

**UNIVERSIDAD NACIONAL DEL CALLAO  
FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA  
UNIDAD DE POSGRADO**



**“ESTIMACIÓN DEL MODELO IDGREY Y CONTROL DE  
VELOCIDAD POR INTERACCIÓN ADAPTATIVA APLICADO A UN  
MOTOR DC EMG30”**

**TESIS PARA OPTAR EL GRADO ACADÉMICO DE  
DOCTOR EN INGENIERÍA ELÉCTRICA**

**AUTOR: RICARDO RAÚL RODRÍGUEZ BUSTINZA**

**Callao, 2021**

**PERÚ**



## HOJA DE REFERENCIA DEL JURADO

### MIEMBROS DEL JURADO

Dr. MARCELO NEMESIO DAMAS NIÑO : PRESIDENTE  
Dr. JACOB ASTOCONDOR VILLAR : SECRETARIO  
Dr. NOE MANUEL JESÚS CHÁVEZ TEMOCHE : MIEMBRO  
Dr. CESAR AUGUSTO MEJÍA SANTOS : MIEMBRO  
Dr. NICANOR RAÚL BENITES SARAVIA : ASESOR

Nº DE LIBRO : 01  
FOLIO : 107  
FECHA DE APROBACIÓN : 18 de agosto 2021  
RESOLUCIÓN DIRECTORAL : 036-2021-DUPFIEE

## **DEDICATORIA**

A mi esposa Carmen Rosa

## **AGRADECIMIENTOS**

Deseo expresar mi agradecimiento a Dios por encontrarlo en mi vida y ser mi guía y darme siempre la luz. A mi nieta Vania por devolverme la alegría. Un agradecimiento especial a mi Maestro y amigo, Dr. NICANOR RAÚL BENITES SARAVIA por su apoyo incondicional en la revisión de la tesis.

# ÍNDICE

<b>HOJA DE REFERENCIA DEL JURADO</b> .....	i
<b>DEDICATORIA</b> .....	ii
<b>AGRADECIMIENTOS</b> .....	iii
<b>ÍNDICE DE TABLAS</b> .....	1
<b>ÍNDICE DE GRÁFICOS</b> .....	2
<b>RESUMEN</b> .....	3
<b>ASTRATTO</b> .....	4
<b>RESUMO</b> .....	5
<b>I. PLANTEAMIENTO DE LA INVESTIGACIÓN</b> .....	6
1.1 Descripción de la realidad problemática.....	6
1.2 Formulación del problema .....	7
1.2.1 Problema general.....	7
1.2.2 Problemas específicos .....	7
1.3 Objetivos de la investigación.....	7
1.3.1 Objetivo general .....	7
1.3.2 Objetivos específicos .....	7
1.4 Limitantes de la investigación .....	8
1.4.1 Teórico.....	8
1.4.2 Temporal .....	8
1.4.3 Espacial.....	8
<b>II. MARCO TEÓRICO</b> .....	9
2.1 Antecedentes .....	9
2.2 Bases teóricas.....	12
2.2.1 Base ontológica.....	12
2.2.2 Base metodológica .....	12
2.2.3 Base epistemológica .....	13
2.2.4 Base axiológica .....	13
2.2.5 Base ética .....	14
2.3 Conceptual.....	14
2.3.1 Introducción .....	14
2.3.2 Modelo matemático .....	15
2.3.3 Adquisición de datos .....	20
2.3.4 Estimación de modelos con IDGREY .....	24

2.3.5	Parametrizando modelo IDGREY .....	26
2.3.6	Modelo de una neurona biológica.....	29
2.3.7	Teoría de la interacción adaptativa .....	37
2.3.8	El algoritmo de Brandt-Lin .....	38
2.4	Definición de términos básicos .....	45
2.4.1	Glosario de términos básicos .....	45
2.4.2	Abreviaturas utilizadas .....	46
<b>III.</b>	<b>HIPÓTESIS Y VARIABLES</b> .....	<b>47</b>
3.1	Hipótesis .....	47
3.1.1	Hipótesis general .....	47
3.1.2	Hipótesis específicas.....	47
3.2	Definición conceptual de las variables.....	47
3.2.1	Variables Independientes .....	47
3.2.2	Variables dependientes.....	48
3.2.3	Variables intervinientes:.....	48
3.2.4	Operacionalización de variables.....	48
<b>IV.</b>	<b>DISEÑO METODOLÓGICO</b> .....	<b>49</b>
4.1	Tipo y diseño de investigación.....	49
4.1.1	Regresión lineal .....	49
4.1.2	Control a lazo abierto .....	53
4.1.3	Estimación del modelo motor DC EMG30.....	56
4.2	Método de investigación .....	66
4.3	Población y muestra .....	66
4.4	Lugar de estudio y periodo desarrollado .....	66
4.5	Técnicas e instrumentos para la recolección de la información.....	67
4.6	Análisis y procesamiento de datos.....	67
4.6.1	Registro de datos.....	67
4.6.2	Análisis del escalamiento .....	68
<b>V.</b>	<b>RESULTADOS</b> .....	<b>71</b>
5.1	Resultados descriptivos .....	71
5.2	Resultados inferenciales.....	73
5.2.1	Resultado 1. ....	73
5.2.2	Resultado 2. ....	74
5.2.3	Resultado 3.....	75
5.2.4	Resultado 4.....	77

<b>VI. DISCUSIÓN DE RESULTADOS</b> .....	79
6.1 Contrastación y demostración de la hipótesis con los resultados.....	79
6.1.1 Contrastación hipótesis 1: .....	79
6.1.2 Contrastación hipótesis 2: .....	79
6.1.3 Contrastación hipótesis 3: .....	80
6.3.1.1 Prueba de tiempo real 1 .....	81
6.3.1.2 Prueba de tiempo real 2 .....	82
6.3.1.3 Prueba de tiempo real 3 .....	83
6.3.1.4 Prueba de tiempo real 4 .....	84
6.2 Contrastación de la hipótesis con otros estudios similares .....	85
6.3 Responsabilidad ética de acuerdo con los reglamentos vigentes .....	86
<b>CONCLUSIONES</b> .....	87
<b>RECOMENDACIONES</b> .....	88
<b>BIBLIOGRAFÍA</b> .....	89
<b>ANEXO A</b> .....	91
<b>Matriz de Consistencia</b> .....	91
<b>ANEXO B</b> .....	93
<b>Programa en MATLAB</b> .....	93
<b>ANEXO C</b> .....	96
<b>Hoja Técnica del Motor DC EMG30</b> .....	96
- RB-Dev-40 Devantech 12V, 30:1 Gear Motor w/ Encoder .....	97
<b>ANEXO D</b> .....	100
<b>PROGRAMAS SIMULINK - LabVIEW</b> .....	100
- Diagrama del sistema de control por interacción adaptativa en Simulink. ....	101
- Diagrama del sistema de control por interacción adaptativa en LabVIEW.....	102
- Toolkit de Simulación por interacción adaptativa en LabVIEW DAQ Assistant. ....	102



## ÍNDICE DE TABLAS

Tabla N°01. Parámetros del Motor DC. ....	16
Tabla N°02. Parámetros cuantitativos del Motor DC EMG30. ....	19
Tabla N°03. Datos medidos desde el MAX para la entrada y salida análoga. ....	24
Tabla N°04. Características de las variables. ....	48
Tabla N°05. Algoritmo descriptivo.....	64
Tabla N°06. Valores del estado del foco de predicción. ....	71
Tabla N°07. Datos inferenciales 1.....	73
Tabla N°08. Datos inferenciales 2.....	74
Tabla N°09. Datos inferenciales 3.....	75
Tabla N°10. Datos inferenciales 3.....	77
Tabla N°11. Datos para pruebas 1 de tiempo real.....	81
Tabla N°12. Datos para pruebas 2 de tiempo real.....	82
Tabla N°13. Datos para pruebas 3 de tiempo real.....	83
Tabla N°14. Datos para pruebas 4 de tiempo real.....	84
Tabla N°15. Parámetros del Motor DC EMG30. ....	85
Tabla N°16. Pines de conectores del Motor DC EMG30. ....	98

## ÍNDICE DE GRÁFICOS

Figura N°01. Modelo del motor DC .....	15
Figura N°02. Modelo del motor DC en Simulink.....	19
Figura N°03. Simulaciones para la curva de velocidad en MATLAB.....	20
Figura N°04. Módulo físico del motor EMG30 y sus periféricos.....	20
Figura N°05. Módulo físico de la tarjeta NIDAQ USB-6009.....	21
Figura N°06. Firmware MAX de LabVIEW.....	21
Figura N°07. Completando el test del dispositivo.....	22
Figura N°08. Configurando canal de entrada analógica.....	22
Figura N°09. Configurando canal de salida analógica.....	23
Figura N°10. Curva de la velocidad a lazo abierto desde el MAX.....	23
Figura N°11. Diagrama de bloques de un motor DC controlado por armadura.....	27
Figura N°12. Neurona biológica.....	30
Figura N°13. Elementos básicos de la red neuronal.....	31
Figura N°14. Función de activación lineal.....	32
Figura N°15. Función de activación sigmoide.....	33
Figura N°16. Función de activación tangente hiperbólica.....	34
Figura N°17. Red neuronal multicapa 2-2-1.....	34
Figura N°19. Los dispositivos y sus interconexiones.....	39
Figura N°20. Red neuronal multicapa.....	41
Figura N°21. Configuración a lazo cerrado del control neuronal.....	42
Figura N°22. Controlador de la red neuronal.....	43
Figura N°23. Configuración del controlador neuronal adaptativo.....	43
Figura N°24. Control de velocidad desde entrada analógica de la DAQ USB-6009.....	49
Figura N°25. Trayectoria lineal para el móvil.....	50
Figura N°26. Línea recta desde la regresión.....	50
Figura N°27. Interpolando mediante una línea recta.....	52
Figura N°28. Diagrama de bloques para el control manual de la velocidad.....	54
Figura N°29. VI para el control de velocidad a lazo abierto.....	54
Figura N°30. Curva de la velocidad del motor EMG30 a lazo abierto.....	55
Figura N°31. VI para el control de velocidad a lazo abierto.....	56
Figura N°32. Gráficas de entrada y salida.....	58
Figura N°33. Comparación de respuestas debido a una entrada pulso.....	59
Figura N°34. Diagrama de bloques del motor DC con ganancia de tacómetro.....	60

Figura N°35. Curva característica del motor DC EMG30 a lazo abierto. ....	61
Figura N°36. Curva senoidal para cinco ciclos. ....	64
Figura N°37. Respuestas del sistema de control. ....	65
Figura N°38. Diagrama de LabVIEW para obtener datos de F-V y Voltaje AO. ....	68
Figura N°39. Curva de regresión lineal. ....	68
Figura N°40. Ecuación de la recta del escalamiento. ....	69
Figura N°41. Respuesta escalada de 0 a 5VDC en 450 muestras. ....	70
Figura N°42. Respuestas de adaptación 1 a una entrada senoidal. ....	73
Figura N°43. Respuestas de evolución de pesos sinápticos 1. ....	74
Figura N°44. Respuestas de adaptación 2 a una entrada senoidal. ....	75
Figura N°45. Respuestas de adaptación 3 a una entrada senoidal. ....	76
Figura N°46. Respuestas de evolución de pesos sinápticos 2. ....	76
Figura N°47. Respuestas de evolución de pesos sinápticos. ....	77
Figura N°48. Respuestas de evolución de pesos sinápticos 3. ....	78
Figura N°49. Verificación de performance de identificación de modelo. ....	80
Figura N°50. Front panel de las señales en pruebas 1 de tiempo real. ....	81
Figura N°51. Front panel de las señales en pruebas 2 de tiempo real. ....	82
Figura N°52. Front panel de las señales en pruebas 3 de tiempo real. ....	83
Figura N°53. Front panel de las señales en pruebas 4 de tiempo real. ....	84
Figura N°54. Respuestas comparativas del modelo del motor DC EMG30. ....	85
Figura N°55. Motor DC EMG30 .....	97
Figura N°56. Motor DC EMG30 con bracket de montaje. ....	97
Figura N°57. Esquema de disposición de pines del encoder. ....	98
Figura N°58. Esquema pull-up para salidas del encoder. ....	99

## RESUMEN

El presente proyecto de tesis doctoral corresponde al área de ingeniería aplicada, en el cual se propone el desarrollo del modelo experimental mediante la adquisición de datos de un motor de corriente continua, mediante la identificación de un modelo caja gris. Una manera de validar la exactitud del modelo experimental es resolviendo el problema del control de la velocidad del motor DC, en nuestro caso, se aplicará la técnica de control por interacción adaptativa propuesta por Brand-Lin aplicado a una red multicapa perceptron. La característica importante del algoritmo por interacción adaptativa, es que no requiere trabajar el algoritmo de propagación inversa, ni que la señal de error actualice los pesos, solo es necesario la red neuronal multicapa para su propagación en modo directo. Con el desarrollo del algoritmo por interacción adaptativa se busca mejorar el tiempo computacional y la performance para el control de la velocidad angular del motor DC, haciendo que la acción de control sea capaz de lograr un error en estado estacionario nulo, con mínimo sobre impulso y de respuesta rápida en un mínimo tiempo de establecimiento. Estos requerimientos que logra el algoritmo por interacción adaptativa, permitirán reducir el envejecimiento prematuro de la vida útil del motor DC.

Palabras claves: Adquisición de Datos, Estimación de Modelo, Algoritmo por Interacción Adaptativa, Redes Neuronales MLP. Control de Velocidad Angular.

El Autor

## ASTRATTO

Questo progetto di tesi di dottorato corrisponde all'area dell'ingegneria applicata, in cui viene proposto lo sviluppo del modello sperimentale acquisendo dati da un motore a corrente continua, identificando un modello a scatola grigia. Un modo per validare l'accuratezza del modello sperimentale è risolvere il problema del controllo della velocità del motore DC, nel nostro caso verrà applicata la tecnica di controllo dell'interazione adattiva proposta da Brand-Lin applicata a una rete multistrato di perceptron. La caratteristica importante dell' algoritmo di interazione adattativa è che non richiede l'algoritmo di propagazione inversa per funzionare, né che il segnale di errore aggiorni i pesi, solo la rete neurale multistrato è necessaria per la sua propagazione in modalità diretta. Con lo sviluppo dell'algoritmo di interazione adattativa, si cerca di migliorare il tempo di calcolo e le prestazioni per il controllo della velocità angolare del motore DC, rendendo l'azione di controllo in grado di raggiungere un errore di stato stazionario pari a zero, con un minimo impulso e risposta rapida in un tempo di installazione minimo. Questi requisiti, raggiunti dall'algoritmo di interazione adattativa, consentiranno di ridurre l'invecchiamento prematuro della vita utile del motore in corrente continua.

Parole chiave: Acquisizione dati, Stima del modello, Algoritmo di interazione adattiva, Reti neurali MLP. Controllo della velocità angolare.

L'autore

## RESUMO

O presente projeto de tese de doutoramento corresponde à área da engenharia aplicada, o desenvolvimento do modelo experimental é proposto através da aquisição de dados de um motor de corrente contínua, através da identificação do modelo caixa cinzenta. Uma forma de validar a acurácia do modelo experimental é resolvendo o problema de controle de velocidade do motor DC. No nosso caso, a técnica de controle de interação adaptativa proposta por Brand-Lin será aplicada através da aplicação de uma rede perceptron multicamadas. A característica importante do algoritmo de interação adaptativa é que ele não requer o funcionamento do algoritmo de propagação reversa, não requer o sinal de erro para atualizar os pesos, apenas é necessária a rede neural multicamadas para sua propagação em modo direto. Com o desenvolvimento do algoritmo de interação adaptativa, o objetivo é melhorar o tempo computacional e o desempenho para o controle da velocidade angular do motor DC, tornando a ação de controle capaz de atingir um erro de regime permanente nulo, com um mínimo de sobre-impulso e resposta rápida em tempo mínimo de configuração. Esses requisitos, alcançados pelo algoritmo de interação adaptativa, não permitirão o envelhecimento prematuro da vida útil do motor CC.

Palavras-chave: Aquisição de dados, Estimativa de modelo, Algoritmo de interação adaptativa, Redes Neurais MLP. Controle de velocidade angular.

O autor

# I. PLANTEAMIENTO DE LA INVESTIGACIÓN

## 1.1 Descripción de la realidad problemática

El presente trabajo de investigación nace de las experiencias desarrolladas en los campos de la estimación de modelos de sistemas lineales y no lineales en el área de la ingeniería de control en aplicaciones académicas e industriales. Uno de los problemas a resolver es la estimación del modelo del motor DC considerado como un sistema de caja gris (Grey-Box). Los modelos estimados por IDGREY derivan su estructura de los principios básicos o las percepciones físicas. El ejercicio de construir modelos IDGREY es, en gran medida, la estimación de parámetros tomando como principio a los modelos de caja blanca (White-Box) y modelos de caja negra (Black-Box). En modelos IDGREY, existen estimaciones de modelos en espacio estado lineales y no lineales, como en los sistemas de caja negra.

