

Práctica 5

Optimización de una prótesis de pie

Equipo 4

Jorge Fuentes, Tania Hernandez, Anahi Herrera, Gustavo Díaz, Miriam Mata, Alejandro Ramos

12 de noviembre de 2022

Resumen

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 tecnología en las prótesis ha sido objeto de una transformación sustancial en las últimas décadas, sobre todo con la introducción de materiales y mecanismos elásticos pero también en la geometría, la masa, la alineación. Con el pasar de los años y el avance tecnológico la mayor parte de las prótesis contienen rasgos más humanos, obteniendo múltiples ventajas para proporcionar una segunda oportunidad a personas que han perdido una de sus extremidades. Por esta razón el documento trata sobre el estudio de los diferentes avances obtenidos hasta la actualidad, los cuales brindan un mejor estilo de vida a la persona.

Índice

1. Introducción	3
2. Desarrollo	3
2.1. Nombre y definición de la forma geometría	3
2.1.1. Prótesis de pie	3
2.1.2. Prótesis Biónica	3
2.1.3. Prótesis Sach	4
2.2. Estado del arte	5
2.2.1. Materiales de las prótesis de pie	5
2.3. Propuesta de la geometría, alcances y limitaciones.	6
2.3.1. Alcances y limitaciones	7
2.4. Pasos del desarrollo de la programación	8
2.5. Resultados de la optimización	14
3. Conclusiones	17
3.1. Jorge Fuentes	17
3.2. Tania Hernandez	17
3.3. Anahi Herrera	17
3.4. Gustavo Díaz	17
3.5. Miriam Mata	17
3.6. Alejandro Ramos	17

1. Introducción

Las prótesis de pie se tratan de aparatos médicos destinados a sustituir de forma artificial un pie, tobillo o parte del pie faltante en el cuerpo de un paciente a causa de una amputación total o parcial del miembro.

Un pie protésico se trata de un aparato médico diseñado justo a la medida de cada paciente para suplantar estética y funcionalmente la parte faltante de la extremidad a causa de una amputación.

Esta tiene la función tanto de aparentar que existe la sección faltante como de ser un punto de apoyo para que el paciente pueda caminar, utilizar calzado y desempeñar sus labores de forma normal o lo más normal posible.

Las prótesis de pie tienen costos personalizados que dependen de las necesidades de cada paciente, el nivel de amputación y de los materiales y componentes mecánicos que tengan como aditamentos para el movimiento.

2. Desarrollo

2.1. Nombre y definición de la forma geometría

2.1.1. Prótesis de pie

Es común al momento sufrir una amputación, el reemplazar el miembro por una extremidad artificial; una prótesis. La selección de una prótesis es altamente dependiente de las necesidades y capacidades de cada el paciente. Su función puede variar de puramente estética a una necesidad funcional para un paciente que desea recuperar la independencia en realizar actividades de la vida diaria, o ya sea en el campo deportivo.

La tecnología en las prótesis ha sido objeto de una transformación sustancial en las últimas décadas, sobre todo con la introducción de materiales y mecanismos elásticos pero también en la geometría, la masa, la alineación. Con el pasar de los años y el avance tecnológico la mayor parte de las prótesis contienen rasgos más humanos, obteniendo múltiples ventajas para proporcionar una segunda oportunidad a personas que han perdido una de sus extremidades[3]. Por esta razón el documento trata sobre el estudio de los diferentes avances obtenidos hasta la actualidad, los cuales brindan un mejor estilo de vida a la persona.

2.1.2. Prótesis Biónica

Este tipo de pies protésicos deben contener múltiples adaptaciones y propiedades de rigidez al caminar, Además, de proporcionan un alto torque a la salida. Durante la fase de impulsión, la posición de la articulación del tobillo debe ser controlado con el fin de evitar la caída del pie en contacto con el talón. Esta es un prototipo de una TT (PPAMs) "Prótesis de pie con músculos artificiales". El prototipo TT está equipada con tres PPAMs; uno se coloca en frente, dos se colocan en la parte posterior y su trabajo es de forma paralela. Una gama de tobillo se establece de movimiento de 30 grados. La prótesis consiste tanto en una rodilla articulada y la articulación del tobillo. Dos cilindros neumáticos se utilizan para tener la extremidad inferior de la prótesis completa. Este es un prototipo de una TT (PPAMs).

2.1.3. Prótesis Sach

El pie SACH no soporta mucha carga en la zona de los dedos. Esta falta de apoyo del dedo del pie crea una experiencia de "drop-off". El pie SACH está hecho de una quilla de madera cubierto por espuma de poliuretano[1]. Este pie se caracteriza por ser rígido en el región de madera (media del pie), y por ser bastante flexible en la espuman (zona de los dedos). Estas características se reflejan en el roll-over forma obtenida por el PFLA Una concha de pie polietileno reforzado se usa como un cosmético cubierto. Esta estructura produce una constante, sobre la forma del talón al dedo del pie, lo que se traduce en una constante, el apoyo continuo de peso. Los PFLA está diseñado de modo que una fuerza aplicada al final de la viga se duplica más o menos en el punto de contacto con el pie protésico. Este punto de contacto es proporcionado por un diseño personalizado de la placa aluminio.

Son algunas de las tipos de prótesis investigadas, respecto a los tipos de forma que pueden existir dependiendo de las necesidades del paciente.

