

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

Laboratorio de Biomecánica

Práctica 2. Diseño del Marco de una Bicicleta

Nombre:	Matricula:	Carrera:
Campos Miranda Julio Emilio	1917048	IMTC
Cruz Puente Sergio Eduardo	1903115	IMTC
Cynthia Belen Guerrero Pardo	1026215	IMTC
Jorge Luis Salinas Garza	1916367	IMTC
Acuña Rodríguez Christopher Russ	1894698	IMTC

Maestro: Ing Isaac Estrada

Grupo: 309

Hora: N5

Cd. Universitaria, San Nicolás de los Garza.

19 de Octubre de 2022

Práctica 2. Diseño del Marco de una Bicicleta

Objetivo

El estudiante mismo estudiará la geometría del cuadro y los parámetros que lo conforman, para realizar el diseño del cuadro de una bicicleta de montaña, conociendo los materiales comúnmente empleados por la industria en la fabricación y utilizando el software de Matlab.

Nombre y definición de la forma GEOMETRÍA

La geometría de un cuadro de bicicleta seguramente te importe bien poco, y cuando compras un CUADRO o una BICICLETA COMPLETA, tu ESCALA DE VALORES será por este orden:

- ❖ Peso del cuadro.
- ❖ Material del cuadro.

MAL.

- ❖ Componentes con los que viene montada.
- ❖ Peso del cuadro.
- ❖ Material del cuadro

La GEOMETRÍA DE UN CUADRO ha ido asociada hasta hace bien poco al concepto de ANTROPOMETRÍA, o lo que es lo mismo: no pasábamos de buscar una adaptación del cuadro a las medidas corporales del ciclista.

Y esto se completaría después con una adaptación a nivel BIOMECÁNICO, normalmente a manos de un especialista, donde se jugaría con las variables de altura y posición de sillín, largo de bielas, altura y posición de manillar, ancho de manillar.

Los parámetros STACK y REACH son los que utilizaremos para saber cuál es nuestra TALLA.

El resto de los parámetros los utilizaremos para saber cómo se COMPORTA ese cuadro incluso antes de habernos montado.

¿Por qué Stack y Reach? Hasta hace bien poco todos los cuadros tenían lo que ahora se denomina GEOMETRÍA TRADICIONAL.

Pero con la aparición de GEOMETRÍA SLOPING y diferentes diseños según modalidad (XC, Enduro, DH, etc), se impuso la necesidad de medir la ALTURA y la LONGITUD o alcance de los cuadros, de una manera independientes a otras medidas y ángulos que podrían afectarlas.

EL STACK

Es el punto más alto de un cuadro de bicicleta. Esta medida es la distancia VERTICAL entre el centro de la pipa de dirección y el centro del eje del pedalier. Siendo una medida más precisa que las ofrecidas antiguamente: de centro de caja de pedalier a unión de tubo sillín con tubo horizontal.

EL REACH

Es el alcance de un cuadro de bicicleta. Esta medida es la distancia HORIZONTAL entre el centro de la pipa de dirección y el eje del pedalier.

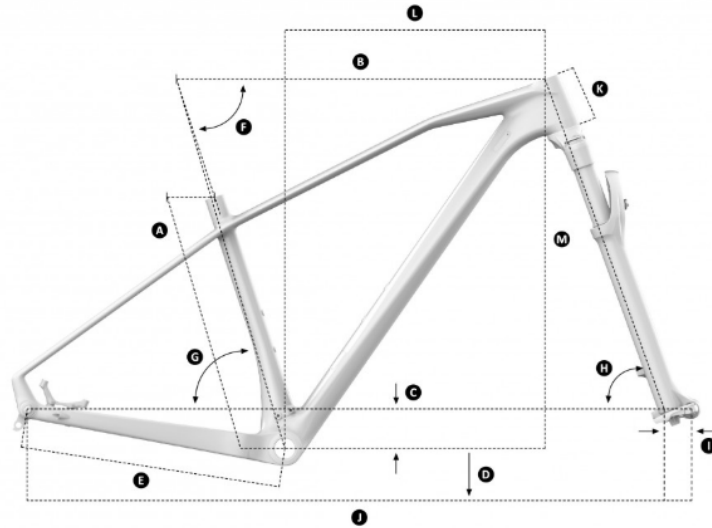
Mucho más precisa que la ofrecida antiguamente: de centro de la pipa de dirección al centro del tubo del sillín.

