



UNIVERSIDAD DE LA REPÚBLICA  
FACULTAD DE INGENIERÍA



# Construcción y Control del Péndulo de Furuta

MEMORIA DE PROYECTO PRESENTADA A LA FACULTAD DE  
INGENIERÍA DE LA UNIVERSIDAD DE LA REPÚBLICA POR

Alejandro Bellati, Fabián Cancela, Nicolás Pérez

EN CUMPLIMIENTO PARCIAL DE LOS REQUERIMIENTOS  
PARA LA OBTENCIÓN DEL TÍTULO DE  
INGENIERO ELECTRICISTA.

## TUTOR

Pablo Monzón ..... Universidad de la República  
Nicolas Perez ..... Universidad de la República

## TRIBUNAL

Sofía Valentina Machín..... Universidad de la República  
Rafael Canetti ..... Universidad de la República  
Alejandro Romanelli ..... Universidad de la República

Montevideo  
martes 26 enero, 2021

*Construcción y Control del Péndulo de Furuta*, Alejandro Bellati, Fabián Cancela,  
Nicolás Pérez.

Esta tesis fue preparada en L<sup>A</sup>T<sub>E</sub>X usando la clase iietesis (v1.1).  
Contiene un total de 187 páginas.  
Compilada el martes 26 enero, 2021.  
<http://iie.fing.edu.uy/>

Para toda tesis existe una antítesis igualmente válida.

IMMANUEL KANT

Esta página ha sido intencionalmente dejada en blanco.

# Resumen

El Péndulo de Furuta, también conocido como péndulo rotatorio invertido, consiste en un brazo impulsado que gira en el plano horizontal, unido a un péndulo que puede girar libremente en el plano vertical. Fue inventado en 1992 en el *Tokyo Institute of Technology* por Katsuhisa Furuta y sus colegas [1]. El péndulo es un ejemplo de un oscilador no lineal complejo de interés en la teoría de sistemas de control. Desde su creación cientos de documentos y tesis han utilizado el péndulo para probar y desarrollar técnicas de control.

Este proyecto desarrolló con éxito la construcción de un prototipo de laboratorio que puede ser utilizado tanto para enseñanza como para investigación. Se eligió un modelo físico para el péndulo y posteriormente se caracterizaron los parámetros del mismo.

El prototipo cuenta con un motor para impulsar el brazo, dos encoders y un sensor de corriente para medir sus variables de estado. Además se cuenta con un microcontrolador que aplica las estrategias de control implementadas y se comunica con el software de simulación, este permite cargar y evaluar distintos controladores de forma rápida.

Finalmente, se diseñó e implementó con éxito un controlador para mantener el péndulo en posición invertida y un controlador capaz de llevar el péndulo desde la posición de reposo hasta la posición invertida. Para esto se utilizó el modelo desarrollado e identificado junto con el software en el cual se integró el sistema, lo que termina de validarlo por completo.

Esta página ha sido intencionalmente dejada en blanco.

# Tabla de contenidos

<b>Resumen</b>	<b>III</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Motivación . . . . .	2
1.2. Descripción del proyecto . . . . .	2
1.3. Especificaciones . . . . .	3
1.4. Antecedentes . . . . .	4
1.5. Estructura del documento . . . . .	5
<b>2. Diseño y Construcción del Péndulo</b>	<b>7</b>
2.1. Investigación y elección del diseño . . . . .	8
2.2. Elección de Componentes . . . . .	11
2.2.1. Elección del Motor DC . . . . .	11
2.2.2. Elección del Driver . . . . .	11
2.2.3. Elección de Encoders . . . . .	12
2.2.4. Elección del Sensor de Corriente . . . . .	13
2.2.5. Elección del Microcontrolador . . . . .	14
2.2.6. Elección de Alimentación . . . . .	14
2.3. Construcción . . . . .	14
2.4. Conexionado . . . . .	16
2.5. Diseño de PCB . . . . .	18
2.5.1. Esquemático . . . . .	18
2.5.2. PCB Layout . . . . .	19
2.5.3. Resultado final del PCB . . . . .	21
<b>3. Acondicionamiento de sensores y actuadores</b>	<b>23</b>
3.1. Sensor de corriente . . . . .	23
3.2. Driver y Motor DC . . . . .	24
3.2.1. Salida . . . . .	25
3.2.2. Causa . . . . .	26
3.2.3. Compensación . . . . .	28
3.3. Encoder . . . . .	32
3.3.1. Midiendo posición . . . . .	32
3.3.2. Testeando medidas de velocidad . . . . .	33
3.3.3. Algoritmo de tiempo fijo . . . . .	33

## Tabla de contenidos

3.3.4. Algoritmo de espacio fijo . . . . .	33
3.3.5. Algoritmo utilizado: Controlador PI . . . . .	34
<b>4. Modelado del Sistema/Péndulo</b>	<b>37</b>
4.1. Péndulo . . . . .	38
4.1.1. Introducción . . . . .	38
4.1.2. Hipótesis del modelo . . . . .	38
4.1.3. Medidas del péndulo . . . . .	39
4.1.4. Estimaciones . . . . .	41
4.1.5. Construcción del modelo . . . . .	42
4.1.6. Momentos de inercia y centro de masa . . . . .	45
4.1.7. Comparando con la literatura . . . . .	46
4.2. Motor . . . . .	47
4.2.1. Limitación del modelo . . . . .	47
4.3. Resumen del modelado . . . . .	48
4.3.1. Ecuaciones en variables de estado . . . . .	48
4.3.2. Linealización Jacobiana . . . . .	49
<b>5. Identificación del Sistema</b>	<b>51</b>
5.1. Péndulo de Furuta . . . . .	51
5.1.1. Péndulo libre, soporte en U fijo . . . . .	51
5.1.2. Soporte en U . . . . .	58
5.2. Motor . . . . .	61
5.2.1. Parámetros iniciales . . . . .	62
5.2.2. Parámetros ajustados . . . . .	65
5.3. Resultados . . . . .	69
5.3.1. Sobre la zona muerta . . . . .	72
5.4. Resumen: modelo y parámetros . . . . .	74
<b>6. Integración en Software de Simulación</b>	<b>77</b>
6.1. Bloques básicos . . . . .	78
6.1.1. Encoder . . . . .	78
6.1.2. Sensor de corriente y calibrador . . . . .	80
6.1.3. Actuador . . . . .	82
6.2. Otros bloques . . . . .	83
6.2.1. Seguidor . . . . .	83
6.2.2. Estabilizador de bajada . . . . .	84
6.2.3. Máquina de estados . . . . .	84
6.2.4. Bloque controlador . . . . .	84
6.3. Ejemplo total . . . . .	85
6.4. Restricciones del prototipo, condiciones de prueba y consideraciones de uso . . . . .	86



<b>7. Controladores</b>	<b>89</b>
7.1. Control estabilizante . . . . .	89
7.1.1. Marco teórico . . . . .	90
7.1.2. Características de la planta . . . . .	93
7.1.3. Metodología . . . . .	94
7.1.4. Ley de control . . . . .	95
7.1.5. Resultados . . . . .	97
7.1.6. Análisis . . . . .	104
7.2. <i>Swing Up</i> . . . . .	111
7.2.1. Metodología . . . . .	111
7.2.2. Energy control y el péndulo simple . . . . .	112
7.2.3. <i>Swing up</i> del péndulo de Furuta . . . . .	115
7.2.4. Simulaciones . . . . .	117
7.2.5. Pruebas y elección del controlador . . . . .	120
7.2.6. Resultados y análisis . . . . .	121
7.3. <i>Swing up</i> , control estabilizante y control supervisor . . . . .	126
7.3.1. Control estabilizante de bajada y desenredador . . . . .	126
7.3.2. Condiciones de cambio entre los controladores . . . . .	127
7.3.3. Resultados . . . . .	129
<b>8. Conclusiones</b>	<b>133</b>
8.1. Trabajos Futuros . . . . .	135
<b>Apéndices</b>	<b>138</b>
<b>A. Estimación de Máxima Caída de Voltajes en Pistas del PCB</b>	<b>139</b>
<b>B. Cálculo de los momentos de inercia y del centro de masa</b>	<b>141</b>
<b>C. Estimación de Incertidumbres Mediante Simulación</b>	<b>145</b>
C.1. Distribución de $J$ . . . . .	145
C.2. Distribución de $\omega$ en las medidas para el cálculo del $K$ del motor . . . . .	146
<b>D. Códigos</b>	<b>149</b>
<b>E. Implementación de los filtros derivativos</b>	<b>151</b>
<b>F. Un descubrimiento</b>	<b>155</b>
<b>Referencias</b>	<b>159</b>
<b>Glosario</b>	<b>163</b>
<b>Índice de tablas</b>	<b>169</b>
<b>Índice de figuras</b>	<b>170</b>

Esta página ha sido intencionalmente dejada en blanco.

# Capítulo 1

## Introducción

Los sistemas mecánicos subactuados son sistemas donde se tienen menos entradas de control que grados de libertad. Esta falta de entradas trae consigo una gran dificultad a la hora de controlar dichos sistemas. El control de sistemas subactuados es actualmente un área activa de investigación debido a sus diversas aplicaciones como en robótica, aeronáutica, vehículos marinos, etc.

El péndulo de Furuta, también conocido como péndulo rotatorio invertido, es un sistema subactuado y extremadamente no lineal debido a la fuerza gravitatoria junto al Efecto Coriolis y a la fuerza centrípeta. El mismo consiste en un sistema con dos grados de libertad y dos articulaciones y está comprendido esencialmente por tres elementos; un motor, un *brazo* conectado al eje del motor que gira en el plano horizontal y un *péndulo* unido a ese brazo que puede girar libremente en el plano perpendicular al brazo. En la figura 1.1 se muestra una ilustración del mismo. Fue inventado en 1992 en el Tokyo *Institute of Technology* por Katsuhisa Furuta y sus colegas [1]. Hoy en día es uno de los sistemas subactuados más conocidos el cual es un excelente sistema de pruebas para muchos problemas de control, siendo de interés para la teoría, atrayendo a los investigadores durante años.

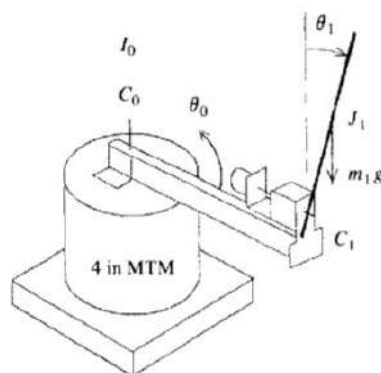


Figura 1.1: Ilustración de un péndulo de Furuta. Extraída de [1]

Desde su creación cientos de documentos y tesis han utilizado el péndulo para

## Capítulo 1. Introducción

probar y desarrollar técnicas de control. Actualmente, *Google Scholar* despliega mil quinientos ochenta resultados de artículos y documentos relacionados al Péndulo de Furuta. En estos se estudia su modelo, construcción y diferentes técnicas de control entre otras cosas.

### 1.1. Motivación

En el comienzo del capítulo se mencionaron cuáles son los beneficios que surgen al estudiar los sistemas subactuados y sus posibles aplicaciones. Se enfatizó en las ventajas que tiene el Péndulo de Furuta para el estudio de este tipo de sistemas. Es natural para nosotros querer trabajar con un sistema el cual se utiliza actualmente para avanzar la frontera del conocimiento, en el centro de la investigación de sistemas subactuados. Como aspirantes a ingenieros es muy gratificante ser capaces de contribuir a la institución que nos formó con un nuevo sistema, complejo pero lo suficientemente práctico y didáctico como para permitir abordar estos aspectos de la rama del control en sus cursos de grado y posgrado dado que captan mucho la atención en varias partes del mundo.

Lograr crear un nuevo instrumento de laboratorio para el instituto ya es de por sí una gran motivación para el grupo y es el principal propósito del proyecto, pero cabe mencionar también otros aspectos no menores que forman parte de la decisión de abordar este desafío. Una de estos es el proceso de ingeniería que conlleva el desarrollo de un prototipo en todas sus etapas. Es aquí donde uno logra visualizar las dificultades que se deben afrontar en cada una de las mismas, en este caso fue especialmente enriquecedor enfrentarse a un sistema no lineal complejo y entender la repercusión que tiene cada una de las decisiones tomadas en el transcurso del proyecto. También es de gran interés el tener que pasar por el proceso del control: modelar, identificar, diseñar e implementar un control. Y por último pero no menos importante, es el simple de hecho de realizar el proyecto de grado de la carrera de ingeniería eléctrica, cerrando un ciclo y una etapa de nuestro proceso académico.

### 1.2. Descripción del proyecto

Este proyecto consiste en el diseño y construcción de un sistema que sea sencillamente manipulable por el usuario pero a su vez de dinámica compleja. Específicamente se abordará la construcción de un Péndulo de Furuta. Este sistema debe permitir al usuario desarrollar y evaluar distintas técnicas de control para su posterior estudio, debe contar consecuentemente con los componentes necesarios para accionar y observar el sistema en un instante dado.

Se desglosa en el presente documento el desarrollo de un prototipo de principio a fin. Se pueden diferenciar distintas etapas para la elaboración del mismo. Estas conforman su diseño, construcción, modelado, identificación de parámetros y finalmente la implementación de al menos un controlador. A lo que se refiere con

### 1.3. Especificaciones

el diseño del sistema, se debe tomar en cuenta sus posibles restricciones (dimensionales, mecánicas, eléctricas, etc), la elección de materiales y los elementos que lo componen. En cuanto del modelo, se deben presentar las hipótesis tomadas y como llegar a un modelo capaz de predecir el comportamiento del sistema dentro de las condiciones de uso.

Una vez obtenido el modelo, es inmediato notar que se deben realizar ensayos, presentar resultados y métodos de ajuste para la obtención de los parámetros de nuestro modelo que mejor predican el comportamiento de nuestro Péndulo de Furuta.

Además se incluye en el proyecto la integración del sistema con un software capaz de simular y actuar sobre el sistema. Con esto se busca establecer una rápida transición entre simulación e implementación en un entorno de desarrollo centralizado. Esto permite ejecutar en el sistema real los controladores ya testeados y transforma el proceso de diseño en una tarea rápida y amigable. Se obtienen de esta manera simulaciones y relevamientos en un mismo software, facilitando al usuario en la evaluación de los controladores implementados.

Finalmente se incluye en los objetivos del proyecto implementar al menos una estrategia de control estabilizante para el Péndulo de Furuta, esto es, lograr implementar un lazo de control que sea capaz de mantenerlo en una posición de equilibrio naturalmente inestable.

Como adicional se implementó además un control de *swing up*, es decir, uno que logra llevar el péndulo desde su posición de reposo a su posición invertida.

### 1.3. Especificaciones

Se deberá construir un prototipo de laboratorio de un Péndulo de Furuta, es decir, un péndulo invertido de dos brazos montado sobre una plataforma giratoria. El péndulo tiene que contar con un motor para poder actuar sobre la plataforma giratoria. Debe disponer de distintos sensores para medir el estado actual del mismo. Estos son dos codificadores ópticos, uno para medir la posición del péndulo y otro para medir la posición del motor y un sensor de corriente para medir la corriente por el motor. También deberá contar con un microprocesador para manejar la inteligencia del dispositivo. Finalmente se deberá implementar una estrategia de control sobre el péndulo.

El péndulo será utilizado para enseñanza, investigación y en muestras de ingeniería. Esto define algunos requerimientos para el mismo. Deberá ser de un tamaño adecuado para ser trasladado de forma de poder ser utilizado en muestras, con la suficiente consistencia física como para que esto no ocasione problemas. También deberá contar con una interfaz de comunicación con un PC de forma de poder

## Capítulo 1. Introducción

programar la estrategia de control deseada o adquirir datos para comparar con simulaciones. Además se espera que el prototipo pueda integrarse con un software de simulación desde el cuál se pueda manejar el sistema.

### 1.4. Antecedentes

Los antecedentes de trabajos con el péndulo de Furuta son muchos y en diferentes aspectos. Hay documentos centrados en los aspectos constructivos, artículos centrados en el modelado y las diferentes formas para caracterizar la dinámica del sistema, y hay otros atacando los distintos problemas de control que el sistema ofrece. El punto de partida en los antecedentes es el artículo de Furuta en 1992 [1], donde se presenta el sistema por primera vez. Desde entonces el sistema ha sido estudiado por varios autores.

De los problemas de control que trae el péndulo de Furuta consigo, destacan principalmente dos. El primero es lograr levantar el péndulo desde la posición de reposo hasta la posición invertida, llamado *swing up*. El otro es el problema de estabilizar el péndulo entorno a su posición invertida. Asociados con éstos está el desafío de levantar y estabilizar el péndulo, donde típicamente se observan estrategias que combinan las soluciones de cada problema. Además de estos hay otros problemas interesantes como lograr estabilizar oscilaciones o controlar el péndulo utilizando análisis de bifurcaciones.

El problema del *swing up* fue resuelto por primera vez por Amstrom y Furuta con un modelo reducido utilizando un método llamado *energy control*. Sobre los controladores de *swing up* se han desarrollado varias estrategias, siendo lo más común utilizar *energy control*, pero también aparecen otras técnicas como plantear el diseño como un problema de optimización, *bang-bang state feedback*, *vibrational input*, entre otros.

En cuanto al problema de estabilizar el péndulo en su posición invertida, la solución simple es aproximar el sistema por un modelo lineal, con linealización Jacobiana, y aplicar un controlador lineal. Típicamente se utiliza como control lineal un regulador cuadrático lineal, conocido como *LQR*. A pesar de esta simple solución hay varias razones para el estudio de otra gran variedad de soluciones que entran en el terreno del control no lineal. En esta área aparecen técnicas como *energy control*, *Forwarding*, *Backstepping*, linealización entrada-salida, *Singular Perturbations*, entre otros.

Para el modelado en general todos siguen el camino de derivar las ecuaciones del movimiento utilizando las ecuaciones de Euler-Lagrange. En la determinación de un *lagrangiano*  $\mathcal{L}$  para el sistema todos utilizan  $K - V$  pero no todos derivan de la misma forma la energía cinética  $K$ . Algunos van por el cálculo directo [2] mientras que la mayoría descomponen el sistema en dos rígidos y utilizan la iner-

## 1.5. Estructura del documento

cia de dichos rígidos. También se encuentra autores que optan por un formalismo *hamiltoniano* a la hora de derivar las ecuaciones, por ejemplo [2].

En cuanto a los aspectos constructivos no son muchos los proyectos que se centran en esto, en [3] y [4] se ven dos ejemplos de prototipos cuya finalidad es académica. Además en [5] se puede ver un prototipo comercial. Si bien en todos los casos se quiere construir un prototipo del péndulo de Futura los resultados son distintos estructuralmente. Sin embargo, en general son equivalentes desde el punto de vista de las ecuaciones que modelan el movimiento.

## 1.5. Estructura del documento

En el próximo capítulo se comienza contando el proceso de diseño y presentando el diagrama esquemático del péndulo de Furuta. Luego se detalla cómo se realizó la elección de los componentes, la construcción del prototipo y el conexionado.

El capítulo 3 se centra en los componentes. Se detallan los principales problemas, circuitos de acondicionamiento y los test realizados.

En el capítulo 4 se desarrolla el modelo del péndulo y del motor, este desarrollo se centra en el prototipo construido en este proyecto. Se presentan las ideas, se detallan las hipótesis, se deriva el modelo y se estudian las limitaciones del mismo. Sobre el final del capítulo se resume el modelo, se presentan distintas características del mismo y finalmente se compara contra los distintos modelos desarrollados en la literatura.

En el capítulo siguiente se realiza la identificación de parámetros del modelo. Se presentan los ensayos realizados y se muestran los resultados de los mismos obteniendo lo que se llaman los parámetros iniciales. Luego de un ajuste se llegan a los parámetros del modelo. En el mismo capítulo se explica cómo y por qué se hace este ajuste. Finalmente se muestran los resultados del modelo identificado, viendo su capacidad para predecir la respuesta del sistema.

El capítulo que sigue es el 6, este se centra en la integración con el software de simulación. Se pretende mostrar la implementación de distintos bloques necesarios para el manejo del sistema desde el PC. Se muestra cómo se fue construyendo a partir de bloques sencillos los bloques más complejos y cómo interactúan entre ellos para lograr leer el estado del sistema y actuar en él a partir de dicha lectura.

En el capítulo 7 se exponen los controladores implementados. Los dos más importantes son el controlador *swing up* y el control estabilizante de los cuales se presenta la metodología aplicada para diseñar cada uno. Además se muestran los resultados obtenidos así como un análisis de los mismos. Sobre el final del capítulo se combinan estos dos controladores junto con un control supervisor para lograr

## Capítulo 1. Introducción

que el péndulo se levante por sí solo desde la posición de reposo hacia la posición invertida y que se estabilice en esta última.

Para finalizar en el capítulo 8 se presentan las conclusiones, las principales reflexiones del proyecto y el trabajo a futuro.



## Capítulo 2

# Diseño y Construcción del Péndulo

En este capítulo se presenta el proceso de diseño y construcción del péndulo. Este proceso comenzó con la búsqueda de otros proyectos o artículos donde se construyera un péndulo de Furuta, haciendo énfasis en el proceso de construcción, tamaño, peso, materiales, conexionado, etc. Las referencias con este enfoque fueron escasas, teniendo principalmente [3], [4]. Esta parte fue muy ilustrativa, explorando el diseño, las estructuras, utilizadas en distintos péndulos. Este camino también nos permitió ganar conocimientos sobre valores razonables para los parámetros del péndulo, sirviendo como guía a la hora de buscar componentes en plaza como motores o sensores. En esta búsqueda se recopiló información sobre materiales, tamaño, dimensiones, estructuras posibles, peso de las distintas partes, tipo de sensores utilizados, tipo de motores y sus características. También se buscó péndulos comerciales pero los mismos presentaban menor información.

En base a la investigación se decidió un primer diseño el cual fue acordado entre el grupo y los tutores. La construcción fue realizada por un tornero, se agradece a Gustavo Bellati por su colaboración honoraria. Se intentó utilizar materiales con los que ya disponíamos por lo que el costo del péndulo fue muy bajo. El péndulo construido puede desarmarse en su totalidad. Esto permite fácil traslado, una característica deseada en las especificaciones. Además el prototipo construido permite cambiar fácilmente el motor, el encoder y el péndulo.

El motor elegido es de 12 V, tiene un encoder incremental integrado para medir la posición del brazo, con  $11 \times 78$  PPR. En cuanto a los sensores, se utilizó un encoder incremental con 600 PPR para medir la posición del péndulo. El sensor de corriente utilizado es el *ACS712* que se basa en el efecto hall que impone una resistencia muy pequeña. Tiene una capacidad de 5 A pero se cambiará este fondo de escala, mediante un circuito acondicionador, por uno más adecuado a la corriente que circula por el motor esto.

## 2.1. Investigación y elección del diseño

Previo a la búsqueda, no se conocía ningún prototipo ni se tenía experiencia en la construcción de uno. Se buscaron proyectos y artículos donde se expusiera la construcción un péndulo de Furuta de forma de tener indicios sólidos sobre el desarrollo del mismo. Aquí se presentan solo las principales referencias.

Se comenzó explorando los prototipos comerciales de *Quanser*. Particularmente en el prototipo presentado en [5] y sus variantes. Es un prototipo muy liviano, una de sus versiones tiene un brazo es de  $42,5\text{ cm}$  y el péndulo de  $12\text{ cm}$ . Es claro el acondicionamiento estético para ser un prototipo comercial.

En [3] se presenta un proyecto de construcción de un péndulo de Furuta de bajo costo, de base circular de 3 pilares. El brazo mide  $9,55\text{ cm}$  y pesa  $186\text{ g}$  mientras que el péndulo mide  $29,8\text{ cm}$  y pesa  $65\text{ g}$ . Se utilizó un encoder de 300 PPR y un motor DC de  $12\text{ V}$  con encoder incorporado. El prototipo se construyó en aluminio, el brazo es hueco con una guía en el medio que tiene rodamiento libre gracias a un rodamiento que lo conecta con el péndulo. La guía permite que el encoder copie los movimientos del péndulo de manera de poder obtener su posición.



Figura 2.1: A la izquierda se observa el prototipo construido en [4] y a la derecha el de [3]

Otra referencia de interés es el artículo [4], donde se muestra un prototipo similar al anterior. Construido esencialmente en aluminio con una base cuadrada de 4 pilares, un brazo de  $14\text{ cm}$  y un péndulo de  $14,75\text{ cm}$  con un peso de  $38\text{ g}$ . Se diferencia en cuanto a la ubicación del encoder encargado de medir la posición del péndulo que, para este caso, se ubicó sobre el brazo en el extremo donde está el péndulo.

En la figura 2.1 se muestra una imagen de estos prototipos que fueron nuestra

## 2.1. Investigación y elección del diseño

principal inspiración. En base a las investigaciones y a los acuerdos realizados con los tutores se eligió un prototipo. En la figura 2.2 se observa la proyección explosiva del diseño adoptado. En la figura 2.3 se muestra una vista frontal y lateral del péndulo, junto con las medidas de importancia.

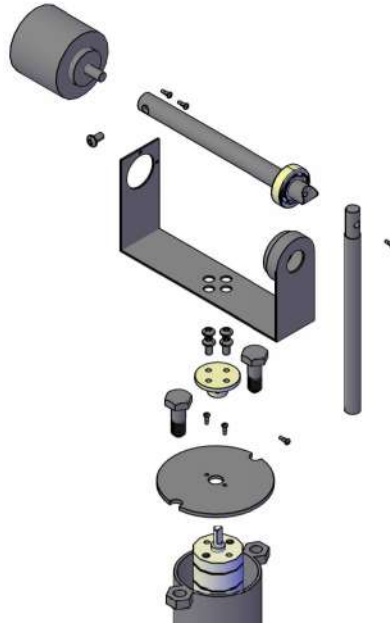


Figura 2.2: Diseño del péndulo de Furuta, proyección explosiva.

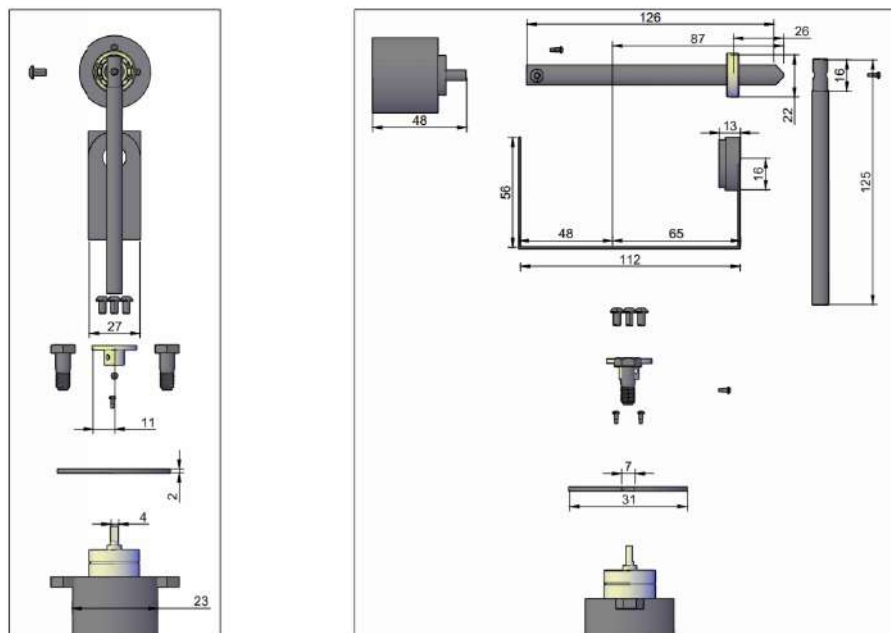


Figura 2.3: Diseño del péndulo de Furuta, vista frontal y lateral. Las medidas son en *mm*

## Capítulo 2. Diseño y Construcción del Péndulo

El mismo se construyó principalmente en metal, con una base circular y un solo pilar para la base. En cuanto a la ubicación del encoder se optó por imitar a [3], ya que lleva a tener un brazo más simétrico y también porque es bastante más práctico manejar los cables del sensor. En la figura 2.4 se muestra una imagen del prototipo construido.



Figura 2.4: Imagen del prototipo construido

Para el sistema diseñado se desarrolló un modelo basado en el siguiente sistema de ecuaciones diferenciales, a las que también llamamos ecuaciones del movimiento.

$$\begin{aligned}\ddot{\varphi} (\gamma + \alpha \sin^2 \theta) + 2\alpha\dot{\varphi}\dot{\theta} \cos \theta \sin \theta + \beta\ddot{\theta} \cos \theta - \beta\dot{\theta}^2 \sin \theta &= \tau + \tau_{fu} \\ \alpha\ddot{\theta} + \beta\dot{\varphi} \cos \theta - \alpha\dot{\varphi}^2 \cos \theta \sin \theta - D \sin \theta &= \tau_{fp}\end{aligned}$$

donde  $\varphi$  es el ángulo del brazo,  $\theta$  es el ángulo del péndulo colgante,  $\tau$  es el torque ejercido por el motor y  $\tau_{fu}$ ,  $\tau_{fp}$  son fricciones. Los parámetros  $\alpha$ ,  $\beta$ ,  $\gamma$  y  $D$  dependen de valores como longitudes, masas, centros de masa e inercias.

El modelo coincide con el presentado en [6] luego de hacer el cambio de variable  $q_1 = \pi - \theta$  y  $q_2 = \varphi$ . También se puede ver que coincide con [7] y con [2]. El trabajo [3] presenta un péndulo bastante parecido al nuestro pero los autores han decidido no mostrar las cuentas en su completitud por lo cual se dificulta el seguimiento. En el capítulo 4 se presentará el desarrollo de las ecuaciones y se profundizará sobre el modelo.

## 2.2. Elección de Componentes

Antes de adentrarnos en la construcción del prototipo, se detallará el proceso de elección para cada uno de los componentes. Esto es, los requisitos que deben cumplir y las características de los mismos. Aquí se incluye motor, driver, encoders, sensor de corriente y microcontrolador. Las pruebas realizadas para validarlos se explican en el capítulo 3.

Se estableció en conjunto con los tutores priorizar los componentes disponibles en plaza para evitar largos tiempos de espera. Permitiendo un desarrollo más fluido del proyecto sin tiempos muertos.

### 2.2.1. Elección del Motor DC

Para la elección del motor DC se tomaron en cuenta las siguientes características: que sea capaz de manejar la carga esperada en condiciones nominales, disponible en plaza, y con encoder integrado para facilitar el montaje del mismo.

Respecto al primer requisito, se fijaron como punto de referencia los utilizados en [4] y en [3], dado que nuestro prototipo es de dimensiones similares. En el documento [4] se presenta un motor DC de voltaje nominal 12 V y torque máximo de 1,34 Nm y el presentado en [3] es de 12 V y torque máximo de 0,83 Nm.

Teniendo en cuenta estos datos se encuentra un único modelo de motor disponible en plaza con encoder integrado (11 × 78 PPR) que satisface nuestras necesidades. El modelo utilizado es el *JGA25-370*. Se presenta en la tabla 2.1 las características del mismo, estos datos provienen de los distribuidores del producto, no se dispone de una cartilla oficial del mismo. Un detalle a mencionar es que el motor cuenta con engranajes con una relación de 78:1. Esto no es óptimo para la construcción de un Péndulo de Furuta debido a los efectos mecánicos que pueden generar, como aumentar el *backlash* o *juego*, o aportar a la fricción. Aun así se opta por este modelo.

Voltaje Nominal	Carga Nominal			
	Velocidad	Corriente	Torque	Potencia
12 V	62rpm	330 mA	0,32 Nm	2 W
	Sin Carga		Carga Máxima	
	82 rpm	100 mA	1,12 Nm	1500 mA

Tabla 2.1: Datos provistos por el vendedor del motor DC utilizado.

### 2.2.2. Elección del Driver

Dado que los pines de Arduino no pueden manejar las corrientes que se deben entregar al motor DC, se debe utilizar un driver que funcione como actuador. El

## Capítulo 2. Diseño y Construcción del Péndulo

mismo debe ser capaz de manejar las corrientes exigidas por el motor. Estas fueron presentadas en la tabla 2.1 con un consumo máximo de 1,5 A. Otro requisito es que el driver sea capaz de entregar corriente en ambas direcciones para lograr ambos sentidos de giro en el motor.

El driver bidireccional *L298N*, basado en el chip *L298 Dual H-Bridge Motor Driver* fue obtenido en plaza. El mismo cuenta con un rango de operación de alimentación de 5 V – 35 V y capaz de soportar corrientes máximas de 2 A. En la figura 2.5 se muestran el driver y el motor utilizados.

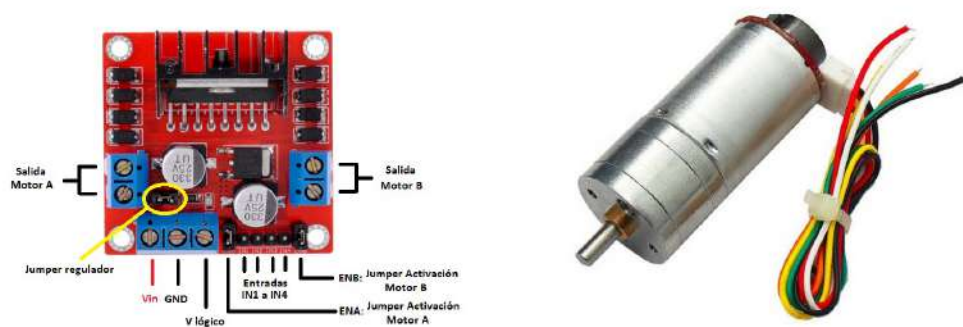


Figura 2.5: Ilustración del driver L298N y motor utilizado.

Extraídas de <https://www.prometec.net/l298n/> y de <http://electronica.uy/robotica/motores>

### 2.2.3. Elección de Encoders

Para poder medir posición y velocidad angular de ambos ejes se debe contar con dos sensores con los que se pueda detectar posición angular. Para esto se utilizaron codificadores ópticos o usualmente llamados encoders (por su nombre en inglés). Estos pueden ser incrementales, es decir, que detectan únicamente cambios de posición o absolutos, que indican la posición actual. Se optó por incrementales por haber más opciones en plaza y por la gran diferencia de precio entre ambos.

Como fue mencionado en la sección 2.2.1, el motor DC ya cuenta con un encoder integrado diseñado para funcionar en conjunto con nuestro modelo de motor. Se verificó, sin embargo, que contara con una buena apreciación. El encoder del motor tiene una resolución de 11 pulsos por revolución (PPR). Al estar conectado antes de los engranajes (de relación 78:1) la resolución se amplifica en la misma proporción. Además se configura la lectura del encoder de modo que la resolución se multiplica por cuatro. Este último efecto se explicará en la sección 3.3. Consecuentemente, la resolución es equivalente a la de un encoder con  $11 \times 4 \times 78$  PPR que se traduce a  $0,10^\circ$  de apreciación, lo cual se considera más que suficiente.

## 2.2. Elección de Componentes



Figura 2.6: Foto del encoder utilizado para medir la posición del péndulo.

Con respecto al encoder del péndulo, el modelo utilizado es el *Rotary Encoder LPD3806-600BM-G5-24C* que cuenta con 600 PPR. Al igual que el encoder anterior, se puede configurar para tener su resolución multiplicada por cuatro. Finalmente se tiene una resolución equivalente a la de un encoder de 2400 PPR que se traduce a  $0,15^\circ$  de apreciación. Esta se considera una resolución adecuada para poder medir la posición angular del péndulo. En la figura 2.6 se muestra una imagen del encoder utilizado para medir la posición del péndulo.

### 2.2.4. Elección del Sensor de Corriente

Para poder cuantificar el consumo del motor en tiempo real, se debe contar con sensor de corriente. El componente incorporado es un PCB prefabricado basado en el chip *ACS712* de *Allegro Microsystems*. El mismo tiene un rango de operación de hasta  $5 A$  y una sensibilidad de  $185 mV/A$ . Su ancho de banda es de  $80 KHz$ , el cuál es mucho mayor al necesario para medir el sistema. En la figura 2.7 se muestra una foto del componente utilizado.

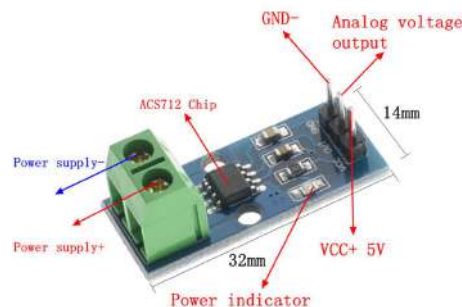


Figura 2.7: Foto del componente utilizado con el sensor ACS712.

## Capítulo 2. Diseño y Construcción del Péndulo

### 2.2.5. Elección del Microcontrolador

Se optó por utilizar un Arduino. Esta elección se basó en la comodidad que ofrece para programarlo dadas las librerías que existen, además de la facilidad para conseguir diferentes modelos en plaza. Otra razón es que se constató su uso en proyectos similares. También es ventajoso a la hora de implementar una comunicación con *Simulink* ya que existe una librería tanto para *Simulink* como para *Matlab*.

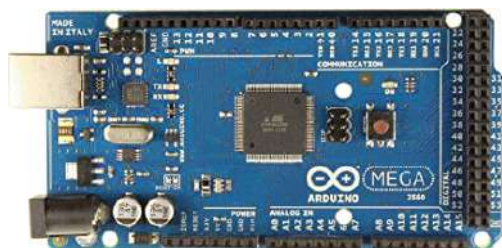


Figura 2.8: Foto de Arduino MEGA 2560.

En cuanto al modelo se optó por el Mega 2560 por varios motivos. Los más relevantes son la capacidad de memoria, la velocidad de procesamiento y la cantidad de pines digitales disponibles para configurar para interrumpir ya que en el esquema elegido se necesitaban al menos 4 de estos pines. En la figura 2.8 se presenta una imagen del componente utilizado.

### 2.2.6. Elección de Alimentación

Para la alimentación del sistema se necesitan dos entradas, una de 5 V y una de 12 V. El Arduino se puede alimentar mediante la entrada USB o su conexión a 5 V. Luego desde la salida de 5 V del Arduino se alimenta a los encoders, el sensor y su circuito acondicionador. Un convertor AC/DC de 12 V es utilizado para alimentar el driver. El modelo del convertor soporta una corriente nominal de 2 A la cuál es acorde con la máxima soportada por el driver.

## 2.3. Construcción

El prototipo se construyó con materiales con los que ya se disponía, principalmente metal, bajando el costo del mismo. Éste fue construido por un tornero, quien no solo brindó su asesoramiento sino también los materiales y las herramientas necesarias.

Se utilizó un solo pilar circular al cual se le soldó un apoyo de forma de tener buena estabilidad. El pilar tiene una longitud de 22,5 cm, es hueco de forma de poder ubicar el motor en el medio. Además el pilar cuenta con una abertura para poder sacar los cables del motor y del encoder.



### 2.3. Construcción

En el extremo del pilar se colocó una base que funciona de soporte del motor. Este se atornilla a la base para asegurarlo al pilar. El eje del motor atraviesa la base. Mediante una platina se acopla el eje del motor con el brazo del péndulo. La platina cuenta con un tornillo de manera de asegurar la misma al eje del motor. Todo esto se puede observar en la figura 2.9.



Figura 2.9: Foto del acople del motor

El péndulo está compuesto por un portaeje que consiste de una plancha de metal en forma de U, de ahora en más *Soporte en U*, de largo horizontal 12,6 cm. En un extremo tiene un soporte para el encoder y en el otro extremo tiene un soporte para un rodamiento. En el portaeje también se tienen los agujeros para atornillar el brazo a la platina de acople. El eje o brazo del prototipo está apoyado sobre el rodamiento en un extremo y sobre el eje del encoder en el otro. Este eje es solidario al péndulo, acoplando el giro del último al giro del encoder. En la figura 2.10 se ve en detalle el soporte en U con el brazo, el péndulo y el encoder.

1. Encoder
2. Agujeros para asegurar la platina de acople
3. Soporte para el encoder
4. Eje o brazo
5. Rodamiento
6. Soporte para el rodamiento
7. Acople brazo-péndulo.
8. Péndulo



Figura 2.10: Nomenclatura del péndulo.

## 2.4. Conexionado

En la figura 6.1 se muestran los componentes del prototipo y cómo se comunican entre ellos. Se utilizaron dos encoders,  $ENC_M$  y  $ENC_P$  para medir ángulos y velocidades, y un sensor de corriente para medir la corriente del motor. En su conjunto nos permiten sensar las variables de estado del sistema. Las flechas indican el sentido de los datos. Además se muestra el tipo de información que se comparte.

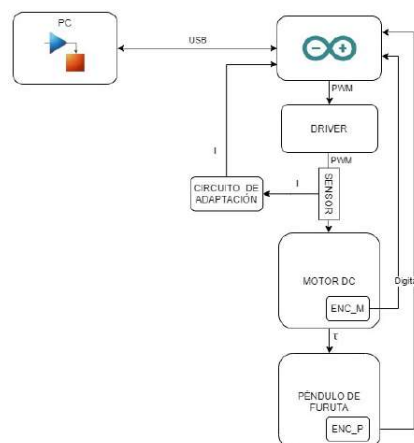


Figura 2.11: Diagrama lógico del sistema.

### Nota

La comunicación USB entre la PC y el Arduino es opcional si se desea correr el programa directamente en el micro-controlador utilizando la herramienta *Deploy to Hardware* de *Simulink*.

En primer lugar se encuentra la comunicación USB entre la PC y el Arduino. En la PC corre la aplicación Simulink en donde se desarrollan los controladores y se programa el microcontrolador, aquí también se almacena la información de los ensayos para su posterior análisis. El Arduino, por otro lado, se conecta hacia los demás dispositivos como lo son los encoders (lecturas digitales), el driver del motor DC (escrituras PWM) y el sensor de corriente (una vez acondicionada la señal mediante el circuito de adaptación, detallado en la sección 3.1). Por último, el driver es el encargado de imponer la tensión en bornes del Motor DC a partir del valor PWM que reciba.

Para esto el driver se encarga de generar una onda PWM para establecer distintos voltajes en los bornes del motor. Además consta de un puente  $H$  para poder hacer circular corriente en ambos sentidos. Permite utilizar una fuente de alimentación desde 5 V hasta 35 V. Tiene dos modos de funcionamiento discriminando según la alimentación sea superior o inferior a 12 V. Se utiliza el modo inferior a 12V.

## 2.4. Conexionado

Para manejar el driver es necesario utilizar tres pines del microcontrolador. Dos de ellos digitales utilizados para fijar el sentido de circulación de la corriente por el motor, y por lo tanto el sentido de giro. El tercero es un pin analógico en el cuál se le indica a tensión a aplicar en bornes del motor. En este pin analógico el Arduino genera una onda PWM, logrando un valor de continua entre  $0\text{ V}$  y  $5\text{ V}$ . El driver impone un valor entre  $0\text{ V}$  y  $V_{cc} = 12\text{ V}$ , proporcional al valor de continua en este pin. Todos los pines son de  $5\text{ V}$ .

Los encoders tienen dos salidas digitales cada uno. Es necesario disponer de resistencias de *pull up* para el manejo de estas señales, para ello se utilizaron las resistencias de internas del Arduino.

Los encoders no se pueden conectar a cualquier pin digital ya que para llevar la cuenta de la posición del encoder se utilizan interrupciones. Cada vez que se produce un cambio en una de las señales de los encoders se produce una interrupción. No todos los pines digitales del Arduino pueden hacer esto. Se utilizan los pines digitales 18 y 19 para el encoder del motor, y los pines digitales 2 y 3 para el encoder que mide la posición del péndulo.

La salida del sensor de corriente se conecta a un circuito de adaptación de escala que mejora su resolución (detallado en la sección 3.1). La salida del circuito se conecta pin analógico *A0* del Arduino. La alimentación del circuito de adaptación y del sensor son ambas de  $5\text{ V}$ .

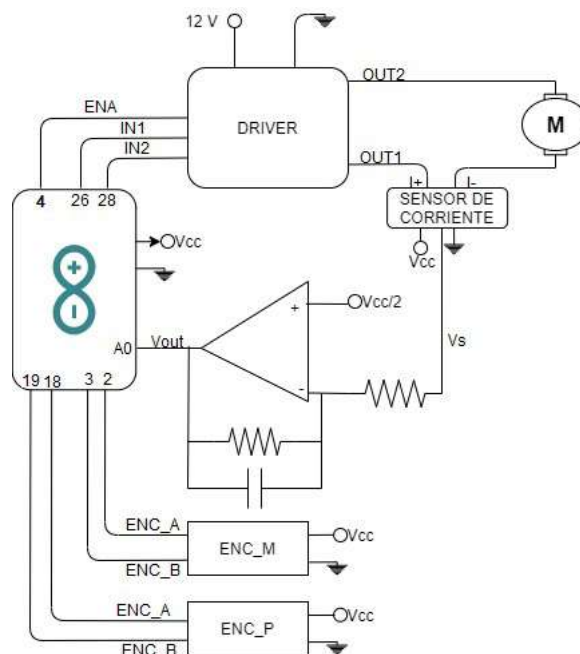


Figura 2.12: Diagrama de conexionado eléctrico.

## Capítulo 2. Diseño y Construcción del Péndulo

Se presenta en la figura 2.12 el esquema de conexionado resultante, donde se indican los pines del Arduino utilizados.

### 2.5. Diseño de PCB

En esta sección del capítulo se comentará sobre el proceso de diseño del PCB donde se realizaron las conexiones necesarias para el prototipo.

Esto incluye montaje para el Arduino Mega, Driver del motor (L298N) y sensor de corriente (ACS712) con su respectivo circuito de adaptación. También se agrega al PCB los conectores para alimentación y para los componentes que van hacia el Péndulo de Furuta (para encoders y motor DC).

El diseño fue realizado con la ayuda de la herramienta Eagle, donde se realizó el esquemático, el diseño de la placa y se generó el archivo *Gerber* que incluye la información necesaria para la fabricación de la placa. Este archivo incluye por ejemplo información sobre cantidad de capas, su contenido y detalles como su espesor, posición de los agujeros, pines y dimensiones entre otras cosas. Se hizo uso de las reglas de diseño proporcionadas por Sparkfun<sup>1</sup> y se buscó un fabricante. Se optó por PCB Way (Ubicada en Shenzhen, China) para la fabricación del circuito bajo recomendación de buen servicio y velocidad de entrega, quienes cumplieron nuestras expectativas.

Antes de adentrarnos en el resultado final se detallan los criterios de diseño utilizados:

- Se optó por un diseño independiente de sus componentes, es decir que cada uno de ellos sea sustituible sin necesidad de desoldar de ser posible.
- Sin considerar restricciones de tamaño a no ser por la estética del prototipo.
- Reutilizar los componentes ya adquiridos. La sustitución de algún componente implicaría remodelar el bloque que lo conforma. Esto se debe a que la placa se fabricó en las etapas finales del proyecto.

#### 2.5.1. Esquemático

Comenzamos presentando el esquemático del circuito, en el mismo se presentan todos los objetos que debe incluir la placa. Se presentan a modo de visualización de las conexiones necesarias, qué componentes se utilizarán, su tamaño, modelo y tipo de conector. La mayoría de los componentes utilizados se obtuvieron a partir de la librería proporcionada por Sparkfun<sup>2</sup> donde se proporcionan un conjunto

---

<sup>1</sup>Extraídas de <https://www.sparkfun.com/tutorials/115>

<sup>2</sup>Librería extraída de Extraída de <https://github.com/sparkfun/SparkFun-Eagle-Libraries>

elementos básicos y reglas que facilitan el proceso de diseño.

En la figura 2.13 se puede observar el esquemático del circuito a diseñar. Aquí se definen los conectores para los encoders, el driver, el motor, el sensor de corriente y la alimentación. Se incluye también el circuito de adaptación presentado en la sección 3.1. Es necesario incluir los agujeros de la placa (en la margen inferior de la figura) que se utilizarán para el montaje del Driver y para afirmar la placa hacia un soporte externo. Las conexiones fueron realizadas acorde a las indicadas en la sección Conexionado.

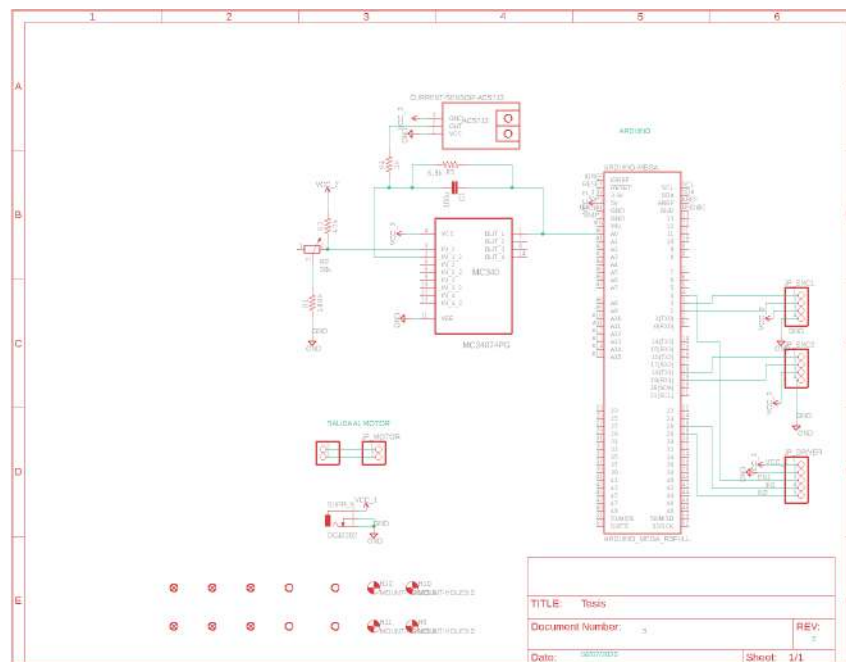


Figura 2.13: Esquemático del circuito a realizar

### 2.5.2. PCB Layout

Una vez concluido el esquemático se deben posicionar los componentes en la placa y definir las pistas. Para esto se utiliza la sección *board* de la herramienta Eagle. La ubicación de los componente se hace de manera que los conectores hacia la PC y la alimentación se ubiquen hacia el frente de la placa (donde se ubicaría el usuario) y en el lado opuesto, se colocan los conectores hacia el péndulo, agrupando los conectores por componente facilitando al usuario a interpretar qué conector refiere a qué componente.

Por último, dada la cantidad de espacio disponible se decide utilizar pistas anchas, disminuyendo la posible caída de voltaje en las mismas y aumentando los puntos de contacto entre los aros de cobre y las pistas. Se fue más generoso aún



### 2.5.3. Resultado final del PCB

Luego de realizar chequeo de continuidad acordé a las conexiones se procede a montar los componentes. En la figura 2.15 se muestra el resultado final del PCB.

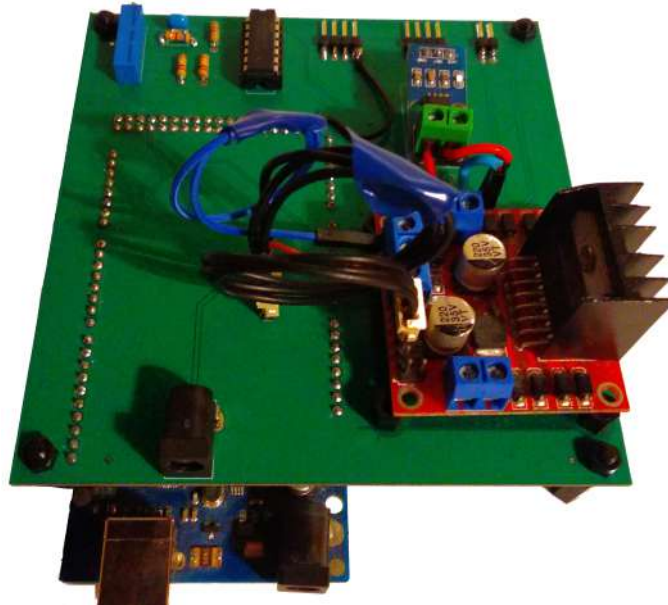


Figura 2.15: Resultado final del PCB

Para facilitar el conexionado, se presenta en la figura 2.16 el código de colores utilizado para los conectores. En la tabla 2.2 se detalla el conexionado de los conectores.

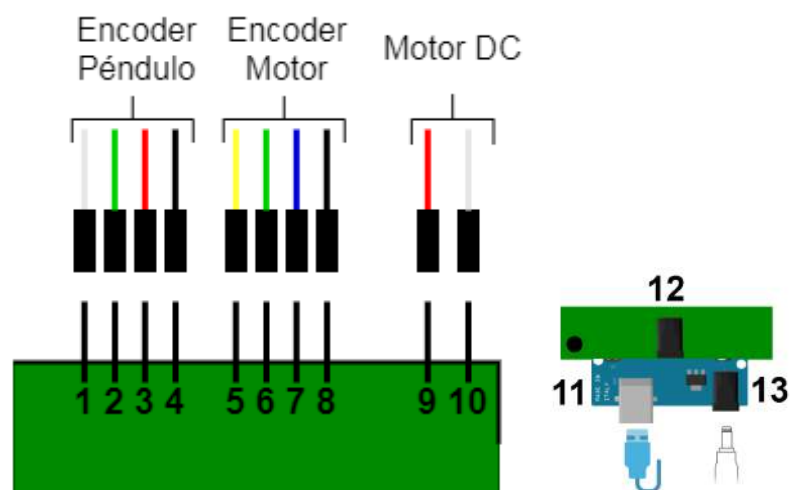


Figura 2.16: Diagrama de conexiones del PCB

## Capítulo 2. Diseño y Construcción del Péndulo

PIN	Color	Componente	Referencia
1	Blanco	Enc Péndulo	Fase A
2	Verde	Enc Péndulo	Fase B
3	Rojo	Enc Péndulo	5V
4	Negro	Enc Péndulo	GND
5	Amarillo	Enc Motor	Fase A
6	Verde	Enc Motor	Fase B
7	Azul	Enc Motor	5V
8	Negro	Enc Motor	GND
9	Rojo	Motor	V+
10	Blanco	Motor	V-

PIN	Componente	Referencia
11	Conector USB	Conexión PC-Arduino
12	Conector Jack	12V
13	Conector Jack	5V (Opcional)

Tabla 2.2: Código de colores para conectores del PCB

Se expone a continuación los componentes principales de la placa:

- Montaje del estilo *Shield* para Arduino Mega.
- Montaje sensor de corriente ACS712.
- Dos conectores de 4 pines hacia encoders.
- Conector de 2 pines hacia el motor.
- Circuito de adaptación para sensor de corriente.
- Montaje para ubicar la placa en el soporte.
- Montaje Driver L298n para motor DC.
- Conector alimentación Jack 12V.
- Conector alimentación Jack 5V(Arduino).
- Conector USB(Arduino).

Con esto se da por finalizado el diseño y construcción del PCB.



## Capítulo 3

# Acondicionamiento de sensores y actuadores

En este capítulo se mostrarán las pruebas realizadas para validar cada uno de los sensores y el actuador. Además se mostrarán los problemas que se presentaron en cada caso, las causas y las soluciones implementadas. Sobre este punto el mayor problema se presentó en el driver dado que al conectarlo al motor se observó un comportamiento que no era el esperado. También se mostrará el circuito amplificador para la medida del sensor de corriente.

### 3.1. Sensor de corriente

La salida del sensor es una tensión de la forma

$$V_o = \frac{V_{CC}}{2} + sI$$

Donde  $s$  es la sensibilidad del sensor, cuyo valor típico  $s = 185 \frac{mV}{A}$ , y  $V_{CC}$  es la tensión de alimentación del sensor. El conversor AD de los pines analógicos del Arduino es de 10 bits. Se trabaja con nivel de 5 V, por lo tanto una cuenta del conversor es

$$5 V/1024 = 4,9 mV$$

Por lo tanto la apreciación de la medida sería de:

$$4,9mV/185 \frac{mV}{A} = 26 mA$$

Dado que el fondo de escala del sensor es de 5 A y el consumo máximo del motor es de 2 A contamos con un margen para poder ajustar la escala. Para ello se utiliza un circuito amplificador. En la imagen 3.1 se presenta el circuito acondicionador utilizado.

### Capítulo 3. Acondicionamiento de sensores y actuadores

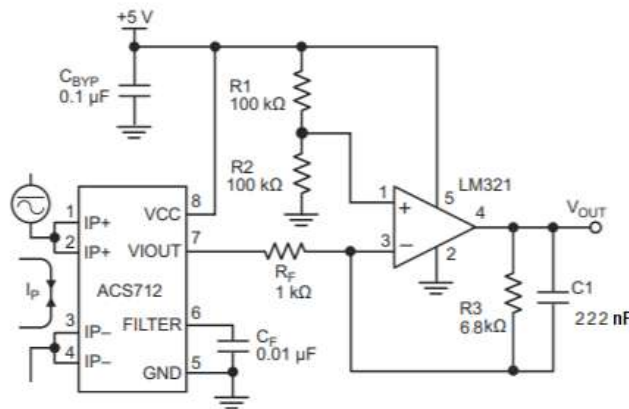


Figura 3.1: Circuito acondicionador para el sensor de corriente. Se observan los componentes con sus valores de diseño. Extraído de <https://www.sparkfun.com/datasheets/BreakoutBoards/0712.pdf>, Aplicación 3.

La ganancia del circuito implementado se relevó en  $6,85 V/V$ , esto mejora la apreciación en la medida de la corriente. La nueva apreciación es de  $3,8 mA$ , y la nueva sensibilidad es:

$$6,85 \times 185 \frac{mV}{A} = 1267 \frac{mV}{A}$$

Teniendo  $5 V$  como nivel máximo del conversor del Arduino con la nueva sensibilidad se logra medir un rango de  $-2 A$  a  $2 A$  aproximadamente. El ancho de banda del sensor es de  $80 kHz$ , [8]. La señal de corriente hacia el motor es una onda PWM de  $1 kHz$ . Esto es porque la señal PWM de salida del driver es originada en base a la señal PWM del Arduino (que por defecto es de  $1 kHz$ ). El polo del motor es de algunos  $Hz$ . En conclusión el ancho de banda del sensor es más que suficiente para medir el sistema y además, para obtener el valor medio de la corriente se debe filtrar. Como se observa en la figura 3.1 se utiliza un filtro RC para filtrar la señal con las resistencia  $R_3$  y  $C_1$ . Finalmente, se ubica el polo del filtro en

$$\frac{1}{2\pi R_3 C_1} = 100 Hz$$

## 3.2. Driver y Motor DC

Para poder colocar la tensión deseada a la salida del driver es necesario conocer la relación entre el PWM indicado por el Arduino y la tensión de salida del driver. Aquí por tensión de salida nos referimos al voltaje medio en bornes del driver siempre y cuando el driver esté conectado al motor. En el Arduino el *duty cycle* del PWM se indica con un valor entre 0 y 255, mediante una relación lineal.

