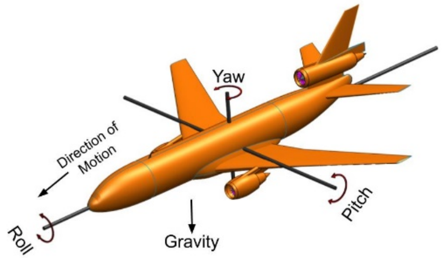
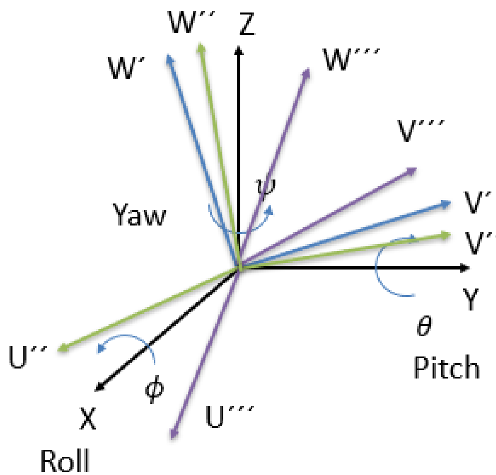
ROBÓTICA INDUSTRIAL



HERRAMIENTAS MATEMATICAS



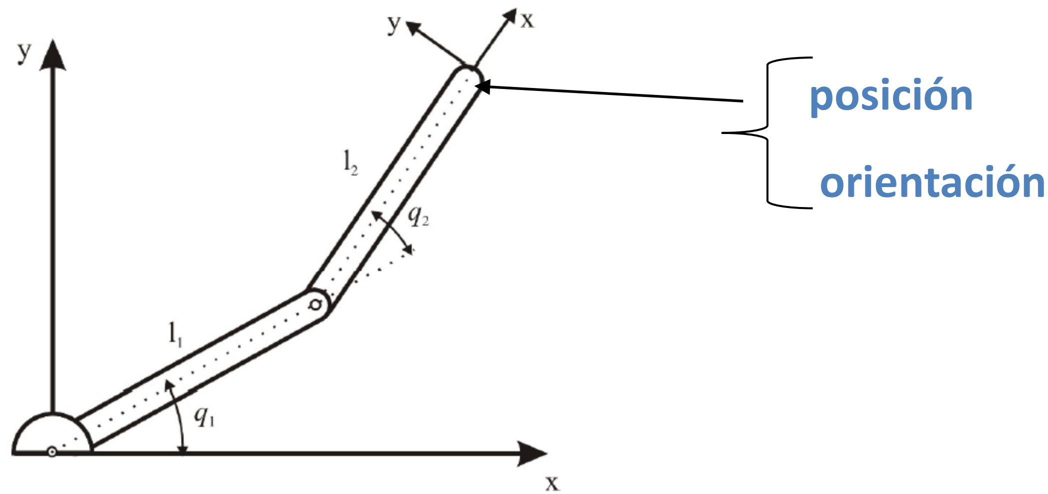
$$a_S = T \cdot a_{S'} = Rz(\psi) \cdot Ry(\theta) \cdot Rx(\phi) \cdot a_{S'}$$

Contenidos

- Representación de la posición
- Representación de la orientación
- Cuaternios
- Matrices de transformación homogénea MTH

Introducción

- La manipulación de piezas exige el movimiento espacial del extremo del robot



- Se necesitan herramientas matemáticas para especificar

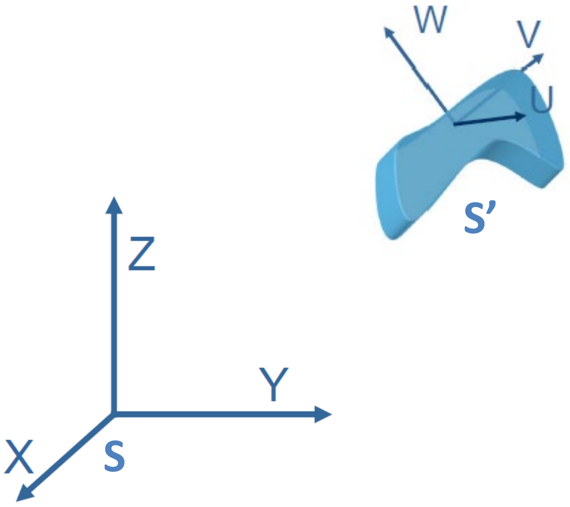
<u>posición y orientación</u> de las <u>**piezas**</u> y del <u>**extremo del robot**</u>

Localización espacial

- Necesitamos especificar la posición y orientación de un sólido rígido en el espacio, con respecto a un sistema de referencia fijo **S={O,X,Y,Z}**

- Al sólido rígido se le asocia a un sistema de coordenadas (dextrógiro)
S'= {O',U,V,W}

- La posición y orientación del sólido con respecto al sistema S, queda totalmente determinada por la del sistema S´ con respecto a S

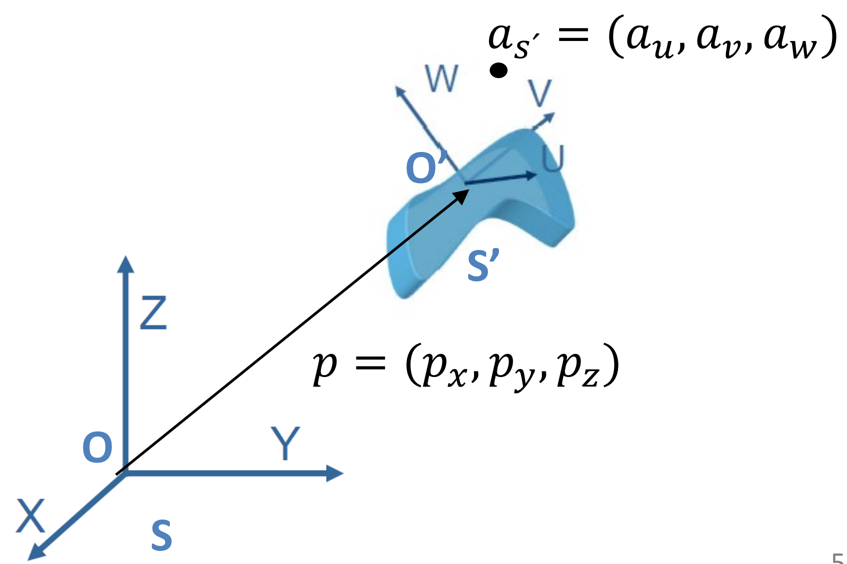


Representación de la posición

- **Vector de posición: p**

Une el punto O con el punto O'

- **Coordenadas cartesianas**
- Cilíndricas
- Esféricas



- Permite conocer las coordenadas del punto **a** en S (a_S), a partir de las coordenadas en S' $(a_{S'})$

$$a_S = p_S + a_{S'}$$

TRASLACIÓN

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} + \begin{bmatrix} a_u \\ a_v \\ a_w \end{bmatrix}$$

Matrices de rotación 2D

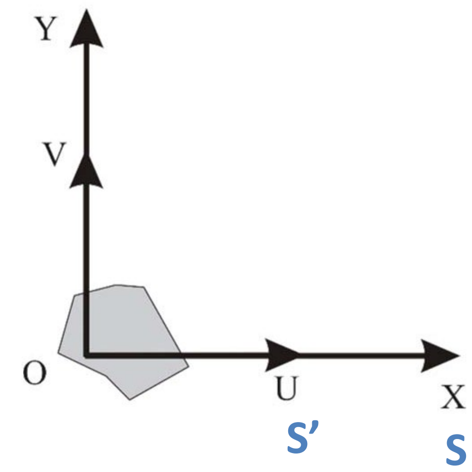
- El sistema S' es el resultado de rotar un ángulo α el sistema S.

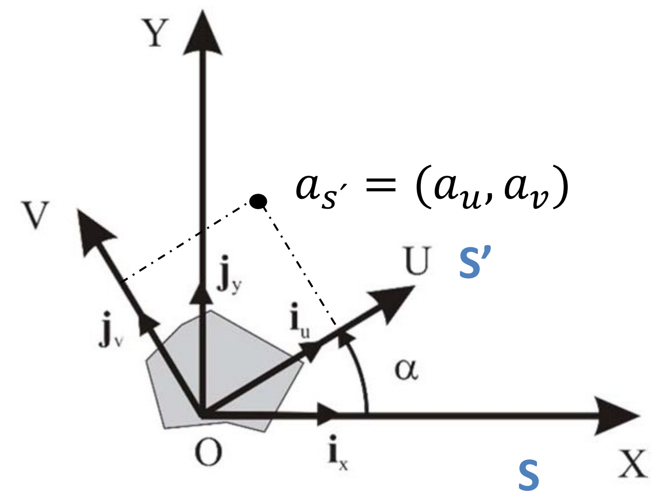
- Permite conocer las coordenadas del punto **a** en S (a_s), a partir de las coordenadas en S' ($a_{s'}$)

$$a_S = R \cdot a_{S'}$$

$$\begin{bmatrix} a_x \\ a_y \end{bmatrix}_S = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} \begin{bmatrix} a_u \\ a_v \end{bmatrix}_{S'}$$

