



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FIME



FACULTAD DE INGENIERIA MECANICA Y ELECTRICA

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERIA MECANICA Y ELECTRICA

Práctica #4

U.A: Laboratorio de Biomecánica

Equipo #2

Integrantes del Equipo:

Melissa Alejandra Jasso Maciel	1897337	IMTC
Carlos Antonio Caballero Padilla	1900864	IMTC
Omar Gerardo Ríos Gaytán	1902388	IMTC
Edgar Alan Carrizales Treviño	1904406	IMTC
Nestor Eliud Cano Garcia	1909644	IMTC

BRIGADA: 204 AULA: 12BMC

DOCENTE: Yadira Moreno Vera

Fecha de entrega: 17/10/22

Semestre Agosto– diciembre 2022



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FIME



FACULTAD DE
INGENIERÍA MECÁNICA
Y ELÉCTRICA

Nombre y Definición de la forma geométrica

Los teleféricos son un medio de transporte utilizado desde hace muchos años. En algunos países como México se están instalando para facilitar la movilidad de los ciudadanos a lugares de difícil acceso para el transporte público. El teleférico es un ingenio que sirve para el transporte de personas y mercancías en zonas de alta montaña o con pronunciados desniveles. Entre sus características principales, destaca que dicho transporte se efectúa sobre vagonetas o cabinas que viajan suspendidas de un fuerte cable de acero.

Este tipo de transportes singulares formados por un cable, encargado de dar movimiento a vehículos ya sean aéreos o terrestres tienen diferente tipo de cable y de empalme o forma de sujetar con los vehículos en función de las características de la instalación.



Figura 1.1. Teleférico común y corriente.

El diámetro del cable, los cordones del cable, la longitud del empalme, el sistema de sujeción de los extremos del cable son datos que se calculan en el proceso de diseño por ingenieros donde la precisión es muy importante por los costes elevados que tienen al errar en la fabricación de un cable.

Los fabricantes de cables crean la bobina de cable a demanda del cliente, siendo un producto personalizado para cada instalación.



Figura 1.2-3. Construcción de un soporte de teleférico y del cable del mismo..

Estado del arte

El teleférico es un sistema de transporte aéreo constituido por cabinas colgadas de una serie de cables que se encargan de hacer avanzar a las unidades a través de las estaciones. Cuando las cabinas van por tierra se denomina Funicular. Por ser una estructura poco convencional no se cuenta con un código que norme su diseño y construcción. Un teleférico debe ser visualizado como sistema estructural en el que sus componentes (anclajes, apoyos, cables) tienen comportamientos diferentes pero que funcionan en conjunto.

Algunos teleféricos usan dos cabinas por tramo (trayecto entre estación y estación) a fin de crear un contrapeso. Otros sistemas más complejos tienen varias cabinas suspendidas simultáneamente en cada dirección.

El sistema está compuesto por uno o más cables (dependiendo del tipo); el primer cable está fijo y sirve para sostener las cabinas, el segundo está conectado a un motor (ubicado en la estación) y hace mover las cabinas.

Un teleférico es un sistema en realidad muy sencillo; es mucho más complejo un vehículo de los que anda por las calles.

El sistema tiene tres elementos básicos (o subsistemas):

1. sistema de carga.
2. sistema de soporte.
3. sistema de tracción.

Es importante el desarrollo de todos los aspectos de los componentes y su correcto funcionamiento, los cuales ofrecen a los pasajeros el máximo confort

y seguridad. La línea tiene diferentes componentes, como las pilonas, los balancines y el cable.

Este tipo de transportes singulares formados por un cable, encargado de dar movimiento a vehículos ya sean aéreos o terrestres tienen diferente tipo de cable y de empalme o forma de sujetar con los vehículos en función de las características de la instalación.

El diámetro del cable, los cordones del cable, la longitud del empalme, el sistema de sujeción de los extremos del cable son datos que se calculan en el proceso de diseño por ingenieros donde la precisión es muy importante por los costes elevados que tienen al errar en la fabricación de un cable.

Propuestas del diseño de la geometría.

El cable es el encargado de soportar y/o transportar los vehículos. El cable está dimensionado para cada instalación, para que soporte el peso de las cabinas con carga máxima.



El cable está formado por un conjunto de hilos de acero de este o diferente tamaño trenzados de tal forma que generan otros cables que a su vez se trenzan alrededor del alma formando el cable del teleférico.

El centro del cable se le denomina alma y está formado normalmente por un plástico flexible que se adapta a los

hilos del cable.

