

Jornadas de Automática

Generación automática de gemelo-sombra digital de CDPR en CoppeliaSim

Riveiro, Enrique^{a,*}, Silva-Muñiz, Diego^a, Sáez, Juan^a, Garrido, Julio^a

^aGrupo en Ingeniería Eficiente y Digital (EN.EDI), Dpto. de Ingeniería de Sistemas y Automática, Universidade de Vigo, C/Maxwell, nº 9, 36310, Vigo, España

To cite this article: Riveiro, E., Silva-Muñiz, D., Sáez, J., Garrido, J. 2025. Automatic generation of CDPR digital shadow-twin in CoppeliaSim. Jornadas de Automática, 46. <https://doi.org/10.17979/ja-cea.2025.46.12182>

Resumen

La construcción de la sombra digital en aplicaciones de simulación para un robot paralelo actuado por cables (CDPR) es laboriosa, debiendo realizar los cálculos de posición inicial de los puntos de anclaje de los cables de forma manual para los distintos parámetros dimensionales. Este artículo presenta una solución para generar de forma automática gemelos-sombras digitales de CDPR en aras de agilizar el desarrollo de nuevas configuraciones y reducir errores por intervención manual. El *script* empleado para la generación extrae los parámetros del controlador y construye, sin intervención manual, un modelo virtual que replica la configuración del CDPR. Así, se obtiene un entorno seguro y rápidamente configurable donde probar trayectorias y ajustar estrategias sin necesidad de disponer de robot físico. Posteriormente, se realiza un experimento de seguimiento de una trayectoria. Los resultados mostraron un error cuadrático medio de 0.866mm a 100mm/s y de 1.841mm a 400mm/s, evidenciando la alta fidelidad y repetibilidad de la sombra digital autogenerada.

Palabras clave: Controladores lógicos programables, Implementación digital, Robots manipuladores, Sistemas de control de movimiento, Tecnología robótica

Automatic generation of CDPR digital shadow-twin in CoppeliaSim

Abstract

Generating digital shadows in simulation applications for a cable-driven parallel robot (CDPR) is a time-consuming task, requiring manual calculus of the initial position of the cable anchor points for the different dimensional parameters. This article presents a solution for automatically generating digital shadow twins of CDPRs in order to streamline the development of new configurations and reduce errors caused by manual intervention. The script used for generation extracts the parameters from the controller and constructs, without manual intervention, a virtual model that replicates the CDPR configuration. This provides a safe and quickly configurable environment in which to test trajectories and adjust strategies without the need for a physical robot. Subsequently, a trajectory tracking experiment is performed. The results showed a mean square error of 0.866mm at 100mm/s and 1.841mm at 400mm/s, demonstrating the high fidelity and repeatability of the self-generated digital shadow.

Keywords: Digital implementation, Motion Control Systems, Programmable logic controllers, Robots manipulators, Robotics technology

1. Introducción

Los robots paralelos actuados por cables (CDPR por sus siglas del inglés *Cable-Driven Parallel Robot*) utilizan cables para posicionar y orientar una plataforma móvil. El uso de cables le confiere características como baja inercia, gran

relación carga-peso y amplios volúmenes de trabajo. Desde los pioneros SkyCam y RoboCrane hasta desarrollos recientes como IPANEMA, los CDPR han demostrado su versatilidad en logística, fabricación aditiva y simuladores (Zhang et al., 2022). No obstante, su uso industrial sigue siendo limitado

*Autor para correspondencia: enrriveiro@uvigo.gal
Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

porque los fabricantes de robots industriales no comercializan este tipo de robots, los controladores comerciales raramente soportan configuraciones cableadas “listas para usar” y los requisitos de seguridad dificultan la puesta en marcha (Zhang et al., 2022).

Cada variante de CDPR exige ajustar geometría, longitudes y límites de cable sin modificar la lógica de control. Probar esos cambios de forma segura y rápida, antes de accionar motores reales, requiere un entorno flexible que reproduzca la cinemática en tiempo real y permita verificar trayectorias. Las librerías académicas WireX, CASPR o CDPR-Studio facilitan el análisis off-line (simulación) de CDPRs (McDonald et al., 2022), pero no ejecutan el mismo código que residirá en el controlador industrial real.

La principal contribución de este trabajo es un procedimiento autogenerado que, sin intervención manual, crea gemelos-sombras digitales (en adelante, sombras digitales) de CDPRs a partir de los parámetros que caracterizan al robot y que están disponibles en el controlador industrial frente a métodos más tradicionales en los que la caracterización del modelo a autogenerar se encuentra definido en ficheros estructurados (XML, JSON, etc.). La representación digital genera el rol de sombra digital en el sentido de (Kritzinger et al., 2018): se autoconstruye con los datos calculados en el controlador y recibe de él todas las variables de estado y consignas, sin devolver órdenes. Para ello, se parte del modelo cinemático del CDPR, descrito en la sección 2 y que implementa directamente en el controlador.