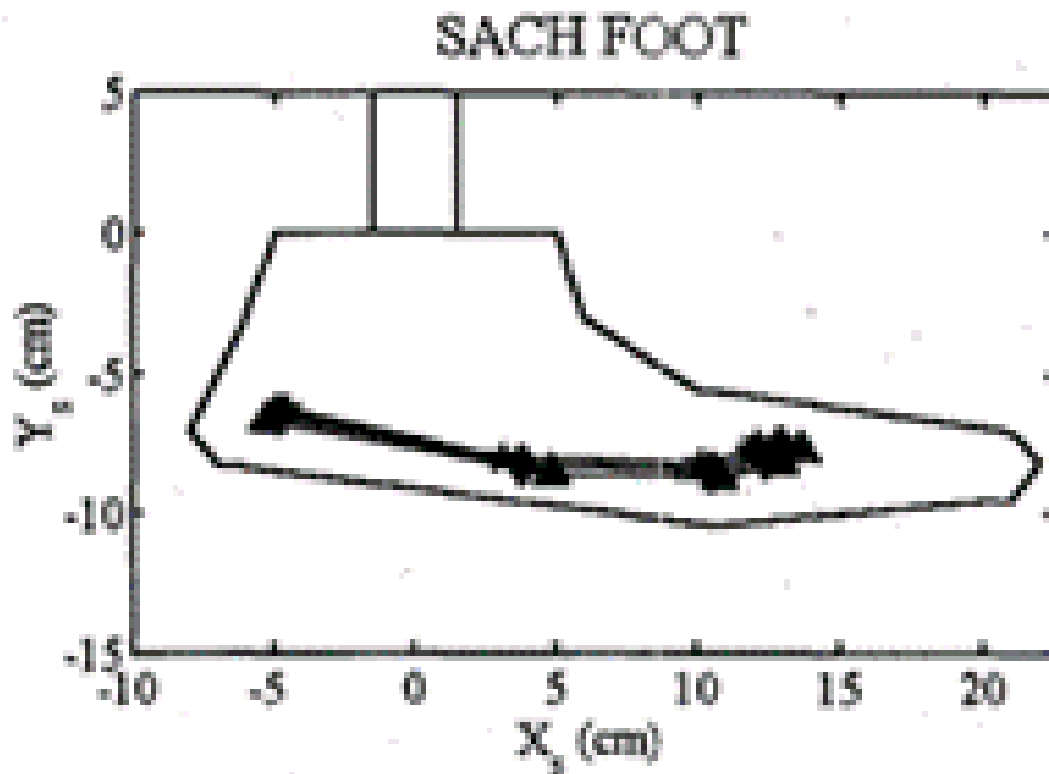


Figura 1: Prótesis Sach.

2.2. Estado del arte

Las prótesis de pie se tratan de aparatos médicos destinados a sustituir de forma artificial un pie, tobillo o parte del pie faltante en el cuerpo de un paciente a causa de una amputación total o parcial del miembro. Estos aparatos protéticos tienen tanto finalidades funcionales para la rehabilitación del paciente como funciones estéticas sumamente avanzadas para imitar de forma bastante realista la parte del cuerpo faltante. Los pies son una de las extremidades más importantes y básicas no sólo para poder caminar, sino para simplemente poder sostenerse parado y tener el equilibrio necesario para todo tipo de actividades, por lo que las prótesis de pie y tobillo son de las más importantes y avanzadas en casos de amputación por accidentes, enfermedades o agenesia[2]. Actualmente existe una enorme variedad de prótesis según las necesidades y características de cada persona, así como el tipo de amputación y muñón, por lo que es importante que las conozcas para determinar cuál es la mejor para ti como paciente o para alguna persona conocida que la necesite.

Dependiendo del tipo de amputación, la prótesis podrá estar compuesta de diversos elementos, como lo pueden ser:

- Socket o encaje con el muñón.
- Plantillas.
- Almohadillas.
- Estabilizadores para el talón.
- Elementos estéticos.

De igual manera, existen prótesis mucho más avanzadas que pueden tener dispositivos y elementos robóticos, mecánicos, mioeléctricos y biomecánicos para otorgar movimientos especiales en caso de ser necesario.

2.2.1. Materiales de las prótesis de pie

El pie protético puede ser fabricado de uno o más materiales, los cuales son seleccionados para brindar el soporte, agarre y diseño necesario para suplantar la parte faltante del cuerpo humano. Por lo general se utiliza:

- Silicona médica.
- Plantillas de carbono.
- Rellenos de materiales elásticos.
- Aluminio.
- Titanio.
- Acero inoxidable.
- Plásticos.

2.3. Propuesta de la geometría, alcances y limitaciones.

Después de un análisis de esfuerzos, costos de fabricación, facilidad de construcción, ventajas y desventajas que presentaban los prediseños realizados de la prótesis de pie se llegó a un diseño final que tiene como ventaja un análisis de ingeniería en cuanto a sus características físicas y mecánicas para darle un mayor confort al usuario y poder garantizar su perfecto funcionamiento.

Para el diseño de cada una de las piezas que conforman la prótesis del pie se tuvo en cuenta algunos diseños realizados por compañías dedicadas exclusivamente a la fabricación de prótesis con el fin de hacer que la prótesis fuera modular y así poder adaptarse fácilmente a las prótesis existentes en el mercado.

Para poder realizar nuestra simulación se basó en la información siguiente: Se realizó un primer boceto y basados en éste se empezaron a realizar modificaciones, para que fuera más ergonómica y funcional. A partir del diseño básico (siguiente figura) se empezaron a analizar las dimensiones pertinentes teniendo en cuenta los parámetros de antropometría del pie, según las dimensiones del grupo objetivo y la forma de los huesos del pie de acuerdo con el esqueleto humano.