La obtención del modelo del motor DC EMG30 se realizará en forma experimental, adquiriendo los valores análogos del voltaje de entrada-salida mediante una interfase entre el motor DC y la computadora; esta interfase es la tarjeta de adquisición de datos DAQ USB-6009 del fabricante National Instruments. La exactitud del modelo es una parte importante en el presente trabajo de investigación, ya que, en nuestro caso, es requerida por la ley de control por interacción adaptativa aplicado a una red MLP, será capaz de resolver las condiciones de error en estado estacionario nulo con mínimo sobre impulso y un menor tiempo de establecimiento que son características idóneas para una buena performance del sistema de control. En resumen, el desarrollo del algoritmo por interacción adaptativa propone una metodología eficiente para llevar a cabo la optimización del gradiente descendente en el espacio paramétrico. A diferencia del enfoque basado en el algoritmo de propagación inversa que propaga la señal de error; el algoritmo por interacción adaptativa no requiere que se propague la señal de error, lo que resulta bastante beneficioso desde el punto de vista computacional, ya que no es necesario desarrollar el algoritmo de propagación inversa.

## 1.2 Formulación del problema

### 1.2.1 Problema general

¿De qué manera la estimación del modelo IDGREY y la aplicación del algoritmo por interacción adaptativa aplicado a una red neuronal multicapa contribuirán en la mejora de la performance del control de la velocidad del motor DC?

### 1.2.2 Problemas específicos

- a. Problema específico 1: ¿Cómo influye la obtención del modelo experimental en la actuación del control de la velocidad del motor DC EMG30?
- b. Problema específico 2: ¿Cómo afecta la estimación IDGREY cuando se compara con otros modelos provenientes de identificación paramétrica recursiva para sistemas Black-Box aplicados en modelos teóricos del motor DC?
- c. Problema específico 3: ¿Cómo se valida la performance del sistema de control de velocidad del motor DC mediante la técnica de control por interacción adaptativa aplicado a una red multicapa?

## 1.3 Objetivos de la investigación

### 1.3.1 Objetivo general

Estimar el modelo experimental IDGREY y desarrollar el algoritmo por interacción adaptativa aplicado al control de velocidad de un motor DC EMG30.

### 1.3.2 Objetivos específicos

- a. Objetivo Específico 1: Implementar un sistema de adquisición de datos en base a las plataformas de programación, LabVIEW y MATLAB para el procedimiento de estimación del modelo del sistema de control.



- b. Objetivo Específico 2: Encontrar el modelo estimado del proceso mediante un análisis de datos experimentales, y el desarrollo del algoritmo de caja gris del Toolbox de MATLAB.
  
- c. Objetivo Específico 3: Desarrollo del algoritmo de control por interacción adaptativa aplicando una red neuronal multicapa y verificación de la actuación de la señal de control del motor DC.

#### 1.4 Limitantes de la investigación

##### 1.4.1 Teórico

Como, limitaciones teóricas, se establece que debido a que no se cuenta con fuentes de información bibliográfica suficiente que trate el tema relacionado con la técnica de control por interacción adaptativa aplicado a una red MLP. Fue necesario acudir a las fuentes informativas, artículos, papers, y portales de internet que han sido citados por Robert D. Brandt and Feng Lin en el desarrollo de su estudio.

##### 1.4.2 Temporal

En nuestro proyecto de tesis, se requiere métodos cuantitativos, que consiste en estudiar los fenómenos experimentales que pudieran ocurrir como es la asignación de la frecuencia de muestreo de la procedencia de los datos y no permitir la ocurrencia del fenómeno del aliasing.

##### 1.4.3 Espacial

Una limitante espacial concierne al área geográfica en donde se desarrollará la investigación. Siempre será una limitante no contar con laboratorios, instrumentos, equipos y dispositivos que nos permitan realizar las pruebas experimentales y poder validar los resultados. Por ello se diseñaron las tarjetas electrónicas y las pruebas de laboratorio en los ambientes de ISCA & Electronics.

## II. MARCO TEÓRICO

### 2.1 Antecedentes

#### Investigaciones Internacionales

- GEORGE SAIKALIS, FENG LIN. “A Neural Network Controller by Adaptive Interaction”. Hitachi America, Ltd., R&D, Farmington Hills Michigan 48323. Wayne State University, ECE Dept., Detroit. El artículo, propone un acercamiento a los controladores basados en redes neuronales mediante el uso de un nuevo algoritmo de adaptación. El algoritmo se deriva de la teoría de interacción adaptativa. El principio detrás del algoritmo de adaptación es simple pero eficiente desde el punto de vista metodológico, ya que realiza la optimización del descenso de gradiente en el espacio paramétrico, (A Neural Network Controller by Adaptive Interaction, 2001).
- KUMPATI S. NARENDRA, AND KANNAN PARTHASARATHY. “Identification and Control of Dynamical Systems Using Neural Networks”. El Paper demuestra que una red neuronal es empleada para identificar y controlar un sistema dinámico no lineal, (Identification and Control of Dynamical Systems using Neural Networks, 1990).
- ESFANDIARI, RAMIN S., AND LU, BEI.” Modeling and Analysis of Dynamic Systems.” Taylor & Francis, CRC Press, (2018). El libro propone métodos de identificación de sistemas dinámicos usando métodos paramétricos y no paramétricos, (ESFANDIARI, 2018) (Optimization of Dynamical Systems Containing Neural Networks, 1991).
- KUMPATI S. NARENDRA, AND KANNAN PARTHASARATHY. “Optimization of Dynamical Systems Containing Neural Networks”. IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 2, NO. 2, MARCH 1991. El Paper propone algoritmo de optimización a los sistemas dinámicos no lineales, (Optimization of Dynamical Systems Containing Neural Networks, 1991).

- FU-CHUANG CHEN, AND HASSAN K. KHA, “Adaptive Control of a Class of Nonlinear Discrete-Time Systems Using Neural Networks”. JEE TRANSACTIONS ON AUTOMATIC CONTROL, VOL. 40, NO. 5, MAY 1995. El Paper analiza las redes neuronales en capas se utilizan de forma no lineal en la solución del problema de control adaptativo autoajustable, (Adaptive Control of a Class of Nonlinear Discrete-Time Systems Using Neural Networks, 1995).
- FENG LIN, ROBERT D. BRANDT, AND GEORGE SAIKALIS, Proceedings of the American Control Conference. “Self-Tuning of PID Controllers by Adaptive Interaction”. Chicago, Illinois June 2000. El Paper propone un nuevo algoritmo de autoajuste o adaptación para controladores PID basados en una teoría de interacción adaptativa, (Self-Tuning of PID Controllers by Adaptive Interaction, 2000).
- ROBERT D. BRANDT A, FENG LIN B, “Adaptive interaction and its application to neural networks an Intelligent Devices”, Inc., 465 Whittier Ave., Glen Ellyn, IL 60137, USA Electrical and Computer Engineering, Wayne State University, 5050 Anthony Wayne Drive, Detroit, MI 48202, USA. El Paper analiza la interacción adaptativa es un nuevo enfoque para introducir la adaptabilidad en sistemas inteligentes. En este enfoque, un sistema se descompone en subsistemas interconectados que llamamos dispositivos y la adaptación ocurre en las interacciones de los dispositivos, (Adaptive interaction and its application to neural networks an Intelligent Devices, 2001).
- ROJAS ARTURO, Control Avanzado Diseño y Aplicaciones en Tiempo Real, Universidad Nacional de Ingeniería, Publicación independiente, 2001. El autor desarrolla algoritmos de control avanzado usando técnicas de control no lineal, (Control Avanzado Diseño y Aplicaciones en Tiempo Real, 2001).
- PHILLIPS - NAGLE. “Control System Analysis and Design”. Ed. Prentice Hall. 4th edition. 2015. El autor resuelve el problema del algoritmo RLS, Recursive Least Square, (Control System Analysis and Design, 2015).

- FRANKLIN – POWELL - EMAMI-NAEINI. “Feedback Control of Dynamics System”. Addison Wesley. 8th edition. 2019. Libro de consulta con tópicos de control de sistemas dinámicos, (Feedback Control of Dynamics System, 2019).
- ALLEN LOUIS EDWARDS, “An introduction to linear regression and correlation”, W H Freeman & Co; 2nd edition June 1, 1984. El autor propone métodos numéricos de ingeniería para resolver la regresión lineal, (An introduction to linear regression and correlation, 1984).
- NATIONAL INSTRUMENTS. USER GUIDE NI USB-6008/6009 Bus-Powered Multifunction DAQ USB Device. Guía técnica para configurar la tarjeta de adquisición de datos, (USER GUIDE NI USB-6008/6009 Bus-Powered Multifunction DAQ USB Device, 2016).
- MATHWORKS. Control System Toolbox. 2017. Documento de MathWorks relacionado con las librerías en inteligencia Artificial y Sistemas de Control, (Control System Toolbox, 2017).
- JOSÉ GONÇALVES, JOSÉ LIMA, PAULO J. COSTA AND A. PAULO MOREIRA. Modeling and simulation of the EMG30 Geared motor with encoder resorting to SimTwo: The official Robot@Factory Simulator. 1 INESC TEC (formerly INESC Porto) and Polytechnic Institute of Bragança, Portugal, (Modeling and simulation of the EMG30 Geared motor with encoder resorting to SimTwo: The official Robot@Factory Simulator, 2011).

#### Investigaciones Nacionales

- R. RODRIGUEZ BUSTINZA, V. PAREDES, G. NOA, L. TRUCIOS, 2019 8th International Conference on Mechatronics and Control Engineering IOP Conf. Series: Materials Science and Engineering 707 (2019) 012003 IOP Publishing doi:10.1088/1757-899X/707/1/012003, (Grey Box Identification and Adaptive Control in a Water Level System, 2019).

- CARDOZO GALVEZ, ERICK OCTAVIO, “Diseño de un Controlador PID Con Auto sintonía Basado En Un Modelo De Redes Neuronales Dinámicas Y Control Adaptativo” Tesis Universidad Ricardo Palma, 2018. Expone el diseño de un control PID basado en redes neuronales, (Diseño de un Controlador PID Con Auto sintonía Basado En Un Modelo De Redes Neuronales Dinámicas y Control Adaptativo, 2018).

## 2.2 Bases teóricas

### 2.2.1 Base ontológica

El fundamento ontológico, representa la razón del ser de plantear y obtener un resultado experimental mediante la estimación y control por interacción adaptativa aplicado a un actuador, tratando de alcanzar la mejor performance de sistema en general, en la sinergia del sistema.

Desde el contexto de la naturaleza de la realidad o verdad, podemos indicar que la investigación efectuada ha estado en la búsqueda de obtener la mejora y de la acción de control para que el sistema siga a las referencias constantes y trayectorias arbitrarias como una forma de medir la performance en la convergencia del error.

### 2.2.2 Base metodológica

El presente trabajo, por ser considerado en la rama de la ingeniería, usa como método las mediciones de los datos del sistema físico para encontrar un modelo experimental siendo de carácter cualitativo. Además, se realizan los diseños y cálculos referidos a la investigación.

La elaboración del marco metodológico, al igual que el planteamiento del problema y el marco teórico forman las bases de la investigación y su relevancia e importancia son muy alta. Engloba y justifica los aspectos operacionales, técnicos y procedimentales que llevaremos a cabo durante el proceso de la investigación, comprende los siguientes aspectos.

- Enfoque de la investigación
- Diseño de la investigación
- Alcance de la investigación
- Población y muestra
- Técnicas e instrumentos de recolección de datos
- Validación

### 2.2.3 Base epistemológica

La base epistemológica busca el desarrollo teórico y práctico del conocimiento en la obtención del modelo experimental que permita al controlador mediante una ley de control accionar con amplia libertad dando indicadores de robustez en el desempeño del sistema en tareas asignadas.

Es importante tener presente que la información que se tiene y se va a utilizar debe ir de acuerdo de los avances y adelantos tecnológicos que se van sucediendo. Considerando especialmente la investigación realizada que tenga que ver con la naturaleza del conocimiento y explicar cómo se obtuvo el conocimiento.

### 2.2.4 Base axiológica

En lo que respecta a la base axiológica podemos indicar, el valor de crear el trabajo de investigación y realizarlo, siendo un rol protagónico de nuestro ser. Es básicamente un producto de la mente humana, que nos permite valorar y mostrar los resultados para recibir las apreciaciones individuales y colectivas de los especialistas.

El valor que le damos a nuestro trabajo está relacionado con el deseo que resulte una investigación exitosa y sobre todo que sea de interés en las actividades académicas e industriales. Los factores que intervienen es la experiencia personal que se tiene del tema, la opinión u opiniones recibidas de los especialistas.

### 2.2.5 Base ética

Desde el aspecto de la ética es la ciencia en la cual se experimenta con elementos y una vez empleada es normal que la filosofía esté relacionada con la ética. El hombre ha hecho grandes avances científicos y ha creado maravillas, entonces, dependerá del hombre analizar donde y en que empleará sus conocimientos para obtener beneficios sin perjudicar los valores morales.

En este sentido, podemos indicar que la ciencia es una actividad social por excelencia y en su ejercicio se presentan numerosas situaciones en las que se deben aplicar principios éticos. No sólo se trata de cumplir con las obligaciones morales del hombre hacia los demás, sino también el ejercicio del método científico, que tiene como meta arribar a la verdad del conocimiento. También podemos indicar el respeto irrestricto por la autoría de los trabajos de investigación en el mundo que nos compromete a someternos en una disciplina que ayudara de sobremanera a conducirnos con amplios valores éticos.

## 2.3 Conceptual

Las bases teóricas proporcionan los fundamentos epistemológicos, metodológicos y ontológicos de las teorías vigentes conforme a las variables, características o dimensiones establecidas convencionalmente por la ciencia y tecnología. En nuestro caso, trataremos estos fundamentos dentro de las bases teóricas que formula, plantea la solución y la resuelve dentro de un marco de investigación científica.

### 2.3.1 Introducción

Para resolver problemas relacionados con la identificación del modelo del sistema motor DC, es necesario adquirir datos que provengan desde los experimentos. La interfase es la encargada de comunicar el sistema con el computador a través del puerto USB y con el proceso a través de los puertos analógicos que miden voltaje de entrada/salida al motor DC, de ese modo los archivos almacenados harán posible procesarlos desde el software de simulación grafica o basado en texto.

Muchas veces tenemos un motor DC y no sabemos su procedencia ni sus características de placa, en este caso se trata de un sistema de caja negra, mientras que, en algunos casos, conocemos algunos parámetros nominales del motor, estos serán muy útiles para una estimación del modelo caja gris, ya que en conjunto con el modelo representado en espacio estado complementarán convenientemente la estrategia de la estimación del modelo.

Algunos fabricantes de motores DC realizan una venta a usuarios (empresas) que desean confidencialidad en las características del motor DC, en este caso estamos ante un caso de caja negra. Algunos otros fabricantes proporcionan los parámetros del motor DC que han sido obtenidos desde el laboratorio, el tema de identificar no sería relevante ya que se trata de un sistema caja blanca, sin embargo, sería útil, de estimar el modelo debido a que se validarían los algoritmos y mediante las pruebas verificar la performance. Dentro del contexto de las bases teóricas también están las relacionadas con los tópicos de las redes neuronales multicapa, sus relaciones matemáticas del modelo e interpretación de la estructura a emplear; así mismo en la tesis esta metodología es propia del control por interacción adaptativa los cuales son el soporte de la investigación de carácter cualitativo.

### 2.3.2 Modelo matemático

El modelo matemático lineal de un motor DC es muy popular y lo encontramos en muchas publicaciones y libros en el mundo. El motor DC es un sistema electromecánico (ver Figura N°01) que tiene una parte eléctrica a la entrada y otra parte mecánica a través del eje que nos proporciona la velocidad y posición angular en el eje.

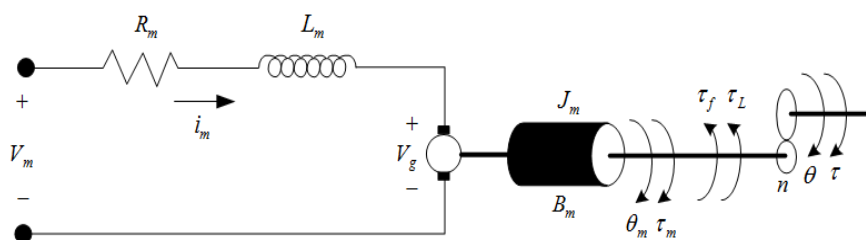


Figura N°01. Modelo del motor DC

Fuente: Elaboración propia.



El modelo del motor DC usualmente viene representado por una función de transferencia que es la relación de la velocidad angular  $\omega$  y la entrada de voltaje  $u$ , las unidades de la función de transferencia están en rad/s. Muchos autores para su estudio del motor DC utilizan parámetros normalizados y despreciando la inductancia de armadura, esto conlleva a las funciones de transferencia  $G(s)$  que viene representadas por.

$$G(s) = \frac{\Theta(s)}{V(s)} = \frac{1}{s(s + 1)}$$

En nuestro caso se hará un estudio desde la perspectiva de la parte eléctrica y mecánica desarrollando el modelo matemático que involucran las leyes de voltaje de Kirchhoff, así como las relaciones electromecánicas (fuerza contraelectromotriz) que están relacionadas con la corriente y torque del motor DC. El motor DC es un sistema electromecánico, por ello para expresar su modelo dinámico se requiere resolver ecuaciones eléctricas y electromecánicas. La Tabla N°01 nos muestra los parámetros del motor DC que usaremos en el modelamiento.

