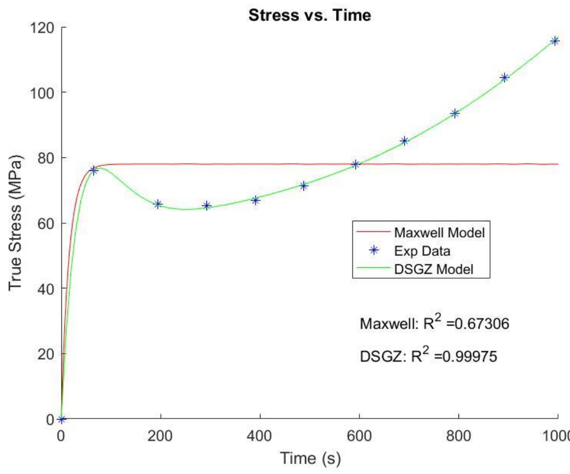
Optimized Material Coefficients: $C_1 = -31.14$ $C_2 = 90.09$

Maxwell Discrete Element Model:

The Maxwell model was also fit to the experimental data. Although this model accounts for the time dependency of the material, it does not take into account the plastic contributions of the material. PMMA exhibits strain softening and strain hardening, which this model is not able to predict. The material coefficients for the Maxwell model were found through trial and error. The strain-rate was a constant value of 0.001/s. The timespan ranged from 0 to 1000 seconds and the temperature was 323 K. The strain input plot is also shown below for reference.

Optimized Material Coefficients: ***k = 5000*** ***c = 78000***





Predictive Experiment:

A predictive experiment that could be done to validate the constitutive equation and the coefficients would be a stress relaxation test. For this test, the material is strained with a constant strain-rate for a time period. Then the material is kept at constant strain which means the strain-rate will be zero after this time and the material experience stress relaxation. The force required to keep the material displaced at the constant strain value will decrease overtime. I found data for a stress relaxation test performed with PMMA. My next step would be to validate the DSGZ model by inputting the strain and strain-rate data to the model. If the model and coefficients produce the same results as the stress results from the stress relaxation test, then the DSGZ model would be a valid model.

Stage 4:

See Power-point presentation.

Stage 5:

Reflection:

I thought this project was a great learning experience and implementation of the concepts taught in class. This project was a great ending to the course because it allowed me to put together all of the concepts from the semester into a single project. Not only did it cover the concepts in class, but it emphasized the skills of time management and professional presentation both of which are needed later in life. I had a wonderful experience with this project from the beginning. It made me do the research and compile the results into something presentable. It was a little time consuming but it is totally doable if one starts ahead of time.

The project helped me see the real-world application of continuum mechanics that I would have not otherwise realized. It was extremely beneficial to spend some time on a single constitutive model and fully understand how it is derived and implemented. I don’t really see a way the project could be improved except that maybe do the presentations over a course of two days. It’s easy to lose focus when the session is that long and the presenters may not get the full attention of the audience. Other than that, I thought the project is great ending to the course and definitely should be kept as part of this course.

Appendix I: References

[1] Duan, Y., Saigal, A., Greif, R. and Zimmerman, M. A. (2001), A uniform phenomenological constitutive model for glassy and semicrystalline polymers. *Polym Eng Sci*, 41: 1322–1328. doi:10.1002/pen.10832

[2] Hengyi Wu, Gang Ma, Yuanming Xia, Experimental study of tensile properties of PMMA at intermediate strain rate, *Materials Letters*, Volume 58, Issue 29, November 2004, Pages 3681-3685, ISSN 0167-577X, http://doi.org/10.1016/j.matlet.2004.07.022. (http://www.sciencedirect.com/science/article/pii/S0167577X04005270)

[3] Jianjun Zhang, Tao Jin, Zhihua Wang, Longmao Zhao, Experimental investigation on yield behavior of PMMA under combined shear–compression loading, *Results in Physics*, Volume 6, 2016, Pages 265-269, ISSN 2211-3797, http://doi.org/10.1016/j.rinp.2016.05.004.

[4] Nadia Achour, George Chatzigeorgiou, Fodil Meraghni, Yves Chemisky, Joseph Fitoussi, Implicit implementation and consistent tangent modulus of a viscoplastic model for polymers, *International Journal of Mechanical Sciences*, Volume 103, November 2015, Pages 297-305, ISSN 0020-7403, http://doi.org/10.1016/j.ijmecsci.2015.09.010.

Appendix 2: Matlab Code

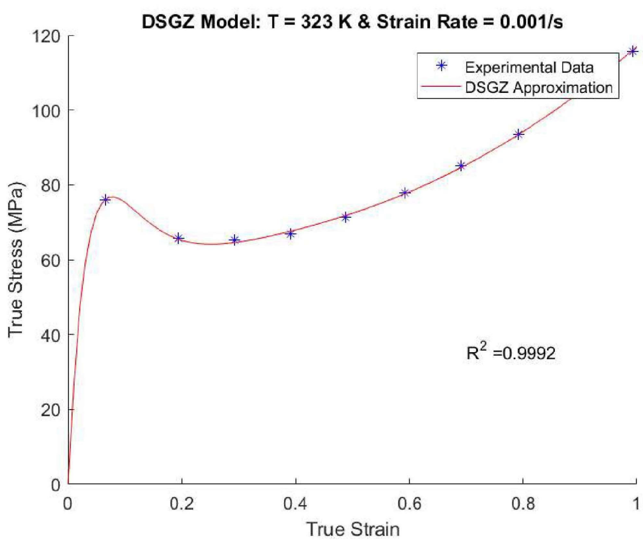
```matlab
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%Comparison of The DSGZ Model and Experimental Results
%
%Author: Gurpal Singh
%
%Date: 4/29/2017
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%Experimental Data for Duan Figure 1
%Temperature: 323 K
%Strain Rate: 0.001/s
Data_1= [
                 0.0650   75.8767
                 0.1936   65.8009
                 0.2928   65.1989
                 0.3907   66.8687
                 0.4874   71.3780
                 0.5914   77.7775
                 0.6912   85.1249
                 0.7911   93.6079
                 0.8912  104.5513
                 0.9928  115.6836];

 %True Strain Rate: 0.001/s
 strainrate = 0.001;

 %Temperature: 323K
 T = 323;

%Material Coefficients
disp('The Material Coefficients are: ')
C1 = 1.379
C2 = 1.722
C3 = 0.003
C4 =10.25
K = 3.5
a = 1196
alpha = 12
m = 0.06036


%DSGZ Model Calculation
 se = 0:.01:1;
 se = se';
 for i = 1:length(se)
    strain = se(i,1);
    y(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit

strain_data = zeros(length(Data_1(:,2)),1);
```