El flujo de trabajo (*pipeline*), presentado en la sección 3, conecta el controlador industrial con CoppeliaSim, permitiendo testear cualquier configuración de cables mediante la reutilización íntegra del programa de control que se ejecutaría en el robot real. La sección 4 detalla el mecanismo de generación automática de la sombra digital, desde la obtención de los parámetros del sistema hasta la construcción del entorno simulado. La fidelidad de las trayectorias se cuantifica mediante el error RMS entre los datos obtenidos de la cinemática directa del CDPR real y los de la sombra digital (sección 5). Finalmente, la sección 6 resume las conclusiones y líneas futuras.

2. Modelo cinemático

Para obtener el modelo cinemático del CDPR, se considera que los cables son sin masa y no elásticos. El lazo vectorial cerrado mostrado en la Figura 1 para el cable i –ésimo se utiliza para definir las ecuaciones cinemáticas del CDPR completamente restringido por ocho cables de longitud l_i , siendo A_i los puntos de anclaje proximales y B_i los puntos de anclaje distales para cada cable, O representa el sistema de referencia base y P es el punto de referencia del TCP (*Tool Center Point*) en el efector final. La ecuación (1) representa el resultado de la cinemática inversa:

$$l_i = \|\vec{L}_i\| = \|\vec{a}_i - R(\vec{b}_i) - \vec{r}\|, \quad i = 1, 2, \dots, 6 \quad (1)$$

donde, \vec{r} representa el vector de posición objetivo desde el sistema de coordenadas de la plataforma móvil al sistema de coordenadas base, \vec{b}_i y \vec{a}_i son los vectores que conectan los puntos distal y proximal de cada cable con el TCP en el efec-

tor final y la base respectivamente, y R es la matriz de rotación *Roll-Pitch-Yaw*.

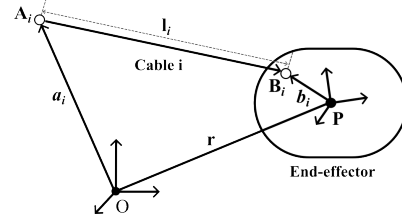


Figura 1: Ilustración esquemática del modelo cinemático de bucle cerrado de un cable del CDPR.

Siendo $\vec{l}_i = (l_{ix}, l_{iy}, l_{iz})^T$ el vector del cable y $\vec{l}_{iXY} = (l_{ix}, l_{iy}, 0)$ su proyección sobre el plano horizontal. El ángulo de elevación del cable con el punto A_i se determina aplicando el teorema del coseno entre \vec{l}_i y \vec{l}_{iXY} ,

$$\theta_{iz} = \arccos\left(\frac{\vec{l}_i \cdot \vec{l}_{iXY}}{\|\vec{l}_i\| \|\vec{l}_{iXY}\|}\right), \quad (2)$$

que cuantifica la inclinación real del cable respecto al plano XY. Para describir la orientación azimutal de esa proyección se emplea \vec{l}_{iXY} , con la que se forman los ángulos de la proyección con los ejes cardinales $\vec{e}_X = (1, 0, 0)$ y $\vec{e}_Y = (0, 1, 0)$,

$$\theta_{iX} = \arccos\left(\frac{\vec{e}_X \cdot \vec{l}_{iXY}}{\|\vec{l}_{iXY}\|}\right), \quad \theta_{iY} = \arccos\left(\frac{\vec{e}_Y \cdot \vec{l}_{iXY}}{\|\vec{l}_{iXY}\|}\right). \quad (3)$$

Así, cada cable queda caracterizado por $(\theta_{iz}, \theta_{iX}, \theta_{iY})$, parámetros que la cinemática inversa utiliza para modelar tanto la elevación real del cable como la orientación de su proyección sobre el plano XY.

Para calcular la cinemática directa del CDPR, se plantea un problema de optimización debido a la redundancia (8 cables, 6 grados de libertad) y a las imprecisiones geométricas y de medida. Se minimiza la discrepancia entre las longitudes teóricas y las medidas mediante un ajuste de mínimos cuadrados (4), y se resuelve iterativamente aplicando el método de Gauss-Newton (Bieber et al., 2023). En este contexto, el parámetro a optimizar es la pose $\mathbf{p} \in \mathbb{R}^6$ de la plataforma.

$$\min_{\mathbf{p}^* \in \mathbb{R}^6} \|\mathbf{l}(\mathbf{p}^*) - \mathbf{l}^*\|, \quad (4)$$

En el controlador industrial es imprescindible obtener la cinemática directa con precisión para el proceso de inicialización de grupo cinemático (grupo de servomotores actuando en conjunto con una cinemática específica). Al crearse un grupo cinemático, el controlador envía la posición destino calculada al servoamplificador que regula cada servomotor que controla la longitud del cable. Si la cinemática directa no se corresponde con la realidad física, esta diferencia entre lo calculado y lo real tiene como consecuencia una consigna que genera aceleraciones instantáneas elevadas, lo que provoca tirones en los ejes de los servomotores en el instante inicial.

