

**UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN**  
**FACULTAD DE INGENIERIA MECANICA Y ELECTRICA**

**Laboratorio de Biomecánica**  
**PRACTICA 3**

**Grupo: 109**

**Día: Lunes**

**Hora: N5**

**Equipo 3.**

<b>Nombre</b>	<b>Matrícula</b>
<b>LUIS GARCIA GONZALEZ</b>	1604958
<b>ADAN ASIS BRIONES TORRES</b>	1732258
<b>SERGIO ESTEBAN CANTU CARRASCO</b>	1863714
<b>OSCAR MARCELO FRAGOSO MARTINEZ</b>	1894650
<b>ALFREDO CARDENAS MENA</b>	1902495

Date: 17/10/2022.

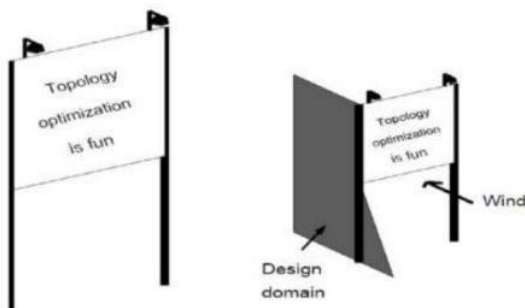
# Objetivo

Presentar una propuesta análisis de formas y de la programación para la ejecución de la optimización de características de trabajo específicas.

## Marco Teórico

Una estructura panorámica es el soporte sobre el cual se posicionará un anuncio publicitario, ya sea de una cara o de tres caras. Estas estructuras usualmente se encuentran en medio de diversos paisajes urbanos y sostienen diseños publicitarios con el objetivo de promocionar un producto, servicio o transmitir un mensaje. Cada país tiene ciertas normativas en cuanto a dónde es apropiado o no colocar estos soportes para anuncios publicitarios. En algunos no está permitido que se construyan estructuras panorámicas a los lados de autopistas porque estos pueden distraer a los conductores. Los panorámicos se exponen a altas ráfagas de viento, por lo que su estructura ocupa ser muy rígida para soportar estas fuerzas.

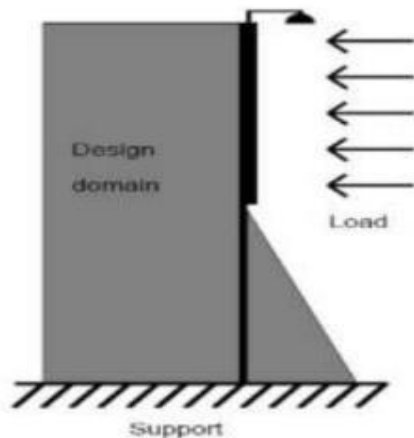
En la figura se muestra el panorámico que será el espacio de diseño a evaluar, éste será de 2 dimensiones, con cargas y apoyos como se muestra a continuación:



Existen diversos materiales que las agencias especializadas usan en la creación y diseño para estas estructuras. Usualmente los postes panorámicos están contruidos de metal y acero para que sean lo suficientemente resistentes al clima, lluvias y cualquier otro fenómeno de la naturaleza

Por otra parte, el panorámico en sí mismo son hechos de lona, vallas de PVC, plástico, tela, metal o acrílico. También existen espectaculares digitales o electrónicos que tienen luces, pantallas eléctricas y música.

En la figura 2 se puede ver el espacio de diseño para esta práctica. Se espera una fracción volumétrica aproximada de 0.20% del espacio de diseño. Supongamos que el panorámico es muy rígido 1, y sus patas son del mismo material que el marco.



## Estado del arte

En esta práctica se busca realizar un diseño de la estructura de un panorámico definiendo estructuras funcionales y seguras utilizando la optimización topológica para hacer este diseño.

Los panorámicos se enfrentan a distintas cargas como las cargas muertas, vivas, de viento o sismo. Estas estructuras suelen ser muy variadas tanto en ubicación, forma, soporte, geometría o dimensiones. Una característica esencial de los panorámicos es la forma en que se conectan los elementos para darle estabilidad al sistema, lo que a su vez define la forma en que debe fabricarse e instalarse la estructura.

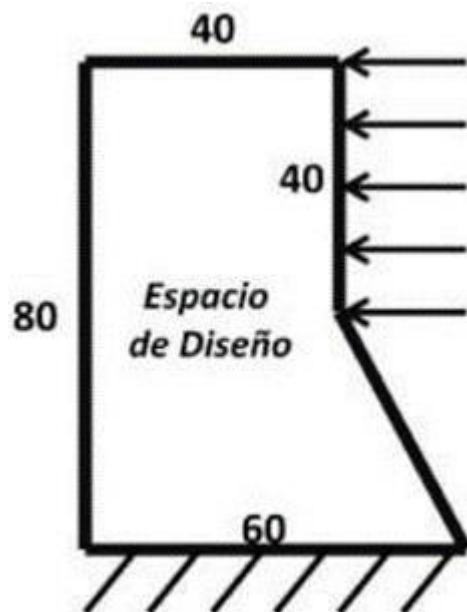
Normalmente la ubicación de estos es en lugares de mucho tránsito de personas, donde puedan llegar a un sector de población más alto lo que los lleva a verse a lo largo de las vías terrestres más transitadas y con más densidad en las áreas urbanas. Por lo tanto, a pesar de que estas estructuras no son tan importantes por sí mismas, debe de garantizarse su seguridad porque