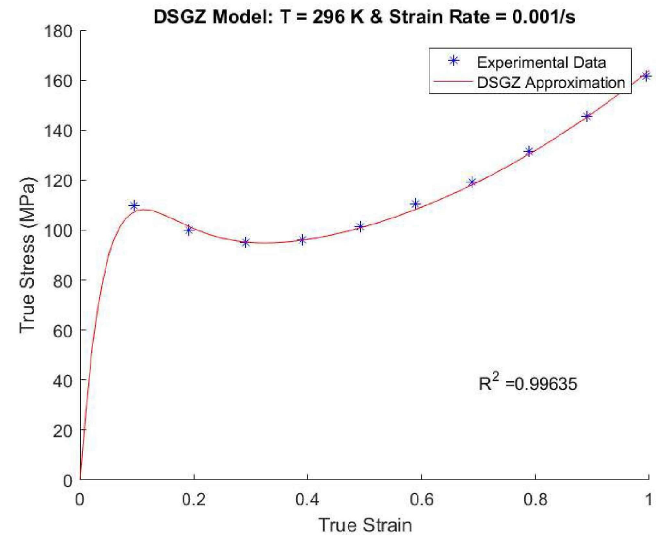
```matlab
%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2 rmse] = rsquare(exp_data,f);

%Plotting the figure
figure
xlabel('True Strain')
ylabel('True Stress (MPa)')
title('DSGZ Model: T = 323 K & Strain Rate = 0.001/s')
hold on
plot(Data_1(:,1),Data_1(:,2),'b*')
hold on
plot(se,y,'r')
hold on
s1 = 'R^2 = ';
s2 = num2str(r2);
s = strcat(s1,s2);
text(0.7,36, s ,'FontSize',10)
legend('Experimental Data', 'DSGZ Approximation')


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Experimental Data for Duan Figure 1
%Temperature: 296 K
%Strain Rate: 0.001/s
Data_2 = [0.0950  109.7465
          0.1920   99.8708
          0.2909   94.9157
          0.3902   96.2064
          0.4927  101.2816
          0.5897  110.3333
          0.6897  119.1948
          0.7884  131.4637
          0.8902  145.6243
          0.9950  161.6765];

%Temperature & Strainrate
T = 296;
strainrate = 0.001;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
```