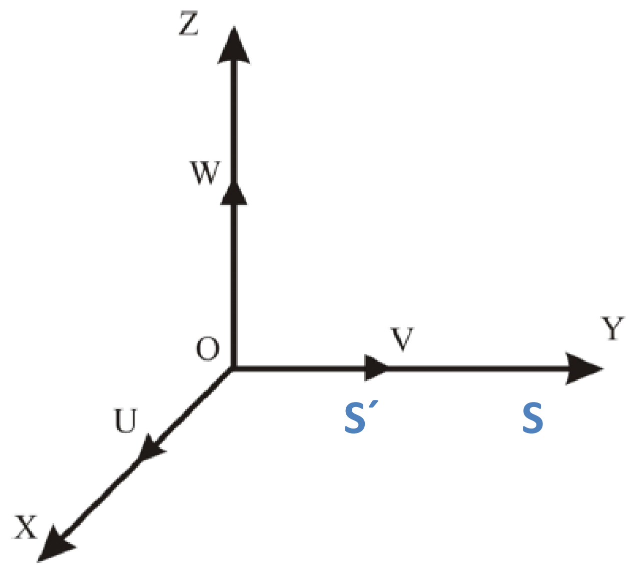
Por ejemplo: $\begin{bmatrix} 1 \\ 0 \end{bmatrix}_{S'} \rightarrow \begin{bmatrix} 0.866 \\ 0.5 \end{bmatrix}_S$





Representación de la orientación
Matrices de rotación 3D

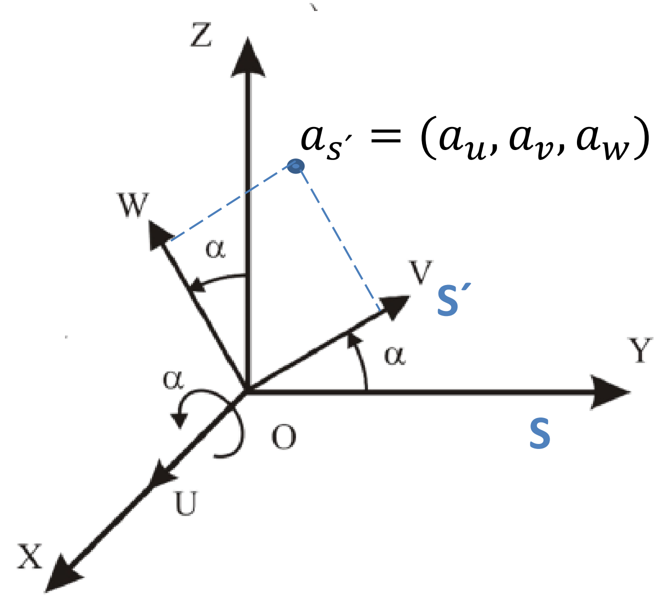
- El sistema S’ es el resultado de rotar un ángulo α el sistema S sobre uno de sus ejes → 3 matrices diferentes Rx, Ry, Rz



$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}_S = R(\alpha) \begin{bmatrix} a_u \\ a_v \\ a_w \end{bmatrix}_{S'} \qquad a_S = R \cdot a_{S'}$$

$$R\boldsymbol{x}(\alpha) = \begin{bmatrix} \mathbf{1} & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix}$$

Ejemplo Rx



$$R\boldsymbol{y}(\alpha) = \begin{bmatrix} \cos(\alpha) & 0 & \sin(\alpha) \\ 0 & \mathbf{1} & 0 \\ -\sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix}$$

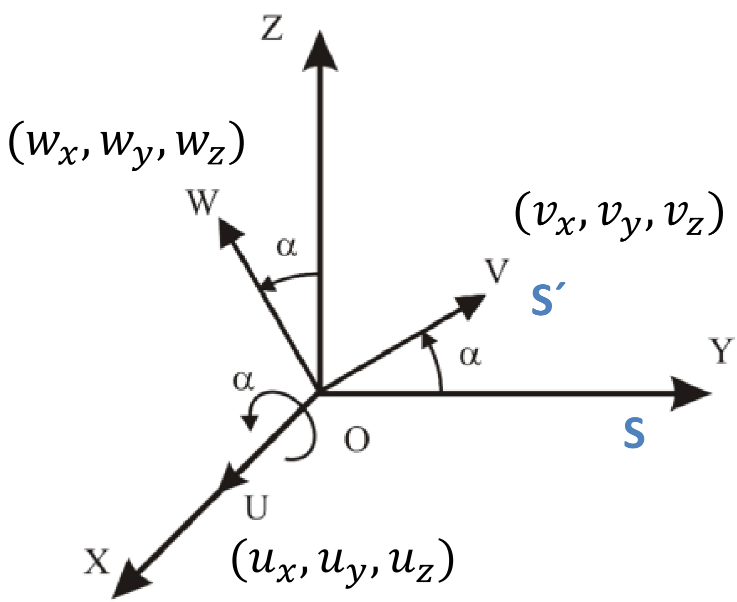
$$R\boldsymbol{x}(\alpha) = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & \mathbf{1} \end{bmatrix}$$

Matrices de rotación 3D. Interpretación geométrica

- Si $R_{S\rightarrow S'}$ es la matriz que relaciona {UVW} con {XYZ}

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}_S = R(\alpha) \begin{bmatrix} a_u \\ a_v \\ a_w \end{bmatrix}_{S'}$$

- Entonces las columnas de la matriz $R_{S\rightarrow S'}$ son las coordenadas de los vectores {UVW} en la base S



$$R_{S\rightarrow S'} = \begin{pmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{pmatrix}$$

Matrices de rotación 3D. Propiedades

- Son matrices **ortonormales**:
  - Sus vectores por columnas o por filas son ortonormales entre si:
    - Producto escalar
      - de un vector por otro cualquiera = 0
      - de un vector por si mismo =1
    - Producto vectorial
      - de un vector por el siguiente da el tercero
- Su Inversa coincide con su Traspuesta $R^{-1} = R^T$
- Su determinante es la unidad $|R| = 1$

Matrices de rotación 3D. Usos

La matriz de rotación R permite:

- Representar la orientación del sistema móvil S’ con respecto al fijo S

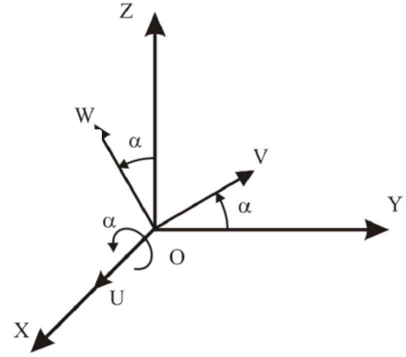
$$S' = R(\alpha) \cdot S$$

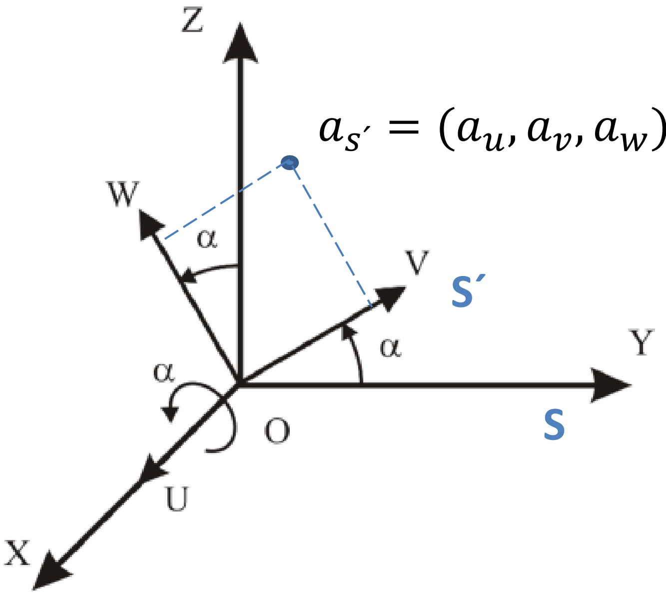
- Obtener las coordenadas de un punto en el sistema S, conocidas sus coordenadas en el sistema móvil S’

$$a_S = R(\alpha) \cdot a_{S'}$$

- Obtener en que punto **b** se convierte un punto **a** si se rota.

$$b_S = R(\alpha) \cdot a_S$$





Matrices de rotación 3D. Composición

- Si queremos hacer varias rotaciones seguidas podemos concatenarlas mediante la multiplicación de matrices.
- Las rotaciones se especifican **respecto del sistema fijo** S(0,X,Y,Z)

PREMULTIPLICAR

Orden:

Rotación OX

Rotación OY

Rotación OZ

$$a_S = T \cdot a_{S'} = Rz \cdot Ry \cdot Rx \cdot a_{S'}$$

$$a_S = T \cdot a_{S'}$$

$$= \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & \mathbf{1} \end{bmatrix} \cdot \begin{bmatrix} \cos(\alpha) & 0 & \sin(\alpha) \\ 0 & \mathbf{1} & 0 \\ -\sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix} \cdot \begin{bmatrix} \mathbf{1} & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \cdot a_{S'}$$

Representación de la orientación

Matrices de rotación 3D. Composición

- Si queremos hacer varias rotaciones seguidas podemos concatenarlas mediante la multiplicación de matrices.
- Las rotaciones se especifican **respecto del sistema fijo** S(0,X,Y,Z)

PREMULTIPLICAR

Orden:

$$a_S \; = T \cdot \; a_{S'} = Rx \cdot Rz \cdot Ry \cdot \; a_{S'}$$

Rotación OY

Rotación OZ

Rotación OX

$$a_S \; = T \cdot \; a_{S'} = \cdot \begin{bmatrix} \mathbf{1} & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix}$$

$$\begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & \mathbf{1} \end{bmatrix} \cdot \begin{bmatrix} \cos(\alpha) & 0 & \sin(\alpha) \\ 0 & \mathbf{1} & 0 \\ -\sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix} a_{S'}$$

Otro ejemplo

Matrices de rotación 3D. Composición

- Si especificamos las rotaciones **respecto del sistema móvil** S'(0',U,V,W)

