



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Dpto. de Ingeniería de Sistemas y Automática

Diseño y validación de dos controladores de admitancia
para el robot UR3

Trabajo Fin de Máster

Máster Universitario en Automática e Informática Industrial

AUTOR/A: Torres Ricalde, Diego

Tutor/a: Zotovic Stanisic, Ranko

CURSO ACADÉMICO: 2021/2022

RESUMEN

Este trabajo se enfoca principalmente en la implementación de dos controladores de admitancia distintos en el robot UR3e de Universal Robots. La diferencia entre estos controladores reside en que uno de ellos es capaz de modificar la inercia aparente del sistema, mientras que el otro mantiene la inercia propia del robot. De esta manera, la implementación de uno de los controladores frente al otro difiere en gran medida debido a la complejidad de diseñar un controlador que modifique efectivamente la inercia aparente del sistema.

La implementación de estos controladores es posible debido a las características dinámicas propias del robot UR3e, las cuales son exploradas con el fin de establecer las bases para el diseño de los controladores. De este modo, al aprovechar estas características, los controladores diseñados presentan un desempeño bastante satisfactorio al reproducir prácticamente a la perfección el comportamiento deseado.

Así mismo, se exploran las limitaciones del robot UR3e y se proponen soluciones con el fin de permitir que el controlador pueda imponer la dinámica deseada sobre el sistema. Adicionalmente, con el fin de mejorar el desempeño del controlador frente a estas limitaciones, se siguen ciertas directrices cuyo efecto es comprobado mediante los experimentos realizados, obteniendo aun mejores resultados.

Dado que los controladores presentados son capaces de imponer correctamente la dinámica deseada, es posible predecir la respuesta del sistema cuando este interactúa con su entorno si se conocen las fuerzas de interacción, lo que significa que el sistema puede ser simulado en un entorno virtual. La capacidad de hacer esto representa una gran ventaja, puesto que se pueden probar diferentes configuraciones de los parámetros y observar el resultado sin la necesidad de utilizar el robot. Más aun, debido a que la dinámica impuesta por el controlador es la de un sistema masa-resorte-amortiguador, es posible predecir las características del comportamiento que tendrá únicamente en base a los parámetros elegidos.

ÍNDICE DE CONTENIDOS

INTRODUCCIÓN	3
1. ANTECEDENTES	5
1.1. Introducción a los conceptos	5
1.2. Control de impedancia	11
1.3. Control de admitancia	11
2. ESTADO DEL ARTE	13
2.1. Control de admitancia	13
2.2. Control de admitancia variable	19
3. MODELO DINÁMICO DEL ROBOT UR3e	27
3.1. Modelo dinámico de un robot rígido	27
3.2. Modelo dinámico de un robot con articulaciones elásticas	29
3.3. Modelo dinámico del robot UR3e	30
4. DISEÑO DE LOS CONTROLADORES DE ADMITANCIA	37
4.1. Consideraciones preliminares	37
4.2. Controlador de admitancia sin modificación de la inercia aparente	38
4.3. Controlador de admitancia con modificación de la inercia aparente	39
5. EXPERIMENTOS Y RESULTADOS	42
5.1. Plataforma experimental y preparación de los experimentos	42
5.2. Resultados del controlador de admitancia sin modificación de la inercia aparente	46
5.3. Resultados del controlador de admitancia con modificación de la inercia aparente	57
CONCLUSIONES	97
OBSERVACIONES Y RECOMENDACIONES	98
BIBLIOGRAFÍA	100

INTRODUCCIÓN

En los últimos años el uso de brazos robóticos en la industria ha incrementado considerablemente (con un total de 12 millones de robots en funcionamiento a nivel mundial en el año 2021¹), sobre todo en lo que respecta al uso de robots colaborativos, también conocidos como “cobots”. De hecho, está previsto que estos constituyan un 34% de las ventas totales de robots para finales del año 2025¹. Así mismo, se espera que para el año 2030, el mercado de los robots colaborativos alcance un valor de 1990.2 millones de dólares (USD)². Una prueba del protagonismo que están comenzando a tener los cobots se puede apreciar al comparar la cantidad de robots que fueron instalados en el año 2017 (alrededor de 11000) y en el año 2020 (alrededor de 22000)³.

Los cobots son particularmente atractivos para las pequeñas y medianas empresas debido a que suponen una forma sencilla y efectiva de introducir la automatización en sus procesos. Adicionalmente, los cobots presentan un bajo coste (en comparación a los robots industriales), una gran facilidad para adaptarse a diferentes entornos y condiciones de trabajo, y un sistema de seguridad que les permite trabajar cerca a las personas (con o sin una valla de seguridad)⁴. Todas estas características resultan atractivas para las empresas debido a que estos robots pueden operar en espacios más reducidos sin dejar de lado la seguridad tanto del robot como de las personas que se encuentren cerca de este, lo cual reduce en gran medida los costes de instalación.

¹ Robotics Statistics You Need To Know in 2022. Enterprise Engineering Solutions. Enlace:

<https://www.eescorporation.com/robotics-statistics/#:~:text=From%202020%20to%202022%2C%20a,the%20time%20we%20reach%202028>

² Size of the collaborative (cobot) robot market worldwide in 2020 and 2021, with a forecast for 2022 to 2030. Enlace: <https://www.statista.com/statistics/748234/global-market-size-collaborative-robots/>

³ Collaborative robots worldwide - statistics & facts. Enlace: https://www.statista.com/topics/8062/collaborative-robots-worldwide/#topicHeader__wrapper

⁴ KNOW YOUR MACHINE: INDUSTRIAL ROBOTS VS. COBOTS. Enlace: <https://www.universal-robots.com/blog/know-your-machine-industrial-robots-vs-cobots/>

De esta forma, considerando que el 88% de las industrias a nivel global tienen planeado implementar soluciones de automatización robótica¹, el estudio de la implementación de controladores avanzados para estos robots resulta importante y, a su vez, beneficioso. Ahora, si bien existen varios algoritmos de control avanzados, algunos de los controladores más populares son el control de impedancia y el control de admitancia. Estos algoritmos permiten controlar la impedancia del efector final, lo que ayuda a que los robots puedan entrar en contacto con su entorno de una manera más segura al controlar las fuerzas de interacción.

No obstante, es usual que no se puedan implementar los algoritmos de control tal y como se deducen debido a que no es posible acceder a los comandos de bajo nivel del robot y, por lo tanto, no es posible enviar una señal de control a los motores de forma directa. Además, incluso si se pudiera acceder a estos comandos, normalmente no se tienen disponibles muchas de las características del robot (tales como el índice de reducción utilizado, la inercia de los eslabones, entre otras cosas), las cuales pueden influir negativamente en la implementación del controlador, dando lugar a un comportamiento no deseado o uno que resulte impredecible a priori.

Es por este motivo que el presente trabajo pretende sacar a la luz algunas de las características del robot UR3e, explicar la influencia de estas en el control y demostrar cómo se puede implementar un control de admitancia que presente un comportamiento estable y predecible. Adicionalmente, se expondrán las consideraciones que se tomaron para implementar dicho control debido a limitaciones en la programación del UR3e, y como estas consideraciones afectan al comportamiento del robot en ciertos casos extremos.

De esta forma, se comenzará por revisar los conceptos utilizados en estos algoritmos de control para luego mostrar algunos de los algoritmos más utilizados, su implementación y sus limitaciones. Luego, se expondrá brevemente el modelo dinámico de un robot que interactúa con su entorno, algunas de las características del UR3e y como se obtuvieron, y la influencia de la matriz jacobiana en el control. Seguidamente, se presentará el modelo utilizado, la deducción del algoritmo de control para el espacio cartesiano y el esquema de control resultante. Después de esto, se presentarán los resultados de los experimentos y se compararán con las simulaciones realizadas para validar el modelo. Finalmente, se discutirán los resultados y se explicarán las limitaciones, ventajas y desventajas de la implementación del modelo deducido.

CAPÍTULO 1

ANTECEDENTES

Esta sección pretende aportar información necesaria para poder entender tanto los conceptos de impedancia y admitancia, como el comportamiento del modelo impuesto por el controlador. Ahora, al hablar de control de impedancia, este normalmente se refiere, tanto al control de impedancia como al control de admitancia. Por lo tanto, es usual inferir que al referirse al control de impedancia sin mencionar el control de admitancia este englobe a ambos conceptos.

1.1. Introducción a los conceptos

El control de impedancia es un esquema de control capaz de lidiar con tareas que comprenden una interacción mecánica con el entorno. En este aspecto, la impedancia se refiere a la relación dinámica que existe entre las variables propias del movimiento y las variables de las fuerzas debido al contacto. De esta forma, el objetivo de este esquema de control es el de controlar dicha relación de la forma deseada para cumplir con los requerimientos de una tarea de manipulación. Así, es posible mejorar el desempeño del robot al mismo tiempo que se garantiza la seguridad en la interacción humano-robot [1].

Con el fin de entender mejor las ventajas y limitaciones del control de impedancia, se presenta la Tabla 1.1, la cual presenta una comparación entre otros esquemas de control usualmente utilizados. Aquí, es posible apreciar la ventaja que presenta el control de impedancia sobre otros esquemas de control al poder ser utilizado para cualquier tipo de movimiento y no necesitar un conocimiento previo del entorno. No obstante, es necesario señalar que, para obtener un comportamiento predecible del controlador, es necesario obtener un buen modelo

de la impedancia del robot, lo cual suele representar una gran dificultad, con el fin de que este se comporte de la forma deseada.

Tabla 1.1: Comparación entre diferentes esquemas de control. Modificado de [1].

Esquema de control	Espacio de trabajo	Variables medidas	Situaciones de aplicación	Objetivos de control	Limitaciones potenciales	Característica de impedancia
<i>Control de posición</i>	Espacio de tareas	Posición	Movimiento libre	Posición deseada	Fuerza de contacto excesiva	Alta impedancia
<i>Control de fuerza</i>	Espacio de tareas	Fuerza de contacto	Movimiento restringido	Fuerza de contacto deseada	Movimiento no controlado	Baja impedancia
<i>Control híbrido posición / fuerza</i>	Subespacio de posición	Posición	Todo tipo de movimientos	Posición deseada	Conocimiento previo del entorno	Alta impedancia
	Subespacio de fuerza	Fuerza de contacto		Fuerza de contacto deseada		Baja impedancia
<i>Control de impedancia</i>	Espacio de tareas	Posición, fuerza de contacto	Todo tipo de movimientos	Impedancia deseada	Modelo de impedancia incorrecto	Impedancia concebible

Así mismo, se presentan las diferentes estrategias utilizadas para implementar un control de impedancia en la Fig. 1.1. En esta es posible distinguir dos grandes grupos, aquellos que utilizan un control de impedancia constante, y aquellos en lo que la impedancia varía dependiendo de la situación en la que se encuentre el robot. Además, dentro del grupo de los controladores de impedancia variable, se tiene a aquellos controladores capaces de aprender la mejor combinación de parámetros que debe usar para obtener el comportamiento deseado.

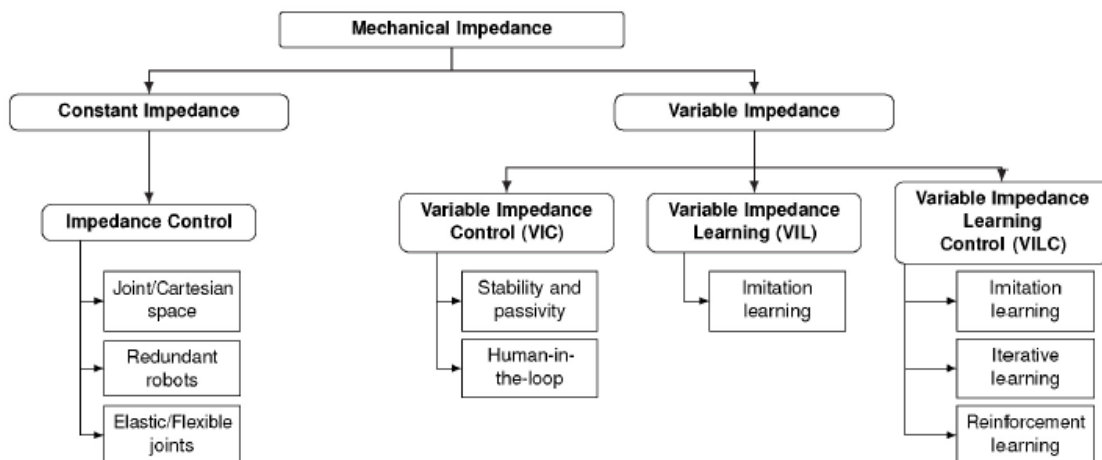


Fig. 1.1: Taxonomía de los enfoques para los algoritmos de control de impedancia. Extraído de [2].

Por otro lado, es necesario entender la impedancia como la relación entre la velocidad y la fuerza de interacción, tal y como se presenta en la ecuación 1.1 (en el dominio de Laplace). Así mismo, la impedancia representa la inercia, el amortiguamiento y la rigidez del sistema, como se puede observar en la ecuación 1.2. Entonces, al reemplazar la ecuación 1.2 en la ecuación 1.1, se obtiene la ecuación 1.3. Esta ecuación es el punto de partida para el desarrollo de un controlador de impedancia. Cabe señalar que el comportamiento del sistema, al utilizar este controlador, se puede entender como el comportamiento de un sistema masa-resorte-amortiguador (Fig. 1.2).

$$Z(s) = \frac{F(s)}{\dot{X}(s)} \quad (1.1)$$

$$Z(s) = ms + b + \frac{k}{s} \quad (1.2)$$

$$\frac{F(s)}{\dot{X}(s)} = ms + b + \frac{k}{s} \quad (1.3)$$

Donde:

- $Z(s)$: Impedancia mecánica del sistema
- $F(s)$: Fuerza de interacción
- $\dot{X}(s)$: Velocidad
- m : Inercia del sistema
- b : Amortiguamiento del sistema
- k : Rigidez del sistema

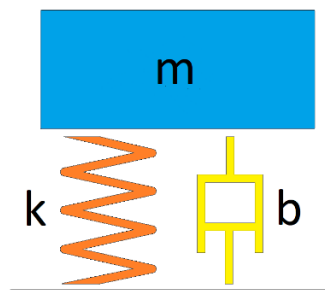


Fig. 1.2: Sistema masa-resorte-amortiguador.

Ahora, es posible darle la forma de un sistema de segundo orden (ecuación 1.4) a la ecuación 1.3 con el fin de poder analizar y predecir su comportamiento en base a la elección de la dinámica deseada. De esta manera, se obtiene la ecuación 1.5, la cual representa la función de transferencia del sistema en la forma de un sistema de segundo orden (en su forma canónica). Así mismo, es posible hacer uso de las propiedades de un sistema de este tipo, con lo que se

obtienen las ecuaciones 1.6, 1.7, 1.8 y 1.9, con las cuales es posible clasificar y predecir el tipo de comportamiento que tendrá el sistema en base a la elección de los parámetros deseados. Es importante señalar que, con el fin de obtener la forma correcta, ahora se tiene una relación entre la posición y la fuerza ejercida sobre el sistema.

$$\frac{X(s)}{F(s)} = K \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (1.4)$$

$$\frac{X(s)}{F(s)} = \frac{\frac{1}{m}}{s^2 + \frac{b}{m}s + \frac{k}{m}} \quad (1.5)$$

$$\omega_n = \sqrt{\frac{k}{m}} \quad (1.6)$$

$$\zeta = \frac{b}{2\sqrt{k \cdot m}} \quad (1.7)$$

$$K = \frac{1}{k} \quad (1.8)$$

$$\omega_d = \omega_n \sqrt{1 - \zeta^2} \quad (1.9)$$

Donde:

- ω_n : Frecuencia natural del sistema
- ω_d : Frecuencia natural amortiguada del sistema
- ζ : Factor de amortiguamiento del sistema
- K : Ganancia estática del sistema

De esta forma, ω_n y ω_d determinan la frecuencia del sistema en la ausencia de fuerzas externas cuando no existe y cuando existe amortiguamiento (solo para sistemas sub amortiguados), respectivamente. Así mismo, ζ indica el tipo de sistema amortiguado, y K define la ganancia del sistema. Entonces, dependiendo del factor de amortiguamiento (ζ), el sistema puede ser oscilatorio ($\zeta = 0$), sub amortiguado ($\zeta < 1$), críticamente amortiguado ($\zeta = 1$), o sobre amortiguado ($\zeta > 1$).

$$s_{1,2} = -\zeta\omega_n \pm \omega_n\sqrt{\zeta^2 - 1} = -\zeta\omega_n \pm j\omega_d \quad (1.10)$$

Por otro lado, el comportamiento también se puede analizar observando los polos del sistema ($s_{1,2}$), los cuales se calculan utilizando la ecuación 1.10. Así, es posible apreciar que, si aumenta el factor de amortiguamiento hasta $\zeta = 1$, los polos tienden a ir hacia el eje real, y si disminuye hasta $\zeta = 0$, los polos van hacia el eje imaginario. Además, al incrementar y disminuir el valor de ω_n , los polos tienden a crecer y decrecer, respectivamente, de forma radial (Fig. 1.3). Ahora, cuando $\zeta = 1$, los polos se encuentran en el mismo lugar y se sitúan sobre el eje real dependiendo del valor de ω_n (Fig. 1.4a). Luego, cuando $\zeta > 1$ los polos se separan, de tal forma que, conforme incrementa el valor del factor de amortiguamiento, uno de los polos va hacia el origen, mientras que el otro va en sentido contrario (Fig. 1.4b). Finalmente, si $\zeta = 0$, ambos polos se sitúan de sobre el eje imaginario dependiendo del valor de ω_n (Fig. 1.4c).

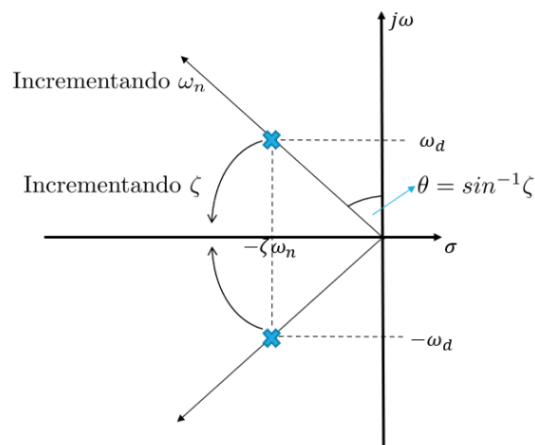


Fig. 1.3: Diagrama de polos de un sistema subamortiguado⁵.

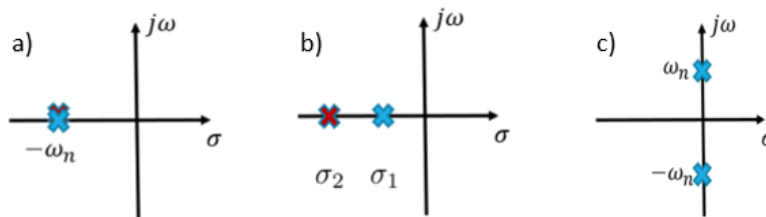


Fig. 1.4: Diagrama de polos de un sistema críticamente amortiguado (a), sobre amortiguado (b) y oscilatorio (c)⁵.

Entonces, recordando que el lugar de los polos determina las características del sistema, si los polos se encuentran más cerca al origen, el sistema es más lento, mientras que, si estos se encuentran lejos del origen, el sistema es más rápido. Así mismo, es útil recordar que los polos situados sobre el eje real que estén más cercanos al origen serán siempre los que tengan un mayor efecto sobre la dinámica del sistema. Esto explica porque un sistema sobre amortiguado presenta un comportamiento lento a pesar de tener un polo más alejado del origen que el otro.

⁵ Sistemas de Segundo Orden. Enlace: <https://controlautomaticoeducacion.com/control-realimentado/sistemas-de-segundo-orden/>

Finalmente, la ecuación diferencial (general) que describe el comportamiento deseado del sistema (ecuación 1.11) se obtiene a partir de la ecuación 1.3 al resolver para $F(s)$, pasar al dominio del tiempo y establecer los valores deseados para las variables del movimiento. Cabe señalar que, para el diseño del controlador, se utilizan los coeficientes m_d , b_d y k_d que permitan obtener el comportamiento deseado del sistema. Así mismo, es importante notar que se pueden obtener distintas formulaciones de la ecuación del sistema dependiendo de las variables de la ecuación 1.11 que se deseen considerar. Por ejemplo, si se toman $\ddot{x}_d = 0$ y $\dot{x}_d = 0$, se tiene la ecuación 1.12, la cual es utilizada posteriormente para el diseño de uno de los controladores. Así mismo, si ahora también se considera $m_d = m_r$ (por lo que la inercia aparente no es modificada), se tiene la ecuación 1.13, la cual es utilizada para el diseño del otro controlador.

$$F(t) = m_d(\ddot{x} - \ddot{x}_d) + b_d(\dot{x} - \dot{x}_d) + k_d(x - x_d) \quad (1.11)$$

$$F(t) = m_d\ddot{x} + b_d\dot{x} + k_d(x - x_d) \quad (1.12)$$

$$F(t) = m_r\ddot{x} + b_d\dot{x} + k_d(x - x_d) \quad (1.13)$$

Donde:

- $F(t)$: Fuerza de interacción
- m_d : Inercia deseada
- m_r : Inercia del robot
- b_d : Amortiguamiento deseado
- k_d : Rigidez deseada
- \ddot{x} : Aceleración actual
- \ddot{x}_d : Aceleración deseada
- \dot{x} : Velocidad actual
- \dot{x}_d : Velocidad deseada
- x : Posición actual
- x_d : Posición deseada

A continuación, se presentan características propias del control de impedancia y el control de admitancia con el propósito de entender mejor las diferencias entre estos y las razones por las que suele utilizar uno u otro.

1.2. Control de impedancia

El control por impedancia se caracteriza por generar una referencia de fuerza en base a al error que exista entre las variables de movimiento deseadas y las reales (Fig. 1.5). De esta manera, se obtiene una cierta complicitad por parte del robot que depende de dicho error (de posición, velocidad y/o aceleración según como se haya implementado el controlador). No obstante, una limitación para la implementación de este tipo de controladores es que es necesario controlar los torques de los motores directamente, lo cual no suele ser posible en robots comerciales [2].

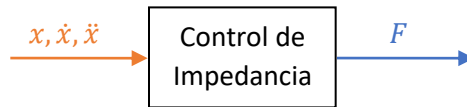


Fig. 1.5: Representación del controlador de impedancia.

Es necesario mencionar que, al implementar un control por impedancia, es posible prescindir de un sensor de fuerza siempre y cuando la inercia deseada sea igual a la inercia del robot. Esto representa una ventaja debido a que los sensores de fuerza suelen tener un precio considerable, además de tener una cantidad significativa de ruido.

La ecuación 1.14 representa lo que sucede en el interior del bloque de control de impedancia mostrado en la Fig. 1.5 (en el dominio de Laplace).

$$F(s) = \dot{X}(s)Z(s) = \dot{X}(s) \left(ms + b + \frac{k}{s} \right) \quad (1.14)$$

1.3. Control de admitancia

Dado que la admitancia es complementaria a la impedancia, el control de admitancia genera una referencia de velocidad en base a la fuerza de interacción ejercida sobre el robot (Fig. 1.6). Es usual utilizar este tipo de control cuando no se puede acceder a los controles de bajo nivel del robot (por ejemplo, el torque de las articulaciones o la corriente de los motores) [2]. Por lo tanto, el uso de un control de admitancia es una forma efectiva de solucionar la limitación que presenta el control de impedancia en su implementación cuando se trata de robots comerciales.

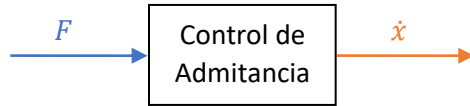


Fig. 1.6: Representación del controlador de admitancia.

Cabe señalar que para implementar un control de admitancia es necesario que el robot cuente con un sensor de fuerza, ya sea interno (como en el caso de algunos cobots) o externo (acoplado al efector final del robot).

La ecuación 1.15 representa lo que sucede en el interior del bloque de control de admitancia mostrado en la Fig. 1.6 (en el dominio de Laplace).

$$\dot{X}(s) = \frac{F(s)}{Z(s)} = \frac{F(s)}{\left(ms + b + \frac{k}{s}\right)} \quad (1.15)$$

CAPÍTULO 2

ESTADO DEL ARTE

En esta sección se presentan los esquemas de control y las estrategias utilizadas para asegurar el cumplimiento de diversas tareas mediante el uso de un control de admitancia, ya sea con parámetros fijos o variables.

2.1. Control de admitancia

En el control de admitancia con parámetros fijos, se ajustan los parámetros previamente para que el sistema se comporte de una forma predeterminada y predecible todo el tiempo.

De esta forma, en [3] se concluye que mediante un controlador de admitancia básico (Fig. 2.1) no es posible disminuir la inercia percibida por debajo de la propia inercia del robot al mismo tiempo que se asegura la estabilidad de este para todos los casos posibles de interacción con humanos. Esto se debe a que cualquier intento de reducción de la inercia percibida (por debajo de la inercia del propio robot) conduce a la inestabilidad acoplada, es decir, la inestabilidad cuando el robot interactúa con su entorno o con un humano. Adicionalmente, este tipo de control también puede llegar a ser inestable al interactuar con un entorno muy rígido.

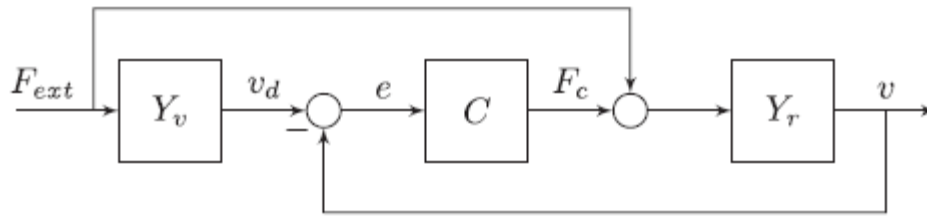


Fig. 2.1: Controlador de admitancia básico. F_{ext} es la fuerza externa, Y_v es el controlador de admitancia, C es el controlador de velocidad del robot, y Y_r es la dinámica del robot. Extraído de [3].

Por tal motivo, se proponen una serie de directrices que pretenden mejorar el control y la estabilidad de los robots. La aplicación de algunos de los métodos propuestos permite incluso disminuir la inercia percibida por el usuario por debajo de la propia inercia del robot, aunque esta nunca pueda disminuir más allá de la inercia que se encuentra después del sensor de fuerza utilizado sin que se tenga un comportamiento inestable.

Es necesario mencionar que las directrices presentadas son obtenidas a partir del análisis de un robot rígido de un solo grado de libertad. Por lo tanto, en robots con varios grados de libertad, el acoplamiento energético con otro sistema no lineal podría resultar en efectos de inestabilidad ausentes en el análisis realizado. Adicionalmente, no se ha evaluado el uso combinado de directrices, por lo que esto podría resultar en una interacción inesperada.

A manera de resumen, se presenta la Tabla 2.1, donde se presentan la motivación para la implementación de cada directriz.

Tabla 2.1: Directrices y motivación para su implementación. Modificado de [3].

	Directriz	Motivación
1	Usar un control con feed-forward	Disminuye efectivamente la inercia que debe ser reducida por el control de admitancia, permitiendo que la inercia se reduzca, como máximo, hasta igualar la inercia luego del sensor.
2	Evitar el filtrado de la fuerza	El filtrado introduce un retardo de fase excesivo en el modelo de inercia virtual marginalmente pasivo.
3	Compensar la inercia luego del sensor	Reduce la inercia aparente, pero también los márgenes de la estabilidad acoplada.
4	Usar un poco de amortiguamiento virtual	Permite un mejor seguimiento de la admitancia a bajas frecuencias.
5	Modificar la referencia de velocidad	El adelanto de fase incrementa los márgenes de la estabilidad acoplada y puede mejorar el seguimiento del modelo de inercia pura.
6	Incrementar el ancho de banda del lazo de velocidad	Empuja el retardo excesivo a frecuencias más altas, con lo que se requiere un entorno más rígido para desestabilizar al sistema acoplado.
7	Optimizar para la rigidez del robot	Los modos de resonancia interna introducen un retardo de fase entre la medición del sensor de fuerza y la medición del sensor de velocidad.

Por otro lado, en [4] se presenta un controlador de admitancia con amortiguación pura. Así, dado que este no pretende modificar la inercia aparente del robot y tampoco se desea que el manipulador se comporte como un resorte, no se incluye el término de inercia y los valores de la matriz K_p son fijados a cero. Además, dado que los robots utilizados (dos robots UR5) incorporan la compensación de gravedad se consiguió un controlador de admitancia como el que se muestra en la ecuación 2.1, donde h es la fuerza medida en el efector final, A relaciona la derivada temporal de los ángulos de Euler a la velocidad angular, K_D es la matriz de coeficientes de amortiguación, \dot{x}_e es la velocidad actual del efector final, y \dot{x}_d es la velocidad deseada del efector final.

$$\dot{x}_d = \dot{x}_e - K_D^{-1} A^T(\phi_e) h \quad (2.1)$$

Entonces, el control de admitancia implementado recibe una referencia de velocidad debido a que el robot es teleoperado. De esta forma, al utilizar una referencia de velocidad, la velocidad deseada para el robot puede ser fácilmente calculada y pasada directamente al controlador de velocidad, el cual presenta un mejor comportamiento que el controlador de posición debido a que este último sufre de ligeras vibraciones.

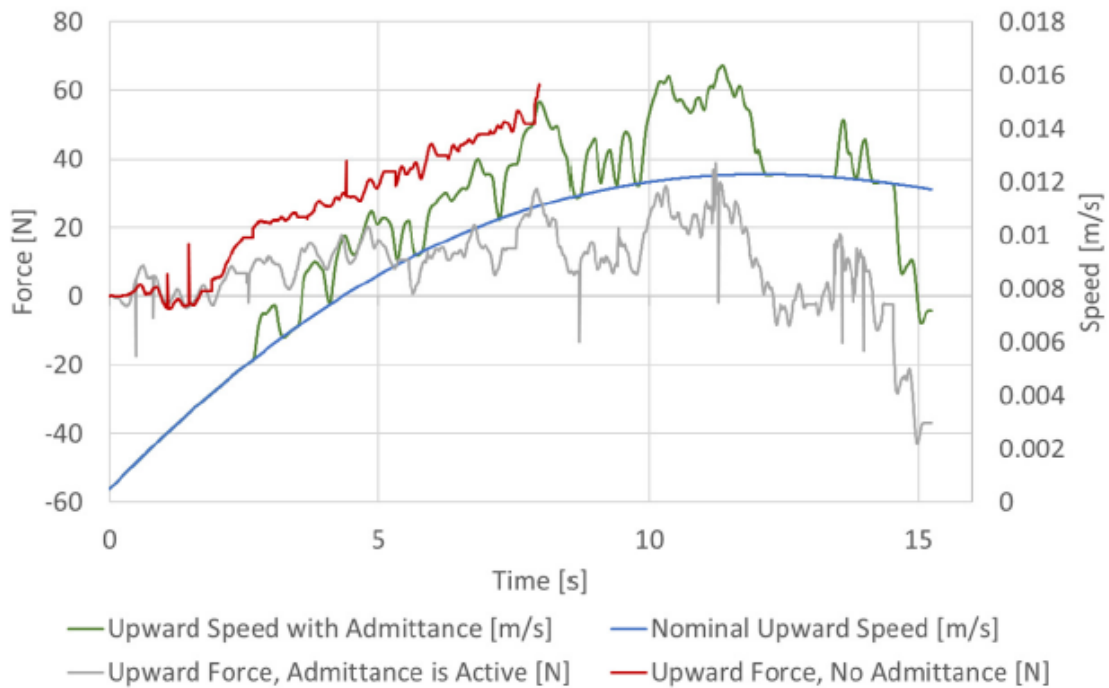


Fig. 2.2: Datos de fuerza y velocidad durante la ejecución de la tarea de abrir una puerta. Extraído de [4].

En la Fig. 2.2, se puede comprobar la eficacia del controlador de admitancia implementado en [4] para completar la tarea de abrir una puerta de forma autónoma. La curva roja representa la fuerza medida al no utilizar el control de admitancia. Como se puede observar, esta sobrepasó

el límite de seguridad, por lo que el robot detuvo la acción y no pudo completar la tarea. Por otra parte, cuando se utilizó el control de admitancia, el robot trató de seguir la referencia de velocidad (curva azul), pero se adaptó al encontrar resistencia por parte del entorno (curva verde).

Otro controlador interesante es el desarrollado en [5], donde se presenta un método robusto que hace uso de una estrategia que involucra la detección de 3 fases en base a la fuerza de interacción y un control de admitancia con el cual se asegura el correcto cumplimiento de la tarea de conectar dos objetos. Este método es implementado haciendo uso de dos robots UR3.

El control de admitancia implementado resulta importante en este caso debido a que permite superar el problema de la orientación de los objetos que serán conectados, así como los posibles problemas debido a errores de calibración e incertidumbre al sujetar la pieza. Un ejemplo de esto se puede apreciar en la Fig. 2.3. Así mismo, el efecto del control de admitancia utilizado se puede observar en la zona azul de la Fig. 2.4, donde se ve claramente como el robot se mueve ligeramente en los 3 ejes con el fin de acomodar correctamente el objeto a conectar.

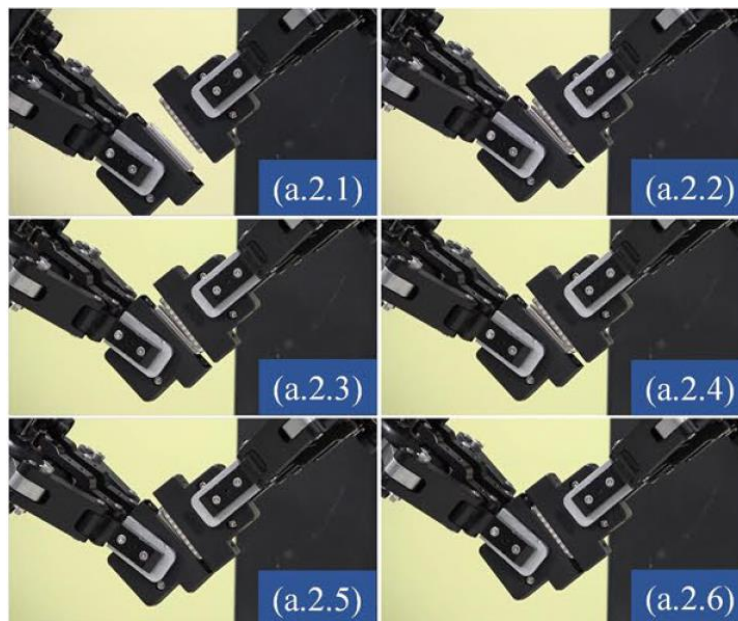


Fig. 2.3: Ejemplo de una aplicación del método propuesto. Extraído de [5].

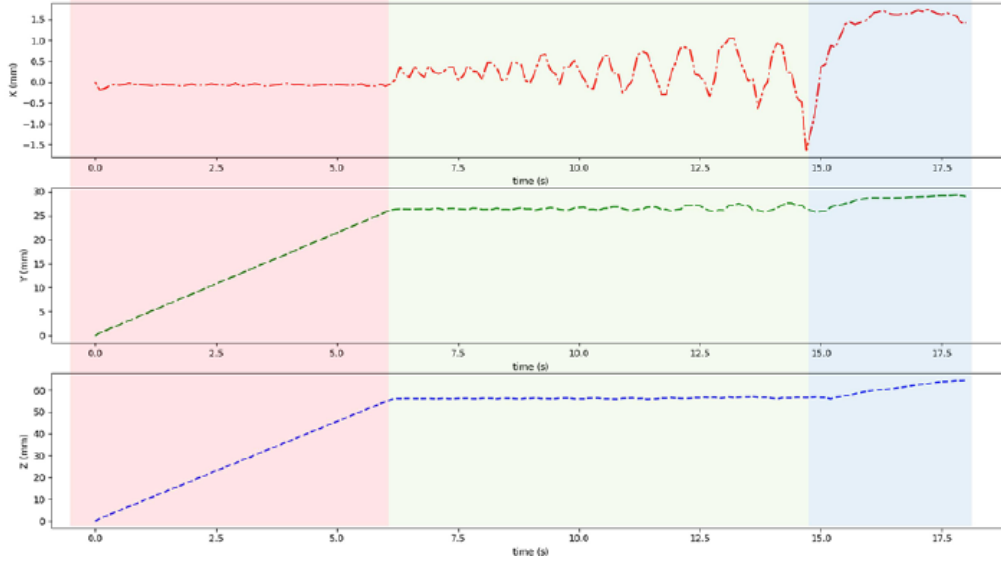


Fig. 2.4: Cambios en la posición de la pinza de acoplamiento. Extraída de [5].

El control de admitancia utilizado en [5] sigue la ley de control de Hogan, la cual se presenta en la ecuación 2.2, donde p_i^{hnd} es la posición de la mano que sostiene la pieza y F_i es la fuerza de interacción en el instante i . Así mismo, m , c , k son la masa, el amortiguamiento y la rigidez deseada del sistema, respectivamente, mientras que dt es el tiempo de muestreo.

$$p_{i+1}^{hnd} = \frac{F_i + m \frac{(2p_i^{hnd} - p_{i-1}^{hnd})}{dt^2} + c \frac{p_i^{hnd}}{dt} + k p_i^{hnd}}{\frac{m}{dt^2} + \frac{c}{dt} + k} \quad (2.2)$$

Por otra parte, en [6] se utilizó una matriz K_a de ganancias para convertir las mediciones del sensor de fuerza (F_{EE}) en comandos de velocidad (\dot{x}) (ecuación 2.3), por lo que este controlador también podría considerarse un controlador de amortiguamiento puro donde la referencia de velocidad es igual a cero.

$$\dot{x} = K_a \cdot F_{EE} \quad (2.3)$$

Así mismo, para solucionar el problema de las singularidades debido al uso de la inversa de la matriz jacobiana para pasar del espacio cartesiano al espacio de articulaciones, el trabajo realizado en [6] propone el uso de una matriz jacobiana amortiguada (J^*). De esta manera, el término λ en la ecuación 2.4 representa dicha amortiguación.

$$J^* = J^T (J \cdot J^T + \lambda^2 I)^{-1} \quad (2.4)$$

Con el fin de comprobar el buen desempeño del controlador desarrollado en [6], se realizaron experimentos con el robot UR3, para lo cual se fijó el valor de λ en 0.1, mientras que los valores de la matriz K_a fueron diferentes para cada experimento. A pesar de que, hipotéticamente, un mayor valor de K_a resultaría en una mejor respuesta del controlador, esto no sucedió debido a que el desempeño en el cumplimiento de la tarea dependió directamente de la comodidad del usuario con los parámetros utilizados. Por tal motivo, se llegó a la conclusión de que los parámetros utilizados deben ser dependientes del usuario y deben ser elegidos para cada uno.

Por otra parte, un método de control de fuerza-a-distancia basado en un controlador de admitancia es propuesto en [7] debido a que es más seguro y eficiente para tareas de contacto y tareas basadas en fuerza. Ahora, dado que el entorno con el que interactúa el robot es conocido, los parámetros del controlador de admitancia son fijados previamente con el fin de obtener el comportamiento deseado.

Por lo tanto, la tarea del agente (el cual aprende gracias al método de aprendizaje reforzado basado en la política del teorema del gradiente) utilizado en [7] es la de generar la referencia de fuerza que debe ejercer el robot sobre la superficie de contacto. De esta forma, a partir del error en la fuerza medida por el sensor, el controlador de admitancia genera una referencia de posición.

Así, se obtiene un comando de bajo nivel al pasar del espacio cartesiano al espacio de articulaciones (haciendo uso de la transpuesta de la matriz jacobiana), de tal forma que la fuerza de interacción se convierte en un comando para el motor, el cual es llevado a cabo por un controlador PD. Dicho comando es expresado en la ecuación 2.6, donde u representa la señal de control (el torque de los motores), K_d la matriz de amortiguamiento, y K_p la matriz de rigidez, mientras que q_{des} y \dot{q}_{des} representan la posición y la velocidad angulares deseadas, respectivamente. Cabe señalar que la función $ID(\cdot)$ representa la ecuación de la dinámica inversa de un robot, la cual está definida en la ecuación 2.5, donde, $D(q)$ representan la matriz de inercias, $C(q, \dot{q})$ la matriz de centrífugas y Coriolis, $G(q)$ la matriz de gravedad, τ el torque de los motores, y \ddot{q}_{des} la aceleración angular deseada.

$$\tau = ID(\ddot{q}_{des}) = D(q)\ddot{q}_{des} + C(q, \dot{q})\dot{q}_{des} + G(q) \quad (2.5)$$

$$u = ID(K_d(\dot{q} - \dot{q}_{des}) + K_p(q - q_{des})) \quad (2.6)$$

El sistema completo desarrollado en [7] se presenta en la Fig. 2.5, donde es posible apreciar al robot, que interactúa con el entorno, y también al modelo de aprendizaje y el método de control utilizado para completar la tarea de obtener imágenes del interior del cuerpo de una persona mediante ultrasonido de forma autónoma.

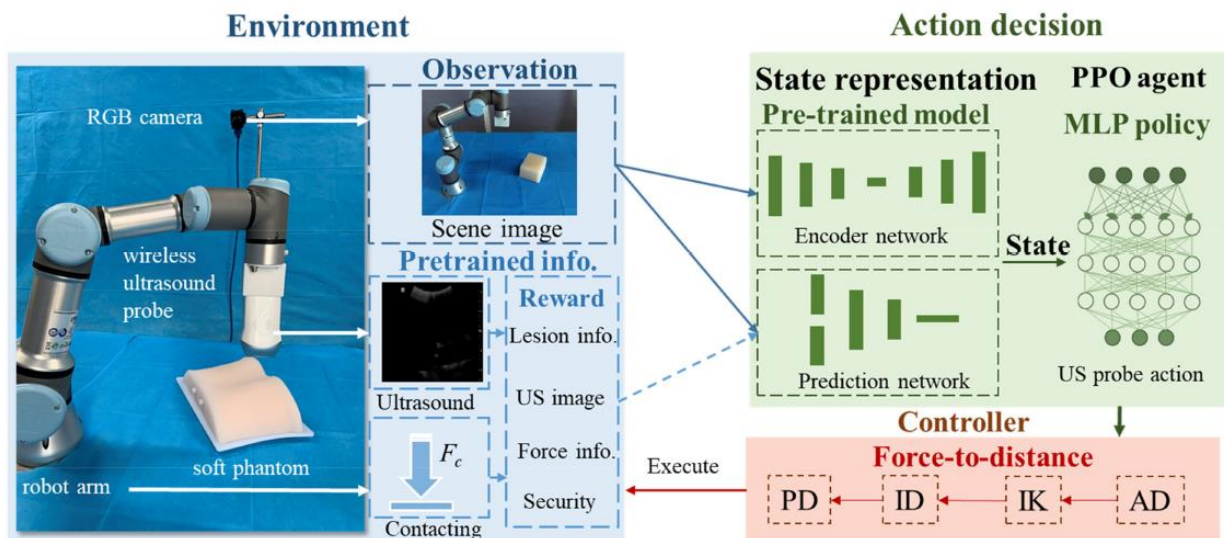


Fig. 2.5: Esquema del método y el marco de trabajo propuesto para el sistema de ultrasonido automático. Extraído de [7].

2.2. Control de admitancia variable

Por otro lado, en el control de admitancia variable, los parámetros son modificados en tiempo real según la situación del robot. Esto normalmente está regulado por un algoritmo que procura modificar los parámetros para alcanzar un comportamiento deseado en base a ciertas reglas.

De esta forma, el trabajo realizado en [8] está enfocado en asegurar que la interacción humano-robot se da de forma segura. Por lo tanto, investiga el problema que surge en el control de admitancia cuando el robot se encuentra en contacto con una superficie muy rígida. En este caso, dicha situación se da cuando el operador sostiene el efector final del robot muy rígidamente.

Por tal motivo, propone un método para detectar el momento en el que, debido a la interacción con el humano, el robot comienza a presentar signos de inestabilidad. Además, propone un algoritmo para hacer que el robot recobre la estabilidad al variar los parámetros del controlador de admitancia en base al principio de tanques de energía.

