UNIVERSIDAD DE MÁLAGA

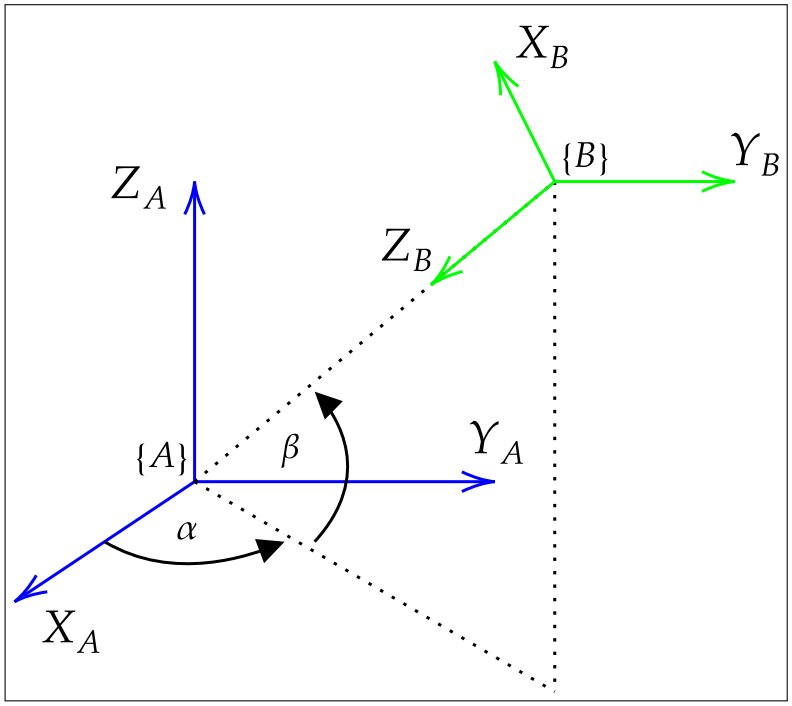
ESCUELA DE INGENIERÍAS INDUSTRIALES

Grado en ingeniería en electrónica, robótica y mecatrónica

Representación de la posición y la orientación.

Transformadas homogéneas.

Tarea # 2



Trabajo realizado por:
Jorge Benavides Macías

27 de marzo de 2022

Fundamentos de robótica

Ejercicios

Problema # 1

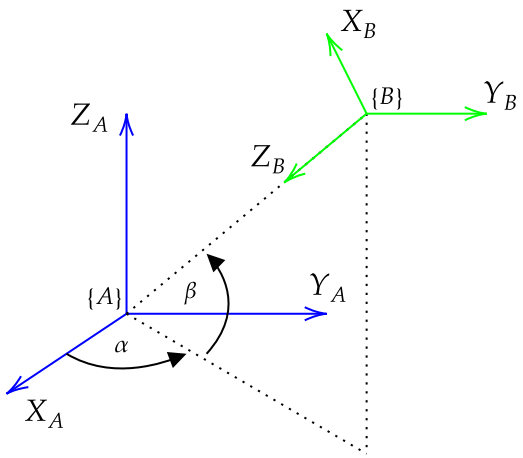


Figura 1: Localización del sistema B con respecto al A

La figura 1 muestra la localización del sistema B con respecto al A cuando la posición de este último viene dada por el vector ${}^{A}T_B = (x, y, z)^T$. La orientación de B con respecto al A se define de la siguiente forma:

El eje Z_B apunta hacia el origen de coordenadas del sistema $\{A\}$, el eje X_B avanza en el sentido positivo del eje Z_A, y por último, el eje Y_B está contenido en un plano paralelo al formado por X_A e Y_A.

- Se pide el cálculo de la localización de $\{B\}$ con respecto al $\{A\}$, representada por la transformada homogénea ${}^{A}T_B$. Para ello, se emplea la siguiente metodología:
  - Plantear los ángulos de giro a y de elevación b, que definen la orientación del sistema $\{B\}$, en función del vector de posición ${}^{A}T_B = (x, y, z)^T$.
  - Suponer un sistema móvil coincidente con $\{A\}$ y que se verá modificado a través de rotaciones y desplazamientos.
  - Elaborar una secuencia de transformaciones homogéneas elementales que, tras aplicarlas, consiga que este sistema móvil coincida perfectamente con $\{B\}$.
- Construir una función MATLAB que posea como argumento el vector de posición del sistema de coordenadas B, y retorne la matriz de transformación homogénea que relacione el mencionado sistema con el global de referencias $\{A\}$ y utilizarla en el caso de ${}^{A}T_B = (100, 70, 150)^T$. Representar ambos sistemas con `createFRAME`.

