CUATERNIONES

Javier Duque Melguizo
Miguel Ángel Ordoñez Morales

Índice

01

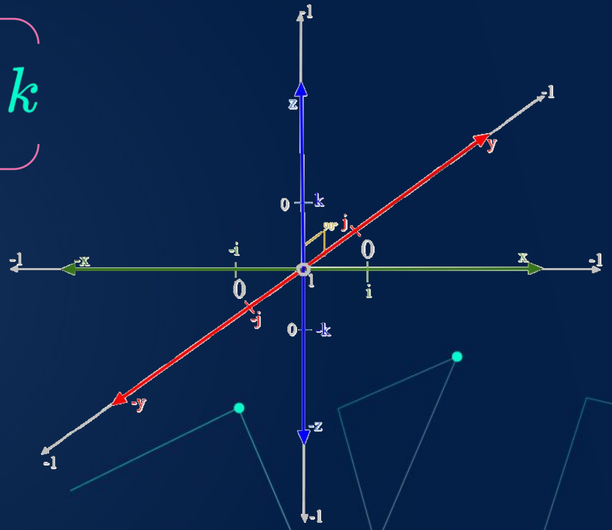
DEFINICIÓN

Extensión números complejos

$$\underbrace{A}_{\text{Parte escalar}} + \underbrace{B*i + C*j + D*k}_{\text{Parte vectorial}}$$

$$\{A, B, C, D\} \epsilon \mathbb{R}$$

$$\{i, j, k\} \epsilon \mathbb{C}$$



Propiedad distributiva

Propiedad asociativa

Propiedad conmutativa

02

ÁLGEBRA

Necesaria para resolver la cinemática directa

Conjugado

$$Q = A + Bi + Cj + Dk$$

$$\overline{Q} = A - Bi - Cj - Dk$$

Inversión del signo de los componentes agregados de la parte vectorial del cuaternión.
Representado por: $\overline{Q}$

Suma

$$Q_1 = A_1 + B_1 i + C_1 j + D_1 k$$

$$Q_2 = A_2 + B_2 i + C_2 j + D_2 k$$

$$Q_1 + Q_2 = (A_1 + A_2) + (B_1 + B_2)i + (C_1 + C_2)j + (D_1 + D_2)k$$

Cada componente del primero se suma únicamente con su equivalente en el segundo.

La suma de cuaterniones se corresponde a un desplazamiento del primero sobre el segundo.

Multiplicación

$$Q_1 * Q_2 =$$
$$(A_1 A_2 - B_1 B_2 - C_1 C_2 - D_1 D_2)$$
$$+(A_1 B_2 + B_1 A_2 + C_1 D_2 - D_1 C_2) * i$$
$$+(A_1 C_2 - B_1 D_2 + C_1 A_2 + D_1 B_2) * j$$
$$+(A_1 D_2 + B_1 C_2 - C_1 B_2 + D_1 A_2) * k$$

La multiplicación entre cuaterniones corresponde a un giro del primero sobre el segundo.

¿Cómo se llega a ese resultado?

Multiplicación
Explicación(I)

1. Desarrollar multiplicación aplicando propiedad distributiva.

$$Q_1 * Q_2 = (A_1 + B_1 i + C_1 j + D_1 k) * (A_2 + B_2 i + C_2 j + D_2 k)$$

$$= A_1(A_2 + B_2 i + C_2 j + D_2 k) + \ldots + (D_1 k)(A_2 + B_2 i + C_2 j + D_2 k)$$

$$= \ldots + (C_1 A_2 + C_1 B_2 i + C_1 C_2 j + C_1 D_2 k) j + (D_1 A_2 + D_1 B_2 i + D_1 C_2 j + D_1 D_2 k) k$$

2. Aplicar a los productos cruzados de las componentes complejas del cuaternión la siguiente tabla de multiplicación.

Las celdas coloreadas marcan la propiedad **NO conmutativa** de los cuaterniones

*	i	j	k
i	-1	k	-j
j	-k	-1	i
k	j	-i	-1

Multiplicación
Explicación(II)

3 Obtendremos el resultado expuesto inicialmente.

$$
\begin{aligned}
Q_1 * Q_2 = \\
&(A_1A_2 - B_1B_2 - C_1C_2 - D_1D_2) \\
&+(A_1B_2 + B_1A_2 + C_1D_2 - D_1C_2) * i \\
&+(A_1C_2 - B_1D_2 + C_1A_2 + D_1B_2) * j \\
&+(A_1D_2 + B_1C_2 - C_1B_2 + D_1A_2) * k
\end{aligned}
$$

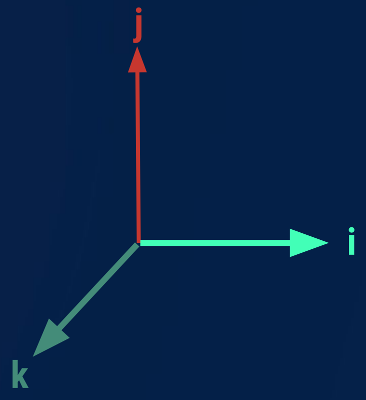
La tabla de multiplicación puede ser replicada de forma intuitiva de la siguiente forma:

1. Sistema de coordenadas orientado a derechas con ejes X,Y,Z correspondiéndole, respectivamente, las componentes i,j,k .

2. Producto entre 2 componentes distintos devolverá el 3º no presente. Otro caso, -1.

$$\boxed{i^2 = j^2 = k^2 = -1}$$

3. El signo lo marcará si la dirección de la 3º componente resultado y el dedo pulgar de la mano derecha, orientada en sentido del 1º desde la izquierda con el resto de dedos apuntando en sentido del 2º, coinciden(+) o no(-) .



