UCL Mechanical Engineering 2020/2021

ENGF0004 Coursework 2

NCWT3

April 15, 2021

1 Question 1

1.1 a

For the line integral to be independent from the path of integration, the following conditions must be fulfilled:

$$I = \int_A^B \left(\frac{\partial u}{\partial x}\,\mathrm{d}x + \frac{\partial u}{\partial y}\,\mathrm{d}y \right) \tag{1.1}$$

$$P(x,\,y) = \frac{\partial u}{\partial x} \text{ and } Q(x,\,y) = \frac{\partial u}{\partial y} \tag{1.2}$$

$$\frac{\partial P(x,\,y)}{\partial y} = \frac{\partial Q(x,\,y)}{\partial x} \tag{1.3}$$

Considering the integral:

$$I = \int_A^B \left[e^{-\alpha xy} \left(\frac{\alpha - 2}{x} \right) \mathrm{d}x - \frac{1}{\alpha y} \left(e^{-\alpha xy} - 1 \right) \mathrm{d}y \right] \tag{1.4}$$

$$P(x,\,y) = e^{-\alpha xy} \left(\frac{\alpha - 2}{x} \right) \text{ and } Q(x,\,y) = -\frac{1}{\alpha y} \left(e^{-\alpha xy} - 1 \right) \tag{1.5}$$

$$\frac{\partial P(x,\,y)}{\partial y} = -\alpha x \left(\frac{\alpha - 2}{x} \right) e^{-\alpha xy} = \left(2\alpha - \alpha^2 \right) e^{-\alpha xy} \tag{1.6}$$

$$\frac{\partial Q(x,\,y)}{\partial x} = -\frac{1}{\alpha y} (-\alpha y)\, e^{-\alpha xy} = e^{-\alpha xy} \tag{1.7}$$

$$\therefore 2\alpha e^{-\alpha xy} - \alpha^2 e^{-\alpha xy} = e^{-\alpha xy} \tag{1.8}$$

$$e^{-\alpha xy} \left(\alpha^2 - 2\alpha + 1 \right) = 0 \tag{1.9}$$

$$e^{-\alpha xy} = 0 \rightarrow \text{no solutions} \tag{1.10}$$

$$(\alpha - 1)^2 = 0 \tag{1.11}$$

$$\alpha = 1 \tag{1.12}$$

1.2 b

Calculating the line integral of 1.13 from $O\,(0,\,0)$ to $A(1,\,e-1)$ along $y = e^x - 1$:

$$I = \int_O^A \left(ye^{-2x}\right)(\mathrm{d}x + \mathrm{d}y) \tag{1.13}$$

$$y = e^x - 1 \tag{1.14}$$

$$\mathrm{d}y = e^x\,\mathrm{d}x \tag{1.15}$$

$$I = \int_0^1 \left((e^x - 1)\left(e^{-2x}\right) + (e^x - 1)\left(e^{-2x}\right)(e^x)\right)\mathrm{d}x \tag{1.16}$$

$$= \int_0^1 \left(e^{-x} - e^{-x} - e^{-2x} + 1\right)\mathrm{d}x \tag{1.17}$$

$$= \int_0^1 \left(1 - e^{-2x}\right)\mathrm{d}x \tag{1.18}$$

$$= \left[x + \frac{e^{-2x}}{2}\right]_0^1 \tag{1.19}$$

$$= 1 + \frac{e^{-2}}{2} - 0 - \frac{1}{2} \tag{1.20}$$

$$I = \frac{1}{2}\left(e^{-2} + 1\right) \tag{1.21}$$

1.3 c

1.3.1 i

$$\underline{F}\,(x,\,y,\,z) = \begin{pmatrix} \frac{y}{x^2} \\ \frac{x}{y^2} \end{pmatrix} \tag{1.22}$$

$$\nabla \cdot \underline{F} = \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix} \cdot \begin{pmatrix} \frac{y}{x^2} \\ \frac{x}{y^2} \end{pmatrix} \tag{1.23}$$

$$= \frac{\partial}{\partial x}\left(\frac{y}{x^2}\right) + \frac{\partial}{\partial y}\left(\frac{x}{y^2}\right) \tag{1.24}$$