Dicho algoritmo hace uso de la abstracción de un tanque en el cual se almacena la energía disipada (energía virtual) por el sistema, la cual es utilizada solo cuando es estrictamente necesario. Por ejemplo, cuando se realiza un cambio en los parámetros del controlador, el uso de esta energía asegura la estabilidad durante la transición. Esto permite mantener una relación constante entre el amortiguamiento y la inercia aparente, lo que resulta más intuitivo para el operador que interactúa con el robot.

Un ejemplo del comportamiento descrito anteriormente se presenta en la Fig. 2.6, donde se observan dos casos en la adaptación de la inercia. La curva azul representa la variación de la inercia utilizando el principio de tanques de energía, mientras que la curva verde representa esta variación con el principio de pasividad. Así mismo, es posible apreciar los momentos en los que se detecta la inestabilidad y la inercia varía para mantener estable al sistema.

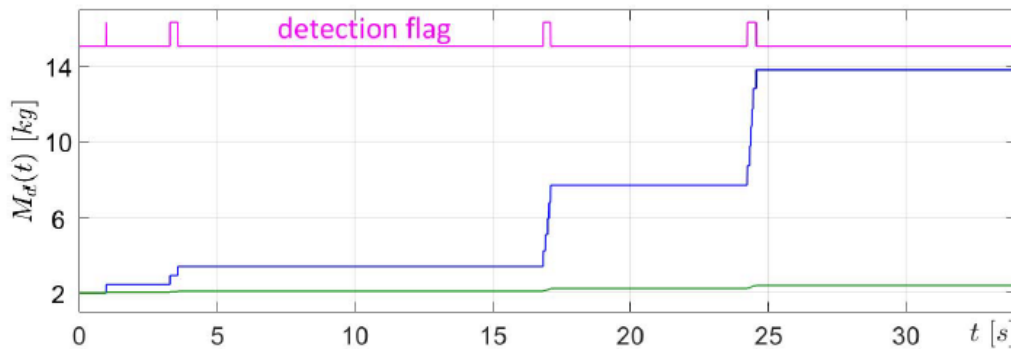


Fig. 2.6: Comparación entre la variación de la inercia con la condición del tanque de energía (curva azul) y la condición de pasividad (curva verde). Extraído de [8].

Entonces, con el fin de mejorar algunos aspectos del algoritmo presentado anteriormente, los autores desarrollaron ciertas mejoras para el mismo en [9].

Una de estas mejoras es el método estadístico propuesto para ajustar el umbral de detección de la inestabilidad. Así mismo, presenta un algoritmo mejorado para la recuperación de la estabilidad del sistema. Este utiliza una estrategia de distribución de energía ponderada con el fin de considerar las traslaciones y rotaciones de forma separada. De esta manera, los parámetros del controlador pueden variar en mayor medida al mismo tiempo que se mantiene la estabilidad. Adicionalmente, se agrega un factor de olvido con el fin de regresar los parámetros a sus valores iniciales de forma progresiva luego de que el sistema haya recuperado la estabilidad.

Este comportamiento se puede apreciar en la Fig. 2.7, donde la curva azul representa la variación de la inercia al detectarse la inestabilidad en el sistema (para la traslación en un solo eje), mientras que la curva magenta representa la variación en el índice de detección de dicha inestabilidad.

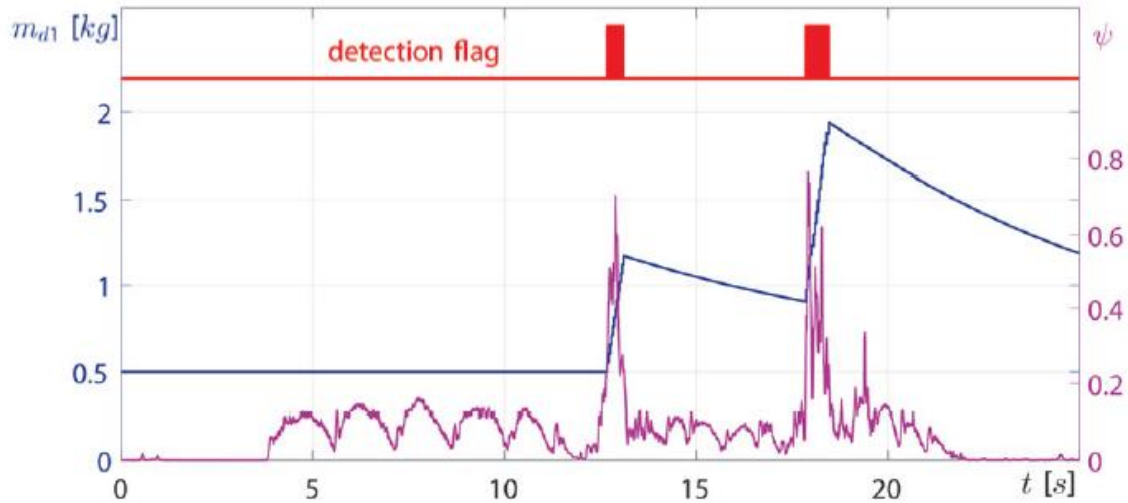


Fig. 2.7: Evolución en el tiempo de la inercia (curva azul) y el índice de detección (curva magenta). Extraído de [9].

Además, en [9] se valida tanto el método para ajustar el umbral como el algoritmo haciendo que dos robots distintos interactúen con 50 usuarios realizando 2 tareas que emulan una operación industrial. Con esto, pretende comprobar que el algoritmo no depende ni del usuario ni del robot.

Por otro lado, en [10] se propone un controlador de admitancia variable, así como un método para determinar la intención directa del humano y otro para determinar la intención indirecta. Con esto, los parámetros del controlador son modificados según la intención del usuario.

El método presentado para la intención directa busca asegurar la estabilidad del robot al establecer límites en la variación de los parámetros, en particular, del amortiguamiento del sistema. Esto resulta útil debido a que, al variar el amortiguamiento, el movimiento en una dirección se vuelve más fácil de realizar, sin embargo, es importante considerar que un amortiguamiento muy pequeño puede llevar a la inestabilidad del sistema.

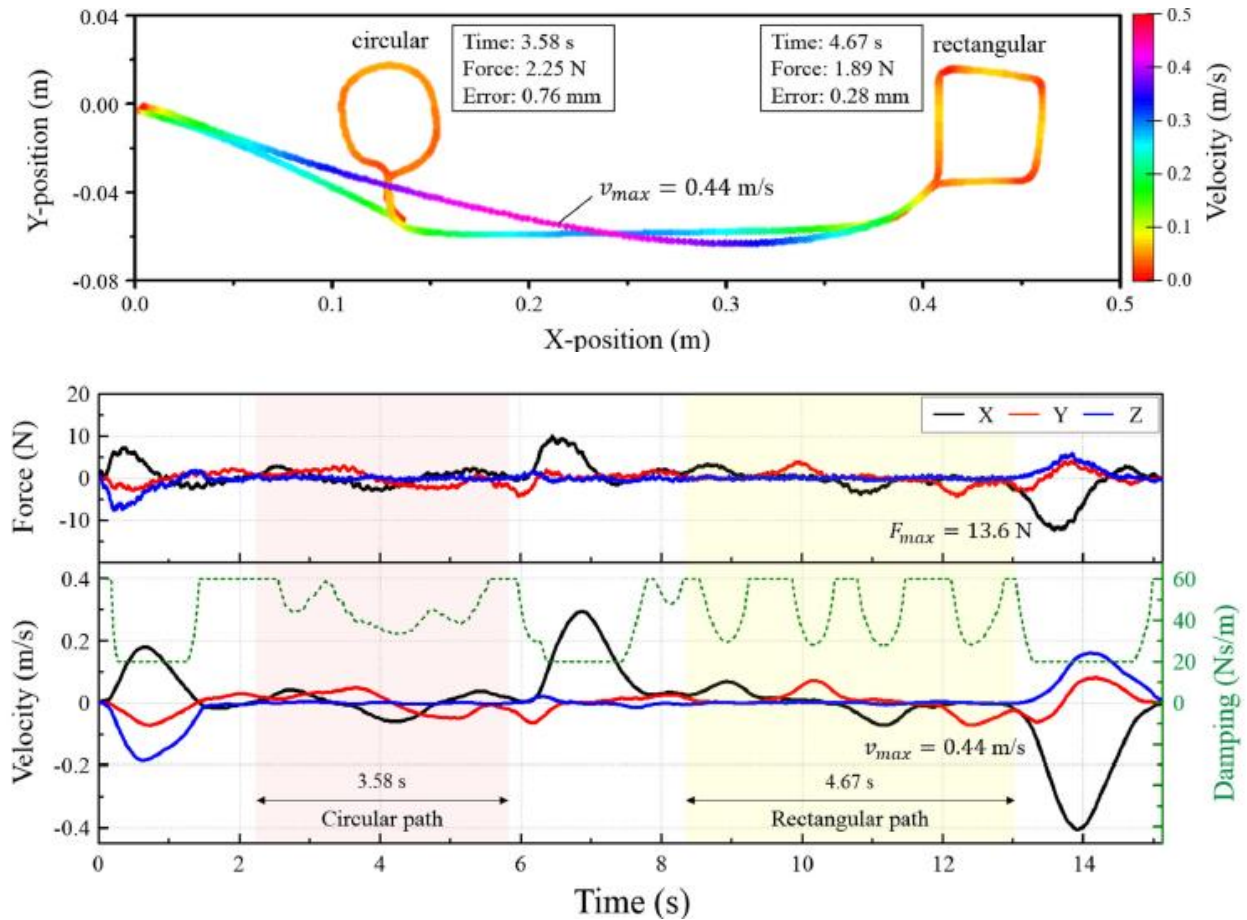


Fig. 2.8: Ejemplo de una tarea completada con el método propuesto. Arriba la trayectoria seguida y abajo la gráfica de velocidades junto con los valores de amortiguamiento del control de admitancia. Extraído de [10].

El método para la intención indirecta pretende predecir hacia donde debe dirigirse el robot en base a la intención del humano, modificando los parámetros del controlador para que el movimiento sea más fácil en esa dirección. Este método solo es implementado para la posición del robot en el espacio, mas no para la rotación.

Se presenta un ejemplo del comportamiento del sistema al implementar los métodos mencionados y el control de admitancia desarrollados en [10] en la Fig. 2.8, donde la primera gráfica presenta la trayectoria del robot y el módulo de la velocidad, mientras que, en la segunda gráfica, se presentan las fuerzas y las velocidades en cada eje junto con los valores de amortiguamiento calculados en base al algoritmo propuesto durante la ejecución de la tarea.

Otro controlador de admitancia interesante es el presentado en [11], donde se utilizó un control de admitancia variable en dos robots UR3. Dado que el objetivo de este controlador era el de permitir la co-manipulación cuando no haya contacto y disminuir la admitancia en caso de

oscilaciones durante el contacto, los sensores de fuerza se encontraban unidos tanto al efector final del robot como a las herramientas utilizadas por los sujetos (Fig. 2.9).

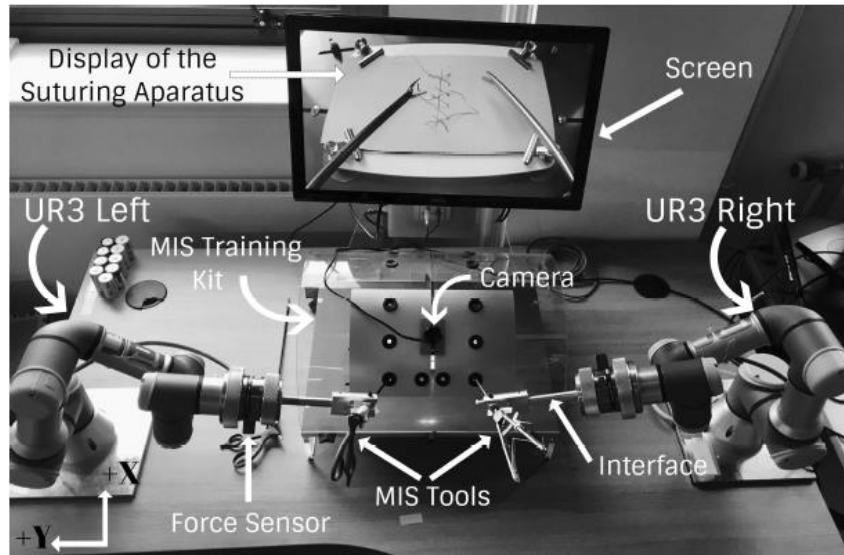


Fig. 2.9: Montaje experimental: Kit de entrenamiento con robots UR3 integrados. Extraído de [11].

El control de admitancia implementado en [11] comprende solamente los parámetros de inercia (M_a) y amortiguamiento (D_a), como se observa en la ecuación 2.7. De esta forma, el comportamiento del sistema queda establecido por los valores de las matrices diagonales M_a y D_a , con lo que la referencia de velocidad (V_{ref}) se puede calcular al conocer la fuerza de interacción (F_s) medida por el sensor.

$$F_s = M_a \dot{V}_{ref} + D_a V_{ref} \quad (2.7)$$

Así mismo, se observa que existe un intercambio entre el rendimiento y la estabilidad del sistema. Esto es aún más crítico al considerar que la inversa de la matriz jacobiana es utilizada para pasar del espacio cartesiano al espacio de articulaciones, por lo que se debe mantener al robot lejos de las singularidades para asegurar un correcto funcionamiento.

Por tal motivo, la robustez del sistema presentado en [11] es mejorada mediante el uso de un observador de estabilidad háptico (HSO por sus siglas en inglés). El HSO utiliza dos índices (I y If), los cuales modifican los valores de amortiguamiento y, subsecuentemente, de masa al detectarse una oscilación. De esta manera, los parámetros son incrementados, disminuyendo la admitancia y aumentando la estabilidad del sistema. No obstante, debido a que esta variación solo ocurre una vez detectada la oscilación, es necesario que el sistema pueda

tolerar, inicialmente, un comportamiento no deseado hasta que se puedan adecuar los parámetros y se asegure la estabilidad del sistema.

Un caso interesante del uso de los controladores de admitancia es aquel donde los parámetros deseados no son impuestos por un humano, sino que son aprendidos. De esta forma, en [12] se propone un sistema de control con un lazo exterior y un lazo interior con el fin de permitirle al robot interactuar con un entorno desconocido mediante el uso de un observador, un controlador de admitancia, el método de control óptimo adaptativo y una red neuronal.

Así, el lazo exterior del sistema propuesto en [12] es utilizado para determinar los parámetros del controlador de admitancia, en un entorno desconocido, basado en el método de control óptimo adaptativo. Dicho método hace uso de una función de coste que define el rendimiento de la interacción en base al seguimiento de la trayectoria de referencia y el torque estimado.

Luego, con el torque estimado por el observador (basado en el enfoque del momento generalizado), es posible generar una referencia de trayectoria a partir del modelo de admitancia adaptativo (en el espacio de articulaciones) basado en la ecuación 2.8. En esta ecuación, M_d , C_d , G_d son los parámetros deseados del controlador de admitancia, $J(q)$ es la matriz jacobiana, τ_{ext} es el torque de interacción estimado por el observador, q_d y q_r son la posición deseada y la referencia de posición, respectivamente, y $k(\cdot)$ es la función de la cinemática directa.

$$M_d J(q) (\ddot{q}_r - \ddot{q}_d) + (M_d \dot{J}(q) + C_d J(q)) (\dot{q}_r - \dot{q}_d) + G_d (k(q_r) - k(q_d)) = -J^{-T}(q) \tau_{ext} \quad (2.8)$$

Por último, el lazo interno del sistema consiste en un controlador basado en una red neuronal (del tipo RBFNN por sus siglas en inglés), la cual es utilizada para garantizar el seguimiento de la trayectoria de referencia.

El esquema de control del sistema completo desarrollado en [12] se presenta en la Fig. 2.10, donde es posible apreciar el lazo exterior (encargado de determinar los parámetros del controlador de admitancia para obtener el comportamiento deseado) y el lazo interior (encargado de seguir la trayectoria generada por el controlador de admitancia).

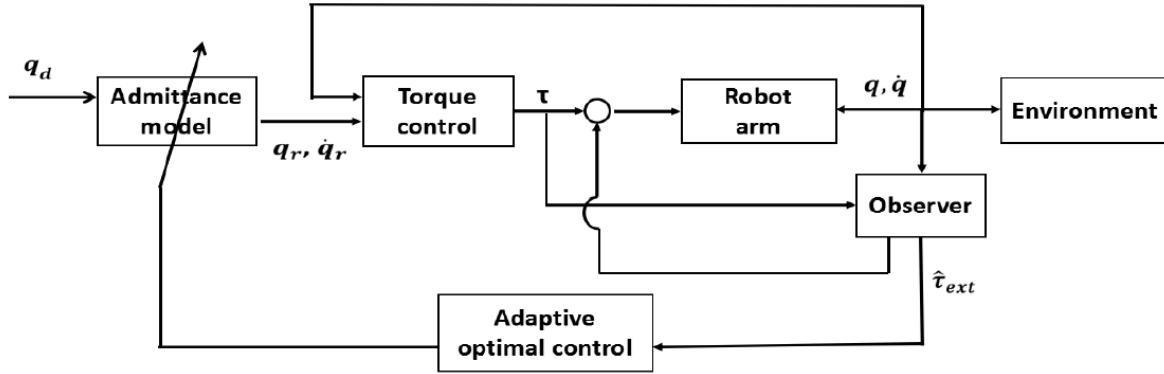


Fig. 2.10: Diagrama de bloques del sistema de control. Extraído de [12].

Por otro lado, el enfoque tomado en [13] resulta interesante debido a que el aprendizaje se realiza con el robot real y no en una simulación del mismo, para lo cual se implementó un algoritmo de seguridad que detiene la sesión de entrenamiento si se incumplen ciertas condiciones de seguridad, con lo que el entrenamiento puede realizarse prácticamente sin supervisión, permitiendo que el robot ejecute múltiples iteraciones sin la necesidad de un operario.

En este caso, se utiliza una estrategia de aprendizaje reforzado mediante el método de Soft-Actor-Critic (SAC) debido a que este método permite el aprendizaje sin la necesidad de un modelo matemático de por medio. De esta forma, el agente SAC varía el parámetro de rigidez del controlador de admitancia (basado en la ecuación 2.9) y, puesto que el valor de la inercia en cada eje es constante, es posible calcular el amortiguamiento a razón de la rigidez establecida. Esto se hace con el fin de reducir la complejidad del controlador y evitar que pueda llegar a tener un comportamiento inestable.

$$F_{ext} = m_d \ddot{x} + b_d \dot{x} + k_d x \quad (2.9)$$

Por otro lado, al incluir información sobre las violaciones a las condiciones de seguridad del sistema implementado en la estrategia de aprendizaje reforzado, es posible acelerar el aprendizaje y disminuir el número de colisiones.

Otro punto interesante es que el agente SAC es capaz de ajustar los parámetros del controlador de tal manera que el comportamiento del sistema se adapte correctamente a cada situación, permitiendo que el robot siga adecuadamente la trayectoria mientras no está en contacto con una superficie a la vez que le permite realizar tareas de alta precisión como insertar un eje en un agujero con una baja tolerancia.

El diagrama de bloques que representa el sistema de control implementado en [13] se presenta en la Fig. 2.11. Es mediante la implementación de este diagrama que el agente SAC puede aprender a modificar los parámetros, tanto del controlador PD (utilizado para seguir la trayectoria deseada) como del controlador de admitancia (utilizado para controlar la interacción con el entorno), permitiendo que el robot complete las tareas satisfactoriamente.

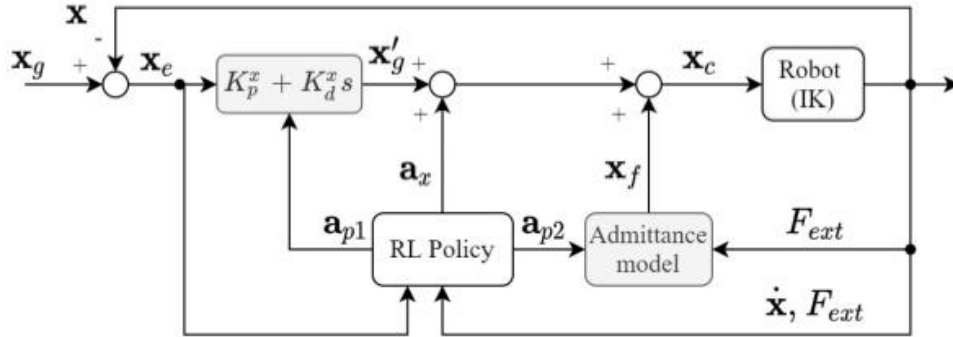


Fig. 2.11: Diagrama de control para el aprendizaje del control de admitancia. Extraído de [13].

CAPÍTULO 3

MODELO DINÁMICO DEL ROBOT UR3e

Esta sección presenta el modelo dinámico de un robot rígido con el fin de mostrar la complejidad del modelo y, al mismo tiempo, comprender las consideraciones tomadas al momento de diseñar el controlador de admitancia utilizado en este trabajo. Así mismo, se explica brevemente el modelo dinámico de un robot elástico y las razones por las cuales no se ha considerado en este caso. Finalmente, se expone el procedimiento seguido para obtener algunos de los parámetros (relevantes para el modelo dinámico) propios del robot UR3e.

3.1. Modelo dinámico de un robot rígido

Cuando se habla de un robot rígido, se asume que los motores (a través de un mecanismo de reducción) están rígidamente acoplados a los eslabones, por lo que no existe un desfase entre el ángulo del motor y el ángulo del eslabón. De esta manera, y considerando los efectos más relevantes que actúan sobre el eslabón, se tiene el modelo dinámico para el espacio de articulaciones presentado en la ecuación 3.1 [14]. En esta ecuación, \ddot{q} , \dot{q} y q , son los vectores de aceleración, velocidad y posición angular, respectivamente. Cabe señalar que la función $sgn(\dot{q})$ representa la función signo de las velocidades de cada articulación, y que $J^T(q)h_e$ representa los torques que experimentan las articulaciones debido a la interacción del robot con el entorno.

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + F_v\dot{q} + F_s sgn(\dot{q}) + g(q) = \tau - J^T(q)h_e \quad (3.1)$$

Donde:

- $M(q)$: Matriz de inercias en el espacio de articulaciones
- $C(q, \dot{q})$: Matriz de centrífugas y Coriolis en el espacio de articulaciones
- $g(q)$: Matriz de gravedad en el espacio de articulaciones
- F_v : Matriz de coeficientes de fricción viscosa en el espacio de articulaciones
- F_s : Matriz de coeficientes de fricción de Coulomb (fricción seca) en el espacio de articulaciones
- τ : Vector de torques de los motores en el espacio de articulaciones
- $J(q)$: Matriz jacobiana en el espacio de articulaciones
- h_e : Fuerzas en el espacio de articulaciones debido a la interacción con el entorno

Debido a que el control de admitancia se implementa normalmente en el espacio de operaciones (también llamado de la tarea), es conveniente utilizar un modelo dinámico del robot que considere las propiedades de este espacio. Por tal motivo, se tiene la ecuación 3.2 [14], la cual expresa la interacción del robot con su entorno en el espacio de operaciones (en este caso en coordenadas cartesianas). En esta ecuación \ddot{x}_e , \dot{x}_e y x_e representan la aceleración, velocidad y posición del efector final en el espacio de operaciones, respectivamente. Así mismo, se tienen las ecuaciones 3.3, 3.4 y 3.5 [14], las cuales presentan la transformación del espacio de articulaciones al espacio de operación para la matriz de inercias, de centrífugas y Coriolis, y de gravedad. De esta forma, es posible utilizar la ecuación 3.2 siempre que J_A no sea singular. Cabe señalar que estas ecuaciones son derivadas para un manipulador no redundante en una configuración no singular y que, además, el efecto, tanto de la fricción seca y como de la viscosa no es considerado en el modelo.

$$M_A(x_e)\ddot{x}_e + C_A(x_e, \dot{x}_e)\dot{x}_e + g_A(x_e) = \gamma_A - h_A \quad (3.2)$$

$$M_A = J_A^{-T} B J_A^{-1} \quad (3.3)$$

$$C_A \dot{x}_e = J_A^{-T} C \dot{q} - M_A \dot{J}_A \dot{q} \quad (3.4)$$

$$g_A = J_A^{-T} g \quad (3.5)$$

Donde:

- M_A : Matriz de inercias en el espacio de operaciones
- C_A : Matriz de centrífugas y Coriolis en el espacio de operaciones
- g_A : Matriz de gravedad en el espacio de operaciones
- J_A : Matriz jacobiana analítica en el espacio de operaciones
- γ_A : Fuerzas en el espacio de operaciones debido a la acción de los motores
- h_A : Fuerzas en el espacio de operaciones debido a la interacción con el entorno

3.2. Modelo dinámico de un robot con articulaciones elásticas

A pesar de que el modelo rígido permite satisfacer una gran cantidad de tareas y predecir el comportamiento de la gran mayoría de robots utilizados actualmente, existen casos en los que, debido a las consideraciones tomadas en dicho modelo y las características del manipulador, el comportamiento no puede ser adecuadamente modelado. En estos casos, es necesario acudir a una formulación más completa de la dinámica del robot, ya sea considerando el efecto de la fricción (seca y/o viscosa) y/o la elasticidad de las articulaciones y/o de los eslabones.

Sin embargo, debido a que los efectos de fricción normalmente son pequeños en los robots comerciales y los eslabones de los mismos son usualmente bastante rígidos, suele ser suficiente utilizar un modelo matemático para manipuladores que considere únicamente la elasticidad de sus articulaciones. Esto se consigue al representar la unión entre el eje del motor y el eslabón correspondiente como un resorte (Fig. 3.1), con lo que se permite que exista un desfase entre la posición angular de estos.

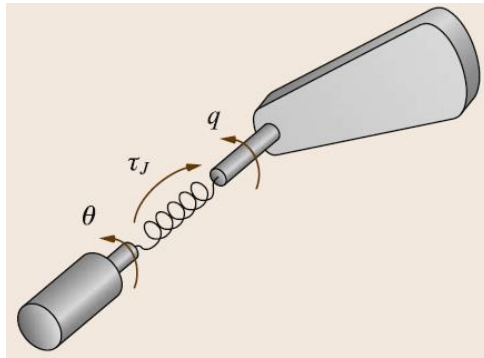


Fig. 3.1: Representación de una articulación elástica. Extraído de [15].

Entonces, tomando esto en consideración, el modelo dinámico del manipulador puede ser representado mediante la expresión matricial presentada en la ecuación 3.6 [15]. En esta ecuación $\ddot{\theta}$, $\dot{\theta}$ y θ representan la aceleración, velocidad y posición angular de los rotores de los motores, respectivamente. Algo a resaltar de este modelo es que los rotores de los motores no actúan directamente sobre los eslabones, sino que estos deforman el resorte mediante el cual se unen a los eslabones, lo cual genera una fuerza (debido a dicha deformación) que es la que actúa sobre dichos eslabones y hace que estos se muevan. Una aclaración importante, es que la parte superior de la ecuación matricial representa la dinámica de los eslabones del manipulador, mientras que la parte inferior represente la dinámica de los motores.

$$\begin{pmatrix} M(q) & S(q) \\ S^T(q) & B \end{pmatrix} \begin{pmatrix} \ddot{q} \\ \ddot{\theta} \end{pmatrix} + \begin{pmatrix} c(q, \dot{q}) + c_1(q, \dot{q}, \dot{\theta}) \\ c_2(q, \dot{q}) \end{pmatrix} + \begin{pmatrix} g(q) + K(q - \theta) \\ K(\theta - q) \end{pmatrix} = \begin{pmatrix} 0 \\ \tau \end{pmatrix} \quad (3.6)$$

Donde:

- $S(q)$: Matriz de acoplamiento de las inercias entre los rotores y los eslabones anteriores en la cadena del robot
- B : Matriz de inercia de los rotores
- $c(q, \dot{q})$: Matriz de centrífugas y Coriolis desde el lado de los eslabones
- $c_1(q, \dot{q}, \dot{\theta})$: Matriz de acoplamiento de centrífugas y Coriolis entre los rotores y los eslabones
- $c_2(q, \dot{q})$: Matriz de centrífugas y Coriolis desde el lado de los motores.
- K : Matriz de rigidez de las articulaciones

Ahora, dado que en este caso el robot interactúa con su entorno y, por lo tanto, existe una fuerza de interacción de que debe ser incluida en el modelo, se formula la ecuación 3.7. Esto se hace siguiendo la formulación del modelo presentado por De Luca [15]. Cabe señalar que, en este caso, el modelo solo se presenta en el espacio de las articulaciones, mas no en el espacio de operaciones.

$$\begin{pmatrix} M(q) & S(q) \\ S^T(q) & B \end{pmatrix} \begin{pmatrix} \ddot{q} \\ \ddot{\theta} \end{pmatrix} + \begin{pmatrix} c(q, \dot{q}) + c_1(q, \dot{q}, \dot{\theta}) \\ c_2(q, \dot{q}) \end{pmatrix} + \begin{pmatrix} g(q) + K(q - \theta) \\ K(\theta - q) \end{pmatrix} = \begin{pmatrix} J^T(q)h_e \\ \tau \end{pmatrix} \quad (3.7)$$

3.3. Modelo dinámico del robot UR3e

Dado que ahora se conocen los modelos normalmente utilizados para representar la dinámica de brazos robóticos, es necesario identificar los parámetros correspondientes para poder obtener el modelo matemático del robot y poder, posteriormente, determinar un algoritmo de control que satisfaga los requerimientos del sistema.

De esta forma, el primer parámetro a identificar es la inercia de los eslabones debido a que es a partir de esta que se derivan las matrices de acoplamiento entre rotores y eslabones, y también las matrices de centrífugas y Coriolis. Una vez se obtenga la matriz de inercia, se podrá determinar el modelo a utilizar en base a la relevancia de la matriz de inercia en el modelo del robot. Esto se debe a que, si la matriz de inercia no representa una magnitud significativa para el modelo, es posible despreciarla, simplificando el modelo en gran medida. Así mismo, puesto que el desfase entre las magnitudes angulares de los rotores y los eslabones depende en cierta forma de la inercia que deba vencer el resorte que los une, una baja inercia significa una baja relevancia para el término de rigidez, permitiendo que se utilice el modelo dinámico para un robot rígido sin problemas.

Cabe señalar que, según Universal Robots, la inercia de los eslabones del robot UR3e son lo suficientemente pequeñas para despreciarlas debido al efecto de la etapa de reducción (el cual se explicará posteriormente). Es por esto que, en los parámetros dinámicos establecidos para

el UR3e, los valores de los tensores de inercia son todos cero a excepción del valor de la inercia en el eje Z para el último eslabón, el cual se presume que representa la inercia del sensor de fuerza acoplado al efector final del robot. La forma de acceder a estos y otros parámetros se explica en el Anexo A.

No obstante, se consideró prudente determinar los valores del tensor de inercia con el fin de comprobar que estos podían ser despreciados sin comprometer el modelo matemático del robot presentado anteriormente. A continuación, se explica brevemente la forma en la que se identificaron los valores de los tensores de inercia con el fin de determinar su relevancia en la dinámica del robot.

En primer lugar, se obtuvo el modelo 3D del robot, el cual contiene las dimensiones exactas de cada eslabón⁶ (Fig. 3.2). Luego, se procedió a determinar las densidades de los eslabones (Tabla 3.1), para lo cual se hizo uso del volumen calculado por el programa Inventor (Fig. 3.3) y las magnitudes de la masa de cada eslabón⁷. Una vez determinada la densidad de cada eslabón, fue posible calcular la inercia de estos haciendo uso de Inventor (Fig. 3.4). Por último, puesto que algunos de los ejes del modelo 3D no coinciden con los ejes utilizados por Universal Robots para determinar los parámetros de Denavit-Hartenberg⁷, estos tuvieron que ser modificados con el fin de que los sistemas de referencia coincidan (Tabla 3.2).

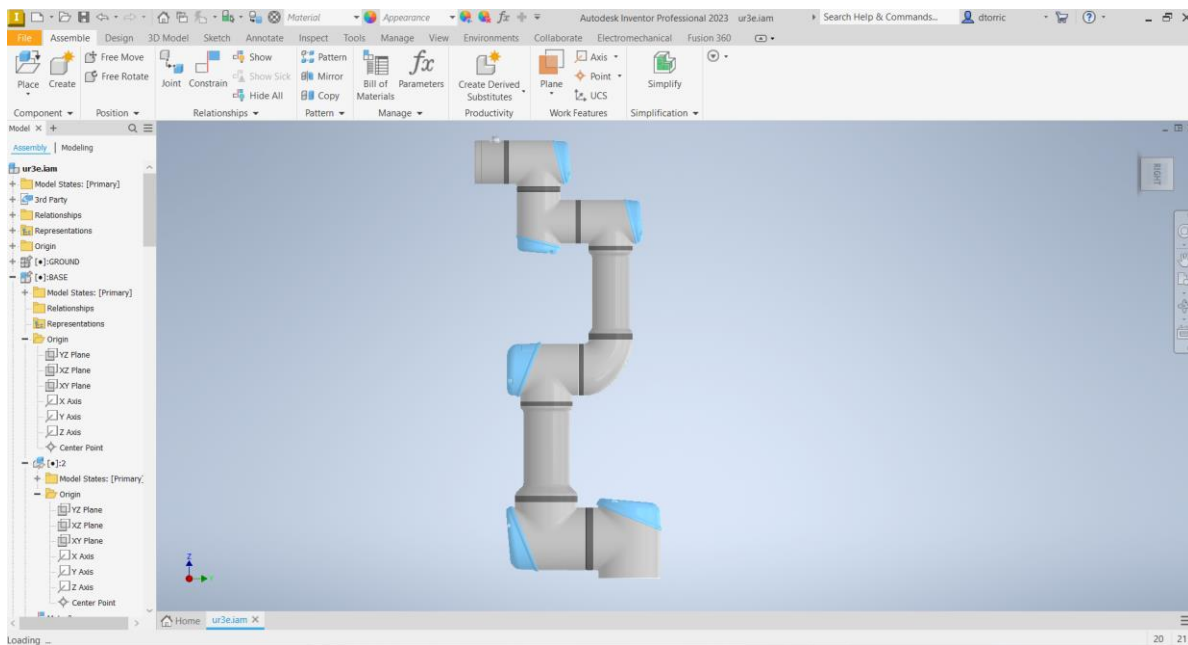


Fig. 3.2: Modelo 3D del robot UR3e en Inventor.

⁶ JT FILE FORMAT FOR UR ROBOTS. Enlace: <https://www.universal-robots.com/articles/ur/application-installation/jt-file-format-for-ur-robots/>

⁷ DH PARAMETERS FOR CALCULATIONS OF KINEMATICS AND DYNAMICS. Enlace: <https://www.universal-robots.com/articles/ur/application-installation/dh-parameters-for-calculations-of-kinematics-and-dynamics/>

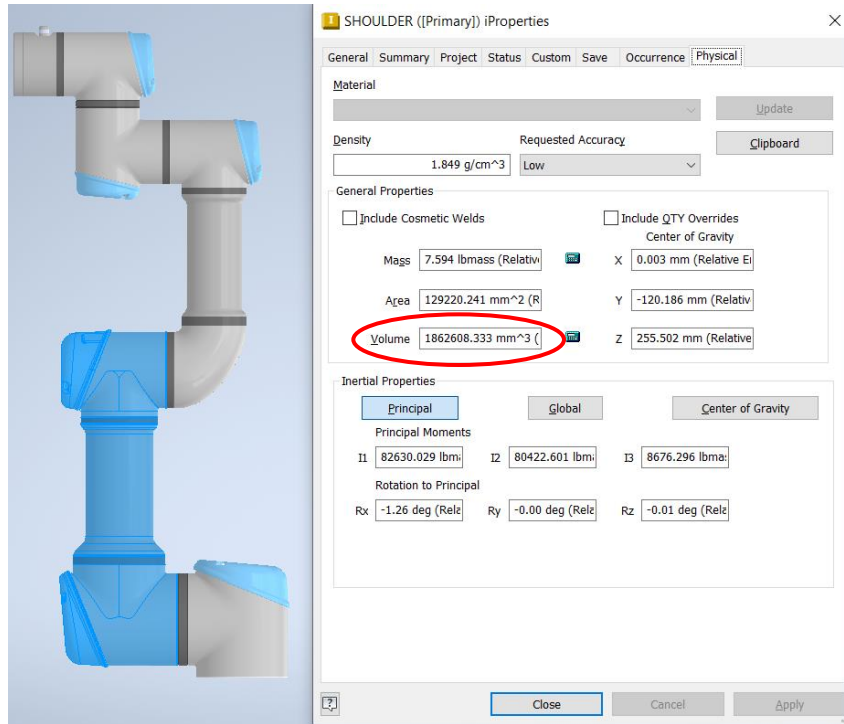


Fig. 3.3: Obtención del volumen de cada eslabón del robot UR3e.

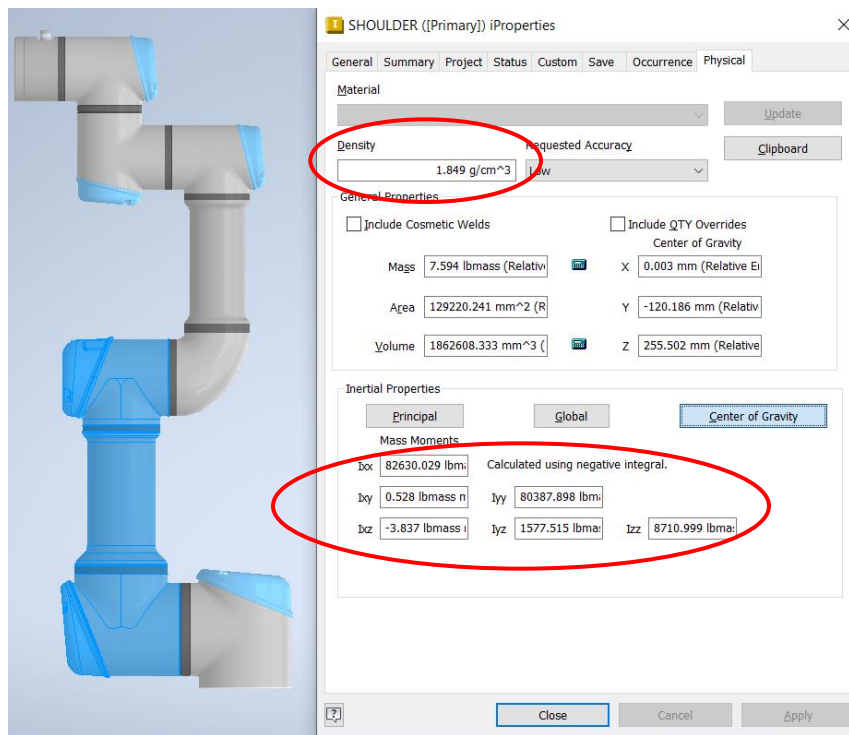


Fig. 3.4: Obtención del tensor de inercias de cada eslabón del robot UR3e.

Tabla 3.1: Estimación de la densidad de los eslabones del robot UR3e.

Eslabón	Masa (kg)	Volumen (mm ³)	Densidad (g/cm ³)
1	1.98	860478.947	2.301044095
2	3.4445	1862608.333	1.849288409
3	1.437	888906.757	1.616592504
4	0.871	374926.477	2.323122141
5	0.805	316876.926	2.540418484
6	0.261	145961.569	1.788141918

Tabla 3.2: Relación entre los ejes del modelo 3D del robot UR3e y los ejes utilizados por Universal Robots.

Inventor	Eslabón 1	Eslabón 2	Eslabón 3	Eslabón 4	Eslabón 5	Eslabón 6
X	X	X	X	X	X	X
Y	Y	Z	Z	Z	Y	Z
Z	Z	-Y	-Y	-Y	Z	-Y

Con todo esto, fue posible determinar los valores de los tensores de inercia para cada uno de los eslabones del robot UR3e, los cuales se presentan a continuación. Cabe señalar que los valores de los tensores de inercia están en $kg \cdot mm^2$.

Tabla 3.2: Tensores de inercia de los eslabones del robot UR3e.

Eslabón	Tensor de inercia del lado del eslabón
1	$I_1 = \begin{pmatrix} 0.003367 & 5.45 \cdot 10^{-9} & -1.27 \cdot 10^{-8} \\ 5.45 \cdot 10^{-9} & 0.002875 & 7.80 \cdot 10^{-5} \\ -1.27 \cdot 10^{-8} & 7.80 \cdot 10^{-5} & 0.002605 \end{pmatrix}$
2	$I_2 = \begin{pmatrix} 0.037514 & 1.74 \cdot 10^{-6} & 2.40 \cdot 10^{-7} \\ 1.74 \cdot 10^{-6} & 0.003955 & -0.00072 \\ 2.40 \cdot 10^{-7} & -0.00072 & 0.036496 \end{pmatrix}$
3	$I_3 = \begin{pmatrix} 0.010977 & -1.82 \cdot 10^{-9} & 1.36 \cdot 10^{-9} \\ -1.82 \cdot 10^{-9} & 0.000975 & 0.000765 \\ 1.36 \cdot 10^{-9} & 0.000765 & 0.010726 \end{pmatrix}$
4	$I_4 = \begin{pmatrix} 0.000959 & 2.2 \cdot 10^{-7} & 1.52 \cdot 10^{-7} \\ 2.2 \cdot 10^{-7} & 0.000735 & -5.90 \cdot 10^{-6} \\ 1.52 \cdot 10^{-7} & -5.90 \cdot 10^{-6} & 0.000671 \end{pmatrix}$
5	$I_5 = \begin{pmatrix} 0.00074 & 3.18 \cdot 10^{-9} & -2.27 \cdot 10^{-9} \\ 3.18 \cdot 10^{-9} & 0.000495 & 6.15 \cdot 10^{-6} \\ -2.27 \cdot 10^{-9} & 6.15 \cdot 10^{-6} & 0.000661 \end{pmatrix}$
6	$I_6 = \begin{pmatrix} 0.000115 & 0 & 0 \\ 0 & 0.000116 & 5.54 \cdot 10^{-8} \\ 0 & 5.54 \cdot 10^{-8} & 0.000132 \end{pmatrix}$

Entonces, con los tensores de inercia calculados para cada eslabón, el siguiente paso es determinar la inercia reflejada del lado del motor, para lo cual se debe hacer uso de la ecuación 3.8 [14]. En esta ecuación, la inercia reflejada es calculada al dividir la inercia del eslabón entre el cuadrado del factor de reducción, por lo que un factor de reducción alto reduce significativamente la inercia percibida por el motor. De esta forma, se obtienen los tensores de la inercia reflejada del lado del motor presentados en la Tabla 3.3. Las inercias de los rotores, los factores de reducción y otros datos adicionales, se presentan en la Tabla 3.4. Es necesario resaltar que las inercias de los motores, los factores de reducción y otros parámetros se obtuvieron directamente de los datos almacenados en los archivos del robot UR3e (Anexo A).

$$I_{eq} = \left(I_m + \frac{I}{k_r^2} \right) \quad (3.8)$$

Donde:

- I_{eq} : Inercia equivalente del lado del motor
- I_m : Inercia del rotor del motor
- I : Tensor de inercia del eslabón
- k_r : Factor de reducción

Tabla 3.3: Tensores de la inercia reflejada de los eslabones.

