

# Práctica 3

## Diseño de la estructura de un panorámico

Equipo 4  
Omar Gutierrez

26 de octubre de 2022

### 1. Introducción

En esta práctica se llevará a cabo una propuesta de análisis de formas, análisis y solución del diseño de la estructura de un panorámico con la ayuda de del código de 99 líneas de optimización topológica, tomando en cuenta las cargas y los puntos fijos de un determinado panorámico para así desarrollar como su nombre lo indica, una optimización topológica.

### 2. Objetivo

El estudiante deberá presentar una propuesta de análisis de formas y de la programación para la ejecución de la optimización (descripción funcional) de características de trabajo específicas que presenta la(s) ventaja(s) (mencionar ventajas).

La metodología que se utilizará para la propuesta se compondrá en los siguientes pasos:

- 1) Nombre y definición de la geometría
- 2) Estado del arte
- 3) Propuesta de diseño de la geometría, alcances y limitaciones
- 4) Pasos del desarrollo de la programación
- 5) Resultados de la optimización
- 6) Conclusiones

### 3. Nombre y definición de la forma geométrica

En esta práctica se aplicará optimización topológica a un panorámico. Hay que aclarar que existen diferentes tipos de panorámicos y cada uno de ellos está diseñado de forma diferente por lo que la geometría de los panorámicos que se encuentran en muchas de las avenidas transitadas no tendrá la misma distribución de volumen.

Por esto, primero se empieza a definir cuál es la geometría que se va a analizar, en este caso optimizaremos uno de los panorámicos bajos, que se encuentran a poca altura de su base de empotramiento como el de la figura ??, para así analizar las cargas a las que se exponen como a altas ráfagas de viento, y con esto necesitaremos así diseñar una estructura trasera muy rígida para soportar estas fuerzas.



Figura 1: Panorámico propuesto

## 4. Estado del arte

### 4.1. Optimización topológica

La optimización de la topología (TO) es un método matemático que optimiza el diseño del material dentro de un espacio de diseño dado, para un conjunto dado de cargas, condiciones de contorno y restricciones con el objetivo de maximizar el rendimiento del sistema. TO es diferente de la optimización de la forma y la optimización del tamaño en el sentido de que el diseño puede alcanzar cualquier forma dentro del espacio de diseño, en lugar de tratar con configuraciones predefinidas.

La formulación convencional de TO utiliza un método de elementos finitos [FEM] para evaluar el desempeño del diseño [2]. El diseño se optimiza utilizando técnicas de programación matemática basadas en gradientes, como el algoritmo de criterios de optimalidad y el método de mover asíntotas o algoritmos no basados en gradientes, como los algoritmos genéticos.

En el proceso de optimización topológica, 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 Optimización Topológica:

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

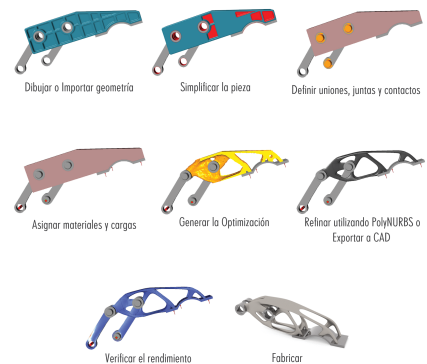


Figura 2: Flujo de trabajo

## 4.2. Panorámicos

Los anuncios panorámicos espectaculares se conocen así debido a su gran tamaño. La palabra panorámico viene de panorama, que significa vista. Por ejemplo, la vista que se tiene desde un lugar alto. En pocas palabras, es un anuncio publicitario con gran tamaño, aunque tiene distintas medidas. Este tipo de anuncio fue diseñado para servir al aire libre. Su principal función es capturar la atención de conductores y peatones.

El tamaño de un panorámico depende principalmente de dónde se encuentre localizado. En Monterrey, México, el tamaño promedio de la renta de anuncio panorámico es de 12 metros de largo y 7.2 metros de ancho. Este aspecto en los espectaculares varía según su tamaño. Sin embargo, el peso de estos anuncios se encuentran entre 5 y 10 toneladas aproximadamente [1].