Estado del Arte

Es un vehículo de dos ruedas de igual diámetro y dispuestas en línea, propulsado por la fuerza del propio conductor. En este caso, la fuerza del conductor son las propias piernas, ya que estas son las que ponen en movimiento los pedales que están conectados a la cadena y las ruedas. Las bicicletas se caracterizan principalmente por su ausencia de motor, por el funcionamiento vía pedales, y por ser un vehículo ecológico (usa energía renovable que es la fuerza del conductor).

El desplazamiento se obtiene al girar con las piernas la caja de los pedales que a través de una cadena hace girar un piñón que a su vez hace girar la rueda trasera sobre el pavimento.

Propuesta de diseño de la geometría, alcances y limitaciones



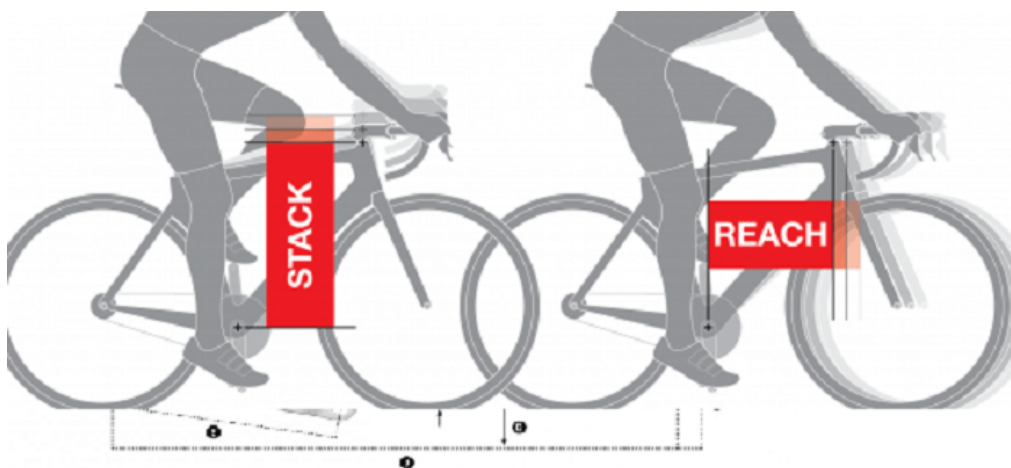
	FRAME SIZE	S	M	L	XL
A	Longitud tubo sillín	395mm	430mm	470mm	520mm
B	Longitud tubo superior	590mm	610mm	630mm	650mm
C	Caída eje pedalier	-65mm	-65mm	-65mm	-65mm
D	Altura eje pedalier	305mm	305mm	305mm	305mm
E	Longitud vainas	430mm	430mm	430mm	430mm
F	Ángulo tubo sillín real	72°	72°	72°	72°
G	Ángulo tubo sillín efectivo	73.5°	73.5°	73.5°	73.5°
H	Ángulo dirección	70°	70°	70°	70°
I	Fork Offset	46mm	46mm	46mm	46mm
J	Distancia entre ejes	1081mm	1101mm	1122mm	1143mm
K	Longitud pipa	85mm	90mm	100mm	115mm
L	Reach	411mm	430mm	447mm	463mm
M	Stack	605mm	610mm	619mm	632mm

Los parámetros STACK y REACH son los que utilizaremos para saber cuál es nuestra TALLA.

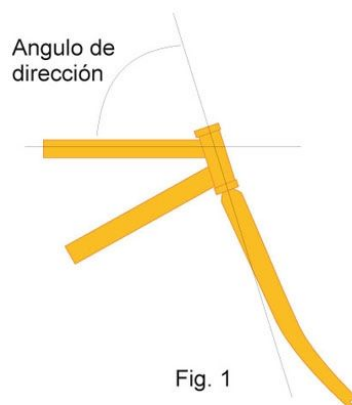
El resto de los parámetros los utilizaremos para saber cómo se COMPORTA ese cuadro incluso antes de habernos montado.