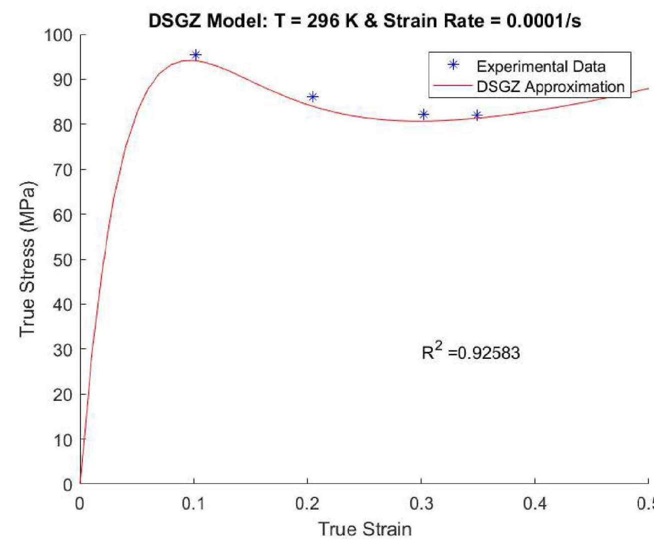
```
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_2(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_2(:,2)),1);
strain_data(:,1) = Data_2(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_2(:,2);

%Function for R^2
[r2 rmse] = rsquare(exp_data,f);

%Plotting
figure
xlabel('True Strain')
ylabel('True Stress (MPa)')
title('DSGZ Model: T = 296 K & Strain Rate = 0.001/s')
hold on
plot(Data_2(:,1),Data_2(:,2),'b*')
hold on
plot(se,y,'r')
hold on
s1 = 'R^2 = ';
s2 = num2str(r2);
s = strcat(s1,s2);
text(0.7,40, s ,'FontSize',10)
legend('Experimental Data', 'DSGZ Approximation')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Experimental Data for Duan Figure 2
%Temperature: 296 K
%Strain Rate: 0.0001/s
Data_5 = [
          0.1018   95.3330
          0.2044   86.1193
          0.3024   82.1904
          0.3489   81.9892];

%Temperature & Strainrate
T = 296;
strainrate = 0.0001;

%DSGZ Model Calculation
se = 0:.01:.5;
se = se';
y = zeros(length(se),1);
for i = 1:length(se)
    strain = se(i,1);
    y(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_5(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_5(:,2)),1);
strain_data(:,1) = Data_5(:,1);

%Calculation at Experimental Points
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_5(:,2);

%Function for R^2
[r2 rmse] = rsquare(exp_data,f);

%Plotting
figure
xlabel('True Strain')
ylabel('True Stress (MPa)')
title('DSGZ Model: T = 296 K & Strain Rate = 0.0001/s' )
hold on
plot(Data_5(:,1),Data_5(:,2),'b*')
hold on
plot(se,y,'r')
hold on
s1 = 'R^2 = ';
s2 = num2str(r2);
s = strcat(s1,s2);
text(0.3,30, s ,'FontSize',10)
legend('Experimental Data', 'DSGZ Approximation')
```

```
The Material Coefficients are:

C1 =

    1.3790


C2 =

    1.7220


C3 =

    0.0030
```

```
C4 =

   10.2500

K =

    3.5000

a =

        1196

alpha =

    12

m =

    0.0604
```







```matlab
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%Sensitivity Study of the Material Coefficients for the DSGZ Model
%
%Author: Gurpal Singh
%
%Date: 4/29/2017
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Optimized Coeffs were found from curve fit:
C1 = 1.379;
C2 = 1.722;
C3 = 0.003;
C4 =10.25;
K = 3.5;
a = 1196;
alpha = 12;
m = 0.06036;

%Study will be performed by changing each coefficient one at
%a time for the data that was initially fit.

%Experimental Data for Duan Figure 1
%Temperature: 323 K
%Strain Rate: 0.001/s

strainrate = 0.001;
T = 323;

Data_1= [0.0650   75.8767
         0.1936   65.8009
         0.2928   65.1989
         0.3907   66.8687
         0.4874   71.3780
         0.5914   77.7775
         0.6912   85.1249
         0.7911   93.6079
         0.8912  104.5513
         0.9928  115.6836];

 %DSGZ Model Calculation
 se = 0:.01:1;
 se = se';
 for i = 1:length(se)
    strain = se(i,1);
    y(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y rmse] = rsquare(exp_data,f);

%%%%%%%%%------C1 Decreased by 50%-------%%%%%%%%%
C1 = C1/2;

%DSGZ Model Calculation
se = 0:.01:1;
se = se';
for i = 1:length(se)
    strain = se(i,1);
    y1(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y1 rmse] = rsquare(exp_data,f);

%%%%%%%%%------C1 Increased by 50%-------%%%%%%%%%

% Increase by 50 (extra factor of 2 because of decrease earlier%
C1 = C1*2*2;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y2(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);
```

```matlab
%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y2 rmse] = rsquare(exp_data,f);

%Plotting
figure
xlabel('True Strain')
ylabel('True Stress (MPa)')
title('Coefficient Sensitivity Study: C_{1}')
hold on
% plot(Data_1(:,1),Data_1(:,2),'b*')
hold on
plot(se,y,'b')
plot(se,y1,'r')
plot(se,y2,'g')
hold on
txt1 = ' T = 323 K';
text(0.7,20,txt1)
txt2 = ' Strain Rate = 0.001';
text(0.7,10,txt2)
s1 = 'R^2 =  ';

s2 = num2str(r2_y);
s = strcat(s1,s2);
text(0.4,65, s ,'FontSize',10)

s2 = num2str(r2_y1);
s = strcat(s1,s2);
text(0.35,100, s ,'FontSize',10)

s2 = num2str(r2_y2);
s = strcat(s1,s2);
text(0.35,40, s ,'FontSize',10)
legend('Original Curve', 'C_{1} decreased by 50%', 'C_{1} increased by 50%')


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Coefficient: C2

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Coeffs were found from curve fit:
C1 = 1.379;
C2 = 1.722;
C3 = 0.003;
C4 =10.25;
K = 3.5;
a = 1196;
alpha = 12;
m = 0.06036;

%DSGZ Model Calculation
for i = 1:length(se)
    strain = se(i,1);
    y(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y rmse] = rsquare(exp_data,f);

%%%%%%%%%------C2 Decreased by 50%-------%%%%%%%%%
C2 = C2/2;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y1(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
```

```
[r2_y1 rmse] = rsquare(exp_data,f);

%%%%%%%%%------C2 Increased by 50%-------%%%%%%%%%
% Increase by 50 (extra factor of 2 because of decrease earlier%
C2 = C2*2*2;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y2(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y2 rmse] = rsquare(exp_data,f);

%Plotting
figure
xlabel('True Strain')
ylabel('True Stress (MPa)')
title('Coefficient Sensitivity Study: C_{2}')
hold on
plot(se,y,'b')
plot(se,y1,'r')
plot(se,y2,'g')
hold on
txt1 = ' T = 323 K';
text(0.7,20,txt1)
txt2 = ' Strain Rate = 0.001';
text(0.7,10,txt2)
s1 = 'R^2 = ';

s2 = num2str(r2_y);
s = strcat(s1,s2);
text(0.4,65, s ,'FontSize',10)

s2 = num2str(r2_y1);
s = strcat(s1,s2);
text(0.35,100, s ,'FontSize',10)

s2 = num2str(r2_y2);
s = strcat(s1,s2);
text(0.35,49, s ,'FontSize',10)
legend('Original Curve', 'C_{2} decreased by 50%', 'C_{2} increased by 50%')


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Coefficient: C3

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Coeffs were found from curve fit:
C1 = 1.379;
C2 = 1.722;
C3 = 0.003;
C4 =10.25;
K = 3.5;
a = 1196;
alpha = 12;
m = 0.06036;

%DSGZ Model Calculation
for i = 1:length(se)
    strain = se(i,1);
    y(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y rmse] = rsquare(exp_data,f);

%%%%%%%%%------C3 Decreased by 50%-------%%%%%%%%%
C3 = C3/2;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y1(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
```

```
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y1 rmse] = rsquare(exp_data,f);

%%%%%%%%%------C3 Increased by 50%-------%%%%%%%%%
% Increase by 50 (extra factor of 2 because of decrease earlier%
C3 = C3*2*2;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y2(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y2 rmse] = rsquare(exp_data,f);

%Plotting
figure
xlabel('True Strain')
ylabel('True Stress (MPa)')
title('Coefficient Sensitivity Study: C_{3}')
hold on
% plot(Data_1(:,1),Data_1(:,2),'b*')
hold on
plot(se,y,'b')
plot(se,y1,'r')
plot(se,y2,'g')
hold on
txt1 = ' T = 323 K';
text(0.7,20,txt1)
txt2 = ' Strain Rate = 0.001';
text(0.7,10,txt2)
s1 = 'R^2 = ';

s2 = num2str(r2_y);
s = strcat(s1,s2);
text(0.4,65, s ,'FontSize',10)

s2 = num2str(r2_y1);
s = strcat(s1,s2);
text(0.15,45, s ,'FontSize',10)

s2 = num2str(r2_y2);
s = strcat(s1,s2);
text(0.2,80, s ,'FontSize',10)
legend('Original Curve', 'C_{3} decreased by 50%', 'C_{3} increased by 50%')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Coefficient: C4

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Coeffs were found from curve fit:
C1 = 1.379;
C2 = 1.722;
C3 = 0.003;
C4 =10.25;
K = 3.5;
a = 1196;
alpha = 12;
m = 0.06036;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y rmse] = rsquare(exp_data,f);
```

```
%%%%%%%%%------C4 Decreased by 50%-------%%%%%%%%%

% Decrease by 50%
C4 = C4/2;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y1(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y1 rmse] = rsquare(exp_data,f);

%%%%%%%%%------C4 Increased by 50%-------%%%%%%%%%

% Increase by 50 (extra factor of 2 because of decrease earlier%
C4 = C4*2*2;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y2(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);
```

```
%Function for R^2
[r2_y2 rmse] = rsquare(exp_data,f);

%Plotting
figure
xlabel('True Strain')
ylabel('True Stress (MPa)')
title('Coefficient Sensitivity Study: C_{4}')
hold on
% plot(Data_1(:,1),Data_1(:,2),'b*')
hold on
plot(se,y,'b')
plot(se,y1,'r')
plot(se,y2,'g')
hold on
txt1 = ' T = 323 K';
text(0.7,20,txt1)
txt2 = ' Strain Rate = 0.001';
text(0.7,10,txt2)
s1 = 'R^2 = ';

s2 = num2str(r2_y);
s = strcat(s1,s2);
text(0.4,65, s ,'FontSize',10)

s2 = num2str(r2_y1);
s = strcat(s1,s2);
text(0.15,45, s ,'FontSize',10)

s2 = num2str(r2_y2);
s = strcat(s1,s2);
text(0.2,80, s ,'FontSize',10)
legend('Original Curve', 'C_{4} decreased by 50%', 'C_{4} increased by 50%')


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Coefficient: K

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Coeffs were found from curve fit:
C1 = 1.379;
C2 = 1.722;
C3 = 0.003;
C4 =10.25;
K = 3.5;
a = 1196;
alpha = 12;
m = 0.06036;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end
```

```
%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y rmse] = rsquare(exp_data,f);

%%%%%%%%%------K Decreased by 50%-------%%%%%%%%%
K = K/2;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y1(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y1 rmse] = rsquare(exp_data,f);

%%%%%%%%%------K Increased by 50%-------%%%%%%%%%
% Increase by 50 (extra factor of 2 because of decrease earlier%
K = K*2*2;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y2(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end
```

```
%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y2 rmse] = rsquare(exp_data,f);

%Plotting
figure
xlabel('True Strain')
ylabel('True Stress (MPa)')
title('Coefficient Sensitivity Study: K')
hold on
hold on
plot(se,y,'b')
plot(se,y1,'r')
plot(se,y2,'g')
hold on
txt1 = ' T = 323 K';
text(0.7,20,txt1)
txt2 = ' Strain Rate = 0.001';
text(0.7,10,txt2)
s1 = 'R^2 = ';

s2 = num2str(r2_y);
s = strcat(s1,s2);
text(0.4,60, s ,'FontSize',10)

s2 = num2str(r2_y1);
s = strcat(s1,s2);
text(0.15,25, s ,'FontSize',10)

s2 = num2str(r2_y2);
s = strcat(s1,s2);
text(0.2,150, s ,'FontSize',10)
legend('Original Curve', 'K decreased by 50%', 'K increased by 50%')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Coefficient: a

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Coeffs were found from curve fit:
C1 = 1.379;
C2 = 1.722;
```

