

Laboratorio de biomecánica

Práctica #1: optimización topológica

Nombre	Matricula	carrera
Victor Hugo Puente Alvarez	1929757	IMTC
Alejandro Hernandez Navarro	1923272	IMTC
Angela Rodriguez Flores	1896624	IMTC
Diego Ávila González	1853396	IMTC
Osiris Acosta Cisneros	1992234	IMTC

Día y hora: martes V1

Brigada: 204

Docente: Isaac Estrada

Fecha de entrega: 06 de septiembre de 2022

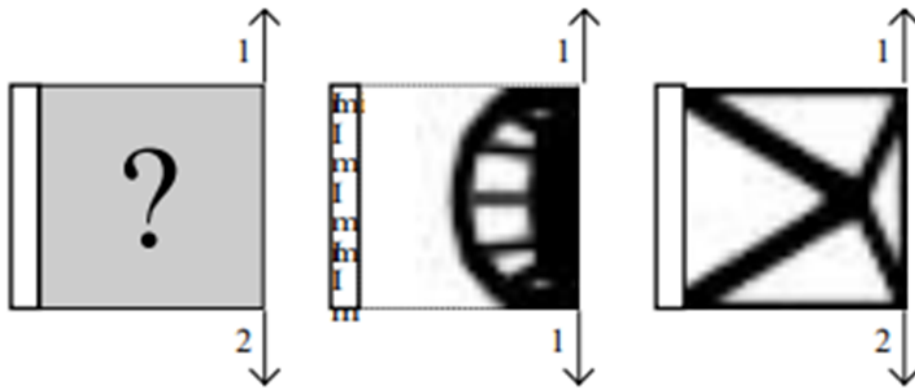
Cd. Universitaria, San Nicolás de los Garza, Nuevo León

1- Nombre y definición de la programación, mencionar un ejemplo de la geometría

La programación para mencionar dentro de esta propuesta se conoce como Optimización de Topología para la minimización del cumplimiento de estructuras cargadas estáticamente. Tiene como principal objetivo la educación en ingeniería para el campo de la optimización topológica y de misma manera cuenta con múltiples usos como podrían ser la realización de extensiones como casos de carga múltiple, esquemas alternativos de independencia de fracción para que luego se muestre la topología óptima correcta para su comparación.

El enfoque del programa está basado en la homogeneización ya que de esta manera se pueden proporcionar límites en el rendimiento teórico de las estructuras.

Ejemplo de geometría optimizada con este código



Optimización de la topología de una viga en voladizo con dos casos de carga. Izquierda: dominio de diseño, centro: viga con topología optimizada utilizando un caso de carga y derecha: viga con topología optimizada utilizando casos de carga

2) Estado del arte

La optimización topológica

La optimización topológica (OT) es una herramienta matemática que le permite al diseñador sintetizar topologías óptimas. En ingeniería mecánica se entiende como topología óptima a una pieza o parte mecánica diseñada especialmente para maximizar o minimizar alguna característica deseada.

Es una técnica englobada dentro del campo del análisis estructural. Se basa en el análisis mecánico de un componente o estructura, su objetivo es el aligeramiento estructural manteniendo las funcionalidades mecánicas del componente objetivo.

La optimización topológica es una técnica que soluciona numéricamente el problema, por medio de algoritmos iterativos que maximizan o minimizan una función objetivo (o de costo).

En estructuras continuas.

Cuando se considera la optimización topológica a partir de un medio continuo, lo que se busca es la óptima configuración de la estructura, donde el dominio se discretiza en elementos finitos que representan divisiones del material. La optimización topológica de estructura continua será abordada en este trabajo.

En estructuras discretas.

La optimización topológica a partir de un dominio discreto se divide en dos categorías, optimización topológica de estructura de malla continua y optimización topológica de estructuras discretas. Donde la optimización topológica de estructura de malla continua, considera un dominio continuo de la estructura discretizando por un número infinito barras rígidas distanciadas por un espacio infinitesimal, cuya solución óptima es obtenida analíticamente por la teoría de la elasticidad.