## 3.2. Driver y Motor DC

Idealmente el driver entrega una onda cuadrada de una cierta frecuencia y cierto ciclo de trabajo. Variaciones en el ciclo de trabajo permiten obtener distintos voltajes medios a la salida. La excursión de salida del dispositivo queda determinada por la electrónica que lo compone, con un rango de operación aproximadamente entre  $[0V, V_{CC} - 1,5V]$ , donde  $V_{CC}$  es de alimentación. Teniendo esto en cuenta es sensato asumir que la relación PWM - salida es lineal y así generalmente se asume. El problema es que lo anterior no es tan cierto. En la subsección 3.2.1 veremos que la relación entrada-salida está lejos de ser lineal. En 3.2.2, en base a observaciones del chip, se dan razones teóricas para este comportamiento, y simulaciones que las respaldan. Finalmente en la sección 3.2.3 se plantea una solución para poder tener una relación lineal.

### 3.2.1. Salida

Lo primero a ver es la respuesta del driver en *vacío*. Como esto no es posible, se coloca una resistencia grande entre sus bornes. En este caso el resultado es el esperado, una recta desde el origen hasta el punto de tensión máxima. Sin embargo, al medir la respuesta con el motor la relación medida cambia significativamente, y está lejos de ser lineal. No solo esto, sino que además, en cierto rango, supera la tensión de vacío. En la figura 3.2 se observan superpuestas la salida en vacío y con el motor conectado.

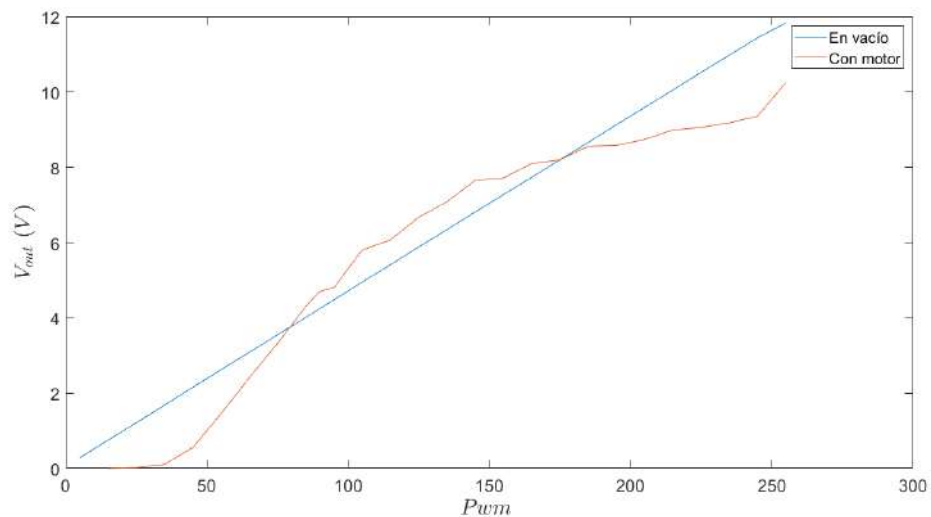


Figura 3.2: Salida del driver con el motor conectado vs salida en vacío.

Si observamos con un osciloscopio la salida que el driver está entregando en bornes del motor, obtenemos la onda mostrada en la figura 3.3. La misma no es una onda cuadrada. En la parte alta entrega 12 V pero en la parte baja no entrega 0 V sino que vemos un sobrepico de tensión negativa y luego una tensión constante de aproximadamente 6 V.

### Capítulo 3. Acondicionamiento de sensores y actuadores

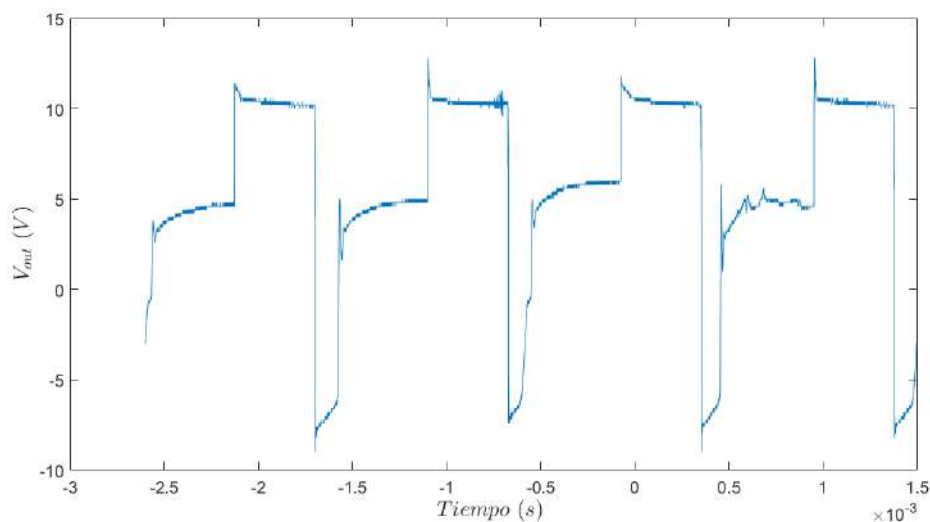


Figura 3.3: Captura del osciloscopio: voltaje PWM entregado por el Driver en bornes del motor.

#### 3.2.2. Causa

Si observamos el esquemático del driver se puede ver que en nivel alto impone  $V_{ON}$  pero en nivel bajo no impone  $0 V$ , sino que impone  $I = 0 A$ .

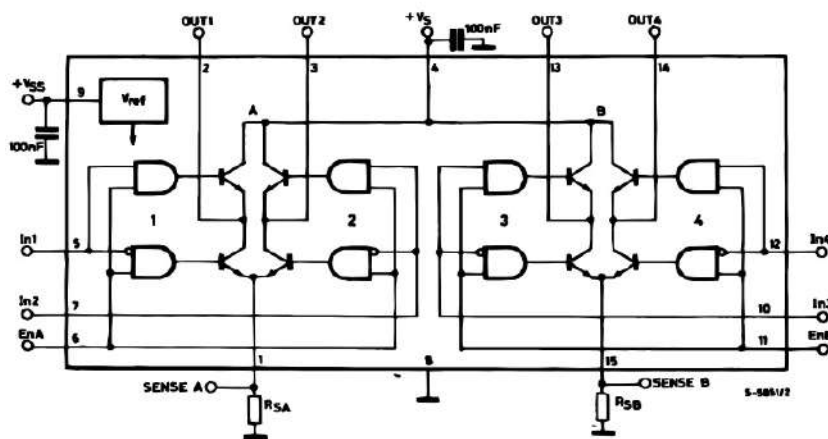


Figura 3.4: Esquemático del driver. Extraído de [9]

Esto se corrobora haciendo un relevo de la corriente. Se presenta el resultado en la figura 3.5 donde se observan algunos ciclos de PWM y se ve claramente como en la parte OFF del ciclo la corriente es aproximadamente  $0 A$ .

Esta diferencia da lo mismo a la hora de tener una carga resistiva, pero no a la hora de tener una carga inductiva. Para contar con una explicación teórica,

### 3.2. Driver y Motor DC

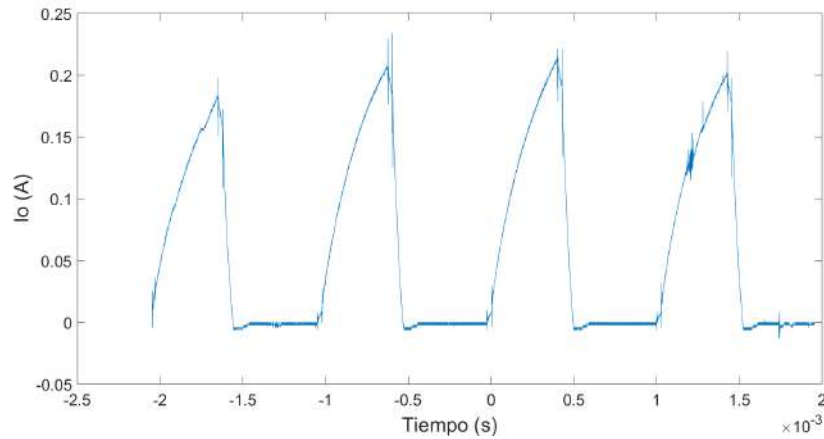


Figura 3.5: Corriente entregada por el driver al motor.

asumamos el modelo del motor mostrado en la figura 3.6 donde  $e(t) = kw(t)$  y  $\tau(t) = ki(t)$ .

A efectos de esta explicación, supongamos que la fricción es simplemente viscosa y lineal, es decir,  $f = bw$ . Llamemos  $i_0$  y  $w_0$  a la corriente y la velocidad, respectivamente, que tiene el motor antes de que el driver imponga  $i = 0$ . Si resolvemos las ecuaciones desde ese momento en adelante, obtenemos

$$v(t) = kw_0 e^{-\frac{bt}{J}} - Li_0 \delta(t) \approx kw_0 - Li_0 \delta(t)$$

donde la aproximación se justifica en el hecho de que la constante de tiempo de la exponencial está formada por parámetros puramente del motor, por lo cual la constante de tiempo es mucho más lenta que el período del PWM, a recordar  $1 \text{ ms}$ .

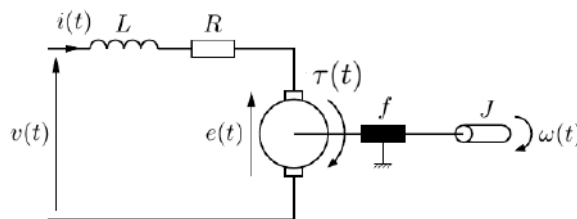


Figura 3.6: Modelo Motor de corriente continua.

Esta solución teórica coincide, *a menos de  $\epsilon$  en distribuciones*, con la respuesta vista en la figura 3.3. Se verifica que  $kw_0$  es igual al voltaje constante observado.

Otra observación es que el sobrepico observado en 3.3 parece estar recortado. Esto se explica en que la placa que utiliza el chip del driver trae incorporado unos diodos de protección que limitan las tensiones en los bornes del driver, ver [10].

## Capítulo 3. Acondicionamiento de sensores y actuadores

La causa propuesta es respaldada por una simulación. Se simuló la salida del driver cuando se conecta al modelo del motor de la explicación previa. Para esto se impuso de entrada una onda cuadrada seguida de una llave. La llave en estado ON se modela como una resistencia de  $0 \Omega$  y en estado OFF por una resistencia de valor  $350 \Omega$ . Con la llave se simula el corte que hace el driver cuando se le pide  $V = 0$ . Además de esto se limitó el voltaje en  $-10 V$ . En la figura 3.7 se observan los resultados de la simulación, los mismos confirman la forma de la señal en bornes del driver. Los parámetros y el modelo de la llave se eligieron de forma arbitraria, por lo cual la idea de esta simulación es que sea únicamente un insumo cualitativo.

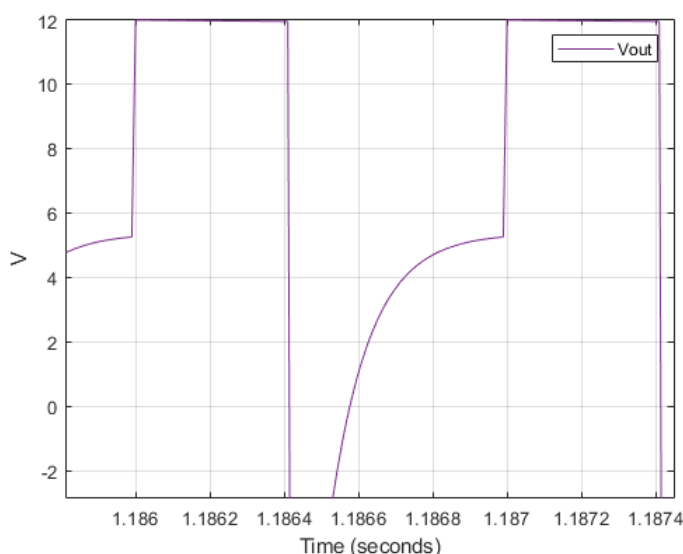


Figura 3.7: Resultados de la simulación

Por último, también se probaron *snubbers* de sobretensiones RC, es decir, un circuito formado por la serie de un capacitor y una resistencia colocado en bornes del motor. Lo mismo funcionó, dando otro apoyo a la explicación teórica. Vale aclarar que los circuitos probados eliminaron el sobrepico de tensión pero solamente modificaron la tensión constante que sigue luego del mismo.

### 3.2.3. Compensación

#### Una posible solución

Una forma estándar de atacar estos problemas es utilizar un *snubber* de sobretensiones, en particular un *snubber* RCD, es decir circuitos formados por resistencias, capacitores y diodos. La idea es que estos circuitos supriman el sobrepico de tensión y que además permitan tener una tensión cercana a 0 cuando el driver esta OFF. El snubber clásico es utilizar un *diodo de rueda libre*<sup>1</sup>. La idea sería colocar

<sup>1</sup>también conocido como *flyback diode*, *kickback diode*, *snubber diode*, *commutating diode*, *suppression diode*, *clamp diode*, o *catch diode*.

### 3.2. Driver y Motor DC

un diodo en bornes del motor. Un esquema de la solución se muestra en la figura 3.8.

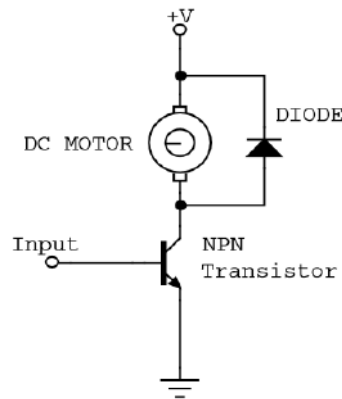


Figura 3.8: Ejemplo de un diodo de rueda libre, para un driver sin puente en H.

Esto sería una solución ideal, el problema es que solo funciona con motores que van a girar en un solo sentido, pues el diodo no permitirá que gire en uno de los sentidos. Esto prácticamente descarta la posibilidad de usar un snubber RCD, dejando a prueba los snubbers RC. Al final se optó por utilizar una solución por software, dejando de lado esta posibilidad. Para el lector interesado en seguir por este camino, se recomienda fuertemente [11], [12] y [13], como referencia para el diseño de circuitos snubbers.

#### Solución Implementada

Idealmente, la respuesta del driver es una función  $f : [-255, 255] \rightarrow [-10,36, 10,25]$  que lleva PWM a voltaje en bornes del motor. Aquí se está suponiendo que el PWM entrega un voltaje medio independientemente del nivel de carga que presente el motor. La relación dada por  $f$  no es nada obvia, pero es invertible. Esto lleva a la solución mostrada en la figura 3.9, donde  $V_{in}$  es un voltaje elegido por el Arduino.

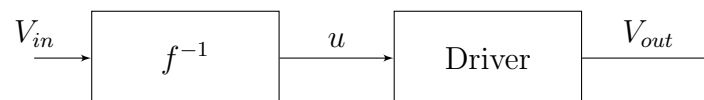


Figura 3.9: Solución Implementada

Idealmente este procedimiento implica que  $V_{out} = V_{in}$  cuando  $-10,36 V < V_{in} < 10,25 V$ , dando una manera sencilla de manejar el voltaje medio en bornes del motor. Esto permite que la variable de control sea el mismo  $V_{out}$ .

### Capítulo 3. Acondicionamiento de sensores y actuadores

Para chequear el buen funcionamiento se implementa y se mide la relación  $V_{in} \mapsto V_{out}$ . En las siguientes figuras se muestran los resultados. En la figura 3.10 se fue variando el voltaje  $V_{in}$  de  $-10\text{ V}$  a  $10\text{ V}$ . En la figura 3.11 se fue variando  $V_{in}$  de  $10\text{ V}$  a  $-10\text{ V}$ . En ambas figuras se representa la identidad en rojo y en azul la curva relevada.

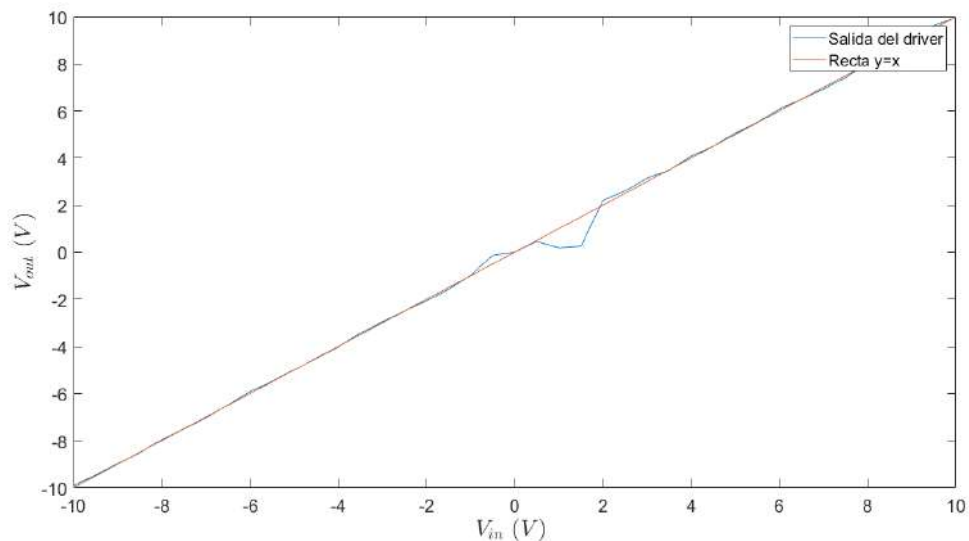


Figura 3.10: Subida: relevo de  $V_{in} \mapsto V_{out}$  con el compensador. Se hizo crecer  $V_{in}$  de  $-10\text{ V}$  a  $10\text{ V}$ .

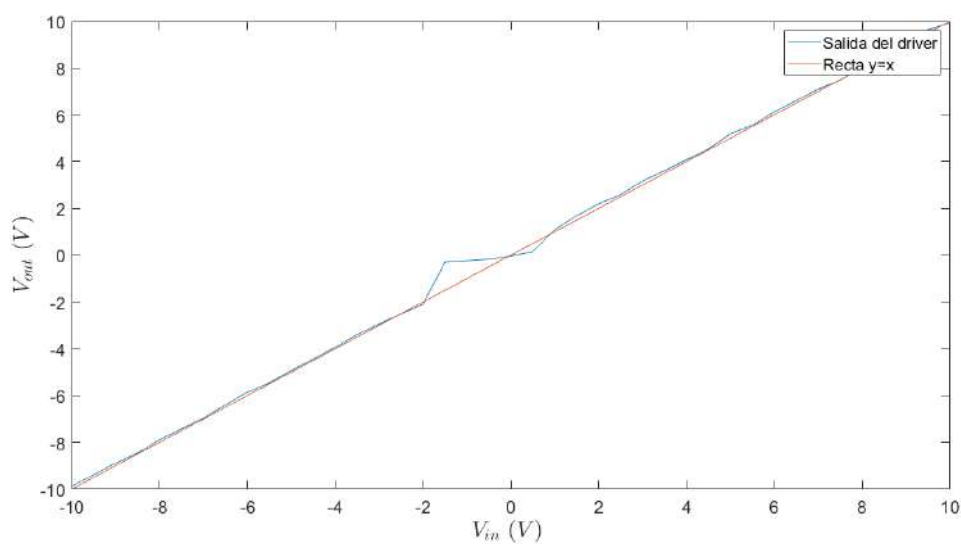


Figura 3.11: Relevo de  $V_{in} \mapsto V_{out}$  con el compensador. Se hizo decrecer  $V_{in}$  de  $10\text{ V}$  a  $-10\text{ V}$ .

Los resultados presentan una fuerte coincidencia fuera del origen con la curva



### 3.2. Driver y Motor DC

esperada. Donde presentan una diferencia es cerca del cero, y solo de un lado del mismo. Esta diferencia se provoca en la zona muerta<sup>2</sup>. Según datos relevados la zona muerta del motor provoca que se salte de un voltaje pequeño (aproximadamente  $0,4\text{ V}$ ) a los  $2\text{ V}$ . Por lo tanto la zona  $[-2\text{ V}, 2\text{ V}]$  corresponde a la zona muerta. Es por esto que no se tiene un buen relevo de  $f$  en dicha zona, explicando que haya diferencias.

Para tener algunos datos de  $f$  en dicha zona se utilizó la histéresis, es decir, el hecho de que la zona muerta es más pequeña si uno se mueve del movimiento al reposo, en lugar del reposo al movimiento. De esta manera se relevó  $f$  con el motor entrando en la zona muerta teniendo movimiento, con esto se logró relevar más datos (pues el motor se detiene con un voltaje menor), pero no muchos más.

Esto explica por qué hay diferencias cerca del cero pero solo de un lado del mismo. En el caso de *subida* el motor entra en la zona  $[-2\text{ V}, 0\text{ V}]$  estando en movimiento, pero el motor entra en la zona  $[0\text{ V}, 2\text{ V}]$  estando en reposo. Es por esto que en la gráfica de subida la curva azul se parece más a la identidad en  $[-2\text{ V}, 0\text{ V}]$  que en  $[0\text{ V}, 2\text{ V}]$ . En el primer intervalo se tiene datos sobre  $f$  y en el segundo de hecho el motor está quieto, y no hay datos sobre  $f$ . El mismo análisis vale para la gráfica de bajada. Para terminar, en la figura 3.12 se presenta la comparación entre los dos relevos, mostrando que solo difieren en la zona muerta.

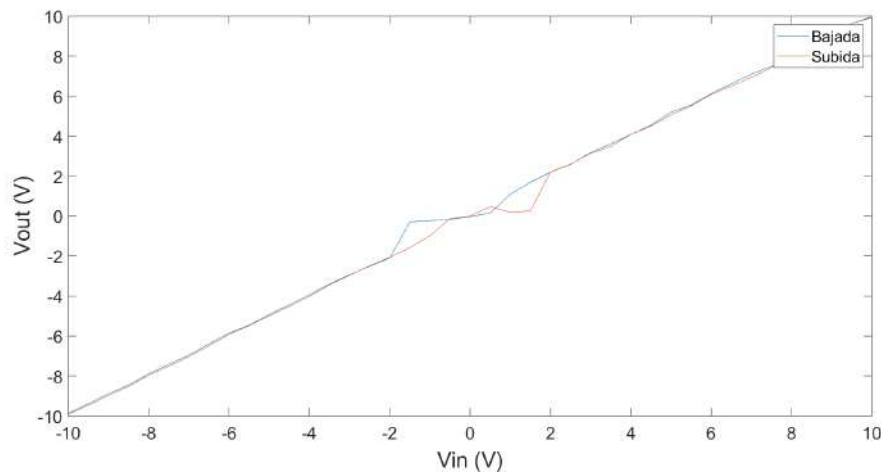


Figura 3.12: Comparación entre la curva de subida y la de bajada. Las diferencias se presentan en la zona muerta.

<sup>2</sup>Se entiende por zona muerta a la mínima señal de entrada necesaria para que el sistema responda

### 3.3. Encoder

En esta sección se describe las pruebas realizadas sobre el encoder para medir posición, la investigación y el desarrollo de algoritmos para medir velocidad y las pruebas que se hicieron para testear los mismos. Las principales referencias utilizadas son [14], [15] y [16].

#### 3.3.1. Midiendo posición

Los encoders naturalmente miden la posición angular de su eje por lo cual no es necesario el desarrollo de un algoritmo muy sofisticado para poder medir. Los encoders utilizados son incrementales. Los mismos se basan en dos señales  $A$  y  $B$  que dependen de la posición del eje, y en donde siempre hay ondas cuadradas defasadas 90 grados. La figura 3.13 ilustra este hecho mostrando el funcionamiento de un encoder óptico.

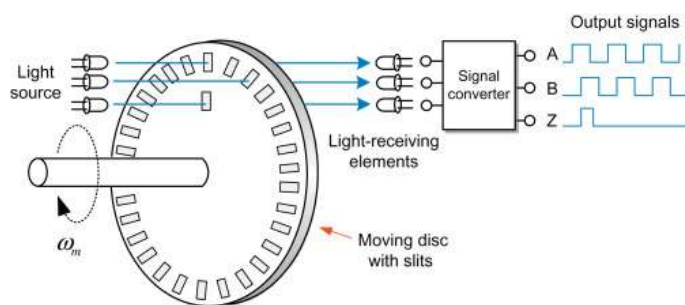


Figura 3.13: Ilustración del funcionamiento de un Encoder Incremental Óptico. Extraído <https://www.sciencedirect.com/topics/engineering/incremental-encoder>

Además las señales  $A$  y  $B$  tienen una cierta cantidad conocida de períodos por revolución, PPR. Mirando el patrón de señales  $A$  y  $B$  se puede determinar el sentido de giro del eje, y mirando cuántos pulsos han pasado se puede saber el desplazamiento angular.

Teniendo en cuenta lo anterior, se decidió implementar un algoritmo que funcionara con interrupciones provocadas por las señales  $A$  y  $B$ . Específicamente, se escribió una rutina de atención a las interrupciones que se dispara en cada flanco de las señales  $A$  y  $B$ , tanto de subida como de bajada. De esta manera, cada vez que se dispara la interrupción se tiene que el eje se ha movido en  $1/4$  de período de las ondas cuadradas. Además leyendo el valor actual de las señales y teniendo el valor anterior se puede determinar hacia donde giro el eje. En efecto, los patrones son:

$$\begin{aligned}
 AB &= 10 - 11 \quad 11 - 01 \quad 01 - 00 \quad 00 - 10 \quad \text{sentido 1} \\
 AB &= 10 - 00 \quad 00 - 01 \quad 01 - 11 \quad 11 - 10 \quad \text{sentido 2}
 \end{aligned}$$

donde, por ejemplo,  $AB = 11 - 01$  significa que primero llegó 11 y luego 01.

A modo de ejemplo, para el encoder que está posicionado en el brazo del péndulo, se tiene una resolución de 600 PPR por lo cual con este algoritmo se obtiene una resolución de  $360^\circ/2400 = 0,15^\circ$ . Los distintos test fueron exitosos.

### 3.3.2. Testeando medidas de velocidad

Para testear los distintos algoritmos para medir velocidad se utilizaron básicamente dos pruebas. La primera consiste en medir distintas velocidades de un torno y la segunda medir las velocidades del motor. El primer test surgió debido a que cuando se comenzaron a desarrollar los algoritmos del encoder no se contaba con el motor.

El primer test consistía en colocar el brazo del péndulo en las mordazas del torno y prender el mismo a distintas velocidades. El encoder se encontraba al final del brazo con su eje acoplado al mismo. Las velocidades que se pudieron testear con este método fueron 15 *RPM*, 30 *RPM* y 130 *RPM*.

En el segundo test no solo se pudo testear velocidades constantes de entre 0 *RPM* y 80 *RPM* sino también observar y evaluar distintas respuestas transitorias.

La forma más simple de estimar velocidad es realizar el cociente incremental  $v = \frac{\Delta \text{Pos}}{\Delta T}$ . Hay dos opciones clásicas, un algoritmo de tiempo fijo y uno de espacio fijo.

### 3.3.3. Algoritmo de tiempo fijo

De tiempo fijo se refiere a ejecutar código cada cierto tiempo fijo ( $\Delta T$  constante), medir el cambio en la posición y realizar el cociente incremental. Se testeó este algoritmo y distintas variantes del mismo, se programó el microcontrolador para realizar interrupciones a distintas frecuencias. Los resultados son buenos a velocidades altas pero no lo son a bajas. El problema de este tipo de algoritmo es que la resolución de la velocidad termina siendo de  $\frac{1 \text{ count}}{\Delta T}$ , por lo cual si la frecuencia de muestreo es muy alta el error a bajas velocidades es muy alto. Para obtener una buena estimación de bajas velocidades se debe tener una frecuencia de muestreo demasiado lenta. Lo anterior se comprobó en las pruebas ya mencionadas.

Al trabajar con un péndulo las velocidades son esperables de ser bajas, e incluso de rondar el reposo. Como resultado se concluyó descartar el algoritmo.

### 3.3.4. Algoritmo de espacio fijo

De espacio fijo se refiere a ejecutar código cada vez que cambian las señales del encoder ( $\Delta \text{pos}$  constante), medir el tiempo transcurrido y realizar el cociente

### Capítulo 3. Acondicionamiento de sensores y actuadores

incremental. En este caso la resolución de la velocidad es variable. Idealmente este método no tiene error, pero teniendo en cuenta la resolución del timer esto ya no es así. En efecto la velocidad medida será  $\frac{\Delta_{\text{pos}}}{\Delta T + e}$  con  $e$  un cierto error en la medida, error independiente del tiempo transcurrido. Mientras más pequeña es la velocidad mayor es el tiempo transcurrido y por lo tanto mayor es la precisión en la velocidad. Esto se debe a que  $e$  es más despreciable frente a  $\Delta T$  y a que el cociente es menos sensible a los errores. A velocidades altas el error  $e$  es considerable frente a  $\Delta T$  y además el cociente muy sensible al denominador, por lo cual la precisión es muy mala.

Las pruebas mostraron buenos resultados para este tipo de algoritmo a velocidades bajas, pero se constataron otros problemas. Un primer problema es el tiempo variable en el que una muestra se actualiza, siendo el mismo, dependiente de la velocidad del eje. Otro problema es que a altas velocidades el microcontrolador recibe muchas más interrupciones. Las velocidades cercanas al 0 también son un problema, ya que cuanto más lenta es la velocidad más demora en actualizarse. En el caso límite, si el eje se detiene de golpe entonces la velocidad no se actualiza por lo cual mide un valor de velocidad 0. Como resultado se decidió descartar el algoritmo.

Otros tipos de errores, tanto de usar tiempo fijo como espacio fijo, se pueden encontrar en [16].

#### 3.3.5. Algoritmo utilizado: Controlador PI

Este algoritmo implementa en digital el filtro/controlador mostrado en la figura 3.14.

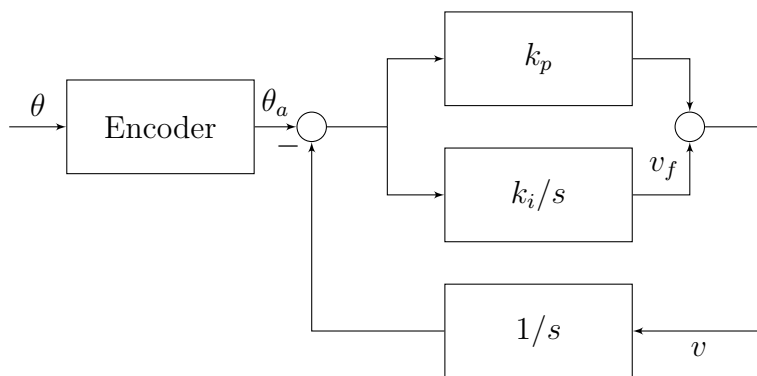


Figura 3.14: Diagrama de bloques del filtro utilizado

Una forma de entender el controlador es pensarlo como un seguidor de la señal  $\theta_a$ , utilizando los bloques proporcional e integral para asegurar estabilidad, buen tiempo de respuesta o buen ancho de banda, y proporcionar suficiente integradores en el lazo para poder seguir rampas. La señal  $\theta_a$  del encoder es una señal escalonada, la idea es que el sistema intente seguir a esta señal pero que el ancho de banda

sea suficientemente bajo como para que no logre copiar los escalones, produciendo una versión suavizada de  $\theta_a$ , y obteniendo su derivada en el nodo  $v$ .

Estudiando el controlador propuesto se observa que los polos del sistema están dados por los ceros del polinomio  $p(s) = s^2 + k_p s + k_i$ , siendo tan sencillo que ni se necesita lugar de las raíces para entender cómo se trasladan los polos. Se ubican los polos en el eje real de manera de no tener sobretiro y se eligen polos de manera de tener buena velocidad pero no tanta para poder rechazar los movimientos bruscos de  $\theta_a$ .

Si denotamos por  $\hat{x} = \mathcal{L}\{x\}$ , la transformada de Laplace de  $x$ , entonces la transferencia hacia la velocidad es

$$\frac{\hat{v}}{\hat{\theta}_a}(s) = k_p \frac{s(s + k_i/k_p)}{s^2 + k_p s + k_i}$$

lo cual efectivamente es un derivador a bajas frecuencias pero deja pasar las altas frecuencias! mostrando carencia en el argumento cualitativo anterior. Sin embargo, observando  $v_f$  se obtiene

$$\frac{\hat{v}_f}{\hat{\theta}_a}(s) = k_i \frac{s}{s^2 + k_p s + k_i}$$

lo cual es un derivador a bajas frecuencias y filtra las altas frecuencias. Además, dada la frecuencia  $w_d$  hasta la cual se desea tener un derivador, esta transferencia demuestra que lo más conveniente es comenzar a filtrar desde  $w_d$  en adelante, y para no tener banda pasante se debe colocar un polo doble.

En resumen, se elige  $k_i = k_p^2/4$  para tener un polo doble en  $s = -k_p/2$ , y se toma de salida a la señal  $v_f$ .

Para probar esta idea se implementó un algoritmo mediante interrupciones de un timer con frecuencias entre 5 kHz y 10 kHz. Se eligió el polo del filtro experimentalmente verificando que llegue a régimen arbitrariamente rápido y observando la respuesta en transitorio. Este ajuste fue cualitativo al no contar con un instrumento de medida externo. De aquí se determinó que con el polo en 16 Hz el filtro mostraba un comportamiento adecuado. El algoritmo superó ampliamente los resultados de las pruebas antedichas, sin presentar flaquezas en ningún rango de las velocidades trabajadas. En la figura 3.15 se muestra un relevo de la respuesta al escalón del motor, midiendo la velocidad con el algoritmo del encoder. Este ensayo se realizó en Arduino con una rutina que modificaba el voltaje impuesto en bornes del motor cada un tiempo determinado. Al mismo tiempo se relevaban los valores del encoder y se imprimieron en pantalla. Con estos datos se obtuvo la figura mencionada.

### Capítulo 3. Acondicionamiento de sensores y actuadores

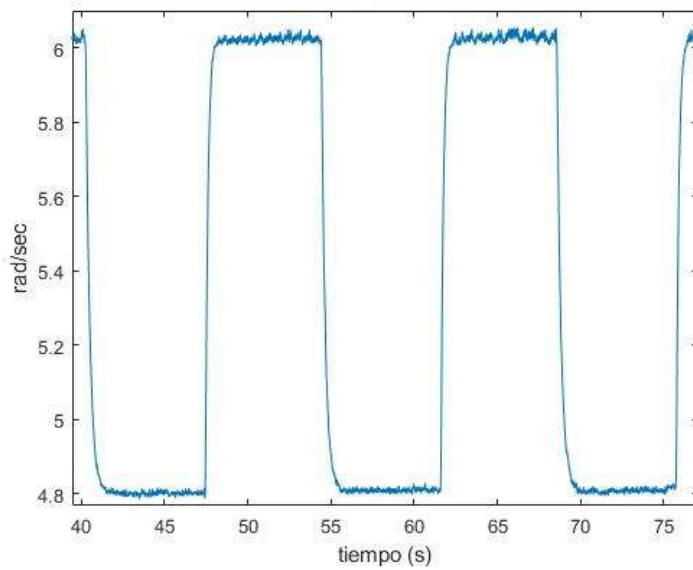


Figura 3.15: Relevo de respuestas al escalón del motor.

Es importante mencionar que para la implementación en el sistema final se tendrán más requerimientos sobre todo en la frecuencia de muestreo a utilizar. Esto requiere hacer algunas pruebas nuevamente con todo el sistema, lo cual se verá en el capítulo 6.

# Capítulo 4

## Modelado del Sistema/Péndulo

En este capítulo se presenta un modelo que comprende al sistema completo. El mismo se divide en dos grandes secciones, el Péndulo de Furuta y el motor junto con el driver.

A la hora de desarrollar un modelo es importante dejar claro los objetivos del mismo ya que son estos quienes determinan el nivel de profundidad del modelado. El modelo es una construcción teórica que pretende representar o predecir al sistema físico en cuestión. El objetivo aquí es obtener un modelo que permita diseñar controladores sobre el sistema real. Para eso el modelo debe tener una buena predicción pero a la vez ser lo suficientemente sencillo. No se pretende que el modelo prediga las señales en función del tiempo con excelencia, pero sí que logré reproducir cualitativamente el comportamiento del sistema real.

Para testear el modelo y darlo como definitivo se decidió que el mismo debe pasar por dos instancias de evaluación. En la primera se compara la respuesta del sistema físico con la respuesta del sistema modelado, ambos con una cierta entrada de voltaje en común. En función de los resultados se decide si el modelo es apropiado o es necesario hacer cambios. Si el modelo supera dicha instancia entonces queda aprobado siendo sujeto a cambios si y sólo si no pasa la segunda instancia. En la segunda instancia se somete el modelo a su objetivo principal, se intenta controlar el péndulo real diseñando un controlador con dicho modelo.

Entonces para verificar el modelo se necesita identificar los parámetros del mismo y someterlo a ensayos, y eventualmente al objetivo, controlar el péndulo. Es por esto que la verificación del modelo queda para los capítulos siguientes. En este capítulo se desarrolla el modelo dejando claro las hipótesis. En el capítulo siguiente se identifican los parámetros del mismo y se realiza la primer instancia de test. Al final del proyecto se realiza la segunda.

## 4.1. Péndulo

### 4.1.1. Introducción

En esta sección se presenta la derivación de las ecuaciones del péndulo de Furuta con el que trabajamos. La misma se ha llevado a cabo luego de revisar brevemente la literatura acerca del modelado del péndulo. Se ha seguido el enfoque llevado a cabo en la mayoría de los artículos revisados, derivar la dinámica a través del Lagrangiano en  $\mathbb{R}^2$  utilizando coordenadas locales, específicamente los ángulos.<sup>1</sup> Los apuntes [19] fueron utilizados como consulta de mecánica. En el desarrollo del modelo se ha consultado diversa bibliografía ([1–4, 6, 7, 18, 20–25]), si bien no todos se dedican a una explicación extensiva del modelado.

El capítulo se estructura como sigue: En la sección 4.1.2 se presentan hipótesis tomadas para derivar el modelo, en la sección 4.1.3 se muestran las medidas realizadas, en 4.1.5 se deducen las ecuaciones del movimiento. A lo largo de todo el capítulo se utilizará la terminología presentada en el 2. Recordamos aquí dicha nomenclatura.

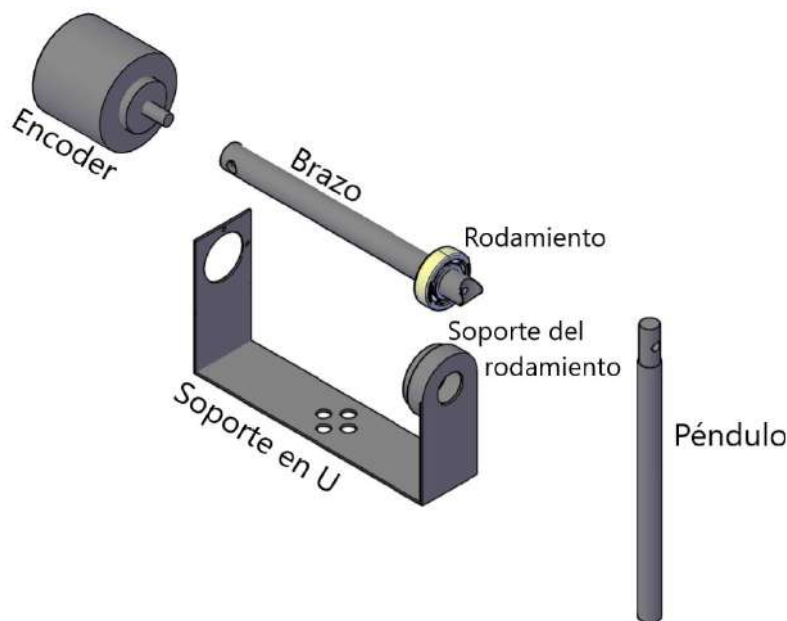


Figura 4.1: Nomenclatura utilizada sobre el péndulo de Furuta.

### 4.1.2. Hipótesis del modelo

Aquí se enumeran las hipótesis del modelo.

---

<sup>1</sup>Algunos autores sugieren el uso de  $S^1 \times S^1$  como espacio natural para derivar la dinámica del sistema, ver [17, 18].



1. Para describir la dinámica del sistema es suficiente modelar el mismo como los dos siguientes rígidos:  
Rígido 1: Soporte  $U$  + Soporte del Rodamiento + Rodamiento Exterior + Encoder  
Rígido 2: Brazo + Péndulo + Rodamiento Interior  
Respecto al Rodamiento, la mayoría de su peso esta en el anillo exterior y en el interior por lo cual se desprecia las municiones (ver sec 4.1.4, cuadro 4.2); el anillo exterior se considera parte del rígido 1 y el interior parte del brazo, aquí se considera como masa puntual.
2. El soporte del rodamiento puede modelarse como una figura bidimensional, de radio fijo.
3. El rodamiento exterior puede modelarse como una figura bidimensional, de radio fijo.
4. El brazo y el péndulo pueden modelarse como objetos unidimensionales.
5. El soporte en  $U$  puede modelarse como bidimensional.
6. El encoder puede modelarse como un rígido homogéneo con forma de cilindro.
7. El rodamiento interior puede modelarse como una masa puntual adherida al brazo.
8. Se puede despreciar la platina de acople del soporte en  $U$ .
9. No se considera el eje del motor, tiene inercia despreciable.
10. El rígido 1 y el rígido 2 cuentan con fricción estática y viscosa.
11. Conexiones rígidas entre entre el motor y el rígido 1, y entre el encoder y el brazo.
12. No hay juego (*backlash*) en las piezas.
13. Todos los materiales son homogéneos.
14. La base no se mueve.

### 4.1.3. Medidas del péndulo

En lo que sigue, el subíndice  $b$  se utilizará para el brazo, el subíndice  $p$  para el péndulo, el subíndice  $sr$  para soporte del rodamiento, el subíndice  $u$  para soporte en  $U$ ,  $e$  para encoder,  $re$  para rodamiento externo,  $ri$  para rodamiento interno. En general  $i$  y  $d$  se usan para izquierda y derecha. A su vez, las letras  $m, r, a, l$  se utilizan para denotar masa, radio, ancho y largo respectivamente. En la figura 4.2 y en la figura 4.3 se muestran algunas medidas junto con bases vectoriales útiles (más adelante se detallan mejor). En el cuadro 4.1 se resumen todas las medidas.

#### Capítulo 4. Modelado del Sistema/Péndulo

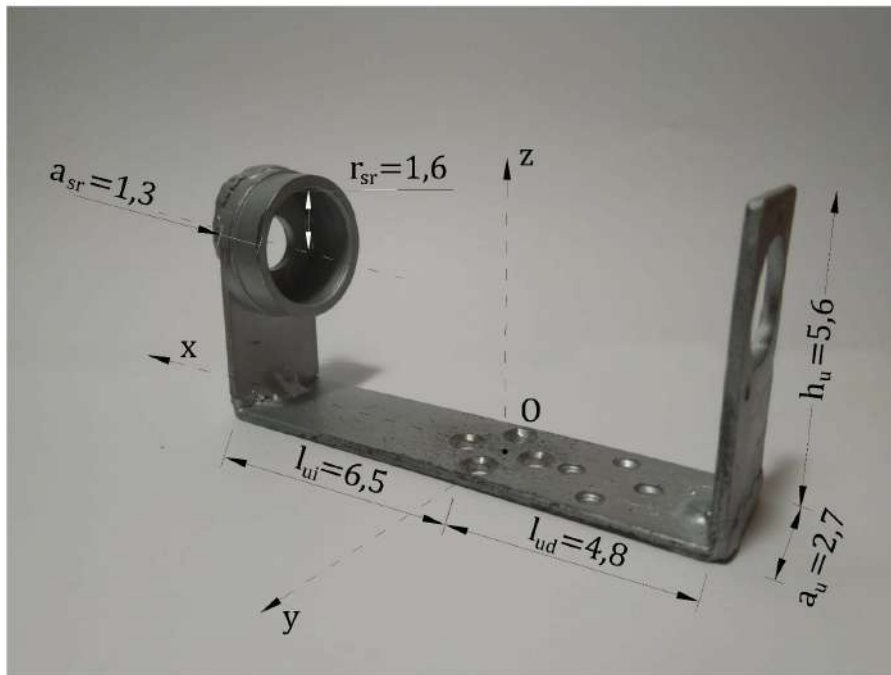


Figura 4.2: Soporte en U + Soporte del rodamiento: Se muestran algunas de sus medidas. Todas las medidas están en *cm*.

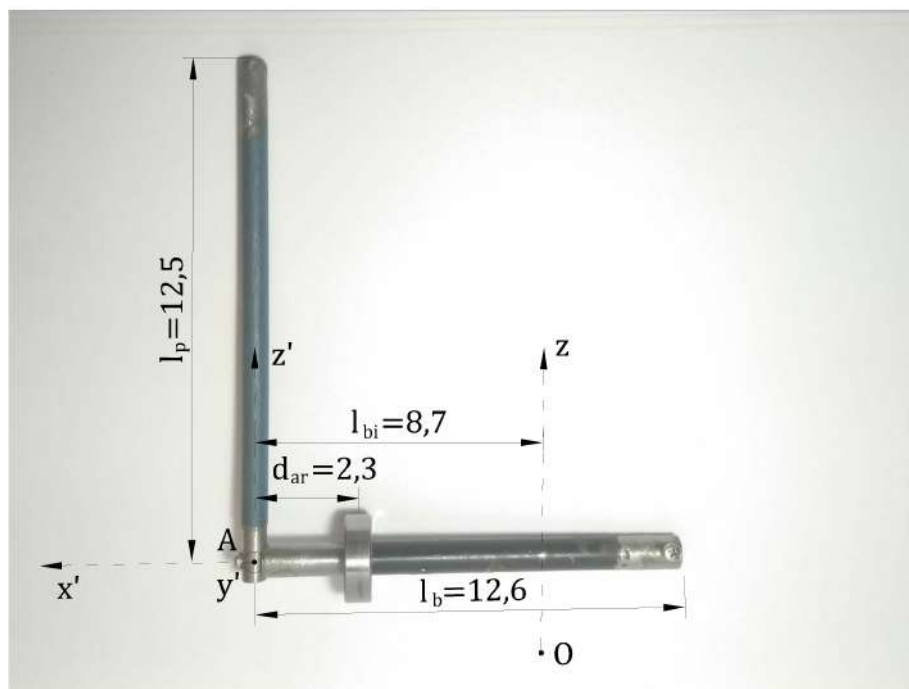


Figura 4.3: Péndulo + Brazo + rodamiento. Solo la parte interior del rodamiento forma parte del rígido 2. Todas las medidas están en *cm*.

## 4.1. Péndulo

	Nombre	Parámetro	Valor
Rígido 1	Rodamiento Exterior	$m_{re}(4.1)$	$7,7 \pm 0,05 \text{ g}$
		$r_{re}$	$1,3 \pm 0,05 \text{ cm}$
		$a_{re}$	$0,8 \pm 0,05 \text{ cm}$
	Soporte rodamiento	$m_{sr}(4.2)$	$31,6 \pm 0,05 \text{ g}$
		$r_{sr}$	$1,6 \pm 0,05 \text{ cm}$
		$a_{sr}$	$1,3 \pm 0,05 \text{ cm}$
	Soporte $U$	$l_{ui}$	$6,5 \pm 0,05 \text{ cm}$
		$l_{ud}$	$4,8 \pm 0,05 \text{ cm}$
		$a_u$	$2,7 \pm 0,05 \text{ cm}$
		$h_u$	$5,6 \pm 0,05 \text{ cm}$
		$m_u(4.3)$	$109,5 \pm 0,05 \text{ g}$
	Encoder	$m_e(4.4)$	$73 \pm 0,05 \text{ g}$
		$r_e$	$3,5 \pm 0,05 \text{ cm}$
$a_e$		$3,3 \pm 0,05 \text{ cm}$	
Rígido 2	Péndulo	$l_p$	$15,2 \pm 0,05 \text{ cm}$
		$m_p$	$57 \pm 0,05 \text{ g}$
	Brazo	$l_b$	$12,6 \pm 0,05 \text{ cm}$
		$m_b$	$35 \pm 0,05 \text{ g}$
		$l_{bi}$	$8,7 \pm 0,05 \text{ cm}$
	Rodamiento Interior	$m_{ri}(4.5)$	$6 \pm 0,05 \text{ g}$
		$d_{ar}$	$2,3 \pm 0,05 \text{ cm}$
	Masa total	$m_2$	$98 \pm 0,05 \text{ g}$

Tabla 4.1: Medidas

- (1) La masa de la parte exterior del rodamiento fue estimada en base a las medidas de un rodamiento similar. Por detalles ver el cuadro 4.2 en la sección 4.1.4 .
- (2) La masa del soporte del rodamiento se estimó midiendo el peso del tubo de donde salió dicho soporte, y asumiendo homogeneidad. Por detalles ver la sección 4.1.4.
- (3) Para medir la masa del soporte  $U$  se midió la masa del soporte en  $U$  junto con el soporte del rodamiento (naturalmente, están soldados), y se le restó  $m_{sr}$ . Se midió sin la platina de acople.
- (4) Se midió sin el cable.
- (5) Vale el mismo comentario que en (1), con *interior* en lugar de *exterior*.

### 4.1.4. Estimaciones

Primero, a los efectos de estimar las partes internas del rodamiento se mide un rodamiento similar desarmado.

## Capítulo 4. Modelado del Sistema/Péndulo

Parte	Peso	Porcentaje
Peso total	21 g	100 %
Círculo exterior	9 g	42,8 %
Círculo interior	7 g	33,3 %
Municiones	5 g	23,8 %

Parte	Peso Medido	Peso Estimado
Peso total	18 ± 0,05 g	-
Círculo exterior	-	7,7 ± 0,03 g
Círculo interior	-	6 ± 0,02 g
Municiones	-	4,3 ± 0,02 g

Tabla 4.2: Izq: Composición de un rodamiento similar al nuestro. Der: Estimación del peso de las partes de nuestro rodamiento.

Respecto al soporte del rodamiento, se midió el peso del tubo de donde salió dicho soporte, el mismo pesaba 51 g y tenía forma cilíndrica de largo 2,1 cm. El soporte del rodamiento es de forma cilíndrica, del mismo radio pero de largo 1,3 cm. Si asumimos homogeneidad del material entonces el soporte del rodamiento debería pesar  $\frac{1,3}{2,1}51 \text{ g} = 31,6 \text{ g}$ .

### 4.1.5. Construcción del modelo

Para la construcción del modelo seguimos la metodología expuesta en la mayoría de los artículos estudiados. Consideremos las coordenadas vistas en la figura 4.4, junto con las bases vectoriales mostradas.

La base  $\mathcal{B} = \{x, y, z\}$  es solidaria al rígido 1 mientras que la base  $\mathcal{C} = \{x', y', z'\}$  es solidaria al rígido 2. El origen de coordenadas será el denotado por  $O$ , esto es, todo vector posición  $\vec{r}_P$  denota el vector que va de  $O$  a  $P$ . Comenzamos por calcular la energía cinética de los rígidos. Primero el rígido 1, para esto sea  $I_O$  el tensor de inercia en el punto  $O$ . Entonces se tiene

$$K_1 = \frac{1}{2} \vec{w}_1 \cdot I_O(\vec{w}_1)$$

donde  $K_1$  y  $\vec{w}_1$  denotan la energía cinética y la velocidad angular del rígido 1 respectivamente. Para la velocidad angular se tiene  $\vec{w}_1 = \dot{\varphi} \mathbf{z}$ . Si consideramos  $I =_{\mathcal{B}} (I_O)_{\mathcal{B}}$ , la matriz asociada a  $I_O$  en la base  $\mathcal{B}$ , entonces

$$K_1 = \frac{1}{2} (0, 0, \dot{\varphi}) \cdot I \cdot \begin{pmatrix} 0 \\ 0 \\ \dot{\varphi} \end{pmatrix} = \frac{1}{2} I_{33} \dot{\varphi}^2 \quad (4.6)$$

por lo cual solo es relevante el momento de inercia respecto al eje z. De ahora en más llamémosle  $J$  a  $I_{33}$ . Para el rígido 2 utilizaremos el punto  $A$  (ver figura

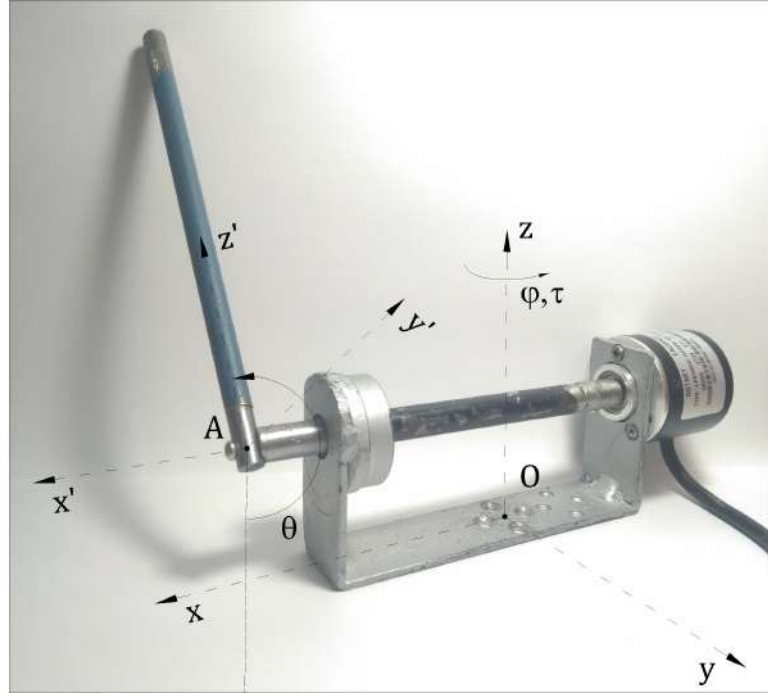


Figura 4.4: Rígidos 1 y 2 acoplados: Coordenadas y bases utilizadas.