Tabla N°01. Parámetros del Motor DC.

Parámetros	Notación	Unidades
Inductancia de armadura	$L_m$	$H$
Resistencia de armadura	$R_m$	$\Omega$
Tensión contra electromotriz	$V_g$	$V$
Tensión de entrada	$V_m$	$V$
Corriente de armadura	$i_m$	$A$
Constante contra electromotriz	$K_b$	$V/rad/seg$
Constante de motor	$K_m$	$Nm/A$
Inercia de motor	$J_m$	$Kg - m^2$
Coefficiente de fricción	$B_m$	$Nm/rad/seg$
Torque de fricción	$\tau_f$	$Nm$
Torque de carga	$\tau_L$	$Nm$
Torque de motor	$\tau_m$	$Nm$

Velocidad angular	$\omega_m$	<i>rad/seg</i>
Posición angular	$\theta_m$	<i>rad</i>
Coeficiente de reducción	$n$	
Torque del eje del motor	$\tau$	<i>Nm</i>
Angulo de posición angular	$\theta$	<i>rad</i>

La relación de voltaje de armadura viene definida según la ecuación diferencial.

$$V_m = L_m \frac{di_m}{dt} + R_m i_m + V_g \quad (1)$$

La relación de la tensión contraelectromotriz es.

$$V_g = K_b \omega_m \quad (2)$$

Reemplazando (2) en (1).

$$V_m = L_m \frac{di_m}{dt} + R_m i_m + K_b \omega_m \quad (3)$$

La ecuación del torque de motor DC es.

$$\tau_m = K_t i_m = \tau_f + J_m \frac{d\omega_m}{dt} + B_m \omega_m + \tau_L \quad (4)$$

Igualando la potencia eléctrica y la potencia mecánica.

$$V_g i_m = \tau_m \omega_m \quad (5)$$

Reemplazando (2) y (4) en (5).

$$K_b \omega_m i_m = K_t i_m \omega_m \quad (6)$$

De este modo obtenemos.

$$K_b = K_t \quad (7)$$

Aplicando la transformada de Laplace a las ecuaciones (3), (4) y (7).

$$(L_m s + R_m)I_m(s) + K_b \omega_m(s) = V_m(s) \quad (8)$$

$$K_b I_m(s) - (J_m s + B_m)\omega_m(s) = \tau_f(s) \quad (9)$$

Encontrando la función de transferencia de la velocidad angular, para ello despejamos la corriente de (8).

$$I_m(s) = \frac{(J_m s + B_m)}{K_b} \omega_m(s) + \frac{1}{K_b} \tau_f(s) \quad (10)$$

Reemplazando (10) en (8).

$$(L_m s + R_m) \left\{ \frac{(J_m s + B_m)}{K_b} \omega_m(s) + \frac{1}{K_b} \tau_f(s) \right\} + K_b \omega_m(s) = V_m(s) \quad (11)$$

Resolviendo para la velocidad angular  $\omega_m(s)$ .

$$\left\{ \frac{(L_m s + R_m)(J_m s + B_m) + K_b^2}{K_b} \right\} \omega_m(s) = V_m(s) - \frac{(L_m s + R_m)}{K_b} \tau_f(s)$$

$$\omega_m(s) = \frac{K_b}{(L_m s + R_m)(J_m s + B_m) + K_b^2} V_m(s) - \frac{(L_m s + R_m)}{(L_m s + R_m)(J_m s + B_m) + K_b^2} \tau_f(s) \quad (12)$$

Asumiendo el torque de fricción  $\tau_f(s)$  igual a cero obtenemos la función de transferencia del modelo lineal de velocidad del motor DC según.

$$\frac{\omega_m(s)}{V_m(s)} = \frac{K_b}{(L_m s + R_m)(J_m s + B_m) + K_b^2} \quad (13)$$

El modelo del motor DC en Simulink es mostrado en la Figura N°02.

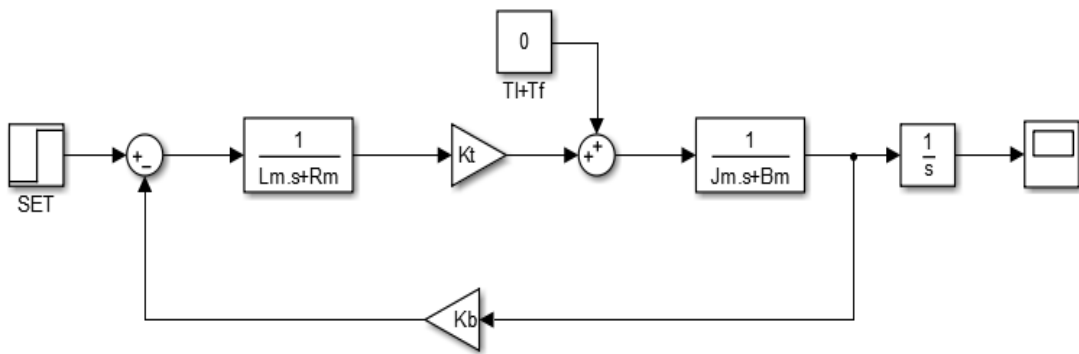


Figura N°02. Modelo del motor DC en Simulink.

Fuente: Elaboración propia.

Consideraremos los parámetros de un motor DC de 12V definido en la Tabla N°02.

Tabla N°02. Parámetros cuantitativos del Motor DC EMG30.

$L_a$	Inductancia de armadura	H	$3.4 \times 10^{-3}$
$R_a$	Resistencia de armadura	$\Omega$	7.101
$J$	Inercia	$\text{Kg} - \text{m}^2$	0.00567
$B$	Fricción viscosa	$\text{Nm/rad/s}$	0.000391
$K_t$	Constante de motor	$\text{Nm/A}$	0.509
$K_b$	Constante contra electromotriz	$\text{V/rad/s}$	0.509

Las simulaciones son realizadas usando el software MATLAB. La entrada es una señal escalón unitario aplicando la superposición (ver Figura N°02) cuando se incluye el torque de fricción como lo demuestra las simulaciones para la velocidad angular según lo mostrado en la Figura N°03.

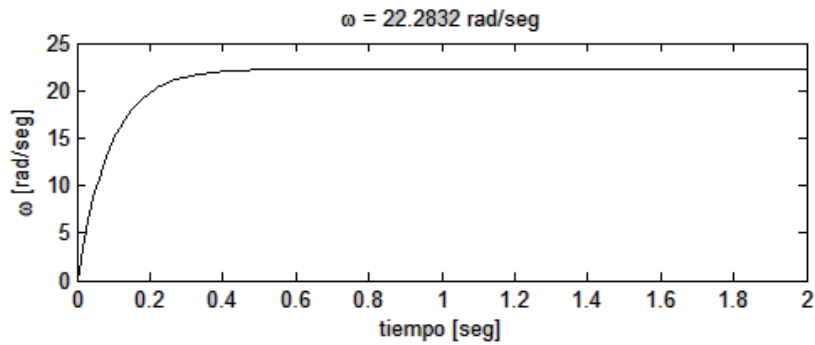


Figura N°03. Simulaciones para la curva de velocidad en MATLAB.

Fuente: Elaboración propia.

### 2.3.3 Adquisición de datos

El sistema de adquisición de datos, cuenta con el módulo de control (ver Figura N°04) y la interfase NIDAQ USB-6009 (ver Figura N°05), de modo que estos dos módulos interactúan. El Motor DC EMG30 de 12V está equipado con codificador y una caja de engranes de 30:1, es ideal para aplicaciones de robótica, además posee una velocidad de 216RPM sin carga se caracteriza por incluir un encoder o codificador de cuadrante de efecto hall, que envía un tren de pulsos cuando gira el eje del motor permitiendo que el circuito externo conozca la velocidad real a la que gira el motor y cuantas vueltas dará. También está la tarjeta electrónica que contiene (modulador de ancho de pulso, convertidor frecuencia a voltaje y un amplificador de potencia o driver). La interfase interactúa con la PC (Personal Computer) a través del puerto USB.

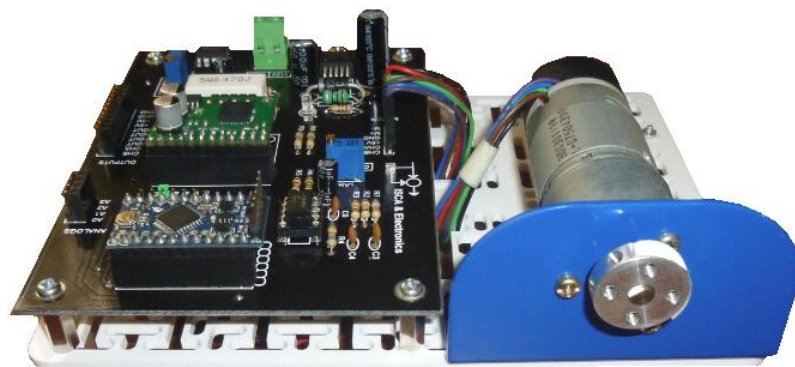


Figura N°04. Módulo físico del motor EMG30 y sus periféricos.

Fuente: Elaboración propia.

El módulo interfase compuesto por la tarjeta de adquisición de datos DAQ USB-6009 de National Instruments (ver Figura N°05). Posee 8 entradas análogas y 2 salidas análogas, dos puertos DIOs (Digital Input-Output), uno puerto P0 de 8 bits y otro puerto P1 de 4 bits, para la comunicación con el módulo del motor DC.



Figura N°05. Módulo físico de la tarjeta NIDAQ USB-6009.

Fuente: Elaboración propia.

La interfase NIDAQ posee un firmware denominado MAX (Measurement & Automation o simplemente MAX) según la Figura N°06.



Figura N°06. Firmware MAX de LabVIEW.

Fuente: Elaboración propia.

Desde el menú del MAX, Devices & Interfaces, procedemos a realizar el test de prueba con la DAQ USB-6009 que es asignada con “Dev2” (ver Figura N°07).

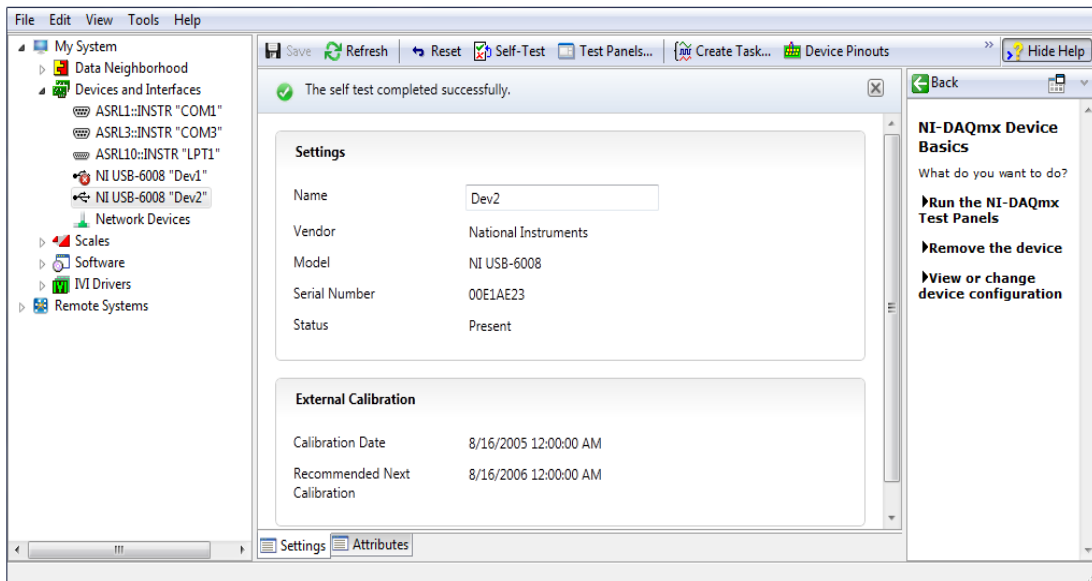


Figura N°07. Completando el test del dispositivo.

Fuente: Elaboración propia.

Desde el menú Devices & Interfaces, procedemos a abrir la ventana de configuración de analog input (ai1) como se muestra en la Figura N°08.

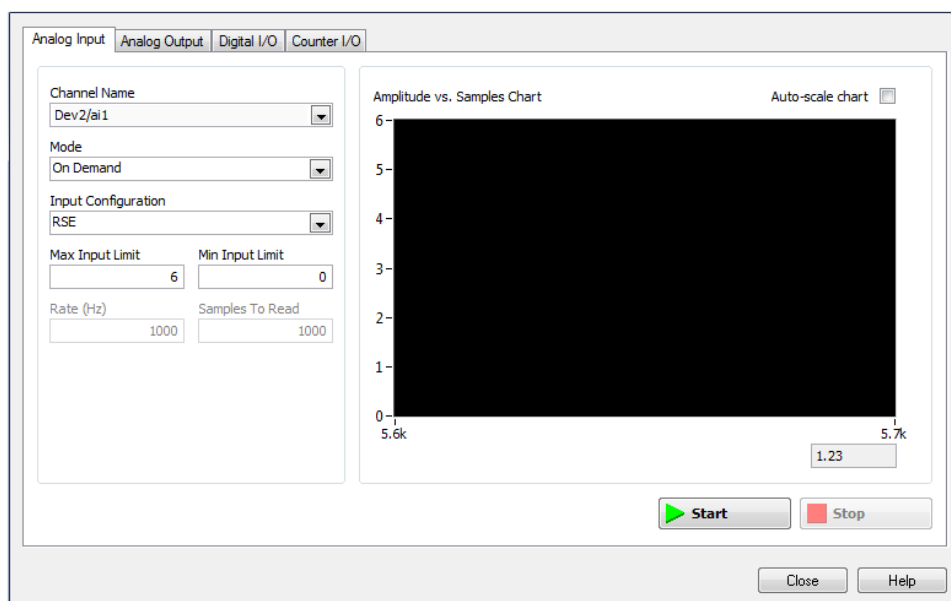


Figura N°08. Configurando canal de entrada analógica.

Fuente: Elaboración propia.

En forma análoga configuraremos analog output (ao1) según la Figura N°09.

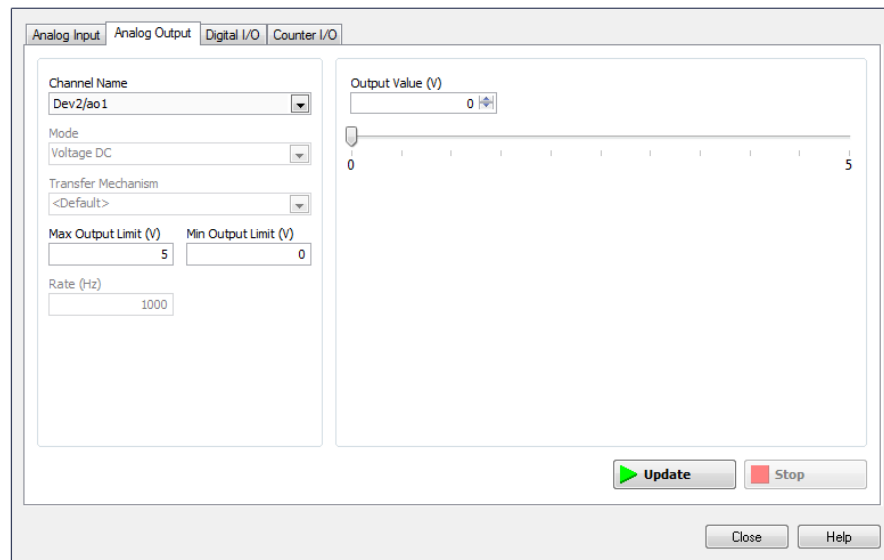


Figura N°09. Configurando canal de salida analógica.

Fuente: Elaboración propia.

Ahora procedemos a ingresar valores desde el canal analog output (ao1), por ejemplo 0V y con “update” actualizamos el valor del voltaje, al ejecutar la entrada analógica “START”, observaremos en forma simultánea este valor desde el canal de entrada analógica (ai1). Repetimos esta operación para valores de 1V, 2V, 3V, 4V y 5V y observaremos la curva característica de voltaje del sistema de control a lazo abierto (ver Figura N°10).