Eslabón	Inercia reflejada del lado del motor
1	$I_1 = \begin{pmatrix} 3.3 \cdot 10^{-7} & 5.34 \cdot 10^{-13} & -1.25 \cdot 10^{-12} \\ 5.34 \cdot 10^{-13} & 2.82 \cdot 10^{-7} & 7.64 \cdot 10^{-9} \\ -1.25 \cdot 10^{-12} & 7.65 \cdot 10^{-9} & 2.55 \cdot 10^{-7} \end{pmatrix}$
2	$I_2 = \begin{pmatrix} 3.68 \cdot 10^{-6} & 1.71 \cdot 10^{-10} & 2.35 \cdot 10^{-11} \\ 1.71 \cdot 10^{-10} & 3.88 \cdot 10^{-7} & -7.02 \cdot 10^{-8} \\ 2.35 \cdot 10^{-11} & -7.02 \cdot 10^{-8} & 3.58 \cdot 10^{-6} \end{pmatrix}$
3	$I_3 = \begin{pmatrix} 1.08 \cdot 10^{-6} & -1.78 \cdot 10^{-13} & 1.34 \cdot 10^{-13} \\ -1.78 \cdot 10^{-13} & 9.55 \cdot 10^{-8} & 7.5 \cdot 10^{-8} \\ 1.34 \cdot 10^{-13} & 7.5 \cdot 10^{-8} & 1.05 \cdot 10^{-6} \end{pmatrix}$
4	$I_4 = \begin{pmatrix} 9.4 \cdot 10^{-8} & 2.15 \cdot 10^{-11} & 1.49 \cdot 10^{-11} \\ 2.15 \cdot 10^{-11} & 7.21 \cdot 10^{-8} & -5.79 \cdot 10^{-10} \\ 1.49 \cdot 10^{-11} & -5.79 \cdot 10^{-10} & 6.58 \cdot 10^{-8} \end{pmatrix}$
5	$I_5 = \begin{pmatrix} 7.25 \cdot 10^{-8} & 3.12 \cdot 10^{-13} & -2.23 \cdot 10^{-13} \\ 3.12 \cdot 10^{-13} & 4.86 \cdot 10^{-8} & 6.03 \cdot 10^{-10} \\ -2.23 \cdot 10^{-13} & 6.03 \cdot 10^{-10} & 6.48 \cdot 10^{-8} \end{pmatrix}$
6	$I_6 = \begin{pmatrix} 1.13 \cdot 10^{-8} & 0 & 0 \\ 0 & 1.14 \cdot 10^{-8} & 5.43 \cdot 10^{-12} \\ 0 & 5.43 \cdot 10^{-12} & 1.29 \cdot 10^{-8} \end{pmatrix}$

Ahora, a la inercia reflejada del lado del motor (calculada anteriormente) se le debe agregar la inercia de los rotores. Sin embargo, al observar la inercia de los rotores es posible apreciar que esta es la inercia dominante en el modelo, por lo que es posible concluir que la inercia de los eslabones no tiene un efecto significativo en el mismo y, por lo tanto, puede ser despreciada.

Tabla 3.4: Parámetros de los motores del robot UR3e.

Articulación	Inercia del rotor ($kg \cdot mm^2$)	Resistencia del motor (Ω)	Constante de torque	Factor de reducción
1	$6 \cdot 10^{-5}$	0.78	0.10756	101
2	$6 \cdot 10^{-5}$	0.78	0.10756	101
3	$2.08 \cdot 10^{-5}$	1.65	0.07695	101
4	$2.08 \cdot 10^{-5}$	1.65	0.07695	101
5	$2.08 \cdot 10^{-5}$	1.65	0.07695	101
6	$7 \cdot 10^{-6}$	2.394	0.035685	101

Entonces, dado que la inercia de los eslabones no representa una magnitud significativa para el modelo matemático de robot es posible utilizar el modelo dinámico de un robot rígido de forma adecuada. De esta manera, el modelo utilizado como base en el presente trabajo se presenta en la ecuación 3.9. Es este modelo sobre el cual se trabajará con el fin de que el robot tenga el comportamiento deseado.

$$g_A(x_e) = \gamma_A - h_A \quad (3.9)$$

Por otra parte, es necesario mencionar que, a pesar de que el modelo utilizado está en el espacio de operaciones, los comandos para el movimiento del robot se ejecutan en el espacio de articulaciones. Debido a esto, el controlador del UR3e debe hacer uso de uno de los dos métodos normalmente utilizados para calcular dichos comandos: el método de la matriz jacobiana inversa (o de forma más general, pseudo inversa) o el de la matriz jacobiana transpuesta. Cabe resaltar que estos métodos pueden ser implementados tanto para la matriz jacobiana geométrica como para la matriz jacobiana analítica.

En un sistema de control de lazo cerrado, el método de la matriz jacobiana inversa es computacionalmente rápido en comparación al método de la matriz jacobiana transpuesta, pero presenta singularidades cuando el robot adopta ciertas poses, por lo que no es un método que pueda considerarse robusto. Por el contrario, el método de la matriz jacobiana transpuesta tiene una convergencia más lenta al necesitar múltiples operaciones debido a que los valores de la matriz suelen ser pequeños, pero, al no tener la necesidad de invertir la matriz, no presenta singularidades y por lo tanto es posible considerarlo como un método robusto.

Se presume que el controlador del UR3e hace uso del método de la matriz jacobiana transpuesta cuando se utiliza un comando de velocidad, mientras que para un comando de posición hace uso del método de la matriz jacobiana inversa [16]. De esta forma, se obtiene un movimiento que no presenta singularidades y que no depende fuertemente de la pose del robot al utilizar un comando de velocidad, lo cual permite obtener un mejor resultado al implementar el algoritmo de control para obtener el comportamiento deseado. Por tal motivo, en los experimentos realizados se utilizan únicamente comandos de velocidad para desplazar el efector final del robot.

CAPÍTULO 4

DISEÑO DE LOS CONTROLADORES DE ADMITANCIA

Este capítulo presenta la deducción de los algoritmos de control utilizados basándose en el modelo matemático obtenido en el capítulo anterior y el modelo de admitancia presentado en el capítulo 1. De esta manera, se desarrollan dos algoritmos de control: uno que no modifica la inercia aparente del sistema y otro que si lo hace.

4.1. Consideraciones preliminares

Debido a la forma en la que el comando *speedl* del UR3e funciona, es imperativo especificar el vector de velocidades (tanto de desplazamiento como de rotación) y la aceleración que tendrá el efector final para alcanzar dichas velocidades. Sin embargo, dado que esta aceleración no es un vector sino una magnitud escalar, es la misma para todos los ejes, lo cual afecta la capacidad del robot para seguir la referencia de velocidad, evitando que se obtenga el comportamiento deseado al no poder realizar los movimientos necesarios a tiempo. Por lo tanto, es necesario elegir una aceleración lo suficientemente grande para que la referencia de velocidad sea alcanzada rápidamente y sea posible obtener el comportamiento deseado.

Por otro lado, tanto el sensor de fuerza como el sensor de velocidad presentan ruido, siendo el ruido de este último el más significativo. Esto se puede deber a que la velocidad se calcula en base a la variación de la posición (la cual también presenta ruido) y no es obtenida desde la lectura de un sensor destinado para este propósito. Debido a esta limitación, no es factible implementar un control adecuado en base a la lectura de la velocidad actual del robot, puesto que se tendría que utilizar un filtro, lo cual reduciría el ancho de banda, retrasaría la señal y afectaría negativamente tanto al comportamiento como a la estabilidad del sistema.

Un aspecto importante que señalar es que el controlador de velocidad del robot UR3e compensa efectivamente el efecto de la gravedad, por lo que la dinámica deseada puede ser impuesta en el sistema directamente y sin la necesidad de compensar otros efectos, lo cual simplifica considerablemente la implementación del controlador. Esto se puede entender mejor al recordar la ecuación 3.9, en la cual, si se compensa la gravedad en dicha ecuación, se obtiene que las fuerzas de interacción corresponden directamente a las fuerzas de los motores, lo que significa que, al utilizar el modelo de admitancia presentado en el capítulo 1, basta con calcular la velocidad que debe tener el efector final para que el robot presente el comportamiento deseado. En la ecuación 4.1 es posible observar esta relación en el espacio de Laplace, mientras que en la ecuación 4.2 se puede apreciar la ecuación resultante al pasar al espacio del tiempo e incluir las respectivas referencias para el movimiento. Es este último el que será utilizado para diseñar los controladores.

$$h_A(s) = \gamma_A(s) = \dot{X}(s)Z(s) = \dot{X} \left(ms + b + \frac{k}{s} \right) \quad (4.1)$$

$$h_A = \gamma_A = m_d(\ddot{x} - \ddot{x}_d) + b_d(\dot{x} - \dot{x}_d) + k_d(x - x_d) \quad (4.2)$$

4.2. Controlador de admitancia sin modificación de la inercia aparente

El hecho de definir la inercia aparente igual a la inercia del robot (La cual se desprecia, dando lugar a $m_d = m_r = 0$), significa que no es necesario calcular la aceleración que debe tener el efector final para presentar el comportamiento deseado, con lo que la velocidad de referencia para el controlador del UR3e (\dot{x}_{ref}) puede ser calculada fácilmente.

De este modo, se tiene el modelo utilizado (en el espacio del tiempo) para este caso en la ecuación 4.3 (obtenida al despejar la velocidad de la ecuación 4.2 y definir $\dot{x}_d = 0$), donde el objetivo es calcular la velocidad que debe tener el efector final de acuerdo con el modelo de admitancia con el propósito de que esta sea la referencia de velocidad para el controlador del UR3e. Cabe señalar que dicha ecuación se debe implementar para cada uno de los ejes en los que se desea obtener este comportamiento, pudiendo elegir parámetros distintos para cada uno. Adicionalmente, en la Fig. 4.1 se presenta el diagrama de bloques resultante de este controlador.

$$\dot{x}_{ref} = \frac{h_A}{b_d} - \frac{k_d}{b_d}(x_d - x) \quad (4.3)$$

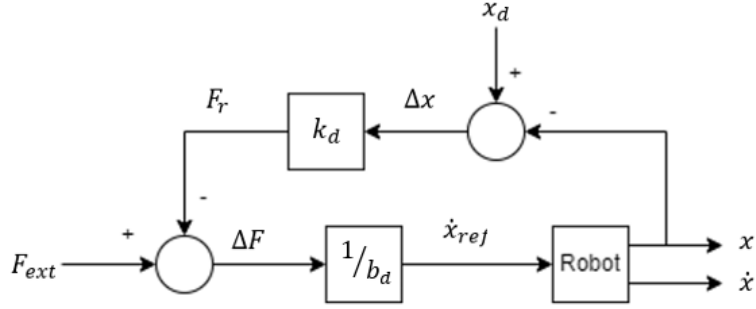


Fig. 4.1: Diagrama de bloques del controlador sin modificación de la inercia aparente. F_r es la fuerza del resorte, mientras que F_{ext} es la fuerza de interacción.

Como se puede apreciar, el comportamiento del robot es dictado por el amortiguamiento (b_d) y la rigidez (k_d) deseados. Por otra parte, es importante notar que, al actuar una fuerza externa (h_A) sobre el efector final (donde se encuentra el sensor de fuerza), el robot se desplaza en dirección de dicha fuerza hasta que la fuerza ejercida por el resorte es igual a la fuerza externa. Con lo que, una vez que se deja de ejercer una fuerza sobre el efector final, el robot regresa a la posición deseada por la acción del resorte. Así mismo, la acción del amortiguador regula la velocidad con la que el robot se desplaza debido a los efectos antes mencionados.

4.3. Controlador de admitancia con modificación de la inercia aparente

Debido a que con este controlador se pretende modificar la inercia aparente del sistema, es necesario calcular la velocidad actual del efector final considerando el efecto de esta inercia. De esta forma, al integrar la ecuación 4.4 (obtenida al despejar la aceleración de la ecuación 4.2 y definir $\ddot{x}_d = 0$ y $\dot{x}_d = 0$), se obtiene la ecuación 4.5. Con el fin de mantener la claridad en las ecuaciones, se define $V = \dot{x}$. Ahora, si bien se podría obtener la velocidad de la lectura del sensor, este presenta demasiado ruido, provocando un efecto negativo en el sistema.

$$\dot{V} = \frac{h_A}{m_d} - b_d \frac{V}{m_d} - k_d \frac{(x_d - x)}{m_d} \quad (4.4)$$

$$\int_0^V dV = \int_0^t \left(\frac{h_A}{m_d} - b_d \frac{V}{m_d} - k_d \frac{(x_d - x)}{m_d} \right) dt \quad (4.5)$$

Sin embargo, debido a que la función de la fuerza externa no es conocida, y que el controlador será implementado de forma discreta, una solución aproximada de la ecuación 4.5 se puede obtener al integrar dicha ecuación de forma discreta. Así, la velocidad considerando el efecto de la inercia se puede calcular haciendo uso de la ecuación 4.6. Adicionalmente, para mantener la ecuación breve, se tienen las ecuaciones 4.7, 4.8 y 4.9, con las cuales se obtiene la ecuación 4.10.

$$V_{est} = \frac{1}{m_d} \int_0^t h_A dt - \frac{b_d}{m_d} \int_0^t V dt - \frac{k_d}{m_d} \int_0^t (x_d - x) dt \quad (4.6)$$

$$\int_0^t h_A dt \approx h_{A_{int}}(k) = h_{A_{int}}(k-1) + \left(\frac{h_A(k) + h_A(k-1)}{2} \right) T \quad (4.7)$$

$$\int_0^t V dt \approx V_{int}(k) = V_{int}(k-1) + \left(\frac{V_{ref}(k-1) + V_{ref}(k-2)}{2} \right) T \quad (4.8)$$

$$\int_0^t (x_d - x) dt \approx \Delta x_{int}(k) = \Delta x_{int}(k-1) + \left(\frac{(x_d - x(k)) + (x_d - x(k-1))}{2} \right) T \quad (4.9)$$

$$V_{est}(k) = \frac{1}{m_d} h_{A_{int}}(k) - \frac{b_d}{m_d} V_{int}(k) - \frac{k_d}{m_d} \Delta x_{int}(k) \quad (4.10)$$

Donde:

- $h_{A_{int}}$: Valor acumulado de la fuerza externa
- V_{int} : Valor acumulado de la velocidad
- Δx_{int} : Valor acumulado de la diferencia de posición
- V_{est} : Velocidad estimada del efector final
- T : Periodo de muestreo

Haciendo uso de esta técnica es posible obtener una solución aproximada para la ecuación 4.6, con lo que es posible obtener el valor estimado de la velocidad del efector final. No obstante, dado que en este caso el efecto de la aceleración juega un papel importante, y la velocidad de referencia utilizada para calcular la velocidad estimada se encuentra atrasada un periodo de muestreo, es necesario incluir la aceleración en el cálculo de la velocidad de referencia para el controlador del UR3e con el fin de obtener una respuesta satisfactoria. Esto se logra al utilizar la ecuación 4.11 para calcular la aceleración de referencia (a_{ref}) para cada eje, y la ecuación 4.12 obtener la velocidad de referencia del respectivo eje para luego pasar el vector de velocidades al controlador de velocidad del UR3e. De esta manera, haciendo uso de un valor lo suficientemente grande de aceleración para el controlador del UR3e y utilizando la ecuación 4.12, es posible obtener el comportamiento deseado para cada uno de los ejes donde se implemente el controlador. Por último, en la Fig. 4.2 se presenta el diagrama de bloques resultante de este.

$$a_{ref}(k) = \frac{h_A(k)}{m_d} - b_d \frac{V_{est}(k)}{m_d} - k_d \frac{(x_d - x(k))}{m_d} \quad (4.11)$$

$$V_{ref}(k) = V_{est}(k) + a_{ref}(k) \cdot T \quad (4.12)$$

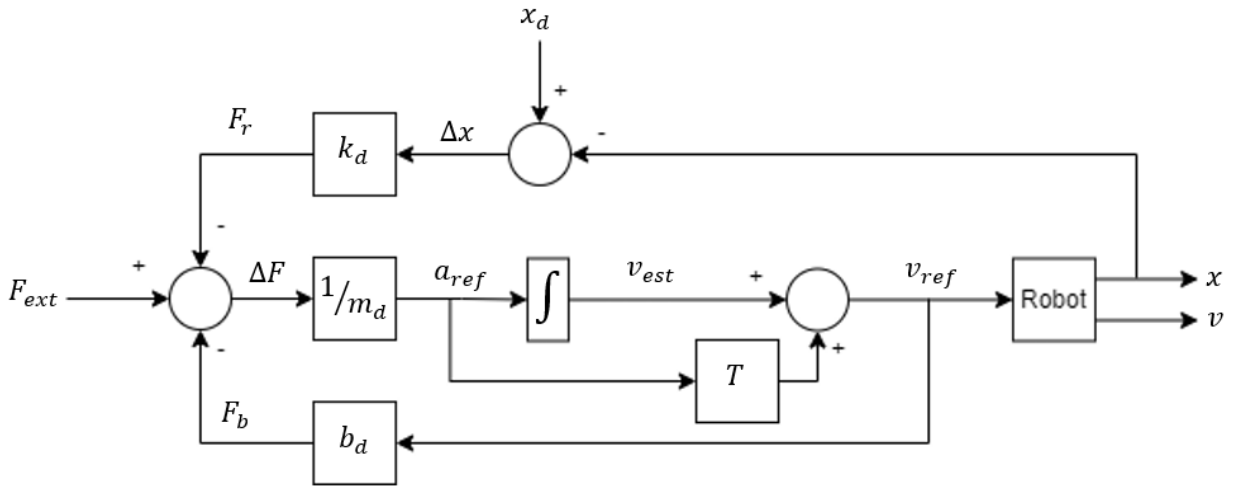


Fig. 4.2: Diagrama de bloques del controlador con modificación de la inercia aparente. F_r es la fuerza del resorte, F_b es la fuerza del amortiguador, F_{ext} es la fuerza de interacción, y T es el periodo de muestreo.

CAPÍTULO 5

EXPERIMENTOS Y RESULTADOS

Este capítulo comprende la plataforma experimental, los experimentos realizados en el robot UR3e, y los resultados obtenidos de estos. En este caso se presentan los resultados más relevantes, algunos de los cuales son utilizados para mostrar los límites del robot UR3e y como algunas de estas limitaciones pueden ser superadas mediante el uso de los controladores diseñados en el capítulo anterior.

5.1. Plataforma experimental y preparación de los experimentos

La plataforma utilizada en este trabajo comprende a un robot UR3e de la empresa Universal Robots, la caja de control del UR3e, un PC, y un sensor de fuerza integrado en el efector final del robot. Las características de cada uno de los componentes y los programas utilizados se encuentran detalladas en [17].

De esta forma, el robot envía los datos hacia el PC mediante un cable ethernet haciendo uso del protocolo TCP. Luego, el PC recibe los datos y los condiciona para que puedan ser almacenados en un documento en formato "csv" gracias a un programa desarrollado en Python (el cual se encuentra explicado paso a paso en [17]). Una vez se han recolectado todos los datos del experimento, estos son cargados en MATLAB para ser comparados con los resultados de las simulaciones, del modelo de admitancia correspondiente, realizadas en Simulink. A manera de resumen, en la Fig. 5.1 se presenta un esquema de la plataforma experimental. El modelo utilizado en Simulink recibe como entrada la fuerza medida por el sensor situado en el efector final del UR3e, y da como resultado las variables de movimiento (posición, velocidad, aceleración) del efector final en el eje correspondiente.

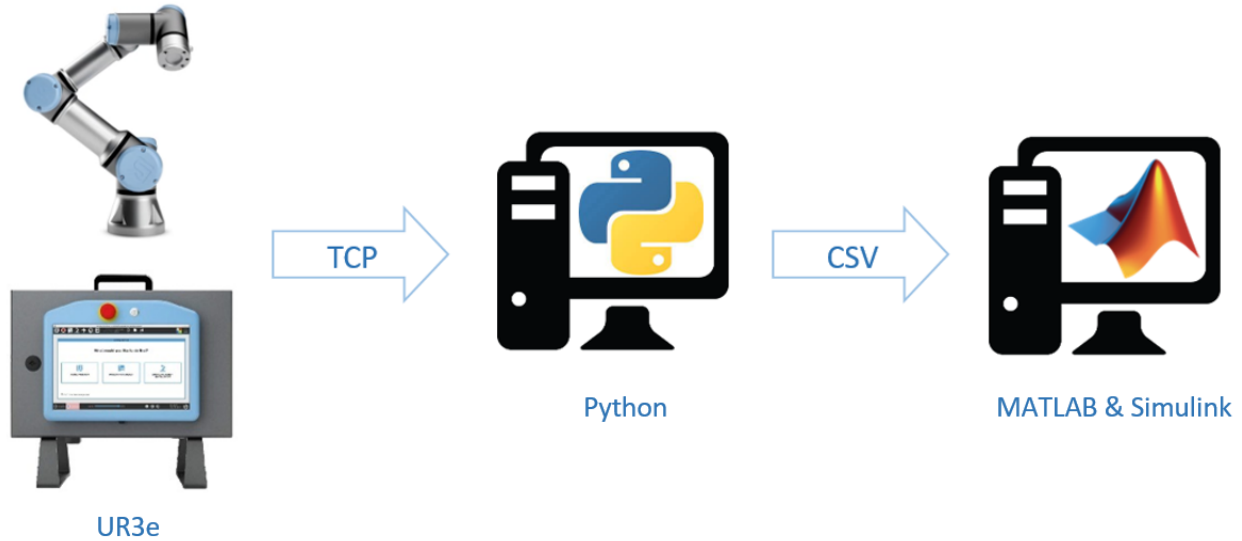


Fig. 5.1: Esquema de la plataforma experimental utilizada.

Por otro lado, se prepararon los experimentos con el fin de verificar que los controladores de admitancia reproducían correctamente el comportamiento deseado. Además, también se plantearon algunos experimentos con el propósito de observar el efecto de utilizar la directriz número 5 presentada en el capítulo 2 (en la Tabla 2.1), para lo cual se parte de la ecuación 5.1 (en el espacio de Laplace) [3], y se obtiene la ecuación 5.2 con el fin de implementarla en el espacio de tiempo discreto.

$$Y_v(s) = (ska + 1) \cdot Y'_v(s) \quad (5.1)$$

$$Y_v(k) = (ka + 1) \cdot Y'_v(k) - ka \cdot Y'_v(k - 1) \quad (5.2)$$

Donde:

- Y'_v : Dinámica deseada del sistema
- Y_v : Dinámica modificada
- ka : Ganancia de la acción “feed-forward”

De esta manera, se presentan primero los parámetros de los experimentos para el controlador sin modificación de la inercia aparente (abreviado como SMIA) en las Tablas 5.1 y 5.2, donde “kx”, “ky” y “kz” (en N/m), definen los valores de rigidez, mientras que “bx”, “by” y “bz” (en N^s/m), definen la amortiguación para cada eje.

Es importante señalar que en estos experimentos se utilizó una aceleración de 0.5 m/s^2 , por lo que en ciertos casos (normalmente cuando el amortiguamiento deseado era pequeño) el robot no fue capaz de realizar los movimientos en un tiempo adecuado y, por lo tanto, no consiguió mostrar el comportamiento esperado. No obstante, esto resulta útil para observar claramente el efecto que tiene el utilizar la ecuación 5.2 en el control.

Tabla 5.1: Experimentos del controlador SMIA sin la acción "feed-forward".

Experimento	k_x	k_y	k_z	b_x	b_y	b_z	k_a
1_1 SMIA	20	20	20	50	50	50	0
1_2 SMIA	50	50	50	50	50	50	0
1_3 SMIA	100	100	100	50	50	50	0
1_4 SMIA	20	20	20	20	20	20	0

Tabla 5.2: Experimentos del controlador SMIA con la acción "feed-forward".

Experimento	k_x	k_y	k_z	b_x	b_y	b_z	k_a
2_1 SMIA	20	20	20	50	50	50	1
2_2 SMIA	50	50	50	50	50	50	1
2_3 SMIA	100	100	100	50	50	50	1
2_4 SMIA	20	20	20	20	20	20	1

Luego, se presentan las Tablas 5.3, 5.4, 5.5 y 5.6 (donde "m" define la masa del sistema, la cual se traduce en la inercia aparente que tendrá el efector final del robot), en las cuales se definen los parámetros utilizados para cada experimento del controlador con modificación de la inercia aparente (abreviado como CMIA). En estos experimentos se utilizó una aceleración de 3 m/s^2 , por lo que el robot fue capaz de mostrar el comportamiento esperado.

Entonces, a pesar de que para aplicaciones con robots un comportamiento oscilatorio no es deseado, se definieron un par de experimentos considerando este caso con el fin de evaluar la capacidad del robot de reproducir el modelo propuesto e identificar si el sistema tiende hacia la inestabilidad o hacia el reposo en un caso extremo como este. Adicionalmente, se planteó un experimento para verificar si el robot era capaz de seguir correctamente la referencia cuando solo se tiene la amortiguación deseada para el sistema.

Tabla 5.3: Experimentos sistema oscilatorio y no oscilatorio (con la acción "feed-forward").

Experimento	k_x	k_y	k_z	b_x	b_y	b_z	m	k_a
Oscilatorio	50	50	50	0	0	0	5	1
Osci_XYZ	50	50	50	0	0	0	5	1
No_Osci	0	0	0	20	20	20	5	1

Así mismo, al igual que en el caso del SMIA, se plantearon algunos experimentos para comprobar la eficacia de la acción "feed-forward".

Tabla 5.3: Experimentos del controlador CMIA con y sin la acción "feed-forward".

Experimento	k_x	k_y	k_z	b_x	b_y	b_z	m	k_a
1_1 CMIA	50	50	50	20	20	20	5	0
1_2 CMIA	100	100	100	20	20	20	5	0
1_3 CMIA	20	20	20	50	50	50	5	0
2_1 CMIA	50	50	50	20	20	20	5	1
2_2 CMIA	100	100	100	20	20	20	5	1
2_3 CMIA	20	20	20	50	50	50	5	1

Luego, dado que se busca validar que el controlador puede imponer el comportamiento deseado en el sistema, se realizaron múltiples experimentos considerando un sistema críticamente amortiguado, puesto que este punto marca la línea entre un comportamiento oscilatorio y un comportamiento no oscilatorio. De este modo, se busca probar al sistema en este límite con el fin de determinar que tan fiable es el controlador y que tan predecible es el modelo. Cabe señalar que para todos los experimentos se hizo uso de la acción "feed-forward".

Tabla 5.4: Experimentos del controlador CMIA para un sistema críticamente amortiguado

Experimento	k_x	k_y	k_z	b_x	b_y	b_z	m	k_a
1 CMIA_Crit	10	10	10	40	40	40	40	1
2 CMIA_Crit	20	20	20	40	40	40	20	1
3 CMIA_Crit	40	40	40	40	40	40	10	1
4 CMIA_Crit	40	40	40	20	20	20	2.5	1
5 CMIA_Crit	25	25	25	10	10	10	1	1

Por último, se decidió que era necesario evaluar el comportamiento del sistema con diferentes parámetros en cada eje con el fin de descartar la existencia de un acoplamiento entre estos que pudiera resultar perjudicial. De esta manera, se realizaron dos experimentos, uno con la acción "feed-forward" y otro sin esta con el fin de observar su efecto sobre el sistema.

Tabla 5.6: Experimentos del controlador CMIA con diferentes parámetros para cada eje, con y sin acción "feed-forward"

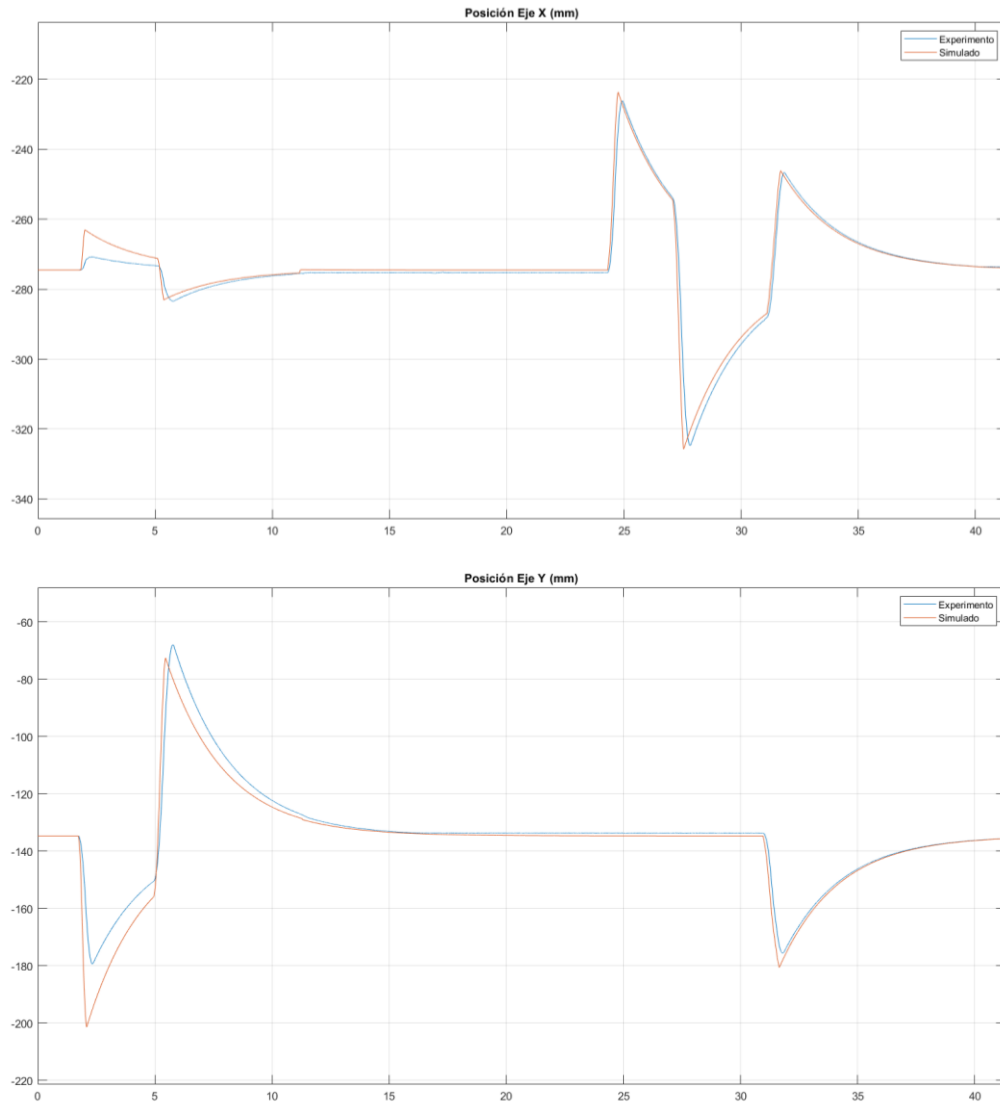
Experimento	k_x	k_y	k_z	b_x	b_y	b_z	m	k_a
1 CMIA_Dif_P	50	20	10	20	50	10	10	1
2 CMIA_Dif_P	50	20	10	20	50	10	10	0

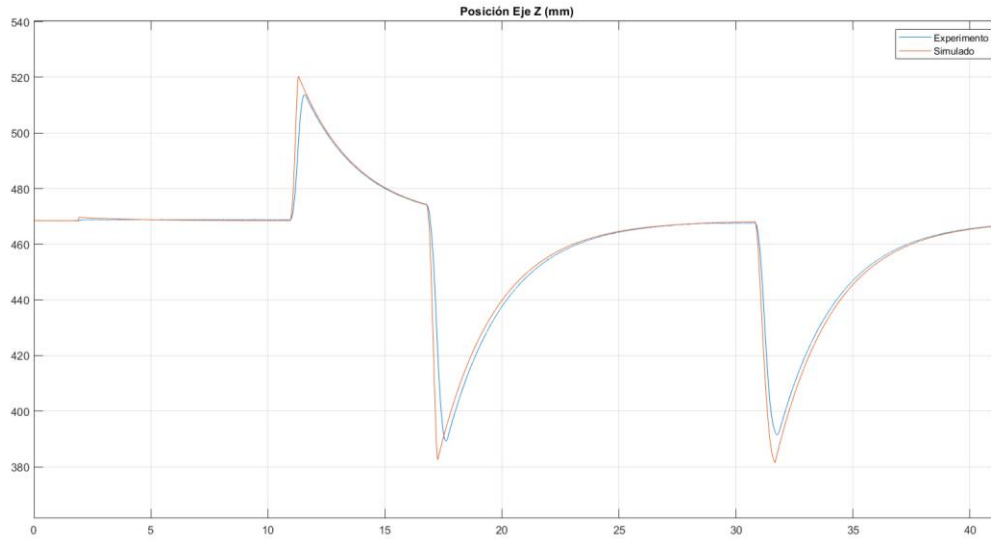
5.2. Resultados del controlador de admitancia sin modificación de la inercia aparente

A continuación, se muestran los resultados de los experimentos realizados con este controlador, junto con una breve explicación de estos.

1_1 SMIA

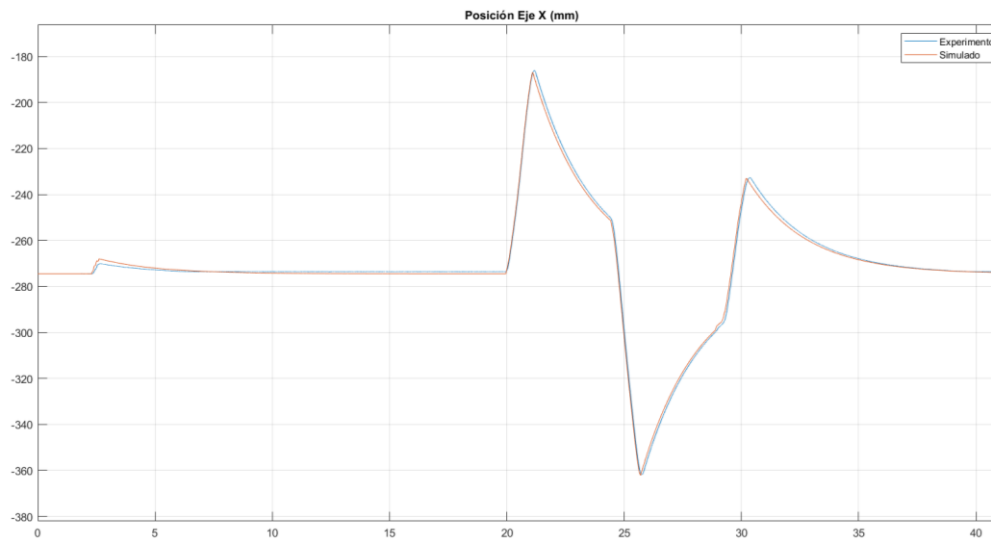
En este experimento es posible apreciar como el robot ejecuta el comportamiento deseado con una ligera desviación debido a que la aceleración utilizada no permite que el robot alcance la referencia de velocidad (obtenida del modelo) a tiempo. Esto es más evidente cuando surgen cambios bruscos en el movimiento.

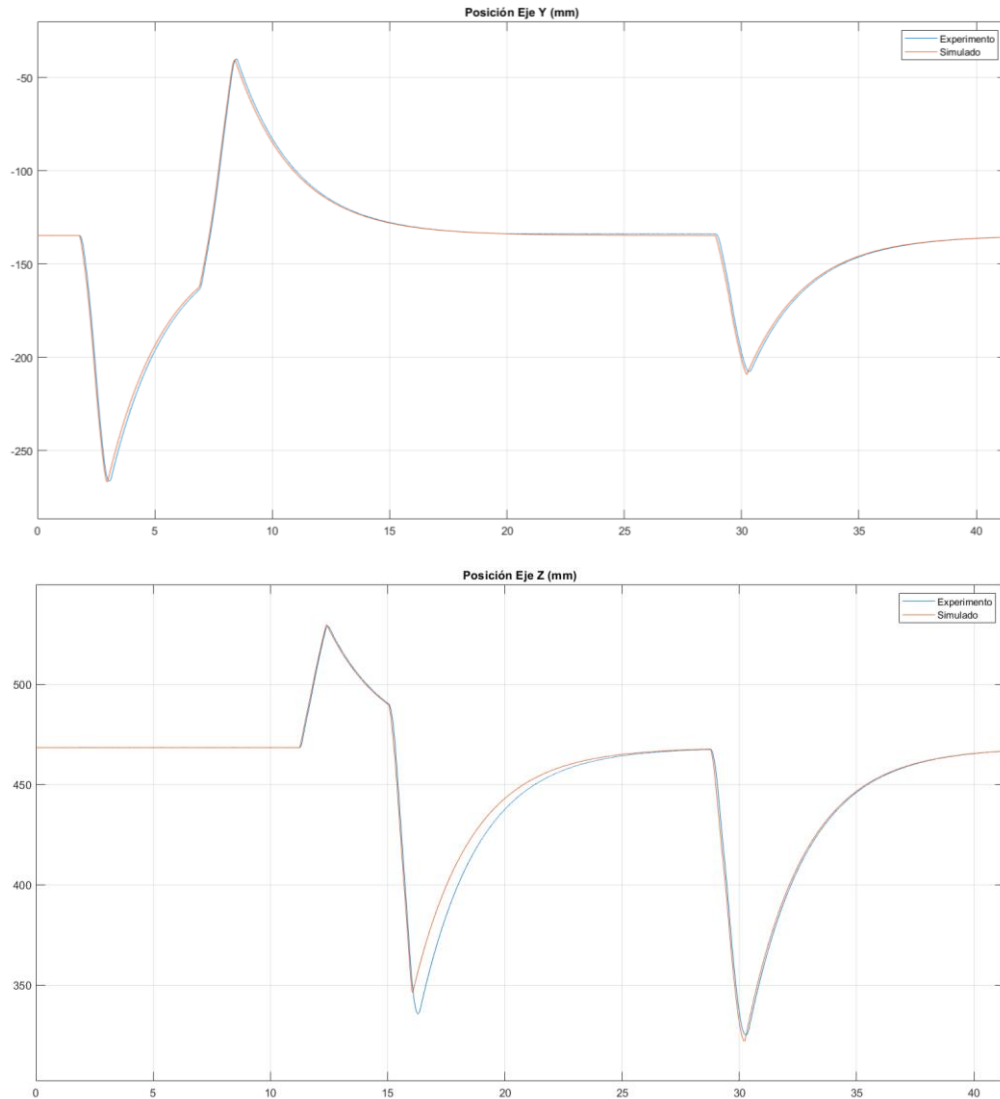




2_1 SMIA

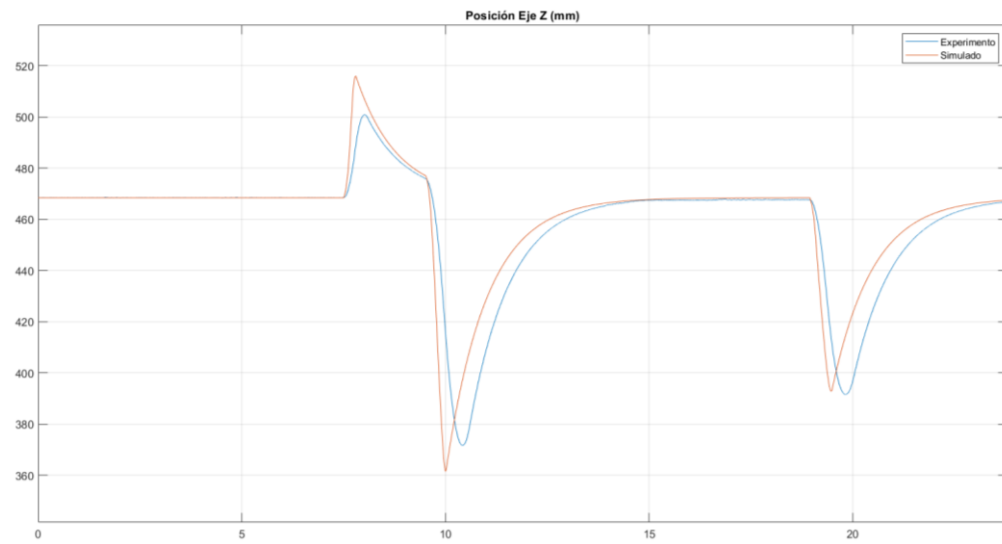
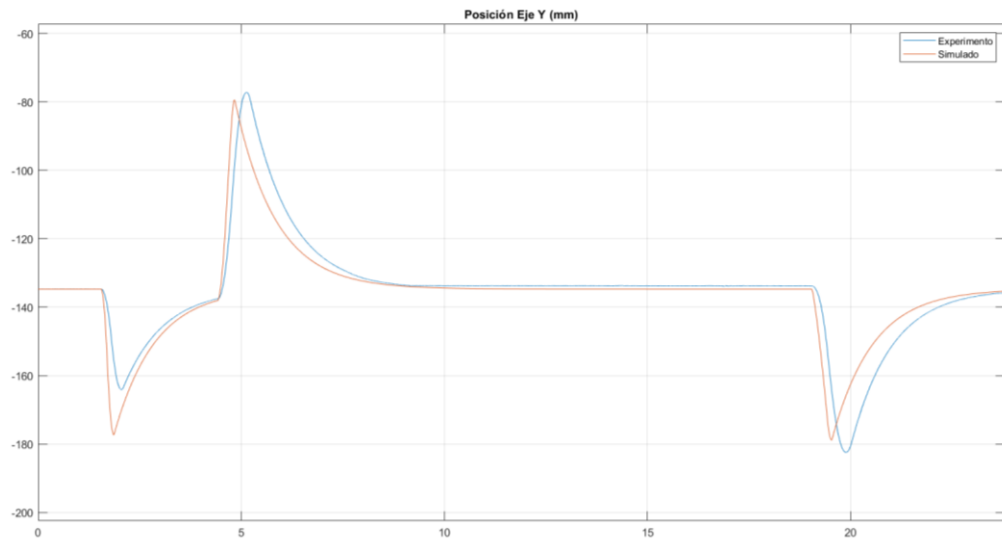
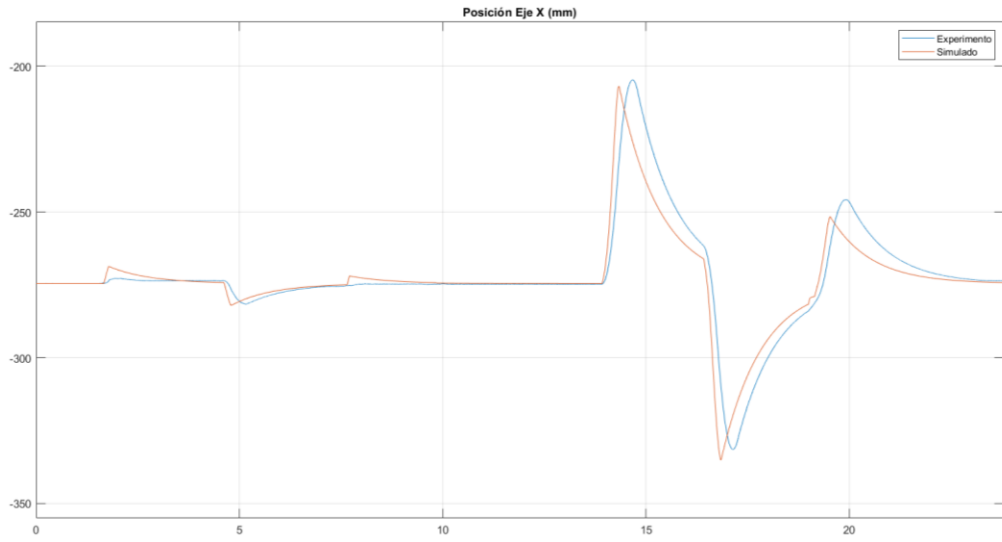
Ahora, haciendo uso de la acción “feed-forward” es posible observar una clara mejora en la respuesta del sistema. No obstante, aun existen ocasiones en las que el robot no es capaz de alcanzar la velocidad de referencia lo suficientemente rápido, alejándose de la respuesta ideal, tal y como se puede observar en la gráfica de la posición en el eje Z.