03

Cinematica Directa

Variables articulares → Variables cartesianas

Rotación

Rotación de un punto alrededor de un vector θ grados

$$Q = cos\left(\frac{\theta}{2}\right) + sen\left(\frac{\theta}{2}\right) \cdot \vec{n}$$

Coordenadas del punto que se ha rotado

$$O = Q \cdot (\ 0, \vec{r}\ ) \cdot \overline{Q}_i$$

Rotación

Rotación de un punto alrededor de un vector θ grados

$$Q = cos\left(\frac{\theta}{2}\right) + sen\left(\frac{\theta}{2}\right) \cdot \vec{n}$$

¿Por qué se usa θ/2 en vez de θ?

¿Por qué seno y coseno?

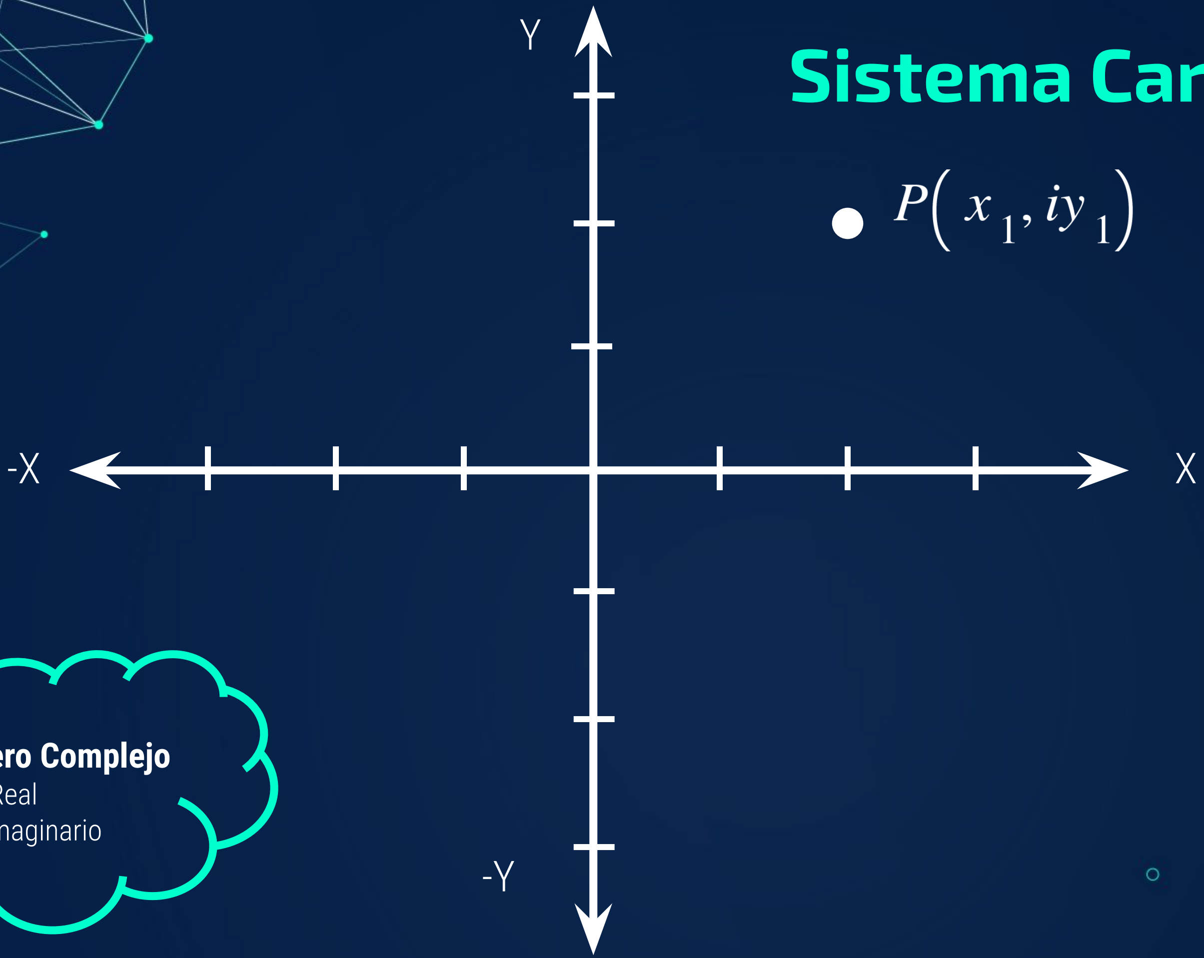
Coordenadas del punto que se ha rotado

$$O = Q \cdot (0, \vec{r}) \cdot \overline{Q}_i$$

¿Por qué se multiplica por una cuaternión y su conjugación?