El cable de un teleférico es continuo, teniendo las dos puntas unidas en el empalme del cable. Este empalme se crea en diferentes nudos. Los empalmes de los cables pueden llegar a medir varias decenas de metros.

Los cables tienden a alargarse, sobre todo cuando son nuevos, cada instalación tiene un margen de alargado que al estar próximo o en el límite de estirado del cable se tiene que acortar el cable.

La forma en como el cable es colocando es con un cierto grado de inclinación le cual le facilita la movilidad para transportar las personas, siendo que el cable esta reforzado siendo que se compone de varios cables de acero entrelazados de la misma forma en como una cuerda está hecha, siendo que las limitaciones que el diseño puede tener es que la velocidad que tendrá el teleférico será lenta, pero esto se debe a que para recorrer el cable de subida es necesario de tensores que irán tensando el cable arrastrando el cable y atrayendo el teleférico de un punto a otro sin poder hacerlo de forma rápida para disminuir los riesgos de alta fricción que puedan desgastar el cable, por otro lado esta forma de utilizar la pendiente para transportar de un lado a otro es algo útil debido a que es la ruta más corta, pese a ser un movimiento relativamente lento es un trayecto más corto, siendo que el cable esta reforzado es seguro mientras no se exceda el límite de peso permitido para que así la tensión en el cable no sea extrema, de misma forma recomendando disminuir el movimiento dentro del teleférico para que estos no perturben el equilibrio del mismo.

DESARROLLO DEL CÓDIGO DE LA PRÁCTICA

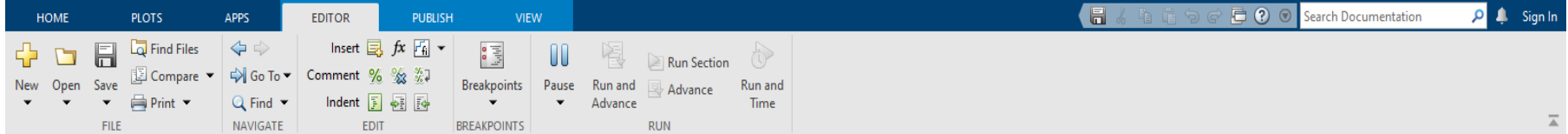
Para realizar este ejercicio o problema propuesto tendremos que primero cambiar algunas cosas de nuestro código antes de correrlo con `topp1(40,80,0.2,3.0,5)`

Primeramente el vacío del código que ya teníamos se modificara a

```
for ely = 1:nely
for elx = 1:nelx
if ely>21
if elx<31
passive(ely,elx) = 1;
else
passive(ely,elx) = 0;
end
end
end
end
```

Después de esto tendremos que modificar el vacío de otra optimización que está dentro del mismo código solo cambiando unas partes del mismo como estas:

```
for ely = 1:nely
for elx = 1:nelx
if ely>21
if elx<21
passive(ely,elx) = 1;
elseif elx>41
passive(ely,elx)=1;
else
passive(ely,elx) = 0;
end
end
end
end
```



Current Folder: D:\matlab2.0\bin

Workspace: topp4.m, getExtendedErrorCallback.m

Command Window: Editor - topp4.m

File Explorer contents:

- icutzdata
- m3iregistry
- ode
- programas
- util
- win32
- win64
- bulbas.jpg
- bulbasur.fig
- bulbasur.m
- charizard.fig
- charizard.m
- charizarddd.png
- crash_analyzer.cfg
- deploytool.bat
- ejem_gui.fig
- ejem_gui.m
- ejem_gui-pokedex.m
- grafica de señal cosenoidal con respecto al tiempo.fig
- hechizo.png
- inicial.fig
- inicial.m
- lcdata.xml
- lcdata.xsd
- lcdata_utf8.xml
- matlab.exe
- matlab_jenv.bat
- matlab - Acceso directo.lnk
- mbuild.bat
- mcc.bat
- mex.bat
- mexext.bat
- mw_mpiexec.bat
- pickaa.png
- pickachu.fig

