

Universidad Autónoma de Nuevo León
Facultad de Ingeniería Mecánica y Eléctrica

PRÁCTICA #2:

Diseño del marco de una Bicicleta

Asignatura: Lab. Biomecanica

Maestro(a): Yadira Moreno Vera

Grupo/Brigada: 309

Nombre	Matricula
Victor Alan Cavazos Ramírez	1902881
Arturo Mariscal Picón	1806989
José Francisco Juárez Segundo	1992319
Jorge Eduardo Ortiz Cruz	1992029

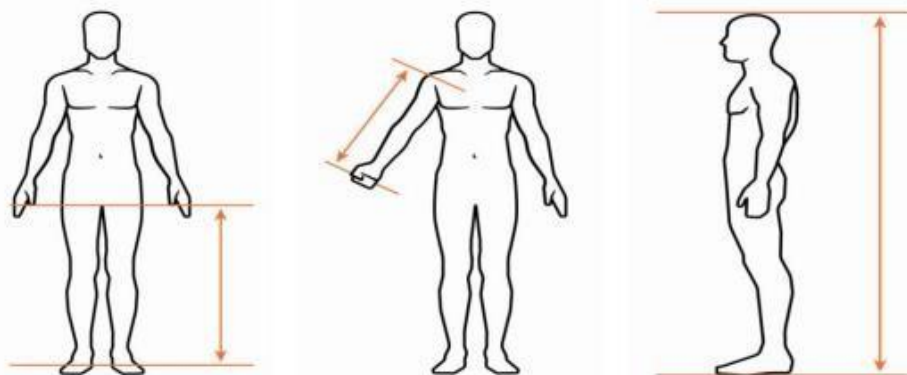
Definición

Aprender a utilizar la lógica del código de Matlab para colocación de cargas, apoyos y fuerzas dentro de un espacio de diseño propuesto.

Estado del arte:

Con el fin de seguir unos estándares de fabricación, las marcas de bicicletas establecen unas medidas estándar que son clasificadas en diferentes tallas, según las necesidades del cliente. Esto facilita y abarata mucho la producción en serie, con lo que se consiguen mejores precios para el cliente.

- Altura Corporal: La medición se realiza sin calzado, y va desde la clavícula hasta la planta del pie.
- Longitud de piernas: La medida se realiza desde la entrepierna hasta la planta del pie.
- Longitud de brazos: Se realiza la medición desde encima de la articulación del hombro hasta el centro de la palma de la mano.



En cambio, también existen marcas que son capaces de realizar y fabricar bicicletas únicas y personalizadas para cada cliente. Es un proceso que implica un elevado coste, ya que tu bicicleta es fabricada exclusivamente para una persona, con lo que el precio de venta aumente considerablemente.

En las medidas de su diseño se empiezan a definir los diferentes ángulos que tiene el cuadro y los efectos que tienen estos sobre el comportamiento de la bicicleta.

- Angulo de la dirección.
- Angulo del sillín.
- Angulo de potencia
- Avance de la horquilla.
- Avance de la dirección.
- Longitud de las vainas traseras.

➤ Longitud del tubo horizontal superior.

➤ Altura eje del pedalier

La optimización mediante FEA representa un campo de estudio creciente en la ingeniería. Aunque hay muchos programas y técnicas disponibles para realizarla, los estudios industriales, los estudios de sensibilidad y la optimización de la forma son los más utilizados en la actualidad. Los dos métodos más utilizados de optimización de la forma son la búsqueda de gradientes y el diseño de experimentos (DoE). Este último está basado en cálculos de superficie de respuesta y produce soluciones “sólidas”, es decir, eficaces en la gama más amplia de posibles condiciones de servicio durante la vida del producto.

