

UNIVERSIDAD AUTÓNOMA DE
NUEVO LEÓN
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

Laboratorio de Biomecánica
Práctica 1

INSTRUCTOR@: YADIRA MORENO VERA
MARTES V1 BRIGADA 204

Equipo #2:

Matricula	Nombre	Carrera
1991908	Covarrubias Becerril Brian Eduardo	IMTC
1991966	Luis Javier Rodríguez Vizcarra	IMTC
1925324	Rubén Cantú Espinoza	IMTC
1926098	Ángel Fernando Mexquitic Rodríguez	IMTC
1895460	Elmer Javier Delgadillo García	IMTC
1847932	Francisco Javier Velazco Rivas	IMTC

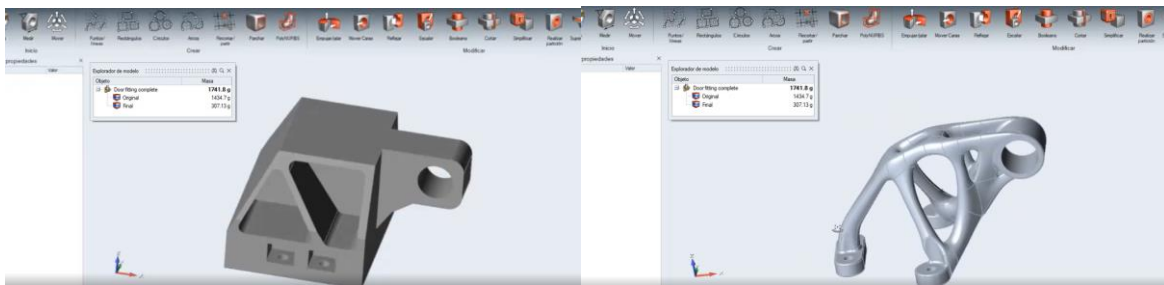
5 de septiembre del 2022, San Nicolás de los Garza, Nuevo León

Objetivo

El estudiante conocerá cada una de las secciones que integran el código de optimización topológica, como se debe crear el archivo (.m) en MATLAB y como se ejecuta el análisis.

Marco Teórico

La optimización topológica es una técnica que pertenece al análisis estructural, y consiste, básicamente, en analizar un componente o estructura y, en función de cómo se cargue, eliminar material ahí donde no es necesario. Por ejemplo:



La solución de este problema puede ser planteada utilizando dos estrategias: como un problema de optimización de forma o de optimización de la topología.

La optimización de forma consiste en modificar la geometría del dominio preservando su topología, es decir sin crear huecos o cavidades en su interior. Este tipo de análisis es usualmente conocido como análisis de sensibilidad al cambio de forma y sus bases matemáticas se encuentran bien establecidas. El principal inconveniente del análisis de sensibilidad al cambio de forma es que sólo permite cambios en la frontera del dominio, lo que limita su campo de aplicación.

Una manera más general de controlar un dominio es mediante modificaciones de su topología, lo que permite obtener la configuración deseada partiendo de una morfología inicial distante de la óptima. Los métodos de homogenización son posiblemente los más utilizados para la optimización topológica. Estos consisten en caracterizar la topología a través de su densidad, es decir, los huecos se identifican con regiones de densidad nula. De esta forma la solución del programa resulta en una distribución ficticia de material.

En el proceso de optimización topológica, se deben 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.

Para ello, en esta práctica se implementará un código de optimización topológica de 99 líneas en Matlab que divide en 36 líneas para la programación principal, 12 líneas para los criterios de optimización, 16 líneas para el filtro de mallado y 35 líneas para el código de elemento finito. De hecho, excluyendo las líneas de comentarios y líneas asociadas con la producción y el análisis de elementos finitos, el código resultante es de solo 49 líneas. Este código fue desarrollado por O. Sigmund, Department of Solid Mechanics, Building 404, Technical University of Denmark, DK-2800 Lyngby, Denmark. El código puede ser descargado desde la página del autor: <http://www.topopt.dtu.dk>.

MATLAB es una plataforma de programación y cálculo numérico utilizada por millones de ingenieros y científicos para analizar datos, desarrollar algoritmos y crear modelos. combina un entorno de escritorio perfeccionado para el análisis iterativo y los procesos de diseño con un lenguaje de programación que expresa las matemáticas de matrices y arrays directamente. Las toolboxes de MATLAB se desarrollan de forma profesional, pasan pruebas rigurosas y están totalmente documentadas.