4.4). Denotemos por  $\vec{w}_2$  el vector velocidad angular del rígido 2 y por  $G$  el centro de masa. Sea  $I_A$  el tensor de inercia en  $A$ . Para un punto como  $A$  que no tiene velocidad nula, y que no es el centro de masa, se tiene una fórmula más compleja para la energía en términos de la inercia (ver por ejemplo [19], página 98). Para la energía cinética la fórmula queda

$$K_2 = \frac{1}{2}m_2 \|\vec{v}_A\|^2 + m_2 \vec{v}_A \cdot [\vec{w} \times (\vec{r}_G - \vec{r}_A)] + \frac{1}{2}\vec{w}_2 \cdot I_A(\vec{w}_2)$$

Trabajemos término a término. Para el primer término, se tiene que  $\vec{r}_A = h_u \vec{z} + l_{bi} \vec{x}$ . Derivando se obtiene  $\vec{v}_A = l_{bi} \dot{\varphi} \vec{y}$  y por lo tanto

$$\frac{1}{2}m_2 \|\vec{v}_A\|^2 = \frac{1}{2}m_2 l_{bi}^2 \dot{\varphi}^2$$

Para el siguiente término utilizaremos la base  $\mathcal{C}$ . Observemos que  $\vec{w}_2 = \dot{\theta} \vec{x}' + \dot{\varphi} \vec{z} = \dot{\theta} \vec{x}' - \dot{\varphi} \sin \theta \vec{y}' - \dot{\varphi} \cos \theta \vec{z}'$ . Además  $\vec{r}_G - \vec{r}_A = c_x \vec{x}' + c_z \vec{z}'$ , referirse a 4.1.6 para los cálculos de las constantes  $c_x$  y  $c_z$ . Entonces

$$m_2 \vec{v}_A \cdot [\vec{w}_2 \times (\vec{r}_G - \vec{r}_A)] = m_2 \left| \begin{pmatrix} 0 & -l_{bi} \dot{\varphi} \cos \theta & l_{bi} \dot{\varphi} \sin \theta \\ \dot{\theta} & -\dot{\varphi} \sin \theta & -\dot{\varphi} \cos \theta \\ c_x & 0 & c_z \end{pmatrix} \right| = m_2 l_{bi} c_x \dot{\varphi}^2 + m_2 l_{bi} c_z \dot{\varphi} \dot{\theta} \cos \theta$$

Para el tercer término resulta que el mapa  $I_A$  es diagonal en la base  $\mathcal{C}$ , llamémosle  $I_x$ ,  $I_y$  e  $I_z$  a los términos de la diagonal. Entonces

$$\frac{1}{2}\vec{w}_2 \cdot I_A(\vec{w}_2) = \frac{I_x \dot{\theta}^2}{2} + \frac{I_y \dot{\varphi}^2 \sin^2 \theta}{2} + \frac{I_z \dot{\varphi}^2 \cos^2 \theta}{2}$$

## Capítulo 4. Modelado del Sistema/Péndulo

finalmente sumando los términos se llega a

$$K_2 = \frac{1}{2}m_2l_{bi}^2\dot{\varphi}^2 + m_2l_{bi}c_x\dot{\varphi}^2 + m_2l_{bi}c_z\dot{\varphi}\dot{\theta}\cos\theta + \frac{I_x\dot{\theta}^2}{2} + \frac{I_y\dot{\varphi}^2\sin^2\theta}{2} + \frac{I_z\dot{\varphi}^2\cos^2\theta}{2} \quad (4.7)$$

y sumando 4.6 y 4.7 obtenemos la expresión para la energía cinética.

$$K = \left( \frac{1}{2}J + \frac{1}{2}m_2l_{bi}^2 + m_2l_{bi}c_x + \frac{I_y}{2}\sin^2\theta + \frac{I_z}{2}\cos^2\theta \right) \dot{\varphi}^2 + \frac{I_x}{2}\dot{\theta}^2 + m_2l_{bi}c_z\dot{\theta}\dot{\varphi}\cos\theta \quad (4.8)$$

Continuemos con la energía potencial, si observamos el rígido 1, el centro de masa del mismo nunca cambia su altura por lo cual podemos suponer que su energía potencial es 0. Para el rígido 2, tomando de referencia el brazo del péndulo se tiene

$$V = m_2gc_z\sin(\theta - \pi/2) = -m_2gc_z\cos\theta$$

entonces llamándole  $\mathbf{q} = \begin{pmatrix} \varphi \\ \theta \end{pmatrix}$ , y utilizando que para figuras planas  $I_y = I_x + I_z$  se llega al lagrangiano  $\mathcal{L} : \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}$  dado por

$$\begin{aligned} \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) = K(\mathbf{q}, \dot{\mathbf{q}}) - V(\mathbf{q}) &= \left( \frac{1}{2}J + \frac{1}{2}m_2l_{bi}^2 + m_2l_{bi}c_x + \frac{I_z}{2} + \frac{I_x}{2}\sin^2\theta \right) \dot{\varphi}^2 \\ &+ \frac{I_x}{2}\dot{\theta}^2 + m_2l_{bi}c_z\dot{\theta}\dot{\varphi}\cos\theta + m_2gc_z\cos\theta \end{aligned} \quad (4.9)$$

Sea  $\tau$  el par que ejerce el motor, como se muestra en la figura 4.4. Consideremos un par de fricción  $\tau_{fu}$  para el soporte en  $U$  y  $\tau_{fp}$  para el péndulo. Cuando las fuerzas presentes derivan de un potencial la solución al problema es un punto estacionario de un funcional de acción dado por  $\mathcal{L}$ . Los puntos críticos vienen dados por el variacional de dicho funcional. En  $\mathbb{R}^n$  dichas cuentas terminan dando las ecuaciones de Euler Lagrange. Si hay fuerzas no conservativas las ecuaciones de Euler Lagrange valen con mínimas modificaciones. En este caso vienen dadas por:

$$\begin{aligned} \frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{\varphi}} - \frac{\partial \mathcal{L}}{\partial \varphi} &= \tau + \tau_{fu} \\ \frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{\theta}} - \frac{\partial \mathcal{L}}{\partial \theta} &= \tau_{fp} \end{aligned}$$

Luego de realizar los cálculos correspondientes se llega a

$$\begin{aligned} \ddot{\varphi} (J + 2m_2l_{bi}c_x + m_2l_{bi}^2 + I_z + I_x\sin^2\theta) \\ + 2\dot{\varphi}\dot{\theta}I_x\cos\theta\sin\theta + m_2l_{bi}c_z\ddot{\theta}\cos\theta - m_2l_{bi}c_z\dot{\theta}^2\sin\theta &= \tau + \tau_{fu} \end{aligned} \quad (4.10)$$

$$I_x\ddot{\theta} + m_2l_{bi}c_z\ddot{\varphi}\cos\theta - \dot{\varphi}^2I_x\cos\theta\sin\theta + m_2gc_z\sin\theta = \tau_{fp} \quad (4.11)$$

que además se puede escribir como

$$M(\theta)\ddot{\mathbf{q}} + D(\theta, \dot{\varphi})\dot{\mathbf{q}} + F(\theta) = U \quad (4.12)$$

donde

$$\begin{aligned} M(\theta) &= \begin{pmatrix} J + m_2 l_{bi}^2 + 2m_2 l_{bi} c_x + I_z + I_x \sin^2 \theta & m_2 l_{bi} c_z \cos \theta \\ m_2 l_{bi} c_z \cos \theta & I_x \end{pmatrix} \\ D(\theta, \dot{\varphi}) &= \begin{pmatrix} 2I_x \dot{\theta} \sin \theta \cos \theta & -m_2 l_{bi} c_z \sin \theta \dot{\theta} \\ -\dot{\varphi} I_x \cos \theta \sin \theta & 0 \end{pmatrix} \\ F(\theta) &= \begin{pmatrix} 0 \\ m_2 g c_z \sin \theta \end{pmatrix} \\ U &= \begin{pmatrix} \tau + \tau_{fu} \\ \tau_{fp} \end{pmatrix} \end{aligned}$$

#### 4.1.6. Momentos de inercia y centro de masa

En esta sección presentamos los valores de los momentos de inercia utilizados,  $J$ ,  $I_x$ ,  $I_y$  e  $I_z$ . Además se presenta el valor de las constantes del centro de masa  $c_x$  y  $c_z$ . Para ver los cálculos referirse al apéndice B. Comenzamos con el momento de inercia del soporte en  $U$ ,

$$J = 0,0185 \pm 0,00096 \text{ kg} \cdot \text{m}^2$$

Para calcular la incertidumbre de  $J$  se utilizaron variables aleatorias uniformes y se relevó la distribución de  $J$ . Para profundizar en esto ver el apéndice C.

Para el cálculo de  $I_x$ ,  $I_y$  e  $I_z$  observemos que al ser el rígido 2 *plano* se cumple  $I_x + I_z = I_y$ . Recordemos que en la base  $\mathcal{C}$  el mapa  $I_A$  diagonaliza, por lo cual lo único a calcular es  $I_x$  e  $I_z$ .

$$\begin{aligned} I_x &= 4,39 \times 10^{-4} \pm 2,8 \times 10^{-5} \text{ kg} \cdot \text{m}^2 \\ I_z &= 1,88 \times 10^{-4} \pm 4,5 \times 10^{-5} \text{ kg} \cdot \text{m}^2 \\ I_y &= I_z + I_x \end{aligned}$$

Por último, recordamos que  $c_x$  y  $c_z$  se corresponden a las coordenadas del vector  $\vec{r}_G - \vec{r}_A$  en la base  $\mathcal{C}$ . Es decir  $\vec{r}_G - \vec{r}_A = c_x \mathbf{x}' + c_z \mathbf{z}'$ . Se tiene

$$\begin{aligned} c_x &= -2,3 \pm 0,042 \text{ cm} \\ c_z &= 4,4 \pm 0,016 \text{ cm} \end{aligned}$$

### 4.1.7. Comparando con la literatura

De la literatura revisada en general todos siguen el camino de derivar las ecuaciones del movimiento utilizando las ecuaciones de Euler-Lagrange, es un camino más sistemático llevando a menos errores. En la determinación de un lagrangiano  $\mathcal{L}$  para el sistema todos utilizan  $K - V$  pero no todos derivan de la misma forma la energía cinética  $K$ . Algunos van por el cálculo directo [2] mientras que la mayoría descomponen el sistema en dos rígidos y utilizan la inercia de dichos rígidos.

Como señala [7], a pesar de la gran atención que el péndulo ha recibido, varios de los que siguen el camino de utilizar la inercia han cometido errores. El autor muestra que varios artículos de la literatura han olvidado considerar el carácter matricial del tensor de inercia, tomando el mismo solamente escalar. Incluso el autor señala libros con dicho error, entre ellos [26]. De los artículos aquí revisados el autor señala también [27]. Otros constatados por nosotros son [4, 20, 22]. Afortunadamente el error cometido por estos artículos no lleva a ecuaciones del movimiento muy distintas, de hecho, cualitativamente tienen la misma forma general o pequeñas diferencias.

Respecto a las ecuaciones de nuestro péndulo, por tratarse de una versión diferente a la versión estándar no se puede pretender que las ecuaciones sean idénticas, para comenzar simplemente por tener más parámetros. Nuestro péndulo se compone de un rígido plano (péndulo+brazo) y del soporte en U, cuando la versión estándar simplemente de dos rígidos prácticamente unidimensionales. A pesar de estas diferencias las ecuaciones del movimiento derivadas tienen la misma forma que las ecuaciones del Furuta estándar. Para ver esto lo más fácil es comparar los lagrangianos utilizados. Sea  $\alpha = I_x$ ,  $\beta = m_2 l_{bi} c_z$ ,  $\gamma = J + m_2 l_{bi}^2 + 2m_2 l_{bi} c_x + I_z$  y  $D = -m_2 g c_z$  entonces el lagrangiano derivado en esta sección es

$$\mathcal{L} = \frac{1}{2}\alpha\dot{\theta}^2 + \beta\dot{\theta}\dot{\varphi}\cos\theta + \frac{1}{2}(\gamma + \alpha\sin^2\theta)\dot{\varphi}^2 - D\cos\theta$$

que coincide con el presentado en [6] luego de hacer el cambio de variable  $q_1 = \pi - \theta$  y  $q_2 = \varphi$ . También se puede ver que coincide con [7] y con [2].

El trabajo [3] presenta un péndulo bastante parecido al nuestro pero los autores han decidido no mostrar las cuentas en su completitud por lo cual se dificulta el seguimiento. Lo que se puede afirmar es que han tenido en cuenta correctamente el mapa lineal de la inercia.



## 4.2. Motor

En la figura 4.5 se muestra el modelo utilizado para representar el motor DC sin carga. Se eligió el modelo debido a que la gran mayoría de referencias utilizaban el mismo.

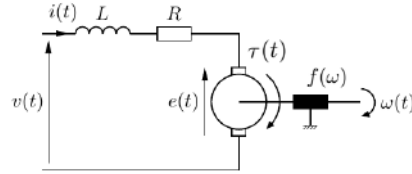


Figura 4.5: Modelo del motor DC sin carga.

La entrada  $v(t)$  es la señal de control, es el voltaje elegido desde el microcontrolador y corresponde al valor *medio* del PWM entregado por el driver. La señal  $i(t)$  se corresponde con la corriente *media* por el motor. Luego  $\tau$  es el par en el eje del motor,  $J$  es la inercia del eje y  $w$  es la velocidad, que se corresponde con  $\dot{\varphi}$  de la sección anterior. La señal  $e(t)$  es el voltaje inducido en el bobinado del motor. Se asume  $\tau = ki$  y  $e = kw$ . Finalmente,  $f(w)$  es la fricción sobre el eje. Aquí la fricción se ha decidido modelar como una función no lineal en la velocidad. Las ecuaciones del modelo anterior son

$$\begin{cases} v - Ri - L \frac{di}{dt} - kw = 0 \\ J\dot{w} = ki - f(w) \end{cases}$$

Ahora en nuestro caso el motor se encuentra cargado y no con una rígido sino con el péndulo de Furuta. Entonces las ecuaciones pasan a ser

$$\begin{cases} v - Ri - L \frac{di}{dt} - kw = 0 \\ \tau = ki \end{cases}$$

y la dinámica de  $w$  queda dado por  $\dot{\varphi}$  en las ecuaciones 4.10 y 4.11 con  $\tau_{fu}(w) = f(w)$  y torque dado por  $ki$ .

### 4.2.1. Limitación del modelo

Una limitación del modelo surge de que el motor se maneja mediante un driver PWM pero el modelo se maneja con una fuente ideal  $v(t)$  de valor igual al valor medio del PWM. Para comenzar esto implica que solo podremos predecir los valores medios de las magnitudes  $v(t)$  e  $i(t)$ .

Una consecuencia más importante se sigue de las observaciones hechas en 3.2.2. Observe que en el sistema real al colocar  $v = 0$  el driver en realidad intenta imponer  $i = 0$  y la tensión no queda determinada. En el modelo anterior del motor este efecto no está incluido, al colocar  $v = 0$  la corriente queda determinada por la malla.

## Capítulo 4. Modelado del Sistema/Péndulo

Se decide no modelar este efecto ya que el modelo para el mismo debería ser bastante más complejo.

### 4.3. Resumen del modelado

En esta sección se resumen las ecuaciones del sistema, juntando las ecuaciones del motor acoplado con las ecuaciones del péndulo.

$$\begin{cases} \ddot{\varphi} (J + 2m_2l_{bi}c_x + m_2l_{bi}^2 + I_x + I_x \sin^2 \theta) \\ + 2\dot{\varphi}\dot{\theta}I_x \cos \theta \sin \theta + m_2l_{bi}c_z\ddot{\theta} \cos \theta - m_2l_{bi}c_z\dot{\theta}^2 \sin \theta = \tau + \tau_{fu}(\dot{\varphi}) \\ I_x\ddot{\theta} + m_2l_{bi}c_z\ddot{\varphi} \cos \theta - \dot{\varphi}^2 I_x \cos \theta \sin \theta + m_2gc_z \sin \theta = \tau_{fp}(\dot{\theta}) \\ \tau = ki \\ v - Ri - L\frac{di}{dt} - k\dot{\varphi} = 0 \end{cases} \quad (4.13)$$

#### 4.3.1. Ecuaciones en variables de estado

Para llevar el sistema a variables de estados es muy conveniente utilizar 4.12. Para eso primero veamos que  $M$  es invertible para todo  $\theta$ . Esto es un resultado general y se puede ver del hecho de que la matriz asociada a una cierta métrica expresada en coordenadas  $(\theta, \varphi, \dot{\theta}, \dot{\varphi})$ . Un camino más simple, que en el fondo explota esto, es observar que

$$\dot{\mathbf{q}}^T M \dot{\mathbf{q}} = K \geq 0$$

donde  $\mathbf{q} = \begin{pmatrix} \varphi \\ \theta \end{pmatrix}$  y  $K$  es la energía cinética del sistema, que fue calculada en 4.8. Como la energía cinética es positiva excepto para velocidad nula, entonces el producto anterior está diciendo que la matriz  $M$  es definida positiva para todo  $\theta$ . Como consecuencia se sigue que  $M$  es invertible. Entonces uno puede operar en (4.12) con  $M^{-1}$  sin sentir culpa. Se obtiene entonces el modelo en variables de estado

$$\ddot{\mathbf{q}} = M(\theta)^{-1} [U - D(\theta, \dot{\varphi})\dot{\mathbf{q}} - F(\theta)] \quad (4.14)$$

utilizando la fórmula para la inversa de una matriz de 2x2 se sigue que

$$\ddot{\varphi} = \frac{1}{d(\theta)} \begin{pmatrix} I_x(\tau_{fu} - 2I_x\dot{\theta}\dot{\varphi} \sin \theta \cos \theta + m_2l_{bi}c_z \sin \theta \dot{\theta}^2) \\ -m_2l_{bi}c_z \cos \theta(\tau_{fp} + \dot{\varphi}^2 I_x \cos \theta \sin \theta - m_2gc_z \sin \theta) + I_x\tau \end{pmatrix}$$

y

$$\ddot{\theta} = \frac{1}{d(\theta)} \begin{pmatrix} -m_2 l_{bi} c_z \cos \theta (\tau_{fu} - 2I_x \dot{\theta} \dot{\varphi} \sin \theta \cos \theta + m_2 l_{bi} c_z \sin \theta \dot{\theta}^2) \\ +(J + m_2 l_{bi}^2 + 2m_2 l_{bi} c_x + I_z + I_x \sin^2 \theta) (\tau_{fp} + \dot{\varphi}^2 I_x \cos \theta \sin \theta - m_2 g c_z \sin \theta) \\ -m_2 l_{bi} c_z \cos \theta \tau \end{pmatrix}$$

donde  $d(\theta) = \det M(\theta) = (J + m_2 l_{bi}^2 + 2m_2 l_{bi} c_x + I_z + I_x \sin^2 \theta) I_x - m_2^2 l_{bi}^2 c_z^2 \cos^2 \theta$ . La ecuación 4.14 muestra que el sistema, a pesar de ser no lineal, es afín con la entrada  $\tau$ , es decir, toma la forma estándar  $\dot{x} = f(x) + g(x)\tau$ .

### 4.3.2. Linealización Jacobiana

Para contar con las herramientas que provee la teoría del control lineal, realizamos la linealización del sistema del péndulo de Furuta, sin motor, entorno al punto de equilibrio dado por  $x_{eq} = (\theta, \varphi, \dot{\theta}, \dot{\varphi}) = (\pi, 0, 0, 0)$  con entrada  $\tau = 0$ . Si le llamamos  $d = (J + 2m_2 l_{bi} c_x + m_2 l_{bi}^2 + I_z) I_x - m_2^2 l_{bi}^2 c_z^2$  y aplicando diferenciación implícita sobre las primeras 2 ecuaciones de 4.13 se tiene

$$\begin{aligned} \frac{\partial \ddot{\varphi}}{\partial \theta} &= \frac{m_2^2 c_z^2 l_{bi} g}{d} & \frac{\partial \ddot{\theta}}{\partial \theta} &= \frac{m_2 g c_z (J + 2m_2 l_{bi} c_x + m_2 l_{bi}^2 + I_z)}{d} & \frac{\partial \ddot{\varphi}}{\partial \dot{\varphi}} &= -\frac{I_x b_u}{d} & \frac{\partial \ddot{\theta}}{\partial \dot{\varphi}} &= -\frac{m_2 l_{bi} c_z b_u}{d} \\ \frac{\partial \ddot{\varphi}}{\partial \dot{\theta}} &= -\frac{m_2 l_{bi} c_z b_p}{d} & \frac{\partial \ddot{\theta}}{\partial \dot{\theta}} &= -\frac{(J + 2m_2 l_{bi} c_x + m_2 l_{bi}^2 + I_z) b_p}{d} & \frac{\partial \ddot{\varphi}}{\partial u} &= \frac{I_x}{d} & \frac{\partial \ddot{\theta}}{\partial u} &= \frac{m_2 l_{bi} c_z}{d} \end{aligned}$$

donde las derivadas están evaluadas en  $x_{eq}$ . Para obtener dicha linealización se consideró solamente el carácter viscoso de la fricción, que se asume lineal con coeficiente  $b_p$  para el péndulo y  $b_u$  para el soporte en  $U$ . Considerar el verdadero aspecto de la fricción, como tener en cuenta la fricción estática, no es una opción ya que la misma no es diferenciable, además de que el modelo de fricción no es preciso en torno a velocidades nulas. En resumen el sistema linealizado esta dado por

$$\dot{x} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{m_2 g c_z (J + 2m_2 l_{bi} c_x + m_2 l_{bi}^2 + I_z)}{d} & 0 & -\frac{(J + 2m_2 l_{bi} c_x + m_2 l_{bi}^2 + I_z) b_p}{d} & -\frac{m_2 l_{bi} c_z b_u}{d} \\ \frac{m_2^2 c_z^2 l_{bi} g}{d} & 0 & -\frac{m_2 l_{bi} c_z b_p}{d} & -\frac{I_x b_u}{d} \end{pmatrix} x + \begin{pmatrix} 0 \\ 0 \\ \frac{m_2 l_{bi} c_z}{d} \\ \frac{I_x}{d} \end{pmatrix} \tau$$

con  $x = (\theta - \pi, \varphi, \dot{\theta}, \dot{\varphi})^T$ , obteniendo un modelo de la forma  $\dot{x} = Ax + B\tau$ .

Esta página ha sido intencionalmente dejada en blanco.

# Capítulo 5

## Identificación del Sistema

Aquí se describe el método por el cual se ensayan y obtienen los parámetros del modelo (4.13). El capítulo se divide en dos secciones. En la sección 5.1 se identifican todos los parámetros y funciones que aparecen en las ecuaciones (4.10,4.11), a excepción del parámetro  $J$  que es, en cierto sentido, un parámetro que pertenece tanto a las ecuaciones del péndulo como a las ecuaciones del motor acoplado con el soporte en U. Este parámetro se identifica en la sección 5.2, junto con los parámetros propios del motor.

La estrategia para la identificación de los distintos parámetros ha sido desacoplar el sistema completo en distintas partes, reduciendo las ecuaciones del modelo en ecuaciones más sencillas pero con los mismos parámetros, permitiendo realizar la identificación del sistema identificando los parámetros de sistemas más simples.

Finalmente, en la sección 5.3 se muestran los resultados del modelado e identificación llevados a cabo en el capítulo 4 y a lo largo de este capítulo, comparando la respuesta del péndulo real con la predicción del modelo.

### 5.1. Péndulo de Furuta

Para realizar la identificación de los parámetros involucrados en el modelo (4.10, 4.11), se trabaja separadamente con el péndulo (rígido 2) libre con el soporte en U fijo por un lado, y el soporte en U acoplado al motor, pero sin el péndulo, por otro. Esto reduce las ecuaciones del movimiento en ecuaciones bastante más simples y mantiene los parámetros involucrados. Es así que identificando los parámetros de dos sistemas más simples se obtienen los parámetros del modelo completo.

#### 5.1.1. Péndulo libre, soporte en U fijo

Si observamos la ecuación del movimiento del péndulo (4.11) al colocar  $\varphi \equiv \varphi_0$ , se llega a

## Capítulo 5. Identificación del Sistema

$$I_x \ddot{\theta} + m_2 g c_z \sin \theta = \tau_{fp}(\dot{\theta}) \quad (5.1)$$

que corresponde a la ecuación de un péndulo simple con rozamiento, lo cual era lo esperado.

Para realizar la identificación de parámetros se probaron tres métodos distintos.

### Método 1, Sistema lineal de segundo orden:

Si consideramos que solo la fricción viscosa es apreciable entonces un modelo para la fricción es  $\tau_{fp}(\dot{\theta}) = -b_2 \dot{\theta}$ . Si además estudiamos el péndulo realizando pequeñas oscilaciones ( $|\theta| < 15^\circ$ ) entonces el sistema (5.1) se puede aproximar por

$$I_x \ddot{\theta} + b_2 \dot{\theta} + m_2 g c_z \theta = 0$$

al ser un sistema lineal de segundo orden la identificación de sus parámetros es bastante simple. Si se releva el sistema partiendo de cierta condición inicial entonces se observará una senoide cuya amplitud cae de manera exponencial. Si reescribimos la ecuación del movimiento en la forma canónica  $\ddot{\theta} + 2\zeta w_n \dot{\theta} + w_n^2 \theta = 0$  entonces se cumple:

$$w_d = \frac{2\pi}{T} \quad \zeta = \frac{\ln \frac{x_i}{x_{i+1}}}{\sqrt{4\pi^2 + \left(\frac{x_i}{x_{i+1}}\right)^2}} \quad w_n = \frac{w_d}{\sqrt{1 - \zeta^2}}$$

donde  $T$  es el período, y  $x_i, x_{i+1}$  son dos amplitudes consecutivas. Para calcular los parámetros se relevaron 5 respuestas del péndulo partiendo del reposo con  $\theta_0 \approx 15^\circ$ . Si tomamos los parámetros resultantes de esta identificación, simulamos y comparamos los resultados contra un relevo experimental se obtiene los resultados de la figura 5.1.

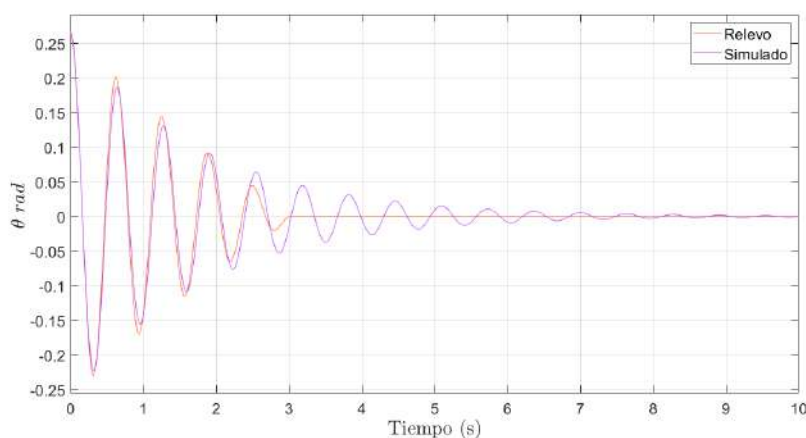


Figura 5.1: Datos vs modelo obtenido.

## 5.1. Péndulo de Furuta

por lo cual los resultados no son aceptables. Si uno observa con atención el problema es que el modelo no puede imitar la caída en la amplitud del relevo. De hecho la figura 5.2 muestra que este decaimiento es lineal, indicio de la presencia de rozamiento estático. En base a estos resultados se concluye que no se puede despreciar la fricción de Coulumb y por lo tanto se descarta este método.

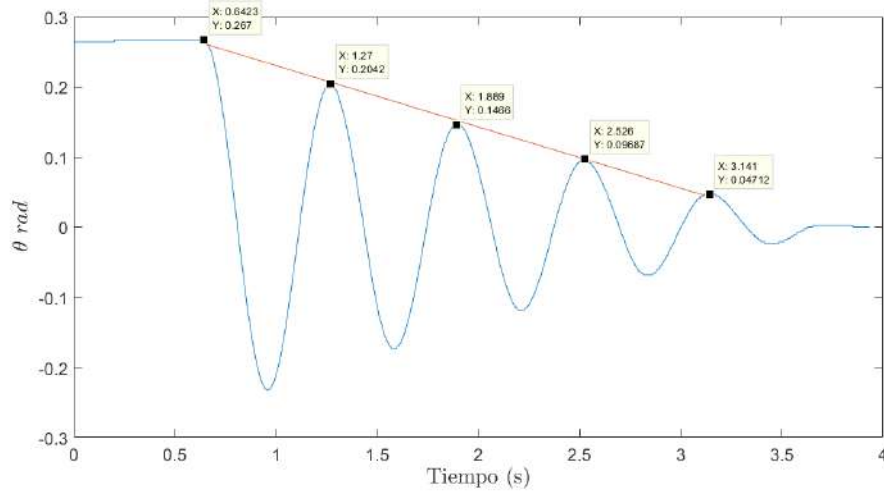


Figura 5.2: Evidencia de fricción de Coulomb.

### Método 2, Mínimos cuadrados lineales:

Una manera simple de agregar la fricción de Coulomb es considerar  $\tau_{fp} = -b_2\dot{\theta} - C_{ep}\text{sign}(\dot{\theta})$ . Para no tener problemas numéricos se utiliza una sigmoide en lugar de la función signo, aquí utilizamos  $\left(1 - \frac{2}{1+e^{200\dot{\theta}}}\right) = \tanh(100\dot{\theta})$

Sea  $\{t_1, t_2, \dots, t_n\}$  los instantes de muestreo y sea

$$\ddot{\theta} = \begin{pmatrix} \ddot{\theta}(t_1) \\ \ddot{\theta}(t_2) \\ \vdots \\ \ddot{\theta}(t_n) \end{pmatrix} \quad \dot{\theta} = \begin{pmatrix} \dot{\theta}(t_1) \\ \dot{\theta}(t_2) \\ \vdots \\ \dot{\theta}(t_n) \end{pmatrix} \quad \tau_{fu} = \begin{pmatrix} \tau_{fu}(t_1) \\ \tau_{fu}(t_2) \\ \vdots \\ \tau_{fu}(t_n) \end{pmatrix} \quad \mathbf{V} = \begin{pmatrix} \sin \theta(t_1) \\ \sin \theta(t_2) \\ \vdots \\ \sin \theta(t_n) \end{pmatrix}$$

entonces la ecuación (5.1) dice que  $\ddot{\theta} \in \text{span}\{\dot{\theta}, \tau, \mathbf{V}\}$  ahora los parámetros que queremos identificar son los coeficientes de la combinación lineal que genera  $\ddot{\theta}$ , entonces simplemente hay que resolver el sistema lineal de ecuaciones:

$$\ddot{\theta} = x_1\dot{\theta} + x_2\tau_{fu} + x_3\mathbf{V}$$

en la práctica el sistema es incompatible y por lo tanto optamos por minimizar la norma 2 de la diferencia, es decir, resolver quiénes son los coeficientes que

## Capítulo 5. Identificación del Sistema

minimizan el error cuadrático, más conocido como el problema de mínimos cuadrados lineales. El que sea lineal tiene la ventaja de que la solución existe, es única y tiene una fórmula explícita muy simple. Además  $m_2$ ,  $c_z$ ,  $g$  y  $I_x$  son conocidos, por lo tanto conociendo  $x_1$ ,  $x_2$  y  $x_3$  se puede conocer  $C_{ep}$ ,  $b_2$  y verificar el valor teórico de  $I_x$ . Aprovechando que el método trabaja con un sistema no lineal se utilizaron relevos del péndulo partiendo de  $\theta = 90^\circ$ , se tomaron 5 de estos relevos. En la figura 5.3 se observa un resultado típico, en color rosado se observa los datos relevados mientras que en naranja se observa el resultado del modelo obtenido.

Los resultados muestran que el modelo elegido replica bien cualitativamente el comportamiento de los relevos, por lo cual parece el modelo correcto. Sin embargo, cuantitativamente, es claro que la solución obtenida no es la óptima. Creemos que el error proviene del cálculo de  $\ddot{\theta}$ . Como ya vimos en la sección 3.3 el cálculo de  $\dot{\theta}$  no fue trivial debido a la cuantización y el ruido. Para  $\ddot{\theta}$  los problemas se agravan y el resultado no es tan bueno.

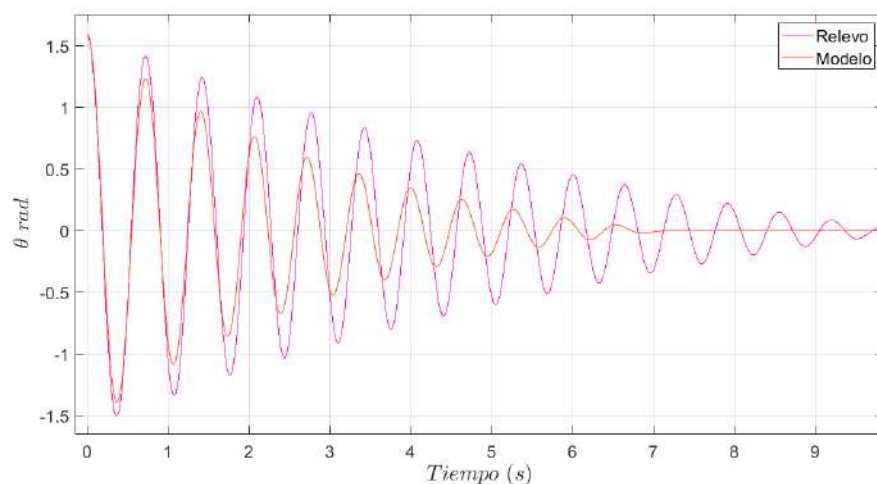


Figura 5.3: Resultado típico del método 2, en rosado se observa el relevo de datos y en naranja el modelo con los parámetros resultantes del método de ajuste.

Uno puede intentar mejorar la solución obtenida por el método iterando a mano, sin embargo, es deseable una solución más automática. Dicha solución es el método utilizado y es el que se presenta a continuación.

### Método 3

La idea de este método es utilizar la idea anterior pero planteada de forma distinta. El objetivo es hallar  $x_1$ ,  $x_2$  y  $x_3$  que minimicen  $\|\theta - \hat{\theta}(x_1, x_2, x_3)\|$  donde  $\theta$  es la posición medida por el encoder y  $\hat{\theta}(x_1, x_2, x_3)$  es la solución numérica de (5.1) con parámetros  $x_1 = b_2/I_x$ ,  $x_2 = C_{ep}/I_x$  y  $x_3 = m_2gc_z/I_x$ . Para calcular  $\hat{\theta}$  se utiliza la función *Ode45* de matlab. Luego para minimizar la función objetivo se



## 5.1. Péndulo de Furuta

utilizó *fminsearch* tomando de semilla inicial los parámetros resultantes del método 2.

Para cada relevo tomado se ajustaron los mejores parámetros, en la figura 5.4 se muestra un relevo particular en azul y en naranja el modelo con los parámetros que mejor ajusta este relevo particular. Se observan resultados satisfactorios tanto cualitativamente como cuantitativamente, la norma 2 del error vale 0,1144 mientras que la norma infinito del error vale 0,1023 que es menos de  $6^\circ$ .

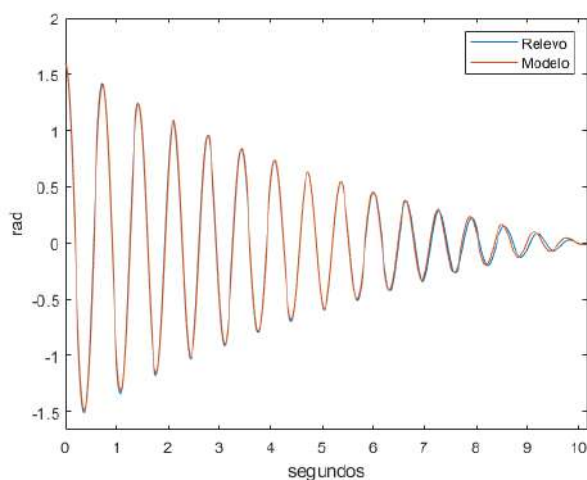


Figura 5.4: En azul se observa un relevo de datos mientras que en naranja se observa el modelo con los parámetros que mejor ajustan los datos según el método 3.

En la tabla 5.1 se observan los resultados del ajuste para cada relevo.

Lo primero a observar es el valor de  $I_x$ . Recordamos que el valor teórico de  $I_x$  es de  $4,39 \times 10^{-4} \text{ kg m}^2$  lo que está muy cerca de los resultados, corroborando en parte el método. Respecto al resto de resultados, observamos que tienen una varianza relativamente grande. Para obtener valores medios solo se utilizan los relevos 3, 4 y 5, ya que los primeros dos relevos fueron descartados y los últimos seguramente presenten más error por trabajar con señales más pequeñas. Los resultados son

$$I_x = 4,18 \times 10^{-4} \text{ kg m}^2 \quad b_2 = 6,64 \times 10^{-5} \text{ Nms} \quad C_{ep} = 6,51 \times 10^{-4} \text{ Nms}$$

Si simulamos el modelo obtenido y comparamos contra los distintos relevos podemos apreciar que si bien los resultados de fricción cuentan con una gran varianza, esta variación no provoca un gran cambio en la dinámica resultando en una buena aproximación del modelo con parámetros medios.

## Capítulo 5. Identificación del Sistema

Relevo	$I_x(kgm^2)$	$b_2(Nms)$	$C_{ep}(Nm)$	$\ e\ _2$	$\ e\ _\infty$
1*	$4,29 \times 10^{-4}$	$5,88 \times 10^{-5}$	$1,00 \times 10^{-3}$	0,16	0,11
2*	$4,19 \times 10^{-4}$	$4,55 \times 10^{-5}$	$8,79 \times 10^{-4}$	0,23	0,16
3	$4,17 \times 10^{-4}$	$5,19 \times 10^{-5}$	$7,40 \times 10^{-4}$	0,23	0,14
4	$4,29 \times 10^{-4}$	$7,37 \times 10^{-5}$	$6,25 \times 10^{-4}$	0,35	0,50
5	$4,19 \times 10^{-4}$	$7,35 \times 10^{-5}$	$5,87 \times 10^{-4}$	0,11	0,10
6	$4,20 \times 10^{-4}$	$1,33 \times 10^{-4}$	$4,27 \times 10^{-4}$	0,0076	0,012
7	$4,18 \times 10^{-4}$	$1,49 \times 10^{-4}$	$4,20 \times 10^{-4}$	0,0070	0,011
8	$4,26 \times 10^{-4}$	$1,75 \times 10^{-4}$	$4,09 \times 10^{-4}$	0,015	0,031
9	$4,32 \times 10^{-4}$	$1,91 \times 10^{-4}$	$3,89 \times 10^{-4}$	0,023	0,047
10	$4,12 \times 10^{-4}$	$1,29 \times 10^{-4}$	$4,25 \times 10^{-4}$	0,012	0,023

Tabla 5.1: Resultados del ajuste para cada relevo. Los relevos del 1 al 5 corresponden al péndulo partiendo de  $\theta \approx 90^\circ$ , los relevos del 6 al 10 corresponden al péndulo partiendo de  $\theta \approx 15^\circ$ . Los relevos 1 y 2 se descartan pues al inspeccionarlos se encontraron datos sospechosos de mala práctica.

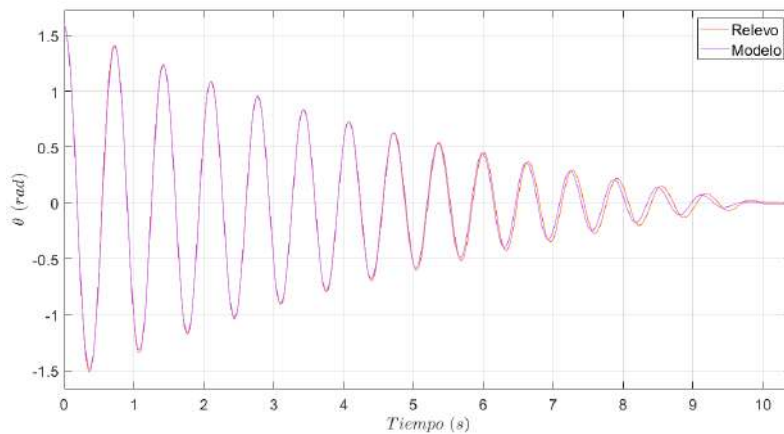


Figura 5.5: Se compara el relevo 5 contra el modelo obtenido.

## 5.1. Péndulo de Furuta

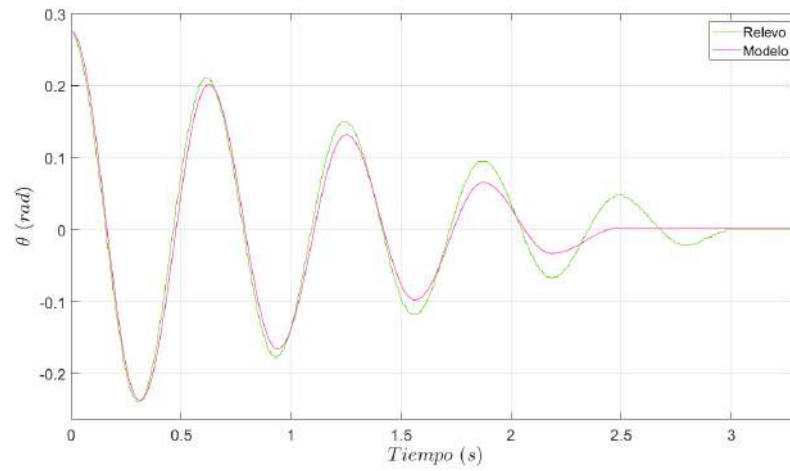


Figura 5.6: Se compara el relevo 6 contra el modelo obtenido.

### 5.1.2. Soporte en U

Si consideramos solamente el motor acoplado con el soporte en U, entonces las ecuaciones que describen la dinámica son

$$\begin{cases} v - Ri - L\frac{di}{dt} - kw = 0 \\ J\dot{w} = ki - \tau_{fu}(w) \end{cases}$$

entonces en régimen  $ki = \tau_{fu}(w)$  por lo cual midiendo la corriente tenemos una medida del rozamiento. De esta forma midiendo la corriente en régimen y luego multiplicando por  $k$  se releva la fricción  $\tau_{fu}(w)$ .

Para no depender de  $k$ , aquí relevamos  $\tau_{fu}/k(w)$ , midiendo directamente la corriente  $i$  por el motor. Vale aclarar que aquí medimos la corriente media por el motor.

En lugar de fijar la tensión en el motor y relevar la velocidad y la corriente, utilizamos un controlador PI que no permite fijar la velocidad, como se muestra en la figura 5.7, donde  $w_s$  es el setpoint.

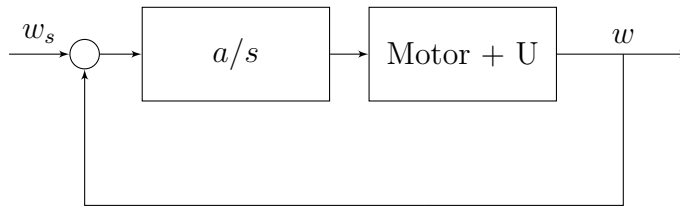


Figura 5.7: Controlador PI para fijar la velocidad.

La primera razón para hacer esto es la comodidad ya que nos interesa relevar la función  $w \mapsto i$  en régimen, por lo cual fijar  $w$  es muy cómodo.

La segunda razón es que es muy cómodo para relevar la corriente de arranque, pues si se coloca  $a$  pequeño, y  $w_s$  una velocidad que se corresponda a apenas salir de la zona muerta del motor, entonces al prender el motor no arrancará, y el integrador empezará a aumentar el voltaje en bornes del motor lentamente hasta que logré arrancar. La idea es que sea cuasiestático para considerar que vale el régimen. Qué tan lento depende del valor de  $a$ , por eso a la hora de medir esta corriente se tomó  $a$  pequeño pero a la hora de medir corrientes fuera de la zona muerta utilizamos un valor de  $a$  tal que se llegue rápido al régimen.

Utilizando este controlador se obtiene el relevo mostrado en la figura 5.8. En dicha figura se han superpuesto dos relevos. Teniendo en cuenta que  $\tau_{fu} = ki$ , la figura muestra la forma de la fricción. Se observa como la fricción de arranque es mayor que la fricción para velocidades pequeñas. También se observa como la fricción en movimiento tiene una cierta tendencia a ser una recta. Para elegir un modelo de la fricción se revisó brevemente la literatura.

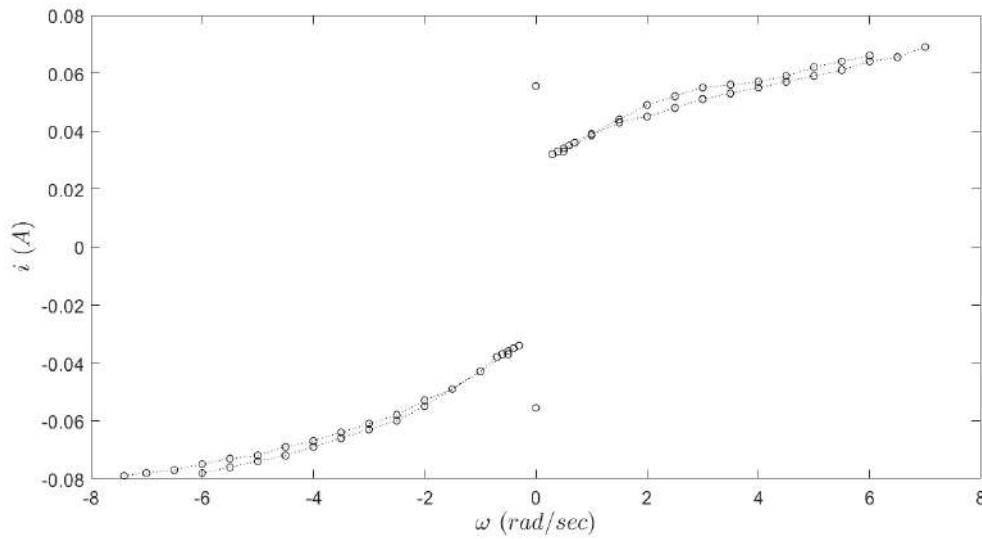


Figura 5.8: Corriente vs velocidad en régimen

Como se comenta en [28], es imposible trabajar con un modelo de la fricción que se base en la interacción microscópica entre las superficies en contacto. Dichos modelos son adecuados para entender el comportamiento de la fricción pero computacionalmente son muy costosos y complicados. Para aplicaciones en ingeniería usualmente es necesario construir leyes empíricas de la fricción, leyes basadas en observaciones, leyes fenomenológicas. Este tipo de leyes dependen de cantidades macroscópicas como el desplazamiento, la velocidad, temperatura, variables de estado, etc.

Dentro de la literatura revisada aparecen principalmente dos tipos de modelos, aquellos que son sin memoria y aquellos que tienen memoria. Los primeros refieren a expresar la fricción como una función de las variables de estado mientras que en los segundos la dependencia de la fricción entre dos superficies con la velocidad relativa entre ellas se modela con ecuaciones diferenciales dándole un estado interno al modelo. Estos últimos modelos son más complicados pero con ellos se puede lograr modelar incluso la histéresis [29]. Dentro de los modelos más mencionados están el modelo de LuGre [24, 29–31], el de Dahl [29, 31], el Maxwell-Slip y el GMS [29, 30].

Dentro de los modelos sin memoria, el más elemental es el modelo de Fricción de Coulomb. La fricción de Coulomb no depende en la magnitud de la velocidad, solamente en el sentido,  $F = -F_c \text{sign}(v)$ . Típicamente la fricción de Coulomb, también llamada *Sliding kinetic friction* o *dynamic friction* es menor a la fricción estática [32, 33]. Otros modelos son de fricción estática. La fricción estática *static friction* o *stiction* existe cuando se está en reposo y previene el movimiento inicial. La fricción estática depende de la fuerza neto o torque neto externo y no de la

## Capítulo 5. Identificación del Sistema

velocidad [33]. Un modelo mencionado en [32] es

$$F = \begin{cases} -F_e & \text{si } v = 0 \wedge |F_e| < F_s \\ -F_s \operatorname{sgn}(F_e) & \text{si } v = 0 \wedge |F_e| \geq F_s \end{cases}$$

donde  $F_e$  es la fuerza externa y  $F_s$  es un coeficiente de fricción estática.

Dentro de los modelos más sencillos está el modelo de fricción viscosa. La fricción viscosa ocurre típicamente cuando hay algún tipo de lubricación entre las superficies y es proporcional a la velocidad,  $bw$ .

La tres fricciones se pueden modelar como una función de la velocidad  $F(v)$  con la llamada curva de Stribeck, que además incluye el efecto Stribeck, la cual se representa en la figura 5.9. El efecto Stribeck refiere a un tipo de fricción que ocurre típicamente cuando se usa lubricación para el contacto entre distintas partes mecánicas. Lo que ocurre es que a bajas velocidades la fricción es decreciente hasta una cierta velocidad llamada la velocidad de Stribeck. El efecto Stribeck se debe a la transición entre distintos regímenes de lubricación, los llamados *Boundary Lubrication*, *Mixed Lubrication* y *Hydrodynamic Lubrication* [28, 32, 33]. En un extremo se tiene un contacto áspero significativo entre las superficies donde la lubricación se ve interrumpida por el contacto, donde el principal mecanismo de soporte son el contacto entre las distintas rugosidades de las superficies. En el otro extremo prácticamente no se tiene contacto entre las superficies, el principal contacto es a través del film lubricante [34].

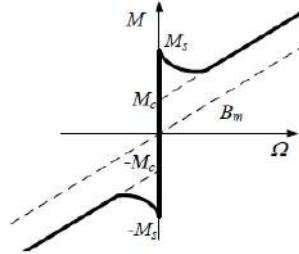


Figura 5.9: Curva Stribeck, en este caso torque vs velocidad angular. Tomada de [33].

Hay diferentes formas de describir la curva  $F(v)$  pero la más común es utilizar

$$\tau_{fu}(w) = \left( F_c + (F_s - F_c) e^{-\left(\frac{w}{w_s}\right)^\delta} \right) \tanh(\sigma w) + bw \quad (5.2)$$

como se utilizó en [35]. Este modelo también aparece en [32] y con una pequeña variante se utiliza en [33]. La curva *stribeck* también aparece en [36]. En el libro [28] aparecen varios modelos para la fricción entre los cuales se encuentra 5.2. Al parecer el modelo 5.2 es una adaptación de un modelo propuesto por [37].

Aquí elegimos el modelo dado por la ecuación (5.2) con  $\delta = 2$  para identificar la fricción del motor acoplado con el soporte en U.

$$\tau_{fu}/k = \hat{b}w + \left( \hat{F}_c + (\hat{F}_s - \hat{F}_c)e^{-\left(\frac{w}{w_s}\right)^2} \right) \text{sign}(w)$$

donde los coeficientes  $\hat{b}$ ,  $\hat{F}_c$ ,  $\hat{F}_s$  se relacionan con los coeficientes de fricción mediante la multiplicación por  $k$ . Para ajustar el modelo lo que hacemos es determinar el modelo de la fricción viscosa, ajustando una recta a los datos de la fricción en movimiento. Para esto juntamos backward y forward y hacemos el ajuste

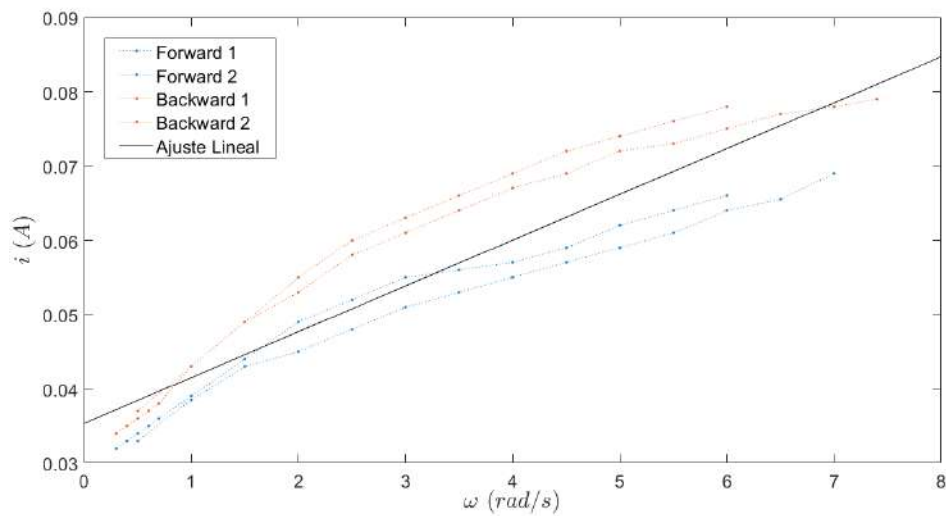


Figura 5.10: Ajuste lineal de la corriente vs velocidad

Esto determina  $\hat{b}$  y  $\hat{F}_c$ . Luego para  $\hat{F}_s$  utilizamos el promedio entre las corrientes de arranque relevadas. El parámetro  $w_s$  se ajusta de manera que la fricción relevada caiga en la fricción viscosa, pero sin otro criterio más que este. Se utiliza  $w_s = 0,05 \text{ rad/sec}$ . Finalmente, en la figura 5.11 se observa el resultado del ajuste. Los valores obtenidos son:

$$\hat{F}_c = 0,03534 \text{ A} , \hat{F}_s = 0,0556 \text{ A} , \hat{b} = 0,00617 \frac{\text{A}}{\text{rad/s}}$$

## 5.2. Motor

El modelo utilizado para describir la dinámica del motor consta de una fuente ideal cuando en la práctica se trata de una fuente PWM. Esto es un punto importante a tener en cuenta pues si bien la velocidad en régimen del motor coincide con las distintas fuentes, tanto simulaciones como relevos muestran que el PWM cambia el tiempo de respuesta, siendo este más lento.

## Capítulo 5. Identificación del Sistema

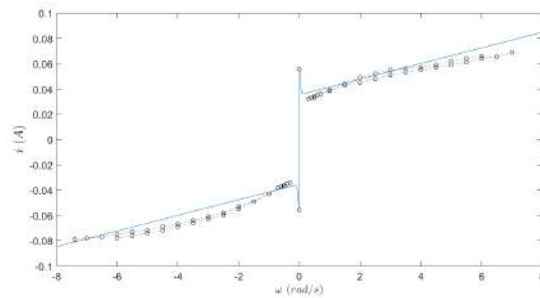


Figura 5.11: Modelo ajustado para la fricción del motor con soporte en U.

Por lo tanto, si uno releva los parámetros del motor utilizando ensayos de respuesta estacionaria, no es de extrañarse que el modelo quede más rápido de lo que realmente es, pues el modelo utiliza una fuente DC.

Para esquivar este problema y tener un modelo que sea bueno tanto en régimen como en el transitorio, lo que se hace es determinar un set de parámetros iniciales mediante ensayos de respuesta estacionaria y posteriormente se realiza un ajuste de estos parámetros que contempla la respuesta transitoria del sistema.

### 5.2.1. Parámetros iniciales

#### Determinación de $L_0$ y $R_0$

Para obtener los parámetros del modelo eléctrico del motor primero se midió la resistencia en bornes del motor con el motor en reposo, por lo tanto el valor obtenido es  $R_{in}$ . El resultado fue de  $9,3 \pm 0,5 \Omega$ . Para obtener  $L$  se ensayó con el motor en reposo, por lo que la impedancia vista hacia el motor es  $R_{in} + L_{in}s$ . Lo que se hizo fue colocar voltajes sinusoidales y buscar la frecuencia de cambio de  $3dB$ . Para esto se agregó una resistencia  $R_p = 4,8 \pm 0,5 \Omega$  para medir la caída de tensión con el osciloscopio y obtener la corriente que circula por el circuito indirectamente. El diagrama eléctrico del ensayo se observa en 5.12 donde  $R_o$  es la resistencia de salida del generador de señales que es  $50 \Omega$ .

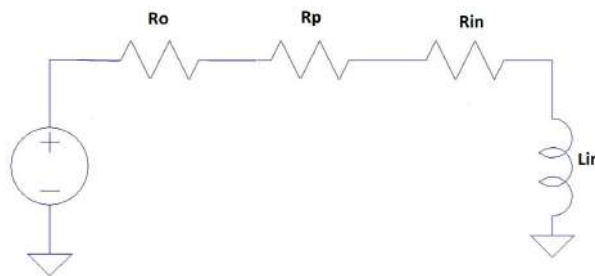


Figura 5.12: Diagrama eléctrico del ensayo



## 5.2. Motor

Con este esquema se tiene que  $L = \frac{R_o + R_{in} + R_p}{\omega}$  donde  $\omega$  es la frecuencia de  $3dB$ . El valor de  $\omega$  obtenido es de  $1489 \pm 49 \frac{rad}{s}$ . Por lo tanto el valor de  $L_{in}$  es  $43,1 \pm 4,1 mHy$ .

### Determinación de $k_0$

Como se mostró en la sección 4.2 la ecuación de la malla en el motor es:

$$v - Ri - L \frac{di}{dt} - kw = 0 \quad (5.3)$$

Según esta ecuación en estado estacionario vale  $v - Ri - kw = 0$ . Por lo tanto en esa condición si se releva la entrada  $v$ , la corriente que circula por el motor  $i$  y la velocidad del eje del motor  $w$  se puede obtener el valor de  $k$ .

Para este ensayo se utilizó el driver, que como ya se mencionó es una salida PWM pero para nuestro modelo lo vamos a suponer DC, por lo tanto solo se tomara el valor DC de las variables relevadas. De las variables a relevar la velocidad  $w$  se filtra con la propia dinámica, sin embargo  $v$  e  $i$  tienen una frecuencia fundamental de  $1 kHz$  por lo tanto es importante recalcar que medimos el valor medio de ellas.

El ensayo consistió en aplicar una tensión constante con el driver y relevar la corriente entregada hacia el motor y la velocidad angular del eje. De (5.3) es inmediato que en estado estacionario vale

$$k = \frac{v - Ri}{w}$$

Al igual que se hizo cuando se relevó la fricción, se relevan datos en ambos sentidos de giro. Se decidió estimar el  $k$  en los sentidos de giro por separado dado que se podrían observar diferencias. En la tabla 5.2 se presentan los datos para el ensayo en sentido antihorario y en la tabla 5.3 los datos en sentido horario.

El valor en la incertidumbre de  $w$  se calculó mediante simulaciones dado que se tiene un filtro para calcular la velocidad donde no está del todo claro la propagación de incertidumbre del ángulo medido por lo tanto se optó por simulaciones. Se observó que la incertidumbre no cambiaba para diferentes velocidades por lo que se optó por usar una única incertidumbre. Para ver más detalles dirigirse al apéndice C.

Es claro cómo el valor obtenido de  $k$  disminuye al aumentar la velocidad angular. Además para el sentido de giro horario es levemente menor. A partir de los resultados se busca el mejor representante para  $k$ .

Para buscar el mejor representante a  $k$  se hace mínimos cuadrados lineales en cada caso donde se busca el  $k$  que mejor ajusta a los datos en la ecuación  $v = Ri + kw$  con  $R = 9,3 \Omega$ . El coeficiente de correlación para los datos de  $k$  en sentido horario es 0,99, mientras que para el del sentido antihorario es 0,97. Para

## Capítulo 5. Identificación del Sistema

$w \left(\frac{rad}{s}\right)$	corriente(mA)	Tension(V)	$k \text{ (Nm/A)}$
0,5 ± 0,067	33 ± 0,14	1,03 ± 0,01	1,439 ± 0,34
1,0 ± 0,067	38,5 ± 0,15	1,76 ± 0,01	1,398 ± 0,17
1,5 ± 0,067	43 ± 0,15	2,49 ± 0,03	1,39 ± 0,23
2,0 ± 0,067	45 ± 0,16	3,22 ± 0,04	1,40 ± 0,09
2,5 ± 0,067	48 ± 0,16	3,92 ± 0,04	1,39 ± 0,07
3,0 ± 0,067	51 ± 0,16	4,63 ± 0,04	1,38 ± 0,06
3,5 ± 0,067	53 ± 0,16	5,32 ± 0,05	1,38 ± 0,05
4,0 ± 0,067	55 ± 0,17	6,03 ± 0,05	1,38 ± 0,05
4,5 ± 0,067	57 ± 0,17	6,73 ± 0,06	1,38 ± 0,04
5,0 ± 0,067	59 ± 0,17	7,43 ± 0,06	1,38 ± 0,04
5,5 ± 0,067	61 ± 0,17	8,21 ± 0,06	1,37 ± 0,03
6,0 ± 0,067	64 ± 0,17	8,80 ± 0,06	1,37 ± 0,03
6,5 ± 0,067	65 ± 0,17	9,5 ± 0,07	1,37 ± 0,03
7,0 ± 0,067	69 ± 0,18	10,2 ± 0,07	1,36 ± 0,03

Tabla 5.2: Resultados del ensayo para obtener K en el sentido de giro antihorario.