SolidWorks Simulation utiliza un método de optimización basado en DoE. Para ejecutar un problema, el ingeniero proporciona los valores máximos y mínimos de sus variables de diseño de cotas y, a continuación, selecciona una optimización “Estándar” o de “Alta calidad”. El método estándar funciona suponiendo que la curva de respuesta objetiva entre los valores límite sea lineal y calcula sólo la respuesta a estos valores. La optimización de alta calidad tiene en cuenta la posibilidad de una respuesta de segundo orden entre los límites y evalúa un valor medio así como los extremos. La ilustración 7 muestra las iteraciones de diseño automático realizadas para una suspensión de automóvil en la que el tamaño de los tres redondeos es diferente. En este caso, la restricción del diseño consiste en no superar el límite elástico del material bajo la carga estática.

Estado del arte

1	Carátula	Facultad de Ingeniería Mecánica y Eléctrica. Diseño del marco de una bicicleta. 19/10/2022
2	Objetivo	Utilizar el método de optimización para la realización de un marco de bicicleta, así como aprender a utilizar la lógica del código de Matlab para colocación de cargas, apoyos y fuerzas dentro de un espacio de diseño propuesto.
3	Contenido	Ejemplo de programa realizado en MatLAB para realizar mejoras en las especificaciones
4	Palabras Clave	Bicicleta, optimización, Matlab
5	Conclusión	Tras realizar la optimización adecuada pudimos ver y cambiar los valores para observar el marco de bicicleta de las mejores maneras posibles
6	Referencias Bibliográficas	➤ Fundación ESTEYCO, (2003), Pilar Carrizosa, La ingeniería de la bicicleta, Editado en Madrid.

Propuestas:

Las diferencias entre las bicicletas de los hombres y de las mujeres es el diseño del cuadro. La bicicleta de las mujeres está diseñada para hacer más fácil de montar mientras que las bicicletas del hombre no. En esta práctica vamos a optimizar el diseño del marco para mejorarlo.

Primero vamos a considerar solo la carga y el apoyo. Para esto editaremos líneas 80 y 81 del código de matlab:

➤ 80 $F(2,1)=1$; ➤ 81 $\text{fixddofs} = 2 * \text{nelx} * (\text{nely} + 1) + 1:2 * (\text{nelx} + 1) * (\text{nely} + 1)$;

Guarde el código en el mismo directorio. Luego ejecute Matlab con:

➤ $\text{top}(20,20,0.33,3.0,1.5)$ El resultado debe ser similar a la figura 2.3. El dominio de diseño se discretiza 20 veces en 20 elementos finitos. Tal vez piense que la magnitud de la fuerza no es realista y que el modulo de Young debe ser corregido en la línea 89. Comúnmente $E=2 * 10^{11} \text{N/m}^2$



Figura 2.3 : Resultado de la primera optimización



Figura 2.4 : Resultado de la segunda optimización

También el tamaño de los elementos finitos se define como una vez por 1 unidad. Estos valores se tienen que cambiar para obtener valores correctos de optimización, pero no siempre es necesario cambiarlos ya que son solo correcciones de las escalas. Sin embargo, si se quieren corregir se puede modificar la línea 41 para mantener la precisión mientras se resuelven las ecuaciones:

➤ 41 while ((l2-l1)/l2 > 1e-4)

Penalización y filtro de radio

La sintaxis de la función es:

➤ top(nelx,nely,volfrac,penal,rmin)

Donde las variables denotan lo siguiente:

- nelx es el número de elementos finitos en la dirección horizontal.
- nely es el número de elementos finitos en la dirección vertical.
- volfrac es la fracción de volumen en el dominio de diseño.
- penal es la penalización de las densidades intermedias. Una penalización alta hará la solución en blanco y negro, es decir los elementos finitos estarán llenos o vacíos. Una penalización = 1 significa que no hay penalización de las densidades intermedias.
- rmin es un radio de filtro para un filtro que hace que el diseño de malla independiente.

Definición de regiones vacías.