3. Arquitectura

En esta sección se presenta la arquitectura de sistemas para la generación automática de la sombra digital a partir de la información disponible en el controlador industrial. Como se

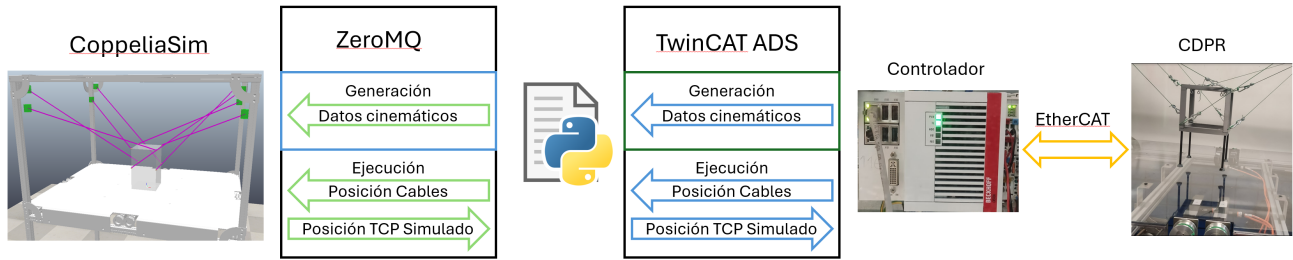


Figura 2: Arquitectura de comunicaciones del pipeline: un puente Python-ZeroMQ conecta el controlador industrial con Coppeliasim, donde se despliega la sombra digital del CDPR. Los datos cinemáticos generados en el controlador se envían al simulador, que devuelve la posición TCP simulada y las referencias de cable; paralelamente, el controlador puede gobernar el robot real enviando las consignas a través de EtherCAT.

ilustra en la Figura 2, la arquitectura consta de tres componentes principales: el robot (real y su sombra digital), el controlador industrial, y un *script* en Python que actúa tanto como generador automático de la sombra digital como pasarela de comunicación entre los otros dos elementos. La sombra digital se implementa en Coppeliasim, un simulador robótico que permite crear, configurar y simular sistemas robóticos complejos. Dado que el objetivo es validar el comportamiento del controlador industrial bajo distintas configuraciones geométricas del CDPR y no modelar de forma exhaustiva la dinámica de los cables, la simplificación empleada en Coppeliasim resulta adecuada. El controlador real está basado en tecnología Beckhoff y ejecuta el control del robot mediante una aplicación en C++ que utiliza la librería “Kinematic Transformation” para implementar cálculos cinemáticos personalizados. La comunicación entre el controlador y el entorno de simulación se realiza mediante un *script* en Python, que, empleando las librerías adecuadas, permite la sincronización de datos en tiempo real entre ambos entornos. Estas librerías son:

- PyAds: Librería que permite emplear herramientas para conectar una aplicación con un controlador Beckhoff empleando TwinCAT ADS (Automation Device Specification) (Stefan Lehmann, 2015), protocolo de comunicación que permite crear un canal de comunicación directo entre el controlador industrial y la aplicación para lectura/escritura de variables y símbolos, como se emplea en la literatura (Moreno and Torres, 2024).
- `coppeliasim_zmqremoteapi_client`: Librería orientada a crear un cliente de ZeroMQ para el control de la simulación (CoppeliasimRobotics, 2025), permitiendo usar las funciones definidas para los *scripts* propios de la aplicación Coppeliasim de forma remota, como se emplea en la literatura (Marcus et al., 2025).

4. Generación automática de modelos de la sombra digital de CDPR

La construcción de la sombra digital en una aplicación de simulación es laboriosa, debiendo realizar los cálculos de posición inicial de los cables de forma manual para los distintos parámetros dimensionales (posiciones espaciales, ángulos de salida del cable, puntos de anclaje en el efector final, etc.). La construcción ad-hoc del modelo digital también implica el tener una pose inicial fija, obligando a que la vinculación entre este y el controlador se tenga que realizar en poses cercanas a la digital. La propia modificación de un modelo ya construido para adaptarlo a una máquina con dimensiones distintas, o a

un cambio del efector final, no reduce en gran medida el trabajo de construir el modelo digital desde cero. La posibilidad de realizar la construcción de este modelo digital de manera automática, extrayendo datos del controlador industrial e iniciando la simulación a partir de la pose actual, facilita en gran medida las pruebas con distintas configuraciones espaciales y distintos efectores finales de forma rápida y con un coste bajo.

