

Autoevaluación

OCW 2020: *Parametrización y representación gráfica de superficies construidas*

Tema 2. Ejercicios propuestos resueltos

Equipo docente del curso

*Martín Yagüe, Luis
Barrallo Calonge, Javier
Soto Merino, Juan Carlos
Lecubarri Alonso, Inmaculada*

Departamento de Matemática Aplicada
Escuela de Ingeniería de Bilbao, Edificio II-I (EIB/BIE)
ETS de Arquitectura de Donostia-San Sebastián (ETSASS/DAGET)



EJERCICIOS DEL TEMA 2. PARAMETRIZACIÓN DE CURVAS EN EL ESPACIO

Ejercicio nº1

Enunciado

La bóveda cuatrimpartita o de crucería está asociada al estilo gótico y sus arcos se forman a partir de la construcción *vesica piscis*.

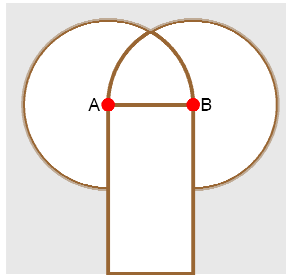


Imagen 1. Generación del arco gótico (imagen propia)

Una bóveda de crucería se divide en cuatro partes iguales por medio de seis arcos (dos arcos fajones en la nave, dos arcos formeros en los muros y dos arcos cruzados) además de dos aristas en forma de cruz atravesando la clave (dovela central).



Imagen 2. Bóveda de crucería (imagen propia)

Se supone un tramo de bóveda cuatrimpartita formada a partir de arcos con forma de *vesica piscis* canónica (cada circunferencia pasa por el centro de la otra) contenida en un cuadrado de 10 metros de lado en planta.

Parametrice y represente las líneas que definen los seis arcos apuntados y las dos aristas de la bóveda.

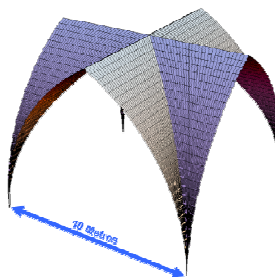


Imagen 3. Simulación de la bóveda (imagen propia)

Resolución

`Remove["Global`*"]`

- La bóveda resulta de la intersección de cuatro superficies cilíndricas.
- Se establece un sistema cartesiano de referencia tal que el plano OXY contiene los puntos de arranque de los seis arcos, los ejes OX y OY coinciden con las proyecciones sobre OXY de las aristas y el eje OZ es la recta vertical que pasa por la clave.
- Para cualquier consulta sobre la *vesica piscis* se remite al *Ejercicio nº5* del fichero de *Ejercicios resueltos* del Tema 1 (*Parametrización de curvas planas*).
- La distancia entre los centros y el radio de las circunferencias que generan la *vesica piscis* equilátera coinciden; en este caso: $R = 10$ m

$r = 10$;

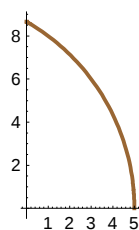
- Se establece una sola parametrización que define la base, $x(t)$, y la altura, $z(t)$, de la mitad de uno de los arcos; el resto de curvas se obtiene a partir de este arco por simetría

`{x[t_] = -r/2 + r Cos[t], z[t_] = r Sin[t]}`

`{-5 + 10 Cos[t], 10 Sin[t]}`

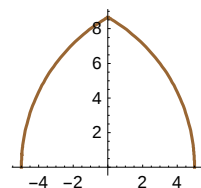
- Se representa en el plano uno de los arcos
 - la parametrización permite la representación de una mitad

`ParametricPlot[{x[t], z[t]}, {t, 0, Pi/3}, PlotStyle -> {Brown, Thick}]`



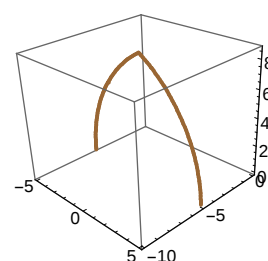
- la otra mitad se obtiene por simetría respecto al eje de ordenadas

`ParametricPlot[{x[t], z[t]}, {-x[t], z[t]}, {t, 0, Pi/3}, PlotStyle -> {Brown, Brown, Thick}]`



- Ahora hay que representar en el espacio ese tipo de arcos:
 - arcos fajones, son paralelos y distan 10 metros (se trazan en los planos $y = -5$ m, $y = 5$ m)

`arc1 = ParametricPlot3D[{x[t], -5, z[t]}, {-x[t], -5, z[t]}, {t, 0, Pi/3}, PlotStyle -> {Brown, Brown, Thick}]`



```
arc2 = ParametricPlot3D[{x[t], 5, z[t]}, {-x[t], 5, z[t]}],
{t, 0, Pi/3}, PlotStyle -> {Brown, Brown, Thick}];
```

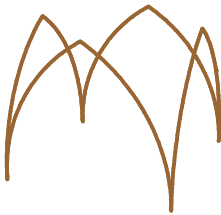
- arcos formeros, son paralelos y distan 10 metros (se trazan en los planos $x = -5$ m, $x = 5$ m)

```
arc3 = ParametricPlot3D[{-5, x[t], z[t]}, {-5, -x[t], z[t]}],
{t, 0, Pi/3}, PlotStyle -> {Brown, Brown, Thick}];
```

```
arc4 = ParametricPlot3D[{5, x[t], z[t]}, {5, -x[t], z[t]}],
{t, 0, Pi/3}, PlotStyle -> {Brown, Brown, Thick}];
```

- representación conjunta de estos cuatro arcos

```
Show[arc1, arc2, arc3, arc4, PlotRange -> All, Boxed -> False, Axes -> False]
```



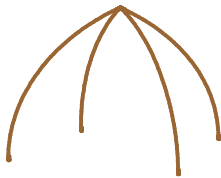
- arcos cruzados (ubicados en los planos $x = -y$, $x = y$)

```
arc5 = ParametricPlot3D[{x[t], x[t], z[t]}, {-x[t], -x[t], z[t]}],
{t, 0, Pi/3}, PlotStyle -> {Brown, Brown, Thick}];
```

```
arc6 = ParametricPlot3D[{x[t], -x[t], z[t]}, {-x[t], x[t], z[t]}],
{t, 0, Pi/3}, PlotStyle -> {Brown, Brown, Thick}];
```

- representación conjunta de los dos arcos cruzados

```
Show[arc5, arc6, PlotRange -> All, Boxed -> False, Axes -> False]
```



- Representación de las aristas

```
aristas =
```

```
ParametricPlot3D[{t, 0, z[Pi/3]}, {0, t, z[Pi/3]}], {t, -5, 5}, PlotStyle -> {Brown, Brown, Thick}];
```

- Representación gráfica de todo el conjunto de arcos y aristas

```
Show[arc1, arc2, arc3, arc4, arc5, arc6, aristas, PlotRange -> All, Boxed -> False, Axes -> False]
```



Ejercicio nº2

Enunciado

El arquitecto argentino César Pelli fue el autor del Master Plan de Abandoibarra en Bilbao que articuló alrededor de una gran torre diseñada por su estudio para la empresa Iberdrola.



