Laboratorio 1 – Análisis de las Características de un Robot Industrial y Modelo Geométrico Directo

Parte II

5. Iniciando

5.3 Compare los dos métodos

La realización del modelo del robot KUKA KR 340 R3330 para ambos *toolboxes* contó con procedimientos similares, en ambos, se crean objetos que representan la geometría de los eslabones, para esto, una lectura de la ficha técnica del manipulador es necesaria para tomar de allí sus dimensiones. Mientras la herramienta propia de MATLAB, el *Robotics System Toolbox* (de ahora en adelante RST), necesita de varias líneas de código para la creación de el eslabón, pues requiere de la información de su transformación respecto a una base fija, la asociación a un elemento de tipo articulación e incluso un nombre particular para poder ser asociado luego con los demás eslabones de la cadena; el *Robotics Toolbox* de Peter Corke para MATLAB (de ahora en adelante RVC) emplea una sintaxis más simplificada por medio de la creación de elementos tipo *Link* que terminan recibiendo toda la información necesaria para definir un eslabón, y además, su articulación asociada. Los elementos creados, los eslabones y articulaciones como cuerpos rígidos para el RST y los *Links* para el RVC, son reunidos en una sola cadena cinemática a través de métodos distintos, el primero requiere del comando *addBody()* que asocia cada eslabón (hijo) con su igual inmediatamente anterior (padre), dando la posibilidad incluso de crear diferentes ramas en un mismo manipulador; dicha asociación en el RVC se realiza con la función *SerialLink* que simplifica el tratamiento al recibir un simple arreglo con los eslabones de manera independiente. A pesar de sus diferencias, algo es común en ambos, es fundamental para describir apropiadamente el manipulador deseado su análisis de los parámetros de Denavit-Hartemberg modificados (D&H mod), sin estos, se hace mucho más complicado para el RST y casi imposible para el RVC, el cual solicita explícitamente estos parámetros para poder definir correctamente al manipulador.

6. Modelo Geométrico Directo

6.5 Compare los métodos

Para realizar los métodos de la cinemática directa para encontrar la pose del efector final de acuerdo a unos ciertos valores de articulación también es diferente para ambos *toolboxes*, para el RST, la posibilidad de realizar una imagen con la configuración del robot para una configuración dada se torna poco intuitiva, usando la función del RST *show()* se puede agregar como argumento el robot que se desea visualizar y su configuración, además de otras opciones; precisamente, esta configuración es una estructura que contiene los nombres de la articulación y los valores asociados, de manera que no puede ser configurado sino tan solo entrando en una de las características de la estructura, por medio de la sintaxis *config(i).JointPosition*, algo que no se espera cuando se quiere realizar una modificación rápida a la posición del manipulador. Por otra parte, el RVC simplifica mucho eso al simplemente recibir un arreglo con los valores de articulación *q* como argumento de la función *robot.teach()*, es este caso, no solo que muestra visualización del robot en la posición requerida, sino que también despliega unos controles con la posibilidad de modificar en tiempo real estos valores y así observar el robot en movimiento articular.

Adicionalmente, cada *toolbox* tiene la función propia para la obtención de la matriz de transformación homogénea a partir de una configuración entregada, para el caso del RST, la función es *getTransform()* que solicita el robot, la configuración (estructura con nombres y valores de las articulaciones) y el nombre del elemento donde se quiere analizar dicha matriz, usualmente, el efector final; por el lado del RVC, la función es *robot.fkine()* cuyo único argumento es el arreglo de los valores articulares *q*.