Code Editor (topp4.m):

```
1  %%% A 99 LINE TOPOLOGY OPTIMIZATION CODE BY OLE SIGMUND, OCTOBER 1999 %%%
2  function new_pr42_f(nelx,nely,volfrac,penal,rmin);
3  % INITIALIZE
4  x(1:nely,1:nelx) = volfrac;
5  for ely = 1:nely
6  for elx = 1:nelx
7  if ely>21
8  if elx<21
9  passive(ely,elx) = 1;
10 elseif elx>41
11 passive(ely,elx)=1;
12 else
13 passive(ely,elx) = 0;
14 end
15 end
16 end
17 end
18 x(find(passive))=0.001;
19 loop = 0; change = 1.;
20 % START ITERATION
21 while change > 0.01
22 loop = loop + 1;
23 xold = x;
24 % FE-ANALYSIS
25 [U]=FE(nelx,nely,x,penal);
26 % OBJECTIVE FUNCTION AND SENSITIVITY ANALYSIS
27 [KE] = 1k;
28 c = 0.;
29 %
30 for ely = 1:nely
31 for elx = 1:nelx
32 n1 = (nely+1)*(elx-1)+ely;
33 n2 = (nely+1)* elx +ely;
34 dc(ely,elx)=0.;
35 for i=1:2
```

Figure 1: A plot showing a binary image of a tree-like structure, likely representing a topology optimization result. The structure consists of a central vertical stem and a fan-like canopy at the top, rendered in black pixels on a white background.

MATLAB R2021a

HOME PLOTS APPS EDITOR PUBLISH VIEW

New Open Save Find Files Find Compare Go To Find Find Comment % Insert Indent Breakpoints Continue Step Step In Step Out Run to Cursor Function Call Stack: FE Quit Debugging

FILE NAVIGATE EDIT BREAKPOINTS DEBUG

Current Folder: D:\matlab2.0\bin

Workspace: Command Window

topp4.m

```
1 %*** A 99 LINE TOPOLOGY OPTIMIZATION CODE BY OLE SIGMUND, OCTOBER 1999 ***
2 function new_prf42.f(nelx,nely,volfrac,penal,rmin)
3 % INITIALIZE
4 x(1:nely,1:nelx) = volfrac;
5 for ely = 1:nely
6     for elx = 1:nelx
7         if ely>21
8             if elx<21
9                 passive(ely,elx) = 1;
10                elseif elx>41
11                    passive(ely,elx)=1;
12                else
13                    passive(ely,elx) = 0;
14                end
15            end
16        end
17    end
18    x(find(passive))=0.001;
19    loop = 0; change = 1.;
20    % START ITERATION
21    while change > 0.01
22        loop = loop + 1;
23        xold = x;
24        % FE-ANALYSIS
25        [U]=FE(nelx,nely,x,penal);
26        % OBJECTIVE FUNCTION AND SENSITIVITY ANALYSIS
27        [KE] = 1k;
28        c = 0.;
29
30        for ely = 1:nely
31            for elx = 1:nelx
32                n1 = (nely+1)*(elx-1)+ely;
33                n2 = (nely+1)* elx +ely;
34                dc(ely,elx)=0.;
35            end
36        end
```

8 occurrences of "nelx" found

UTF-8

MATLAB R2021a

HOME PLOTS APPS EDITOR PUBLISH VIEW

New Open Save Find Files Compare Go To Find Print

Insert Comment Indent Breakpoints Continue Step Step In Step Out Run to Cursor Function Call Stack: FE Quit Debugging

FILE NAVIGATE EDIT BREAKPOINTS DEBUG

Current Folder: D:\matlab2.0\bin

Workspace: topp4.m, getExtendedErrorCallback.m

Command Window

```

34 - dc(ely,elx)=0.;
35 - for i=1:2
36 -     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);
37 -     c = c + x(ely,elx)^penal*Ue'*KE*Ue;
38 -     dc(ely,elx) = dc(ely,elx)-penal*x(ely,elx)^(penal-1)* Ue'*KE*Ue;
39 - end
40 - end
41 - end
42 - % FILTERING OF SENSITIVITIES
43 - [dc] = check(nelx,nely,rmin,x,dc);
44 - % DESIGN UPDATE BY THE OPTIMALITY CRITERIA METHOD
45 - [x] = OC(nelx,nely,x,volfrac,dc,passive);
46 - % PRINT RESULTS
47 - change = max(max(abs(x-xold)));
48 - disp([' It.: ' sprintf('%4i',loop) ' Obj.: ' sprintf('%10.4f',c) ...
49 -     'Vol.: ' sprintf('%6.3f',sum(sum(x))/(nelx*nely)) ...
50 -     ' ch.: ' sprintf('%6.3f',change )])
51 - % PLOT DENSITIES
52 - colormap(gray); imagesc(-x); axis equal; axis tight; axis off; pause(1e-6);
53 - end
54 - %***** OPTIMALITY CRITERIA UPDATE *****
55 - function [xnew]=OC(nelx,nely,x,volfrac,dc,passive)
56 -     l1 = 0; l2 = 100000; move = 0.2;
57 -     while (l2-l1 > 1e-4)
58 -         lmid = 0.5*(l2+l1);
59 -         xnew = max(0.001,max(x-move,min(1.,min(x+move,x.*sqrt(-dc./lmid)))));
60 -         xnew(find(passive))=0.001;
61 -         if sum(sum(xnew)) - volfrac*nelx*nely > 0.2
62 -             l1 = lmid;
63 -         else
64 -             l2 = lmid;
65 -         end
66 -     end
67 -     %***** MESH-INDEPENDENCY FILTER *****
68 -     function [denl,check(nely,nely,rmin,x,dc)]