Definición de la programación y ejemplo de la geometría

Como ya lo mencionamos anteriormente, la optimización topológica es una técnica englobada dentro del campo de análisis estructural, basada en el análisis mecánico de un componente, sistema o estructura. Tiene como objetivo el aligeramiento estructural manteniendo las características y propiedades mecánicas del componente principal. A diferencia de las demás optimizaciones, esta ofrece un nuevo concepto de diseño estructural enfocado a aquellas aplicaciones donde el peso del componente es crucial, como la industria aeroespacial.

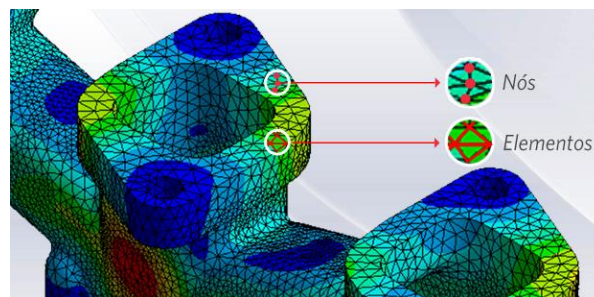
La optimización topológica, en resumen, consiste en utilizar un software concreto para “eliminar” el material que no posee los soportes. Algunos programas utilizados para realizar esta tarea son Within Labs, Inspire, Tosca, Ansys, Netfabb, etc.

Para comenzar a realizar una optimización de este tipo, primero debemos realizar el modelo 3D en la fase de boceto, donde colocaremos las distintas cargas o fuerzas sobre la pieza. Después, el mismo software será el encargado de simular y calcular todas las tensiones que son aplicadas sobre la pieza. Conociendo los resultados de la simulación, podemos realizar un recorte a la pieza para retirar las partes que no son sometidas a ningún tipo de tensión o fuerza. Una vez hecho esto, tendremos una nueva geometría que cumplirá con los requisitos mecánicos y de diseño que necesitamos.

Estado del arte

El artículo proporcionado por el instructor de laboratorio presenta una implementación de un código de implementación topológica para la minimización de la conformidad de estructuras cargadas estáticamente. El número total de líneas de entrada de Matlab es de 99, incluyendo el optimizador y la subrutina de elementos finitos. Las 99 líneas están divididas en 36 líneas para el programa principal, 12 líneas para el optimizador basado en criterios de optimización, 16 líneas para un filtro de dependencia de malla y, por último, 35 líneas para el código de elementos finitos.

Como paréntesis, el método de elementos finitos consiste en proponer que un número infinito de variables desconocidas, sean sustituidas por un número limitado de elementos de comportamiento bien definido. Esas divisiones pueden tener diferentes formas, tales como triangular, cuadrangular, entre otros, dependiendo del tipo y tamaño del problema. Como el número de elementos es limitado, son llamados de “elementos finitos” – palabra que da nombre al método. Los elementos finitos están conectados entre sí por puntos, que se llaman nodos o puntos nodales. Al conjunto de todos estos ítems – elementos y nodos – se lo denomina malla. Debido a las subdivisiones de la geometría, las ecuaciones matemáticas que rigen el comportamiento físico no se resolverán de una manera exacta, sino aproximada por este método numérico. La precisión de los Métodos dos Elementos Finitos depende de la cantidad de nodos y elementos, del tamaño y de los tipos de elementos de la malla. Por lo tanto, cuanto menor sea el tamaño y mayor el número de elementos en una malla, más precisos serán los resultados del análisis.