A continuación se describe el procedimiento implementado para la generación automática del sombra digital del CDPR. Este permite que la sombra digital refleje fielmente y en tiempo real la configuración física del robot, empleando la información obtenida directamente desde el controlador industrial. Los cálculos cinemáticos (sección 2) están implementados como un objeto tecnológico programado en C++ y ejecutado en el controlador industrial (con un tiempo de ciclo de 2ms). Este objeto permite la creación de símbolos para su conexión con la aplicación de PLC así como la autorización del acceso por parte de una aplicación externa empleando el protocolo de comunicaciones TwinCAT ADS. Todos los datos cinemáticos, así como estructurales, son calculados en el objeto tecnológico y compartidos a través de estos símbolos.

Siguiendo el procedimiento indicado la sección 3, la aplicación programada en Python accederá a la información compartida a través de estos símbolos para obtener los puntos de anclaje proximales respecto al sistema de coordenadas global así como los puntos de anclaje distales respecto al TCP.

De manera general, el procedimiento realizado para la creación de la sombra digital y su correcta puesta en marcha se resume en:

1. Puesta en marcha e inicialización de la cinemática en el controlador industrial.
2. Extracción de los datos que definen la estructura mecánica del robot (empleando TwinCAT-ADS).
3. Creación de la sombra digital en Coppeliasim empleando las funciones de ZeroMQ.
4. Puesta en marcha de la simulación.
5. Lectura de las longitudes actuales de los cables reales y modificación de las longitudes de los cables simulados.
6. Lectura de la posición actual del efector final de la simulación y envío de esta información al sistema de control para su monitorización.

4.1. Modelo base de cable

La escena de simulación se genera automáticamente a partir de una plantilla base que contiene el modelo genérico de cable con sus distintos componentes (“CableBase”). Para cada cable del CDPR se crea una representación física y visual

en la simulación. Dado que CoppeliaSim no tiene un objeto cable flexible nativo, se utiliza una estructura de articulaciones y cuerpos rígidos que emula el comportamiento del cable. En concreto, se sigue una estrategia común en la literatura: insertar una articulación prismática que representa la longitud variable del cable, junto con elementos pasivos para manejar la dirección y tensión (Zake et al., 2019). Los elementos móviles que forman el cable son:

- Junta rotativa: aplica la orientación azimuthal según el ángulo θ_{ix} (3).
- Segunda junta rotativa: aplica el ángulo θ_{iz} (2) de forma que el cable se alinea con el vector que une el punto de salida del cable con la unión al efector final.
- Articulación prismática activa: su desplazamiento equivale a la longitud efectiva del cable. Está configurada en modo posición-controlada y se agrega un “*Custom control*” para la aplicación de un PID que garantice el seguimiento.
- Articulación prismática pasiva en modo resorte-amortiguador. Mantiene el cable en tensión absorbiendo los esfuerzos de compresión que podría realizar la articulación prismática activa.
- Junta esférica que concede libertad al extremo del cable en el punto de unión con el efector final.
- Punto de referencia (*dummy*) que se enlaza mediante una restricción dinámica con *dummy-B* en la plataforma móvil, cerrando la cadena cinemática.

Las distintas articulaciones se unen entre sí empleando pequeños sólidos, estos también permiten incluir la masa del cable si esta se considera dinámicamente relevante. En CoppeliaSim, el árbol quedaría de la siguiente manera (Figura 3).

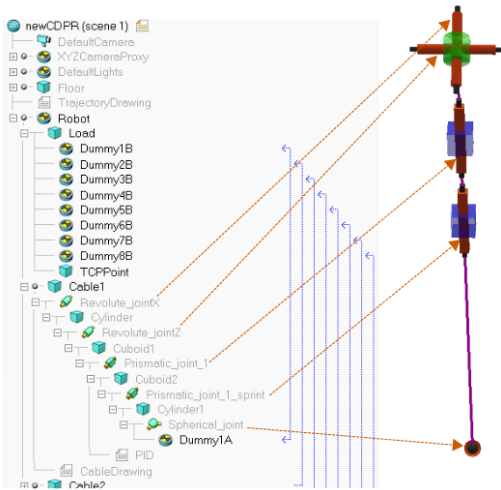


Figura 3: Árbol jerárquico del modelo de un cable en CoppeliaSim.