Figura 3: Panorámico

Este tipo de estructura aparentemente es muy sencilla porque cuenta solamente con tres partes principales que son:

- Mampara
- Pedestak
- Cimentación

Sin embargo, vista en forma minuciosa, una mampara consta de varios componentes y accesorios que hacen que esta estructura sea realmente muy compleja tanto en su diseño estructural, como en su construcción y también en su comportamiento sobre todo ante viento como el producido por huracán o tromba.

Tanto la cimentación como el pedestal y la mampara elevada pueden constar de diversos elementos tales como: anclas suelo-zapata, vigas estabilizadoras, anclas pedestal-zapata, lastres, placas-base, acartelamientos, el tubo del pedestal, escaleras externas e internas, andamios, placas de conexión pedestal-mampara, travesano principal de la mampara, placas verticales del travesano, armaduras, pernos de sujeción, láminas de la mampara, accesorios de iluminación, ganchos o argollas de sujeción, travesanos secundarios; además, se tienen diversos elementos de sujeción o conexión tales como pernos, tornillos, remaches, soldaduras, etc.

## 5. Propuesta de diseño de la gemetría, alcances y limitaciones

Lo más común de donde se ubican dichas estructuras es en vías públicas donde se pueden encontrar la mayor parte de la población dependiendo dicho sector de la ciudad, teniendo en cuenta esto, se debe garantizar su seguridad porque su colapso podrá afectar a muchas personas. Los rótulos deben ser diseñados de una forma conservadora para proteger a las personas para evitar una catástrofe, como así también evitar pérdidas económicas directamente, como por ejemplo los costos de la estructura misma y daños secundarios. Gracias a la optimización topológica se busca la reducción del volumen de la estructura del panorámico y que soporte las cargas necesarios que esta será

sometida como por ejemplo la carga del viento mediante movimientos horizontales de masas de aire provocado por las diferentes unidades de presión en los distintos sectores de la atmósfera y teniendo en cuenta la rotación de la tierra, para estos movimientos que ocurren habitualmente es esencial tomar en cuenta la velocidad y presión de los vientos para llevar a cabo el diseño y la sumatoria de las fuerzas que se estarían aplicando en dicha estructura.

## 5.1. Propuesta

Para la propuesta de la estructura de un panorámico se genera una geometría rectangular elevada simple, fuerzas múltiples siendo 5 cargas, los apoyos tendrán restricciones en “X”,”Y” para que al aplicarle el algoritmo de optimización topológica se pueda optimizar la geometría y el espacio de diseño para esta práctica será de:

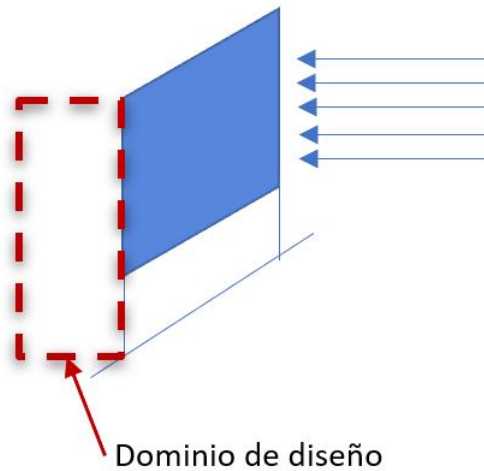


Figura 4: Dominio de diseño

## 6. Desarrollo de programación

Para realizar esta práctica, tenemos que hacer algunos cambios a las líneas del código de optimización topológica desarrollado por Ole Sigmund [3], para esto tenemos que usar cinco fuerzas, dos empotramientos y generar un espacio vacío, los cambios realizados se muestran a continuación:

### 6.1. Elementos pasivos y generación de rectángulo lateral

En el archivo del uso del código de 99 líneas existe una sección donde se habla de elementos pasivos el cual sirve de ayuda para determinar un espacio en blanco, en el ejemplo del archivo viene como hacer un círculo, y nosotros necesitamos un rectángulo, para esto se modificaron y agregaron las siguientes líneas:

#### ■ Líneas 7 a 16

```
for ely = 1:nely
    for elx = 1:nelx
        if (((ely-(nely*0.5)<(2*elx)-(1.36*nelx)) || (ely <(1+nely))) &&
            (elx > (1+nelx)*0.6666))
            passive(ely,elx) = 1;
        else
            passive(ely,elx) = 0;
        end
    end
end
```

```

        end
    end
    x(find(passive))=0.001;

```

■ **Línea 43**

```
[x] = OC(nelx , nely , x , volfrac , dc , passive );
```

■ **Línea 43 y 53**

```
[x] = OC(nelx , nely , x , volfrac , dc , passive );
function [xnew]=OC(nelx , nely , x , volfrac , dc , passive )
```

## 6.2. Cargas y empotramientos

Si observamos la figura 4, nos encontramos con cinco fuerzas y dos empotramientos, para cambiar el empotramiento del espacio de diseño a otra posición se tiene que cambiar la línea con la instrucción `fixeddofs`, para esto se modificaran las siguientes líneas:

■ **Líneas 32 a 37**

```

for i = 1:5
    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) = dc(ely , elx)-penal*x(ely , elx)^(penal-1)*Ue'*KE*Ue;
end

```

■ **Líneas 82 a 85**

```

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),5); U=zeros(2*(nely+1)*(nelx+1),5);

```

■ **Líneas 94 a 102**

```

% DEFINE LOADS AND SUPPORTS (HALF MBB-BEAM)
F(2*nelx*(nely+1)+2,1) = 1;
F(2*nelx*(nely+1)+(nely/4),2) = 1;
F(2*nelx*(nely+1)+(nely/2),3) = 1;
F(2*nelx*(nely+1)+(nely),4) = 1;
F(2*nelx*(nely+1)+(nely*1.2),5) = 1;
fixeddofs = 2*(nely+1):2*(nely+1):2*(nelx+1)*(nely+1);
alldofs = [1:2*(nely+1)*(nelx+1)];
freedofs = setdiff(alldofs , fixeddofs );

```

Habiendo realizado cada una de las modificaciones anteriormente presentadas, se corrigen errores de sintaxis en el código y con esto ya acabado se corre el código con las siguientes especificaciones en la función principal:

$$topp3(40, 20, 0, 2, 3, 0, 0, 5); \quad (1)$$

## 7. Resultados

Una vez realizado hay que comprobar que el rectángulo y el espacio vacío en la esquina superior izquierda se genero representa correctamente en base a la figura 4, para esto, en la consola de Matlab se introduce la siguiente línea del código **topp3(40,20,1,3.0,0.5)**, donde definimos los valores de la función superior pero solo generamos una iteración que nos permitirá ver la geometría generada.

Como podemos observar, en la figura 5 se generó correctamente el espacio vacío que representa el espacio que ocupara una parte de la rueda delantera.



Figura 5: Geometría generada

Con esto comprobado se procede a introducir nuevamente la línea de código, pero ahora modificamos volfrac y penal **topp3(40,20,0.15,3.6,1.5)** para que se produzcan las iteraciones y con los parámetros especificados al llamar la instrucción lo que se hace es discretizar 40 veces en 20 elementos finitos el dominio de diseño.

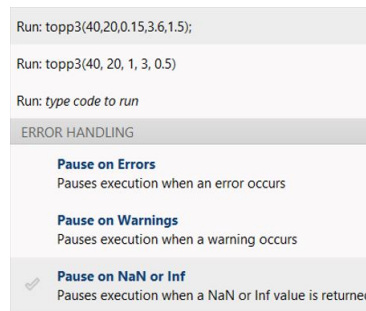


Figura 6: Modificación de función superior

Realizando la operación, finalmente obtenemos la optimización de la geometría, como podemos observar en la figura 7 el código propuesto determinó la distribución de masa más eficiente para cumplir exactamente con los requisitos de diseño impuestos a la pieza, tiene en cuenta el espacio permitido, las restricciones y condiciones de carga de la pieza y las tensiones máximas admisibles en el material, por lo que con esto se obtiene el resultado esperado y se da por concluida la optimización.