$w \left(\frac{rad}{s}\right)$	corriente(mA)	Tension(V)	$k \text{ (Nm/A)}$
-0,5 ± 0,067	-37 ± 0,15	-1,05 ± 0,01	1,404 ± 0,33
-1,0 ± 0,067	-43 ± 0,15	-1,78 ± 0,01	1,375 ± 0,17
-1,5 ± 0,067	-49 ± 0,16	-2,51 ± 0,03	1,336 ± 0,11
-2,0 ± 0,067	-53 ± 0,16	-3,22 ± 0,04	1,360 ± 0,09
-2,5 ± 0,067	-58 ± 0,17	-3,91 ± 0,04	1,346 ± 0,07
-3,0 ± 0,067	-61 ± 0,17	-4,61 ± 0,04	1,345 ± 0,06
-3,5 ± 0,067	-64 ± 0,17	-5,29 ± 0,05	1,34 ± 0,05
-4,0 ± 0,067	-67 ± 0,17	-5,96 ± 0,05	1,33 ± 0,04
-4,5 ± 0,067	-69 ± 0,18	-6,63 ± 0,06	1,33 ± 0,04
-5,0 ± 0,067	-72 ± 0,18	-7,30 ± 0,06	1,33 ± 0,04
-5,5 ± 0,067	-73 ± 0,18	-7,97 ± 0,06	1,32 ± 0,04
-6,0 ± 0,067	-75 ± 0,18	-8,62 ± 0,06	1,32 ± 0,04
-6,5 ± 0,067	-77 ± 0,18	-9,25 ± 0,07	1,31 ± 0,03
-7,0 ± 0,067	-78 ± 0,18	-9,90 ± 0,07	1,31 ± 0,03
-7,4 ± 0,067	-79 ± 0,18	-10,4 ± 0,07	1,31 ± 0,02

Tabla 5.3: Resultados del ensayo para obtener K en el sentido de giro horario.

el sentido horario se obtiene un valor de  $k = 1,37 \pm 0,03 \text{ Nm/A}$  y para el sentido antihorario  $k = 1,32 \pm 0,04 \text{ Nm/A}$ . Destacamos el parecido de los resultados. Se decide usar el valor medio de ambos  $k = 1,35 \pm 0,04 \text{ Nm/A}$ .

Valor de  $J_0$

Como parámetro inicial de la inercia del soporte en  $U$  se toma el valor teórico calculado en 4.1.6. En dicha sección se obtuvo

$$J_0 = 0,0185 \pm 0,00096 \text{ kg} \cdot \text{m}^2$$

### 5.2.2. Parámetros ajustados

Para obtener un modelo que sea más consistente en la respuesta transitoria es necesario ajustar los parámetros iniciales. La forma en la que se hizo eso fue minimizando una cierta función objetivo, la cual tiene en cuenta el transitorio y el régimen a la vez.

Para hacer esto se relevaron 3 respuestas al escalón en velocidad,  $y_1(t)$ ,  $y_2(t)$  e  $y_3(t)$ , en distintos rangos de voltaje, a saber, de 3 a 4 V, de 4 a 5 V y de 8 a 9 V. Luego se construyó la siguiente función objetivo:

$$U : \mathbb{R}_+^4 \longrightarrow \mathbb{R}$$

$$U(R, L, k, J) = \|w_1 - y_1\|_{I_1} + \|w_2 - y_2\|_{I_2} + \|w_3 - y_3\|_{I_3}$$

donde  $\|f\|_{I_i}^2 = \int_{I_i} f^2(x)dx$  y  $w_i$  es la solución de la ecuación diferencial que modela la dinámica del motor con el soporte en  $U$  del prototipo:

$$\begin{cases} L \frac{di}{dt} = -Ri - kw_i + V_i \\ J\dot{w}_i = ki - \tau_{fu}(w_i) \\ \tau_{fu}(w_i) = k\hat{b}w_i + k \left[ \hat{F}_c + (\hat{F}_s - \hat{F}_c) e^{-\left(\frac{w_i}{v_s}\right)^2} \right] \tanh(1000w_i) \end{cases}$$

La idea es que eligiendo los intervalos de tiempo adecuados,  $I_i$ , la función  $U$  estará midiendo el error entre la respuesta experimental  $y_i$  y la predicha por el modelo  $w_i$ . El intervalo de tiempo debe ser uno que contenga parte del transitorio y parte del régimen.

Al moverse en el dominio de  $U$  se cambian todos los parámetros del modelo pero manteniendo la estructura del mismo. Respecto a la fricción se mantiene fija la relación  $w \mapsto \tau_{fu}/k$ , la cual se determinó en la sección 5.1.2.

Con dicha función objetivo  $U$ , se obtienen los parámetros ajustados minimizando  $U$  mediante un algoritmo numérico<sup>1</sup>, que utiliza como semilla a los parámetros iniciales  $R_0$ ,  $L_0$ ,  $k_0$  y  $J_0$ . En las figuras 5.13, 5.14 y 5.15 se muestran las tres respuestas al escalón utilizadas, junto con los intervalos de tiempo  $I_i$ , y las respuestas de los modelos con los parámetros iniciales y con los parámetros ajustados.

<sup>1</sup>Se utilizó la función de Matlab `fminsearch`

## Capítulo 5. Identificación del Sistema

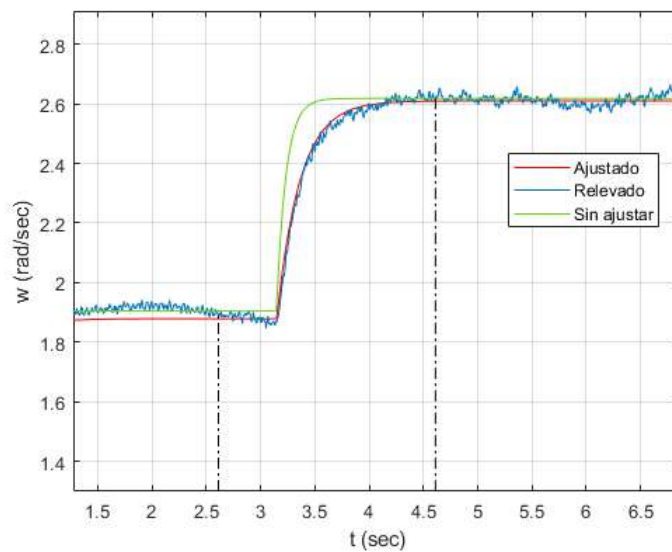


Figura 5.13: Respuesta al escalón de 3 a 4 V. Se utilizó el intervalo de tiempo  $I_1$  marcado en negro.

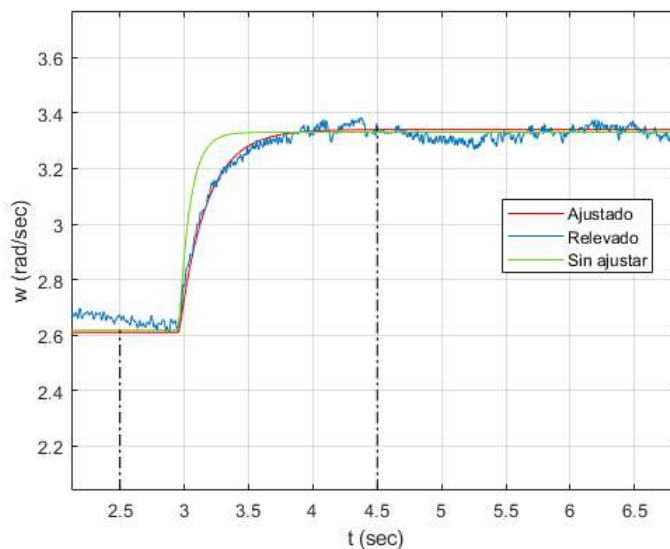


Figura 5.14: Respuesta al escalón de 4 a 5 V. Se utilizó el intervalo de tiempo  $I_2$  marcado en negro.

Como se puede observar en las dos primeras figuras, el modelo con parámetros sin ajustar es más rápido que el relevado, mientras que el modelo con parámetros ajustados copia el transitorio del relevado. También se observa que el modelo con cualquiera de los dos juegos de parámetros coincide en continua con el relevado.

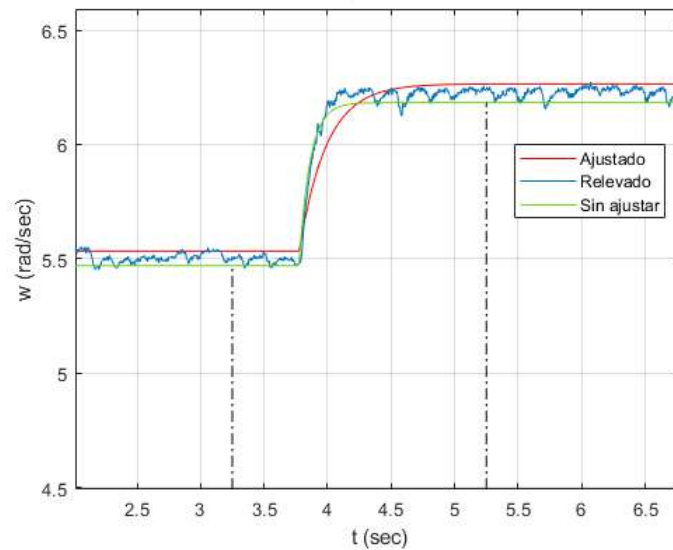


Figura 5.15: Respuesta al escalón de 8 a 9 V. Se utilizó el intervalo de tiempo  $I_3$  marcado en negro.

Respecto a la última figura, notamos que el motor ahora tiene un mejor tiempo de respuesta, tal es así que ahora el modelo con parámetros sin ajustar copia mejor el transitorio que el modelo con parámetros ajustados. De todas formas observamos un buen resultado por parte del modelo con parámetros ajustados. Nuevamente en continua no hay nada que discutir.

Si bien los parámetros ajustados son buenos para predecir la velocidad  $w$  aún queda la duda de qué tan buenos son para predecir el resto de las variables. En las tablas 5.4 y 5.5 se muestra el desempeño del modelo para medir la respuesta estacionaria del resto de las variables del motor. Para hacer este ensayo se utiliza el motor acoplado solo con el soporte en U, y se impone la velocidad mediante un controlador PI, como en la figura 5.7.

En ambas tablas se ve que los parámetros ajustados y los sin ajustar prácticamente tienen el mismo desempeño. Además se observa que la predicción es prácticamente lo mejor que se puede hacer, teniendo en cuenta la diferencia que hay entre el motor girando en una dirección u otra.

En la tabla 5.6 resumimos los resultados de los parámetros obtenidos para el motor.

## Capítulo 5. Identificación del Sistema

$\dot{\phi}$ (rad/s)	Corriente (mA)			
	forward	backward	Sin Ajustar	Ajustados
0,5	33 ± 0,14	37 ± 0,15	38,4	38,4
1,0	38,5 ± 0,15	43 ± 0,15	41,5	41,5
1,5	43 ± 0,15	49 ± 0,16	44,6	44,6
2,0	45 ± 0,16	53 ± 0,16	47,7	47,7
2,5	48 ± 0,16	58 ± 0,17	50,8	50,7
3,0	51 ± 0,16	61 ± 0,17	53,9	53,8
3,5	53 ± 0,16	64 ± 0,17	56,9	56,9
4,0	55 ± 0,17	67 ± 0,17	60,0	60,0
4,5	57 ± 0,17	69 ± 0,18	63,1	63,1
5,0	59 ± 0,17	69 ± 0,18	66,2	66,2
5,5	61 ± 0,17	73 ± 0,18	69,3	69,3
6,0	64 ± 0,17	75 ± 0,18	72,4	72,4
6,5	66 ± 0,17	77 ± 0,18	75,4	75,4
7,0	69 ± 0,18	78 ± 0,18	78,5	78,5
7,5	-	79 ± 0,18	81,6	81,7

Tabla 5.4: Corriente en estado estacionario. Forward y Backward refieren al motor girando en distintas direcciones.

$\dot{\phi}$ (rad/s)	Tensión (V)			
	forward	backward	Sin Ajustar	Ajustados
0,5	1,03 ± 0,01	1,05 ± 0,01	1,03	1,11
1,0	1,76 ± 0,01	1,78 ± 0,01	1,74	1,79
1,5	2,49 ± 0,03	2,51 ± 0,03	2,44	2,48
2,0	3,22 ± 0,04	3,22 ± 0,04	3,14	3,16
2,5	3,92 ± 0,04	3,91 ± 0,04	3,85	3,84
3,0	4,63 ± 0,04	4,61 ± 0,04	4,55	4,53
3,5	5,32 ± 0,05	5,29 ± 0,05	5,25	5,21
4,0	6,03 ± 0,05	5,96 ± 0,05	5,96	5,89
4,5	6,73 ± 0,06	6,63 ± 0,06	6,66	6,57
5,0	7,43 ± 0,06	7,30 ± 0,06	7,37	7,26
5,5	8,12 ± 0,06	7,97 ± 0,06	8,07	7,94
6,0	8,80 ± 0,06	8,62 ± 0,06	8,77	8,62
6,5	9,50 ± 0,07	9,25 ± 0,07	9,47	9,30
7,0	10,2 ± 0,07	9,90 ± 0,07	10,18	9,99
7,5	-	10,40 ± 0,07	10,88	10,53

Tabla 5.5: Tensión en régimen. Forward y Backward refieren al motor girando en distintas direcciones.

Tipo	$R(\Omega)$	$L (mHy)$	$k (\frac{Nm}{A})$	$J(kg m^2)$
Sin ajustar	$9,3 \pm 0,5$	$43 \pm 4,1$	$1,35 \pm 0,04$	$0,0185 \pm 0,00096$
Ajustados	12,16	84	1,29	0,0321

Tabla 5.6: Resumen de resultados de la sección motor

### 5.3. Resultados

Los primeros resultados sobre el modelo son los mostrados en las secciones anteriores. Sin embargo allí solo se puede evaluar el rendimiento del modelo por separado. En esta sección se somete el modelo a una prueba final, donde se compara la predicción del mismo con la respuesta del sistema completo. De esta manera se evalúa el modelo conseguido. Para esto se releva la respuesta del sistema a un escalón de voltaje. En la figura 5.16 se recuerda el esquema del sistema

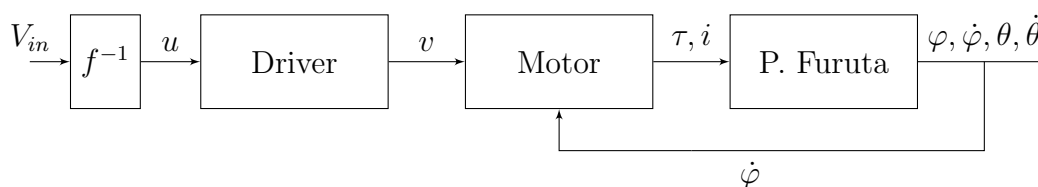


Figura 5.16: Esquema del sistema

Para el ensayo se coloca en  $V_{in}$  un escalón de 8,6 V que comienza en  $t = 0,5$  s y termina en  $t = 1$  s, siendo  $t$  el tiempo de ejecución. Se relevan las señales  $i, \varphi, \dot{\varphi}, \theta, \dot{\theta}$  durante 10 s. Se repitió el ensayo 10 veces para obtener una serie de medidas. Por otro lado se tomó el modelo desarrollado en este capítulo y el anterior con los parámetros ajustados y con los parámetros sin ajustar, y se simuló el modelo con la entrada que se le colocó al sistema. De esta manera el ensayo utilizado somete al modelo a varias pruebas. Para comenzar el sistema parte de la zona muerta del motor, y además es sometido a un escalón de gran amplitud.

Los resultados se observan en las figuras 5.17, 5.18, 5.19, 5.20 y 5.21, donde solo se despliega en los intervalos de tiempo adecuados.

En la figura 5.17 observamos la respuesta de  $\varphi$ . Como se puede ver, en régimen, tanto los parámetros ajustados como los parámetros sin ajustar predicen bien la respuesta del modelo. En cuanto al transitorio ambos modelos copian la forma de la respuesta del sistema pero ambos están fuera del margen de incertidumbre. Aun así la predicción es muy buena, siendo mejor la de los parámetros ajustados, como era esperable en transitorios.

## Capítulo 5. Identificación del Sistema

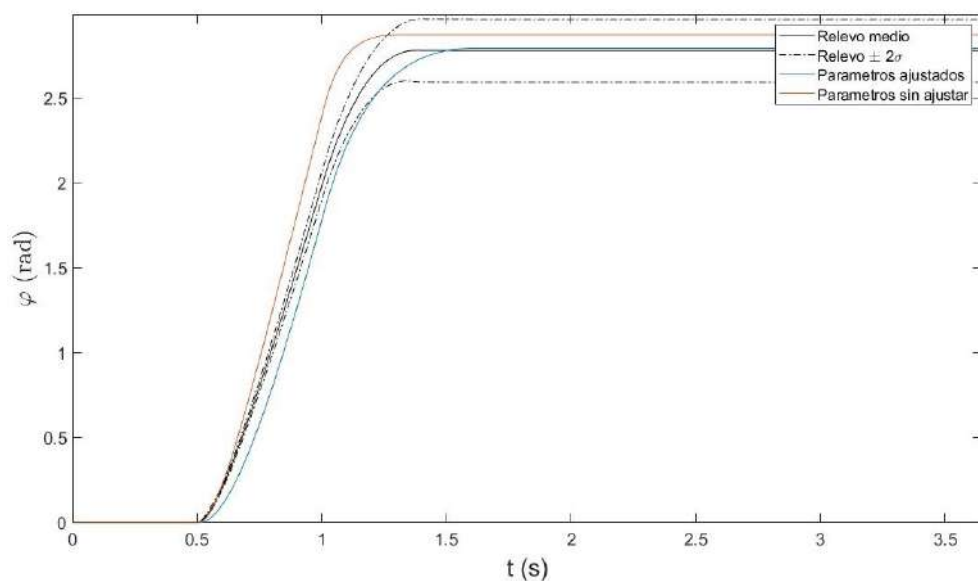


Figura 5.17: Sistema vs Modelo: Respuesta de  $\varphi$ .

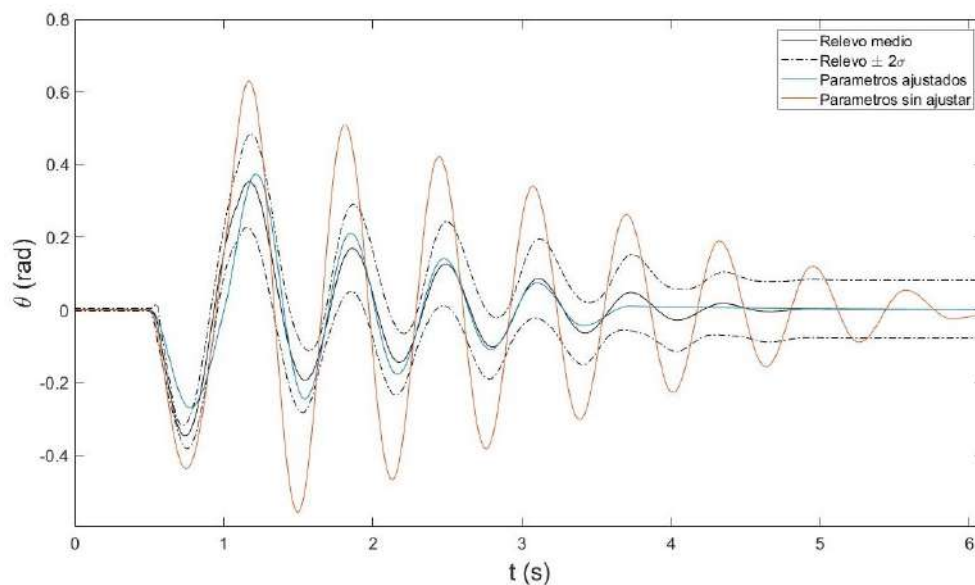


Figura 5.18: Sistema vs Modelo: Respuesta de  $\theta$ .

En la figura 5.18 observamos la respuesta de  $\theta$ . Nuevamente tanto el modelo con parámetros ajustados como sin ajustar copian cualitativamente la curva de respuesta del sistema. Cuantitativamente la respuesta dada por los parámetros sin ajustar ya no es tan buena como antes, teniendo su mayor error en el tiempo de parada. Respecto a la respuesta del sistema con parámetros ajustados, el mismo



### 5.3. Resultados

sigue la respuesta del sistema con gran precisión, teniendo su mayor diferencia en el tiempo de parada. El modelo con parámetros ajustados se detiene antes que la respuesta del sistema. La respuesta del sistema y de los parámetros ajustados se mantienen iguales prácticamente hasta que la amplitud llega a menos de  $0,1 \text{ rad}$ . A partir de ese entonces el modelo se detiene mientras que el sistema sigue una oscilación más. Este es un punto difícil de modelar, el carácter más complicado de la fricción entra en juego. Más allá de este detalle, la predicción del modelo nunca se sale del margen de incertidumbre validando el resultado.

En la figura 5.19 se puede ver la respuesta de  $\dot{\varphi}$ . Otra vez ambos juegos de parámetros copian cualitativamente la respuesta del sistema. Los resultados dados por los parámetros sin ajustar no son tan buenos mientras que los de los parámetros ajustados se mantienen muy buenos, si bien por momentos se sale de la zona de incertidumbre. Recordar que el modelo está intentando predecir la respuesta del sistema partiendo de la zona muerta y aun así los resultados son muy buenos.

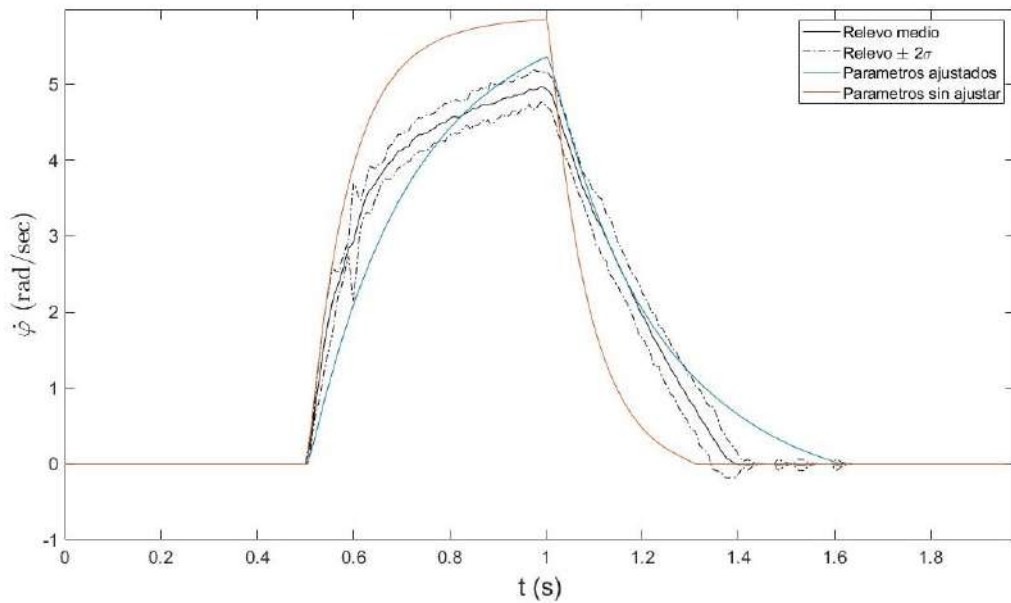


Figura 5.19: Sistema vs Modelo: Respuesta de  $\dot{\varphi}$ .

En la figura 5.20 vemos la respuesta de  $\dot{\theta}$ . Los resultados son consistentes con los de  $\theta$ . Las respuestas copian bien al relevo. El set de parámetros sin ajustar no es bueno para la predicción cuantitativa sin embargo los parámetros ajustados se mantienen en casi todo momento dentro del margen de incertidumbre. Naturalmente se observa el mismo resultado que antes en cuanto al tiempo de parada, aun así los resultados son muy buenos siguiendo al sistema hasta velocidades pequeñas.

Finalmente en la figura 5.21 vemos los resultados para la corriente. Como observamos el modelo copia cualitativamente la respuesta del sistema hasta  $t = 1 \text{ s}$ .

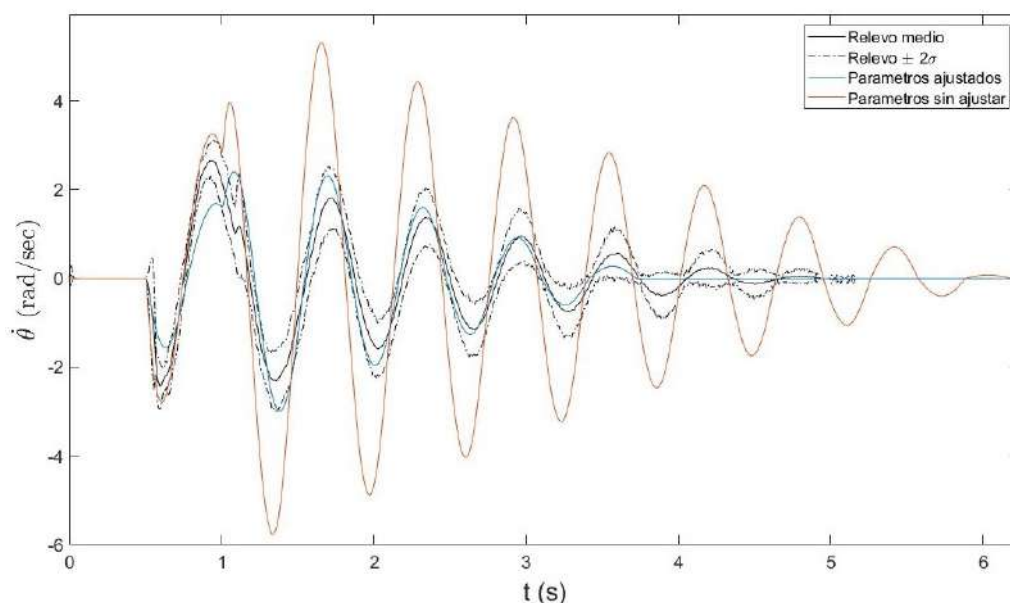


Figura 5.20: Sistema vs Modelo: Respuesta de  $\dot{\theta}$ .

Sin embargo, a partir de ese tiempo en adelante el sistema corta la corriente mientras que el modelo indica corrientes negativas de igual forma al primer intervalo de tiempo. Esto era lo esperable, dada la limitación aclarada en 4.2.1. Recordamos que esto se debe que al pedirle  $0\text{ V}$  al driver en realidad el mismo impone  $i = 0$ . Como ya se mencionó, este efecto se decidió no modelar por lo cual no nos sorprende, ni molesta, la respuesta del modelo para  $t > 1\text{ s}$ .

En cuanto a lo cuantitativo, en el transitorio el modelo falla incluso hasta un 100%. Aquí hay que tener en cuenta la otra limitación comentada en 4.2.1, en realidad la corriente tiene una frecuencia de  $1\text{ kHz}$  y lo que relevamos es el filtrado con un sistema de primer orden con polo en  $100\text{ Hz}$ , por lo cual tampoco podemos pretender que el modelo copie este comportamiento, pues solo se pretendió predecir el valor medio de  $i$  cuando  $i$  está en régimen. Respecto a esto se tienen los resultados de la tabla 5.4. Aquí nos conformamos con tener un modelo que prediga cualitativamente el comportamiento del valor medio de la corriente, y que en régimen predice dicho valor de manera muy buena. Quien marca la conformidad es el test final, controlar. El modelo ha mostrado ser bueno para diseñar distintos controladores, como veremos en 7.3, y por lo tanto el modelo cumple con el objetivo para el cual fue derivado.

### 5.3.1. Sobre la zona muerta

Aquí mostramos una de las bondades del modelo, el mismo simula la zona muerta del motor. Veremos esto con dos simulaciones. En ambas simulaciones uti-

### 5.3. Resultados

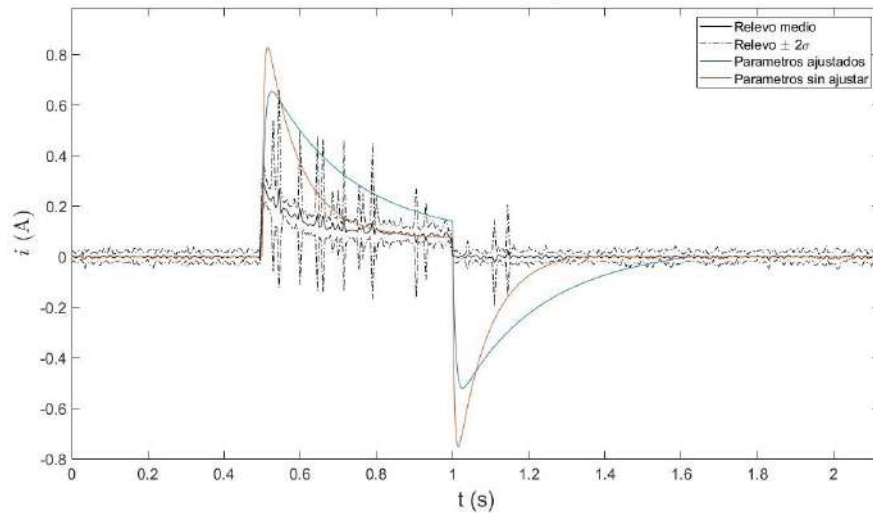


Figura 5.21: Sistema vs Modelo: Respuesta de  $i$ .

lizamos solamente el motor acoplado al soporte en U.

En la primera utilizamos el controlador PI como en 5.7, colocamos una velocidad tan chica como  $0,5 \text{ rad/sec}$  y una constante del integrador bien pequeña. De esta forma el motor parte del reposo con acción de control nula, como el motor se encuentra en la zona muerta no arrancará y entonces el integrador empezará a acumular el error. Como la constante del integrador es pequeña entonces la acción de control crecerá lentamente hasta que el motor arranque. Observamos los resultados en la figura 5.22.

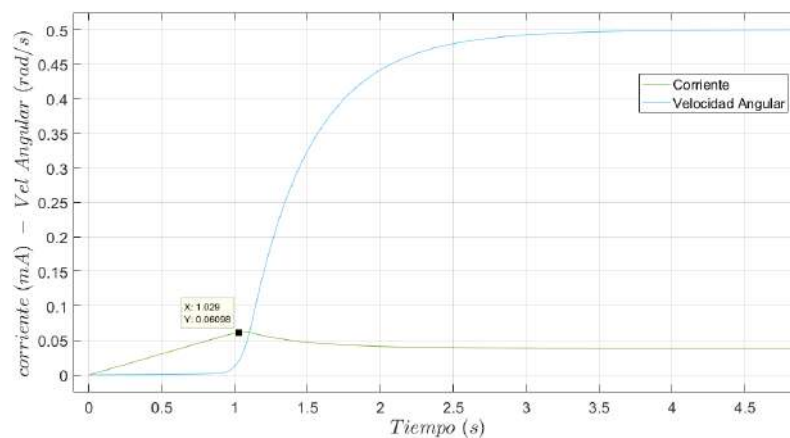


Figura 5.22: Respuesta dada por el motor+Soporte U frente a un seguidor de velocidad. Se observa el efecto de la zona muerta.

## Capítulo 5. Identificación del Sistema

Recordamos aquí que el par ejercido por el motor es directamente proporcional a la corriente, por lo cual al mirar la corriente estamos viendo el torque. En la figura es claro cómo durante el primer segundo el motor no arranca aunque que la corriente aplicada no es nula. El motor solo arranca cuando la corriente supera el valor de  $60\text{ mA}$  aproximadamente, lo cual coincide con la corriente de arranque relevada en la práctica. Esto muestra cómo el modelo simula una zona muerta al momento de arrancar.

En la segunda simulación relevamos la curva de respuesta estacionaria tensión-velocidad. En la figura 5.23 se ven los resultados. Los mismos muestran la típica curva tensión velocidad de un motor con zona muerta, evidenciando que el modelo implementa la el efecto de la misma. Además el modelo predice el voltaje límite de la zona muerta encontrado en la práctica.

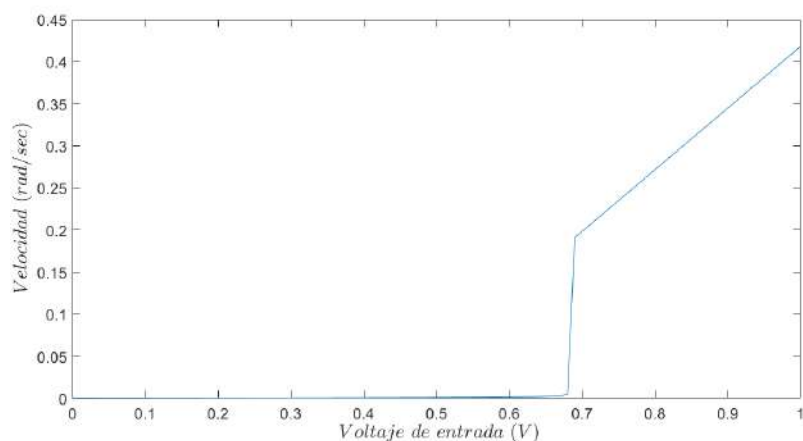


Figura 5.23: Curva de respuesta estacionaria tensión velocidad del motor+Soporte U. Se observa el efecto de la zona muerta.

Se está muy conforme con los resultados obtenidos tanto a nivel puntual como en los ensayos del sistema completo. La excepción a estos buenos resultados es la corriente. Si bien se conocen las causas de porque el modelo falla se decide no probar modelar ese efecto ya que se considera que el objetivo del modelo está cumplido. Es claro como el ajuste de parámetros realizado mejora los resultados a nivel cuantitativo. Como virtudes extras del modelo se destaca que logra predecir la zona muerta del motor naturalmente y ajusta muy bien la corriente de arranque del motor en DC.

### 5.4. Resumen: modelo y parámetros

A modo de cierre de este capítulo y del proceso de modelado e identificación de los parámetros comenzado en el capítulo anterior se presenta a continuación las

#### 5.4. Resumen: modelo y parámetros

ecuaciones del modelo utilizado y también se muestran los valores utilizados para los parámetros.

$$\left\{ \begin{array}{l} \ddot{\varphi} (J + 2m_2l_{bi}c_x + m_2l_{bi}^2 + I_z + I_x \sin^2 \theta) \\ + 2\dot{\varphi}\dot{\theta}I_x \cos \theta \sin \theta + m_2l_{bi}c_z\ddot{\theta} \cos \theta - m_2l_{bi}c_z\dot{\theta}^2 \sin \theta = \tau + \tau_{fu}(\dot{\varphi}) \\ I_x\ddot{\theta} + m_2l_{bi}c_z\ddot{\varphi} \cos \theta - \dot{\varphi}^2 I_x \cos \theta \sin \theta + m_2gc_z \sin \theta = \tau_{fp}(\dot{\theta}) \\ \tau = ki \\ v - Ri - L\frac{di}{dt} - k\dot{\varphi} = 0 \\ \tau_{fp}(\dot{\theta}) = -b_2\dot{\theta} - C_{ep} \tanh(1000\dot{\theta}) \\ \tau_{fu}(\dot{\varphi}) = b\dot{\varphi} + \left( F_c + (F_s - F_c)e^{-\left(\frac{\dot{\varphi}}{w_s}\right)^2} \right) \tanh(1000\dot{\varphi}) \end{array} \right. \quad (5.4)$$

Como se observa se utiliza la función  $\tanh(1000x)$  como aproximación de la función *signo*.

Parámetro	Valor
$J$	0,0321 $kgm^2$
$m_2$	98 $g$
$l_{bi}$	8,7 $cm$
$C_x$	-4 $cm$
$I_x$	$4,39 \times 10^{-4} kgm^2$
$I_z$	$2,24 \times 10^{-4} kgm^2$
$C_z$	4,4 $cm$
$k$	1,29 $Nm/A$
$R$	12,16 $\Omega$
$L$	84 $mHy$
$F_c$	0,0456 $Nm$
$F_s$	0,0717 $Nm$
$b$	0,00796 $Nms$
$C_{ep}$	$6,51 \times 10^{-4} Nm$
$b_2$	$6,64 \times 10^{-5} Nms$

Tabla 5.7: Parámetros utilizados para el modelo.

Esta página ha sido intencionalmente dejada en blanco.

## Capítulo 6

# Integración en Software de Simulación

En este capítulo se muestra la implementación del sistema con Arduino en el Software de simulación. Se recuerda el diagrama de comunicación del sistema mostrado en 2.4. La idea es que el software de simulación permita interactuar con cada uno de los componentes a través del Arduino.

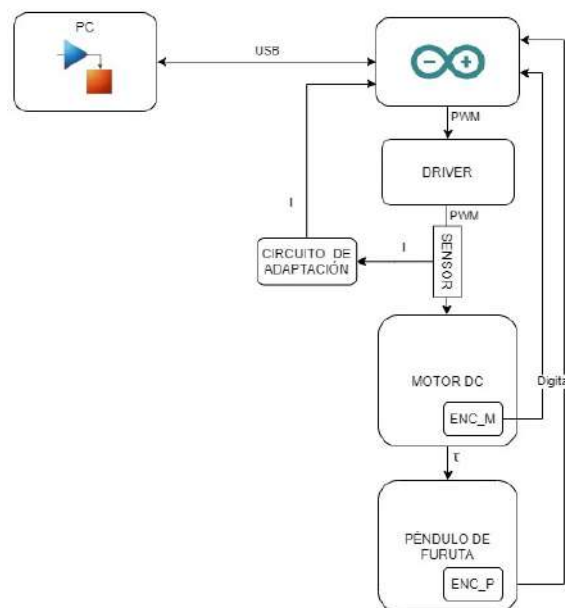


Figura 6.1: Diagrama de comunicación del sistema.

Una vez logrado ese primer objetivo lo siguiente es poder ejecutar en el sistema los controladores diseñados. Para esto se debe poder interactuar con todos los componentes a la vez e implementar bloques con las leyes de control. Por último, se implementa el relevamiento las variables de interés para poder comparar con las simulaciones.

Existe una gran cantidad de ventajas al utilizar un software centralizado para diseñar, simular e implementar los bloques en el mismo programa. El software

## Capítulo 6. Integración en Software de Simulación

elegido es Simulink. Esta elección se basó en la versatilidad de la herramienta. La misma dispone de potentes herramientas de simulación y a su vez logra desarrollar el código que corre en Arduino fácilmente. Esto permite realizar simulaciones, almacenar los resultados obtenidos en el software y luego comparar directamente contra los resultados de un ensayo sin tener que hacer ningún tipo de procesamiento para su posterior análisis. Esto acelera sustancialmente el proceso de diseño de los controladores. De ser necesario implementar cada controlador en el Arduino, esto implicaría una etapa de implementación adicional donde pueden ocurrir errores de implementación. Es muy sencillo generar el código deseado para las pruebas gracias a su interfaz de usuario y al plugin *Simulink Support Package for Arduino Hardware* ofrecido. Éste se encarga del pasaje de un sistema de bloques a la generación de código para ser ejecutado directamente en el microcontrolador. Además ofrece la posibilidad de correr el programa en distintos modos. Se destaca particularmente el modo *external* donde el código corre en Arduino pero mantiene la comunicación con Simulink en tiempo de ejecución permitiendo relevar diferentes señales de interés y observar señales en tiempo real. Luego, en el modo *Deploy to Hardware* el código corre directamente en la placa de forma independiente del Simulink, como si se programara directamente en Arduino IDE.

A continuación se comentará sobre los bloques implementados y por último se indicarán las condiciones en las que se probó en el prototipo y también algunas consideraciones de uso.

### 6.1. Bloques básicos

Los bloques que cumplen la función de comunicarse con un componente específico del sistema los llamamos bloques básicos. Estos consisten en un bloque para obtener la posición y velocidad angular a partir de las lecturas de un encoder, otro para realizar lecturas del sensor de corriente y finalmente un bloque para manejar el actuador del motor. Las pruebas realizadas para cada uno de los componentes fueron explicadas en el capítulo 3, estas pruebas fueron implementadas directamente en Arduino IDE, sin la utilización de la herramienta Simulink. En este capítulo nos centraremos en el diseño y test del bloque.

#### 6.1.1. Encoder

Se tiene un bloque para cada encoder los cuales se nombran *Encoder Péndulo* y *Encoder Motor*. Estos bloques se encargan de manejar las interrupciones generadas por los pines conectados a los encoders, leer el estado de dichos pines y llevar la cuenta para obtener la posición, luego realizar la conversión a radianes y finalmente implementar un filtro para calcular la velocidad angular. El diseño y las pruebas realizadas para la obtención de este filtro se explican en la sección 3.3. En la figura 6.2 se muestra una ilustración de los bloques.

Estos bloques no tienen entradas y tienen dos salidas:



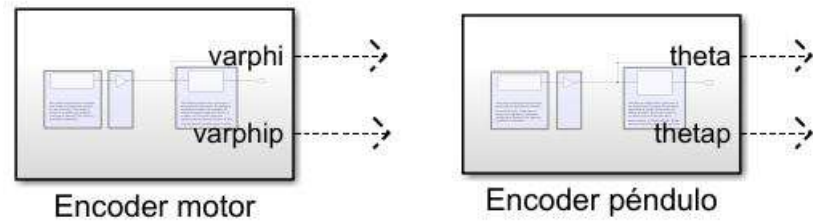


Figura 6.2: Ilustración de los bloques implementados en Simulink

- $\text{varphi}$ : Posición del encoder del motor en radianes.
- $\text{varphi}$  dot: Velocidad angular del motor en  $\text{rad}/\text{s}$ .
- $\text{theta}$ : Posición del encoder del péndulo en radianes.
- $\text{thetap}$ : Velocidad del péndulo en  $\text{rad}/\text{s}$ .

En la figura 6.3 se muestra cómo está construido el bloque *Encoder Péndulo*.

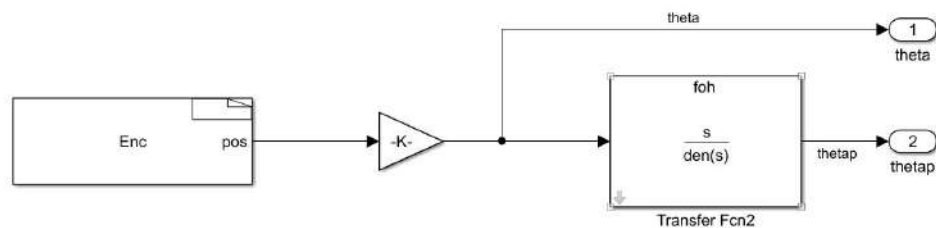


Figura 6.3: Detalle de implementación del bloque *Encoder Péndulo*

El bloque *Enc* es una *S-Function*<sup>1</sup> encargada de llevar la cuenta de los pulsos del encoder. En la configuración de este bloque se debe indicar a cuáles pines está conectado el encoder, que en el caso del péndulo son el 2 y el 3, y el tiempo de muestreo.

El bloque funciona por medio de interrupciones, por lo que en el bloque se escribió la rutina de atención a la interrupción (ISR) de estos pines, en el apéndice D se muestra este código. Se configuran los pines en modo *Change*, esto hace que interrumpen en cada cambio de nivel de cada uno de los pines. De este modo se obtiene una resolución de 2400 cuentas por revolución. Dentro de la ISR se aumenta o disminuye la cuenta según corresponda.

El tiempo de muestreo definido indica cada cuánto se actualizará la salida del bloque. La constante  $-K-$  es el factor multiplicativo para pasar a radianes.

<sup>1</sup>S-functions (system-functions) es una descripción en lenguaje informático de un bloque de Simulink escrito en MATLAB®, C, C++ o Fortran. Las funciones C, C++ y Fortran S se compilan como archivos MEX utilizando la utilidad mex.

## Capítulo 6. Integración en Software de Simulación

El bloque *Transfer Fcn2* es un filtro en tiempo discreto para obtener la derivada de la posición. Este es un tipo de bloque de Simulink para implementar filtros, en él se deben definir los coeficientes para el numerador y denominador de la transferencia. Es importante que el tiempo de muestreo de este bloque coincida con el definido en el bloque anterior para que la velocidad calculada sea la correcta. Este filtro implementa en tiempo discreto la transferencia de tiempo continuo:

$$\frac{s}{\left(\frac{1}{2\pi 30}\right)^2 s^2 + \frac{2}{2\pi 30} s + 1}$$

La transferencia de tiempo continuo fue diseñada en 3.3. La idea del filtro es tener un comportamiento derivativo a bajas frecuencias y luego un polo doble de forma de atenuar las frecuencias que no son de interés, para eso el polo doble en 30 Hz da una caída de primer orden luego de esa frecuencia. Además se busca no dejar pasar señales de frecuencias que no están en la zona de interés. Para la implementación en tiempo discreto se utiliza la transferencia  $H(z)$  dada por el teorema de la transmitancia muestreada para un retenedor de orden uno (FOH). Para ver detalles referirse al apéndice E.

El bloque que implementa el encoder del motor no se detalla porque es análogo al del péndulo. Las diferencias se dan en la configuración de la *S-Function* donde se establecen los pines 18 y 19, y en el valor de la constante multiplicativa para obtener la posición en radianes

$$\text{Encoder Péndulo: } \frac{2\pi}{600 \cdot 4} \quad \text{Encoder Motor: } \frac{2\pi}{11 \cdot 78 \cdot 4}$$

El filtro de velocidad de este encoder es idéntico al anterior.

### 6.1.2. Sensor de corriente y calibrador

Se implementó un bloque para leer la corriente que circula por el sensor, el cual se nombra *Sensor de Corriente*. En la figura 6.4 se muestra el bloque.

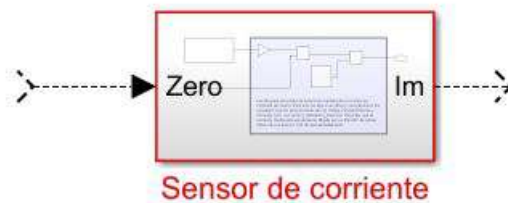


Figura 6.4: Bloque de lectura de corriente implementado en Simulink

Este bloque tiene una entrada y una salida:

- zero: Referencia de corriente 0 en Voltios.
- Im: Corriente medida en Amperios.

## 6.1. Bloques básicos

El bloque se encarga de leer el pin A0 que es el utilizado para adquirir la salida del sensor, para esto se utiliza un bloque de la librería *Simulink Support Package for Arduino Hardware* llamado *Analog Input*. A este bloque se le indica el pin que se desea leer y el tiempo de muestreo.

La salida de este bloque es un número entre 0 y 1023. Éste valor se traduce a tensión con un bloque proporcional. Luego se resta el valor en tensión correspondiente a corriente 0 y este valor se divide entre la sensibilidad de nuestro instrumento, que como se mostró antes es  $1,267 \text{ V/A}$  (ver 3.1), obteniendo el valor de corriente medido. En la figura 6.5 se puede observar la implementación del bloque.

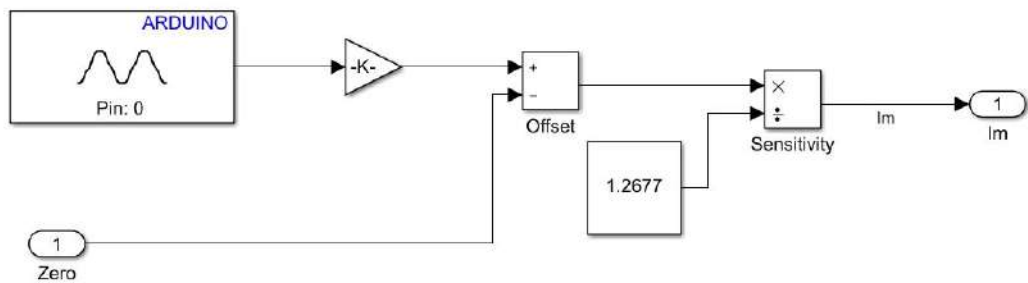


Figura 6.5: Detalle de implementación del bloque current sensor.

La referencia de tensión correspondiente al valor de corriente  $0 \text{ A}$  se pasa como entrada al bloque, típicamente es  $V_{CC}/2$ . No se tiene un valor fijo dentro del bloque porque en la práctica se observó que puede variar en el orden de  $100 \text{ mV}$ . Para poder obtener el valor correcto de referencia se implementó un script llamado *Calibrador* que lo que hace es relevar el valor de tensión correspondiente a corriente  $0 \text{ A}$ . En la figura 6.6 se muestra cómo está hecho el script.

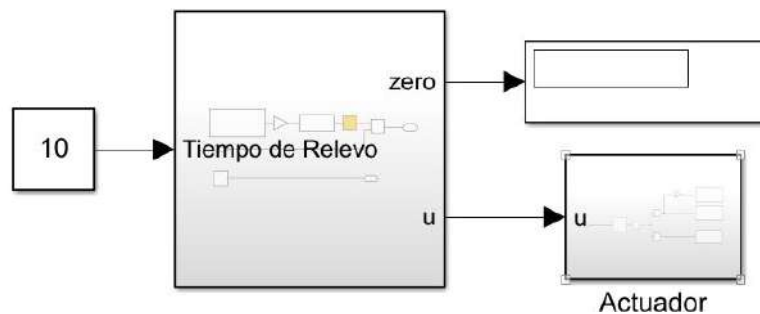


Figura 6.6: Imagen del script calibrador.

La idea del script es imponer durante el tiempo de relevo  $0 \text{ V}$  en el actuador de forma de estar seguros que no circula corriente por el motor para luego hacer un promedio. Este tiempo de relevo debe ser el mismo que se configure en la duración

## Capítulo 6. Integración en Software de Simulación

del script en Simulink. Una vez finalizado el script, se despliega en el display el resultado obtenido.

En la imagen 6.7 se observa como está compuesto el bloque central de la figura 6.6. Se releva el pin A0 al igual que lo hace el bloque *Sensor de Corriente*, se integra estos valores y se divide durante el tiempo de relevo, obteniendo como resultado el valor medio de la salida del sensor en reposo.

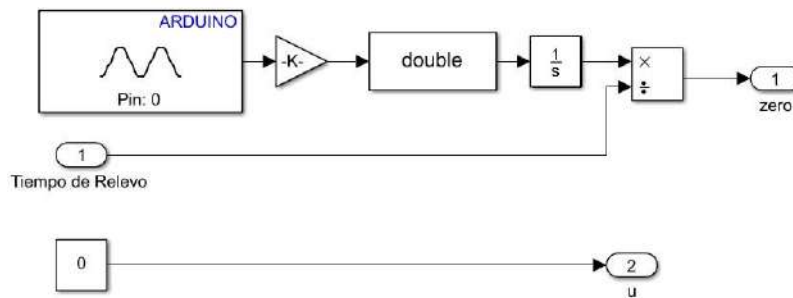


Figura 6.7: Detalle de implementación del calibrador.

Se debe correr este script cada vez que se comienza a trabajar con el prototipo.

### 6.1.3. Actuador

El bloque *Actuador* es el encargado de manejar la tensión aplicada al motor a través al driver. En la figura 6.8a se muestra el bloque. Tiene una entrada,  $u$ , que es la tensión que se quiere aplicar en el motor.

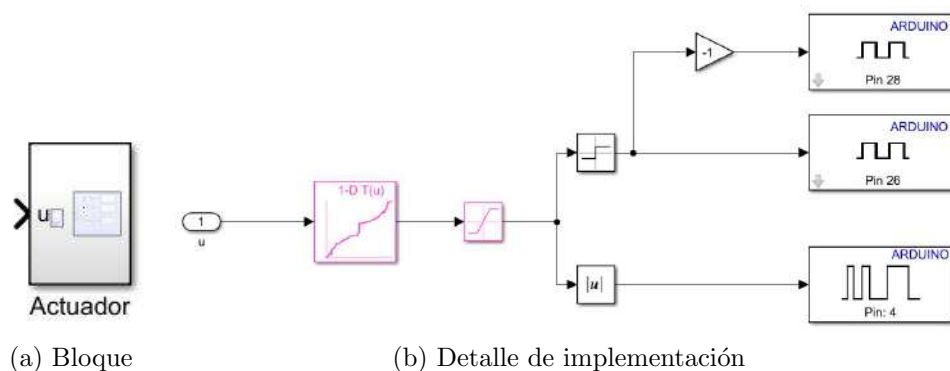


Figura 6.8: Ilustraciones del bloque Actuador.

Como se mostró en 3.2 al driver se le indica la polaridad de la tensión mediante dos pines digitales (que en este caso se usan los pines 26 y 28) y luego en otra entrada se le indica el valor PWM. En la figura 6.8b se muestra la implementación

realizada. Además como se mostró en 3.2.1 la curva de tensión de la salida del driver no es lineal con respecto a la entrada. Para solucionar esto se tiene una *Lookup table* que dada una entrada de voltaje, devuelve el valor de PWM a aplicar para que a la salida del driver este el voltaje deseado. A esta salida se le aplica un bloque signo para saber qué nivel aplicar en los pines 26 y 28 y un bloque módulo para obtener el valor de PWM a aplicar en el pin 4. Finalmente obtenemos como resultado, la tensión y sentido de giro esperadas en el motor.

## 6.2. Otros bloques

Una vez diseñados los bloques básicos para manejar las diferentes partes del sistema se implementaron otras funcionalidades sencillas. Estas fueron usadas de forma recurrente por lo que se decidió crear bloques para hacer las aplicaciones más simples. A continuación se muestran los más importantes.

### 6.2.1. Seguidor

Un bloque que es muy usado es el seguidor de posición para el motor. Su principal utilidad es desenredar el cable del encoder cuando se dan varias vueltas. Es un controlador de posición proporcional que solo mira  $\varphi$ :

$$u = -k(\varphi - \varphi_0)$$

De este modo logramos dejar el motor en la posición deseada. El valor de  $k$  es elegido en la práctica. Para valores de  $k$  menores a 10 se observa que el brazo no llega al objetivo, esto sucede porque es un controlador proporcional y cuando la diferencia con el objetivo es pequeña la tensión aplicada no es suficiente para vencer la zona muerta del motor. Para valores de  $k$  mayores a 30 se observa que se tiene un pequeño sobretiro, en nuestra aplicación este sobretiro no molesta y se decide usar  $k = 50$  en busca de tener un mejor tiempo de levantamiento, y presumiblemente un mejor tiempo de asentamiento (efectivamente así lo es). En la figura 6.9 se puede observar la implementación del mismo.

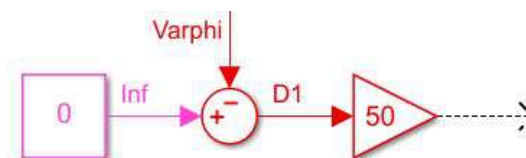


Figura 6.9: Implementación del seguidor.

### 6.2.2. Estabilizador de bajada

Cuando se está controlando el péndulo en posición invertida y éste se cae el ángulo  $\theta$  oscila durante unos segundos. Para evitar estas oscilaciones se utilizó un controlador estabilizante de bajada. Se implementó un controlador que tiene como punto de referencia llevar todos los estados a cero excepto  $\varphi$  que se le pasará un ángulo al cual seguir. Esto último permite estabilizar al péndulo en la posición  $\varphi_0$  donde cayó. En el siguiente capítulo se mostrarán mas detalles.

### 6.2.3. Máquina de estados

Para la lógica de transición entre los diferentes controladores es necesario una máquina de estados. Un flujo normal en el funcionamiento del péndulo podría comenzar con *swing up* para luego pasar a un control estabilizante, luego eventualmente un control estabilizador de bajada precedido por el seguidor de posición para evitar que el cable del encoder quede enredado obstruyendo el giro.

Los estados que se tendrían son: *Inicial* (aquí actúa el controlador estabilizante de bajada), *SwingUp*, *Estabilizante* y *Des*. Luego se deben definir las condiciones para las transiciones y eventualmente modos de funcionamiento.

En la imagen 6.10 se presenta la implementación de una máquina de estados. Aquí se tienen en cuenta dos modos de funcionamiento dependiendo del valor de la señal *modo*. Esta señal determina qué se debe hacer cuando el péndulo se cae durante un control estabilizante. Cuando *modo* es 0 se entra directamente al estado *swingUp*. Por otro lado, si *modo* es 1 se entra al estado *Inicial* en donde actúa el control estabilizante de bajada.

Para implementar esta máquina de estados se utiliza el bloque de Simulink *Chart*. La salida *y* del bloque es utilizada para indicarle a un bloque externo qué ley de control aplicar en cada comento. Luego, la salida *sh* se utiliza como variable auxiliar para pasar a los controladores estabilizantes qué valor de  $\varphi$  debe seguir.

### 6.2.4. Bloque controlador

Este bloque se implementa utilizando el bloque *Matlab Function* de Simulink. Tiene las leyes de control de todos los controladores existentes en un sistema. Recibe como entradas el relevo de todos los estados del sistema y las señales *y* (estados de la máquina) y *sp* (set point) que se generan a partir de la máquina de estados.

La lógica es muy simple, se encuentra un *switch case* en función de la señal *y* para elegir qué controlador aplicar. La señal *sh* se levanta cada vez que se ingresa al estado *Estabilizante* o al estado *Inicial*, eso hace que se copie a *sp* el valor de  $\varphi$  de ese momento. El valor que tiene *sp* será la referencia a seguir para los controladores estabilizantes.

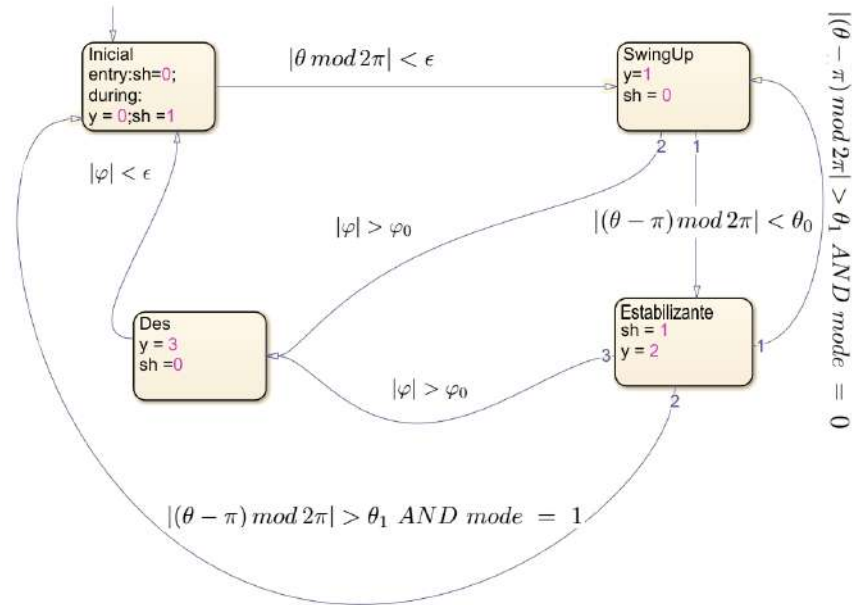


Figura 6.10: Ejemplo de una máquina de estado para manejar la lógica de funcionamiento

### 6.3. Ejemplo total

En esta sección se presenta un ejemplo de funcionamiento completo. En la figura 6.11 se muestra el script utilizado. A la izquierda se observan los bloques básicos (excepto el *actuador* que acciona el motor) de los que se obtiene el valor instantáneo de las variables de estado del sistema. Luego, la exponencial compleja para  $\theta$  se utiliza para medir la distancia a  $\pi \bmod 2\pi$ , es decir, sin importar si el péndulo dio muchas vueltas. La idea de esto es medir la distancia en  $\mathbb{C}$  de  $e^{j\theta}$  a  $-1$ . Con esto y con el valor del *mode* se cuenta con toda la información necesaria para la máquina de estados.