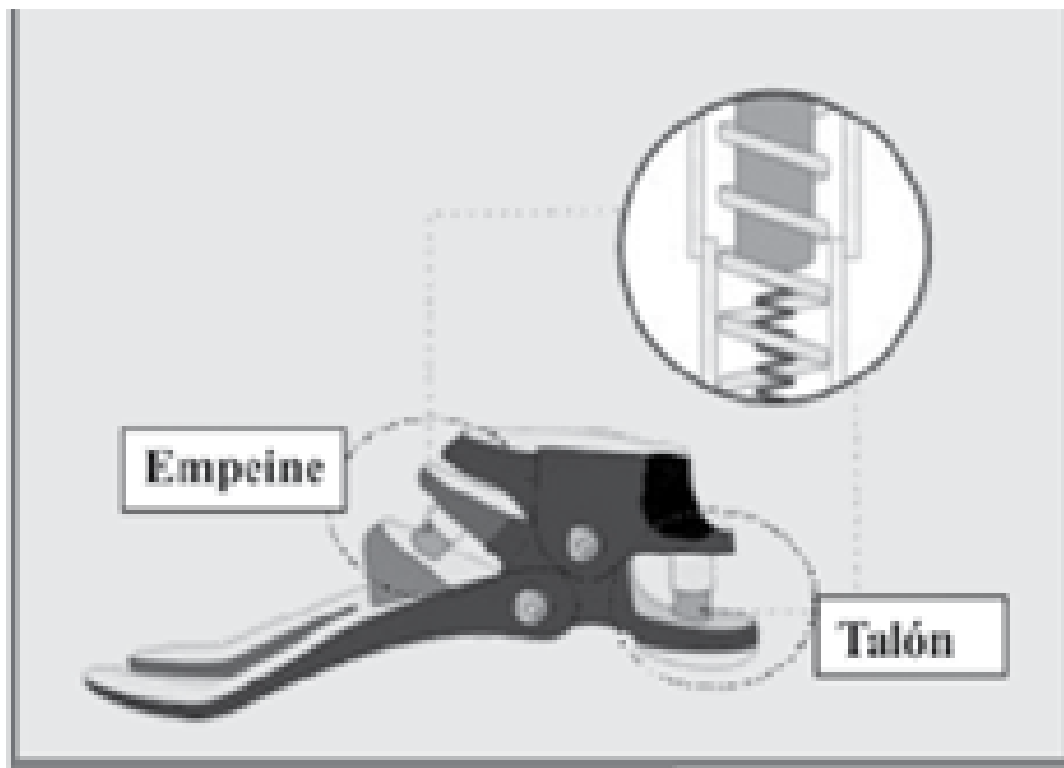


Figura 2: Primer boceto del pie.

A partir de este boceto se empezaron a realizar cambios sustanciales en el diseño, para poder suplir las necesidades y, de igual manera, cumplir con los objetivos planteados. Para el segundo boceto se redujo la abertura de la planta y se quitó el doble sistema de amortiguación dejando solamente el resorte de nitinol con una sola articulación a nivel del tobillo (siguiente figura).

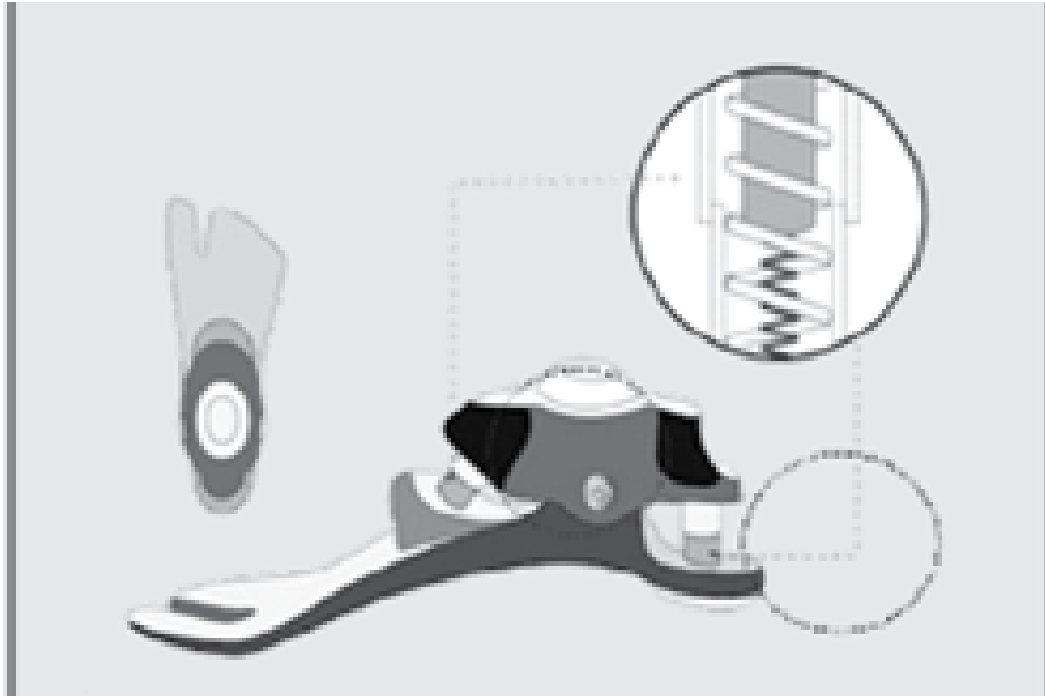


Figura 3: Segundo boceto del pie.

2.3.1. Alcances y limitaciones

Alcances:

- Diseño, simulación y análisis de la prótesis de pie sin llegar a la construcción de un prototipo físico.
- En el proceso de selección de materiales, se buscará un diseño con materiales económicos que permitan cumplir con las exigencias del diseño.

Limitaciones:

- Todos los cálculos se realizarán tomando como base la marcha en un plano horizontal.
- Se desarrollará un diseño para adultos tomando como referencia los estándares físicos masculinos de la población colombiana.
- Para el análisis, se tendrán en cuenta que la prótesis de pie solo tiene un grado de libertad.