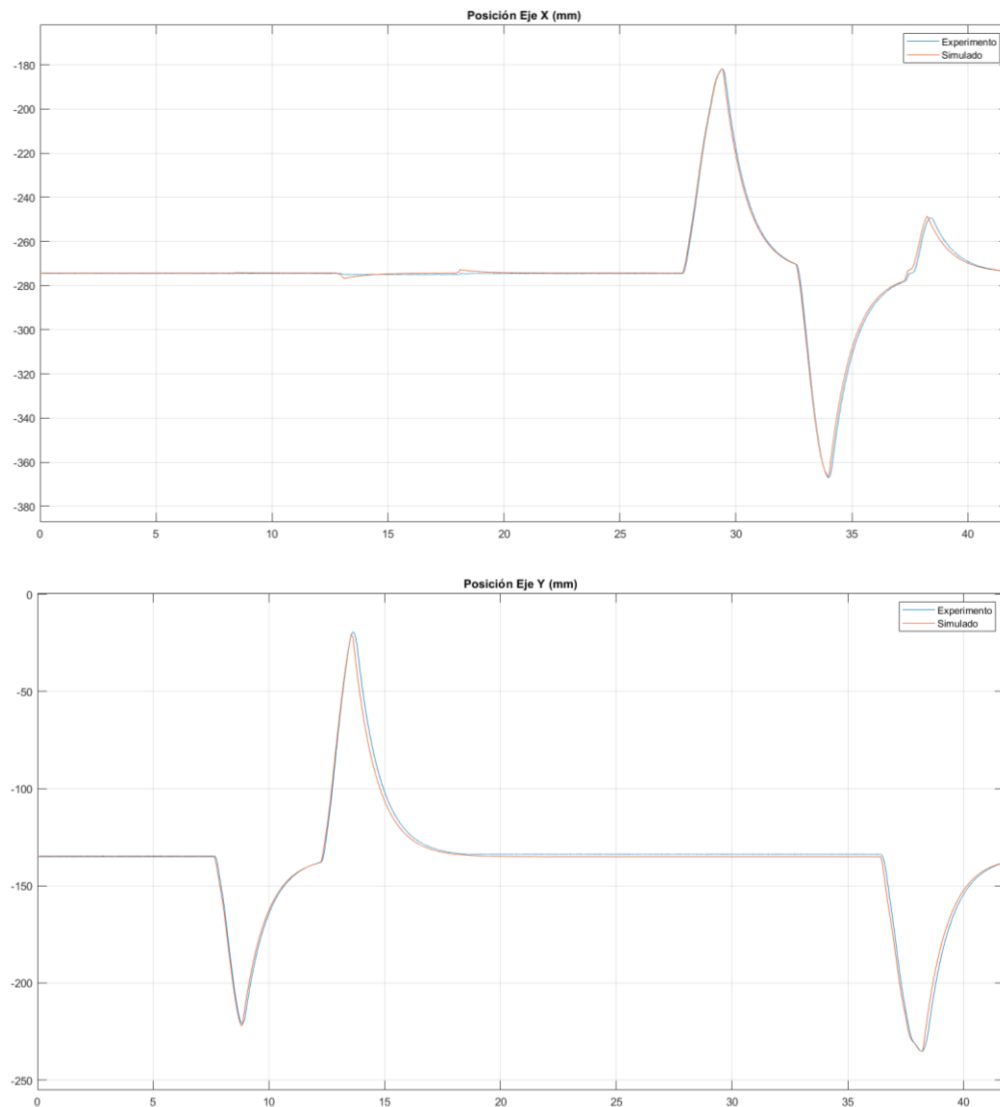
1_2 SMIA

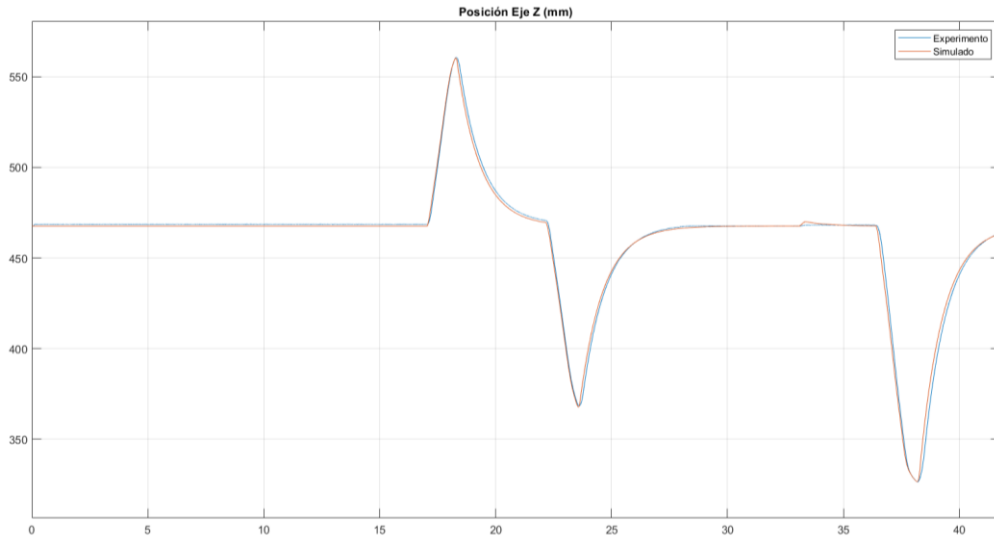
En esta ocasión, debido a que la rigidez del sistema es mayor, la frecuencia natural (ω_n) del sistema también lo es, lo que significa que la velocidad con la que este actúa al desaparecer el estímulo exterior (la fuerza de interacción en este caso) es mayor a la del caso anterior. Por lo tanto, dado que la aceleración no es suficiente, el robot no logra moverse lo suficientemente rápido y, por ende, no logra reproducir el comportamiento deseado. En las gráficas mostradas se puede apreciar como el robot siempre se queda atrás con respecto al modelo simulado.



2_2 SMIA

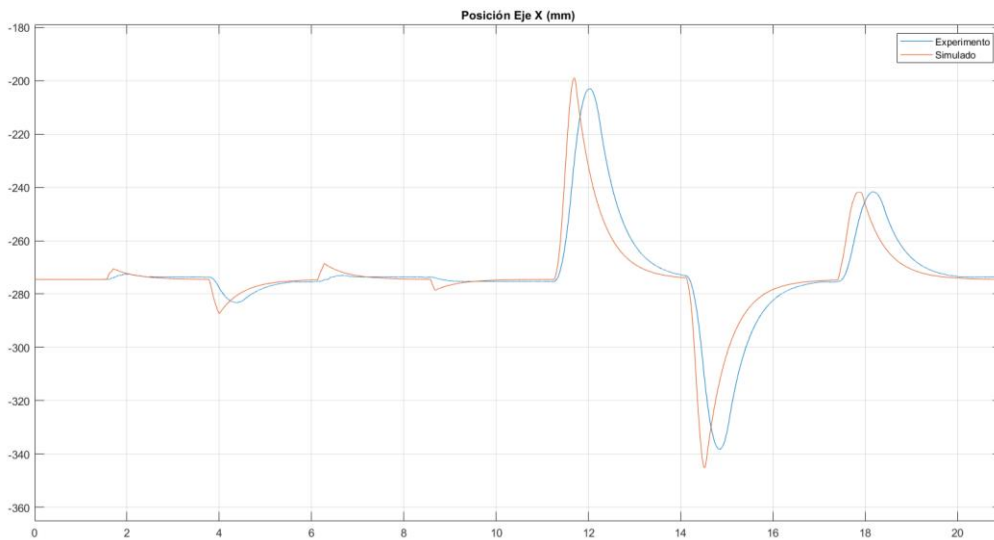
Una vez más, es posible observar una clara mejora al utilizar la acción “feed-forward”. Sin embargo, a pesar de que no es significativo, aún se puede apreciar como el robot se encuentra ligeramente atrasado con respecto al sistema simulado.

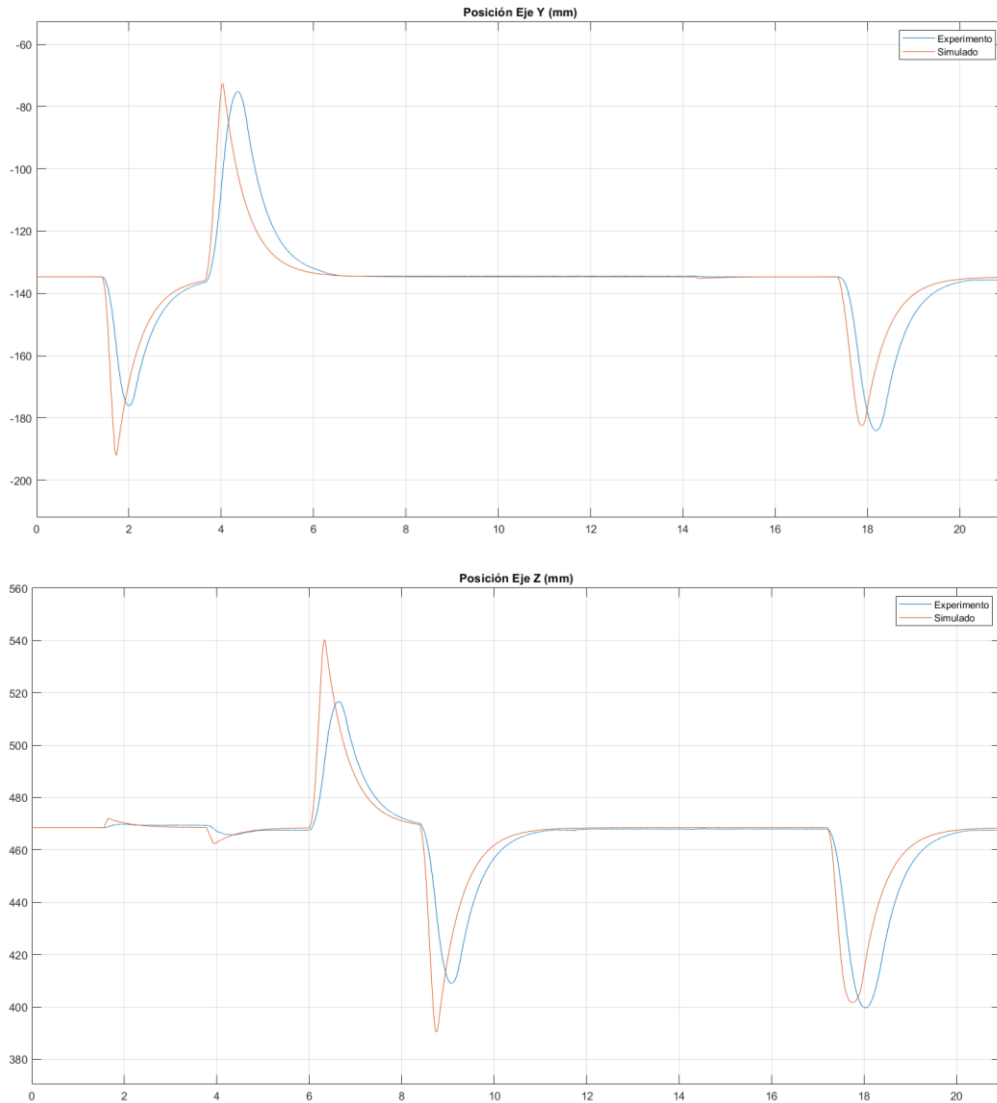




1_3 SMIA

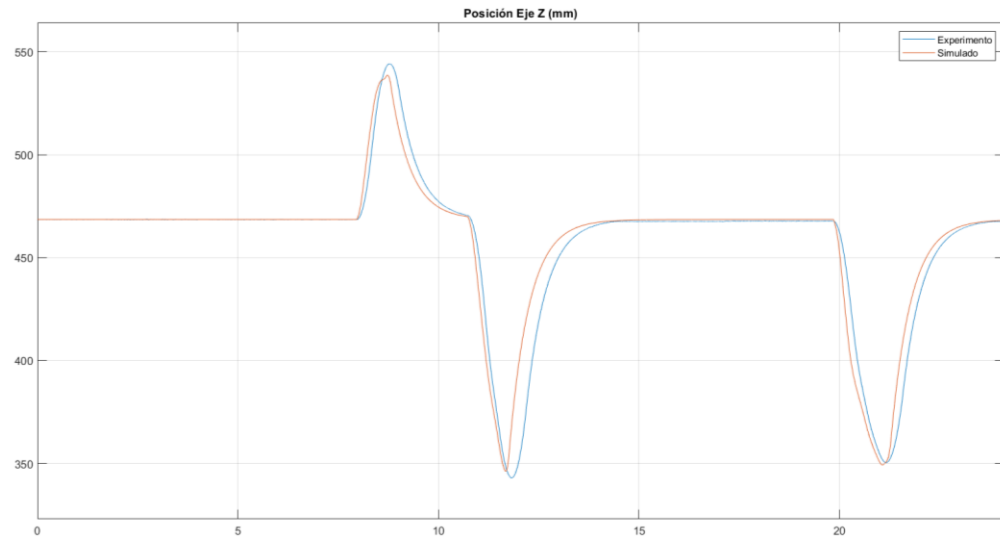
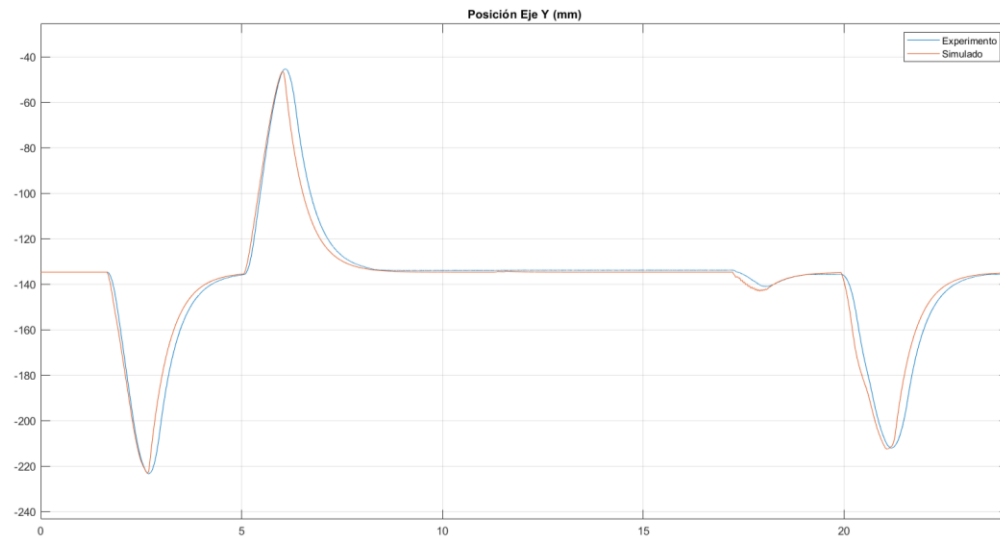
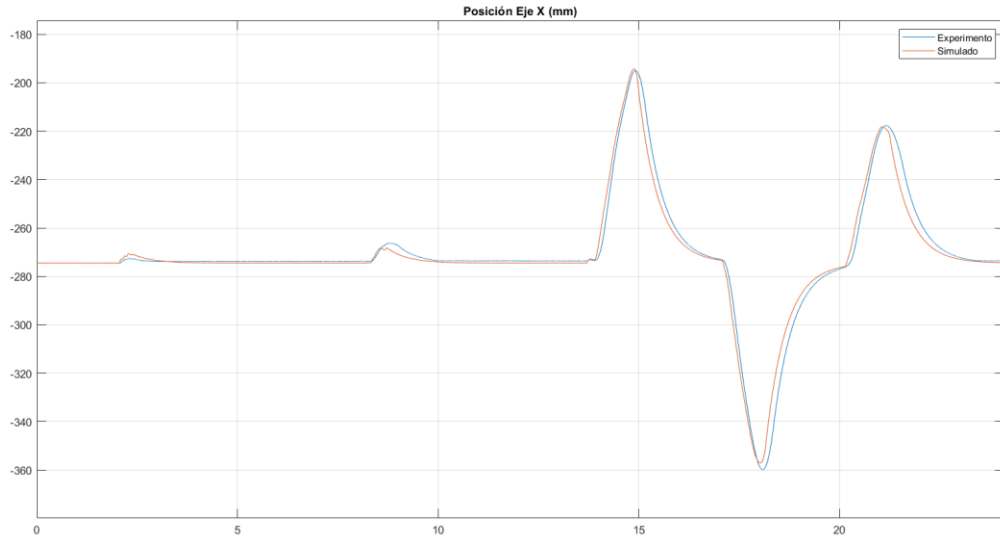
En este caso la frecuencia natural (ω_n) es aún mayor, por lo que la incapacidad de alcanzar la referencia de velocidad es más notoria. De hecho, es posible apreciar como el modelo simulado presenta picos más finos, mientras que el robot, al realizar un movimiento más lento, presenta picos más anchos y redondos.





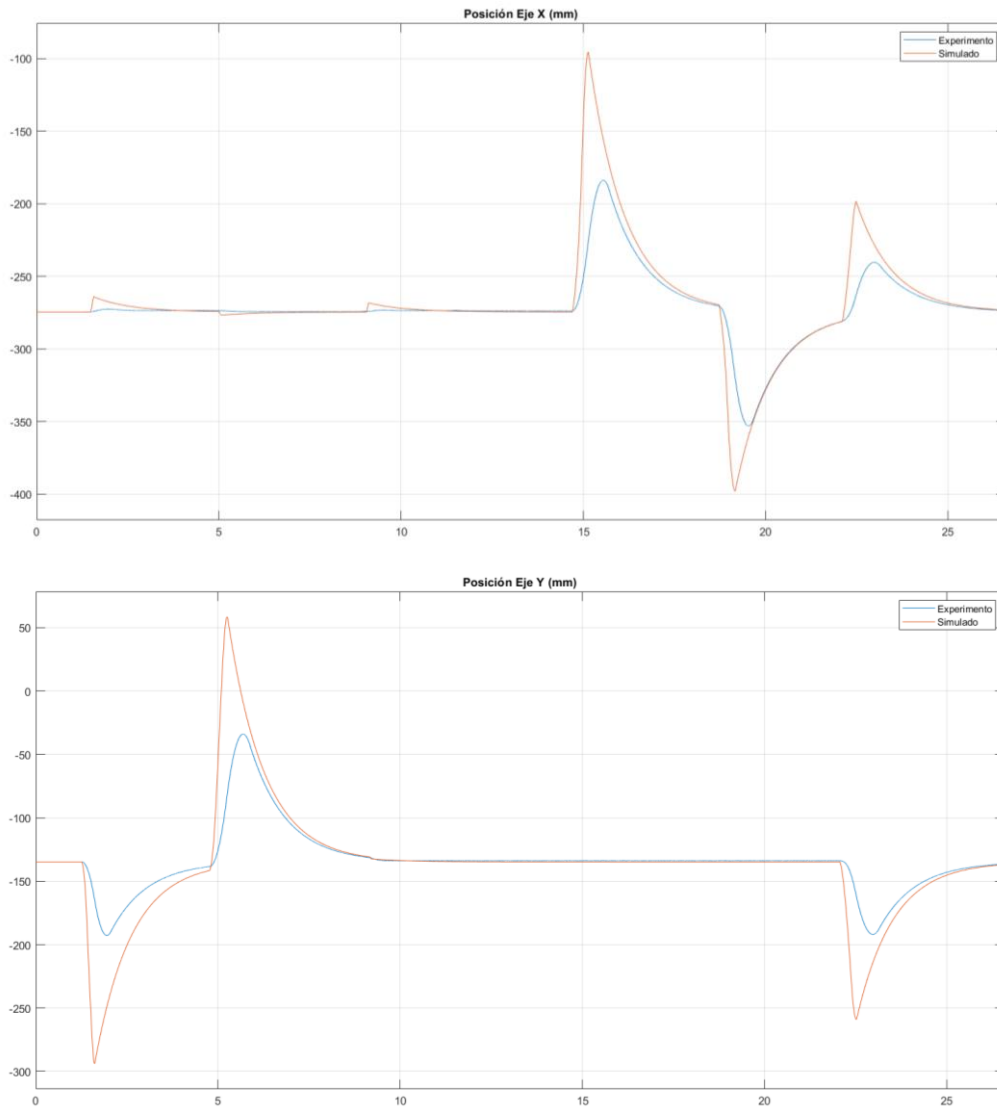
2_3 SMIA

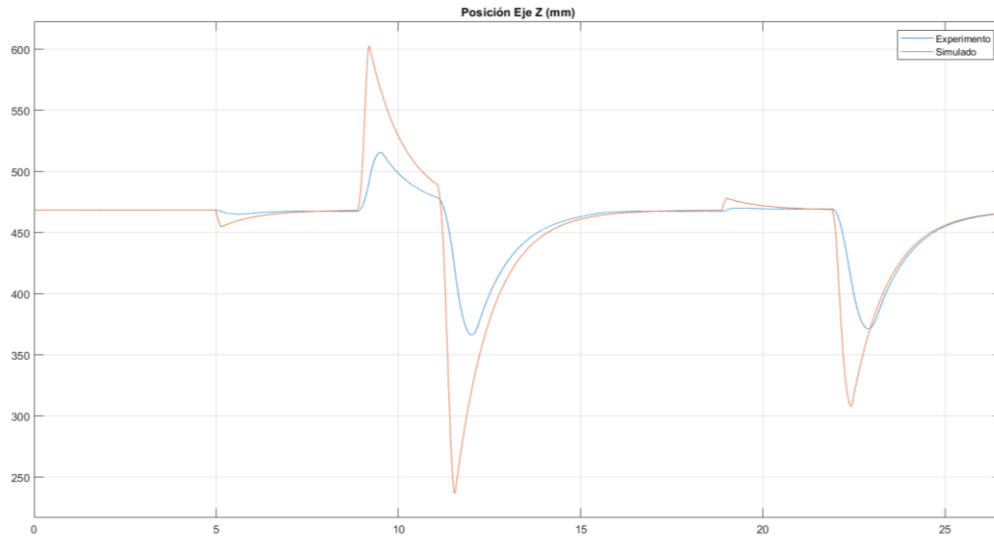
A pesar de que presenta una mejora sustancial en comparación al caso anterior, aun con la acción "feed-forward", el robot presenta una clara dificultad para seguir la referencia de velocidad, por lo que este no es capaz de reproducir fielmente el comportamiento esperado. Algo interesante a notar es que el robot es capaz de moverse adecuadamente cuando se aplica una fuerza en el efector final, pero, una vez que esta deja de actuar, el robot no puede alcanzar la velocidad necesaria para reproducir correctamente el modelo debido a que la frecuencia del sistema aumenta.



1_4 SMIA

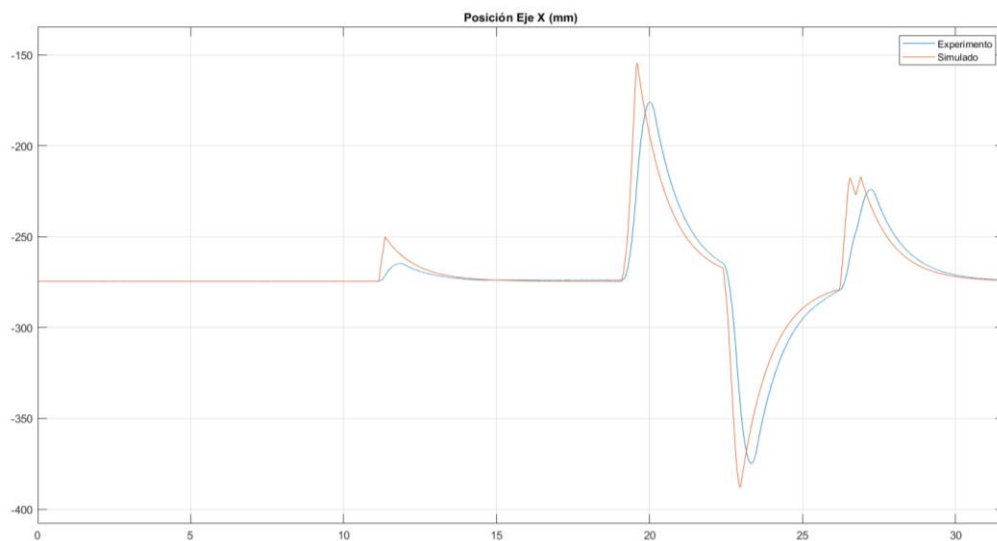
Este experimento pretende mostrar un caso extremo en el que el robot falla, en gran medida, en su tarea de realizar los movimientos necesarios para demostrar el comportamiento deseado. Esto se debe a que, al disminuir el amortiguamiento del sistema, la frecuencia natural ω_n aumenta, lo que significa que la velocidad con la que funciona el sistema también lo hace y por lo tanto el robot no es capaz de alcanzar la velocidad necesaria para reproducir el comportamiento esperado.

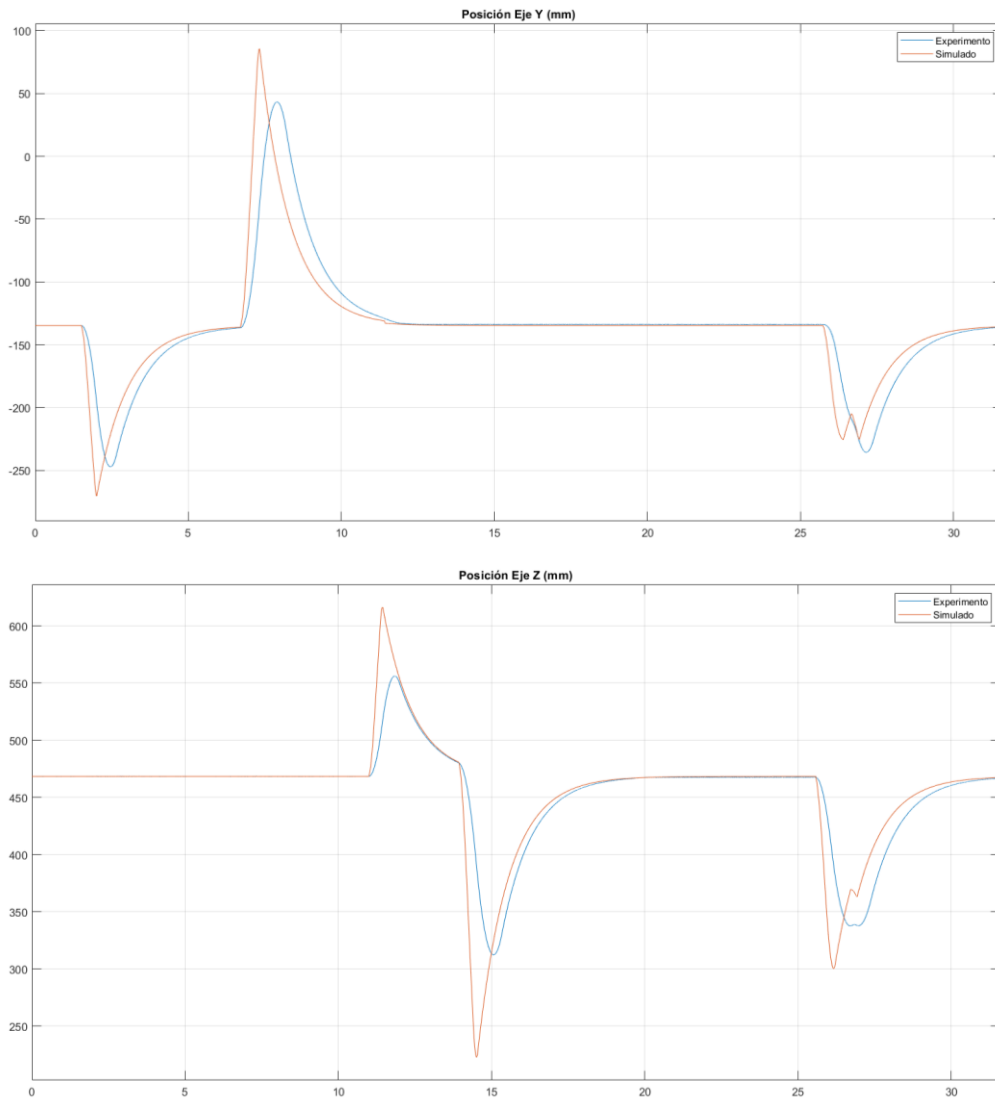




2_4 SMIA

Otra vez es posible apreciar el efecto que tiene la acción “feed-forward” sobre el sistema, mejorando significativamente la respuesta. Sin embargo, en este caso dicha acción no es suficiente para hacer que el robot reproduzca la dinámica deseada correctamente.





Resumen: comparación del controlador SMIA con y sin acción “feed-forward”

A manera de resumen y para poder comparar mejor los resultados se tiene la Tabla 5.7, donde se presentan los errores cuadráticos de cada experimento. Es necesario señalar que se eligió esta métrica debido a que permite evidenciar fácilmente la magnitud de la diferencia entre los datos.

Tabla 5.7: Error cuadrático con y sin acción “feed-forward” con el controlador SMIA.

Experimento	X	Y	Z
1_1 SMIA	8.09	27.30	19.71
2_1 SMIA	2.03	3.56	21.79
1_2 SMIA	45.45	33.87	103.46
2_2 SMIA	1.82	7.55	7.00
1_3 SMIA	91.73	62.17	87.65
2_3 SMIA	15.36	29.31	90.25
1_4 SMIA	342.49	634.84	814.36
2_4 SMIA	122.80	438.18	639.09

5.3. Resultados del controlador de admitancia con modificación de la inercia aparente

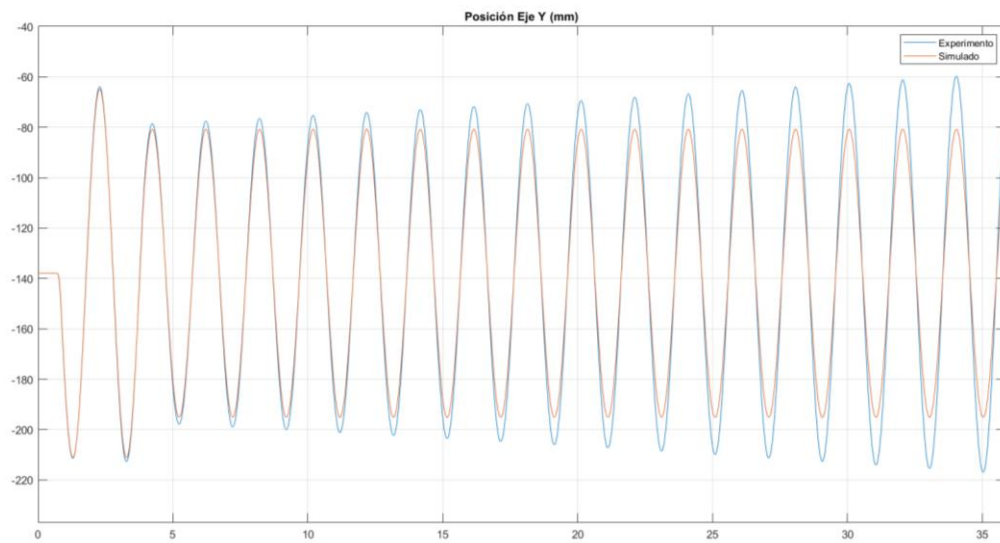
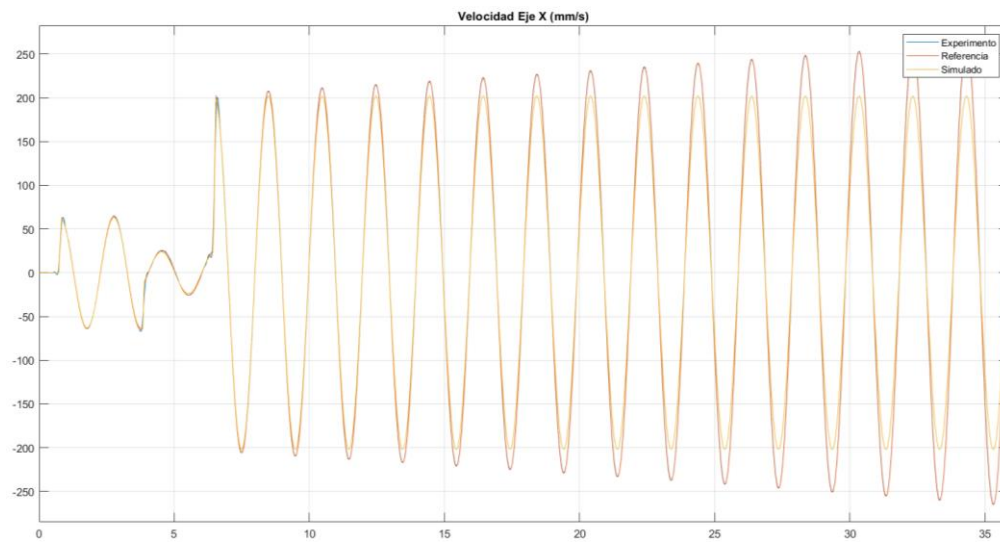
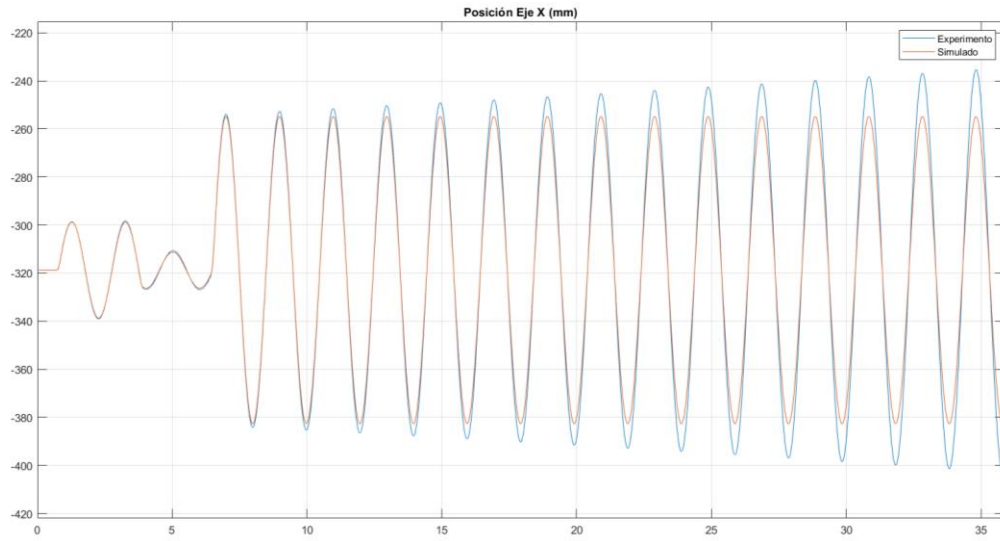
Al igual que en el inciso anterior, se mostrarán los resultados obtenidos junto con una breve descripción de lo que sucede en cada uno. En estos experimentos se presentan tanto las gráficas de posición como de velocidad con el fin de entender mejor lo que está sucediendo. Además, las gráficas de velocidad muestran tanto la velocidad de referencia como la velocidad real del robot para poder apreciar la exactitud del modelo y, al mismo tiempo, el efecto de las limitaciones del robot en su capacidad para reproducir el comportamiento esperado.

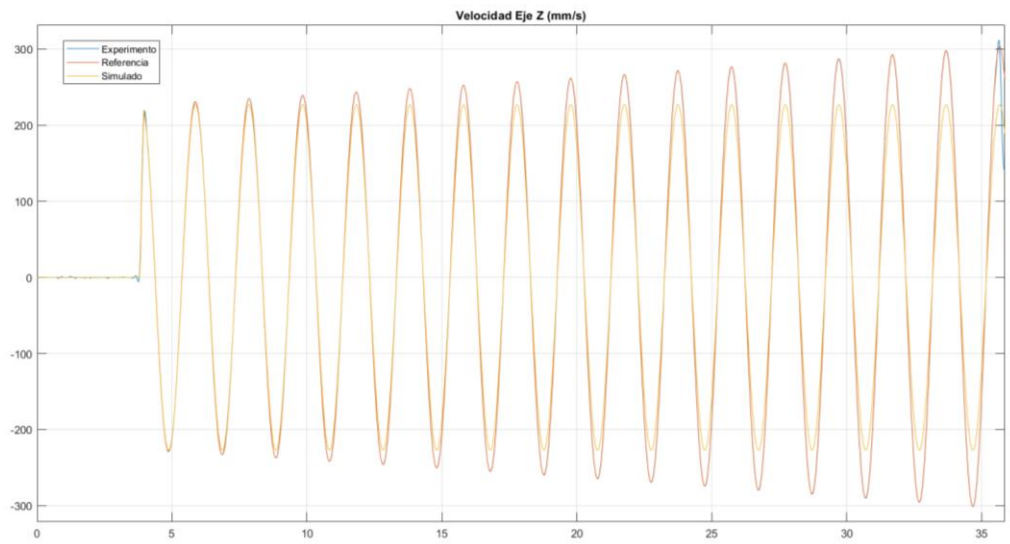
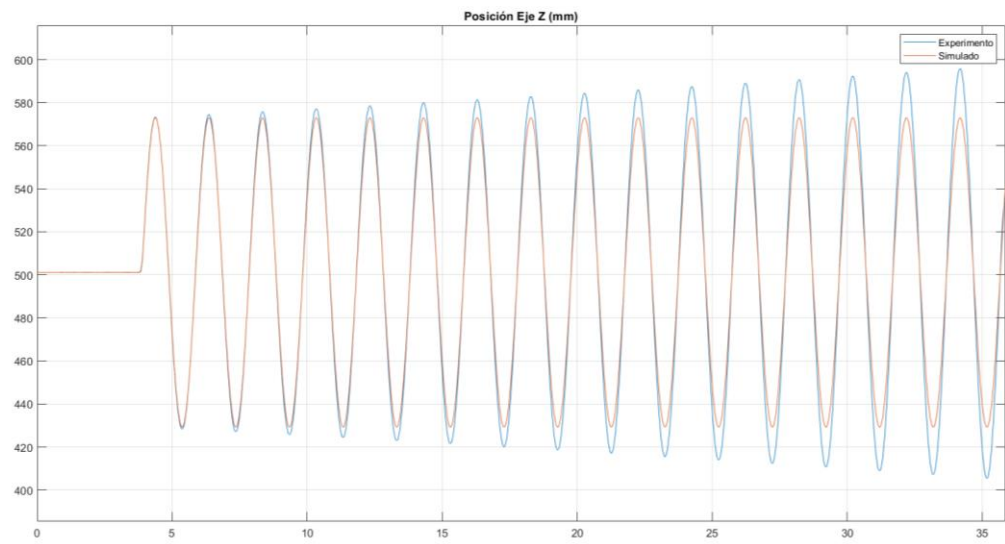
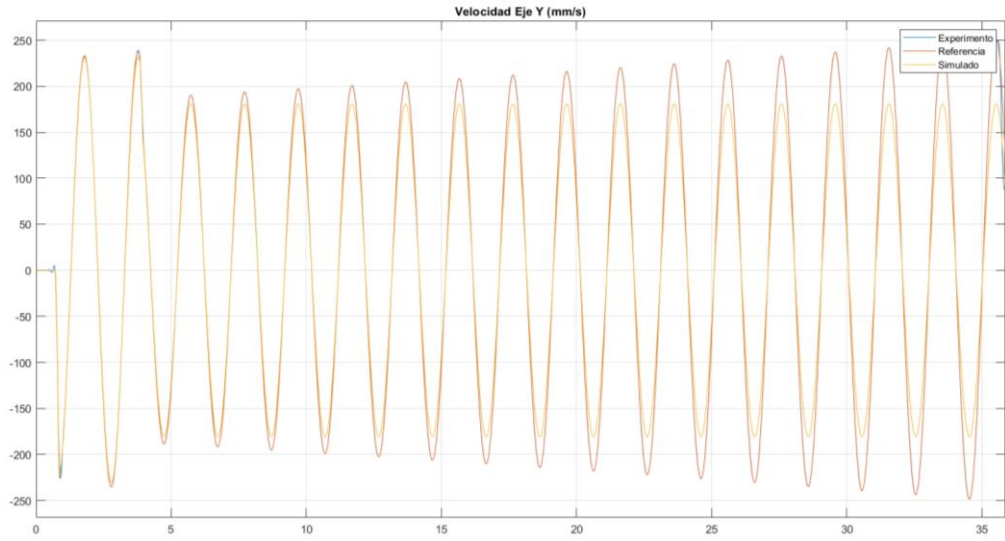
Oscilatorio

Este experimento se hizo con la intención de poner a prueba los límites del sistema debido a que este no es un comportamiento que sea deseable en aplicaciones robóticas. No obstante, provee una idea de los límites del sistema y los efectos no lineales que afectan al mismo, evitando que el robot pueda reproducir el comportamiento y tienda hacia la inestabilidad.

En este caso se excitaron los tres ejes al ejercer una fuerza en cada uno de ellos de forma progresiva (primero el eje Y, luego el Z y por último el eje X), con lo que a partir de cierto punto el robot se movía en los tres ejes al mismo tiempo. De esta forma, es posible apreciar como no solo la posición, sino también la velocidad del robot tiende hacia la inestabilidad dado que su valor aumenta (de forma aparentemente exponencial) conforme pasa el tiempo. Es importante notar que las pequeñas oscilaciones presentes son debido a la inexactitud de la dirección en la aplicación de la fuerza, sin embargo, estas no influyen significativamente en los resultados.}

Por otro lado, es interesante observar como el robot, a pesar de tender a la inestabilidad, sigue respetando la frecuencia natural del sistema ($\omega_n = 3.16 \text{ rad/s}$) en todos los ejes, la cual coincide con la frecuencia del modelo simulado.