A continuación, el bloque de *Sample and hold* se utiliza para retener el valor de  $\varphi$  en el momento que se levanta la señal *sh*. Su salida, *sp*, se inserta en el bloque controlador actualizando el valor a seguir por los controladores.

Como se observa en la figura en este ejemplo se agregó un *switch* previo al actuador. De esta forma se puede elegir, con un click, qué valor de tensión aplicar al motor, pudiendo elegir si aplicar la salida de la lógica de controladores o poder desenredar manualmente con la salida del bloque seguidor cuando se quiera. Los detalles sobre los controladores utilizados y los resultados obtenidos se mostrarán en el siguiente capítulo.

## Capítulo 6. Integración en Software de Simulación

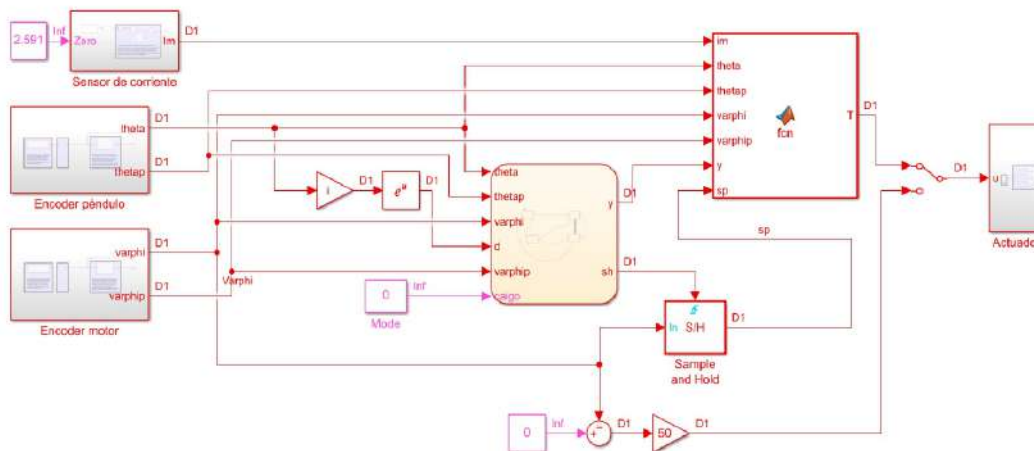


Figura 6.11: Ilustración de un script que implementa todas las funcionalidades desarrolladas.

### 6.4. Restricciones del prototipo, condiciones de prueba y consideraciones de uso

En esta sección se listarán las condiciones en las cuales el sistema fue probado, además de remarcar algunas de las limitaciones a tener en cuenta para el diseño de controladores.

- Frecuencia de muestreo del sistema entre  $50\text{ Hz}$  y  $200\text{ Hz}$ . En estas condiciones se corrobora un correcto funcionamiento.
- Utilizar un tiempo de muestreo único en todos los bloques.
- Serial baud rate: Se probó a 115200 baudios y funciona bien.
- Configuración del solver: Type: *Fixed step*, Solver: *auto*.
- La velocidad del péndulo relevada no superó los  $30\text{ rad/s}$ . Vale aclarar que no es necesario alcanzar velocidades mayores para un correcto funcionamiento del *swing up* o del control estabilizante. No se buscó hasta qué velocidad el sistema funciona bien, pero hasta los  $30\text{ Hz}$  funciona bien.
- Los polos del filtro para el encoder están en  $30\frac{\text{rad}}{\text{s}}$ . Si se requiere es posible cambiarlos en los bloques correspondientes pero no se probaron otros polos.
- Sobre el sensor de corriente: el fondo de escala del sensor son  $2\text{ A}$ .<sup>2</sup>

<sup>2</sup>Se recuerda que se debe calibrar el nivel cero cuando se comience a trabajar y que la corriente realmente es una señal a  $1\text{ kHz}$ . El valor medido es la corriente filtrada por un filtro de primer orden con frecuencia de corte  $100\text{ Hz}$ .



#### 6.4. Restricciones del prototipo, condiciones de prueba y consideraciones de uso

- Sobre el driver: La corriente máxima según el fabricante son  $2\text{ A}$ , la corriente máxima relevada fue de  $1,5\text{ A}$ . La frecuencia del PWM utilizada fue  $1\text{ kHz}$ . Si bien es posible cambiarla esto no se probó.
- A la hora de diseñar controladores tener en cuenta que si bien la alimentación es de  $12\text{ V}$  la tensión máxima aplicada al motor es de aproximadamente  $10,5\text{ V}$  por la caída en el driver.

Esta página ha sido intencionalmente dejada en blanco.

# Capítulo 7

## Controladores

En este capítulo se presenta el desarrollo e implementación de un controlador estabilizante y de un controlador de *swing up*. Los mismos cumplen los objetivos inicialmente planteados. La síntesis de estos controladores se llevó a cabo utilizando el sistema creado a lo largo de la tesis, sirviendo como validación del sistema completo. Además se explica cómo se combinan estos controladores, junto con un control supervisor, para lograr un único objetivo común: que el péndulo se levante por sí solo y quede estabilizado en la posición invertida. El control supervisor se encarga de sincronizar los dos controladores anteriores. Además evita que el sistema se enrede con el cable del encoder que mide el ángulo del péndulo, devolviendo el sistema al estado inicial. También elimina algunos tiempos de espera cuando el sistema vuelve a iniciar.

El capítulo se divide en tres grandes partes. En 7.1 se profundiza en el control estabilizante mientras que la sección 7.2 se trata del *Swing Up*. Finalmente, en la sección 7.3 se estudian estos dos controladores sincronizados por el control supervisor.

### 7.1. Control estabilizante

En general dado un sistema de la forma  $\dot{x} = f(x, v)$ ,  $x_0$  es un punto de equilibrio si  $f(x_0, 0) = 0$ . Entendemos por control estabilizante una ley  $v = \Phi(x)$  tal que el sistema en lazo cerrado  $\dot{x} = f(x, \Phi(x))$  tiene a  $x_0$  como punto de equilibrio localmente asintóticamente estable.

Teniendo en cuenta los distintos aspectos del sistema *real* (como la zona muerta del motor, el juego en el eje, el verdadero carácter de la corriente, entre otros) no es posible obtener un control estabilizante *estrictamente*, esto es en el sentido de la definición. En lugar de eso nos conformamos con obtener un control que mantenga  $\theta$  cerca de  $\pi \bmod 2\pi$ , por ejemplo a menos de  $10^\circ$ . Para las demás variables no exigimos un desempeño similar pero sí que se mantengan oscilando en torno a cero. Entendemos por esto un controlador estabilizante *en sentido amplio*.

## Capítulo 7. Controladores

La estrategia de diseño consistió en linealizar el sistema y aplicar resultados de control lineal. Se obtuvieron dos controladores con performance levemente diferente.

En la subsección 7.1.1 se presentan los resultados teóricos que sustentan el desarrollo de la ley de control. En 7.1.2 resumimos el modelo linealizado junto a sus características. En 7.1.3 describimos la metodología llevada a cabo para el diseño e implementación de un controlador en el sistema real. En 7.1.4 se explica cómo se derivaron las leyes de control utilizadas. En 7.1.5 se presentan los resultados obtenidos y finalmente en 7.1.6 se analizan los mismos.

### 7.1.1. Marco teórico

Resumimos aquí los principales resultados en los cuales se sustenta el desarrollo llevado a cabo. Los resultados pueden encontrarse en cualquier referencia de control lineal, por ejemplo [38], [39]. Las siguientes definiciones pueden tratarse en un contexto general, pero nos limitamos a hacerlo para cierta clase de sistemas. Consideraremos el siguiente sistema de tiempo continuo

$$\begin{cases} \dot{x} = Ax + Bv \\ x(0) = x_0 \end{cases} \quad (7.1)$$

donde  $x \in \mathbb{R}^n$  es el vector de estados.

$v : [0, \infty) \rightarrow \mathbb{R}^m$  es la señal de control.

$A \in \mathcal{M}_{n \times n}(\mathbb{R})$

$B \in \mathcal{M}_{n \times m}(\mathbb{R})$

#### Definición.

1. Se dice que un estado  $z$  es controlable en tiempo  $T$  si existe  $v : [0, T] \rightarrow \mathbb{R}^m$  tal que el sistema 7.1 con condición inicial  $x(0) = z$  tiene como solución  $x : [0, T] \rightarrow \mathbb{R}^n$  tal que  $x(T) = 0$ . Si esto ocurre se denota por  $z \rightsquigarrow_T 0$ .
2. Se dice que un estado  $z$  es controlable si existe  $T \geq 0$  tal que  $z \rightsquigarrow_T 0$ . Si esto ocurre se denota por  $z \rightsquigarrow 0$ .
3. El sistema 7.1 es controlable si para todo  $x \in \mathbb{R}^n$  se tiene  $x \rightsquigarrow 0$ .

El concepto de controlabilidad fue introducido por R.Kalman en 1960, [38, 40]. El mismo juega un rol importante en el diseño de sistemas de control en el espacio de estados, al punto de que la solución al problema de control puede no existir si no se tiene controlabilidad. Por otro lado, el tener la controlabilidad resuelve completamente el problema del control.

## 7.1. Control estabilizante

En un comienzo la condición de controlabilidad es una propiedad dinámica del sistema pero la estructura lineal del mismo hace que el álgebra lineal aporte al estudio de la controlabilidad. Es así que la propiedad de controlabilidad se convierte en una propiedad algebraica de las matrices  $A$  y  $B$ , como lo muestra la siguiente proposición.

**Prop 1.** *El sistema 7.1 es controlable si y sólo si  $\text{rank}(C) = n$  donde*

$$C = [B \mid AB \mid A^2B \mid \dots \mid A^{n-1}B]$$

*A la matriz  $C$  se le denomina el Grammiano de controlabilidad.*

Un gran aporte del álgebra lineal al problema de la controlabilidad es la posibilidad de utilizar cambios de bases para obtener una representación equivalente del sistema 7.1, con otras matrices  $\tilde{A}$  y  $\tilde{B}$ , que sean más simples de entender, las llamadas formas canónicas. Resulta que si un sistema es controlable entonces existe un cambio de base donde las nuevas matrices del modelo en variables de estados son muy simples. Las nuevas matrices son tan simples que permiten obtener el siguiente resultado, el cual resuelve gran parte del problema del control.

**Prop 2.** *Sea el sistema de tiempo continuo 7.1, y propongamos como señal de control  $v = -Kx$  donde  $K \in \mathcal{M}_{n \times m}(\mathbb{R})$ . Entonces, se puede elegir la posición de los valores propios de  $A - BK$  prácticamente sin restricciones eligiendo adecuadamente  $K$ , si y sólo si el sistema es controlable. Dicho de otra forma,*

*Para casi toda colección de  $n$  números  $\{\lambda_1, \dots, \lambda_n\} \subset \mathbb{C}$  existe  $K$  tal que los valores propios de  $A - BK$  son exactamente  $\lambda_1, \dots, \lambda_n$ .  $\iff$  el sistema es controlable.*

*La única restricción en la elección de los valores propios es que si  $\lambda$  es un valor propio también lo es  $\bar{\lambda}$ .*

Los valores propios de  $A - BK$  se le llaman polos del sistema en lazo cerrado. Un controlador de la forma  $v = -Kx$  se le llama *state feedback controller* o realimentación de estados. Entonces si el sistema es controlable la proposición anterior muestra que se puede diseñar un controlador prácticamente eligiendo el valor de los polos. Esta técnica es conocida como *full state feedback* o *pole placement*, y es la utilizada para diseñar el controlador que estabiliza al péndulo en la posición invertida. En *matlab* el comando *place* recibe  $A$ ,  $B$  y los polos y devuelve la matriz  $K$  que cumple lo deseado.

Otra característica deseable de un sistema lineal es su observabilidad. Esta característica refiere a la posibilidad de conocer el estado del sistema  $x$  a partir de la observación de la entrada y la salida. Notar que sin el conocimiento de  $x$  no es posible implementar el control por realimentación de estados comentado por lo cual la observabilidad es importante. Nuestro sistema claramente es observable

## Capítulo 7. Controladores

pues de hecho medimos el estado directamente. Es por esto que no ahondamos en definiciones precisas de esta característica.

La controlabilidad del sistema trae consigo varias herramientas, por ejemplo el siguiente resultado debido a Kalman, sobre el control óptimo para el sistema 7.1. La teoría del control óptimo trata de operar un sistema dinámico a un costo mínimo. El caso donde el sistema está dado por 7.1 y la función de costo es cuadrática se llama el problema de control óptimo cuadrático o *LQ problem*. Un resultado importante es que la solución está dada por el regulador lineal cuadrático o *LQR*. Precisamente, si el sistema 7.1 es controlable, si  $Q$  y  $R$  son matrices simétricas,  $Q$  definida positiva (o semidefinida) y  $R$  definida positiva, fijamos la condición inicial  $x_0$  y definimos la función de costo

$$F(x_0, v) = \int_0^{\infty} x^T Q x + v^T R v dt$$

donde  $x$  es la solución a  $\dot{x} = Ax + Bv$  con condición inicial  $x_0$ . Entonces la ley de control  $v$  que minimiza<sup>1</sup> la función de costo es  $v = -Kx$  con  $K = R^{-1}(B^T P + N^T)$  donde  $P$  es solución de la llamada ecuación de *Riccati*. En resumen, la ley de control óptimo en el sentido de la función de costo es el control por realimentación de estados. Además la solución es independiente de  $x_0$ . Este método da una manera sistemática de obtener un control óptimo que además permite pesar distinto cada variable, ya sean variables de estado o la señal de control. En *matlab* el comando *LQR* recibe las matrices  $A$ ,  $B$ ,  $Q$  y  $R$ , y devuelve el  $K$  deseado. Para más detalles sobre el *LQR* ver por ejemplo [38].

En resumen, el control por realimentación de estados es un excelente control para aplicar sobre sistemas de la forma 7.1. Si uno opta por posicionar los polos queda la pregunta de cómo elegirlos. Una posible estrategia clásica es elegir tener uno o dos polos dominantes, eligiendo los polos dominantes para cumplir las especificaciones. Aquí se debe tener cuidado de no colocar polos muy a la izquierda para que la acción de control no sea muy grande. Otra posible opción es no elegir la posición de los polos y utilizar *LQR* para definir la ley de control basados en los comportamientos que uno debe penalizar, por ejemplo acción de control o algún estado particular como puede ser  $\theta$ .

El método de *pole placement* tiene la ventaja de atacar directamente especificaciones sobre la respuesta del sistema ya sea en el dominio temporal o en frecuencia, pero tiene la desventaja de no tener en cuenta el esfuerzo de la señal de control. El *LQR* tiene la ventaja de ser una metodología fácil de iterar, que elige polos de lazo cerrado haciendo un balance entre el error de los estados y el esfuerzo del control. La desventaja es que a veces no es fácil vincular la función de costo con la respuesta transitoria.

---

<sup>1</sup>El mínimo es tomando  $v$  en cierto espacio funcional, el resultado es cierto tomando  $v \in \mathcal{L}^\infty$  el espacio de las funciones medibles esencialmente acotadas. Para un tratamiento riguroso del tema se puede ver [39]

## 7.1.2. Características de la planta

Recordamos aquí el modelo del sistema, presentado en 4.13.

$$\begin{cases} \ddot{\varphi} (J + 2m_2l_{bi}c_x + m_2l_{bi}^2 + I_x + I_x \sin^2 \theta) \\ + 2\dot{\varphi}\dot{\theta}I_x \cos \theta \sin \theta + m_2l_{bi}c_z\ddot{\theta} \cos \theta - m_2l_{bi}c_z\dot{\theta}^2 \sin \theta = \tau + \tau_{fu}(\dot{\varphi}) \\ I_x\ddot{\theta} + m_2l_{bi}c_z\ddot{\varphi} \cos \theta - \dot{\varphi}^2 I_x \cos \theta \sin \theta + m_2gc_z \sin \theta = \tau_{fp}(\dot{\theta}) \\ \tau = ki \\ v - Ri - L\frac{di}{dt} - k\dot{\varphi} = 0 \end{cases}$$

Donde el estado del sistema es  $(i, \theta, \dot{\theta}, \varphi, \dot{\varphi})$  y la entrada es  $v$ , el voltaje en bornes del motor. Para atacar el problema de la estabilización es de gran utilidad contar con los resultados que provee el control lineal, en particular los mencionados en 7.1.1. Para esto linealizamos el modelo en torno a la posición de equilibrio que se desea estabilizar, es decir, en torno al punto de equilibrio inestable  $i = 0$  A,  $\varphi = \varphi_0 \bmod 2\pi$  (rad),  $\dot{\varphi} = 0$  (rad/s),  $\theta = \pi \bmod 2\pi$  (rad) y  $\dot{\theta} = 0$  (rad/s). Para la linealización se considera el modelo sin fricción.<sup>2</sup> Teniendo en cuenta los resultados de la sección 4.3.2 sobre linealización y que las ecuaciones del motor ya son lineales, se obtiene

$$\dot{x} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ \frac{m_2gc_z(J+2m_2l_{bi}c_x+m_2l_{bi}^2+I_x)}{d} & 0 & 0 & 0 & \frac{m_2l_{bi}c_z}{d} \\ \frac{m_2^2c_z^2l_{bi}g}{d} & 0 & 0 & 0 & \frac{I_x}{d} \\ 0 & 0 & 0 & \frac{k}{L} & -\frac{R}{L} \end{pmatrix} x + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -\frac{1}{L} \end{pmatrix} v$$

donde  $x = (\theta, \varphi, \dot{\theta}, \dot{\varphi}, i)$ . Llamándole  $A$  a la primer matriz y  $B$  a la segunda entonces contamos con un modelo en variables de estado, lineal, invariante en el tiempo, de la forma  $\dot{x} = Ax + Bv$ . Al provenir de una linealización Jacobiana el modelo obtenido aproxima localmente la dinámica del péndulo de Furuta en torno al punto de equilibrio inestable.

Utilizando los valores identificados en el capítulo de 3 y 4 se puede ver que el Grammiano de controlabilidad  $\mathcal{C}$  es de rango completo. Esto se verificó utilizando un software numérico. En conclusión el sistema lineal es controlable, entonces podemos hacer *pole placement* y obtener un control lineal que también estabiliza localmente al no lineal.

Otro dato de interés son los valores propios de  $A$ . Para calcular los mismos utilizamos nuevamente software numérico, obteniendo

$$\lambda_1 = 0 \quad \lambda_2 = -140,4 \quad \lambda_3 = 10,12 \quad \lambda_4 = -10,18 \quad \lambda_5 = -4,34$$

<sup>2</sup>El modelo de la fricción no es preciso para velocidades nulas.

### 7.1.3. Metodología

Para encontrar una ley de control que logre estabilizar el péndulo en posición invertida se sigue una metodología que involucra elección de un control, simulación y prueba. En esta sección se explican los pasos seguidos. La metodología es basada en simulaciones, en el sentido de que la ley de control propuesta es primero sometida a simulaciones y si supera los requerimientos necesarios se pasa a probar en el sistema. Se siguió la siguiente metodología de trabajo:

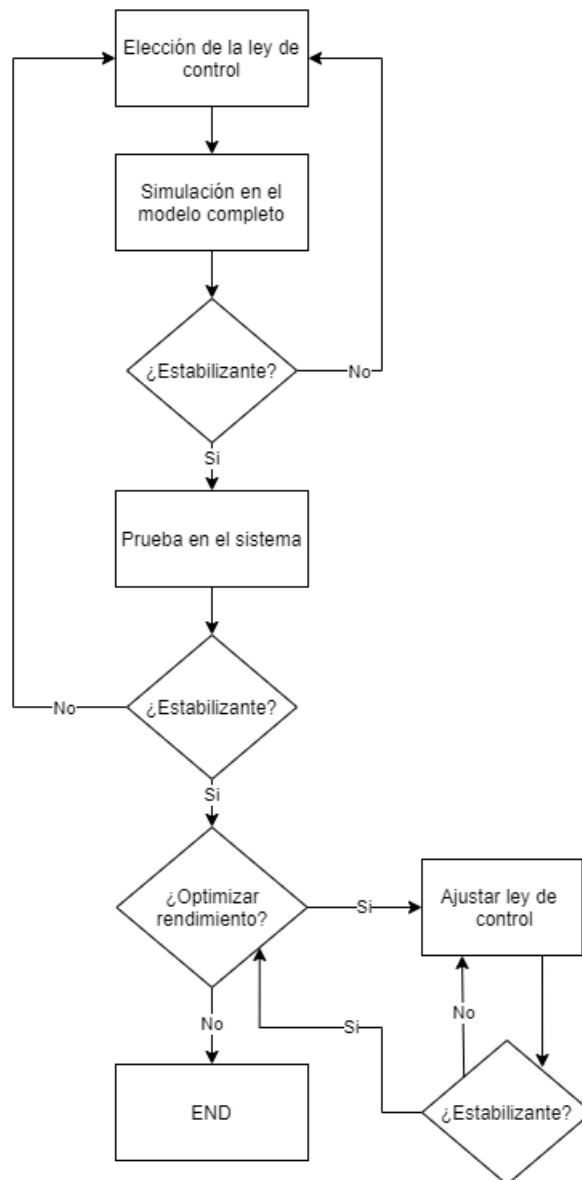


Figura 7.1: Flujo de trabajo para el diseño del controlador estabilizante.

Para la elección de la ley de control  $v$  se utiliza el modelo linealizado en torno



a la posición de equilibrio inestable,  $\dot{x} = Ax + Bv$ . El modelo linealizado es sin fricción. Una vez elegida la señal  $v$  a probar se simula dicha señal con el modelo completo, es decir, utilizando fricción viscosa y estática y también la saturación de voltaje del motor. Si el controlador logra estabilizar el péndulo se pasa a probar en el sistema físico. En general no cualquier control que superó la simulación funcionó en físico. Si el controlador logra estabilizar el sistema se pasan a estudiar aspectos de rendimiento como pueden ser la robustez, la acción de control, y otros. En caso de desconformidad con el rendimiento se pasa a ajustar los parámetros del controlador. Una vez conformes se da fin al proceso de diseño.

#### 7.1.4. Ley de control

Dadas las características de la planta a controlar se decidió probar las técnicas presentadas en la sección de marco teórico. Al final se utiliza *pole placement* para determinar la ley de control. La técnica de diseño comienza con la determinación de los polos de lazo cerrado que son que son deseables basados en la respuesta transitoria y/o en frecuencia, velocidad, factor de amortiguación, ancho de banda, requerimientos de respuesta en régimen, etc.

En el enfoque convencional del diseño de un controlador se piensa solamente en la ubicación de los polos dominantes del lazo cerrado, se los ubica para que tengan una cierta frecuencia natural  $w_n$  y un cierto factor de amortiguación  $\zeta$ . Sin embargo, al utilizar *pole placement* es necesario especificar la posición de todos los polos de lazo cerrado, no solamente los dominantes. De todas formas elegir tener polos dominantes es una estrategia posible, pero vale aclarar que no es la estrategia que se siguió aquí.

Para elegir la ley de control en una primera instancia se probó utilizar el método *LQR* variando las matrices  $Q$  y  $R$ . En todos los casos se utilizó  $Q$  diagonal. Se comenzó por matrices diagonales de igual peso. Luego se probó utilizar matrices  $Q$  y  $R$  que penalicen las distintas variables. Entre estas variantes se probaron estrategias de *cheap control*, donde se le permite al controlador ejercer una señal de control considerable, y también de *expensive control*, donde se penaliza fuertemente la señal de control frente al resto de las variables. La principal referencia en cómo penalizar las distintas variables es [41]. Ahí se comenta que una buena estrategia es colocar una gran penalización a  $\theta$  y baja penalización en  $\varphi$ . También sugiere penalizar  $\dot{\varphi}$  y  $\dot{\theta}$  para reducir oscilaciones. Además se sugiere la magnitud de los polos.

Si bien la mayoría de las leyes derivadas superan la etapa de simulación comentada en 7.1.3, ninguna logró la estabilidad en el sistema físico. Una inspección de los polos mostró que para todas las leyes derivadas siempre se tenía un polo de magnitud mayor a  $140 \text{ rad/s}$ . No se logró elegir  $R$  y  $Q$  para bajar este polo. Esto nos dió la sugerencia de utilizar *pole placement* con los polos que daba un *LQR* pero trayendo el polo mayor a 140 a valores más pequeños. Simplemente este cambio

## Capítulo 7. Controladores

logró reducir el tamaño de los coeficientes de la matriz  $K$ , incluso en ordenes de magnitud. Además de esto, la referencia [41], quién trata con un péndulo de Furuta sin motor, sugiere colocar dos de los polos para que tengan aproximadamente la misma velocidad que los polos de lazo abierto ( $\approx 6 \text{ rad/s}$ ). De los restantes dos polos sugiere colocar uno bien rápido ( $> 20 \text{ rad/s}$ ) y uno bien lento ( $< 2 \text{ rad/s}$ ). En base a estas dos sugerencias se pueden hacer varios controladores. Finalmente nos quedamos con los mencionados en la siguiente tabla, los cuales pasaron todas las etapas de diseño mencionadas en 7.1.3. En el análisis se verá las diferencias en el desempeño de estos controladores.

Coefficientes	Polos del lazo cerrado
$K = [-7.96, 35.45, 3.51, -1.66, -2.19]$	$\lambda_1 = -14 \quad \lambda_2 = -12 \quad \lambda_3 = -10$ $\lambda_4 = -8 \quad \lambda_5 = -6$
$K = [-8.30, 25.79, 2.55, -0.69, -1.76]$	$\lambda_1 = -20 \quad \lambda_2 = -8 \quad \lambda_3 = -7$ $\lambda_4 = -6 \quad \lambda_5 = -5$

Tabla 7.1: Coeficientes de los controladores implementados y los polos de lazo cerrado.

Recordamos que la ley de control es  $v = -Kx$  con  $v$  el voltaje del motor y  $x = (i, \theta - \pi, \dot{\theta}, \varphi, \dot{\varphi})^T$ .

La ley de control  $v = -Kx$  tiene una desventaja. Como  $x \in \mathbb{R}^n$  el controlador *piensa* que  $\theta$  y  $\theta + 2k\pi$  son coordenadas que refieren a puntos distintos, a diferentes configuraciones del péndulo. Esto trae problemas sobre todo a la hora de combinar este controlador con otro controlador que no distinga dichos puntos. Un ejemplo se da en la sección 7.3 donde un controlador se encarga de levantar el péndulo, llevando  $\theta$  a  $\pi \text{ mod } 2\pi$  y luego un control estabilizante mantiene el péndulo en posición invertida. El controlador que se encarga de levantar el péndulo puede dejarle al control estabilizante una condición inicial  $\theta_0 = \pi$  como  $\theta_0 = -\pi$  o cualquier otro equivalente. Entonces un control estabilizante que distinga estos puntos no tiene un único setpoint y eso es un problema. Si colocamos como setpoint  $\theta = \pi$  y el controlador que levanta el péndulo deja  $\theta_0 = -\pi$  entonces el estabilizante pensará que está muy lejos de su objetivo y no funcionará. Para solucionar este problema cambiamos la ley  $v = -k_1 i - k_2 \theta - k_3 \dot{\theta} - k_4 \varphi - k_5 \dot{\varphi}$  por

$$v = -k_1 i + k_2 \arcsin \sin \theta - k_3 \dot{\theta} - k_4 \varphi - k_5 \dot{\varphi}$$

El truco se basa en que en un entorno suficientemente grande de  $\theta + 2k_0\pi$  la función  $\arcsin \sin \theta = -(\theta - 2k_0\pi)$ . Es claro que este problema también lo sufre  $\varphi$ , sin embargo, para dicho ángulo se resuelve el problema de otra forma. Como se mencionó en 6.2.4 y cómo se volverá a ver en 7.3, cuando el sistema completo active el control estabilizante se elegirá el setpoint de  $\varphi$ , llamémosle  $\varphi_0$ , como el último valor del ángulo antes de entrar al estado. En resumen la ley de control es

$$v = -k_1 i + k_2 \arcsin \sin \theta - k_3 \dot{\theta} - k_4 (\varphi - \varphi_0) - k_5 \dot{\varphi} \quad (7.2)$$

## 7.1. Control estabilizante

y los coeficientes son los mostrados en la tabla 7.1. De ahora en más le llamamos *Control 1* y *Control 2* respectivamente a los controladores presentados en la tabla.

### 7.1.5. Resultados

En esta subsección se presentan los resultados de los controladores implementados en el sistema postergando para la siguiente la discusión y el análisis. Para el relevo de datos se coloca el péndulo en una condición inicial cercana al punto de equilibrio inestable, con  $\varphi_0 = 0 \text{ rad}$  y se releva la respuesta del sistema controlado.

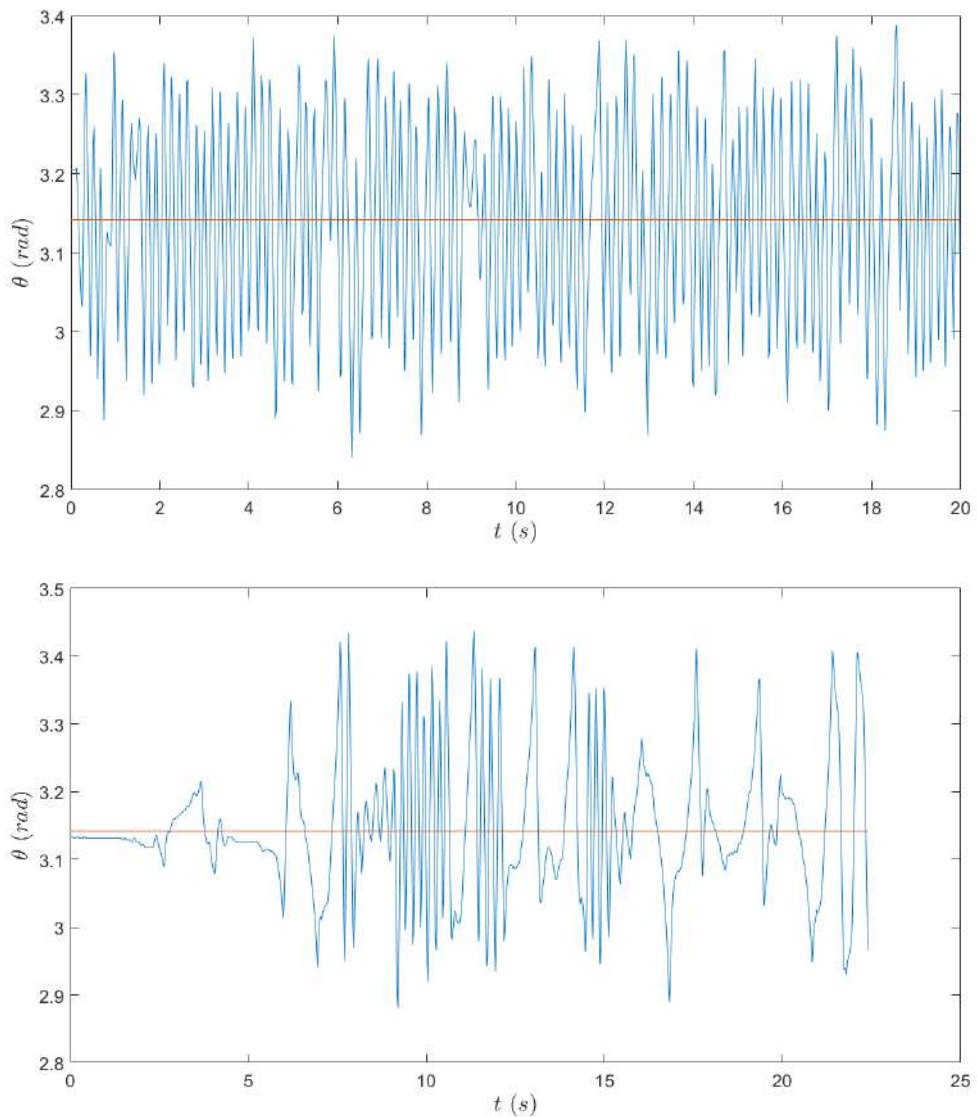


Figura 7.2: Evolución de  $\theta$  durante la estabilización. En la figura de arriba se observa el control 1 y en la de abajo el control 2. En ambas figuras se denota el objetivo con naranja.

## Capítulo 7. Controladores

En la figura 7.2 se observa el rendimiento del control 1 y del control 2 para la variable  $\theta$ , variable clave del objetivo. A su vez se muestra el objetivo,  $\pi$ , con una recta horizontal naranja. En la figura 7.3 se puede ver la respuesta para  $\dot{\theta}$ . En este caso el objetivo no se marca pues el mismo es el  $0 \text{ rad/s}$ .

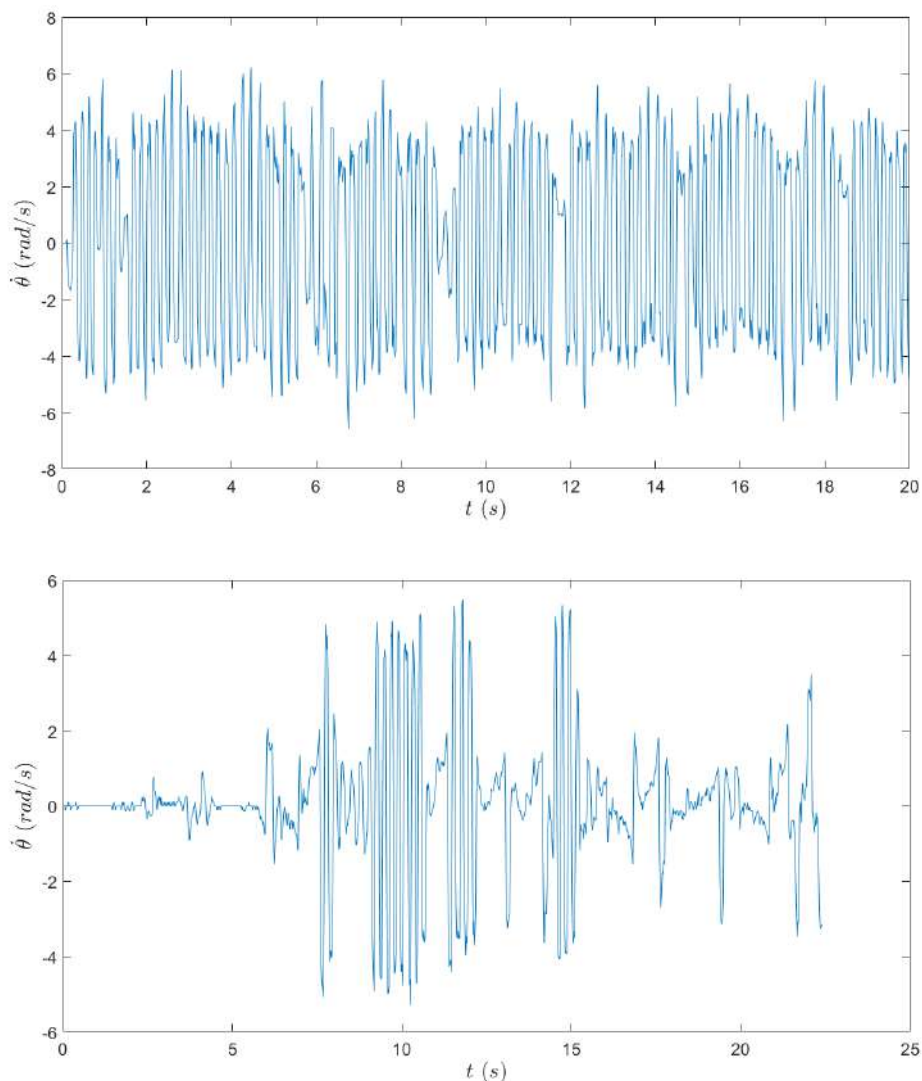


Figura 7.3: Evolución de  $\dot{\theta}$  durante la estabilización. En la figura de arriba se observa el control 1 y en la de abajo el control 2.

Como se observa en las figuras anteriores ambos controladores logran estabilizar tanto  $\theta$  como  $\dot{\theta}$  en el sentido amplio. Sin embargo no se obtiene el mismo resultado para  $\varphi$  como se puede ver en la figura 7.4. De los resultados mostrados hasta aquí se puede intuir que el controlador 2 es más suave que el controlador 1, la figura 7.4 muestra que esta suavidad fue demasiado para estabilizar  $\varphi$  adecuadamente.

## 7.1. Control estabilizante

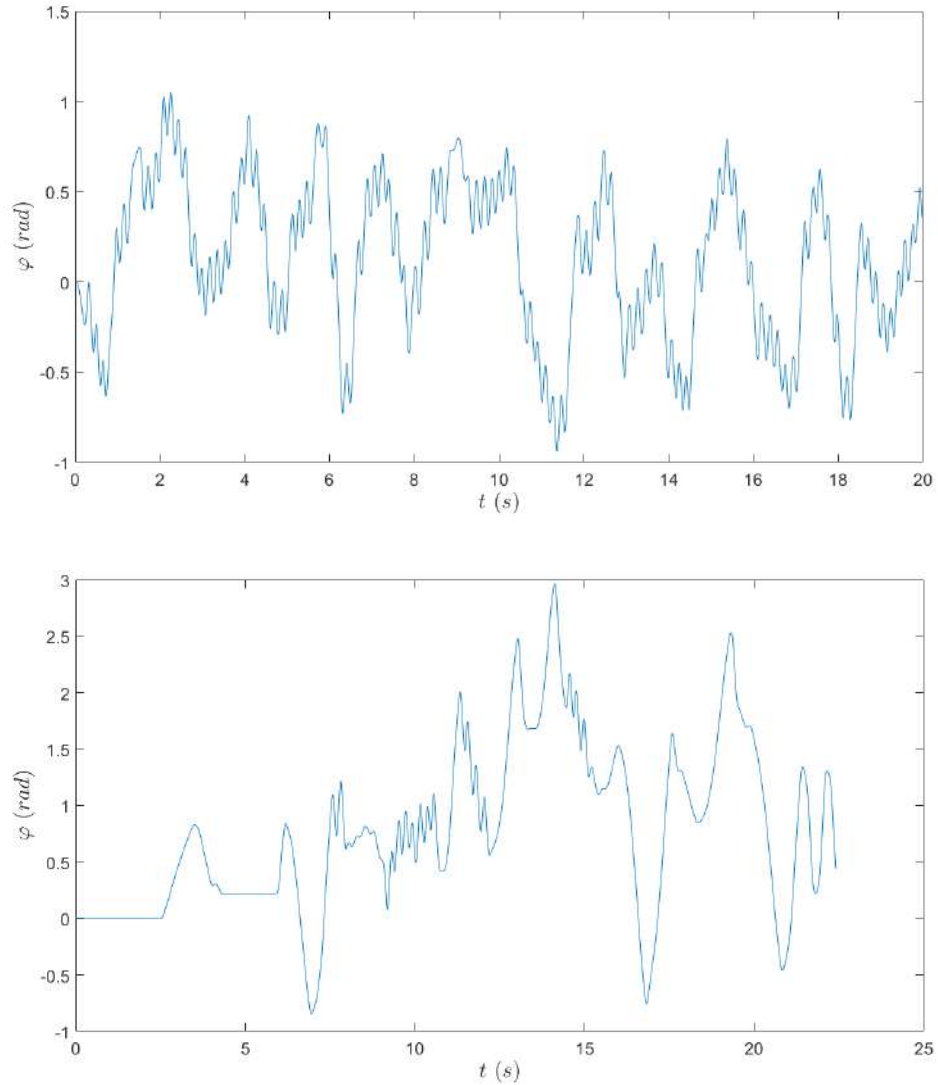


Figura 7.4: Evolución de  $\varphi$  durante la estabilización. En la figura de arriba se observa el control 1 y en la de abajo el control 2.

El controlador 2 no logra controlar  $\varphi$ . A pesar de esto ambos controladores logran mantener  $\dot{\varphi}$ , como se observa en la figura 7.5. Nuevamente el movimiento de  $\dot{\varphi}$  es más suave para el controlador 2 pero esta vez logra mantener el valor medio en  $0 \text{ rad/s}$ , como se verá en la tabla 7.2.

## Capítulo 7. Controladores

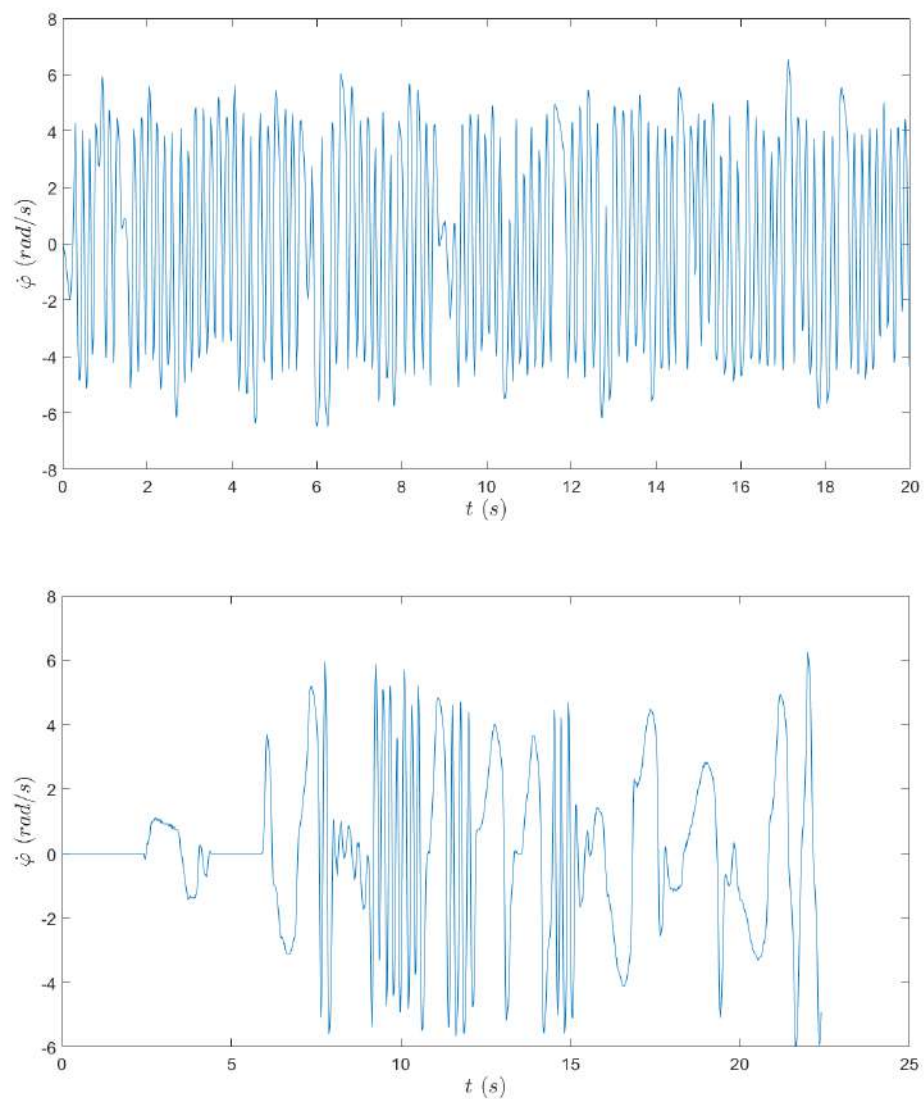


Figura 7.5: Evolución de  $\dot{\varphi}$  durante la estabilización. En la figura de arriba se observa el control 1 y en la de abajo el control 2.

En la figura 7.6 mostramos la evolución de la corriente. Recordamos al lector el verdadero carácter de la corriente (el cuál se presentó y fue discutido en 3.2.2) y que lo relevado está filtrado por un filtro RC con polo en  $100 \text{ Hz}$ .

## 7.1. Control estabilizante

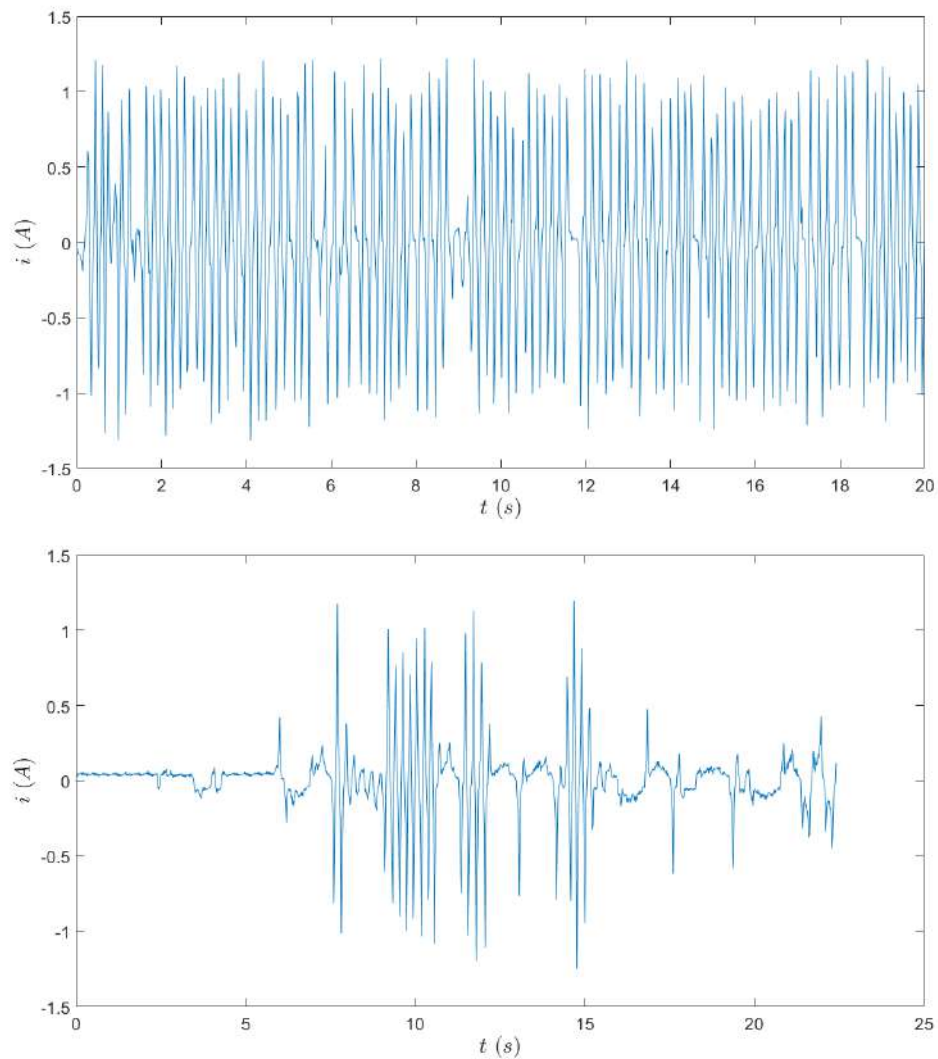


Figura 7.6: Evolución de  $i$  durante la estabilización. En la figura de arriba se observa el control 1 y en la de abajo el control 2.

Por último en la figura 7.7 mostramos la acción de control, es decir, el voltaje en bornes del motor, aplicada por cada uno de los controladores.

Además de las gráficas presentamos una serie de cálculos que indican qué *tan bien* cada controlador ha logrado su objetivo. Antes de presentar los cálculos explicamos qué medimos exactamente. Uno desea medir qué tanto se parece  $x(t)$  respecto a  $x_{eq}$ , para esto comparamos coordenada con coordenada. Una forma de medir esto es tomar la norma 2 del error, si  $x_i : [0, T] \rightarrow \mathbb{R}$  es una coordenada cuyo setpoint es  $x_{i0}$  entonces

## Capítulo 7. Controladores

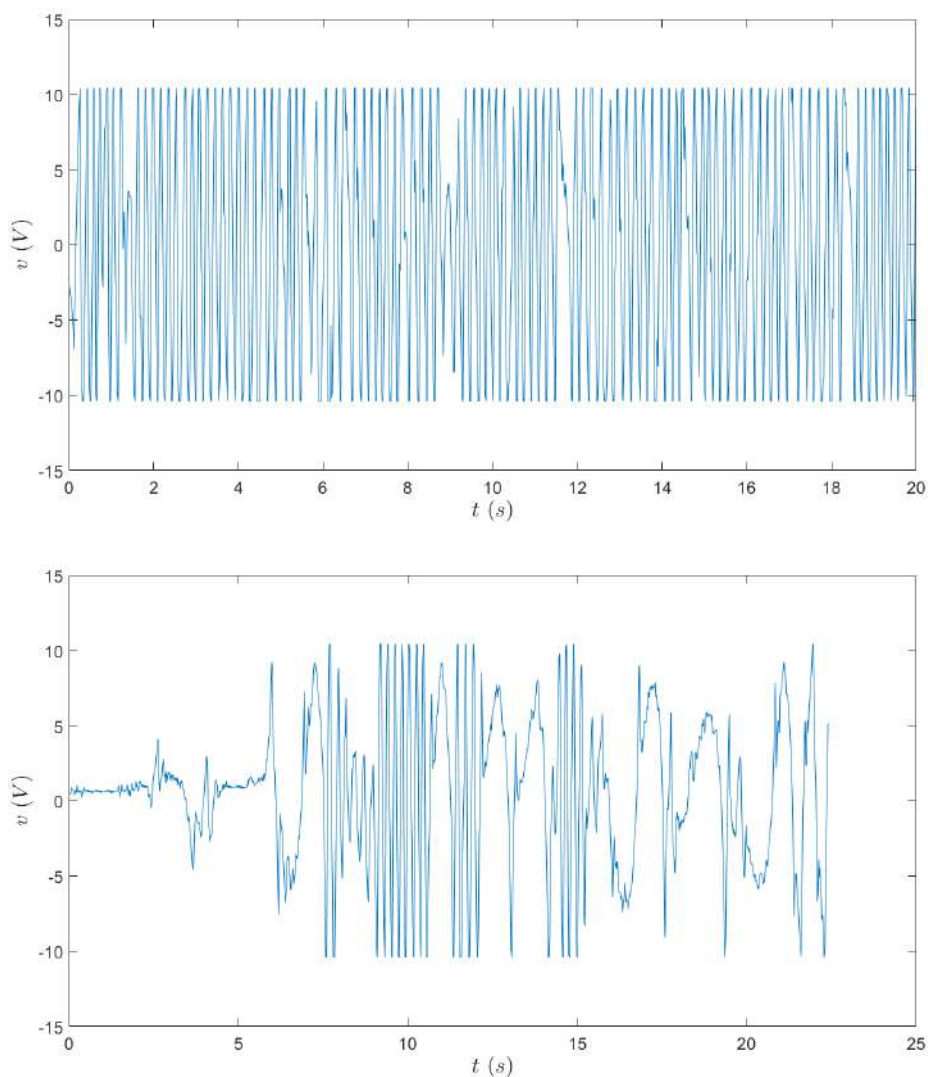


Figura 7.7: Evolución de  $v$  durante la estabilización. En la figura de arriba se observa el control 1 y en la de abajo el control 2.

$$\sqrt{\frac{1}{T} \int_{[0,T]} (x_i(t) - x_{i_0})^2 dt}$$

pero en realidad solo tenemos la información de  $x_i$  en los tiempos de muestreo  $\{0, \dots, t_n = T\}$ , por lo cual solo podemos integrar pesando esos tiempos de muestreo. Para esto hacemos que cada tiempo mida  $\frac{1}{n+1}$  de manera que  $\{t_0, \dots, t_n\}$  mida 1. De esta forma obtenemos una aproximación discreta de la ecuación anterior

$$\|x_i - x_{i_0}\|_2 = \sqrt{\frac{1}{n+1} \sum_{k=0}^n (x_i(t_k) - x_{i_0})^2} \quad (7.3)$$



## 7.1. Control estabilizante

A modo de ejemplo  $x_i$  podría ser el ángulo  $\theta$ , en cuyo caso  $x_{i_0} = \pi$ . Para todas las demás coordenadas  $x_{i_0} = 0$ . Esta manera de medir la diferencia con lo esperado es típica y permite varias interpretaciones. Además de verlo como la norma 2 se puede ver como el valor RMS del error. Otra interpretación surge de considerar  $\Omega = \{t_1, \dots, t_n\}$  junto con una medida de probabilidad  $\mathbb{P}$  uniforme y ver a  $x_i : \Omega \rightarrow \mathbb{R}$  como una variable aleatoria. Entonces la desviación estandar de dicha variable (considerando que la esperanza es  $x_{i_0}$ ) es

$$\sqrt{\int_{\Omega} (x_i - x_{i_0})^2 d\mathbb{P}} = \sqrt{\frac{1}{n+1} \sum_{k=0}^n (x_i(t_k) - x_{i_0})^2}$$

Por lo cual al medir la norma 2 de la diferencia entre  $x_i$  con su objetivo esencialmente estamos mirando la desviación estándar, o el valor RMS. Siguiendo con la interpretación como variable aleatoria otro valor interesante a medir es la esperanza. De manera análoga la esperanza coincide con el promedio de los valores relevados

$$\frac{1}{n+1} \sum_{k=0}^n x_i(t_k)$$

En resumen, para cada variable relevada calculamos el promedio y la norma 2 del error. Estas medidas dan una idea de rendimiento por coordenada.

En general un controlador que utiliza mayor señal de control es capaz de lograr mejores resultados. . Es por esto que es importante medir que tanta señal de control utiliza cada controlador. Nuevamente, utilizamos como costo la norma de la señal  $\|v\|$ , donde la norma se calcula de igual manera que en la ecuación 7.3.

$$\|v\| = \sqrt{\frac{1}{n+1} \sum_{k=0}^n v(t_k)^2}$$

De esta manera obtenemos la tabla 7.2.

Como se observa en la tabla para el control 2 se tiene un valor medio  $\varphi_0 = 0,781 \text{ rad}$  por lo cual no se logró estabilizar adecuadamente  $\varphi$ . Esta observación ya fue hecha al presentar el resultado gráfico. A pesar de esto, es importante mencionar que no siempre se obtiene un valor medio tan desfasado. A modo de ejemplo, en 7.3.3 se verá un relevo donde el error entre el objetivo y el valor medio de  $\varphi$  es del orden de  $0,05 \text{ rad}$ .

	Controlador 1		Controlador 2	
Variable	Promedio	$\ \text{error}\ $	Promedio	$\ \text{error}\ $
$\theta$ (rad)	3,14	0,114	3,15	$97,1 \times 10^{-3}$
$\dot{\theta}$ (rad/s)	$-8,33 \times 10^{-3}$	3,41	$-7,15 \times 10^{-3}$	1,67
$\varphi$ (rad)	0,117	0,444	0,781	1,08
$\dot{\varphi}$ (rad/s)	$13,6 \times 10^{-3}$	3,43	$20,0 \times 10^{-3}$	2,50
$i$ (mA)	-92	541	14,0	221
$v$ (V)	0,562	7,55	$-5,28 \times 10^{-3}$	4,84

Tabla 7.2: Promedio y desviación del objetivo para cada estado. Además se presenta la desviación de  $x$  de su objetivo.

### 7.1.6. Análisis

En esta subsección se estudian los resultados presentados en 7.1.5. El análisis se divide en dos partes principales. En la primera se comparan los resultados experimentales de los controladores contra la teoría y la simulación, analizando las diferencias. En la segunda parte se comparan los controladores entre sí. Es importante mencionar que el análisis llevado a cabo se basa en los dos relevos experimentales recientemente presentados por lo que no se puede concluir resultados estadísticos.

#### Comparando contra la teoría y la simulación

Lo más notable de los resultados es que los controladores no logran estabilidad asintótica en el sentido estricto, sino que logran mantener el estado en un entorno del objetivo, manteniendo una vibración que para algunas variables es pequeña pero para otras es mayor. Idealmente, esto es en el modelo del sistema que se derivó pero sin fricción, las simulaciones muestran cómo se logra estabilidad asintótica con una respuesta típica de un sistema lineal, para cualquiera de los dos controladores desarrollados. Además, esto es lo que dice la teoría pues  $A - BK$  es una matriz con valores propios de parte real negativa.

Al hacer simulaciones se ve que este resultado se mantiene si se agrega la fricción viscosa. Sin embargo, al agregar la fricción estática (lo cual corresponde a simular con el modelo completo) las simulaciones muestran que los controladores desarrollados no logran estabilidad asintótica y en cambio oscilan entorno al punto objetivo, como se observó en la realidad. Esto se cumple para todas las variables de estado. A efectos de ilustrar, en la figura 7.8 y 7.9 se observa la respuesta de  $\theta$  y  $\varphi$  de la simulación del controlador 2. Para el controlador 1 las simulaciones muestran resultados completamente similares, presentando oscilaciones que prácticamente no difieren en su amplitud y frecuencia de las presentadas en 7.8 y 7.9.