Proceso

Se deben de tener en cuenta varios aspectos; el espacio de diseño, el o los casos de carga que va a sufrir la pieza en cuestión, el material y la tecnología con que se va a realizar su fabricación, la reducción de costes mediante la minimización de soportes y aprovechamiento de la cuba de impresión, en caso de utilizar tecnologías aditivas, y muchos más.

Pasos.

1. Dibujar o Importar geometría
2. Simplificar la pieza y definir el espacio de diseño
3. Establecer uniones, juntas y contactos
4. Asignar materiales
5. Definir los casos de carga
6. Generar la optimización
7. Refinar la geometría
8. Exportar a CAD o generar STL
9. Verificar el rendimiento
10. Fabricar

La optimización topológica se realiza sobre un modelo existente, específicamente en la parte final de la etapa de diseño, con la finalidad de hacer más ligero el objeto. Primeramente, se establece el espacio de diseño mínimo necesario para la pieza, después se definen las cargas externas, las condiciones fronteras, restricciones y propiedades del material, además de establecer las áreas de exclusión o ubicaciones fijas. Posteriormente el análisis del elemento finito divide el espacio de diseño en áreas más pequeñas, la optimización topológica crea una malla básica de este espacio de diseño más pequeño utilizando elementos finitos. Después el análisis del elemento finito evalúa la distribución de tensión de la malla y la energía de deformación para encontrar la carga o tensión óptima que cada elemento puede manejar. Luego el programa de optimización topológica presiona digitalmente el diseño desde varios ángulos, evalúa su integridad estructural y encuentra el material innecesario a eliminar. El software prueba la rigidez, la tensión y la desviación de cada elemento finito con respecto al requisito definido para encontrar material redundante. Por último, el análisis de elementos finitos une las piezas para completar el patrón.

La optimización topológica tiene la ventaja de que elimina el peso del material y lo hace más ligero y a la vez aumenta la rigidez. El hecho de eliminar material innecesario hace que se reduzca la materia prima necesaria y por lo tanto, será sostenible para el planeta. Entre las desventajas que tiene la optimización topológica es que el precio de producción es más caro dado que algunos diseños

con la optimización topológica se vuelven más complejos de realizar a tal punto que no se pueden utilizar métodos de fabricación tradicionales.

La optimización topológica se utilizan ultimadamente debido a que muchas industrias buscan métodos de diseño avanzado como la optimización topológica y el diseño generativo, entre estas industrias están la aeroespacial, automotriz y la médica, en cuanto a la industria aeroespacial se busca reducir la masa de las aeronaves pero sin perder la fuerza puesto que los viajes aéreos son costosos, en la industria automotriz se busca reducir la masa de las piezas para aumentar la economía de combustible, potencia, estabilidad y la fuerza de un automóvil, por último, en el área médica se busca crear implantes y prótesis eficientes, con esto se pueden tener partes del cuerpo artificiales ligeras pero que a su vez sean muy resistentes y puedan aplicar cierto nivel de fuerza.

3) Procedimiento de la programación

De la línea 65 a 99 se escribe el código de elementos finitos.

De la línea 70-77 la matriz de rigidez global está formada por un bucle sobre todos los elementos

Cada nodo tiene dos grados de libertad (horizontal y vertical), por lo que el comando $F(2,1)=-1$. (línea 79) se aplica una fuerza de fuerza unitaria vertical en la esquina superior izquierda.

Para la línea 82 es más fácil definir los grados de libertad que son fijos (fixeddofs) a partir de entonces el freedofs se encuentran automáticamente usando el operador de Matlab setdiff que encuentra los grados libres de libertad como la diferencia entre todos los grados de libertad y el fijo grados de libertad.

Matlab puede implementar los soportes eliminando los grados de libertad fijos de las ecuaciones lineales, esto en la línea 84 con: $U(\text{libertades,:})=K(\text{libertades, libertades})/F(\text{libertades,:})$;

La matriz de rigidez del elemento se calcula en las líneas 86–99.

El módulo de Young E y la relación nu de Poisson pueden modificarse en las líneas 88 y 89.