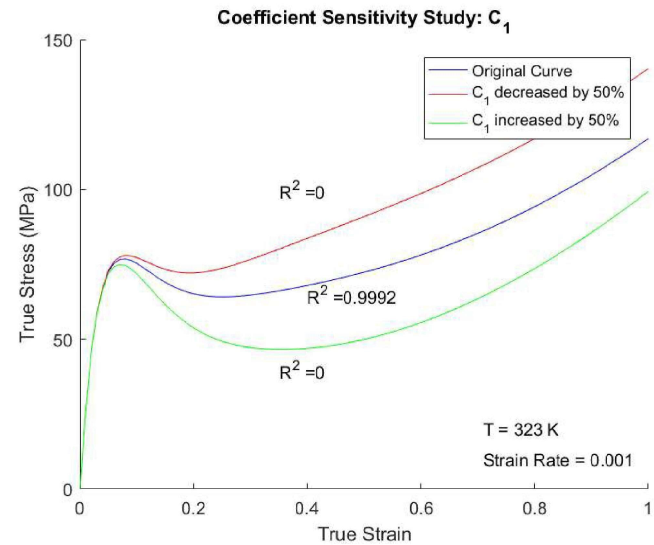
```
C3 = 0.003;
C4 =10.25;
K = 3.5;
a = 1196;
alpha = 12;
m = 0.06036;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y rmse] = rsquare(exp_data,f);

%%%%%%%%%------a Dncreased by 50%-------%%%%%%%%%

% Decrease by 50%
a = a/2;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y1(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
```

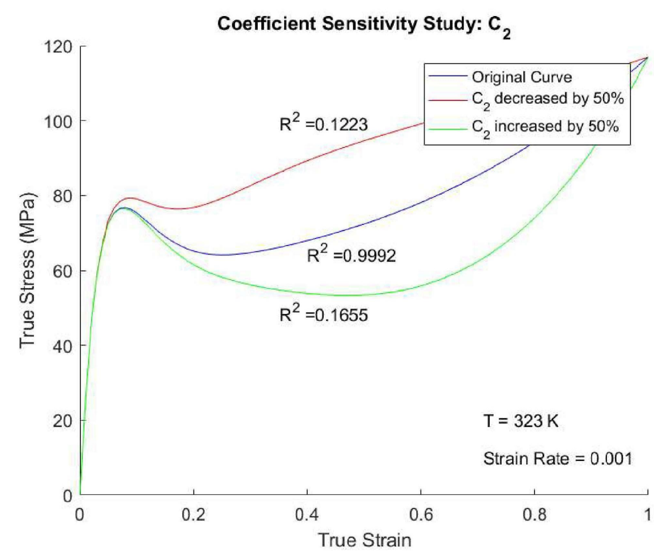
```
exp_data = Data_1(:,2);

%Function for R^2
[r2_y1 rmse] = rsquare(exp_data,f);

%%%%%%%%%------a Increased by 50%-------%%%%%%%%%
% Increase by 50 (extra factor of 2 because of decrease earlier%
a = a*2*2;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y2(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y2 rmse] = rsquare(exp_data,f);

%Plotting
figure
xlabel('True Strain')
ylabel('True Stress (MPa)')
title('Coefficient Sensitivity Study: a')
hold on
hold on
plot(se,y,'b')
plot(se,y1,'r')
plot(se,y2,'g')
hold on
txt1 = ' T = 323 K';
text(0.7,1000,txt1)
txt2 = ' Strain Rate = 0.001';
text(0.7,800,txt2)
s1 = 'R^2 = ';

s2 = num2str(r2_y);
s = strcat(s1,s2);
text(0.4,500, s ,'FontSize',10)

s2 = num2str(r2_y1);
s = strcat(s1,s2);
```