El resultado en la figura 2.3 no deja ninguna zona hueca para la rueda delantera. Vamos a llamar a los elementos finitos en este pasivo vacío, y definir una matriz con ceros en elementos libres y seros en pasiva. Agregue las siguientes líneas al código de MATLAB entre la línea 5 y 6 para hacer esto:

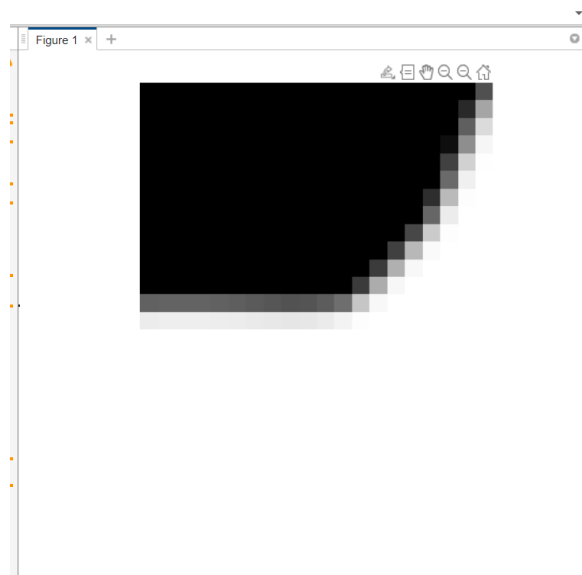
```
for ely = 1:nely
    for elx = 1:nelx
        if ((elx)^2+(ely-nely)^2) < (0.65*nelx)^2
            passive(ely,elx) = 1;
        else
            passive(ely,elx) = 0;
        end
    end
end
x(find(passive))=0.001;
```

El último comando inicializa todos los elementos de la zona hueca en el bajo valor 0.001. También tenemos que actualizar la línea 29 y 40 e insertar una línea adicional entre 43 y 44:

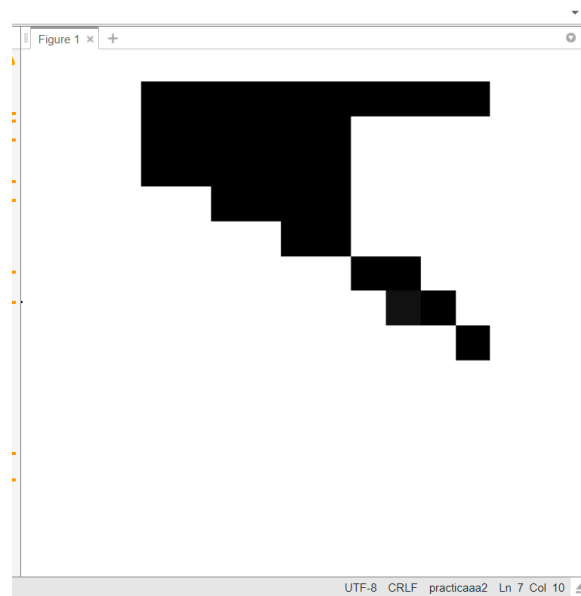
- 29 [x] = OC(nelx,nely,x,volfrac,dc,passive);
- 40 function [xnew]=OC(nelx,nely,x,volfrac,dc,passive)
- 43b xnew(find(passive)) = 0.001;

Realiza estos cambios y ejecuta con:

- top(20,20,0.33,3,1.5)



top(10,10,0.33,3.0,0.5)