Para resolver el ejemplo de voladizo corto que se muestra en la figura, solo las líneas 79 y 80 deben cambiarse a:

- 79 $F(2*(nelx+1)*(nely+1),1) = -1$;
- 80 grados de libertad fijos = $[1:2*(nely+1)]$;

En el caso de dos casos de carga, los vectores de fuerza y desplazamiento deben definirse como vectores de dos columnas, lo que significa que la línea 69 se cambia a:

- 69 $F = \text{escaso}(2*(nely+1)*(nelx+1),2);$
- $U = \text{escaso}(2*(nely+1)*(nelx+1),2);$

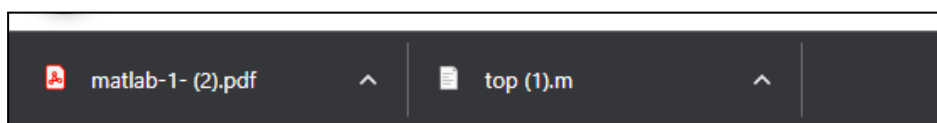
En algunos casos, se puede requerir que algunos de los elementos tomen el valor de densidad mínima (por ejemplo, un orificio para una tubería). Una matriz $nely \times nelx$ pasiva con ceros en los elementos libres para cambiar y unos en los elementos fijos para ser cero se puede definir en el programa principal y transferir a la subrutina OC (agregando pasiva a la llamada en las líneas 28 y 38).

4) Implementación o desarrollo de la programación en sus diferentes vistas

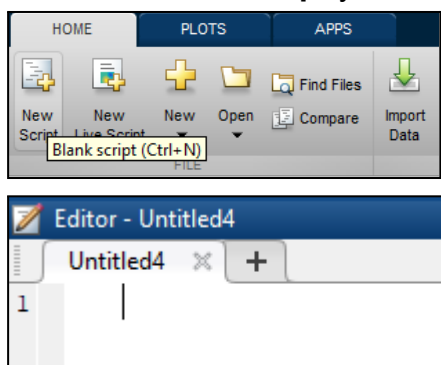
En esta práctica utilizaremos el código de las 99 líneas de optimización topológica que nos presentó el ingeniero para que lo descargamos y probemos, para ello tendremos que descargar el programa de MATLAB para poder correrlo y después poder ver una gráfica que representa una viga con cargas.

1. Primero tendremos que descargar el software de MATLAB ya sea donde se te sea más cómodo y accesible de descargar en tu computadora.
2. Después tendremos que ir al link que nos puso el ingeniero en las instrucciones de la práctica, donde se mostrará un link que al darle click de nos descargara, este es el código de 99 líneas que tenemos que abrir en MATLAB.

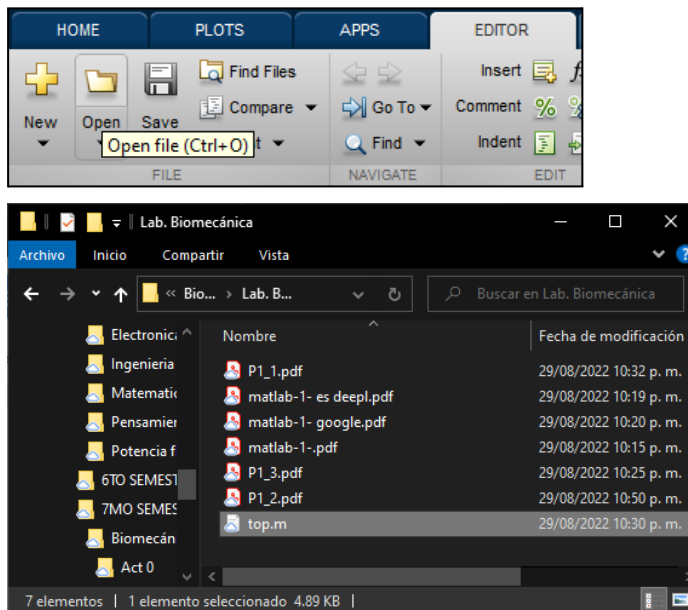
<https://www.topopt.mek.dtu.dk/apps-and-software/a-99-line-topology-optimization-code-written-in-matlab>