su colapso podría afectar a muchas personas. Por dicha razón, los rótulos deben ser diseñados de una forma conservadora para proteger las vidas de las personas y también evitar pérdidas económicas directas, como costo de la estructura misma y daños a otros inmuebles e indirectas, como los tiempos muertos de tránsito vehicular o paralizar momentáneamente áreas comerciales.

Mediante la optimización topológica se busca la reducción del volumen de la estructura del panorámico y que soporte las cargas a las que este está sometido como sería la carga por viento mediante movimientos horizontales de masas de aire provocados por las diferencias de presión en las distintas zonas de la atmósfera y a la rotación terrestre. Estos movimientos ocurren constantemente; sin embargo, para el diseño estructural interesan esencialmente los vientos que tienen velocidades muy grandes.

## Propuesta de diseño de la geometría, alcances y limitaciones

Se tomarán ciertas consideraciones para la solución de esta práctica: 5 cargas, los apoyos tendrán restricciones en “X”, “Y” y el espacio de diseño para esta práctica será de:



## Pasos del desarrollo de la programación

Fuerzas múltiples.

Ingresamos las 5 fuerzas y cambiar el anclaje del espacio de diseño a otra posición se tiene que cambiar la línea con la instrucción FE-ANALYSIS

Esta es la instrucción original:

```
%%%%%%%%%% 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 = zeros(2*(nely+1)*(nelx+1),1);
```

Se hace la modificación y queda de esta manera:

```
%%%%%%%%%% 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;
        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
```

Esta parte del código corresponde a

El cual debe ser reemplazado por:

```
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],i)
    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
```

Las siguientes líneas son reemplazadas:

```
F(2,1) = -1;  
fixeddofs = union([1:2:2*(nely+1)], [2*(nelx+1)*(nely+1)]);  
alldofs = [1:2*(nely+1)*(nelx+1)];
```

Por los siguiente

```
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);
```

Empotramiento diagonal (elementos pasivos)

Las siguientes líneas se modificaron y/o agregaron para crear la incrustación diagonal, o crear el espacio en blanco para recrear la incrustación en la parte inferior derecha, en la sección del archivo de uso de código de 99 líneas que analiza los elementos pasivos. Hacer un círculo es el ejemplo proporcionado aquí, pero también necesitamos un rectángulo y un triángulo para hacer esto.

```
%%% 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;
```

Líneas modificadas:

```

%% 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.;
%Declarando vacio
for ely = 1:nely
    for elx = 1:nelx
        if ((ely-(nely*0.5)<(2*elx)-(1.36*nelx)) | (ely <(1+nely*0.5))) & (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

```

Además originalmente la siguiente parte del código estaba así

```

% DESIGN UPDATE BY THE OPTIMALITY CRITERIA METHOD
[x] = OC(nelx,nely,x,volfrac,dc);

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [xnew]=OC(nelx,nely,x,volfrac,dc)

```

```

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;

```

Estas tres partes se modifican y quedan de la siguiente manera:

```

% DESIGN UPDATE BY THE OPTIMALITY CRITERIA METHOD
[x] = OC(nelx,nely,x,volfrac,dc,passive);