Imagen 4. Torre Iberdrola, Bilbao, Bizkaia (imagen propia)

La torre se concibe como la intersección de tres conos imaginarios de 1000 metros que quedan truncados a una altura de 165 metros. La sección adquiere la forma de un Triángulo Reuleaux isósceles ya que los centros de los conos no son equidistantes produciendo en el edificio el efecto de una proa de barco varada en la ría bilbaína.

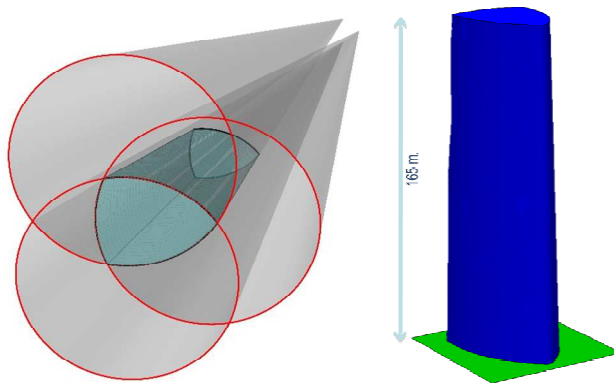


Imagen 5. Simulación de la generación de la torre (imagen propia)

Dado que la sección de los conos disminuye con la altura también lo hace la superficie de cada planta. Así en cota cero las circunferencias que delimitan la planta tienen un radio de 42 metros mientras que en la cubierta el radio disminuye hasta los 38 metros. Los centros de los conos son $(28,0)$, $(-3,24)$ y $(-3,-24)$ para cualquier altura, tal y como se muestra en el gráfico adjunto donde se indican las ecuaciones de las circunferencias en cota cero y 165 metros.

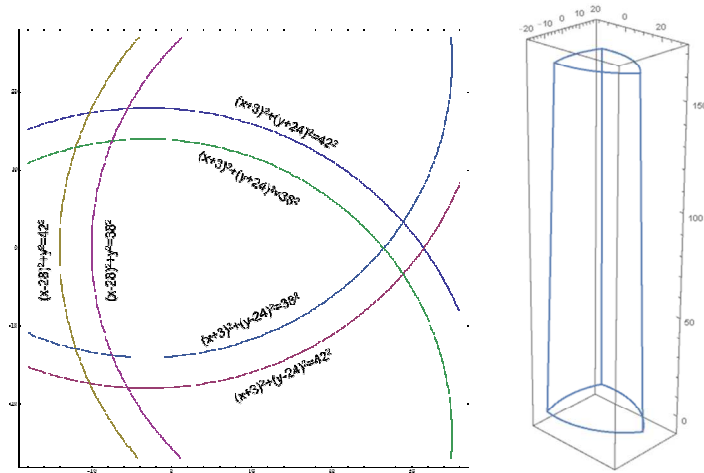


Imagen 6. Líneas en cota 0 y cota 165 junto con el contorno básico (imagen propia)

Con los datos facilitados, parametrize y represente las nueve líneas que definen el contorno básico de la Torre Iberdrola.

Resolución

Remove ["Global`*"]

- Las ecuaciones de las circunferencias vienen dadas respecto a un sistema cartesiano ya fijado y cuyo origen se sitúa en cota cero (suelo).
- El triángulo Reuleaux isósceles viene delimitado por las siguientes circunferencias

$$\blacksquare \text{ en cota cero: } C_0 \equiv \begin{cases} (x+3)^2 + (y-24)^2 = 42^2 \\ (x+3)^2 + (y+24)^2 = 42^2 \\ (x-28)^2 + y^2 = 42^2 \end{cases}$$

$$\blacksquare \text{ en cota 165: } C_{165} \equiv \begin{cases} (x+3)^2 + (y-24)^2 = 38^2 \\ (x+3)^2 + (y+24)^2 = 38^2 \\ (x-28)^2 + y^2 = 38^2 \end{cases}$$

- Parametrización de las circunferencias $\forall t \in [0, 2\pi]$:

r1 = 42; r2 = 38;

- en cota cero:

$$C_1 \equiv \begin{cases} x(t) = -3 + 42 \cos(t) \\ y(t) = 24 + 42 \sin(t) \\ z(t) = 0 \end{cases} \quad C_2 \equiv \begin{cases} x(t) = -3 + 42 \cos(t) \\ y(t) = -24 + 42 \sin(t) \\ z(t) = 0 \end{cases} \quad C_3 \equiv \begin{cases} x(t) = 28 + 42 \cos(t) \\ y(t) = 42 \sin(t) \\ z(t) = 0 \end{cases}$$

c1 = {-3 + r1 * Cos[t], 24 + r1 * Sin[t], 0};

c2 = {-3 + r1 * Cos[t], -24 + r1 * Sin[t], 0};

c3 = {28 + r1 * Cos[t], r1 * Sin[t], 0};