$$= -\frac{2y}{x^3} - \frac{2x}{y^3} \tag{1.25}$$

$$= -2\left(\frac{y}{x^3} + \frac{x}{y^3}\right) \tag{1.26}$$

1.3.2 ii

$$I = \int_1^2 \int_1^2 \left(-2\left(\frac{y}{x^3} + \frac{x}{y^3}\right)\right) \mathrm{d}x \, \mathrm{d}y \tag{1.27}$$

$$= \int_1^2 \left[-2\left(\frac{y}{-2x^2} + \frac{x^2}{2y^3}\right)\right]_1^2 \mathrm{d}y \tag{1.28}$$

$$= \int_1^2 \left[-2\left(-\frac{y}{8} + \frac{2}{y^3} + \frac{y}{2} - \frac{1}{2y^3}\right)\right] \mathrm{d}y \tag{1.29}$$

$$= \int_1^2 \left(-\frac{3y}{4} - \frac{3}{y^3}\right) \mathrm{d}y \tag{1.30}$$

$$= \left[-\frac{3y^2}{8} + \frac{3}{2y^2}\right]_1^2 \tag{1.31}$$

$$= -\frac{3}{2} + \frac{3}{8} + \frac{3}{8} - \frac{3}{2} \tag{1.32}$$

$$I = -\frac{9}{4} \tag{1.33}$$

1.4 d

1.4.1 i

$$I = \int (\sin x \cos y \, \mathrm{d}y + \cos x \sin y \, \mathrm{d}y) \tag{1.34}$$

$$y = 0 \qquad \mathrm{d}y = 0 \tag{1.35}$$

$$I_{AB} = \int_{x=0}^{\pi} (\sin x) \, \mathrm{d}x = [-\cos x]_0^{\pi} = 2 \tag{1.36}$$

$$x = \pi \qquad \mathrm{d}x = 0 \tag{1.37}$$

$$I_{BC} = \int_{y=0}^{\pi} (-\sin y) \, \mathrm{d}y = [\cos y]_0^{\pi} = -2 \tag{1.38}$$

$$\therefore I = I_{AB} + I_{BC} = 2 - 2 = 0 \tag{1.39}$$

1.4.2 ii

$$I = \int (\sin x \cos y \, \mathrm{d}y + \cos x \sin y \, \mathrm{d}y) \tag{1.40}$$

$$y = x \qquad \mathrm{d}y = \mathrm{d}x \tag{1.41}$$

$$I_{AC} = \int_0^{\pi} (\sin x \cos x + \sin x \cos x) \, \mathrm{d}x \tag{1.42}$$

$$= \int_0^{\pi} \big(\sin(2x)\big) \, \mathrm{d}x \tag{1.43}$$

$$I_{AC} = \left[-\frac{1}{2}\cos(2x)\right]_0^{\pi} = \frac{1}{2} - \frac{1}{2} = 0 \tag{1.44}$$

$$\tag{1.45}$$

1.5 e

1.5.1 i

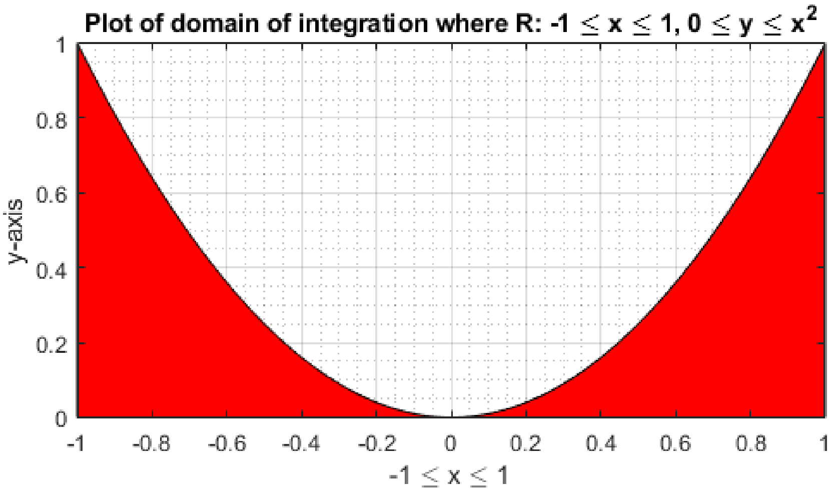

Figure 1: Domain of integration where $R: -1 \leq x \leq 1, 0 \leq y \leq x^2$.

```
clf
x=-1:0.001:1;
y=x.^2;
A=area(x,y);
set(A(1),'FaceColor','red');
axis('image');
xlabel('-1 \leq x \leq 1')
ylabel('y-axis')
title('Plot of domain of integration where R: -1 \leq x \leq 1, 0 \leq y
    \leq x^2')
grid on
grid minor
```