Programa

```
%Práctica 2
%Brigada 408
%Sem Agosto Diciembre 2022
%Jorge Eduardo Ortiz Cruz
%Victor Alan Cavazos Ramírez
%José Francisco Juárez Segundo
%Arturo Mariscal Picón
function practicaaa2(nelx,nely,volfrac,penal,rmin)
% INITIALIZE
x(1:nely,1:nelx) = volfrac;
loop = 0;
for ely = 1:nely
for elx = 1:nelx
if((elx)^2+(ely-nely)^2)<(0.65*nelx)^2 passive(ely,elx) = 1;
else
passive(ely,elx)=0;
end
end
end
x(find(passive))=0.001; change=1.;
while (change>0.01)
    loop=loop + 1;
    xold=x;
% FE-ANALYSIS
[U]=FE(nelx,nely,x,penal);
[KE]= lk;
c=0.;
for ely= 1:nely
for elx=1:nelx
    n1= (nely+1) *(elx-1)+ely
    n2 = (nely+1)*elx+ely;
    Ue=U([2*n1-1; 2*n1; 2*n2-1; 2*n2; 2*n2+1; 2*n2+2; 2*n1+1; 2*n2+2],1);
    c=c+x(ely,elx)^penal*Ue'*KE*Ue;
    dc(ely,elx)=-penal*x(ely,elx)^(penal-1)*Ue'*KE*Ue;
end
```

```

end
% FILTERING OF SENSITIVITIES
[dc] = check(nelx,nely,rmin,x,dc);
% DESIGN UPDATE BY THE OPTIMALITY CRITERIA METHOD
[x] = OC(nelx,nely,x,volfrac,dc,passive);
% PRINT RESULTS
change = max(max(abs(x-xold)));
disp(['It.: ' sprintf('%4i',loop) 'Obj.: ' sprintf(' %10.4f',c) ...
      'Vol.: ' sprintf('%6.3f',sum(sum(x))/(nelx*nely)) ...
      'ch.: ' sprintf('%6.3f',change )])
% PLOT DENSITIES
colormap(gray); imagesc(-x); axis equal; axis tight; axis off; pause(1e-6);
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% OPTIMALITY CRITERIA UPDATE %%%%%%%%%%%%%%
function [xnew]=OC(nelx,nely,x,volfrac,dc,passive)
l1 = 0;
l2 = 100000;
move = 0.2;
while (l2-l1 > 1e-4)
    lmid = 0.5*(l2+l1);
    xnew = max(0.001,max(x-move,min(1.,min(x+move,x.*sqrt(-dc./lmid))));
    if sum(sum(xnew)) - volfrac*nelx*nely > 0;
        l1 = lmid;
    else
        l2 = lmid;
    end
end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% MESH-INDEPENDENCY FILTER %%%%%%%%%%%%%%
function [dcn]=check(nelx,nely,rmin,x,dc)
dcn=zeros(nely,nelx);
for i = 1:nelx
    for j = 1:nely
        sum=0.0;
        for k = max(i-round(rmin),1):min(i+round(rmin),nelx)
            for l = max(j-round(rmin),1):min(j+round(rmin), nely)
                fac = rmin-sqrt((i-k)^2+(j-l)^2);
                sum = sum+max(0,fac);
                dcn(j,i) = dcn(j,i) + max(0,fac)*x(l,k)*dc(l,k);
            end
        end
        dcn(j,i) = dcn(j,i)/(x(j,i)*sum);
    end
end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FE-ANALYSIS %%%%%%%%%%%%%%
function [U]=FE(nelx,nely,x,penal)
[KE] = lk;
K = sparse(2*(nelx+1)*(nely+1), 2*(nelx+1)*(nely+1));
F = sparse(2*(nely+1)*(nelx+1),1);
U = sparse(2*(nely+1)*(nelx+1),1);
for ely = 1:nely
    for elx = 1:nelx
        n1 = (nely+1)*(elx-1)+ely;
        n2 = (nely+1)* elx +ely;
        edof = [2*n1-1; 2*n1; 2*n2-1; 2*n2; 2*n2+1; 2*n2+2; 2*n1+1; 2*n1+2];
        K(edof,edof) = K(edof,edof) + x(ely,elx)^penal*KE;
    end
end
end
% DEFINE LOADSAND SUPPORTS(HALF MBB-BEAM)