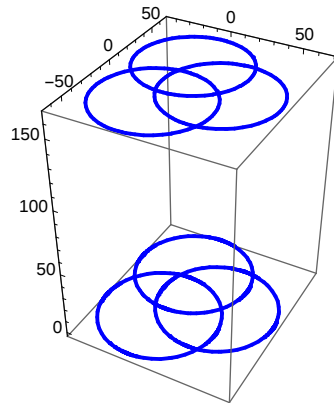
$$\blacksquare \text{ en cota 165: } C_{165} \equiv \begin{cases} (x+3)^2 + (y-24)^2 = 38^2 \\ (x+3)^2 + (y+24)^2 = 38^2 \\ (x-28)^2 + y^2 = 38^2 \end{cases}$$

$$C_4 \equiv \begin{cases} x(t) = -3 + 38 \cos(t) \\ y(t) = 24 + 38 \sin(t) \\ z(t) = 165 \end{cases} \quad C_5 \equiv \begin{cases} x(t) = -3 + 38 \cos(t) \\ y(t) = -24 + 38 \sin(t) \\ z(t) = 165 \end{cases} \quad C_6 \equiv \begin{cases} x(t) = 28 + 38 \cos(t) \\ y(t) = 38 \sin(t) \\ z(t) = 165 \end{cases}$$

```
c4 = {-3 + r2 * Cos[t], 24 + r2 * Sin[t], 165};
c5 = {-3 + r2 * Cos[t], -24 + r2 * Sin[t], 165};
c6 = {28 + r2 * Cos[t], r2 * Sin[t], 165};
```

■ Representación gráfica

```
ParametricPlot3D[{c1, c2, c3, c4, c5, c6}, {t, 0, 2 Pi},
  PlotStyle -> {Blue, Blue, Blue, Blue, Blue, Blue, Thick}, PlotRange -> All]
```



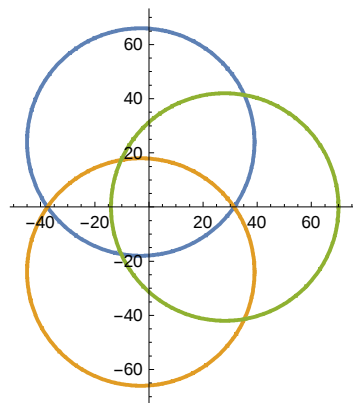
■ Para representar el contorno básico del edificio se necesitan los puntos de intersección entre las circunferencias que lo delimitan

■ Cota 0: triángulo de Reuleaux

- una representación gráfica de las circunferencias ayuda a determinar los puntos

```
c10 = {-3 + r1 * Cos[t], 24 + r1 * Sin[t]};
c20 = {-3 + r1 * Cos[t], -24 + r1 * Sin[t]};
c30 = {28 + r1 * Cos[t], r1 * Sin[t]};
```

```
planta = ParametricPlot[{c10, c20, c30}, {t, 0, 2 Pi}, PlotStyle -> {Thick}, PlotRange -> All]
```



■ $C_1 \cap C_3 : P_{13} = (-10.2392, -17.3714, 0)$

```
NSolve[{(x + 3)^2 + (y - 24)^2 == r1^2, (x - 28)^2 + y^2 == r1^2}, {x, y}]
{{x -> 35.2392, y -> 41.3714}, {x -> -10.2392, y -> -17.3714}}
```

■ $C_1 \cap C_2 : P_{12} = (31.4674, 0, 0)$

```
NSolve[{(x + 3)^2 + (y - 24)^2 == r1^2, (x + 3)^2 + (y + 24)^2 == r1^2}, {x, y}]
{{x -> -37.4674, y -> 0.}, {x -> 31.4674, y -> 0.}}
```

- $C_2 \cap C_3 : P_{23} = (-10.2392, 17.3714, 0)$

```
NSolve[{(x + 3)^2 + (y + 24)^2 == r1^2, (x - 28)^2 + y^2 == r1^2}, {x, y}]
```

```
{{x -> 35.2392, y -> -41.3714}, {x -> -10.2392, y -> 17.3714}}
```

- representación gráfica con los puntos de intersección

```
p13 = Graphics[{Red, PointSize[Large], Point[{-10.2392, -17.3714}], Black, Text["P13", {-20, -20}]}];
```

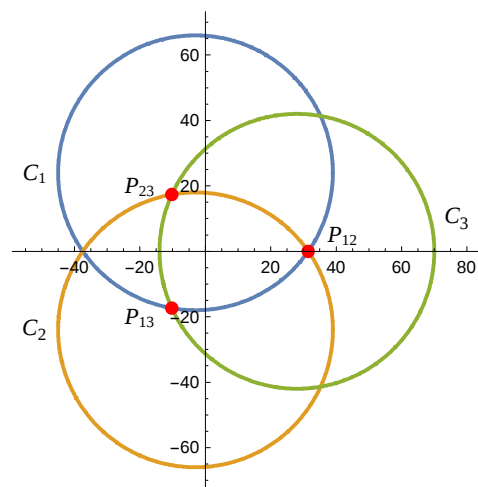
```
p12 = Graphics[{Red, PointSize[Large], Point[{31.4674, 0}], Black, Text["P12", {42, 5}]}];
```

```
p23 = Graphics[{Red, PointSize[Large], Point[{-10.2392, 17.3714}], Black, Text["P23", {-20, 20}]}];
```

```
label =
```

```
Graphics[{Black, Text["C1", {-52, 24}], Black, Text["C2", {-52, -24}], Black, Text["C3", {77, 10}]}];
```

```
Show[planta, label, p13, p12, p23, PlotRange -> All]
```



- obtención de los parámetros que delimitan los arcos y representación gráfica

- $C_1 : t \in [-1.74400, -0.6082]$

```
NSolve[c1[[1]] == 31.46737587922818, t]
```

```
*** NSolve: Inverse functions are being used by NSolve, so some solutions may not be found; use Reduce for complete solution information.
```

```
{{t -> -0.608246}, {t -> 0.608246}}
```

```
NSolve[c1[[1]] == -10.23916480733909, t]
```

```
*** NSolve: Inverse functions are being used by NSolve, so some solutions may not be found; use Reduce for complete solution information.
```

```
{{t -> -1.74402}, {t -> 1.74402}}
```

```
arc10 = ParametricPlot3D[c1, {t, -1.7440, -0.6082}, PlotStyle -> {Blue, Thickness[0.015]}];
```

- $C_2 : t \in [0.6082, 1.7440]$

```
NSolve[c2[[1]] == 31.46737587922818, t]
```

```
*** NSolve: Inverse functions are being used by NSolve, so some solutions may not be found; use Reduce for complete solution information.
```

```
{{t -> -0.608246}, {t -> 0.608246}}
```

```
NSolve[c2[[1]] == -10.23916480733909, t]
```