POSTMULTIPLICAR

$$a_S = T \cdot a_{S'} = Rx \cdot Ry \cdot Rz \cdot a_{S'}$$

Orden:

Rotación OX

Rotación O'V

Rotación O'W

$$a_S = T \cdot a_{S'} = \begin{bmatrix} \mathbf{1} & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \cdot \begin{bmatrix} \cos(\alpha) & 0 & \sin(\alpha) \\ 0 & \mathbf{1} & 0 \\ -\sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix} \cdot \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & \mathbf{1} \end{bmatrix} \cdot a_{S'}$$

Matrices de rotación 3D. Composición

- Podemos combinar rotaciones respecto del sistema **fijo** (PREMULTIPLICAR)con rotaciones respecto del sistema **móvil** (POSTMULTIPLICAR)

Orden:

Rotación OX

Rotación O'V

Rotación O'Z

$$a_S = T \cdot a_{S'} = Rz \cdot Rx \cdot Ry \cdot a_{S'}$$

$$a_S = T \cdot a_{S'} = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & \mathbf{1} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{1} & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \cdot \begin{bmatrix} \cos(\alpha) & 0 & \sin(\alpha) \\ 0 & \mathbf{1} & 0 \\ -\sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix} \cdot a_{S'}$$

Representación de la orientación

Matrices de rotación 3D. Ventajas e inconvenientes

- Su composición se realiza mediante el álgebra de matrices (facilidad de uso) ✓
- Precisan 9 elementos (redundancia) ✗
- Riesgo de inconsistencia tras varias operaciones (redondeos)
  - Adecuadas para la formulación y el cálculo manual ✓
  - Inadecuadas para el calculo computacional ✗
- No podemos interpolar. Las matrices intermedias no son ortonormales ✗

Representación de la orientación

Ángulos de Euler

- Según el teorema de la rotación de Euler, cualquier rotación puede describirse usando giros consecutivos entorno a 3 ejes, debiendo ser perpendicular cada eje con el siguiente.
- Posibilidad de especificar los giros sobre ejes fijos (XYZ) o sobre ejes móviles (UVW)
- 12 combinaciones posibles sobre ejes fijos y 12 sobre ejes móviles

Ejes fijos

XYZ	XYX
XZY	XZX
YXZ	YXY
YZX	YZY
ZXY	ZXZ
ZYX	ZYZ

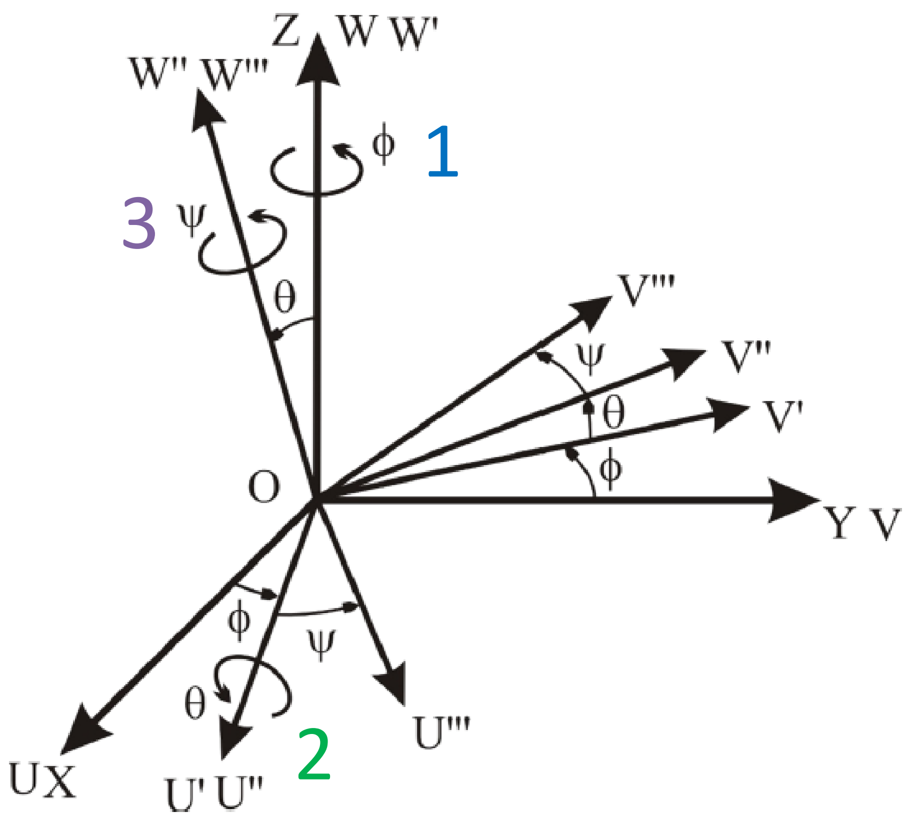
Ejes móviles

UVW	UVU
UWV	UWU
VUW	VUV
VWU	VWV
WUV	**WUW**
WVU	**WVW**

Más comunes $\left\{ \begin{array}{l} \textbf{XYZ} \\ \textbf{WUW} \\ \textbf{WVW} \end{array} \right.$

Ángulos de Euler. WUW

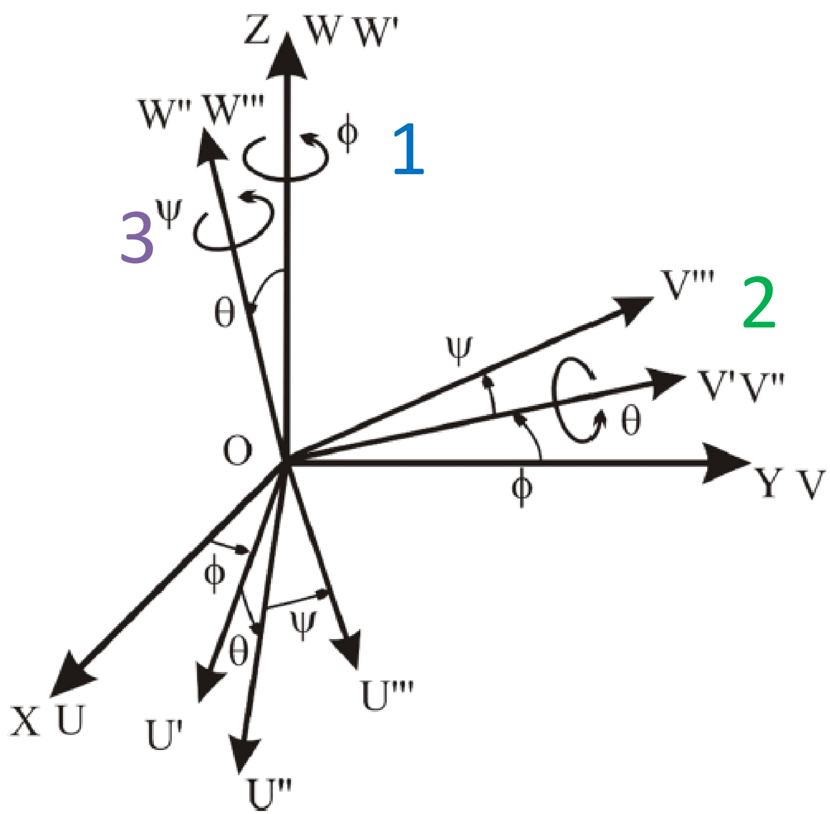
- Giramos sobre OZ un ángulo ϕ convirtiéndose en OU’V’W’
- Giramos OU’V’W’ sobre OU´ un ángulo θ convirtiéndose en OU’’V’’W’’
- Giramos OU’’V’’W’’ sobre OW´´ un ángulo ψ convirtiéndose en OU’’’V’’’W’’’



$$a_S = T \cdot a_{S'} = Rz(\phi) \cdot Rx(\theta) \cdot Rz(\psi) \cdot a_{S'}$$

Representación de la orientación
Ángulos de Euler. WVW

- Giramos sobre OZ un ángulo ϕ convirtiéndose en OU’V’W’
- Giramos OU’V’W’ sobre OV´ un ángulo θ convirtiéndose en OU’’V’’W’’
- Giramos OU’’V’’W’’ sobre OW´´ un ángulo ψ convirtiéndose en OU’’’V’’’W’’’