4.2. Lectura de datos desde el controlador industrial

Del controlador industrial se extraen:

- Posiciones de anclaje: puntos de anclaje de los cables en la estructura A_i , y puntos de sujeción en el efector final B_i . Estos datos definen la geometría base del CDPR.
- Longitudes de cables: el estado actual de cada cable, L_i .
- Orientación del punto proximal de los cables: ángulos θ_{iz} y θ_{ix} que definen la orientación de cada cable.
- Última pose cartesiana $p = (x, y, z, \alpha, \beta, \gamma)$ del TCP empleada para el cálculo de la cinemática inversa.

Todos estos valores se convierten cuando son leídos desde milímetros y grados a metros y radianes, facilitando así su uso posterior en el entorno de simulación. Este conjunto de valores conforman la descripción completa del estado y geometría del robot físico en ese instante. El *script* almacena estos parámetros y procede a usarlos para generar la escena de simulación.

En el caso del robot empleado en este artículo, los ocho cables están situados en la zona alta, siendo así un CDPR de ocho cables suspendido, como se aprecia en la Figura 2. Las distancias x e y de los puntos proximales son (1325,45, 1096,56)mm y (1396,56, 1025,45)mm para los cables superiores e inferiores, estando a una altura (z) de 825mm y 700mm respectivamente. Los puntos distales están separados en (200, 200)mm, tanto los superiores como los inferiores. Los cables superiores conectan con el efector final (proximal) a una altura de 175mm y los inferiores a una altura de 330mm (cruzados). Además se encuentran rotados en la conexión con el efector final en 90° los cables superiores y -90° los inferiores.

Todos estos parámetros constituyen una configuración concreta del CDPR. Cualquier cambio en ellos implica una nueva configuración lo cual supone volver a calcular los datos de los puntos de anclaje y los ángulos de salida de los cables; reconstrucción completa de la sombra digital.

4.3. Generación automática de la escena

Con los datos disponibles, el *script* en Python inicia la conexión con CoppeliaSim usando ZeroMQ. La generación automática del modelo completo del CDPR consiste en el siguiente iterativo, que se compone de la siguiente secuencia de pasos:

Paso 1 - Creación de la base del robot y del efector final. Se crea un objeto de referencia (*dummy*) que hará la función de sistema de referencia global. Todos los demás objetos creados serán hijos de este elemento. A continuación, a partir de la información obtenida de los puntos de anclaje del efector final se agrega un sólido básico (*cuboid*) que lo representará, tendrá comportamiento dinámico y una masa de 1Kg, así como una pequeña esfera que representará el TCP como objeto padre del efector final. Como objeto dependiente del efector final creado, se agrega un nuevo *dummy* que será clonado posteriormente para crear los puntos de anclaje de los cables, permitiendo en estos puntos el cierre de las distintas cadenas cinemáticas.

Paso 2 – Duplicación y posicionamiento inicial del cable. Para cada cable, se clona el modelo base (“*CableBase*”), obteniendo así una copia específica para el cable i -ésimo. Esta copia es trasladada desde su ubicación original hasta la posición exacta del punto fijo de anclaje obtenido previamente desde el controlador industrial.

Paso 3 – Corrección de la orientación inicial del cable. Se ajusta la orientación del cable empleando dos articulaciones rotativas consecutivas que permiten direccionar el cable desde el punto de anclaje proximal al punto de anclaje distal. La primera articulación rotativa, alrededor del eje vertical y se ajusta mediante (5). La segunda articulación rotativa se orienta empleando el ángulo calculado entre el vector del cable y su proyección horizontal,

$$\theta'_{ix} = c_{dir} \theta_{ix} + c_{off} \quad c_{dir} \in \{\pm 1\}, \quad c_{off} \in \{0^\circ, 180^\circ\}, \quad (5)$$

donde los coeficientes c_{dir} y c_{off} se determinan automáticamente según el cuadrante en el plano horizontal en el que

se sitúa el punto de anclaje. Esta corrección asegura que las orientaciones angulares en la simulación sean coherentes con las convenciones empleadas en el controlador industrial.

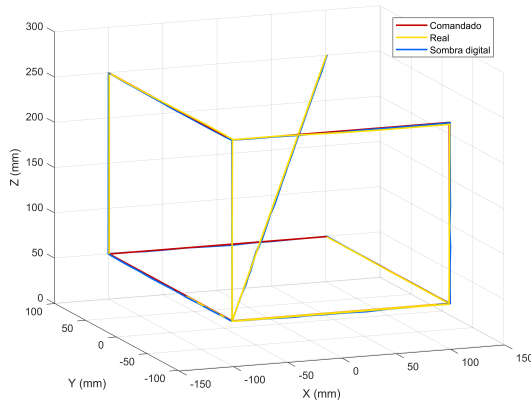


Figura 4: Trayectoria realizada a 400 mm/s.