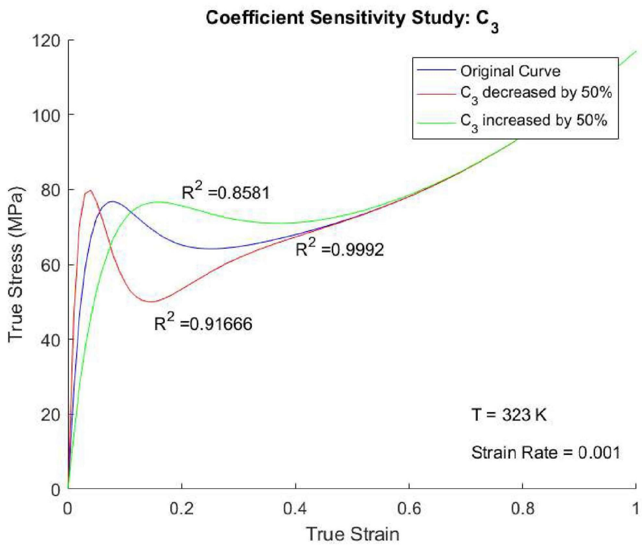
```
text(0.4,300, s ,'FontSize',10)

s2 = num2str(r2_y2);
s = strcat(s1,s2);
text(0.2,2400, s ,'FontSize',10)
legend('Original Curve', 'a decreased by 50%', 'a increased by 50%')


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Coefficient: alpha

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Coeffs were found from curve fit:
C1 = 1.379;
C2 = 1.722;
C3 = 0.003;
C4 =10.25;
K = 3.5;
a = 1196;
alpha = 12;
m = 0.06036;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y rmse] = rsquare(exp_data,f);

%%%%%%%%%------alpha Decreased by 50%-------%%%%%%%%%
alpha = alpha/2;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y1(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y1 rmse] = rsquare(exp_data,f);

%%%%%%%%%------alpha Increased by 50%-------%%%%%%%%%
% Increase by 50 (extra factor of 2 because of decrease earlier%
alpha = alpha*2*2;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y2(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y2 rmse] = rsquare(exp_data,f);

%Plotting
figure
xlabel('True Strain')
ylabel('True Stress (MPa)')
title('Coefficient Sensitivity Study: \alpha')
hold on
plot(se,y,'b')
plot(se,y1,'r')
plot(se,y2,'g')
```

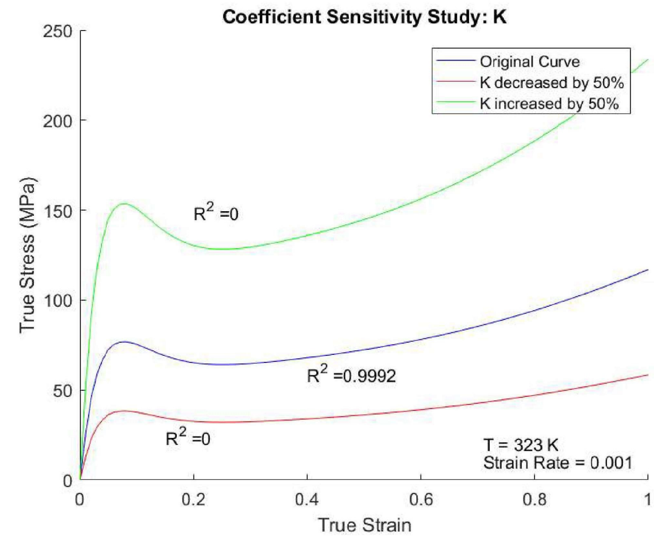
```
hold on
txt1 = ' T = 323 K';
text(0.7,40,txt1)
txt2 = ' Strain Rate = 0.001';
text(0.7,30,txt2)
s1 = 'R^2 = ';

s2 = num2str(r2_y);
s = strcat(s1,s2);
text(0.5,85, s ,'FontSize',10)

s2 = num2str(r2_y1);
s = strcat(s1,s2);
text(0.2,45, s ,'FontSize',10)

s2 = num2str(r2_y2);
s = strcat(s1,s2);
text(0.2,80, s ,'FontSize',10)
legend('Original Curve', '\alpha decreased by 50%', '\alpha increased by 50%')


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Coefficient: m

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Coeffs were found from curve fit:
C1 = 1.379;
C2 = 1.722;
C3 = 0.003;
C4 =10.25;
K = 3.5;
a = 1196;
alpha = 12;
m = 0.06036;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y rmse] = rsquare(exp_data,f);

%%%%%%%%%------m Decreased by 50%-------%%%%%%%%%
m = m/2;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y1(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_y1 rmse] = rsquare(exp_data,f);

%%%%%%%%%------m Increased by 50%-------%%%%%%%%%
% Increase by 50 (extra factor of 2 because of decrease earlier%
m = m*2*2;

%DSGZ Model Calculation
 for i = 1:length(se)
    strain = se(i,1);
    y2(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Experimental Data
exp_data = Data_1(:,2);
```