... **NSolve**: Inverse functions are being used by NSolve, so some solutions may not be found; use Reduce for complete solution information.

```
{{t -> -1.74402}, {t -> 1.74402}}
```

```
arc20 = ParametricPlot3D[c2, {t, 0.6082, 1.7440}, PlotStyle -> {Blue, Thickness[0.015]}];
```

- $C_3: t \in [2.71518, 2\pi - 2.71518]$; el programa da dos soluciones en el intervalo $t \in [-\pi, \pi]$

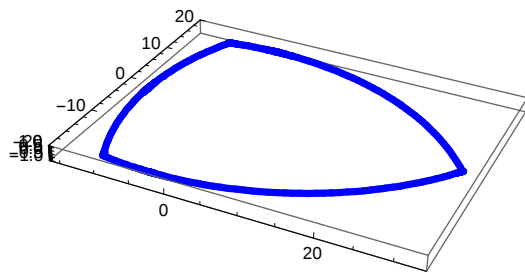
```
NSolve[c3[[1]] == -10.23916480733909, t]
```

... **NSolve**: Inverse functions are being used by NSolve, so some solutions may not be found; use Reduce for complete solution information.

```
{{t -> -2.71518}, {t -> 2.71518}}
```

```
arc30 = ParametricPlot3D[c3, {t, 2.7152, -2.7152 + 2 Pi}, PlotStyle -> {Blue, Thickness[0.015]}];
```

```
Cota0 = Show[arc30, arc20, arc10, PlotRange -> All]
```



- Cota 165 (cubierta): triángulo de Reuleaux

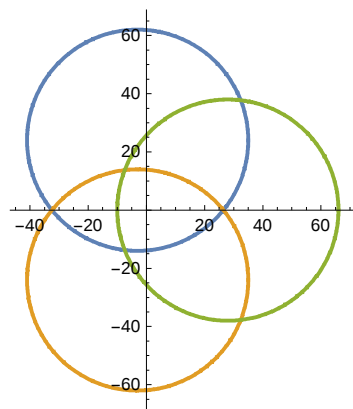
- una representación gráfica de las circunferencias ayuda a determinar los puntos

```
c4c = {-3 + r2 * Cos[t], 24 + r2 * Sin[t]};
```

```
c5c = {-3 + r2 * Cos[t], -24 + r2 * Sin[t]};
```

```
c6c = {28 + r2 * Cos[t], r2 * Sin[t]};
```

```
cubierta = ParametricPlot[{c4c, c5c, c6c}, {t, 0, 2 Pi}, PlotStyle -> {Thick}, PlotRange -> All]
```



- $C_4 \cap C_6: P_{46} = (-7.4286, -13.7411, 165)$

```
NSolve[{(x + 3)^2 + (y - 24)^2 == r2^2, (x - 28)^2 + y^2 == r2^2}, {x, y}]
```

```
{{x -> 32.4286, y -> 37.7411}, {x -> -7.42856, y -> -13.7411}}
```

- $C_4 \cap C_5 : P_{45} = (26.4618, 0, 165)$

```
NSolve[{(x + 3)^2 + (y - 24)^2 == r2^2, (x + 3)^2 + (y + 24)^2 == r2^2}, {x, y}]
{{x -> -32.4618, y -> 0.}, {x -> 26.4618, y -> 0.}}
```

- $C_5 \cap C_6 : P_{56} = (-7.4286, 13.7411, 165)$

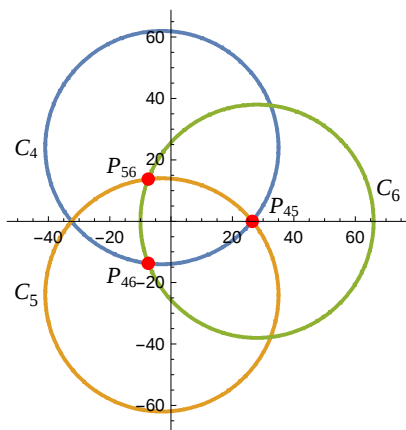
```
NSolve[{(x + 3)^2 + (y + 24)^2 == r2^2, (x - 28)^2 + y^2 == r2^2}, {x, y}]
{{x -> 32.4286, y -> -37.7411}, {x -> -7.42856, y -> 13.7411}}
```

- representación gráfica con los puntos de intersección

```
p46 = Graphics[{Red, PointSize[Large], Point[{-7.4286, -13.7411}], Black, Text["P46", {-16, -18}]}];
p45 = Graphics[{Red, PointSize[Large], Point[{26.4618, 0}], Black, Text["P45", {37, 5}]}];
p56 = Graphics[{Red, PointSize[Large], Point[{-7.4286, 13.7411}], Black, Text["P56", {-16, 18}]}];

labelc =
  Graphics[{Black, Text["C4", {-47, 24}], Black, Text["C5", {-47, -24}], Black, Text["C6", {71, 10}]}];

Show[cubierta, labelc, p46, p45, p56, PlotRange -> All]
```



- obtención de los parámetros que delimitan los arcos y representación gráfica

- $C_4 : t \in [-1.6876, -0.6836]$

```
NSolve[c4[[1]] == 26.46183972531247, t]
```

*** NSolve: Inverse functions are being used by NSolve, so some solutions may not be found; use Reduce for complete solution information.

```
{{t -> -0.683588}, {t -> 0.683588}}
```

```
NSolve[c4[[1]] == -7.428564682530212, t]
```

```
{{t -> -1.6876}, {t -> 1.6876}}
```

```
arc4c = ParametricPlot3D[c4, {t, -1.6876, -0.6836}, PlotStyle -> {Blue, Thickness[0.015]}];
```

- $C_5 : t \in [0.6836, 1.6876]$

```
NSolve[c5[[1]] == 26.46183972531247, t]
```

```
{{t -> -0.683588}, {t -> 0.683588}}
```

```
NSolve[c5[[1]] == -7.428564682530212, t]
```

```
{{t -> -1.6876}, {t -> 1.6876}}
```

```
arc5c = ParametricPlot3D[c5, {t, 0.6836, 1.6876}, PlotStyle -> {Blue, Thickness[0.015]}];
```

- $C_6 : t \in [2.71518, 2\pi - 2.71518]$; el programa da dos soluciones en el intervalo $t \in [-\pi, \pi]$

```
NSolve[c6[[1]] == -7.428564682530212, t]
```

```
{{t -> -2.7716}, {t -> 2.7716}}
```

```
arc6c = ParametricPlot3D[c6, {t, 2.7716, -2.7716 + 2 Pi}, PlotStyle -> {Blue, Thickness[0.015]}];
```

```
Cota165 = Show[arc4c, arc5c, arc6c, PlotRange -> All];
```

