**UNIT: DYNAMICS OF PLANAR MECHANISMS**

**COURSE: MECHATRONICS**

**TITLE: LAB WORK 1**

**MEMBERS:** WASHINGTON KIGANI KAMADI ENM221-0063/2017

FELIX WANYOIKE GATERU ENM221-0056/2017

MICHAEL GITAU KIMANI ENM221-0066/2017

BROWN IRIA KIMUNYI ENM221-0067/2017

**DATE: November, 1st 2020**

**OBJECTIVES**

1. To implement the Chebyshev spacing formula using MATLAB
2. To implement the Freudenstein’s equation using MATLAB
3. To use Freudenstein's equation to design a four bar linkage with given design parameters.
4. To use the Least Square method to solve for the link lengths of the four bar linkage.
5. To compute the structural errors occurring in a given range of input angles.
6. To compute the transmission angle of the mechanism in a particular range of input angles.

**THEORY**

The synthesis of mechanism is the design or creation of a mechanism to produce a desired output motion for a given input motion. It is the determination of proportions of a mechanism for the given input and output motion. In designing a mechanism to generate a particular function, it is usually impossible to accurately produce the function at more than a few points. The points at which the generated and desired functions agree are known as precision points or accuracy points and must be located so as to minimize the error generated between these points. The best spacing of the precision points, for the first trial, is called Chebyshev’s spacing. Denoted by the formula:

**xj** = Precision points

The subscripts S and F indicate start and finish positions respectively.

Freudenstein's equation relates the input and output angles to the ratios of the lengths of the links denoted by K1, K2 and K3. It is expressed as:

It is possible to design a mechanism to give least deviation from the specified positions. This is done by using least square technique denoted by:

**PROCEDURE**

Our MATLAB program allowed for the user to input the length of the fixed link, and a range of input angles for a four-bar linkage. Using the range of input angles provided and the output angles calculated, Chebyshev’s spacing was applied in order to find three precision points.

Each of these precision points was then fed into the Freudenstein’s equation and solving these equations linearly allowed us to find the three ratios K1, K2 and K3. The lengths of the crank, coupler and follower were then subsequently calculated with the help of the three ratios calculated above.

The transmission angle was then calculated using the entered range of input angles in steps of 5 degrees. This was done by applying the cosine rule twice with the input angle being considered in the first instance and the transmission angle being considered the second time round. Since both of these angles share an opposite length, rearranging the equation allowed for the calculation of the transmission angles.

The variation of the input angles with the input angles was then plotted using the subplot command in MATLAB.

Structural errors were then calculated by rearranging Freudenstein's equation for the given range of input angles. The variation of structural errors as a function of input angles was then plotted.

Using Chebyshev’s spacing five precision points were calculated. We then used these precision points to re-evaluate the length ratios K1, K2 and K3. Using these new ratios, the lengths of the crank, coupler and follower were then calculated. Structural errors were also re-calculated using these newly calculated ratios for the given range of input angles at an increment of 5 degrees

These structural errors were then plotted as a function of input angles on the same axis as the structural errors in the previous scenario.

**DATA PRESENTATION AND DISCUSSION**

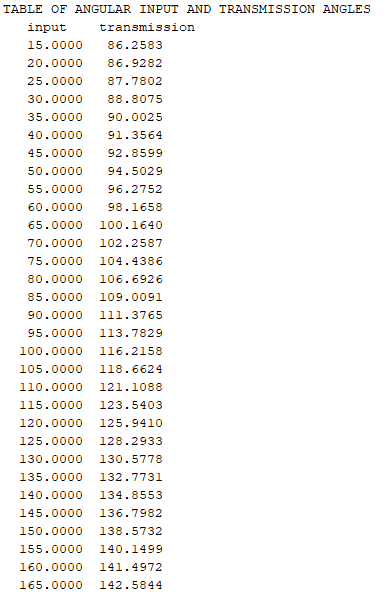
From the computer program’s implementation of the Chebyshev spacing formula and the Freudenstein’s equation, using three precision points, the values of K1, K2 and K3 were as follows;

* K1 = **-7.1003**
* K2 = **-3.4090**
* K3 = **-0.7100**

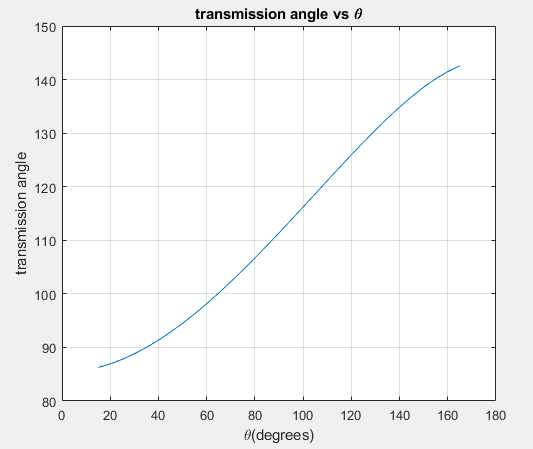
From the length ratios K1, K2 and K3, the lengths of the links were calculated as follows (With the length of the fixed link given as 400mm);

* Coupler ⇒ **442.44926mm**
* Follower ⇒ **120.26793mm**
* Crank ⇒ **57.74377mm**

For a range of input angles ranging from 5° to 165° increasing by 5°, the transmission angles were computed using the formula below;



The values of transmission angles were plotted against the corresponding input angles and the graph is shown below.



