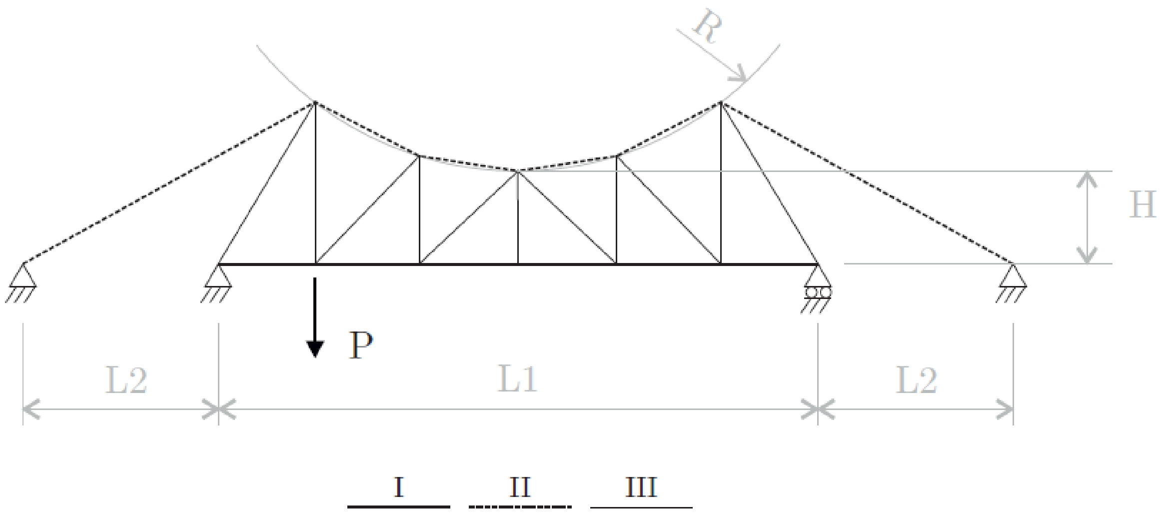
SIMULACIÓN DE UN PUENTE

Trabajo de Simulación de Materiales y Estructuras

Universidad Loyola Andalucía
Facultad de Ingeniería
Grado Ingeniería Informática y Tecnologías Virtuales



Autor

Ferrer Haba, Laura

Curso Académico **2021-22**

ÍNDICE DE CONTENIDO

ÍNDICE DE FIGURAS

1. INTRODUCCION Y OBEJTIVOS

El modelo de la Figura 1 representa un puente de acero que conecta dos orillas de un puente:

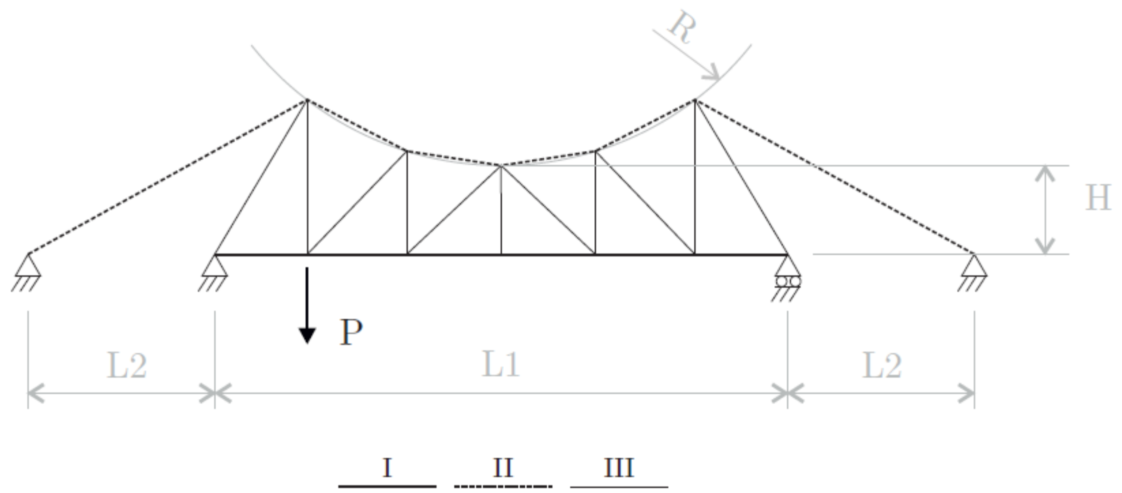


Figura 1:Figura 2D

La estructura está formada por vigas de tres diferentes geometrías diferentes:

- Tipo 1: A = 26 m^2
- Tipo 2: A = 22 m^2
- Tipo 3: A = 18 m^2

Sobre la estructura actúa una carga vertical y de sentido hacia abajo **P** de 200 kN. Además sobre las vigas de Tipo 1 descansa la superficie de rodaje, que puede considerarse homogénea y que tiene un **peso total** de 240 kN. El peso propio de las vigas puede despreciarse. El acero utilizado tiene un módulo de **Young** de 210 GPa y un límite elástico de 235 MPa. Las constantes geométricas son **L** = 36 m, **R** = 30 m, **H** = 6 m, $\mathbf{L_2}$ = $\mathbf{2L_1}$ = **L**. Se pide determinar:

1. Desplazamiento máximo absoluto que se produce en la estructura y su localización
2. Configuraciones deformadas (escaladas) e indeformada de forma gráfica.
3. Tensión máxima soportada por los elementos de la elementos de la estructura (tracción/compresión)
4. Seguridad de la estructura (comprobar si alguna barra plastifica)
5. Represente gráficamente la evolución del desplazamiento máximo de la estructura al variar la longitud L.

El objetivo de esta práctica poder ser capaz de crear un solver numérico (**SolverTruss2D**) que permita la resolución paramétrica de estructuras de elementos (**barra biarticuladas**). Dicha función recibirá un único argumento, que será una estructura, con toda la información necesaria y devolverá una tabla con la solución, representando gráficamente la estructura deformada e indeformada. Todas las funciones que se consideren necesarias deberán ser indexadas al final del propio código o bien incluidas en archivos independientes.

2. EXPLICACIÓN DE LA IMPLEMENTACIÓN

En este apartado explicaré como he resuelto la práctica mediante la utilización de un script, en el cual se engloban dos funciones (**SolverTruss** y **PlotSol**)

2.1. DATOS

Los datos proporcionados para esta práctica son, los siguientes:

```
A1 = 2.6e-3;                % m2 tipo 1
A2 = 2.2e-3;                % m2 tipo 2
A3 = 1.8e-3;                % m2 tipo 3
P = 200000;                 % N
PT = 240000;                % N
E = 2.1e11;                 % 210 GPa modulo de Young
EL = 2.35e8;                % 235 MPa  limite de elasticidad
L = 36;                     % m constantes geometricas
H = 6;                      % m constantes geometricas
R = (L^2+4*H^2)/(8*H);      % m constantes geometricas
L1 = 2*L;                   % m
L2 = L1/4;                  % m L2 ha sido modificada puesto
                            % que salen los puntos muy alejados
```

Figura 2: Datos Proporcionados

Como podemos observar existen tres tipos de vigas, dichas vigas tienen un área diferente cada una.

2.2. CREACIÓN DE MATRICES

Para la correcta implementación deberemos crear una Matriz de elementos y de nodos. La matriz de elementos contendrá los elementos de la figura 3 (que en mi caso son 23), el área de cada tipo de elemento y el módulo de Young. He enumerado los elementos y los nodos de la siguiente forma:

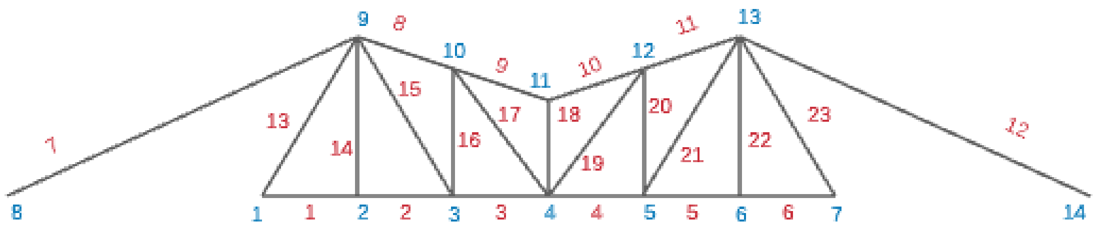


Figura 3: Enumeración de los Elementos

- Nodos: Color azul
- Elementos: Color rojo

Siendo los nodos 1, 8 y 14 los que no se mueven en ninguno de sus ejes (ni x ni y). En cambio el nodo 7 en su eje x si se mueve. Una vez tenido en cuenta esto, pasamos a crear las matrices de elementos y nodos.

<u>La matriz de elementos</u>: será una 23x5, debido a que hay 23 elementos. Es una matriz 23x5 debido a que el primer elemento es el nodo, el segundo y el tercero la conexión entre los nodos, el cuarto el área y el quinto el módulo de Young. Para ello creamos dos bucles for. El primer for irá del elemento 1 hasta el elemento 6. Los elementos se unen entre i e i+1, siendo i el nodo en el que se encuentra el elemento. El segundo for irá del elemento 7 hasta el 23. Al no haber un patrón los datos han sido introducidos a manos.

La matriz de nodos: será una 14x3, debido a que hay 14 nodos. En ella guardaremos la coordenadas que existen entre los nodos, siendo estas los elementos dos y tres respectivamente de la matriz. Para ello elegiremos el centro de coordenadas en el nodo 4. Hemos de tener en cuenta H, R, L1 y L2.

2.3. CONDICIONES DE DESPLAZAMIENTOS

Las condiciones de desplazamiento son los desplazamientos que hay que tener en cuenta. Estos serán las componentes x e y de los nodos. Estos indican que nodos no se mueven que son los de sujeción, estos nodos son el 1, 8 y el 14, y que nodos si se mueven, por lo que el resto de los nodos se mueven salvo el nodo 7 que solo se mueve en la componente horizontal.

2.4. CONDICIONES DE FUERZA

Las condiciones de fuerza son las fuerzas que hay que tener en cuenta. Estas serán las fuerzas que se les aplique a un nodo, el único nodo al que se le aplica una fuerza es el nodo 2, pero existe una fuerza que se le aplica entre todos los nodos de la viga 1, siendo esta 240KN.

2.5. FUNCIÓN SolverTruss

2.5.1. DATOS

Lo primero que hay que hacer es sacar el tamaño del problema. Para ello sacamos el tamaño de Mnod (Matriz de nodos), estos serán los grados de libertad por 2 dado que cada nodo tiene componente x e y. Con los grados de libertad construimos la matriz K, esta será la matriz de rigidez local. Y por último sacamos el tamaño de la matriz de elementos. Este tamaño servirá para iterar en cada elemento.

Los nodos i y j representan las columnas 2 y 3 de la matriz de elementos, siendo estos los nodos que une.

2.5.2. PROPIEDADES DE LA VIGA

Sacamos las propiedades de la vida una vez dentro del for. En mi caso la Matriz de Elementos tiene como parámetros el área (A1, A2, A3) de los tipos de barras que hay y el módulo de Young (E). La L podemos calcularla con la fórmula: sqrt((x2 – x1)^2 + (y2 – y1)^2) puesto que conocemos las componentes x e y de los nodos. Asimismo podemos calcular theta debido a que conocemos el cateto opuesto y su hipotenusa (será la componente y de los nodos) por lo que podemos aplicar el arcoseno para hallar el ángulo.

2.5.3. MATRIZ DE RIGIDEZ LOCAL

La matriz de rigidez local se calcula con la fórmula $k = T^T k' T$, siendo $T = \begin{bmatrix} l & m \\ -m & l \end{bmatrix} \rightarrow \begin{cases} m = \sin\sigma \\ l = \cos\sigma \end{cases}$ y $k' = EA/L$. . Por otro lado, creamos 4 nuevos índices para establecer i y j puesto que al entrar en el bucle se super ponen para poder hacer esto sabemos que se cumple la condición de que el nodo 1 s 1 y 2 por lo que la componente x de i será i*2 y su componente y será (i*2)-1, pasará lo mismo para j.