2.4. Pasos del desarrollo de la programación

A continuación, veremos la codificación de la programación en MATLAB.

```

1 %%%% A 99 LINE TOPOLOGY OPTIMIZATION CODE BY OLESIGMUND, OCTOBER 1999 %%%
2 function topp5a(nelx,nely,volfrac,penal,rmin);
3 % INITIALIZE
4 x(1:nely,1:nelx) = volfrac;
5 loop = 0;
6 change = 1.;
7 % START ITERATION
8 while change > 0.01
9     loop = loop + 1;
10    xold = x;
11    % FE-ANALYSIS
12    [U]=FE(nelx,nely,x,penal);
13    % OBJECTIVE FUNCTION AND SENSITIVITY ANALYSIS
14    [KE] = lk;
15    c = 0.;
16    for ely = 1:nely
17        for elx = 1:nelx
18            n1 = (nely+1)*(elx-1)+ely;
19            n2 = (nely+1)* elx +ely;
20            dc(ely,elx)=0.;
21            for i=1:5
22                Ue = U([2*n1-1;2*n1; 2*n2-1;2*n2; 2*n2+1; 2*n2+2;
23                    2*n1+1;2*n1+2],i);
24                c = c + x(ely,elx)^penal*Ue'*KE*Ue;
25                dc(ely,elx) = dc(ely,elx)-penal*x(ely,elx)^(penal-1)*Ue'*KE*Ue;
26            end
27        end
28    end
29    % FILTERING OF SENSITIVITIES
30    [dc] = check(nelx,nely,rmin,x,dc);
31    % DESIGN UPDATE BY THE OPTIMALITY CRITERIA METHOD
32    [x] = OC(nelx,nely,x,volfrac,dc);
33    % PRINT RESULTS
34    change = max(max(abs(x-xold)));
35    disp(['It.: ' sprintf('%4i',loop) 'Obj.: ' sprintf('%10.4f',c) ...
36        ' Vol.: ' sprintf('%6.3f',sum(sum(x))/(nelx*nely)) ...
37        ' ch.: ' sprintf('%6.3f',change)]])
38    % PLOT DENSITIES
39    colormap(gray); imagesc(-x); axis equal; axis tight;
40 end
41 %%%%%%%%%% OPTIMALITY CRITERIA UPDATE %%%%%%%%%%
42 function [xnew]=OC(nelx,nely,x,volfrac,dc)
43 l1 = 0; l2 = 100000; move = 0.2;
44 while (l2-l1 > 1e-4)
45     lmid = 0.5*(l2+l1);
46     xnew = max(0.001,max(x-move,min(1.,min(x+move,x.*sqrt(-dc./lmid)))));
47     if sum(sum(xnew)) - volfrac*nelx*nely > 0;
48         l1 = lmid;
49     else
50         l2 = lmid;
51     end
52 end
53 %%%%%%%%%% MESH-INDEPENDENCY FILTER %%%%%%%%%%
54 function [dcn]=check(nelx,nely,rmin,x,dc)
55 dcn=zeros(nely,nelx);
56 for i = 1:nelx
57     for j = 1:nely
58         sum=0.0;
59         for k = max(i-round(rmin),1):min(i+round(rmin),nelx)
60             for l = max(j-round(rmin),1):min(j+round(rmin),nely)
61                 fac = rmin-sqrt((i-k)^2+(j-l)^2);
62                 sum = sum+max(0,fac);
63                 dcn(j,i) = dcn(j,i) + max(0,fac)*x(l,k)*dc(l,k);
64             end

```



```

65         end
66         dcn(j,i) = dcn(j,i)/(x(j,i)*sum);
67     end
68 end
69 %%%%%%%%%%% FE-ANALYSIS %%%%%%%%%%%
70 function [U]=FE(nelx,nely,x,penal)
71 [KE] = lk;
72 K = sparse(2*(nelx+1)*(nely+1), 2*(nelx+1)*(nely+1));
73 F = sparse(2*(nely+1)*(nelx+1),5); U = sparse(2*(nely+1)*(nelx+1),5);
74 for ely = 1:nely
75     for elx = 1:nelx
76         n1 = (nely+1)*(elx-1)+ely;
77         n2 = (nely+1)* elx +ely;
78         edof = [2*n1-1; 2*n1; 2*n2-1; 2*n2; 2*n2+1;2*n2+2;2*n1+1; 2*n1+2];
79         K(edof,edof) = K(edof,edof) + x(ely,elx)^penal*KE;
80     end
81 end
82 % DEFINE LOADS AND SUPPORTS (HALF MBB-BEAM)
83 F(3222,1) = -1;
84 F(3782,2) = -1;
85 F(2662,3) = -1;
86 F(2942,4) = -1;
87 F(3502,5) = -1;
88 fixeddofs = union([560:2*(nely+1):1260],[3920:2*(nely+1):4620]);
89 alldofs = [1:2*(nely+1)*(nelx+1)];
90 freedofs = setdiff(alldofs,fixeddofs);
91 % SOLVING 127
92 U(freedofs,:) = K(freedofs,freedofs) \ F(freedofs,:);
93 U(fixeddofs,:) = 0;
94 %%%%%%%%%%% ELEMENT STIFFNESS MATRIX %%%%%%%%%%%
95 function [KE]=lk
96 E = 1.;
97 nu = 0.3;
98 k=[ 1/2-nu/6 1/8+nu/8 -1/4-nu/12 -1/8+3*nu/8 ...
99     -1/4+nu/12 -1/8-nu/8 nu/6 1/8-3*nu/8];
100 KE = E/(1-nu^2)*[ k(1) k(2) k(3) k(4) k(5) k(6) k(7) k(8)
101     k(2) k(1) k(8) k(7) k(6) k(5) k(4) k(3)
102     k(3) k(8) k(1) k(6) k(7) k(4) k(5) k(2)
103     k(4) k(7) k(6) k(1) k(8) k(3) k(2) k(5)
104     k(5) k(6) k(7) k(8) k(1) k(2) k(3) k(4)
105     k(6) k(5) k(4) k(3) k(2) k(1) k(8) k(7)
106     k(7) k(4) k(5) k(2) k(3) k(8) k(1) k(6)
107     k(8) k(3) k(2) k(5) k(4) k(7) k(6) k(1)];