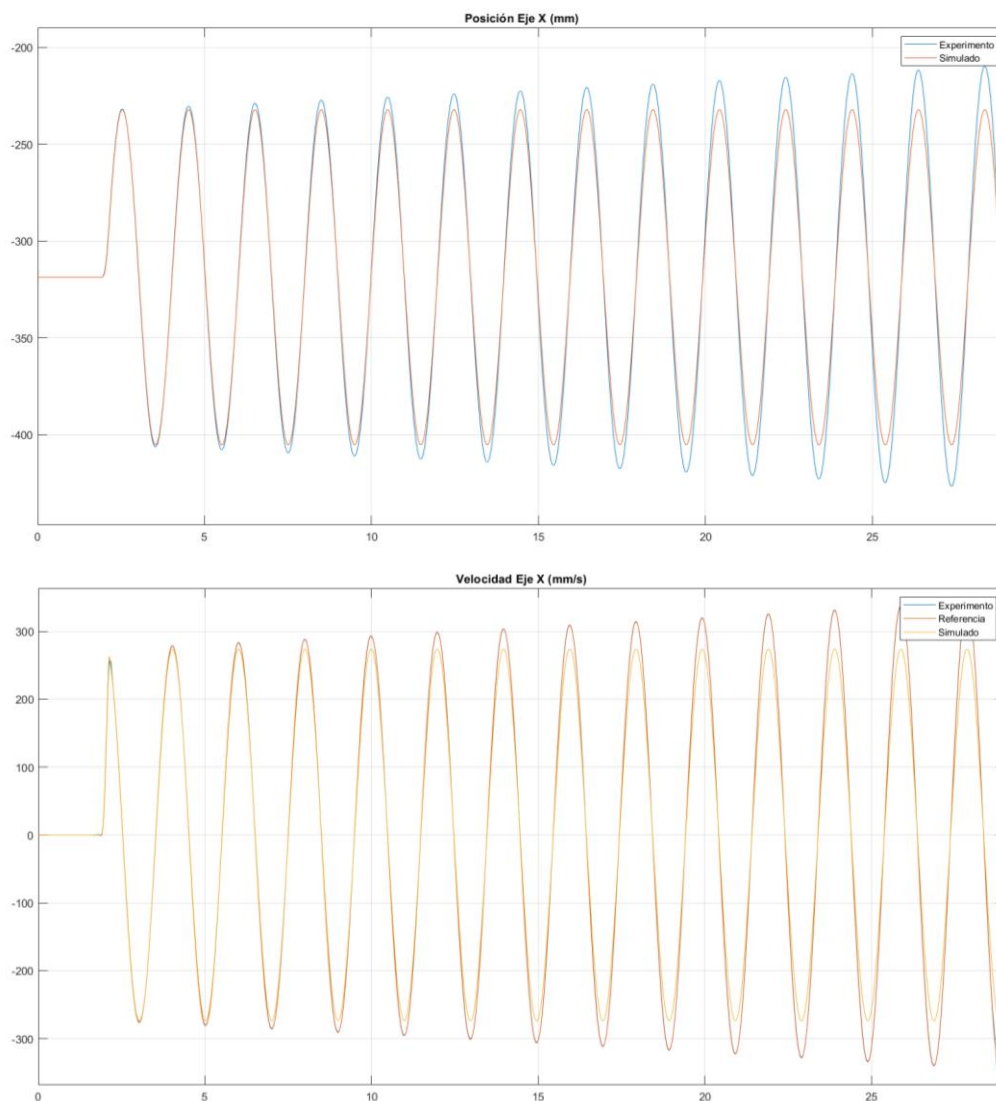


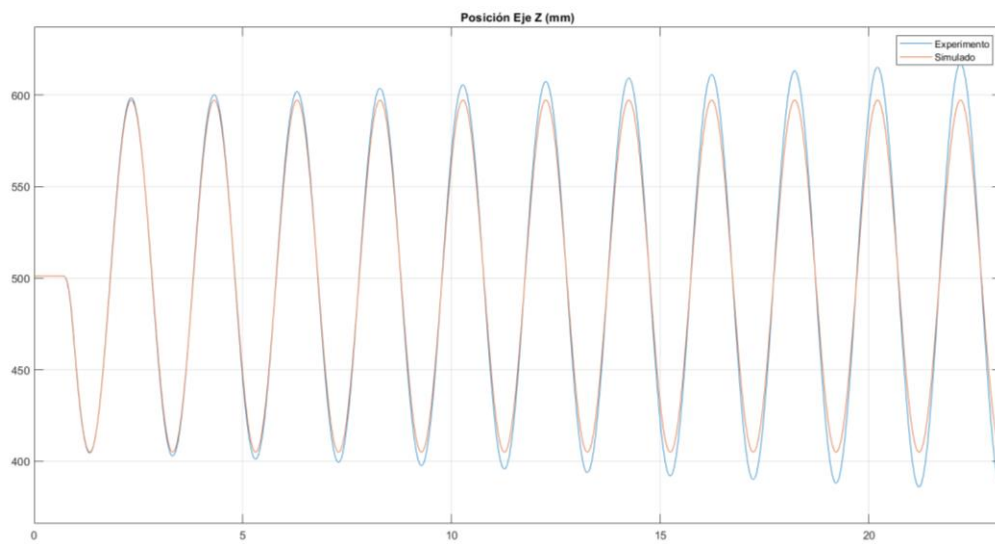
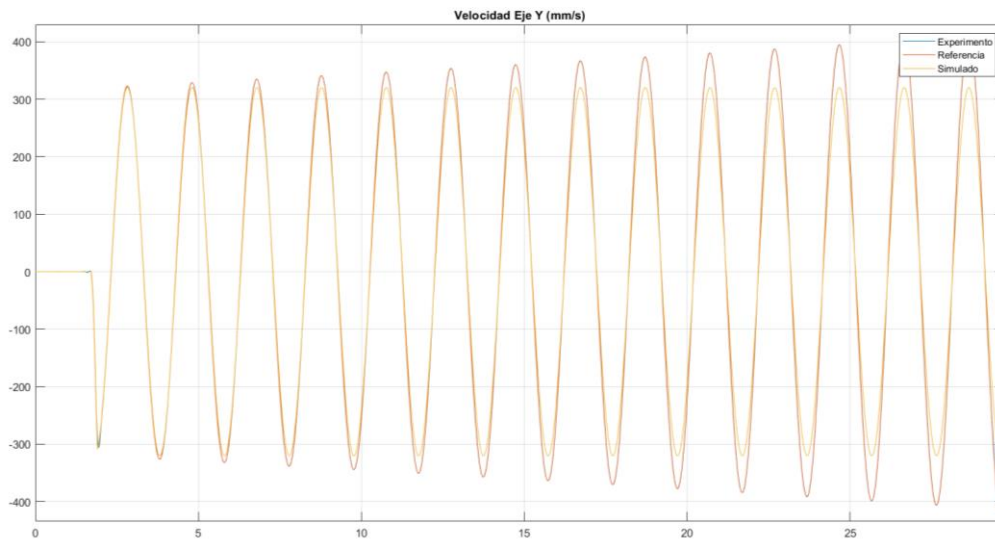
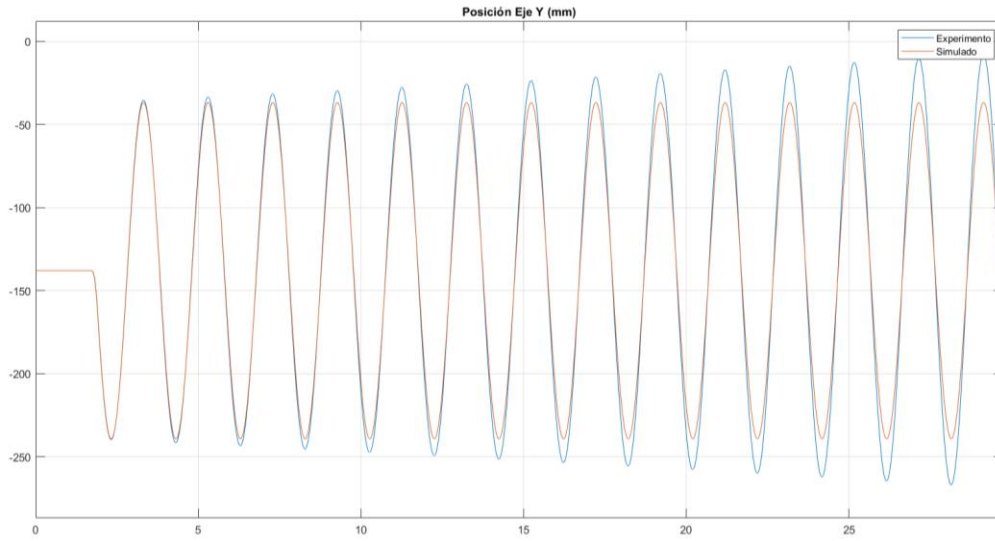


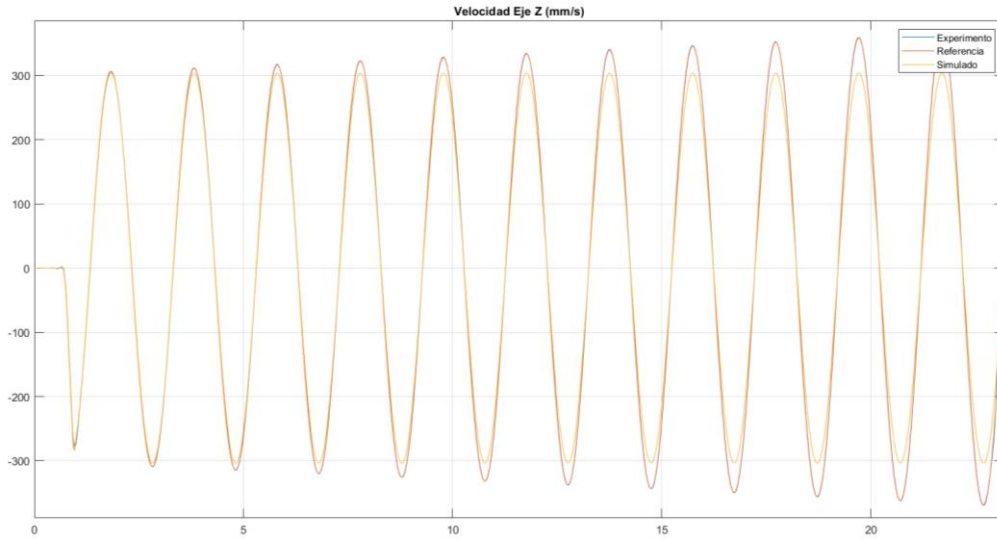
Osci_XYZ

En este caso, los ejes se accionan cada uno por separado, con lo que nunca se tienen movimientos en más de un eje a la vez. Con esto se pretendía evaluar, en conjunto con el experimento anterior, si existía algún tipo de acoplamiento entre los movimientos ejecutados en cada eje.

De esta forma, al observar las gráficas se ve que el comportamiento mostrado es similar al del experimento anterior, con lo que, a simple vista, parece no haber ningún acoplamiento entre los movimientos de los ejes.

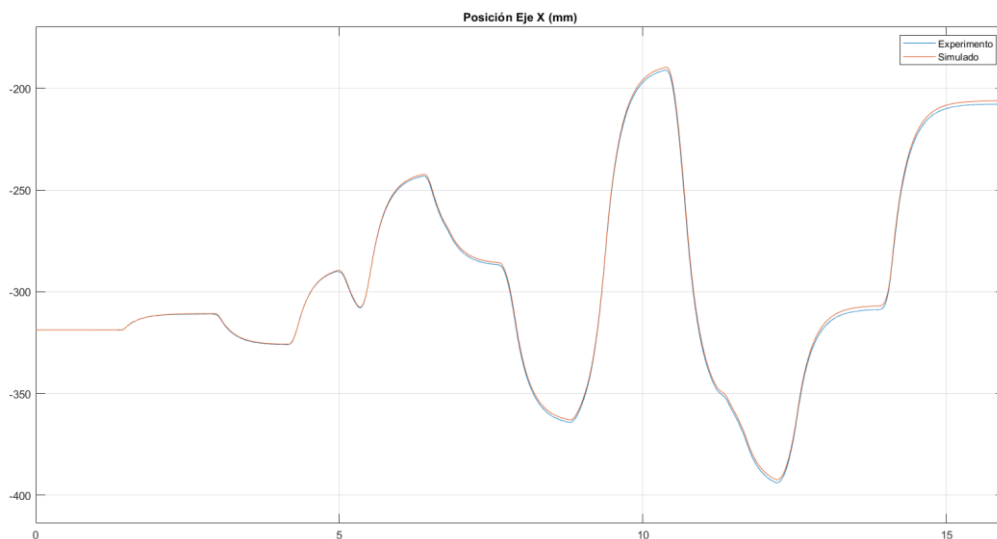


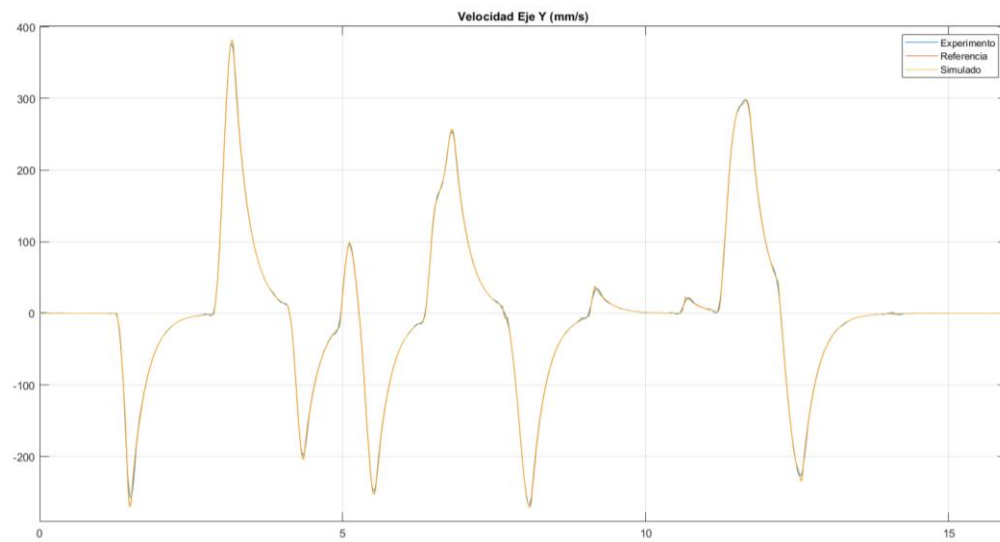
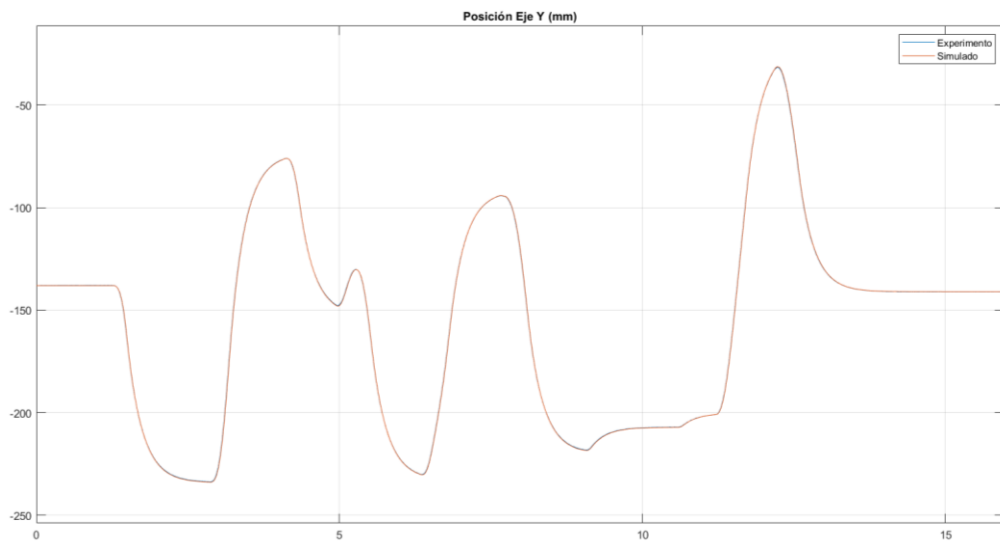
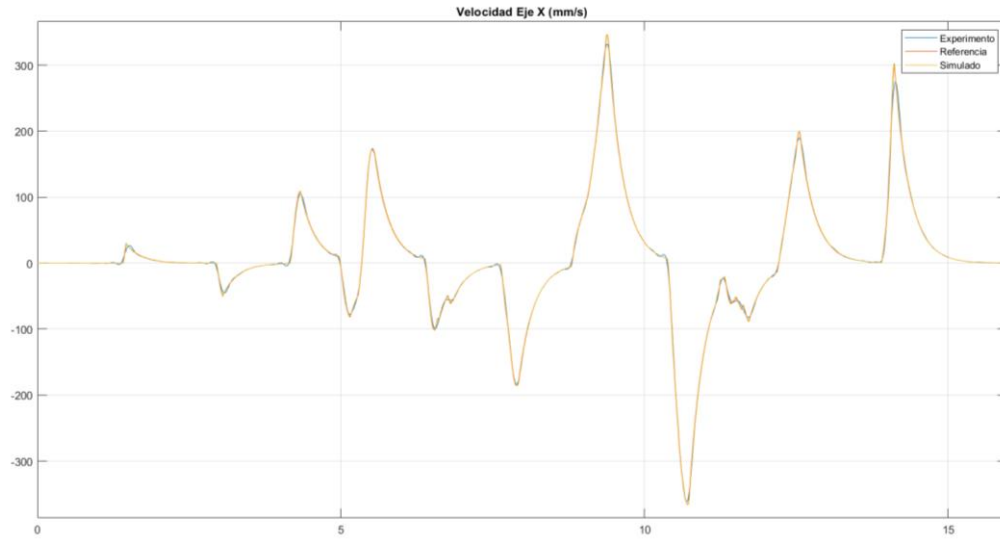


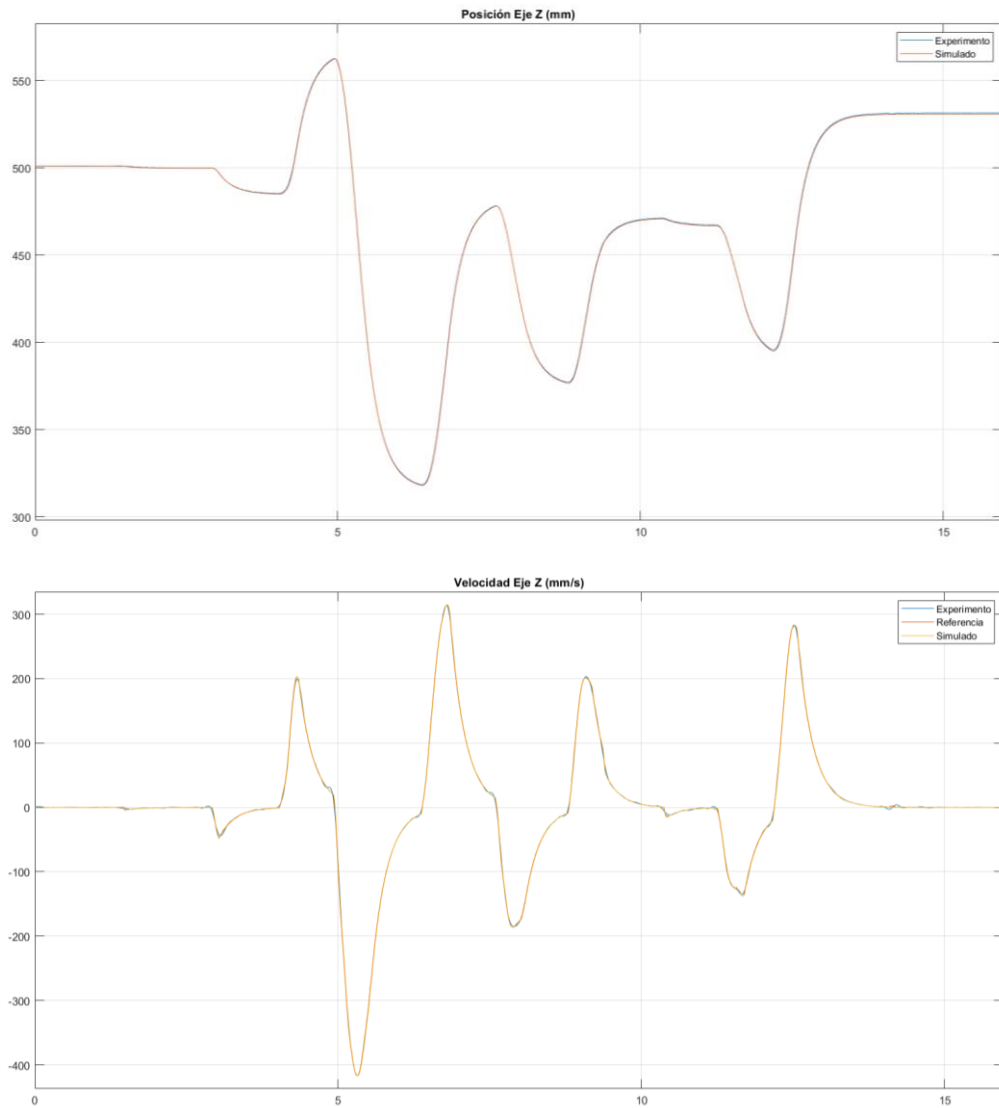


No_Osci

A diferencia de los experimentos anteriores, este si presenta un comportamiento que puede ser útil en aplicaciones robóticas, por lo que se realizó con el propósito de verificar que el robot podía funcionar de forma efectiva solo utilizando el amortiguamiento y la inercia. Así, al observar los resultados, es posible apreciar que el robot ejecuta los movimientos casi a la perfección, confirmando que este es capaz de reproducir el comportamiento deseado.







Resumen: experimentos con y sin oscilaciones

De la misma manera que en los experimentos realizados para el controlador SMIA, en este caso se tiene la Tabla 5.8, la cual presenta los errores cuadráticos de los experimentos con y sin oscilaciones.

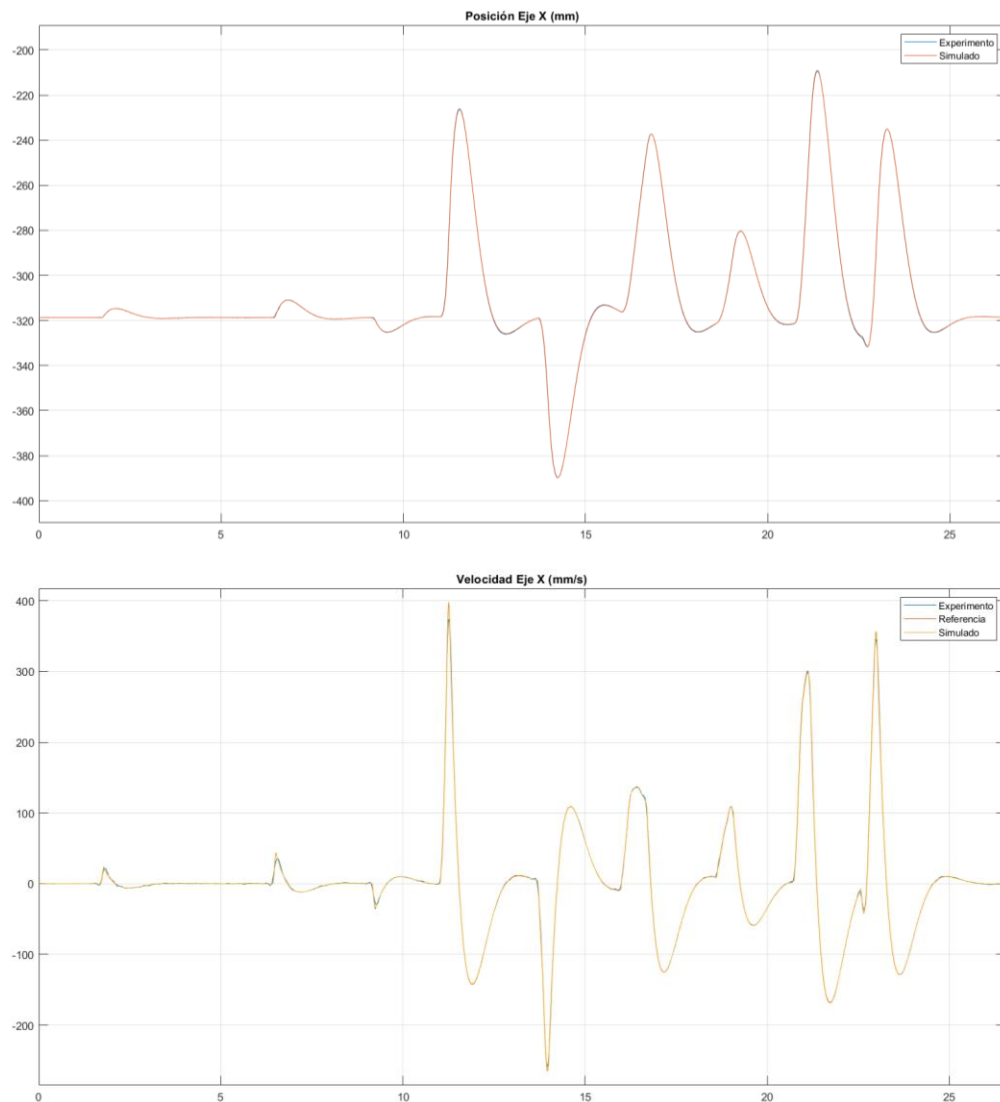
Tabla 5.8: Error cuadrático con y sin oscilaciones con el controlador CMIA

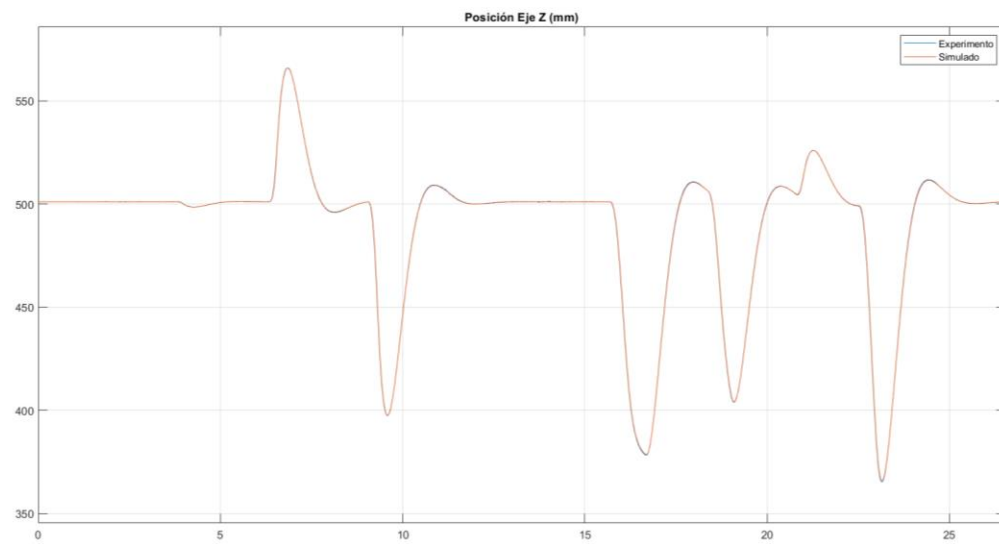
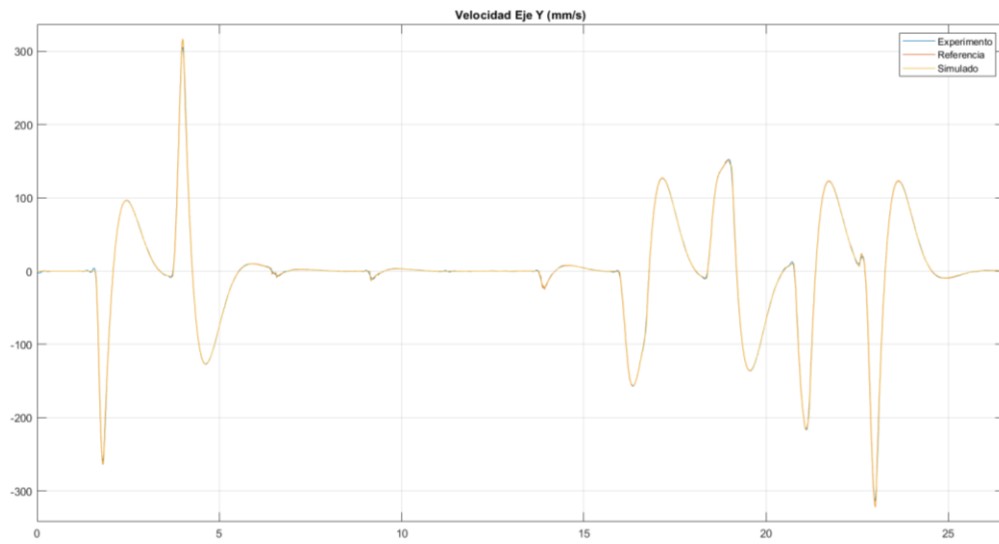
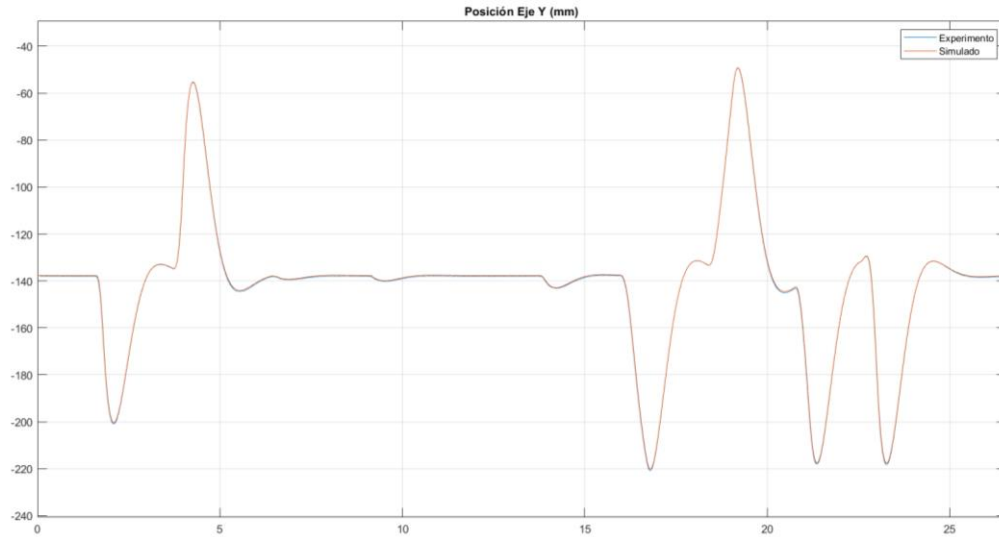
<i>Experim.</i>	<i>X</i>	<i>Y</i>	<i>Z</i>	<i>Vx</i>	<i>Vy</i>	<i>Vz</i>	<i>Vref_x</i>	<i>Vref_y</i>	<i>Vref_z</i>
<i>Oscilatorio</i>	54.21	74.34	79.23	528.48	754.26	809.41	537.99	769.53	825.34
<i>Osci_XYZ</i>	75.78	127.52	66.77	760.09	1282.4	664.61	790.29	1320.1	677.02
<i>No_Osc</i>	1.24	0.08	0.28	7.78	4.03	3.08	4.29	5.44	3.78

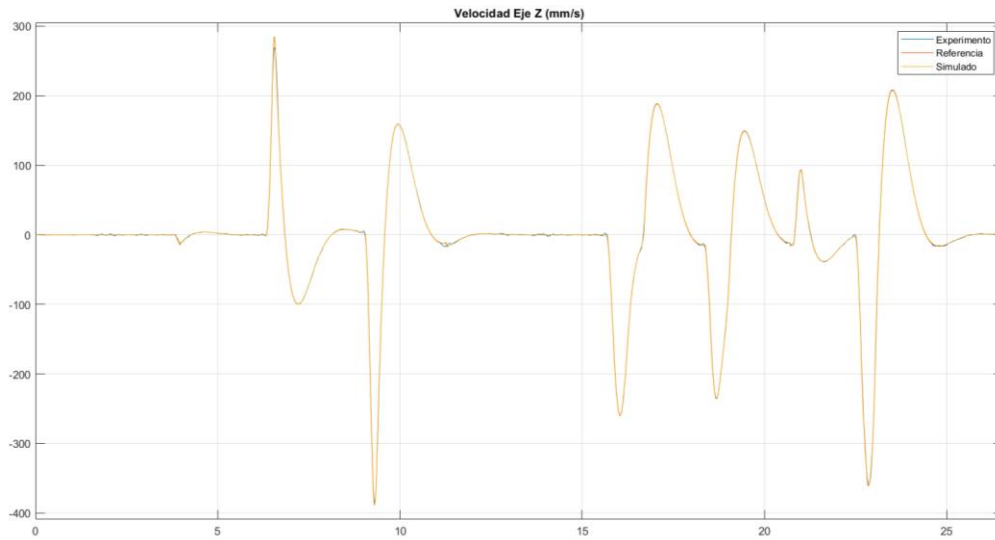
1_1 CMIA

En este experimento, los parámetros utilizados dan como resultado un comportamiento sub amortiguado ($\zeta = 0.63$), el cual se manifiesta en las oscilaciones mostradas en las gráficas. De esta manera, la parte de la izquierda del pico en la gráfica de posición es debido a la aplicación de una fuerza sobre el robot, mientras que la parte derecha es debido a la rigidez del sistema. Así, una vez que se deja de aplicar la fuerza sobre el efector final del robot, es posible apreciar como el sistema presenta pequeñas oscilaciones antes de volver a su estado estacionario.

En este caso se puede ver como el robot es capaz de seguir, prácticamente a la perfección, el comportamiento deseado gracias a la correcta elección de la magnitud de la aceleración para el controlador de velocidad del UR3e. Esto se aprecia mejor en la gráfica de velocidad, donde el robot es capaz de alcanzar los picos más pronunciados y, por lo tanto, es capaz de ejecutar el movimiento esperado.

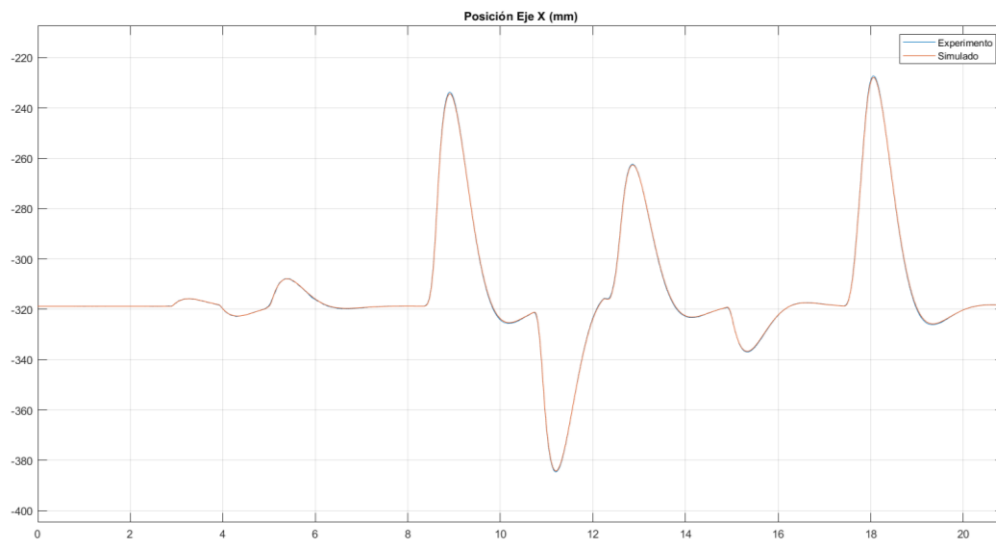


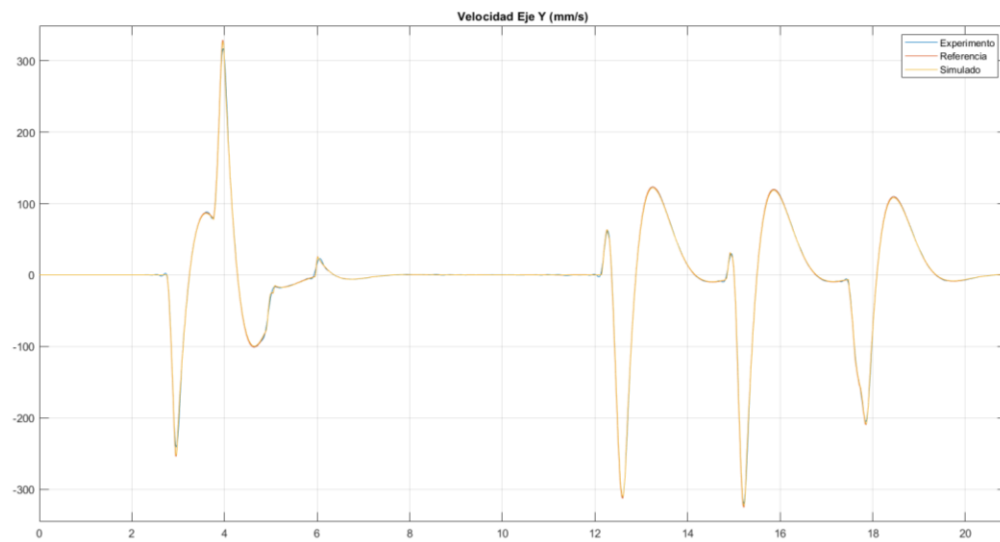
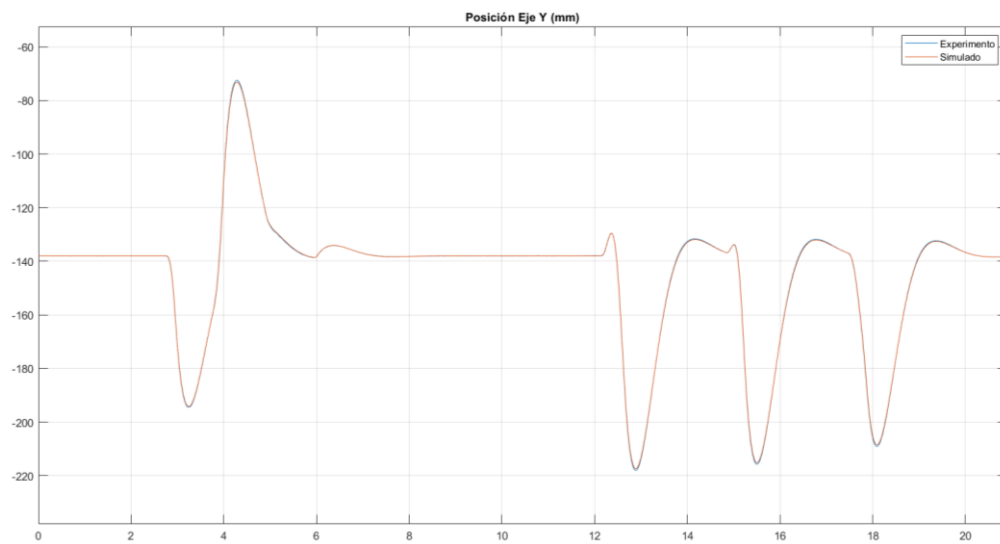
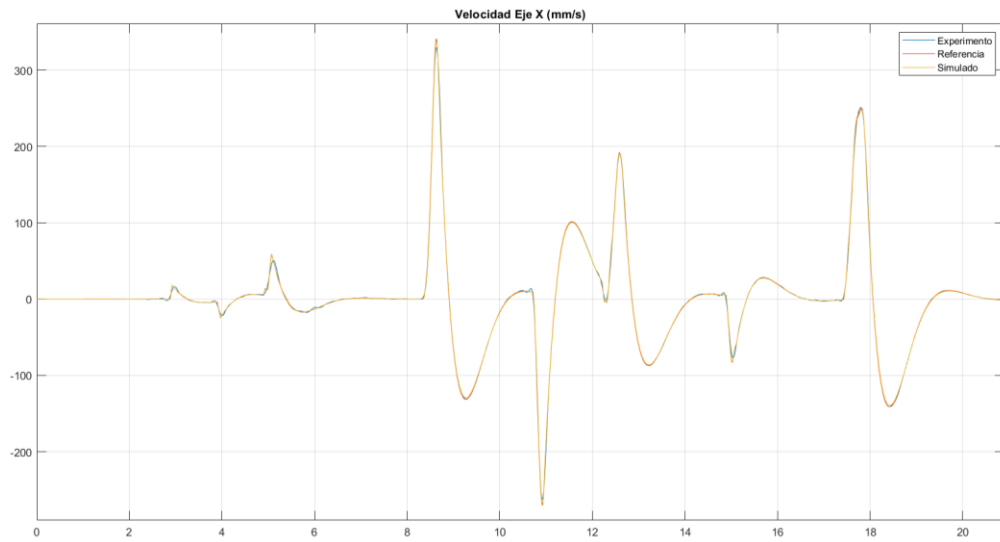


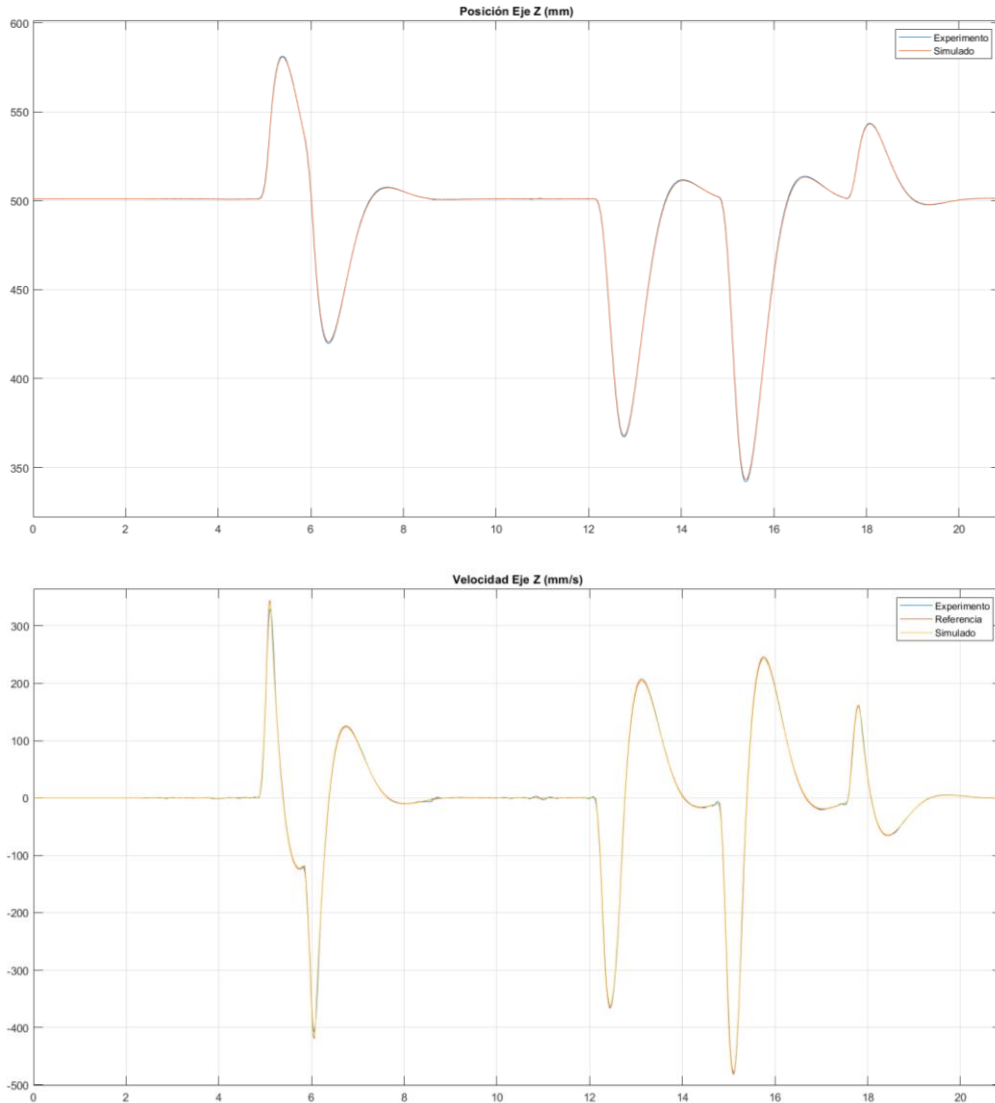


2_1 CMIA

Dado que, en este caso la aceleración es suficiente para que el robot ejecute los movimientos a tiempo, el uso de la acción “feed-forward” no muestra una mejora significativa en el desempeño del controlador.