The transmission angle is of good quality since it lies between the range **40° < ϑ < 140°**

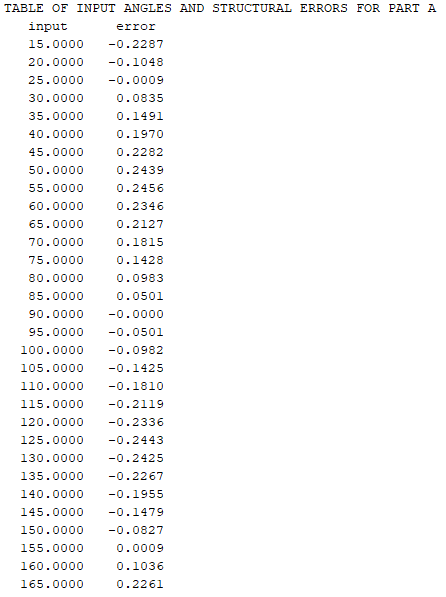
Using the Chebyshev spacing equation to evaluate the values of five precision points, the values of K1, K2 and K3 were solved using the **Least Squares method** and their values found as below;

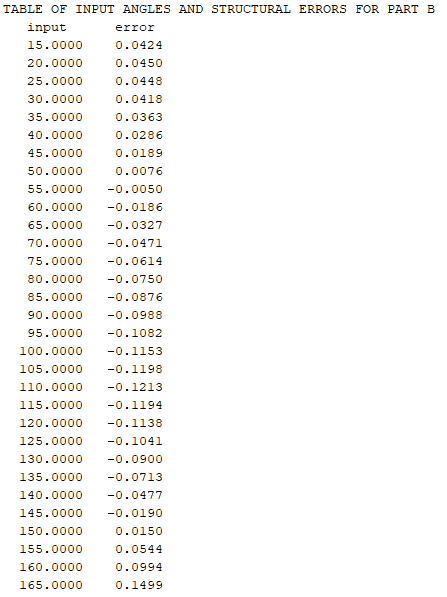
* K1 = **-1.7293**
* K2 =  **-0.70609**
* K3 = **0.4632**

From the values of K1, K2 and K3 obtained from the Least Squares method, the lengths of the links were found as follows (With the length of the fixed link given as 400mm);

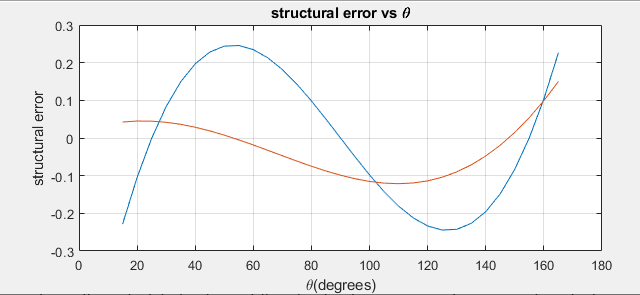
* Coupler ⇒ **658.743mm**
* Follower ⇒ **580.662mm**
* Crank ⇒ **237.090mm**

From the values of the link ratios obtained by using the two methods i.e. Freudenstein’s equation and the Least Squares method, the values of structural errors were calculated by use of the formula below, while incrementing the input angle by 5 degrees in the range, **15° < ϑ < 165°;**





The values obtained for the structural errors were plotted against the corresponding input angle for both a and b. The graph is included below;



From the calculated values of the structural error, a greater error is found when using Freudenstein's equation with three precision points as compared to the structural error found while using the **Least Square method**.

**REFERENCES**

1. Lecture notes by Mrs. Leila Mbagaya
2. J.K. Gupta & R.S. Khurmi. (1976). Theory of Machines. Eurasia Publishing House

**THE CODE**

clc;

%part a (Freudenstein's)

fixed = input('Input the length of the fixed link: ');

theta\_2 = [];

theta\_4 = [];

theta\_not = input('Input the starting range of theta 2: ');

theta\_f = input('Input the final value of the range of theta 2: ');

n = input('Input the number of precision points: ');

j = 1;

while (j < 4)

precision\_2 = 0.5\*(theta\_not+theta\_f)- 0.5\*(theta\_f-theta\_not)\*cosd(180\*(2\*(j)-1)/(2\*n));

precision\_4 = (precision\_2)\*0.43 +65;

theta\_2 = [theta\_2, precision\_2];

theta\_4 = [theta\_4, precision\_4 ];

j = j+1;

end

A = [cosd(theta\_4(1,1)), -cosd(theta\_2(1,1)), 1 ; cosd(theta\_4(1,2)), -cosd(theta\_2(1,2)), 1 ; cosd(theta\_4(1,3)), -cosd(theta\_2(1,3)), 1];