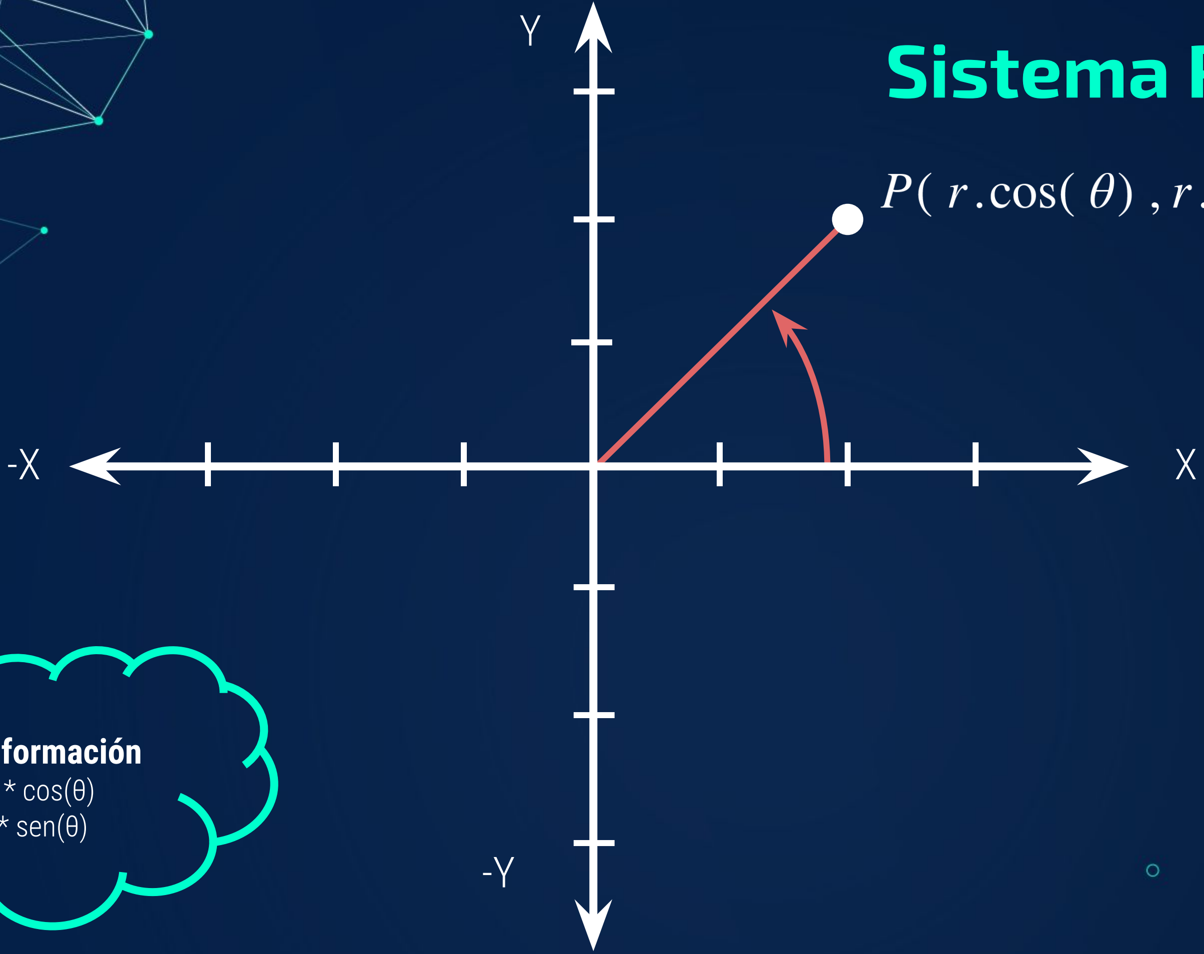
Sistema Cartesiano

$$P\left(x_1, iy_1\right)$$



Numero Complejo
X $\longrightarrow$ Real
Y $\longrightarrow$ Imaginario

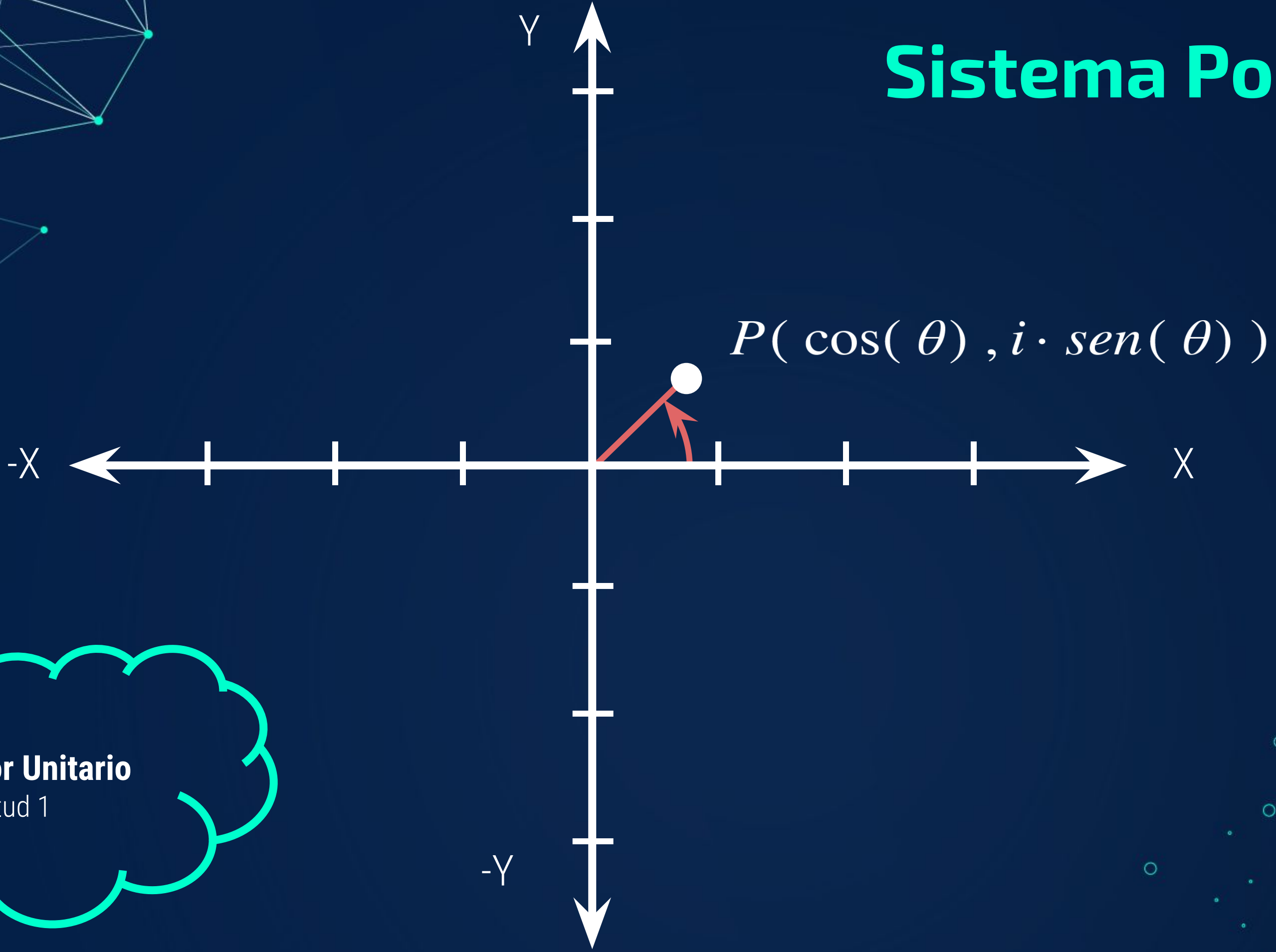
Sistema Polar