## 7.1. Control estabilizante

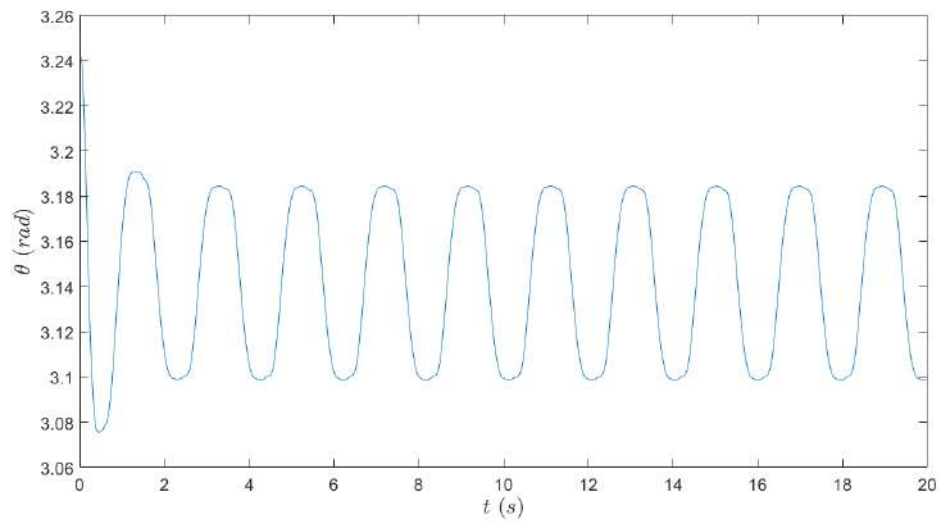


Figura 7.8: Simulación: evolución de  $\theta$  utilizando el controlador 2. La oscilación presenta una frecuencia de 0,5 Hz.

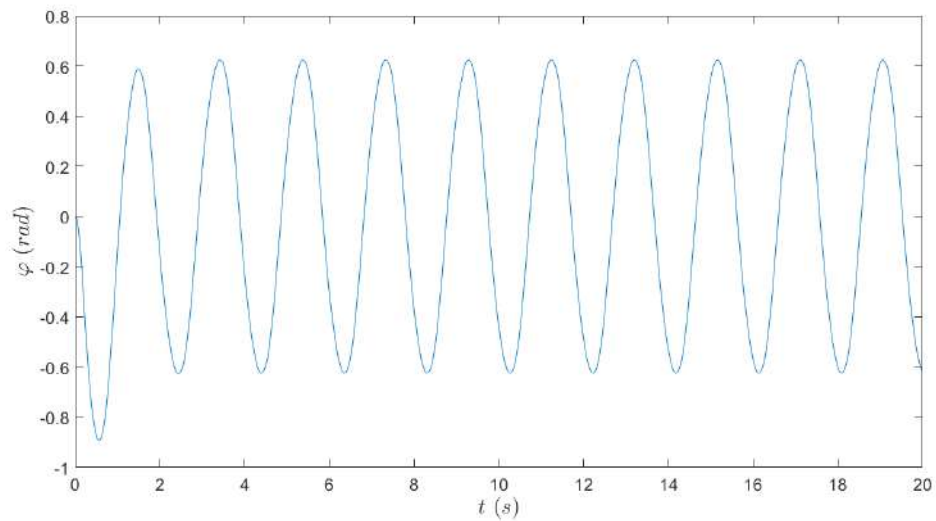


Figura 7.9: Simulación: evolución de  $\varphi$  utilizando el controlador 2. La oscilación presenta una frecuencia de 0,5 Hz.

Esto no solo ocurrió con los dos controladores exitosos sino con cualquier otro controlador diseñado. Vale mencionar que esto no es una contradicción con los resultados teóricos utilizados ya que al momento de diseñar los controladores se utilizó el modelo sin fricción. Creemos que la aparición de las oscilaciones en el simulador se deben a que la fricción estática aporta el efecto de zona muerta, como se vió en 5.3.1. Es un resultado conocido que una zona muerta en la realimentación de un sistema lineal puede causar la aparición de oscilaciones (ver [42], sec 10.4), por lo cual no es de extrañar que aquí aparezcan. A efectos de comparar esta

## Capítulo 7. Controladores

simulación con el relevo experimental se presenta la tabla 7.3.

Controlador 1

Variable	Promedio		error	
	Simulado	Relevado	Simulado	Relevado
$\theta$ (rad)	3,14	3,14	0,033	0,11
$\dot{\theta}$ (rad/s)	-0,0029	-0,0083	0,17	3,41
$\varphi$ (rad)	-0,012	0,12	0,25	0,44
$\dot{\varphi}$ (rad)	0,016	0,014	-1,07	3,43

Controlador 2

Variable	Promedio		error	
	Simulado	Relevado	Simulado	Relevado
$\theta$ (rad)	3,14	3,15	0,037	0,097
$\dot{\theta}$ (rad/s)	$-7,0 \times 10^{-3}$	$-7,2 \times 10^{-3}$	0,14	1,67
$\varphi$ (rad)	-0,021	0,78	0,4535	1,08
$\dot{\varphi}$ (rad)	-0,031	0,020	1,47	2,50

Tabla 7.3: Promedio y desviación del objetivo para las variables angulares.

Es claro que el error cuadrático mide la distancia de la señal al objetivo, el tamaño del error. Lo que no es obvio es que el error cuadrático también mide qué tanto vibra/oscila una señal, es decir, mientras más oscilaciones mayor error cuadrático. De hecho esto es un teorema, la información de las oscilaciones de una señal  $y$  es capturada por la transformada de Fourier de  $y$ , llamémosle  $\hat{y}$ . El teorema de Parsevall establece que el mapa  $y \mapsto \hat{y}$  es una isometría lineal<sup>3</sup>, es decir  $\|y\| = \|\hat{y}\|$ . Por lo tanto, la norma de una señal no solo mide el tamaño de la señal sino también el tamaño de sus oscilaciones. Teniendo esto en cuenta, de la tabla y de las figuras anteriores hay conclusiones bastante directas: las simulaciones justifican las oscilaciones que se presentaron pero no en su completitud. Más precisamente, las simulaciones muestran una oscilación en  $\varphi$ ,  $\theta$  y  $\dot{\varphi}$  con un error cuadrático del orden del encontrado en la práctica, pero menor. Que el error cuadrático sea menor es consistente, pues es claro de la simple inspección de las figuras que las simulaciones presentan menos oscilaciones. Por otro lado las simulaciones presentan un error cuadrático un orden menor para  $\dot{\theta}$ . A su vez también es claro que *la frecuencia* de las vibraciones no coinciden.

Para un estudio cuidadoso del contenido frecuencial de las señales relevadas  $\theta$  y  $\varphi$  conviene graficar su transformada de Fourier discreta, FFT. Para estudiar el contenido frecuencial de  $\theta$  se le resta el objetivo  $\pi$ . En otras palabras se estudia  $\widehat{\theta - \pi}$ . En la figura 7.10 y en la figura 7.11 se grafica  $|\widehat{\theta - \pi}|$  y  $|\widehat{\varphi}|$  respectivamente, donde  $\theta$  y  $\varphi$  provienen del relevo del controlador 1.

<sup>3</sup>prácticamente en cualquiera de las versiones de la transformada de Fourier, DFT (FFT), DTFT, en  $\mathcal{L}^2(\mathbb{R})$ , varios otros.

## 7.1. Control estabilizante

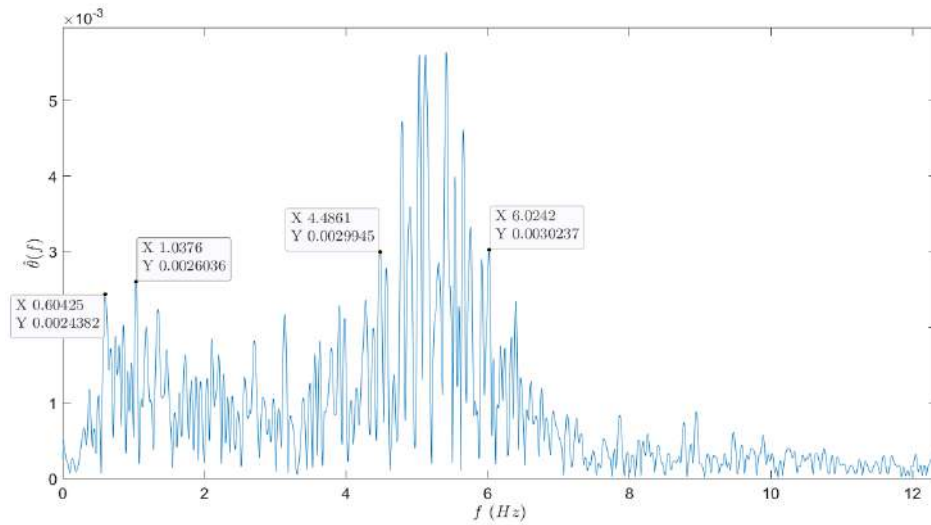


Figura 7.10: Se observa la transformada de Fourier discreta de  $\theta - \pi$ , donde  $\theta$  es el relevo del ángulo del péndulo para el controlador 1. En la figura se marcan ciertos puntos que delimitan las principales componentes frecuenciales de la señal.

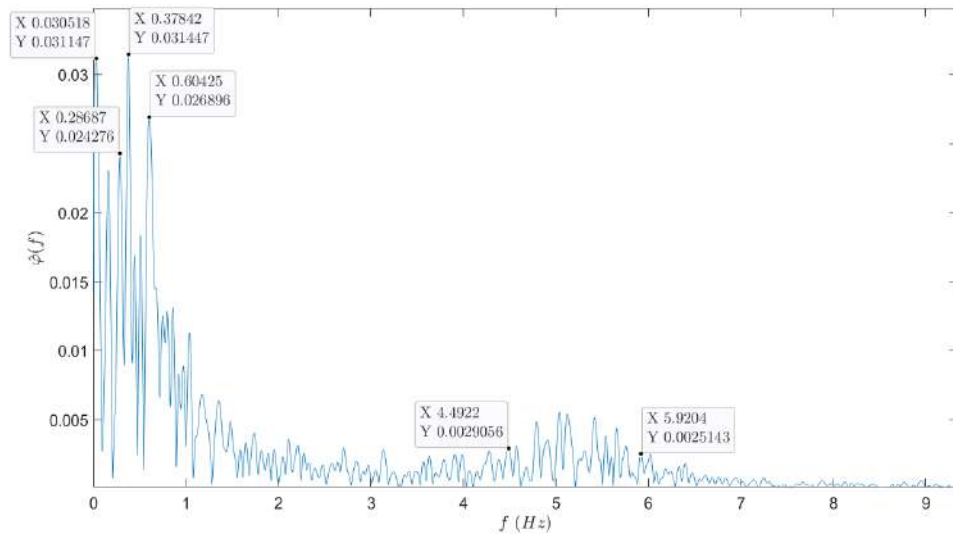


Figura 7.11: Se observa la transformada de Fourier discreta de  $\varphi$ , donde  $\varphi$  es el relevo del ángulo del motor para el controlador 1. En la figura se marcan ciertos puntos que delimitan las principales componentes frecuenciales de la señal.

Observando el contenido frecuencial de  $\theta$  se puede ver que hay dos componentes principales. Una entre  $4,49 \text{ Hz}$  y  $6,03 \text{ Hz}$  y luego una de menor frecuencia del orden de  $0,6 \text{ Hz}$  a  $1 \text{ Hz}$ . Observar que esta segunda componente tiene los valores frecuenciales del mismo orden de los observados en la simulación (figura 7.8 y figura 7.9), mientras que la componente de mayor contenido frecuencial no. Esto sugiere que la fricción estática causa la componente de baja frecuencia, mien-

## Capítulo 7. Controladores

tras que la componente de más alta frecuencia tiene otra causa que no es modelada.

Si observamos la respuesta en frecuencia de  $\varphi$  se concluye lo mismo solo que en este caso la oscilación predominante (mayor amplitud) es la de baja frecuencia. Además en este caso se puede apreciar una componente de muy baja frecuencia, asociado al valor de continua de  $\varphi$ . Finalmente, si se hace el mismo estudio para el controlador 2 los resultados son completamente similares con la diferencia de que la componente de alta frecuencia va desde  $3,4 \text{ Hz}$  hasta  $5 \text{ Hz}$  y tiene la mitad de amplitud.

Todo esto nos lleva a concluir que la fricción es una causa de las oscilaciones en el sistema, pero que no es la única causa de las oscilaciones. Creemos que el *juego* que presenta el eje del motor juega un rol fundamental. Las simulaciones logran explicar parte de las oscilaciones. Más precisamente, las simulaciones logran explicar gran parte del error cuadrático y también del contenido frecuencial, ambas medidas de la vibración. Concluimos que la fricción estática es causante de las vibraciones con contenido frecuencial del orden de  $0,5 \text{ Hz}$ , como lo muestra la simulación, y que quizás el juego sea el causante de las vibraciones de mayor contenido frecuencial ( $\approx 5 \text{ Hz}$ ). Aun así, es claro que no podemos dejar de considerar otros posibles efectos como aportantes a la causa de las oscilaciones (tanto de baja frecuencia como de alta), por ejemplo, los engranajes del motor, la naturaleza de la corriente del driver y/o otra dinámica no modelada. Como se puede ver en el apéndice F, sobre el final del proyecto apareció nueva evidencia que afirma la conclusión de que las simulaciones explican las oscilaciones de baja frecuencia lo cual apoya aun más el hecho de que la fricción sea causante de estas componentes. No utilizamos dicha evidencia aquí pues no se analizó en profundidad, por quedar fuera del alcance del proyecto.

La otra diferencia con lo esperado es que el controlador 2 no logra estabilizar adecuadamente  $\varphi$ . Más específicamente, a veces el controlador lo logra (como se verá en 7.3.3, figura 7.28b) y a veces no lo hace (como en 7.1.5, figura 7.4). En las figuras 7.8 y 7.9 se pueden observar los resultados de simular el controlador 2. Como se puede ver el simulador no predice el problema de la controlabilidad de  $\varphi$ .

Para ganar un poco de intuición sobre donde puede estar la causa analizamos brevemente la ley de control para el control 2. El coeficiente que multiplica a  $\varphi$  es  $-0,69 \text{ V/rad}$ . El valor medio de  $\varphi$  relevado experimentalmente es  $\varphi_0 = 0,781 \text{ rad}$ . Si suponemos que  $\varphi$  se encuentra cerca de  $\varphi_0$  y todas las demás variables están perfectamente estabilizadas en su objetivo entonces la ley de control es prácticamente  $v \approx 0,69\varphi_0 = 0,54 \text{ V}$ . Ahora este voltaje se encuentra dentro de la zona muerta del motor (hasta  $0,8 \text{ V}$ ) y por lo tanto el motor no actuará para compensar este desvío.

Un análisis como el anterior no prueba nada pero revela que puede tratarse de un problema causado por la zona muerta y por lo pequeño que es el coeficiente de  $\varphi$  para el control 2. Es importante observar que a pesar de tener un modelo con

zona muerta el mismo no detecta este problema por lo cual es totalmente plausible que la causa de esto también se deba a otros efectos no modelados, nuevamente, el *juego* que posee el eje del motor, la dinámica de la corriente, la dinámica de los engranajes u otros ni siquiera conocidos.

Otra gran diferencia, ya comentada en varias oportunidades y siendo sumamente esperada, es la corriente. Idealmente, es decir en el modelo derivado sin fricción, la corriente se estabiliza asintóticamente en cero. Ya al agregar la fricción estática aparecen las oscilaciones ya mencionadas. Aún así estas oscilaciones en la corriente no son similares a las que se presentan en la corriente relevada. Esto es más que esperable recordando una vez más que nuestro modelo utiliza una fuente de tensión ideal pero en la práctica contamos un driver PWM el cuál provoca una onda de corriente de 1  $kHz$ . Dicha onda es filtrada por un filtro RC y eso es lo adquirido por el sistema, por lo cual la corriente esta lejos de provenir de una fuente de tensión ideal.

Dejando de lado las diferencias anteriores, obtenemos los resultados que predijo la simulación, controladores estabilizantes en el sentido amplio. Más precisamente, controladores que logran mantener la variable de estado de principal interés,  $\theta$ , en un entorno suficientemente pequeño del objetivo, con un valor medio idéntico al objetivo. La simulación además acertó, a excepción de  $\varphi$  para el control 2, la estabilidad en el sentido amplio para el resto de las variables. Más aún, las vibraciones observadas en la realidad se ven justificadas, en gran parte, por las simulaciones con el modelo desarrollado e identificado en capítulos anteriores. Las simulaciones predicen las componentes de baja frecuencia de las señales y también gran parte del error cuadrático. Esto demuestra que el modelo construido no solamente ha sido útil para el desarrollo de controladores sino también para el entendimiento de los resultados.

### Comparando los dos controladores

Cómo comparar dos controladores es una pregunta difícil, sobre todo si la comparación tiene como objetivo determinar cuál es la mejor estrategia de control, pues es difícil diseñar un test que sea independiente de los parámetros de los controladores. Por suerte aquí la estrategia de control es la misma y estamos comparando dos controladores ya ajustados. Aun así, a la hora de comparar hay varias dimensiones a evaluar y cada una de estas dimensiones tiene múltiples formas de ser evaluada. Distintos aspectos de interés pueden ser: distancia al objetivo, especificaciones en el dominio temporal, especificaciones en frecuencia, energía empleada, esfuerzo de control, robustez, rechazo a las perturbaciones, complejidad del algoritmo, sencillez para ajustarlo, requerimientos de hardware, zonas de atracción, y otros. Aquí nos limitamos a responder dos preguntas típicas, ¿Cuál logra mejor el objetivo? y ¿Cuál utiliza menos señal de control? Para responder/evaluar estas preguntas utilizamos el error cuadrático, entonces para responder la primer pregunta evaluamos cuál mantiene las variables,  $x_i$ , a menos distancia del objetivo en norma 2, y para responder la segunda cuál ejerce una señal de control  $v$  con menos norma.

## Capítulo 7. Controladores

De la simple inspección de las gráficas presentadas en la sección 7.1.5 se puede ver que quién mantiene las variables más cerca del objetivo (a excepción de  $\varphi$ ) es el control 2. Además es claro también que dicho controlador utiliza menos acción de control. Para tener números precisos están los resultados de la tabla 7.2, donde se observa cuantitativamente cómo las variables de estado se desvían menos al utilizar el controlador 2, a excepción de  $\varphi$  que además tiene un valor medio *equivocado*. Por último, en la misma tabla se puede ver cómo el control 2 utiliza *la mitad* de la entrada que utiliza el control 1.

Que el control 2 tenga menos error cuadrático en casi todas las variables y además utilice menos señal de control no es de extrañar. El controlador 2 fue diseñado para lograr polos de lazo cerrado que son más pequeños que los del control 1, de esta forma necesita menos señal de control y así se confirma al comparar los coeficientes de los dos controladores (ver tabla 7.1): esencialmente todos los coeficientes del control 2 son más pequeños que el control 1. El primer coeficiente de la corriente aumentó un 24 %, luego el coeficiente de  $\theta$  disminuyó en un 30 %, el coeficiente de  $\dot{\theta}$  también disminuyó en un 30 %. En cambio, para  $\varphi$  el coeficiente se redujo un 60 %. Por último el coeficiente de  $\dot{\varphi}$  se redujo en un 20 %.

Es por este cambio en los coeficientes que la acción de control es menor y la respuesta más suave. Otro justificativo que apoya esta conclusión es que la respuesta frecuencial de las variables de interés son de menor frecuencia para el controlador 2 que para el 1, y de menor amplitud. Sin embargo, la suavidad ha sido tal que el control 2 tiene problemas para controlar  $\varphi$  adecuadamente. Justamente el coeficiente de  $\varphi$  es el que más ha cambiado, se redujo al 40 % de su valor original.

Es claro que quién utiliza menos señal de control es el control 2. Lo segundo que está claro es que si no fuese por el problema del control 2 para estabilizar  $\varphi$  adecuadamente entonces el control 2 es mejor que el control 1 para lograr el objetivo, pues es quién dista menos al objetivo. Si somos estrictos con las medidas realizadas el problema con la estabilización de  $\varphi$  deja por fuera de competencia al control 2, mostrando que el control 1 es el más adecuado para cumplir el objetivo para el cuál fue diseñado, la estabilización. Con los ensayos relevados y las medidas realizadas creemos que no es posible responder cuál controlador cumple mejor el objetivo, quién dista menos del objetivo. Creemos que es necesario realizar más ensayos y considerar otras métricas, entre otros.

En resumen, se ha respondido a la pregunta de cuál controlador usa menos señal de control, pero no se logró decidir cuál es el mejor a la hora de mantener las variables cerca del objetivo, pues uno acierta mejor con el valor medio pero presenta un error cuadrático mayor que el otro. Es importante mencionar que si uno pretende decidir cuál es el *mejor* control entonces no solo es necesario responder esta última pregunta, también es importante ensayar otros aspectos.



## 7.2. *Swing Up*

El término *Swing Up* se utiliza típicamente para referirse a un controlador capaz de transferir un péndulo de su posición de reposo a la posición invertida. En este caso la posición de reposo corresponde a  $\theta = 0 \bmod 2\pi$ ,  $\dot{\theta} = 0$ ,  $\varphi = \varphi_0 \bmod 2\pi$  y  $\dot{\varphi} = 0$ , y la posición invertida a  $\theta = \pi \bmod 2\pi$ ,  $\dot{\theta} = 0$ ,  $\varphi = \varphi_1 \bmod 2\pi$  y  $\dot{\varphi} = 0$ .

A diferencia de la estabilización del péndulo en la posición invertida, el problema del *swing up* no puede lograrse utilizando un control lineal. De hecho, se puede ver que las técnicas más comunes del control no lineal no funcionan, ver [43]. En la literatura se encuentran diversas técnicas para levantar el péndulo como por ejemplo plantear el diseño como un problema de optimización, *bang-bang state feedback*, *vibrational input* y *energy control*, ver [1, 21, 44].

La estrategia elegida aquí consiste en diseñar un *swing up* para el péndulo simple utilizando *energy control* y luego adaptar dicho controlador al péndulo de Furuta. Las referencias consultadas son [1, 43–48] pero principalmente [44, 46], en quienes nos basamos para el desarrollo del control.

En la sección 7.2.1 se comenta la metodología llevada a cabo para el desarrollo e implementación de una ley de control. En 7.2.2 se explica la técnica *energy control* y se aplica al diseño de un *swing up* para el péndulo simple. En la sección 7.2.3 se deriva una ley de control para el péndulo de Furuta. En 7.2.4 se realizan simulaciones para respaldar la ley. En 7.2.5 se realizan pruebas y se ajusta el control y finalmente en la sección 7.2.6 se muestran los resultados experimentales.

### 7.2.1. Metodología

En esta sección se explican los pasos seguidos para la elección de una ley de control capaz de lograr el *swing up* de manera robusta. A diferencia de la metodología llevada a cabo para el diseño del control estabilizante, la cual se basó fuertemente en simulaciones, aquí las simulaciones se usan en una instancia inicial para verificar la elección de la ley de control, pero luego el ajuste de dicho control se realizó mediante ensayos en el sistema. Se siguió la siguiente metodología de trabajo:

#### 1. Estudio de las referencias

Aquí se estudió principalmente el *swing up* para el péndulo simple que propone [44]. Para esto se realizaron simulaciones, se evaluaron parámetros, etc. Luego se estudió [46], donde se presenta un *swing up* del péndulo de Furuta basado en las ideas de [44].

## Capítulo 7. Controladores

### 2. Diseño de una primer ley de control para el Furuta

Basados en el estudio anterior se deriva una primer ley para el péndulo de Furuta. La ley es una adaptación de la ley para el péndulo simple que propone [44], similar a lo hecho por [46]. Aquí el resultado en sí no es una ley de control sino una estructura general para una ley de control, donde se deja cierta libertad, más allá de los parámetros. Específicamente se consigue una ley  $v = \Phi(x, u(x))$  donde  $x$  es el estado del péndulo y  $u(x)$  es una ley de *swing up* para el péndulo simple (hay varias posibles).

### 3. Simulaciones

Tomando una  $u(x)$  particular se simula la ley propuesta en el paso anterior, chequeando la validez de la adaptación propuesta. Para esto se utiliza una ley de  $u$  propuesta por [44].

### 4. Pruebas en el sistema

En este paso se comienza tomando una  $u(x)$  particular y se ajusta sus parámetros en base a pruebas. En base a los resultados de la práctica se va modificando  $u$  hasta llegar a un control apropiado, en el sentido de que logre el *swing up* de manera robusta.

## 7.2.2. Energy control y el péndulo simple

El objetivo del *swing up* es lograr llegar a un punto de equilibrio inestable. Para hacer esto, la técnica energy control hace uso de ciertas órbitas especiales, las llamadas órbitas homoclínicas. Una *órbita homoclínica* es una orbita del flujo de un sistema dinámico que parte de un punto de equilibrio inestable y vuelve al mismo<sup>4</sup>. En la figura 7.12 se ilustra este concepto.

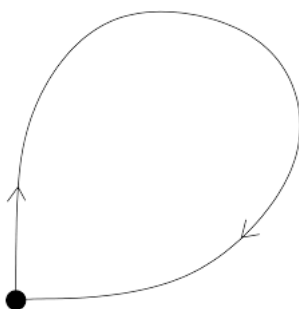


Figura 7.12: Ilustración de una órbita homoclínica.

---

<sup>4</sup>Más precisamente, la órbita tiende al punto de equilibrio asintóticamente hacia el pasado y hacia el futuro,  $\lim_{t \rightarrow \pm\infty} x(t) = x_{eq}$ .

Si uno desea diseñar una política de control que lleve el estado al punto de equilibrio inestable una estrategia razonable es hacer que el sistema se acerque a una órbita homoclínica. Una vez cerca la dinámica natural del sistema llevará el estado a las cercanías del punto de equilibrio. Si la órbita homoclínica queda dada por la curva de nivel de la energía del sistema, para lograr llegar a esta órbita especial lo que se hace es diseñar un control que maneje la energía del sistema, haciendo converger la misma al valor de energía que posee la órbita.

Veamos esta técnica aplicada al péndulo simple sin fricción, donde la entrada  $u$  es la aceleración horizontal en el pivot. La dinámica de la planta se puede derivar de diversas maneras, por ejemplo, una forma eficaz es usando un marco de referencia no inercial, acompañando el pivot.

$$J\ddot{\theta} + mgl \sin \theta + mul \cos \theta = 0 \quad (7.4)$$

donde  $J$  es la inercia del péndulo,  $m$  la masa,  $l$  la longitud del pivot al centro de masa,  $u$  la aceleración horizontal en el pivot y  $\theta$  es el ángulo del péndulo tomando  $\theta = 0$  como la posición de reposo. La energía del péndulo esta dada por

$$E = \frac{J}{2}\dot{\theta}^2 - mgl(\cos \theta + 1)$$

En la figura 7.13 se observa el diagrama de fase del péndulo con entrada  $u$  nula, junto con la curva de nivel 0 de  $E$ .

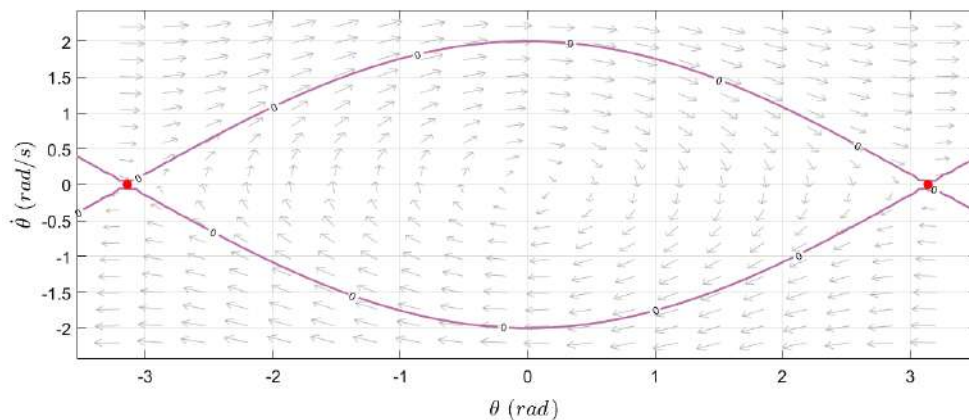


Figura 7.13: Diagrama de fase del péndulo simple sin controlar. En rojo se marca el objetivo y en punteado la curva de nivel 0 de la energía, la cual contiene una órbita homoclínica.

El objetivo consiste en partir desde un punto cualquiera, típicamente cercano al  $(0, 0)$  y llegar al punto de equilibrio inestable, marcado en rojo en la figura. Recordar que los dos puntos marcados en rojo son en realidad el mismo. Notamos del diagrama de fase una órbita homoclínica<sup>5</sup> contenida en  $\{x \mid E(x) = 0\}$ . Entonces la idea es diseñar  $u$  que cumpla que el sistema realimentado tenga la misma

<sup>5</sup>En realidad son dos, una para velocidades positivas y otras para velocidades negativas.

## Capítulo 7. Controladores

órbita homoclínica pero que además logre hacer que  $E \rightarrow 0$ . Si esto ocurre el sistema realimentado se acercará a la órbita homoclínica y luego la misma hará que pase cerca del objetivo. Para lograr esto consideramos una función de Lyapunov  $V : \mathbb{R}^2 \rightarrow \mathbb{R}$  dada por  $V = E^2/2$ . Entonces calculando la derivada de  $V$  a lo largo de las orbitas, derivada de Lyapunov, se obtiene

$$\dot{V} = E\dot{E} = E(J\ddot{\theta} + mgl \sin(\theta)\dot{\theta}) = -Euml\dot{\theta} \cos \theta$$

Esto permite entender cómo debe ser  $u$  para *empujar* en el sentido correcto, en el sentido que lleva  $E$  a 0. Hay varias  $u$  que logran  $\dot{V} \leq 0$ , a modo de ejemplo veamos dos. Si tomamos

$$u = kE\dot{\theta} \cos \theta \quad (7.5)$$

entonces se obtiene  $\dot{V} = -mlk(E\dot{\theta} \cos \theta)^2 \leq 0$ . Ahora si uno desea cambiar la energía lo más rápido posible la señal  $u$  debe ser lo más grande posible. Esto se puede obtener utilizando la ley de control

$$u = u_m \text{sign}(E\dot{\theta} \cos \theta) \quad (7.6)$$

donde  $u_m$  es el máximo posible de la señal de control.

Que  $\dot{V} \leq 0$  no quiere decir que el objetivo se cumpla, sin embargo así lo es. Esto se puede demostrar utilizando el teorema de *LaSalle*, o también con otros caminos, ver [44]. Aquí nos conformamos con apreciar el resultado en las figuras 7.14 y 7.15. Como se puede ver, para la ley de control 7.5 mientras mayor sea  $k$  más rápido sube el péndulo, lo cual es lo esperado. Con  $k = 0,3$  el péndulo realiza varias oscilaciones antes de llegar al objetivo, mientras que con  $k = 1$  logra subir con un solo *swing*<sup>6</sup>.

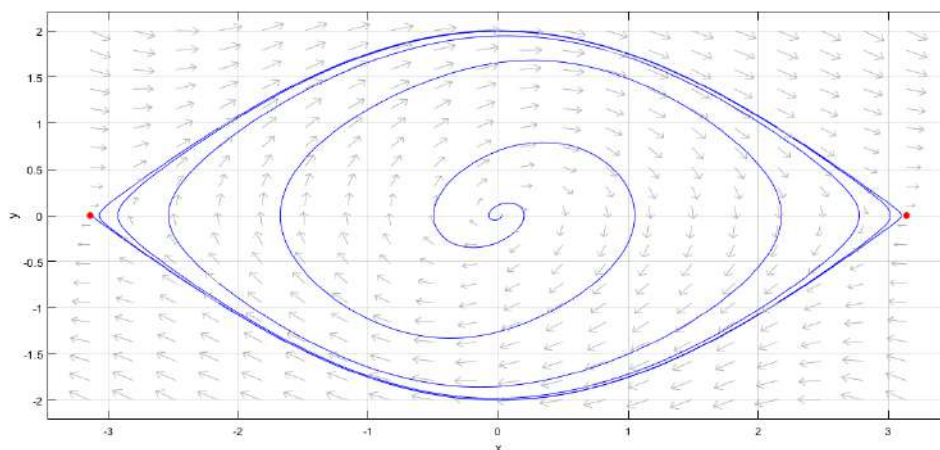


Figura 7.14: Diagrama de fase del sistema controlado. *Swing up* del péndulo simple con  $u$  dada por 7.5 y  $k = 0,3$ . Se bosqueja además una órbita típica.

<sup>6</sup>Columpio o balanceo. De ahora en más se utilizará *swing*.

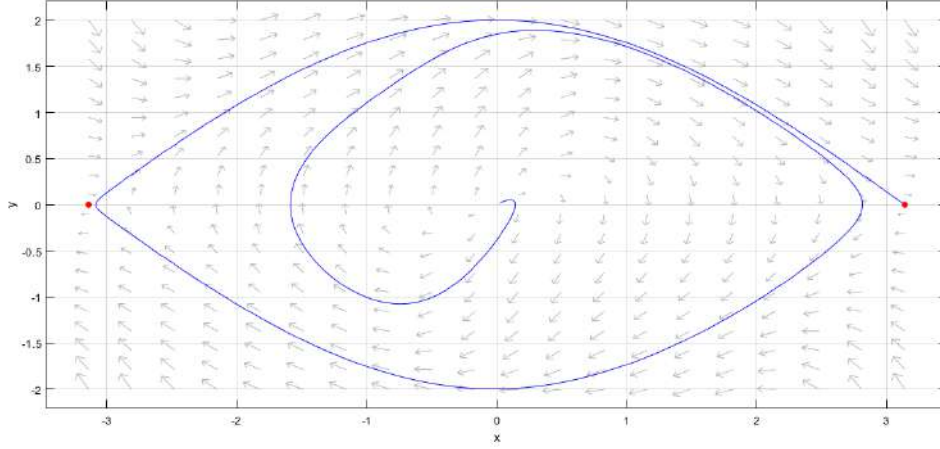


Figura 7.15: Diagrama de fase del sistema controlado. *Swing up* del péndulo simple con  $u$  dada por 7.5 y  $k = 1$ . Se bosqueja además una órbita típica.

En cualquiera de los casos se puede ver cómo la órbita se acerca primero a la órbita homoclínica y luego la misma la hace pasar cerca del objetivo cada cierto tiempo. De esta forma se construye un control de *swing up* para el péndulo simple.

### 7.2.3. *Swing up* del péndulo de Furuta

Si se sigue estrictamente la estrategia anterior deberíamos buscar una órbita homoclínica para el dinámica del péndulo de Furuta. Esto, a priori, se vuelve más complicado que el caso anterior ya que tenemos una dinámica de dimensión 4. Para el lector interesado en explorar este camino se puede consultar [47]. El enfoque utilizado aquí es utilizar el *Swing up* del péndulo simple. Recordamos que la dinámica del péndulo de Furuta se rige por la siguiente ODE

$$\begin{aligned} \ddot{\varphi} (J + 2m_2 l_{bi} c_x + m_2 l_{bi}^2 + I_z + I_x \sin^2 \theta) \\ + 2\dot{\varphi} \dot{\theta} I_x \cos \theta \sin \theta + m_2 l_{bi} c_z \ddot{\theta} \cos \theta - m_2 l_{bi} c_z \dot{\theta}^2 \sin \theta = \tau + \tau_{fu} \end{aligned} \quad (7.7)$$

$$I_x \ddot{\theta} + m_2 l_{bi} c_z \ddot{\varphi} \cos \theta - \dot{\varphi}^2 I_x \cos \theta \sin \theta + m_2 g c_z \sin \theta = \tau_{fp} \quad (7.8)$$

donde estamos considerando  $\tau$  como entrada. Para el siguiente diseño nos olvidamos de la fricción. Observemos que la aceleración lineal en el pivot del péndulo es  $l_{bi} \ddot{\varphi} e_\varphi$ , por lo tanto, inspirados en el péndulo simple, llamemosle  $u = l_{bi} \ddot{\varphi}$  e imaginemos que podemos manejar  $u$ . Entonces la segunda ecuación queda

$$I_x \ddot{\theta} + m_2 g c_z \sin \theta = -m_2 c_z u \cos \theta + \dot{\varphi}^2 I_x \cos \theta \sin \theta \quad (7.9)$$

que coincide con la ecuación del péndulo simple 7.4 a menos del término  $\dot{\varphi}^2 I_x \cos \theta \sin \theta$  que podemos ver como una perturbación. Para lograr un *swing up* que lleve  $(\theta, \dot{\theta})$  al objetivo diseñamos  $u$  de la forma  $u = u_{sw} + u_2$  donde  $u_2$  se

## Capítulo 7. Controladores

utiliza para cancelar la perturbación y  $u_{sw}$  es un *swing up* para el péndulo simple, por ejemplo cualquiera de los presentados en la sección previa. Entonces

$$u = u_{sw} + u_2 = u_{sw} + \frac{\dot{\varphi}^2 I_x \sin \theta}{m_2 c_z}$$

Al colocar esta ley de control en 7.9 se obtiene

$$I_x \ddot{\theta} + m_2 g c_z \sin \theta = -m_2 c_z u_{sw} \cos \theta$$

que es el péndulo simple con un controlador de *Swing up*, por lo tanto  $(\theta, \dot{\theta})$  se moverá hacia el objetivo, logrando el *Swing up*. Ahora queda la pregunta, ¿es posible manejar  $u$ ? La respuesta es afirmativa, la clave de esto es que  $\tau$  entra de manera afin en las ecuaciones de variable de estado. En efecto, de la ecuación de  $\ddot{\varphi}$  se tiene

$$\begin{aligned} \ddot{\varphi} &= \frac{1}{d(\theta)} \left( I_x (m_2 l_{bi} c_z \sin \theta \dot{\theta}^2 - 2 I_x \dot{\theta} \dot{\varphi} \sin \theta \cos \theta) \right. \\ &\quad \left. - m_2 l_{bi} c_z \cos \theta (\dot{\varphi}^2 I_x \cos \theta \sin \theta - m_2 g c_z \sin \theta) + I_x \tau \right) \\ &= \frac{1}{d(\theta)} (f(x) + I_x \tau) \end{aligned}$$

por lo cual eligiendo  $\tau = \frac{d\ddot{\varphi} - f(x)}{I_x}$  se logra imponer  $\ddot{\varphi}$  y por lo tanto controlar  $u$ . Entonces el valor de  $\tau$  para imponer el  $u$  deseado es

$$\begin{cases} \tau = \frac{d(\theta)}{I_x l_{bi}} u + 2 I_x \dot{\theta} \dot{\varphi} \sin \theta \cos \theta - m_2 l_{bi} c_z \sin \theta \dot{\theta}^2 \\ \quad + \frac{m_2 l_{bi} c_z}{I_x} \cos \theta (\dot{\varphi}^2 I_x \cos \theta \sin \theta - m_2 g c_z \sin \theta) \\ u = u_{sw} + u_2 = u_{sw} + \frac{\dot{\varphi}^2 I_x \sin \theta}{m_2 c_z} \end{cases} \quad (7.10)$$

**Una ley de voltaje** La ley 7.10 logra un *Swing up* del péndulo de Furuta pero utilizando como entrada el torque del motor  $\tau$  cuando en verdad la señal de control es el voltaje del motor  $v$ . Para obtener una ley de control que utilice el voltaje existen alternativas, las más conocidas se llaman *backstepping* y *sliding mode controller* o control deslizante [42].

En este proyecto se intentó utilizar el control deslizante pero no se tuvo éxito. En este caso el control deslizante consistía en diseñar un control que haga que el torque  $\tau$  converja al control deseado, llamemosle  $\Phi(\theta, \dot{\theta}, \varphi, \dot{\varphi})$ , el cual está dado por 7.10. Para hacer esto se define el error  $s = \tau - \Phi(\theta, \dot{\theta}, \varphi, \dot{\varphi})$  y se calcula  $\dot{s}$ . Se puede probar que se puede elegir  $v$  de forma tal que  $s$  converja a cero en tiempo finito. Típicamente la forma del control es  $v = v_{eq} - \lambda \text{sign}(s)$  donde  $v_{eq}$  logra  $\dot{s} = 0$  y la función signo hace que la derivada sea negativa. Para demostrar que esto funciona es necesario utilizar una función de Lyapunov y el lema de comparación de ecuaciones diferenciales. Como este método no funcionó no desarrollamos más sobre el mismo, el lector interesado puede consultar [42] o [49].

Vale mencionar que el sliding mode controller si bien quedó complejo, funcionó sin problemas en las simulaciones, donde se veía cómo  $s \rightarrow 0$  y a partir de ese momento se activaba el *swing up*, el cual levantaba el péndulo sin problemas. En la práctica se observaba que  $s$  no convergía a cero. También se podía ver cómo el controlador exigía un voltaje mayor al máximo del motor, por lo cual el voltaje estaba saturado. Creemos que ésta es la razón de mal funcionamiento. Además no se logró tunear el controlador para que no sature el voltaje y que además el *swing up* funcioné. Respecto a esto último, como las simulaciones lo prevén, el *swing up* debe tunearse de manera de tener la suficiente *fuerza* como para vencer dinámica no modelada como la fricción.

Por suerte una idea mucho más simple funcionó en la práctica, la idea fue ignorar la dinámica del motor y tomar como voltaje de control la ley de torque 7.10. En definitiva considerar

$$\begin{cases} v = H \left( \frac{d(\theta)}{I_x l_{bi}} u + 2I_x \dot{\varphi} \sin \theta \cos \theta - m_2 l_{bi} c_z \sin \theta \dot{\theta}^2 \right. \\ \left. + \frac{m_2 l_{bi} c_z}{I_x} \cos \theta (\dot{\varphi}^2 I_x \cos \theta \sin \theta - m_2 g c_z \sin \theta) \right) \\ u = u_{sw} + u_2 = u_{sw} + \frac{\dot{\varphi}^2 I_x \sin \theta}{m_2 c_z}, \end{cases} \quad (7.11)$$

Donde  $H$  es una constante con las dimensiones adecuadas. Esta elección, la de olvidar la dinámica del motor, no es una mala idea, sino todo lo contrario. Hay tres razones para esto. La primera es que la relación entre torque y voltaje es, intuitivamente, directa. La segunda es que varios autores suelen olvidar la dinámica del motor [46]. La tercera es que las simulaciones muestran que la ley 7.11 funciona, tomando como  $u_{sw}$  cualquiera de las derivadas en la sección 7.2.2. Vemos esto en la siguiente sección.

#### 7.2.4. Simulaciones

En esta sección probamos la ley de control 7.11 utilizando como  $u_{sw}$  una leve variante de 7.6. Utilizamos la ley propuesta por [44]

$$u_{sw} = \text{sat}_{u_m} \left( k E \text{sign}(\dot{\theta} \cos \theta) \right) \quad (7.12)$$

donde  $\text{sat}_{u_m}$  es una función lineal de pendiente 1 que satura en  $\pm u_m$ . Este cambio es porque el control 7.6 puede provocar *chattering*<sup>7</sup> entorno al objetivo  $E = 0$ . Igualmente se cambia  $\text{sign}$  por  $\tanh(20 \cdot)$ . Además de lo mencionado se utiliza  $u_2 = 0$ , dejando la perturbación sin cancelar. Para la simulación se utiliza el modelo completo desarrollado en el capítulo 3 e identificado en el capítulo 4, incluyendo las fricciones y la dinámica del motor. Además a la entrada del motor se coloca una saturación a los 10,4 V, correspondiente al voltaje máximo con el que se cuenta en la práctica.

<sup>7</sup>Las funciones discontinuas en ecuaciones diferenciales pueden provocar oscilaciones.

## Capítulo 7. Controladores

Los resultados son buenos. Para valores suficientemente grandes de  $k$  y  $u_m$  el péndulo logra subir. Mientras mayor  $u_m$  más rápido sube el péndulo. Cuando los valores no son suficientes el péndulo logra oscilar hasta una posición intermedia. Todo esto es lo esperado. En el caso de  $u_m$ , intuitivamente mientras mayor sea su valor más fuerza tendrá el controlador por lo cual mejor enfrentará a la fricción o podrá superar a la perturbación no cancelada. Por el lado de  $k$ , es la pendiente de la saturación. El que el control sea una saturación implica que cuando  $E$  está cerca de 0 entonces el controlador se achica, pues el mismo se encontrará en la parte lineal. Esto le saca fuerza al control perdiendo las bondades del punto anterior. Qué tan cerca debe estar  $E$  de 0 para que este efecto aparezca depende en  $k$ . Mientras mayor  $k$  más cerca debe estar  $E$  de 0 para que esto ocurra. Es por esto que para valores pequeños de  $k$  el controlador no logra su objetivo, porque el control pierde fuerza y no puede superar efectos no modelados como la fricción.

En resumen, se verifica el funcionamiento de la ley de control 7.11. A modo de ejemplo presentamos los resultados de una simulación con parámetros  $k = 100000 \text{ s}^2/\text{kgm}^2$ ,  $u_m = 20 \text{ m/s}^2$  y  $H = 1 \text{ V/Nm}$ .

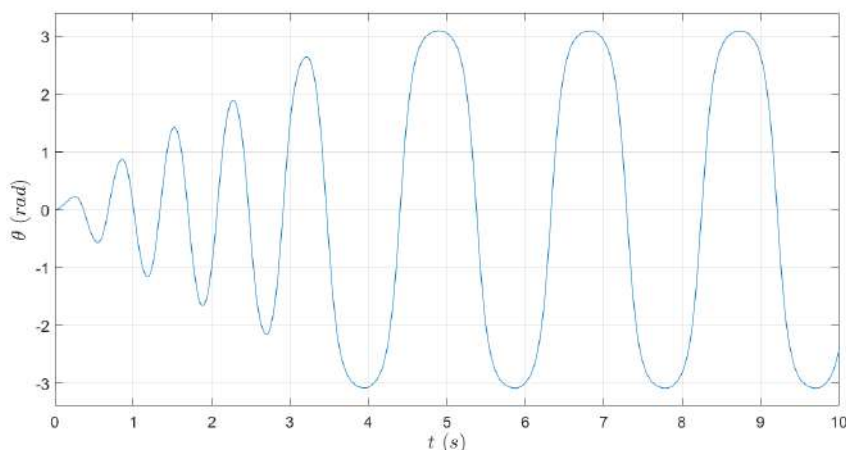


Figura 7.16: Evolución de  $\theta$ .

Como se observa en la figura 7.16, la variable  $\theta$  empieza a oscilar aumentando su amplitud y luego a partir de cierto  $t_0$  se mantiene con amplitud aproximadamente  $\pi$ . En la figura 7.17 se puede ver como la energía evoluciona de manera monótona hasta llegar prácticamente a cero, el momento en el cual esto ocurre se corresponde con  $t_0$ . A partir de ahí la órbita de  $(\theta, \dot{\theta})$  se mantiene cerca de la órbita homoclínica del péndulo simple, y por lo tanto pasa cerca del objetivo de manera recurrente. En la figura 7.18 observamos la dinámica de  $(\theta, \dot{\theta})$  con un diagrama de fase en el plano. Observar la similitud con la figura 7.14.

Recordar que a efectos prácticos  $\theta = \pi$  es equivalente a  $\theta = -\pi$ . Para obtener una imagen más fiel de la dinámica observada en lugar de utilizar  $\mathbb{R}^2$  como espacio de estados se puede utilizar uno donde se identifique los  $\theta$  que difieran  $2k\pi$ . Intui-



## 7.2. Swing Up

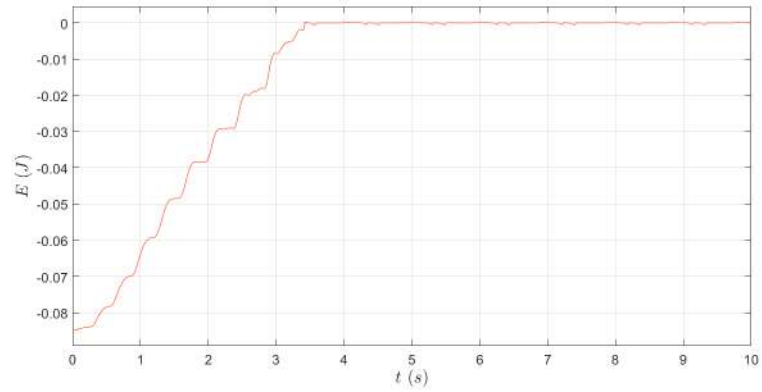


Figura 7.17: Evolución de la energía del péndulo simple.

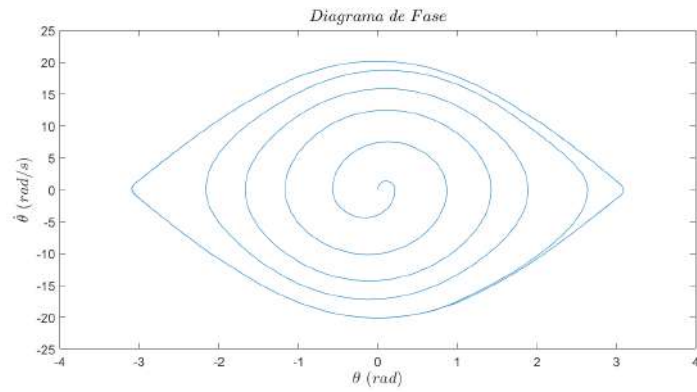


Figura 7.18: Diagrama de fase de  $(\theta, \dot{\theta})$ .

tivamente, una forma de hacer esto es tomar el diagrama de fase de la figura 7.18 y enrollar el mismo en un cilindro de radio 1. El mapa  $(\theta, \dot{\theta}) \mapsto (\cos \theta, \sin \theta, \dot{\theta})$  realiza dicha identificación. Entonces aplicando dicho mapa al diagrama de la figura 7.18 se obtiene un diagrama de fase en el cilindro.

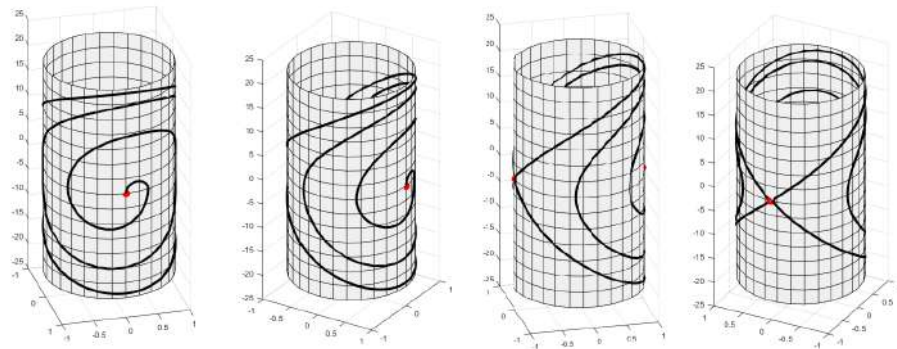


Figura 7.19: De izquierda a derecha desde el origen al objetivo.

### 7.2.5. Pruebas y elección del controlador

La ecuación 7.11 provee una ley de control para lograr el *Swing up* del péndulo de Furuta, donde tenemos varias posibilidades para  $u_{sw}$ . Las dos opciones más sencillas son las mostradas en 7.5 y 7.6. Ambas leyes de control empujan el péndulo en el momento adecuado. Se comenzó probando la ley 7.6 pues la misma *empuja* con la máxima fuerza posible y en este sentido es óptima.

Para probar 7.6 tuvimos el cuidado de utilizar su variante 7.12 para evitar el chattering. En la práctica es necesario ajustar el valor de  $k$  y el valor de  $u_m$ . Se fueron ajustando experimentalmente pero no se logró un *swing up* correcto. El péndulo logró hacer swings y aumentar su amplitud como lo muestra 7.16 pero cuando  $|\theta| > \theta_0 > \pi/2$  entonces el controlador hizo que el péndulo empezará a vibrar bruscamente. Esto impidió que la amplitud siguiera aumentando para llegar al objetivo.

Una inspección de la acción de control muestra que la causa de dicha vibración es que el control cambiaba entre  $u_m$  y  $-u_m$  bruscamente en el momento incorrecto. El signo del control  $u$  está dado por el signo de  $E\dot{\theta} \cos \theta$  por lo tanto cuando  $|\theta| > \theta_0$  y el péndulo se encuentra subiendo el signo de  $\dot{\theta} \cos \theta$  no cambia por lo cual el cambio de signo violento está dado por  $E$ . Teóricamente la energía parte desde  $E = -0,0842 J$  y crece de manera monótona hasta  $E = 0$  por lo cual mantiene su signo. Sin embargo, en la práctica cuando la energía se encuentra en el entorno de  $-0,02 J$  aparecen picos, como si se tratase de ruido, que alteran el signo de  $E$ . Ver figura 7.22. En estos cambios de signos  $u$  alterna entre  $\pm u_m$  cuando no debería. Creemos que ésta es la causa de falla de esta ley.

La siguiente ley a probar fue 7.5. Dicha ley mostró ejercer su mayor fuerza cuando  $\theta \approx 0$  y luego prácticamente no actuó. Esto es consistente ya que en dicha posición  $\cos \theta \approx 1$ ,  $|E|$  esta en su máximo valor y  $|\dot{\theta}|$  también. Luego, a medida que el péndulo comienza a subir  $|\dot{\theta}|$  y  $|E|$  empiezan a disminuir y por lo tanto  $u_{sw}$  también. Esto eliminó las vibraciones pero el control era muy suave, empujando el péndulo prácticamente solo en  $\theta = 0$ . En cuanto a resultados, el péndulo comenzaba a hacer *swings* aumentado su amplitud pero llegado cierto umbral el péndulo queda oscilando prácticamente manteniendo su amplitud hasta que finalmente, en varios casos, lograba subir. Por lo tanto el *swing up* no quedó muy robusto.

Se concluyó que se necesitaba una ley de control que tuviera un poco de los dos, es decir, que atenuara su acción cerca de  $\theta = \pi \bmod 2\pi$  y que actuará lo más posible en  $|\theta| < \pi/2 \bmod 2\pi$ , y no solamente en el entorno de  $0 \bmod 2\pi$ .

Para conseguir dicha ley recordamos que necesitamos cumplir con  $\dot{V} \leq 0$  y que  $u_{sw}$  mantenga la órbita homoclínica del péndulo simple. Las leyes dadas por 7.5 y 7.6 cumplen estas condiciones. Si definimos una función  $f : \mathbb{R} \rightarrow \mathbb{R}^+ \cup \{0\}$  entonces la ley  $u_{sw} = kf(\theta)E\text{sign}(\dot{\theta} \cos \theta)$  satisface estas condiciones. En efecto

$$\dot{V} = E\dot{E} = -Euml\dot{\theta} \cos \theta = -mlkf(\theta)E^2\text{sign}(\dot{\theta} \cos \theta)\dot{\theta} \cos \theta \leq 0$$

También  $u_{sw}|_{\{E=0\}} = 0$ . Además un detalle técnico es pedirle a  $f$  que *no se anule demasiado*. Aquí utilizaremos  $f$  anulándose solo en  $\theta = \pi \text{ mod } 2\pi$ . La filosofía de  $f$  es que permite ponderar la acción de control. De esta forma podemos elegir  $f$  para que la ley de control atenué su acción cerca de  $\theta = \pi \text{ mod } 2\pi$  y que actúe fuertemente en  $|\theta \text{ mod } 2\pi| < \pi/2$ . En algunos artículos se utiliza  $|\theta - \pi|^n$  como función ponderante. Nosotros utilizamos una versión periódica para no distinguir puntos que corresponden a la misma configuración del péndulo, precisamente utilizamos  $f(\theta) = (\text{arc cos}(\cos(\theta - \pi)))^3$ , cuyo bosquejo se muestra en 7.20.

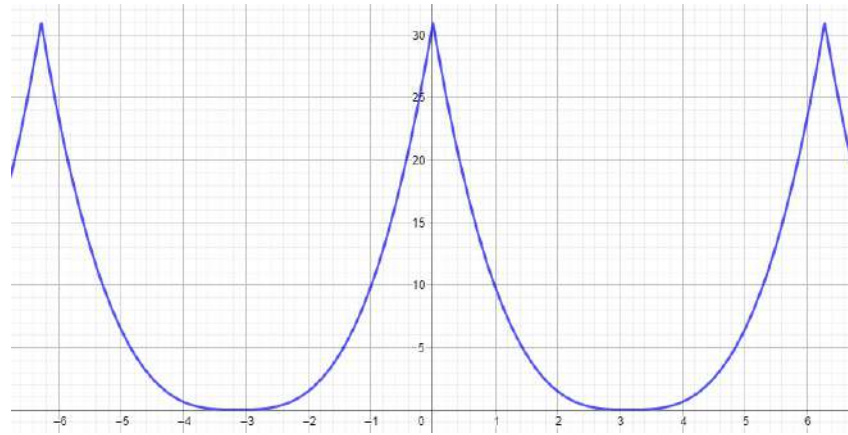


Figura 7.20: Bosquejo de la función  $f(\theta)$  utilizada.

Resumiendo, el control propuesto es

$$\begin{cases} v = H \left( \frac{d(\theta)}{I_x l_{bi}} u + 2I_x \dot{\theta} \dot{\varphi} \sin \theta \cos \theta - m_2 l_{bi} c_z \sin \theta \dot{\theta}^2 \right. \\ \left. + \frac{m_2 l_{bi} c_z}{I_x} \cos \theta (\dot{\varphi}^2 I_x \cos \theta \sin \theta - m_2 g c_z \sin \theta) \right) \\ u = u_{sw} = k(\text{arc cos}(\cos(\theta - \pi)))^3 E \tanh(20\dot{\theta} \cos \theta) \end{cases} \quad (7.13)$$

Donde se ha cambiado la función  $\text{sign}(a)$  por  $\tanh(20a)$ . Un control como el de 7.13 ha demostrado tener la estructura correcta para ser utilizado en la práctica, debiendo ajustar solamente el parámetro  $k$  y  $H$ , pudiendo elegir estos para cumplir el objetivo.

### 7.2.6. Resultados y análisis

Presentamos aquí los resultados de ensayar en el sistema la ley de control dada por 7.13. Como se mencionó en la sección anterior los ensayos muestran que la ley de control propuesta es capaz de llevar  $\theta$  a  $\pi \text{ mod } 2\pi$  mediante una adecuada elección de  $k$ .

## Capítulo 7. Controladores

Teóricamente, esto es, en el péndulo simple sin fricción cualquier valor de  $k$  es suficiente para realizar el *swing up*, si uno elige un valor pequeño simplemente el sistema irá ganando energía de a poco, ganando amplitud en su oscilación y eventualmente subirá, aunque deba pasar años para que esto ocurra. También teóricamente, un valor muy grande de  $k$  simplemente hara que el péndulo se eleve en un solo swing, pero el mismo llegará suavemente a su objetivo sin importar que tan grande sea  $k$ . Estos hechos no pueden estar más lejos de la realidad. En la práctica uno cuenta con dinámica no modelada por ejemplo la fricción, que no fue tomada en cuenta, el término de perturbación mencionado en 7.2.3, y otros efectos no contemplados como puede ser la dinámica de los engranajes o el juego del motor. Es bastante intuitivo que si  $k$  es pequeño el sistema no vencerá a la fricción y entonces no podrá subir. Esto efectivamente se observa en la práctica. También se observa que si  $k$  es suficientemente grande entonces el péndulo llega a  $\pi$  con cierta velocidad. El resto de las diferencias con la realidad lo iremos viendo resultado por resultado.