Paso 4 – Ajuste inicial de la longitud del cable. La longitud inicial de cada cable se establece situando el *dummy* que representa punto distal a una distancia igual a esta longitud del punto proximal. Esta longitud inicial se empleará como *offset* para asignar la posición a la articulación prismática que representa la longitud del cable (inicialmente su longitud es cero).

Paso 5 – Cierre de la cadena cinemática del cable. El extremo móvil del cable es conectado con la plataforma mediante la creación de un punto de referencia móvil (*Dummy* i-ésimo B). Este punto se obtiene copiando el *dummy* genérico previamente generado y asociado al efector final, y después se ubica en la posición correspondiente de conexión a la plataforma. Finalmente, se enlaza al punto correspondiente ubicado en el extremo del cable (*Dummy* i-ésimo A), cerrando así la cadena cinemática que describe al cable en la simulación.

Paso 6 – Posicionamiento del TCP. Tras finalizar la generación automática del modelo, se procede al posicionamiento del TCP aplicando la pose actual obtenida del controlador. Al finalizar la generación del modelo, se inicia la ejecución de la sombra digital. Para ello, el *script* restablece la conexión con el controlador industrial para ejecutar un ciclo continuo de intercambio de datos (de media entre 48-60ms), donde se actualizan las posiciones objetivo de las articulaciones prismáticas de cada cable simulado con las nuevas longitudes obtenidas

del controlador industrial. Simultáneamente, la posición actual del efector simulado se devuelve al controlador, donde se monitoriza la trayectoria virtual realizada.

5. Experimento y resultados

Para verificar la adecuación del modelo de simulación generado se realizó el siguiente experimento. Se ejecutó una trayectoria en el controlador industrial muestreando (a 2ms) la posición comandada, la realizada por el robot (calculada a través de la cinemática directa en el controlador industrial) y la realizada por la sombra digital. La trayectoria se compone de movimientos lineales (mm) en los rangos: $X=[-100, 100]$, $Y=[-100, 100]$ y $Z=[50, 250]$ y un movimiento diagonal de ida y vuelta entre los puntos $[-100, -100, 50]$ y $[100, 100, 250]$ como se muestra en la Figura 4. Esta trayectoria se realizó cinco veces a velocidades de 100 y 400 mm/s, siendo este último un valor cercano al límite de velocidad del CDPR real.

La Figura 5 muestra una comparativa visual de una ejecución completa de trayectoria. La Tabla 1 muestra las métricas comparativas entre la trayectoria real y la ejecutada en el simulador de CoppeliaSim. Para el cálculo de las métricas de error se corrigió previamente el desfase de los datos (35 muestras para 100 y 400 mm/s respectivamente). Este desfase se calculó aplicando métodos de correlación cruzada (MathWorks, 2024). Del controlador se obtiene un nuevo dato en cada instante de muestreo, mientras que los datos obtenidos de la sombra digital deben ser adaptados, ya que el muestreo se realiza cada 2ms y los datos de la sombra digital se actualizan cada 50ms. Por esto los últimos puntos obtenidos se repiten haciendo un efecto de escalera. Para aproximar estos datos a la realidad del comportamiento de la sombra digital se realiza una interpolación lineal.

Observando los datos de la tabla se observan errores RMS similares en los desplazamientos individuales de cada eje, siendo aproximadamente el doble en el caso de los movimientos a 400mm/s ($0,866\text{mm} \pm 0,088\text{mm}$ frente a $1,841\text{mm} \pm 0,323\text{mm}$). También se calcula la distancia Fréchet, que cuantifica hasta qué punto dos trayectorias superponen sus recorridos en proximidad espacial y sincronía temporal, para los distintos desplazamientos. En los datos obtenidos se destaca un mayor error continuo al realizarse desplazamientos en el