```

MATLAB R2021a

HOME PLOTS APPS EDITOR PUBLISH VIEW

New Open Save Find Files Find Compare Go To Comment Indent Breakpoints Continue Step Step In Step Out Run to Cursor Function Call Stack: FE Quit Debugging

FILE NAVIGATE EDIT BREAKPOINTS DEBUG

Current Folder: D:\matlab2.0\bin

Workspace: Command Window Editor - topp4.m

8 occurrences of "nelx" found

```

67 ***** MESH-INDEPENDENCY FILTER *****
68 function [dcn]=check(nelx,nely,rmin,x,dc)
69 dcn=zeros(nely,nelx);
70 for i = 1:nelx
71 for j = 1:nely
72 sum=0.0;
73 for k = max(i-round(rmin),1): min(i+round(rmin),nelx)
74 for l = max(j-round(rmin),1): min(j+round(rmin),nely)
75 fac = rmin-sqrt((i-k)^2+(j-l)^2);
76 sum = sum+max(0,fac);
77 dcn(j,i) = dcn(j,i) + max(0,fac)*x(l,k)*dc(l,k);
78 end
79 end
80 dcn(j,i) = dcn(j,i)/(x(j,i)*sum);
81 end
82 end
83 ***** FE-ANALYSIS *****
84 function [U]=FE(nelx,nely,x,penal)
85 [KE] = 1k;
86 K = sparse(2*(nelx+1)*(nely+1), 2*(nelx+1)*(nely+1));
87 F = sparse(2*(nely+1)*(nelx+1),2); U = sparse(2*(nely+1)*(nelx+1),2);
88 for ely = 1:nely
89 for elx = 1:nelx
90 n1 = (nely+1)*(elx-1)+ely;
91 n2 = (nely+1)* elx +ely;
92
93 edof = [2*n1-1; 2*n1; 2*n2-1; 2*n2; 2*n2+1; 2*n2+2;2*n1+1; 2*n1+2];
94 K(edof,edof) = K(edof,edof) + x(ely,elx)^penal*KE;
95 end
96 end
97 % DEFINE LOADSAND SUPPORTS (HALF MBB-BEAM)
98 F(40,1) = -1.; F(9760,2)=1.;
99 fixeddofs = 2*(nely+1):2*(nely+1):2*(nelx+1)*(nely+1);
100 alldofs = [1:2*(nely+1)*(nelx+1)];
101 freedofs = setdiff(alldofs,fixeddofs);