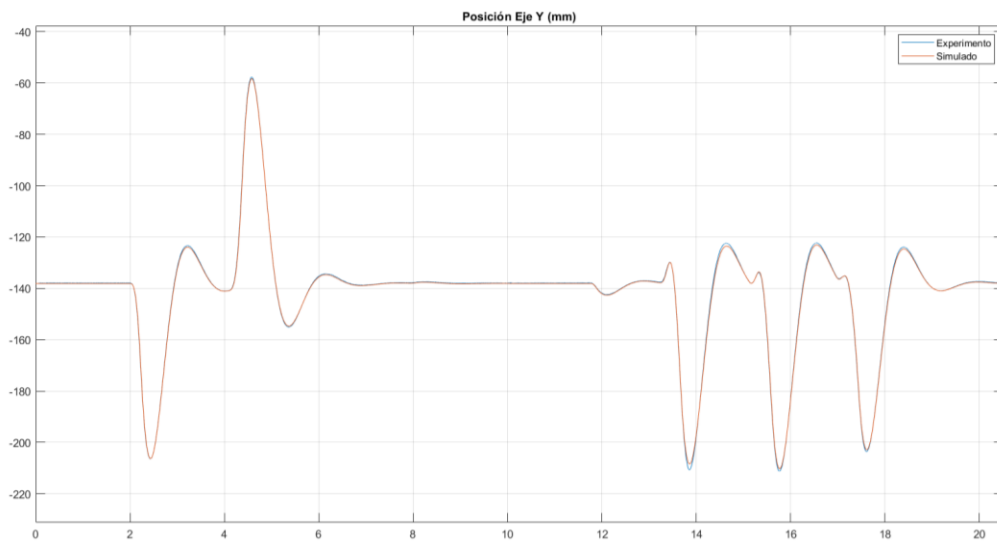
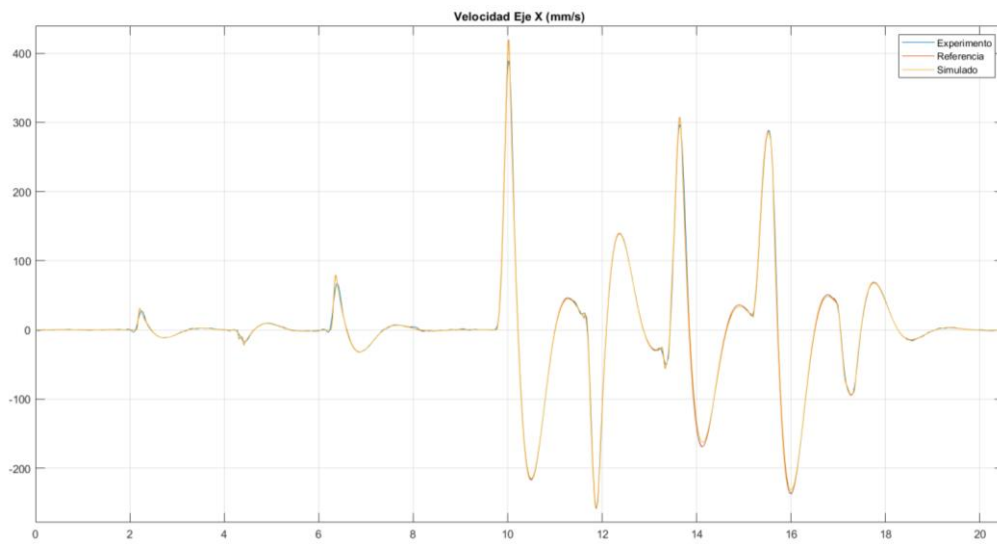
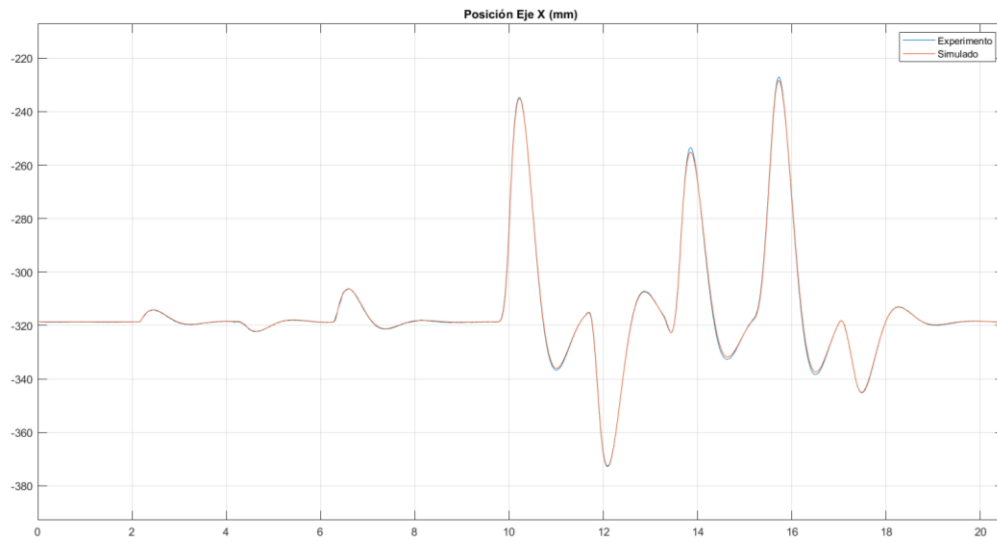


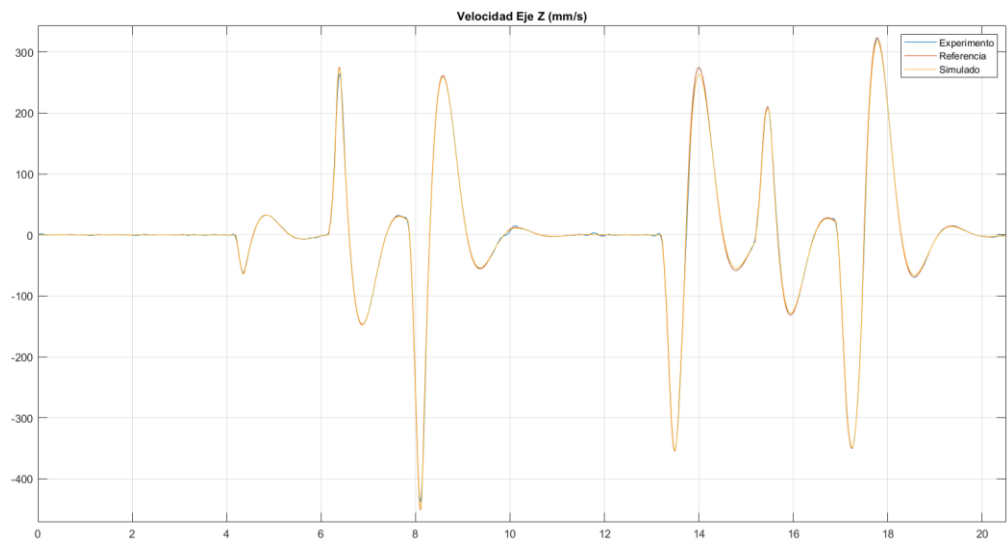
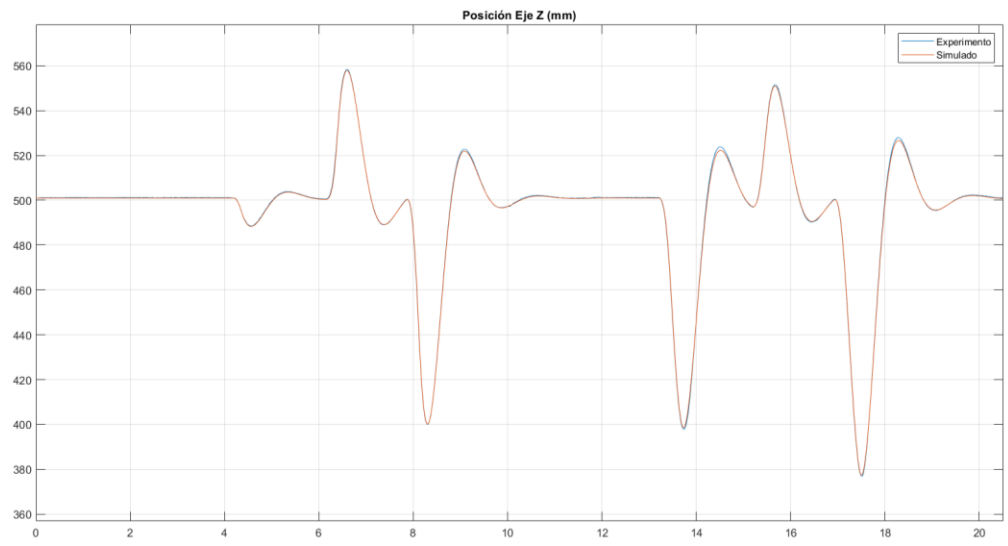
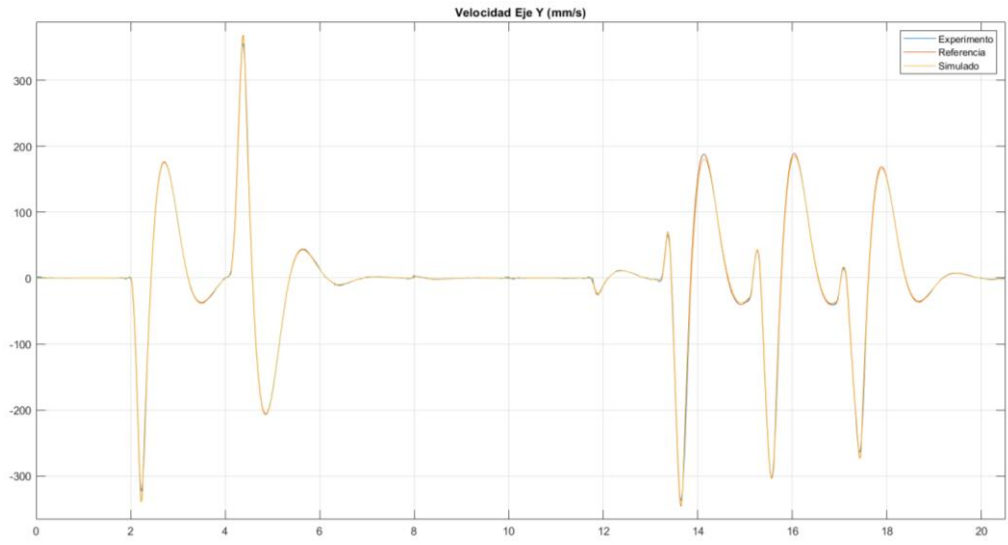


1_2 CMIA

En este experimento también se tiene un comportamiento sub amortiguado ($\zeta = 0.45$), aunque en este caso la frecuencia amortiguada del sistema ($\omega_d = 4 \text{ rad/s}$) es mayor a la del experimento anterior, por lo que el robot debe alcanzar una velocidad más alta para poder reproducir el comportamiento deseado.

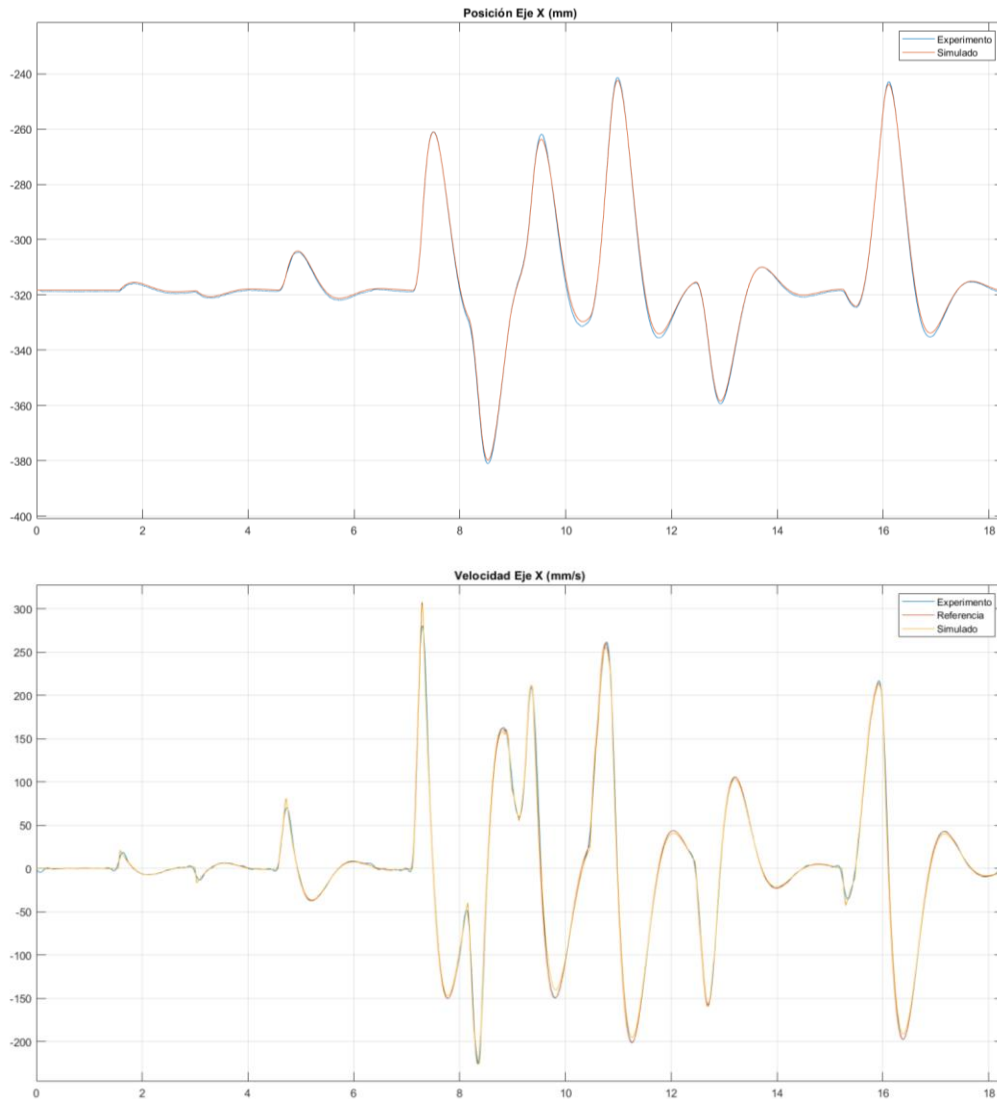
Algo a notar en este experimento es que, en los picos de las gráficas de posición, se observa como el robot sobre pasa por poco la posición esperada por el modelo. Esto se debe principalmente a que el modelo implementado en el UR3e sobre estima ligeramente la referencia de velocidad del robot (esto es evidente en los picos de las gráficas de velocidad), haciendo que este vaya más rápido de lo que debería, lo cual provoca que este se desplace un poco más de lo necesario.

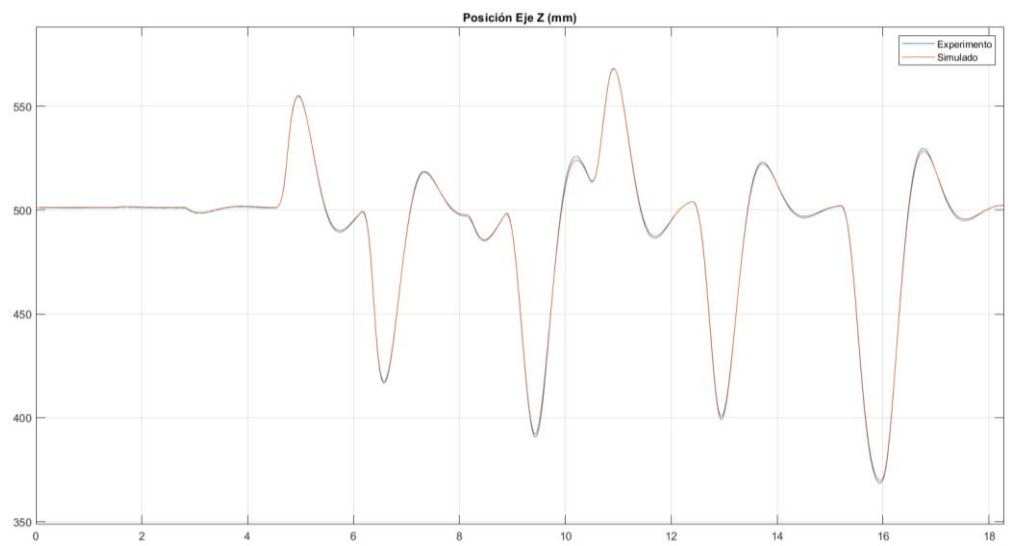
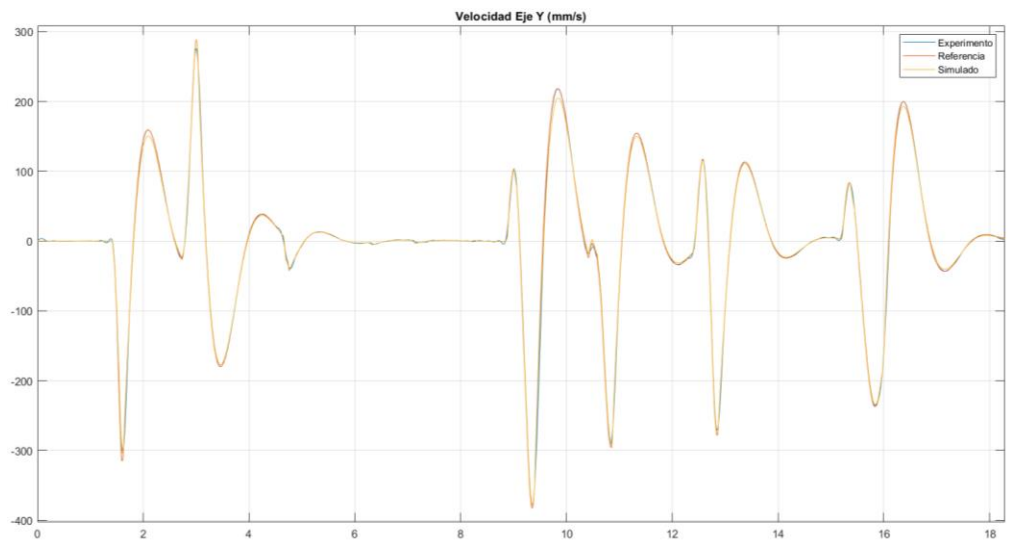
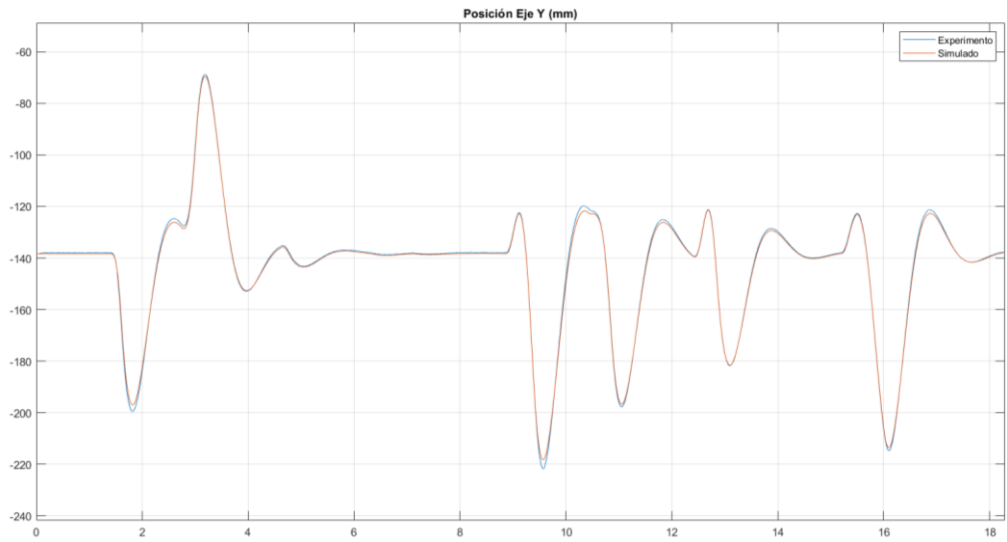


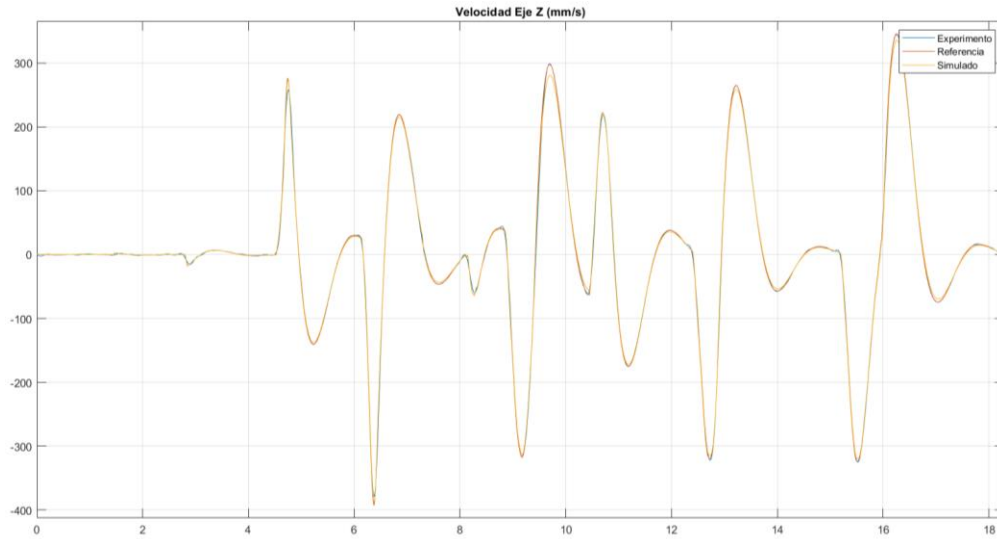


2_2 CMIA

A pesar de que los movimientos son un poco más bruscos que en el caso anterior, es posible ver como el robot es capaz de seguir la referencia de velocidad durante casi todo el experimento. No obstante, al igual que en el caso anterior, esta referencia tiene un valor un poco mayor al ideal, por lo que el robot termina desplazándose más de lo que debería.

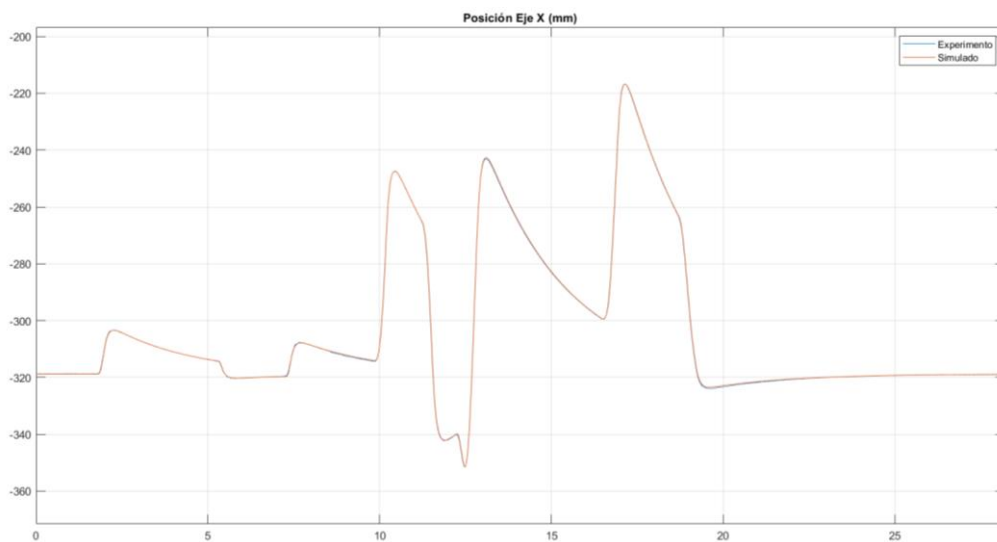


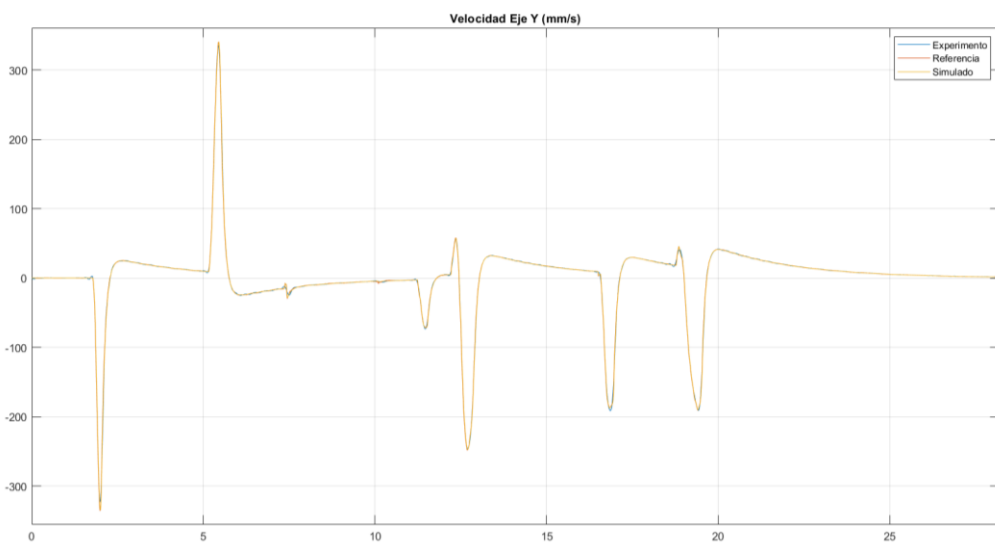
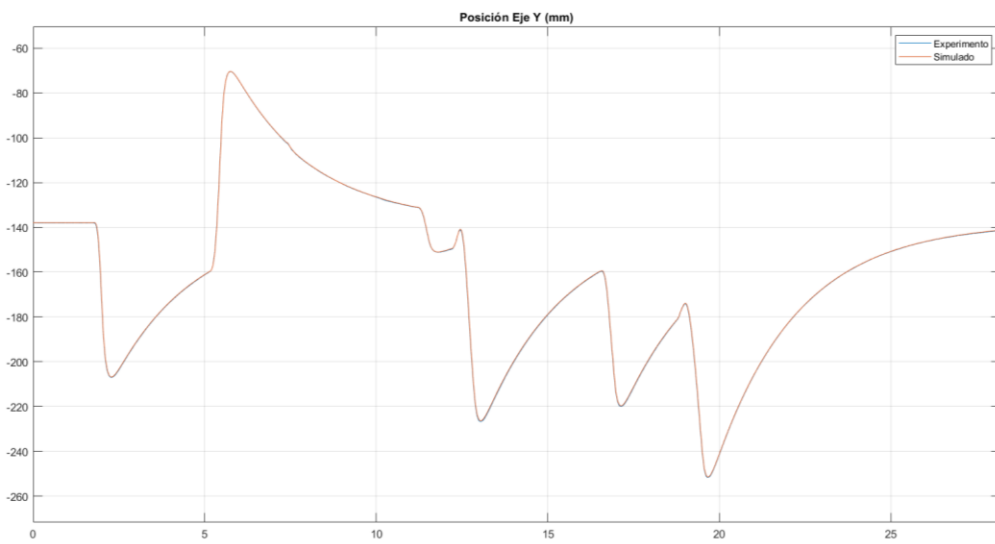
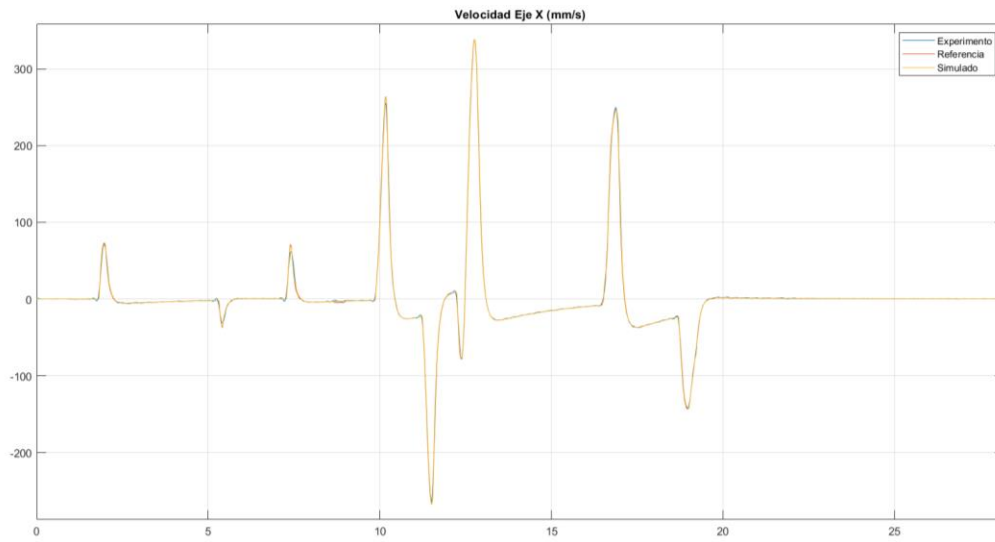


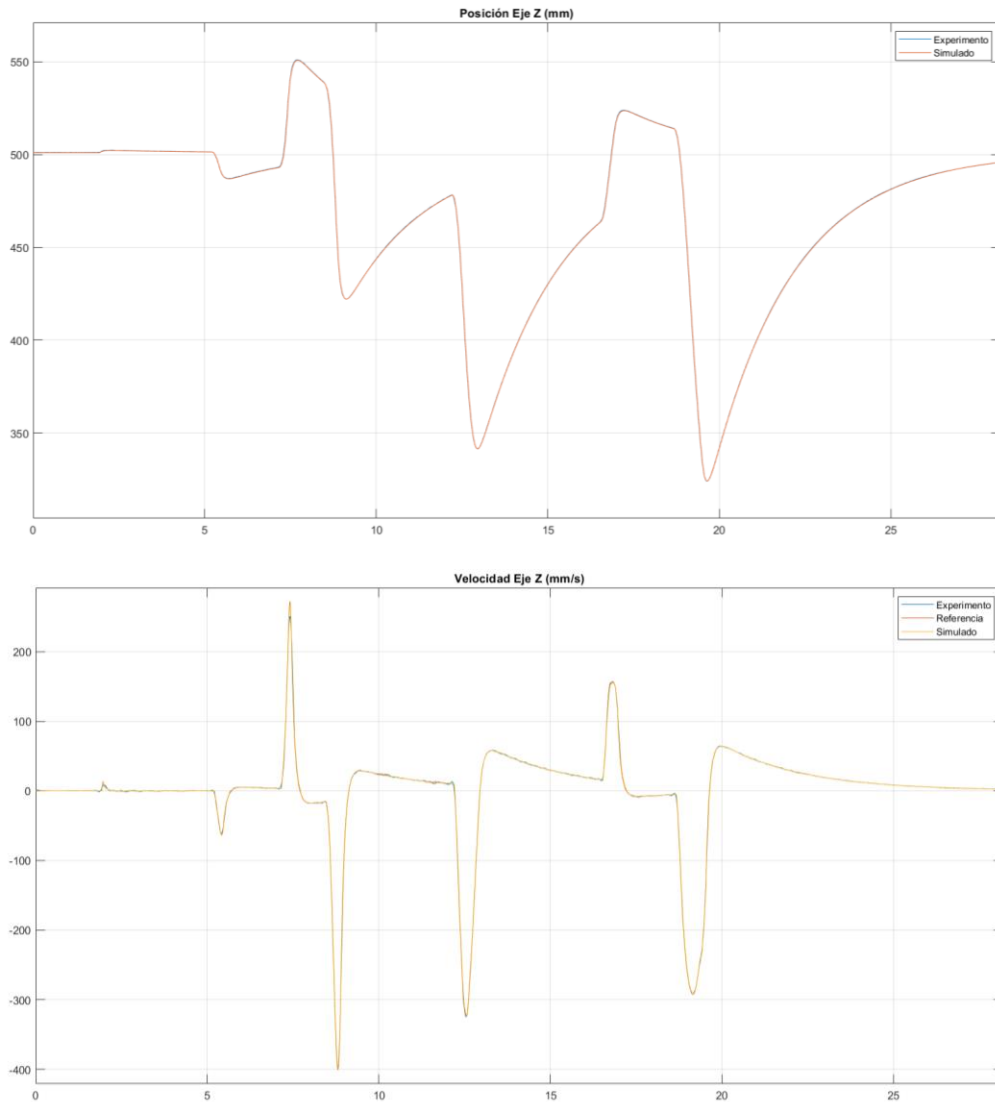


1_3 CMIA

Dado que en este caso se tiene a un sistema sobre amortiguado ($\zeta = 2.5$), los movimientos que debe realizar el robot son considerablemente más lentos en comparación a los experimentos anteriores ($\omega_n = 2 \text{ rad/s}$), por lo que le es más sencillo seguir la referencia de velocidad, permitiendo que este pueda reproducir, prácticamente a la perfección, el comportamiento deseado.



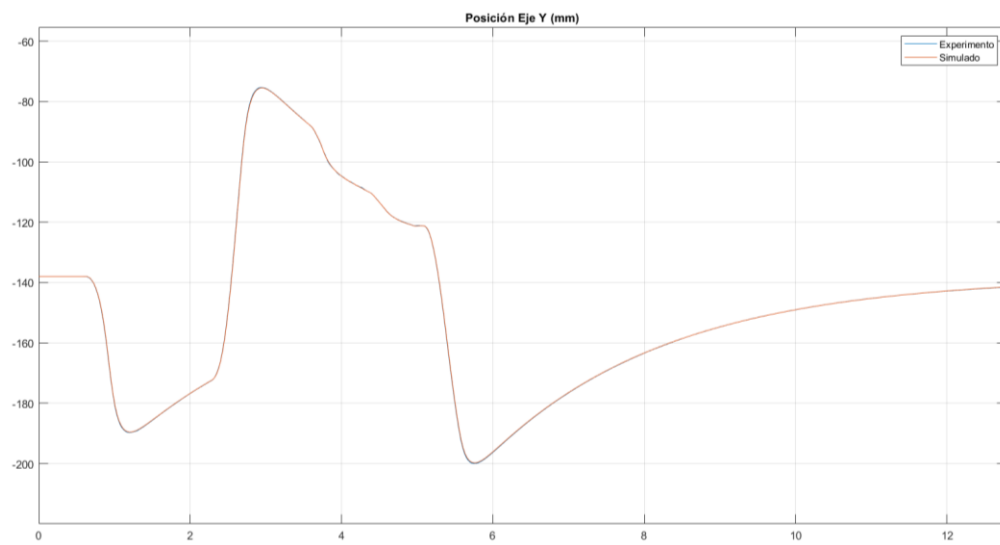
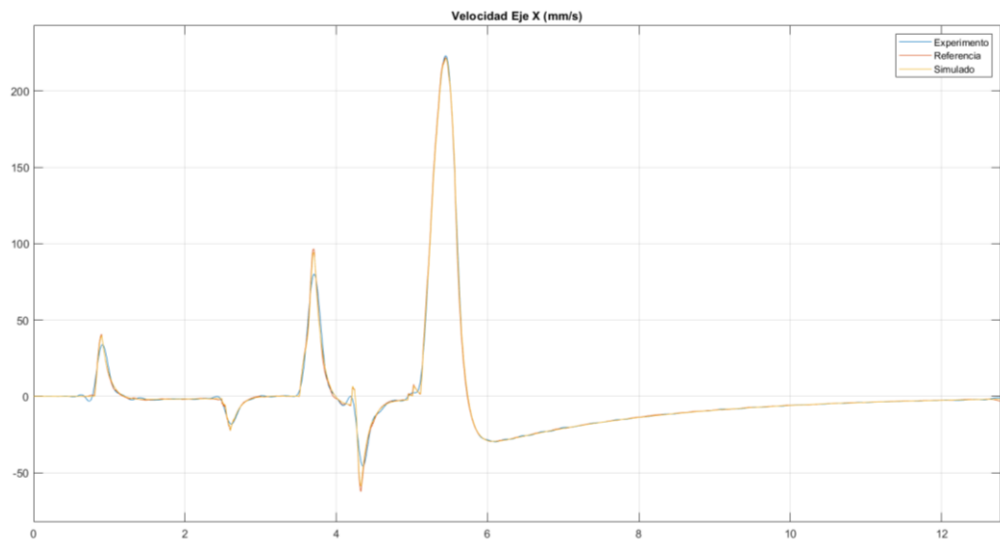
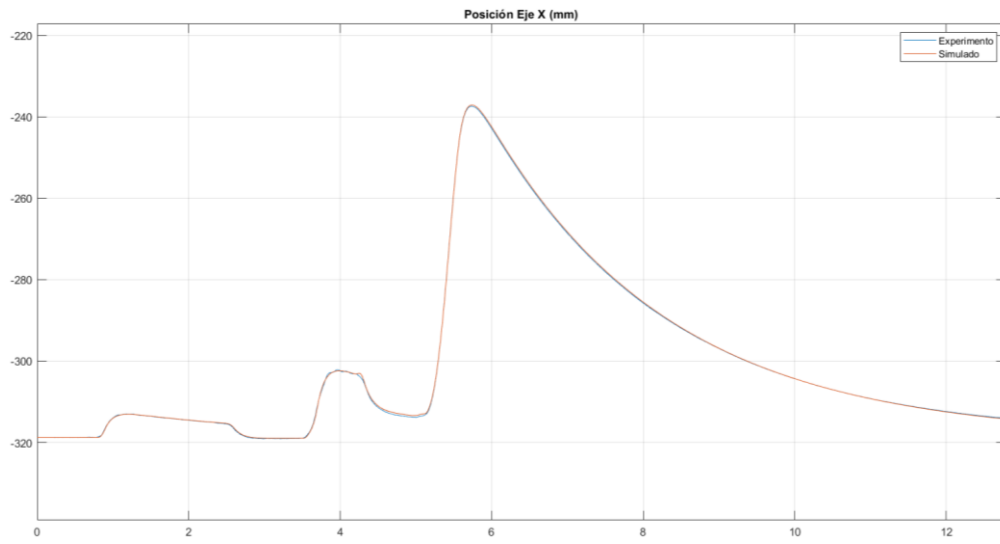


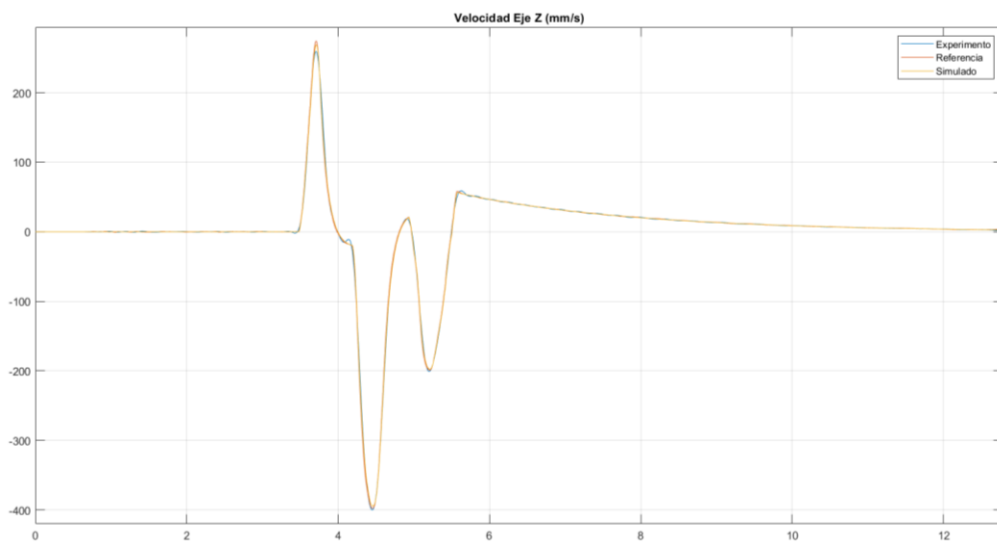
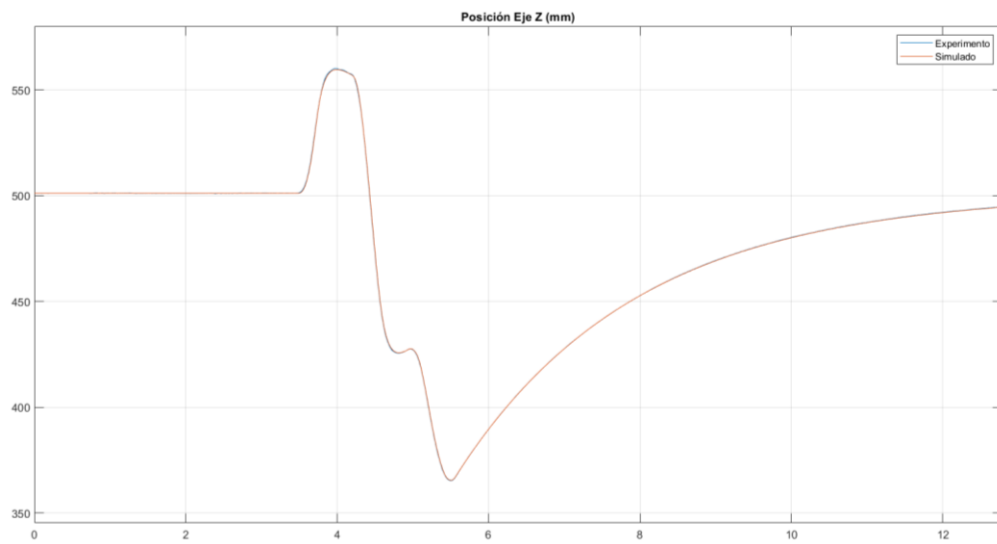
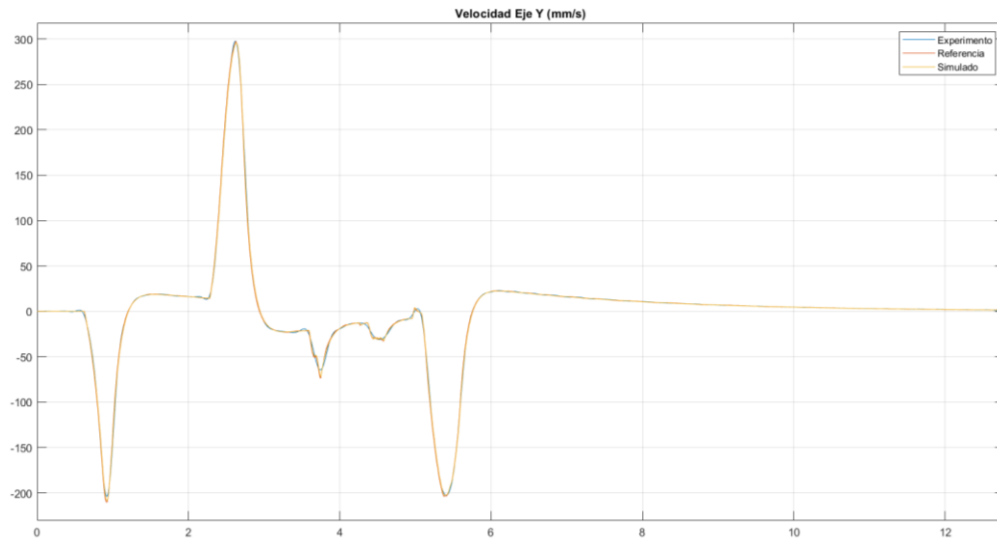


2_3 CMIA

Puesto que, de por sí el robot ejecuta correctamente los movimientos y reproduce adecuadamente la dinámica deseada para el sistema, el efecto de la acción “feed-forward” es insignificante.

Es interesante notar que, a pesar de que la referencia de velocidad está de acuerdo con el modelo, el robot parece tener dificultades para seguirla e incluso presenta pequeñas oscilaciones (sobre todo en el eje X). Sin embargo, es capaz de alcanzar la posición esperada y reproducir correctamente el comportamiento deseado.





Resumen: comparación del controlador CMIA con y sin acción “feed-forward”

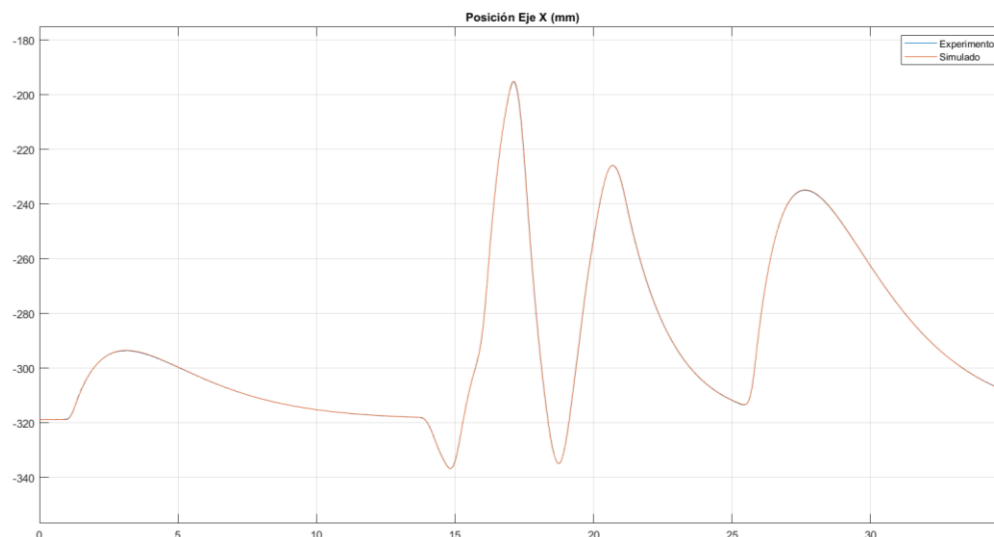
A continuación, se presentan los errores cuadráticos obtenidos de los experimentos. Como se puede apreciar, la diferencia entre utilizar o no la acción “feed-forward” no es tan significativa como en el caso del controlador SMIA, donde la aceleración utilizada era un factor limitante. No obstante, al evaluar los resultados de la Tabla 5.9, es posible observar que el uso de esta acción mejora, aunque poco, la respuesta del sistema, haciendo que corresponda aún más al modelo ideal (aunque no en todos los casos).