```

```

1 %%%% A 99 LINE TOPOLOGY OPTIMIZATION CODE BY OLESGMUND, OCTOBER 1999 %%%
2 function topp5b(nelx ,nely ,volfrac ,penal ,rmin);
3 % INITIALIZE
4 x(1:nely,1:nelx) = volfrac;
5 loop = 0;
6 change = 1.;
7 % START ITERATION
8 while change > 0.01
9     loop = loop + 1;
10    xold = x;
11    % FE-ANALYSIS
12    [U]=FE(nelx ,nely ,x ,penal);
13    % OBJECTIVE FUNCTION AND SENSITIVITY ANALYSIS
14    [KE] = lk
15    c = 0.;
16    for ely = 1:nely
17        for elx = 1:nelx
18            n1 = (nely+1)*(elx-1)+ely;
19            n2 = (nely+1)* elx +ely;
20            dc(ely ,elx)=0.;
21            for i=1:5
22                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);
23                c = c + x(ely ,elx)^penal*Ue'*KE*Ue;
24                dc(ely ,elx) = dc(ely ,elx)-penal*x(ely ,elx)^(penal-1)*Ue'*KE*Ue;
25            end
26        end
27    end
28    % FILTERING OF SENSITIVITIES
29    [dc] = check(nelx ,nely ,rmin ,x ,dc);
30    % DESIGN UPDATE BY THE OPTIMALITY CRITERIA MEIHOD
31    [x] = OC(nelx ,nely ,x ,volfrac ,dc);
32    % PRINT RESULTS
33    change = max(max(abs(x-xold)));
34    disp(['It.: ' sprintf('%4i',loop) 'Obj.: ' sprintf('%10.4f',c) ...
35        ' Vol.: ' sprintf('%6.3f',sum(sum(x))/(nelx*nely)) ...
36        ' ch.: ' sprintf('%6.3f',change)]])
37    % PLOT DENSITIES
38    colormap(gray); imagesc(-x); axis equal; axis tight; axis off; pause(1e6);
39 end
40 %%%%%%%%%% OPTIMALITY CRITERIA UPDATE %%%%%%%%%%
41 function [xnew]=OC(nelx ,nely ,x ,volfrac ,dc)
42 l1 = 0; l2 = 100000; move = 0.2;
43 while (l2-l1 > 1e-4)
44     lmid = 0.5*(l2+l1);
45     xnew = max(0.001,max(x-move,min(1.,min(x+move,x.*sqrt(-dc./lmid)))));
46     if sum(sum(xnew)) - volfrac*nelx*nely > 0;
47         l1 = lmid;
48     else
49         l2 = lmid;
50     end
51 end
52 %%%%%%%%%% MESH-INDEPENDENCY FILTER %%%%%%%%%%
53 function [dcn]=check(nelx ,nely ,rmin ,x ,dc)
54 dcn=zeros(nely ,nelx);
55 for i = 1:nelx
56     for j = 1:nely
57         sum=0.0;
58         for k = max(i-round(rmin),1):min(i+round(rmin),nelx)
59             for l = max(j-round(rmin),1):min(j+round(rmin), nely)
60                 fac = rmin-sqrt((i-k)^2+(j-l)^2);
61                 sum = sum+max(0,fac);
62                 dcn(j,i) = dcn(j,i) + max(0,fac)*x(l,k)*dc(l,k);
63             end
64         end
65         dcn(j,i) = dcn(j,i)/(x(j,i)*sum);
66     end
67 end
68 %%%%%%%%%% FE-ANALYSIS %%%%%%%%%%

```

```

69 function [U]=FE(nelx ,nely ,x ,penal)
70 [KE] = lk;
71 K = sparse(2*(nelx+1)*(nely+1), 2*(nelx+1)*(nely+1));
72 F = sparse(2*(nely+1)*(nelx+1),5); U =sparse(2*(nely+1)*(nelx+1),5);
73 for ely = 1:nely
74     for elx = 1:nelx
75         n1 = (nely+1)*(elx-1)+ely;
76         n2 = (nely+1)* elx +ely;
77         edof = [2*n1-1; 2*n1; 2*n2-1; 2*n2; 2*n2+1;2*n2+2;2*n1+1; 2*n1+2];
78         K(edof,edof) = K(edof,edof) + x(ely ,elx)^penal*KE;
79     end
80 end
81 % DEFINE LOADSAND SUPPORTS(HALF MBB-BEAM)
82 F(3222,1) = -1;
83 F(3782,2) = -1;
84 F(2662,3) = -1;
85 F(2942,4) = -1;
86 F(3502,5) = -1;
87 fixeddofs = [3920:2*(nely+1):4620];
88 alldofs = [1:2*(nely+1)*(nelx+1)];
89 freedofs = setdiff(alldofs ,fixeddofs);
90 % SOLVING 127
91 U(freedofs,:) = K(freedofs ,freedofs) \F(freedofs ,:);
92 U(fixeddofs,:) = 0;
93 %%%%%%%%%%ELEMENT STIFFNESS MATRIX %%%%%%%%%%
94 function [KE]=lk
95 E = 1.;
96 nu = 0.3;
97 k=[ 1/2-nu/6 1/8+nu/8 -1/4-nu/12 -1/8+3*nu/8 ...
98     -1/4+nu/12 -1/8-nu/8 nu/6 1/8-3*nu/8];
99 KE = E/(1-nu^2)*[ k(1) k(2) k(3) k(4) k(5) k(6) k(7) k(8)
100     k(2) k(1) k(8) k(7) k(6) k(5) k(4) k(3)
101     k(3) k(8) k(1) k(6) k(7) k(4) k(5) k(2)
102     k(4) k(7) k(6) k(1) k(8) k(3) k(2) k(5)
103     k(5) k(6) k(7) k(8) k(1) k(2) k(3) k(4)
104     k(6) k(5) k(4) k(3) k(2) k(1) k(8) k(7)
105     k(7) k(4) k(5) k(2) k(3) k(8) k(1) k(6)
106     k(8) k(3) k(2) k(5) k(4) k(7) k(6) k(1)];