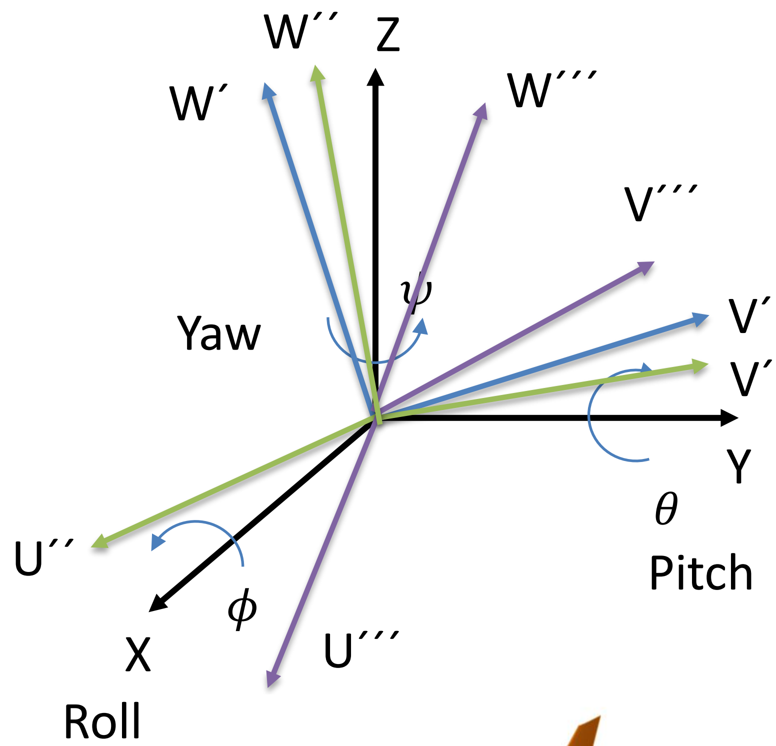


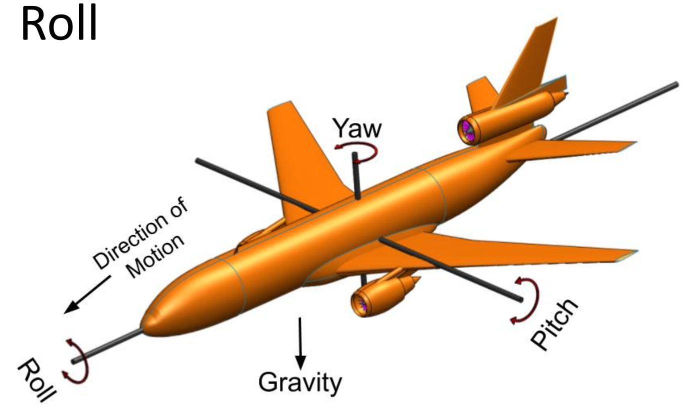
$$a_S = T \cdot a_{S'} = Rz(\phi) \cdot Ry(\theta) \cdot Rz(\psi) \cdot a_{S'}$$

Representación de la orientación
Ángulos de Euler. XYZ

- Giramos sobre OX un ángulo ϕ convirtiéndose en OU’V’W’
- Giramos OU’V’W’ sobre OY un ángulo θ convirtiéndose en OU’’V’’W’’
- Giramos OU’’V’’W’’ sobre OZ un ángulo ψ convirtiéndose en OU’’’V’’’W’’’

$$a_S = T \cdot a_{S'} = Rz(\psi) \cdot Ry(\theta) \cdot Rx(\phi) \cdot a_{S'}$$





Ángulos de Euler. Ventajas e inconvenientes

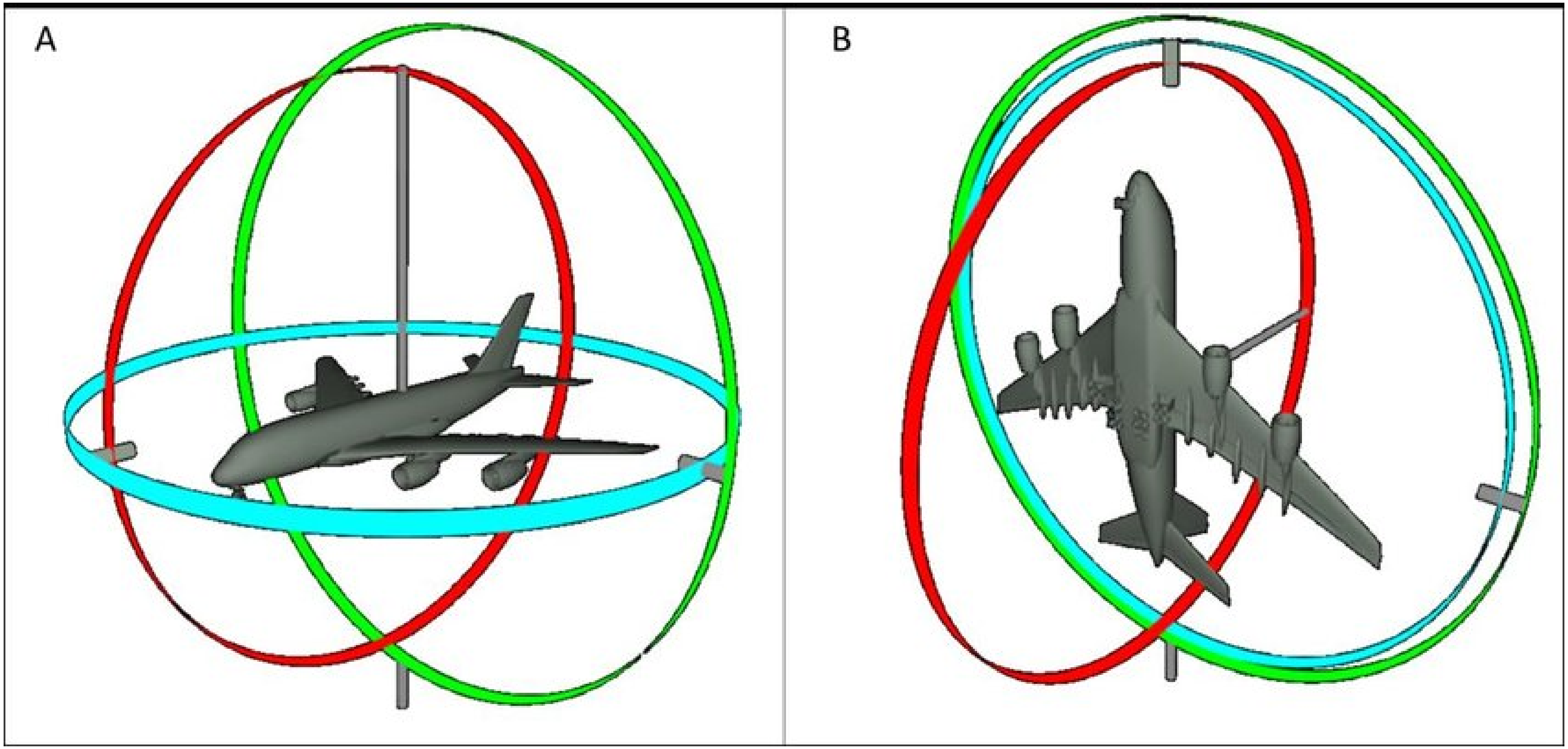
- No existe un álgebra para concatenar o componer rotaciones ✗
- Precisan 3 elementos ✓
- Poco intuitivos
  - Inadecuadas para la formulación y el cálculo manual ✗
  - Inadecuados para el cálculo computacional ✗
- Gimbal LOCK: Pérdida de un grado de libertad cuando dos ejes están alineados ✗
- Podemos interpolar ✓

Representación de la orientación

Ángulos de Euler. Gimbal Lock

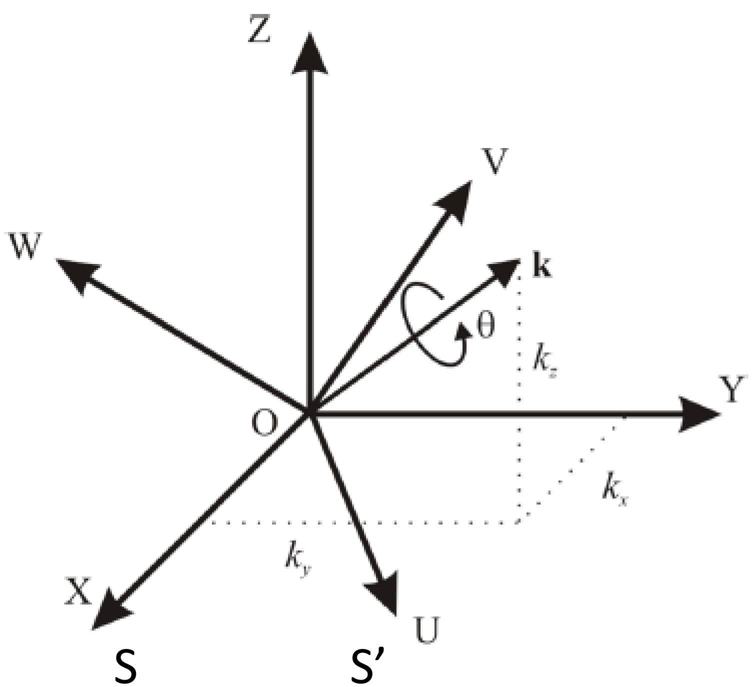
- Gimbal LOCK: Pérdida de un grado de libertad cuando dos ejes están alineados

En el ejemplo Roll y Yaw producen el mismo efecto



Par de rotación

- Dados los sistemas {O,X,Y,Z} y {O',U,V,W}, existe un único vector $\bar{k}\ (kx, ky, kz)$, tal que al girar alrededor de él un ángulo θ se convierte el primer sistema en el segundo. $Rot(k, \theta)$