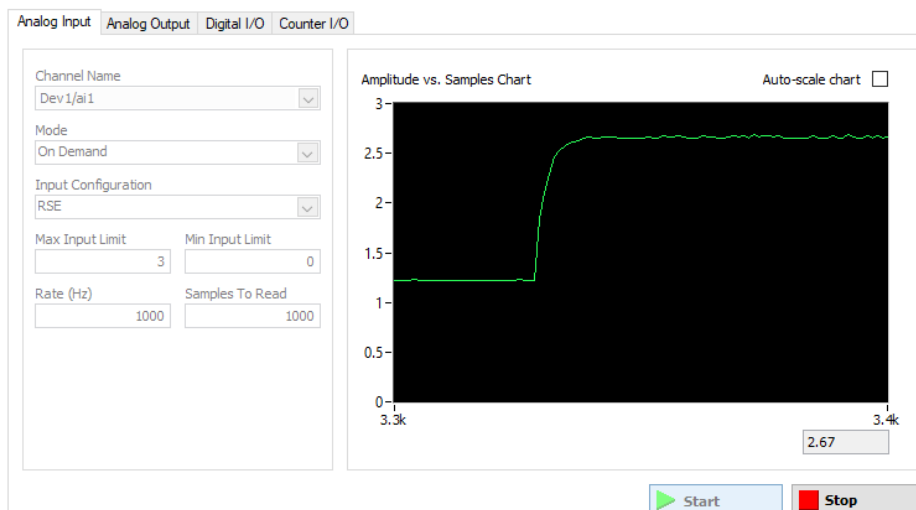


Figura N°10. Curva de la velocidad a lazo abierto desde el MAX.

Fuente: Elaboración propia.



Ahora realizaremos la toma de datos y escribimos por el puerto ai1 los valores medidos archivo bloc de notas y completamos los datos según la Tabla N°3. Es importante mencionar que hay un voltaje de offset de 1.22V que en adelante será sujeto al análisis de la regresión lineal a manera de escalamiento.

Tabla N°03. Datos medidos desde el MAX para la entrada y salida análoga.

ao1	ai1
0	1.22
1	1.43
2	1.74
3	2.05
4	2.34
5	2.67

#### 2.3.4 Estimación de modelos con IDGREY

Los modelos de sistemas lineales describen bastante bien a los sistemas reales, incluso cuando muchos sistemas son no lineales siempre y cuando haya un punto de operación. Algunos sistemas no lineales se aproximan mediante modelos lineales cuando las entradas y salidas están dentro de ciertos rangos o puntos de operación específicos. Sin embargo, a veces los modelos lineales no capturan el comportamiento del sistema, especialmente al operar en una amplia gama de puntos de operación donde la ocurrencia de los fenómenos no lineales se presenta, tales como, el caos, ciclos limites, entre otros.

Si entendemos la física de un sistema, lo expresamos como una serie de ecuaciones diferenciales ordinarias, y a través de estas ecuaciones, representamos el modelo espacio estado con algunos parámetros desconocidos y otros parámetros conocidos analizados en el dominio de tiempo continuo (o ecuaciones de diferencia de tiempo discreto). Con esta representación es posible utilizar una identificación de sistemas lineales o no lineales de caja gris. Como se mencionó anteriormente, el modelado IDGREY es útil cuando se conoce la relación entre variables, restricciones o ecuaciones explícitas que representa la dinámica del sistema. En nuestro caso, se estimó el modelo de caja gris en tiempo continuo-

discreto, para datos que han sido adquiridos experimentalmente y utilizando las herramientas del Toolbox IDGREY de MATLAB.

Un primer paso para modelar sistemas de caja gris, consiste en escribir una función que retorne matrices espacio estado es usando parámetros definidos e información acerca del modelo.

El siguiente formato para implementar modelos lineales grey-box se da mediante la función:  $[A, B, C, D] = \text{mifuncion}(\text{par1}, \text{par2}, \dots, \text{parN}, T_s, \text{aux1}, \text{aux2}, \dots)$ .

Donde los argumentos de salida son las matrices espacio estado, además.

- mifuncion es el nombre del archivo
- par1, par2, ..., parN son los  $N$  parámetros del modelo. Cada entrada es un escalar, vector o matriz.
- $T_s$  es el periodo de muestreo.
- aux1, aux2, ... son los argumentos de entrada opcional para que mifuncion use en cómputo de las matrices espacio estado adicionalmente a los parámetros y periodo de muestreo.

La variable aux contiene variables auxiliares de nuestro sistema. Usamos variables auxiliares para variar los parámetros del sistema en la entrada de la función y evitar la edición del archivo.

Lo interesante de IDGREY es la posibilidad de permitirnos trabajar con modelos espacio estado en dominio de tiempo continuo o dominio de tiempo discreto, u ambos dentro del contexto de programación. Cuando aplicamos el comando idgrey de MATLAB se construye un modelo lineal de caja gris con parámetros que serán identificables. Utilizando el modelo mifuncion especificamos la naturaleza de los argumentos de salida de mifuncion.

- Un modelo espacio estado en tiempo continuo tiene la forma:

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) + Ke(t) \\ y(t) &= Cx(t) + Du(t) + e(t) \\ x(0) &= x_0\end{aligned}\tag{14}$$

Donde  $A, B, C, D$  son las matrices de estado que son parametrizadas por los parámetros según  $par1, par2, \dots, parN$ . La matriz de ruido  $K$  y el vector inicial  $x_0$  no son parametrizados por la función `mifuncion`. Para configurar el manejo de los estados iniciales,  $x_0$  y la perturbación,  $K$  durante la estimación usamos el conjunto de opciones que nos proporciona `greyestOptions`.

- Un modelo espacio estado en tiempo discreto tiene la forma.

$$\begin{aligned}x(k + 1) &= Ax(k) + Bu(k) + Ke(k) \\y(k) &= Cx(k) + Du(k) + e(k) \\x(0) &= x_0\end{aligned}\tag{15}$$

Donde  $A, B, C, D$  ahora son las matrices de estado en el dominio del tiempo discreto que son parametrizadas por los parámetros según  $par1, par2, \dots, parN$ . La matriz de ruido  $K$  y el vector inicial  $x_0$  no son directamente parametrizados siendo estimadas si es requerido por la configuración de las opciones de estimación correspondiente.

### 2.3.5 Parametrizando modelo IDGREY

Se requieren desarrollar los siguientes pasos para lograr obtener el procesamiento de la DATA obtenida experimentalmente, y por ende llegar a encontrar el modelo estimado.

#### (a) Parametrizando modelos con disturbio y estados iniciales

En algunas aplicaciones queremos expresar la matriz de ruido  $K$  y el vector inicial  $x_0$  como parametrizados escogiendo parámetros tales como las matrices  $A, B, C, D$ . Para manejar estos casos, escribimos un archivo ODE en `mifuncion` de modo que retorne  $K$  y  $x_0$  según.

$$[A, B, C, D, K, x_0] = \text{mifuncion}(par1, par2, \dots, parN, Ts, aux1, aux2, \dots)$$

En este caso, la matriz de ruido  $K$  y el vector inicial,  $x_0$  por lo tanto son tratados de la misma manera que las matrices de estado con las funciones de los parámetros  $par1, par2, \dots, parN$ .

(b) Construyendo un IDGREY Objeto

Antes de crear una función con la estructura del modelo, debemos definir el modelo idgrey – object. Se crea una función del modelo grey-box. Para ello es preciso representar la ecuación de estado de un motor DC controlado por armadura, según el esquema de la Figura N°11.

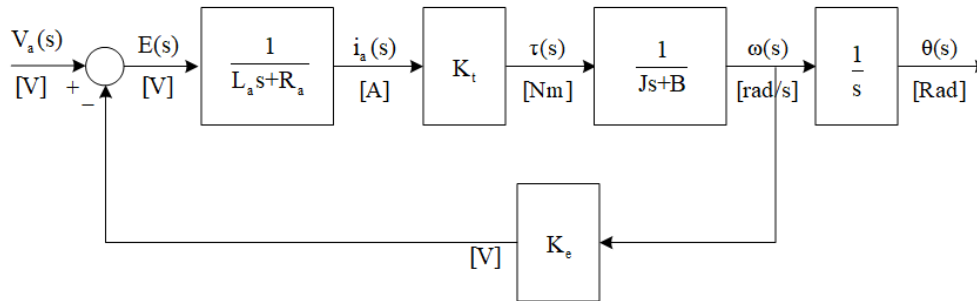


Figura N°11. Diagrama de bloques de un motor DC controlado por armadura.

Fuente: Elaboración propia.

Los parámetros del motor DC son: Inercia (J), la fricción viscosa (B), la inductancia ( $L_a$ ), la resistencia ( $R_a$ ), la constante contra electromotriz ( $K_b$ ) y la constante de motor ( $K_t$ ). En el diagrama de la Figura N°11, despreciamos la inductancia de armadura,  $L_a \approx 0$ , y considerando como salida la velocidad angular, encontraremos la función de transferencia de primer orden definida según.

$$\frac{\omega(s)}{V_a(s)} = \frac{K}{\tau s + 1} \quad (16)$$

Dónde:

$$K = \frac{K_t}{R_a B + K_b K_t} \quad \tau = \frac{J R_a}{R_a B + K_b K_t}$$

El modelo a estimar será considerado como un sistema de primer orden. Asignando el estado  $x$  a la velocidad angular,  $x = \omega$  mientras que la entrada es  $u = V_a$ . Según esto se reescribe las ecuaciones de estado y salida.

$$\dot{x}(t) = -\frac{1}{\tau}x(t) + \frac{K}{\tau}u(t)$$

$$y(t) = x(t)$$

$$x(0) = 0$$

(c) Estructuras de modelo IDGREY

El modelo espacio estado anterior es representado como una estructura de un modelo de espacio estado de tiempo continuo, según.

$$\dot{x}(t) = \theta_1 x(t) + \theta_2 u(t)$$

$$y(t) = x(t)$$

$$x(0) = 0$$

Esta ecuación espacio estado, representa al sistema de velocidad de un motor DC, en donde los parámetros son considerados según:

- Los parámetros  $\theta_1$  y  $\theta_2$  como “conocidos” bajo ciertas condiciones iniciales o valores iniciales, mientras que la siguiente estructura:

Otro posible modelo espacio estado es representado, según.

$$\dot{x}(t) = -\theta_1 x(t) + K\theta_1 u(t)$$

$$y(t) = x(t)$$

$$x(0) = 0$$

Esta nueva ecuación espacio estado, también representa al sistema de velocidad de un motor DC, en donde los parámetros son considerados según.

- Un parámetro constante de tiempo desconocido,  $\theta_1$  y otro parámetro  $K$  conocido. El parámetro  $-\theta_1 = 1/\tau$ .

Esta última estructura es muy usual en estimación de parámetros de caja gris.

(d) Función en tiempo discreto

Para realizar la función motorODE.m se requiere conocer una expresión para programar el modelo en tiempo discreto para el modelo. Se requiere el siguiente análisis.

Sea la matriz exponencial.

$$e^A = \sum_{k=0}^{\infty} \frac{1}{k!} A^k = 1 + A + \frac{A^2}{2} + \dots + \quad (17)$$

Encontrar la matriz de transición.

$$\begin{aligned}x(k + 1) &= \Phi x(k) + \Gamma u(k) \\y(k) &= C_d x(k) + D_d u(k)\end{aligned}\tag{18}$$

La matriz  $\Phi$  del sistema discreto es representada según la serie.

$$\Phi = e^{AT} = \sum_{k=0}^{\infty} \frac{1}{k!} (AT)^k = 1 + AT + \frac{(AT)^2}{2} + \dots +$$

Desde la perspectiva de la función de transferencia pulso obtenemos.

$$\Gamma = A^{-1}(\Phi - I)B = (\Phi - I)A^{-1}B$$

$$\begin{bmatrix} \Phi & \Gamma \\ 0 & I \end{bmatrix} = \exp\left(\begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix}\right) h\tag{19}$$

La ecuación (19) es la expresión matricial para el programa con IDGREY.

### 2.3.6 Modelo de una neurona biológica

Durante muchas décadas, el objetivo de la ciencia y la ingeniería ha sido desarrollar máquinas inteligentes con una gran cantidad de elementos simples, estas se encuentran en referencias en la literatura científica del siglo XIX. Durante la década de 1940, los investigadores cuyo estudio fue el modelo del cerebro humano, desarrollaron modelos simples de hardware, posteriormente en software para simulaciones de neuronas biológicas y sus sistemas de interacción ya aun no estaban en desarrollo el advenimiento tecnológico de las computadoras personales. McCulloch y Pitts publicaron el primer estudio sistemático de la red neuronal artificial. Cuatro años más tarde, los mismos autores exploraron paradigmas de red para el reconocimiento de patrones utilizando un perceptron de capa única, en donde se destacaba el reconocimiento de patrones mediante los datos proporcionados por las funciones lógicas AND y OR. En las décadas de 1950 y 1960, un grupo de investigadores combinó estos conocimientos biológicos y psicológicos para producir la primera red neuronal artificial (ANN, Artificial Neural Network). Inicialmente implementados como circuitos electrónicos, luego se convirtieron en un medio más flexible de simulación por computadora. Sin embargo, los investigadores Minsky y Papert, cuestionaron estos trabajos realizados por

McCulloch y Pitts, debido que la red perceptron propuesta no era capaz de clasificar a la función lógica XOR, en un procesamiento de símbolos fácilmente modelado en la computadora de tipo Von Neumann.

El perceptron propuesto por Rosenblat resultó ser más limitado de lo esperado. Aunque otras investigaciones en ANN continuaron durante la década de 1970 por varios investigadores pioneros como Grossberg, Kohonen, Widrow y otros investigadores, en sus inicios, sus trabajos recibieron relativamente menor atención. Los factores principales para el reciente resurgimiento del interés en el área de las redes neuronales son la extensión de los trabajos de Rosenblat, Widrow y Hoff que se ocuparon del aprendizaje en una red compleja de múltiples capas, base matemática de Hopfield para comprender la dinámica de una clase importante de redes, así como computadoras mucho más rápidas que las de los años 60.

El interés en las redes neuronales proviene de la capacidad de las redes para emular al cerebro humano, así como de su capacidad para aprender y responder. Como resultado, las redes neuronales se han utilizado en una gran cantidad de aplicaciones y han demostrado ser eficaces para realizar funciones complejas en una variedad de campos. Estos incluyen reconocimiento de patrones, clasificación, visión, sistemas de control y predicción.

Desde el punto de vista fisiológico, el sistema nervioso humano es una red neuronal muy compleja. El cerebro es el elemento central del sistema nervioso humano, que consta de cerca de  $10^{10}$  neuronas biológicas que están conectadas entre sí a través de subredes. Cada neurona en el cerebro está compuesta por un cuerpo, un axón y multitud de dendritas (ver Figura N°12).

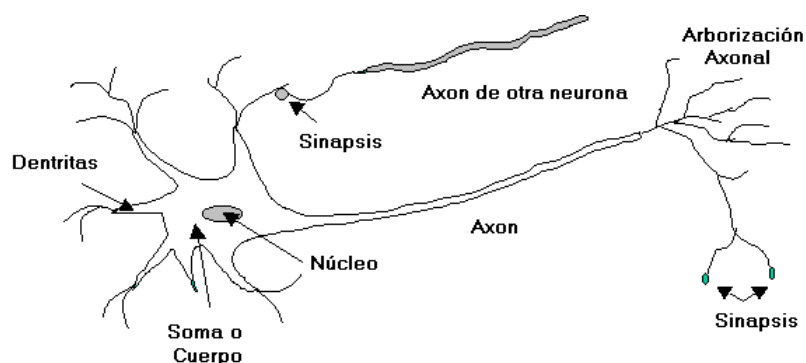


Figura N°12. Neurona biológica.

Fuente: Elaboración propia.

El modelo de una neurona sirvió como base para implementar el modelo de la neurona artificial.

- Las dendritas reciben señales de otras neuronas.
- El axón se considera como un tubo largo, que se divide en ramas que terminan en pequeños bulbos terminales.
- La pequeña brecha entre una bombilla terminal y una dendrita se llama sinapsis.
- El axón de una sola neurona forma conexiones sinápticas con muchas otras neuronas.

Dependiendo del tipo de neurona, el número de conexiones de sinapsis de otras neuronas varían de unos pocos cientos a  $10^4$ .

Un modelo de la neurona artificial, como se muestra en la Figura N°13, es el elemento básico de una red neuronal. Se compone de tres componentes básicos que incluyen, los pesos sinápticos, el umbral y la función de activación.

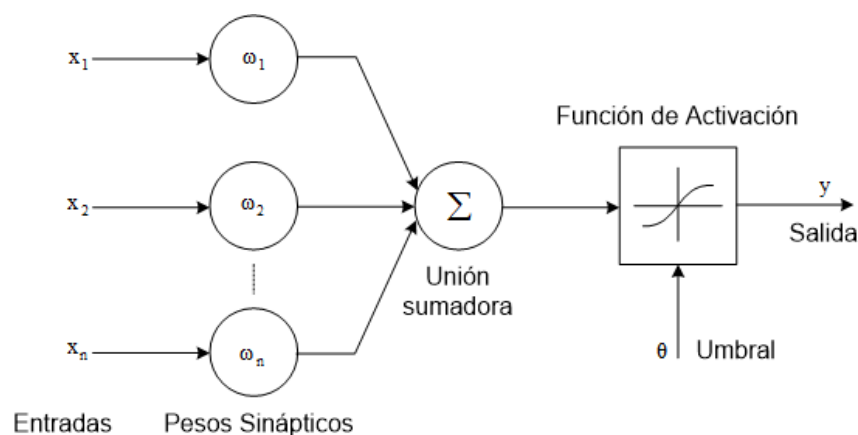


Figura N°13. Elementos básicos de la red neuronal.