El código 1 es una **función** que mediante trigonometría calcula los ángulos α, β y la matriz que representan los distintos giros y desplazamientos, en este caso *móviles*, que consiguen el sistema $\{B\}$. La función minimiza los fallos usando el seno y coseno junto con la función `atan2` que ofrece *MATLAB*.

```
function [matrix, alpha, beta] = calculate_transformation(vector)
x = vector(1);
y = vector(2);
z = vector(3);
c = sqrt(x^2+y^2);
sin_alpha = y/c;
cos_alpha = x/c;
L = sqrt(c^2+z^2);
sin_beta = z/L;
cos_beta = c/L;
alpha = atan2(sin_alpha,cos_alpha);
beta = atan2(sin_beta,cos_beta);
I = eye(4);
matrix = I*rotY(-pi/2)*rotX(alpha)*rotY(-(beta))*move(0,0,-sqrt(c^2+z^2));
end
```

Código 1: Matriz de la transformada ${}^{A}T_{B}$.

Con la función ya implementada en el código 1 generamos el código 2, en el que definimos el punto solicitado y lo intruducimos en la función `calculate_transformation`.

```
punto = [100,70,150];
[ATB, alpha, beta] = calculate_transformation(punto);
```

Código 2: Matriz de la transformada ${}^{A}T_{B}$.

Obtenemos los siguientes resultados:

```
BTO =
     0     1     0    11
    -1     0     0    10
     0     0     1     1
     0     0     0     1
```

```
view(57,6);
grid on
Base =[1 0 0 0;0 1 0 0; 0 0 1 0; 0 0 0 1];
createFRAME(Base,'b','Base',200);
createFRAME(ATB,'r','ATB',250);
```

Código 3: Representación del sistema mediante las funciones del createFrame

La representación de una base y del sistema hallado, mediante el código 3, da el siguiente resultado:

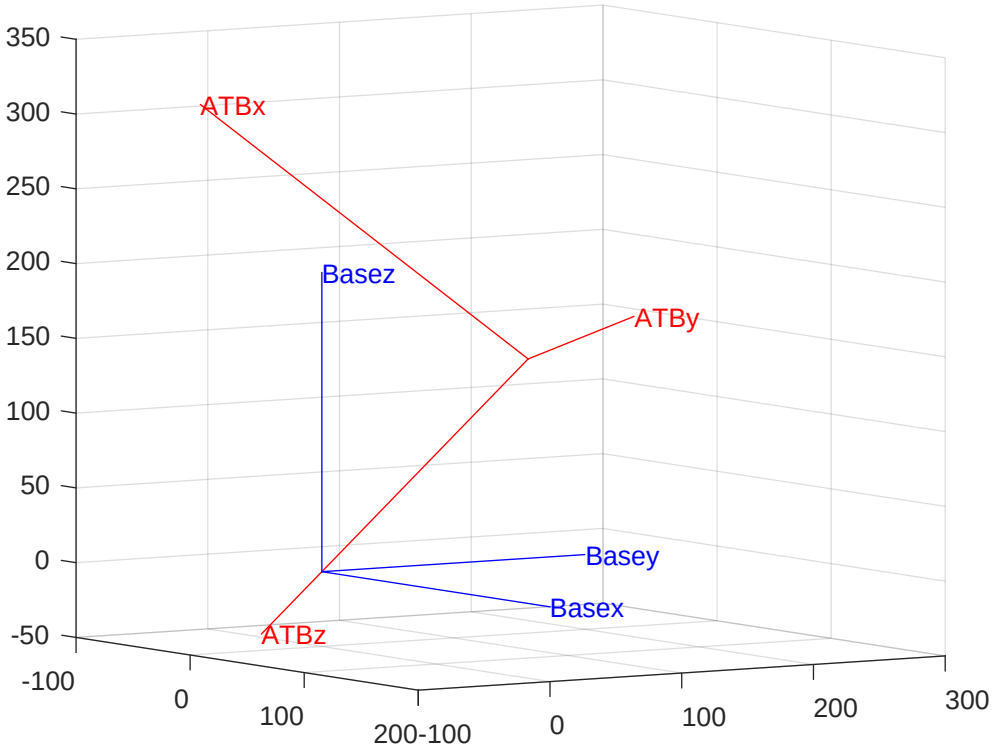


Figura 2: Representación de los sistemas.