1.5.2 ii

1.6 f

1.6.1 ii

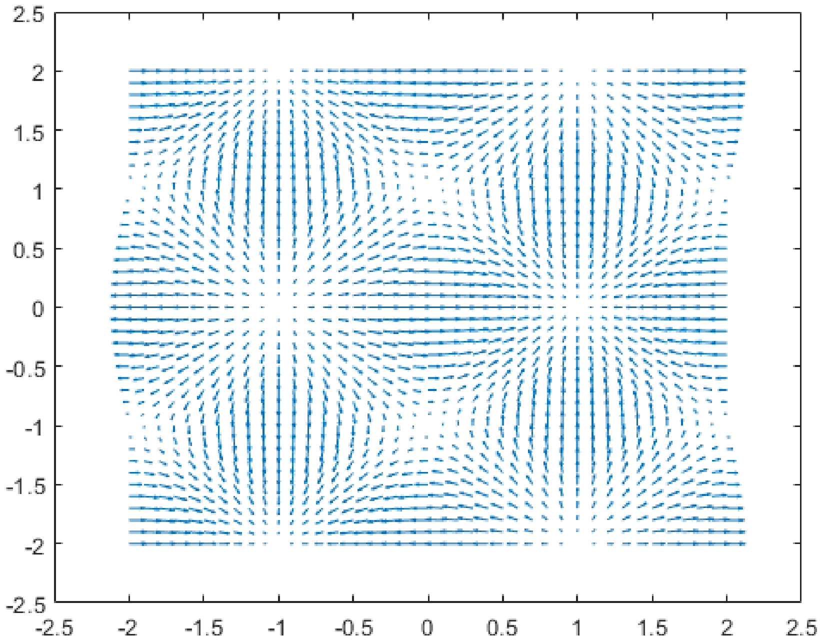

Figure 2:

```
clc
clear
close all

syms x y
z = x*y*exp(-sqrt(x^2 + y^2));
f = (sin((pi/2)*x))*(cos((pi/2)*y));
g = gradient(f,[x,y]);

[X, Y] = meshgrid(-2:0.1:2,-2:0.1:2);
G1 = subs(g(1),[x y],{X,Y});
G2 = subs(g(2),[x y],{X,Y});
quiver(X,Y,G1,G2)
```

2 Question 2

2.1 a

In our series of equations, there are three unknown internal bar forces N_{12}, N_{23}, N_{13}, and three unknown reaction forces, R_{2x}, R_{2y}, R_{3y}. We also have two unknown angles, α and β, and the force F. Given that there are six unknowns that we would like to find and six equations with those variables, the conditions are fulfilled to solve this using matrices. Our answer would be in terms of the variables α, β and F. Values may be assumed, measured or calculated for these.

2.2 b

$$\begin{bmatrix} -\cos\alpha & 0 & \cos\beta & 0 & 0 & 0 \\ -\sin\alpha & 0 & -\sin\beta & 0 & 0 & 0 \\ \cos\alpha & 1 & 0 & 1 & 0 & 0 \\ \sin\alpha & 0 & 0 & 0 & 1 & 0 \\ 0 & -1 & -\cos\beta & 0 & 0 & 0 \\ 0 & 0 & \sin\beta & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} N_{12} \\ N_{23} \\ N_{13} \\ R_{2x} \\ R_{2y} \\ R_{3y} \end{bmatrix} = \begin{bmatrix} 0 \\ F \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{2.1}$$

2.3 c

```
clc
clear
close all

alpha = 0.927295;
beta = 0.643501;
F = 1000;

A = [-cos(alpha) 0 cos(beta) 0 0 0;
     -sin(alpha) 0 -sin(beta) 0 0 0;
      cos(alpha) 1 0 1 0 0;
      sin(alpha) 0 0 0 1 0;
      0 -1 -cos(beta) 0 0 0;
      0 0 sin(beta) 0 0 1];
B = [0; F; 0; 0; 0; 0];

sol = A\B;
```