Fuente: Elaboración propia.

Una red neuronal presenta los siguientes elementos característicos.

### 1. Factores de ponderación

Los valores  $\omega_1, \omega_2, \omega_3, \dots, \omega_n$  son factores de peso asociados con cada nodo para determinar la fuerza del vector de fila de entrada  $x = [x_1, x_2, x_3, \dots, x_n]^T$ . Cada entrada se multiplica por el peso asociado de la conexión neuronal  $x^T \omega$ .



Dependiendo de la función de activación, si el peso es positivo,  $x^T \omega$  comúnmente excita la salida del nodo; mientras que, para pesos negativos,  $x^T \omega$  tiende a inhibir la salida del nodo.

## 2. Umbral

El umbral interno del nodo  $\theta$  es el desplazamiento de magnitud que afecta la activación de la salida del nodo y de la siguiente manera.

$$y = \sum_{i=1}^n (x_i \omega_i) - \theta_k \quad (20)$$

## 3. Función de Activación

Se presentan diversidad de funciones de activación, tanto lineal como no lineal. Una función de activación realiza una operación matemática en la salida de señal. Las funciones de activación más sofisticadas también se utilizan según el tipo de problema a resolver por la red. Entre las más destacadas podemos mencionar.

### 3.1 Función lineal

Como se sabe, una función lineal satisface el concepto de superposición, esta función se muestra en la Figura N°14.

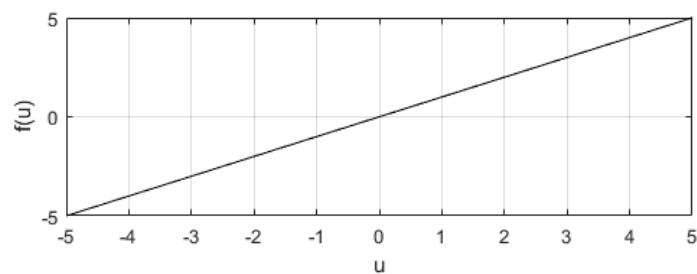


Figura N°14. Función de activación lineal.

Fuente: Elaboración propia.

La ecuación matemática se escribir como.

$$y = f(u) = au \quad (21)$$

Donde  $\alpha$  es la pendiente de la función lineal de (21). Si la pendiente  $\alpha$  es 1, la función de activación lineal se llama función de identidad. La salida ( $y$ ) de la función de identidad es igual a la función de entrada ( $u$ ).

### 3.2 Función Sigmoide (Forma de S)

Esta función no lineal es el tipo más común de activación utilizada para construir las redes neuronales. Tiene una función matemáticamente continua, diferenciable y estrictamente creciente.

Una función de transferencia sigmoide se escribe.

$$y = f(u) = \frac{1}{1 + e^{-\alpha u}}, \quad 0 \leq f(u) \leq 1 \quad (22)$$

Donde  $\alpha$  es el parámetro de forma de la función sigmoidea. Al variar este parámetro, obtiene diferentes formas de la función como se ilustra en la Figura N°15. Esta función es continua monótonica y diferenciable.

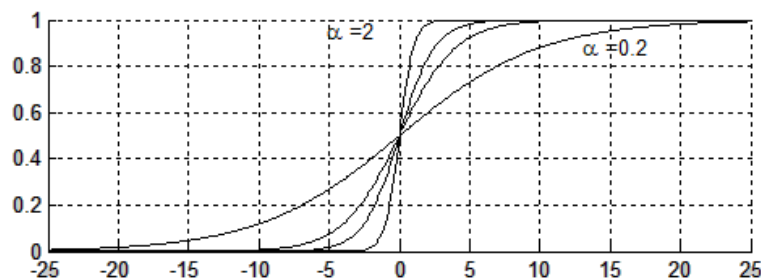


Figura N°15. Función de activación sigmoide.

Fuente: Elaboración propia.

### 3.3 Función Tangente Hiperbólica

Esta función de transferencia se describe mediante la siguiente forma matemática.

$$y = f(u) = \frac{e^{\alpha u} - e^{-\alpha u}}{e^{\alpha u} + e^{-\alpha u}}, \quad -1 \leq f(u) \leq 1 \quad (23)$$

Es interesante observar que las derivadas de las ecuaciones (22) y (23) se expresan en términos de la función individual en sí misma.

La Figura N°16 se muestra de activación tangente hiperbólica.

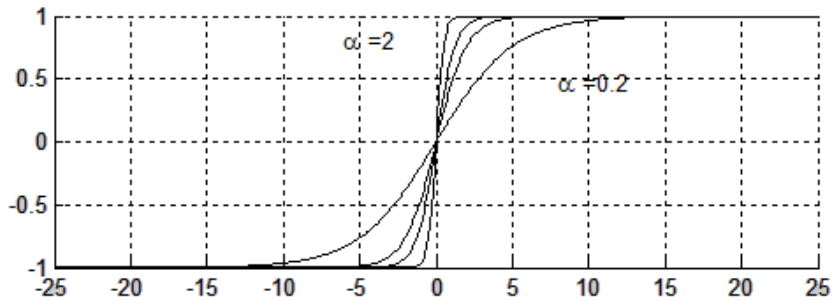


Figura N°16. Función de activación tangente hiperbólica.

Fuente: Elaboración propia.

#### 4. Arquitectura General

El perceptron multicapa representa una generalización del perceptron monocapa como se describió en la sección anterior. Un perceptron de una sola capa forma una región de decisión de medio plano. Por otro lado, el perceptron multicapa forma regiones de decisión arbitrariamente complejas y está separada varios patrones de entrada.

La capacidad del perceptron multicapa proviene de las no linealidades utilizadas dentro de los nodos. Si los nodos fueran elementos lineales, se podría utilizar una red de una sola capa con el peso adecuado en lugar de perceptron de dos o tres capas. En la Figura N°17 se muestra una estructura de red neuronal perceptron multicapa típica de estructura 2-2-1.

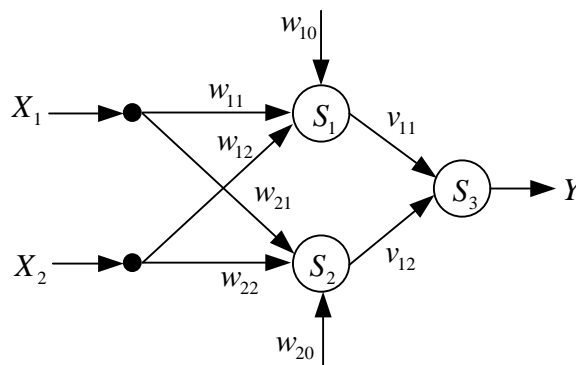


Figura N°17. Red neuronal multicapa 2-2-1.

Fuente: Elaboración propia.

La red consta de las siguientes capas.

- Capa o nodo de entrada: una capa de neuronas que recibe información de fuentes externas y pasa esta información a la red para su procesamiento. Estos son las entradas sensoriales ( $X_1, X_2$ ) o señales de otros sistemas fuera del que se está modelando.
- Capa oculta: una capa de neuronas que recibe información de la capa de entrada y las procesa de forma oculta compuesta por las neuronas ( $S_1, S_2$ ). No tiene conexiones directas con el mundo exterior (entradas o salidas). Todas las conexiones de la capa oculta son a otras capas dentro del sistema.
- Capa de salida: es una capa (más usual) que tiene una o más neuronas ( $S_3$ ), presenta una interconexión directa similar al axón ya que comunica la información al exterior, de modo que interactúe con otras neuronas.
- Los pesos sinápticos: los pesos sinápticos son la fortaleza de una red neuronal presenta los pesos en la capa oculta ( $w_{ij}$ ) y capa de salida ( $v_{kj}$ ) definidos por las matrices.

$$w = \begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix}, v = (v_{11} \quad v_{12})$$

## 5. Algoritmo en una red neuronal 2-2-1

### 5.1 Inicialización

- Inicializar los pesos  $w$  y pesos  $v$  con valores aleatorios pequeños.
- Inicializar la razón de aprendizaje  $\eta$  con valores pequeños menores a 1.
- Escoger la función de activación por capas.

### 5.2 Selección de patrones

- Los patrones son los datos E/S que se le proporciona a la red.

### 5.3 Feedforward

- Se representa por las siguientes ecuaciones.
- Activación en la neurona 1.

$$net_1 = w_{10}X_0 + w_{11}X_1 + w_{12}X_2 \quad (24)$$

- Activación en la neurona 2.

$$net_2 = w_{20}X_0 + w_{21}X_1 + w_{22}X_2 \quad (25)$$

- Salida de la neurona oculta 1.

$$Y_1 = \tanh^{-1}(net_1) \quad (26)$$

- Salida de la neurona oculta 2.

$$Y_2 = \tanh^{-1}(net_2) \quad (27)$$

- Activación en la neurona de salida.

$$net_3 = v_{10}Y_0 + v_{11}Y_1 + v_{12}Y_2 \quad (28)$$

- Salida de la red.

$$Y = \tanh^{-1}(net_3) \quad (29)$$

#### 5.4 Error feedback

- Calcula salida de error.

$$\delta_3 = (D - Y_3)(1 - Y_3^2) \quad (30)$$

- Calcula error feedback para las neuronas de la capa oculta.

$$\delta_2 = \delta_3 v_{11}(1 - Y_1^2) \quad (31)$$

$$\delta_1 = \delta_3 v_{12}(1 - Y_2^2) \quad (31)$$

#### 5.5 Cambio de pesos

$$\begin{aligned} \Delta v_{11} &= \eta \delta_3 Y_1, & \Delta v_{12} &= \eta \delta_3 Y_2, & \Delta v_{10} &= \eta \delta_3 \\ \Delta w_{11} &= \eta \delta_1 X_1, & \Delta w_{12} &= \eta \delta_1 X_2, & \Delta w_{10} &= \eta \delta_1 \\ \Delta w_{21} &= \eta \delta_2 X_1, & \Delta w_{22} &= \eta \delta_2 X_2, & \Delta v_{10} &= \eta \delta_2 \end{aligned} \quad (33)$$

### 2.3.7 Teoría de la interacción adaptativa

Desde su renacimiento en 1980, las redes neuronales han encontrado aplicaciones en muchos campos de la ingeniería, entre ellos, la identificación de los sistemas lineales y no lineales incluyendo al control adaptativo neuronal. Los controladores neuronales controlan no sólo los sistemas lineales, sino también sistemas no lineales que es en donde alcanzan un buen desempeño. (Adaptive interaction and its application to neural networks an Intelligent Devices, 2001).

Los diseños de control neuronal se dividen en dos categorías principales:

- (1) El diseño directo cuando el controlador es una red neuronal (implementación no muy conveniente).
- (2) El diseño indirecto cuando el controlador no es en sí una red neuronal, sino que utiliza principios de las redes neuronales en su diseño y adaptación (implementación conveniente).

Interrogantes como la robustez y la estabilidad también sobre el control de la red neuronal se han discutido en muchos libros. Sin embargo, hay dos factores principales que contribuyen a la popularidad de las redes neuronales.

- La capacidad de las redes neuronales para aproximar las funciones no lineales. Esto es importante porque en muchos casos los objetivos de control logran más eficazmente mediante el uso de un controlador no lineal.
- La capacidad de las redes neuronales a “adaptarse”. De hecho, el camino para redes neuronales es una adaptación muy natural. No requiere la construcción de modelos o parámetros de identificación. La adaptabilidad "natural" de sistemas es algo único, esto hace del diseño del control un gran trabajo fácil. Por ejemplo, sabemos lo difícil que es diseñar un sistema de control no lineal. Sin embargo, es más flexible, si dejamos que un controlador basado en una red neuronal se adapte.

### 2.3.8 El algoritmo de Brandt-Lin

Este algoritmo propuesto por Brandt-Lin, hace que los errores necesarios para la adaptación se infieran la información local de tal manera que el error de propagación inversa se hace implícitamente más que explícito. Como resultado, el algoritmo de Brandt-Lin es implementado en una manera simple y directa sin utilizar el algoritmo de propagación inversa en una red neuronal. Matemáticamente, sin embargo, se demuestra que el algoritmo de Brandt-Lin es equivalente para el algoritmo de propagación inversa.

Además, el algoritmo de Brandt-Lin se aplica a sistemas arbitrarios (sistemas de fase no mínima), incluyendo sistemas híbridos como el que nos ocupa en los sistemas de control de redes neuronales. Es porque el algoritmo de Brandt-Lin se deriva de una teoría de la interacción de adaptación que es aplicable a una amplia clase de sistemas. Se ha aplicado a controladores PID autoajustables y la estimación de parámetros.

El algoritmo desarrolla la teoría de interacción adaptativa que se aplica para adaptar los coeficientes del sistema. Dependiendo de su aplicación y configuración del algoritmo, los coeficientes ajustados son los pesos de la red neuronal, ganancias del controlador PID o coeficientes de las funciones de transferencia.

Para aplicar el algoritmo a un sistema de control, la única información necesaria sobre la planta es la derivada de Fréchet, que se aproxima por una constante determinada. Esto hará que el algoritmo sea robusto para incertidumbres del sistema y eventuales cambios, por tanto, es aplicable a una amplia clase de sistemas.

La teoría de la adaptación interactiva considera  $N$  subsistemas llamados dispositivos, estos dispositivos son generalmente parámetros que constituyen una ley de control o interactúan con otros elementos del sistema. Las interacciones entre los dispositivos se consiguen mediante las conexiones tal como se observa en la Figura N°19. El conjunto de todas las conexiones se denota por  $C$ , y las conexiones pre y post sináptica son referidas a la transmisión que existe entre los dispositivos.

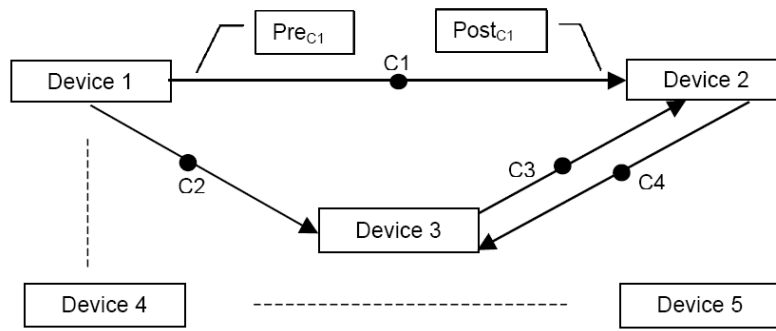


Figura N°19. Los dispositivos y sus interconexiones.

Fuente: Elaboración propia.

Para representar las relaciones entre dispositivos y conexiones empleamos.

- $pre_c$  es el dispositivo cuya salida se transmite por la conexión  $c$ .
- $post_c$  es el dispositivo cuya entrada depende de la señal transmitida por  $c$ .
- $I_n = \{c: pre_c = n\}$  es el conjunto de las interacciones de entrada para el dispositivo  $n$ -ésimo potencia.
- $O_n = \{c: post_c = n\}$  es el conjunto de las interacciones de salida para el dispositivo de  $n$ -ésima potencia.

Cada dispositivo (indexados por  $n \in N := \{1, 2, \dots, N\}$ ) tienen una salida  $y_n$  de la señal integrable y una señal de entrada  $x_n$  integrable. La dinámica de cada dispositivo se describe como una función causal.

$$F_n: X_n \rightarrow Y_n, \quad n \in \mathbb{N}$$

Donde  $X_n$  y  $Y_n$  son los espacios de entrada y de salida respectivamente. Así mismo la relación entre la entrada y la salida del  $n$ -ésimo dispositivo es.

$$y_n(t) = (F_n \circ X_n)(t) = F_n[X_n(t)], \quad n \in \mathbb{N}$$

Donde  $(\circ)$  denota la composición de funciones.

Asumimos la interacción lineal entre dispositivos y de señales externas  $u_n(t)$ , es.

$$x_n(t) = u_n(t) + \sum_{c \in I_n} \alpha_c y_{pre_c}(t), \quad n \in \mathbb{N}$$

Donde  $\alpha_c$  es la conexión de los pesos



Con esta interacción lineal, la dinámica del sistema es descrita por:

$$y_n = F_n \left[ u_n(t) + \sum_{c \in I_n} \alpha_c y_{pre_c}(t) \right], \quad n \in \mathbb{N} \quad (34)$$

El objetivo del algoritmo de adaptación es adaptar la conexión de pesos  $\alpha_c$  por lo que el índice de performance de  $E(y_1, \dots, y_n, u_1, \dots, u_n)$  es una función de las entradas y salidas que deben ser minimizadas.

