



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

COMPARACIÓN DE UN CONTROLADOR CLÁSICO
Y UN CONTROLADOR DISCONTINUO EN UN
SISTEMA SOBREACTUADO

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

Ingeniero Eléctrico - Electrónico

PRESENTA:

Ruiz Gasca Marco Antonio

DIRECTOR DE TESIS:

M.I. Rafael Iriarte Vivar Balderrama



2012

*Dedicado a:
mi familia, hermanas, Estefanía y Casandra, Padres, Aurora y Reyes
amigos y compañeros, Mildred, Cuauhtémoc, Silvia, Alan, Juan Carlos, Fernando,
Sebastian, Thod y una larga lista.
A mis asesores, Liset, Rafael
Al equipo de investigación del grupo de modos deslizantes, Toño, Hector, Marco Tulio,
Leonid, Roger, Antonio.*

Contenido

Lista de figuras	vii
Lista de tablas	xi
1 Introducción	1
1.1 Presentación	1
1.2 Objetivo	2
1.3 Motivación y estado del arte	2
1.3.1 Estado del arte	2
1.3.2 Motivación	2
1.4 Planteamiento del problema	5
1.5 Metodología	6
1.6 Contribuciones	8
1.7 Estructura de la tesis	9
2 Teoría básica empleada	11
2.1 Ecuaciones de Lagrange	11
2.1.1 Formulación básica	11
2.2 Controladores PD	13
2.3 Introducción a los Modos Deslizantes	14
2.3.1 Sistemas de Control de Estructura Variable	16
2.3.2 Control equivalente	18
3 Modelo matemático del robot paralelo	23
3.1 Descripción de la planta	23
3.2 Modelo matemático del robot	25

CONTENIDO

3.2.1	Cinemática del robot paralelo	26
3.2.2	Modelo dinámico: Cadena cinemática abierta	27
3.2.3	Modelo dinámico sin fuerzas de restricción	29
3.2.4	Modelo dinámico del robot paralelo con fuerzas de restricción	29
3.2.5	Modelo en espacio de estados	34
4	Control del Robot Paralelo	35
4.1	Planteamiento del problema	35
4.1.1	Objetivo de control	36
4.2	Estructura de control	36
4.3	Estrategia de control	38
4.4	Control nominal	38
4.5	Control No Lineal	39
4.5.1	Robustificación del controlador	39
4.5.2	Planteamiento del problema	40
4.5.3	Diseño del controlador	40
5	Resultados	43
5.1	Parámetros dinámicos usados	43
5.2	Resultados de Simulaciones	44
5.2.1	Simulación trayectoria recta con PD nominal	45
5.2.2	Simulación trayectoria circular con PD nominal	48
5.2.3	Simulación trayectoria recta con PD+Twisting	50
5.2.4	Simulación trayectoria circular con PD+Twisting	55
5.3	Resultados experimentales	62
5.3.1	Experimento trayectoria recta con PD nominal	62
5.3.2	Experimento trayectoria circular con PD nominal	65
5.3.3	Experimento trayectoria recta con PD+Twisting	67
5.3.4	Experimento trayectoria circular con PD+Twisting	70
5.3.5	Experimentos con diferentes parámetros	72
6	Conclusiones	75
6.1	Conclusiones generales	75
6.2	Comparación de resultados basados en RSME	76

A	79
A.1 Tarjeta GT-400-SV	79
A.2 Sistema de Servo Amplificadores PY2	81
A.3 Entorno de programación C++	82
A.4 MATLAB Simulink	84
A.5 Calibración de los encoders absolutos	84
Bibliografía	87

CONTENIDO

Lista de figuras

1.1	Dispositivo de manipulación de alta precisión	3
1.2	Microensamblado	3
1.3	Trípode	4
1.4	Precisión	5
1.5	Robustez	6
1.6	Estructura del sistema	7
1.7	Metodología de programación	8
2.1	PD	14
2.2	Plano de fase del sistema	17
2.3	Plano de fase del modo deslizante	18
2.4	Plano de fase	19
2.5	Acción de Control	20
3.1	Robot paralelo 1	24
3.2	Robot paralelo 2	25
3.3	Parámetros geométricos del robot	26
3.4	Cadena cinemática abierta	27
3.5	Efector final del robot dentro del espacio de trabajo	32
3.6	Efector final fuera del espacio de trabajo	33
4.1	Control nominal	37
4.2	Control aumentado	37
4.3	Estructura del sistema	39
5.1	Error en la coordenada x del efector final, control PD trayectoria recta .	45

LISTA DE FIGURAS

5.2	Error en la coordenada y del efector final, control PD trayectoria recta .	46
5.3	Error de posición actuador 1 con PD trayectoria recta	47
5.4	Error de posición actuador 2 con PD trayectoria recta	47
5.5	Error de posición actuador 3 con PD trayectoria recta	48
5.6	Error en la coordenada x del efector final en trayectoria circular	49
5.7	Error en la coordenada y del efector final en trayectoria circular	49
5.8	Error de posición actuador 1 con PD trayectoria circular	50
5.9	Error de posición actuador 2 con PD trayectoria circular	51
5.10	Error de posición actuador 3 con PD con PD trayectoria circular	51
5.11	Error en la coordenada x del efector final en trayectoria recta	52
5.12	Error en la coordenada y del efector final en trayectoria recta	53
5.13	Error en la posición angular 1 con PD+Twisting, trayectoria recta	54
5.14	Error en la posición angular 2 con PD+Twisting, trayectoria recta	54
5.15	Error en la posición angular 3 con PD+Twisting, trayectoria recta	55
5.16	Control en actuador 1 con PD+Twisting, trayectoria recta	56
5.17	Control en actuador 2 con PD+Twisting, trayectoria recta	56
5.18	Control en actuador 3 con PD+Twisting, trayectoria recta	57
5.19	Error en la coordenada x del efector final en trayectoria circular	57
5.20	Error en la coordenada y del efector final en trayectoria circular	58
5.21	Error en la posición angular 1 con PD+Twisting, trayectoria circular	59
5.22	Error en la posición angular 2 con PD+Twisting, trayectoria circular	59
5.23	Error en la posición angular 3 con PD+Twisting, trayectoria circular	60
5.24	Control en actuador 1 con PD+Twisting, trayectoria circular	61
5.25	Control en actuador 2 con PD+Twisting, trayectoria circular	61
5.26	Control en actuador 3 con PD+Twisting, trayectoria circular	62
5.27	Error en actuador 1 de resultado experimental PD trayectoria recta	63
5.28	Error en actuador 2 de resultado experimental PD trayectoria recta	64
5.29	Error en actuador 3 de resultado experimental PD trayectoria recta	64
5.30	Error en actuador 1 de resultado experimental PD trayectoria circular	65
5.31	Error en actuador 2 de resultado experimental PD trayectoria circular	66
5.32	Error en actuador 3 de resultado experimental PD trayectoria circular	66
5.33	Error en actuador 1 de resultado experimental PD+Twisting trayectoria recta	68

LISTA DE FIGURAS

5.34	Error en actuador 2 de resultado experimental PD+Twisting trayectoria recta	69
5.35	Error en actuador 3 de resultado experimental PD+Twisting trayectoria recta	69
5.36	Error en actuador 1 de resultado experimental PD+Twisting trayectoria circular	70
5.37	Error en actuador 2 de resultado experimental PD+Twisting trayectoria circular	71
5.38	Error en actuador 3 de resultado experimental PD+Twisting trayectoria circular	72
5.39	Experimento PD+Twisting trayectoria recta con diferentes coordenadas	73
5.40	Experimento PD+Twisting trayectoria circular y alta ganancia	74
A.1	Tarjeta GT-400-SV	79
A.2	Filtro digital para cada motor	81
A.3	Servo amplificadores PY2	82
A.4	Parámetros del amplificador PY2	83
A.5	Entorno C++ 6.0	83
A.6	Entorno Visual Studio 2010	84
A.7	Señal de error en el amplificador PY	85
A.8	Software PY para los amplificadores	86

LISTA DE FIGURAS

Lista de tablas

5.1	Parámetros dinámicos del robot	43
5.2	Ganancias de controladores	44
6.1	Tabla comparativa de simulaciones trayectoria recta	77
6.2	Tabla comparativa de simulaciones trayectoria circular	77
6.3	Tabla comparativa de experimentos trayectoria recta	77
6.4	Tabla comparativa de experimentos trayectoria circular	78

LISTA DE TABLAS

Introducción

1.1 Presentación

La ingeniería desempeña un papel fundamental en el desarrollo de la ciencia y la tecnología. A través de los años se ha diversificado en muchos campos disciplinares, cada una de estas áreas o campos disciplinares (electrónica, eléctrica, mecánica, biomedicina, robótica, aeronáutica, etc.) están enfocados a un problema de diferente naturaleza.

Entonces es clara la necesidad de crear técnicas más sofisticadas, para controlar los sistemas en el área industrial, procesos automáticos, aplicaciones aeroespaciales, médicas, etc. Todas estas aplicaciones exigen un diseño óptimo de los algoritmos de control.

En este trabajo se aborda el estudio de sistemas sobreactuados (número de entradas mayor al número de grados de libertad), proponiendo un algoritmo de control adicionado al nominal. Este algoritmo llamado Twisting, es un algoritmo por modos deslizantes de segundo orden. Dicho controlador presenta discontinuidades y oscilaciones cuyo análisis detallado queda fuera del alcance de este trabajo.

El sistema sobreactuado es un robot paralelo de dos grados de libertad. Algunas aplicaciones de los robots paralelos están enfocadas a montaje automatizado, sobre todo en el micromontaje.

La precisión necesaria para los procesos de micromontaje y las velocidades deseadas requieren de un diseño integral que incorpore óptimamente: sensores adecuados, proce-



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

1. INTRODUCCIÓN

samiento de imágenes y sistema de control. Existen gran variedad en las configuraciones de estos robots paralelos, además de sus aplicaciones diversas.

1.2 Objetivo

Este trabajo tiene por objetivo la comparación de un control nominal, el bien conocido control Proporcional Derivativo, con un PD aumentando un controlador discontinuo. Para ello es necesario el diseño e implementación de un control robusto con respecto a perturbaciones e incertidumbres no modeladas, esto por medio del algoritmo Twisting para un sistema mecánico sobreactuado, comparando el comportamiento del control nominal clásico con el control no lineal.

1.3 Motivación y estado del arte

1.3.1 Estado del arte

Algunos de los trabajos actuales relacionados con el tema de los manipuladores paralelos planares, son las investigaciones realizadas por Weiwei Shang, Shuang del Department of Automation, University of Science and Technology of China Cong (1). En dicha investigación se desarrolla la implementación técnicas de control adaptable, variando los parámetros de fricción en el modelo dinámico, dependiendo esta misma fricción de las velocidades que alcanzan los actuadores. Otro de los trabajos relacionados con el tema es del Institute of Machine Tools and Production Engineering (IWF por sus siglas en Alemán: Institut für Werkzeugmaschinen und Fertigungstechnik), en este instituto se trabajan con manipuladores paralelos con ligeros cambios en sus configuraciones, esos manipuladores. Esta última referencia está enfocada principalmente a las aplicaciones industriales de micromontaje de circuitos impresos electrónicos. Algunos ejemplos de estos manipuladores de última generación se encuentran en las Figuras 1.1, 1.2 y 1.3.

1.3.2 Motivación

Se tiene un robot paralelo (sistema sobreactuado) con incertidumbres (dinámicas no modeladas, tales como fricciones) y perturbaciones externas (materiales en contacto, cambios de temperatura, presión, intensidad de la luz).



Figura 1.1: Dispositivo de manipulación de alta precisión - Esta máquina está equipada con un sistema de control altamente dinámico que compensa todas las perturbaciones que afectan a la precisión dentro del intervalo de 1 micra.

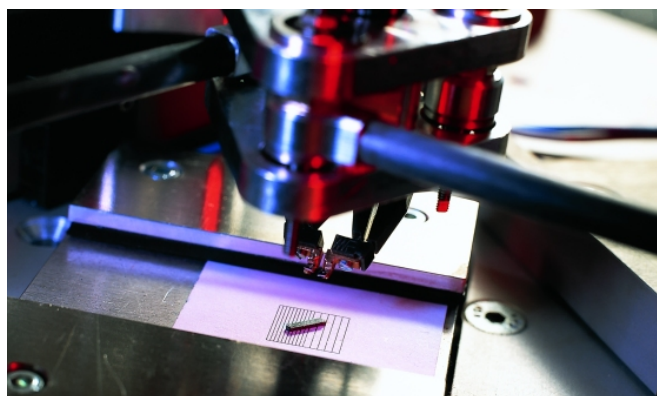


Figura 1.2: Microensamblado - ejemplo de robot paralelo y un microchip en un proceso de ensamblado

1. INTRODUCCIÓN

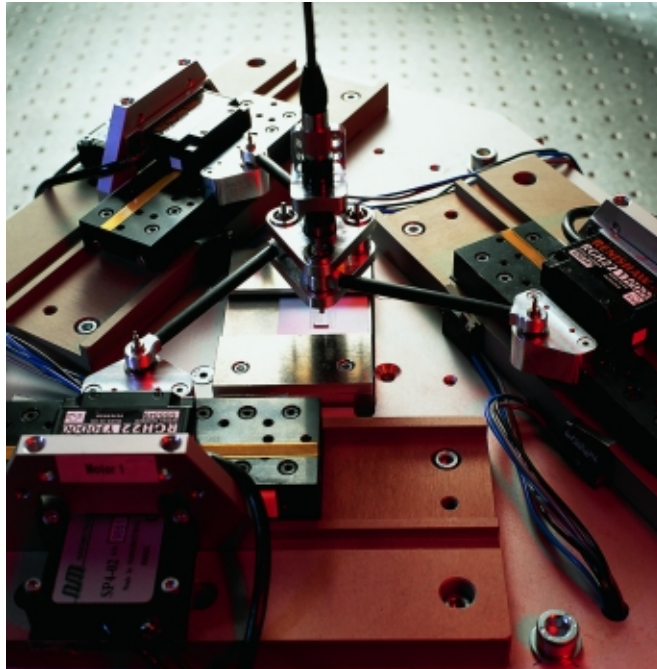


Figura 1.3: Trípode - Robot de micromontaje con estructura plana trípode

Se busca que este robot haga un seguimiento de trayectoria deseada, mediante curvas paramétricas programadas en un ciclo 'for' punto a punto, cuya secuencia de seguimiento es el llamado 'tracking'. Cabe recalcar que ese seguimiento de trayectoria siempre se encontrará en el espacio de tareas del robot, dentro de dicho espacio de tareas el modelo del robot no presentará indeterminaciones.

Para resolver el problema se proponen dos leyes de control y se compararán los resultados obtenidos. La primera propuesta es el bien conocido control PD (proporcional derivativo), mientras que la segunda propuesta de control se trata del mismo control PD pero agregando un término no lineal, el llamado algoritmo Twisting el cual es una técnica de control por modos deslizantes de segundo orden. Al proponer esta segunda solución al problema, buscamos aprovechar las ventajas que tiene este control no lineal ante las perturbaciones, lo cual se podrá traducir en robustez y precisión en el desempeño sistema.

En las figuras 1.4 y 1.5 se muestran imágenes que ilustran el concepto de precisión

y robustez. La imagen de la figura 1.1 es un ejemplo de poca precisión, si todos los puntos coincidieran en el centro, existiría perfecta precisión.

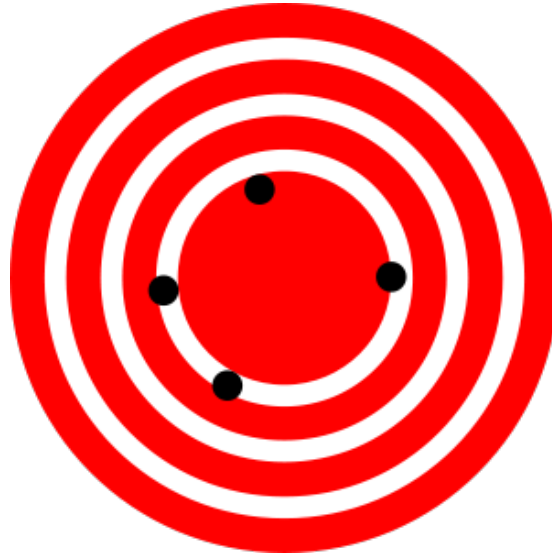


Figura 1.4: Precisión - La precision de un sistema está directamente relacionada con el error y la repetitividad

1.4 Planteamiento del problema

Se tiene el siguiente sistema mecánico sobreactuado, que se encuentra bajo incertidumbres o perturbaciones debido a dinámicas no modeladas o factores externos que afectan al sistema. A partir de las ecuaciones de Lagrange,

$$M(z)\ddot{z} + C(z, \dot{z})\dot{z} + f(\dot{z}) = \tau + \delta(t, z, \dot{z}) \quad (1.1)$$

y se desea diseñar una ley de control para seguimiento de trayectoria. Para ello se propone el siguiente controlador:

$$u = u_0 + u_1 \quad (1.2)$$

donde el u_0 es el PD y el u_1 es el twisting. Las constantes del twisting c_1 y c_2 se eligen con base en una condición para que el control tenga convergencia finita. Este controlador se sumará al nominal con el fin de compensar las incertidumbres del sistema y así mejorar su precisión.

1. INTRODUCCIÓN



Figura 1.5: Robustez - Un sistema es robusto si es capaz de mantenerse funcionando ante perturbaciones externas

1.5 Metodología

Se tiene un objetivo principal, y para llegar a él se establece cierta estructura del sistema que nos permite visualizar el problema planteado y acotar cada una de las tareas que lleva cada parte del trabajo, esto se ilustra gráficamente en la figura 1.6.

El primer bloque correspondiente a la trayectoria deseada es el llamado 'tracking' (seguimiento de trayectoria), basado en la generación de curvas paramétricas. La trayectoria punto a punto es aquella que el robot hará seguimiento de ellas.

En el bloque de control se programan los algoritmos a los que le llegan las señales del bloque de procesamiento de datos junto con la referencia de las trayectorias deseadas, este bloque es el más importante pues en él se basará gran parte de este trabajo.

El bloque del sistema (robot) representa toda la dinámica conocida de la planta, de ella se hablará con detalle en el capítulo correspondiente al modelo matemático del robot paralelo.

En cuanto al bloque de control no lineal, en él se encuentran los algoritmos no lineales de control que se le adicionarán a nuestro control nominal. Los algoritmos otorgarán una mayor robustez y precisión al sistema de control.

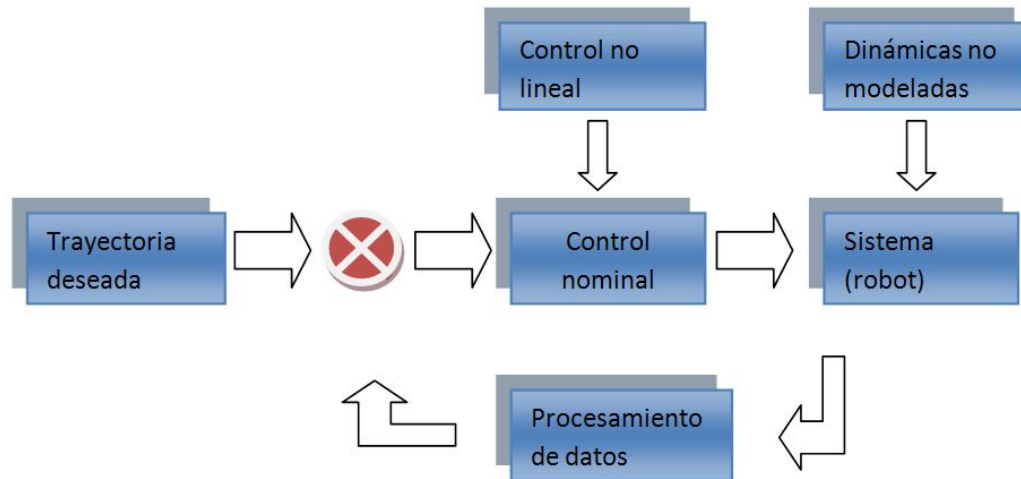


Figura 1.6: Estructura del sistema -

En el bloque correspondiente a las dinámicas no modeladas, se encuentran todas aquellas entradas al sistema que no conocemos. En nuestro modelo matemático, son todas las perturbaciones y cualquier variable no considerada que afecte el desempeño de nuestro objetivo de control.

Finalmente se encuentra el bloque de procesamiento de datos, en éste consideramos toda la comunicación que realizamos con nuestro sistema de adquisición de datos. Esos datos son los que nos proporcionan los sensores (encoders) para conocer la posición angular de cada actuador, en resumen es la comunicación entre el sistema y la computadora personal (PC).

En la Figura 1.7 se muestra la metodología usada para la programación de los algoritmos de control en la tarjeta GT-400-SV, esta tarjeta controladora de movimiento es la que permite la interfaz PC-Robot, para más detalle e información sobre la misma consultar el apéndice correspondiente. Dicha tarjeta se programa desde un lenguaje C++. Sin la utilización de estas herramientas de programación no hubiera sido posible la implementación real de los algoritmos de control. De la Figura 1.4 el bloque correspondiente a las instrucciones de la tarjeta GT-400-SV es el que hace posible el uso de un diferente algoritmo de control. En la sección A.1 de apéndices se describe con exactitud la herramienta usada para realizar esta tarea. El uso de las sentencias if, for, así como la programación de clases y metodos corresponden a las herramientas

1. INTRODUCCIÓN

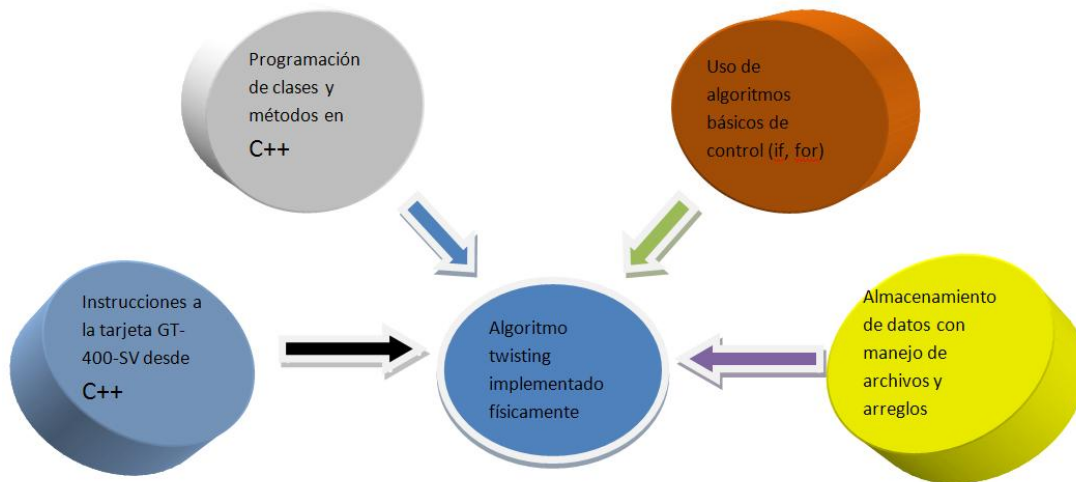


Figura 1.7: Metodología de programación -

propias de programación necesarias para construir los algoritmos. Finalmente el uso de arreglos y manejo de archivos nos permite rescatar la información y poder exportarlos a programas más sofisticados para su análisis (como MATLAB).

1.6 Contribuciones

La principal contribución de este trabajo es la comparación de un algoritmo no lineal, diferente completamente al nominal, en este sistema. Basado en la teoría de sistemas lineales clásicos y la teoría de Modos Deslizantes. Para poder realizar esta comparación, fue necesario el estudio detallado del sistema y poder implementar de manera física este algoritmo.

Otra contribución importante fue la puesta en funcionamiento del sistema, pues se encontraba en un estado inutilizable. Para ello se resolvieron muchos detalles técnicos. Se utilizó teoría básica para la resolución de fallas eléctricas. Esto no se mencionara a detalle en este trabajo pues de hacerlo, debido a la cantidad de información, el trabajo podría perder énfasis en su objetivo principal.

Las contribuciones van más allá de lo que tiene alcance el objetivo principal de este trabajo, pues es el primer paso para futuras investigaciones en este tipo de aplicaciones (sistema sobreactuado), dentro del laboratorio de modos deslizantes de la Facultad de Ingeniería de la Universidad Nacional Autónoma de México. Al poder implementar un

control diferente al nominal en un sistema de este tipo, se podrán tener más trabajos futuros, investigaciones y aplicaciones.

Ejemplo de aplicaciones e investigaciones a futuro, se encuentran la implementación de observadores por modos deslizantes. Algoritmos aún más sofisticados pero que no podrían implementarse sin haber pasado por este proceso previo. También se encuentran muchas posibilidades en el área de algoritmos a prueba de fallas, ejemplo de ello sería el modelar la falla quitando un actuador en algún punto de operación. Por último cabe destacar el estudio detallado de los sistemas implícitos, y la propuesta de controladores creados especialmente para este tipo de sistemas.

1.7 Estructura de la tesis

Este trabajo está dividido en cinco capítulos. Al principio se encuentra la parte de la presentación, objetivos, introducción, etc.

El segundo capítulo es una introducción a la teoría básica que se emplea en este trabajo, una sección enfocada a la mecánica de Lagrange para el estudio de los sistemas dinámicos, después una descripción breve de la teoría de control clásico (controladores PD) y finalmente la sección introductoria de los modos deslizantes junto con un resumen acerca de sus características y propiedades.

El tercer capítulo lleva por nombre "modelo matemático del robot paralelo". Este capítulo es uno de los más importantes pues se describe a detalle el modelo dinámico, cuyas características son bastante interesantes.

En el cuarto capítulo "control del robot paralelo" se describe brevemente la teoría del control empleada y el algoritmo Twisting usado en este trabajo. También las bases teóricas para poder justificar el uso del algoritmo implementado.

El quinto capítulo titulado "Resultados" corresponde a los experimentos numéricos en el entorno MATLAB SIMULINK [(2)] y [(3)]. En este capítulo se resume la teoría aplicada a un sistema virtual de simulación. También se encuentran los 'Resultados experimentales' que corresponden a la implementación física del algoritmo de control. A pesar de que es un capítulo corto en comparación con los demás, el que más trabajo

1. INTRODUCCIÓN

y tiempo implicó para poder concluirlo.

Finalmente se encuentra la parte de apéndices que menciona de manera breve algunas secciones, como son la descripción de los módulos de potencia, la programación de la tarjeta GT-400-SV para la comunicación entre robot-pc, la programación en MATLAB simulink para las simulaciones y la calibración de los encoders.

Teoría básica empleada

2.1 Ecuaciones de Lagrange

Existen muchos métodos para la formulación de las ecuaciones dinámicas de un sistema mecánico. Todos los métodos generan conjuntos equivalentes de ecuaciones, pero algunas formas de las ecuaciones pueden ser más adecuadas para el cálculo o el análisis. Se hará uso de la teoría de las ecuaciones de Lagrange, que se basa en las propiedades de energía de los sistemas mecánicos para calcular las ecuaciones de movimiento. [(4)].

2.1.1 Formulación básica

Considere un sistema de n partículas que obedece la segunda ley de Newton-la velocidad de variación de impulso de una partícula es proporcional a la fuerza aplicada a ésta. Sea F_i la fuerza aplicada sobre la i -ésima partícula, m_i la masa de dicha partícula, y r_i sea su posición, entonces la ley de Newton se convierte en:

$$F_i = m_i \ddot{r}_i \quad r_i \in \mathbb{R}^3, i = 1, \dots, n. \quad (2.1)$$

Nuestro interés no está en un conjunto de partículas independientes, sino más bien en partículas que están unidas y tienen grados limitados de libertad. Para describir esta interconexión, se introducen restricciones entre las posiciones de nuestras partículas. Cada restricción está representada por una función $g_j : \mathbb{R}^{3n} \rightarrow \mathbb{R}$ tal que:

$$g_j(r_1, \dots, r_n) = 0 \quad j = 1, \dots, k \quad (2.2)$$

Una restricción que se puede escribir de esta forma, como una relación algebraica entre las posiciones de las partículas, se denomina una restricción holonómica. Res-

2. TEORÍA BÁSICA EMPLEADA

tricciones de carácter muy propio de los cuerpos rígidos, lo que implica r_i también se puede presentar en múltiples cadenas o eslabones.