2.5.4. REDUCIMOS EL SISTEMA

Reducimos el sistema para quedarnos con la parte que nos interesa de él, es decir, quitamos los desplazamientos que sean cero.

2.5.5. RESUELVO EL SISTEMA REDUCIDO

Resolvemos el sistema reducido, eliminando los elementos correspondientes a desplazamientos nulos y obtengo la matriz de contorno completa.

2.6. FUNCIÓN PlotSol

2.6.1. INDEFORNADO

Sacamos los elementos de las columnas 2 y 3 de la matriz de elementos. Después obtenemos la componente x e y de cada elemento (nodo). A cada tipo de barra se le atribuye un color, en mi caso la barra1 será de color azul, mientras que la barra 2 será verde y la barra 3 será roja.

2.6.2. DEFORMADO

Sacamos los elementos de las columnas 2 y 3 de la matriz de elementos. Después obtenemos la componente x e y de cada elemento (nodo). A cada tipo de barra se le atribuye un color, en mi caso la barra1 será de color azul, mientras que la barra 2 será verde y la barra 3 será roja. Por otro lado, creamos 4 nuevos índices para establecer i y j puesto que al entrar en el bucle se super ponen para poder hacer esto sabemos que se cumple la condición de que el nodo 1 s 1 y 2 por lo que la componente x de i será i*2 y su componente y será (i*2)-1, pasará lo mismo para j. Asimismo multiplicamos U por 100 para que se observe el cambio del puente.

2.6.3. NUMERACIÓN DE LOS PUNTOS

Para enumerar solo habrá que convertir el nodo en número ya que lo toma como un string y usar la función text

3. DESPLAZAMIENTO MÁXIMO ABSOLUTO QUE SE PRODUCE EN LA ESTRUCTURA Y SU LOCALIZACIÓN

Para calcular el desplazamiento máximo calcularemos el módulo de cada nodo para obtener cuanto se mueven. Para ello crearemos una nueva función a la cual le pasaremos la U calculada previamente y haremos un bucle que recorra todos los elementos del vector U de dos en dos y calcule el módulo. Una vez calculado el módulo sacaremos el máximo usando la función max de Matlab.

```
function [Umax, U_nu]  = desMax(U)
    %Leo el tamaño del problema
    [Nel, ~] = size(U);
    U_nu = zeros(Nel/2, 1);
    a = 1;

    for spring = 1:2:Nel
        %Identifico los nodso i-j del vector U
        i = U(spring, 1);
        j = U(spring+1, 1);
        U_nu(a) =  sqrt(i^2+j^2);
        a = a + 1;
    end

    %Sacamos el desplazamiento maximo absoluto
    Umax = max(U_nu);

end
```