$$P(\, r.\cos(\,\theta)\;,r.i\cdot sen(\,\theta)\;)$$

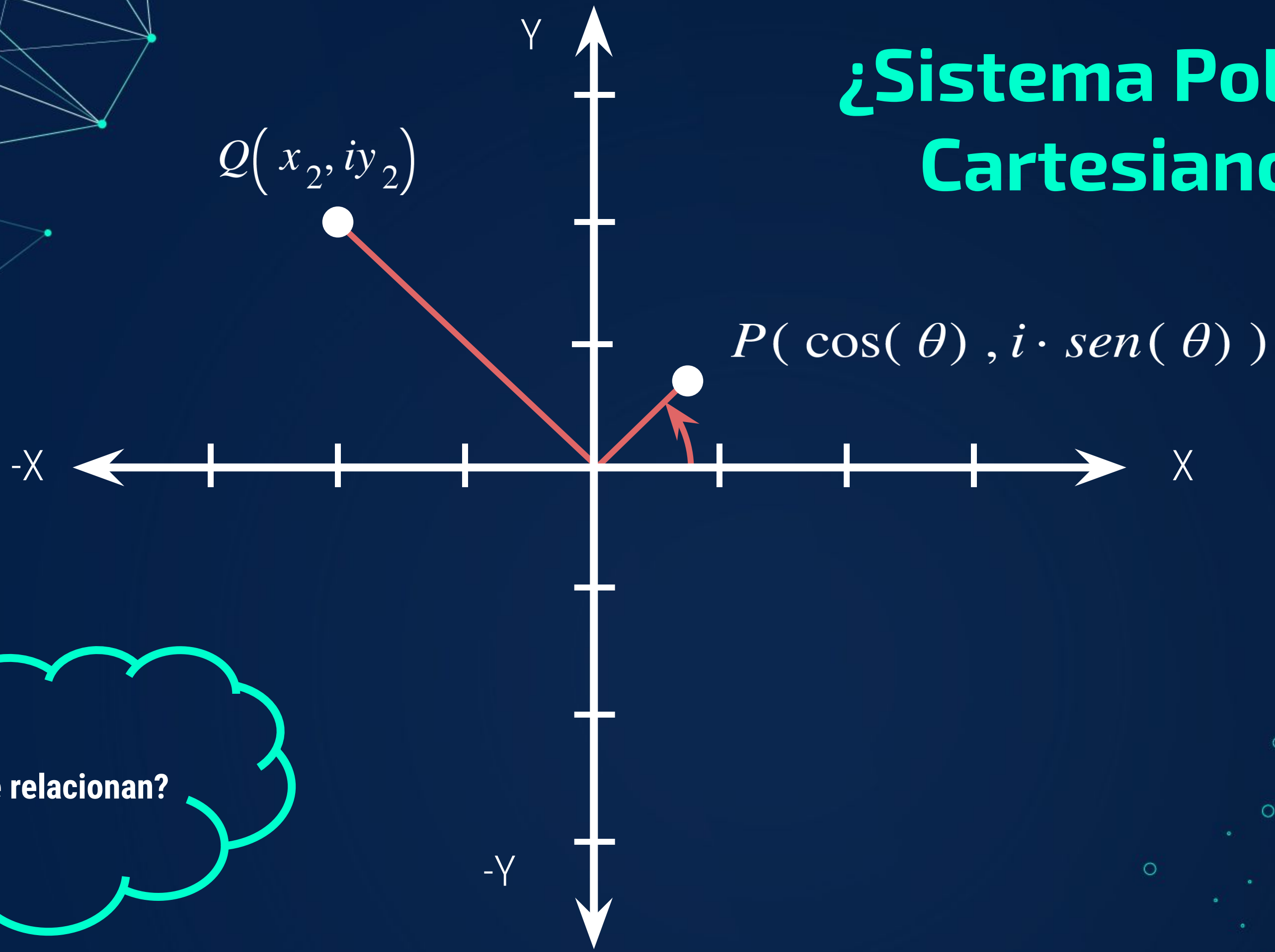
Transformación
X ⟶ r * cos(θ)
Y ⟶ r * sen(θ)

Sistema Polar



Vector Unitario
Magnitud 1

¿Sistema Polar o Cartesiano?



¿Cómo se relacionan?