Una restricción actúa sobre un sistema de partículas mediante la aplicación de fuerzas de restricción. Las fuerzas de restricción se determinan de tal manera que la restricción en la ecuación (2.2) se cumple siempre. Si consideramos la restricción como una superficie lisa en \mathbb{R}^n , las fuerzas de restricción son normales a la superficie y se debe restringir la velocidad del sistema para que sea tangente a la superficie en todo momento. Por lo tanto, podemos reescribir 2.1:

$$F = \begin{bmatrix} m_1 I & \dots & 0 \\ \vdots & & \vdots \\ 0 & \dots & m_n I \end{bmatrix} \begin{bmatrix} \ddot{r}_1 \\ \cdot \\ \ddot{r}_1 \end{bmatrix} + \sum_{j=1}^k \Gamma_j \lambda_j, \quad (2.3)$$

donde los vectores $\Gamma_1, \dots, \Gamma_k \in \mathbb{R}^{3n}$ son una base para las fuerzas de ligadura y λ_j es el factor de escala para el elemento de base j -ésimo. No se requiere que $\Gamma_1, \dots, \Gamma_k$ sea ortonormal. Por restricciones de la forma en (2.2), Γ_j puede ser tomado como el gradiente de g_j , que es perpendicular al nivel establecido $g_j(r) = 0$.

Los escalares $\lambda_1, \dots, \lambda_k$ se denominan multiplicadores de Lagrange. Se deben determinar los multiplicadores de Lagrange para resolver las $3n + k$ ecuaciones dadas por (2.2) y (2.3) para las $3n + k$ variables $r \in \mathbb{R}^{3n}$ y $\lambda \in \mathbb{R}^k$. Los valores λ_i sólo dan las magnitudes relativas de las fuerzas de restricción ya que los vectores Γ_j no son necesariamente ortonormales.

Esta forma de lidiar con las restricciones holonómicas con las limitaciones es intuitivamente simple pero computacionalmente complejo, ya que debe realizar un seguimiento del estado de todas las partículas en el sistema, aunque no sean capaces de un movimiento independiente. Un enfoque más atractivo es describir el movimiento del sistema en términos de un conjunto más pequeño de variables que describe completamente el configuración del sistema. Para un sistema de n partículas con k restricciones, buscamos un conjunto de $m = 3n - k$ variables q_1, \dots, q_m y funciones suaves f_1, \dots, f_n tales que:

$$r_i = f_i(q_1, \dots, q_m) \iff g_j(r_1, \dots, r_n) = 0 \quad (2.4)$$

$$i = 1, \dots, n \quad j = 1, \dots, k$$

Llamamos a las q_i un conjunto de coordenadas generalizadas para el sistema. Para una robot manipulador que consiste en uniones rígidas, estas coordenadas generalizadas casi siempre son elegidas para ser los ángulos de las articulaciones. La especificación de

estos ángulos únicamente determina la posición de todas las partículas que constituyen el robot.

Como los valores de las coordenadas generalizadas son suficientes para especificar la posición de las partículas, podemos volver a escribir las ecuaciones de movimiento para el sistema en términos de las coordenadas generalizadas. Para ello, también se expresan las fuerzas externas aplicadas al sistema en términos de componentes a lo largo de las coordenadas generalizadas. Nosotros llamamos a estas fuerzas las fuerzas generalizadas para distinguirlas de las fuerzas físicas, que están siempre representadas como vectores en \mathbb{R}^3 . Para un robot manipulador con los ángulos de las articulaciones que actúan como coordenadas generalizadas, las fuerzas generalizadas son los pares aplicados sobre los ejes de las articulaciones.

Para escribir las ecuaciones de movimiento, se define la función de Lagrange, L , como la diferencia entre la energía cinética y la energía potencial del sistema, es decir:

$$L(q, \dot{q}) = T(q, \dot{q}) - V(q) \quad (2.5)$$

donde T es la energía cinética y V es la energía potencial del sistema, ambos escritos en coordenadas generalizadas. Finalmente se establece el siguiente Teorema (ver (5) para más detalle):

Teorema 2.1 (5) *Las ecuaciones de movimiento para un sistema mecánico con coordenadas generalizadas $q \in \mathbb{R}^m$ y Lagrangiano L vienen dadas por:*

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = \tau_i \quad (2.6)$$

donde τ_i es la fuerza externa actuando en la i -ésima coordenada generalizada. La expresión anterior se conoce como ecuación de Lagrange.

2.2 Controladores PD

Un control Proporcional Derivativo (PD) es un mecanismo de control por realimentación que calcula la desviación o error entre un valor medido y el valor que se quiere obtener, para aplicar una acción correctora que ajuste el proceso. El algoritmo de cálculo del control PD se da en dos parámetros distintos: el proporcional, y el derivativo. El valor proporcional determina la reacción del error actual, el derivativo determina la reacción

2. TEORÍA BÁSICA EMPLEADA

del tiempo en el que el error se produce. La suma de estas dos acciones es usada para ajustar el proceso, por medio de un elemento de control como la posición de una válvula de control o la energía suministrada a un calentador. Ajustando estas dos variables en el algoritmo de control del PD, el controlador puede proveer un control diseñado para lo que requiera el proceso a realizar. La respuesta del controlador puede ser descrita en términos de un error, el grado en el cual el controlador llega al "set point" y el grado de oscilación del sistema. Algunas aplicaciones pueden sólo requerir de uno o dos modos de los que provee este sistema de control.

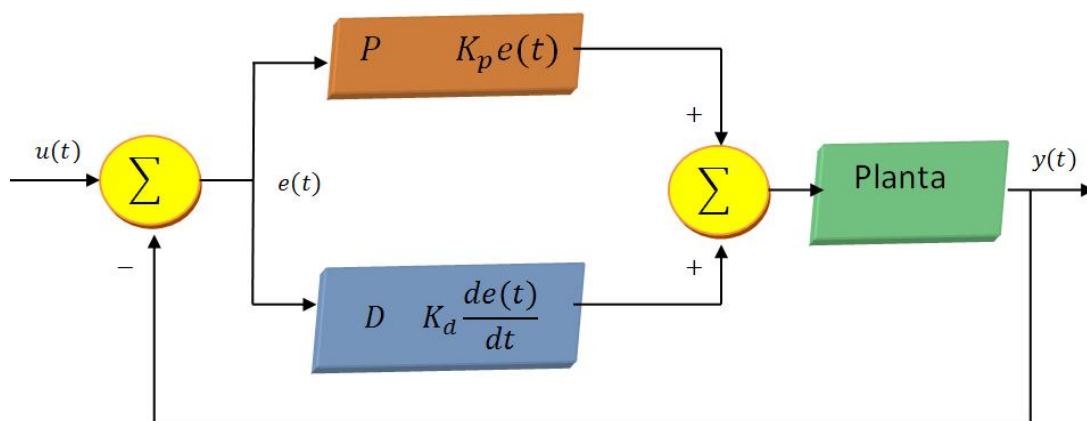


Figura 2.1: PD - Diagrama de bloques de un controlador PD

En la Figura 2.1 se puede apreciar un esquema basado en diagramas de bloques de un controlador PD, el bloque "planta" corresponde a la dinámica conocida del sistema. Los bloques P y D corresponden a la acción proporcional y derivativa, respectivamente.

2.3 Introducción a los Modos Deslizantes

Los modos Deslizantes resultan atractivos cuando se requiere robustez en el diseño del controlador, ya que un sistema controlado por modos deslizantes es insensible a perturbaciones e incertidumbres acopladas al control. En general, el control por modos deslizantes garantiza que el sistema satisfaga una restricción previamente establecida a pesar de perturbaciones e incertidumbres. Normalmente esta restricción se satisface con la aplicación de acciones de control discontinuo, como los conmutadores. Las ideas

2.3 Introducción a los Modos Deslizantes

originales acerca del concepto de modos deslizantes pueden encontrarse en publicaciones que se remontan a la década de los cincuentas [(6)].

Los Sistemas de Control por Modos deslizantes son una generalización de los sistemas de control de Estructura Variable (VSS por sus siglas en inglés). Los VSS tienen la característica de cambiar de estructura por medio de alguna ley, de manera de satisfacer características deseadas, lo que puede significar en el caso de sistema lineales por ejemplo, cambiar entre dos ganancias los lazos de realimentación. Los sistemas de control por modos deslizantes tienen una elevada aceptación dentro la comunidad científica e ingenieril. Su aplicación e implementación se deben a los grandes avances en la electrónica de potencia, que permiten la implementación de dispositivos de conmutación de muy alta velocidad. En 1980, la parte principal de la teoría clásica del control por modos deslizantes se finalizó y que después reportó el profesor Utkin en su monografía en Rusia 1981. En dicha monografía, el procedimiento de dos etapas para el diseño del control por modos deslizantes se especificó claramente[(7)]. Los pasos a seguir son los siguientes:

1. Diseño de la superficie de deslizamiento.
2. Controladores discontinuos que garantizan los modos deslizantes.

Las principales ventajas del control por modos deslizantes son:

- Robustez frente a perturbaciones.
- Compensación exacta (insensibilidad) de las incertidumbres acotadas.
- Reducción del orden de las ecuaciones del sistema.
- Convergencia en tiempo finito a la superficie de deslizamiento.

Sin embargo las principales desventajas de SMC son evidentes:

- Chattering (movimiento de alta frecuencia).
- Insensibilidad solo comparable con las perturbaciones.

2. TEORÍA BÁSICA EMPLEADA

- Las variables de deslizamiento convergen en tiempo finito, pero las variables de estado sólo convergen asintóticamente.
- El rendimiento del lazo cerrado no ideal depende de la discretización, ruido y dinámicas parásitas.
- El diseño de la superficie deslizante se limita a tener un grado relativo con respecto al control, es decir, derivadas de orden mayor son necesarios para el diseño de la superficie deslizante.

2.3.1 Sistemas de Control de Estructura Variable

Como se mencionó en la sección anterior, en los sistemas de control de estructura variable, la ley de control depende del estado del sistema.

La ley de control esta dada por:

$$u(t) \begin{cases} -1 & \text{si } s(y, \dot{y}) > 0 \\ 1 & \text{si } s(y, \dot{y}) < 0 \end{cases} \quad (2.7)$$

de (2.7) la función de conmutación es:

$$s(y, \dot{y}) = my + \dot{y}, \quad (2.8)$$

donde m es un escalar positivo. La razón para el uso del término "función de conmutación" es claro, si la función dada en (2.8) es usada para decidir que estructura de control se pondrá en uso para cualquier punto (y, \dot{y}) en el plano de fase, (2.7) queda de la siguiente manera:

$$u(t) = -\text{sgn}(s(t)) \quad (2.9)$$

donde $\text{sgn}(\cdot)$ es la función signo, la cual tiene la siguiente propiedad:

$$s \cdot \text{sign}(s) = |s| \quad (2.10)$$

Para ilustrar la ley de estructura variable consideramos el doble integrador dado por:

$$\ddot{y}(t) = u(t) \quad (2.11)$$

Y (2.7) es usada para controlar este doble integrador. Para una gran cantidad de valores de \dot{y} en el plano de fases como se muestra en la Figura 2.2

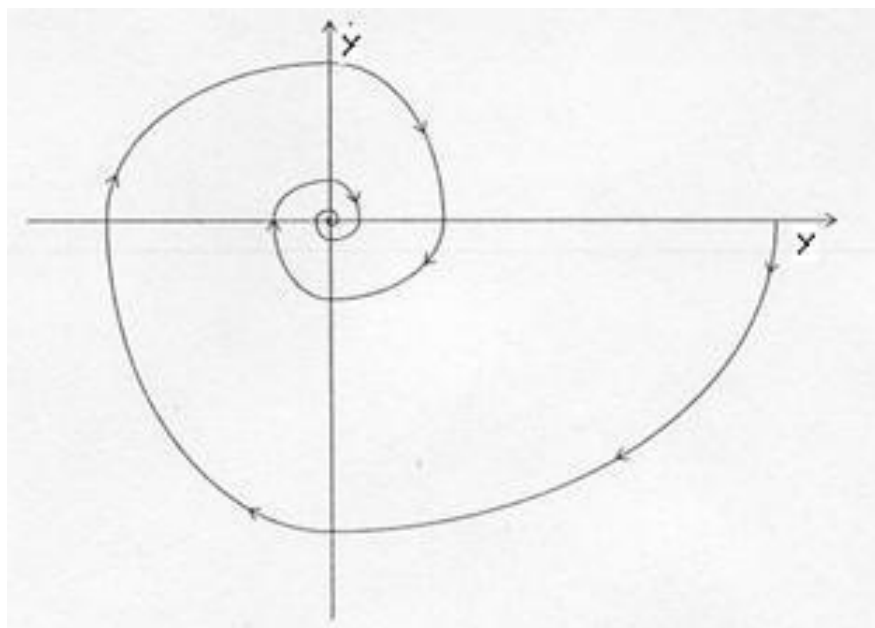


Figura 2.2: Plano de fase del sistema -

En dicho plano de fase existe una recta que representa el conjunto de puntos para los cuales $(y, \dot{y}) = 0$, en este caso una línea recta que atraviesa el origen con una pendiente $-m$. Y los valores de \dot{y} que satisfacen la desigualdad $m|\dot{y}| < 1$ entonces:

$$s\dot{s} = s(m\dot{y} - \ddot{y}) = s(m\dot{y} - \text{sgn}(s)) < |s|(m|\dot{y}| - 1) < 0 \quad (2.12)$$

Consecuentemente, cuando $m|\dot{y}| < 0$ las trayectorias del sistema a ambos lados de la línea apuntan hacia la línea L_s .

Por tanto, una frecuencia de conmutación alta entre las dos estructuras de control tomará lugar cuando el sistema de trayectorias cruce repetidamente la línea L_s . Este movimiento de alta frecuencia se describe como chattering. Cuando el movimiento es limitado a la línea L_s se satisface la ecuación diferencial $(y, \dot{y}) = 0$ y por lo tanto:

$$\dot{y}(t) = -my(t) \quad (2.13)$$

Esto representa una descomposición de primer orden y las trayectorias se deslizan a lo largo de la línea L_s como se muestra en la figura 2.3:

Dicho comportamiento dinámico es descrito como un Modo Deslizante Ideal y la línea L_s representa la Superficie Deslizante. Durante el modo deslizante el sistema se

2. TEORÍA BÁSICA EMPLEADA

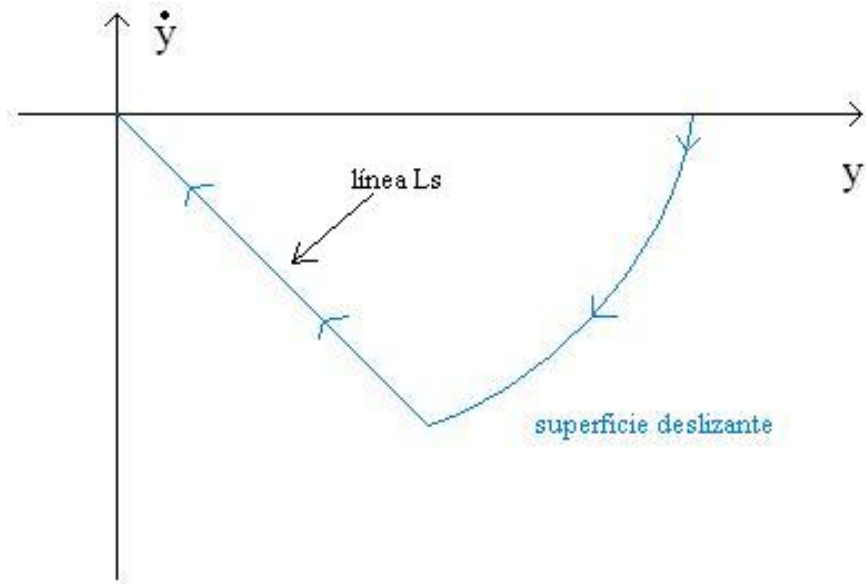


Figura 2.3: Plano de fase del modo deslizante -

comporta como un sistema de orden reducido. La acción de control asegura que las condiciones dadas en (2.6) se satisfagan y esto garantiza que $s(y, \dot{y}) = 0$. La condición dada en (2.12) se puede escribir más convenientemente como:

$$s\dot{s} < 0 \quad (2.14)$$

que se refiere a la condición de accesibilidad. La ley de control es diseñada entonces para garantizar que esta condición se satisfaga y ésta sólo se satisface en un dominio del plano de fase

$$\Omega = \{(y, \dot{y}) : m|\dot{y}| < 1\}. \quad (2.15)$$

2.3.2 Control equivalente

Considerese el doble integrador de la ecuación (2.11) y la ley de control dada en (2.7). El plano de fase de la Figura 2.4 es el resultado de la simulación del comportamiento en lazo cerrado cuando $m=1$, y las condiciones iniciales son $y(0) = 1, \dot{y}(0) = 0$. En la Figura 2.5 se muestra la acción de control asociada a esta simulación. En ambas

2.3 Introducción a los Modos Deslizantes

gráficas puede apreciarse que el deslizamiento toma lugar después de 0.7232 segundos cuando una frecuencia de conmutación alta aparece.

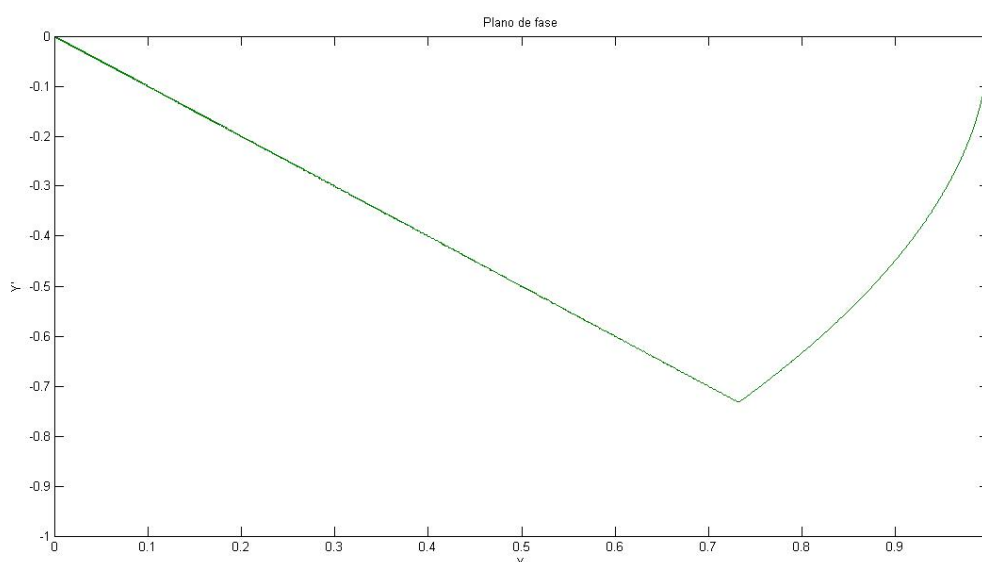


Figura 2.4: Plano de fase - Simulación del plano de fase en matlab simulink con $y(0) = 0$, $\dot{y}(0) = 1$

Antes de considerar las propiedades del modo deslizante, haremos una interpretación de la señal de control mostrada, considerándola como un promedio o una señal de baja frecuencia.

Suponiendo que en un tiempo t_s se alcanza la superficie de conmutación y que el modo deslizante ideal toma lugar, es decir que la función de conmutación satisface $s(t) = 0$ para todo $t > t_s$, lo cual implica que $\dot{s}(t) = 0$ para todo $t \geq t_s$. Sin embargo, de las ecuaciones (2.8) y (2.11)

$$\dot{s}(t) = m\dot{y}(t) + u(t) \quad (2.16)$$

y por tanto ya que $\dot{s}(t) = 0$ para todo $t \geq t_s$, la ley de control que mantiene la trayectoria en L_s es:

$$u(t) = -m\dot{y}(t) \quad (2.17)$$

2. TEORÍA BÁSICA EMPLEADA

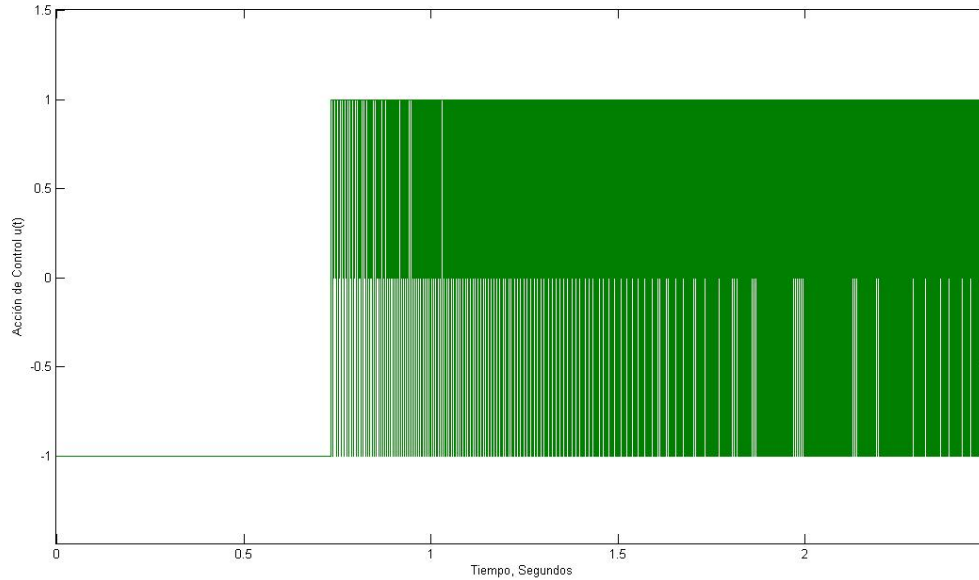


Figura 2.5: Acción de Control - Simulación de la acción de control del doble integrador

Esta ley de control se conoce como la acción de control equivalente. Esta no es la señal de control que está aplicada a la planta pero podemos decir que es la señal de control que está aplicada en promedio. La señal de control aplicada a la planta se puede componer entonces de una señal de baja frecuencia u_p y otra señal de alta frecuencia $(u - u_p)$, de modo que:

$$u(t) = u_p(t) + (u(t) - u_p(t)) \quad (2.18)$$

Para obtener la componente de baja frecuencia $u_p(t)$ podemos pasar la señal de control $u(t)$ a través de un filtro paso bajas

$$\tau \dot{u}_p + u_p(t) = u(t). \quad (2.19)$$

Aparte de la propiedad de robustez mostrada por el modo deslizante, otro beneficio que aparece por esta situación es que la planta es forzada a comportarse como un sistema de primer orden. Esto garantiza que no ocurra un sobrepaso cuando se quiere regular el sistema desde una condición arbitraria hasta desplazarlo al origen.

2.3 Introducción a los Modos Deslizantes

Las dos propiedades del modo deslizante ideal descritas anteriormente, rechazo de perturbaciones y reducción del orden, son propiedades claves que han motivado para el estudio de los controladores que incluyen los modos deslizantes.

Los modos deslizantes estándar poseen ciertas propiedades especiales además de las ya mencionadas. Son alcanzados en tiempo finito, esto quiere decir que las trayectorias cruzan por algún punto de deslizamiento. El movimiento sobre la superficie es llevado a cabo en una superficie discontinua y puede ser entendido como el límite del movimiento cuando las perturbaciones desaparecen y la frecuencia de conmutación tiende a infinito. Cualquier generalización de la noción de los modos deslizantes deberá heredar estas propiedades, ver [(8)].

2. TEORÍA BÁSICA EMPLEADA

Modelo matemático del robot paralelo

En este capítulo se presenta una descripción de lo que es nuestro sistema de experimentación: el modelo del robot paralelo de dos grados de libertad, así como también se describe su funcionamiento y sus elementos básicos que lo componen.

La obtención del modelo matemático se realiza minuciosamente pues consideramos detalles bastante interesantes, como son las restricciones del mismo. Se empieza con una descripción de la cinemática del sistema, para después tomar en cuenta la dinámica del sistema, todo bajo el enfoque de Euler-Lagrange.

Finalmente el modelo se lleva a la forma de espacio de estados para poder aplicar las teorías de control e introducir el modelo a simulación, además de facilitar el análisis del sistema para ver el comportamiento de sus estados, entradas y salidas.

3.1 Descripción de la planta

Se define un robot paralelo como: "Aquel robot en el que el extremo final está unido a la base por más de una cadena cinemática independiente".

La arquitectura de los robots paralelos es muy diferente a la de los robots serie, lo que conduce a que la mayoría de los problemas teóricos se deban reconsiderar. De hecho existe una extraña dualidad entre ambos tipos de mecanismos, ya que un problema de difícil solución para un tipo de estructura, es fácilmente resoluble por el otro tipo de estructura, y viceversa [(9)] [(10)].

3. MODELO MATEMÁTICO DEL ROBOT PARALELO

Para este trabajo se utilizó un equipo llamado GPM2002 de la empresa Googol Technology. Este equipo mostrado en la Figura 3.1 consta de un movimiento planar en coordenadas x, y para el efector final, aunque el robot está equipado con tres actuadores que coordinan el movimiento de los mecanismos que componen el robot.

El equipo también cuenta con un módulo de potencia para los motores, la pc para controlar el sistema y modificar algunos parámetros vía una tarjeta PCI llamada GT400-SV.

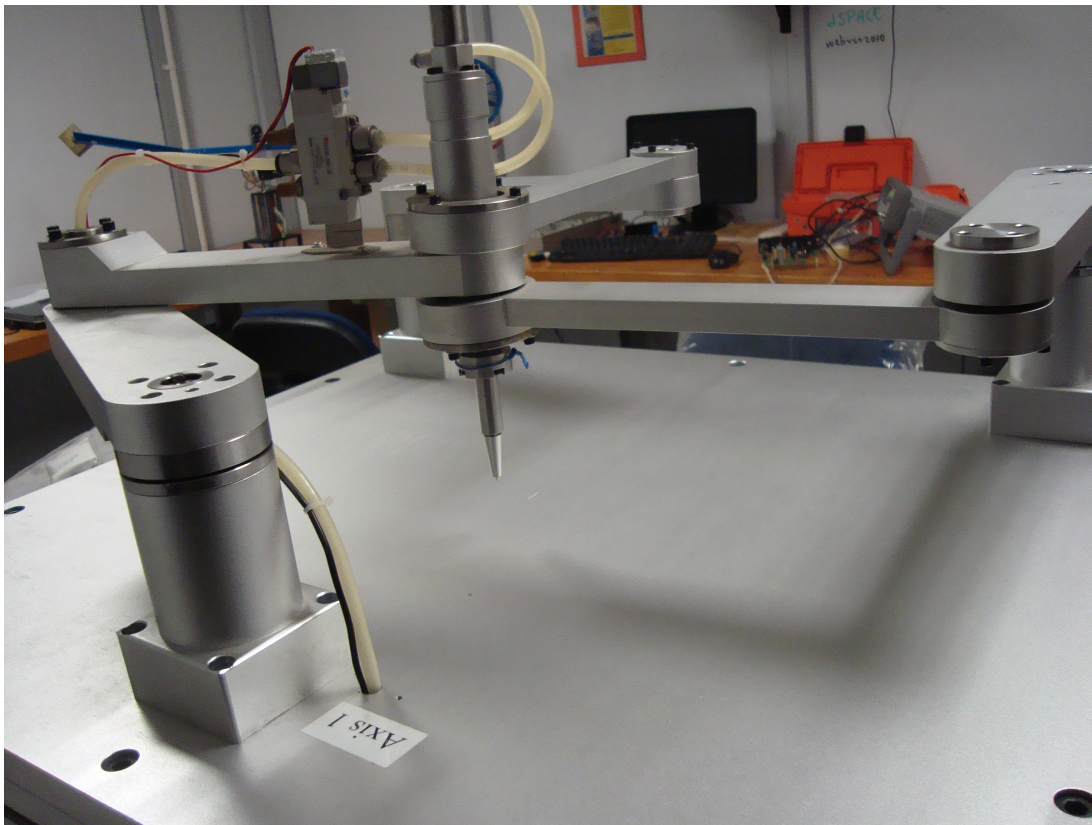


Figura 3.1: Robot paralelo 1 - vista frontal del robot paralelo

Sin embargo en este capítulo nos enfocaremos a analizar la dinámica del robot y el movimiento planar del efector final (parte instalada en el extremo final del robot equivalente a la mano humana) en coordenadas cartesianas.

