

Practica 1:

Jesús Martínez García Samuel Gerardo Cuéllar Dávila
Karime Montserrat Cantú Ramírez Adrian Isaac Gomez Ocon
Joselyn Zacarías Chávez Seini Armando Ramos Durán

6 de septiembre de 2022

1. Introducción

Un problema clásico de la ingeniería consiste en determinar la configuración geométrica óptima de un cuerpo que minimice o maximice una cierta función, que satisfaga las restricciones o condiciones de contorno del problema. La solución de este problema puede ser planteada utilizando dos estrategias: como un problema de optimización de forma o de optimización de la topología.

Si bien, en este trabajo se verá un resumen del artículo que se facilitó en previas sesiones de laboratorio, acerca de la optimización topológica. Se asignó una estructura para analizar y se mostrará el resultado arrojado por el código.

2. Nombre y definición de la programación

Se le llama código de optimización de topología de 99 líneas”, en la que se presenta una cantidad de 99 líneas de código en Matlab en la que las 99 líneas de entrada corresponden al optimizador y a las subrutina de elemento finito.

El código de 99 líneas se divide en varias estructuras que componen cada una a cierta parte del programa que realiza cierta acción. De las 99 líneas de código: 36 líneas pertenecen al programa principal 12 líneas pertenecen a criterio optimizador 16 líneas a la independencia de mallado y 35 líneas al código de elemento finito

Viendo el código sin líneas de comentarios obtenemos que con tan solo 49 líneas de código son realmente necesarias para hacer dicha optimización de geometrías.

Agregando 3 líneas más de códigos seremos capaces de agregar múltiples cargas.

3. Estado del arte

Esta técnica es utilizada dentro del análisis estructural, esta es una herramienta matemática que permite sintetizar topologías óptimas. Esto con el objetivo de aligerar la estructura, pero manteniendo las funcionalidades mecánicas de interés en la pieza analizada. Las partes mecánicas con un menor peso generalmente implican un menor costo debido a que se usa menos material, además de que también se pueden tener otros beneficios, tales como disminuir la cantidad de energía necesaria para la operación si es que esta pieza pertenece a una máquina, esto debido a que se necesita menos esfuerzo para mover a pieza [1].

Como se hace a optimización topológica

Para a realización de este proceso se toma en cuenta el diseño, la carga que se aplicara y en que punto se aplicara, el material a utilizar, entre muchos otros. El proceso básicamente consiste en diseñar una pieza la cual buscaremos optimizar, para esto hay que definir la forma, material, uniones, puntos de contacto y restricciones de la pieza y posteriormente se aplicarán cargas simuladas a esta pieza y se evaluara las partes donde exista un mayor esfuerzo

y también las partes en las cuales el esfuerzo sea menor. Posteriormente el software se encarga de generar una nueva geometría para la pieza la cual tendrá una menor cantidad de material, pero buscará cumplir con todas las especificaciones mecánicas requeridas. Por ultimo se hace una verificación para probar que la pieza generada cumple con las características especificadas y si es así se completa el proceso de diseño.

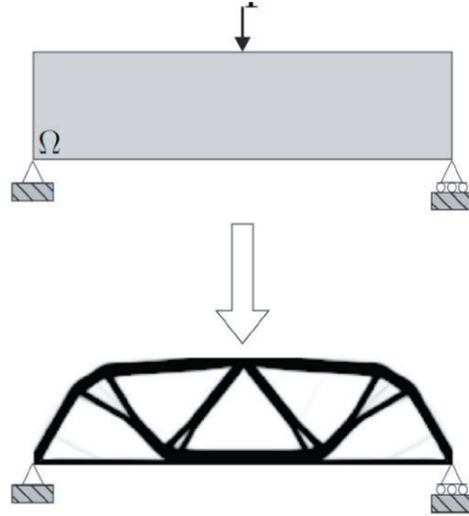


Figura 1: Ejemplo de una optimización topológica.

Algunas de las aplicaciones de la optimización topológica son el diseño de mecanismos flexibles, micro mecanismos, Sistemas Micro-Electro-Mecánicos, resonadores, aletas para intercambio de calor, etc. Actualmente existen muchos softwares encargados de realizar este tipo de optimización, entre los cuales podemos encontrar varios programas tales como Dassault Systèmes, Ansys, Inspire de SolidThinking, Simufact Additive, entre muchos otros. Cabe resaltar que no todos los programas de CAD cuentan con esta función integrada, sin embargo debido a su gran cantidad de aplicaciones en la industria muchos programas integran su propio software para hacer esta función.