Para presentar el algoritmo, primero tenemos que introducir la derivada de Fréchet que se describe, como una transformación  $T$  definida en un dominio abierto  $D$  en un espacio normado  $X$  que debe tener un rango en un espacio normado  $Y$ . Si para un  $x \in D$  fijo y para cada elemento  $h \in X$  existe  $\delta T(x; h) \in Y$  que es lineal y continuo con respecto a  $h$  tal que.

$$\lim_{\|h\| \rightarrow 0} \frac{\|T(x+h) - T(x) - \delta T(x; h)\|}{\|h\|} = 0 \quad (35)$$

Entonces  $T$  se dice que es diferenciable de Fréchet en  $x$  y  $\delta T(x; h)$  de  $T$  en  $x$  con  $h$  como incremento. En nuestro caso,  $T(x) = F_n(x)$  y  $\delta T(x; h) = F'_n(x) \circ h$  donde  $F'_n(x)$  es la derivada de Fréchet. El algoritmo de adaptación se da en (34). Asumiremos que la conexión de los pesos  $\alpha_c$  son adaptables de acuerdo a la expresión.

$$\dot{\alpha}_c = \left( \sum_{s \in O_{post_c}} \alpha_s \dot{\alpha}_s \frac{\frac{dE}{dy_{post_s}} F'_{post_s}[x_{post_s}]}{\frac{dE}{dy_{post_s}} F'_{post_s}[x_{post_s}] \circ y_{pre_c}} - \gamma \frac{\partial E}{\partial y_{post_c}} \right) \circ F'_{post_c}[x_{post_c}] \circ y_{pre_c} \quad c \in C \quad (36)$$

Donde  $\gamma > 0$  es el coeficiente de adaptación. Si la ecuación (36) tiene una solución única para un  $\dot{\alpha}_c, c \in C$  (el determinante Jacobiano no debe ser cero en la región de interés), entonces el índice de performance  $E(y_1, \dots, y_n, u_1, \dots, u_n)$  que decrece monótonicamente en el tiempo y la siguiente ecuación se satisface siempre.

$$\dot{\alpha}_c = -\gamma \frac{\partial E}{\partial \alpha_c}, \quad c \in C \quad (37)$$

Es importante observar que si  $F_n$  y  $E$  son funciones instantáneas, entonces la composición funcional ( $\circ$ ) es sustituida por la multiplicación. La ecuación (36) entonces se simplifica a.

$$\dot{\alpha}_c = F'_{post_c}[x_{post_c}] \left( \frac{y_{pre_c}}{y_{post_c}} \right) \sum_{s \in O_{post_c}} \alpha_s \dot{\alpha}_s - \gamma F'_{post_c}[x_{post_c}] y_{pre_c} \frac{\partial E}{\partial y_{post_c}} \quad (38)$$

Las ecuaciones anteriores se aplican a una clase muy general de sistemas incluyendo las redes neuronales, como se muestra a continuación. Una red neuronal se descompone en varios dispositivos como se describe en la Figura N°20 que muestra una representación gráfica de una red neuronal sencilla.

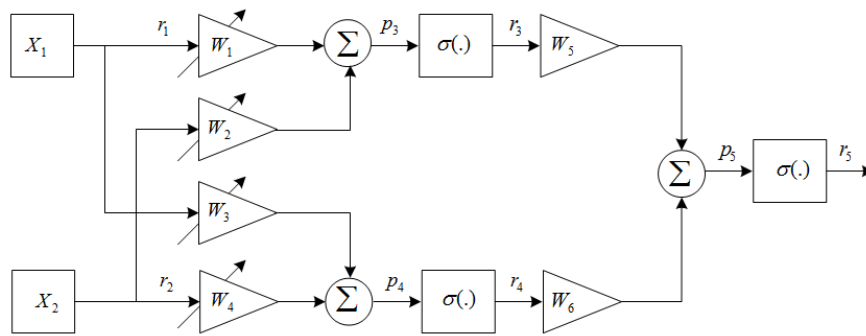


Figura N°20. Red neuronal multicapa.

Fuente: Elaboración propia.

- $n$  es la etiqueta particular de la neurona.
- $s$  es la etiqueta particular de una sinapsis.
- $D_n$  es el conjunto de dendritas (entrada) las sinapsis de la neurona,  $n$ .
- $A_n$  es el conjunto axón (salida) las sinapsis de la neurona,  $n$ .
- $pre_s$  la neurona pre sináptica correspondiente a la sinapsis,  $s$ .
- $post_s$  la neurona post sináptica correspondiente a la sinapsis,  $s$ .
- $w_s$  es la fuerza (peso) de la sinapsis,  $s$ .
- $p_n$  es el potencial de membrana de la neurona,  $n$ .
- $r_n$  es la tasa de disparo de la neurona,  $n$ .
- $\gamma$  es el coeficiente de captación directa de todas las neuronas.
- $f_n$  es la señal de retroalimentación directa.
- $\sigma$  es la función sigmoidea  $\sigma(x) = \frac{1}{1+e^{-x}}$

Matemáticamente, la red neuronal y el algoritmo de adaptación se describen como sigue.

$$p_n = \sum_{s \in D_n} w_s r_{pre_s} \quad (39)$$

La salida de la red.

$$r_n = \sigma(p_n) \quad (40)$$

Además, denotamos.

$$\phi_n = \frac{1}{2} \frac{d}{dt} \sum_{s \in A_n} w_s^2 = \sum_{s \in A_n} w_s \dot{w}_s \quad (41)$$

Entonces aplicamos la ley de adaptación dada en (38), los pesos de adaptación resultan.

$$\dot{w}_s = r_{pre_s} (\phi_{post_s} \sigma(-p_{post_s}) + \gamma f_{post_s}) \quad (42)$$

Las ecuaciones (41) y (42) Describen el algoritmo de Brandt-Lin para la adaptación de la red neuronal. Es equivalente al algoritmo de propagación inversa desarrollado para la función XOR, sin embargo, no requiere realimentación para propagar el error. Se define la teoría del algoritmo de adaptación de Brandt-Lin aplicado a una red neuronal en el diseño del controlador. La configuración propuesta en lazo cerrado se muestra en la Figura N°21.

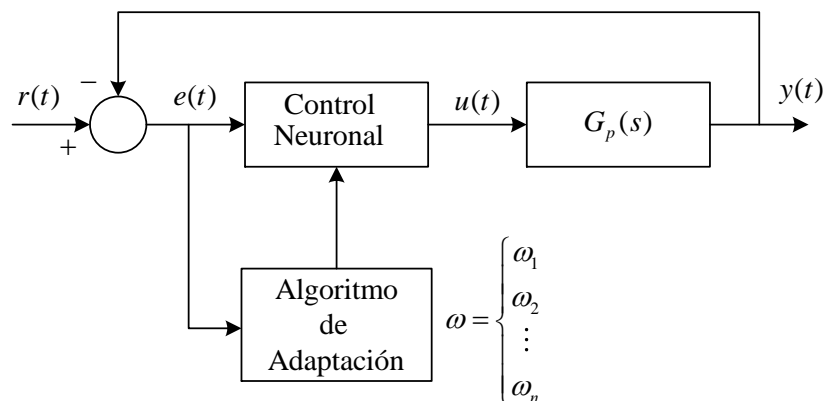


Figura N°21. Configuración a lazo cerrado del control neuronal.

Fuente: Elaboración propia.

Para ser más específico, el controlador de red neuronal tiene dos entradas  $e_1$  y  $e_2$ . Donde  $e_1$  es el error entre el punto de ajuste y la salida de la planta y  $e_2$  es una señal retardada basado en  $e_1$ . La razón para introducir  $e_2$  es como sigue. Puesto que el controlador de la red neuronal es un dispositivo con memoria a fin de que la salida de control no dependa solo de la entrada actual (error en nuestro caso), sino también en las entradas anteriores, algunas señales retardadas deben ser introducidas según la Figura N°22.

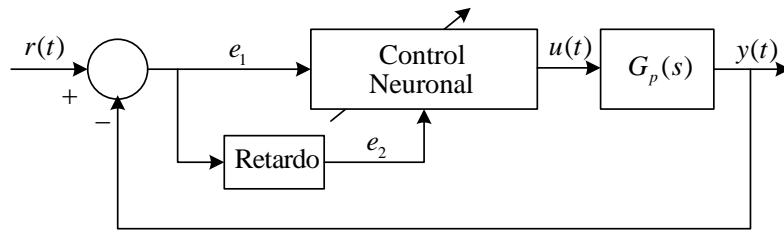


Figura N°22. Controlador de la red neuronal.

Fuente: Elaboración propia.

Si se utiliza la red neuronal simple con dos neuronas ocultas, entonces el controlador de red neuronal tendrá la estructura mostrada en la Figura N°23 es una red más sofisticada que se utiliza para mejorar la performance.

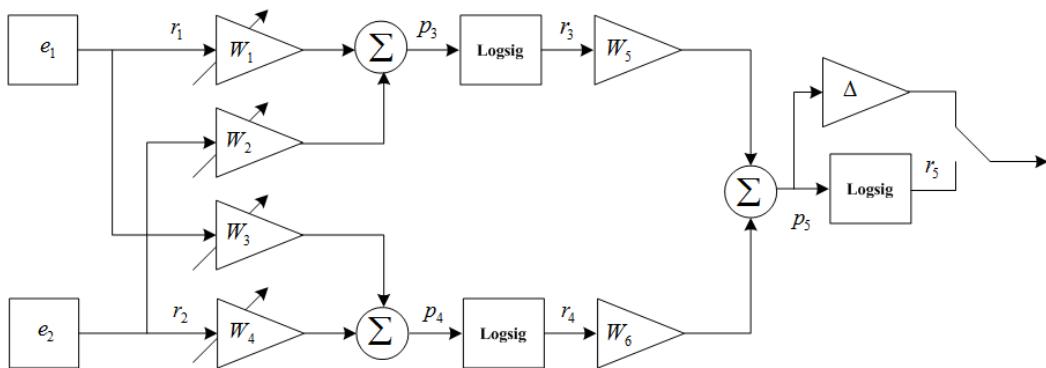


Figura N°23. Configuración del controlador neuronal adaptativo.

Fuente: Elaboración propia.

En la Figura N°23, se proponen dos maneras de configurar que la etapa de salida del controlador:

1. Función de activación tangente sigmoidea

## 2. Ganancia de salida constante

La razón detrás de la sigmoide tangente hiperbólica (tansig) es la capacidad de proporcionar una doble polaridad de la señal a la salida. Basándose en los resultados de simulación, la simple constantes de ganancia a la salida también funciona ya que a menudo ofrecen un mejor resultado. Según el esquema mostrado en la Figura N°23 se desarrolla matemáticamente, las relaciones entrada y salida de las neuronas según lo siguiente.

Entradas a la red.

$$r_1 = e_1, \quad r_2 = e_2 \quad (43)$$

La ponderación en las neuronas de la capa oculta.

$$p_3 = w_1 r_1 + w_2 r_2, \quad p_4 = w_3 r_1 + w_4 r_2 \quad (44)$$

Entradas a la capa de salida.

$$r_3 = \sigma(p_3), \quad r_4 = \sigma(p_4) \quad (45)$$

Salida de la red.

$$p_5 = w_5 r_3 + w_6 r_4 \quad (46)$$

Sea el error cuadrático.

$$E = e_1^2 = (r - y)^2 = r^2 - 2ry + y^2 \quad (47)$$

Entonces.

$$\frac{\partial E}{\partial y} = -2r + 2y = -2(r - y) = -2e_1 \quad (48)$$

Aplicando el algoritmo de Brandt-Lin a las ecuaciones según las ecuaciones (41) y (42) obtenemos.

$$\begin{aligned} \dot{w}_1 &= r_1(\phi_3 \sigma(-p_3) + \gamma \cdot 0) = e_1 \phi_3 \sigma(-p_3) \\ \dot{w}_2 &= r_2(\phi_3 \sigma(-p_3) + \gamma \cdot 0) = e_2 \phi_3 \sigma(-p_3) \\ \dot{w}_3 &= r_1(\phi_4 \sigma(-p_4) + \gamma \cdot 0) = e_1 \phi_4 \sigma(-p_4) \\ \dot{w}_4 &= r_2(\phi_4 \sigma(-p_4) + \gamma \cdot 0) = e_2 \phi_4 \sigma(-p_4) \end{aligned} \quad (48)$$

Donde,  $\phi_3 = w_5 \cdot \dot{w}_5$  y  $\phi_4 = w_6 \cdot \dot{w}_6$ .

La ley de adaptación para  $w_5$  y  $w_6$  es más complicada, ya que está vinculada a la planta para ser controlada.

Analizando a partir de la ecuación (4), ya que  $O_{post_c}$  está vacía, tenemos

$$\dot{w}_5 = -\gamma F'_{post_c}[u]r_3(-2e_1) \quad (49)$$

Si la derivada de Fréchet se aproxima por una constante que se absorbe en  $\gamma$ , entonces la expresión anterior es aproximada por.

$$\dot{w}_5 = \gamma r_3 e_1 \quad (50)$$

Similarmente.

$$\dot{w}_6 = \gamma r_4 e_1 \quad (51)$$

La constante  $\gamma$  se considera como la tasa de adaptación o aprendizaje. Será variada para analizar el ritmo de adaptación del control de la red neuronal.

## 2.4 Definición de términos básicos

### 2.4.1 Glosario de términos básicos

- Adquisición de datos: es medir un fenómeno eléctrico o físico como voltaje, corriente, temperatura, presión, sonido, etcétera. La adquisición de datos mediante dispositivos electrónicos utiliza una combinación de hardware, software y un elemento para realizar las mediciones según el procedimiento.

- Adquisición
- Análisis
- Validación

La adquisición de datos incorpora señales, sensores, actuadores, acondicionamiento de señal, dispositivos de adquisición de datos y, la aplicación en software.

- Estimación IDGREY: es una técnica de estimación de parámetros en la que se construye modelos a través de procesamiento de datos recopilados en forma experimental. Estos modelos construidos se representan en espacio

estado en dominio de tiempo continuo o dominio de tiempo discreto. Cuando creamos con el comando IDGREY de MATLAB que construye un modelo lineal de caja gris con parámetros identificables.

- Control por Interacción adaptativa: es una técnica de control que desarrolla la teoría de interacción adaptativa que se aplica para adaptar los coeficientes del sistema. Dependiendo de su aplicación y configuración del algoritmo, los coeficientes ajustados son los pesos de la red neuronal, ganancias del controlador o coeficientes de las funciones de transferencia. Para aplicar el algoritmo a un sistema de control, la única información necesaria sobre la planta es la derivada de Fréchet, que se aproxima por una constante determinada. Esto hará que el algoritmo sea robusto para incertidumbres del sistema y eventuales cambios, por tanto, es aplicable a una amplia clase de los sistemas.

#### 2.4.2 Abreviaturas utilizadas

ADC	: Analog Digital Converter (convertidor análogo a digital)
DAC	: Digital Analog Converter (convertidor digital a análogo)
MLP	: Multilayer Perceptron (red neuronal multicapa)
ANN	: Artificial Neural Networks (red neuronal artificial)
Delay	: Termino que indica un retardo en una señal
CD	: Corriente directa
DC	: Direct current (corriente directa)
PWM	: Pulse With Modulation (Modulador por ancho de pulso)
DAQ	: Termino utilizado en referencia de la tarjeta de adquisición de datos
EMG30	: Modelo del modelo del motor de corriente continua
MSE	: Mean Square Error (Error cuadrático medio)
FPE	: Focus Prediction Error (Foco de Predicción)
Black-Box	: Caja Negra
White-Box	: Caja Blanca

### III. HIPÓTESIS Y VARIABLES

#### 3.1 Hipótesis

##### 3.1.1 Hipótesis general

El diseño del control por interacción adaptativa aplicado a una red MLP permitirá un adecuado control de velocidad del motor DC.

##### 3.1.2 Hipótesis específicas

- a. Hipótesis específica 1: Una adecuada configuración de la interfase, el modo de adquisición y la frecuencia de muestreo, facilitaran la obtención de los datos de entrada y salida del proceso.
- b. Hipótesis específica 2: El tratamiento y validación de los datos provenientes de la adquisición de datos permitirá resolver el modelo experimental del proceso.
- c. Hipótesis específica 3: El desarrollo del algoritmo de control por interacción adaptativa aplicado a una red MLP, permitirá verificar la performance del sistema de control de velocidad del motor DC, haciendo que la señal de control cumpla con las consideraciones de obtener la mejor performance.

#### 3.2 Definición conceptual de las variables

##### 3.2.1 Variables Independientes

Es la variable de acción de control que está determinada por la tensión de entrada al motor de corriente continua que proporciona el torque y se denota por la variable (u).



### 3.2.2 Variables dependientes

El sistema cuenta con un grado de libertad representados por la velocidad angular  $\dot{q}_1 = \omega$  desarrollada en el eje del motor DC y cuya medida será leída en RPM (revoluciones por minuto).

### 3.2.3 Variables intervinientes:

En este caso se considera la variable de perturbaciones presentes como el ruido blanco en el sistema, sea en la entrada o en la salida.

### 3.2.4 Operacionalización de variables

Las variables que conforman el objeto de estudio deben ser medidas, lo que nos permitirá obtener datos e información de importancia, que serán tabulados, ordenados y procesados; lo que significa que deberá usarse formulaciones estadísticas, que permitan una cuantificación y análisis de los datos correspondientes a las variables participantes en el sistema, según la Tabla N°04.