¿Por qué Stack y Reach? Hasta hace bien poco todos los cuadros tenían lo que ahora se denomina GEOMETRÍA TRADICIONAL.

Pero con la aparición de GEOMETRÍA SLOPING y diferentes diseños según modalidad (XC, Enduro, DH, etc), se impuso la necesidad de medir la ALTURA y la LONGITUD o alcance de los cuadros, de una manera independiente a otras medidas y ángulos que podrían afectarlas.

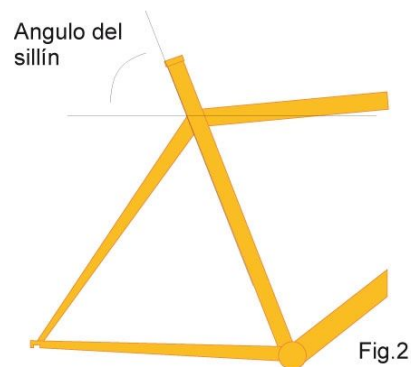


ÁNGULO DE DIRECCIÓN



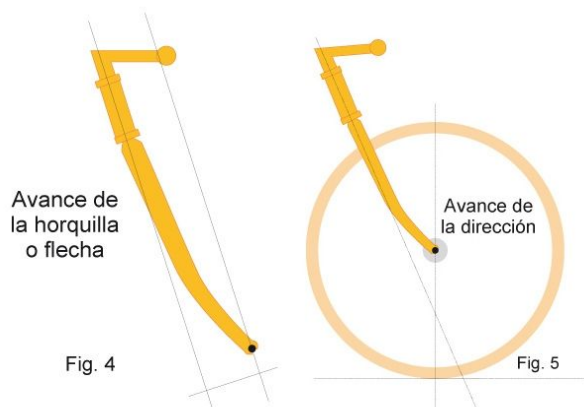
Este es el que se forma entre la pipa de la dirección y el suelo. Siendo un ángulo habitual 73° , podemos apreciar que cuanto más se acerca a 90° más nerviosa es la dirección. Y cuanto más nos alejamos de 90° , encontramos una dirección más dócil y tenderá a corregir sola la trayectoria.

ÁNGULO DE SILLÍN



Es el formado entre suelo y tubo del sillín. Siendo un ángulo habitual 72° , podemos apreciar que cuanto más se acerca a 90° estaremos sentados más lejos del eje de la rueda trasera = menos tracción. Por el contrario, cuanto más se aleja de 90° tendremos mayor tracción.

AVANCE DE HORQUILLA | OFFSET



Es la distancia que separa el eje del tubo de dirección del eje de giro de la rueda delantera.

La medida más habitual es de 4,5 cm. El OFFSET sirve para reducir o aumentar el TRAIL de la horquilla.

En modalidades muy técnicas (All Mountain, Enduro, DH), donde el ángulo de dirección necesita ser muy relajado el mayor offset o avance de la horquilla ayuda a reducir un trail demasiado grande para conseguir un óptimo equilibrio entre maniobrabilidad y estabilidad de la bicicleta, mientras que en modalidades más de pedaleo (XC, Maratón), con una dirección más vertical un menor offset aumenta el trail para lograr el mismo equilibrio.

AVANCE DE DIRECCION | TRAIL

De esta medida dependerá la capacidad de “autodireccionarse” de una bicicleta: a mayor avance mejor mantendremos la línea recta, especialmente a gran velocidad, y a menor avance maniobraremos pero, especialmente a poca velocidad.