En manera breve el funcionamiento del sistema consiste en enviar las señales de voltaje a los 3 motores que conforman el robot paralelo, de tal forma que se consiga

3.2 Modelo matemático del robot

una coordenada x, y deseada, y así hacer seguimiento de trayectorias mediante curvas paramétricas. En la Figura 3.2 se puede apreciar al centro el efector final. Para el análisis detallado de los sistemas robóticos se puede consultar en [(4)] y [(11)].

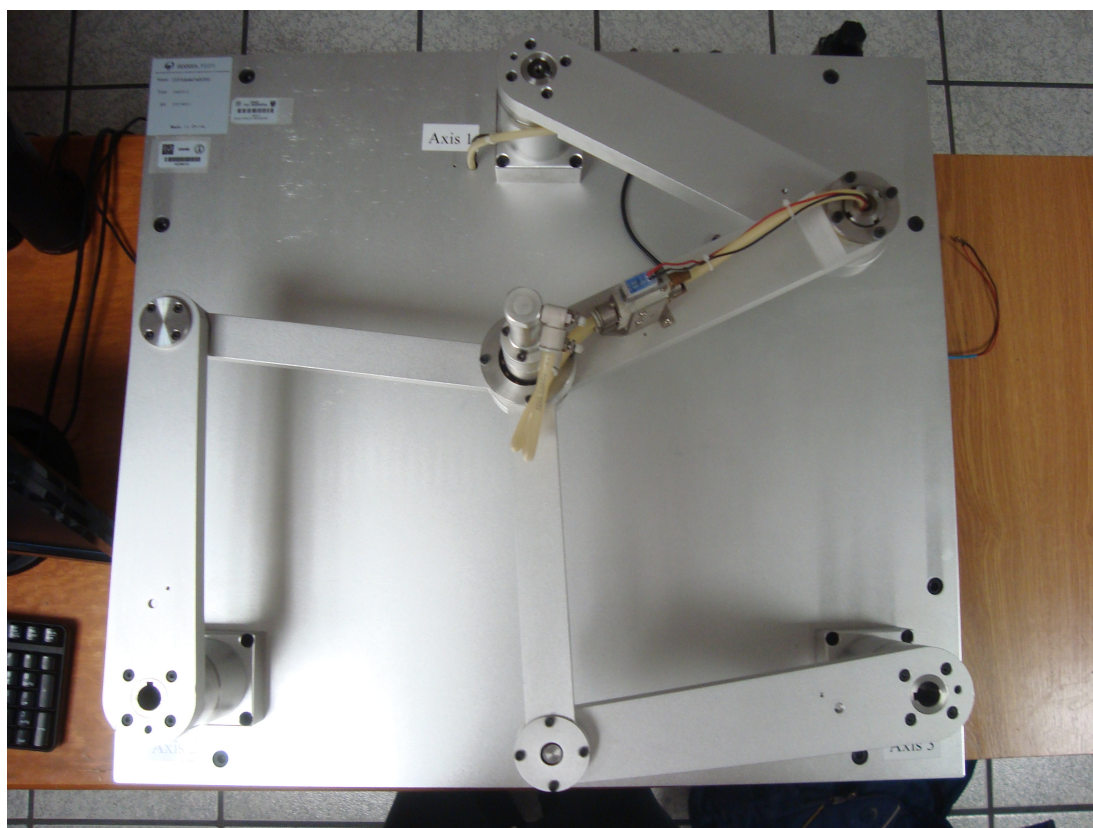


Figura 3.2: Robot paralelo 2 - vista superior de la unidad GPM2002

3.2 Modelo matemático del robot

A partir de aquí comienza la deducción para poder llegar a las 6 ecuaciones diferenciales que describen el sistema. Se comienza hablando de la cinemática del robot paralelo. Primero la cinemática directa y después la cinemática inversa. Después se obtiene un modelo dinámico en función de los ángulos de los actuadores. Teniendo el modelo dinámico en función de los ángulos, se lleva al espacio de tareas, con las restricciones algebraicas del sistema. Más adelante esas ecuaciones diferenciales serán reducidas para obtener finalmente un modelo en espacio de estados.

3. MODELO MATEMÁTICO DEL ROBOT PARALELO

3.2.1 Cinemática del robot paralelo

En esta sección abordaremos la cinemática del robot paralelo. En el caso de los robots paralelos, no es posible aplicar algoritmos como el de Denavit-Hartenberg [(12)] para obtener de manera sistemática la cinemática directa. Como se muestra en la figura 3.3.

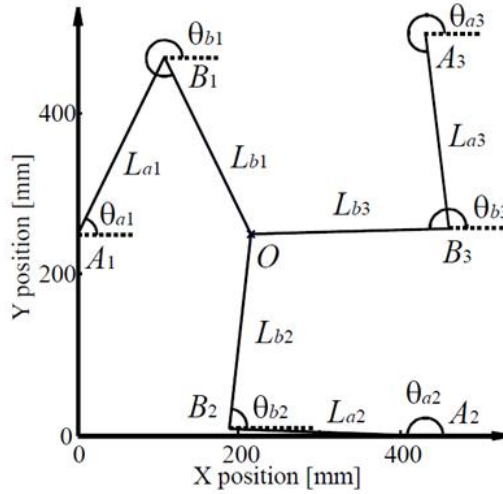


Figura 3.3: Parámetros geométricos del robot -

En la Figura 3.3 se muestran los parámetros geométricos correspondientes a nuestro sistema. Más adelante se siguen expresiones generales y algunas definiciones.

Cinemática directa: esta consiste en obtener la posición del efector final $P = (p_x, p_y)$ a partir de los valores de los ángulos $z = (\theta_{a1}, \theta_{a2}, \theta_{a3}, \theta_{b1}, \theta_{b2}, \theta_{b3})$, por simplificación escribiremos $z = (a_1, a_2, a_3, b_1, b_2, b_3)$, mediante la siguiente expresión:

$$\begin{pmatrix} x_{ai} + l \cos a_i + l \cos b_i \\ x_{bi} + l \cos a_i + l \cos b_i \end{pmatrix}, i = 1, 2, 3. \quad (3.1)$$

Por tanto, la cinemática inversa [(13)] consiste en obtener los valores de los ángulos $z = (a_1, a_2, a_3, b_1, b_2, b_3)$ a partir de la posición del efector final $P = (p_x, p_y)$

$$a_i = \arctan \left(\frac{y_0 - y_{ai}}{x_0 - x_{ai}} \right) + \arccos \frac{d_i}{2l} \quad (3.2)$$

$$b_i = \arctan\left(\frac{y_0 - y_{ai} - l \sin a_i}{x_0 - x_{ai} - l \cos a_i}\right) + \arccos \frac{d_i}{2l}; i = 1, 2, 3 \quad (3.3)$$

$$\text{donde, } d_i = \sqrt{(x_0 - x_{ai})^2 + (y_0 - y_{ai})^2} \quad (3.4)$$

3.2.2 Modelo dinámico: Cadena cinemática abierta

En esta sección abordaremos una cadena individual de las tres con las que cuenta nuestro robot paralelo con el enfoque de Euler-Lagrange, más adelante combinaremos estas expresiones para obtener un modelo general.

En el enfoque de Euler Lagrange primero se calcula el Lagrangiano, vease ecuación (2.5)

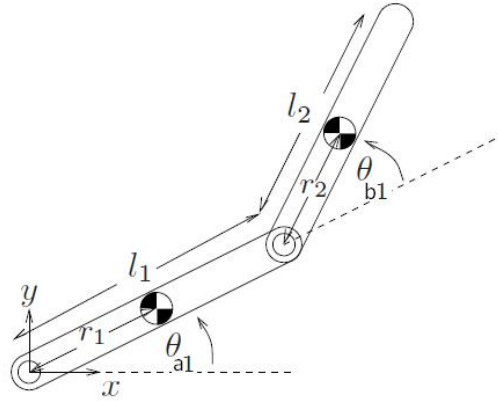


Figura 3.4: Cadena cinemática abierta -

Basándonos en la Figura 3.4, tenemos:

$$\begin{aligned} \bar{x}_1 &= r_1 c_1 & \dot{\bar{x}}_1 &= -r_1 s_1 \dot{a}_1 \\ \bar{y}_1 &= r_1 s_1 & \dot{\bar{y}}_1 &= -r_1 c_1 \dot{a}_1 \\ \bar{x}_2 &= l_1 c_1 + r_2 c_2 & \dot{\bar{x}}_2 &= -(l_1 s_1 + r_2 s_2) \dot{a}_1 - r_2 c_2 \dot{b}_2 \\ \bar{y}_2 &= l_1 s_1 + r_2 s_2 & \dot{\bar{y}}_2 &= (l_1 c_1 + r_2 s_2) \dot{a}_1 + r_2 c_2 \dot{b}_2 \end{aligned} \quad (3.5)$$

Donde: $s_i = \sin a_i$, $s_{ij} = \sin(a_i - a_j)$ y de manera similar para c_i y c_{ij}

La energía cinética queda de la siguiente forma:

$$T(\theta, \dot{\theta}) = \frac{1}{2} m_{a1} (\dot{\bar{x}}_1^2 + \dot{\bar{y}}_1^2) + \frac{1}{2} J_{a1} \dot{\theta}_{a1}^2 + \frac{1}{2} m_{b1} (\dot{\bar{x}}_2^2 + \dot{\bar{y}}_2^2) + \frac{1}{2} J_{b1} (\dot{\theta}_{a1}^2 + \dot{\theta}_{b1}^2) \quad (3.6)$$

3. MODELO MATEMÁTICO DEL ROBOT PARALELO

Luego de agrupar algunos términos y hacer $T = L$, se resuelve la ecuación de Euler-Lagrange (2.6). Después de agrupar algunos términos y ciertas sustituciones, además de llevarlo a una forma matricial, tenemos la siguiente expresión:

$$M_i(z)\ddot{z} + C_i(z, \dot{z})\dot{z} + N_i(z, \dot{z}) = \tau_i; i = 1, 2, 3 \quad (3.7)$$

donde:

- M_i es la matriz de inercia.
- C_i es la matriz de fuerzas centrífugas y de Coriolis (en nuestro caso sólo contamos con fuerzas centrífugas).
- N_i representa las fuerzas gravitatoria y de fricción (debido a que el robot se encuentra en posición horizontal se considera despreciable la fuerza gravitatoria)
- τ_i es el torque del actuador A_i

Dichas matrices tienen la forma siguiente:

$$M_i = \begin{pmatrix} \alpha_i & \gamma_i \cos a_i - b_i \\ \gamma_i \cos a_i - b_i & \beta_i \end{pmatrix} \quad (3.8)$$

$$C_i = \begin{pmatrix} 0 & (\gamma_i \cos a_i - b_i)\dot{b}_i \\ (\gamma_i \cos a_i - b_i)\dot{b}_i & 0 \end{pmatrix} \quad (3.9)$$

donde:

- $\alpha_i = J_{a1} + m_{ai}r_{ai}^2 + m_{bi}l_2$.
- $\beta_i = J_{bi} + m_{bi}r_{ai}^2$.
- $\gamma_i = m_{bi}r_{bi}l$; $i=1,2,3$.

Propiedades:

- M_i es una matriz simétrica definida positiva.
- $\frac{dM_i}{dt} - 2C_i$ es una matriz simétrica distorsionada (torcida).

3.2.3 Modelo dinámico sin fuerzas de restricción

Cuando se combinan las dinámicas de las tres cadenas cinemáticas que componen al robot se tiene el siguiente modelo:

$$M(z)\ddot{z} + C(z, \dot{z})\dot{z} + f(\dot{z}) = \tau_i \quad (3.10)$$

Donde $f \in \mathbb{R}^6$ es la fuerza de fricción en las articulaciones y $\tau = (\tau_{a1}, \tau_{a2}, \tau_{a3}, 0, 0, 0)$ es el torque en las uniones activas.

Las matrices M y C tienen la siguiente forma:

$$M = \begin{pmatrix} \alpha_1 & 0 & 0 & \gamma_1 \cos ab_1 & 0 & 0 \\ 0 & \alpha_2 & 0 & 0 & \gamma_2 \cos ab_2 & 0 \\ 0 & 0 & \alpha_3 & 0 & 0 & \gamma_i \cos ab_3 \\ \gamma_1 \cos ab_1 & 0 & 0 & \beta_1 & 0 & 0 \\ 0 & \gamma_2 \cos ab_2 & 0 & 0 & \beta_1 & 0 \\ 0 & 0 & \gamma_3 \cos ab_3 & 0 & 0 & \beta_1 \end{pmatrix} \quad (3.11)$$

$$C = \begin{pmatrix} 0 & 0 & 0 & (\gamma_1 \cos ab_1)\dot{b}_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & (\gamma_2 \cos ab_2)\dot{b}_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & (\gamma_3 \cos ab_3)\dot{b}_3 \\ (\gamma_1 \cos ab_1)\dot{b}_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & (\gamma_2 \cos ab_2)\dot{b}_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & (\gamma_3 \cos ab_3)\dot{b}_3 & 0 & 0 & 0 \end{pmatrix} \quad (3.12)$$

Por simplificación, el término $ab_i = a_i - b_i$, $i = 1, 2, 3$.

Al igual que en (3.9) y (3.10) las matrices cumplen con las siguientes propiedades:

- M_i es una matriz simétrica definida positiva.
- $\frac{dM_i}{dt} - 2C_i$ es una matriz simétrica distorsionada (torcida).

3.2.4 Modelo dinámico del robot paralelo con fuerzas de restricción

Tomando en cuenta el sistema de cinemática abierta y las fuerzas de restricción debido al sistema de lazo cerrado, el modelo matemático del robot en el espacio de uniones queda de la siguiente manera:

$$M(z)\ddot{z} + C_i(z, \dot{z})\dot{z} + f(\dot{z}) = \tau + R^T \lambda \quad (3.13)$$

3. MODELO MATEMÁTICO DEL ROBOT PARALELO

Donde $R^T \lambda$ es el vector de las fuerzas de restricción. La matriz R es el diferencial de las restricciones de lazo cerrado y λ es el multiplicador de que representa la magnitud de las fuerzas de restricción. Las restricciones de movimiento de lazo cerrado del robot vienen dadas por la matriz L :

$$L(q) = \begin{bmatrix} x_{a_1} + l \cos(a_1) + l \cos(b_1) - x_{a_2} - l \cos(a_2) - l \cos(b_2) \\ y_{a_1} + l \sin(a_1) + l \sin(b_1) - y_{a_2} - l \sin(a_2) - l \sin(b_2) \\ x_{a_1} + l \cos(a_1) + l \cos(b_1) - x_{a_3} - l \cos(a_3) - l \cos(b_3) \\ y_{a_1} + l \sin(a_1) + l \sin(b_1) - y_{a_3} - l \sin(a_3) - l \sin(b_3) \end{bmatrix} = 0 \quad (3.14)$$

Diferenciando, tenemos $\frac{dL(q)}{dt} = \frac{\partial L(q)}{\partial q} \dot{q} = R(q) \dot{q} = 0$ y la matriz R puede escribirse así:

$$R = \begin{bmatrix} -l \sin(a_1) & l \sin(a_2) & 0 & -l \sin(b_1) & l \sin(b_2) & 0 \\ l \cos(a_1) & -l \cos(a_2) & 0 & l \cos(b_1) & -l \cos(b_2) & 0 \\ -l \sin(a_1) & 0 & -l \sin(a_3) & -l \sin(b_1) & 0 & l \sin(b_3) \\ l \cos(a_1) & 0 & -l \cos(a_3) & l \cos(b_1) & 0 & -l \cos(b_3) \end{bmatrix} \quad (3.15)$$

Las fuerzas de restricción $R^T \lambda$ son desconocidas y es difícil medirlas directamente, afortunadamente pueden ser eliminadas por la expresión del espacio nulo de la matriz R . Con la matriz Jacobiana W , tenemos:

$$\dot{z} = W \dot{p} \quad (3.16)$$

donde:

- $\dot{z} = [\dot{a}_1, \dot{a}_2, \dot{a}_3, \dot{b}_1, \dot{b}_2, \dot{b}_3]^T$ representa el vector velocidad de todas las uniones.
- $\dot{p} = [\dot{x}, \dot{y}]^T$ representa el vector velocidad del efector final.

Y la matriz Jacobiana puede definirse como:

$$W = \begin{bmatrix} r_1 \cos(b_1) & r_1 \sin(b_1) \\ r_2 \cos(b_2) & r_2 \sin(b_2) \\ r_3 \cos(b_3) & r_3 \sin(b_3) \\ -r_1 \cos(a_1) & -r_1 \sin(a_1) \\ -r_2 \cos(a_2) & -r_2 \sin(a_2) \\ r_3 \cos(a_3) & -r_3 \sin(a_3) \end{bmatrix} \text{ donde : } r_i = \frac{1}{l \sin(b_i - a_i)} \quad (3.17)$$

Considerando la ecuación de restricción $R \dot{z} = 0$ (Principio de D'Alembert [(5)]) podemos tener $RW \dot{p}_e = 0$ por la relación Jacobiana. El vector velocidad \dot{p} del efector

final contiene coordenadas generalizadas independientes.

Por tanto: $RW = 0$, o su equivalente: $R^T W^T = 0$

De esta forma se puede eliminar el término $R^T \lambda$ y el modelo dinámico en el espacio de tareas puede ser expresado de la siguiente forma:

$$W^T M(z)\ddot{z} + W^T C(z, \dot{z})\dot{z} = W^T \tau - W^T f(z, \dot{z}) + W^T R^T \lambda = W^T \tau - W^T f(z, \dot{z}) \quad (3.18)$$

Derivando \dot{z} de la relación Jacobiana (3.16), tenemos:

$$\ddot{z} = \dot{W}\dot{p} + W\ddot{p} \quad (3.19)$$

y sustituyendo (3.19) y (3.18) en (3.20) tenemos:

$$\widetilde{M}(z)\ddot{p} + \widetilde{C}(z, \dot{z})\dot{p} = W^T \tau - W^T f(\dot{z}) \quad (3.20)$$

donde:

- $\widetilde{M} = W^T M W$ representa la matriz de inercia del espacio de tareas.
- $\widetilde{C} = W^T (M\dot{W} + CW)$ representa la matriz de fuerzas centrífugas en el espacio de tareas.

De la misma manera que en el sistema de cadena cinemática abierta, las matrices \widetilde{M} y \widetilde{C} cumplen las propiedades:

- \widetilde{M} es una matriz simétrica definida positiva.
- $\frac{d\widetilde{M}}{dt} - 2\widetilde{C}$ es una matriz simétrica distorsionada (torcida).

En la Figura 3.5 se puede apreciar un ejemplo de cuando el efector final se encuentra dentro del espacio de trabajo. En la Figura 3.6 se muestra cuando el efector final está fuera del espacio de trabajo, a medida que se aleja del espacio de trabajo las ecuaciones del modelo entran en indeterminación. Lo que se traduce en pérdida de exactitud e incluso de controlabilidad.

3. MODELO MATEMÁTICO DEL ROBOT PARALELO

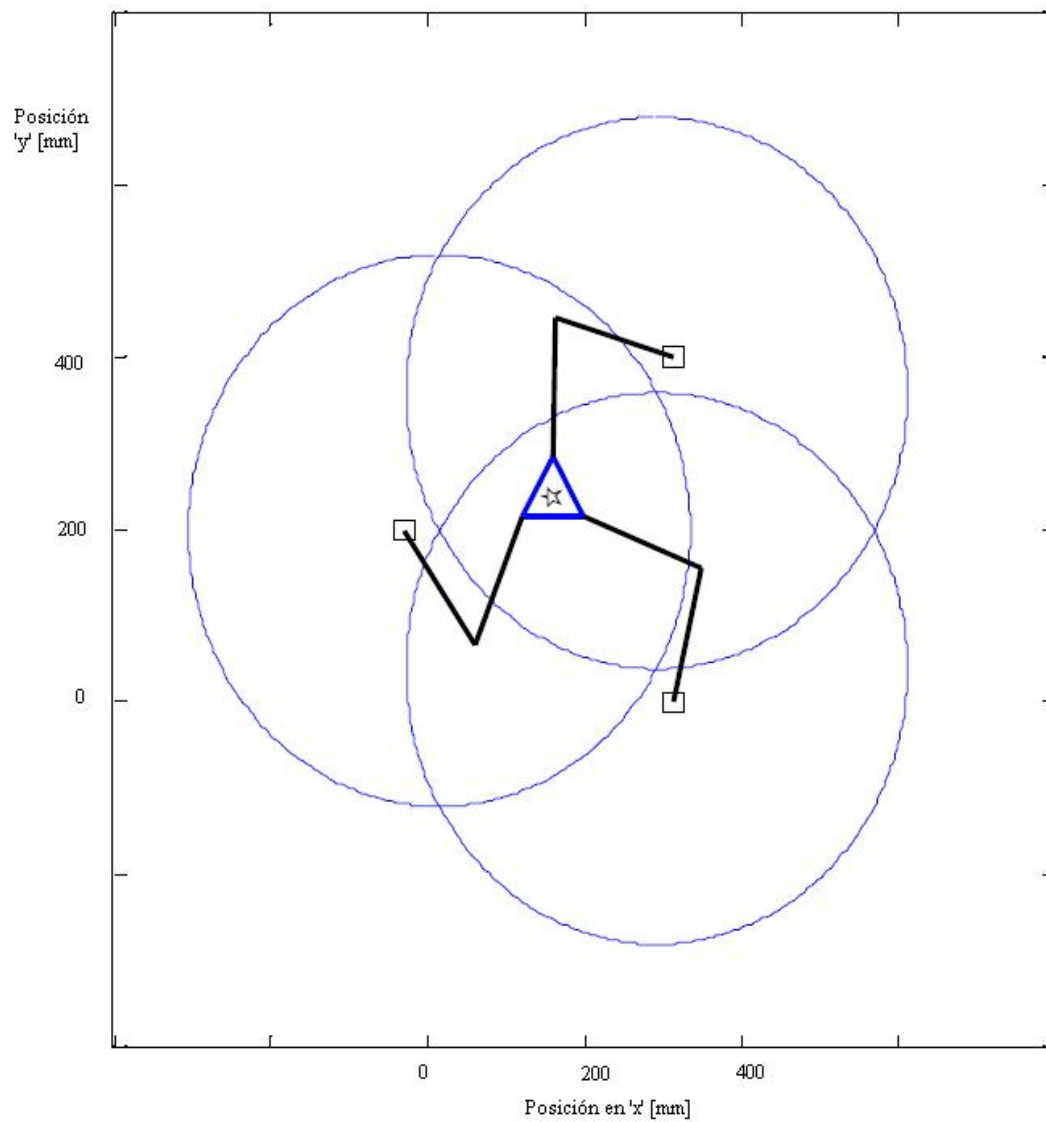


Figura 3.5: Efector final del robot dentro del espacio de trabajo -

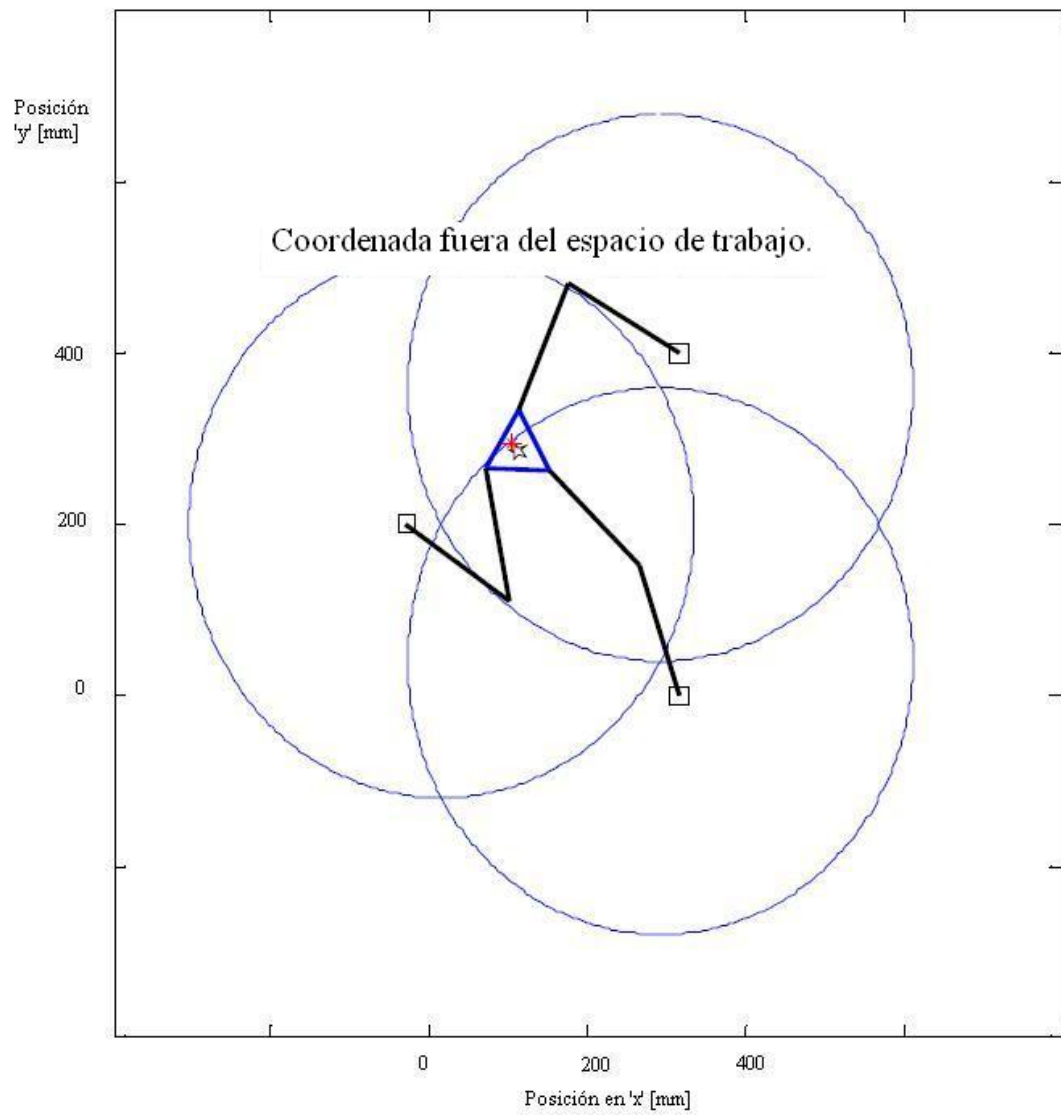


Figura 3.6: Efector final fuera del espacio de trabajo -

3. MODELO MATEMÁTICO DEL ROBOT PARALELO

3.2.5 Modelo en espacio de estados

Con el objetivo de aplicar las técnicas y análisis de control al sistema, transformaremos el modelo (3.20) a la forma de Cauchy. Considerando el cambio de variables:

$$\begin{aligned}x_1 &= p_x \\x_2 &= p_y \\x_3 &= \dot{p}_x \\x_4 &= \dot{p}_y\end{aligned}$$

Por tanto, el modelo queda expresado de la siguiente manera:

$$\dot{X} = A(z, \dot{z})X + B(z)u \quad (3.21)$$

$$Y = (a_1, a_2, a_3)^T \quad (3.22)$$

Donde:

- $X = (x_1, x_2, x_3, x_4)^T$ Es el vector de estados (posiciones y velocidades del efector final).
- $Y = (a_1, a_2, a_3)^T$ Es el vector de mediciones (posición angular de las uniones activas).
- $u = (u_1, u_2, u_3)$ Es el control expresado como el torque de los actuadores.