Tabla N°04. Características de las variables.

Variabes	Dimensiones	Indicadores	Escala Medición
Variable Independiente X: torque	X.1. Obtención de la ley de control por interacción adaptativa	Componente del sistema de control	Cualitativo
	X.2. Medida de la acción de control	Valor observable	Cuantitativa
	X.3. Limitación de la acción de control	Grado de actuación apropiado	Cuantitativa
Variable dependiente Y: velocidad angular	Y.1 Medida del ángulo	Medida del sensor de efecto hall	Cuantitativa
	Y.2. Medida de la velocidad	Valor medido en RPM	Cuantitativa

## IV. DISEÑO METODOLÓGICO

### 4.1 Tipo y diseño de investigación

El presente trabajo de investigación es de tipo experimental (ver Figura N°24) con enfoque cuantitativo. Se aplican herramientas matemáticas, configuraciones de la interfase para el experimento para el modelo y su simulación en lazo abierto, así como para el diseño del control por interacción adaptativa y la simulación de la respuesta del sistema de control en lazo cerrado.

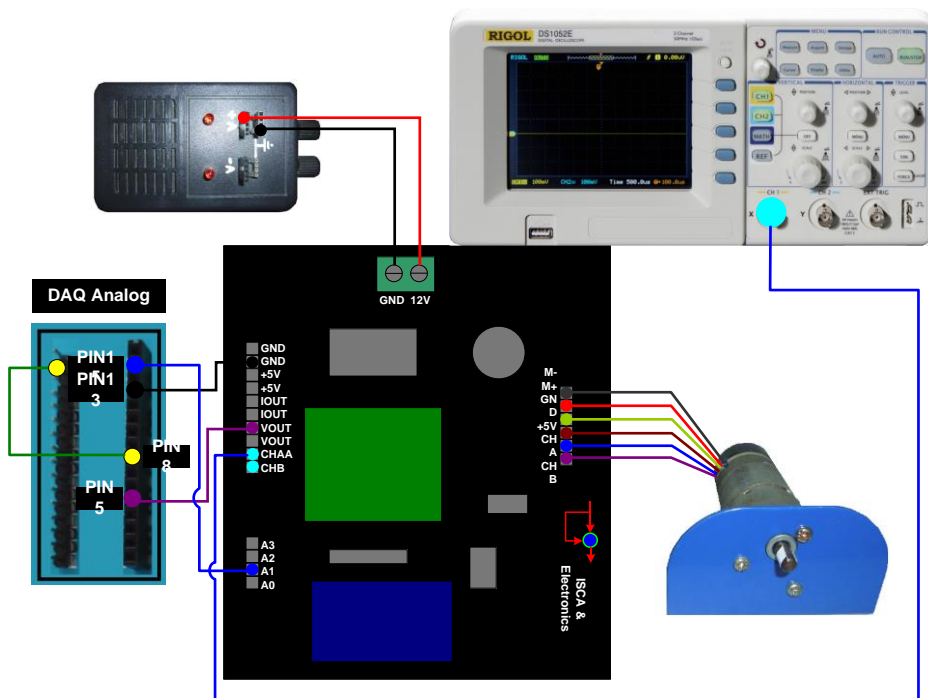


Figura N°24. Control de velocidad desde entrada analógica de la DAQ USB-6009.

Fuente: Elaboración propia.

#### 4.1.1 Regresión lineal

Supongamos que estamos midiendo la posición de un móvil en función del tiempo en un movimiento rectilíneo (ver Figura N°25). Si el móvil está libre de fuerzas, esperamos que la relación entre la posición del móvil y el tiempo sea lineal  $x = x_0 + vt$ . Donde  $x_0$  es la posición del móvil en el instante  $t = 0$ .

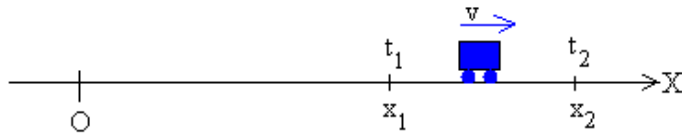


Figura N°25. Trayectoria lineal para el móvil.

Fuente: Elaboración propia.

Si medimos las posiciones del móvil  $x_1$  y  $x_2$  en los instantes  $t_1$  y  $t_2$ , obtenemos un sistema de dos ecuaciones con dos incógnitas de las que determinan las cantidades desconocidas  $x_0, v$ . Ahora bien, esta afirmación solamente es cierta en un experimento ideal libre de errores.

Si efectuamos  $n$  medidas de la posición del móvil, el aspecto de la representación gráfica de nuestras medidas según la Figura N°26.

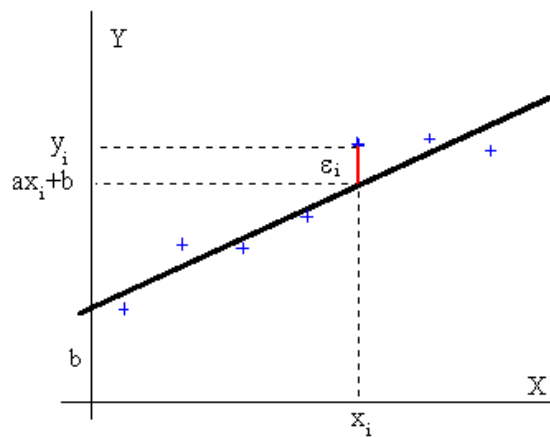


Figura N°26. Línea recta desde la regresión.

Fuente: Elaboración propia.

Los puntos de color azul representan los datos experimentales. La relación entre las ordenadas  $y$ , las abscisas  $x$  de dichos puntos es solamente aproximada, debido a los errores de cada una de las medidas.

Si tomamos únicamente dos puntos para definir la recta el resultado tendría un importante error. Para una mejor estimación de la recta  $y$ , por tanto, de las magnitudes buscadas, se deberán utilizar las  $n$  medidas tomadas.

Supongamos una magnitud física  $y$ , está relacionada con otra variable  $x$ , mediante la función  $y = ax + b$ . Una recta de pendiente  $a$  cuyo intercepto de ordenada en el origen es  $b$ . Las desviaciones  $\varepsilon$  de los valores de  $y$ , viene definido.

$$\varepsilon_1 = y_1 - (ax_1 + b)$$

$$\varepsilon_2 = y_2 - (ax_2 + b)$$

$$\varepsilon_n = y_n - (ax_n + b)$$

Sea  $E(a, b)$  la suma de los cuadrados de todas estas desviaciones:

$$E(a, b) = (y_1 - ax_1 - b)^2 + \dots + (y_i - ax_i - b)^2 + \dots + (y_n - ax_n - b)^2$$

Los valores que minimizan a  $E(a, b)$  son aquellos para los que se obtiene así, un sistema de dos ecuaciones con dos incógnitas  $a$  y  $b$  cuya solución es.

$$E(a, b) = \sum_{i=1}^n (y_i - ax_i - b)^2 \quad (52)$$

Los valores que minimizan a  $E(a, b)$  son aquellos para los que.

$$\frac{\partial E}{\partial a} = 0, \quad \frac{\partial E}{\partial b} = 0$$

Se obtienen así un sistema de dos ecuaciones con dos incógnitas  $a$  y  $b$  cuya solución es.

$$a = \frac{n \sum_1^n x_i y_i - (\sum_1^n x_i)(\sum_1^n y_i)}{n \sum_1^n x_i^2 - (\sum_1^n x_i)^2}, \quad b = \frac{\sum_1^n y_i - a \sum_1^n x_i}{n} \quad (53)$$

Estas ecuaciones son las ingresaremos a MATLAB de modo que tengamos la función de regresión y quede a la espera del ingreso de los datos (DATA).

Las expresiones más elaboradas nos permiten determinar el error de  $a$ ,  $\Delta a$  y el error de  $b$ ,  $\Delta b$  son las que listamos debajo.

$$\sigma = \sqrt{\frac{\sum_1^n (y_i - ax_i - b)^2}{n - 2}}, \quad \Delta a = \frac{\sqrt{n} \sigma}{\sqrt{n \sum_1^n x_i^2 - (\sum_1^n x_i)^2}}, \quad \Delta b = \Delta a \sqrt{\frac{\sum_1^n x_i^2}{n}}$$

La pendiente de la recta se escribirá  $a \pm \Delta a$ , y la ordenada en el origen  $b \pm \Delta b$ . Véase las reglas para expresar una medida y su error de una magnitud.

El coeficiente de correlación es otro parámetro para el estudio de una distribución bidimensional, que nos indica el grado de dependencia entre las variables X e Y. El coeficiente de correlación r es un índice que mide el grado de ajuste de los datos experimentales a la recta.

Se obtiene mediante la fórmula.

$$r = \frac{\sum_1^n (x_i - \langle x \rangle)(y_i - \langle y \rangle)}{\sqrt{\sum_1^n (x_i - \langle x \rangle)^2} \sqrt{\sum_1^n (y_i - \langle y \rangle)^2}} \quad (54)$$

El coeficiente de correlación vale cualquier número comprendido entre  $-1$  y  $+1$ .

- Cuando  $r = 1$ , la correlación lineal es perfecta, directa.
- Cuando  $r = -1$ , la correlación lineal es perfecta, inversa.
- Cuando  $r = 0$ , no existe correlación, independencia total de valores X e Y.

Los resultados de la regresión para la data obtenida desde el control de velocidad a lazo abierto son mostrados en la Figura N°27.

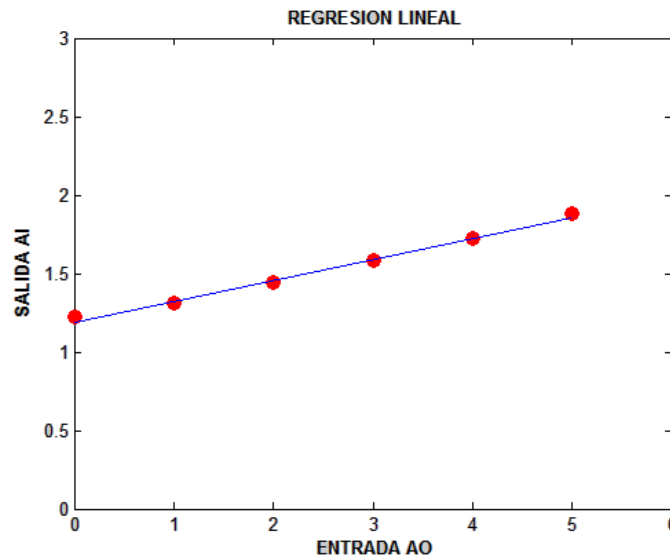


Figura N°27. Interpolando mediante una línea recta.

Fuente: Elaboración propia.

El programa MATLAB nos proporciona la ecuación de la recta:

$$y = 0.27x + 1.162$$

Esta recta servirá para escalar la velocidad manual o a lazo abierto del motor DC. El control manual a lazo abierto será realizado desde LabVIEW, para ello se requiere escalamiento previo a partir de la línea recta encontrada. En este caso el voltaje que se adquiere según:

$$x = \frac{y - b}{m}$$

La salida de la DAQ viene representada por la variable  $y$ , mientras que la variable  $x$  es la nueva salida escalada de 0V a 5V.

#### 4.1.2 Control a lazo abierto

Las características proporcionadas por el fabricante para el motor DC EMG30 son, las cuentas por revolución  $CPR = 360$ , y las máximas revoluciones para el motor sin carga de  $RPM \approx 208$  (ver ANEXO B). La relación para calcular la frecuencia máxima que proporciona el encoder, tomado en consideración que los sensores Hall están dispuestos en  $90^\circ$  es.

$$f(\text{encoder}) = \frac{(CPR) \times (MAXRPM)}{60 \times 4} \text{ [Hz]} \quad (55)$$

Reemplazando obtenemos.

$$f = 360 \frac{\text{pulsos}}{1 \text{ rev}} \times 208 \frac{\text{rev}}{\text{min}} \times \frac{1 \text{ min}}{60 \text{ seg}} \times \frac{1}{4} = 312.5 \frac{\text{pulsos}}{\text{seg}} \approx 312 \text{ Hz}$$

Por otro lado, los valores medidos en forma experimental por el osciloscopio y el tacómetro respectivamente resultaron ser:

- Medida de la frecuencia entre 312Hz a 324Hz
- Medida de las RPM entre 200 a 215

El control a lazo abierto (ver Figura N°28) se inicia desde la adquisición de datos, se da íntegramente con señales de E/S analógica previamente escalada y manejo adecuado del offset.

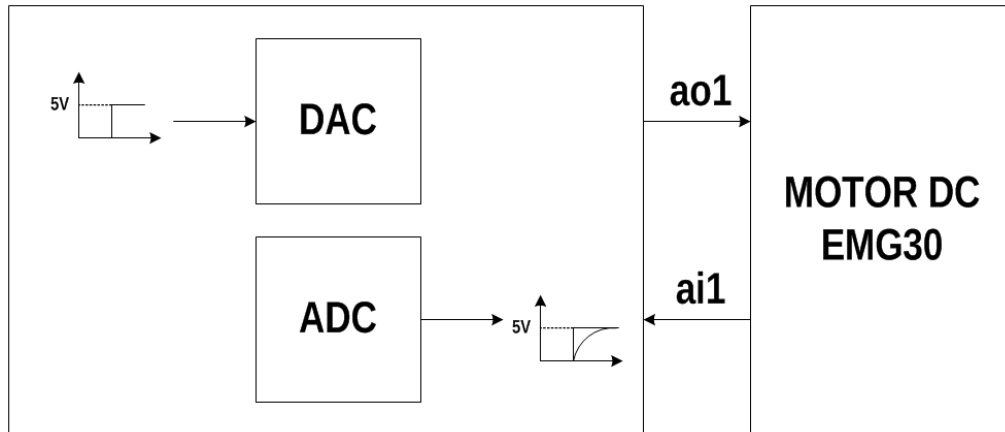


Figura N°28. Diagrama de bloques para el control manual de la velocidad.

Fuente: Elaboración propia.

El VI para el control a lazo abierto es mostrado en la Figura N°29. Observamos la ecuación de la recta para el escalamiento.

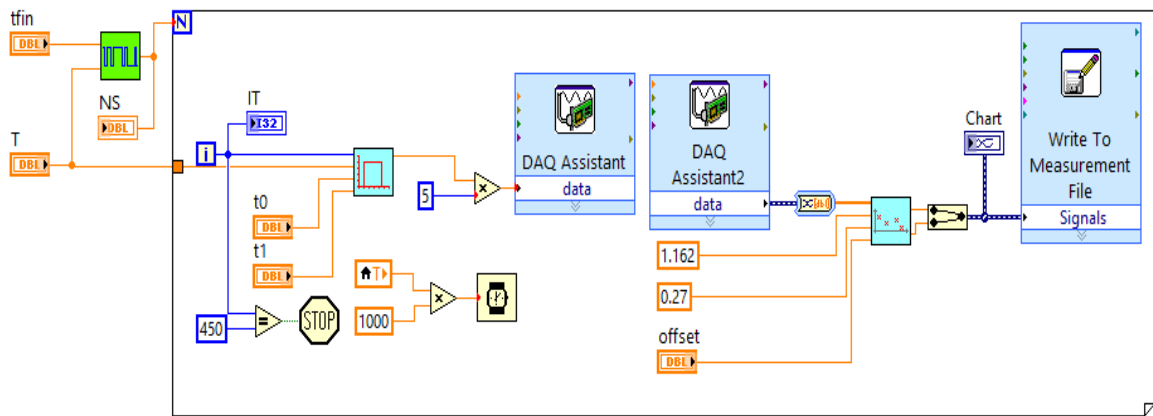


Figura N°29. VI para el control de velocidad a lazo abierto.

Fuente: Elaboración propia.

Los resultados se aprecian en la Figura N°30. Debemos indicar que es una curva que se comporta como un filtro pasa bajo esto es debido a que aún no se ha calculado la ganancia de tacómetro que se analizará más adelante.

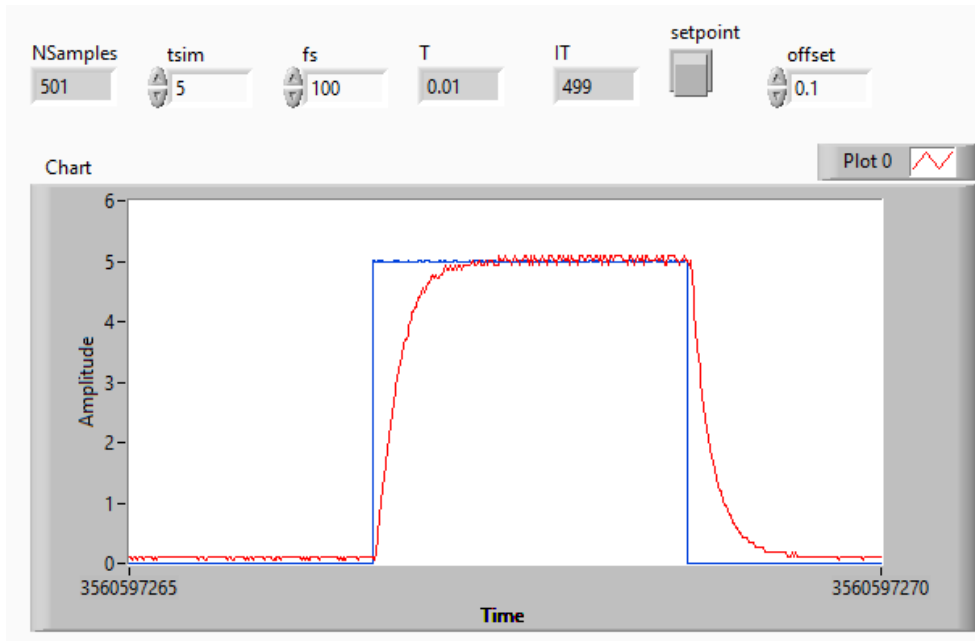


Figura N°30. Curva de la velocidad del motor EMG30 a lazo abierto.  
Fuente: Elaboración propia.