```

UTF-8 new_pr42_f Ln 6 Col 18

```

85 - [KE] = lk;
86 - K = sparse(2*(nelx+1)*(nely+1), 2*(nelx+1)*(nely+1));
87 - F = sparse(2*(nely+1)*(nelx+1), 2); U = sparse(2*(nely+1)*(nelx+1), 2);
88 - for ely = 1:nely
89 -     for elx = 1:nelx
90 -         n1 = (nely+1)*(elx-1)+ely;
91 -         n2 = (nely+1)*elx +ely;
92 -         edof = [2*n1-1; 2*n1; 2*n2-1; 2*n2; 2*n2+1; 2*n2+2; 2*n1+1; 2*n1+2];
93 -         K(edof,edof) = K(edof,edof) + x(ely,elx)^penal*KE;
94 -     end
95 - end
96 - % DEFINE LOADSAND SUPPORTS (HALF MBB-BEAM)
97 - F(40,1) = -1.; F(9760,2)=1.;
98 - fixeddofs = 2*(nely+1):2*(nely+1):2*(nelx+1)*(nely+1);
99 - alldofs = [1:2*(nely+1)*(nelx+1)];
100 - freedofs = setdiff(alldofs,fixeddofs);
101 - % SOLVING
102 - U(freedofs,:) = K(freedofs,freedofs) \F(freedofs,:);
103 - U(fixeddofs,:)= 0;
104 - %***** ELEMENT STIFFNESS MATRIX %*****
105 - function [KE]=lk
106 - E = 1.;
107 - nu = 0.3;
108 - k=[ 1/2-nu/6 1/8+nu/8 -1/4-nu/12 -1/8+3*nu/8 ...
109 -    -1/4+nu/12 -1/8-nu/8 nu/6 1/8-3*nu/8];
110 - KE = E/(1-nu^2)* [ k(1) k(2) k(3) k(4) k(5) k(6) k(7) k(8)
111 -    k(2) k(1) k(8) k(7) k(6) k(5) k(4) k(3)
112 -    k(3) k(8) k(1) k(6) k(7) k(4) k(5) k(2)
113 -    k(4) k(7) k(6) k(1) k(8) k(3) k(2) k(5)
114 -    k(5) k(6) k(7) k(8) k(1) k(2) k(3) k(4)
115 -    k(6) k(5) k(4) k(3) k(2) k(1) k(8) k(7)
116 -    k(7) k(4) k(5) k(2) k(3) k(8) k(1) k(6)
117 -    k(8) k(3) k(2) k(5) k(4) k(7) k(6) k(1)];
118 -

```

También se aplico otro refuerzo de otra manera teniendo así dos optimizaciones, realmente no cambian mucho los códigos entre estas dos optimizaciones y tan solo es la instrucción que se le da al mismo código para que empiece a reforzar el cable. Esta segunda optimización resultaría algo así

MATLAB R2021a

HOME PLOTS APPS EDITOR PUBLISH VIEW

New Open Save Find Files Compare Go To Find Insert Comment Indent Breakpoints Pause Step Run to Cursor Function Call Stack: FE Quit Debugging