```

```

1 %%%% A 99 LINE TOPOLOGY OPTIMIZATION CODE BY OLESGMUND, OCTOBER 1999 %%%
2 function topp(nelx,nely,volfrac,penal,rmin);
3 % INITIALIZE
4 x(1:nely,1:nelx) = volfrac;
5 loop = 0;
6 change = 1.;
7 % START ITERATION
8 while change > 0.01
9 loop = loop + 1;
10 xold = x;
11 % FE-ANALYSIS
12 [U]=FE(nelx,nely,x,penal);
13 % OBJECTIVE FUNCTION AND SENSITIVITY ANALYSIS
14 [KE] = lk;
15 c = 0.;
16 for ely = 1:nely
17 for elx = 1:nelx
18 n1 = (nely+1)*(elx-1)+ely;
19 n2 = (nely+1)* elx +ely;
20 dc(ely,elx)=0.;
21 for i=1:5
22 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);
23 c = c + x(ely,elx)^penal*Ue'*KE*Ue;
24 dc(ely,elx) = dc(ely,elx)-penal*x(ely,elx)^(penal-1)*Ue'*KE*Ue;
25 end
26 end
27 end
28 % FILTERING OF SENSITIVITIES
29 [dc] = check(nelx,nely,rmin,x,dc);
30 % DESIGN UPDATE BY THE OPTIMALITY CRITERIA METHOD
31 [x] = OC(nelx,nely,x,volfrac,dc);
32 % PRINT RESULTS
33 change = max(max(abs(x-xold)));
34 disp(['It.: ' sprintf('%4i',loop) 'Obj.: ' sprintf('%10.4f',c) ...
35 ' Vol.: ' sprintf('%6.3f',sum(sum(x))/(nelx*nely)) ...
36 ' ch.: ' sprintf('%6.3f',change)]])
37 % PLOT DENSITIES
38 colormap(gray); imagesc(-x); axis equal; axis tight; axis off;pause(1e6);
39 end
40 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
41 function [xnew]=OC(nelx,nely,x,volfrac,dc)
42 l1 = 0; l2 = 100000; move = 0.2;
43 while (l2-l1 > 1e-4)
44 lmid = 0.5*(l2+l1);
45 xnew = max(0.001,max(x-move,min(1.,min(x+move,x.*sqrt(-dc./lmid))));
46 if sum(sum(xnew)) - volfrac*nelx*nely > 0;
47 l1 = lmid;
48 else
49 l2 = lmid;
50 end
51 end
52 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
53 function [dcn]=check(nelx,nely,rmin,x,dc)
54 dcn=zeros(nely,nelx);
55 for i = 1:nelx
56 for j = 1:nely
57 sum=0.0;
58 for k = max(i-round(rmin),1):min(i+round(rmin),nelx)
59 for l = max(j-round(rmin),1):min(j+round(rmin),nely)
60 fac = rmin-sqrt((i-k)^2+(j-l)^2);
61 sum = sum+max(0,fac);
62 dcn(j,i) = dcn(j,i) + max(0,fac)*x(l,k)*dc(l,k);
63 end
64 end
65 dcn(j,i) = dcn(j,i)/(x(j,i)*sum);
66 end
67 end
68 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FE-ANALYSIS %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

69 function [U]=FE(nelx ,nely ,x ,penal)
70 [KE] = lk;
71 K = sparse(2*(nelx+1)*(nely+1), 2*(nelx+1)*(nely+1));
72 F = sparse(2*(nely+1)*(nelx+1),5); U =sparse(2*(nely+1)*(nelx+1),5);
73 for ely = 1:nely
74 for elx = 1:nelx
75 n1 = (nely+1)*(elx-1)+ely;
76 n2 = (nely+1)* elx +ely;
77 edof = [2*n1-1; 2*n1; 2*n2-1; 2*n2; 2*n2+1;2*n2+2;2*n1+1; 2*n1+2];
78 K(edof,edof) = K(edof,edof) + x(ely ,elx)^penal*KE;
79 end
80 end
81 % DEFINE LOADSAND SUPPORTS(HALF MBB-BEAM)
82 F(3222,1) = -1;
83 F(3782,2) = -1;
84 F(2662,3) = -1;
85 F(2942,4) = -1;
86 F(3502,5) = -1;
87 fixeddofs = [560:2*(nely+1):1260];
88 alldofs = [1:2*(nely+1)*(nelx+1)];
89 freedofs = setdiff(alldofs ,fixeddofs);
90 % SOLVING 127
91 U(freedofs,:) = K(freedofs ,freedofs) \F(freedofs ,:);
92 U(fixeddofs,:) = 0;
93 %%%%%%%%%%ELEMENT STIFFNESS MATRIX %%%%%%%%%%
94 function [KE]=lk
95 E = 1.;
96 nu = 0.3;
97 k=[ 1/2-nu/6 1/8+nu/8 -1/4-nu/12 -1/8+3*nu/8 ...
98 -1/4+nu/12 -1/8-nu/8 nu/6 1/8-3*nu/8];
99 KE = E/(1-nu^2)*[ k(1) k(2) k(3) k(4) k(5) k(6) k(7) k(8)
100 k(2) k(1) k(8) k(7) k(6) k(5) k(4) k(3)
101 k(3) k(8) k(1) k(6) k(7) k(4) k(5) k(2)
102 k(4) k(7) k(6) k(1) k(8) k(3) k(2) k(5)
103 k(5) k(6) k(7) k(8) k(1) k(2) k(3) k(4)
104 k(6) k(5) k(4) k(3) k(2) k(1) k(8) k(7)
105 k(7) k(4) k(5) k(2) k(3) k(8) k(1) k(6)
106 k(8) k(3) k(2) k(5) k(4) k(7) k(6) k(1)];

```

2.5. Resultados de la optimización

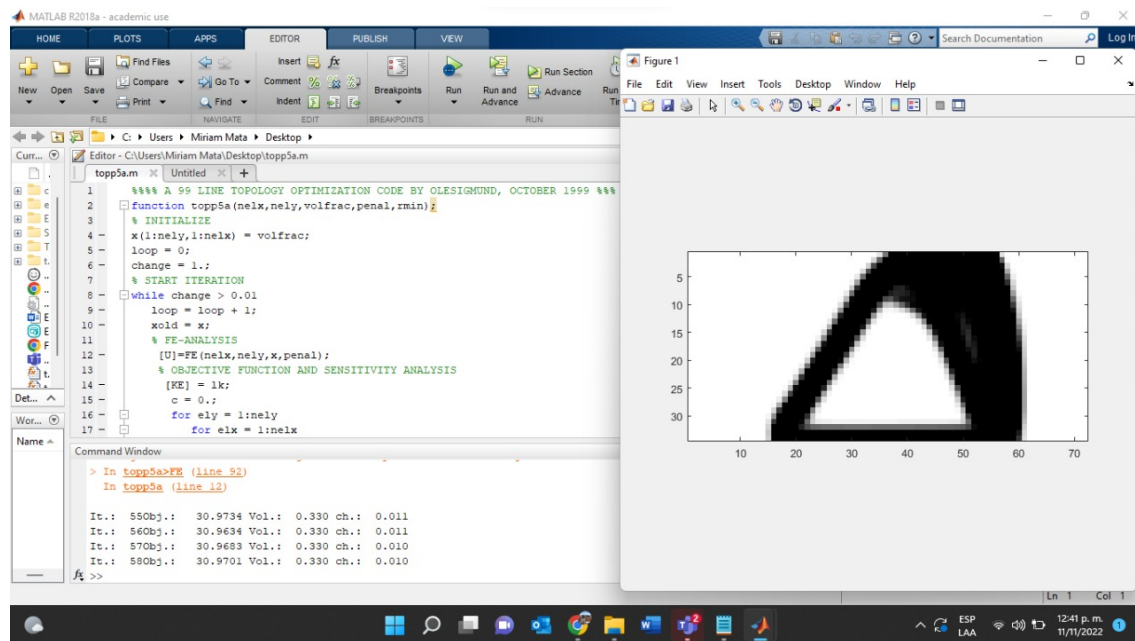
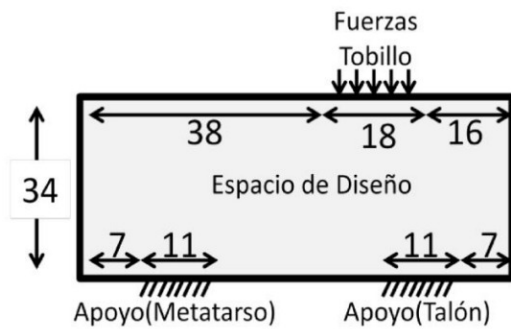
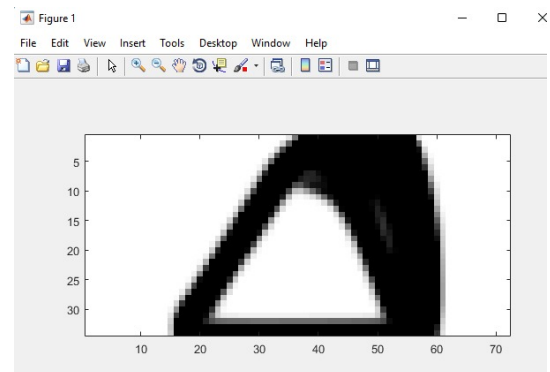