Y las matrices A y B vienen dadas por las expresiones siguientes:

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & A_1[1, 1] & A_1[1, 2] \\ 0 & 0 & A_1[2, 1] & A_1[2, 2] \end{pmatrix}, A_1 = -(\tilde{M})^{-1}\tilde{C}, \quad (3.23)$$

$$B = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ B_1[1, 1] & B_1[1, 2] & B_1[1, 3] \\ B_1[2, 1] & B_1[2, 2] & B_1[2, 3] \end{pmatrix}, B_1 = -(\tilde{M})^{-1}S^T. \quad (3.24)$$

Control del Robot Paralelo

En este capítulo abordaremos el tema central de este trabajo: el control del sistema. Los sistemas de control determinan la efectividad con la que funciona un sistema, dispositivo, proceso, etc., por eso es importante recordar la relevancia de este capítulo pues sin importar que tan bueno sea el diseño mecánico, el sistema electrónico, los amplificadores o la adquisición de datos, el control siempre determinará de manera radical el desempeño del sistema. Lo que nos lleva a considerar los elementos siguientes: la mecánica, la calidad de los sensores, etc. no consideran las incertidumbres y perturbaciones asociadas sus elementos, no es posible compensarlas. El control sí, es por ello que un buen control responderá a esa pregunta con exactitud.

Estudiaremos el sistema de control con el que operará nuestro robot, la idea es que al control nominal proporcional derivativo (PD) le sumemos una señal proveniente de un algoritmo de control no lineal, el control por modos deslizantes, que dadas unas incertidumbres acotadas, podrá compensarlas de una manera eficaz.

4.1 Planteamiento del problema

Se tiene el siguiente sistema mecánico de segundo orden, que se encuentra bajo incertidumbres o perturbaciones debido a dinámicas no modeladas o factores externos que afectan al sistema. A partir de las ecuaciones de Lagrange,

$$M(z)\ddot{z} + C(z, \dot{z})\dot{z} + f(\dot{z}) = \tau + \delta(t, z, \dot{z}) \quad (4.1)$$

donde:

- M es la matriz de inercia.

4. CONTROL DEL ROBOT PARALELO

- C es la matriz de fuerzas centrífugas y de Coriolis (en nuestro caso sólo contamos con fuerzas centrífugas).
- f representa la fuerza de fricción.
- τ_i es el torque del actuador A_i

y se desea diseñar una ley de control para seguimiento de trayectoria. Para ello se propone el siguiente controlador:

$$u = u_0 + u_1 \quad (4.2)$$

donde el u_0 es el PD y el u_1 es el twisting. Las constantes del twisting c_1 y c_2 se eligen con base en una condición para que el control tenga convergencia finita. Este controlador se sumará al nominal con el fin de compensar las incertidumbres del sistema y así mejorar su precisión.

4.1.1 Objetivo de control

Se busca que este robot haga un seguimiento de trayectoria deseada, mediante curvas paramétricas programadas en un ciclo 'for' punto a punto, cuya secuencia de seguimiento a cada uno de estos puntos es el llamado 'tracking'. Cabe recalcar que ese seguimiento de trayectoria siempre se encontrará en el espacio de tareas del robot, dentro de dicho espacio de tareas el modelo del robot no presentará indeterminaciones.

4.2 Estructura de control

Por medio de diagramas de bloques se ejemplifica como es que funciona nuestro control. En la Figura 4.1 se ilustra como actúa el control nominal:

Este control nominal ofrece resultados buenos, dependiendo la aplicación que se requiera, pero esos resultados están limitados por perturbaciones y dinámicas no modeladas, por eso se plantea el siguiente control, parecido al anterior pero con un bloque añadido, ese bloque es el control por compensación usando poca información del sistema.

La idea de este control agregado, es poder controlar dos estados (posición y velocidad) en tiempo finito. Con lo anterior concluimos que la señal de control es la suma de

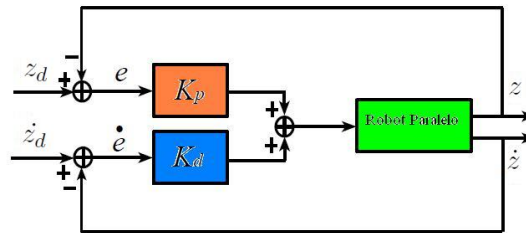


Figura 4.1: Control nominal - acción proporcional y acción derivativa del sistema de control

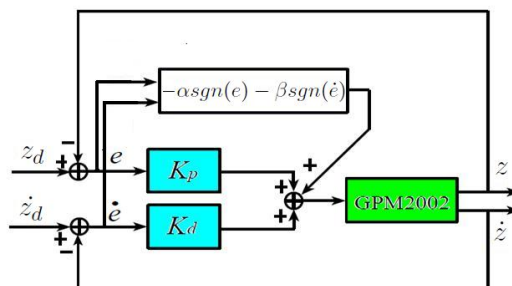


Figura 4.2: Control aumentado - acción proporcional y acción derivativa junto con el controlador 'Twisting' aumentado

4. CONTROL DEL ROBOT PARALELO

una parte nominal y otra de compensación de incertidumbres, como se especifica en la sección siguiente.

4.3 Estrategia de control

Así entonces el problema de control consiste en diseñar una ley de control que dado $x(0) = x_0(0)$, nos garantice que $x(t) = x_0(t)$ para todo $t > 0$.

Sea,

$$u(t) = u_0(t) + u_1(t) \quad (4.3)$$

donde el control $u_0 \in \mathbb{R}^3$ es el control nominal diseñado para el sistema, como un control Proporcional-Derivativo (PD) que tiene el objetivo de realizar el seguimiento de una trayectoria deseada (conocida), además de compensar las dinámicas modeladas en el capítulo anterior. Y el control $u_1 \in \mathbb{R}^3$ es la parte que garantiza la compensación de incertidumbres (fuerzas de fricción) desconocidas del sistema en las uniones activas.

4.4 Control nominal

En esta sección se aborda el control nominal con el que cuenta el robot paralelo, entender su funcionamiento básico es fundamental para poder implementar un controlador distinto a éste. Aunque no se aborda con el riguroso detalle que requiere el diseño de este controlador, pues no es prioridad en este trabajo, sí se mencionan los conceptos básicos que nos servirán para plantear más adelante la justificación de la robustificación del controlador.

La ley de control para los tres actuadores del robot paralelo viene dada por la siguiente expresión del controlador PD:

$$u_0 = K_p(z_d - z) + K_d(\dot{z}_d - \dot{z}) \quad (4.4)$$

Donde:

- z_d es el vector de los ángulos deseados por la trayectoria a seguir.
- z son los ángulos actuales de cada servomotor.

- \dot{z}_d y \dot{z} son las respectivas derivadas de z_d y z .

La razón por la que no se toma en cuenta la parte integral es porque al no saber el último valor guardado en la memoria, este término puede ocasionar sobrepasos y dañar el equipo.

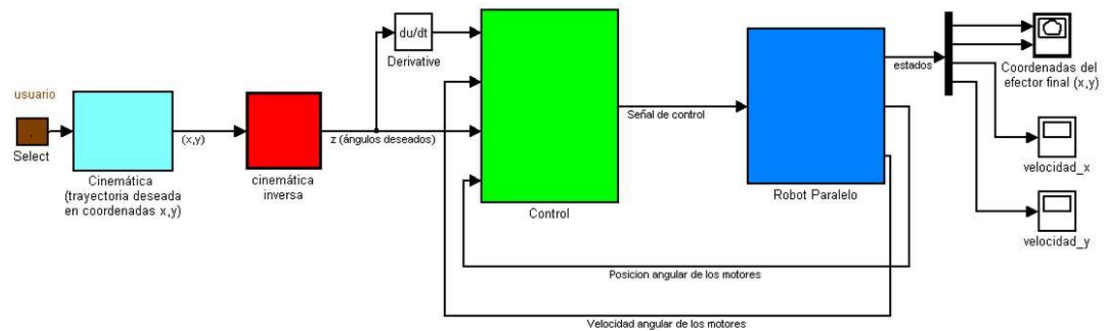


Figura 4.3: Estructura del sistema - acción proporcional y acción derivativa se encuentran en el bloque de control

4.5 Control No Lineal

Las ventajas de los controladores no lineales ya se han mencionado en la parte introductoria de este trabajo, en la anterior sección se habló del controlador clásico PID. Debido a que este controlador no nos puede garantizar convergencia del error a cero en tiempo finito, tenemos que buscar teorías modernas que nos ofrezcan una solución a nuestro problema. Se abordará en las siguientes subsecciones, aunque sin adentrarnos en la estricta formalidad matemática que estos controladores requieren para su análisis.

4.5.1 Robustificación del controlador

A continuación presentaremos el control no lineal, se hablará de manera formal acerca de las ventajas del mismo, así como también sus desventajas. En esta sección se diseña un control robusto por compensación que con base en una diferencia entre el valor deseado y el valor real, cambiará la estructura del controlador para cancelar las incertidumbres (fricciones) no modeladas del sistema.

4. CONTROL DEL ROBOT PARALELO

Así entonces nuestra motivación principal será el poder contar con un esquema robusto de control, sin hablar a detalle de la complejidad de las matemáticas que implica su análisis, pero que con base en investigaciones realizadas podamos implementar de manera sencilla el controlador.

4.5.2 Planteamiento del problema

El modelo general para un sistema mecánico de segundo orden tiene la siguiente forma:

$$M(z)\ddot{z} + C(z, \dot{z})\dot{z} + P(\dot{z}) + G(z) = \tau_i + \delta(t, z, \dot{z}) \quad (4.5)$$

Donde $z \in \mathbb{R}^n$ es un vector de coordenadas generalizadas. $M(z)$ es la matriz de inercia, $C(z, \dot{z})$ es la matriz de Coriolis y fuerzas centrífugas, $P(\dot{z})$ es la fuerza de fricción de Coulomb, que puede contener términos dependientes en \dot{z} , $G(z)$ es el termino de fuerzas de origen gravitatorio, $\delta(t, z, \dot{z})$ es el termino de incertidumbres y τ es el torque producido por los actuadores. Se asume que la entrada de control τ está dada por una función de realimentación conocida.

Introduciendo las variables $x_1 = z$ y $x_2 = \dot{z}$, el modelo puede reescribirse en la forma de espacio de estados:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= f(t, x, y, \tau) + \delta(t, x, y) \end{aligned} \quad (4.6)$$

Donde $f(t, x, y, \tau)$ corresponde a la dinámica nominal del sistema respresentada por las funciones en el modelo general para el sistema mecánico de segundo orden.

Y la parte nominal del sistema dinámico está respresentado por la función:

$$f(t, x, y, \tau) = M^{-1}(z)\ddot{z}[C(z, \dot{z})\dot{z} + P(\dot{z}) + G(z) - \tau] \quad (4.7)$$

4.5.3 Diseño del controlador

Se propone el siguiente diseño de controlador

$$\tau = C(x_1, x_2)x_2 + P(x_2) + G(x_1) - M(x_1)\ddot{x}_1 U \quad (4.8)$$

Y considerese el bien conocido algoritmo Twisting [(14)] dado por:

$$U = -\alpha \text{sgn}(x_1) - \beta \text{sgn}(x_2) \quad (4.9)$$

Sustituyendo (4.6) y (4.7) en (4.5), el sistema de lazo cerrado es descrito por:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\alpha \operatorname{sgn}(x_1) - \beta \operatorname{sgn}(x_2) + \delta(t, x, y) \end{aligned} \quad (4.10)$$

Usando este controlador la incertidumbre puede ser acotada por $|\delta| < D$, tenemos un doble integrador. En ese caso el algoritmo es globalmente uniformemente estable en tiempo finito si se cumple la siguiente desigualdad (para más detalle acerca de el siguiente criterio se puede consultar en[(15)]):

$$\alpha > \beta > 0 \quad (4.11)$$

El controlador twisting, en términos de z_d (posición angular deseada) y z (posición angular actual del motor) así como su velocidad angular, que se le sumará al control nominal tiene la siguiente forma:

$$u_1 = -\alpha \operatorname{sgn}(z - z_d) - \beta \operatorname{sgn}(\dot{z} - \dot{z}_d) \quad (4.12)$$

Y en términos del error:

$$\begin{aligned} u_1 &= -\alpha \operatorname{sgn}(e) - \beta \operatorname{sgn}(\dot{e}) \\ \text{donde :} & \\ e &= z - z_d \\ \dot{e} &= \dot{z} - \dot{z}_d \end{aligned} \quad (4.13)$$

4. CONTROL DEL ROBOT PARALELO

Resultados

En este capítulo nos enfocamos únicamente en los resultados tanto de simulaciones como de experimentos. En ambos casos se presentan las pruebas tanto con el control nominal, como con el aumentado. Cabe mencionar que para las simulaciones se utilizó el lenguaje de programación por bloques MATLAB simulink. Y para las pruebas experimentales se utilizó el entorno visual C++.

Las pruebas consisten en dos partes, la del control nominal y la del control aumentado. Previamente a ellas se especifican los parámetros utilizados tanto en experimentos como en simulaciones.

5.1 Parámetros dinámicos usados

En esta sección se presentan algunos de los parámetros usados tanto para las simulaciones como en la implementación real de los experimentos.

	masa[kg]	longitud de eslabones [m]	r_i [m]	J_i [$kg.m^2$]
L_{a1}	1.2525	0.2440	0.1156	0.0124
L_{a2}	1.3663	0.2440	0.0.067	0.0122
L_{a3}	1.3663	0.2440	0.0.067	0.0122
L_{b1}	1.0771	0.2440	0.1621	0.0098
L_{b2}	0.4132	0.2440	0.1096	0.0036
L_{b3}	0.4132	0.2440	0.1096	0.0036

Tabla 5.1: Parámetros dinámicos del robot paralelo - Estos son los parámetros que fueron introducidos a la simulación, corresponden a los valores reales.

De la tabla 5.1 se tiene,

5. RESULTADOS

- r_i es la distancia entre el centro de masa y la unión.
- J_i es el momento de inercia relativo al centro de masa.

Para completar la información de la tabla anterior también se puede consultar la Figura 3.3 que corresponde a los parámetros geométricos del robot.

Las ganancias del controlador PD, así como las ganancias del controlador Twisting se encuentran en la siguiente tabla:

K_p	K_d	α	β
5	8	0.3	0.15

Tabla 5.2: Parámetros para los controladores - ganancias correspondientes al control PD y al control twisting.

Finalmente para el valor de la fricción seca $f_c = 0.4976[Nm]$, se puede encontrar en [(1)].

5.2 Resultados de Simulaciones

En esta sección se muestran los resultados correspondientes a las pruebas hechas en el simulador. El entorno de simulación usado es MATLAB simulink. Este entorno cuenta con una amplia gama de funciones y librerías que facilitan la representación numérica de nuestro robot paralelo.

Los parámetros de simulación principales fueron los mismos para todos los casos (control nominal, control nominal aumentado, trayectorias recta y circular). Los parámetros son los siguientes:

Tiempo de simulación: 0-4 [s].

Opciones de solución: Fixed-step (Paso fijo)

Método de solución: ode1 (Euler)

Paso de muestreo: 0.001 [s].

5.2.1 Simulación trayectoria recta con PD nominal

En esta subsección se presentan los resultados de simular una trayectoria recta. De una coordenada (x_0, y_0) a una coordenada (x_f, y_f) del efector final. La trayectoria recta va de las coordenadas $(0.2165, 0.250)$ como punto inicial, a las coordenadas $(0.2465, 0.250)$ como punto final, ambas en unidades de metros.

Si definimos el error en la coordenada x como $e_x = x_{deseada} - x_{real}$ y de igual manera para el error en la coordenada y podemos apreciar los resultados en las gráficas correspondientes.

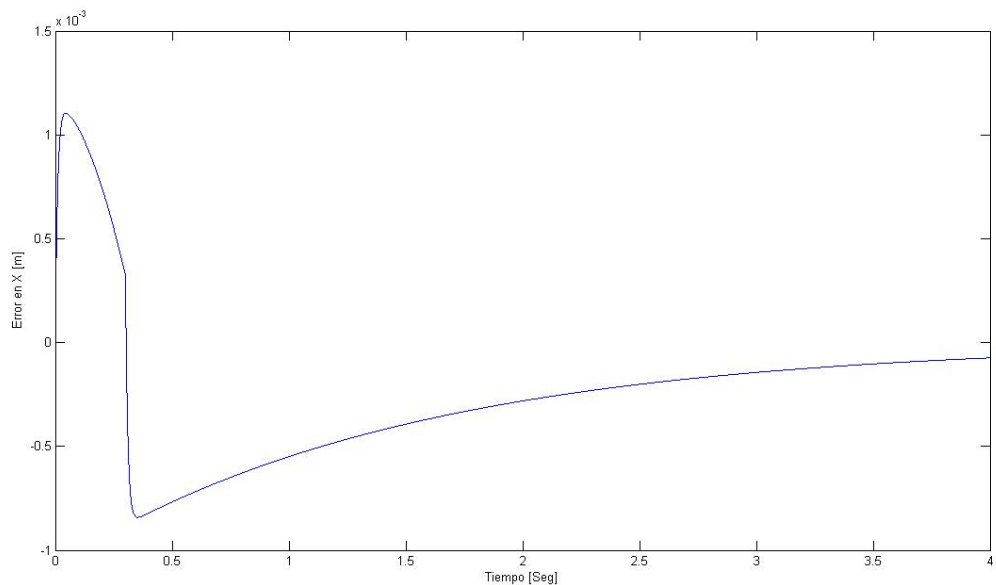


Figura 5.1: Error en la coordenada x del efector final, control PD trayectoria recta -

Podemos notar que el comportamiento de la coordenada x del efector final no llega a cero. Se aproxima solamente, y da un cruce, pero nunca llega. Es evidente desde la primera simulación las limitantes del controlador clásico (poca precisión y lentitud).

De manera similar sucede con el error en la coordenada y del efector final. Instintivamente pensaríamos que el error en esa coordenada debería ser cero, pues el movimiento es únicamente sobre la coordenada x . En conclusión las dinámicas combinadas del sistema y sus propias restricciones ocasionan efectos notables de un actuador sobre todos demás actuadores. Dicho efecto tiende a cero gracias al controlador, pero nunca llega a

5. RESULTADOS

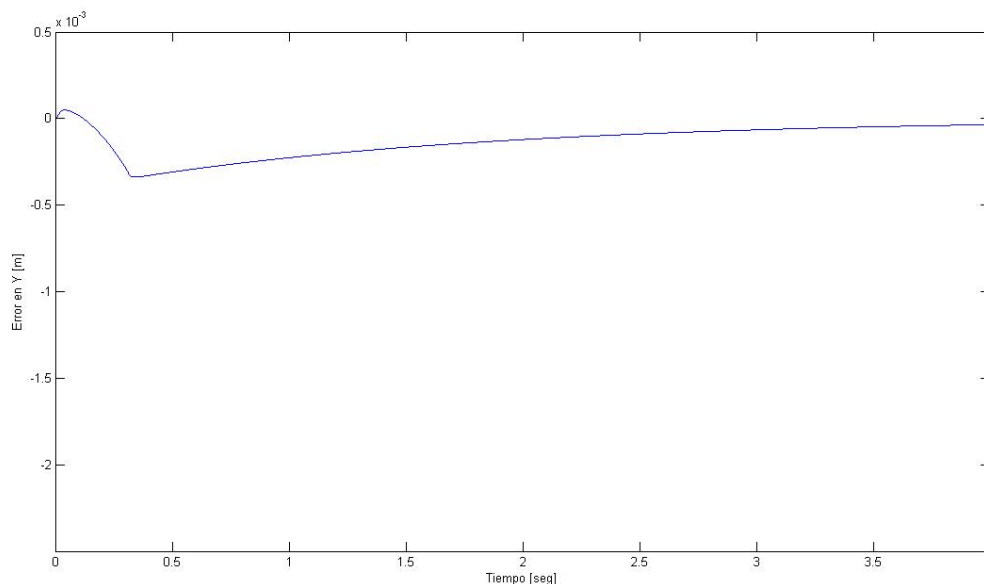


Figura 5.2: Error en la coordenada y del efector final, control PD trayectoria recta -

él. También nos interesa el comportamiento de los ángulos de cada actuador, si definimos el error en la posición angular en cada actuador como: $e_n = z_{ndeseada} - z_{nreal}$, en donde $z_{deseada}$ es la posición angular deseada en grados para el seguimiento de trayectoria y z_{real} es el ángulo real que detecta el sensor (encoder).

En la Figura 5.3, que corresponde al error de posición del actuador 1, podemos apreciar una curva suave. Esta curva suave nos da una idea de como se comportará a nivel experimental la dinámica del error en el actuador. Dicha dinámica podría decirse que es asintótica, pues nunca alcanza al valor de cero el error.

Un comportamiento muy similar puede apreciarse para el actuador 2, como se puede apreciar en la Figura 5.4. Pero con la diferencia de que en este actuador el error es bastante menor, esto significa que realiza menos trabajo que el actuador 1.

En la Figura 5.5, que corresponde al error de posición en el actuador 3, podemos apreciar que la gráfica es idéntica a la Figura 5.4. Esto significa que sus dinámicas de error son muy similares, casi idénticas. De estas tres gráficas podemos concluir que cada motor se comporta de forma muy distinta uno de otro. Algunos realizan más trabajo que otros, todo dependiendo de la tarea que se les exija, pero en los tres casos

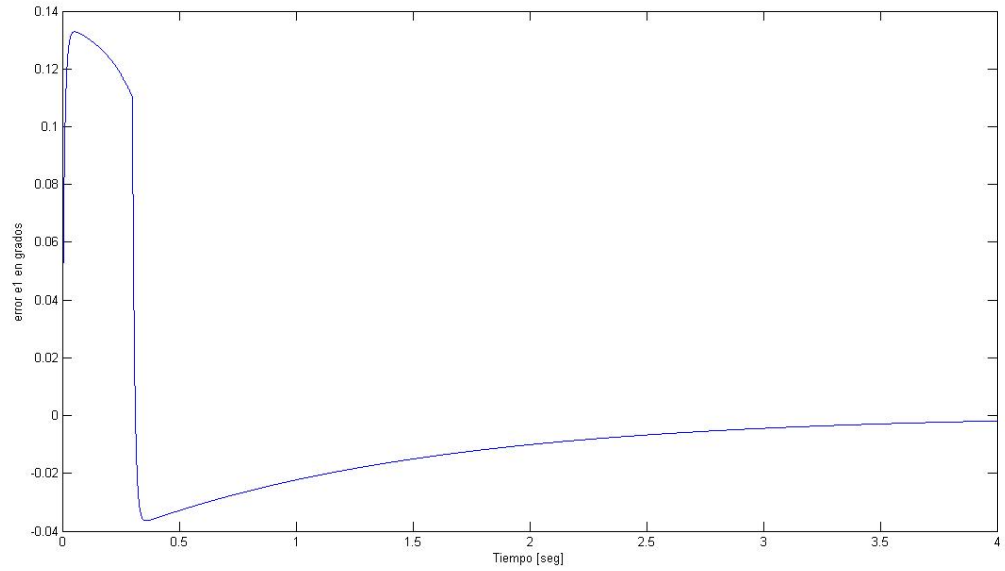


Figura 5.3: Error de posición actuador 1 con PD trayectoria recta -

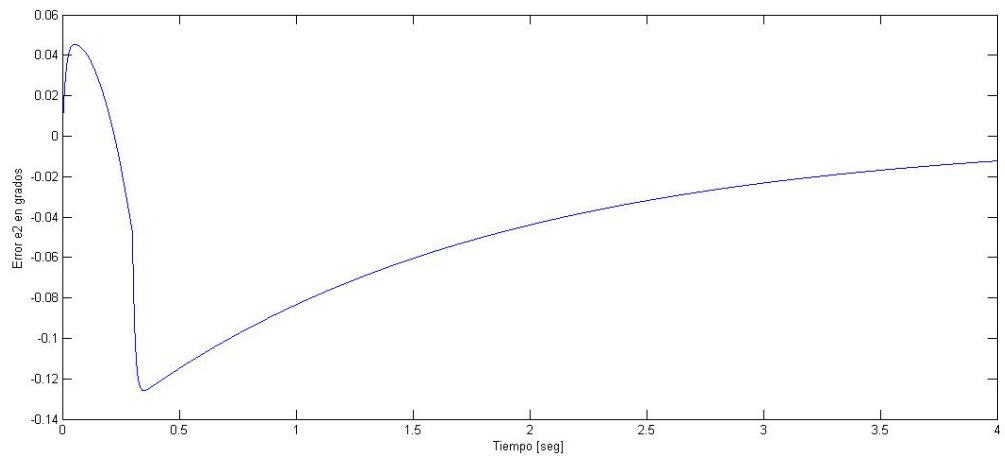


Figura 5.4: Error de posición actuador 2 con PD trayectoria recta -

5. RESULTADOS

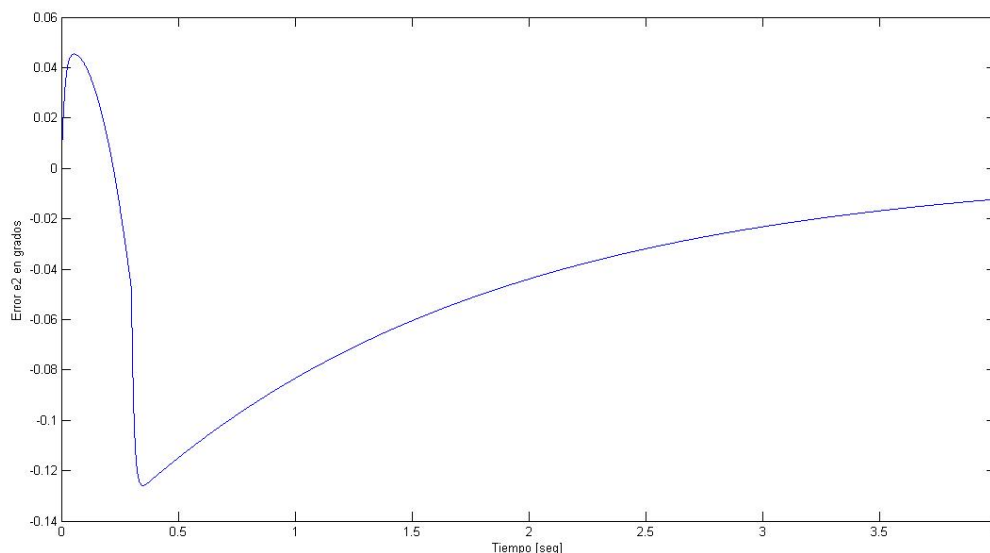


Figura 5.5: Error de posición actuador 3 con PD trayectoria recta -

coinciden que nunca llegan a cero en tiempo finito.

5.2.2 Simulación trayectoria circular con PD nominal

En esta subsección se presentan los resultados de simular un seguimiento de trayectoria circular con un radio $R = 0.03[m]$ y además el movimiento en sentido de las manecillas del reloj. Con un periodo de 4 segundos para que en ese tiempo de simulación complete una vuelta.

La trayectoria circular a seguir con radio de $R = 0.03[m]$ alrededor del punto inicial de operación del efector final que es $(0.2165, 0.250)$. Definimos nuevamente el error en la coordenada x como $x_{deseada} - x_{real}$ y de igual manera para el error en la coordenada y , tenemos los siguientes resultados:

En la Figura 5.6 podemos notar una importante diferencia con el error cuando el seguimiento se trataba de una línea recta. Esta diferencia radica en que al tratarse de una trayectoria más complicada, es más difícil para el controlador hacer cero el error.

De manera similar sucede con la coordenada y del efector final, se puede observar en la gráfica de la Figura 5.7, su comportamiento lejano de poder estabilizarse en cero, varía durante todo el seguimiento de trayectoria.