Figura 4: Desplazamiento Máximo

4. CONFIGURACIONES DEFORMADA (ESCALADA) E INDEFORMADA DE FORMA GRÁFICA.

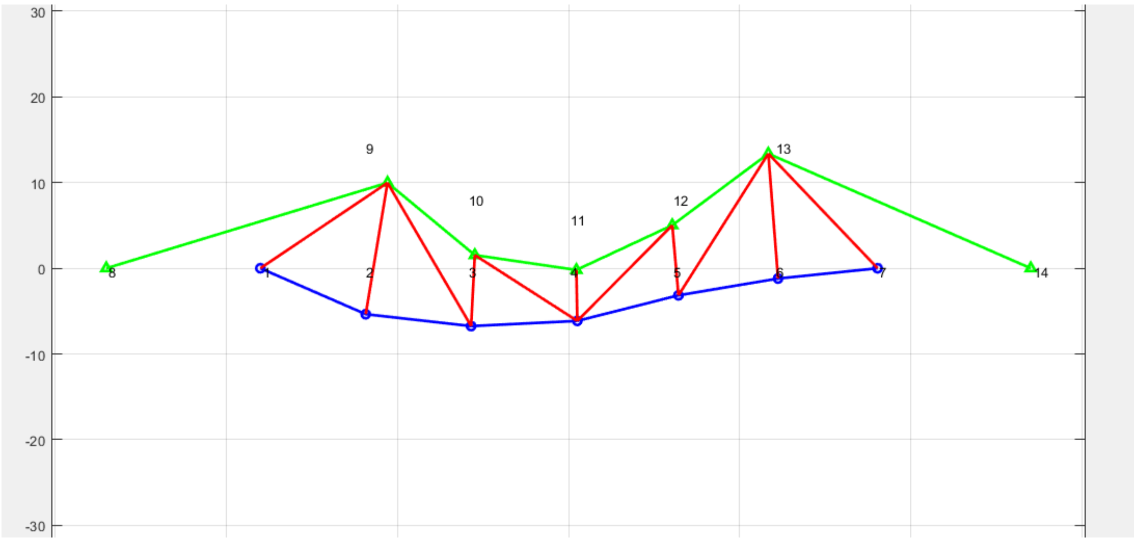


Figura 5:Ploteo Deformado

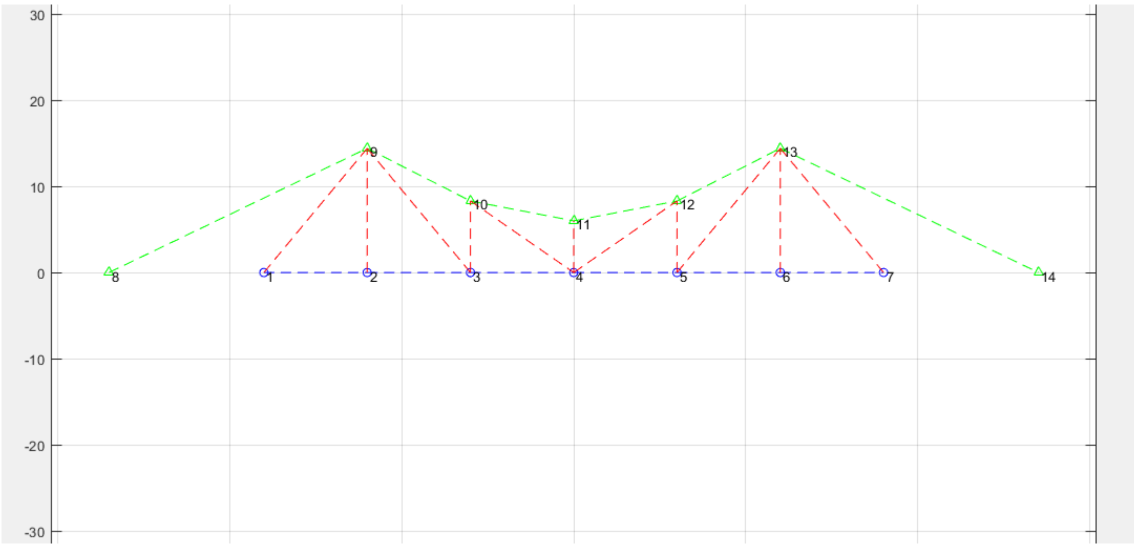


Figura 6: Ploteo Indeformado

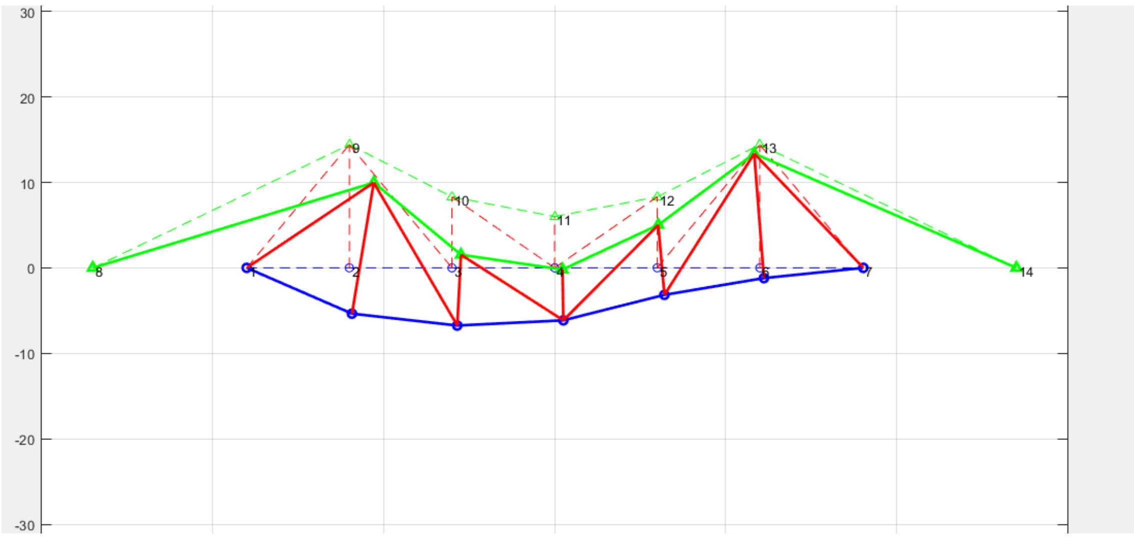


Figura 7: Ploteo Total

5. TENSIÓN MÁXIMA SOPORTADA POR LOS ELEMENTOS DE LA ESTRUCTURA (TRACCIÓN/COMPRESIÓN)

5.1. DATOS

Los datos serán los mismo sacados anteriormente en la función **SolverTruss**.

5.2. PROPIEDADES DE LA VIGA

Las propiedades de la vida serán los mismo sacados anteriormente en la función **SolverTruss**.

5.3. MATRIZ DE RIGIDEZ LOCAL

La matriz de rigidez local será los mismo sacados anteriormente en la función **SolverTruss**. Salvo que en este caso no tendremos que calcular K, sino S. Para ello sacamos U_barra que será solo 4 elementos de la matriz U de desplazamiento. U_barra se multiplicará por T para obtener f_barra. Una vez obtenido f_barra sacaremos S, siendo esta f_barra(1) dividido por el área, f_barra(1) significa que cogemos el primer elemento del vector f_barra ya que el segundo y cuarto elemento no nos interesa y el tercero coincide con el primero. Por último usaremos la funcion max para hallar la tensión máxima.