Figura 4: Vista completa del resultado del código A.



(a)



(b)

Figura 5: F1 posición normal y su resultado.

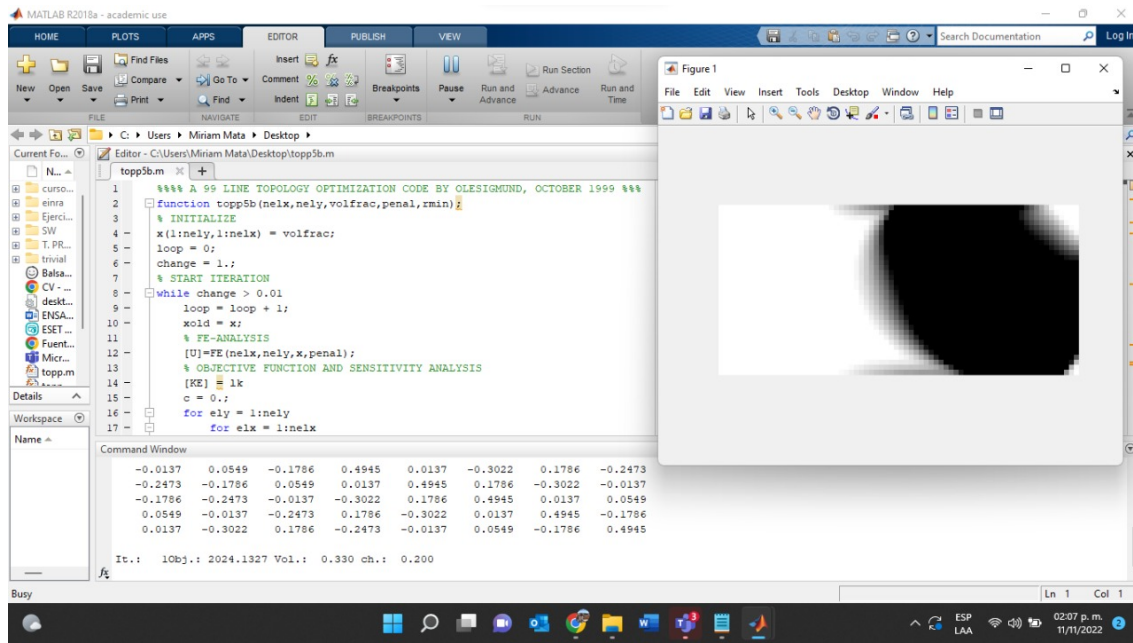
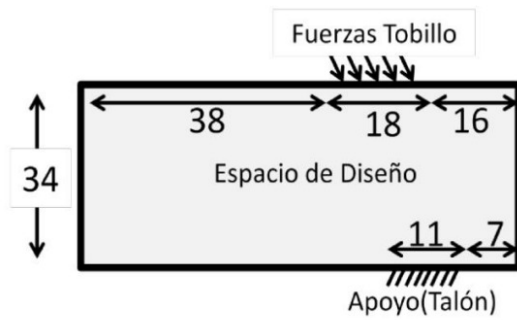
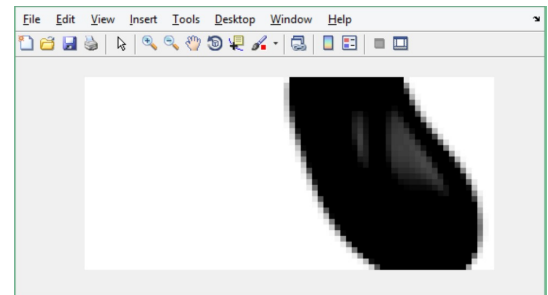


Figura 6: Vista completa del resultado del código B.



(a)



(b)

Figura 7: F2 posición de despegue y su resultado.