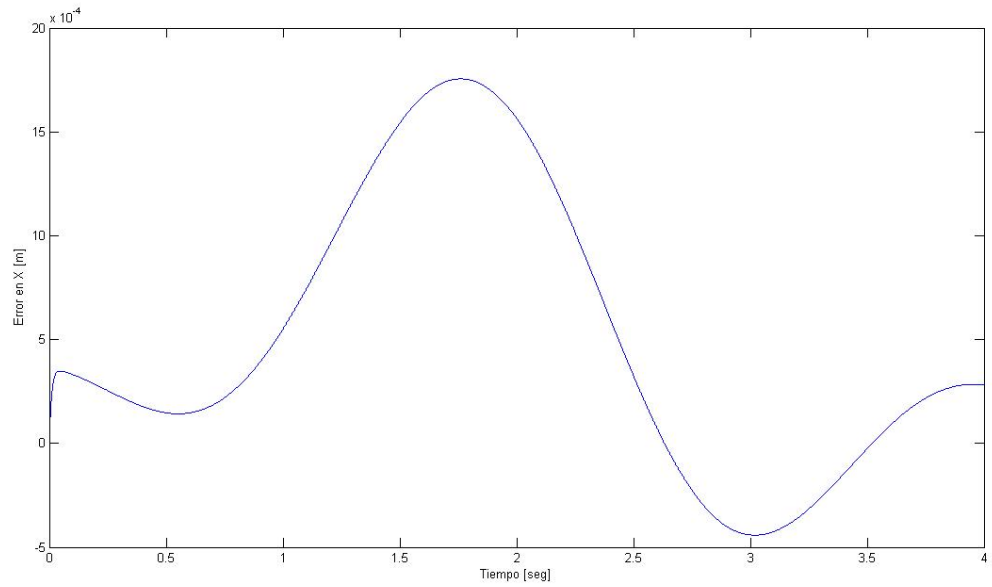


Figura 5.6: Error en la coordenada x del efector final en trayectoria circular -

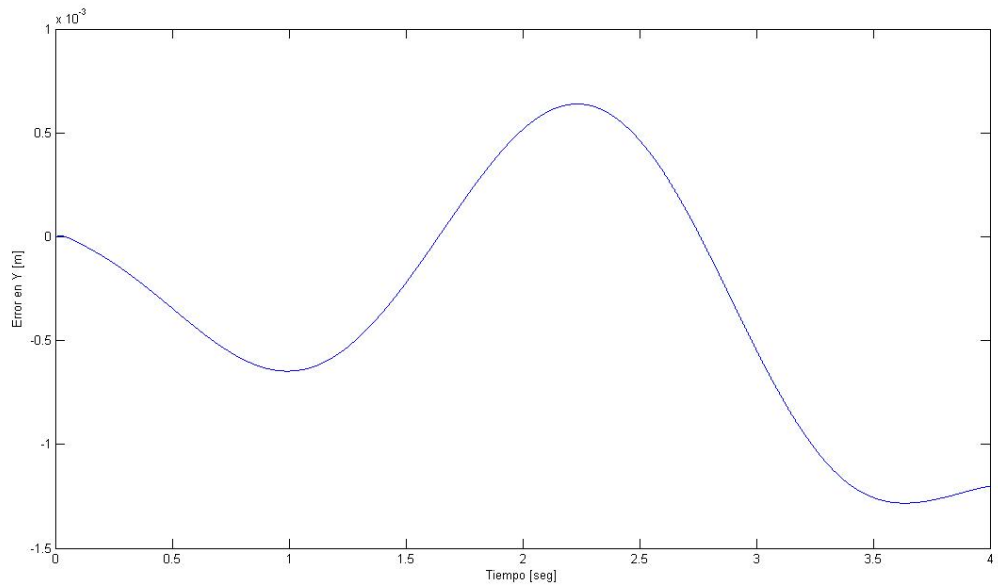


Figura 5.7: Error en la coordenada y del efector final en trayectoria circular -

5. RESULTADOS

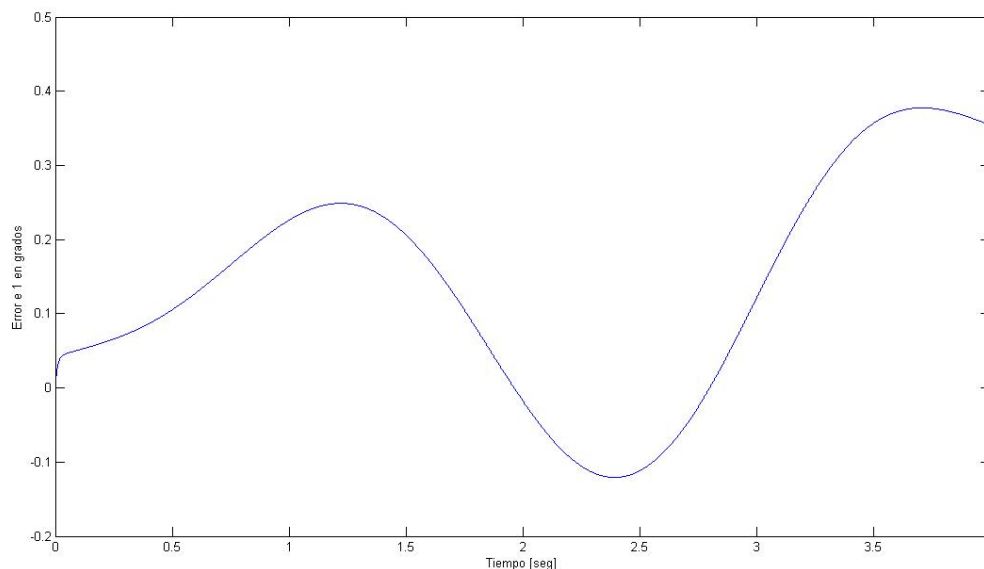


Figura 5.8: Error de posición actuador 1 con PD trayectoria circular -

En la gráfica de la Figura 5.8, aunque es suave, podemos notar que el error crece y decrece. Sin embargo termina en un punto muy alejado de la posición inicial. Este comportamiento puede no ser deseado y traer inconvenientes, pues en el caso de tareas repetitivas, para una aplicación industrial, este error puede ser causa de un paro en la producción y la necesidad de recalibrar los sensores. El tiempo asociado puede traducirse a pérdidas importantes.

El error en el actuador 2, a diferencia del anterior, termina en un punto más cercano al deseado, pero de igual manera nunca logra mantenerse en cero. Cruza por cero dos veces pero eso es debido al cambio de dirección de la trayectoria.

Finalmente en el tercer actuador, podemos notar que termina en un punto cercano a cero. Aunque el problema es claro, que en ninguno de los tres actuadores logra ser cero el error en tiempo finito. Cruza en cierto punto (segundo 1 y 2.5) pero eso se debe a la trayectoria y su cambio de dirección.

5.2.3 Simulación trayectoria recta con PD+Twisting

En esta subsección se presentan los resultados de simular un seguimiento de trayectoria recta, pero ahora con el control Twisting aumentado al control nominal, de una

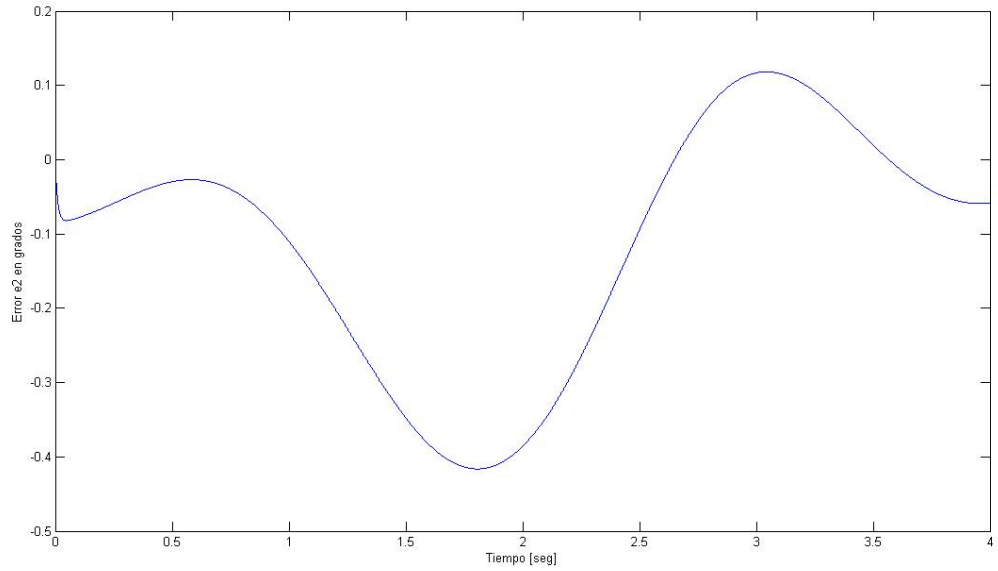


Figura 5.9: Error de posición actuador 2 con PD trayectoria circular -

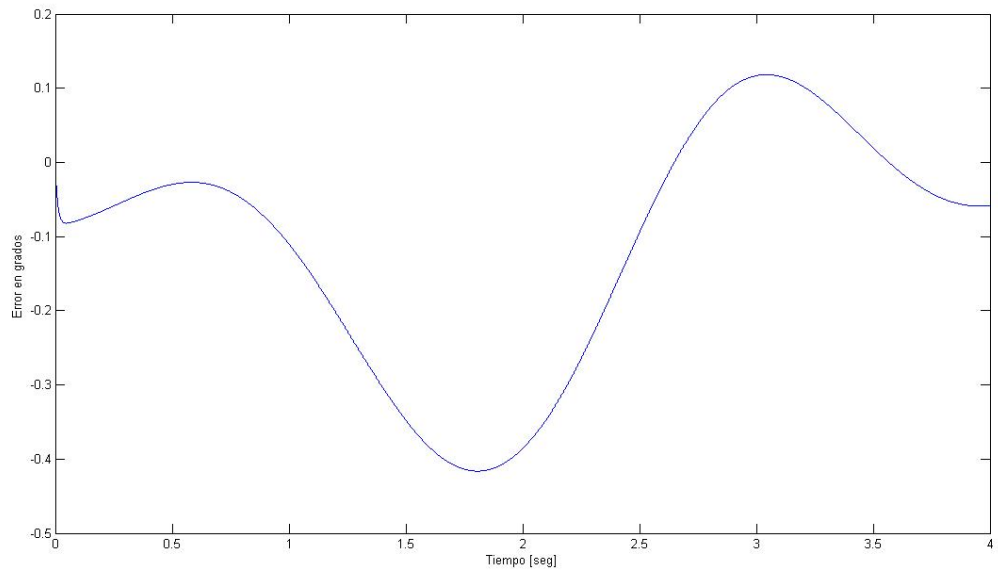


Figura 5.10: Error de posición actuador 3 con PD con PD trayectoria circular

5. RESULTADOS

coordenada (x_0, y_0) a otra coordenada (x_f, y_f) del efector final.

La trayectoria recta va de las coordenadas $(0.2165, 0.250)$ como punto inicial, a las coordenadas $(0.2465, 0.250)$ como punto final, ambas en unidades de metros. Los resultados fueron los siguientes:

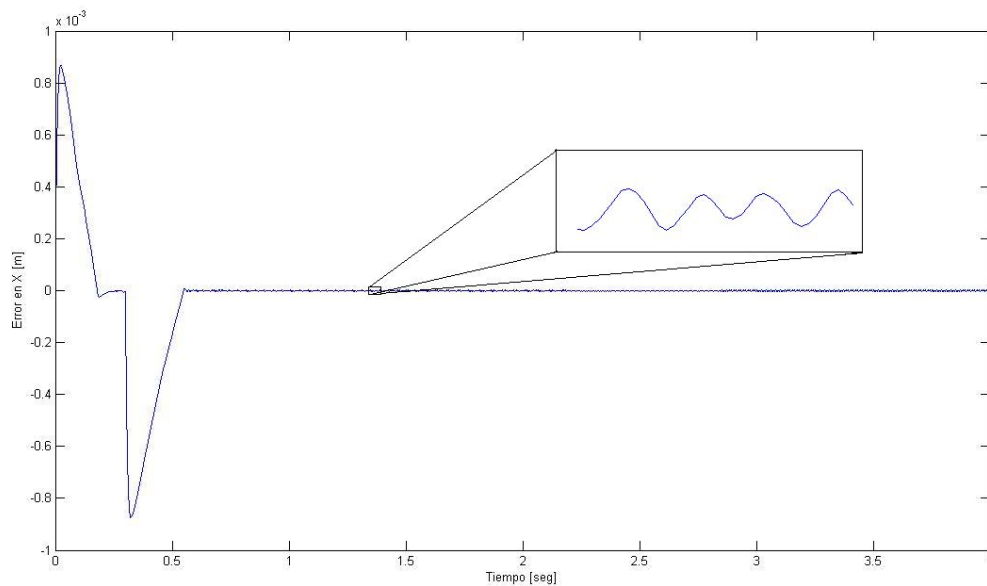


Figura 5.11: Error en la coordenada x del efector final en trayectoria recta -

En esta gráfica podemos notar la primera diferencia con el controlador clásico. En este caso sí llega en un tiempo finito a cero, y este se mantiene después de 0.6 segundos. Es clara la ventaja del controlador, sin embargo también se nota el efecto del "chattering" una vez que alcanza el cero. El acercamiento de la Figura 5.11 nos permite observar a detalle la forma de la oscilación en el error de la coordenada x .

En la gráfica que corresponde a la coordenada y del efector final, podemos observar que esta también presenta el efecto del "chattering". Es lógico pues el algoritmo compensa cualquier incertidumbre no modelada, y cualquier desviación a lo largo de su trayectoria es suficiente para que entre en el modo deslizante. El acercamiento hecho a la figura 5.12 a la oscilación nos permite apreciar su forma, y podemos notar que a diferencia de la Figura 5.11 esta oscilación tiene una mayor frecuencia.

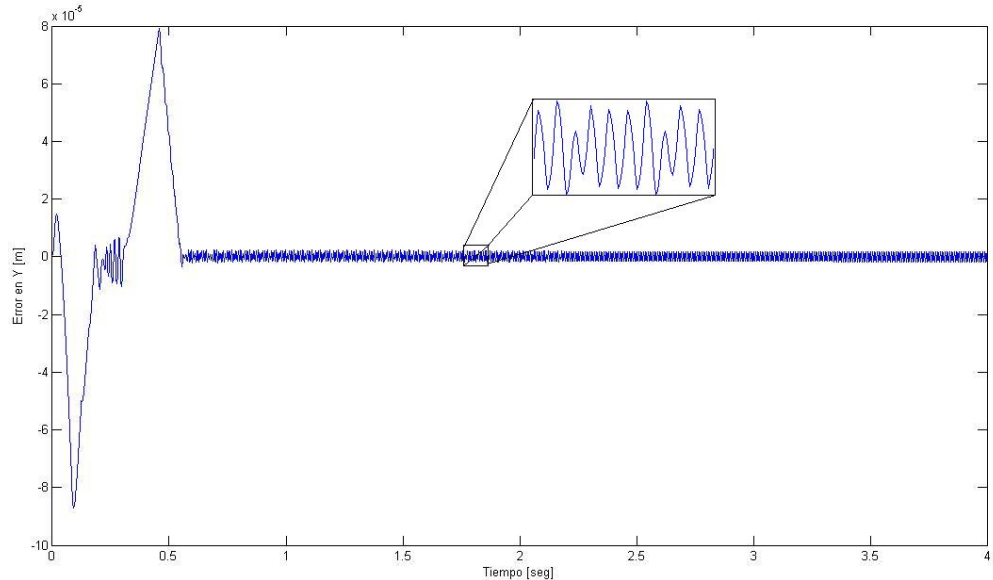


Figura 5.12: Error en la coordenada y del efector final en trayectoria recta -

También es interesante ver que sucede en cada actuador. El error de posición en cada uno de los motores se en las figuras correspondientes.

Si observamos la Figura 5.13, correspondiente al error de posición en el motor 1, se tienen dos picos importantes durante los primeros 0.5 *segundos*. Los picos son en diferentes sentidos (debido al cambio de signo), pero a partir del segundo pico, regresa a una vecindad del origen y ahí se mantiene oscilando.

En la Figura 5.14, que corresponde al error de posición en el actuador 2, podemos ver mayor amplitud en las oscilaciones en la que al igual que en el error del actuador 1, presenta dos picos importantes durante los primeros 0.5 *segundos* de simulación.

Al observar la Figura 5.15, correspondiente al error de posición en el actuador 3, notamos de nuevo dos picos durante los primeros 0.5 *segundos* de simulación pero en este caso hay una diferencia importante, la dinámica del error presenta menos oscilaciones que las dos anteriores. Sería interesante comparar estos cambios con respecto a los resultados experimentales.

De las Figuras 5.13, 5.14 y 5.15 podemos notar un comportamiento muy similar en cada motor, con mínimas diferencias en cuanto a la amplitud. En la figura 5.14, correspondiente a la dinámica del error en el actuador 2, es evidente el acentuamiento

5. RESULTADOS

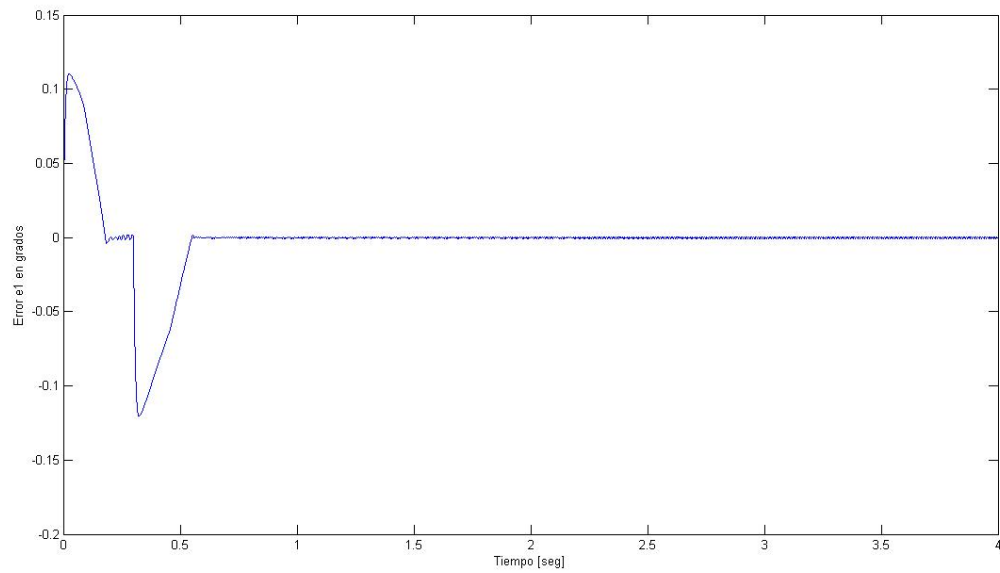


Figura 5.13: Error en la posición angular 1 con PD+Twisting, trayectoria recta

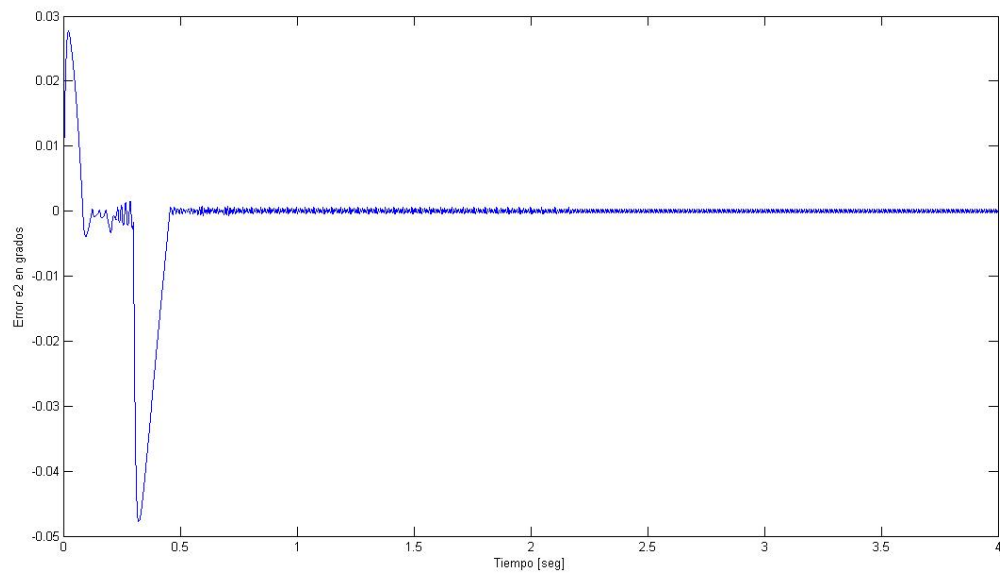


Figura 5.14: Error en la posición angular 2 con PD+Twisting, trayectoria recta

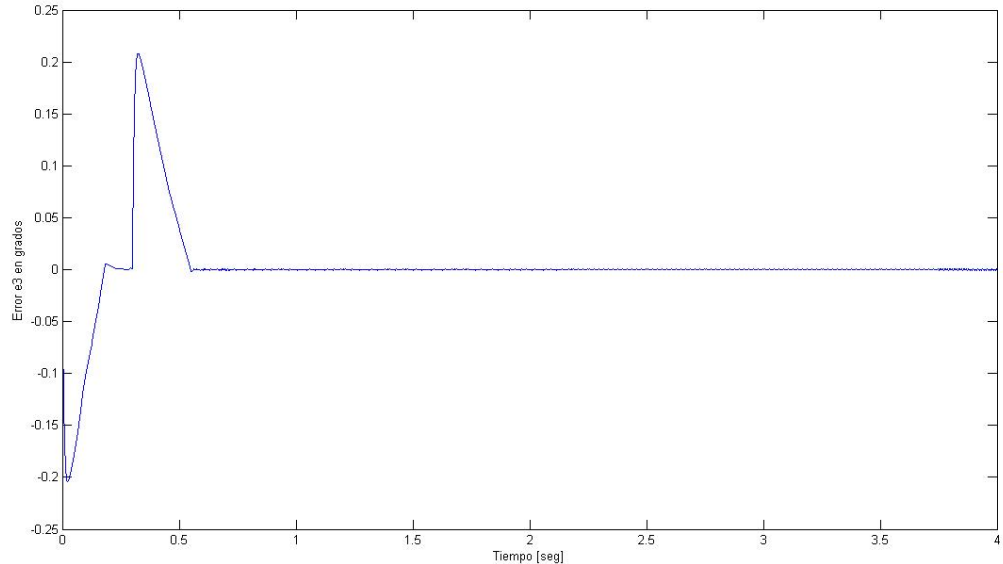


Figura 5.15: Error en la posición angular 3 con PD+Twisting, trayectoria recta

del efecto del "chattering" a diferencia de las gráficas de las figuras 5.13 y 5.15.

Podemos apreciar en las tres gráficas que las tres presentan picos al principio. Las que muestran un pico más alto son las gráficas de las figuras 5.18 y 5.17, mientras que la grafica 5.16 muestra un menor pico. Pero las tres gráficas entran en un estado de conmutación a partir de 0.5 [s], a partir de ese tiempo la amplitud de la oscilación se mantiene en límites más o menos constantes.

5.2.4 Simulación trayectoria circular con PD+Twisting

En esta subsección se presentan los resultados de simular un seguimiento de trayectoria recta, ahora con el algoritmo Twisting aumentado y con un radio $R = 0.03[m]$, centrada alrededor de las coordenadas $(0.2165, 0.250)$.

Las gráficas de las Figuras 5.19 y 5.20 corresponden a los errores de posición en las coordenadas del efector final. El error se define de igual forma que en los casos anteriores.

Podemos notar de la figura 5.19 un cambio relativamente suave en esta gráfica, justo a después de 1 [seg] de simulación. También al principio es notorio un pico pero

5. RESULTADOS

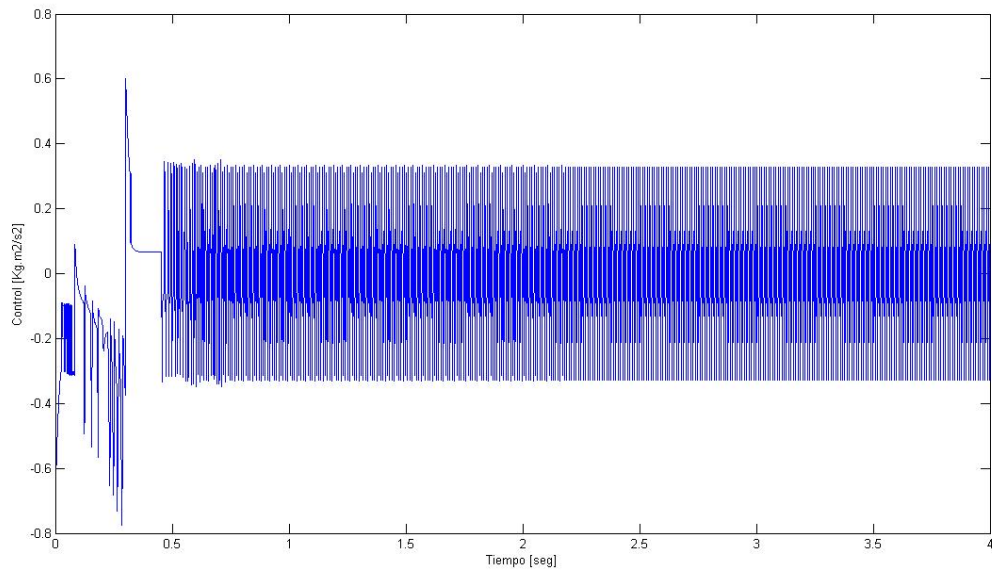


Figura 5.16: Control en actuador 1 con PD+Twisting, trayectoria recta -

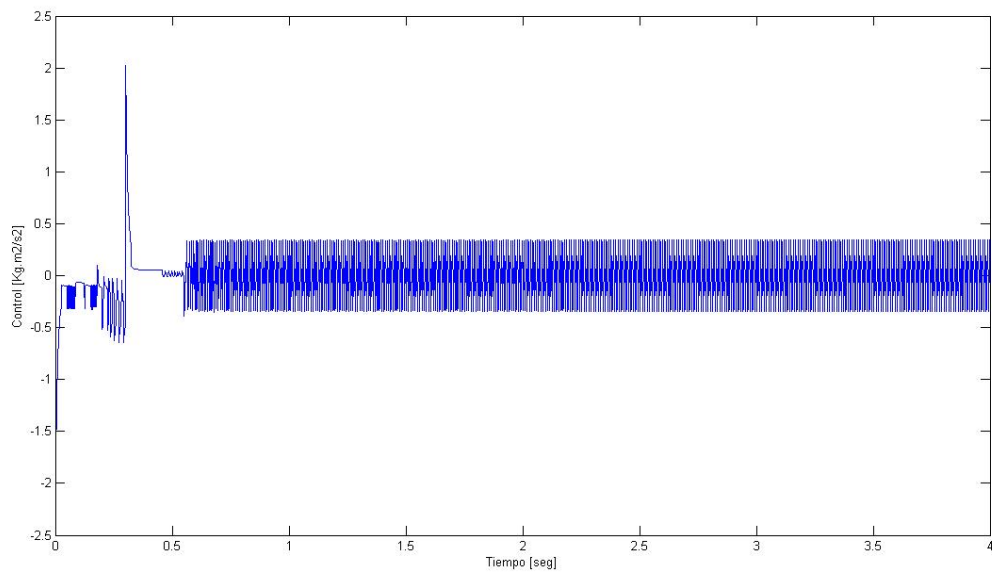


Figura 5.17: Control en actuador 2 con PD+Twisting, trayectoria recta -

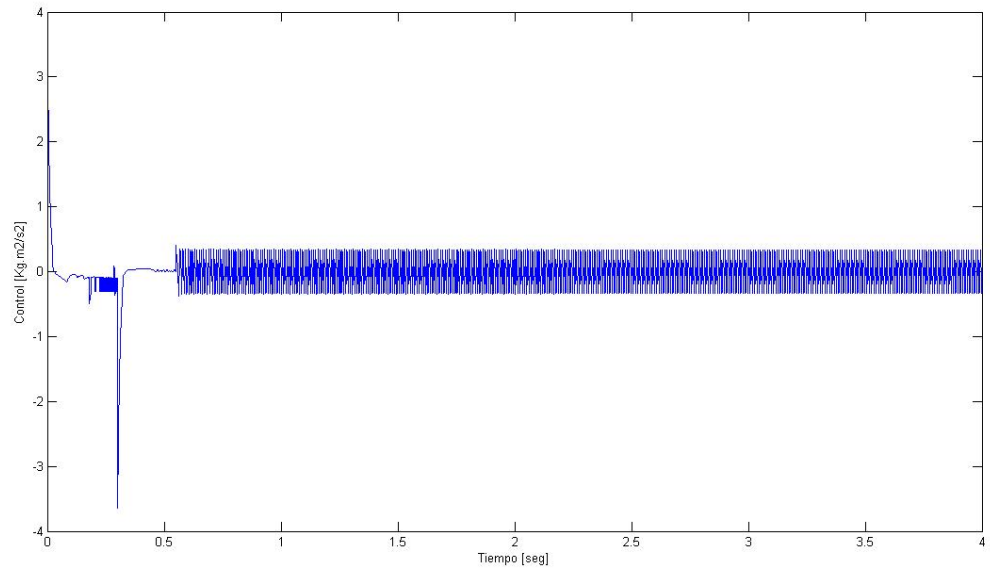


Figura 5.18: Control en actuador 3 con PD+Twisting, trayectoria recta -

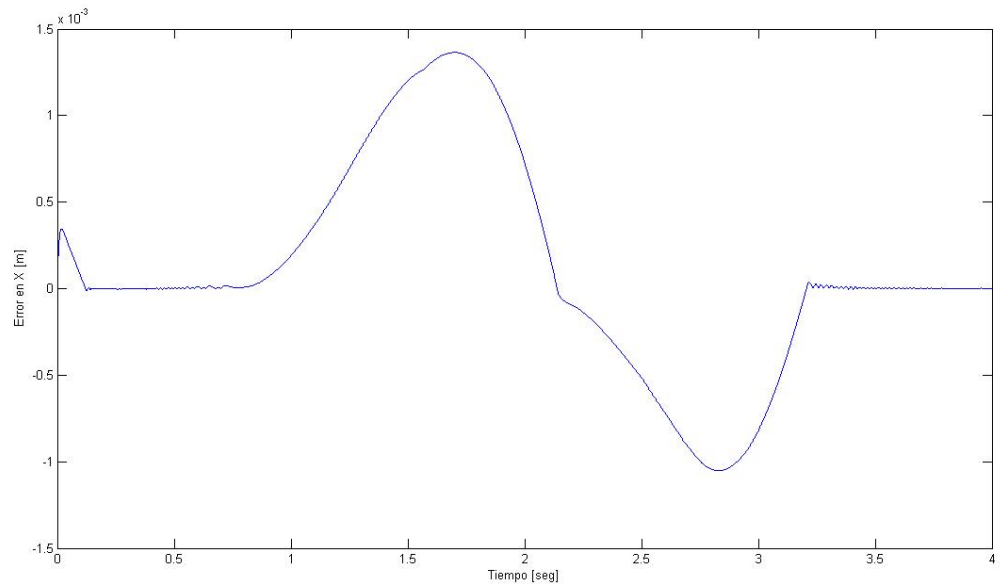


Figura 5.19: Error en la coordenada x del efector final en trayectoria circular -

5. RESULTADOS

es mínimo comparado con la desviación posterior, en 3 [s] el control puede regresar el error a un valor de cero, propiedad que el controlador clásico no tiene.