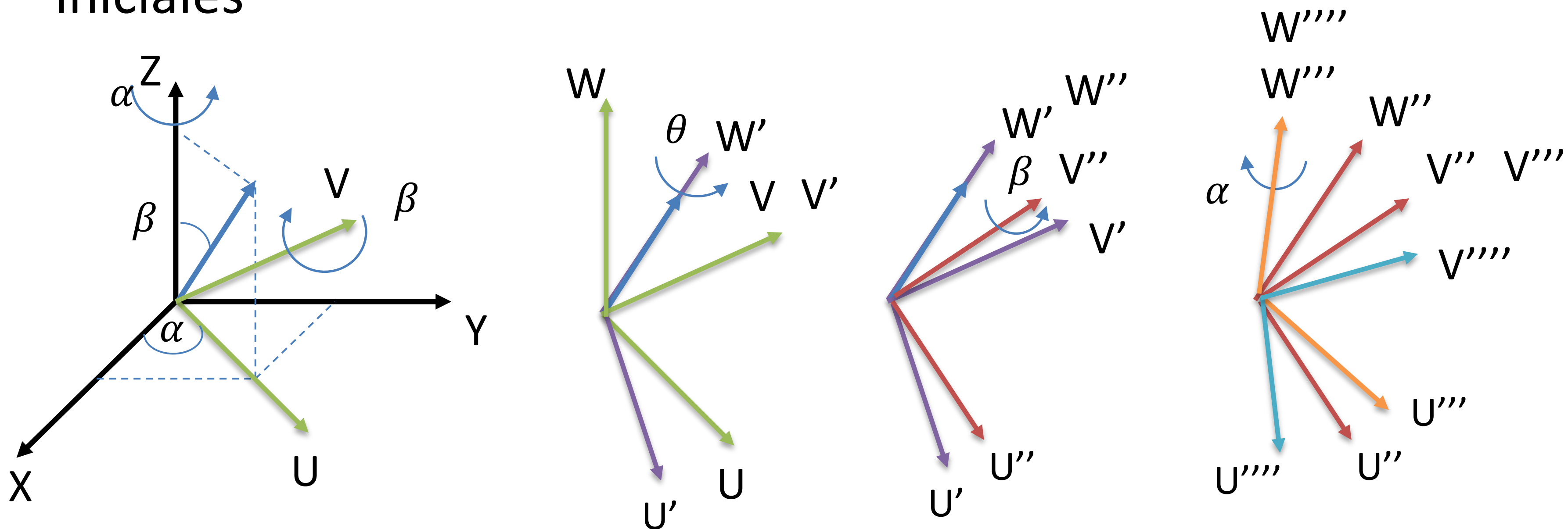
$$a_S = R_k(\theta) \cdot a_{S'}$$

¿Cómo obtenemos $R_k(\theta)$?

- Representamos la orientación con 4 elementos: (θ, kx, ky, kz)

Representación de la orientación
Par de rotación. Obtención de $R_k(\theta)$

- La matriz R_k haciendo coincidir Z del sistema base con el vector $\bar{k}$, después se rota θ y se deshacen las rotaciones iniciales



Alineación Z con K

Deshacemos giros

$$R_k(\theta) = \underbrace{R_z(\alpha)R_y(\beta)}_{\text{Alineación Z con K}}\ R_z(\theta)\ \underbrace{R_y(-\beta)\ R_z(-\alpha)}_{\text{Deshacemos los giros}}$$

$$\alpha = \operatorname{atan}\left(\frac{Ky}{Kx}\right)$$

$$\beta = \operatorname{atan}\left(\frac{\sqrt{Kx^2 + Ky^2}}{Kz}\right)$$

Representación de la orientación

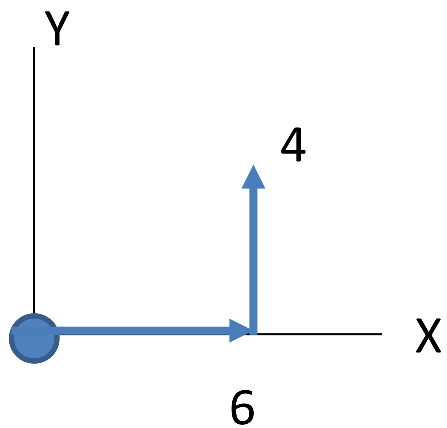
Par de rotación. Ventajas e inconvenientes

- No existe un álgebra para concatenar o componer rotaciones ✗
- Solo precisan 4 elementos ✓
- Poco intuitivos
  - Inadecuadas para la formulación y el cálculo manual ✗
  - Inadecuados para el cálculo computacional ✗
- No hay gimbal lock ✓
- Podemos interpolar ✓

Cuaternios

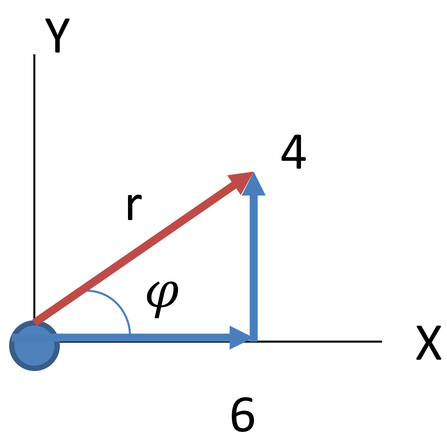
Aproximación histórica desde los números complejos

- Imaginemos que queremos viajar en el plano desde (0,0) a una posición (x,y) por ejemplo (6, 4).



- Avanzamos 6 posiciones a la derecha, y 4 posiciones a la izquierda

- Este desplazamiento lo puedo representar con un número complejo: **6 + 4i**



- Es equivalente a un giro de un ángulo φ y un desplazamiento r. (r, φ)

$$r = \sqrt{6^4 + 4^2} \qquad \varphi = \operatorname{atan}(4/6)$$

Cuaternios
Aproximación histórica desde los números complejos

Números complejos:

- $(a_0, a_1) = a_0 + a_1 \cdot i = A_\varphi = A cos(\varphi) + A sin(\varphi) \cdot i \qquad i^2 = -1$

Producto:

- $(a_0 + a_1 \cdot i) * (b_0 + b_1 \cdot i) = (a_0 b_0 - a_1 b_1) + (a_0 b_1 - a_1 b_0)\ i$
- $A_{\varphi_a} * B_{\varphi_b} = \underbrace{(A \cdot B)}_{\text{Cambio de desplazamiento}}{}_{\underbrace{\varphi_a + \varphi_b}_{\text{Cambio de orientación}}}$

Por tanto **si el número B tiene norma 1, se produce una rotación pura, sin cambio de desplazamiento**

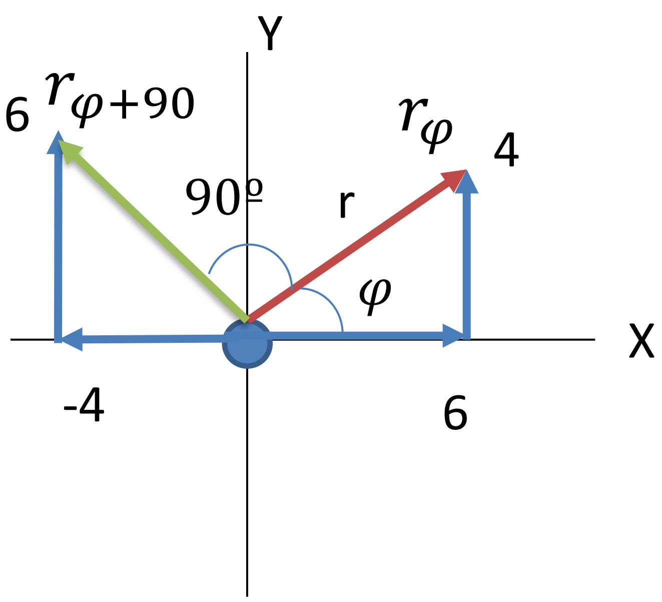
Por ejemplo: $(0\ +\ i)\ \rightarrow 1_{90º}$ rotación de 90º

Aproximación histórica desde los números complejos

- Por ejemplo rotamos el punto (6, 4)   90º.

$$r_\varphi * (0, \mathrm{i}) = (6 + 4\mathrm{i})\mathrm{i} = -4 + 6\mathrm{i}$$

$$r_\varphi * 1_{90º} = r_{\varphi+90}$$



Aproximación histórica desde los números complejos

Matrices de rotación

- <u>Rotamos un vector</u> multiplicándole por una matriz ortonormal

$$\begin{bmatrix} b_x \\ b_y \end{bmatrix}_S = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} \begin{bmatrix} a_x \\ a_y \end{bmatrix}_S$$

- <u>Componemos rotaciones</u> mediante productos de matrices ortonormales.

$$\begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} \begin{bmatrix} \cos(\beta) & -\sin(\beta) \\ \sin(\beta) & \cos(\beta) \end{bmatrix}$$

Números complejos

- <u>Rotamos un vector</u> multiplicándole por un número complejo de norma 1

$$\begin{bmatrix} b_x \\ b_y \end{bmatrix}_S = 1_\alpha \begin{bmatrix} a_x \\ a_y \end{bmatrix}_S$$

- <u>Componemos rotaciones</u> mediante productos de números complejos de norma 1.

$$1_\alpha * 1_\beta$$

Cuaternios

Aproximación histórica desde los números complejos

- Si multiplicar por un número complejo equivale a una rotación + un desplazamiento en 2D. ¿Podemos hacer esto en 3D?
- A este problema se enfrento William Rowan Hamilton
- ¿Podemos extender los números complejos para representar rotaciones en 3D?