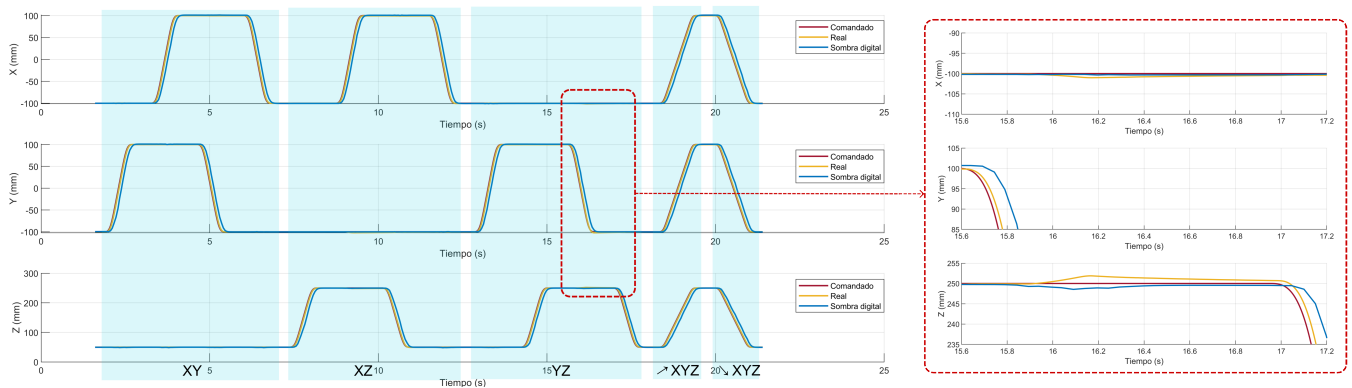


Figura 5: Trayectoria completa realizada a 400 mm/s por el robot en los ejes X (fila superior), Y (fila central) y Z (fila inferior) durante las secuencias: plano XY, plano XZ, plano YZ, diagonal de subida y diagonal de bajada (mostrados con azul sombreado). Se comparan la señal comandada (línea roja), la respuesta real calculada a partir de la cinemática directa (línea amarilla) y la simulación en CoppeliaSim (línea azul). En la parte ampliada (recuadro rojo) se observa como el movimiento en una dirección (Y) perturba la posición de las otras dos, siendo mayor en el eje vertical Z que es donde se soporta la carga.

Tabla 1: Métricas de error tras las 5 ejecuciones de las trayectorias a ambas velocidades considerando cada tramo. Se detallan en negrita los mejores resultados.

Plano	RMS X (mm)		RMS Y (mm)		RMS Z (mm)		RMS X-Y-Z (mm)		Fréchet (mm)	
	100 mm/s	400 mm/s	100 mm/s	400 mm/s	100 mm/s	400 mm/s	100 mm/s	400 mm/s	100 mm/s	400 mm/s
XY	0,564 ± 0,033	0,941 ± 0,087	0,508 ± 0,044	1,020 ± 0,145	0,368 ± 0,090	0,541 ± 0,065	0,849 ± 0,018	1,497 ± 0,085	1,055 ± 0,104	1,773 ± 0,102
XZ	0,580 ± 0,079	1,161 ± 0,231	0,190 ± 0,031	0,786 ± 0,035	0,568 ± 0,062	0,975 ± 0,180	0,837 ± 0,062	1,718 ± 0,206	0,794 ± 0,016	1,880 ± 0,121
YZ	0,345 ± 0,011	0,267 ± 0,006	0,529 ± 0,020	1,422 ± 0,069	0,555 ± 0,017	1,383 ± 0,033	0,841 ± 0,016	2,002 ± 0,050	0,990 ± 0,011	3,049 ± 0,073
Up	0,479 ± 0,087	0,900 ± 0,208	0,370 ± 0,117	0,855 ± 0,212	0,725 ± 0,106	1,491 ± 0,196	0,946 ± 0,171	1,942 ± 0,339	1,184 ± 0,007	1,661 ± 0,065
Down	0,608 ± 0,049	1,309 ± 0,260	0,414 ± 0,047	1,199 ± 0,298	0,437 ± 0,019	1,017 ± 0,213	0,857 ± 0,052	2,047 ± 0,440	1,176 ± 0,002	1,641 ± 0,315
TOTAL	0,515 ± 0,111	0,915 ± 0,402	0,402 ± 0,136	1,057 ± 0,288	0,531 ± 0,140	1,081 ± 0,372	0,866 ± 0,088	1,841 ± 0,323	1,040 ± 0,153	2,001 ± 0,562

plano YZ a velocidad alta. Esto es debido a que en estos casos el eje que no se mueve es el X, siendo este el que menor porcentaje de espacio de trabajo necesita para el experimento realizado. A nivel global de la trayectoria, la distancia obtenida es baja a ambas velocidades ($1,040\text{mm} \pm 0,153\text{mm}$ a 100mm/s y $2,001\text{mm} \pm 0,562\text{mm}$ a 400mm/s), indicando una buena capacidad de la sombra digital para realizar una trayectoria equiparable a la real incluso a mayor velocidad, a pesar de la reducción en la precisión.

En las dos pruebas realizadas, la sombra digital sigue a la consigna con un error que se incrementa cuanto mayor es la altura. Esto se debe a que, al situarse los cables más horizontales, la componente vertical se reduce aumentando la fuerza total que debe realizar la articulación activa. El comportamiento dinámico del simulador muestra este efecto, mientras en los datos de la cinemática directa calculada, al suponerse un cable ideal, se muestra la trayectoria teórica que está siguiendo el TCP. En el caso de los movimientos a 400mm/s los errores aumentan no solo por la dinámica, si no también por acercarse a las velocidades límites de los motores.