This returned the following:

$$\begin{bmatrix} N_{12} \\ N_{23} \\ N_{13} \\ R_{2x} \\ R_{2y} \\ R_{3y} \end{bmatrix} = \begin{bmatrix} -800 \\ 480 \\ -600 \\ 0 \\ 640 \\ 360 \end{bmatrix} \tag{2.2}$$

2.4 d

```
clc
clear
close all

alpha = 0.927295;
beta = 0.643501;
```

```
F = 1000;

A = [-cos(alpha) 0 cos(beta) 0 0 0;
     -sin(alpha) 0 -sin(beta) 0 0 0;
     cos(alpha) 1 0 1 0 0;
     sin(alpha) 0 0 0 1 0;
     0 -1 -cos(beta) 0 0 0;
     0 0 sin(beta) 0 0 1];
B = [0; F; 0; 0; 0; 0];

[L,U] = lu(A); %splits matrix A such that A = L*U, L is lower triangular,
     U is upper triangular
y = L\B;
sol = U\y;
```

This returned the following:

$$\begin{bmatrix} N_{12} \\ N_{23} \\ N_{13} \\ R_{2x} \\ R_{2y} \\ R_{3y} \end{bmatrix} = \begin{bmatrix} -800 \\ 480 \\ -600 \\ 0 \\ 640 \\ 360 \end{bmatrix} \tag{2.3}$$