```

```

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

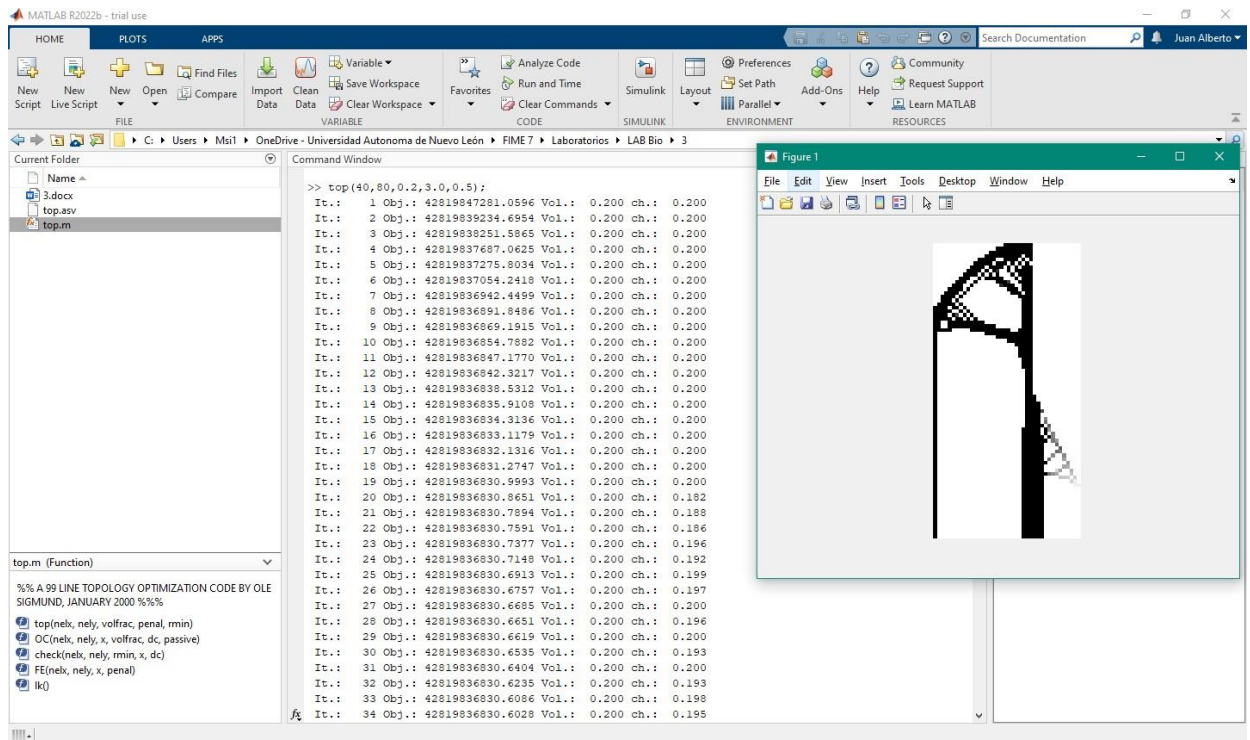
```

```

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;

```

# Resultado de la optimización



## Conclusión

**LUIS GARCIA**

En esta práctica hicimos uso de análisis de elemento finito para objetos como lo fue en esta actividad y además de poder sea utilizado en otros casos, además de generar un buen soporte. representa una herramienta importante para la exploración y la extracción de conocimiento. Siendo el principal objetivo construir modelos que permitan describir posibles patrones estructurales en la información a partir de los datos, con el objetivo de tomar decisiones o hacer predicciones.

**ADAN BRIONES**

En esta práctica se logra optimizar un panorámico mediante el código en el software de Matlab. Con la idea de que la estructura tenga la rigidez suficiente para soportar las cargas a las que se expone como son las cargas por viento, mientras al mismo tiempo se reduzca el volumen mediante la optimización topológica.



## **SERGIO CANTU**

El problema es el diseño de una valla publicitaria que está expuesta a un campo de viento. El soporte de la valla publicitaria necesita algún tipo de marco de soporte para hacerlo más rígido. En la figura del espacio de trabajo se da una ilustración bidimensional de las cargas y el apoyo. Una fracción de volumen de aproximadamente 0,2 del dominio de diseño. Supusimos que la valla publicitaria es muy rígida 1 y las patas disponibles están hechas del mismo material que el marco. En este informe de laboratorio, demostramos el trabajo que se completó a través de la práctica con Matlab. Descubrimos que esto tomó más tiempo en completarse que los experimentos anteriores porque el software tuvo que pasar por un proceso para maximizar sus esfuerzos, y también notamos que los espacios en blanco, que son componentes pasivos, debían tenerse en cuenta para el diagrama. Podemos inferir de nuestro trabajo que, aunque comúnmente se asume que algo se ignora porque es un espacio en blanco dentro de un sistema de intentos, esto no debería ser así y debemos darle valor para el diseño ideal del diagrama.

## **OSCAR FRAGOSO**

Con esta práctica se llegó a comprender el análisis del comportamiento estático de un panorámico y con ello, lograr plasmarlo en un código en Matlab y llevarlo a su representación gráfica con el mismo. Personalmente, esta práctica resultó de alto valor de aprendizaje, ya que puso a prueba al equipo el reto de analizar de manera crítica y poder llegar a la representación gráfica del panorámico.

## **ALFREDO CARDENAS**

Gracias a esta práctica se aprendió a analizar de una manera diferente el comportamiento estático de un panorámico y así poder plasmarlo en un código en Matlab y llevarlo a su representación gráfica con el mismo. Esta práctica resultó de alto valor de aprendizaje para mí, al ponernos como reto el analizar de manera crítica el programa original y con eso poder llegar a la representación gráfica del panorámico.

```

%%% 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.;
%Declarando vacio
for ely = 1:nely
    for elx = 1:nelx
        if ((ely-(nely*0.5)<(2*elx)-(1.36*nelx)) |(ely <(1+nely*0.5))) & (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);
% 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
% FILTERING OF SENSITIVITIES
[dc] = check(nelx,nely,rmin,x,dc);
% DESIGN UPDATE BY THE OPTIMALITY CRITERIA METHOD
[x] = OC(nelx,nely,x,volfrac,dc,passive);
% 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
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [xnew]=OC(nelx,nely,x,volfrac,dc,passive)

```