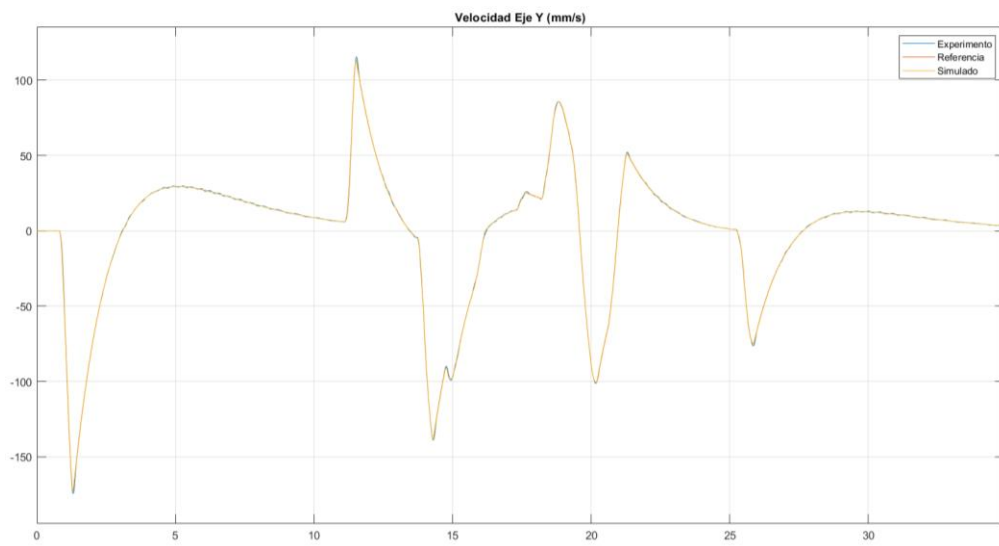
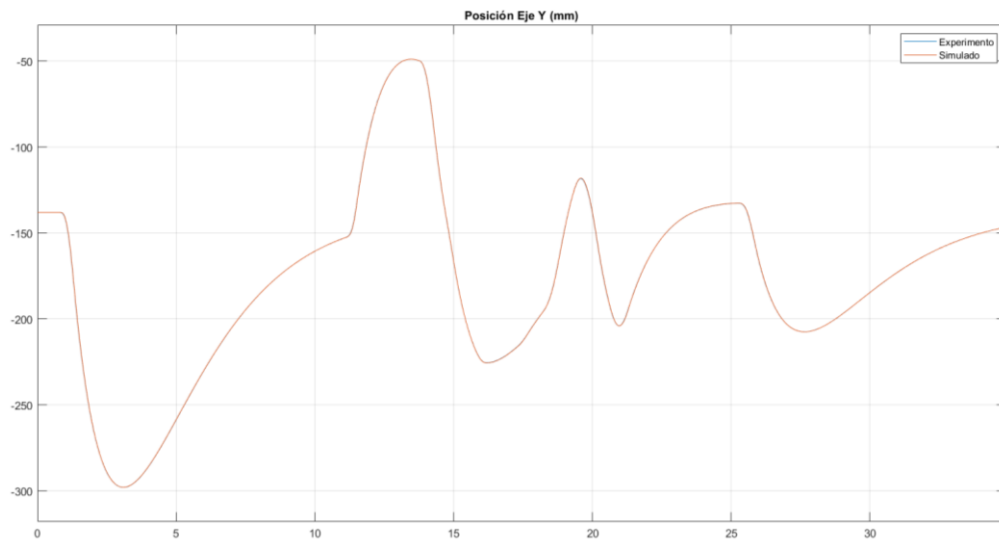
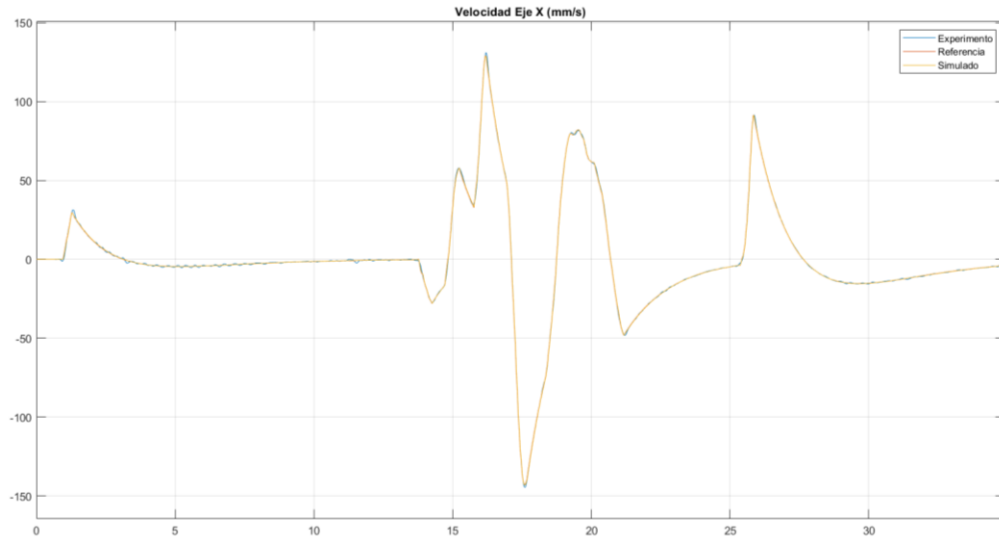
Tabla 5.9: Error cuadrático con y sin acción “feed-forward” con el controlador CMIA.

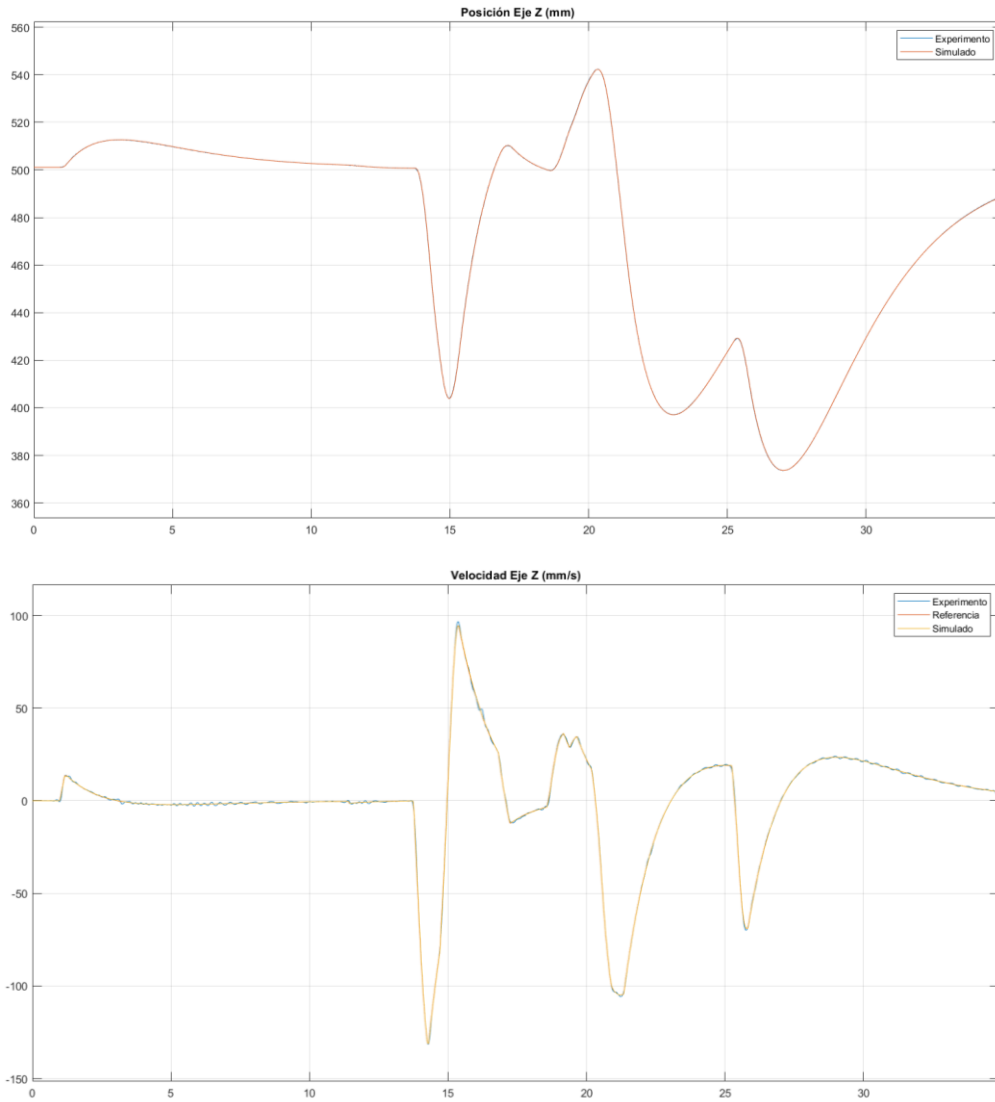
Experimento	X	Y	Z	Vx	Vy	Vz	Vref_x	Vref_y	Vref_z
1_1 CMIA	0.05	0.13	0.07	3.50	1.70	1.71	3.48	2.66	3.52
1_2 CMIA	0.16	0.21	0.24	9.06	8.01	6.69	9.12	11.24	11.00
1_3 CMIA	0.04	0.06	0.06	2.53	2.24	2.13	2.53	2.89	2.79
2_1 CMIA	0.06	0.06	0.13	2.95	2.59	3.74	1.58	2.05	3.63
2_2 CMIA	0.56	0.68	0.53	12.74	18.99	16.98	9.85	16.79	15.59
2_3 CMIA	0.04	0.02	0.04	3.97	1.76	4.16	0.85	1.76	3.28

1 CMIA_Crit

Debido a que la inercia del sistema es bastante alta, la frecuencia natural del mismo es baja ($\omega_n = 0.5 \text{ rad/s}$), lo que da como resultado un sistema lento. De esta manera, dado que el sistema es tan lento, el robot puede seguir a la perfección la referencia de velocidad, obteniendo la respuesta deseada para el sistema.

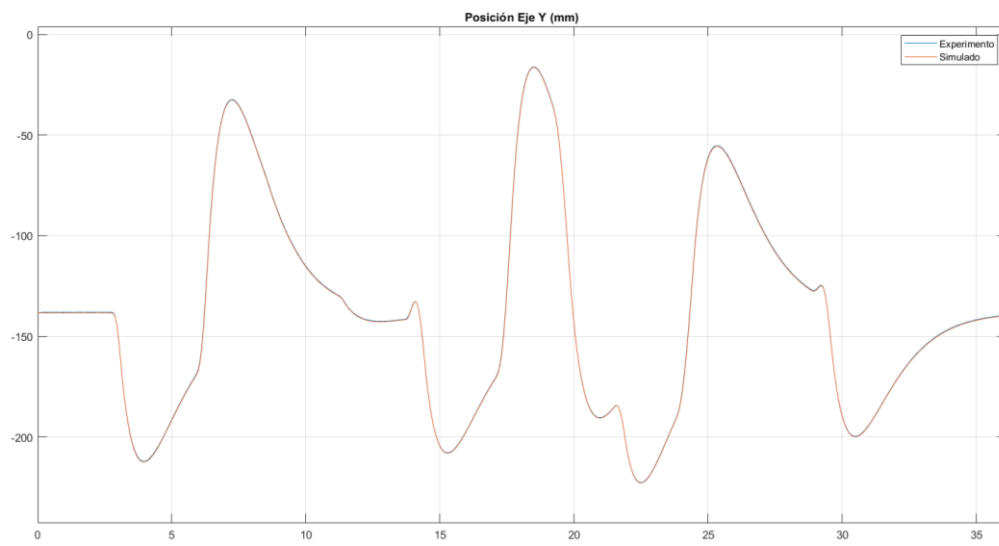
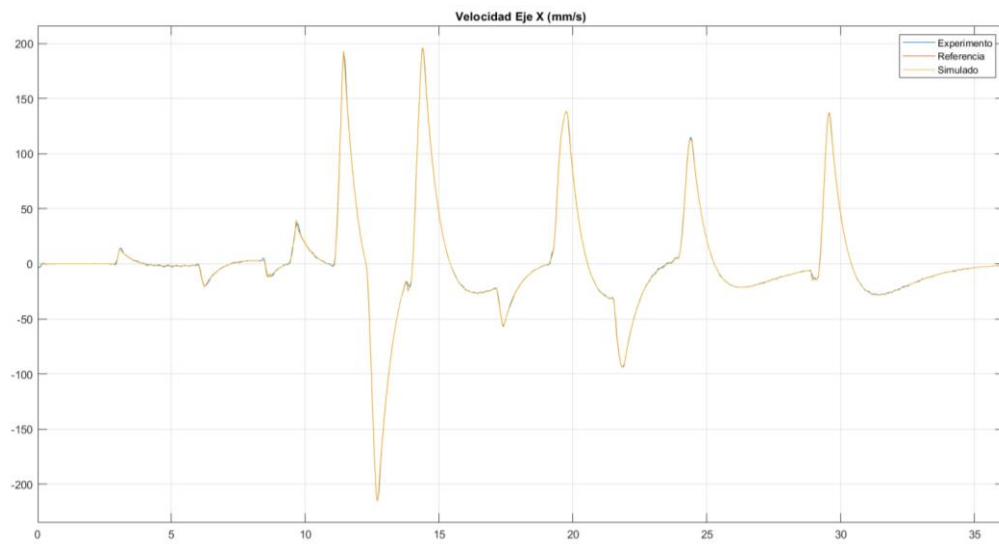
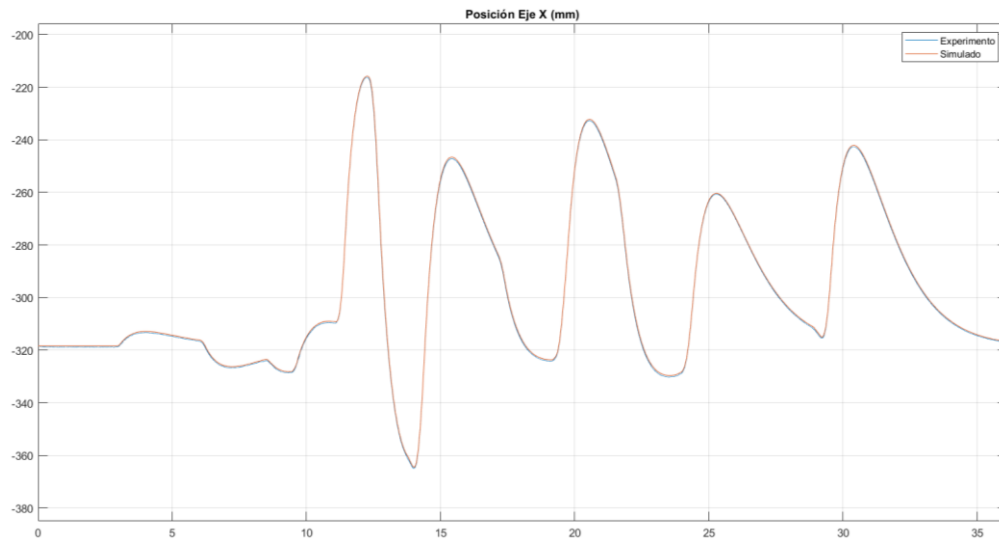


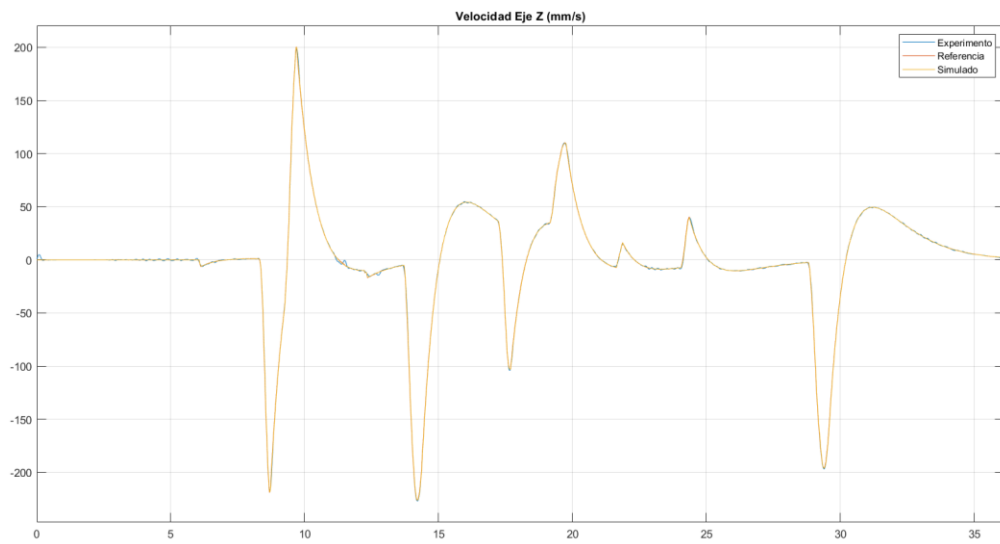
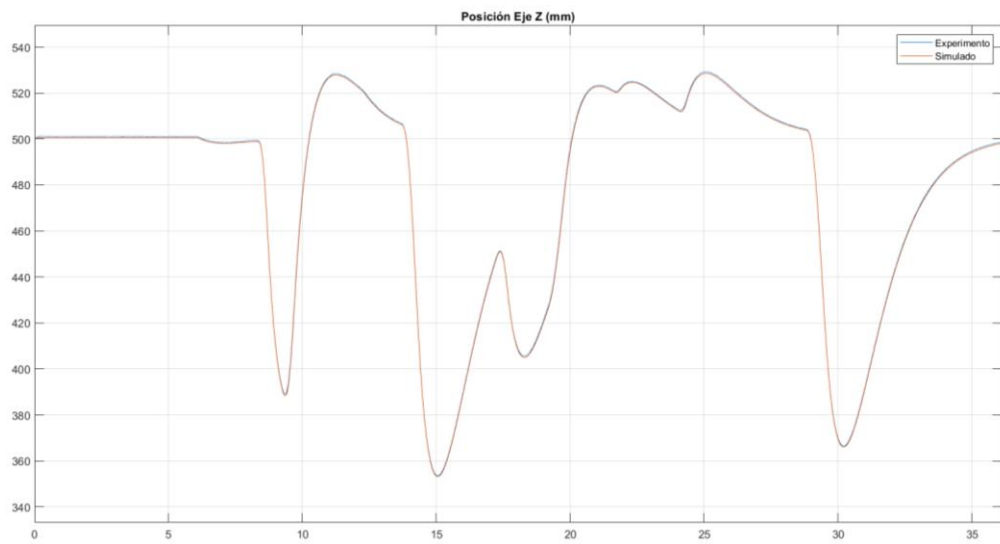
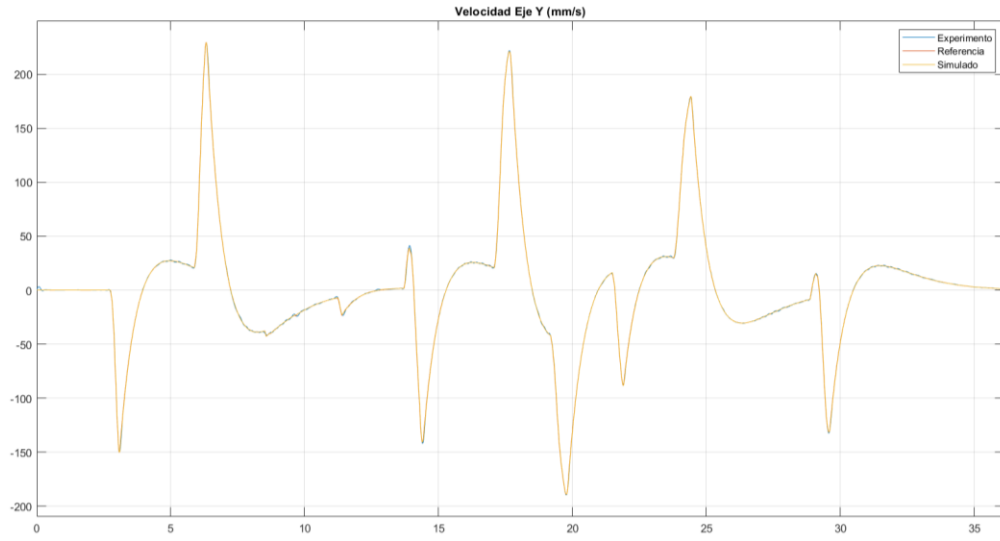




2 CMIA_Crit

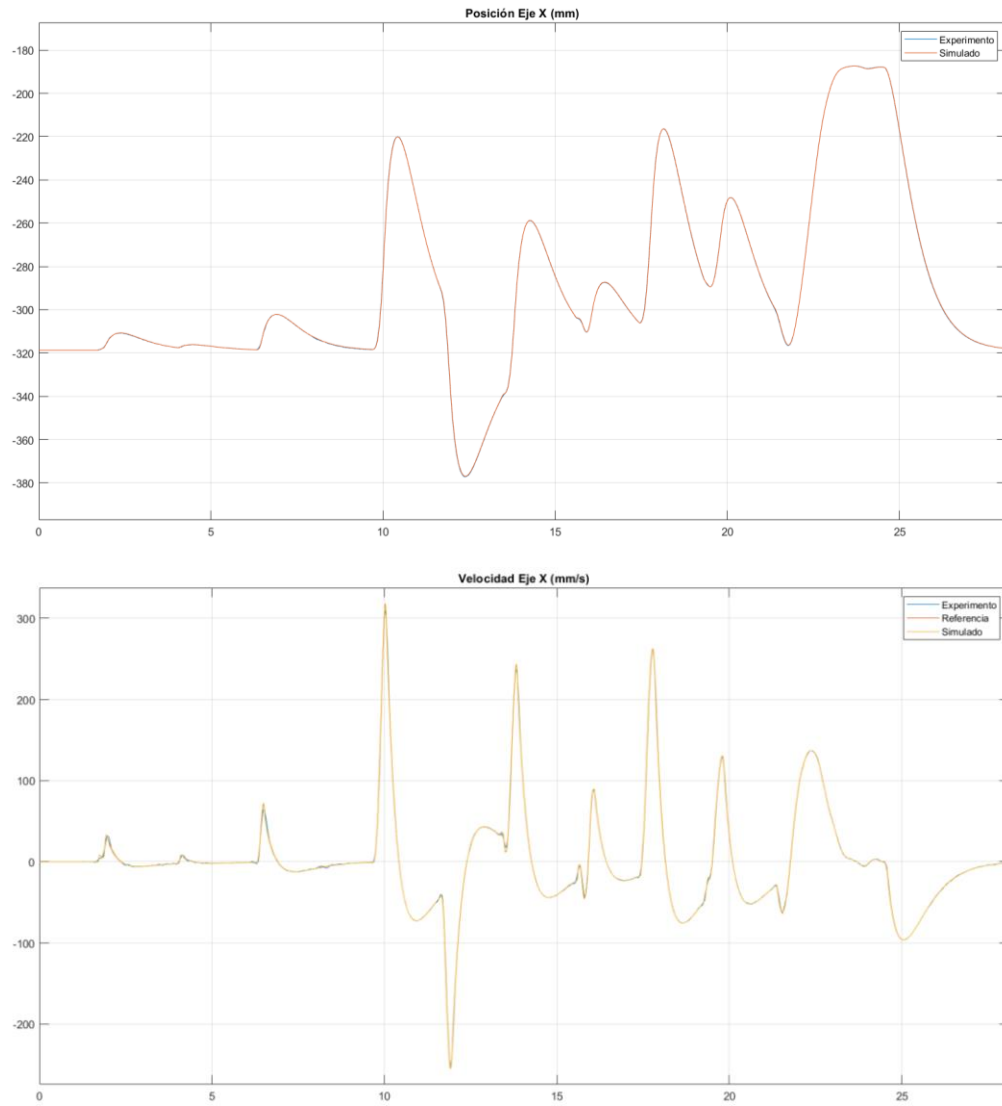
Al duplicar la frecuencia natural del sistema (ahora $\omega_n = 1 \text{ rad/s}$) el robot aun puede seguir correctamente la referencia de velocidad, por lo que el resultado sigue siendo satisfactorio. No obstante, se puede observar una muy ligera diferencia entre la posición del robot y la del modelo en ciertas ocasiones.

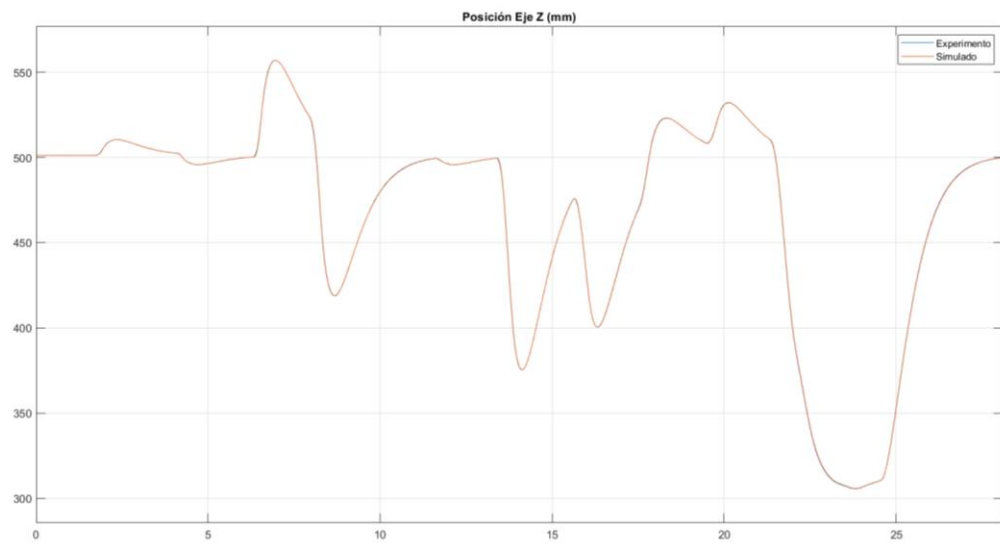
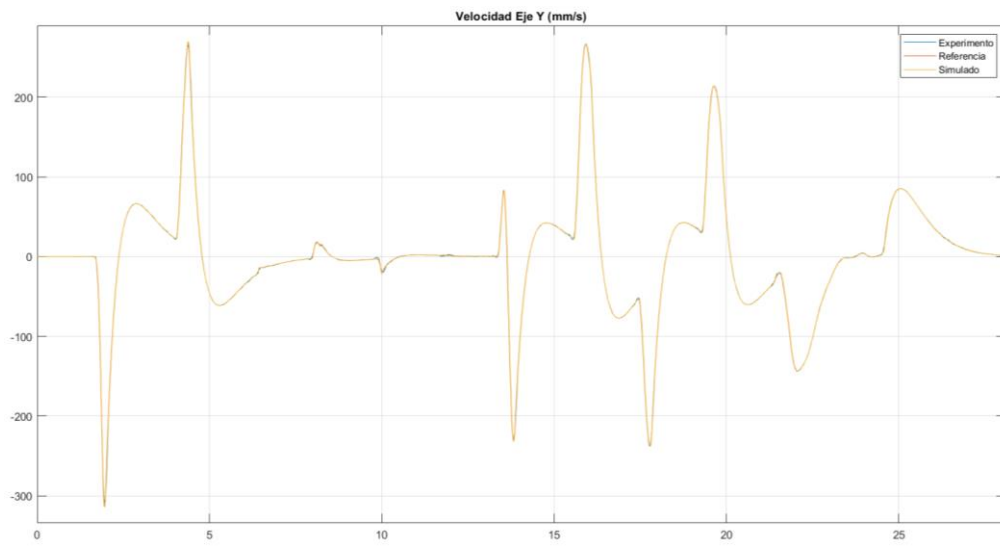
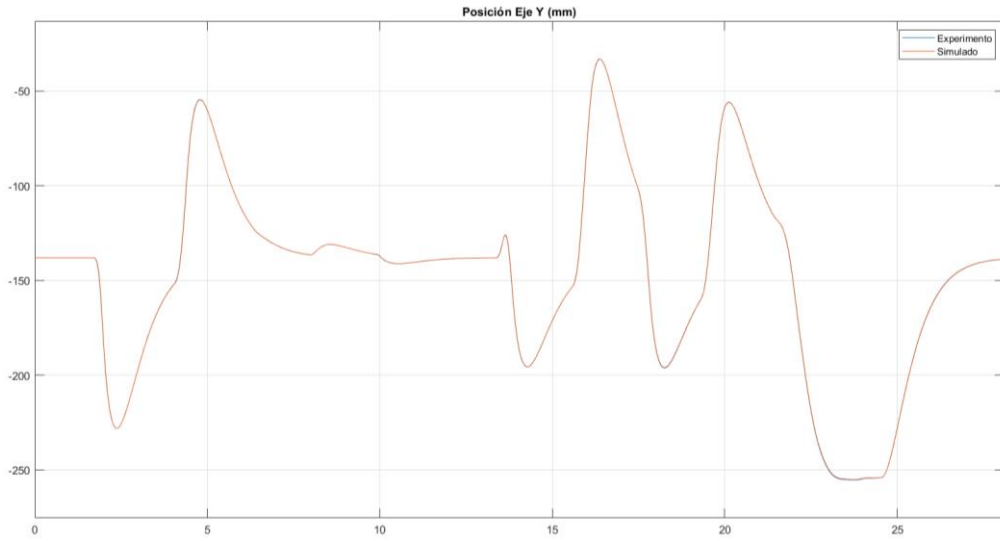


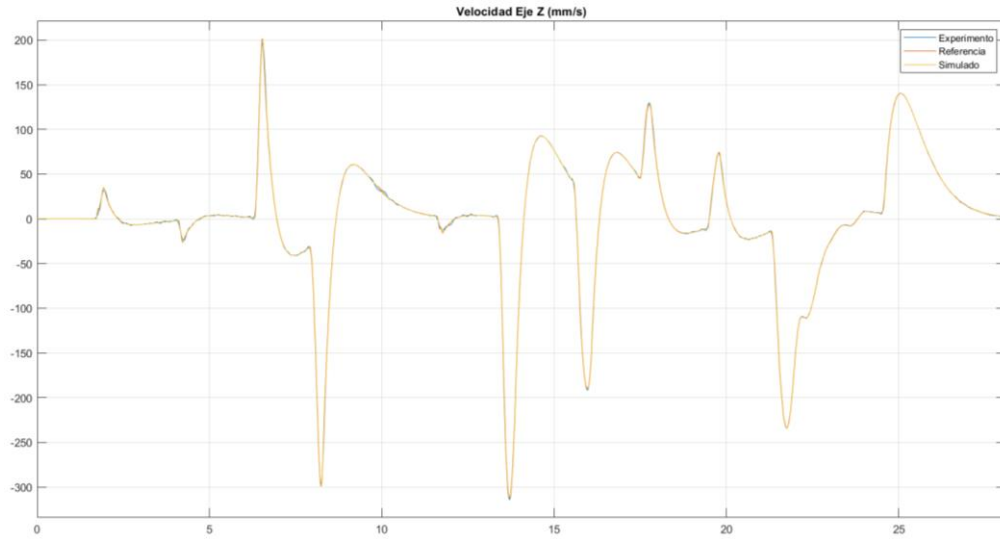


3 CMIA_Crit

Al duplicar una vez más la frecuencia natural del sistema (ahora $\omega_n = 2 \text{ rad/s}$), el robot aun es capaz de ejecutar los movimientos necesarios para reproducir el comportamiento deseado. Curiosamente, en este caso la posición del robot corresponde de forma precisa con la del modelo.



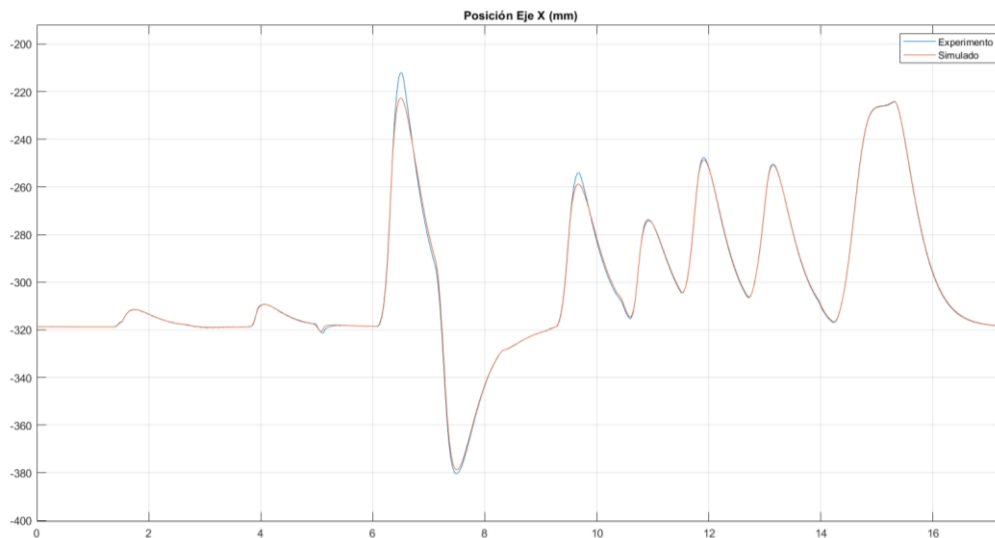


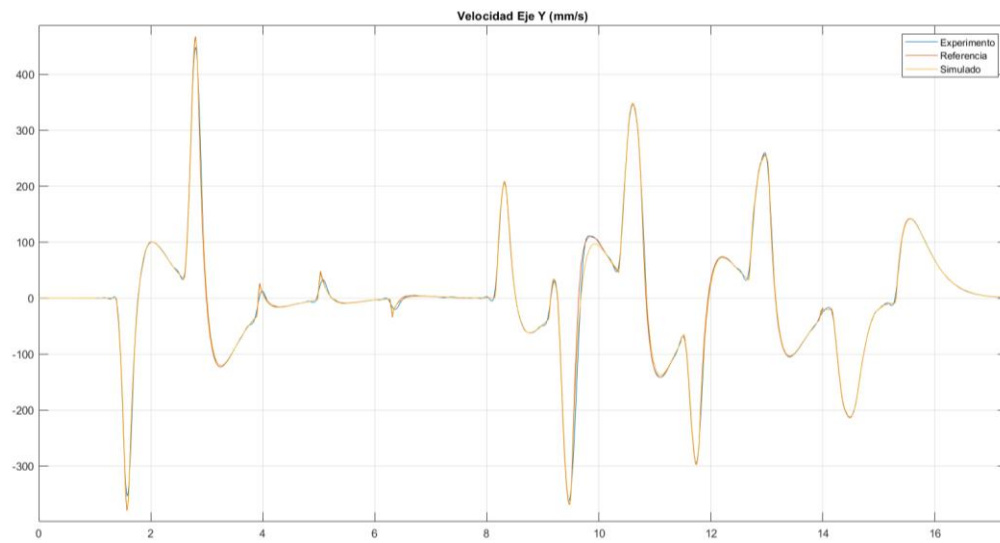
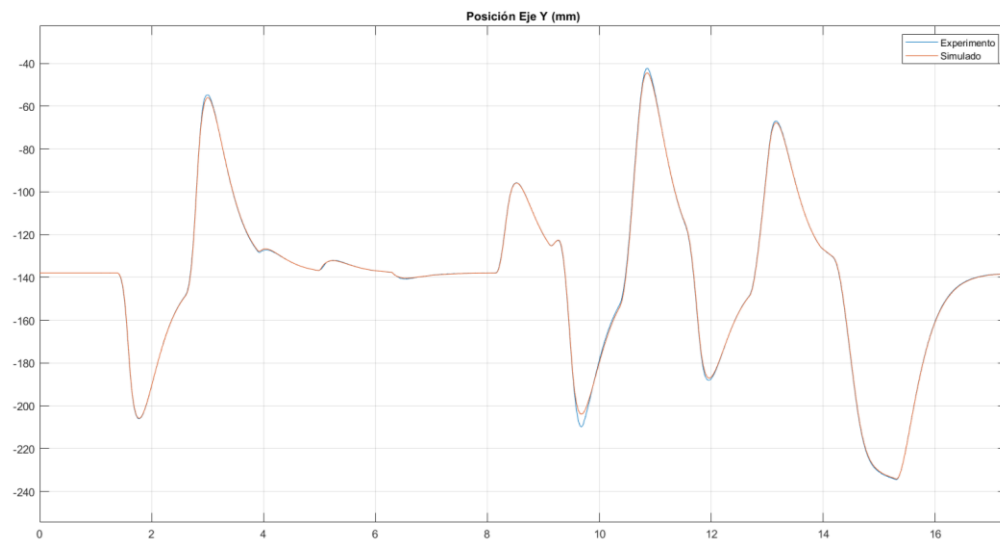
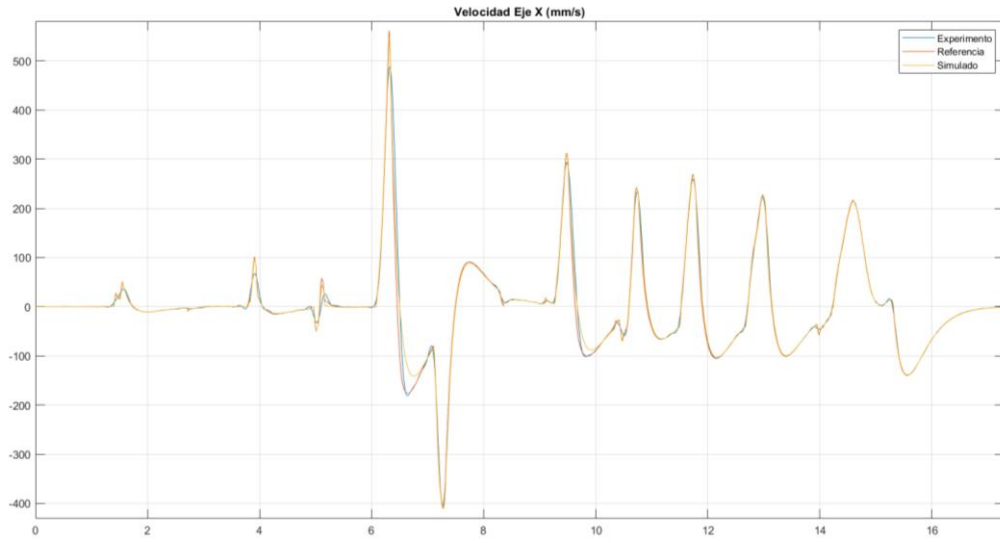


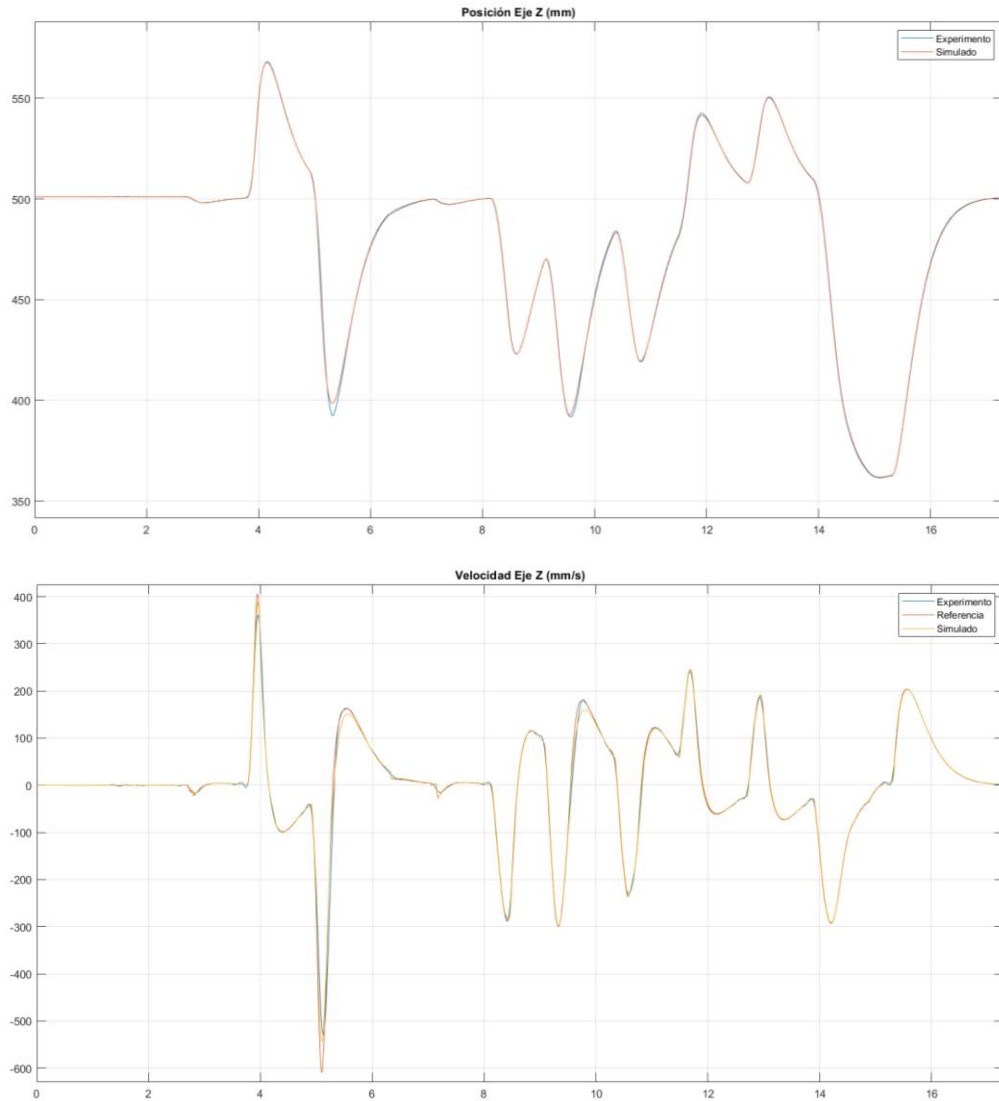
4 CMIA_Crit

En este experimento el robot comienza a mostrar cierta dificultad para estimar correctamente la referencia de velocidad, por lo que termina moviéndose un poco más rápido de lo que debería, lo que a su vez hace que este se desplace un poco más de lo esperado. Sin embargo, al observar las gráficas parece ser que la influencia de la fuerza externa sobre el efector final no se está calculando adecuadamente, lo que da lugar a este comportamiento.

Cabe señalar que en este caso también se duplicó la frecuencia natural del sistema (ahora $\omega_n = 4 \text{ rad/s}$) y el robot fue capaz de seguir la referencia de velocidad adecuadamente.