¿Y si los multiplicamos?

$$P(\cos(\theta), i \cdot sen(\theta)) \quad * \quad Q\left(x_2, iy_2\right)$$

$$(x_2 \cdot \cos(\theta) - y_2 \cdot \sin(\theta)) + i(x_2 \cdot sen(\theta) + y_2 \cdot \cos(\theta))$$

θ = 90º

La multiplicación de un número complejo por otro número complejo **unitario**, es equivalente a rotar en el espacio 2D

$$\underbrace{(i)}_{\text{Rotación}}\underbrace{(x_2 + iy_2)}_{\text{Punto 2D}} = -y_2 + ix_2$$

Multiplica por la izquierda

$$S = (w + ix + jy + kz)$$

$$(i)(\underline{w + ix} + \underline{jy + kz}) = \underline{-x + iw} \underline{- jz + ky}$$

(w, x) se convierten en (-x, w)

90º

(y, z) se convierten en (-z, y)

90º

Multiplicar un cuaternión por un ángulo, el cuaternión gira en dos planos independientes a la vez

Multiplica por la derecha

$$S = (w + ix + jy + kz)$$

$$(i)(w + ix + jy + kz) = -x + iw - jz + ky$$

$$(\underline{w + ix} + \underline{jy + kz})(i) = \underline{-x + iw} + \underline{jz - ky}$$

(w, x) se convierten en (-x, w)

90º

(y, z) se convierten en (z, -y)

90º

El **problema** es que las rotaciones ocurren en dos planos y dependen del lado por el que se multiplica

Multiplica por la izquierda y derecha

$$S = (w + ix + jy + kz)$$

$$(i)(\underline{w + ix} + jy + kz)(i) = \underline{-w - ix} + jy + kz$$

$\boxed{\text{(w, x) se convierten en (-x, -w)}}$

180º

Modifica plano con componente real

$$(i)(w + ix + jy + kz)(\bar{i})$$

$$(i)(w + ix + \underline{jy + kz})(-i) = w + ix \underline{- jy - kz}$$

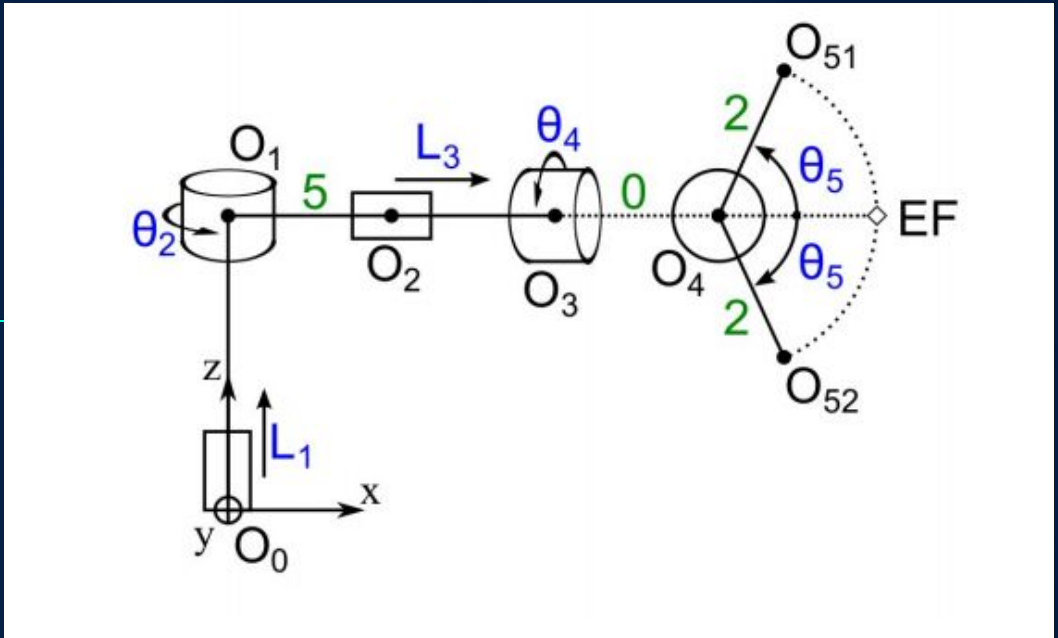
Cinemática directa

Objetivo

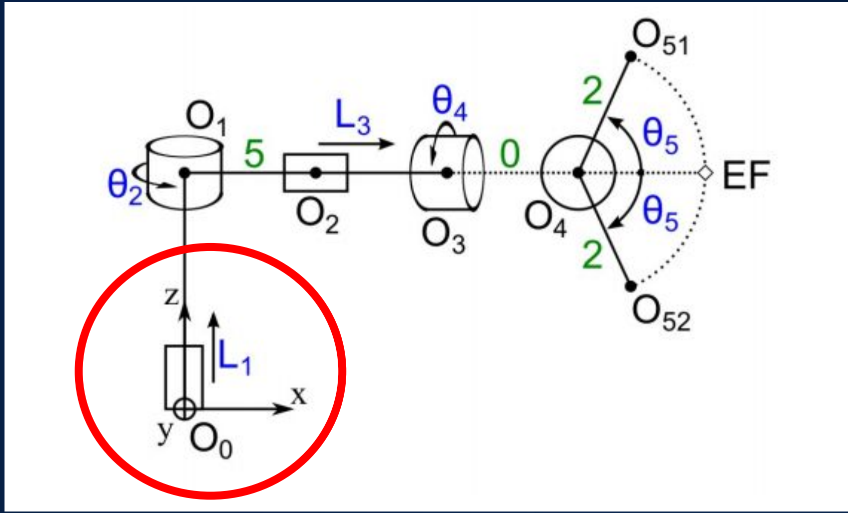
Articulares $\xrightarrow{\text{Matrices / Cuaterniones}}$ **Cartesianas**