```

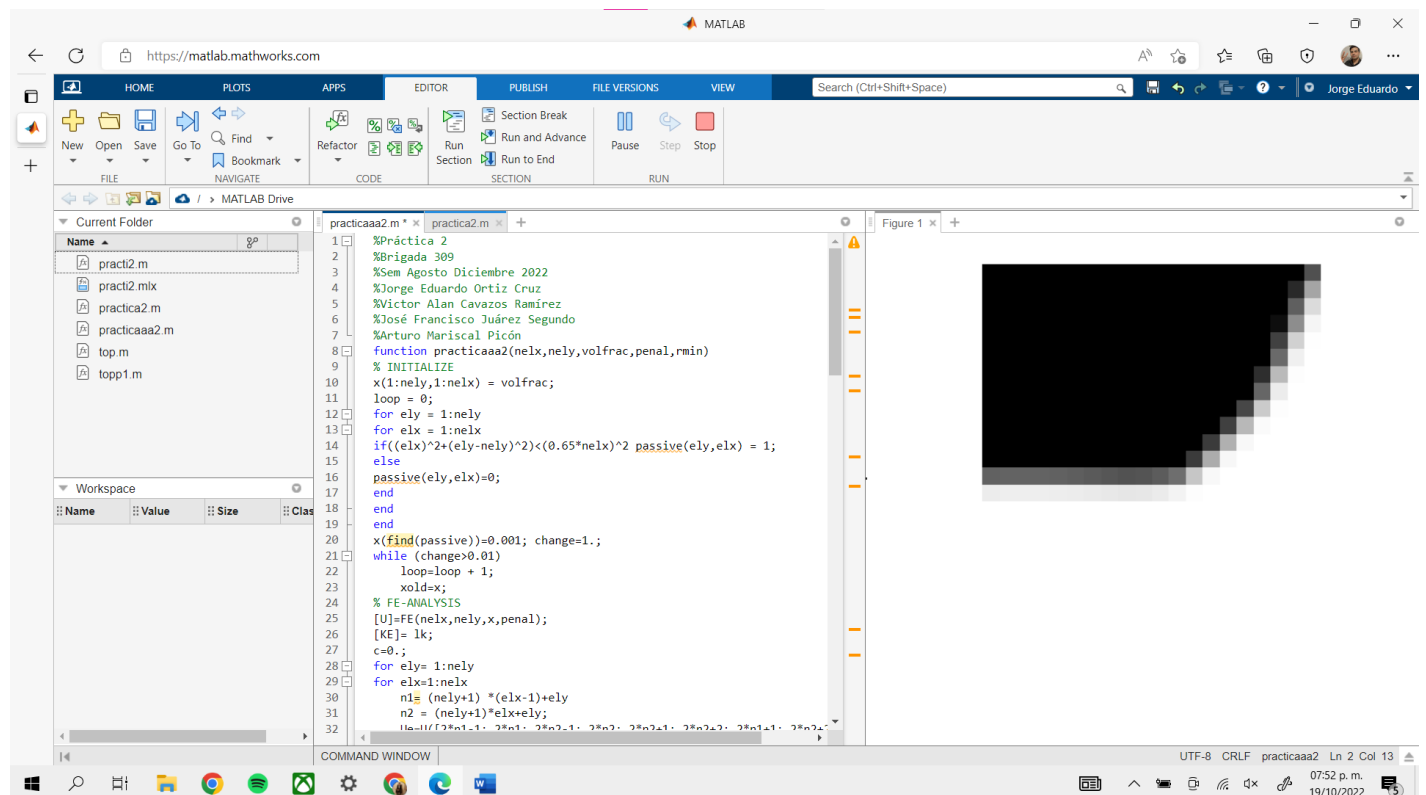


```

F(2,1) = -1;
fixeddofs = 2*nelx*(nely+1)+1:2*(nelx+1)*(nely+1);
alldofs = [1:2*(nely+1)*(nelx+1)];
freedofs = setdiff(alldofs,fixeddofs);
% SOLVING
U(freedofs,:) = K(freedofs,freedofs) \F(freedofs,:);
U(fixeddofs,:)= 0;
%%%%%%%%% ELEMENT STIFFNESS MATRIX %%%%%%%%%%
function [KE]=lk
E = 1.;
nu = 0.3;
k=[ 1/2-nu/6 1/8+nu/8 -1/4-nu/12 -1/8+3*nu/8 ...
-1/4+nu/12 -1/8-nu/8 nu/6 1/8-3*nu/8];
KE = E/(1-nu^2)*[ k(1) k(2) k(3) k(4) k(5) k(6) k(7) k(8)
k(2) k(1) k(8) k(7) k(6) k(5) k(4) k(3)
k(3) k(8) k(1) k(6) k(7) k(4) k(5) k(2)
k(4) k(7) k(6) k(1) k(8) k(3) k(2) k(5)
k(5) k(6) k(7) k(8) k(1) k(2) k(3) k(4)
k(6) k(5) k(4) k(3) k(2) k(1) k(8) k(7)
k(7) k(4) k(5) k(2) k(3) k(8) k(1) k(6)
k(8) k(3) k(2) k(5) k(4) k(7) k(6) k(1)];