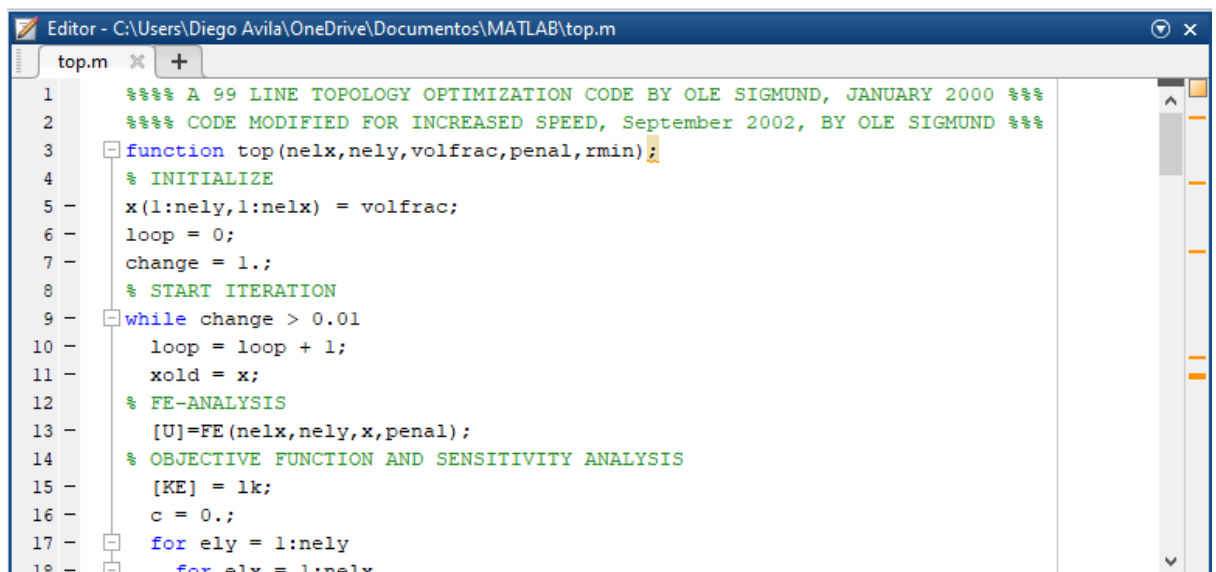
3. Seguido de esto Iniciamos el MATLAB y después nos vamos a la pestaña donde dice **New script** y se nos abrirá una nueva pestaña de editor.



4. Seguido debemos de ir a la pestaña que dice Open, y después nos dirigiremos a la carpeta donde descargamos y guardamos el código que descargamos, se encuentra por el nombre de “top.m”.



5. Se nos abrirán las 99 líneas en la pestaña del editor como se nos muestra en la siguiente imagen.



6. Después tenemos que pasar esas líneas hacia la ventana de comando de código y darle a run, después se nos mostrará una ventana de error ya que aún no hemos declarado las variables de top().

```

Command Window
New to MATLAB? See resources for Getting Started.
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)];
function top(nelx,nely,volfrac,penal,xmin);
↑
Error: Function definition not supported in this context. Create functions in code file.
fx >>