- Aristas que unen los vértices entre las dos cotas

- recta que pasa por los puntos $P_{13} = (-10.2392, -17.3714, 0)$ y $P_{46} = (-7.4286, -13.7411, 165)$

```
P13 = {-10.2392, -17.3714, 0}; P46 = {-7.4286, -13.7411, 165}; v1 = P46 - P13;
```

```
arista1 = {P13[[1]] + v1[[1]] t, P13[[2]] + v1[[2]] t, P13[[3]] + v1[[3]] t};
```

```
ar1 = ParametricPlot3D[arista1, {t, 0, 1}, PlotStyle -> {Blue, Thickness[0.015]}];
```

- recta que pasa por los puntos $P_{23} = (-10.2392, 17.3714, 0)$ y $P_{56} = (-7.4286, 13.7411, 165)$

```
P23 = {-10.2392, 17.3714, 0}; P56 = {-7.4286, 13.7411, 165}; v2 = P56 - P23;
```

```
arista2 = {P23[[1]] + v2[[1]] t, P23[[2]] + v2[[2]] t, P23[[3]] + v2[[3]] t};
```

```
ar2 = ParametricPlot3D[arista2, {t, 0, 1}, PlotStyle -> {Blue, Thickness[0.015]}];
```

- recta que pasa por los puntos $P_{12} = (31.4674, 0, 0)$ y $P_{45} = (26.4618, 0, 165)$

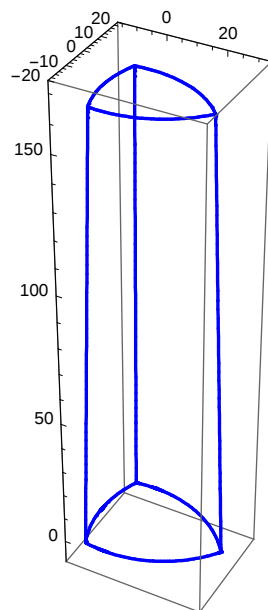
```
P12 = {31.4674, 0, 0}; P45 = {26.4618, 0, 165}; v3 = P45 - P12;
```

```
arista3 = {P12[[1]] + v3[[1]] t, P12[[2]] + v3[[2]] t, P12[[3]] + v3[[3]] t};
```

```
ar3 = ParametricPlot3D[arista3, {t, 0, 1}, PlotStyle -> {Blue, Thickness[0.015]}];
```

- Contorno básico de la torre: los triángulos de Reuleaux de la planta y la cubierta junto con las aristas que unen sus vértices

```
Show[Cota0, Cota165, ar1, ar2, ar3, PlotRange -> All]
```



Ejercicio nº3

Enunciado

La Torre Millennium fue un proyecto fallido diseñado en 1989 por el arquitecto británico Norman Foster. Iba a ser construida en la Bahía de Tokio a dos kilómetros de la costa y hubiera sido con más de 800 metros el edificio más alto del mundo. Debido a problemas financieros el proyecto no llegó a materializarse.

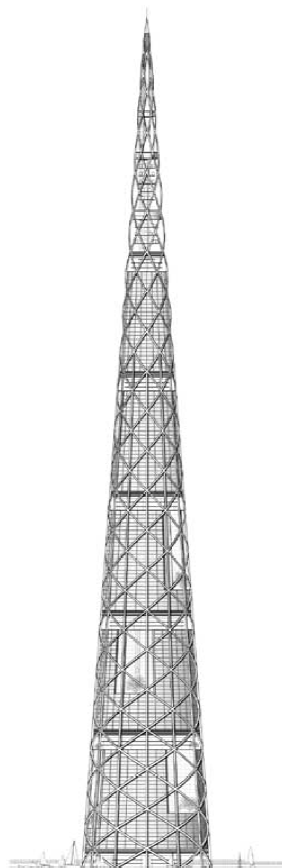


Imagen 7. Boceto de la torre

Imagen obtenida en:

<https://www.e-architect.co.uk/wp-content/uploads/2008/11/millennium-tower-tokyo-building-by-foster-and-partners.jpg>

Se adjunta el siguiente enlace que remite a un artículo de *The Guardian* donde se publica una fotografía de una maqueta de la torre:

<https://www.theguardian.com/cities/2019/jun/11/unbuilt-tokyo-depthscrapers-and-a-million-person-pyramid#img-7>

La torre se proyectó de forma cónica con una base de 130 metros de diámetro. Su estructura exterior estaría formada por doce hélices cónicas de acero que recorrerían la fachada arrancando a pares desde la base en dirección horaria y antihoraria hasta los 800 metros de altura.

Parametrice y represente la estructura helicoidal que recorrería el exterior del edificio.



Imagen 8. Generación de la estructura (imagen propia)

Resolución

`Remove["Global`*"]`

- Radio en la base: $R = 65$; altura: $h = 800$
- En la imagen 7 se observa que las hélices trazan cuatro vueltas alrededor de la torre: $n = 4$

$r = 65$; $h = 800$; $n = 4$;

- Parametrización:

`helice = {x = (t / (2 Pi) - 1) r Cos[n t], y = (t / (2 Pi) - 1) r Sin[n t], z = 800 t / (2 Pi)};`

- De la base arranca la estructura helicoidal formada por seis pares de hélices ; además,
 - cada una de las cuales gira en sentido inverso que el de su pareja
 - cada par se distancia en 60° del siguiente

$$C_2 \equiv x^2 + \left(y - \frac{\sqrt{3}r}{2}\right)^2 = r^2 \quad \forall x \in \left[-\frac{r}{2}, \frac{r}{2}\right]$$

- las ecuaciones de un giro de α grados son las siguientes:

$$\begin{cases} x' = x \cdot \cos(\alpha) - y \cdot \sin(\alpha) \\ y' = x \cdot \sin(\alpha) + y \cdot \cos(\alpha) \end{cases}$$

- Generación de los seis pares de hélices

$k = 0$;

