



ROBÓTICA (86.15)

UNIVERSIDAD DE BUENOS AIRES
FACULTAD DE INGENIERÍA
Año 2019 - 2^{do} cuatrimestre

TRABAJO PRÁCTICO N.º 2
TEMA: Problema de posición directo en inverso
FECHA: 13 de octubre de 2019

INTEGRANTE :

Nastasi, Franco Gabriel
fnastasi@fi.uba.ar

- #100002

1. Introducción

Para resolver el problema de obtener la posición y orientación final de un robot IBR 140 de la marca ABB, se aplica el criterio de Denavit y Hartenberg. Para ello se numeran los eslabones y se colocan las ternas como se observa en la siguiente imagen teniendo en cuenta que se desea maximizar la cantidad de valores nulos en las matrices homogéneas.

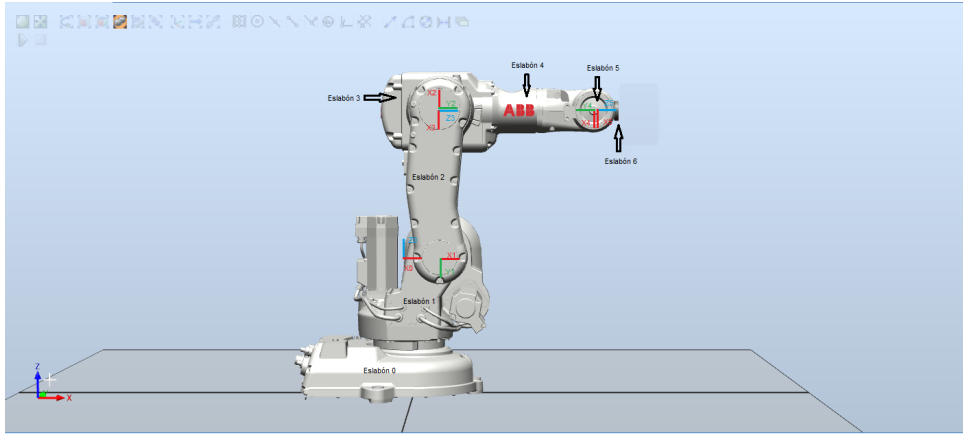


Figura 1.1: Colocación de las ternas según el criterio D-H maximizando la cantidad de valores nulos.

Si bien en la imagen puede parecer lo contrario, el origen de las ternas 2 y 3 es el mismo, así como también el punto de origen de las ternas 4,5 y 6 es el mismo.

Para este robot las variables de articulación son los valores θ_i de los arámetros D-H. Para la figura 1.1 estos valores resultan ser:

$$\vec{\theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \end{bmatrix} = \begin{bmatrix} 0 \\ -90 \\ 180 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (1.1)$$

2. Desarrollo

En esta sección se comentan algunas observaciones sobre el código presentado y las pruebas realizadas. En primer lugar se modificó ligeramente la versión del tp1. En vez de verificar que a_x y a_y sean distintos de cero, se pregunta por la condición en la que son mayores a una tolerancia pequeña igual a 10^{-6} . Esto permitió corregir errores numéricos de aproximación que se producen por aproximar un número irracional como es π por una cantidad finita de dígitos. Con respecto al código que corresponde al segundo trabajo práctico se escribió una función auxiliar *DH_hom_mat* que calcula la matriz homogénea que se obtiene al plantear el criterio Denavit Hartenberg. Al utilizar esta función nuevamente aparecían errores numéricos por lo que se decidió reemplazar por cero los elementos de la matriz que sean menores en módulo al valor de tolerancia mencionada.

Para probar el correcto funcionamiento de las funciones, se utilizaron diferentes variables articulares con distintos índices de configuración. Por ejemplo se probaron los algoritmos con los valores que se obtienen al analizar la figura 1.1:

$$\vec{\theta}_1 = \begin{bmatrix} 0 \\ -90 \\ 180 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (2.1)$$

También se aplicó el algoritmo para las siguientes variables articulares

$$\vec{\theta}_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 90 \\ 0 \end{bmatrix} \quad (2.2)$$

Estos valores corresponden a la configuración del robot que se muestra en la siguientes imágenes:

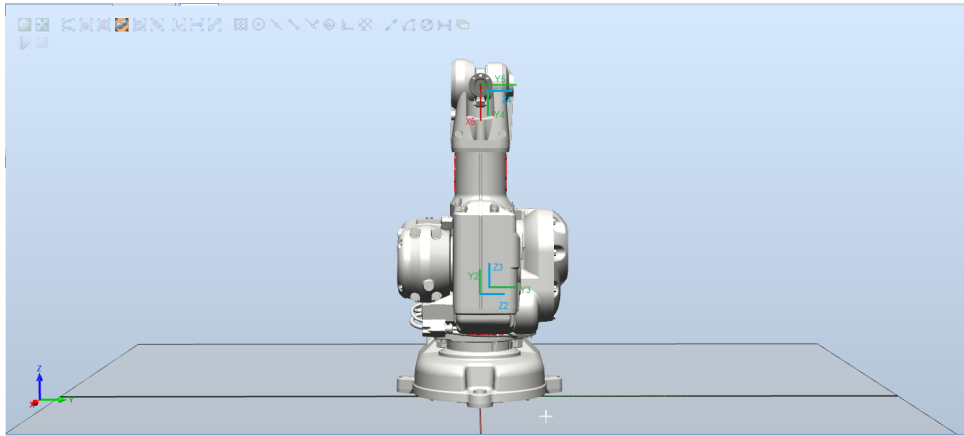


Figura 2.1: Vista de la configuración del robot para las variables articulares $\vec{\theta}_2$ donde se ven las ternas 0 y 1