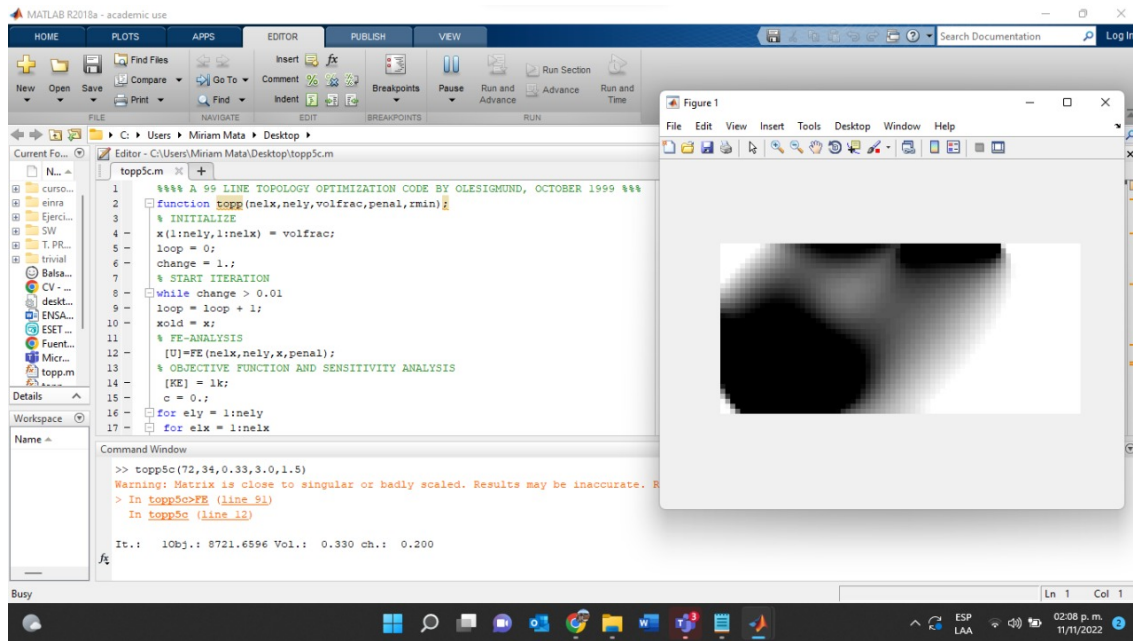


Figura 8: Vista completa del resultado del código C.

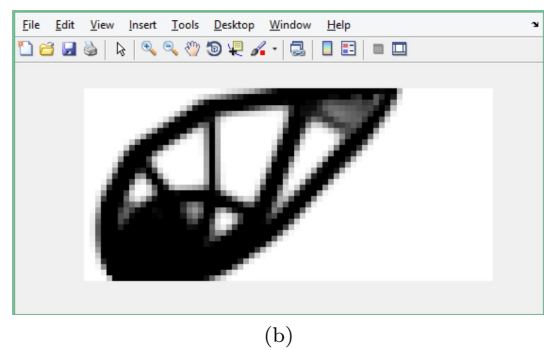
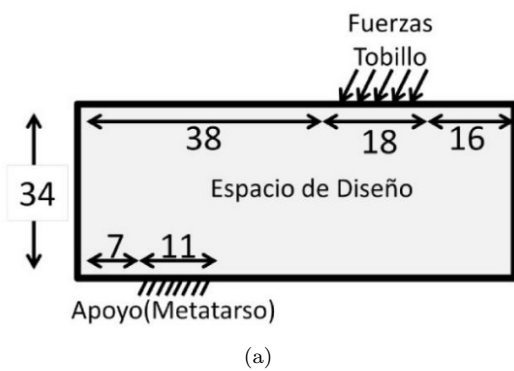


Figura 9: F2 posición de apoyo y su resultado.

3. Conclusiones

3.1. Jorge Fuentes

En esta ocasión tratando de cumplir el objetivo planteado de presentar una propuesta de análisis de formas y de la programación para la ejecución de la optimización de características de trabajo específicas que presenta las ventajas ahora en esta práctica 5 sobre la elaboración de una Optimización de una prótesis de pie, aplicando los conocimientos previos sobre nuestro dominio en MATLAB así como los pasos de la programación logramos elaborar una Optimización resultante para una prótesis de pie.

3.2. Tania Hernandez

La práctica realizada consistió en la elaboración de una prótesis de pie, para la cual como en todas las anteriores prácticas, es por esto, que se tuvo que investigar la forma geométrica, estado del arte, entre otros aspectos relevantes, de mi parte aprendí, que se pueden realizar distintos tipos de formas de diseño del pie, el cual varia dependiendo las necesidades que ocupe el paciente y la optimización en MATLAB..

3.3. Anahi Herrera

Para esta actividad se realizaron varias búsquedas de información para ver las características de este trabajo final, por lo cual, se recopilaron los conocimientos adquiridos durante todo el curso y gracias a ellos fue que se logró con este último objetivo de realizar el diseño de una prótesis de pie.

De igual manera para llegar al resultado se tuvieron que realizar investigaciones sobre el funcionamiento de este tipo de prótesis, algunos modelos comerciales ya realizados para así obtener resultados positivos en nuestro trabajo.

3.4. Gustavo Díaz

Dentro de la investigación requerida para el desarrollo de la practica fue un poco sencillo de encontrar la información dado que dentro de la clase se nos pedía encontrar información de esta rama de la biomecánica dado que las prótesis son una forma en que se aplica a la perfección esta rama de la mecánica, en la cual para poder mejorar dicha prótesis se uso el software de Matlab en el cual se buscaron las diferentes partes en las cuales se puede optimizar de manera correcta y que no afecte la estructura de la misma, y sea sencilla para el fabricante hacerla y sea sencilla y cómoda de usar.

3.5. Miriam Mata

Durante este curso de laboratorio mi equipo y yo nos dimos cuenta que se puede usar Matlab para generar un análisis de elemento finito para objetos de ámbito simple y que se pueden usar para diferentes casos, además de generar un buen soporte que nos ayudará mucho en este caso. A partir de lo que aprendimos, nos damos cuenta que los softwares de hoy en día nos apoyan mucho con cálculos e impresiones que nos facilitan el poder generar nuevas ideas e ir más rápido en nuestras investigaciones.

3.6. Alejandro Ramos

En esta práctica pude observar para empezar el diseño de una prótesis de pie después observamos como la podríamos mejorar de diferentes maneras hasta que decidimos una, lo cual fue complicado porque para eso tuvimos que investigar bastante porque no es tan sencillo mejorarlas debido a que un error y podemos causar daño a la persona que la usara pero para mi si la mejoramos de la manera correcta y espero seguir aprendiendo porque me interesa bastante esta área de biomecánica.

Referencias

- [1] Moore J. Transtibial prosthesis a foot prosthesis, May 2011.
- [2] Otomori M. Matlab code for a level set-based topology optimization method using a reaction diffusion equation., March 2015.
- [3] Michael R. A discrete level-set topology optimization code written in matlab., December 2009.