FILE NAVIGATE EDIT BREAKPOINTS DEBUG

Current Folder D:\matlab2.0\bin

Workspace Command Window Editor - topp4a.m

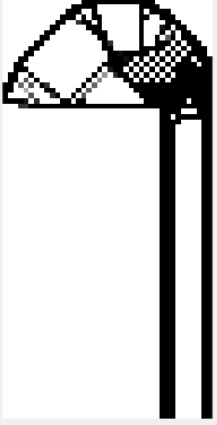
topp4a.m

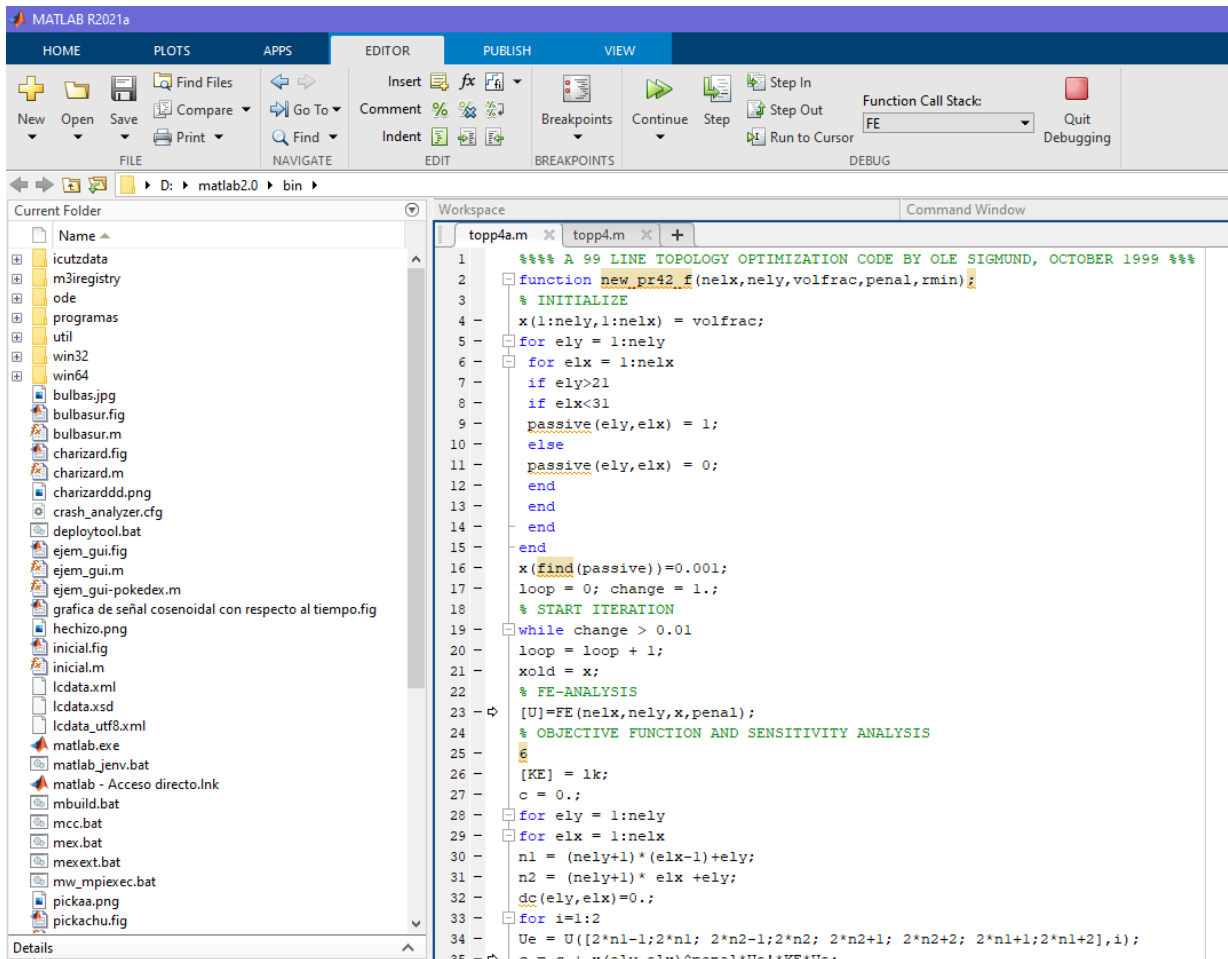
```

82 function [U]=FE(nelx,nely,x,penal)
83 [KE] = lk;
84 K = sparse(2*(nelx+1)*(nely+1), 2*(nelx+1)*(nely+1));
85 F = sparse(2*(nely+1)*(nelx+1),2); U = sparse(2*(nely+1)*
86 for ely = 1:nely
87 for elx = 1:nelx
88
89 n1 = (nely+1)*(elx-1)+ely;
90 n2 = (nely+1)* elx +ely;
91 edof = [2*n1-1; 2*n1; 2*n2-1; 2*n2; 2*n2+1; 2*n2+2;2*n1+1;
92 K(edof,edof) = K(edof,edof) + x(ely,elx)^penal*KE;
93 end
94 end
95 % DEFINE LOADSAND SUPPORTS (HALF MBB-BEAM)
96 F(40,1) = -1;
97 fixeddofs = 2*(nely+1):2*(nely+1):2*(nelx+1)*(nely+1);
98 alldofs = [1:2*(nely+1)*(nelx+1)];
99 freedofs = setdiff(alldofs,fixeddofs);
100 % SOLVING
101 U(freedofs,:) = K(freedofs,freedofs) \ F(freedofs,:);
102 U(fixeddofs,:)= 0;
103 %***** ELEMENT STIFFNESS MATRIX *****
104 function [KE]=lk
105 E = 1.;
106 nu = 0.3;
107 k=[ 1/2-nu/6 1/8+nu/8 -1/4-nu/12 -1/8+3*nu/8 ...
108 -1/4+nu/12 -1/8-nu/8 nu/6 1/8-3*nu/8];
109 KE = E/(1-nu^2)* [ k(1) k(2) k(3) k(4) k(5) k(6) k(7) k(8)
110 k(2) k(1) k(8) k(7) k(6) k(5) k(4) k(3)
111 k(3) k(8) k(1) k(6) k(7) k(4) k(5) k(2)
112 k(4) k(7) k(6) k(1) k(8) k(3) k(2) k(5)
113 k(5) k(6) k(7) k(8) k(1) k(2) k(3) k(4)
114 k(6) k(5) k(4) k(3) k(2) k(1) k(8) k(7)
115 k(7) k(4) k(5) k(2) k(3) k(8) k(1) k(6)
116 k(8) k(3) k(2) k(5) k(4) k(7) k(6) k(1)];

```

Figure 1





MATLAB R2021a

HOME PLOTS APPS EDITOR PUBLISH VIEW

New Open Save Find Files Compare Go To Find Insert Comment Indent Breakpoints Continue Step Run to Cursor Function Call Stack: FE Quit Debugging