4. Código

```

%% A 99 LINE TOPOLOGY OPTIMIZATION CODE BY OLE SIGMUND, OCTOBER 1999
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)) ...
'_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)
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
% DEFINE LOADSAND 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 127
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)];

```

5. Procedimiento de la programación

Bueno, lo primero que se hizo fue tomar el código que se encontró en el documento que se nos proporcionó y ser verifíco su funcionamiento.

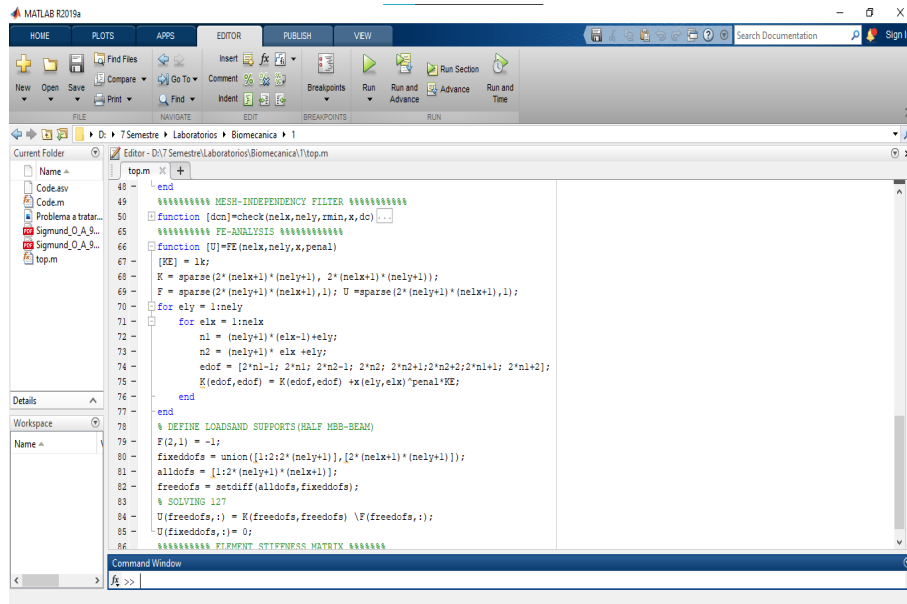


Figura 2: Programa en Matlab

Una vez comprobamos que todo estaba en orden, se planteó una geometría para analizar, se eligió una sencilla ya que creemos que es la mejor forma de comprender el funcionamiento del programa.

El problema planteado fué el siguiente:

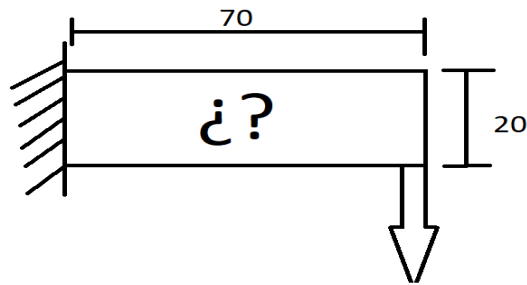


Figura 3: Problema a tratar

6. Implementación del programa en sus diferentes vistas.

Una vez teníamos el problema, solo restaba utilizar el script proporcionado y elegir los parámetros. Todos los parámetros fueron elegidos arbitrariamente.

```
nelx = 70
nely = 20
volfrac = 0.4
penal = 3.0
rmin = 1.2
top(70,20,0.4,3.0,1.2)
```

Después de ejecutar esa línea de comando este fue el relatado obtenido:

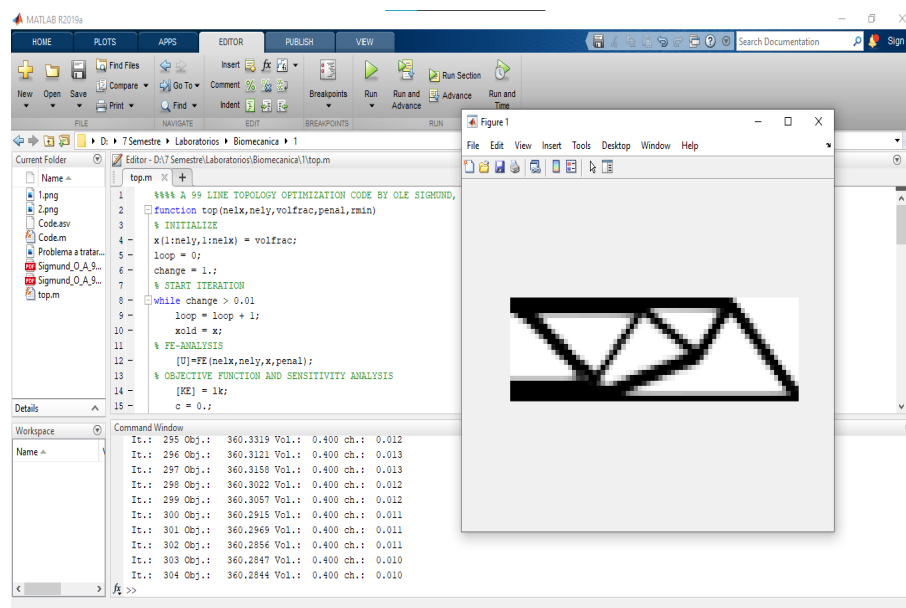


Figura 4: Resultado obtenido

Y aquí podemos analizar más de cerca el resultado:



Figura 5: Resultado obtenido

Vemos que el programa nos arroja resultados buenos, además de que se puede llegar a configurar para distintas situaciones como una doble carga o algún orificio dentro de la estructura, pero como se mencionó, se trabajó con una estructura simple para comprender mejor.

7. Conclusiones

Adrián Isaac Gómez Ocón 1992343:

Personalmente encontré este trabajo bastante interesante ya que me permitió conocer una técnica que no conocía para la optimización de piezas mecánicas. Me resultó muy interesante conocer como es que se pueden lograr piezas que consigan las mismas propiedades mecánicas que otras, pero utilizando una cantidad mucho menor de material. En este trabajo pude aprender un poco sobre como es que funcionan los programas detrás de estas piezas y como es que este tipo de optimización tienen un gran uso dentro de la industria. Considero que esta es una gran muestra de como es posible aplicar otras áreas de conocimiento tales como las matemáticas y la programación para resolver problemas que surgen dentro del ámbito del diseño mecánico y como es que la creación de algoritmos computacionales puede resultar sumamente útil para agilizar procesos de diseño, ya que si bien podríamos intentar realizar este tipo de optimización por nuestra cuenta sin ayuda de software, esto sería sumamente tardado y difícil.

Samuel Gerardo Cuéllar Dávila 1992262:

Resulta bastante interesante como un código de menos de 100 líneas puede ejecutar una optimización de carga para una pieza, por lo que si se sabe que una pieza estará sometida a una carga en cierto punto, la optimización ayudará a la mejora de peso de material, por lo que tendremos una mejor ingeniería aplicada para ciertos casos. La programación hace un papel bastante fundamental para el desarrollo de múltiples cálculos en cuestión de un momento, por lo que resulta bastante rápido y útil una topología así que no solo se puede aplicar para la pieza que vimos en el ejercicio si no que podemos meter diferentes casos de cargas y geometrías.

Jesús Martínez García 1991837:

Para mi este trabajo fue muy interesante, ya que investigando y leyendo los artículos proporcionados pudimos comprender un poco más sobre este proceso de optimización, la verdad es que son cálculos avanzados y se tiene que

tener experiencia en el tema, con la realización de este trabajo no nos adentramos ni un 10 por ciento de todo lo que abarca pero nos podemos ir dando una idea de todo este mundo. Creo que se logró el objetivo pero fue mediante un ejemplo muy sencillo.

Karime Montserrat Cantú Ramírez 1675807:

Este trabajo me interesó mucho, ya que se nos incitó a leer aún más sobre el tema de optimización. Así pues, como conclusión hay que señalar que aunque éste ha sido solamente un trabajo introductorio, pude notar que se ha permitido desarrollar herramientas útiles para definir algoritmos de resolución que proporcionan la solución óptima ante cualquier demanda y/o problema en cuestión posible que se pueda analizar, como lo hicimos previamente.

Seini Armando Ramos Durán 1991948:

Cómo resultado de la actividad se lograron aplicar las competencias en base a la información enfocada en la optimización de tal forma que se llevó a cabo la introducción de nuevas formas de realizar los procedimientos en base a qué podamos trabajar con estas herramientas y lograr hacer actividades con mayor grado de complejidad más adelante por lo cual se me hizo una actividad muy bien estructurada e interesante

Joselyn Zacarías Chávez 1992173:

Gracias a la práctica realizada y la investigación previamente hecha hemos logrado comprender un poco más acerca de la optimización. Siendo a penas la primer práctica realizada creo que nos ha ayudado pues usamos herramientas claves y básicas que nos ayudarán en nuevas prácticas, así como en la resolución de problemas que se nos presenten a futuro.

Referencias

- [1] Ediguer E. Franco Carlos A. Meza, Fernando Tamayo. Optimización topológica aplicada al diseño de componentes estructurales mecánicos de peso reducido, Enero 2015.