

Práctica 4

Refuerzo del cable de un teléferico

Equipo 4
Omar Gutierrez

28 de octubre de 2022

1. Introducción

El problema de Optimización Topológica (OT) consiste en buscar una distribución óptima de material en un dominio de diseño que satisfaga las solicitaciones y las condiciones de borde definidas. Generalmente, la OT ha sido formulada en términos de minimizar la energía de deformación de la estructura analizada.

En la mayoría de trabajos sobre OT reportados se encuentra que los métodos de optimización comúnmente empleados son: el criterio de Optimalidad Estándar (Optimality Criteria), el Método de la Curva de Nivel (Level Set Method) y la eficiencia de Pareto (Pareto Optimal Tracing), entre otros. De forma general, todas estas técnicas exploran el espacio de búsqueda de una manera controlada y tienen la ventaja de no depender del cálculo de derivadas para llevar a cabo el proceso de optimización. Entre estos, con el programa de 99 líneas que se selecciono para el estudio de una pieza definida, se empleó el criterio de Optimalidad Estándar (Optimality Criteria).

Nuevamente, en este trabajo se presenta el estado del arte necesario para abordar la optimización topológica pero en este caso en el refuerzo del cable de un teléferico y junto a esto la descripción del problema de optimización topológica y los resultados que se obtuvieron de la implementación del código.

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

El objetivo de esta práctica es presentar una propuesta de análisis de formas y de la programación para la ejecución de optimizar la estructura base de un teleférico que permitirá reforzar el cable lo máximo posible, esta estructura deberá de soportar dos cabinas del teleférico.

4. Estado del arte

Un teleférico es un aparato móvil cuyo principal objetivo es transportar a una persona desde un punto a hasta un punto b. Lo que lo diferencia entre el resto de medios de transporte es que, por lo general, se hace uso de este medio cuando los terrenos son muy difíciles de navegar como lo pueden ser acantilados, montañas o cañones [1].

El teleférico, principalmente se encuentra compuesto por las siguientes partes:

- Estaciones:

Cada estación es un ejemplar único gracias a su integración individual con la naturaleza y a las diferentes características del terreno y de la propia instalación. Por este motivo, existen diferentes tipos de estación, como por ejemplo las estaciones largas o cortas, la estación intermedia y la estación HCL.



Figura 1: Estaciones

- Grupo motor:

Dependiendo de las necesidades específicas del proyecto, la motriz se puede ubicar tanto en la estación inferior o superior, en configuración enterrada o aérea, en las variantes motriz fija o motriz tensora. Un grupo motor clásico está formado al menos por un motor, un freno de servicio, un freno de emergencia y el reductor de velocidad.



Figura 2: Grupo motor

- Línea:

Ningún otro elemento como la morfología del terreno es capaz de influir tan claramente en las características de la línea de un teleférico. En consecuencia, 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 [2].

- Pilonas:

Las pilonas deben soportar el peso del vehículo con los pasajeros y, por este motivo, su construcción es robusta. Los vehículos pueden circular por ambos lados de la pila. Las pilonas individuales están formadas por una combinación de tubos de acero de diferente longitud, diámetro y espesor de la pared.



Figura 3: Pilonas

- Balancines:

Los balancines sirven para dirigir el cable portante a lo largo de la línea. Cada balancín está compuesto por una disposición de poleas. El número de poleas depende del peso que el cable debe transportar. Cada polea está compuesta por un cuerpo base, el anillo y la rueda.



Figura 4: Balancines

- Cable:

El cable da nombre a todos los sistemas de transporte por cable, los teleféricos. Los cables de acero están compuestos de hilos de cable que se retuercen alrededor del núcleo del cable. Empresas especializadas fabrican los cables y los montan en el lugar.



Figura 5: Cable

- Pinza:

Hay dos tipos de pinzas que se pueden adaptar a las necesidades específicas del sistema correspondiente: la pinza fija y la desembragable. Las pinzas desembragables solo tienen un componente móvil, la mordaza de la pinza. La apertura y el cierre de la pinza se realiza de forma directa sin levas, articulaciones o sistemas de palanca. Esta pinza está compuesta por dos partes principales forjadas las cuales se fijan de forma permanente.



Figura 6: Pinza

5. Propuesta de diseño de la geometría, alcances y limitaciones

Para la propuesta de diseño de la geometría, planteamos un soporte doble que permitirá a la estructura llevar dos teleféricos a la vez, pero esto también perjudicará un poco al momento de optimizar ya que la pieza tiene que ser robusta, y sobre ella actuarán 2 fuerzas. Esta propuesta puede ser observada de manera gráfica en la figura 7.