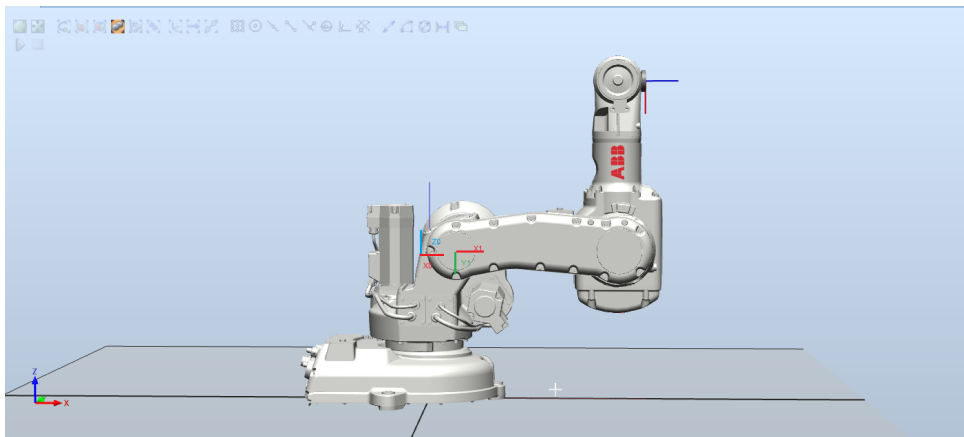


Figura 2.2: Vista de la configuración del robot para las variables articulares $\vec{\theta}_2$ donde se ven las restantes ternas. El origen de la terna 2 y 3 es el mismo y se tomo la terna 6 igual a la terna 5

Donde la matriz homogénea resultante del algoritmo es:

$$A_0^6 = \begin{bmatrix} 0 & 0 & 1 & 430 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 380 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.3)$$

Se aplicó el algoritmo desarrollado para los siguientes variables de articulación, obteniéndose la matriz homogénea que se indica.

$$\vec{\theta}_3 = \begin{bmatrix} 30 \\ 30 \\ 30 \\ 30 \\ 30 \\ 30 \end{bmatrix} \quad (2.4)$$

$$A_0^6 = \begin{bmatrix} -0,556 & -0,429 & 0,712 & 616 \\ 0,612 & 0,368 & 0,67 & 355 \\ -0,562 & 0,824 & 0,058 & 10 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

Para estos valores de variables de articulación se observó que si se aplicaba el algoritmo de problema inverso cambiando el valor de g_1 , el cual resultó ser 1, por -1, se obtenía el error correspondiente a:

$$|\sin \theta_3| > 1$$

Esto puede deberse a que como el índice de configuración g_1 indica si el brazo pasa por delante o por atrás del eje 1 para alcanzar una posición y orientación determinada, al encontrarse en una zona límite que el brazo robótico puede alcanzar, tal configuración solo admite únicamente un solo valor de índice de configuración g_1

Por último se aplicó el algoritmo que resuelve el problema de posición directa a un conjunto de variables articulares y se aplicó el problema inverso. En general se produjeron errores que tenían que ver con la capacidad de representar números racionales o de restar números iguales dando como resultado un valor distinto de cero. La mayoría de estos errores se lograron solucionar aunque si se ejecuta el script *pruebas.py* se obtiene que hay un error ya que el script compara los valores de ángulos 180° y -180° , devolviendo que son ángulos diferentes. A continuación se muestran algunos de los resultados obtenidos, mostrando el caso mencionado. Se menciona además que no se tomó en cuenta las limitaciones físicas del robot para proponer los valores de las variables articulares. Esto fue así para probar los algoritmos incluso en los valores límites donde puede llegar a observar algún error, de existir, en el algoritmo.

θ	Resultado problema directo (A — g)	Resultado problema inverso
$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 & 430 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 380 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$
$\begin{bmatrix} 0 \\ 45 \\ 90 \\ 0 \\ 0 \\ 135 \end{bmatrix}$	$\begin{bmatrix} 0,5 & 0,5 & 0,707 & 593 \\ 0,707 & -0,707 & 0 & 0 \\ 0,5 & 0,5 & -0,707 & -593 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 45 \\ 89,99999 \\ 0 \\ 0 \\ 135 \end{bmatrix}$
$\begin{bmatrix} 15 \\ -180 \\ 0 \\ -135 \\ 0 \\ -45 \end{bmatrix}$	$\begin{bmatrix} 0,96 & 0,26 & 0 & -280 \\ 0,26 & -0,96 & 0 & -75 \\ 0 & 0 & -1 & -380 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 15 \\ 180 \\ 0 \\ -135 \\ 0 \\ -45 \end{bmatrix}$
$\begin{bmatrix} 0 \\ -45 \\ 45 \\ 135 \\ 180 \\ -180 \end{bmatrix}$	$\begin{bmatrix} -0,707 & 0,707 & 0 & 324 \\ 0,707 & 0,707 & 0 & 0 \\ 0 & 0 & -1 & 634 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -45 \\ 45 \\ 135 \\ 180 \\ 180 \end{bmatrix}$

Tabla 2.1: Resultados de pruebas realizadas sobre los algoritmos.

Más pruebas pueden realizarse al ejecutar el archivo *pruebas.py* modificando el archivo *variables_articulares_prueba.txt* donde se encuentran los valores de variables articulares que se desean probar

3. Conclusiones

Se logró implementar 2 algoritmos en lenguaje python que resuelven el problema de posición directa en inversa de un robot IBR 140 de la marca ABB. Se realizaron pruebas con distintas variables articulares obteniéndose resultados positivos una vez que se resolvieron problemas relacionados a la capacidad del ordenador de representar números irracionales.