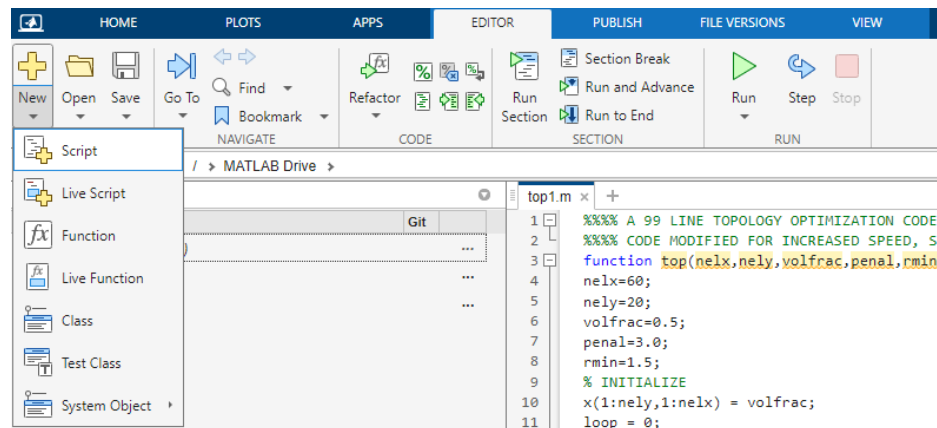


El código Matlab presentado en la página <https://www.topopt.mek.dtu.dk/apps-and-software/a-99-line-topology-optimization-code-written-in-matlab> está destinado a la educación en ingeniería. Los estudiantes y los recién llegados al campo de la optimización topológica pueden encontrar el código aquí y descargarlo. El código se puede utilizar en cursos de optimización estructural en los que se puede asignar a los estudiantes la realización de extensiones, como casos de carga múltiples, esquemas alternativos de independencia de malla, áreas pasivas, etc. Esta versión del código está optimizada con respecto a la velocidad en comparación con la versión original y fue lanzada el 28 de mayo de 2002.

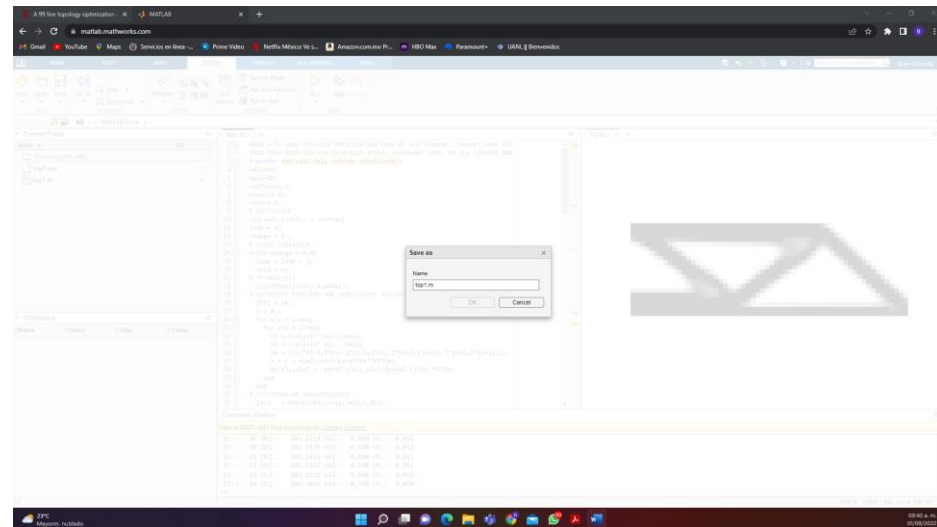
Procedimiento de la programación

Para simular en Matlab el código propuesto de optimización topológica, debemos llevar a cabo los siguientes pasos:

- 1) Abrir Matlab, ya sea la aplicación de escritorio o directamente desde la página oficial usando su repositorio online.
- 2) Crear un nuevo script, para ello seleccionamos la opción en la barra de herramientas.



- 3) Una vez hecho esto, solo tenemos que copiar el código proporcionado que se localiza en la pagina indicada por el instructor.
- 4) Ya con el código copiado, tenemos que guardar primero el documento, en este caso como se utilizó Matlab online, se guardará en la nube de la cuenta que esta iniciada.