```

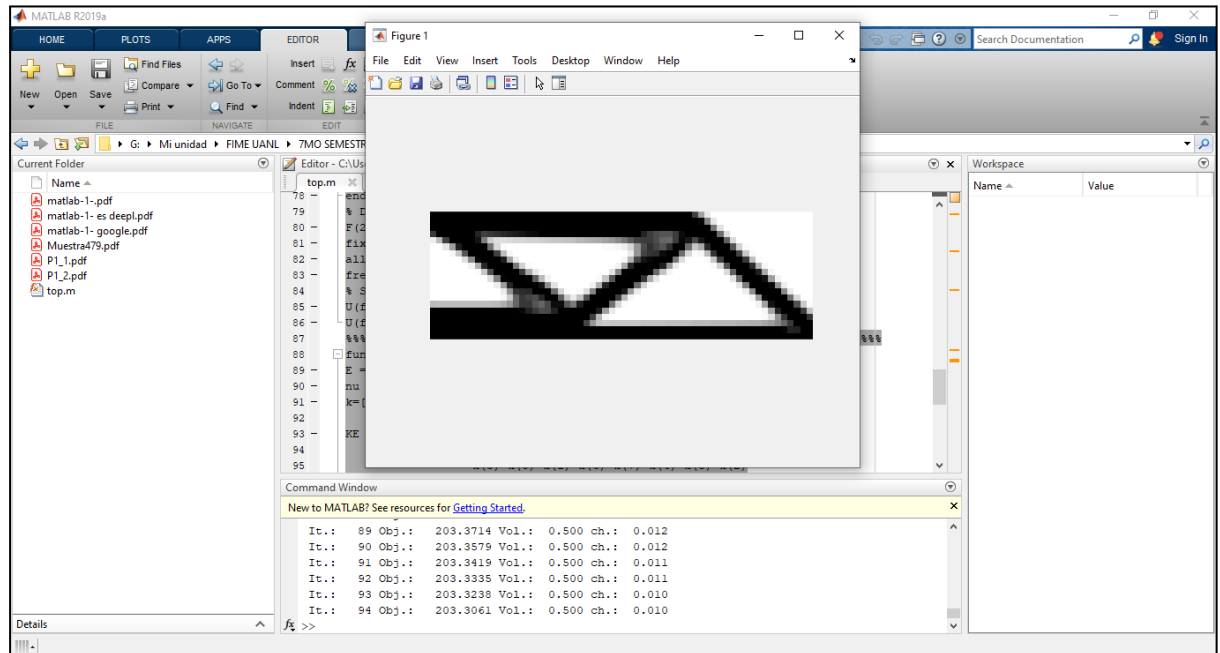
- Una vez esto hecho tenemos que poner el siguiente código en la ventana de comando que se nos muestra en el documento descargado de la práctica:
top(60,20,0.5,3.0,1.5)

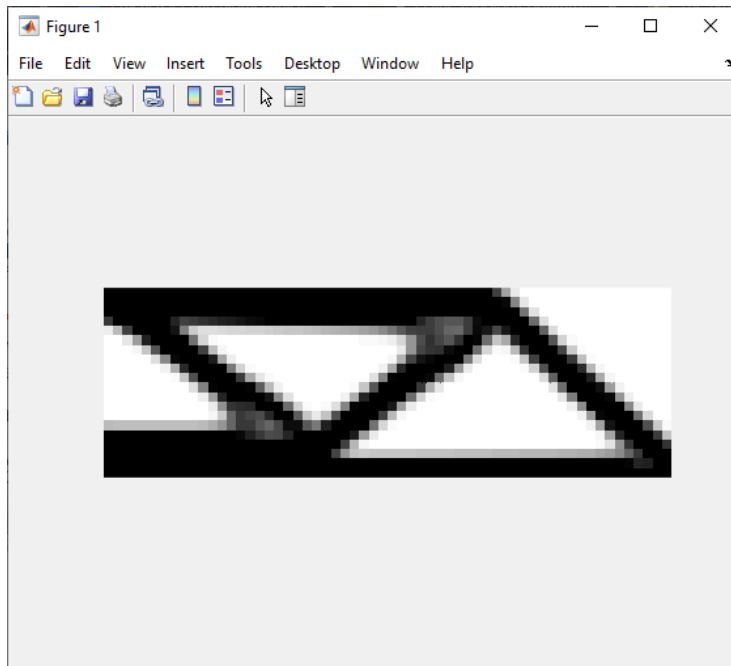
```

Command Window
New to MATLAB? See resources for Getting Started.
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)];
function top(nelx,nely,volfrac,penal,xmin);
↑
Error: Function definition not supported in this context. Create functions in this code file.
fx >> top(60,20,0.5,3.0,1.5)

```

- Después se nos abrirá una nueva ventana de MATLAB aparte con nombre de Figure 1 donde aquí se nos mostrará el diseño de la figura optimizada del diseño de cargas y sus restricciones propuestas.