```

Evidencia del código



Conclusiones:

Victor Alan Cavazos Ramírez

En nuestra primera práctica del Laboratorio de Biomecánica teníamos como objetivo conocer las distintas secciones que componen a un código de optimización topológica. Para poder lograr esto utilizamos el software llamado Matlab que fue de gran ayuda para poder realizar el código de la estructura que elegimos. Gracias a esta práctica pudimos ver cómo es importante conocer este tipo de códigos ya que nos ayudan a poder analizar una estructura y detectar que es necesario modificar o eliminar.

La optimización del diseño puede incrementar el valor de un producto mejorando su rendimiento en su entorno operativo y reduciendo el coste de producción mediante la reducción de la cantidad de material utilizado para fabricarlo. Al utilizar la optimización, el ingeniero de diseño incrementará sus conocimientos acerca del comportamiento de su producto y mejorará el diseño.

Arturo Mariscal Picón

Al momento de analizar el marco de una bicicleta nos pudimos percatar que si se pueden lograr diversas mejoras de las cuales requieren propuestas de diseño y nosotros identificamos que el marco de la bicicleta de un hombre contra una mujer puede tener mejoras para el beneficio de la persona que la usa, sin embargo, con ayuda de matlab se logró simular este caso.

De la misma manera podemos comprender que el uso de MATLAB para esta aplicación es de bastante utilidad, ya que comprende en gran cantidad cálculo y la visualización mediante un código programable. Por ello, comprendemos que la herramienta MATLAB es bastante sencilla de usar y nos ayuda a comprender mejor los temas.

Jorge Eduardo Ortiz Cruz

Tras haber terminado la programación en MatLAB web, se logró ver el marco de bicicleta en distintas partes, cambiando los valores que dimos, y aunque se tuvieron algunos problemas por el uso en la web ya que no dejaba de sacar valores nuevos y no podíamos reiniciar el programa a menos de que se cerrara todo. También se mejoró la inclusión del estado del arte, agregando la tabla con el contenido que se ve de manera más resumida.

Podemos ver cómo Matlab y la función de optimización nos ayudan para el análisis como lo habíamos visto desde la primera práctica, pero en este caso aplicado en el marco de la bicicleta por lo que esta función es muy útil y tiene muchas más aplicaciones de las que hemos visto hasta ahora

José Francisco Juárez Segundo

En base a los realizado en la práctica número 2 del laboratorio de biomecánica, hemos podido relacionar la investigación realizada sobre la optimización topológica, un tema visto en la materia clase, en un programa para las distintas mejoras que se plantea.

Por ello podemos concluir que se ha realizado de manera correcta dicha práctica en base a los conocimientos adquiridos y plasmados.

Referencias

- <https://store.pedalrevolution.co.uk/5859/products/van-nicholas-tuareg-650b-mtb-mountain-bike-frame.aspx>
- Fundación ESTEYCO, (2003), Pilar Carrizosa, La ingeniería de la bicicleta, Editado en Madrid.
- https://www.solidworks.es/sw/docs/SIM_Optimization_WP_ESP.pdf