Cuaternios
Aproximación histórica desde los números complejos

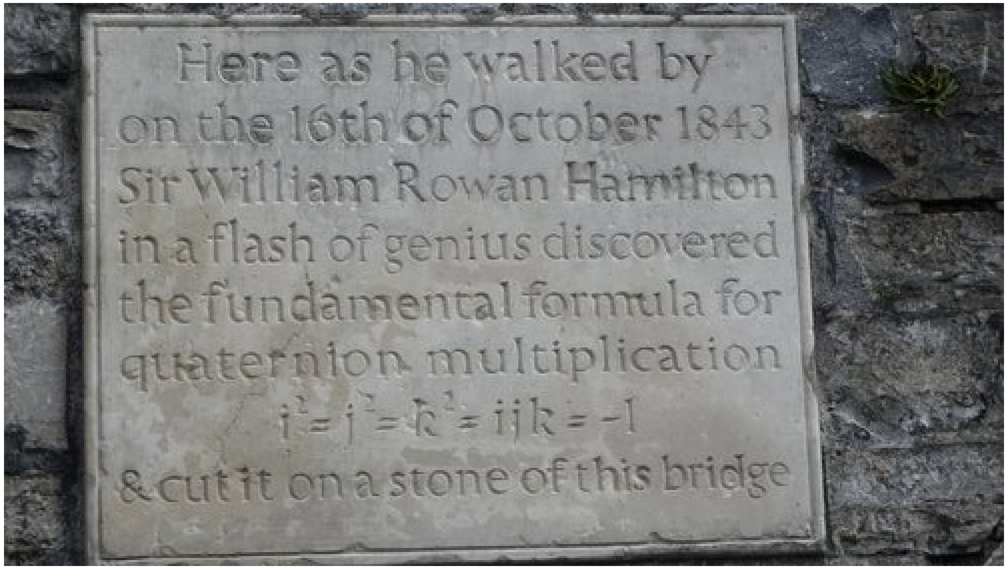
- La primera aproximación directa: Si un número complejo de 2 componentes me sirve en 2D, para 3D necesitaré 3 componentes

- Lo intentó con las tripletas $(a_o + a_1 i + a_2 j),\ i^2 = j^2 = -1$

- La suma le funcionaba pero <u>la multiplicación no</u>

- Se dice que Hamilton descubrió el producto de cuaternios o cuaterniones cuando paseaba por el puente de Brougham Bridge de Dublín y lo anoto en la piedra

$$i^2 = j^2 = k^2 = ijk = -1$$



Cuaternios

Definiciones

- Un cuaternio esta compuesto por 4 componentes $Q = (q_o, q_1, q_2, q_3)$ que lo representan en la base {e,i,j,k}

- Tiene diferentes representaciones:
  - 4 componentes:
    - $Q = (q_o, q_1, q_2, q_3)$
    - $Q = q_0 + q_1 i + q_2 j + q_3 k$

$$i * j = -j * i = k$$
$$j * k = -k * j = i$$
$$k * i = -i * k = j$$

  - Como vector $Q = (\mathrm{s}, \mathrm{v})$
    - s → parte escalar
    - v→ parte vectorial

∘	e	i	j	k
e	e	i	j	k
i	i	-e	k	-j
j	j	-k	-e	i
k	k	j	-i	-e

  - Como un numero complejo
    - $Q = (q_0 + q_1 i) + (q_2 + q_3 i)j$

De esta forma nos permite aplicar el algebra de los números complejos

Cuaternios
Operaciones

- SUMA → Elemento a elemento $(a_o + b_o, a_1 + b_1, a_2 + b_2, a_3 + b_3)$
- PRODUCTO
  - Propiedad distributiva producto y suma
  - Teniendo en cuenta que $i^2 = j^2 = k^2 = ijk = -1$

$$(a_0 + a_1 i + a_2 j + a_3 k)(b_0 + b_1 i + b_2 j + b_3 k) = a_0 (b_0 + b_1 i + b_2 j + b_3 k) + a_1 i (b_0 + b_1 i + b_2 j + b_3 k) + a_2 j (b_0 + b_1 i + b_2 j + b_3 k) + a_3 k (b_0 + b_1 i + b_2 j + b_3 k)$$

De forma vectorial:

Q=$Q = Q1 \circ Q2 = (s_1, v_1) \circ (s_2, v_2) = (s_1 s_2 - v_1 v_2, v_1 \times v_2 - s_1 v_2 + s_2 v_1)$

Computacionalmente

$$q_{30} = q_{10} q_{20} - (q_{11} q_{21} + q_{12} q_{22} + q_{13} q_{23})$$
$$q_{31} = q_{10} q_{21} + q_{11} q_{20} + q_{12} q_{23} - q_{13} q_{22}$$
$$q_{32} = q_{10} q_{22} + q_{12} q_{20} + q_{13} q_{21} - q_{11} q_{23}$$
$$q_{33} = q_{10} q_{23} + q_{13} q_{20} + q_{11} q_{22} - q_{12} q_{21}$$

16 multiplicaciones
12 sumas

Matrices de rotacion

27 multiplicaciones
18 sumas

Cuaternios
Algebra de cuaternios

SUMA $Q_3 = Q_1 + Q_2 = (s_1, \mathbf{v}_1) + (s_2, \mathbf{v}_2) = (s_1 + s_2, \mathbf{v}_1 + \mathbf{v}_2)$

PRODUCTO $Q_3 = Q_1 \circ Q_2 = (s_1, \mathbf{v}_1) \circ (s_2, \mathbf{v}_2) = (s_1 s_2 - \mathbf{v}_1 \mathbf{v}_2, \mathbf{v}_1 \times \mathbf{v}_2 + s_1 \mathbf{v}_2 + s_2 \mathbf{v}_1)$

MULTIPLICACIÓN POR ESCALAR $Q_3 = a Q_2 = a(s_2, \mathbf{v}_2) = (a s_2, a \mathbf{v}_2)$

CUATERNIO CONJUGADO $Q^* = [q_0, -q_1, -q_2, -q_3] = (s, -\mathbf{v})$

NORMA $Q \circ Q^* = (q_0^2 + q_1^2 + q_2^2 + q_3^2)\mathbf{e}$

CUATERIO INVERSO $Q^{-1} = \frac{Q^*}{\|Q\|}$

Rotaciones con cuaternios

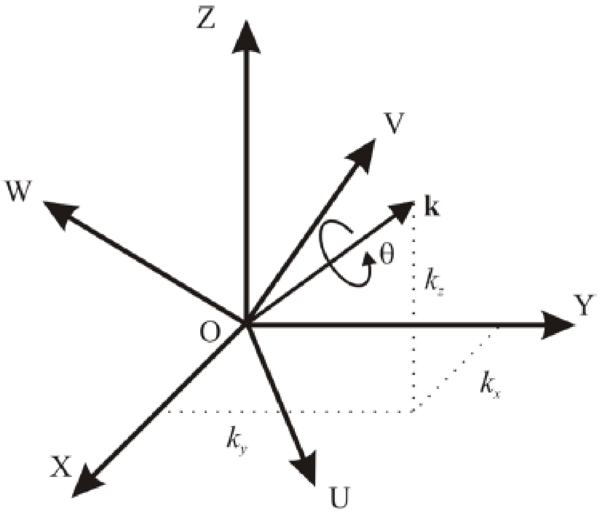
- Al igual que con los números complejos para hacer rotaciones puras puras la norma debe ser 1. **Cuaternio unitario**

$$|Q| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} = 1$$

- Al par de rotación $Rot(k, \theta)$ se le asocia un **cuaternio unitario**.

$$Rot(k, \theta) \rightarrow Q = (cos\left(\frac{\theta}{2}\right), \boxed{\frac{\bar{k}}{|\bar{k}|}} sin\left(\frac{\theta}{2}\right))$$

Aseguramos que k es unitario



$$|Q| = \sqrt{C_\theta^2 + S_\theta^2(k_x^2 + k_y^2 + k_z^2)} = \sqrt{C_\theta^2 + S_\theta^2} = 1$$

Cuaternios
Rotaciones con cuaternios

- Para rotar un punto $a_s$en S con un cuaternio Q, primero lo convertimos en cuaternio de escalar 0. $\mathrm{Q}_a = (0, a_S)$
- Después realizamos la operación

$$\boxed{b_S = Q \circ \mathrm{Q}_a \circ Q^* = Q \circ (0, a_s) \circ Q^*}$$

- La composición de rotaciones la realizamos la realizamos con el producto de cuaternios

$$Q = \mathrm{Q}_1 \circ Q_2$$

- La rotación inversa se realiza con el cuaternio inverso $Q^{-1} = \frac{Q^*}{\|Q\|}$

Como $\|Q\|$=*1* por ser unitario→Q^{-1}= Q^*

$$Q^{-1} = \left(cos\left(\frac{\theta}{2}\right), k \cdot \sin\left(\frac{\theta}{2}\right)\right)^{-1} = \left(cos\left(\frac{\theta}{2}\right), -k \cdot \sin\left(\frac{\theta}{2}\right)\right)$$

Rotaciones con cuaternios. Ejemplos

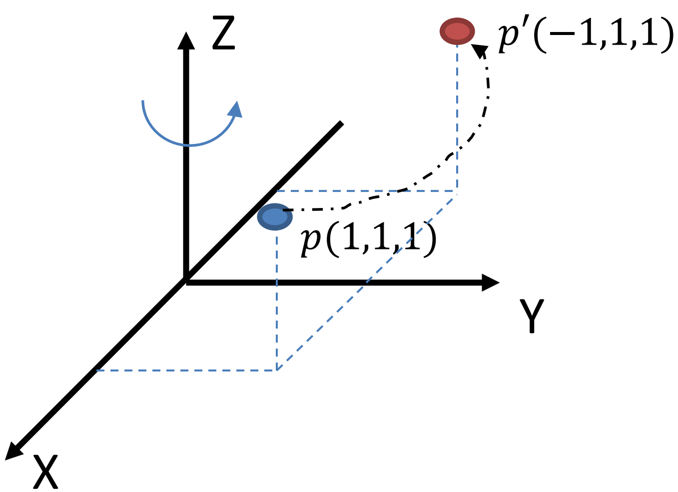
- Cuaternio unitario que represente un giro de 90º en torno al eje Z

$$Rot(k,\theta) \rightarrow Q = (cos\left(\frac{\theta}{2}\right), \frac{\bar{k}}{|\bar{k}|} sin\left(\frac{\theta}{2}\right))$$

$$Rot((0,0,1),\pi/2) \rightarrow Q = \left[cos\left(\frac{\frac{\pi}{2}}{2}\right), sin\left(\frac{\theta}{2}\right)(0,0,1)\right] = \left[\frac{\sqrt{2}}{2}, \left(0,0,\frac{\sqrt{2}}{2}\right)\right]$$