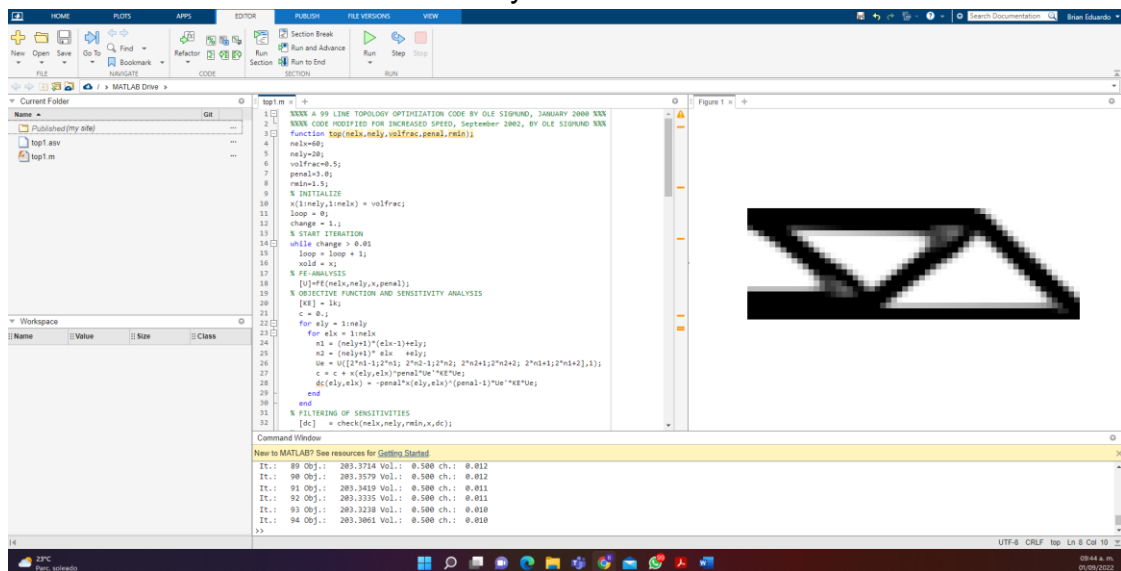
- 5) Antes de ejecutar la simulación, es necesario definir los valores de las variables de entrada (nelx, nely, volfrac, penal, rmin), ya sea sustituyendo directamente los valores dentro de la línea de código donde se define o asignándolos cada uno.

```

top1.m x +
1 %%%% A 99 LINE TOPOLOGY OPTIMIZATION CODE BY OLE SIGMUND, JANUARY 2000 %%%
2 %%%% CODE MODIFIED FOR INCREASED SPEED, September 2002, BY OLE SIGMUND %%%
3 function top(nelx,nely,volfrac,penal,rmin);
4 nelx=60;
5 nely=20;
6 volfrac=0.5;
7 penal=3.0;
8 rmin=1.5;
9 % INITIALIZE
10 x(1:nely,1:nelx) = volfrac;
11 loop = 0;

```

6) Procedemos a correr la simulación y observamos los resultados mostrados.



Implementación o desarrollo de la programación en sus diferentes vistas

El código implementado para la simulación del programa de optimización topológica es el siguiente:

```

%%%%% A 99 LINE TOPOLOGY OPTIMIZATION CODE BY OLE SIGMUND, JANUARY 2000 %%%
%%%%% CODE MODIFIED FOR INCREASED SPEED, September 2002, BY OLE SIGMUND %%%
function top1(nelx,nely,volfrac,penal,rmin);
nelx=60;
nely=20;
volfrac=0.5;
penal=3.0;
rmin=1.5;
% INITIALIZE
x(1:nely,1:nelx) = volfrac;
loop = 0;
change = 1.;
% START ITERATION
while change > 0.01
    loop = loop + 1;
    xold = x;
% FE-ANALYSIS

```

[illegible]

```

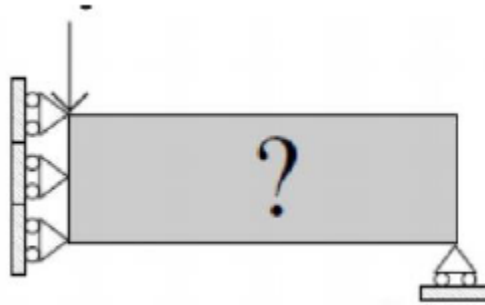
[KE] = lk;
K = sparse(2*(nelx+1)*(nely+1), 2*(nelx+1)*(nely+1));
F = sparse(2*(nely+1)*(nelx+1),1); U = zeros(2*(nely+1)*(nelx+1),1);
for elx = 1:nelx
    for ely = 1:nely
        n1 = (nely+1)*(elx-1)+ely;
        n2 = (nely+1)* elx +ely;
        edof = [2*n1-1; 2*n1; 2*n2-1; 2*n2; 2*n2+1; 2*n2+2; 2*n1+1;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

```

Es importante definir cada una de las variables de entrada, que son las que definimos al principio asignándole los valores de: top(60,20,0.5,3.0,1.5).

- Nelx y nely son el número de elementos en las direcciones horizontales y verticales.
- Volfrac es la fracción de volumen.
- Penal es el poder de penalización.
- Rmi es el tamaño del filtro.