FILE NAVIGATE EDIT BREAKPOINTS DEBUG

Current Folder: D:\matlab2.0\bin

Workspace: Command Window: Editor - topp4a.m

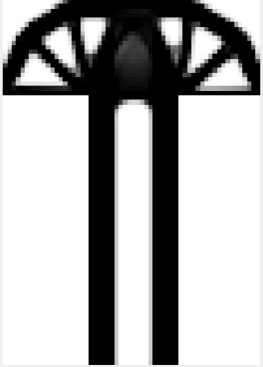
File Edit View Insert Tools Desktop Window Help

icutzdata m3iregistry ode programas util win32 win64 bulbas.jpg bulbasur.fig bulbasur.m charizard.fig charizard.m charizardddd.png crash_analyzer.cfg deploytool.bat ejem_gui.fig ejem_gui.m ejem_gui-pokedex.m grafica de señal cosenoidal con respecto al tiempo.fig hechizo.png inicial.fig inicial.m lcddata.xml lcddata.xsd lcddata_utf8.xml matlab.exe matlab_jenv.bat matlab - Acceso directo.lnk mbuild.bat mcc.bat mex.bat mexext.bat mw_mpiexec.bat pickaa.png pickachu.fig

```

34 - 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);
35 - c = c + x(ely,elx)^penal*Ue'*KE*Ue;
36 - dc(ely,elx) = dc(ely,elx)-penal*x(ely,elx)^(penal-1)* Ue'*KE*Ue;
37 - end
38 - end
39 - end
40 - % FILTERING OF SENSITIVITIES
41 - [dc] = check(nelx,nely,rmin,x,dc);
42 - % DESIGN UPDATE BY THE OPTIMALITY CRITERIA METHOD
43 - [x] = OC(nelx,nely,x,volfrac,dc,passive);
44 - % PRINT RESULTS
45 - change = max(max(abs(x-xold)));
46 - disp([' It.: ' sprintf('%4i',loop) ' Obj.: ' sprintf('%10.4f',c) ...
47 - 'Vol.: ' sprintf('%6.3f',sum(sum(x))/(nelx*nely)) ...
48 - ' ch.: ' sprintf('%6.3f',change )])
49 - % PLOT DENSITIES
50 - colormap(gray); imagesc(-x); axis equal; axis tight; axis off; pause(1e-6);
51 - end
52 - %***** OPTIMALITY CRITERIA UPDATE *****
53 - function [xnew]=OC(nelx,nely,x,volfrac,dc,passive)
54 - l1 = 0; l2 = 100000; move = 0.2;
55 - while (l2-l1 > 1e-4)
56 - lmid = 0.5*(l2+l1);
57 - xnew = max(0.001,max(x-move,min(1.,min(x+move,x.*sqrt(-dc./lmid)))));
58 - xnew(find(passive))==0.001;
59 - if sum(sum(xnew)) - volfrac*nelx*nely > 0.2
60 - l1 = lmid;
61 - else
62 - l2 = lmid;
63 - end
64 - end
65 - %***** MESH-INDEPENDENCY FILTER *****
66 - function [dcn]=check(nelx,nely,rmin,x,dc)
67 - dcn=zeros(nely,nelx);
68 - for i = 1:nely

```





UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FIME

FACULTAD DE INGENIERIA MECANICA Y
ELECTRICA



Conclusiones

Omar Gerardo Ríos Gaytán – 1902388

En lo que respecta a esta práctica, referente a las anteriores, se vio de nuevo el método de la mejora de la pieza con respecto a un método realizado con el software de Matlab. Ahora se trató de mejorar el diseño o la estructura del cableado de un teleférico, y como método de estructuración para su mejora se tuvo que buscar la relación de estos cables con respecto a su posicionamiento, peso de la cabina, longitud del cableado y material propio del cable. Para el desarrollo de la práctica se toma como referencia la estructuración interna del cableado, y se trató de lograr una estructura mucho más compleja que constituya de un apoyo y no solo como un cable lineal. El resultado satisfizo lo que buscábamos y se espera que pueda resistir lo mismo que el cableado común y corriente de los teleféricos.

Melissa Alejandra Jasso Maciel 1897337

Con esta práctica aprendimos que Matlab puede funcionar para solucionar algoritmos de elemento finito para optimizar estructuras complejas como la de un teleférico.

Mediante esta práctica, tomamos en cuenta el estudio con múltiples cargas y carga individual, y en consideración, cuáles son las implicaciones que esto conlleva.