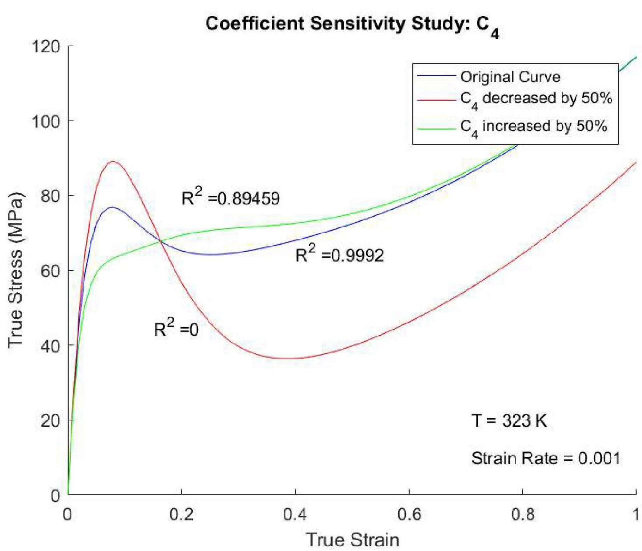
```
%Function for R^2
[r2_y2 rmse] = rsquare(exp_data,f);

%Plotting
figure
xlabel('True Strain')
ylabel('True Stress (MPa)')
title('Coefficient Sensitivity Study: m')
hold on
plot(se,y,'b')
plot(se,y1,'r')
plot(se,y2,'g')
hold on
txt1 = ' T = 323 K';
text(0.7,40,txt1)
txt2 = ' Strain Rate = 0.001';
text(0.7,30,txt2)
s1 = 'R^2 = ';

s2 = num2str(r2_y);
s = strcat(s1,s2);
text(0.5,85, s ,'FontSize',10)

s2 = num2str(r2_y1);
s = strcat(s1,s2);
text(0.2,45, s ,'FontSize',10)

s2 = num2str(r2_y2);
s = strcat(s1,s2);
text(0.2,80, s ,'FontSize',10)
legend('Original Curve', 'm decreased by 50%', 'm increased by 50%')
```

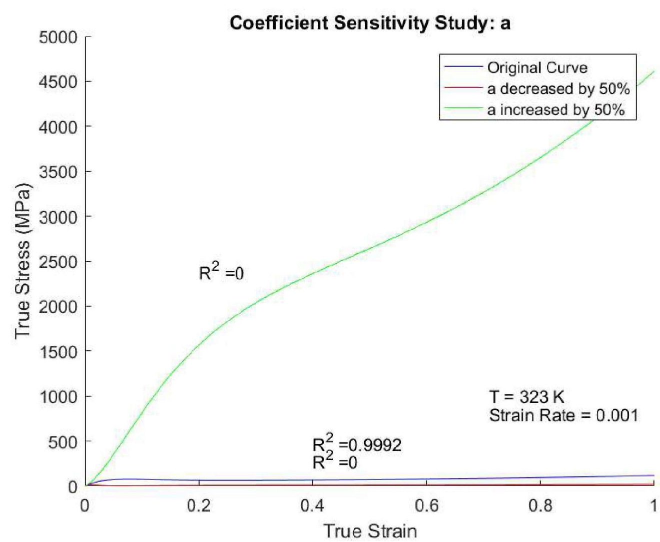


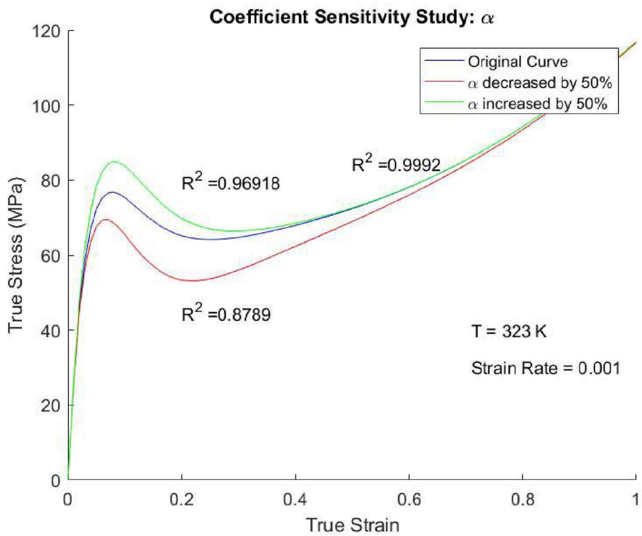


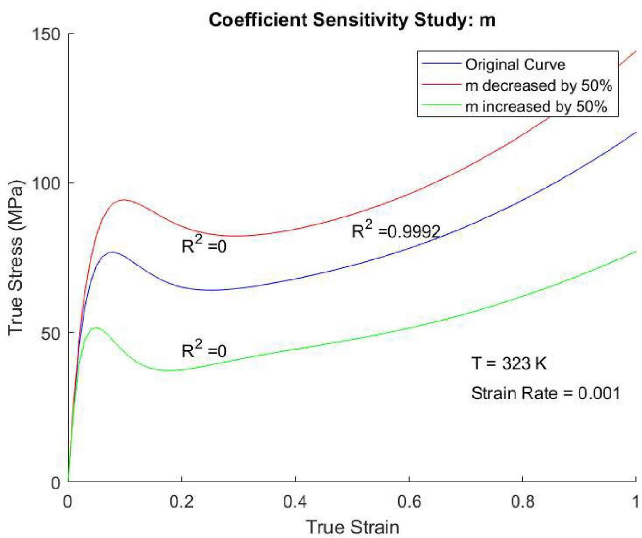

Coefficient Sensitivity Study: C_3
True Stress (MPa)
True Strain
Original Curve
C_3 decreased by 50%
C_3 increased by 50%
R^2 =0.8581
R^2 =0.9992
R^2 =0.91666
T = 323 K
Strain Rate = 0.001


Coefficient Sensitivity Study: K
True Stress (MPa)
True Strain
Original Curve
K decreased by 50%
K increased by 50%
R^2 =0
R^2 =0.9992
R^2 =0
T = 323 K
Strain Rate = 0.001


Coefficient Sensitivity Study: C_4
True Stress (MPa)
True Strain
Original Curve
C_4 decreased by 50%
C_4 increased by 50%
R^2 =0.89459
R^2 =0.9992
R^2 =0
T = 323 K
Strain Rate = 0.001