Pasos del desarrollo de la programación

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

```
F(2,1)=1;  
fixeddofs = 2 * nelx * (nely + 1) + 1:2 * (nelx + 1) * (nely + 1);
```

Si nosotros utilizamos el código de optimización tal y como lo usamos en la práctica 1, no obtendremos el resultado deseado, debido a que el resultado final presenta inconvenientes en su diseño. Por lo tanto es necesario añadir las siguientes líneas de código:

```
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;
```

Código de optimización adaptado

```
function top(nelx,nely,volfrac,penal,rmin);
x(1:nely,1:nelx) = volfrac;
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;
loop = 0;
change = 1.;
while change > 0.01
loop = loop + 1;
xold = x;
[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*n1+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
[dc] = check(nelx,nely,rmin,x,dc);
[x] = OC(nelx,nely,x,volfrac,dc);
```



```

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 )])
colormap(gray); imagesc(-x); axis equal; axis tight; axis
off;pause(1e-6);
end
function [xnew]=OC(nelx,nely,x,volfrac,dc)
l1 = 0; l2 = 100000; move = 0.2;
while ((l2-l1)/l2 > 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
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-floor(rmin),1):min(i+floor(rmin),nelx)
for l = max(j-floor(rmin),1):min(j+floor(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

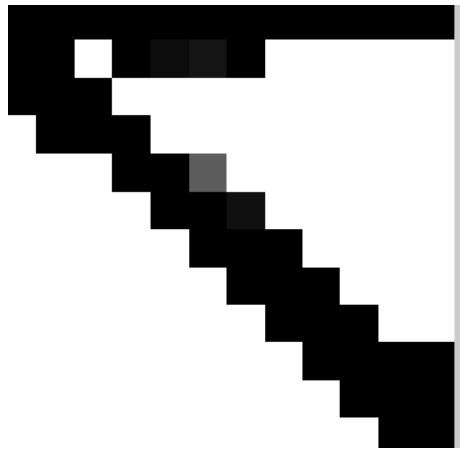
```

```

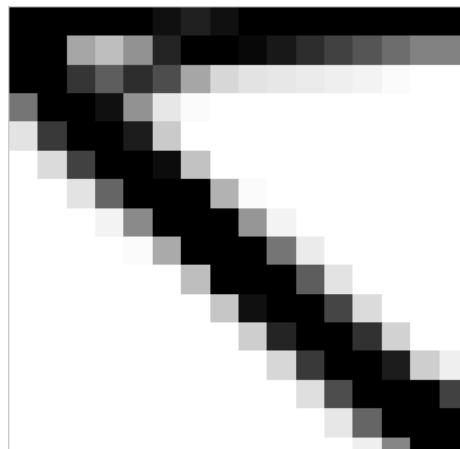
end
function [U]=FE(nelx,nely,x,penal)
[KE] = lk;
K = sparse(2*(nelx+1)*(nely+1), 2*(nelx+1)*(nely+1));
F = sparse(2*(nely+1)*(nelx+1),1); U = zeros(2*(nely+1)*(nelx+1),1);
for elx = 1:nelx
for ely = 1:nely
n1 = (nely+1)*(elx-1)+ely;
n2 = (nely+1)* elx +ely;
edof = [2*n1-1; 2*n1; 2*n2-1; 2*n2; 2*n2+1; 2*n2+2; 2*n1+1; 2*n1+2];
K(edof,edof) = K(edof,edof) + x(ely,elx)^penal*KE;
end
end
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);
U(freedofs,:) = K(freedofs,freedofs) \ F(freedofs,:);
U(fixeddofs,:)= 0;
function [KE]=lk
E= 2 *10^11.;
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)];