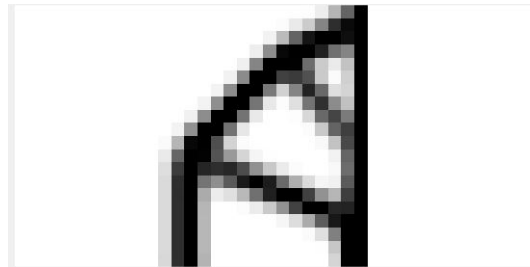


Figura 7: Geometría optimizada

A continuación, se anexa el código final después de las modificaciones realizadas.

```

%% A 99 LINE TOPOLOGY OPTIMIZATION CODE BY OLESIGMUND, OCTOBER 1999
function topp3(nelx ,nely ,volfrac ,penal ,rmin);
% INITIALIZE
x(1:nely,1:nelx) = volfrac;
loop = 0;
%Declarando vacio
for ely = 1:nely
    for elx = 1:nelx
        if (((ely-(nely*0.5)<(2*elx)-(1.36*nelx)) || (ely <(1+nely))) &&(elx > (1+nelx)*0.6666))
            passive(ely ,elx) = 1;
        else
            passive(ely ,elx) = 0;
        end
    end
end
x(find(passive))=0.001;
change = 1.;
% START ITERATION
while change > 0.01
    loop = loop + 1;
    xold = x;
    % FE-ANALYSIS
    [U]=FE(nelx ,nely ,x,penal);
    %13 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; %19
            dc(ely ,elx) = 0.;
            for i = 1:5
                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) = dc(ely ,elx)-penal*x(ely ,elx)^(penal-1)*Ue'*KE*Ue;
            end
        end
    end
end
%25 FILTERING OF SENSITIVITIES
[dc] = check(nelx ,nely ,rmin ,x,dc);
%27 DESIGN UPDATE BY THE OPTIMALITY CRITERIA METHOD
[x] = OC(nelx ,nely ,x,volfrac ,dc,passive);
%29 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
%40 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
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)))));
xnew(find(passive)) = 0.001;
if sum(sum(xnew)) - volfrac*nelx*nely > 0;
    l1 = lmid;
else
l2 = lmid;
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),5); U=zeros(2*(nely+1)*(nelx+1),5);
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*nelx*(nely+1)+2,1) = 1;
F(2*nelx*(nely+1)+(nely/4),2) = 1;
F(2*nelx*(nely+1)+(nely/2),3) = 1;
F(2*nelx*(nely+1)+(nely),4) = 1;
F(2*nelx*(nely+1)+(nely*1.2),5) = 1;
fixeddofs =2*(nely+1):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)];

```

## 8. Conclusiones

Omar Alejandro Gutierrez Carrizales:

Durante esta práctica se propuso una geometría para la estructura de un panorámico con la finalidad de aplicar la optimización topológica y poder optimizar la geometría propuesta, reduciendo la masa y manteniendo la rigidez de la pieza. Para lograr esto se utilizaron conocimientos previos respecto al código de optimización topológica en MatLab para adaptarlo al problema que se está trabajando, cabe recalcar que se hizo todo lo posible para simplificar la geometría propuesta para que todo el análisis sea lo más sencillo posible.

## Referencias

- [1] Blucactus. Anuncios panorámicos espectaculares en la república mexicana. Recuperado de: [https://www.sciencedirect.com/science/article/pii/S0213131515000127](https://www.blucactus.com.mx/panoramicos/#:~:text=La%20publicidad%20en%20anuncio%20panor%C3%A1mico,rutas%20de%20tr%C3%A1fico%20y%20autopistas., 2020.</a></li>
<li>[2] O. Begambre Carrillo C. Millán Páramo. Solución de problemas de optimización topológica empleando el algoritmo simulated annealing modificado. Recuperado de: <a href=), 2016.
- [3] Ole Sigmund. A 99 line topology optimization code written in matlab. Recuperado de: <https://www.topopt.mek.dtu.dk/apps-and-software/a-99-line-topology-optimization-code-written-in-matlab>, September 2002.