Coefficient Sensitivity Study: a
True Stress (MPa)
True Strain
Original Curve
a decreased by 50%
a increased by 50%
R^2 =0
R^2 =0.9992
R^2 =0
T = 323 K
Strain Rate = 0.001





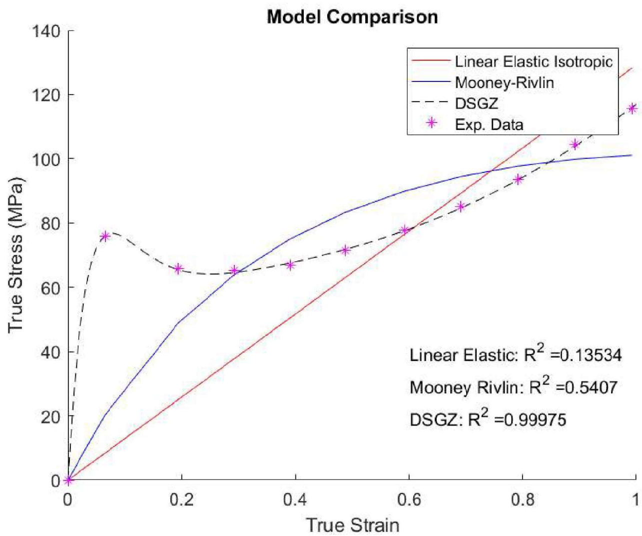
Published with MATLAB® R2016b

```matlab
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%Comparison of The DSGZ Model with other Models
%
%Author: Gurpal Singh
%
%Date: 4/29/2017
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%-----Linear Elastic Isotropic Model-----%%%%%%%%%%%%%%

%Experimental Data for Duan Figure 1
%Temperature: 323 K
%Strain Rate: 0.001/s
T = 323;
strainrate = 0.001;

Data_1= [        0        0
                 0.0650   75.8767
                 0.1936   65.8009
                 0.2928   65.1989
                 0.3907   66.8687
                 0.4874   71.3780
                 0.5914   77.7775
                 0.6912   85.1249
                 0.7911   93.6079
                 0.8912  104.5513
                 0.9928  115.6836];

%For Curve Fitter
x_data = Data_1(:,1);
y_data = Data_1(:,2);


strain = Data_1(:,1);
stress_LEI = zeros(length(strain),1);

%Coefficient found using Curve Fitter
disp('The Material Coefficient for the Linear Elastic Isotropic Model is: ')
C = 129.2

%Calculation
for i = 1:length(strain)
   stress_LEI(i,1) = C*strain(i,1);
end

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_LEI rmse] = rsquare(exp_data,stress_LEI);

%%%%%%%%%%%%%-----Hyper Elastic Mooney-Rivlin Model-----%%%%%%%%%%%%%%
%Coefficients were chosen to provide best fit
```

```matlab
% for transverse data from curve fitting app on matlab
disp('The Material Coefficients for the Mooney-Rivlin Model are:')
C1 = -31.14
C2 = 90.09


stress_MR = zeros(length(Data_1),1);


%Calculation
for i = 1:length(Data_1)
    lambda = sqrt(2*Data_1(i,1) + 1);
    stress_MR(i,1) = 2*C1*(lambda^2 - (1/lambda)) + 2*C2*(lambda - (1/lambda^2));
end

%R^2
[r2_MR rmse] = rsquare(exp_data,stress_MR)

%%%%%%%%%%%%%-----DSGZ Polymer Model-----%%%%%%%%%%%%%%

%Material Coefficients
disp('The Material Coefficients for the DSGZ Model are: ')
C1 = 1.379
C2 = 1.722
C3 = 0.003
C4 =10.25
K = 3.5
a = 1196
alpha = 12
m = 0.06036

%DSGZ Model Calculation
 se = 0:.01:1;
 se = se';
 for i = 1:length(se)
    strain = se(i,1);
    y(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end


strain_data = zeros(length(Data_1(:,2)),1);
%Model evaluated at experimental data points only to compute R^2
stress_DSGZ = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(stress_DSGZ)
    strain = strain_data(i,1);
    stress_DSGZ(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

 %Function for R^2
[r2_DSGZ rmse] = rsquare(exp_data,stress_DSGZ);

%Plotting
figure
title('Model Comparison')
```

```
xlabel('True Strain')
ylabel('True Stress (MPa)')
hold on
plot(strain_data, stress_LEI, 'r')
hold on
plot(strain_data, stress_MR, 'b')
hold on
plot(se, y, 'black--')
hold on
plot(strain_data,Data_1(:,2),'m*')
hold on
legend('Linear Elastic Isotropic', 'Mooney-Rivlin', 'DSGZ', 'Exp. Data')
s1 = 'Linear Elastic: R^2 = ';
s2 = num2str(r2_LEI);
s = strcat(s1,s2);
text(0.6,40, s ,'FontSize',10)
hold on
s1 = 'Mooney Rivlin: R^2 = ';
s2 = num2str(r2_MR);
s = strcat(s1,s2);
text(0.6,30, s ,'FontSize',10)
hold on
s1 = 'DSGZ: R^2 = ';
s2 = num2str(r2_DSGZ);
s = strcat(s1,s2);
text(0.6,20, s ,'FontSize',10)
```

```
The Material Coefficient for the Linear Elastic Isotropic Model is:

C =

  129.2000

The Material Coefficients for the Mooney-Rivlin Model are:

C1 =

  -31.1400


C2 =

   90.0900


r2_MR =

    0.5407


rmse =

   19.2444

The Material Coefficients for the DSGZ Model are:

C1 =

    1.3790


C2 =

    1.7220


C3 =

    0.0030


C4 =

   10.2500


K =

    3.5000


a =

        1196


alpha =

    12


m =

    0.0604
```