Como se puede observar en la figura 2 el eje ${}^{A}T_{B_Z}$ cumple las especificaciones, la prolongación de dicho eje corta el origen el eje ${}^{A}T_{B_y}$ está contenida en un plano paralelo a la base.

Problema # 2

Se desea calcular la matriz de rotación que define la orientación de un sistema móvil $\{H\}$ con respecto a uno de referencias fijo $\{B\}$ definida de la siguiente forma:

Efectuar una rotación de un ángulo α respecto al eje X_B, seguida por un giro β respecto al eje Z_H y seguida por rotación de un ángulo γ respecto a Y_B.

- Deducir la representación en ángulos OAT que corresponde al enunciado propuesto aplicando la metodología denominada el problema inverso de la orientación.

  Implantarlo en MATLAB teniendo en cuenta las soluciones degeneradas.

```
function [O,A,T] = tr2OAT(matrix,m)
if nargin==1, m=1; end
M=sign(m);

% BTH_OAT(1,3)^2+BTH_OAT(2,3)^2 == r13^2+ r23^2
% r13^2+ r23^2 == (cos(A))^2
% r33 == -sin(A)

cos_a = M*sqrt(matrix(1,3)^2+matrix(2,3)^2);
A = atan2(-matrix(3,3),cos_a);

    if  abs(cos_a) > 1e-3
       T = atan2(matrix(3,2)/cos_a,-matrix(3,1)/cos_a);
       O = atan2(matrix(1,3)/cos_a,-matrix(2,3)/cos_a);
    else
       A = 0;
       warning('Configuracion degenerada')
       end
    end
```

Código 4: Función para obtener los valores de la representación *OAT*.

```
Identidad = eye(4);
syms alpha beta gamma real
BTH = rotY(gamma)*rotX(alpha)*Identidad*rotZ(beta)
alpha = deg2rad(45);
beta  = deg2rad(30);
gamma = deg2rad(60);
BTH = rotY(gamma)*rotX(alpha)*Identidad*rotZ(beta);
[O,A,T] = tr2OAT(BTH)
[O,A,T] = tr2OAT(BTH,-1)
```

Código 5: Ángulos OAT

La salida del código 5 es la siguiente:

$$\begin{pmatrix} \cos(\beta)\cos(\gamma)+\sin(\alpha)\sin(\beta)\sin(\gamma) & \cos(\beta)\sin(\alpha)\sin(\gamma)-\cos(\gamma)\sin(\beta) & \cos(\alpha)\sin(\gamma) & 0 \\ \cos(\alpha)\sin(\beta) & \cos(\alpha)\cos(\beta) & -\sin(\alpha) & 0 \\ \cos(\gamma)\sin(\alpha)\sin(\beta)-\cos(\beta)\sin(\gamma) & \sin(\beta)\sin(\gamma)+\cos(\beta)\cos(\gamma)\sin(\alpha) & \cos(\alpha)\cos(\gamma) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

```
O = 0.7137
A = -0.3614
T = 0.9112
O = -2.4279
A = -2.7802
T = -2.2304
```

- Volver a calcular la matriz de orientación detallada en el enunciado si la situación inicial entre el sistema $\{B\}$ y $\{H\}$ se define de la siguiente manera: El eje Z_H coincide con X_B, X_H con $-Y_B$ e Y_H con $-Z_B$. A partir de esta situación se realizarán las transformaciones dadas inicialmente.

  Probar la función elaborada con los siguientes datos: $\alpha = 45°$, $\beta = 30°$ y $\gamma = 60°$.

```
BTH = rotZ(deg2rad(-90))*Identidad*rotX(deg2rad(-90));
BTH_2 = rotY(gamma)*rotX(alpha)*BTH*rotZ(beta)
[O,A,T] = tr2OAT(BTH_2)
[O,A,T] = tr2OAT(BTH_2,-1)
```

Código 6: Ángulos OAT

Obtenemos los siguientes resultados, la nueva matriz y los ángulos OAT.

```
BTH_2 = 4×4
   -0.8365   -0.2241    0.5000         0
   -0.2588    0.9659         0         0
   -0.4830   -0.1294   -0.8660         0
         0         0         0    1.0000

O = 1.5708
A = 1.0472
T = -0.2618
O = -1.5708
A = 2.0944
T = 2.8798
```