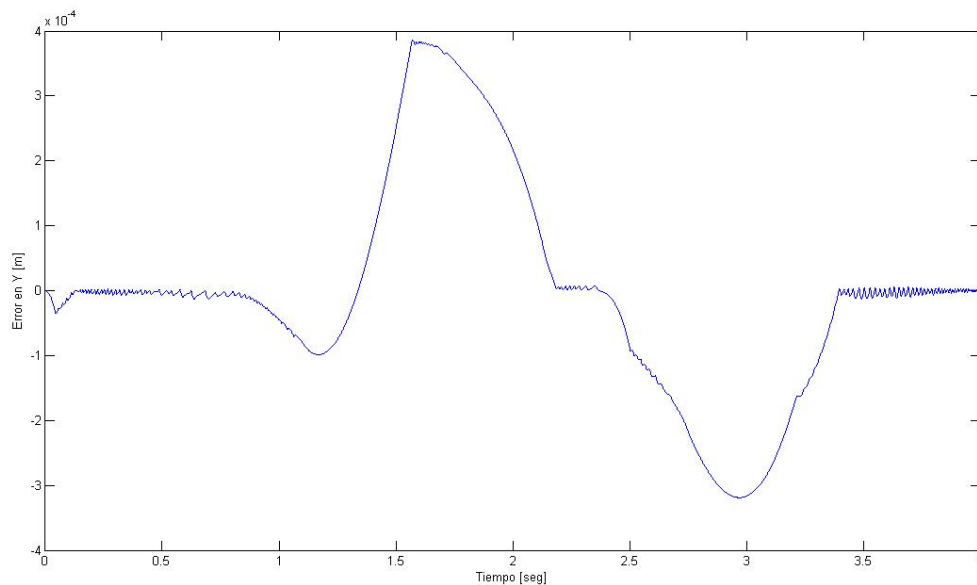


Figura 5.20: Error en la coordenada y del efector final en trayectoria circular -

En Figura 5.20 que corresponde al error en la posición y del efector final, se nota un importante cambio. El error es mucho menor, como se puede apreciar al observar la escala de la gráfica. Aunque los cambios son un poco más abruptos. Lo importante es que al final, igual que en la Figura 5.19 el control es capaz de regresar a cero el valor del error.

En las Figuras 5.21, 5.22 y 5.23 se muestran los errores en grados de las posiciones angulares de los motores. y el error lo definimos igual que en los casos anteriores.

En la Figura 5.21 se pueden apreciar varios cambios tanto positivos como negativos, esto debido a los cambios de trayectoria que implica un seguimiento circular. Y cuando este llega a cero también se puede apreciar el efecto de "chattering"

En la figura 5.22 que corresponde al error de posición en el segundo actuador, se pueden notar dos importantes sobrepasos. Estos sobrepasos corresponden al tiempo en el que el actuador requiere mas fuerza y es difícil mantener el error en cero.

La gráfica de la figura 5.23 es muy parecida al comportamiento del segundo actuador en 5.22, pero en este caso las trayectorias van en sentido contrario. Al final el error

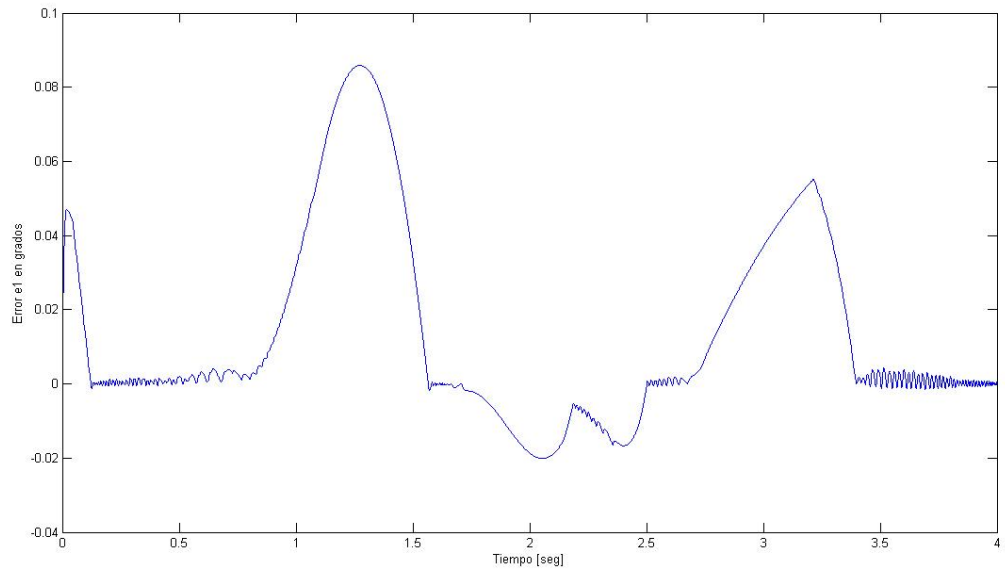


Figura 5.21: Error en la posición angular 1 con PD+Twisting, trayectoria circular -

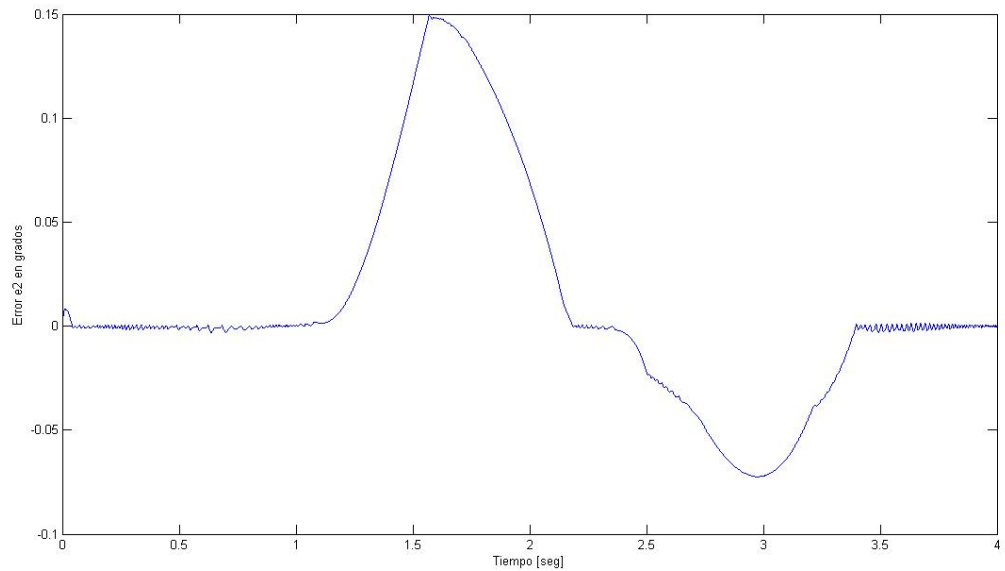


Figura 5.22: Error en la posición angular 2 con PD+Twisting, trayectoria circular -

5. RESULTADOS

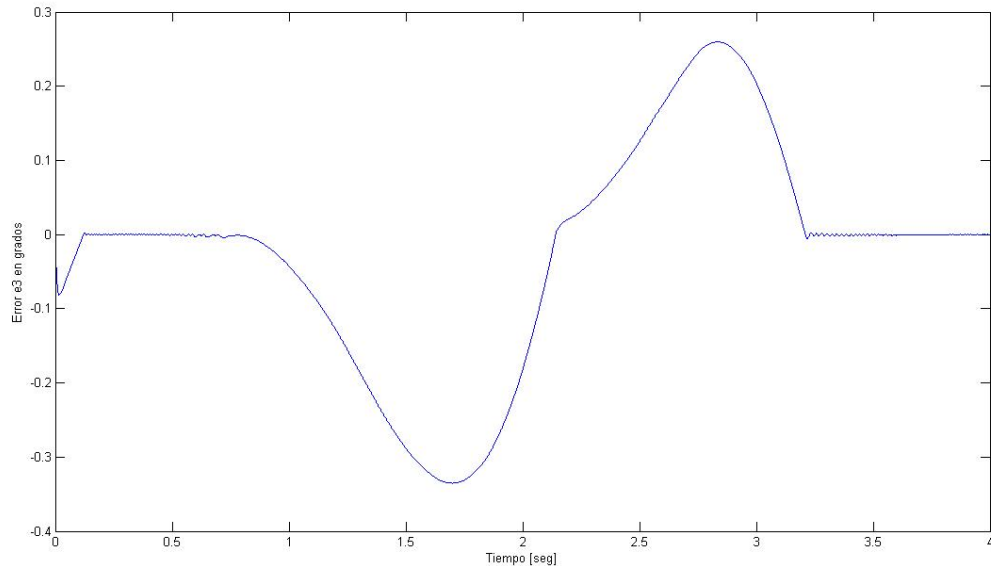


Figura 5.23: Error en la posición angular 3 con PD+Twisting, trayectoria circular -

regresa a ser cero.

En las figuras 5.24, 5.25 y 5.26 se muestra el comportamiento de la señal de control, en ellas se describe el esfuerzo requerido para las tareas del robot.

En la figura 5.24 que describe el comportamiento de la señal de control para el primer actuador, se pueden apreciar importantes oscilaciones con alta frecuencia y notable amplitud.

En cuanto a la figura 5.25 describe un comportamiento similar, con la diferencia de tener segmentos en los que la amplitud de la oscilación es menor. Estos segmentos son tres, y el primero va de 0.5 a 1.5 [s], el segundo de 1.7 a 2.4 [s] y el tercero de 2.6 a 3.4 [s].

Finalmente en la figura 5.26 se encuentra la gráfica que corresponde a la señal de control en el tercer actuador. Esta gráfica, como las anteriores presenta oscilaciones, pero la diferencia es que tiene un segmento de tiempo mayor en el que esas oscilaciones disminuyen la amplitud. Ese segmento va de 0.7 a 3.2 [s], con un sobresalto en 2.1 [s].

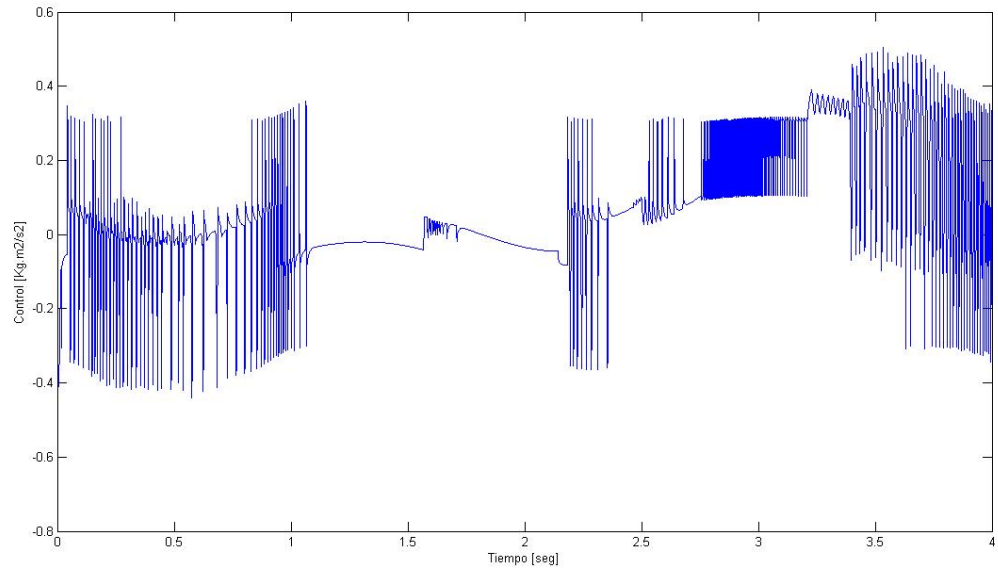


Figura 5.24: Control en actuador 1 con PD+Twisting, trayectoria circular -

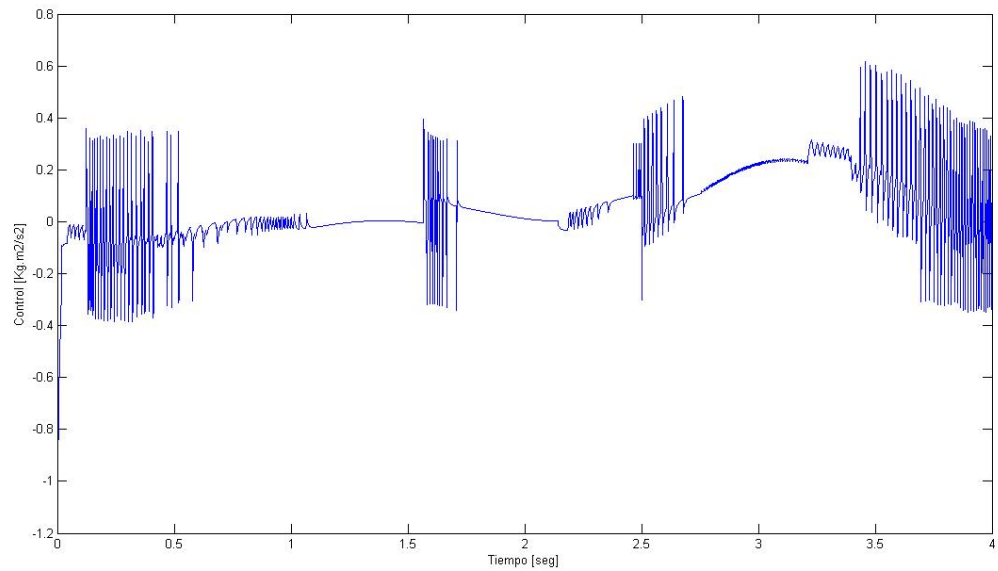


Figura 5.25: Control en actuador 2 con PD+Twisting, trayectoria circular -

5. RESULTADOS

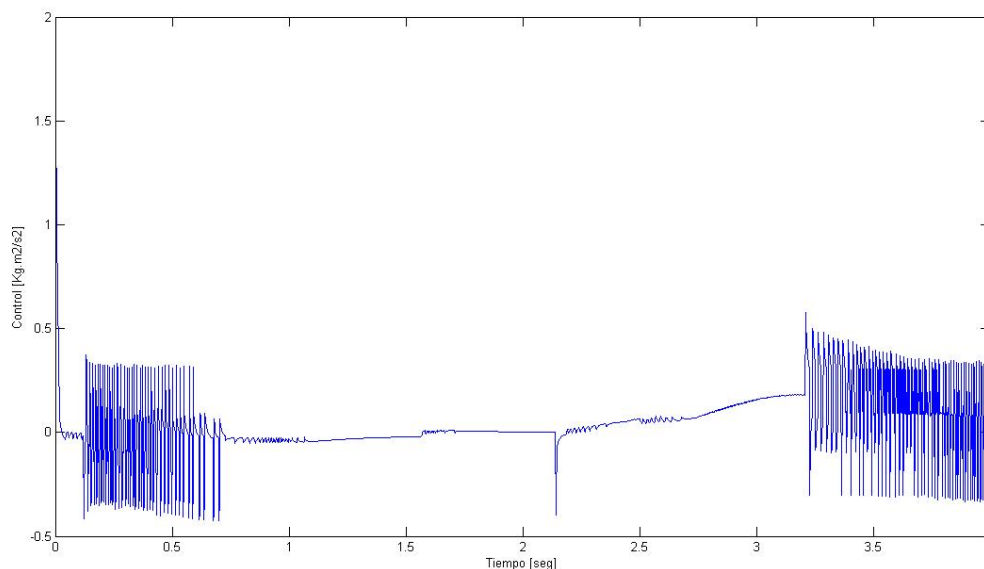


Figura 5.26: Control en actuator 3 con PD+Twisting, trayectoria circular -

5.3 Resultados experimentales

En esta última sección se presentan los experimentos con el control nominal y con el control Twisting, los parámetros son iguales a los de las simulaciones.

Los experimentos físicos para este trabajo se llevaron a cabo en la unidad GPM2002 de Googol Tech. Utilizando para la entrada/salida de datos la tarjeta GT-400-SV y la programación de los algoritmos de control en el entorno de Visual C++.

Debido a las limitaciones actuales relativas a las herramientas de programación (debido a que esto acarrea un tiempo largo para trabajarlo), sólo se pueden mostrar los errores de posición en los actuadores.

5.3.1 Experimento trayectoria recta con PD nominal

A continuación se presentan los resultados del experimento con seguimiento de trayectoria recta. En las figuras siguientes se muestra el error de cada motor para este experimento. La trayectoria recta va de las coordenadas (0.2165, 0.250) como punto inicial, a las coordenadas (0.2465, 0.250) como punto final, ambas en unidades de metros. Los experimentos son idénticos a las simulaciones relacionados con las trayectorias

que debe seguir el robot.

En la Figura 5.27, que es el error correspondiente al motor 1 usando el control Proporcional Derivativo, podemos apreciar un error prácticamente constante. Este error se mantiene durante todo el experimento, sin disminuir ni aumentar un valor considerable. Se mantiene en un aproximado de 11^{-3} grados.

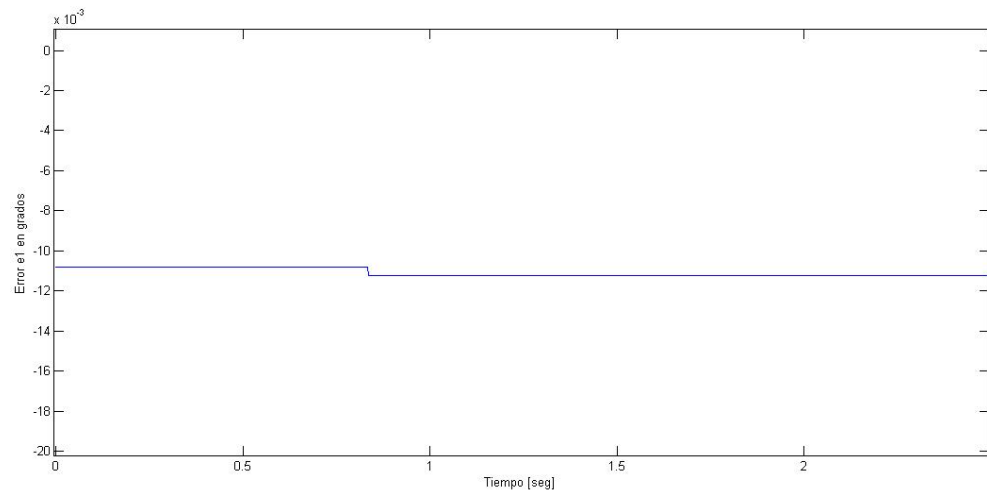


Figura 5.27: Error en actuador 1 de resultado experimental PD trayectoria recta -

Un comportamiento muy similar se puede observar en la figura 5.28, que corresponde al error de posición en el segundo actuador, con apenas mínimas variaciones a la mitad del experimento. En la gráfica pudieran parecer saltos abruptos, pero no lo son, el parecido se debe a la escala y resolución del encoder.

En la figura 5.29 podemos observar un comportamiento similar a las anteriores, variando de manera mínima a lo largo de todo el experimento. Con la diferencia de que la figura 5.29 correspondiente al error de posición en el actuador 3, tiene un error menor al de los actuadores 1 y 2.

Con esta primera parte de los experimentos concluida, podemos hacer observaciones importantes: a diferencia de las simulaciones que nos muestran resultados de curvas suaves, en los experimentos obtenemos datos casi constantes, con cambios mínimos. Esto lo podemos explicar debido a que en nuestra plataforma de experimentación, la etapa de potencia de cada motor tiene su propio controlador independiente, ese

5. RESULTADOS

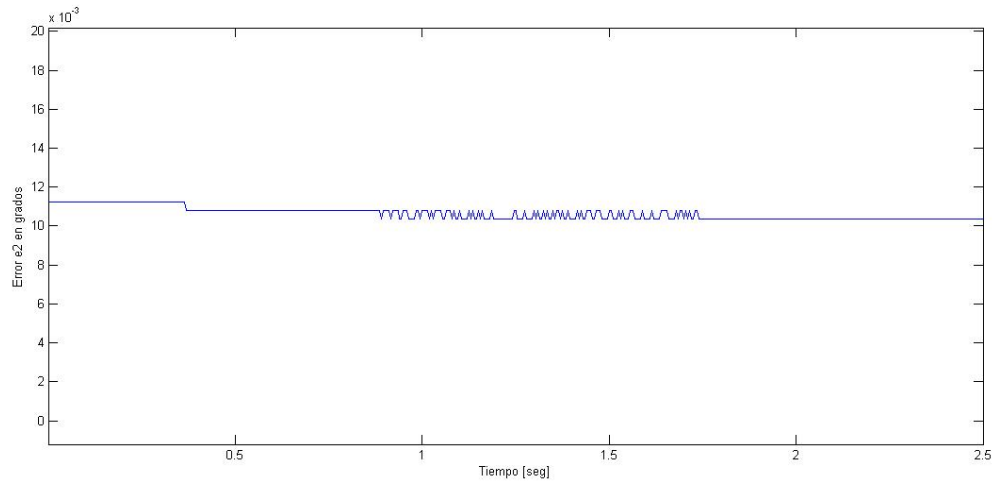


Figura 5.28: Error en actuador 2 de resultado experimental PD trayectoria recta -

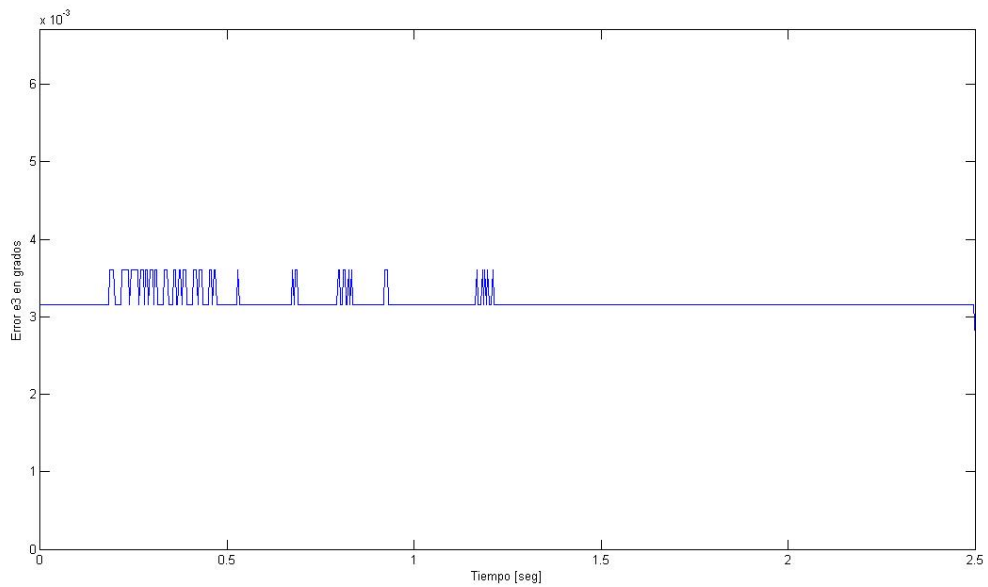


Figura 5.29: Error en actuador 3 de resultado experimental PD trayectoria recta -

controlador nunca se modificó. Otro punto importante es que el error nunca converge a cero en tiempo finito y es algo que tienen en común las simulaciones y los experimentos.

5.3.2 Experimento trayectoria circular con PD nominal

En esta sección se presentan los resultados del experimento usando el controlador Proporcional derivativo y una trayectoria circular. La trayectoria circular tiene un radio de $0.03[m]$ y centro en las coordenadas $(0.250, 0.2156)$.

En la figura 5.30 observamos un resultado muy interesante, el error se mantiene en valores casi constantes como en el caso de la recta. La figura 5.30 que corresponde al error de posición experimental de la trayectoria circular, nos muestra valores sumamente parecidos a los del mismo actuador, pero para una trayectoria recta. No se trata de un error de experimentación, tampoco son las mismas gráficas, simplemente esos fueron los resultados que se obtuvieron.