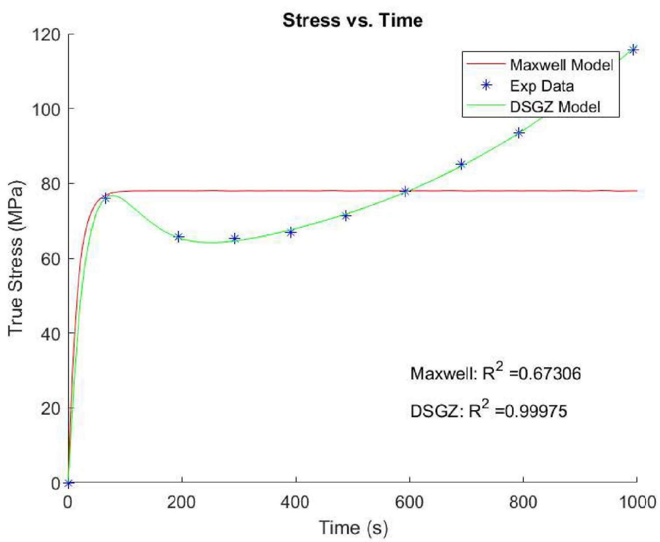


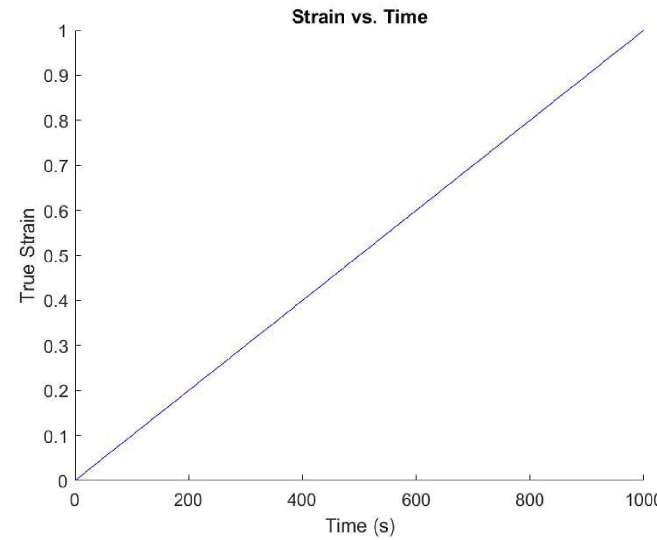
Published with MATLAB® R2016b

```matlab
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%Comparison of The DSGZ Model with the Discrete Element Maxwell Model
%
%Author: Gurpal Singh
%
%Date: 4/29/2017
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Data_1= [        0        0
                 0.0650   75.8767
                 0.1936   65.8009
                 0.2928   65.1989
                 0.3907   66.8687
                 0.4874   71.3780
                 0.5914   77.7775
                 0.6912   85.1249
                 0.7911   93.6079
                 0.8912  104.5513
                 0.9928  115.6836];

time_data = Data_1(:,1)/0.001;
time = [0:10:1000]';

%strain rate is constant
strainrate = 0.001;
%Temperature: 323K
T = 323;
C1 = 1.379;
C2 = 1.722;
C3 = 0.003;
C4 =10.25;
K = 3.5;
a = 1196;
alpha = 12;
m = 0.06036;

s = [0:0.001:1];
t = [0:1:1000];

tspan = 0:1:1000;
%Initial Condition and Coefficients
disp('The material coefficients for the Maxwell Model are:')
k = 5000
c = 78000
y0 = 0;

%Constant Strain rate = 0.0111
[t1, y1] = ode45(@(t,y) fun1(k,c,strainrate,y), tspan, y0);

%Calculating R^2
[time_data,y2] = ode45(@(t,y) fun1(k,c,strainrate,y), time_data, y0);

%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_Maxwell rmse] = rsquare(exp_data,y2);


%%%%%%%%%%%%%%%DSGZ Model%%%%%%%%%%%%%%%%%%%
%  strain = zeros(10,1);
 se = 0:.01:1;
 se = se';
 for i = 1:length(se)
    strain = se(i,1);
    y(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end

%Computing R^2 value of the fit
strain_data = zeros(length(Data_1(:,2)),1);

%Model evaluated at experimental data points only
f = zeros(length(Data_1(:,2)),1);
strain_data(:,1) = Data_1(:,1);

%Calculation
 for i = 1:length(f)
    strain = strain_data(i,1);
    f(i,1) = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T);
 end
%Experimental Data
exp_data = Data_1(:,2);

%Function for R^2
[r2_DSGZ rmse] = rsquare(exp_data,f);

figure
title('Stress vs. Time')
xlabel('Time (s)')
ylabel('True Stress (MPa)')
hold on
plot(t,y1,'r')
hold on
plot(time_data, Data_1(:,2),'b*')

% plot(Data_1(:,1),Data_1(:,2),'b*')
hold on
plot(time,y,'g')
legend('Maxwell Model', 'Exp Data','DSGZ Model')
hold on
s1 = 'Maxwell: R^2 = ';
s2 = num2str(r2_Maxwell);
st = strcat(s1,s2);
text(600,30, st ,'FontSize',10)
hold on
s1 = 'DSGZ: R^2 = ';
s2 = num2str(r2_DSGZ);
st = strcat(s1,s2);
text(600,20, st ,'FontSize',10)
```

```
figure
title('Strain vs. Time')
xlabel('Time (s)')
ylabel('True Strain')
hold on
plot(t,s,'b')
```

```
The material coefficients for the Maxwell Model are:

k =

        5000


c =

       78000
```





Published with MATLAB® R2016b

```matlab
function f = fun(K, C1, C2, C3, C4, a, m, alpha,strain,strainrate,T)

%Function for DSGZ Model
%   Implementation of DSGZ Model

    %f1 is a function of strain
    f1 = (exp(-C1*strain) + strain^C2)*(1 - exp(-alpha*strain));

    %h is a function of strain rate & T
    h = (strainrate^m)*exp(a/T);
    g = h/(strainrate^m);

    %Stress is a function strain, strain_rate, & T
    f = K*(f1+(((strain*exp(1-(strain/(C3*h))))/(C3*h))-f1)*exp((log(g) - C4)*strain))*h;
end
```

```
Not enough input arguments.

Error in fun (line 7)
    f1 = (exp(-C1*strain) + strain^C2)*(1 - exp(-alpha*strain));
```