Operaciones
- Conjugado
- Suma
- Multiplicación

Brazo robótico



5 articulaciones, tanto de rotación como de desplazamiento

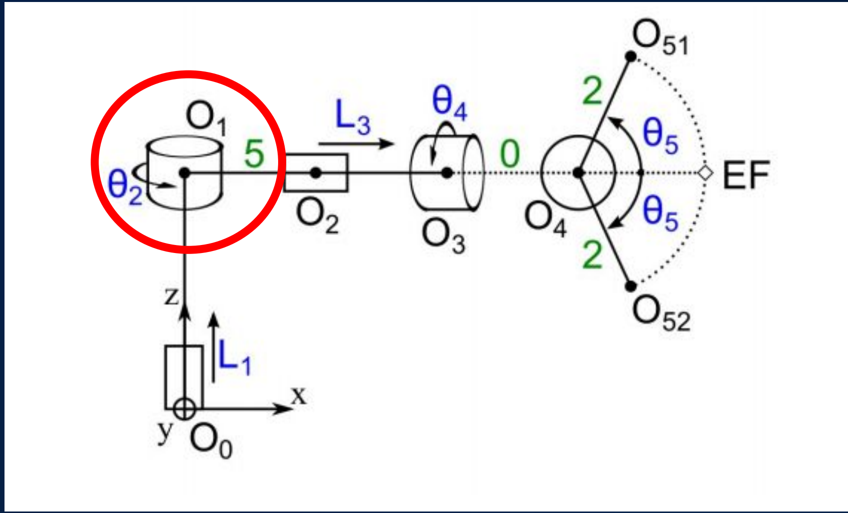


O_0

Cuaterniones **puros** permiten representar en 3D

$$O_0 = (\,0,\ \underbrace{0}_{X},\ \underbrace{0}_{Y},\ \underbrace{0}_{Z})$$

O_1



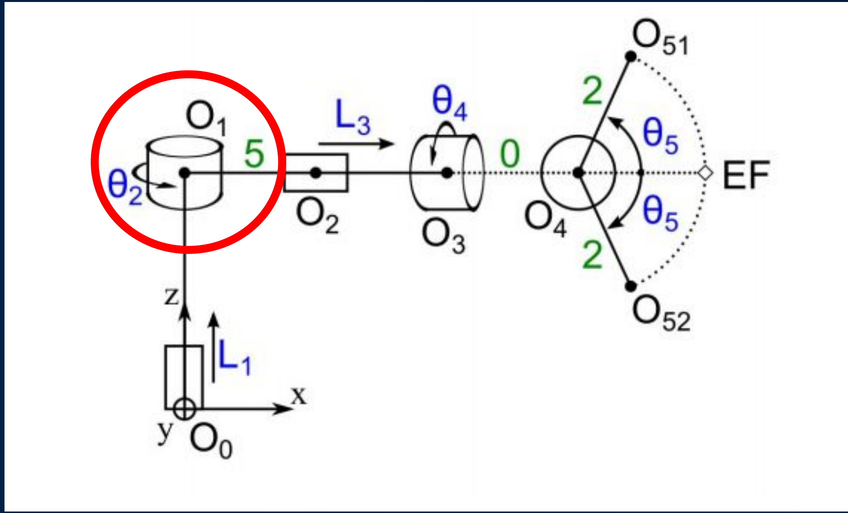
Convenio

$$\underbrace{\vec{n}_1 = (0, 0, 1) \qquad \theta = 0°}$$

$$Q = cos\left(\frac{\theta}{2}\right) + sen\left(\frac{\theta}{2}\right) \cdot \vec{n} \longrightarrow Q_1 = (1, 0, 0, 0)$$

Articulación de desplazamiento

O_1

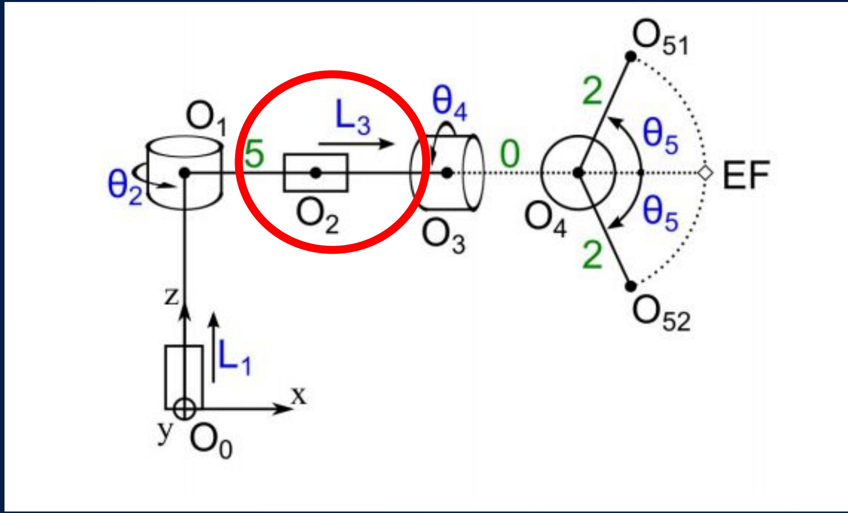


$l_1 = 10$

$$Q_1 = (1, 0, 0, 0) \quad r_1 = (0, 0, 0, l_1) \quad \overline{Q} = (1, 0, 0, 0)$$