- Rotación del punto p (1,1,1) con el cuaternio anterior

$$p' = Q \circ (0,p) \circ Q^* = \left[\frac{\sqrt{2}}{2}, \left(0,0,\frac{\sqrt{2}}{2}\right)\right] [0,(1,1,1)] \left[\frac{\sqrt{2}}{2}, \left(0,0,-\frac{\sqrt{2}}{2}\right)\right] = [0,(-1,1,1)]$$



Usos

Los cuaternios unitarios nos permiten:

- Representar la orientación del sistema móvil S' con respecto al fijo S

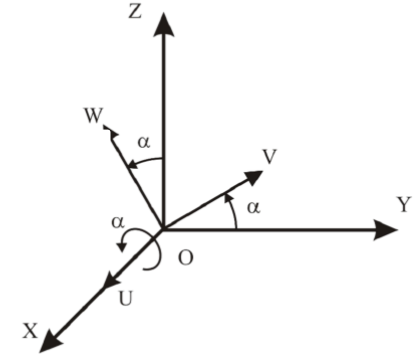
$$Q = (cos\left(\frac{\theta}{2}\right), k \cdot sin\left(\frac{\theta}{2}\right))$$

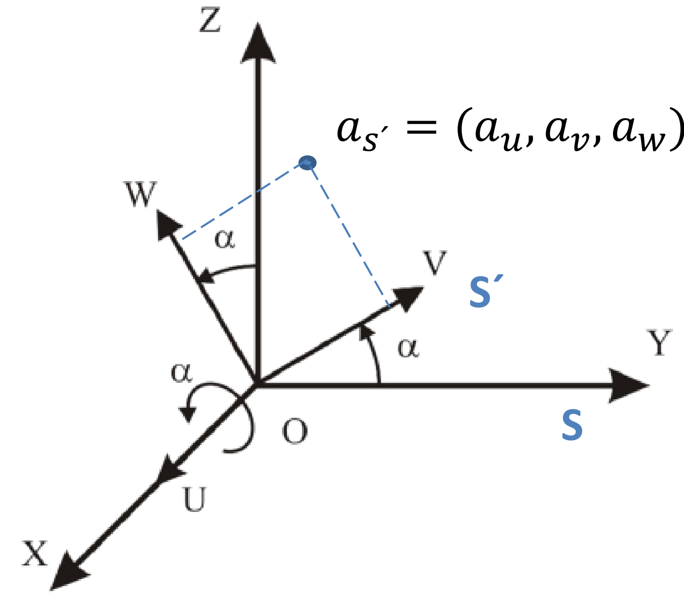
- Obtener las coordenadas de un punto en el sistema S, conocidas sus coordenadas en el sistema móvil S'

$$a_S = Q \circ (0, a_{S'}) \circ Q^*$$

- Obtener en que punto **b** se convierte un punto **a** si se rota.

$$b_S = Q \circ (0, a_S) \circ Q^*$$





Composición

- Podemos combinar rotaciones respecto del sistema **fijo** (PREMULTIPLICAR)con rotaciones respecto del sistema **móvil** (POSTMULTIPLICAR)

Orden:

Rotación OX (α)

Rotación O'V(β)

Rotación O'Z(γ)

$$a_S = Q \circ (0, a_{S'}) \circ Q^*; Q = Qz \cdot Qx \cdot Qy$$

$$= \left[cos\left(\frac{\gamma}{2}\right), sin\left(\frac{\gamma}{2}\right)(0,0,1)\right] \cdot \left[cos\left(\frac{\alpha}{2}\right), sin\left(\frac{\alpha}{2}\right)(1,0,0)\right]$$

$$\cdot \left[cos\left(\frac{\beta}{2}\right), sin\left(\frac{\beta}{2}\right)(0,1,0)\right]$$

Cuaternios
Ventajas e inconvenientes

- Existe un álgebra para concatenar o componer rotaciones ✓
- Solo precisan 4 elementos (poca redundancia) ✓
- Computacionalmente eficientes ✓
- No hay gimbal lock ✓
- Podemos interpolar ✓
- Al principio son difíciles de entender. Algebra poco conocida ✗
- Muy utilizados por fabricantes de robot (ABB) y herramientas de simulación (ROS, Unity, …)

Matrices de transformación homogénea (MTH)
Coordenadas generalizadas

- Coordenadas de un espacio de (n+1) dimensiones para representar objetos de n dimensiones

- Convertimos un punto $p(x, y, z) \rightarrow p(x, y, z, w)$ donde w es el factor de escala

$$p = \begin{bmatrix} a \\ b \\ c \\ 1 \end{bmatrix}_{S'} = \begin{bmatrix} aw \\ bw \\ cw \\ w \end{bmatrix} = \begin{bmatrix} a \\ b \\ c \\ 1 \end{bmatrix}_{S}$$

- Ejemplo: w=2 $\quad p = [2,3,4] \rightarrow [4,6,8,2]$

Definición

- Matriz 4x4 que representa la transformación de un vector en coordenadas homogéneas de un sistema de coordenadas a otro

- Convertimos un punto $p(x, y, z) \rightarrow p(x, y, z, w)$ donde w es el factor de escala

$$T = \begin{bmatrix} R_{3\times3} & p_{3\times1} \\ f_{1\times3} & w_{1\times1} \end{bmatrix} = \begin{bmatrix} Rotacion & Traslacion \\ Perspectiva & escalado \end{bmatrix}$$

- En robótica:
  - $f_{1\times3} = [0,0,0]$
  - $w_{1\times1} = 1$

$$T = \begin{bmatrix} R_{3\times3} & p_{3\times1} \\ 0 & 1 \end{bmatrix}$$

OJO NO SON MATRICES ORTONORMALES

Usos

La matriz de transformación homogénea MTH permite:

- Representar la orientación del sistema móvil S' **<u>girado y trasladado</u>** con respecto al fijo S

$$S' = R_{3\times3} \cdot S + p_{3\times1} = T \cdot S$$

- Obtener las coordenadas de un punto en el sistema S, conocidas sus coordenadas en el sistema móvil S'

$$a_S = T \cdot a_{S'}$$

- Obtener en que punto **b** se convierte un punto **a** si **<u>se rota y se traslada</u>**.

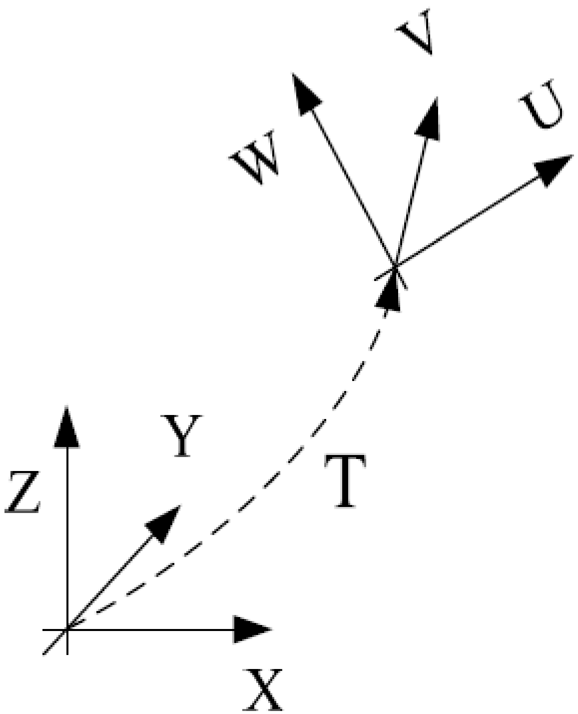
$$b_S = R_{3\times3} \cdot a_S + p_{3\times1} = T \cdot a_S$$

Matrices de transformación homogénea (MTH)
Usos

$$\mathbf{T} = \begin{bmatrix} \text{Rotación} & \text{Traslación} \\ 0 & 1 \end{bmatrix}$$

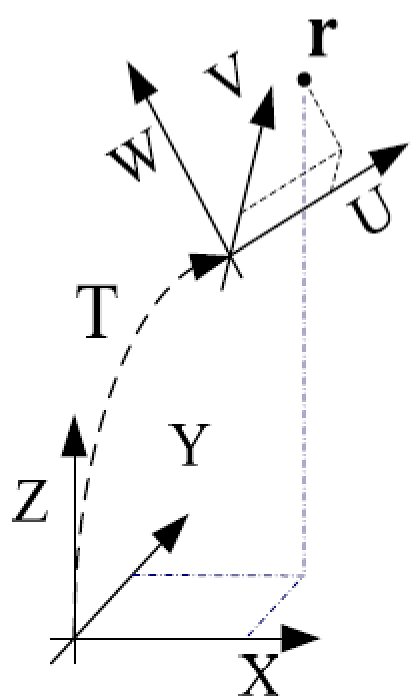
$$\begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \mathbf{T} \begin{bmatrix} r_u \\ r_v \\ r_w \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} r'_x \\ r'_y \\ r'_z \\ 1 \end{bmatrix} = \mathbf{T} \begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix}$$



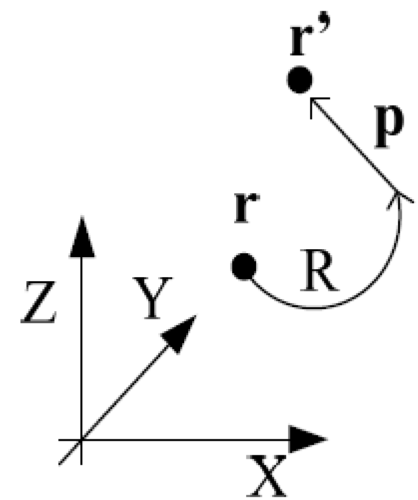
a

Rotación + trasl (Sistema)



b

Cambio de base



c

Rotación + trasl (punto)