Código de 99 líneas de MATLAB:

```

%% A 99 LINE TOPOLOGY OPTIMIZATION CODE BY OLE SIGMUND, JANUARY 2000
%% CODE MODIFIED FOR INCREASED SPEED, September 2002, BY OLE SIGMUND
function top(nelx,nely,volfrac,penal,rmin);
% INITIALIZE
x(1:nely,1:nelx) = volfrac;
loop = 0;
change = 1.;
% START ITERATION
while change > 0.01
    loop = loop + 1;
    xold = x;
% FE-ANALYSIS
    [U]=FE(nelx,nely,x,penal);
% OBJECTIVE FUNCTION AND SENSITIVITY ANALYSIS
    [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
% FILTERING OF SENSITIVITIES
    [dc] = check(nelx,nely,rmin,x,dc);
% DESIGN UPDATE BY THE OPTIMALITY CRITERIA METHOD
    [x] = OC(nelx,nely,x,volfrac,dc);
% 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)) ...

```

[illegible]

```

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)];

```

5) Conclusión

Alejandro Hernandez Navarro

Respecto al trabajo elaborado, se estuvo investigando lo que es la optimización topológica y analizando el código de 99 líneas de Matlab de optimización topológica, se aprendió más sobre este tema, el cómo esta etapa de diseño mejora en el diseño de un objeto aumentando la rigidez de este y a su vez disminuyendo su masa. Se comprendió el funcionamiento del código de optimización topológica, de cada una de las partes del este y como se ejecuta el análisis en general.

Diego Ávila González

En este trabajo aprendí que las operaciones y matemáticas que se aplican en el elemento finito se pueden realizar con el software de MATLAB y eso nos ayudará mucho al momento de querer hacer esos cálculos y nos será de gran ayuda en futuros proyectos que se nos presente y eso nos da entrada hacer a de como funcionan el uso de fuerzas y cargas en diferentes objetos naturales de la biomecánica.

Angela Rodriguez Flores

Para esta práctica primero se investigó sobre la optimización topológica y el código de las 99 líneas, luego se explicó a detalle cómo se usan las 99 líneas, como se pueden cambiar, que va en cada una, etc. Finalmente se utilizó matlab para realizar para observar una forma diferente de uso a la que hemos visto en semestres anteriores, pudimos observar figuras y analizar el programa.

Osiris Acosta Cisneros

En esta práctica se presentó un algoritmo de optimización de topología en base a una programación de matemáticas. El código a pesar de solo contener 99 líneas de programación me pareció muy interesante y complejo. La ejecución del código fue demasiado lento, pero aún así retomé la programación en Matlab después de

mucho tiempo y pude aprender una implementación completamente diferente a las que había usado.

Victor Hugo Puente Alvarez

Por medio de esta actividad conocí más sobre el tema de optimización topológica y entendí que el objetivo de esta es minimizar la masa y maximizar la rigidez. Que tiene una energía de deformación que contribuye o no al mínimo y máximo de la variable del diseño de elementos finitos. que todo esto lo englobamos en el software de matlab al elaborar el código de 99 líneas que nos mostró una figura optimizada del diseño de cargas y sus restricciones propuestas y por medio de este ejemplo me ayudó a comprender más el tema y su importancia en la biomecánica.

Referencias bibliográficas

- MEZA VALENCIA, C., 2022. OPTIMIZACIÓN TOPOLÓGICA EN EL DISEÑO DE ELEMENTOS ESTRUCTURALES MECÁNICOS . [en línea] Red.uao.edu.co.

Disponible en:

<<https://red.uao.edu.co/bitstream/handle/10614/4209/TME01179.pdf;jsessionid=4E3F96D03B78AAF7225F0A83A5031330?sequence=1>>

- Anónimo. (s.f.). Introduction to topology optimization. septiembre 5, 2022, de Engineering product design Sitio web:

<https://engineeringproductdesign.com/knowledge-base/topology-optimization/>

- Velling, A. (2020). Topology optimization. septiembre 5, 2022, de Fractory Sitio web: https://fractory.com/topology-optimisation/#How_Does_It_Work