```

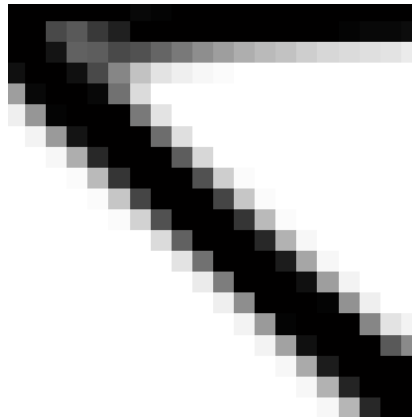
Resultados de la optimización



`top(12,12,0.33,3.0,0.9);`



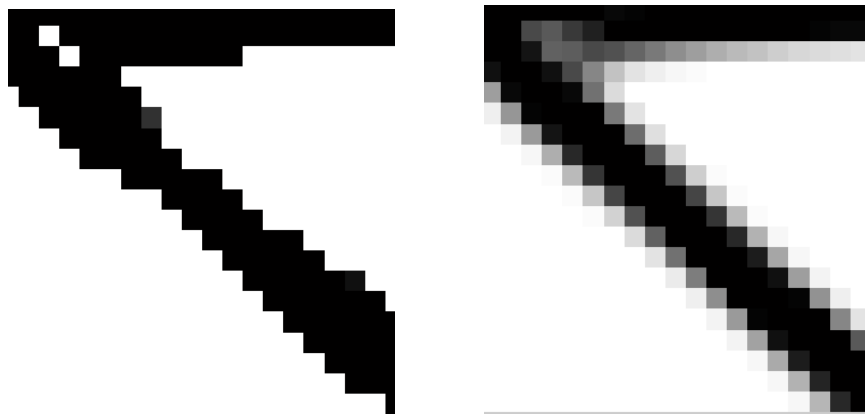
`top(16,16,0.33,3.0,1.2);`



```
top(20,20,0.33,3.0,1.5);
```

Con estos últimos parámetros usados, el resultado ya se asemeja más al marco de una bicicleta. Podemos realizar la comparación acerca de cómo funciona el filtro de mallado. En la última figura, el resultado tiene el filtro desactivado, ahora veremos como se ve con el filtro activado y desactivado.

Para ello comentaremos la línea 30 de nuestro código.



```
top(20,20,0.33,3.0,1.5);
```

Conclusiones

Sergio Eduardo Cruz Puente 1903115

Gracias a la elaboración de esta segunda práctica ahora es posible conocer la geometría del cuadro y los parámetros que conforman a este, así como también se realizó el diseño del cuadro de una bicicleta de montaña y se implementaron los materiales comúnmente empleados por la industria en la fabricación, todo esto se elaboró utilizando el software de Matlab, donde se implementó el código utilizado en la práctica anterior, así como el de SolidWorks, para la elaboración del modelado del marco de la bicicleta y así tener una mejor visión de este. Se retomaron el tema de la optimización topológica que se vio anteriormente en la práctica anterior, esto debido a que es muy eficiente en cuanto a la reducción de costos en el mecanizado y material de la pieza.

Julio Emilio Campos Miranda 1917048

Con la elaboración de este reporte, practicamos la elaboración de piezas en softwares de diseño, en este caso SOLIDWORKS. Utilizamos diversas funciones que proporciona el programa. La más sobresaliente en esta práctica fue la que se aplicó para obtener los esfuerzos de nuestra pieza, así como las propiedades físicas según el material elegido. Asimismo, aprendimos acerca de los conceptos que deben ser tomados en cuenta para construir de forma física una bicicleta.

Jorge Luis Salinas Garza 1916367

En la elaboración de esta práctica consistió en la realización de un marco de bicicleta, donde se realizó un análisis en base a la programación de la misma práctica anterior y tomando como referencia la práctica pasada solo tuvimos que modificar ciertos parámetros, en la cual cumplirían con el objeto a realizar. En el código de programación topológica tuvimos que colocar un valor de carga y un valor de fuerzas dentro del mismo diseño para que saliera el marco de bicicleta.

Cynthia Belen Guerrero Pardo 1921615

Se realizó esta práctica a través del Software de SOLIDWORKS, en esta, analizamos el diseño de una bicicleta de montaña y a través de los conocimientos que generamos alrededor de la anterior práctica, con esto podemos concluir que se logró el objetivo que teníamos

planeado ya que pudimos aprender sobre sus distintas características y cómo poder generarlo y comprenderlo.

Christopher Russ Acuña Rodríguez 1894698

Para finalizar la práctica, se pudo generar un diseño eficiente de manera satisfactoria. El objetivo de crear un marco de bicicleta el cuál sea optimizado por el diseño generativo fue el punto de partida para esta práctica, y al concluir con esta se logró el objetivo. Se usó el código, otorgado por el docente en la práctica pasada, de Matlab en donde se consiguieron estos resultados.