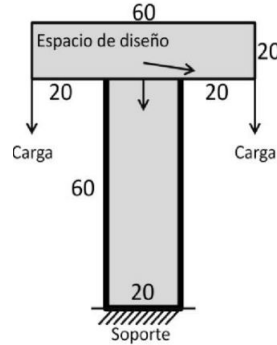


Figura 7: Espacio de diseño para dos cargas

6. Desarrollo de programación

Para poder realizar la optimización de la propuesta de diseño de la geometría se tienen que realizar cambios al programa modificado en la práctica anterior, originalmente este código de 99 líneas para optimización topológica fue obtenido de [3], para cada uno de los cambios realizados se puede visualizar en las siguientes líneas de texto:

- Declaración de vacío de la figura T:

```
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
```

- Declaración de fuerzas y soportes:

```
F(40,1) = -1.;
F(9760,2)=1.;
fixeddofs = 2*(nely+1):2*(nely+1):2*(nelx+1)*(nely+1);
alldofs = [1:2*(nely+1)*(nelx+1)];
freedofs = setdiff(alldofs, fixeddofs);
```

- Modificación de parámetros de función principal

```
top(60, 80, 0.33, 3, 1.5)
```

7. Resultados

Con las modificaciones realizadas en el código, antes de comenzar con la optimización topológica debemos asegurarnos que la figura fue generada correctamente, para esto en la línea de comando donde ingresamos los parámetros de la función principal aumentamos volfrac a 1, esa modificación no permitirá que el código realice las iteraciones y así nos permitirá ver la geometría.

Como podemos observar, la figura 8 cumple con la geometría propuesta, con lo cual ahora podemos realizar la optimización topológica.

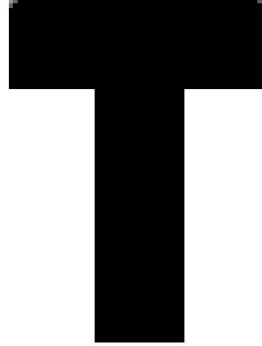


Figura 8: Geometría generada

Ahora para comenzar con la optimización topológica de la geometría generada solo basta con modificar el 1 por 0.33, esto permitirá ahora que se realicen las iteraciones correspondientes.

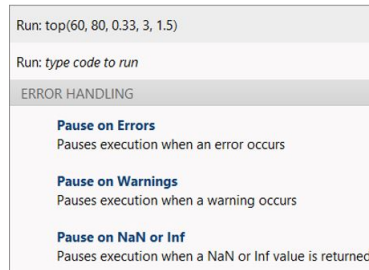
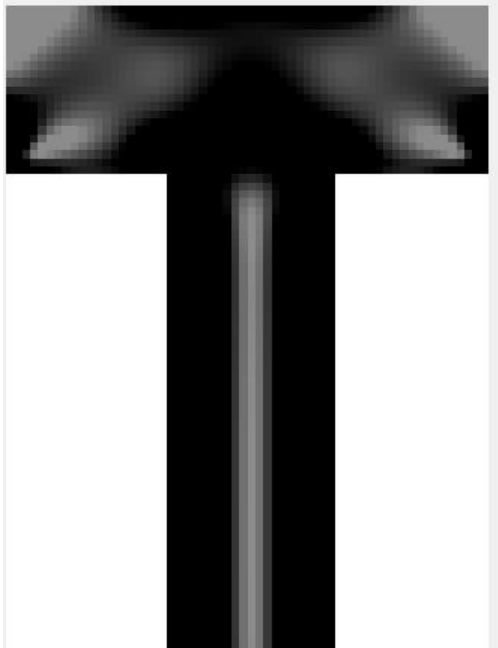
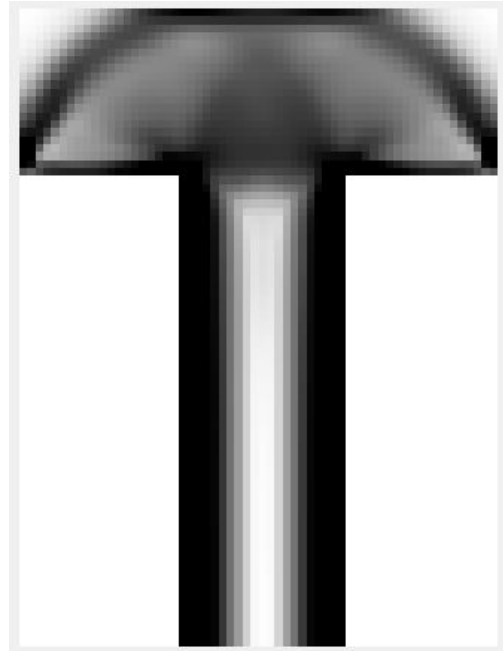