```
function Smax = tensionMaxima(Mele, Mnod, U)
    %Leo el tamaño del problema
    [Nel, ~] = size(Mele);

    %La tension es la que define como se comporta el material,
    %carga total = F/A(Area)

    S = zeros(23, 1);
    for s = 1:Nel
        %Identifico los nodso i-j del elemento y sus gdl correcpondientes
        i = Mele(s, 2); %Cogemos el elemento 2 de la matriz
        j = Mele(s, 3); %Cogemos el elemento 3 de la martiz

        %Saco las porpiedades de la viga
        A =  Mele(s, 4); %Cogemos el elemento 4 de la matriz
        E = Mele(s, 5);  %Cogemos el elemento 5 de la matriz
        L = sqrt((Mnod(j,2) - Mnod(i,2))^2 + (Mnod(j,3) - Mnod(i,3))^2);
        theta = asin((Mnod(j,3) - Mnod(i,3))/L);
        e = (2*i)-1;
        f = 2*i;
        g = (2*j)-1;
        h = 2*j;

        %Construyo su matriz de rigidez local
        kele = zeros(4);
        kele([1 3], [1 3]) = E*A/L*[1 -1; -1 1];
        T0 = [cos(theta) sin(theta); -sin(theta) cos(theta)];
        T = [T0 zeros(2); zeros(2) T0]';
        Kele = T*kele*T;
        U_barra = U([e f g h]);
        u_barra = T*U_barra;
        f_barra = Kele*u_barra;

        S(s) = f_barra(1)/A;
    end

    %Sacamos la tension maxima
    Smax = max(abs(S));

end
```