`{x1 = x Cos[k 60 °] - y Sin[k 60 °], y1 = x Sin[k 60 °] + y Cos[k 60 °]};`

`{x2 = x Cos[k 60 °] + y Sin[k 60 °], y2 = x Sin[k 60 °] - y Cos[k 60 °]};`

$l0 =$

`ParametricPlot3D[{x1, y1, z}, {x2, y2, z}], {t, 0, 2 Pi}, PlotStyle -> {Gray, Gray, Thickness[0.01]}];`

$k = 1$;

`{x1 = x Cos[k 60 °] - y Sin[k 60 °], y1 = x Sin[k 60 °] + y Cos[k 60 °]};`

`{x2 = x Cos[k 60 °] + y Sin[k 60 °], y2 = x Sin[k 60 °] - y Cos[k 60 °]};`

$l1 =$

`ParametricPlot3D[{x1, y1, z}, {x2, y2, z}], {t, 0, 2 Pi}, PlotStyle -> {Gray, Gray, Thickness[0.01]}];`

$k = 2$;

`{x1 = x Cos[k 60 °] - y Sin[k 60 °], y1 = x Sin[k 60 °] + y Cos[k 60 °]};`

`{x2 = x Cos[k 60 °] + y Sin[k 60 °], y2 = x Sin[k 60 °] - y Cos[k 60 °]};`

$l2 =$

`ParametricPlot3D[{x1, y1, z}, {x2, y2, z}], {t, 0, 2 Pi}, PlotStyle -> {Gray, Gray, Thickness[0.01]}];`

```

k = 3;
{x1 = x Cos[k 60 °] - y Sin[k 60 °], y1 = x Sin[k 60 °] + y Cos[k 60 °]};
{x2 = x Cos[k 60 °] + y Sin[k 60 °], y2 = x Sin[k 60 °] - y Cos[k 60 °]};
13 =
  ParametricPlot3D[{ {x1, y1, z}, {x2, y2, z}}, {t, 0, 2 Pi}, PlotStyle -> {Gray, Gray, Thickness[0.01]}}];

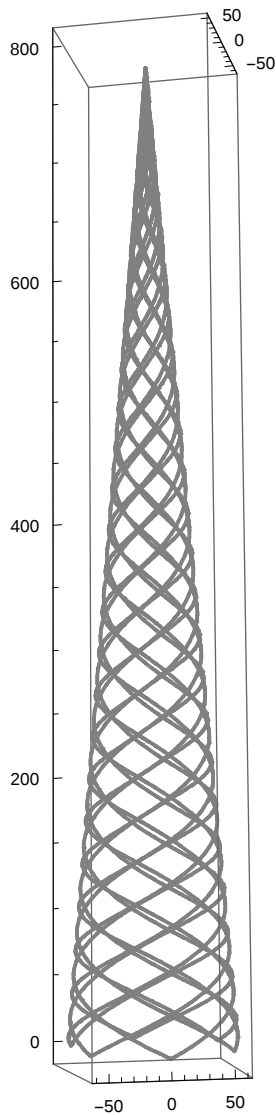
k = 4;
{x1 = x Cos[k 60 °] - y Sin[k 60 °], y1 = x Sin[k 60 °] + y Cos[k 60 °]};
{x2 = x Cos[k 60 °] + y Sin[k 60 °], y2 = x Sin[k 60 °] - y Cos[k 60 °]};
14 =
  ParametricPlot3D[{ {x1, y1, z}, {x2, y2, z}}, {t, 0, 2 Pi}, PlotStyle -> {Gray, Gray, Thickness[0.01]}}];

k = 5;
{x1 = x Cos[k 60 °] - y Sin[k 60 °], y1 = x Sin[k 60 °] + y Cos[k 60 °]};
{x2 = x Cos[k 60 °] + y Sin[k 60 °], y2 = x Sin[k 60 °] - y Cos[k 60 °]};
15 =
  ParametricPlot3D[{ {x1, y1, z}, {x2, y2, z}}, {t, 0, 2 Pi}, PlotStyle -> {Gray, Gray, Thickness[0.01]}}];

```

■ Representación gráfica de la estructura helicoidal:

```
Show[10, 11, 12, 13, 14, 15]
```



Ejercicio nº4

Enunciado

El conjunto Puerta de Europa lo constituyen dos torres levantadas entre 1989 y 1996 en Madrid por el arquitecto estadounidense Philip Johnson, quien en 1979 fue el primer receptor del premio Pritzker de Arquitectura. Johnson se obsesionó en este proyecto con la idea de acabar con el ángulo recto y a partir de esta premisa surgieron los dos primeros grandes edificios inclinados del mundo.



Imagen 9. Puerta de Europa, Madrid, España (imagen propia)

Las torres, con forma de paralelepípedo de base cuadrada, están contenidas en una estructura perimetral de acero inoxidable. Destacan las vigas verticales en las fachadas principales, sirviendo de referencia para apreciar la inclinación del edificio, y las elegantes formas en aspa de las fachadas laterales (Imagen 10). Esta estructura exterior no evita la tendencia natural a caer hacia el interior de ambos edificios, por lo que se encuentran atirantados por unos gruesos cables de acero que recorren el exterior de las fachadas hasta dos gigantescos bloques de hormigón soterrados que actúan como contrapeso.



Imagen 10. Fachada lateral (imagen propia)

Parametrice y represente los segmentos de recta que constituyen la estructura metálica perimetral de ambas torres.

Para ello, debe tenerse en cuenta que los puntos más próximos de ambas torres, en la azotea, se encuentran a 88 metros de distancia.