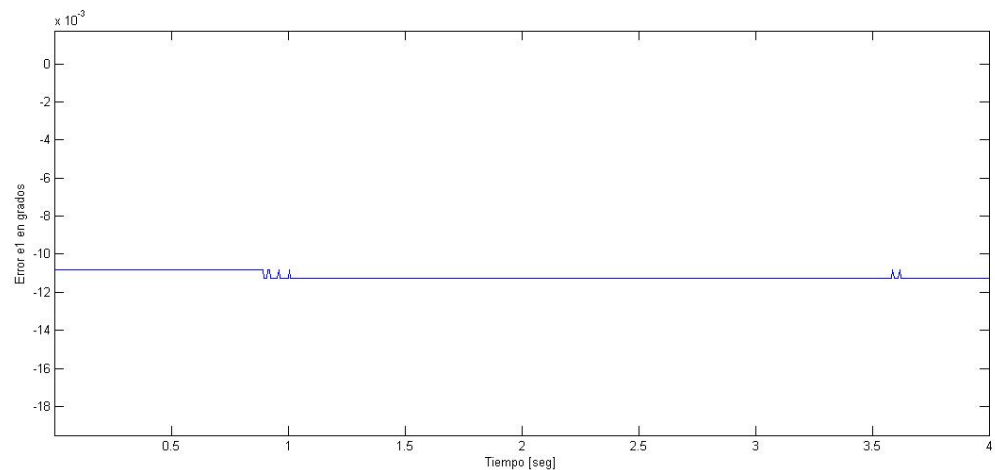


Figura 5.30: Error en actuador 1 de resultado experimental PD trayectoria circular -

En la figura 5.31 observamos un resultado muy similar para el error de posición en el actuador 2, cabe destacar que los signos están invertidos; esto se puede explicar fácilmente debido a que el signo del error depende del tipo de trayectoria que se dibuje y la dirección de la misma.

5. RESULTADOS

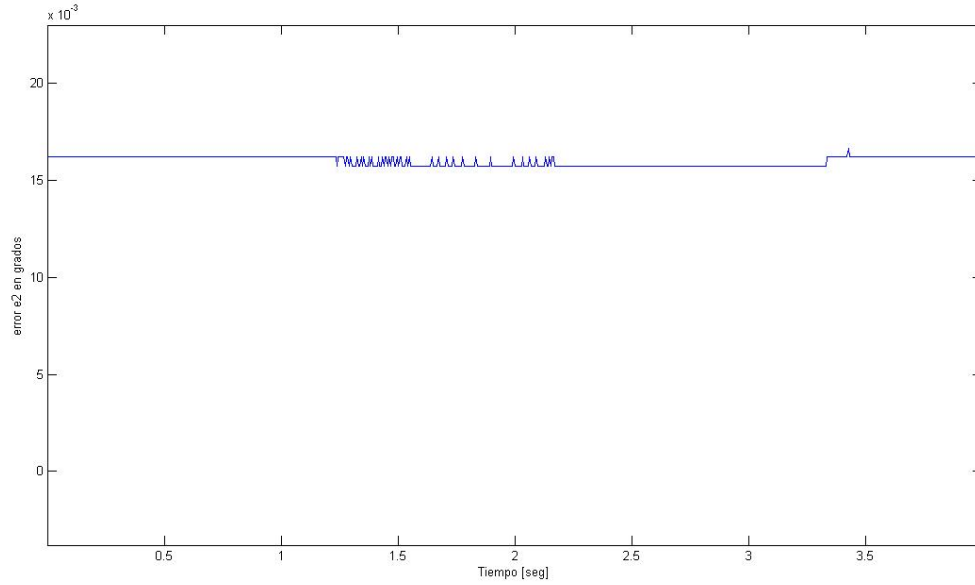


Figura 5.31: Error en actuator 2 de resultado experimental PD trayectoria circular -

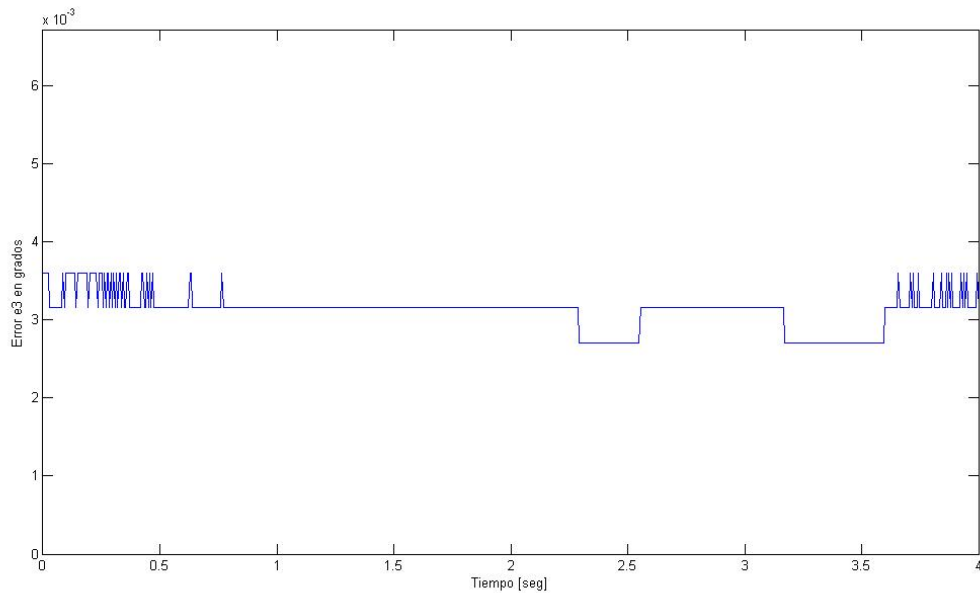


Figura 5.32: Error en actuator 3 de resultado experimental PD trayectoria circular -

En el error de posición para el actuador 3, que se puede ver en la figura 5.32, notamos un valor menor de error que en los casos anteriores. Análogamente al caso de la línea recta, también resultó ser menor el error para este actuador. Se pueden apreciar diminutos cambios en la posición, sin embargo considerando la escala, éstos son mínimos. La dinámica del error se mantiene casi constante, debido a la poca diferencia de valores que toma.

En esta sección, correspondiente a la experimentación de la trayectoria circular con el controlador clásico Proporcional Derivativo, obtenemos importantes conclusiones. La primera de ellas es que en el caso de la experimentación, no existe gran diferencia en la dinámica del error, tratándose de una línea o un círculo, esto lo podemos explicar debido a que el sistema de amplificadores para los motores cuenta con un sistema independiente de control. Consideración que se debe de tomar en cuenta para trabajos futuros. Dicho sistema de control, también es un PID, evidentemente es más rápido que el que cuenta el filtro digital de la tarjeta GT-400S-V.

5.3.3 Experimento trayectoria recta con PD+Twisting

En esta sección se presentan los resultados de los experimentos usando el Control Proporcional Derivativo con el Twisting agregado. Al igual que en los casos anteriores, se lleva el efector final del robot de las coordenadas (0.250, 0.2165) a las coordenadas (0.250, 0.2465).

En la figura 5.33, que corresponde al error de posición del actuador 1, podemos apreciar con éxito el resultado esperado. Ciertamente, no es exactamente igual a las simulaciones, sin embargo cumple con lo que se espera de un controlador Twisting. Cabe mencionar el sobrepaso que tiene esta dinámica del error, en un valor de -6.2^{-3} *grados*, justo en el segundo 0.25 del experimento. Inmediatamente después, la dinámica del error tiende a acercarse a cero, pero se queda oscilando en una vecindad cercana. Se aprecia en la gráfica que tiene un cierto sobrepaso, pero inmediatamente después de 0.5[s] el error alcanza el cero en tiempo finito. Y entonces este error mantiene una oscilación. Conseguimos no sólo convergencia en tiempo finito, también mayor precisión puesto que el error es menor.

En contraste, la figura 5.34, correspondiente al error de posición en el actuador 2, no presenta el comportamiento propio de un modo deslizante. Esta gráfica se comporta

5. RESULTADOS

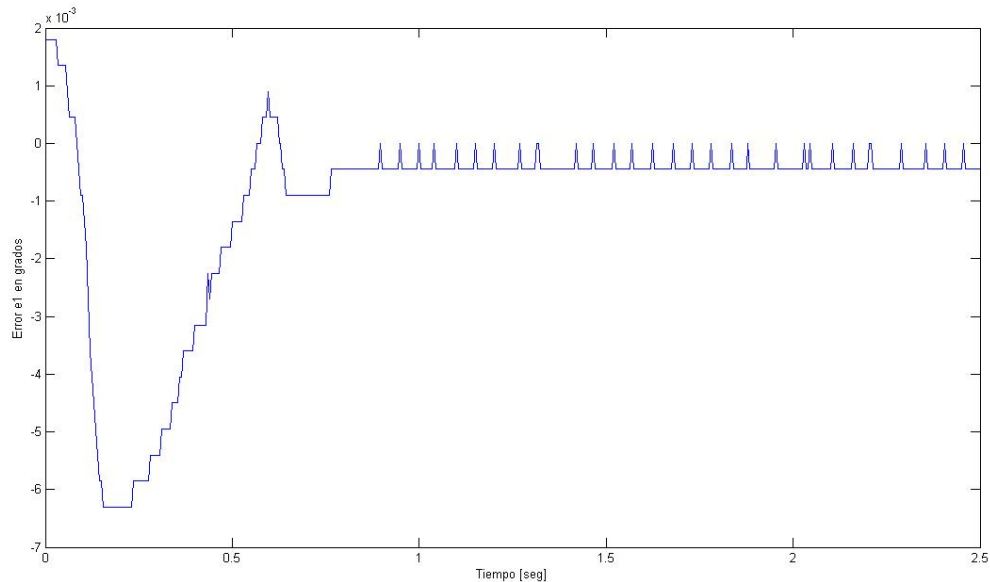


Figura 5.33: Error en actuador 1 de resultado experimental PD+Twisting trayectoria recta -

de una manera casi idéntica al caso Proporcional Derivativo, sin Twisting agregado.

Algo muy similar sucede con la figura 5.35, la dinámica del error no converge a cero en un tiempo finito. De hecho se mantiene en valores no muy alejados unos de otros, pero el error sí es menor que el caso sin Twisting agregado. A pesar de ser un error mínimo, nuestras expectativas eran las de una convergencia a cero en tiempo finito, este fenómeno podría estudiarse a detalle en trabajos futuros.

En esta sección se revisaron los resultados obtenidos de haber probado el controlador Proporcional Derivativo con el algoritmo Twisting agregado. Si bien dos de estas gráficas no cumplen con lo que se tenía esperado, puede tener su explicación tanto en la programación como en la implementación física, inclusive en la dinámica del sistema. Otro de los posibles motivos por las que no convergieron a cero en tiempo finito, es que al ser un sistema mecánico con restricciones holonómicas, existen términos importantes que no están siendo considerados.

5.3 Resultados experimentales

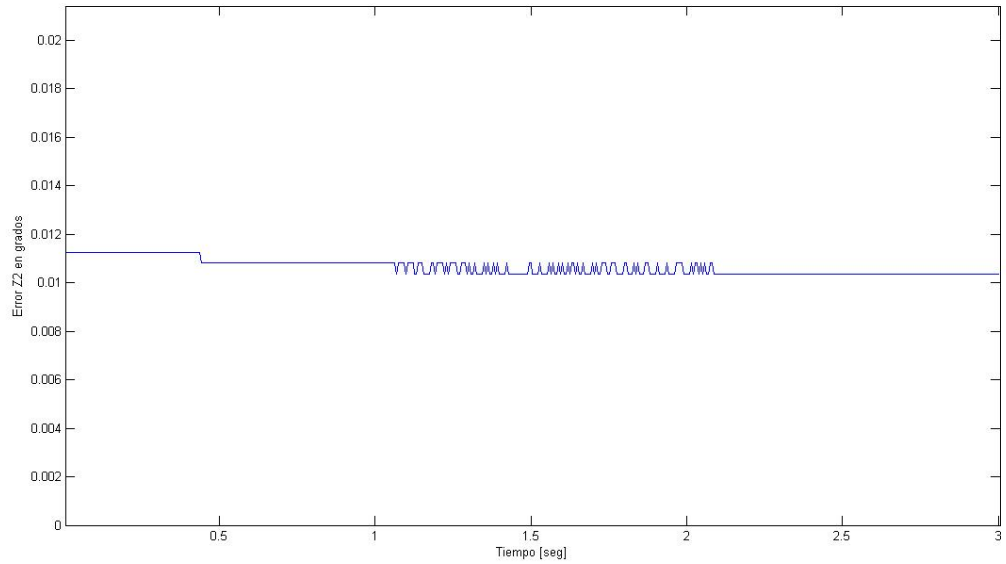


Figura 5.34: Error en actuador 2 de resultado experimental PD+Twisting trayectoria recta -

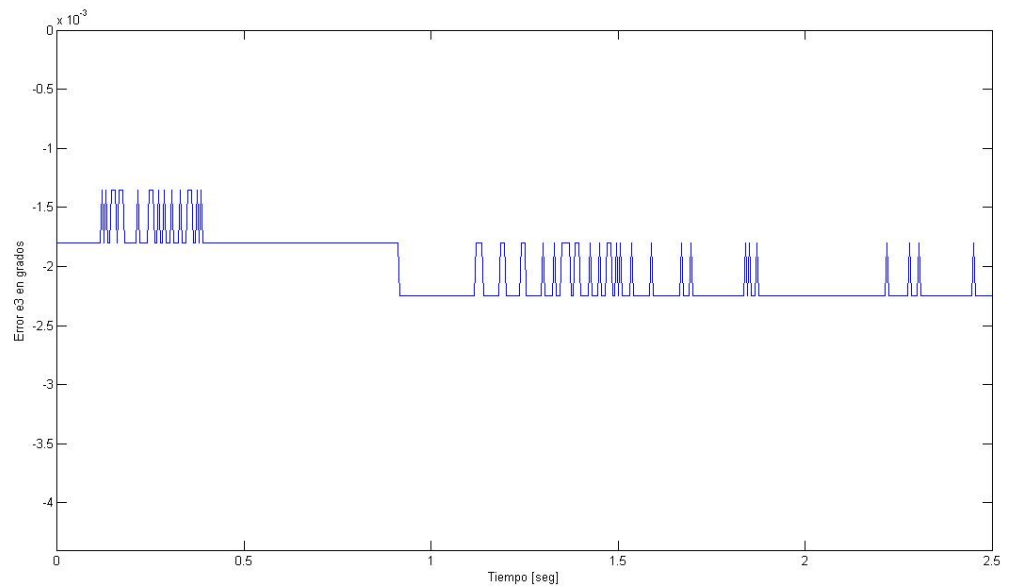


Figura 5.35: Error en actuador 3 de resultado experimental PD+Twisting trayectoria recta -

5. RESULTADOS

5.3.4 Experimento trayectoria circular con PD+Twisting

En esta sección se presentan los resultados de la última réplica experimental de lo probado en simulaciones, el seguimiento de una trayectoria circular, ahora con el controlador Twisting agregado al Proporcional Derivativo. Esta trayectoria circular, tiene un radio de $0.03[m]$ y centro en las coordenadas $(0.250, 0.2165)$. Además este giro para la trayectoria circular es en sentido contrario a las manecillas del reloj. El experimento dura 4 segundos, tiempo que tarda en recorrer la vuelta completa y regresar al punto original.

En la figura 5.36, correspondiente al error de posición del actuador 1, se aprecia inmediatamente la conmutación de la dinámica del error. Se consigue llevar el error a cero en tiempo finito, y las oscilaciones alrededor de dicho valor son evidentes. Estas oscilaciones se encuentran entre -1^{-3} y 0.5^{-3} grados, con sobrepasos mínimos que se alejan por muy poco de esos límites. La no simetría de estos límites puede explicarse por las restricciones holonómicas del sistema, a diferencia de las simulaciones en las que los límites son simétricos alrededor del origen.

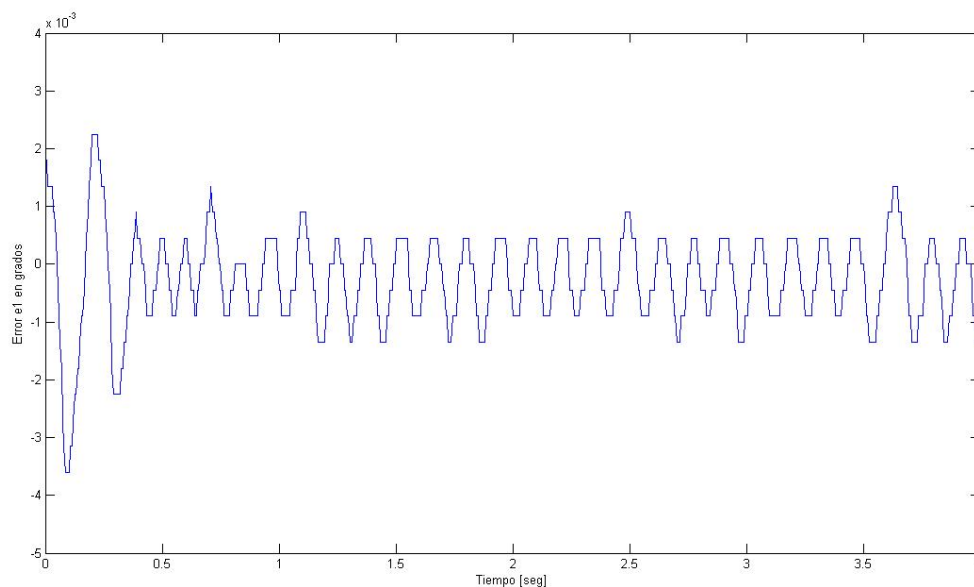


Figura 5.36: Error en actuador 1 de resultado experimental PD+Twisting trayectoria circular -

Si observamos la Figura 5.37, que corresponde al error de posición en el actuador 2, el error tarda un poco más en llegar a cero. Una una vez que la dinámica del error cruza por cero en una ocasión, vuelve a presentarse otra oscilación con un sobrepaso considerable de 11^{-3} *grados* justo en 0.5 [s], inmediatamente despues de ese tiempo la dinámica del error toma valores menores a 5^{-3} *grados*, y decrecen a medida que avanza el tiempo. Una diferencia notoria es que las oscilaciones no son las mismas que en caso anterior, son menos frecuentes. De hecho no podríamos asegurar que se tratan de oscilaciones periódicas como en el caso anterior, en éste la dinámica del error logra acercarse a cero pero nunca oscila alrededor del mismo. A diferencia con el controlador clásico, el error logra disminuirse mucho más, no se comporta como una constante, como lo fue el caso del PD sin Twisting.

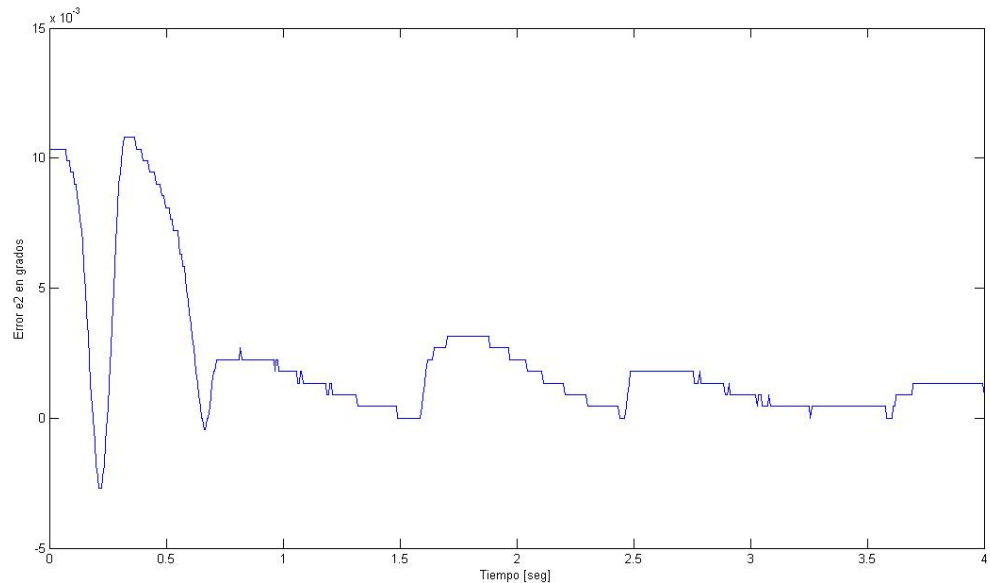


Figura 5.37: Error en actuador 2 de resultado experimental PD+Twisting trayectoria circular -

En la figura 5.38, que corresponde al error de posición del actuador 3, se puede apreciar un resultado contrastante con los dos anteriores. En los casos anteriores, se alcanzaba a cero el error en tiempo finito, sin embargo, para el caso de la figura 5.38 esto no sucede, incluso el comportamiento es muy parecido al caso anterior, en cuanto al patrón de onda se refiere. Cabe destacar que el error no se aproxima a cero, más

5. RESULTADOS

bien la dinámica del error se aproxima al valor de -0.013 *grados*. Fenómeno muy interesante, puesto que no se esperaba este resultado ni en simulaciones.

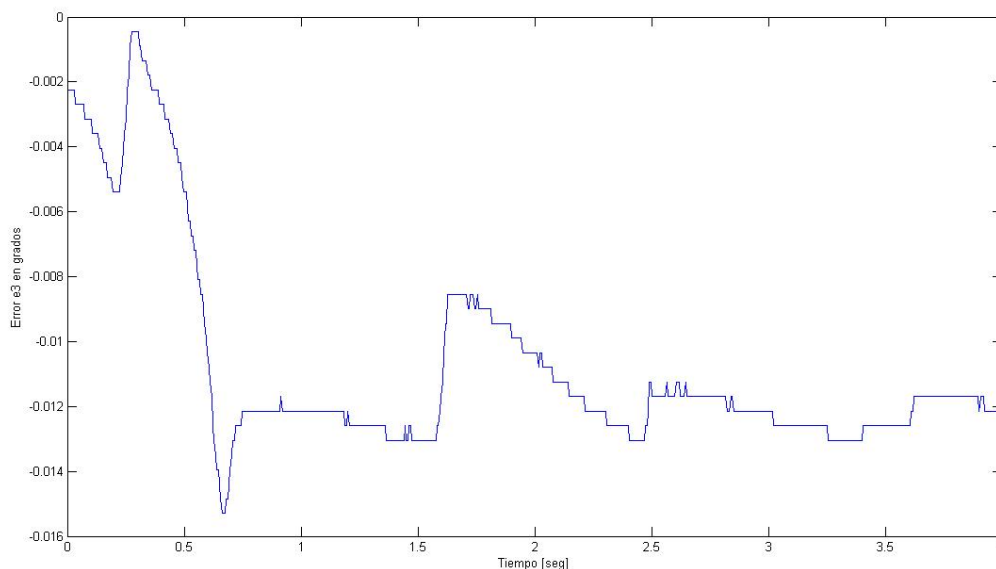


Figura 5.38: Error en actuador 3 de resultado experimental PD+Twisting trayectoria circular -

Este experimento nos ofrece resultados muy interesantes. Existen aún muchas preguntas en pie, pero quedan fuera del alcance de este trabajo responderlas con exactitud, por ejemplo el hecho de que dependiendo la trayectoria deseada puede o no que funcione el algoritmo Twisting. La experimentación tiene muchas diferencias con respecto a la simulación, eso es claro. El error en el actuador 3 es uno de los resultados más interesantes pues era totalmente inesperado, ni siquiera en simulación obtuvimos ese resultado, es evidente que en los tres actuadores surtió efecto el controlador Twisting. La principal diferencia fue el hecho de encontrar oscilaciones en las dinámicas del error y la disminución del mismo.

5.3.5 Experimentos con diferentes parámetros

Para finalizar el capítulo de experimentos, se muestran algunas pruebas con parámetros diferentes a los anteriores, éstas pruebas no fueron reproducidas en simulación. Se incluye esta última sección puesto que se consideró importante mostrar las diferencias

encontradas al variar los parámetros, tanto en el controlador como en las trayectorias deseadas. El primero de ellos corresponde al seguimiento de una trayectoria recta, con las mismas ganancias del controlador PD y del Twisting que se usaron en los experimentos anteriores, pero con la diferencia en que las coordenadas para el punto final es $(0.2465, 0.280)$. Es decir, en contraste a los experimentos anteriores, en los que sólo había movimiento en el eje x , ahora también hay movimiento en y .

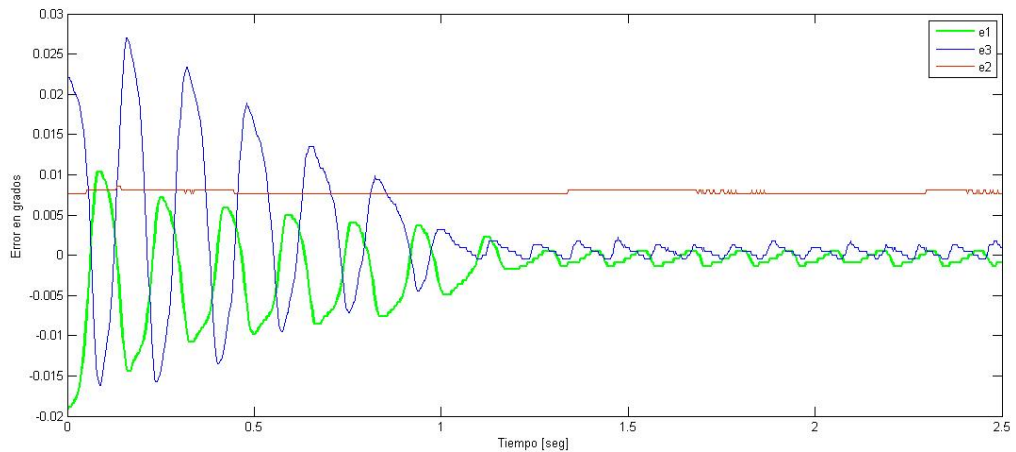


Figura 5.39: Experimento PD+Twisting trayectoria recta con diferentes coordenadas - se puede apreciar en una sola gráfica los tres errores de posición de los actuadores

En la figura 5.39 observamos los resultados experimentales de estos cambios. La dinámica del error en cada actuador, a diferencia de los resultados anteriores, en los que sólo uno de los tres actuadores conseguía llegar a cero el error en tiempo finito, ahora lo consiguen dos de ellos. Las amplitudes de las oscilaciones decrecen durante del primer segundo del experimento. El experimento dura 2.5 segundos, tiempo en que tarda el efector final en recorrer del punto inicial al punto final.

También se muestran los resultados de un experimento con trayectoria circular con radio de $0.06[m]$ (el doble del radio que los experimentos anteriores) y centro en $(0.250, 0.2165)$. Pero en este último experimento también se duplicaron las ganancias del controlador Twisting, pues con las que se usaron anteriormente, en muchas ocasiones se perdía el seguimiento de trayectoria y el sistema se inestabilizaba.

5. RESULTADOS

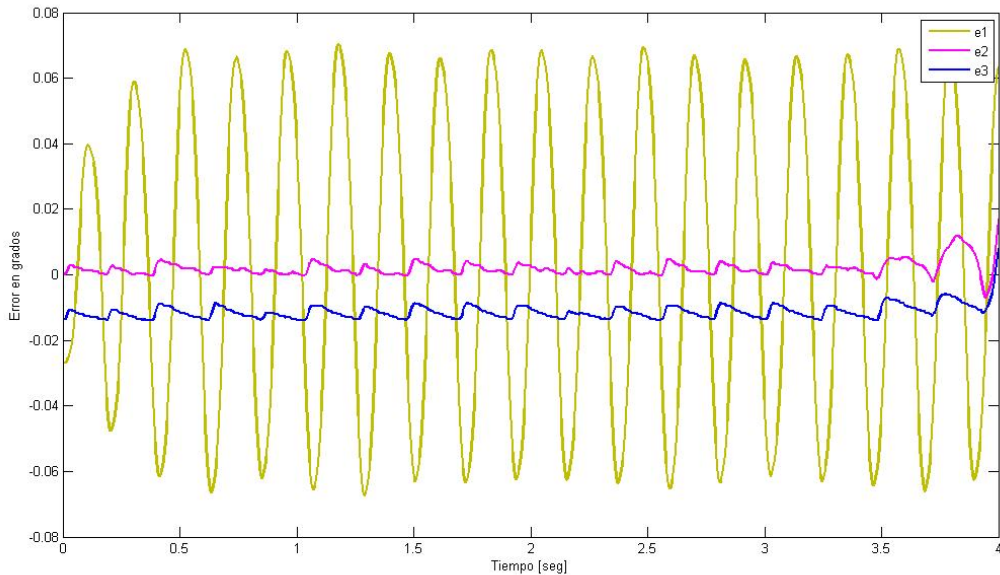


Figura 5.40: Experimento PD+Twisting trayectoria circular y alta ganancia -
Las oscilaciones en los actuadores son notorias

En el último experimento de esta sección, se puede apreciar en la Figura 5.40 oscilaciones con una mayor amplitud. El actuador número 1 es el que presenta las mayores oscilaciones, a él le sigue el actuador número 2 y el 3. De hecho tanto el actuador 2 como el 3, presentan oscilaciones muy similares, con la única diferencia de que la dinámica del actuador 2 está más cercana a cero. El actuador 3 no converge a ese valor durante casi todo el experimento, sólo al final logra acercarse. En este experimento con alta ganancia, podemos apreciar cómo, al final de la prueba casi llegando al segundo 4, el error de los actuadores 2 y 3 logran acercarse y cruzan por cero. El actuador 3 al final del experimento mantiene sus oscilaciones.