B = [cosd(theta\_2(1,1) - theta\_4(1,1)) ; cosd(theta\_2(1,2) - theta\_4(1,2)) ; cosd(theta\_2(1,3) - theta\_4(1,3)) ];

sol = linsolve(A,B);

K1 = sol(1,1);

K2 = sol(2,1);

K3 = sol(3,1);

fprintf('K1 = %.15g\n',K1);

fprintf('K2 = %.15g\n',K2);

fprintf('K3 = %.15g\n',K3);

crank\_length = fixed /K1;

follower\_length = fixed / K2;

coupler\_length = sqrt((crank\_length\*crank\_length)+(follower\_length\*follower\_length)+(fixed\*fixed)-(K3\*2\*crank\_length\*follower\_length));

fprintf('crank\_length = %.15g\n',crank\_length);

fprintf('follower\_length = %.15g\n',follower\_length);

fprintf('coupler\_length = %.15g\n',coupler\_length);

%transmssion\_angle;

theta = 15 :5 :165;

transmission\_angle = acosd(((coupler\_length^2)+(follower\_length^2)-(crank\_length^2)-(fixed^2)+(2\*crank\_length\*fixed\*cosd(theta)))/(2\*coupler\_length\*follower\_length));

disp(' ')

disp('TABLE OF ANGULAR INPUT AND TRANSMISSION ANGLES')

disp(' input transmission ')

disp([theta',transmission\_angle'])

%subplot(1,1,1),plot(theta,transmission\_angle),xlabel('\theta(degrees)'),ylabel('transmission angle'),title('transmission angle vs \theta'),grid on

output = 65 + 0.43\*theta;

%structural error part a)

structural\_error = K1\*cosd(output)- K2\*cosd(theta)+K3-cosd(theta - output);

disp(structural\_error)

disp('TABLE OF INPUT ANGLES AND STRUCTURAL ERRORS')

disp(' input error ')

disp([theta',structural\_error'])

subplot(1,1,1),plot(theta,structural\_error),xlabel('\theta(degrees)'),ylabel('structural error'),title('structural error vs \theta'),grid on

%part b (Least Square)

disp('LEAST SQUARE')

n5 = input('Input the number of precision points: ');

theta\_22 = [];

theta\_42 = [];

p=1;

while (p < 6)

precision\_22 = 0.5\*(theta\_not+theta\_f)- 0.5\*(theta\_f-theta\_not)\*cosd(180\*(2\*(p)-1)/(2\*n5));

precision\_42 = (precision\_22)\*0.43 +65;

theta\_22 = [theta\_22, precision\_22];

theta\_42 = [theta\_42, precision\_42 ];

p = p+1;

end

ci = cosd(theta\_22);

sumci = sum(cosd(theta\_22));

co = cosd(theta\_42);

sumco = sum(cosd(theta\_42));

sumci2 = sum(cosd(theta\_22).^2);

sumco2 = sum(cosd(theta\_42).^2);

diff = cosd(theta\_22-theta\_42);

sumd = sum(diff);

e1 = [

sumco2, - sum(co.\*ci), sum(co);

sum(co.\*ci), -sumci2, sumci;

sumco,-sumci,5];

e2 = [sum(co.\*diff); sum(ci.\*diff); sumd];

als = linsolve(e1,e2);

K1 = als(1);

K2 = als(2);

K3 = als(3);

fprintf('K1 = %.15g\n',K1);

fprintf('K2 = %.15g\n',K2);

fprintf('K3 = %.15g\n',K3);

crank\_length = fixed /K1;

follower\_length = fixed / K2;

coupler\_length = sqrt((crank\_length\*crank\_length)+(follower\_length\*follower\_length)+(fixed\*fixed)-(K3\*2\*crank\_length\*follower\_length));

fprintf('crank\_length = %.15g\n',crank\_length);

fprintf('follower\_length = %.15g\n',follower\_length);

fprintf('coupler\_length = %.15g\n',coupler\_length);

%structural error part b)

structural\_error = K1\*cosd(output)- K2\*cosd(theta)+K3-cosd(theta - output);

disp('TABLE OF INPUT ANGLES AND STRUCTURAL ERRORS')

disp(' input error ')

disp([theta',structural\_error'])

hold all

subplot(1,1,1),plot(theta,structural\_error),xlabel('\theta(degrees)'),ylabel('structural error'),title('structural error vs \theta'),grid on