$$O = Q \cdot \vec{r} \cdot \overline{Q_i} \longrightarrow O_1 = (0, 0, 0, 10)$$

Articulación de desplazamiento



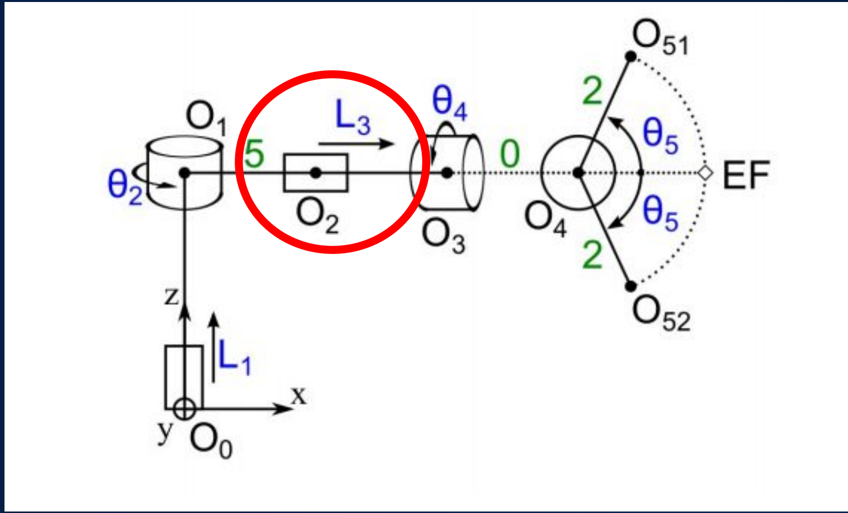
O_2

$$\vec{n}_2 = (0, 0, 1) \qquad \theta = \theta_2$$

$\theta_2 = 180^{\circ}$

$$Q = \cos\left(\frac{\theta}{2}\right) + sen\left(\frac{\theta}{2}\right) \cdot \vec{n} \longrightarrow Q_2 = (0, 0, 0, 1)$$

Articulación de rotación



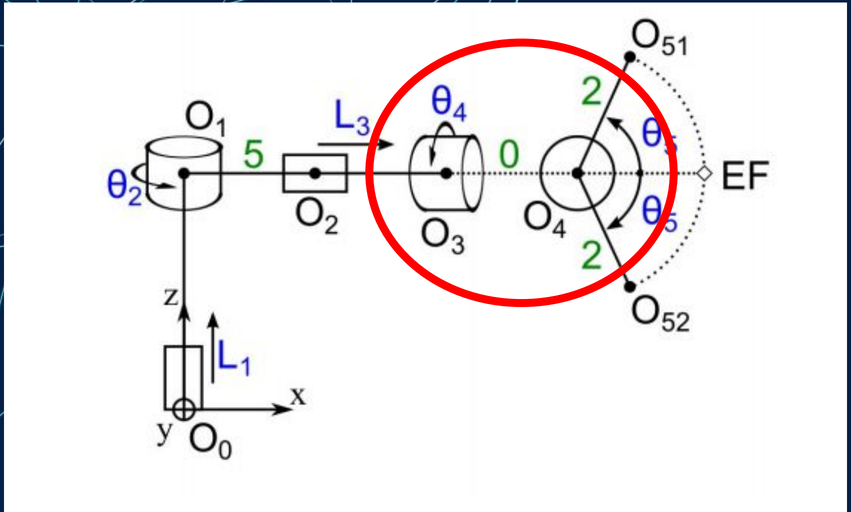
O_2

$$Q_2 = (0, 0, 0, 1) \qquad r_2 = (0,\ 5,\ 0,\ 0) \qquad \overline{Q_2} = (0, 0, 0, -1)$$

$$O_n = \prod_{i=1}^{n} Q_i \cdot (0, \vec{r}) \cdot \prod_{i=n}^{1} \overline{Q_i} + O_{n-1} \longrightarrow O_2 = (0, -5, 0, 10)$$

Articulación de rotación

O_3 y O_4



$$r_3 = (0,\ l_3,\ 0,\ 0)$$

$$r_4 = (0,\ 0,\ 0,\ 0)$$

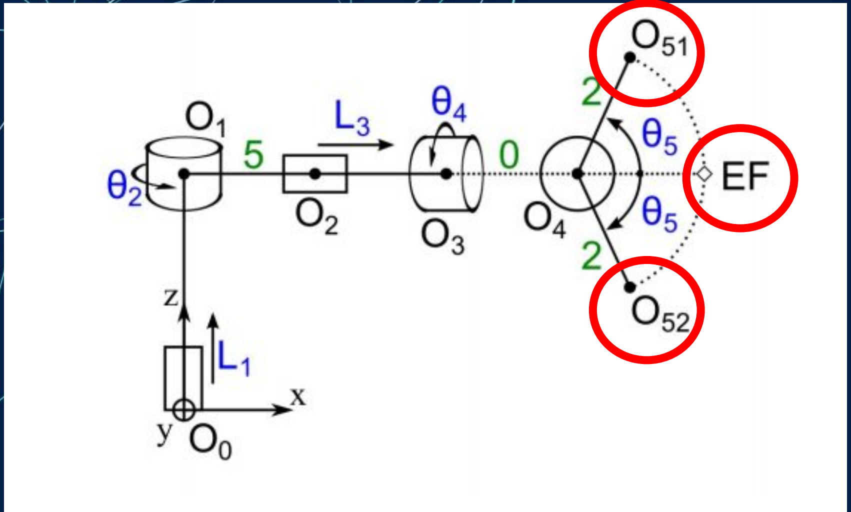
$l_3 = 5$ $\theta_4 = 0°$

$$\vec{n}_3 = (1, 0, 0)$$

$$\vec{n}_4 = (1, 0, 0)$$

$$O_3 = (0,\ -10,\ 0,\ 10)$$

$$O_4 = (0,\ -10,\ 0,\ 10)$$



O_{51} , O_{52} y O_{EF}

$$\overrightarrow{n}_{51} = (0, -1, 0)$$

$$\overrightarrow{n}_{52} = (0, 1, 0)$$

$$r_5 = (0,\ 2,\ 0,\ 0)$$

$$\overrightarrow{n}_{EF} = (0, 1, 0)$$

04

Conclusión

Comparativa y conclusiones.