Figura 9: Comando modificado

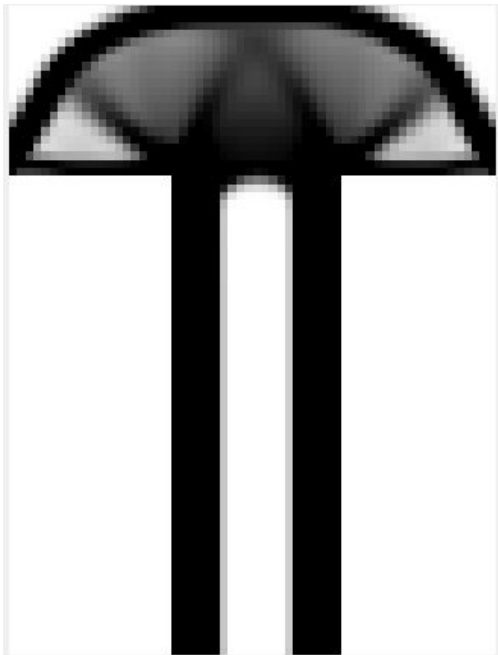
Realizando la operación, finalmente obtenemos la optimización de la geometría, como podemos observar en la figura 10a se muestra la secuencia de como el código propuesto fue determinando la distribución de masa más eficiente para cumplir exactamente con los requisitos de diseño impuestos a la pieza, 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.



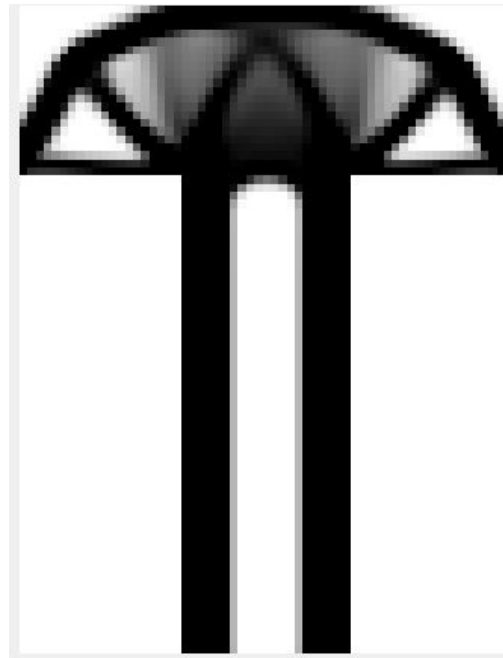
(a) 2da iteración



(b) 4ta iteración



(c) 11va iteración



(d) 20va iteración

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

```

%% A 99 LINE TOPOLOGY OPTIMIZATION CODE BY OLESIGMUND, OCTOBER 1999
function top(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>21
            if elx<21
                passive(ely,elx) = 1;
            elseif elx>41
                passive(ely,elx)=1;
            else
                passive(ely,elx) = 0;
            end
        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;
            dc(ely,elx)=0.;
            for i=1:2
                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
        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 %%%%%%%%%% OPTIMALITY CRITERIA UPDATE %%%%%%%%%%
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;
end

```

```

    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 %%%%%%%%%%
%65 %%%%%%%%%%
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),2); U = sparse(2*(nely+1)*(nelx+1),2);
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(40,1) = -1.;
F(9760,2)=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;
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)];

```


8. Conclusiones

- Omar Alejandro Gutierrez Carrizales :

Para esta práctica, se ha realizado un trabajo más complejo. Sin embargo, guarda cierta similitud con el panorámico debido a las cargas múltiples. Y se retoman conceptos que se han visto en prácticas anteriores. Pero, a pesar de ello, hubo algunas complicaciones, al momento de ejecutar el programa, debido al código tan extenso y, por ende, la posibilidad de equivocarse en algún carácter era mayor. De igual manera, la solicitud del planteamiento, la cual, era reforzar el cable de un teleférico, se puede realizar, reforzando las pilonas de acuerdo a su espacio de trabajo, haciendo uso del procedimiento de optimización topológica y el método de análisis de elementos finitos. De este modo, se logró llegar a un resultado óptimo.

Referencias

- [1] CurioSfera Ciencia. Qué es un teleférico. Recuperado de: <https://curiosfera-ciencia.com/que-es-un-teleferico-como-funciona/>, 2022.
- [2] LEITNER. Elementos de teleféricos. Recuperado de: <https://www.leitner.com/es/empresa/informaciones-utiles/elementos-de-telefericos/>, 2022.
- [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.