Conclusiones

6.1 Conclusiones generales

Se realizó el modelo matemático del robot paralelo a partir de las ecuaciones de Lagrange. Basados en el modelo matemático obtenido, se pudo reproducir en simulación el comportamiento de la planta (robot paralelo).

Se logró comparar cómo se comporta el controlador PD nominal y el controlador PD+Twisting tanto experimentalmente como en simulaciones. El error en cada uno de los motores es analizado, comparándolos en un mismo experimento y también se revisaron las diferencias en el caso para una trayectoria circular o línea recta. Las comparaciones hechas con base en el error de posición en cada motor se traducen directamente en una buena precisión en el sistema.

Para poder realizar dicha comparación en el sistema real se implementó físicamente un algoritmo de control diferente al nominal. Para ello se estudió minuciosamente el funcionamiento de la unidad GPM2002 y la tarjeta controladora GT-400-SV a nivel de programación.

La elección de las ganancias del controlador Twisting se basaron en las incertidumbres no modeladas propias del sistema (fricción seca). Para esta elección de ganancias también se consideraron criterios que las correlacionan, apoyados en investigaciones previas sobre el tema.

6. CONCLUSIONES

Se comprobó que el controlador clásico nunca consigue hacer cero el error en los actuadores en tiempo finito. Y en algunos casos experimentales el controlador Twisting sí consigue llevar a cero el error en tiempo finito, incluso compensar las incertidumbres de forma efectiva.

Se comprobaron las oscilaciones generadas en el sistema mecánico ocasionadas por el algoritmo Twisting. La frecuencia y amplitud de las oscilaciones dependen de las ganancias del controlador.

Experimentalmente no se obtuvo una fiel reproducción de las simulaciones. Esto siempre ocurre, y fue comprobado al comparar no sólo los controladores, sino también las diferencias entre experimentos y simulaciones.

También es importante mencionar el trabajo de programación realizado. Tanto la programación en C++ para los experimentos así como en Matlab para las simulaciones. El mantenimiento adecuado al equipo junto con la resolución de fallas en los servoamplificadores y el sistema de sensores (encoders) fueron exitosos.

Finalmente, se deja la puesta en funcionamiento correcto de este robot paralelo. Todas las comparaciones hechas en el comportamiento de los dos diferentes controladores son el inicio de una extenso de trabajo futuro por realizar (pruebas de robustez, más algoritmos y diferentes técnicas de control). La posibilidad de manipular el algoritmo de control nominal ahora es posible de una manera más directa mediante programación de alto nivel (comprobado en las gráficas experimentales). Con todo esto se deja libre el camino para futuros experimentos y trabajos relacionados con el tema.

6.2 Comparación de resultados basados en RSME

Para poder realizar una comparación exacta del desempeño de cada uno de los controladores, hacemos uso de la raíz cuadrada media del error. RSME por sus siglas en

6.2 Comparación de resultados basados en RSME

inglés (root-square mean error) se define de la siguiente manera:

$$RSME = \sqrt{\frac{1}{N} \sum_{i=1}^N (z_i - z_{id})^2} \quad (6.1)$$

donde z_{id} es la posición angular en el i -ésimo punto de la trayectoria deseada y z_i es la posición angular en el i -ésimo punto de la trayectoria real. La comparación para las simulaciones se muestra en la tabla 6.1.

	<i>RSME</i> motor 1	<i>RSME</i> motor 2	<i>RSME</i> motor 3
<i>PD</i>	2.3534	0.8947	1.7092
<i>PD + Twisting</i>	0.0428	0.0595	0.1140

Tabla 6.1: Tabla comparativa de simulaciones trayectoria recta - RSME para cada uno de los motores del robot.

	<i>RSME</i> motor 1	<i>RSME</i> motor 2	<i>RSME</i> motor 3
<i>PD</i>	1.1018	0.9490	1.0193
<i>PD + Twisting</i>	0.2003	0.1908	0.2350

Tabla 6.2: Tabla comparativa de simulaciones trayectoria circular - RSME para cada uno de los motores del robot.

Cabe mencionar que aquí no es posible comparar simulaciones con experimentos, puesto que existen muchísimos factores que hacen totalmente diferentes ambos resultados, esos factores ya fueron ejemplificados en las conclusiones generales. En la tabla 6.2 se pueden observar las comparaciones entre los controladores para los resultados experimentales.

	<i>RSME</i> motor 1	<i>RSME</i> motor 2	<i>RSME</i> motor 3
<i>PD</i>	0.0111	0.0161	0.0032
<i>PD + Twisting</i>	0.0055	0.0089	0.0087

Tabla 6.3: Tabla comparativa de experimentos trayectoria recta - RSME para cada uno de los motores del robot.

6. CONCLUSIONES

Es evidente que el controlador discontinuo ha logrado mejorar el comportamiento del

	<i>RSME</i> motor 1	<i>RSME</i> motor 2	<i>RSME</i> motor 3
<i>PD</i>	0.0111	0.0160	0.0031
<i>PD + Twisting</i>	0.00085844	0.0035	0.0112

Tabla 6.4: Tabla comparativa de experimentos trayectoria circular - RSME para cada uno de los motores del robot.

sistema en 2 de los 3 motores en el caso experimental, para el motor 3 no podemos decir lo mismo, esto puede ser debido a un problema pendiente por resolver: la distribución de las fuerzas. Cada motor realiza diferente esfuerzo en determinada trayectoria, sin embargo, el análisis a este problema es un tema fuera del alcance de este trabajo y que es parte del trabajo a futuro. Se recomienda por tanto usar estos controladores discontinuos cuyas ventajas se han logrado comprobar con medidas numéricas exactas.

A.1 Tarjeta GT-400-SV

Para la implementación de los algoritmos de control, desde el entorno de Windows, se utilizó la tarjeta GT400-SV la cual tiene un campo de aplicaciones amplio.

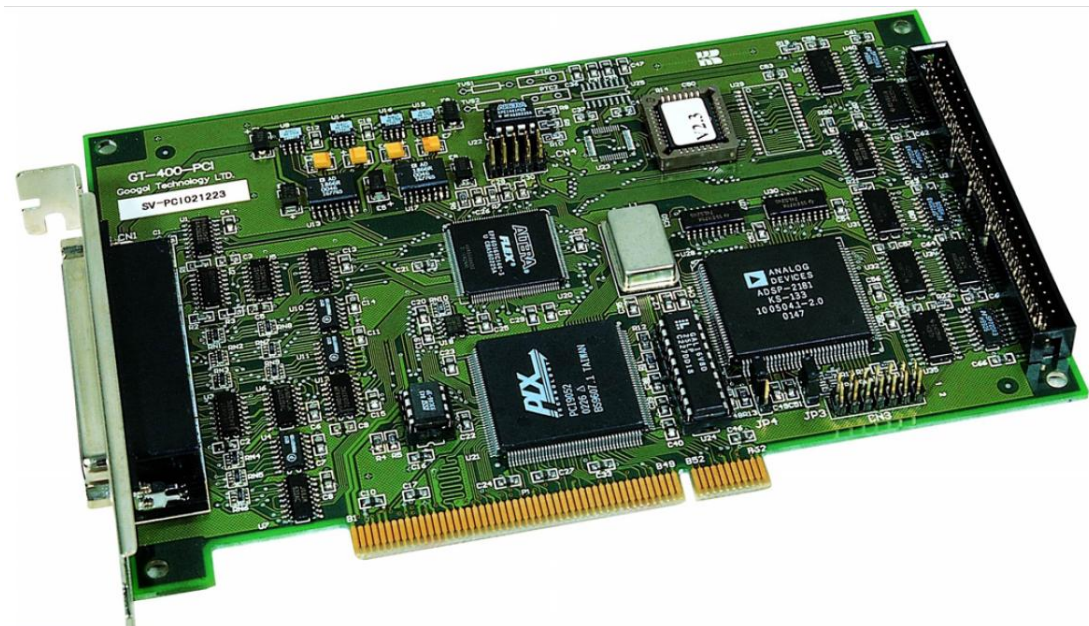


Figura A.1: Tarjeta GT-400-SV -

GT-400-SV de la serie de controladores de movimiento desarrollado por Googol

A.

Technology Ltd es un tipo de propósito general, de 4 ejes, controladores de movimiento basados en DSP. La serie tiene varios modelos de productos que se enumeran a continuación, ampliamente utilizado en aplicaciones que van desde el simple punto a punto de equipos de control de movimiento a los equipos de control de perfil muy complicado de movimiento, tales como máquinas de medición, máquinas de perforación PCB, SMT, máquina de grabado, tornos, centros de mecanizado, corte por chorro de agua, máquinas de corte por láser, robots y más [(16)].

Características principales:

- Puede controlar 4 servos / motores paso a paso.
- Periodo de muestreo programable. El periodo de interpolación mínimo de cuatro ejes es de 200us (GT-400-SG-S es de 400 us). El período mínimo de control de un solo eje de punto a punto el movimiento es 25US.
- Los modos de movimiento: movimiento de punto a punto, interpolación lineal, interpolación circular, control de velocidad, manual de entrada del generador de impulsos y ruedas de fricción electrónica.
- Curva programable trapecio, y S-curva de perfil de velocidad y los parámetros de actualización sobre la marcha.
- Todos los registros de los parámetros de cálculo y los parámetros de la trayectoria de planificación son de 32 bits.
- Se puede establecer el siguiente error límite, límite de la aceleración y el límite de la producción, para garantizar un control seguro y fiable.
- PID (Proporcional-Integral-Derivativo) un filtro digital con velocidad y la aceleración feedforward, y con el límite de la indemnización integral y la parcialidad y filtro de paso bajo.
- Movimiento coordinado de hasta 4 ejes, interpolación lineal, 2-4 ejes y 2 ejes de interpolación circular.
- Función de interpolación continua.

A.2 Sistema de Servo Amplificadores PY2

- Interrupción de eventos programables: Alarma de entrada externa, interrupción de evento y tiempo de interrupción.
- EEPROM para actualizar el firmware y los parámetros.
- Controladores y DLL para Windows98/2000/NT, C y C++ biblioteca de funciones.

En la figura A.2 se muestra el Filtro digital, fue modificado el parámetro "net difference compensation". Con el fin de poder aplicar el algoritmo Twisting de control:

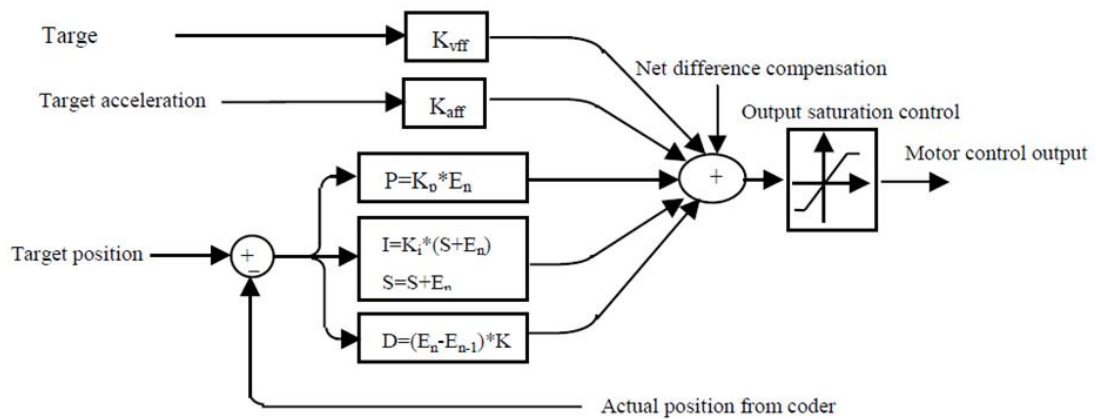


Figura A.2: Filtro digital para cada motor -

A.2 Sistema de Servo Amplificadores PY2

Para la calibración de los encoders, para darle la potencia necesaria a los actuadores, se utilizó el sistema de Servo Amplificadores PY2. También gracias al sistema de alarma con que cuentan estos amplificadores, se detectó con mayor rapidez algunos problemas técnicos durante este trabajo.

Las unidades PY2 pueden operar desde una entrada de 230 VAC (configuración monofásica o trifásica está disponible) y tiene funciones integradas tales como el frenado dinámico y regenerativo de procesamiento de la energía (resistencia regenerativa externa debe ser utilizado en algunas aplicaciones). Las unidades PY2 se fabrican en una instalación de la norma ISO-9001.

También se modificaron los parámetros del modo de operación 5-15 de los amplificadores para el correcto funcionamiento del motor número tres del robot. Al darle el

A.

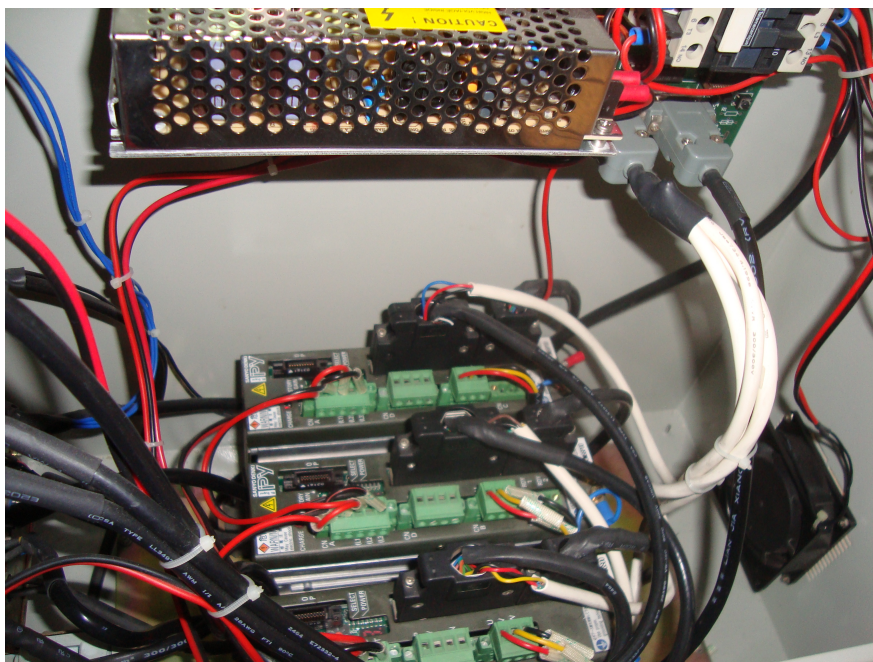


Figura A.3: Servo amplificadores PY2 - Se muestran los amplificadores y el cableado para comunicar con la tarjeta GT-400-SV, motores y encoders del robot

valor máximo a la constante integral del modo de operación, el movimiento del motor ya no presentaba sobrepasos. Para más detalles técnicos se puede consultar en [(17)].

A.3 Entorno de programación C++

Para la aplicación de los algoritmos de control a la tarjeta GT-400-SV, y por tanto al robot paralelo, se utilizó el entorno de programación Microsoft C++ [(18)].

El software con el que se aplican los algoritmos de control se actualizó, cambiando la estructura de algunas sentencias y declaración de variables. Para un manejo más cómodo y visualización de las clases y métodos, se paso del entorno clásico de Microsoft C++ 6.0 (figura A.4) al entorno más actual de Microsoft Visual Studio 2010 (figura A.5).

A.3 Entorno de programación C++

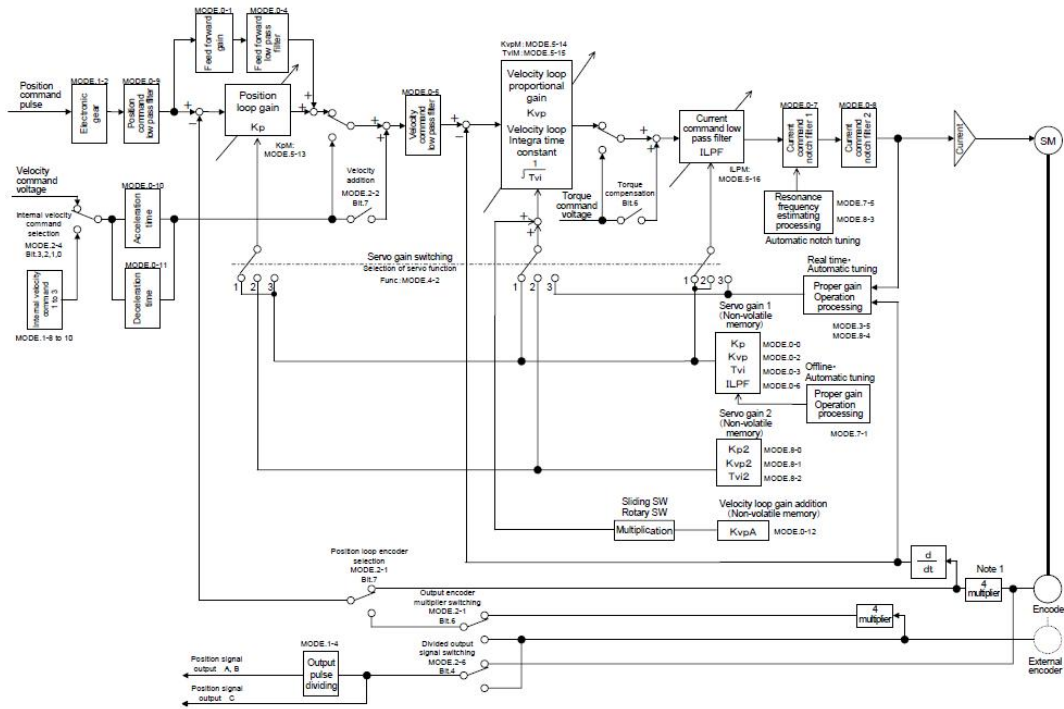


Figura A.4: Parámetros del amplificador PY2 -

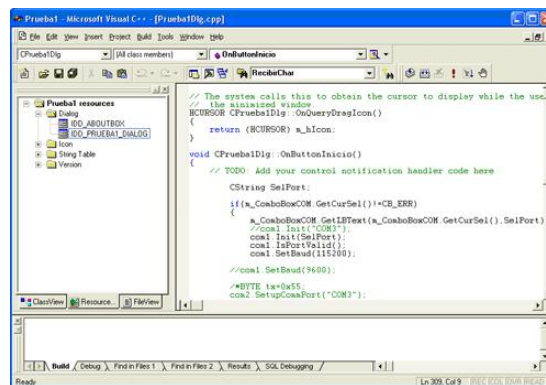


Figura A.5: Entorno C++ 6.0 - Al principio de este trabajo el software operaba con este entorno casi obsoleto

A.

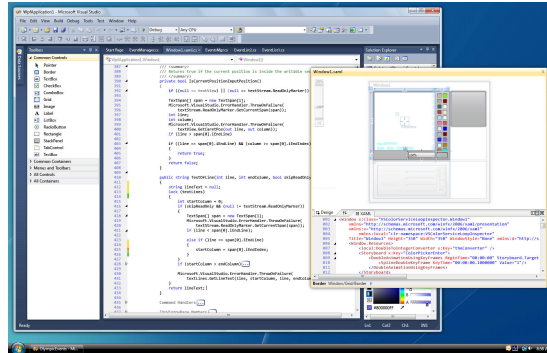


Figura A.6: Entorno Visual Studio 2010 - El nuevo entorno de programación es mas moderno y con mejores herramientas para la edición de código

A.4 MATLAB Simulink

Para la simulación de los procesos y modelos matemáticos se utilizó este software. Matlab (Matrix Laboratory) es un sistema basado en matrices para realizar cálculos matemáticos y de ingeniería. Entre las múltiples herramientas que presenta este programa se encuentra Simulink que es una librería de MATLAB que permite la simulación de procesos mediante diagramas de bloques, para más detalles se puede consultar en [(2)] y [(3)].

Simulink viene a ser una herramienta de simulación de modelos o sistemas, con cierto grado de abstracción de los fenómenos físicos involucrados en los mismos. Se hace hincapié en el análisis de sucesos, a través de la concepción de sistemas (cajas negras que realizan alguna operación)[(19)].

Se emplea arduamente en Ingeniería Electrónica en temas relacionados con el procesamiento digital de señales (DSP), involucrando temas específicos de ingeniería biomédica, telecomunicaciones, entre otros, también es muy utilizado en Ingeniería de Control y Robótica.


A.5 Calibración de los encoders absolutos

El sistema de servomotores de la unidad GPM2002 no puede operar a menos que los amplificadores PY2 no tengan ningun error en su funcionamiento. Una de las causas recurrentes en sus fallas es la necesaria recalibración de los encoders vía hardware. Para ello se utilizó comunicación RS232 además del software correspondiente. Cuando los

A.5 Calibración de los encoders absolutos

amplificadores emitían una señal de error como se muestra en la figura A.7, se tenía que "limpiar" esa señal utilizando el software que se muestra en la figura A.8. Este problema era ocasionado por que los encoders requieren su propia fuente de energía independiente de 3.6 [V] cada uno y cuando se desenergizaban era necesario limpiar esa señal de error.

Alarm Status

Segment LED Display	Alarm Code ALM8,4,2,1	Abbreviation	Alarm Name	Alarm Clear	Contents
	1000	DE1	Encoder disconnection (sensor error)	Not possible	Disconnection of sensor signal (A, B, C or PS signal) line was detected
		DE2	Serial disconnection (sensor error)	Not possible	Disconnection of sensor signal (PS signal) line was detected
		DE3	Encoder initial error (sensor error)	Not possible	Initial data of motor sensor can not be read in.
		DE4	Serial receiving stop (sensor error)	Not possible	No feedback of absolute position data from absolute sensor
		EXDE	External encoder disconnection	Possible	Disconnection of full close sensor signal line was detected.

Alarm code 0,1 indicates: when Func2/ bit7,6 ="0,0", "0"= output open and "1"= output short.

Figura A.7: Señal de error en el amplificador PY - Esta falla es común en estos equipos

A.

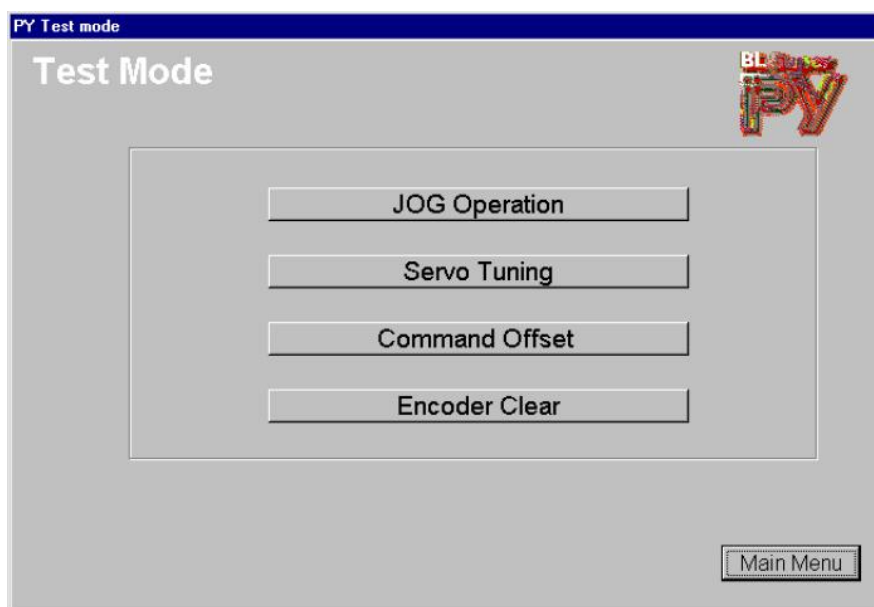


Figura A.8: Software PY para los amplificadores - El boton Encoder Clear sirve para corregir la alarma de error del amplificador PY en cada motor

Bibliografía

- [1] SHUANG CONG WEIWEI SHANG. **Nonlinear adaptive task space control for a 2-DOF redundantly actuated parallel manipulator.** *Springer*, 2010. 2, 44
- [2] HOLLY MOORE. **Matlab para ingenieros.** *Prentice Hall*, 2005. 9, 84
- [3] JONATHAN M. ROSENBERG BRIANT R. HUNT, RONALD L. LIPSMAN. **A guide to matlab, for begginers and experienced users.** *Prentice Hall*, 2006. 9, 84
- [4] S. SHANKAR SASTRY RICHARD M. MURRAY, ZEXIANG LI. **A Mathematical Introduction to Robotic Manipulation.** *CRC Press*, 1994. 11, 25
- [5] L. MEIROVICHM. **Methods of analytical dynamics.** *McGaw-Hill*, 1970. 13, 30
- [6] V.UTKIN. **First Stage of VSS: People and Events. In Variable Structure Systems Toward 21st Century.** *World Scientific Publishing Company*, 2001. 15
- [7] LEONID FRIDMAN RAFAEL IRIARTE, JAIME MORENO. **Sliding Modes after the First Decade of the 21st century.** *Springer*, 2011. 15
- [8] S. SPURGEON C.EDWARDS. **Sliding Mode Control, Theory an applications.** *Taylor and Francis, London*, 1998. 21
- [9] V.B ZAMANOV AND Z.M. SOTIROV. **A contribution to the serial and parallel manipulator duality.** *8th World Congress on the Theory of Machine and Mechanisms*, 1991. 23
- [10] K.J. WALDRON AND Z.M. HUNT, K.H. **Series-parallel dualities in actively coordinated mechanisms.** *The Int. J. of Robotics Research*, 1991. 23
- [11] SETH HUTCHINSON MARK W. SPONG AND M. VIDYASAGAR. **A Mathematical Introduction to Robotic Manipulation.** *CRC Press*, 1994. 25
- [12] L.F. PENIN. C. BLAGUER R. ARACIL BARRIENTOS, A. **Fundamentos de control y robótica.** *McGaw-Hill*, 1997. 26

BIBLIOGRAFÍA

- [13] WILLIAMS II R.L. **Inverse kinematics for planar parallel manipulators.** *Ohio University*, 1999. 26
- [14] KOROVIN SK EMELYANOV, SV AND LV LEVANTOVSKY. **Second order sliding modes in controlling uncertain systems.** *Soviet Journal of Compute and System Science*,, 1986. 40
- [15] JAIME MORENO RAUL SANTIESTEBAN, LEONID FRIDMAN. **Finite-time convergence analysis for Twisting controller via a strict Lyapunov function.** *International Workshop on Variable Structure Systems Mexico City*, 2010. 41
- [16] GOOGOL TECHNOLOGY. **Programming Manual for GT Motion Controller.** 2003. 80
- [17] SANYO DENKI CO. **PY2 Servo Amplifier Instruction Manual.** 2003. 82
- [18] BJARNE STROUSTRUP. **The C++ Programming Language: Special Edition.** 2001. 82
- [19] JUAM M. GOMEZ GLORIA MATA H., VICTOR M. SANCHEZ. **Análisis de sistemas y señales con computo avanzado.** *Facultad de ingeniería*, 2002. 84