Se puede observar que en los resultados de los casos propuestos se tiene una geométrica muy similar entre ellos. En el caso de dos cargas, este como las fuerzas son aplicadas en opuestos simétricos la forma de pieza es simétrica en el eje Y.

Carlos Antonio Caballero Padilla 1900864

En conclusión, a través de esta práctica logré cumplir de manera exitosa el propósito de esta, ya que se presentó una propuesta de análisis de forma con el propósito de diseño para reforzar el cable de un teleférico además de la programación realizada en el software Matlab para la ejecución de la optimización tomando así en cuenta las características de trabajo, las ventajas y desventajas de esta misma para un correcto diseño, análisis y simulación.

Edgar Alan Carrizales Treviño 1904406

Después de la realización de esta práctica pude comprender el análisis de este tipo de cables, pero aun así hubo varios problemas durante la realización del código ya que algunos comandos no funcionaban y habría que modificar nuevamente el código dando como resultado el que se obtuvieran dos optimizaciones, una en donde el código trabajaba bien con los comandos pertinentes y otro donde se modificaban dichos comandos porque no trabajaban de la manera más eficiente cambiándolos a una manera más útil en cuanto al objetivo pensado de la práctica.

Nestor Eliud Cano Garcia 1909644

Durante la investigación de esta práctica fue interesante ver los distintos tipos de factores que pueden alterar algo como lo puede ser un cable de teleférico, siendo que hay demasiados factores que pueden poner en riesgo la vida de las personas, siendo por eso la razón de que los cálculos y simulaciones que tenemos que hacer para este tipo de cosas no deben tener el

mínimo grado de error posible, ya que nosotros como ingenieros al igual que los médicos tenemos la vida de las personas en nuestras manos, de nosotros depende que las personas se encuentren a salvo, un fallo en un cálculo puede equivaler a decenas de vidas de personas inocentes que se perderán por un error. La práctica es interesante ya que si bien debemos de ver por la vida de las personas también hay que buscar un equilibrio entre hacer algo funcional, seguro y de un precio accesible ya que todos obviamente queremos seguridad, pero por desgracia el costo a veces es muy alto y para eso hay que aplicar distintas estrategias como utilizar la geometría para poder aprovechar al máximo nuestros recursos y buscar formas innovadoras para la solución de algún problema.

Raúl Emiliano Soto Salas 1864359

En esta práctica realizamos una propuesta de análisis de formas, para en este caso, un cable de un teleférico además de la programación adecuada para realizar una optimización de esta. Es interesante ver cómo es que hemos ido avanzando de optimizar cosas cotidianas como la estructura de una bicicleta, hasta llegar a lo que realizamos que es el cable de un teleférico, el cual es una pieza importante y cualquier error ya no solo podría afectar a una persona en algo no tan peligroso, si no a muchas personas como lo podría hacer un teleférico, que puede llegar a afectar algo más grave e incluso costar vidas, pero es por eso que se realizan numerosas pruebas para tener una optimización correcta y poder mejorar día con día todo lo que sea posible y tengamos a nuestro alcance.

Bibliografía

- M., J. (2021, 29 octubre). *Autor de la entrada: Javi M. TPC: Funiculares y teleféricos*. Recuperado 17 de octubre de 2022, de <https://transporteporcable.com/como-se-fabrican-los-cables-de-los-funiculares-y-telefericos/>
- De Camino a La Cima TV. (2019, 6 enero). *Cambio y trenzado del cable de un telesilla o remonte*. [Vídeo]. YouTube. Recuperado 17 de octubre de 2022, de <https://www.youtube.com/watch?v=JaNbOYpkSGU>
- Deutsche Welle (www.dw.com). (s. f.). *Teleférico / DW / 24.08.2022*. DW.COM. Recuperado 17 de octubre de 2022, de <https://www.dw.com/es/telef%C3%A9rico/t-41334573>
- *Cable de alambre para teleféricos (vías de cuerda aérea/cuerdas para montañas/telesquí/vías para materiales)*. (s. f.). Bekaert. Recuperado 17 de octubre de 2022, de <https://www.bekaert.com/es-MX/productos/productos-de-consumo/deporte-y-ocio/cable-de-alambre-para-telefericos-vias-de-cuerda-aerea-cuerdas-para-montanas-telesqui-vias-para-materiales>
- *Elementos de teleféricos - LEITNER*. (s. f.). Recuperado 17 de octubre de 2022, de <https://www.leitner.com/es/empresa/informaciones-utiles/elementos-de-telefericos/>