Las demás medidas se presentan en la siguiente imagen:

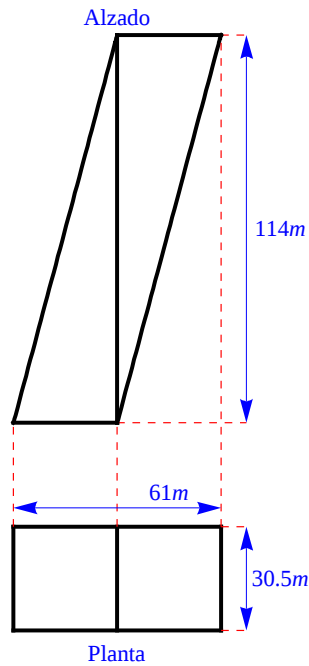


Imagen 11. Alzado y planta de una torre (imagen propia)

Resolución

Remove["Global`*"]

- Se denomina *Torre n°1* a la situada a la izquierda de la imagen y *Torre n°2* a la situada a la derecha
- Dimensiones de cada torre
 - altura: $h = 114 \text{ m}$
 - longitud del lado de la base cuadrada: $b = 30.5 \text{ m}$
 - razón entre la altura y la longitud de la base: $r = \frac{h}{b} \approx 3.7377$
- Distancia entre las torres
 - mínima: $d_{\min} = 88 \text{ m}$
 - máxima: $d_{\max} = d_{\min} + 2b = 149 \text{ m}$

{h = 114, b = 30.5, dmin = 88, dmax = dmin + 2 b, p = h / b};

- Se trata de parametrizar los segmentos de recta que constituyen esa estructura metálica perimetral
- Se recuerda que si se conocen dos puntos de una recta r , $P_0 = (x_0, y_0, z_0)$, $P_1 = (x_1, y_1, z_1) \in r$, puede calcularse un vector director $\vec{v} = (v_1 = x_1 - x_0, v_2 = y_1 - y_0, v_3 = z_1 - z_0)$. De esta forma, puede parametrizarse un segmento de una recta r , $\forall x \in [a, b] \subseteq \mathbb{R}$ como:

$$r \equiv \left\{ \begin{array}{l} x(t) = x_0 + v_1 \cdot t \\ y(t) = y_0 + v_2 \cdot t \\ z(t) = z_0 + v_3 \cdot t \end{array} \right. \quad t \in \left[\frac{a-x_0}{v_1}, \frac{b-x_0}{v_1} \right] \subseteq \mathbb{R}$$

- El origen del sistema cartesiano de ejes se fija en una de las esquinas inferiores de la *Torre n°1* como se muestra en la siguiente imagen

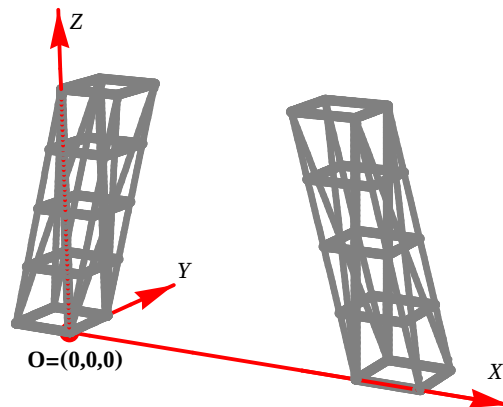


Imagen 12. Establecimiento del sistema de referencia (imagen propia)

- Torre 1

- Vigas horizontales

- Cota 0 (suelo, plano $z = 0\text{ m}$)

$c = 0;$

```
l1c01 = ParametricPlot3D[{-t + b * c / 4, 0, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l1c02 = ParametricPlot3D[{-t + b * c / 4, b, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l1c03 = ParametricPlot3D[{b * c / 4, t, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l1c04 = ParametricPlot3D[{-b * (1 - c / 4), t, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
cota0 = Show[l1c01, l1c02, l1c03, l1c04, AxesLabel -> {X, Y, Z}];
```

- Cota 1 (plano $z = 28.5\text{ m}$)

$c = 1;$

```
l1c11 = ParametricPlot3D[{-t + b * c / 4, 0, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l1c12 = ParametricPlot3D[{-t + b * c / 4, b, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l1c13 = ParametricPlot3D[{b * c / 4, t, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l1c14 = ParametricPlot3D[{-b * (1 - c / 4), t, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
cota1 = Show[l1c11, l1c12, l1c13, l1c14];
```

- Cota 2 (plano $z = 57\text{ m}$)

$c = 2;$

```
l1c21 = ParametricPlot3D[{-t + b * c / 4, 0, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l1c22 = ParametricPlot3D[{-t + b * c / 4, b, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l1c23 = ParametricPlot3D[{b * c / 4, t, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l1c24 = ParametricPlot3D[{-b * (1 - c / 4), t, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
cota2 = Show[l1c21, l1c22, l1c23, l1c24, AxesLabel -> {X, Y, Z}];
```

- Cota 3 (plano $z = 85.5\text{ m}$)

$c = 3;$

```
l1c31 = ParametricPlot3D[{-t + b * c / 4, 0, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l1c32 = ParametricPlot3D[{-t + b * c / 4, b, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
```

```

l1c33 = ParametricPlot3D[{b * c / 4, t, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l1c34 = ParametricPlot3D[{-b * (1 - c / 4), t, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
cota3 = Show[l1c31, l1c32, l1c33, l1c34];

```

- Cota 4 (techo, plano $z = 114\text{ m}$)

```

c = 4;

```

```

l1c41 = ParametricPlot3D[{-t + b * c / 4, 0, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l1c42 = ParametricPlot3D[{-t + b * c / 4, b, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l1c43 = ParametricPlot3D[{b * c / 4, t, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l1c44 = ParametricPlot3D[{-b * (1 - c / 4), t, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
cota4 = Show[l1c41, l1c42, l1c43, l1c44];

```

```

horiz = Show[cota0, cota1, cota2, cota3, cota4];

```

- Aristas verticales

```

l1av1 = ParametricPlot3D[{-b + t, 0, p * t}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.01]}];
l1av2 = ParametricPlot3D[{-b + t, b, p * t}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.01]}];
l1av3 = ParametricPlot3D[{t, 0, p * t}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.01]}];
l1av4 = ParametricPlot3D[{t, b, p * t}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.01]}];
arista = Show[l1av1, l1av2, l1av3, l1av4];

```

- Vigas verticales

```

l1v01 = ParametricPlot3D[{0, 0, t}, {t, 0, 114}, PlotStyle -> {Gray, Thickness[0.015]}];
l1v02 = ParametricPlot3D[{0, b, t}, {t, 0, 114}, PlotStyle -> {Gray, Thickness[0.015]}];
vert = Show[l1v01, l1v02];

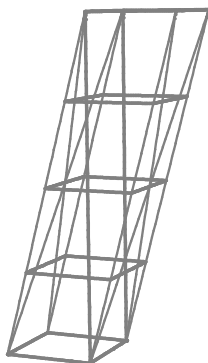
```