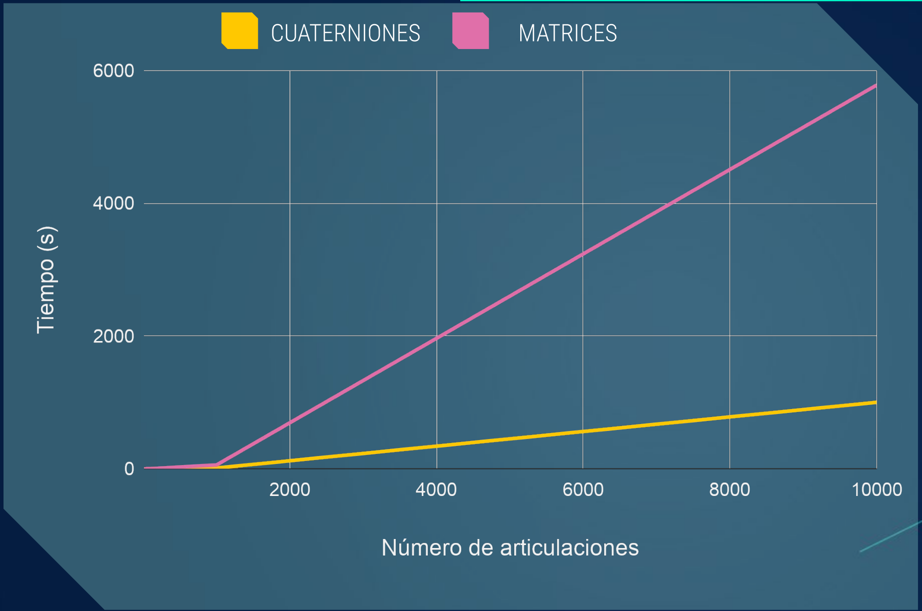
Conclusiones

1. Dada la naturaleza 4D de los cuaterniones, estos son más difíciles de comprender y justificar matemáticamente que las matrices de transformación.
2. Sin embargo, su aplicación para resolver la cinemática directa es mucho más intuitiva y fácil de explicar que la aplicación planteada con matrices, incluso el aproximamiento realizado por Denavit-Hartenberg. Por lo cual, desarrollar un algoritmo que resuelva la cinemática directa utilizando cuaterniones, también lo es.
3. El algoritmo que resuelve la cinemática directa con cuaterniones y el que lo resuelve con matrices, ambos implementan , respectivamente, los siguientes productorios:

$$\prod_1^n (Q_i) * (0, \overrightarrow{r}_n) * \prod_n^1 (\overline{Q}_i) \qquad \prod_n^1 ({}^{i-1}T_i) * (O_n)_n$$

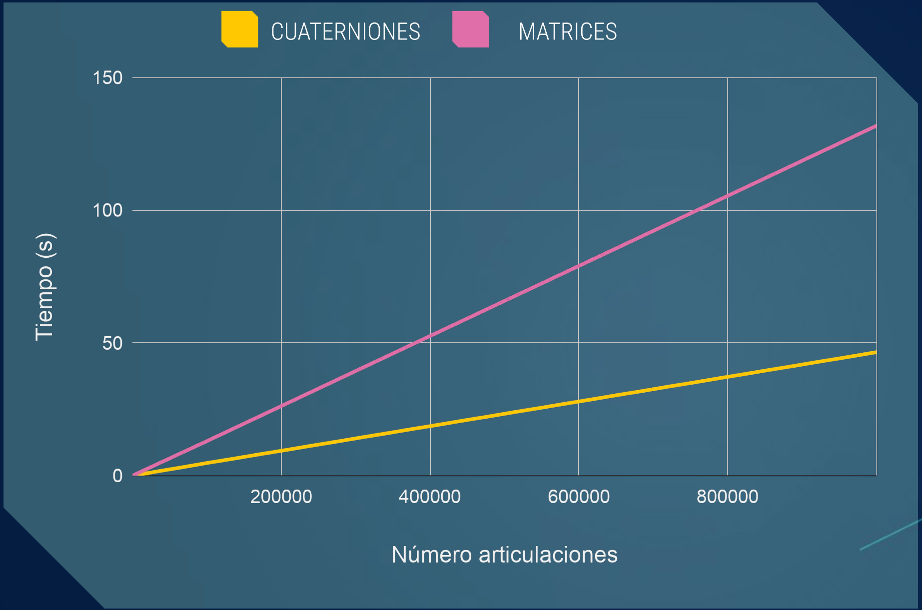
Por lo tanto, ambos tienen complejidad **O(n^2)**, sin embargo, el número de multiplicaciones que realizan difiere, y ello se ve reflejado en los tiempos de ejecución de los 2 algoritmos para un mismo **n** .

Comparativa
(Sin optimización)



$$\underbrace{32n}_{\text{Cuaterniones}} \text{ vs } \underbrace{64 * (n - 1)}_{\text{Matrices}}$$

Comparativa
(Optimizado)



Aprovechando la propiedad asociativa, de forma que resultados previos de ambos productorios son guardados en memoria,reducimos complejidad del algoritmo a **O(n)** .

CONCLUSIÓN

La resolución de la cinemática directa con cuaterniones no es solo más fácil e intuitiva que su contraparte, además, el algoritmo que se implementa con ellos es entre 3 y 5 veces más rápido.

Gracias

Recursos alojados en:
https://github.com/JDM-ULL-93/CD_Cuaternios

Video

¿Alguna pregunta?

Javier Duque Melguizo:
alu0101160337@ull.edu.es

Miguel Ordoñez Morales:
alu0101281087@ull.edu.es