Los ensayos experimentales muestran que para  $k < 35$  no se logra el *swing up* de manera robusta. Mostramos aquí algunos resultados para  $k = 37$ . En la figura 7.21 se observa la órbita proyectada en  $(\theta, \dot{\theta})$  del sistema controlado por 7.13 partiendo del reposo.

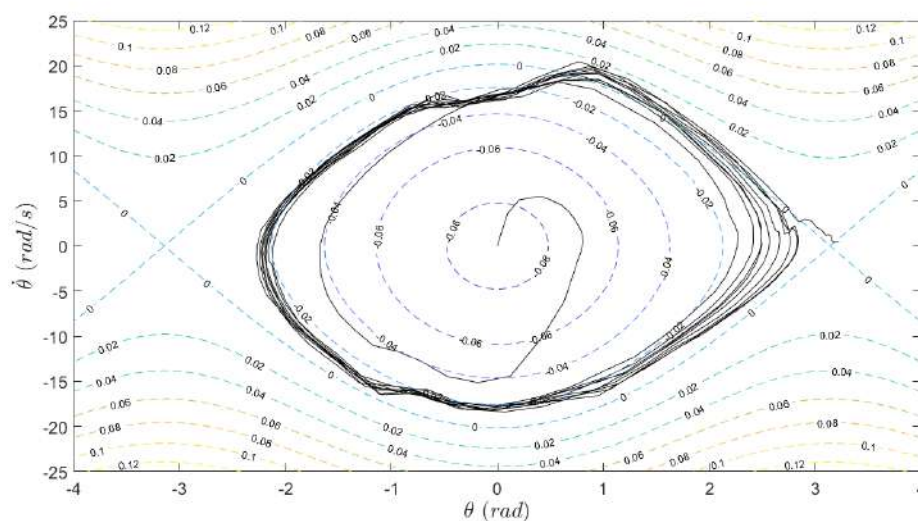


Figura 7.21: *Swing Up*, dinámica de  $(\theta, \dot{\theta})$  junto con las curvas de nivel de energía.

Como se observa en el diagrama de fase, el sistema evoluciona manteniendo  $E$  casi monótona creciente, como es de esperar. Se puede observar que la órbita demora más en salir del nivel  $-0,02 J$ , pero luego de un par de órbitas se sale llegando al nivel  $0 J$ . Una vez en este nivel se llega rápido al punto de equilibrio. Este es la primer distinción con lo simulado. La evolución de la energía también puede visualizarse en la gráfica 7.22 donde se puede ver cómo al sistema le cuesta más salir del nivel  $-0,02 J$ . Además, en esta gráfica se puede ver que la energía

cuenta con ciertos *picos* y que cuando se llega al nivel  $-0,02 J$  dichos picos superan el nivel del  $0 J$ . Al superar dicho nivel, el control  $u_{sw}$  cambia de signo cuando al sistema aún le falta para llegar a su objetivo, *empujando* hacia el lado incorrecto. Es aquí donde es fundamental el efecto atenuante de  $f$ ,  $u_{sw}$  efectivamente cambia de signo pero su magnitud ya es despreciable. Observar que ya la propia energía  $E$  atenúa  $u_{sw}$  al estar cerca del objetivo, sin embargo en la práctica no fue suficiente, necesitando agregar  $f$ .

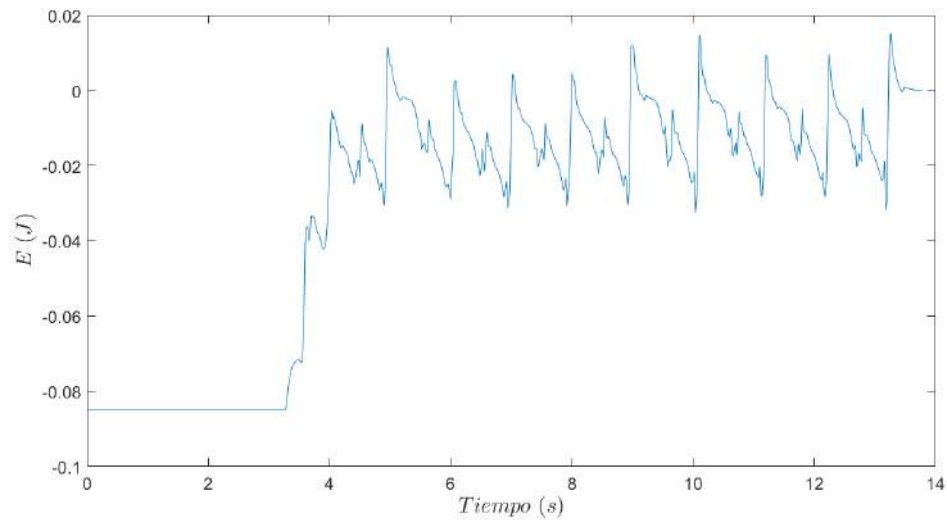


Figura 7.22: *Swing Up*, evolución de la energía.

La aparición de los picos fue reproducida utilizando el simulador y el modelo desarrollado en los capítulos anteriores. Para reproducir los picos se agregó incertidumbre a los parámetros que aparecen en la ley de control. Al hacer esto aparecen los mismos picos. Sin embargo, aún así el sistema lograba subir a  $\pi \bmod 2\pi$  sin detenerse en el nivel  $-0,02 J$ .

Otra forma de hacer aparecer los picos fue al simular el filtro que calcula la velocidad a partir de la salida del encoder. Si el filtro se desviaba mucho en fase de  $\pi$  a las frecuencias de interés, el derivador introducía un error de fase que desincronizaba la velocidad calculada de la velocidad verdadera. Este desfase era suficiente para introducir los picos. En la práctica se probó levantar el polo del filtro y efectivamente se notaron cambios, pero menores.

En la figura 7.23 se puede ver la evolución de  $\theta$  con el tiempo. En la misma se puede ver cómo el péndulo comienza a ganar amplitud hasta cierto valor cercano al objetivo, el cual permanece un par de segundos. Esto se corresponde a la energía manteniéndose en  $E = -0,02 J$ . Finalmente el sistema logra superar dicha amplitud llegando a  $\pi$ .

Otro aspecto importante a señalar que difiere con la teoría es que sucede una

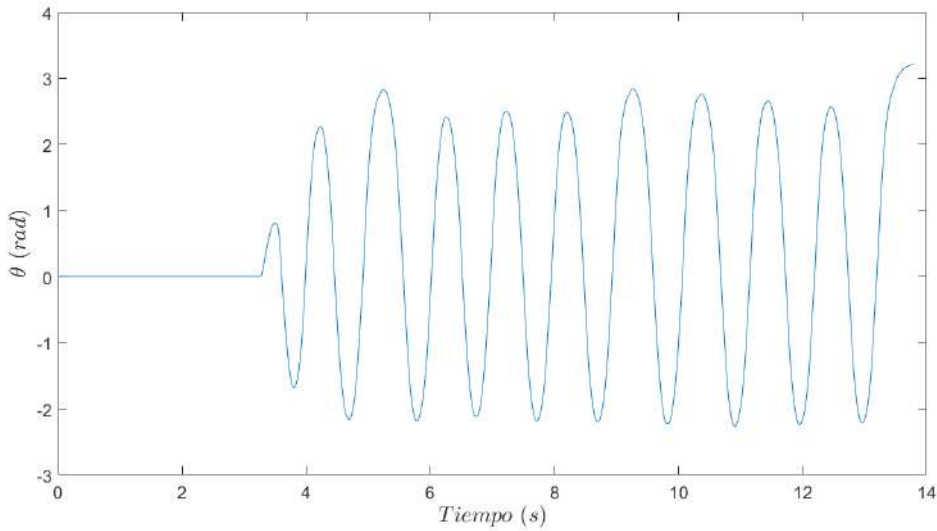


Figura 7.23: *Swing Up*, evolución de  $\theta$ .

vez el *swing up* ha llegado al objetivo. En las gráficas anteriores no se muestra esto. Teóricamente una vez el estado se encuentra en la órbita homoclínica el mismo no debe salirse de la misma y por lo tanto pasará recurrentemente por  $\theta = \pi \text{ mod } 2\pi$  y  $\dot{\theta} = 0$ . Sin embargo esto no ocurre. Lo que sucede es lo que se muestra en la figura 7.24. La energía no se mantiene en los entornos de  $E = 0 \text{ J}$  sino que vuelve a bajar a  $E = -0,02 \text{ J}$ . Luego nuevamente vence esta barrera y vuelve a subir, pero no logra mantenerse en la órbita homoclínica. Esto no es extraño, si bien no es algo predicho por las simulaciones perfectamente puede deberse a dinámica no modelada.

Finalmente en la figura 7.25 se puede ver el resultado de tomar  $k = 50$ . Como se observa el péndulo comienza haciendo solo 3 swings para lograr pasar cerca del objetivo. Precisamente pasa por  $\pi \text{ mod } 2\pi$  con velocidad  $\dot{\theta} = -4,37 \text{ rad/s}$ . Luego de esto el sistema pierde energía hasta llegar a  $-0,02 \text{ J}$  aproximadamente y luego vuelve a subir, esta vez pasando con una velocidad de  $-0,09 \text{ rad/s}$ , la cual es despreciable. Una primera observación entonces es el cambio de comportamiento al utilizar un mayor  $k$ , como es de intuir, el péndulo sube con menos swings cuando parte del reposo pero pasa con mayor velocidad. Observar que esto no es un problema, no se necesita pasar exactamente por el objetivo sino entrar en la zona de atracción del control estabilizante en la posición invertida. La segunda observación es que prácticamente no se detiene en  $-0,02 \text{ J}$  antes de pasar por primera vez por el objetivo. Por lo tanto, utilizar  $k = 50$  puede ser una buena estrategia pues si bien tiene la desventaja de que puede pasar más rápido por  $\theta = \pi \text{ mod } 2\pi$  tiene la ventaja de no detenerse en  $-0,02 \text{ J}$ . Además la desventaja desaparece si la zona de atracción del control estabilizante es suficientemente grande.

## 7.2. Swing Up

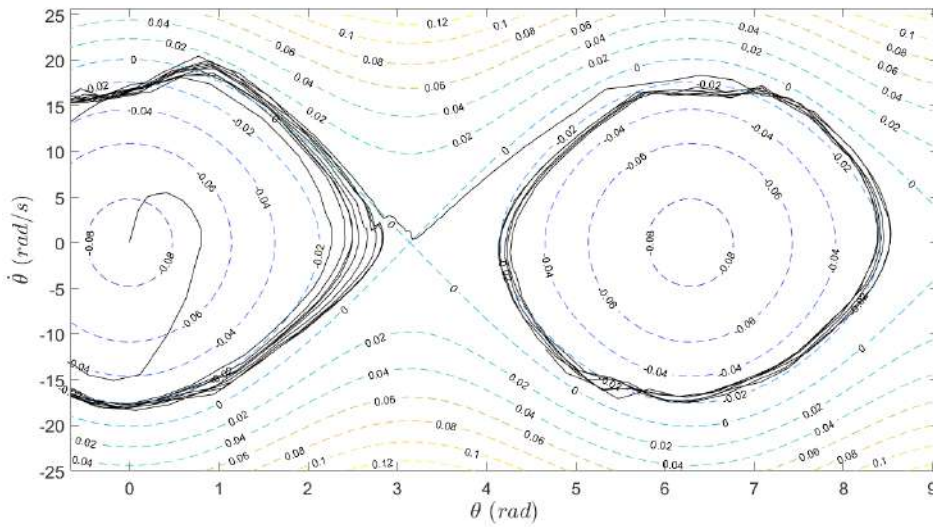


Figura 7.24: Dinámica inmediatamente luego del *Swing up*. Se utilizó  $k = 37,5$

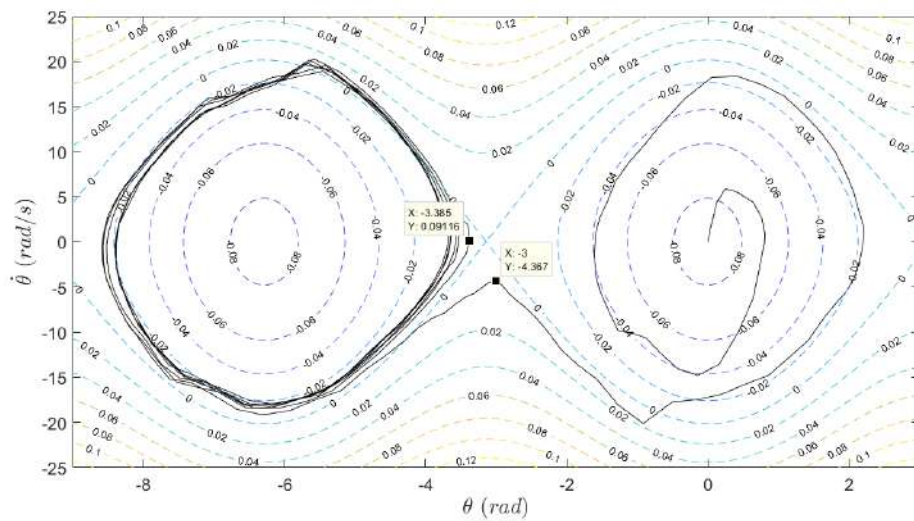


Figura 7.25: *Swing Up*, dinámica de  $(\theta, \dot{\theta})$ . Se utilizó  $k = 50$ .

Sobre las diferencias con la teoría cuando  $E = 0$ , tanto como para cuando  $E$  se *detiene* en  $-0,02 J$  como para cuando la órbita no se logra mantener en  $E = 0$  o cuando el controlador se pasa de su objetivo: creemos que lo que puede estar ocurriendo puede venir del hecho que  $f$  atenúe tanto que la acción de control  $v$  cae dentro de la zona muerta, siendo equivalente a que  $f$  se anule. De esta manera se pierden las propiedades de convergencia aseguradas para el péndulo simple, se pierde controlabilidad cerca de  $E = 0$ . También creemos que la fricción tiene su aporte. Sin embargo, ninguna de estas posibles causas se estudiaron en detalle, un estudio más profundo escapa al alcance del proyecto.

### 7.3. *Swing up*, control estabilizante y control supervisor

En esta subsección se combinan los dos controladores anteriores junto con un control supervisor, obteniendo como resultado un controlador que levanta el péndulo desde su posición de equilibrio hacia su posición invertida y luego lo estabiliza en dicha posición. El control supervisor es una máquina de estados que se encarga de hacer los cambios entre los controladores de manera adecuada y de hacer las elecciones que sean necesarias. Además hace tareas extras como evitar que el péndulo se enrede con el cable del encoder, achicar tiempos muertos y pasar algunos parámetros a los controladores. Para esto cuenta con dos controladores adicionales: el *desenredador* y el control estabilizante de bajada.

El principal desafío consistió en lograr una transición entre el *swing up* al control estabilizante. En las subsecciones anteriores se mostraron ambos controladores funcionando de forma independiente. En el caso del *swing up* se estudió en qué condiciones llega el péndulo a la posición invertida y para el control estabilizante se estudió su rendimiento partiendo cerca del punto de equilibrio.

Para realizar el control supervisor se decidió implementar esencialmente la misma máquina de estados mostrada en 6.2.3. La función de este control no solo es decidir qué controlador aplicar en cada momento sino que también elige el setpoint para los controles estabilizantes, más precisamente elige el valor de  $\varphi$  a seguir. La máquina de estados entonces funciona como control supervisor, eligiendo la ley de control. Cuando el sistema se esté por enredar el control supervisor le pasa el control al desenredador, el cual devuelve el péndulo a su posición inicial. Luego se activa el control estabilizante de bajada de manera de evitar que el péndulo oscile libremente, reduciendo tiempos muertos.

#### 7.3.1. Control estabilizante de bajada y desenredador

Para evitar que el péndulo se enrede se implementó un controlador que se llamó desenredador. En 6.2.1 se presentó una implementación de este controlador como un seguidor de posición. Su ley de control es  $u = -k(\varphi - \varphi_0)$ . Para desenredar el péndulo se utiliza  $\varphi_0 = 0$ . Por lo tanto se obtiene:  $u = -k\varphi$ .

El sistema comienza desde el reposo, es decir todas sus variables en 0. Cuando se quiere volver a esta condición de inicio, porque el péndulo se enredó, dejar que el péndulo oscile libremente consume muchos segundos para llegar a la condición de reposo. Para evitar estas oscilaciones se utiliza el control estabilizante de bajada. Para diseñar el controlador se utilizó la linealización del modelo del sistema en el punto deseado (todas las variables en 0). La linealización del sistema en torno a la posición de reposo es controlable, por esto se optó por utilizar un *State Feedback Control* para determinar los polos del sistema. Se comenzó probando los mismos polos que para el control estabilizante y se observaron buenos resultados, luego de algunas pruebas los polos se fijaron en  $\lambda_1 = -20$  ,  $\lambda_2 = -12$  ,  $\lambda_3 = -10$  ,  $\lambda_4 = -8$  ,  $\lambda_5 = -6$  . En la figura 7.26 se muestra la evolución temporal de  $\theta$  partiendo desde una posición casi vertical para los casos con y sin el controlador estabilizante



### 7.3. *Swing up*, control estabilizante y control supervisor

de bajada. Se observa como el controlador reduce considerablemente el tiempo de asentamiento. De esta manera el control cumple con su objetivo, evitar el tiempo de espera en el que el péndulo oscila antes de llegar al reposo.

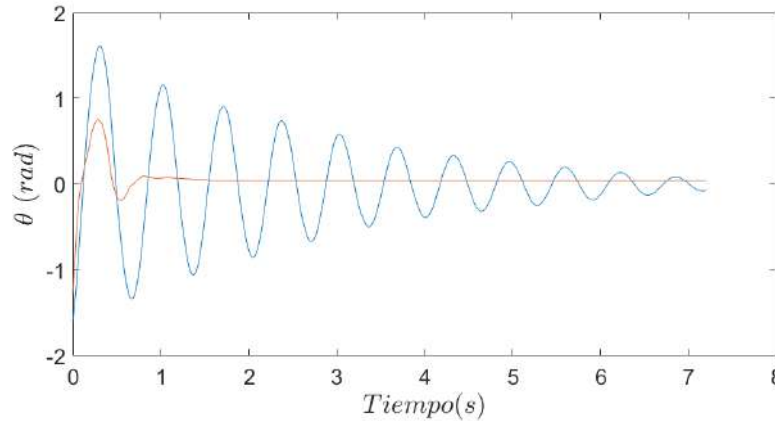


Figura 7.26: Evolución de  $\theta$  durante una caída. La curva que tiene menor tiempo de asentamiento es la relevada con controlador estabilizante de bajada.

#### 7.3.2. Condiciones de cambio entre los controladores

Cuando se inicia el sistema el péndulo está en reposo en la posición vertical hacia abajo. Esta es la condición inicial típica del *swing up* por lo tanto este controlador no tendrá problemas. En cambio el controlador estabilizante cuenta con una cierta región de atracción acotada, pequeña. Como se mostró anteriormente, dependiendo del ajuste de los parámetros del *swing up*, el mismo no lleva al péndulo a la posición invertida con velocidad nula, por lo tanto la transición entre los controladores no es trivial. Por ejemplo, el *swing up* con  $k = 50$ , por momentos, llega a  $\theta = \pi \text{ mod } 2\pi$  con una velocidad del orden de  $\dot{\theta} = 4 \text{ rad/s}$ . Por otro lado con  $k = 36$  suele llegar a con velocidad nula, pero no llega de manera robusta. La estrategia utilizada para hacer esta transición exitosa es utilizar  $k = 50$  para el *swing up* (de manera que sea robusto) y luego realizar el cambio de controlador antes de que el péndulo llegue a posición vertical. De esta manera el control estabilizante debe retener el péndulo para luego estabilizarlo. Visto desde otro punto de vista, el estado del sistema cuando se cambia de controlador debe ser tal que pertenezca a la región de atracción del control estabilizante.

La búsqueda de una condición que permitiera una transición exitosa se hizo haciendo pruebas en el sistema. La condición de cambio propuesta es  $|\theta - \pi \text{ mod } 2\pi| < \theta_0$ . Por comodidad, para implementar esta condición se utilizó la exponencial compleja,  $\theta \mapsto e^{i\theta}$ , como se menciona en el capítulo previo. Se comenzó con un umbral  $\theta_0$  pequeño y se fue aumentando hasta encontrar un ángulo que logré el objetivo y que sea cómodo. Finalmente se utilizó  $\theta_0 = 34^\circ$ . En términos de la exponencial compleja la condición es  $|e^{i\theta} + 1| < 0,6$ .

## Capítulo 7. Controladores

Para un diseño más teórico de la condición de cambio se puede hacer una estimación de la región de atracción del control estabilizante y de esta manera calcular en que momento el *swing up* entra en dicha región. Para estimar la zona de atracción resulta muy útil tener una función de Lyapunov  $V : \Omega \rightarrow \mathbb{R}$  para el sistema realimentado con el controlador estabilizante, donde  $\Omega$  es un entorno del punto de equilibrio. Además existen formas sistemáticas de calcular funciones de Lyapunov para un controlador lineal sobre un sistema linealizado [42]. Aún así hay detalles y el método merece su tiempo. Es por esto que preferimos buscar la condición de manera experimental.

En la lógica del sistema se tiene en cuenta la posibilidad de que durante el control estabilizante el péndulo caiga. En este sentido cuando el péndulo se aparta de la posición invertida un cierto ángulo se asume que el control estabilizante perdió el dominio del mismo y se va a caer. Una vez sucede eso se hace la transición nuevamente a *swing up*. Se decidió utilizar una condición muy permisiva de forma de evitar que el control estabilizante no se vea interrumpido. Además se quiere que el *swing up* no comience a actuar cuando el péndulo aún se encuentra muy arriba. Se utilizó el umbral:  $\theta_1 = 82^\circ$ . En términos de la exponencial compleja la condición queda  $|e^{i\theta} + 1| > 1,3$ .

Finalmente para activar el desenredador se utiliza como condición  $|\varphi| > 2\pi$ . Una vez que se activa este control, hasta que no se cumpla  $|\dot{\varphi}|$  y  $|\varphi| < 0,05$ , no se vuelve al estado inicial. En el estado inicial se activa el controlador de bajada cuyo objetivo es que  $\theta$  y  $\dot{\theta}$  lleguen al reposo más rápido que dejando el péndulo oscilar libremente. Por último, para volver a comenzar el *swing up* se debe cumplir  $|e^{i\theta} - 1| < 0,04$  y  $|\dot{\theta}| < 0,001$ . En la figura 7.27 se muestra la máquina de estados que implementa el control supervisor, en donde se ven todas las condiciones discutidas.

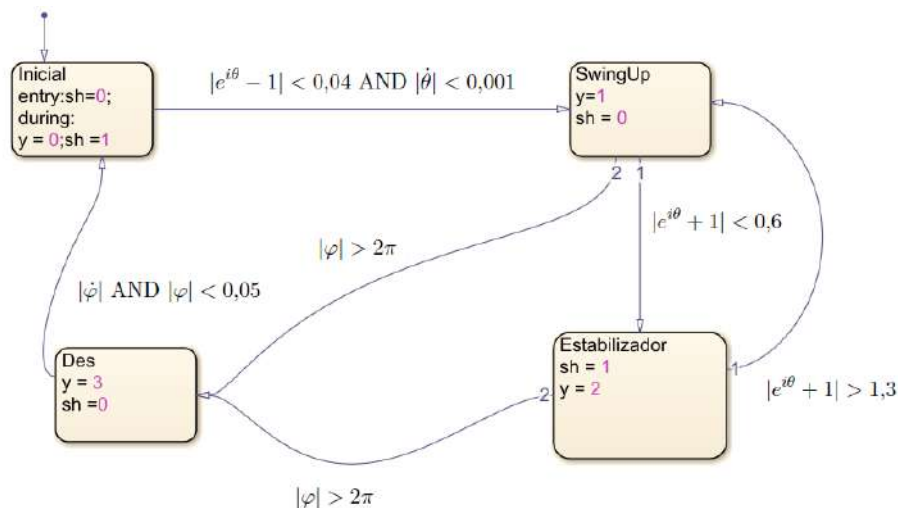


Figura 7.27: Control supervisor

### 7.3. Swing up, control estabilizante y control supervisor

#### 7.3.3. Resultados

Comenzamos resumiendo nuevamente los controladores utilizados. Para el *swing up* se utilizó la siguiente ley de control ya explicada anteriormente:

$$\begin{cases} v = H \left( \frac{d(\theta)}{I_x l_{bi}} u + 2I_x \dot{\theta} \dot{\varphi} \sin \theta \cos \theta - m_2 l_{bi} c_z \sin \theta \dot{\theta}^2 \right. \\ \left. + \frac{m_2 l_{bi} c_z}{I_x} \cos \theta (\dot{\varphi}^2 I_x \cos \theta \sin \theta - m_2 g c_z \sin \theta) \right) \\ u = u_{sw} = k(\arccos(\cos(\theta - \pi)))^3 E \tanh(20\dot{\theta} \cos \theta) \end{cases} \quad (7.14)$$

con  $k = 50 \text{ m/Js}^2$  y  $H = 1 \text{ V/Nm}$ . Con este valor de  $k$  se observaba un comportamiento más robusto durante el *swing up* por eso se decidió utilizarlo. De control estabilizante se utilizó el control 2, recordamos que tiene un desempeño más suave que el control 1. La ley de control es:

$$v = -k_1 i + k_2 \arcsin \sin \theta - k_3 \dot{\theta} - k_4 (\varphi - \varphi_0) - k_5 \dot{\varphi} \quad (7.15)$$

cuyos coeficientes y polos se muestran en la tabla 7.4.

Coeficientes	Polos del lazo cerrado
$K = [-8.30, 25.79, 2.55, -0.69, -1.76]$	$\lambda_1 = -20 \quad \lambda_2 = -8 \quad \lambda_3 = -7$ $\lambda_4 = -6 \quad \lambda_5 = -5$

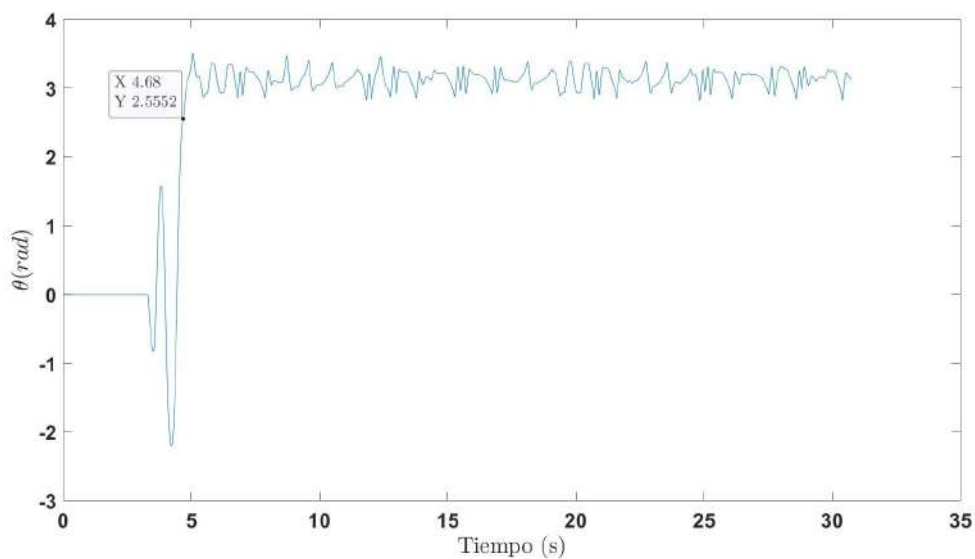
Tabla 7.4: Coeficientes del controlador estabilizante utilizado en esta sección, el controlador 2 y sus polos.

Seguimos mostrando los resultados obtenidos en ensayos sobre el prototipo. En la figura 7.28a se muestra la evolución de  $\theta$ . En dicha figura se marcó el punto en el cual se da el cambio entre *swing up* y control estabilizante. Como se observa  $\theta$  se mantiene suave en el cambio de controladores. Luego del cambio el desempeño es muy similar al del control estabilizante 2 mostrado en 7.1.5, pero con la ventaja de estabilizar adecuadamente  $\varphi$ .

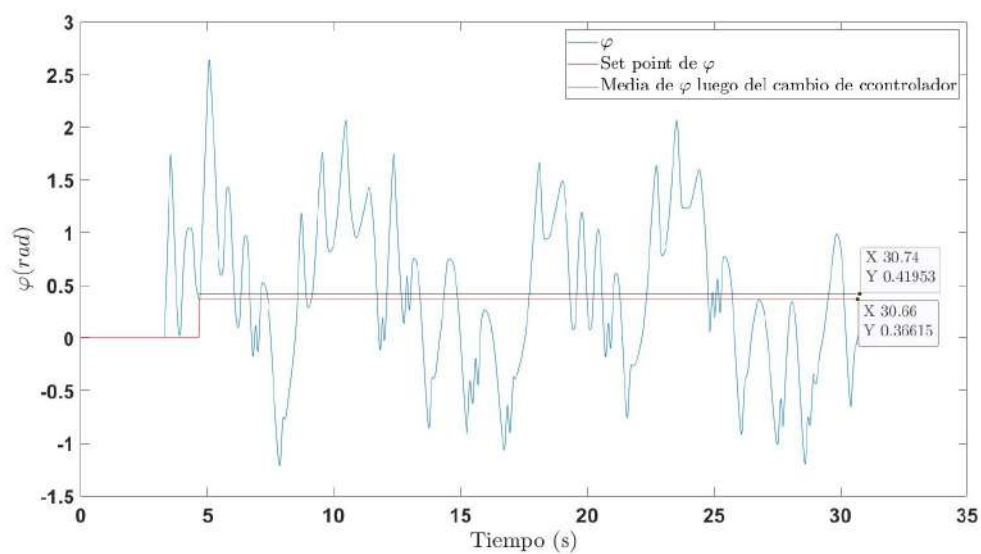
En la figura 7.28b se muestra la evolución de  $\varphi$ . Además se muestra la variable *set point*, recordar que esta variable retiene el valor de  $\varphi$  en el momento de cambio de *swing up* a control estabilizante y es el valor a seguir para este controlador. Recordamos que esta elección es porque eventualmente el *swing up* podría mover mucho esta variable y esto podría generar problemas para el control estabilizante si se quiere llevar  $\varphi$  a 0 y la condición inicial en  $\varphi$  quedó lejos. También se muestra la media de  $\varphi$  luego del cambio de controladores y se ve claramente cómo es muy similar al set point a seguir.

Para finalizar en la figura 7.29 se muestra la dinámica de  $(\theta, \dot{\theta})$ . En la figura se puede observar la acción del *swing up* seguida por el control estabilizante, el cual comienza actuar en el punto señalado. También se puede apreciar que el rendimiento del *swing up* es el mismo que el estudiado en la sección previa. Inmediatamente

## Capítulo 7. Controladores



(a)  $\theta$  durante relevo. El punto marcado hace referencia al momento de cambio del *swing up* al control estabilizante.



(b)  $\varphi$  durante el relevo. El momento donde cambia el set point se corresponde con el cambio del *swing up* al control estabilizante.

Figura 7.28: Dinámica de  $\theta$  y  $\varphi$  durante *swing up* y luego control estabilizante.

luego del cambio de controladores se observa un cambio de pendiente en la órbita. Esto es más que razonable ya que el cambio entre los controladores es un cambio abrupto, no es una transición *suave*. Recordamos aquí que el *swing up* no llega con velocidad nula al objetivo y por eso es fundamental que el control estabilizante lo atrape.

### 7.3. *Swing up*, control estabilizante y control supervisor

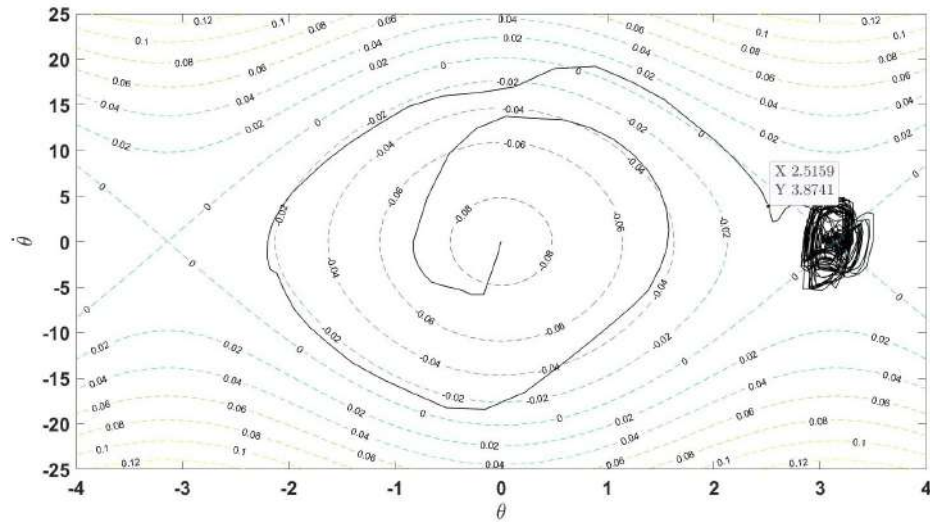


Figura 7.29: Dinámica de  $(\theta, \dot{\theta})$ , se observa la acción del *swing up* precedida por el control estabilizante, quién comienza actuar en el punto señalado.

Como conclusión se ha utilizado de manera exitosa el software en el cual se ha integrado el sistema desarrollado en los primeros capítulos de la tesis, junto con el modelo desarrollado e identificado en el capítulo 4 y 5, para desarrollar un control capaz de llevar el péndulo de su posición de reposo a la posición invertida, manteniendo el mismo en dicha posición. Además el control no deja que el sistema acceda a ángulos  $\varphi$  cumpliendo  $|\varphi| > 2\pi$ . De esta manera el controlador evita que el cable del encoder se enrede. Los resultados son más que satisfactorios, avalando la integración en el software, el sistema desarrollado, el modelo y la identificación y el propio desarrollo de los controladores.

Esta página ha sido intencionalmente dejada en blanco.

# Capítulo 8

## Conclusiones

Comenzamos el capítulo haciendo un breve punteo de cuáles eran los objetivos del proyecto y cuáles eran los problemas a atacar.

- Diseñar y construir un Péndulo de Furuta.
- Obtener un modelo que caracterice su dinámica.
- Identificación de los parámetros del modelo.
- Integración en software de simulación.
- Lograr controlar el péndulo en su posición invertida con al menos una estrategia de control.

Si uno se detiene a observar los objetivos planteados, es claro que en el transcurso del proyecto se logró cumplirlos de forma exitosa. Se comenzó por el diseño y construcción de un Péndulo de Furuta, esta fue quizás la etapa que, a priori, más se desconocía su grado de dificultad. A pesar de esto se logró construir un prototipo de muy bajo costo construido en su mayoría en base a piezas recicladas. El prototipo cumple con el objetivo de ser portable y desmontable, siendo apto para exposiciones y ámbitos de laboratorio.

En cuanto al modelado del sistema, como se desglosa en la sección 4.3, se desarrolló un modelo de nuestro Péndulo de Furuta el cual coincide con el modelo derivado por la mayoría de los artículos que tratan el Péndulo. Para el caso del motor DC su modelo clásico bastó para representar el comportamiento del sistema a pesar de no ser capaz de reproducir fielmente el comportamiento de la corriente. Un punto a destacar sobre esta sección fue el nivel de profundidad que se alcanzó con respecto al comportamiento de las fuerzas no conservativas, contemplando los efectos viscosos, estáticos y de Coulomb de la fricción. Gracias a esto el modelo logra reproducir efectos como la zona muerta del motor, prediciendo el voltaje límite de la misma e incluso imitando la corriente de arranque.

## Capítulo 8. Conclusiones

En cuanto a la identificación de los parámetros del motor, la misma consistió en dos etapas. En la primera se hicieron distintos ensayos experimentales para determinar los parámetros del modelo. Luego se constató de que el modelo no era tan bueno prediciendo el transitorio. Fue por esta razón que dichos parámetros se utilizaron como semilla de un ajuste basado en una función minimizante diseñada por nosotros (la cual tenía en cuenta transitorio y régimen), esta fue la segunda etapa de identificación.

La decisión de utilizar la función minimizante no fue sin considerar sus implicancias y ameritó discusiones entre equipo y tutores. Sin embargo, fue crucial para lograr tener un modelo sencillo pero preciso, sabiendo lo trabajoso que podría ser adentrarse en afinar aún más el modelo del prototipo. Creemos que la decisión tomada fue acertada, siendo concientes de que la naturaleza de las diferencias entre modelo y realidad surgen realmente de efectos no contemplados por nuestra representación del sistema. Un ejemplo claro de estos efectos no contemplados es la dinámica introducida por el driver. Logramos constatar que la utilización del driver reduce el tiempo de levantamiento de la velocidad del motor, provocando un transitorio más lento.

Para el Péndulo de Furuta los parámetros relevados de ensayos experimentales ajustaron con éxito su comportamiento. Como resultado, el modelo completo desarrollado junto con los parámetros, ya ajustados, demostraron predecir la respuesta al escalón del sistema, manteniéndose dentro del margen de incertidumbre encontrado en la práctica, excepto para la corriente. Esto último incluso partiendo desde el reposo, donde la zona muerta del motor tiene su efecto. Es importante remarcar que no se pretendió mejorar el resultado sobre la corriente ya que la misma proviene de una señal PWM de 1 kHz, debido al driver. Otro buen resultado es que el modelo logró replicar el comportamiento del sistema real al ser sometido a los controladores implementados, al punto de reproducir gran parte de las vibraciones debidas, entre otras cosas, a la fricción y *backlash* del motor. Estos resultados cierran el proceso de modelado y el de identificación, dando resultados excelentes.

La integración con el software de simulación resultó muy útil permitiendo simular e implementar distintos controladores en el sistema de forma realmente rápida. En el software se logra simular, actuar sobre el sistema, relevar y guardar variables en un entorno de desarrollo centralizado en el que el usuario logra visualizar y obtener conclusiones de los resultados obtenidos cómodamente. Además es de gran ayuda tener a disposición las potentes herramientas de cálculo, visualización, modelado y análisis de sistemas, así como la gran versatilidad que el programa trae para manipularlos. De esta forma se favorece al usuario con un uso fluido del instrumento de laboratorio construido. Un ejemplo del poder que trae consigo la herramienta se ve en la máquina de estados implementada para el control supervisor, la cual fue una gran ventaja a la hora de combinar distintos controladores.

Es en la etapa de controladores en donde creemos que el equipo de trabajo dio el paso extra, superando con creces los resultados esperados, no solo por los resul-



## 8.1. Trabajos Futuros

tados en sí, sino por haber logrado combinar el *Swing Up* y el control estabilizante de forma exitosa, integrándolos junto con un control supervisor. De esta manera se logró un sistema que auto-supervisa su comportamiento, evitando enredarse y dañar el cableado del encoder y reduciendo tiempos de espera.

Se quiere destacar los siguientes puntos sobre la etapa de controladores:

- Se implementaron dos controladores estabilizantes. Los dos lograron cumplir el objetivo de estabilizar el péndulo. Sin embargo, ambos presentaron vibraciones, pero en distinta magnitud.
- La respuesta de estos controladores es en gran medida la predicha por las simulaciones con el modelo. Incluso las simulaciones predicen en gran parte las vibraciones observadas. El análisis mostró que las vibraciones tenían dos componentes frecuenciales principales. Las simulaciones nos permitieron deducir que la fricción era una causa de la componente de baja frecuencia.
- El *Swing Up* fue implementado con éxito de forma robusta. El mismo se diseñó en base al *Swing Up* de un péndulo simple pero adaptando la ley de control a nuestro péndulo. Para implementarlo con éxito fue necesario probar distintas variantes de la ley de control. Finalmente los intentos fallidos sugirieron el uso de una función ponderante que logró el objetivo demostrando tener la robustez buscada.
- La transición entre *Swing Up* y control estabilizante fue bien lograda. Para esto se optó por hacer que el control estabilizante actué un poco antes de que el *Swing Up* lleve el péndulo a su objetivo.
- Utilizando una máquina de estados se implementó un control supervisor encargado de coordinar el *Swing Up* y el control estabilizante, además de servir como protección para evitar que el prototipo se enrede y dañe su cableado. El control supervisor también optimiza los tiempos muertos permitiendo trabajar con el prototipo de formas más fluida.

En resumen, se diseñaron e implementaron con éxito distintos controladores sobre el prototipo construido, utilizando el modelo desarrollado e identificado, junto con el software en el cuál se integró el sistema. Es por esto que la etapa de controladores resulta ser la principal validación para el sistema completo. Concluimos que nos hemos enfrentando de manera satisfactoria al proceso de ingeniería que conlleva el desarrollo de un prototipo en todas sus etapas así como la integración del mismo en un entorno de desarrollo centralizado. También nos enfrentamos con éxito al proceso del control: modelar, identificar, diseñar e implementar un controlador sobre el prototipo diseñado.

## 8.1. Trabajos Futuros

Esta sección tiene como propósito dar a luz todos los aspectos del proyecto que podrían mejorarse. Replantando las decisiones tomadas desde un punto de vista

## Capítulo 8. Conclusiones

en el que se tuviera que comenzar de nuevo, analizando aquellas que consideramos que no fueron ideales. A su vez esta sección muestra posibles trabajos futuros sobre el péndulo.

Cronológicamente, la primer decisión que nos generó dificultades en las etapas posteriores en el proyecto fue la elección del motor. Decidimos seleccionar un motor disponible en plaza y lo obtenido fue un modelo con engranaje, cuya dinámica genera efectos complejos. De haber obtenido un motor sin engranajes se hubiera simplificado la dinámica no modelada como por ejemplo el *backlash* y parte de la fricción, y por lo tanto las vibraciones observadas a la hora de controlar el péndulo.

Posteriormente, la elección del driver para el motor estuvo lejos de ser ideal. Como fue explicado en la sección 3.2, no fue hasta haber realizado una diversa cantidad de ensayos que notamos los efectos provocados por la estructura interna del driver. En particular, el hecho de imponer corriente 0 a entrada nula en lugar de imponer tensión 0 provoca que la curva entrada-salida de la tensión del driver no sea una recta cuando se carga con el motor y se tenga que compensar. Además al tratarse de una onda PWM de 1 *kHz* el verdadero carácter de la corriente es complejo. Quizás lo ideal sería buscar la forma de imponer en bornes del motor una entrada de tensión DC manejada por el Arduino o PC en lugar de una señal PWM. Una opción sería investigar la posibilidad incluir en el sistema una fuente de DC programable.

Dada la escasa cantidad de información en artículos sobre los aspectos constructivos del Péndulo de Furuta, hubiera sido conveniente consultar con gente con experiencia en la construcción de este tipo de sistemas, de manera de expandir las opciones y evaluar distintas posibilidades de formas de construir el prototipo.

Algo que se debe mencionar es que se podría hacer un diseño mucho más compacto del PCB, en el que el driver, el sensor de corriente y el microcontrolador estén en el PCB y no montados sobre el mismo. Se podría también agregar en la placa un módulo de comunicación inalámbrica y otro acoplado al encoder del péndulo de manera de evitar que el cable entre ellos se enrede al rededor del pilar.

Pasando ya a la etapa de identificación, se realizaron una gran cantidad de ensayos de los cuales hay muchos que no están presentados en este documento dado que no tuvieron utilidad para la identificación del modelo. Quizás hubiera sido pertinente planificar de mejor manera los ensayos a realizar, ya teniendo de antemano el modelo del sistema estudiado.

Con respecto a la etapa de implementación, nos hubiera ahorrado mucho tiempo el haber comenzado desde las etapas iniciales del proyecto utilizando la herramienta Simulink. Es una herramienta potente pero tiene una curva de aprendizaje que se debe tener en cuenta. Existen muchas herramientas disponibles de las cuales desconocemos y quizás podrían simplificar u optimizar nuestro diseño. Podríamos, por ejemplo, haber buscado la forma de aumentar la frecuencia de muestreo. En

## 8.1. Trabajos Futuros

el caso de no ser posible, habría que investigar la manera de hacerlo. Una forma podría ser con el uso de una tarjeta adquisidora en vez de un Arduino.

También con suficiente tiempo se hubiera podido realizar un modelo 3D con el cual se pueda visualizar las simulaciones mediante animaciones.

Pasando ya a la etapa de diseño de controladores, se podría seguir investigando las causas de las vibraciones que aparecen a la hora de estabilizar el péndulo. Y una vez identificadas las fuentes buscar la manera de mitigarlas.

Realmente se abre un mundo de trabajos para hacer una vez se tiene el sistema. Por ejemplo se podría investigar utilizar otros controladores lineales como un PID, LQR, o utilizar otras técnicas como Root Locus, métodos de respuesta en frecuencia y redes neuronales entre otros. Además de diseñar esta la posibilidad de implementar las leyes de control que se encuentran en los varios artículos que hoy en día tratan con el péndulo. Estas técnicas de control son típicamente del mundo del control no lineal. Por ejemplo el artículo [2] presenta una ley de control única  $v = v(x)$  que realiza el *Swing Up* y la estabilización. También hay trabajos para profundizar aun más en el modelado, a modo de ejemplo, utilizar un modelo de fricción dinámico como el tan usado LuGre.

Por último, sería enriquecedor lograr diseñar una metodología que permita estudiar la robustez de cada uno de los controladores, estudiando como varía el rendimiento de los mismos si se modifican las características mecánicas del péndulo, brazo o cualquier otra pieza. Esto se puede hacer, por ejemplo, colocando una masa en distintas partes del mismo. También se podría estudiar como cuantificar el rechazo a las perturbaciones.

Esta página ha sido intencionalmente dejada en blanco.

## Apéndice A

# Estimación de Máxima Caída de Voltajes en Pistas del PCB

Con la información provista por los fabricantes se puede calcular la máxima caída de tensión en las pistas. El peor caso se da en la pistas de alimentación hacia el driver. Estas tienen un ancho de  $34\text{ mm}$  y un grosor predefinido por el fabricante de  $35\text{ }\mu\text{m}$ . El camino más largo de la pista puede ser de hasta  $115\text{ mm}$  (se toma como peor caso el largo de la placa). Utilizando la nomenclatura definida en la figura A.1 y sabiendo que la corriente máxima medida fue de  $1,5\text{ A}$ , la máxima caída de voltaje se calcula como:

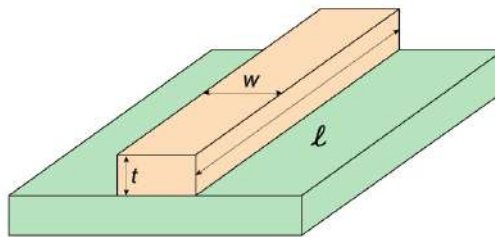


Figura A.1: Nomenclatura usada para las pistas del PCB  
Imagen extraída de <https://www.electronicdesign.com>

$$V_{max} = I_{max}R_{max} = I_{max} \frac{\rho l}{Wt} = 2,5\text{ mV} \quad (\text{A.1})$$

Esta caída se considera despreciable frente a los  $12\text{ V}$  nominales y es a su vez 48 veces menor que el ruido inherente del adaptador AC/DC, típicamente de  $120\text{ mV}$  según las cartillas de productos similares. De aquí se concluye que el ancho de las pistas es más que suficiente para las condiciones de uso.

Esta página ha sido intencionalmente dejada en blanco.

## Apéndice B

### Cálculo de los momentos de inercia y del centro de masa

Primero calculemos  $J$ , el momento de inercia del eje  $z$  del rígido 1. Sea  $R_1, E, U, S_R, R_E \subset \mathbb{R}^3$  el rígido 1, el encoder, el soporte  $U$ , el soporte del rodamiento y el rodamiento exterior respectivamente, entonces:

$$J = \int_{R_1} x^2 + y^2 dm(\vec{x}) = \int_U x^2 + y^2 dm(\vec{x}) + \int_E x^2 + y^2 dm(\vec{x}) \\ + \int_{R_E} x^2 + y^2 dm(\vec{x}) + \int_{S_R} x^2 + y^2 dm(\vec{x})$$

calculemos cada una de estas integrales. Para el soporte en  $U$  la hipótesis de homogeneidad dice que la medida de masa cumple  $dm = \frac{m_u}{A_u} dA$  con  $A_u$  el área del soporte en  $U$ . Entonces

$$\int_U x^2 + y^2 dm(\vec{x}) = \frac{m_u}{A_u} \left( \int_{-a_u/2}^{a_u/2} \int_{-l_{ud}}^{l_{ui}} x^2 + y^2 dy dx + \int_{-a_u/2}^{a_u/2} \int_0^{h_u} l_{ui}^2 + y^2 dz dy \right. \\ \left. + \int_{-a_u/2}^{a_u/2} \int_0^{h_u} l_{ud}^2 + y^2 dz dy \right) \\ = \frac{m_u}{A_u} \left( \frac{a_u^3}{12} (l_{ui} + l_{ud}) + \frac{a_u}{3} (l_{ui}^3 + l_{ud}^3) + \frac{a_u^3 h_u}{6} + a_u h_u (l_{ui}^2 + l_{ud}^2) \right)$$

análogamente para el encoder,

$$\int_E x^2 + y^2 dm(\vec{x}) = \frac{m_e}{v_e} \int_{l_{ud}}^{l_{ud}+a_e} \left( \int_0^{2\pi} \int_0^{r_e} (x^2 + r^2 \sin^2 \theta) r dr d\theta \right) dx = \frac{m_e}{a_e} \left( \frac{a_e r_e^2}{4} + \frac{(l_{ud} + a_e)^3 - l_{ud}^3}{3} \right)$$

## Apéndice B. Cálculo de los momentos de inercia y del centro de masa

Para el soporte del rodamiento tenemos que

$$\begin{aligned} \int_{S_R} x^2 + y^2 dm &= \int_{S_R} (x^2 + y^2) \frac{m_{sr}}{\pi r_{sr}^2 a_{sr}} dS = \frac{m_{sr}}{\pi r_{sr}^2 a_{sr}} \int_{l_{ui}-a_{sr}}^{l_{ui}} \int_0^{2\pi} (x^2 + r_{sr}^2 \cos^2 \theta) r_{sr} d\theta dx \\ &= \frac{m_{sr}}{\pi r_{sr} a_{sr}} \int_{l_{ui}-a_{sr}}^{l_{ui}} 2x^2 \pi + r_{sr}^2 \pi dx = \frac{m_{sr}}{r_{sr} a_{sr}} \left( \frac{2}{3} [l_{ui}^3 - (l_{ui} - a_{sr})^3] + r_{sr}^2 a_{sr} \right) \end{aligned}$$

Para el rodamiento exterior la cuenta es completamente análoga, el resultado es

$$\frac{m_{re}}{r_r a_r} \left( \frac{2}{3} [(l_{ui} - a_{sr} + a_r)^3 - (l_{ui} - a_{sr})^3] + r_r^2 a_r \right)$$

Sumando las expresiones anteriores se obtiene el valor de la inercia,  $J = 0,0185 \pm 0,00096 \text{ kg} \cdot \text{m}^2$ .

Para calcular la incertidumbre de  $J$  se utilizaron variables aleatorias uniformes y se relevó la distribución de  $J$ . Para profundizar en esto ver el apéndice C.

Para el cálculo de  $I_x$ ,  $I_y$  e  $I_z$  observemos que al ser el rígido 2 plano se cumple  $I_x + I_z = I_y$ . Recordemos que en la base  $\mathcal{C}$  el mapa  $I_A$  diagonaliza, por lo cual lo único a calcular es  $I_x$  e  $I_z$ . Para el cálculo la medida de masa  $m = m_p + m_b + m_{ri} \delta$ , donde  $m_p$  es homogénea en el péndulo y nula en el complemento,  $m_b$  es homogénea en el brazo y nula en el exterior y  $\delta$  es una delta de Dirac en la posición del rodamiento. Entonces

$$\begin{aligned} I_x &= \int_{R_2} y^2 + z^2 dm = \int_p z^2 dm = \int_p z^2 dm_p = \int_0^{l_p} z^2 \frac{m_p}{l_p} dz = \frac{m_p l_p^2}{3} \\ I_z &= \int_{R_2} x^2 + y^2 dm = \int_{R_2-p} x^2 dm = \int_{R_2-p} x^2 m_{ri} d\delta + \int_{R_2-p} x^2 dm_b = m_{ri} d_{ar}^2 + \frac{m_b l_b^2}{3} \end{aligned}$$

De donde se obtiene  $I_x = 4,39 \times 10^{-4} \pm 2,8 \times 10^{-5} \text{ kg} \cdot \text{m}^2$  y  $I_z = 1,88 \times 10^{-4} \pm 4,5 \times 10^{-5} \text{ kg} \cdot \text{m}^2$ . Por último veamos el cálculo de  $\vec{r}_G - \vec{r}_A$ , por definición del centro de masa, se tiene

$$\vec{r}_G - \vec{r}_A = \frac{1}{m_2} \int_{R_2} \vec{r} - \vec{r}_A dm(\vec{r})$$

Si expresamos los vectores en la base  $\mathcal{C}$  se tiene

$$\begin{aligned} \vec{r}_G - \vec{r}_A &= \frac{1}{m_2} \left( \int_{R_2} x' dm(\vec{r}), 0, \int_{R_2} z' dm(\vec{r}) \right) = \frac{1}{m_2} \left( \int_{R_2-p} x' dm(\vec{r}), 0, \int_p z' dm(\vec{r}) \right) \\ &= \frac{1}{m_2} \left( \int_{R_2-p} x' m_{ri} d\delta + \int_{R_2-p} x' dm_b, 0, \int_p z' dm_p \right) = \frac{1}{m_2} \left( -m_{ri} d_{ar} - \frac{m_b l_b}{2}, 0, \frac{m_p l_p}{2} \right) \end{aligned}$$

En conclusión,

$$c_x = - \frac{m_{ri} d_{ar} + \frac{m_b l_b}{2}}{m_2} = -2,3 \pm 0,042 \text{ cm}$$

$$c_z = \frac{m_p l_p}{2m_2} = 4,4 \pm 0,016 \text{ cm}$$





Esta página ha sido intencionalmente dejada en blanco.

# Apéndice C

## Estimación de Incertidumbres Mediante Simulación

Se tienen algunas magnitudes de interés que se calculan en base a medidas de otros parámetros, y donde la incertidumbre asociada a estas magnitudes no es fácil de calcular y en algún caso no esta claro como. Ejemplo de esto son  $J$  y la velocidad angular  $\omega$ . Se toma como estrategia en estos casos simular variables aleatorias uniformes para las magnitudes medidas directamente y relevar la distribución de estas magnitudes más complejas.

### C.1. Distribución de $J$

Primero recordar como se calcula  $J$ :

$$J = \frac{m_u}{A_u} \left( \frac{a_u^3}{12} (l_{ui} + l_{ud}) + \frac{a_u}{3} (l_{ui}^3 + l_{ud}^3) + \frac{a_u^3 h_u}{6} + a_u h_u (l_{ui}^2 + l_{ud}^2) \right) + \frac{m_e}{a_e} \left( \frac{a_e r_e^2}{4} + \frac{(l_{ud} + a_e)^3 - l_{ud}^3}{3} \right) + \frac{m_{sr}}{r_{sr} a_{sr}} \left( \frac{2}{3} [l_{ui}^3 - (l_{ui} - a_{sr})^3] + r_{sr}^2 a_{sr} \right) + \frac{m_{re}}{r_r a_r} \left( \frac{2}{3} [(l_{ui} - a_{sr} + a_r)^3 - (l_{ui} - a_{sr})^3] + r_r^2 a_r \right)$$

Si bien es posible realizar propagación de incertidumbres, el proceso es tedioso. Para evitarlo se decidió simular variables uniformes en el intervalo correspondiente teniendo en cuenta la incertidumbre para cada una de las magnitudes que aparecen en el cálculo de  $J$  y luego calcularla.

Se realizo el experimento 100000 veces y en la figura C.1 se muestra el histograma resultante.

Es claro que la distribución de  $J$  es normal. La media es  $0,0182 \text{ kgm}^2$  coincidiendo con el valor calculado con los valores nominales de los parámetros. La desviación estandar calculada es  $0,00032 \text{ kgm}^2$ . Como incertidumbre para la me-

## Apéndice C. Estimación de Incertidumbres Mediante Simulación

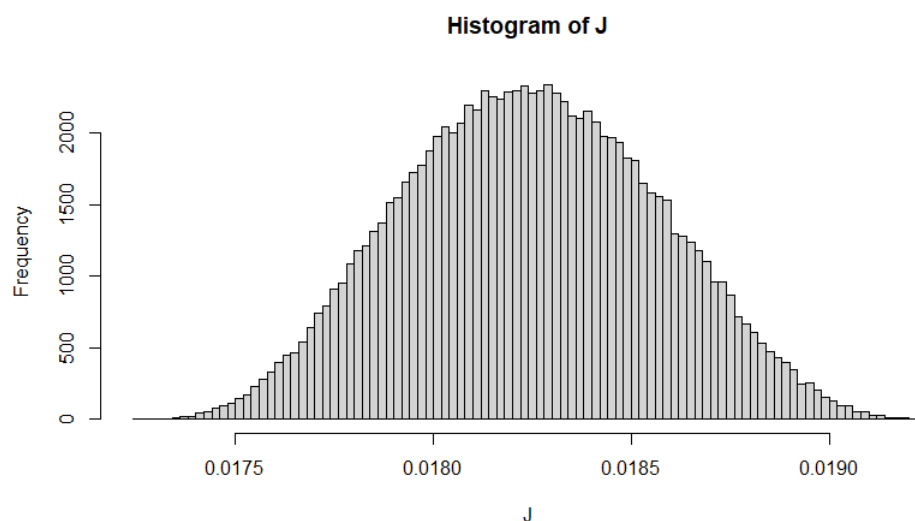


Figura C.1: Histograma de  $J$

didada se decide utilizar  $3\sigma$ . Por lo tanto el obtenemos  $J = 0,0182 \pm 0,00096 \text{ kgm}^2$ . Para estos cálculos se utilizo el programa *RStudio*.