- Aspas verticales

```

l1a01 = ParametricPlot3D[{-b + t, t, p * t}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.01]}];
l1a02 = ParametricPlot3D[{-b + t, b - t, p * t}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.01]}];
l1a03 = ParametricPlot3D[{t, t, p * t}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.01]}];
l1a04 = ParametricPlot3D[{t, b - t, p * t}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.01]}];
aspa = Show[l1a01, l1a02, l1a03, l1a04];
torre1 = Show[aspa, arista, vert, horiz, PlotRange -> All, Boxed -> False, Axes -> False]

```



- Torre 2

- Vigas horizontales

- Cota 0 (suelo, plano $z = 0\text{ m}$)

```

c = 0;

l2c01 =
  ParametricPlot3D[{-t - b * c / 4 + dmax + b, 0, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l2c02 = ParametricPlot3D[{-t - b * c / 4 + dmax + b, b, c * h / 4},
  {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];

l2c03 =
  ParametricPlot3D[{-b * c / 4 + dmax + b, t, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l2c04 = ParametricPlot3D[{b * (-1 + c / 4) + dmax + b, t, c * h / 4},
  {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];

cota02 = Show[l2c01, l2c02, l2c03, l2c04];

  ■ Cota 1 (plano  $z = 28.5\text{ m}$ )

c = 1;

l2c11 =
  ParametricPlot3D[{-t - b * c / 4 + dmax + b, 0, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l2c12 = ParametricPlot3D[{-t - b * c / 4 + dmax + b, b, c * h / 4},
  {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];

l2c13 =
  ParametricPlot3D[{-b * c / 4 + dmax + b, t, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l2c14 = ParametricPlot3D[{-b * (1 + c / 4) + dmax + b, t, c * h / 4},
  {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];

cota12 = Show[l2c11, l2c12, l2c13, l2c14];

  ■ Cota 2 (plano  $z = 57\text{ m}$ )

c = 2;

l2c21 =
  ParametricPlot3D[{-t - b * c / 4 + dmax + b, 0, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l2c22 = ParametricPlot3D[{-t - b * c / 4 + dmax + b, b, c * h / 4},
  {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];

l2c23 =
  ParametricPlot3D[{-b * c / 4 + dmax + b, t, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l2c24 = ParametricPlot3D[{-b * (1 + c / 4) + dmax + b, t, c * h / 4},
  {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];

cota22 = Show[l2c21, l2c22, l2c23, l2c24];

  ■ Cota 3 (plano  $z = 85.5\text{ m}$ )

c = 3;

l2c31 =
  ParametricPlot3D[{-t - b * c / 4 + dmax + b, 0, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l2c32 = ParametricPlot3D[{-t - b * c / 4 + dmax + b, b, c * h / 4},
  {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];

l2c33 =
  ParametricPlot3D[{-b * c / 4 + dmax + b, t, c * h / 4}, {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];
l2c34 = ParametricPlot3D[{-b * (1 + c / 4) + dmax + b, t, c * h / 4},
  {t, 0, b}, PlotStyle -> {Gray, Thickness[0.015]}];

cota32 = Show[l2c31, l2c32, l2c33, l2c34];

  ■ Cota 4 (techo, plano  $z = 114\text{ m}$ )

c = 4;

```

```

l2c41 =
  ParametricPlot3D[{-t - b * c / 4 + dmax + b, 0, c * h / 4}, {t, 0, b}, PlotStyle → {Gray, Thickness[0.015]};
l2c42 = ParametricPlot3D[{-t - b * c / 4 + dmax + b, b, c * h / 4},
  {t, 0, b}, PlotStyle → {Gray, Thickness[0.015]};

l2c43 =
  ParametricPlot3D[{-b * c / 4 + dmax + b, t, c * h / 4}, {t, 0, b}, PlotStyle → {Gray, Thickness[0.015]};
l2c44 = ParametricPlot3D[{-b * (1 + c / 4) + dmax + b, t, c * h / 4},
  {t, 0, b}, PlotStyle → {Gray, Thickness[0.015]};

cota42 = Show[l2c41, l2c42, l2c43, l2c44];

horiz2 = Show[cota02, cota12, cota22, cota32, cota42];

  ■ Aristas verticales

l2av1 = ParametricPlot3D[{-b - t + dmax + b, 0, p * t}, {t, 0, b}, PlotStyle → {Gray, Thickness[0.01]};
l2av2 = ParametricPlot3D[{-b - t + dmax + b, b, p * t}, {t, 0, b}, PlotStyle → {Gray, Thickness[0.01]};
l2av3 = ParametricPlot3D[{-t + dmax + b, 0, p * t}, {t, 0, b}, PlotStyle → {Gray, Thickness[0.01]};
l2av4 = ParametricPlot3D[{-t + dmax + b, b, p * t}, {t, 0, b}, PlotStyle → {Gray, Thickness[0.01]};

arista2 = Show[l2av1, l2av2, l2av3, l2av4];

  ■ Vigas verticales

l2v01 = ParametricPlot3D[{0 + dmax, 0, t}, {t, 0, 114}, PlotStyle → {Gray, Thickness[0.015]};
l2v02 = ParametricPlot3D[{0 + dmax, b, t}, {t, 0, 114}, PlotStyle → {Gray, Thickness[0.015]};

vert2 = Show[l2v01, l2v02];

  ■ Aspas verticales

l2a01 = ParametricPlot3D[{-b - t + dmax + b, t, p * t}, {t, 0, b}, PlotStyle → {Gray, Thickness[0.01]};
l2a02 = ParametricPlot3D[{-b - t + dmax + b, b - t, p * t}, {t, 0, b}, PlotStyle → {Gray, Thickness[0.01]};

l2a03 = ParametricPlot3D[{-t + dmax + b, t, p * t}, {t, 0, b}, PlotStyle → {Gray, Thickness[0.01]};
l2a04 = ParametricPlot3D[{-t + dmax + b, b - t, p * t}, {t, 0, b}, PlotStyle → {Gray, Thickness[0.01]};

aspa2 = Show[l2a01, l2a02, l2a03, l2a04];

torre2 = Show[horiz2, arista2, vert2, aspa2, PlotRange → All];

  ■ Representación conjunta

suelo = ParametricPlot3D[{t, u, 0}, {t, -50, 190}, {u, -20, 50}, Mesh → False, PlotStyle → LightGreen];

figura = Show[torre1, torre2, suelo, PlotRange → All,
  Ticks → {{-30.5, 0, 30.5, 118.5, 149, 179.5}, {0, 30.5}, {0, 114}}]

```