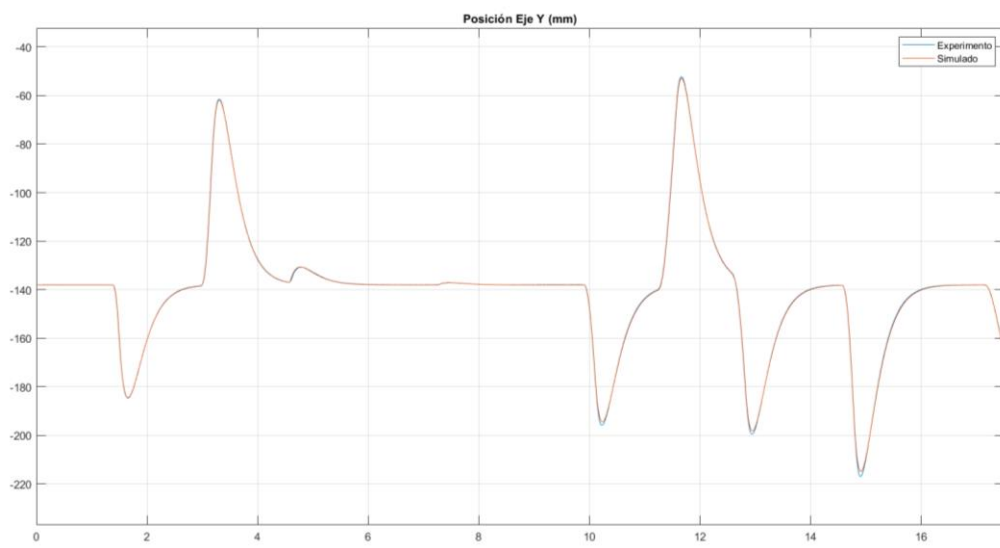
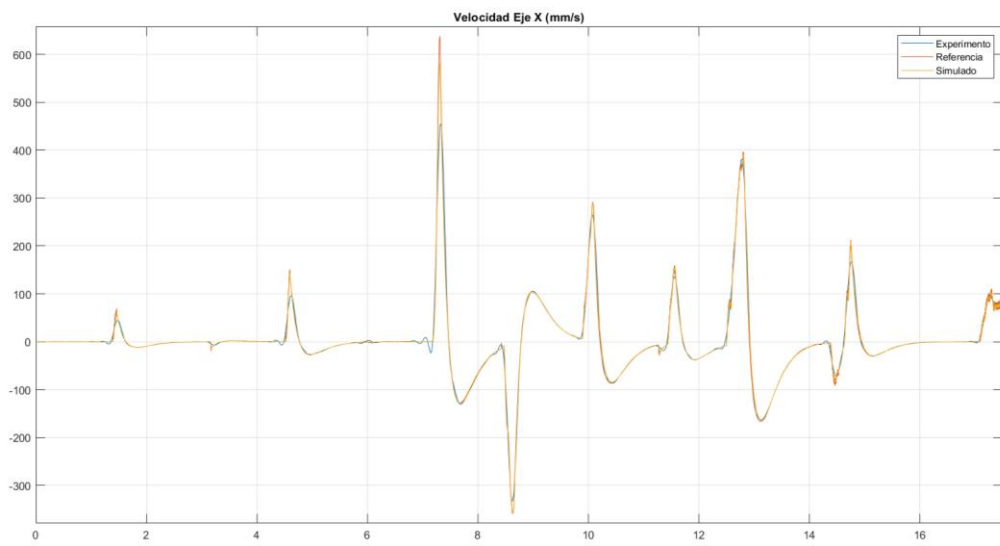
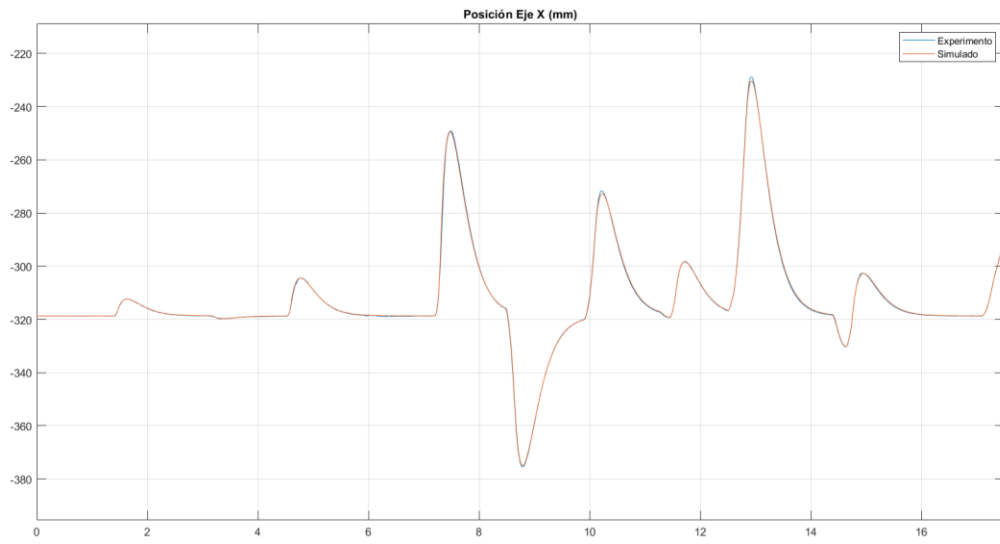


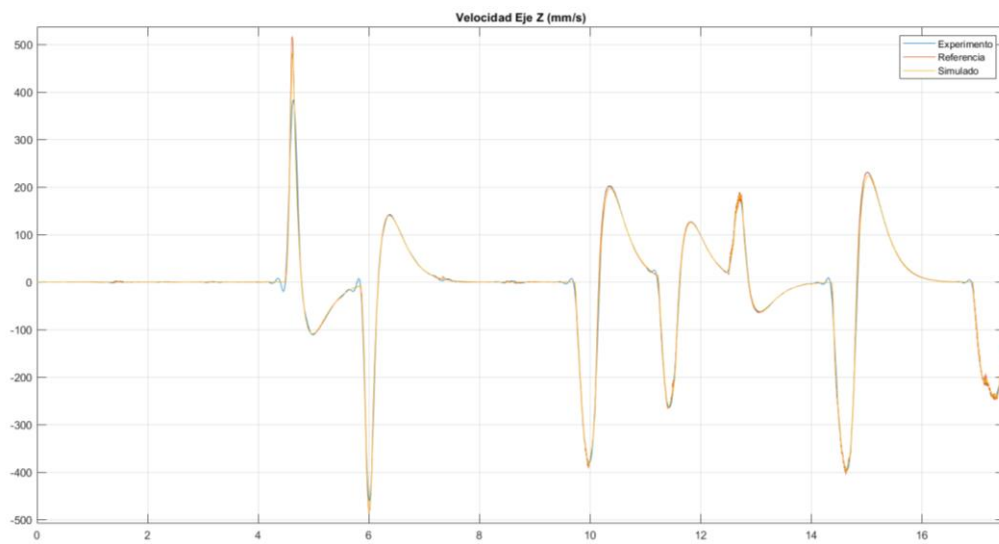
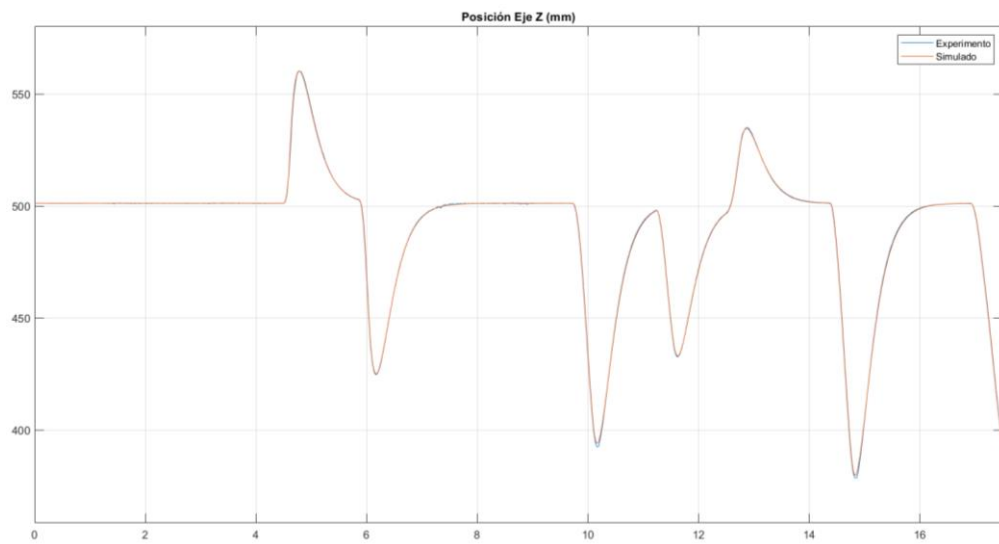
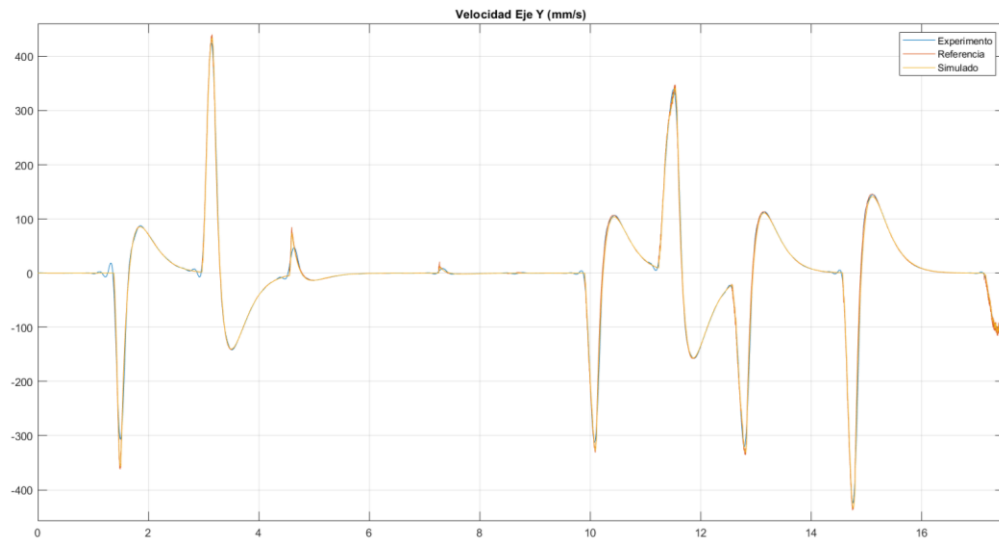




5 CMIA_Crit

En este caso la frecuencia natural del sistema se estableció en $\omega_n = 5 \text{ rad/s}$, por lo que el robot se movía rápidamente. Ahora, si bien era sencillo mover el robot debido a la baja inercia ($m = 1 \text{ kg}$), interactuar con este era considerablemente más complicado que en los casos anteriores debido a la sensibilidad que exhibía. De esta forma, se tuvo especial cuidado al interactuar con este con el fin de poder obtener buenos resultados del experimento. Así, es posible apreciar como el robot fue capaz de mostrar el comportamiento deseado al seguir correctamente la referencia de velocidad (en la mayoría de los casos).





Resumen: controlador CMIA con sistemas críticamente amortiguados

Así como en los experimentos anteriores, en este caso se presenta la Tabla 5.10 con los errores cuadráticos para cada uno. Como se puede apreciar, el caso con el sistema más lento y la mayor inercia (1 CMIA_Crit) es prácticamente igual al sistema ideal.

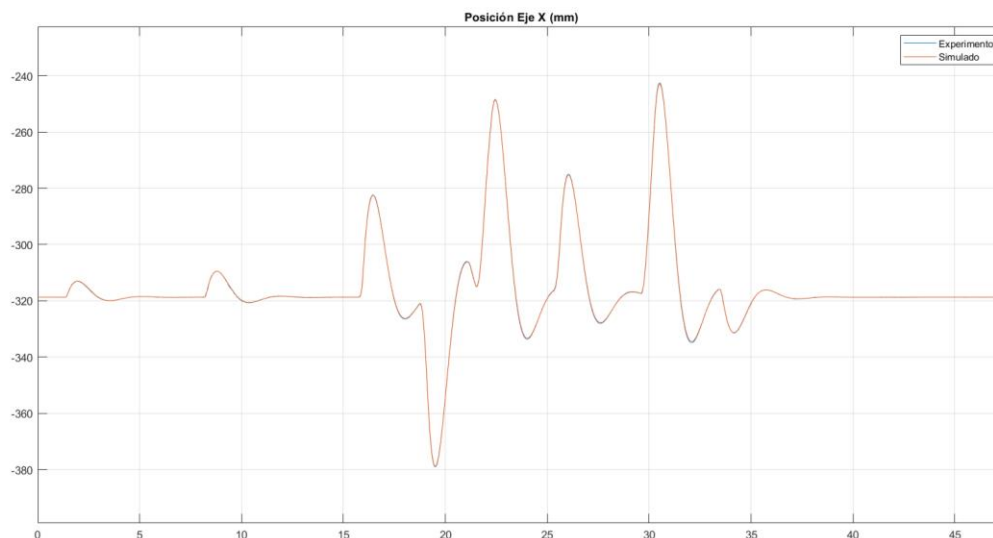
Tabla 5.10: Error cuadrático de sistemas críticamente amortiguados con el controlador CMIA.

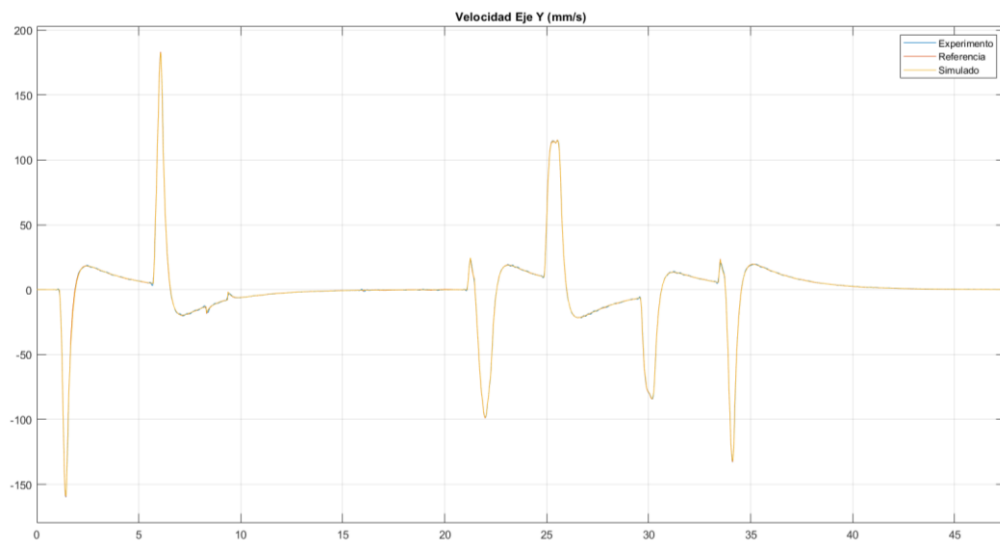
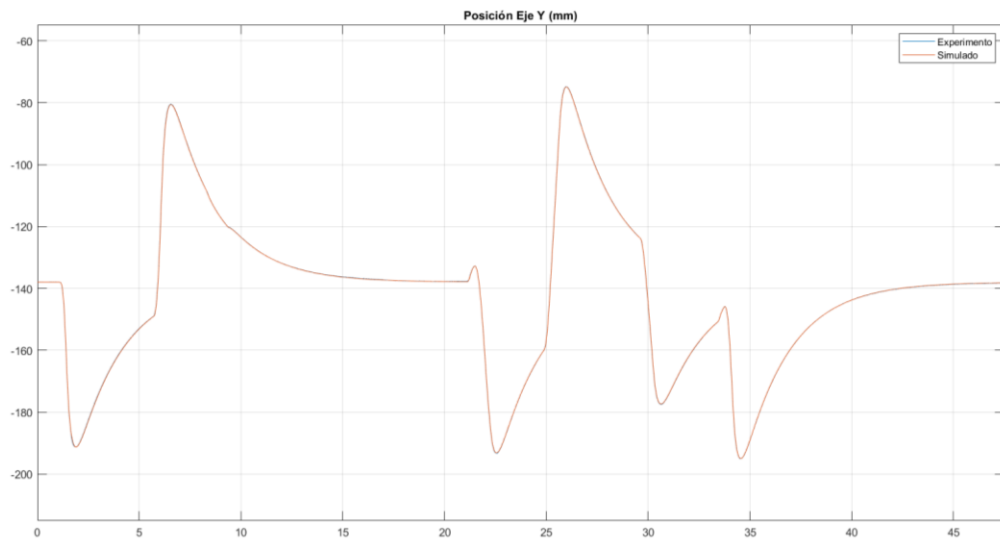
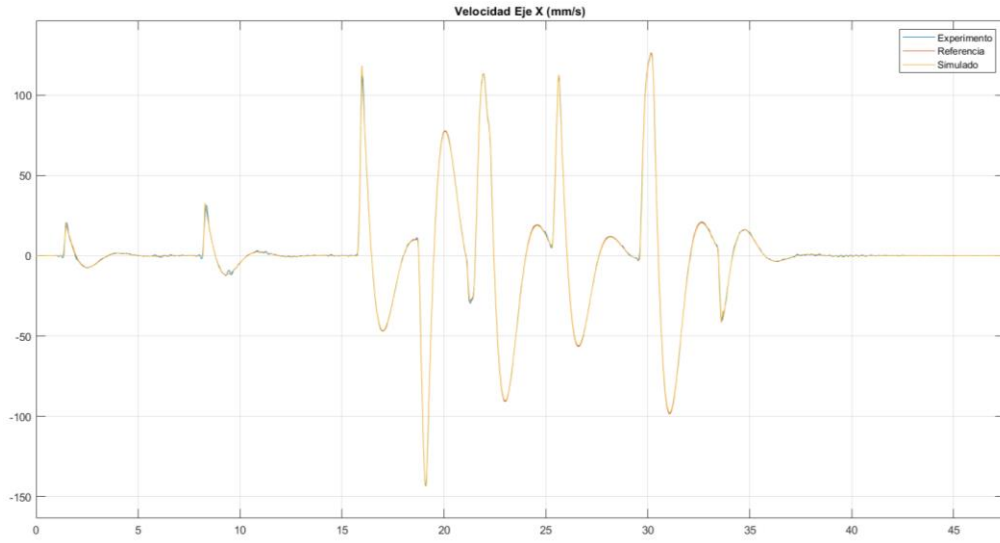
Experimento	X	Y	Z	Vx	Vy	Vz	Vref_x	Vref_y	Vref_z
1 CMIA_Crit	0.02	0.02	0.01	0.27	0.24	0.30	0.17	0.22	0.14
2 CMIA_Crit	0.22	0.13	0.28	0.47	0.38	0.54	0.54	0.67	0.64
3 CMIA_Crit	0.04	0.05	0.07	1.50	0.99	1.03	1.99	2.47	1.95
4 CMIA_Crit	1.57	0.53	0.72	137.4	39.10	97.51	142.4	46.99	63.71
5 CMIA_Crit	0.16	0.13	0.18	106.9	36.69	63.39	34.14	27.78	29.84

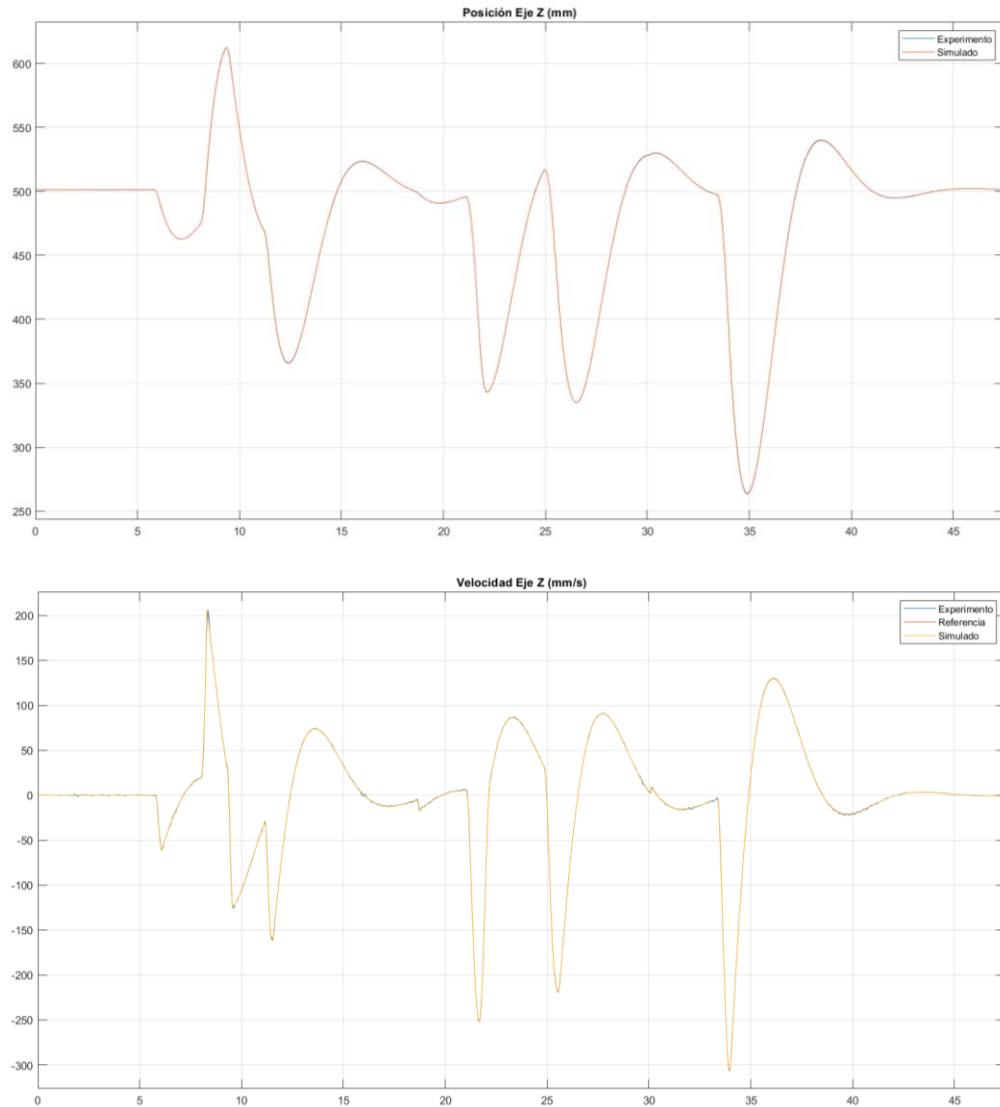
1 CMIA_Dif_P

En este experimento se puede observar claramente la independencia de los movimientos de cada eje, lo cual ayuda a reforzar la idea de que no existe un acoplamiento entre los movimientos del robot en diferentes ejes. En este caso, los ejes X y Z exhiben un comportamiento sub amortiguado ($\zeta_x = 0.45$, $\zeta_z = 0.5$), mientras que el eje Y presenta un comportamiento sobre amortiguado ($\zeta_y = 1.77$). Así mismo, cada eje tiene una frecuencia natural distinta $\omega_x = 2.24 \text{ rad/s}$, $\omega_y = 1.41 \text{ rad/s}$ y $\omega_z = 1 \text{ rad/s}$.

Es interesante observar como el comportamiento del robot coincide perfectamente con el modelo, tanto en la velocidad, como en la posición del mismo.



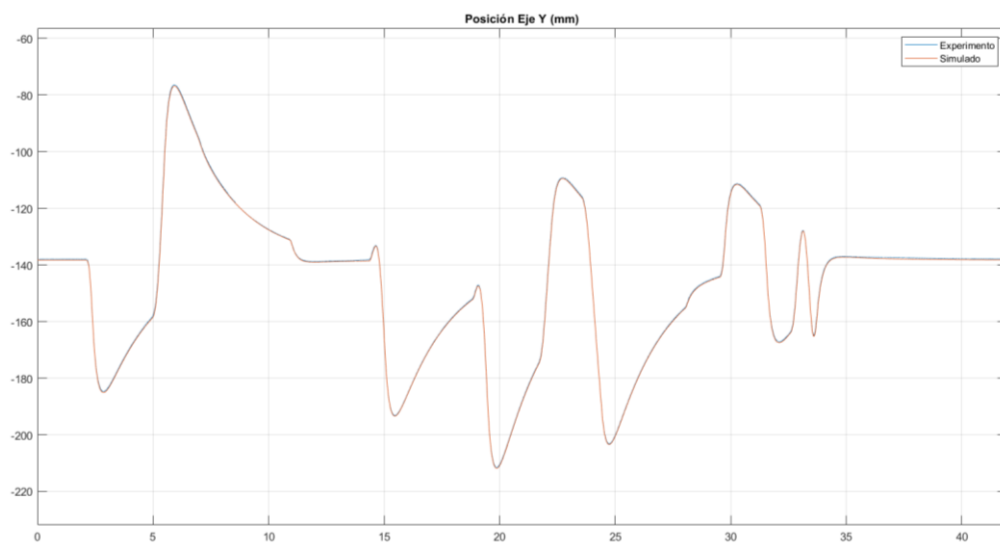
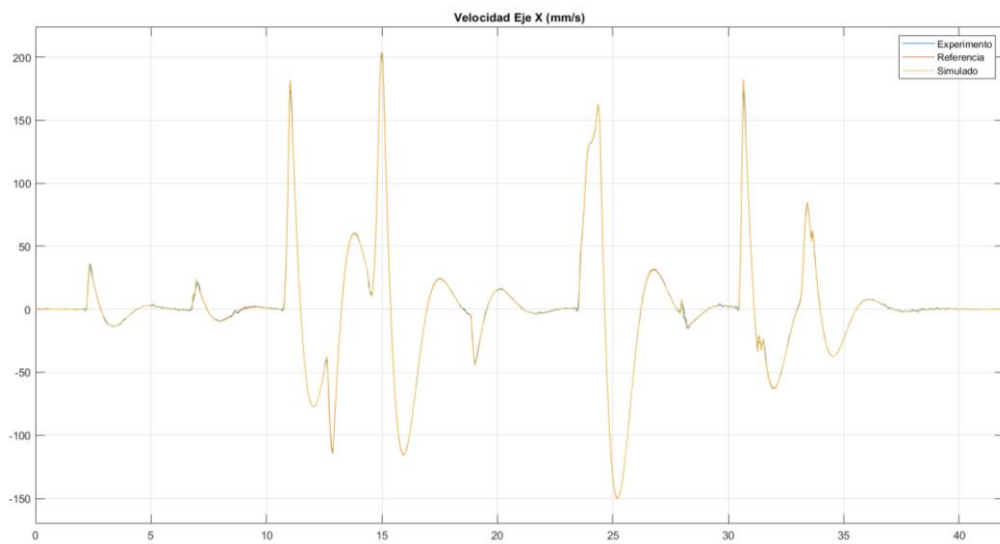
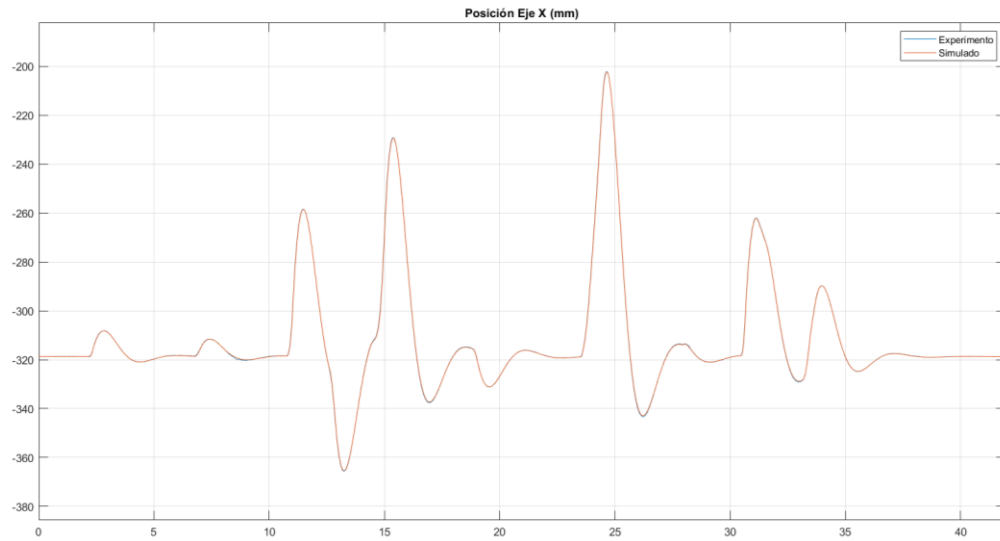


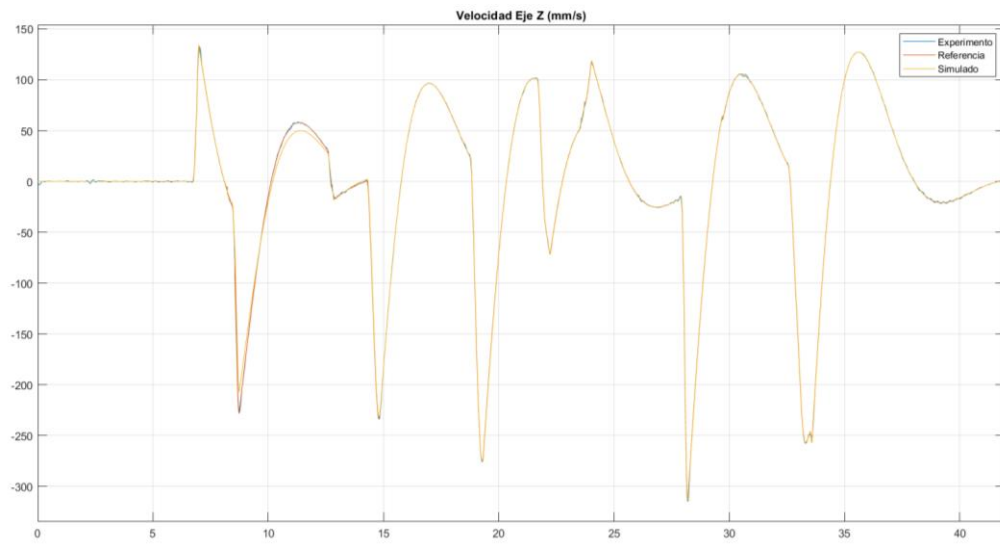
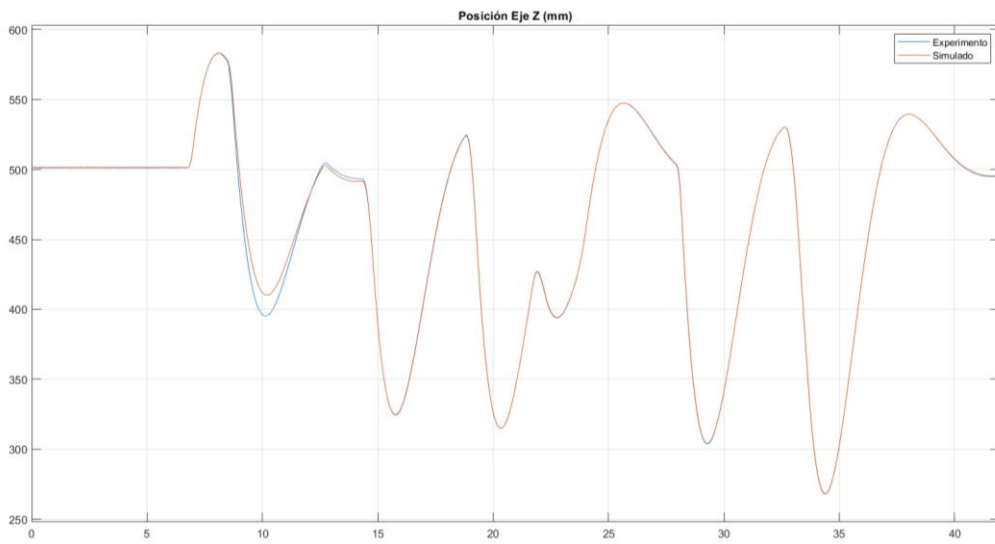
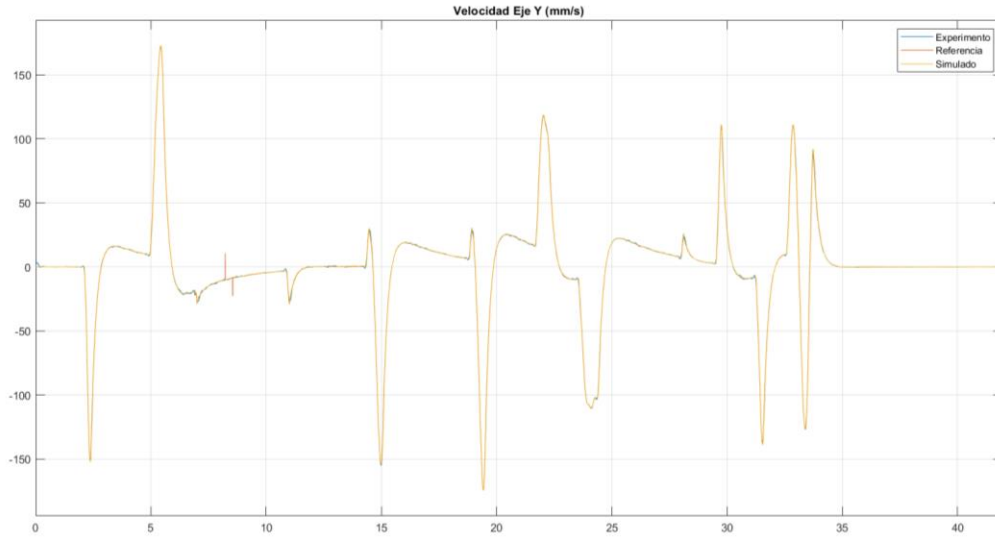


2 CMIA_Dif_P

En este caso es importante notar que la posición en el eje Z presenta una zona donde el comportamiento del robot difiere en gran medida del modelo, por lo que esto podría deberse a la interacción del robot con un elemento muy rígido. Además, en la gráfica de velocidad es posible observar como la referencia se desvía del modelo por unos instantes justo antes de que la posición del robot se vea afectada. Entonces, agregando el hecho de que durante el resto del experimento el robot siga correctamente al modelo, es muy probable que no haya sido un fallo del robot, sino de la interacción con el mismo.

A parte de lo explicado anteriormente, es posible apreciar que el robot es capaz de reproducir el comportamiento deseado sin problemas.





Resumen: sistemas diferentes en cada eje

A pesar de utilizar diferentes parámetros para cada eje, el controlador es capaz de ejecutar los movimientos necesarios para reproducir el comportamiento en cada uno de ellos con gran precisión. Esto se puede apreciar en la Tabla 5.11, donde, dejando de lado el incidente con el eje Z en el segundo experimento, los resultados demuestran la precisión del sistema para reproducir el comportamiento esperado. Así mismo, es posible observar como la acción “feed-forward” (utilizada en 1 CMIA_Crit) ayuda a hacer aún más preciso al sistema.

Tabla 5.11: Error cuadrático de sistemas diferentes en cada eje con el controlador CMIA.

<i>Experimento</i>	<i>X</i>	<i>Y</i>	<i>Z</i>	<i>Vx</i>	<i>Vy</i>	<i>Vz</i>	<i>Vref_x</i>	<i>Vref_y</i>	<i>Vref_z</i>
<i>1 CMIA_Crit</i>	0.02	0.01	0.04	0.42	0.22	0.41	0.16	0.27	0.25
<i>2 CMIA_Crit</i>	0.03	0.11	11.17	0.85	0.46	9.20	0.66	0.93	9.93

CONCLUSIONES

Se han desarrollado dos controladores de admitancia que, con la adecuada elección de la magnitud de la aceleración para los movimientos del robot, son capaces de reproducir el comportamiento deseado prácticamente a la perfección. Por otro lado, dado que el controlador SMIA modifica la rigidez y el amortiguamiento, manteniendo intacta la inercia aparente del robot, su implementación resulta sencilla. Mientras que, debido a que el controlador CMIA, además de modificar estos parámetros, es capaz de modificar la inercia aparente del sistema, presenta una implementación bastante más compleja. De esta manera, es posible apreciar la complejidad que trae consigo modificar la inercia aparente del robot, sin mencionar los problemas de estabilidad que podría presentar debido a esto al disminuir demasiado la inercia.

Además, ha sido posible comprobar, tanto matemáticamente (tomando algunas consideraciones) como mediante experimentos, que el UR3e se puede tratar como un robot rígido debido a las características mecánicas que presenta, simplificando en gran medida los cálculos y disminuyendo el tiempo de cómputo necesario para resolver las ecuaciones de la dinámica del mismo. Adicionalmente, también se ha comprobado que los efectos físicos no lineales (como la fricción seca o la fricción viscosa) no tienen un efecto significativo sobre los movimientos del robot, por lo que estos pueden ser despreciados en este tipo de aplicaciones. Aún más, debido a que esto es posible, y dado que el controlador del UR3e compensa la gravedad, no es necesario compensar los efectos dinámicos al imponer la dinámica deseada sobre el robot, lo cual simplifica bastante el diseño y la implementación de los controladores.

Por otra parte, se ha podido comprobar la efectividad de utilizar una acción “feed-forward” en el controlador, la cual permite utilizar valores de aceleración menores para los movimientos del robot, mejorando así la estabilidad del sistema. Este efecto es evidente en las gráficas y la tabla de los experimentos con el controlador SMIA, donde el uso de esta acción logra compensar la falta de aceleración en los movimientos del robot, mejorando con creces el comportamiento del mismo. Sin embargo, el efecto de esta acción no es tan evidente cuando la aceleración del robot es suficiente para ejecutar los movimientos (como en el caso del controlador CMIA), por lo que se podría prescindir de ella. Aun así, al utilizar la acción “feed-forward” a pesar de que el robot pueda ejecutar correctamente los movimientos sin ella, este suele presentar un comportamiento aún más cercano al del sistema ideal.

Finalmente, el hecho de poder predecir de antemano las características que presentará el robot al implementar el controlador de admitancia (siempre y cuando se conozca la fuerza de interacción) y, el hecho de no tener que compensar la dinámica del robot, representa una gran ventaja, puesto que el sistema puede ser simulado de forma relativamente sencilla (y por lo tanto rápida) en un entorno virtual, permitiendo que se evalúen distintas combinaciones de parámetros con el fin de obtener la que mejor se adapte a los requerimientos del sistema. Además, dado que el modelo es el de un sistema masa-resorte-amortiguador, sus características son ampliamente conocidas.

OBSERVACIONES Y RECOMENDACIONES

Luego de observar las gráficas presentadas en los experimentos *Oscilatorio*, *Osci_XYZ*, *1 CMIA_Dif_P* y *2 CMIA_Dif_P*, es posible deducir que no existe un acoplamiento (o al menos no uno significativo) entre los movimientos, puesto que las respuestas no varían entre un experimento y otro, por lo que se puede decir que los movimientos en cada eje son independientes. De esta forma, es perfectamente factible implementar uno de estos controladores en un eje y tener otros tipos de controladores en los otros ejes para controlar el robot de la forma que mejor se adapte a la tarea que debe realizar.

En ciertos casos, el modelo implementado en el UR3e muestra una diferencia notoria respecto al modelo ideal. Esto puede ser debido a la forma en la que el controlador de admitancia responde ante ser sostenido rígidamente, como se vio en algunos de los artículos presentados en el capítulo 2. De esta forma, la respuesta del controlador influye al hacer que el robot ejerza una fuerza no deseada, haciendo que el humano aplique una fuerza para contrarrestarla, lo que hace que el modelo estime incorrectamente la referencia de velocidad. Este efecto es más notorio en los casos en los que la frecuencia natural del sistema es más alta, y probablemente es debido a que el controlador reacciona con mayor velocidad ante la interacción con un entorno rígido, provocando que este se desplace bastante más de lo necesario.

Por otro lado, a pesar de la gran precisión que muestran los controladores para reproducir el comportamiento esperado, existen ocasiones en las que el controlador CMIA sobre estima ligeramente la velocidad del efector final, haciendo que el robot se desplace un poco más de lo deseado. Un motivo para tal comportamiento podría deberse a la aproximación realizada en el diseño del controlador.

Ahora, si bien el robot es capaz de ejecutar los movimientos necesarios para reproducir el comportamiento esperado, demuestra mejores resultados cuando los movimientos que debe realizar son lentos, por lo que es ideal para aplicaciones donde una alta velocidad no es necesaria. Sin embargo, este controlador podría implementarse en situaciones donde el robot debe moverse rápidamente, pero este no es sostenido, sino que es empujado hacia un lado.

Por otra parte, al despreciar la inercia del robot, es difícil realizar un análisis de pasividad que de como resultado un límite hasta el cual se pueda reducir la inercia aparente del sistema, por lo que, en base a lo observado en los experimentos, no es recomendable disminuir la masa deseada por debajo de los 2.5 kg, siendo recomendable mantener esta masa alrededor de los 5 kg. Esto se debe principalmente a que, con una masa por debajo de este límite, la interacción con el robot resulta complicada y deja de aportar una ventaja para el sistema.

Además, es recomendable siempre buscar que el robot presente un comportamiento críticamente amortiguado o sobre amortiguado para la interacción de este con su entorno (objetos y humanos), puesto que las oscilaciones no son deseadas en aplicaciones robóticas. Aún más, el comportamiento críticamente amortiguado es el que presenta el mejor balance entre velocidad, estabilidad e interacción, por lo que encontrar los parámetros que satisfagan los requerimientos del sistema para la tarea a realizar de tal forma que se obtenga este comportamiento es ideal.

Otro punto interesante es que, si se quita la amortiguación del sistema, este tiende a la inestabilidad. Una razón para que esto suceda es la saturación de los actuadores al tener que cambiar el sentido del movimiento en un pequeño instante de tiempo debido a las oscilaciones que presenta el sistema cuando se le aplica una fuerza. Este aspecto debería de ser estudiado en un próximo trabajo para mejorar la respuesta del sistema y asegurar su estabilidad aun en casos extremos. Puesto que, a pesar de que en este caso se está evaluando el comportamiento, en una aplicación real, el robot debe poder ejecutar el comportamiento deseado en todo momento para asegurar la seguridad del humano al interactuar con el robot o del robot al interactuar con algo inesperado en su entorno.

Finalmente, en el fichero que contiene los parámetros dinámicos del UR3e, la inercia en el eje Z del último eslabón es diferente de cero, lo que sugiere que el robot también compensa la inercia del sensor de fuerza. El hecho de compensar esta inercia ayuda a mejorar el desempeño del robot, sobre todo en este tipo de aplicaciones (como se vio en el capítulo 2).

BIBLIOGRAFÍA

- [1] P. Song, Y. Yu, and X. Zhang, “A Tutorial Survey and Comparison of Impedance Control on Robotic Manipulation,” *Robotica*, vol. 37, no. 5, pp. 801–836, May 2019, doi: 10.1017/S0263574718001339.
- [2] F. J. Abu-Dakka and M. Saveriano, “Variable Impedance Control and Learning—A Review,” *Frontiers in Robotics and AI*, vol. 7. Frontiers Media S.A., Dec. 21, 2020. doi: 10.3389/frobt.2020.590681.
- [3] A. Q. L. Keemink, H. van der Kooij, and A. H. A. Stienen, “Admittance control for physical human–robot interaction,” *International Journal of Robotics Research*, vol. 37, no. 11, pp. 1421–1444, Sep. 2018, doi: 10.1177/0278364918768950.
- [4] A. Zelenak, M. W. Pryor, R. G. Reid, A. Pettinger, and M. Pryor, “Reactive Motion Control for Real-Time Teleoperation and Semi-Autonomous Contact Tasks,” 2019, doi: 10.13140/RG.2.2.18571.03362.
- [5] H. Chen, J. Li, W. Wan, Z. Huang, and K. Harada, “Integrating combined task and motion planning with compliant control: Successfully conducting planned dual-arm assembly motion using compliant peg-in-hole control,” *Int J Intell Robot Appl*, vol. 4, no. 2, pp. 149–163, Jun. 2020, doi: 10.1007/s41315-020-00136-1.
- [6] Y. Lai, S. Sutjipto, M. D. Clout, M. G. Carmichael, and G. Paul, “GAVRe2: Towards Data-driven Upper-Limb Rehabilitation with Adaptive-Feedback Gamification,” in *International Conference on Robotics and Biomimetics*, 2018, pp. 1–6.
- [7] G. Ning, X. Zhang, and H. Liao, “Autonomic Robotic Ultrasound Imaging System Based on Reinforcement Learning,” *IEEE Trans Biomed Eng*, vol. 68, no. 9, pp. 2787–2797, Sep. 2021, doi: 10.1109/TBME.2021.3054413.
- [8] C. T. Landi, F. Ferraguti, L. Sabattini, C. Secchi, and C. Fantuzzi, “Admittance control parameter adaptation for physical human-robot interaction,” in *Proceedings - IEEE International Conference on Robotics and Automation*, Jul. 2017, pp. 2911–2916. doi: 10.1109/ICRA.2017.7989338.
- [9] F. Ferraguti, C. Talignani Landi, L. Sabattini, M. Bonfè, C. Fantuzzi, and C. Secchi, “A variable admittance control strategy for stable physical human–robot interaction,” *International Journal of Robotics Research*, vol. 38, no. 6, pp. 747–765, May 2019, doi: 10.1177/0278364919840415.
- [10] G. Kang, H. S. Oh, J. K. Seo, U. Kim, and H. R. Choi, “Variable Admittance Control of Robot Manipulators Based on Human Intention,” *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 3, pp. 1023–1032, Jun. 2019, doi: 10.1109/TMECH.2019.2910237.
- [11] H. Tugal, B. Gautier, M. Kircicek, and M. Suphi Erden, “Hand-Impedance Measurement During Laparoscopic Training Coupled with Robotic Manipulators,” in *International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 1–7.

- [12] C. Yang, G. Peng, Y. Li, R. Cui, L. Cheng, and Z. Li, "Neural networks enhanced adaptive admittance control of optimized robot-environment interaction," *IEEE Trans Cybern*, vol. 49, no. 7, pp. 2568–2579, Jul. 2019, doi: 10.1109/TCYB.2018.2828654.
- [13] C. C. Beltran-Hernandez *et al.*, "Learning Force Control for Contact-Rich Manipulation Tasks with Rigid Position-Controlled Robots," *IEEE Robot Autom Lett*, vol. 5, no. 4, pp. 5709–5716, Oct. 2020, doi: 10.1109/LRA.2020.3010739.
- [14] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Advanced Textbooks in Control and Signal Processing*. 2009.
- [15] A. de Luca and W. J., *Handbook of Robotics - 11. Robots with Flexible Elements*. 2016.
- [16] R. Pérez-Ubeda, R. Zotovic-Stanisic, and S. C. Gutiérrez, "Force control improvement in collaborative robots through theory analysis and experimental endorsement," *Applied Sciences (Switzerland)*, vol. 10, no. 12, Jun. 2020, doi: 10.3390/app10124329.
- [17] R. Monsalve, G. Rubert, and S. Carlos, "ESTUDIO DE LA APLICACIÓN DE ROBOTS COLABORATIVOS AL FRESADO," 2022.