2.5 e

Matlab App Developer was utilised to create a user friendly interface for inputting the Force F, the lengths of each member (as shown in the diagram) and the coefficient matrix. The code is shown below.

```
classdef q2eApp_exported < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public)
        UIFigure                 matlab.ui.Figure
        ForceEditFieldLabel      matlab.ui.control.Label
        Force                    matlab.ui.control.NumericEditField
        L12EditFieldLabel        matlab.ui.control.Label
        L12                      matlab.ui.control.NumericEditField
        L23EditFieldLabel        matlab.ui.control.Label
        L23                      matlab.ui.control.NumericEditField
        L13EditFieldLabel        matlab.ui.control.Label
        L13                      matlab.ui.control.NumericEditField
        UITable                  matlab.ui.control.Table
        FindForcesButton         matlab.ui.control.Button
        betaGaugeLabel           matlab.ui.control.Label
        betaGauge                matlab.ui.control.NinetyDegreeGauge
        alphaGaugeLabel          matlab.ui.control.Label
        alphaGauge               matlab.ui.control.NinetyDegreeGauge
        ProgrammetocalculateforcesLabel  matlab.ui.control.Label
        UITable2                 matlab.ui.control.Table
        Label                    matlab.ui.control.Label
    end
```

```
    % Callbacks that handle component events
    methods (Access = private)

        % Code that executes after component creation
        function startupFcn(app)
            %initialise table
            ATable = ["-cos(alpha)" "0" "cos(beta)" "0" "0" "0";
                "-sin(alpha)" "0" "-sin(beta)" "0" "0" "0";
                "cos(alpha)" "1" "0" "1" "0" "0";
                "sin(alpha)" "0" "0" "0" "1" "0";
                "0" "-1" "-cos(beta)" "0" "0" "0";
                "0" "0" "sin(beta)" "0" "0" "1"];
            %display table and assign table properties
            set(app.UITable2, 'Visible', 'on');
            set(app.UITable2, 'Data', ATable, 'ColumnFormat',{'char'});
            set(app.UITable2,'ColumnEditable',true(1,6))
        end

        % Button pushed function: FindForcesButton
        function FindForcesButtonPushed(app, event)
            %calculate alpha and beta
            alpha = acos((app.L12.Value^2 + app.L23.Value^2 - app.L13.Value^2)/(2*app.L12.Value*app.L23.Value));
            beta = acos((app.L13.Value^2 + app.L23.Value^2 - app.L12.Value^2)/(2*app.L13.Value*app.L23.Value));

            %conversion for display gauges
            app.alphaGauge.Value = rad2deg(alpha);
            app.betaGauge.Value = rad2deg(beta);

            %matrix maths
            A = get(app.UITable2, 'Data');
            %convert user inputs into expressions and evaluate
            c = size(A);
            c = c(1)*c(2);
            for i = 1:c
                A(i) = eval(A(i));
            end
            A = str2double(A);
            B = [0; app.Force.Value; 0; 0; 0; 0];
            sol = A\B;
            namesForces = ["L12";"L23";"L13";"R2x";"R2y";"R3y"];
            vars = [namesForces sol];

            %output to table
            set(app.UITable, 'Visible', 'on');
            set(app.UITable, 'Data', vars, 'ColumnFormat',{'numeric'});

        end
    end

    % Component initialization
```

```
    methods (Access = private)

        % Create UIFigure and components
        function createComponents(app)

            % Create UIFigure and hide until all components are created
            app.UIFigure = uifigure('Visible', 'off');
            app.UIFigure.Position = [100 100 762 598];
            app.UIFigure.Name = 'MATLAB App';

            % Create ForceEditFieldLabel
            app.ForceEditFieldLabel = uilabel(app.UIFigure);
            app.ForceEditFieldLabel.HorizontalAlignment = 'right';
            app.ForceEditFieldLabel.Position = [62 403 36 22];
            app.ForceEditFieldLabel.Text = 'Force';

            % Create Force
            app.Force = uieditfield(app.UIFigure, 'numeric');
            app.Force.Position = [103 403 100 22];

            % Create L12EditFieldLabel
            app.L12EditFieldLabel = uilabel(app.UIFigure);
            app.L12EditFieldLabel.HorizontalAlignment = 'right';
            app.L12EditFieldLabel.Position = [62 370 26 22];
            app.L12EditFieldLabel.Text = 'L12';

            % Create L12
            app.L12 = uieditfield(app.UIFigure, 'numeric');
            app.L12.Position = [103 370 100 22];

            % Create L23EditFieldLabel
            app.L23EditFieldLabel = uilabel(app.UIFigure);
            app.L23EditFieldLabel.HorizontalAlignment = 'right';
            app.L23EditFieldLabel.Position = [62 349 26 22];
            app.L23EditFieldLabel.Text = 'L23';

            % Create L23
            app.L23 = uieditfield(app.UIFigure, 'numeric');
            app.L23.Position = [103 349 100 22];

            % Create L13EditFieldLabel
            app.L13EditFieldLabel = uilabel(app.UIFigure);
            app.L13EditFieldLabel.HorizontalAlignment = 'right';
            app.L13EditFieldLabel.Position = [62 328 26 22];
            app.L13EditFieldLabel.Text = 'L13';

            % Create L13
            app.L13 = uieditfield(app.UIFigure, 'numeric');
            app.L13.Position = [103 328 100 22];

            % Create UITable
            app.UITable = uitable(app.UIFigure);
            app.UITable.ColumnName = {'Force'; 'Value'};
```

```
            app.UITable.RowName = {};
            app.UITable.Position = [272 59 479 185];

            % Create FindForcesButton
            app.FindForcesButton = uibutton(app.UIFigure, 'push');
            app.FindForcesButton.ButtonPushedFcn = createCallbackFcn(app,
                @FindForcesButtonPushed, true);
            app.FindForcesButton.Position = [103 292 100 22];
            app.FindForcesButton.Text = 'Find Forces';

            % Create betaGaugeLabel
            app.betaGaugeLabel = uilabel(app.UIFigure);
            app.betaGaugeLabel.HorizontalAlignment = 'center';
            app.betaGaugeLabel.Position = [186 117 29 22];
            app.betaGaugeLabel.Text = 'beta';

            % Create betaGauge
            app.betaGauge = uigauge(app.UIFigure, 'ninetydegree');
            app.betaGauge.Limits = [0 90];
            app.betaGauge.Position = [154 154 90 90];

            % Create alphaGaugeLabel
            app.alphaGaugeLabel = uilabel(app.UIFigure);
            app.alphaGaugeLabel.HorizontalAlignment = 'center';
            app.alphaGaugeLabel.Position = [62 117 35 22];
            app.alphaGaugeLabel.Text = 'alpha';

            % Create alphaGauge
            app.alphaGauge = uigauge(app.UIFigure, 'ninetydegree');
            app.alphaGauge.Limits = [0 90];
            app.alphaGauge.Orientation = 'northeast';
            app.alphaGauge.ScaleDirection = 'counterclockwise';
            app.alphaGauge.Position = [34 154 90 90];

            % Create ProgrammetocalculateforcesLabel
            app.ProgrammetocalculateforcesLabel = uilabel(app.UIFigure);
            app.ProgrammetocalculateforcesLabel.HorizontalAlignment = '
               right';
            app.ProgrammetocalculateforcesLabel.FontSize = 20;
            app.ProgrammetocalculateforcesLabel.FontWeight = 'bold';
            app.ProgrammetocalculateforcesLabel.Position = [441 534 306
               56];
            app.ProgrammetocalculateforcesLabel.Text = 'Programme to
               calculate forces';

            % Create UITable2
            app.UITable2 = uitable(app.UIFigure);
            app.UITable2.ColumnName = {'Column 1'; 'Column 2'; 'Column 3'
               ; 'Column 4'; 'Colum 5'; 'Colum 6'};
            app.UITable2.RowName = {};
            app.UITable2.ColumnEditable = true;
            app.UITable2.Position = [272 264 479 193];