### C.2. Distribución de $\omega$ en las medidas para el cálculo del $K$ del motor

Para calcular la velocidad angular se releva las cuentas del encoder y se tiene el ángulo. A partir de este relevo de  $\theta(t)$  se filtra para obtener la velocidad angular. La incertidumbre de la medida del ángulo es clara, pero cómo se propaga a la medida de la velocidad no. Aquí se pretende obtener una estimación de la incertidumbre para la medidas de velocidad realizadas, con el objetivo de calcular la incertidumbre en el parámetro  $k$  del motor. No se pretende lograr una estimación de la incertidumbre en la velocidad angular general para todos los casos de usos.

Recordar que en las medidas tomadas para calcular el valor de  $k$  se tienen relevos en DC, donde la velocidad angular es constante. Se quiere emular esta situación por lo tanto se utilizan entradas rampas para obtener una salida del filtro constante. A estas señales de estilo rampa se le suma una señal aleatoria con distribución uniforme, punto a punto, iid, de media nula y varianza dada por la apreciación del encoder.

Se busca cuantificar la incertidumbre para cada una de las medidas que se realizaron por lo tanto se debe estudiar la distribución de la salida del filtro para cada una de las velocidades relevadas. Sin embarog, se observo que la incertidumbre no depende de la velocidad de régimen. En la figura C.2 se muestra el resultado para una rampa de pendiente  $1 \text{ rad/s}$ . Recordar que la transferencia del filtro es:

## C.2. Distribución de $\omega$ en las medidas para el cálculo del $K$ del motor

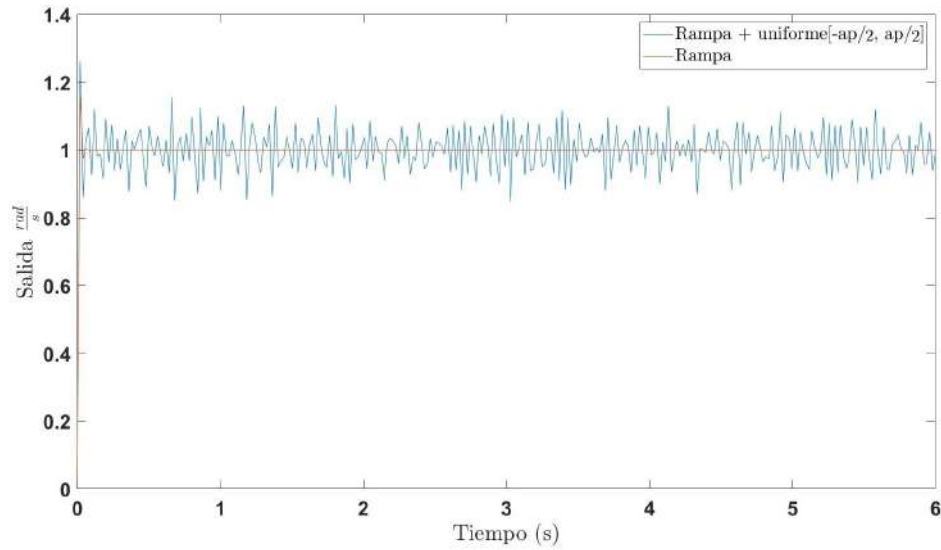
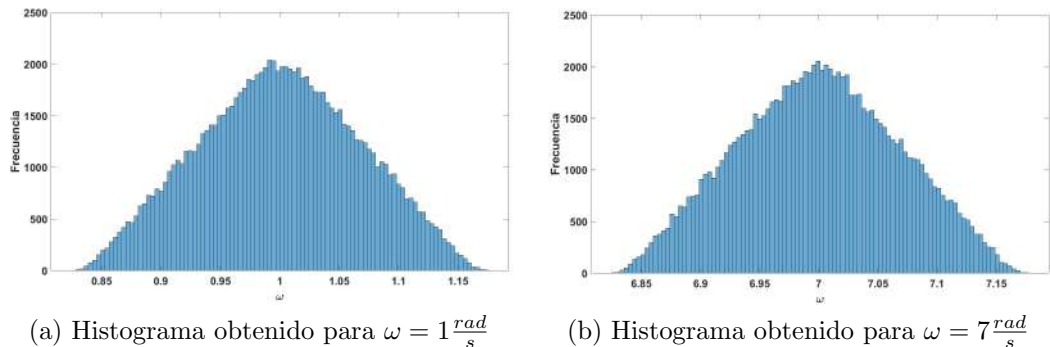


Figura C.2: Salida del filtro para una rampa de pendiente  $1 \text{ rad/s}$  con una señal uniforme superpuesta.

$$\frac{s}{\left(\frac{s}{2\pi 30}\right)^2 + \frac{s}{2\pi 30} + 1}$$

De tomar varias simulaciones se observa que la distribución en régimen no depende del punto, es un proceso estacionario. Entonces la estrategia utilizada es hacer simulaciones de una entrada rampa con ruido y estudiar la distribución de un punto particular de dicha señal en los diferentes ensayos. En la figura C.3b se muestran los histogramas obtenidos para dos velocidades diferentes. En ambos casos se realizaron 100000 simulaciones. Se ve claramente como las distribuciones son iguales. En ambos casos son distribuciones con  $\sigma = 0,067$ . El punto elegido es el correspondiente a  $t = 5 \text{ s}$  estando dentro de la velocidad de régimen.



(a) Histograma obtenido para  $\omega = 1 \frac{\text{rad}}{\text{s}}$

(b) Histograma obtenido para  $\omega = 7 \frac{\text{rad}}{\text{s}}$

Figura C.3: Histogramas para la salida del filtro para entradas de rampa +  $\mathcal{U}\left[-\frac{ap}{2}, \frac{ap}{2}\right]$

Otra estrategia que se podía utilizar es suponer que el proceso en cuestión es

## Apéndice C. Estimación de Incertidumbres Mediante Simulación

ergódico. Una consecuencia de esto es que los promedios estadísticos igualan a los promedios temporales. Entonces en vez de sortear varias señales se podía sortear una sola señal y promediar a lo largo del tiempo. Esto se probó y el resultado fue igual al anterior.

# Apéndice D

## Códigos

```
/* Interrupt Service Routine: change on pin A for Encoder 0 */
void irsEn0(){

    int MSB = digitalRead(Enc[0].pinA);
    int LSB = digitalRead(Enc[0].pinB);
    int Encodado = (MSB << 1) | LSB;

    int sum = (Enc[0].anterior << 2) | Encodado;

    if (sum == 0b1101 || sum == 0b0100 || sum == 0b0010 || sum == 0b1011)
        Enc[0].pos ++;
    if (sum == 0b1110 || sum == 0b0111 || sum == 0b0001 || sum == 0b1000)
        Enc[0].pos --;

    Enc[0].anterior = Encodado;
} /* end ISR pin A Encoder 0
```

Esta página ha sido intencionalmente dejada en blanco.



# Apéndice E

## Implementación de los filtros derivativos

Para obtener la velocidad angular a partir de la lectura de la posición por parte del encoder se decidió filtrar la señal. Como ya se mencionó en 3.3 se probaron estrategias de tiempo fijo y espacio fijo pero se vió que tenían problemas para obtener la velocidad angular.

El filtro que se desea implementar debe comportarse como derivador a bajas frecuencias y se desea filtrar las componentes de alta frecuencia. Concretamente la transferencia en tiempo continuo del filtro que se probó con éxito en la práctica es:

$$\frac{s}{\left(\frac{1}{2\pi 30}\right)^2 s^2 + \left(\frac{2}{2\pi 30}\right)s + 1}$$

Para obtener el filtro en tiempo discreto la primer opción es utilizar un retenedor de orden cero (MOC o ZOH). Si en simulink implementamos en un filtro discreto la transferencia anterior utilizando un ZOH lo que se esta haciendo es lo siguiente:

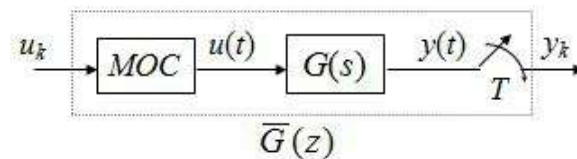


Figura E.1: Muestreo de un sistema continuo. Extraído de [50]

El teorema de la transmitancia muestreada nos dice que el sistema anterior esta dado por una transferencia  $\bar{G}(z)$  y que la misma es:

$$\bar{G}(z) = \frac{Y(z)}{U(z)} = \frac{z-1}{z} \mathcal{Z} \left[ \mathcal{L}^{-1} \left\{ \frac{G(s)}{s} \right\} \Big|_{t=kT} \right]$$

Donde  $\mathcal{Z}$  es la transformada Z y  $\mathcal{L}^{-1}$  es la inversa de la transformada de Laplace. Si aplicamos el teorema a la transferencia de interés obtenemos:

## Apéndice E. Implementación de los filtros derivativos

$$H(z) = \frac{(z-1)T_s a^2 e^{-aT_s}}{(z - e^{-aT_s})^2}$$

Donde  $a = 1/2\pi 30$  el polo doble y  $T_s$  el tiempo de muestreo del sistema. Recordar que el sistema se utiliza con un frecuencia de muestreo entre  $50 \text{ Hz}$  y  $200 \text{ Hz}$ . En la figura E.2 se muestra el diagrama de bode del filtro para una frecuencia de muestreo de  $50 \text{ Hz}$  y para  $200 \text{ Hz}$

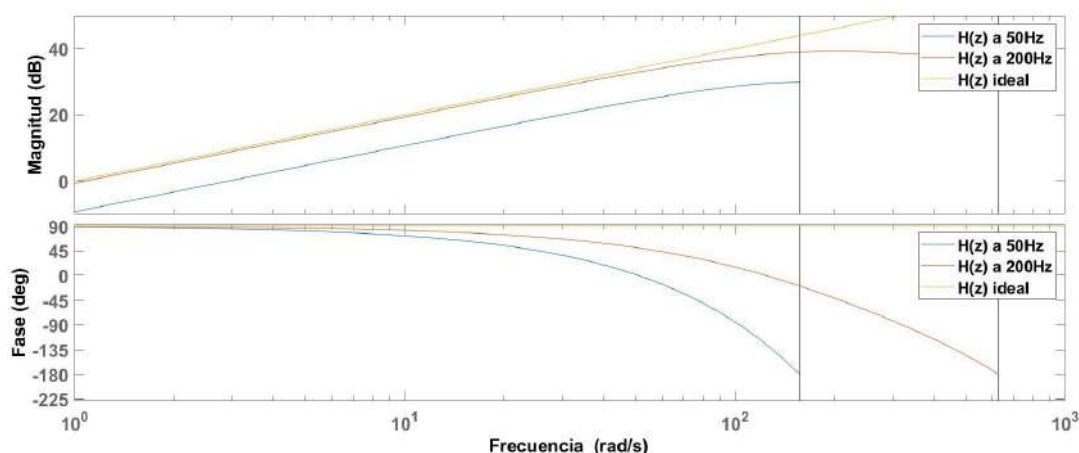


Figura E.2: Diagrama de bode para el filtro implementado con un ZOH para las frecuencias de muestreo de  $50 \text{ Hz}$  y  $200 \text{ Hz}$ . También se muestra un derivador ideal.

Es claro que con una frecuencia de muestreo de  $50 \text{ Hz}$  el resultado no es bueno. El módulo se encuentra uniformemente por debajo de lo esperado unos cuantos decibeles, lo cual es totalmente inaceptable. En la práctica esto se observaba viendo que el filtro obtenía la señal esperada pero con un factor de amplitud equivocado. Para el caso de  $200 \text{ Hz}$  el resultado es considerablemente mejor. Estos resultados no se consideran aceptables por lo que se decide estudiar que se obtiene si se utiliza un retenedor de primer orden (FOH).

Si estudiamos el sistema presentado en E.1 pero en lugar de un MOC utilizamos un FOH, entonces se obtiene un sistema cuya transferencia en tiempo discreto es

$$\overline{G(z)} = \frac{(z-1)^2}{z} \mathcal{Z} \left[ \mathcal{L}^{-1} \left\{ \frac{G(s)}{s^2} \right\} \Big|_{t=kT_s} \right]$$

Si aplicamos el resultado a la transferencia del filtro que se desea implementar se obtiene:

$$H(z) = \frac{(z-1)((1 - (1 + aT_s)e^{-aT_s})z + e^{-aT_s}(e^{-aT_s} + aT_s - 1))}{T_s(z - e^{-aT_s})^2}$$

Nuevamente  $a = 1/2\pi 30$  el polo doble y  $T_s$  el tiempo de muestreo del sistema. En la figura E.3 se muestra el diagrama de Bode para dicha transferencia tanto

para  $50\text{ Hz}$  como para  $200\text{ Hz}$ . Los resultados obtenidos son mucho mejor por lo que se decide utilizar FOH en la implementación del filtro en simulink.

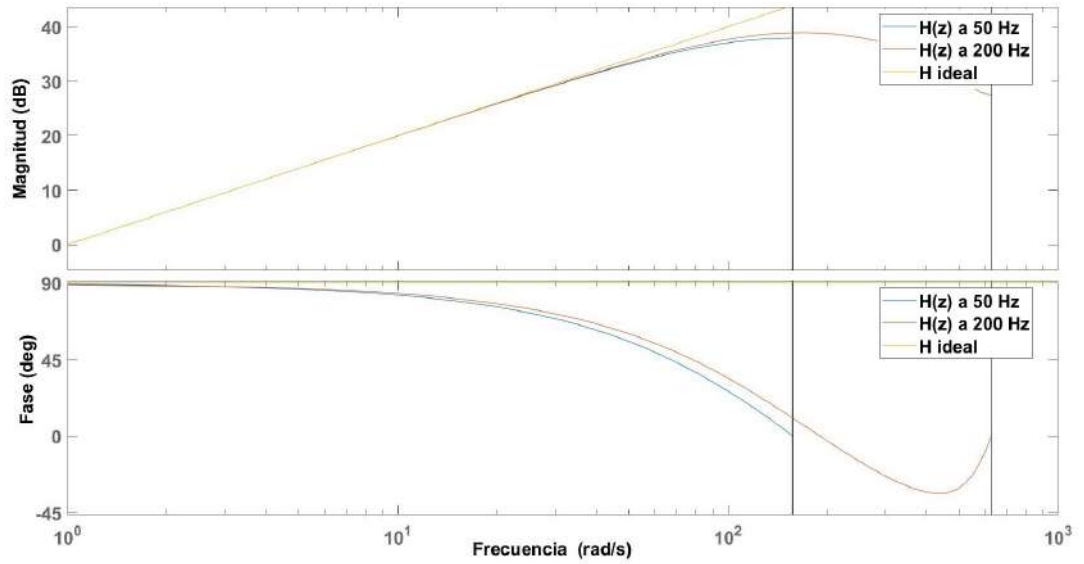


Figura E.3: Diagrama de bode para el filtro implementado con un FOH para las frecuencias de muestreo de  $50\text{ Hz}$  y  $200\text{ Hz}$ . También se muestra un derivador ideal.

Esta página ha sido intencionalmente dejada en blanco.

# Apéndice F

## Un descubrimiento

Ya en los finales del desarrollo de este proyecto se descubrió, accidentalmente, un relevo que permite reafirmar que las simulaciones explican las oscilaciones de baja frecuencia, y por lo tanto el hecho de que la causa de éstas sea la fricción. Como nos parece una pista importante para entender las oscilaciones decidimos presentar los resultados en esta sección. Un análisis profundo de los resultados queda totalmente por fuera del alcance de la tesis. Es por esto que esta evidencia no se ha utilizado en la sección de análisis.

Si se implementa el controlador 1 sobre el péndulo uno puede observar los resultados vistos en 7.1.5, el controlador estabiliza pero presenta vibraciones, más de las esperadas. Ahora si uno presiona *suavemente* con el dedo el encoder, de manera que el soporte en  $U$  se incline lo suficiente como para disminuir el juego presente en el eje del motor, entonces las vibraciones de alta frecuencia se extinguen, quedando solo las vibraciones que la simulación predijo.

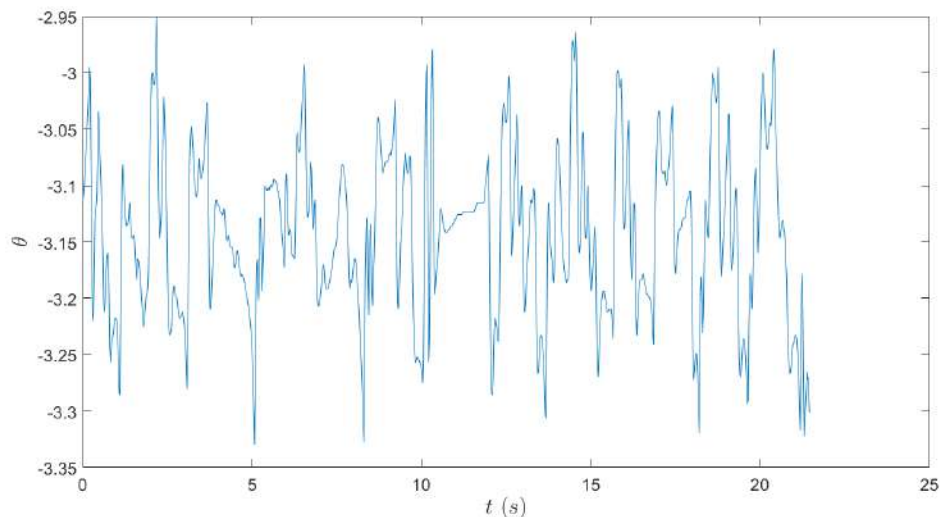


Figura F.1: Respuesta de  $\theta$  para el sistema con el controlador 1. Se presionó el encoder suavemente.

## Apéndice F. Un descubrimiento

En la figura F.1 y F.2 se observan los relevos de  $\theta$  y  $\varphi$ , en la figura F.3 y F.4 la respuesta frecuencial de éstos. Finalmente en la tabla F.1 mostramos los errores cuadráticos.

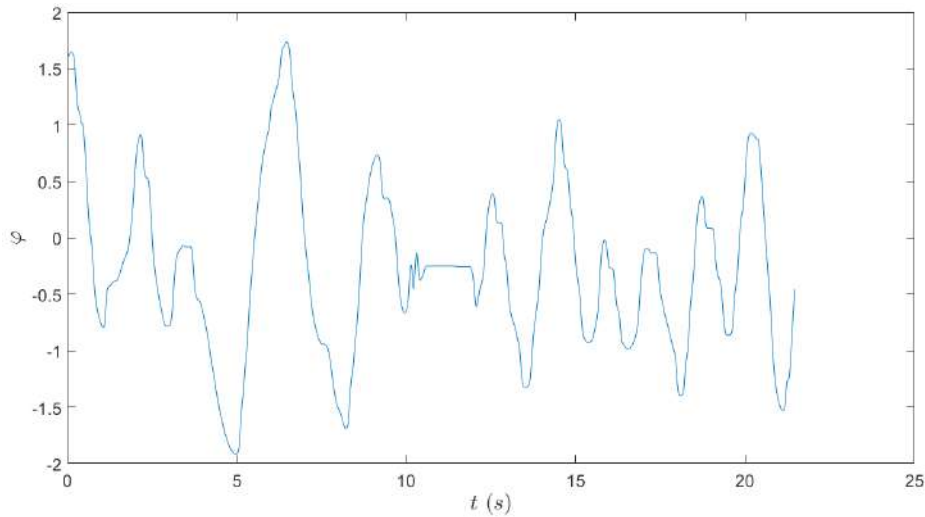


Figura F.2: Respuesta de  $\varphi$  para el sistema con el controlador 1. Se presionó el encoder suavemente.

A simple vista se ve que han disminuido las vibraciones. Para confirmar esto vemos la FFT.

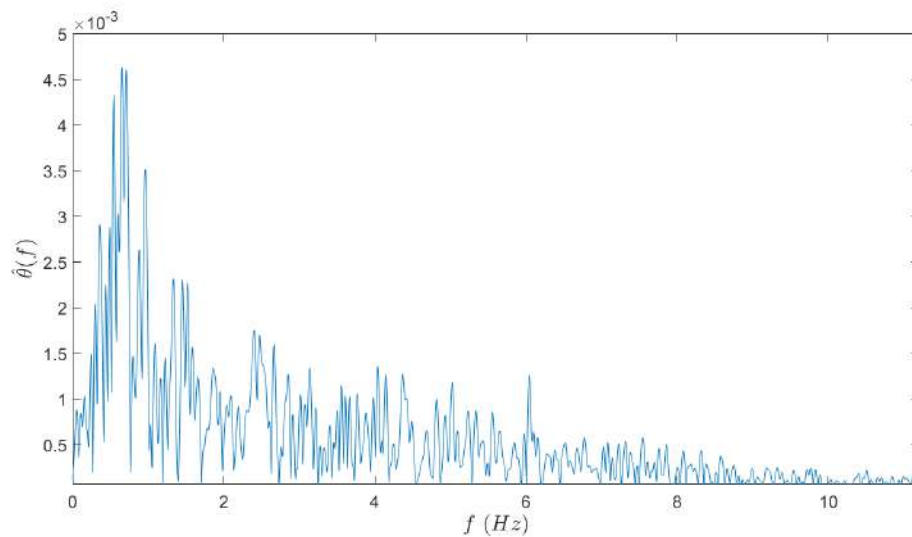


Figura F.3: Se observa la transformada de Fourier de  $\theta + \pi$ , donde  $\theta$  es el relevo del ángulo del péndulo para el controlador 1.

Comparando la figura F.3 con 7.10 se puede ver como la componente de alta frecuencia ha disminuido en más de 4 veces. Lo mismo se puede decir para  $\varphi$ .

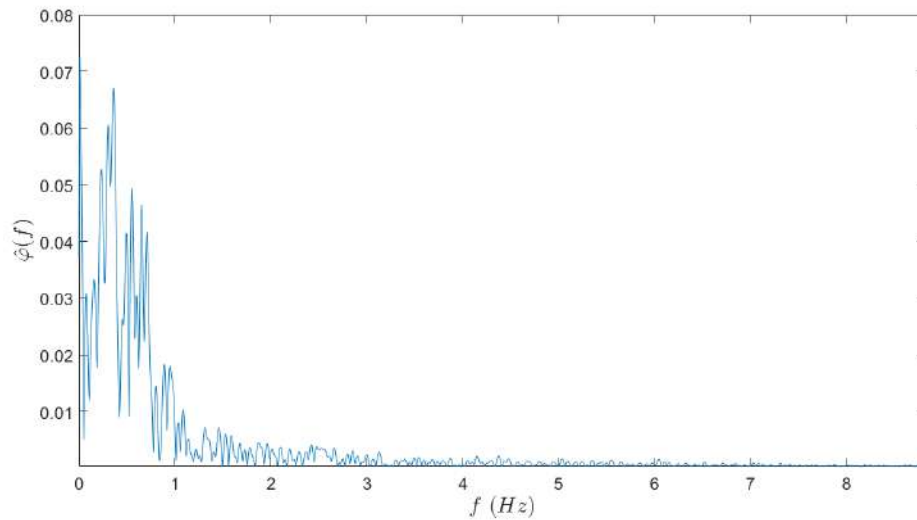


Figura F.4: Se observa la transformada de Fourier de  $\varphi$ , donde  $\varphi$  es el relevo del ángulo del motor para el controlador 1.

¡Entonces solo sobreviven las vibraciones que fueron predichas por la simulación!, las cuales se debían a la zona muerta que produce la fricción estática. Por último, igual que antes, presentamos el promedio y desviación del objetivo para este relevo.

Variable	Promedio		error	
	Relevo	Simulado	Relevo	Simulado
$\theta$ (rad)	3,14	3,14	0,026	0,033
$\varphi$ (rad)	-0,037	-0,012	0,30	0,25

Tabla F.1: Relevo con el encoder levemente presionado para el control 1, promedio y desviación del objetivo para las variables angulares.

¡Ahora los errores cuadráticos dados por la simulación y los errores relevados son bastante más parecidos que el relevo del capítulo 7!, siendo prácticamente iguales para  $\varphi$ . Esto apoya nuevamente la buena predicción del simulador y el modelo.

De un estudio tan breve e informal no se puede concluir que las vibraciones son causadas por el juego, pero esto sí apoya la hipótesis de que las componentes frecuenciales del orden del Hz son las debidas a la zona muerta, ya que la simulación logra reproducir los valores encontrados.

Esta página ha sido intencionalmente dejada en blanco.



# Referencias

- [1] K. Furuta, M. Yamakita, and S. Kobayashi, "Swing-up control of inverted pendulum using pseudo-state feedback," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 206, no. 4, pp. 263–269, 1992.
- [2] J. Acosta, "Furuta's pendulum: A conservative nonlinear model for theory and practise," *Mathematical Problems in Engineering*, vol. 2010, 2010.
- [3] J. L. A. S. R. Andrew Careaga Houck, Robert Kevin Katzschmann, "Furuta pendulum," in *2.151 Advanced System Dynamics & Control*, pp. 3–7, Department of Mechanical Engineering, Massachusetts Institute of Technology, 2013.
- [4] M. Antonio-Cruz, R. Silva-Ortigoza, C. Merlo-Zapata, M. G. Villarreal-Cervantes, D. Muñoz-Carrillo, and V. Hernández-Guzmán, "Modeling and construction of a Furuta pendulum prototype," in *2014 International Conference on Mechatronics, Electronics and Automotive Engineering*, pp. 98–103, IEEE, 2014.
- [5] "Rotary inverted pendulum." <https://www.quanser.com/products/rotary-inverted-pendulum/>.
- [6] S. Nair and N. E. Leonard, "A normal form for energy shaping: application to the furuta pendulum," in *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.*, vol. 1, pp. 516–521, IEEE, 2002.
- [7] B. S. Cazzolato and Z. Prime, "On the dynamics of the furuta pendulum," *Journal of Control Science and Engineering*, vol. 2011, 2011.
- [8] "Datasheet, acs712. fully integrated, hall effect-based linear current sensor with 2.1 kvrms voltage isolation and a low-resistance current conductor." <https://www.sparkfun.com/datasheets/BreakoutBoards/0712.pdf>.
- [9] "Datasheet, l298 dual full-bridge driver." [https://www.sparkfun.com/datasheets/Robotics/L298\\_H\\_Bridge.pdf](https://www.sparkfun.com/datasheets/Robotics/L298_H_Bridge.pdf).
- [10] "User guide, l298n dual h-bridge motor driver." <http://www.handsontec.com/dataspecs/L298N%20Motor%20Driver.pdf>.

## Referencias

- [11] “Snubber capacitors stop spikes.” <https://www.electronicdesign.com/power-management/power-protection/article/21120031/snubber-capacitors-stop-spikes>.
- [12] C. Dubilier, “Application guide: Snubber capacitors,” *Cornell Dubilier Electronics, Inc*, 2011.
- [13] R. Severns and E. Reduce, “Design of snubbers for power circuits,” *International Rectifier Corporation*, 2006.
- [14] “Estimacion de velocidad con encoders incrementales.” <https://forum.arduino.cc/index.php?topic=615619.0>.
- [15] A. Tilli and M. Montanari, “A low-noise estimator of angular speed and acceleration from shaft encoder measurement,” *AUTOMATIKA-ZAGREB*, vol. 42, no. 3/4, pp. 169–176, 2001.
- [16] “How to estimate encoder velocity without making stupid mistakes.” <https://www.embeddedrelated.com/showarticle/158.php>.
- [17] T. Lee, M. Leok, and N. H. McClamroch, *Global formulations of Lagrangian and Hamiltonian dynamics on manifolds*. Springer, 2017.
- [18] T. Lee, M. Leok, and H. McClamroch, “Geometric formulations of furuta pendulum control problems,” *Math. Engrg., Science and Aerospace (MESA)*, vol. 7, no. 1, 2016.
- [19] D. M. Ariel Fernández, *Apuntes de Mecánica Newtoniana*. Instituto de Física - Facultad de Ingeniería - Universidad de la República, 2010.
- [20] M. Antonio-Cruz, R. Silva-Ortigoza, J. Sandoval-Gutiérrez, C. A. Merlo-Zapata, H. Taud, C. Márquez-Sánchez, and V. M. Hernández-Guzmán, “Modeling, simulation, and construction of a furuta pendulum test-bed,” in *2015 International Conference on Electronics, Communications and Computers (CONIELECOMP)*, pp. 72–79, IEEE, 2015.
- [21] Y. Michitsuji, K. Furuta, and M. Yamakita, “Swing-up control of inverted pendulum using vibrational input,” in *Proceedings of the 2000. IEEE International Conference on Control Applications. Conference Proceedings (Cat. No. 00CH37162)*, pp. 226–231, IEEE, 2000.
- [22] M. Hamers and M. van de Molengraft, “Revitalizing the furuta pendulum,” *TU/e internship report*, 2004.
- [23] M. Gäfvert, *Modelling the furuta pendulum*. Department of Automatic Control, Lund Institute of Technology (LTH), 2016.
- [24] M. Gafvert, “Dynamic model based friction compensation on the furuta pendulum,” in *Proceedings of the 1999 IEEE International Conference on Control Applications (Cat. No. 99CH36328)*, vol. 2, pp. 1260–1265, IEEE, 1999.

- [25] V. Gupta, “Full state feedback of furuta pendulum.” <https://www.mathworks.com/matlabcentral/fileexchange/25585-full-state-feedback-of-furuta-pendulum>, 2020. MATLAB Central File Exchange. Retrieved February 24, 2020.
- [26] I. Fantoni and R. Lozano, *Non-linear control for underactuated mechanical systems*. Springer Science & Business Media, 2002.
- [27] K. Furuta and M. Iwase, “Swing-up time analysis of pendulum,” *Bulletin of the Polish Academy of Sciences: Technical Sciences*, vol. 52, no. 3, pp. 153–163, 2004.
- [28] X. Liu and G. S. Chen, *Friction Dynamics: Principles and Applications*, ch. 3, pp. 139–145, 129. Woodhead Publishing, 2016.
- [29] A. K. Padthe, J. Oh, and D. S. Bernstein, “On the lugre model and friction-induced hysteresis,” in *2006 American Control Conference*, pp. 6–pp, IEEE, 2006.
- [30] T. Tjahjowidodo, F. Al-Bender, and H. Van Brussel, “Friction identification and compensation in a dc motor,” *IFAC Proceedings Volumes*, vol. 38, no. 1, pp. 554–559, 2005.
- [31] T. Piatkowski, “Dahl and lugre dynamic friction models—the analysis of selected properties,” *Mechanism and Machine Theory*, vol. 73, pp. 91–100, 2014.
- [32] I. Virgala and M. Kelemen, “Experimental friction identification of a dc motor,” *International journal of mechanics and applications*, vol. 3, no. 1, pp. 26–30, 2013.
- [33] M. Bjekic, S. Antic, and A. Milovanovic, “Permanent magnet dc motor friction measurement and analysis of friction’s impact,” *Int. Rev. Electr. Eng*, vol. 6, no. 5, pp. 2261–2269, 2011.
- [34] “Hd 2 lubrication regimes.” <https://www.youtube.com/watch?v=pBdWcyP-UbA>.
- [35] A. Pascual, “EKF y UKF: dos extensiones del filtro de Kalman para sistemas no lineales aplicadas al control de un péndulo invertido.” Monografía para el curso Tratamiento Estadístico de Señales, 2004.
- [36] N. Guedelha, S. Traversaro, and D. Pucci, “Identification of motor parameters on coupled joints,” in *Proceedings of SAI Intelligent Systems Conference*, pp. 1153–1172, Springer, 2019.
- [37] L. C. Bo and D. Pavelescu, “The friction-speed relation and its influence on the critical velocity of stick-slip motion,” *Wear*, vol. 82, no. 3, pp. 277–289, 1982.
- [38] K. Ogata and Y. Yang, *Modern control engineering*, vol. 4. Prentice hall India, 2002.

## Referencias

- [39] E. D. Sontag, *Mathematical control theory: deterministic finite dimensional systems*, vol. 6. Springer Science & Business Media, 2013.
- [40] R. E. Kalman *et al.*, “Contributions to the theory of optimal control,” *Bol. soc. mat. mexicana*, vol. 5, no. 2, pp. 102–119, 1960.
- [41] “Furuta pendulum project.” <https://www.control.lth.se/education/engineering-program/frtn01-real-time-systems/projects/furuta-pendulum-project/>. Accessed: 2020-08-30.
- [42] H. K. Khalil and J. W. Grizzle, *Nonlinear systems*, vol. 3. Prentice hall Upper Saddle River, NJ, 2002.
- [43] C. C. Chung and J. Hauser, “Nonlinear control of a swinging pendulum,” *automatica*, vol. 31, no. 6, pp. 851–862, 1995.
- [44] K. J. Åström and K. Furuta, “Swinging up a pendulum by energy control,” *Automatica*, vol. 36, no. 2, pp. 287–295, 2000.
- [45] K. J. Åström, K. Furuta, M. Iwashiro, and T. Hoshino, “Energy based strategies for swinging up a double pendulum,” *IFAC Proceedings Volumes*, vol. 32, no. 2, pp. 6450–6455, 1999.
- [46] M. Westcott, “Swing-up and stabilisation of a furuta pendulum using energy control,” 2016. University of Exeter.
- [47] I. Fantoni and R. Lozano, “Stabilization of the furuta pendulum around its homoclinic orbit,” *International Journal of Control*, vol. 75, no. 6, pp. 390–398, 2002.
- [48] R. Tedrake, “Underactuated robotics: Learning, planning, and control for efficient and agile machines course notes for mit 6.832,” *Working draft edition*, vol. 3, 2009.
- [49] J.-J. E. Slotine, W. Li, *et al.*, *Applied nonlinear control*, vol. 199. Prentice hall Englewood Cliffs, NJ, 1991.
- [50] M. Hakas, “Sistemas de control en tiempo discreto.,” pp. 25 – 33.
- [51] K. Furuta, M. Yamakita, and S. Kobayashi, “Swing-up control of inverted pendulum using pseudo-state feedback,” *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 206, no. 4, pp. 263–269, 1992.
- [52] Y. Xu, M. Iwase, and K. Furuta, “Time optimal swing-up control of single pendulum,” *J. Dyn. Sys., Meas., Control*, vol. 123, no. 3, pp. 518–527, 2001.
- [53] W. P. Aung, “Analysis on modeling and simulink of dc motor and its driving system used for wheeled mobile robot,” *World Academy of Science, Engineering and Technology*, vol. 32, pp. 299–306, 2007.

## Referencias

- [54] “Is it possible to solve a nonlinear system with signum function using ode45?.” <https://www.mathworks.com/matlabcentral/answers/44752-is-it-possible-to-solve-a-nonlinear-system-with-signum-function-using->
- [55] “H-bridge motor driver - simulink.” <https://www.mathworks.com/help/physmod/sps/ref/hbridge.html>.

Esta página ha sido intencionalmente dejada en blanco.

# Lista de símbolos

A continuación se presenta la lista de los principales símbolos utilizados a lo largo del documento. La notación se repite, y a veces denotando distinto significado, es por esto que la lista se encuentra dividida en secciones que corresponden a los capítulos. Para usar la lista correctamente, a la hora de buscar un símbolo, se debe buscar en la sección de la lista correspondiente al capítulo donde el símbolo fue visto. En caso de no encontrarse buscar en la sección anterior más próximo.

---

$V_o$	Tensión de salida del sensor de corriente	23
$V_{cc}$	Tensión de alimentación del sensor de corriente	23
$s$	Sensibilidad del sensor de corriente	23
$e$	Tensión inducida en el bobinado del motor	27
$k$	Constante del motor	27
$w$	Velocidad angular en el eje del motor	27
$\tau$	Torque ejercido por el motor sobre el eje	27
$i$	Corriente por el motor	27
$f(w)$	Par de fricción sobre el eje del motor	27
$b$	Coefficiente de fricción viscosa	27
$i_0$	Corriente instantánea por el motor previo a que el driver imponga $i = 0$	27
$w_0$	Velocidad angular por el motor previo a que el driver imponga $i = 0$	27
$v$	Voltaje en bornes del motor. Señal de control del prototipo. Entrada.	27
$J$	Inercia del eje del motor	27
$\delta(t)$	Delta de dirac en $t = 0$	27
$L$	Bobinado del motor	27
$R$	Resistencia del motor	27
$\Delta P_{os}$	Cambio de posición angular medida por el encoder	33
$\Delta T$	Cambio de tiempo	33
$V_{in}$	Voltaje elegido por el microprocesador, la entrada o señal de control. También se le denota por $v$ en otras secciones	29
$V_{out}$	Voltaje medio en bornes del motor	29
$f$	Función que relaciona el voltaje en la entrada del driver con $V_{out}$	29
$\theta$	Señal a medir por el encoder	34
$\theta_a$	Señal adquirida por el encoder	34
$v, v_f$	Nodos del diagrama de bloques del controlador PI para el encoder	34
$k, k_p$	Constantes del controlador PI para el encoder	34
$\hat{x}, \mathcal{L}(x)$	Transformada de Laplace de la señal $x$	35

## Referencias

$s$	Variable de Laplace	35
$\mathbb{R}^n$	Espacio euclídeo n-dimensional	38
$m_{re}$	Masa del anillo exterior del rodamiento	41
$r_{re}$	Radio del anillo exterior del rodamiento	41
$a_{re}$	Ancho del anillo exterior del rodamiento	41
$m_{sr}$	Masa del soporte del rodamiento	41
$r_{sr}$	Radio del soporte del rodamiento	41
$a_{sr}$	Ancho del soporte del rodamiento	41
$l_{ui}$	Largo izquierdo del soporte en U, medido a partir del eje de rotación	41
$l_{ud}$	Largo derecho del soporte en U, medido a partir del eje de rotación	41
$a_u$	Ancho del soporte en U	41
$h_u$	Altura del soporte en U	41
$m_u$	Masa del soporte en U	41
$m_e$	Masa del encoder	41
$r_e$	Radio del encoder	41
$a_e$	Ancho del encoder	41
$l_p$	Largo del péndulo	41
$m_p$	Masa del péndulo	41
$l_b$	Largo del brazo	41
$m_b$	Masa del brazo	41
$l_{bi}$	Largo del brazo izquierdo, medido a partir del eje de rotación	41
$m_{ri}$	Masa del anillo interior del rodamiento	41
$d_{ar}$	Distancia desde el punto $A$ al rodamiento	41
$m_2$	Masa total del rígido 2	41
$\mathcal{B}$	Base solidaria al rígido 1 ( $\{x, y, z\}$ )	42
$\mathcal{C}$	Base solidaria al rígido 2 ( $\{x', y', z'\}$ )	42
$O$	Origen de coordenadas para derivar la ecuación del movimiento	42
$\vec{r}_P$	Vector que va desde $O$ a $P$	42
$I_O$	Tensor de inercia del rígido 1 en $O$	42
$K_i$	Energía cinética del rígido $i$	42
$\vec{\omega}_i$	Vector velocidad angular del rígido $i$	42
$J$	Momento de Inercia respecto al punto $O$ y al eje generado por $z$ del rígido 1	
$G$	Centro de masa del rígido 2	43
$I_A$	Tensor de inercia del rígido 2 en $A$	43
$\vec{r}_X$	Vector desde el origen $O$ al punto $X$	43
$\vec{v}_X$	Velocidad de $X$	43
$\theta$	Posición angular del péndulo	43
$\dot{\theta}$	Velocidad angular del péndulo	43
$\varphi$	Posición angular del brazo	43
$\dot{\varphi}$	Velocidad angular del brazo	43
$c_x, c_z$	Coordenadas de $\vec{r}_G - \vec{r}_A$ en la base $\mathcal{C}$	43
$I_x, I_y, I_z$	Términos de la diagonal de la matriz asociada a $I_A$ en la base $\mathcal{C}$	43



$V$	Energía potencial gravitatoria	44
$g$	Aceleración gravitatoria	44
$\mathcal{L}$	Lagrangiano	44
$\mathbf{q}$	Vector $(\theta, \varphi)^T$	44
$\tau$	Par medio que ejerce el motor sobre su eje	44
$\tau_{fp}$	Par de fricción en el soporte en el péndulo	44
$\tau_{fu}$	Par de fricción en el soporte en U	44
$M, D, F, U$	Matrices del modelo	44
$v$	Voltaje medio en bornes del motor. Señal de control. Entrada	47
$i$	Corriente media por el motor	47
$w$	Velocidad angular en el eje del motor	47
$d(\theta)$	Determinante de la matriz $M(\theta)$	49
$d$	El determinante anterior evaluado en $\theta = \pi$	49
$x_{eq}$	Punto de equilibrio correspondiente al péndulo en posición invertida	49

---

$b_2$	Coefficiente de fricción viscosa en el modelo de $\tau_{fp}$	52
$\zeta$	Factor de amortiguamiento en un sistema de segundo orden genérico	52
$w_n$	Frecuencia natural en un sistema de segundo orden genérico	52
$w_d$	Frecuencia real o frecuencia natural amortiguada de un sistema de segundo orden genérico	52
$x_i, x_{i+1}$	Amplitudes consecutivas de una senoide	52
$C_{ep}$	Coefficiente de fricción estática en el modelo de $\tau_{fp}$	53
$a$	Coefficiente del controlador PI utilizado para controlar la velocidad	58
$w_s$	Velocidad a seguir por el controlador / Velocidad de Stribeck en el modelo de $\tau_{fu}$	58/60
$b$	Coefficiente de fricción viscosa en el modelo de $\tau_{fu}$	61
$F_c$	Coefficiente de fricción de Coulomb en el modelo de $\tau_{fu}$	61
$F_s$	Coefficiente de fricción estática en el modelo de $\tau_{fu}$	61
$\hat{b}, \hat{F}_c, \hat{F}_s$	Coefficientes anteriores divididos por $k$ , la constante del motor	61
$L_0$	Valor inicial del parámetro $L$ del modelo del motor	62
$R_0$	Valor inicial del parámetro $R$ del modelo del motor	62
$R_{in}$	Resistencia de entrada del motor	62
$L_{in}$	Inductancia de entrada del motor	62
$k_0$	Valor inicial del parámetro $k$ del modelo del motor	63
$J_0$	Valor inicial del parámetro $J$ del modelo del péndulo	65

---

y	Señal que comunica el estado del sistema a los bloques de acción	84
modo	Señal utilizada para indicar en que modo debe funcionar el sistema, en el caso de implementar distintos modos	84
sh	Señal que en nivel alto le indica al Sample& Hold que retenga un valor	85
sp	Señal de salida del Sample& Hold. Con esta señal se suele indicar el valor de $\varphi$ a seguir por distintos controladores	85

---

## Referencias

---

$x \xrightarrow{T} z$	$z$ es alcanzable por $x$ en tiempo $T$	90
$x \rightsquigarrow z$	$z$ es alcanzable por $x$ en algún tiempo	90
$\mathcal{C}$	Grammiano de controlabilidad	91
$K$	Matriz de realimentación de estados	91
$Q, R$	Matrices de peso del controlador LQR	92
$A, B$	Matrices del modelo linealizado $\dot{x} = Ax + Bv$	93
$\lambda_i$	Polos del sistema, valores propios de $A$ (lazo abierto) o de $A - BK$ (lazo cerrado)	93,96
$k_i$	Entrada $i$ -ésima de la matriz $K$ . Coeficiente del controlador por realimentación de estados	96
$\hat{y}$	Transformada de Fourier de la señal $y$	
$E$	Energía del péndulo simple sin fricción	113
$u$	Se utiliza para denotar entradas de control de distintos sistemas. Para el prototipo completo se sigue usando $v$	114
$u_m$	Máximo de la señal de control $u$	114
$u_{sw}$	Ley de control para el swing up de un péndulo simple	115
$\text{sat}_a$	Función de $\mathbb{R} \rightarrow \mathbb{R}$ que asocia $x \mapsto a$ si $x > a$ , $x \mapsto -a$ si $x < -a$ y $x \mapsto x$ en otro caso	117
$f(\theta)$	Función ponderante del Swing Up	121
$k$	Constante multiplicativa de distintos controles, por ejemplo en el control final del swing up	121

# Índice de tablas

2.1.	Datos provistos por el vendedor del motor DC utilizado. . . . .	11
2.2.	Código de colores para conectores del PCB . . . . .	22
4.1.	Medidas . . . . .	41
4.2.	Izq: Composición de un rodamiento similar al nuestro. Der: Estimación del peso de las partes de nuestro rodamiento. . . . .	42
5.1.	Resultados del ajuste para cada relevo. Los relevos del 1 al 5 corresponden al péndulo partiendo de $\theta \approx 90^\circ$ , los relevos del 6 al 10 corresponden al péndulo partiendo de $\theta \approx 15^\circ$ . Los relevos 1 y 2 se descartan pues al inspeccionarlos se encontraron datos sospechosos de mala práctica. . . . .	56
5.2.	Resultados del ensayo para obtener K en el sentido de giro antihorario.	64
5.3.	Resultados del ensayo para obtener K en el sentido de giro horario.	64
5.4.	Corriente en estado estacionario. Forward y Backward refieren al motor girando en distintas direcciones. . . . .	68
5.5.	Tensión en régimen. Forward y Backward refieren al motor girando en distintas direcciones. . . . .	68
5.6.	Resumen de resultados de la sección motor . . . . .	69
5.7.	Parámetros utilizados para el modelo. . . . .	75
7.1.	Coefficientes de los controladores implementados y los polos de lazo cerrado. . . . .	96
7.2.	Promedio y desviación del objetivo para cada estado. Además se presenta la desviación de $x$ de su objetivo. . . . .	104
7.3.	Promedio y desviación del objetivo para las variables angulares. . .	106
7.4.	Coefficientes del controlador estabilizante utilizado en esta sección, el controlador 2 y sus polos. . . . .	129
F.1.	Relevo con el encoder levemente presionado para el control 1, promedio y desviación del objetivo para las variables angulares. . . .	157

Esta página ha sido intencionalmente dejada en blanco.

# Índice de figuras

1.1. Ilustración de un péndulo de Furuta. Extraída de [1] . . . . .	1
2.1. A la izquierda se observa el prototipo construido en [4] y a la derecha el de [3] . . . . .	8
2.2. Diseño del péndulo de Furuta, proyección explosiva. . . . .	9
2.3. Diseño del péndulo de Furuta, vista frontal y lateral. Las medidas son en <i>mm</i> . . . . .	9
2.4. Imagen del prototipo construido . . . . .	10
2.5. Ilustración del driver L298N y motor utilizado. . . . .	12
2.6. Foto del encoder utilizado para medir la posición del péndulo. . . .	13
2.7. Foto del componente utilizado con el sensor ACS712. . . . .	13
2.8. Foto de Arduino MEGA 2560. . . . .	14
2.9. Foto del acople del motor . . . . .	15
2.10. Nomenclatura del péndulo. . . . .	15
2.11. Diagrama lógico del sistema. . . . .	16
2.12. Diagrama de conexionado eléctrico. . . . .	17
2.13. Esquemático del circuito a realizar . . . . .	19
2.14. Diseño final del PCB . . . . .	20
2.15. Resultado final del PCB . . . . .	21
2.16. Diagrama de conexiones del PCB . . . . .	21
3.1. Circuito acondicionador para el sensor de corriente. Se observan los componentes con sus valores de diseño. Extraído de <a href="https://www.sparkfun.com/datasheets/Aplicación%203">https://www.sparkfun.com/datasheets/Aplicación 3</a> . . . . .	24
3.2. Salida del driver con el motor conectado vs salida en vacío. . . . .	25
3.3. Captura del osciloscopio: voltaje PWM entregado por el Driver en bornes del motor. . . . .	26
3.4. Esquemático del driver. Extraído de [9] . . . . .	26
3.5. Corriente entregada por el driver al motor. . . . .	27
3.6. Modelo Motor de corriente continua. . . . .	27
3.7. Resultados de la simulación . . . . .	28
3.8. Ejemplo de un diodo de rueda libre, para un driver sin puente en H. . . .	29
3.9. Solución Implementada . . . . .	29
3.10. Subida: relevo de $V_{in} \mapsto V_{out}$ con el compensador. Se hizo crecer $V_{in}$ de $-10 V$ a $10 V$ . . . . .	30

## Índice de figuras

3.11. Relevo de $V_{in} \mapsto V_{out}$ con el compensador. Se hizo decrecer $V_{in}$ de $10\text{ V}$ a $-10\text{ V}$ . . . . .	30
3.12. Comparación entre la curva de subida y la de bajada. Las diferencias se presentan en la zona muerta. . . . .	31
3.13. Ilustración del funcionamiento de un Encoder Incremental Óptico. Extraído <a href="https://www.sciencedirect.com/topics/engineering/incremental-encoder">https://www.sciencedirect.com/topics/engineering/incremental-encoder</a> . . . . .	32
3.14. Diagrama de bloques del filtro utilizado . . . . .	34
3.15. Relevo de respuestas al escalón del motor. . . . .	36
4.1. Nomenclatura utilizada sobre el péndulo de Furuta. . . . .	38
4.2. Soporte en U + Soporte del rodamiento: Se muestran algunas de sus medidas. Todas las medidas están en <i>cm</i> . . . . .	40
4.3. Péndulo + Brazo + rodamiento. Solo la parte interior del rodamiento forma parte del rígido 2. Todas las medidas están en <i>cm</i> . . . . .	40
4.4. Rígidos 1 y 2 acoplados: Coordenadas y bases utilizadas. . . . .	43
4.5. Modelo del motor DC sin carga. . . . .	47
5.1. Datos vs modelo obtenido. . . . .	52
5.2. Evidencia de fricción de Coulomb. . . . .	53
5.3. Resultado típico del método 2, en rosado se observa el relevo de datos y en naranja el modelo con los parámetros resultantes del método de ajuste. . . . .	54
5.4. En azul se observa un relevo de datos mientras que en naranja se observa el modelo con los parámetros que mejor ajustan los datos según el método 3. . . . .	55
5.5. Se compara el relevo 5 contra el modelo obtenido. . . . .	56
5.6. Se compara el relevo 6 contra el modelo obtenido. . . . .	57
5.7. Controlador PI para fijar la velocidad. . . . .	58
5.8. Corriente vs velocidad en régimen . . . . .	59
5.9. Curva Stribeck, en este caso torque vs velocidad angular. Tomada de [33]. . . . .	60
5.10. Ajuste lineal de la corriente vs velocidad . . . . .	61
5.11. Modelo ajustado para la fricción del motor con soporte en U. . . . .	62
5.12. Diagrama eléctrico del ensayo . . . . .	62
5.13. Respuesta al escalón de 3 a 4 V. Se utilizó el intervalo de tiempo $I_1$ marcado en negro. . . . .	66
5.14. Respuesta al escalón de 4 a 5 V. Se utilizó el intervalo de tiempo $I_2$ marcado en negro. . . . .	66
5.15. Respuesta al escalón de 8 a 9 V. Se utilizó el intervalo de tiempo $I_3$ marcado en negro. . . . .	67
5.16. Esquema del sistema . . . . .	69
5.17. Sistema vs Modelo: Respuesta de $\varphi$ . . . . .	70
5.18. Sistema vs Modelo: Respuesta de $\theta$ . . . . .	70
5.19. Sistema vs Modelo: Respuesta de $\dot{\varphi}$ . . . . .	71
5.20. Sistema vs Modelo: Respuesta de $\dot{\theta}$ . . . . .	72

5.21. Sistema vs Modelo: Respuesta de $i$ . . . . .	73
5.22. Respuesta dada por el motor+Soporte U frente a un seguidor de velocidad. Se observa el efecto de la zona muerta. . . . .	73
5.23. Curva de respuesta estacionaria tensión velocidad del motor+Soporte U. Se observa el efecto de la zona muerta. . . . .	74
6.1. Diagrama de comunicación del sistema. . . . .	77
6.2. Ilustración de los bloques implementados en Simulink . . . . .	79
6.3. Detalle de implementación del bloque <i>Encoder Péndulo</i> . . . . .	79
6.4. Bloque de lectura de corriente implementado en Simulink . . . . .	80
6.5. Detalle de implementación del bloque current sensor. . . . .	81
6.6. Imagen del script calibrador. . . . .	81
6.7. Detalle de implementación del calibrador. . . . .	82
6.8. Ilustraciones del bloque Actuador. . . . .	82
6.9. Implementacion del seguidor. . . . .	83
6.10. Ejemplo de una máquina de estado para manejar la lógica de funcionamiento . . . . .	85
6.11. Ilustración de un script que implementa todas las funcionalidades desarrolladas. . . . .	86
7.1. Flujo de trabajo para el diseño del controlador estabilizante. . . . .	94
7.2. Evolución de $\theta$ durante la estabilización. En la figura de arriba se observa el control 1 y en la de abajo el control 2. En ambas figuras se denota el objetivo con naranja. . . . .	97
7.3. Evolución de $\dot{\theta}$ durante la estabilización. En la figura de arriba se observa el control 1 y en la de abajo el control 2. . . . .	98
7.4. Evolución de $\varphi$ durante la estabilización. En la figura de arriba se observa el control 1 y en la de abajo el control 2. . . . .	99
7.5. Evolución de $\dot{\varphi}$ durante la estabilización. En la figura de arriba se observa el control 1 y en la de abajo el control 2. . . . .	100
7.6. Evolución de $i$ durante la estabilización. En la figura de arriba se observa el control 1 y en la de abajo el control 2. . . . .	101
7.7. Evolución de $v$ durante la estabilización. En la figura de arriba se observa el control 1 y en la de abajo el control 2. . . . .	102
7.8. Simulación: evolución de $\theta$ utilizando el controlador 2. La oscilación presenta una frecuencia de $0,5 \text{ Hz}$ . . . . .	105
7.9. Simulación: evolución de $\varphi$ utilizando el controlador 2. La oscilación presenta una frecuencia de $0,5 \text{ Hz}$ . . . . .	105
7.10. Se observa la transformada de Fourier discreta de $\theta - \pi$ , donde $\theta$ es el relevo del ángulo del péndulo para el controlador 1. En la figura se marcan ciertos puntos que delimitan las principales componentes frecuenciales de la señal. . . . .	107
7.11. Se observa la transformada de Fourier discreta de $\varphi$ , donde $\varphi$ es el relevo del ángulo del motor para el controlador 1. En la figura se marcan ciertos puntos que delimitan las principales componentes frecuenciales de la señal. . . . .	107

## Índice de figuras

7.12. Ilustración de una órbita homoclínica. . . . .	112
7.13. Diagrama de fase del péndulo simple sin controlar. En rojo se marca el objetivo y en punteado la curva de nivel 0 de la energía, la cual contiene una órbita homoclinica. . . . .	113
7.14. Diagrama de fase del sistema controlado. <i>Swing up</i> del péndulo simple con con $u$ dada por 7.5 y $k = 0,3$ . Se bosqueja además una órbita típica. . . . .	114
7.15. Diagrama de fase del sistema controlado. <i>Swing up</i> del péndulo simple con con $u$ dada por 7.5 y $k = 1$ . Se bosqueja además una órbita típica. . . . .	115
7.16. Evolución de $\theta$ . . . . .	118
7.17. Evolución de la energía del péndulo simple. . . . .	119
7.18. Diagrama de fase de $(\theta, \dot{\theta})$ . . . . .	119
7.19. De izquierda a derecha desde el origen al objetivo. . . . .	119
7.20. Bosquejo de la función $f(\theta)$ utilizada. . . . .	121
7.21. <i>Swing Up</i> , dinámica de $(\theta, \dot{\theta})$ junto con las curvas de nivel de energía. . . . .	122
7.22. <i>Swing Up</i> , evolución de la energía. . . . .	123
7.23. <i>Swing Up</i> , evolución de $\theta$ . . . . .	124
7.24. Dinámica inmediatamente luego del <i>Swing up</i> . Se utilizo $k = 37,5$ . . . . .	125
7.25. <i>Swing Up</i> , dinámica de $(\theta, \dot{\theta})$ . Se utilizó $k = 50$ . . . . .	125
7.26. Evolución de $\theta$ durante una caída. La curva que tiene menor tiempo de asentamiento es la relevada con controlador estabilizante de bajada. . . . .	127
7.27. Control supervisor . . . . .	128
7.28. Dinámica de $\theta$ y $\varphi$ durante <i>swing up</i> y luego control estabilizante. . . . .	130
7.29. Dinámica de $(\theta, \dot{\theta})$ , se observa la acción del <i>swing up</i> precedida por el control estabilizante, quién comienza actuar en el punto señalado. . . . .	131
A.1. Nomenclatura usada para las pistas del PCB . . . . .	139
C.1. Histograma de $J$ . . . . .	146
C.2. Salida del filtro para una rampa de pendiente $1 \text{ rad/s}$ con una señal uniforme superpuesta. . . . .	147
C.3. Histogramas para la salida del filtro para entradas de rampa + $\mathcal{U} \left[-\frac{ap}{2}, \frac{ap}{2}\right]$ . . . . .	147
E.1. Muestreo de un sistema continuo. Extraído de [50] . . . . .	151
E.2. Diagrama de bode para el filtro implementado con un ZOH para las frecuencias de muestreo de $50\text{Hz}$ y $200\text{Hz}$ . También se muestra un derivador ideal. . . . .	152
E.3. Diagrama de bode para el filtro implementado con un FOH para las frecuencias de muestreo de $50\text{Hz}$ y $200\text{Hz}$ . También se muestra un derivador ideal. . . . .	153
F.1. Respuesta de $\theta$ para el sistema con el controlador 1. Se presionó el encoder suavemente. . . . .	155



F.2. Respuesta de $\varphi$ para el sistema con el controlador 1. Se presionó el encoder suavemente. . . . .	156
F.3. Se observa la transformada de Fourier de $\theta + \pi$ , donde $\theta$ es el relevo del ángulo del péndulo para el controlador 1. . . . .	156
F.4. Se observa la transformada de Fourier de $\varphi$ , donde $\varphi$ es el relevo del ángulo del motor para el controlador 1. . . . .	157



Esta es la última página.  
Compilado el martes 26 enero, 2021.  
<http://iie.fing.edu.uy/>