En el controlador industrial, las longitudes de los cables se estiman mediante un modelo cinemático ideal, calculadas a partir de la posición angular de los motores y del diámetro de los carretes, sin considerar efectos físicos como la flexión o el alargamiento de los cables. En cambio, en la sombra digital, las articulaciones están sometidas a la dinámica de la carga y a las interacciones entre los distintos cables, lo que introduce deformaciones y acoplamientos que afectan a la posición del TCP. Como resultado, el cálculo de la cinemática directa en el controlador y el dato proporcionado por la sombra digital presentan diferencias, derivadas de la discrepancia entre un modelo ideal y uno físicamente más representativo.

6. Conclusiones

La autogeneración de la sombra digital, sin intervención manual a partir de los parámetros actuales del CDPR en el controlador permite probar, de forma ágil, múltiples configuraciones de cables sólo modificando estos parámetros.

La comparación entre el seguimiento de la trayectoria por parte del robot real y por parte de su sombra digital permite concluir que el modelo autogenerado se adapta, cinemáticamente, a la configuración presente en el controlador industrial. Además, las métricas de error calculadas son bajas y consistentes en las distintas trayectorias y con distintas velocidades.

Como trabajo futuro, se estudiará la correlación entre las fuerzas realizadas en las articulaciones prismáticas con los datos de par de los servomotores, con el objetivo de implementar

un modelo dinámico en el controlador apoyado en la sombra digital. Finalmente, se extenderá la generación automática a configuraciones con poleas completas, incorporando los efectos de fricción y desvío de los cables sobre dichas poleas.

Agradecimientos

El proyecto VirtualR3 está financiado por el proyecto EMIL de apoyo financiero a terceros, financiado por la Unión Europea (UE) (HORIZON-CL4-2021-HUMAN-01-06). No obstante, los puntos de vista y opiniones expresados son exclusivamente los del autor/es y no reflejan necesariamente los de la UE. Ni la UE ni la autoridad que concede las ayudas pueden ser consideradas responsables de las mismas.

Referencias

- Bieber, J., Pallmer, S., Beitel Schmidt, M., Nov. 2023. Computationally efficient implementation of the Gauss–Newton method for solving the forward kinematics of redundant cable-driven parallel robots. *Proceedings in Applied Mathematics and Mechanics* 23 (3). DOI: 10.1002/pamm.202300231
- CoppeliaRobotics, Jan. 2025. *simZMQ*. Coppelia Robotics AG. URL: <https://github.com/CoppeliaRobotics/simZMQ>
- Kritzinger, W., Karner, M., Traar, G., Henjes, J., Sih, W., 2018. Digital Twin in manufacturing: A categorical literature review and classification. *IFAC-PapersOnLine* 51 (11), 1016–1022. DOI: 10.1016/j.ifacol.2018.08.474
- Marcus, S. D., Shapiro, A., Giladi, C., Apr. 2025. Enhancing Quadruped Robot Walking on Unstructured Terrains: A Combination of Stable Blind Gait and Deep Reinforcement Learning. *Electronics* 14 (7), 1431. DOI: 10.3390/electronics14071431
- MathWorks, 2024. *xcorr* - correlación cruzada (matlab). Accedido el 27 de mayo de 2025. URL: <https://es.mathworks.com/help/matlab/ref/xcorr.html>
- McDonald, E., Beites, S., Arsenault, M., 2022. CDPR Studio: A Parametric Design Tool for Simulating Cable-Suspended Parallel Robots. In: Gerber, D., Pantazis, E., Bogosian, B., Nahmad, A., Miltiadis, C. (Eds.), *Computer-Aided Architectural Design. Design Imperatives: The Future Is Now*. Vol. 1465. Springer, Singapore, pp. 344–359. DOI: 10.1007/978-981-19-1280-1_22
- Moreno, F.-A., Torres, V., Jul. 2024. ArduTC – Un entrenador de bajo coste para TwinCAT. *Jornadas de Automática* (45). DOI: 10.17979/ja-cea.2024.45.10951
- Stefan Lehmann, 2015. Welcome to pyads's documentation! URL: <https://pyads.readthedocs.io/en/latest/index.html>
- Zake, Z., Caro, S., Roos, A. S., Chaumette, F., Pedemonte, N., 2019. Stability Analysis of Pose-Based Visual Servoing Control of Cable-Driven Parallel Robots. In: Pott, A., Bruckmann, T. (Eds.), *Cable-Driven Parallel Robots*. Vol. 74. Springer International Publishing, Cham, pp. 73–84. DOI: 10.1007/978-3-030-20751-9_7
- Zhang, Z., Shao, Z., You, Z., Tang, X., Zi, B., Yang, G., Gosselin, C., Caro, S., Sep. 2022. State-of-the-art on theories and applications of cable-driven parallel robots. *Front. Mech. Eng.* 17 (3). DOI: 10.1007/s11465-022-0693-3