El dominio de diseño, carga y restricciones propuestas fue el siguiente:



Como notamos en la simulación, una vez simulado el código, la imagen del diseño optimizado fue la siguiente:



Conclusiones

- Covarrubias Becerril Brian Eduardo: en esta práctica pudimos simular y observar una optimización topológica mediante un código de 99 líneas implementado en el software de Matlab. Para ello fue necesario realizar primero una investigación para entrar en contexto y conocer cuál es el objetivo de este método y cómo funciona. Una vez realizado esto, pudimos observar en la simulación la optimización sobre un sistema estructural sometido a una carga y así comprobar su funcionamiento. Observamos que, pese a que se simplificó la estructura, se conservaron las propiedades del material. Esto es básicamente el objetivo de la optimización topológica, eliminar el material restante para aligerar una pieza o simplificar una estructura conservando las características y propiedades de este.

- Delgadillo Garcia Elmer Javier: Es de vital importancia tener un conocimiento teórico como práctico para la implementación de un código optimizado topológicamente en un sistema de cómputo numérico, en este caso MATLAB. Ya que manejar la técnica de optimización topológica permite una serie de ventajas que permiten solucionar numéricamente el problema, por medio de algoritmos iterativos que maximizan o minimizan una función objetivo. De manera práctica, esta técnica permite diseñar estructuras óptimas, con el menor peso posible para un determinado estado de carga y una determinada restricción de volumen.
- Luis Javier Rodriguez Vizcarra: En conclusión, esta práctica me sirvió para conocer el concepto de optimización topológica, sus aplicaciones y por lo tanto su propósito, lo que se hizo fue hacer un ejemplo por medio de un código con el cual se forman diferentes piezas, estas cambian modificando el código principal, el principal objetivo fue al hacer varias pruebas ver como se distribuye el material y que cumpla con las condiciones que se exigen y tenga un aligeramiento la estructura pero que no pierda su funcionalidad mecánica, todo esto fue observado en Matlab. también hablando de Matlab me sirvió esta practica para aprender de la programación con optimización topológica y observar como se comporta el material.
- Francisco Javier Velazco Rivas: Para este reporte se tuvo que evaluar una optimización de unas vigas visto de una manera técnica con el objetivo de mejorarlas. Para el código de 99 líneas se evaluó primero con videos de YouTube como apoyo ya que no tenia idea de como se hacia ni que significaba, al tener un poco más de conocimiento además de recordar un poco la plataforma esto se hace de manera mucho más sencilla.

Al tener que ver opciones o cómo podríamos mejorar la estructura podríamos pensar en cambiar de material pero podríamos hacer algo tan sencillo como eliminar el material que en este caso es lo que se buscaba además de cambiar el diseño podría ser una opción ser algo muy abrupto o imposible.

- Ángel Fernando Mexquitic Rodriguez: es interesante el concepto de optimización topológico y como funciona en el análisis estructural y como afecta en el funcionamiento de una pieza. Y con la ayuda de Matlab, ver como es ese funcionamiento de igual manera es bastante explicativo para entender de mejor manera el concepto, presentando así varias ventajas a la hora de aplicarlo en la pieza para su análisis.

- Rubén Cantú Espinoza: en esta primera práctica se investigó sobre lo que viene siendo la optimización topológica, la cual consiste en utilizar un software en específico para analizar la estructura de una pieza para posteriormente eliminar el material que no sea necesario. En este proceso se deben de tener en cuenta algunos aspectos, como el espacio del diseño, el material, los costos de impresión, la carga que se le aplicará a la pieza, entre muchos otros aspectos. Como se mencionó anteriormente, se requiere utilizar un software en específico, en el caso de esta práctica se propuso utilizar MATLAB, donde se implementó un código el cuál se reparte en líneas de programación principal, líneas para los criterios de optimización, líneas para el filtro de mallado y líneas para el código de elemento finito; todo esto con el objetivo de simplificar la pieza, sin que cambien sus propiedades y que cumpla con la misma función.

Referencias

- 99 Line Topology Optimization Code – O. Sigmund, Department of Solid Mechanics, Building 404, Technical University of Denmark, DK-2800 Lyngby, Denmark.
- eita. (2019). Optimización topológica. 2019, de Estudio de Ingeniería y Tecnología avanzada Sitio web: <https://eitaingenieros.com/optimizacion/>