Matrices de transformación homogénea (MTH)
Traslación pura

Matriz básica de traslación:

$$\mathbf{T}(\mathbf{p}) = \begin{bmatrix} 1 & 0 & 0 & \boxed{\begin{matrix} p_x \\ p_y \\ p_z \end{matrix}} \\ 0 & 1 & 0 & \\ 0 & 0 & 1 & \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Cambio de sistema de coordenadas:

$$\begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_u \\ r_v \\ r_w \\ 1 \end{bmatrix} = \begin{bmatrix} r_u + p_x \\ r_v + p_y \\ r_w + p_z \\ 1 \end{bmatrix}$$

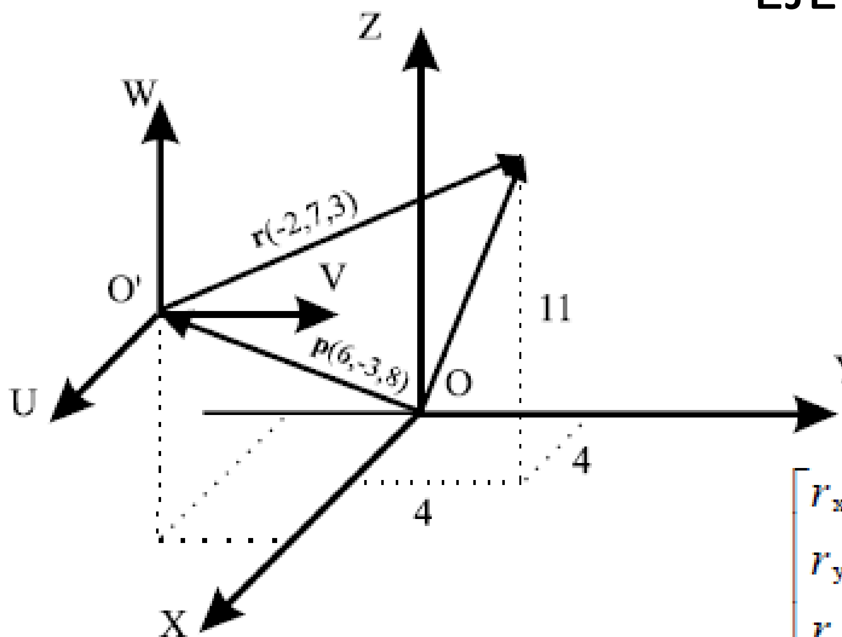
Desplazamiento de un vector:

$$\begin{bmatrix} r'_x \\ r'_y \\ r'_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \begin{bmatrix} r_x + p_x \\ r_y + p_y \\ r_z + p_z \\ 1 \end{bmatrix}$$

Matrices de transformación homogénea (MTH)

Traslación pura

EJEMPLO



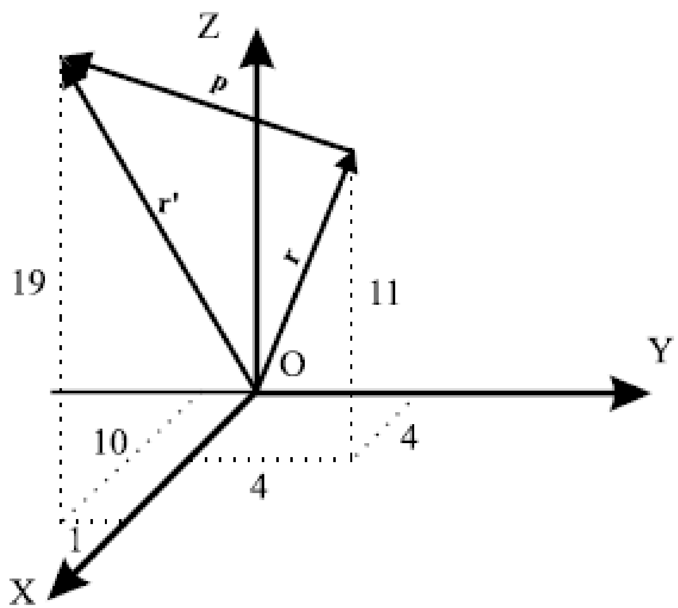
$$\begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 6 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & 1 & 8 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -2 \\ 7 \\ 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \\ 11 \\ 1 \end{bmatrix}$$

¿en qué sistema está? S o S’?

¿Qué operación estamos haciendo?

Matrices de transformación homogénea (MTH)
Traslación pura

EJEMPLO



$$\begin{bmatrix} r'_x \\ r'_y \\ r'_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 6 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & 1 & 8 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 \\ 4 \\ 11 \\ 1 \end{bmatrix} = \begin{bmatrix} 10 \\ 1 \\ 19 \\ 1 \end{bmatrix}$$

¿en qué sistema está? S o S'?

¿Qué operación estamos haciendo?

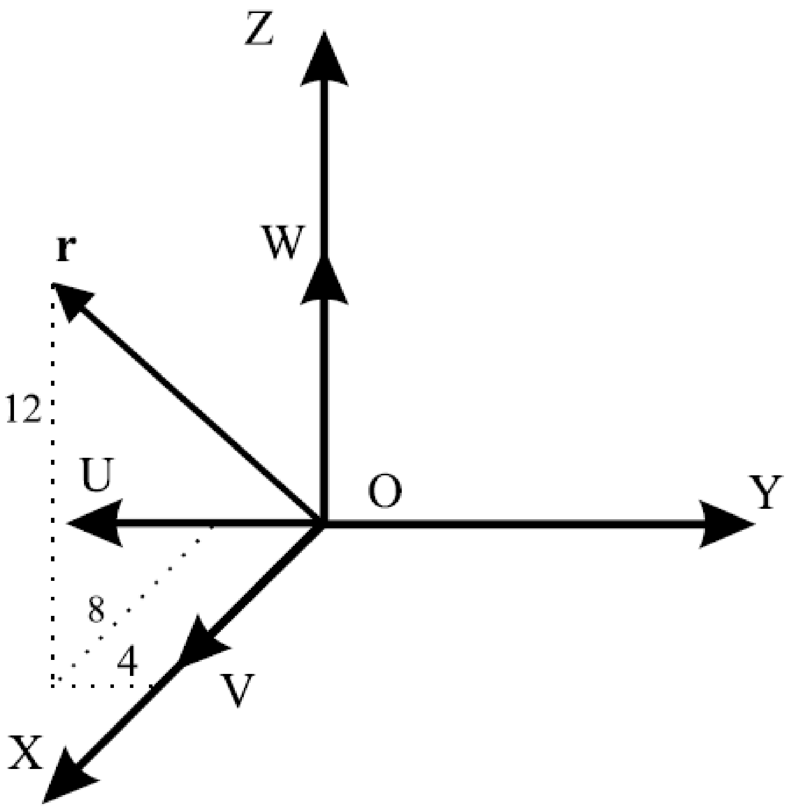
Matrices de transformación homogénea (MTH)
Rotación pura

$$\mathbf{Rotx}(\phi) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\phi & -sen\phi & 0 \\ 0 & sen\phi & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \mathbf{Roty}(\theta) = \begin{bmatrix} \cos\theta & 0 & \operatorname{sen}\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\operatorname{sen}\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{Rotz}(\psi) = \begin{bmatrix} \cos\psi & -\operatorname{sen}\psi & 0 & 0 \\ \operatorname{sen}\psi & \cos\psi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Matrices de transformación homogénea (MTH)

Rotación pura



EJEMPLO

$$\begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 \\ 8 \\ 12 \\ 1 \end{bmatrix} = \begin{bmatrix} 8 \\ -4 \\ 12 \\ 1 \end{bmatrix}$$

¿en qué sistema está? S o S'?

¿Qué operación estamos haciendo?

Composición

- Si queremos hacer varias transformaciones seguidas podemos concatenarlas mediante la multiplicación de matrices.
- **Respecto del sistema fijo** S(0,X,Y,Z)

Orden:

Rotación OX

Rotación OY

Rotación OZ

$$a_S = T \cdot a_{S'} = T_{rz} \cdot T_{ry} \cdot T_{rx} \cdot a_{S'}$$

PREMULTIPLICAR

Composición

- Si queremos hacer varias transformaciones seguidas podemos concatenarlas mediante la multiplicación de matrices.
- **Respecto del sistema móvil** S'(0,U,V,W)

Orden:

Rotación OU

Rotación OV

Rotación OW

$$a_S = T \cdot a_{S'} = T_{rx} \cdot T_{ry} \cdot T_{rz} \cdot a_{S'}$$

POSTMULTIPLICAR

Composición

- Podemos mezclar el sistema móvil y el sistema fijo como vimos con las matrices de rotación

Orden:

Rotación OU

Rotación OY

Rotación OW

$$a_S = T \cdot a_{S'} = T_{ry} \cdot T_{rx} \cdot T_{rz} \cdot a_{S'}$$

Matrices de transformación homogénea (MTH)
Ventajas e inconvenientes

- Su composición se realiza mediante el álgebra de matrices (facilidad de uso) ✓
- Precisan 16 elementos (redundancia) ✗
- Riesgo de inconsistencia tras varias operaciones (redondeos)
  - Adecuadas para la formulación y el cálculo manual ✓
  - Inadecuadas para el calculo computacional ✗
- No podemos interpolar ✗

Bibliografía

- Antonio Barrientos, (2007) Fundamentos de Robótica, 2ª, Mc Graw Hill,
- Aníbal Ollero Baturone, (2001) ROBOTICA Manipuladores y Robots Móviles, Marcombo, 84-267-1313-0,
- K. S. FU., (1988) Robótica: Control, Detección, Visión e Inteligencia, Mc. Graw Hill,