Figura 8: Tensión máxima soportada

6. SEGURIDAD DE LA ESTRUCTURA (COMPROBAR SI ALGUNA BARRA PLASTIFICA)

En esta función comprobaremos que la tensión máxima calculada previamente supera o no el límite elástico. Si supera dicho límite la estructura plastificará, en cambio si no la supera no plastificará.

```
function P = plastifica(Smax, EL)
        %Obtenemos si la estrutura plastifica
    if Smax > EL
        P = ' la estructura plastifica';
    else
        P = ' las estructura no plastifica';
    end
end
```

Figura 9: Seguridad de la Estructura

7. REPRESENTAR GRÁFICAMENTE LA EVOLUCIÓN DEL DESPLAZAMIENTO MÁXIMO DE LA ESTRUCTURA AL VARIAR LA LONGITUD L.

Para poder representar esta gráfica habrá que realizar la función **SolverTruss**, pero diciéndole que L varia, en mi caso L varia de 1 a 100. Estas variaciones pueden realizarse sin ningún problema ya que no dará raíces imaginarias.

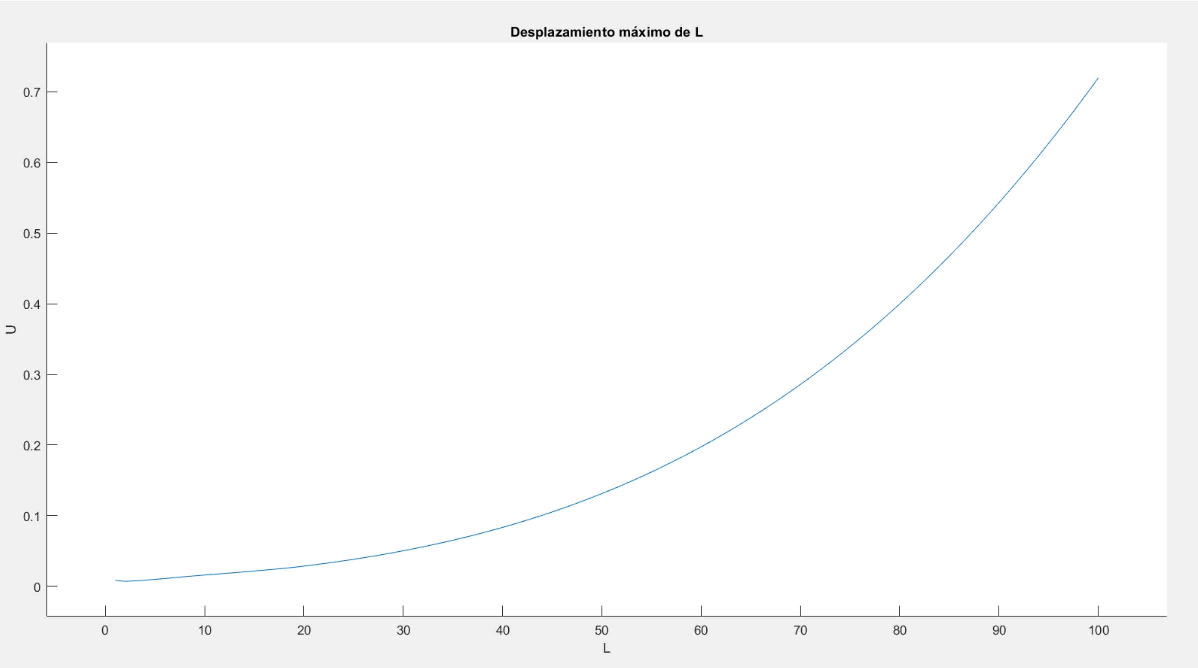


Figura 10: Representación de L

8. SOLUCIONES

```
----------------------Resolución Práctica 1----------------------
Los desplazamientos por nodos son:
         0
    0.0537
    0.0677
    0.0623
    0.0327
    0.0129
    0.0011
         0
    0.0528
    0.0686
    0.0626
    0.0329
    0.0124
         0

Por lo que el desplazamiento máximo es:6.855614e-02 en el nodo:10
La tensión máxima soportada es: 4.094861e+08
Por lo que  la estructura plastifica
```

Figura 11: Resultados