```

```
            % Create Label
            app.Label = uilabel(app.UIFigure);
            app.Label.HorizontalAlignment = 'right';
            app.Label.Position = [202 479 545 56];
            app.Label.Text = {'Please input the force F, the lengths of
               the members L12, L23 and L13. '; 'The programme will then
               calculate the values of alpha and beta and display them to
                you.'; 'If you would like to change the coeffcient matrix
               , look to the table on the right and adjust as you like.';
                'Click "Find Forces" to calculate the values of the
               internal bar forces and the reaction forces.'};

            % Show the figure after all components are created
            app.UIFigure.Visible = 'on';
        end
    end

    % App creation and deletion
    methods (Access = public)

        % Construct app
        function app = q2eApp_exported

            % Create UIFigure and components
            createComponents(app)

            % Register the app with App Designer
            registerApp(app, app.UIFigure)

            % Execute the startup function
            runStartupFcn(app, @startupFcn)

            if nargout == 0
                clear app
            end
        end

        % Code that executes before app deletion
        function delete(app)

            % Delete UIFigure when app is deleted
            delete(app.UIFigure)
        end
    end
end
```

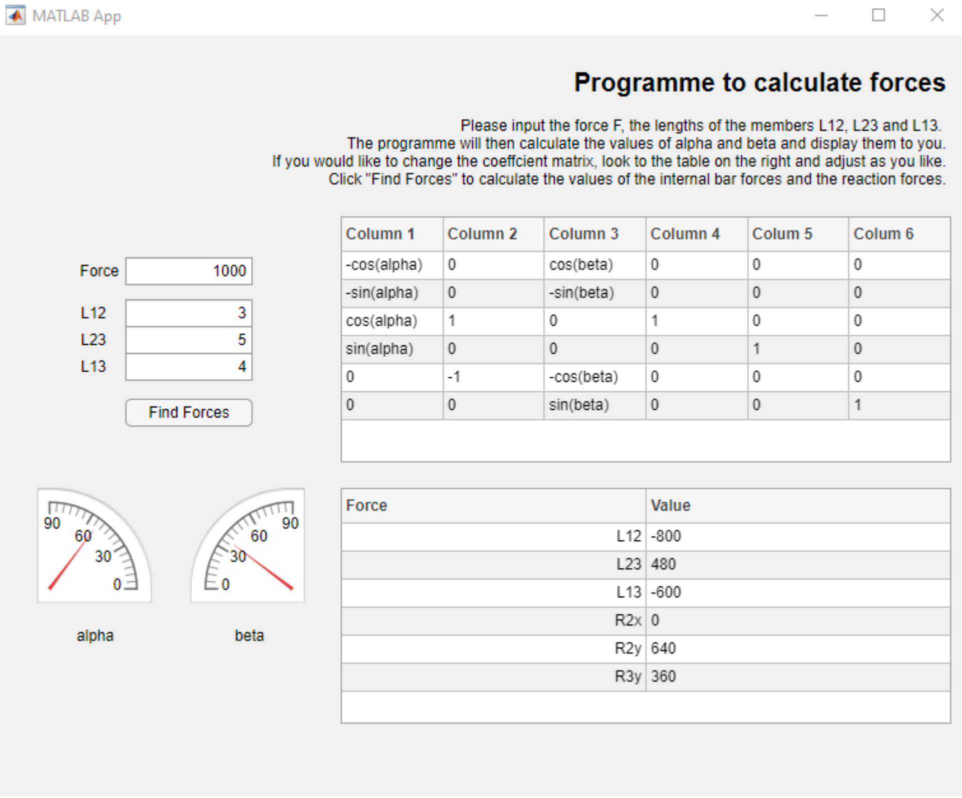

Figure 3: Screenshot from Matlab App, showcasing GUI, input and output parameters.