Las relaciones para expresar la velocidad angular en unidades de RPM o rad/s según:

$$w[RPM] = \frac{V_{in} * F_{max} * 60}{V_{max} * PPR} \quad (56)$$

$$w \left[ \frac{rad}{s} \right] = \frac{V_{in} * F_{max} * 2\pi}{V_{max} * PPR}$$

Por otro lado, la relación nos conduce a una curva característica donde la amplitud sea medida en RPM o en rad/seg como se muestra en la Figura N°30. En este caso se ha tenido en cuenta la relación para calcular ganancia de tacómetro definida según la ecuación.

$$K_{tac} = \frac{V_{max} * PPR}{2\pi * F_{max}} \quad (57)$$

Posteriormente se verifica la identificación del modelo simulando la planta obtenida a un valor referencial de voltaje de 12V, tal como se observa en la Figura N°31.



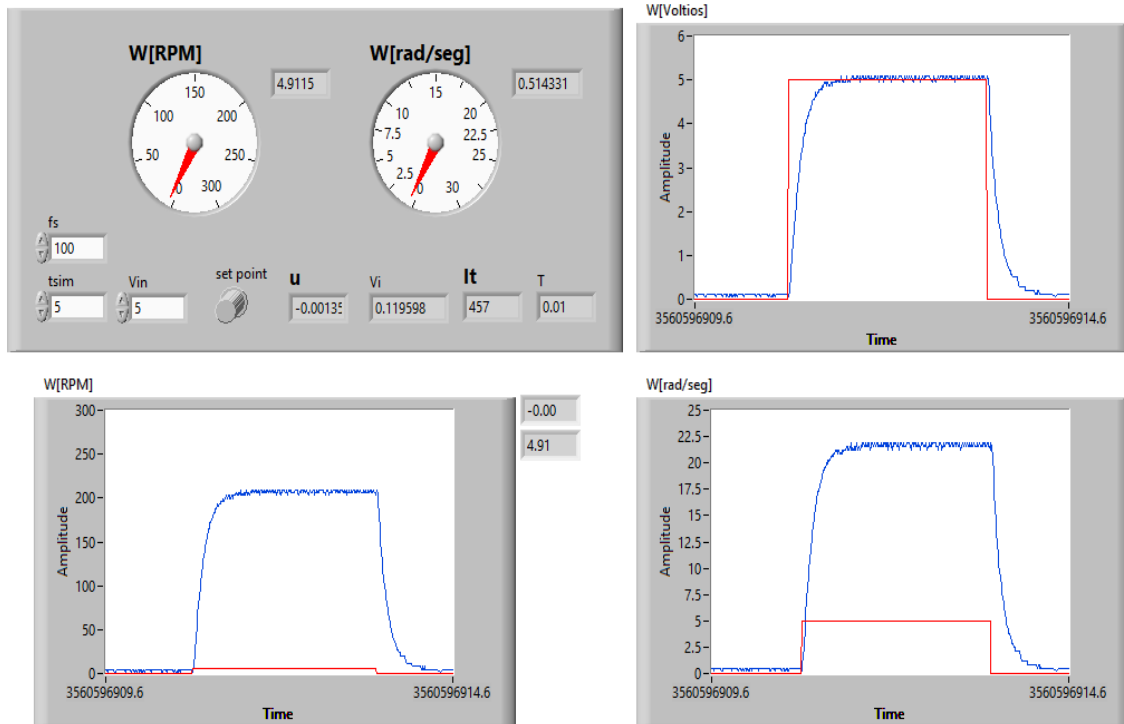


Figura N°31. VI para el control de velocidad a lazo abierto.

Fuente: Elaboración propia.

#### 4.1.3 Estimación del modelo motor DC EMG30

En principio hay dos formas diferentes de obtener los modelos: a partir de un conocimiento previo, es decir, en términos de leyes físicas, o por experimentación sobre un proceso. Un proceso no se caracteriza por un único modelo matemático. Debe representarse por una jerarquía de modelos que van desde los detallados y complejos de simulación hasta los muy sencillos, fáciles de manipular analíticamente.

El control automático de sistemas requiere del conocimiento del modelo matemático obtenido de forma teórica o experimental, tal que represente el comportamiento del mismo. Las estructuras del modelo de caja gris se extraen del conocimiento parcial del proceso afectado por las perturbaciones, como el ruido que está presente en diversas formas, una de las más usuales es el ruido blanco. En nuestro caso un modelo representado por la siguiente ecuación espacio estado:

$$\dot{x}(t) = -\frac{1}{\tau}x(t) + \frac{K}{\tau}u(t)$$

$$y(t) = x(t) + e(t)$$
(58)

Siendo  $e(t)$  la perturbación a la salida de la planta, además se tiene dos parámetros iniciales como son la ganancia  $K$  y la constante de tiempo  $\tau$ . Las condiciones iniciales cero  $x(0) = 0$ , considerado como un sistema causal.

El siguiente procedimiento nos conducirá a encontrar el modelo estimado.

a. Medida de datos de entrada-salida

Como se mencionó anteriormente, todo proceso de identificación parte de un conjunto de datos de entrada/salida obtenidos de forma experimental a partir del sistema físico que se pretende modelar. Generalmente, sólo se utiliza una parte de esos datos para realizar la identificación del modelo, y los restantes se reservan para realizar la validación del mismo.

Estos archivos son cargados en el Workspace de MATLAB en algún tipo de formato, \*.dat, \*.lvm, \*.mat, etc. Las siguientes líneas de código MATLAB nos conducen a la lectura de los datos experimentales.

Lectura de data
<pre>clear; close all; clc filename = 'dataemg30.mat'; load(fullfile('C:\Users\61NB3F\Documents\MATLAB\Data', filename)) who % Your variables are: % % data_emg30m7  filename      u      y whos</pre>

La Figura N°32 muestra la representación de datos de entrada salida obtenidos en forma experimental en unidades de voltaje (V).

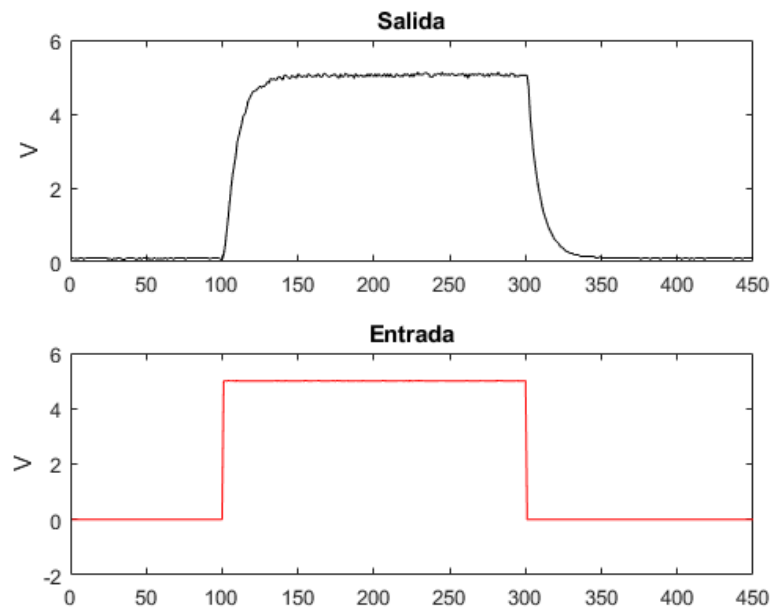


Figura N°32. Graficas de entrada y salida.

Fuente: Elaboración propia.

Para el motor DC EMG30, el modelo se ajusta a la siguiente función de transferencia.

$$\frac{Y(s)}{U(s)} = \frac{G}{\tau s + 1} \quad (59)$$

La ecuación (59) corresponde a un modelo a estimar. Las ecuaciones (18) y (19) nos referencian a la función de MATLAB denominada motorODE.m cuyo programa se lista a continuación.

#### Algoritmo 1: Función ODE

```
function [A,B,C,D]=motorODE(G,tau,Ts)
A = -1/tau;
B = G/tau;
C = 1;
D = 0;
if Ts>0
    [n,m]=size(B); % 2x1
    s=expm( [[A B]*Ts; zeros(m,n+m) ] );
    A=s(1:n,1:n);
    B=s(1:n,n+m);
end
end
```

## b. Modelo estimado

El modelo estimado que se obtendrá desde el IDGREY requiere la función `motorODE` y condiciones iniciales de la constante de tiempo, así como de la ganancia del proceso.

### Algoritmo 2: Función IDGREY

```
fs=50;  
T=1/fs;  
mi=idgrey(@motorODE,{'G',5;'tau',1},'cd',{},T);  
m=greyest([y,u],mi);  
opt = compareOptions('InitialCondition','zero');  
compare([data_emg30m7(:,2),data_emg30m7(:,4)],m,Inf,opt);
```

El modelo estimado para 450 muestras viene representado por la función de transferencia.

$$\hat{G}(s) = \frac{1.01}{0.1776s + 1} \quad (60)$$

Comprobaremos la respuesta debido a una entrada pulso, es decir, comparando con la data, vemos que el modelo aproxima bastante bien al modelo real como se muestra en la Figura N°33.

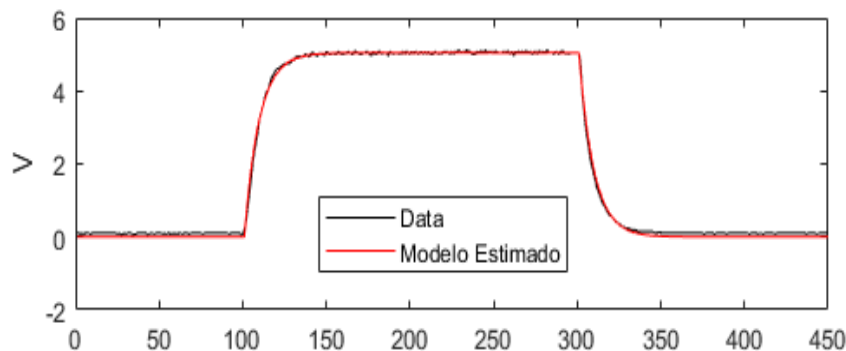


Figura N°33. Comparación de respuestas debido a una entrada pulso.

Fuente: Elaboración propia.

## c. Modelo estimado con ganancia de tacómetro

La elección de la estructura del modelo es una de las decisiones más importantes y difíciles que debe tomar el diseñador. Si no se tiene ningún conocimiento del

sistema que facilite dicha elección. Es el caso del modelo obtenido en (60) presenta una respuesta tipo filtro pasa bajo y no realmente la curva característica del motor DC. Debido a ello es que realizamos algunos ajustes que indicaran el modelo estimado correcto en unidades de rad/s/V (ver Figura N°34).

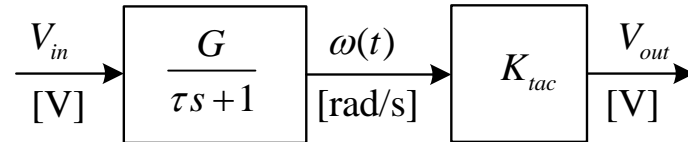


Figura N°34. Diagrama de bloques del motor DC con ganancia de tacómetro.

Fuente: Elaboración propia.

Donde  $K_{tac}$  es una constante proporcional con unidades [V-s/rad] utilizada para convertir la velocidad rotacional  $\omega$  expresada en [rad/s] en una salida de voltaje V. Teniendo en cuenta las relaciones siguientes podemos encontrar la ganancia  $K_{tac}$  según la relación matemática.

$$K_{tac} = \frac{V_{max}PPR}{2\pi F_{max}} \quad (61)$$

Donde.

- PPR son los pulsos por revolución.
- $V_{max}$  es el voltaje máximo aplicado al motor.
- $F_{max}$  es la frecuencia máxima del encoder de efecto hall.

El modelo estimado incorporando la ganancia de tacómetro viene representado por la función de transferencia.

$$\hat{G}(s) = \frac{10.45}{s + 5.631} \quad (62)$$

Comprobaremos la respuesta debido a una entrada escalón unitario de amplitud 12V, (ver Figura N°35) se observa que el modelo presenta una buena performance ya que el ajuste de la estimación esta por 96.51%.

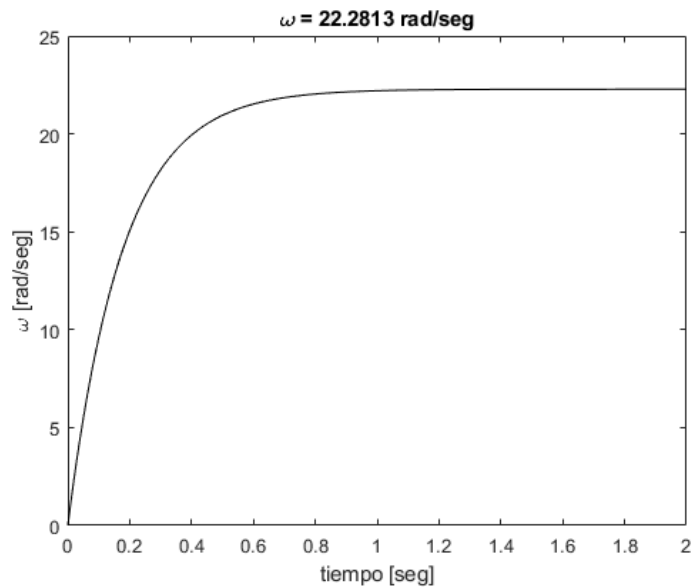


Figura N°35. Curva característica del motor DC EMG30 a lazo abierto.

Fuente: Elaboración propia.

d. Discretización de la planta estimada

Para poder discretizar la planta Identificada de forma correcta, debemos de encontrar el periodo de muestreo más adecuado según.

$$dt = \frac{\tau}{6} \approx 0.02s$$

Este valor del periodo de muestreo coincide con el periodo de muestreo obtenido en la adquisición de datos en la parte experimental.

Realizando la discretización de la planta

$$\frac{\omega(z)}{V(z)} = z^{-1} \frac{0.1978}{1 - 0.8935 z^{-1}}$$

Analizaremos el comportamiento de la planta identificada para una entrada escalón unitario cuando se aplica  $U = 12V$ .

$$V(z) = U \frac{z}{z - 1}$$

Esto nos ayudará a comprender cuál será el comportamiento de la planta frente a un determinado estímulo y a seleccionar la función de activación a la salida del controlador más adecuada.

Analizaremos el teorema del valor final.

$$TVF = \lim_{z \rightarrow 1} [(z - 1)\omega(z)]$$

$$\omega(z) = \frac{0.1978z^{-1}}{1 - 0.8935z^{-1}} \frac{12z}{z - 1}$$

$$TVF = \lim_{z \rightarrow 1} \left[ (z - 1) \frac{0.1978z^{-1}}{1 - 0.8935z^{-1}} \frac{12z}{(z - 1)} \right] \cong 22 \text{ rad/s}$$

¿Qué significado tiene que el valor en estado estacionario debido a una entrada escalón unitario sea casi 22rad/s? ¿De qué forma nos ayuda este resultado a seleccionar la función de activación a la salida del controlador?

Imaginemos que seleccionamos una función de activación tangente hiperbólica (tansig) a la salida del controlador; como sabemos, la función tansig limita su salida a valores comprendidos entre  $-1$  y  $1$ , esto quiere decir que a la entrada de la planta estimada le estarían llegando valores comprendidos entre  $-1$  y  $1$ .

Para la planta estimada estos valores extremos de  $-1$  y  $1$  se comportan como escalones unitarios por lo que la salida de la planta frente a estos estímulos estará entre  $-22$  y  $22$  según lo visto en el teorema del valor final; además, estos valores son cotas en el estado estable para la salida de la planta, esto quiere decir que para una función de activación tansig a la salida del controlador el mínimo y máximo valor estable que alcanza el sistema es  $-22$  y  $22$  que sería el valor de las máxima RPM que el motor DC EMG30 proporciona.

Esto quiere decir que la referencia con una amplitud mayor a  $1$ , por ejemplo  $22$ , el sistema no podrá adaptarse ya que la planta identificada nunca podrá generar el valor de  $22$  en estado estable. Si en lugar seleccionar una función de activación tansig seleccionamos una ganancia entonces, dependiendo del valor de la ganancia, no tendríamos una restricción en la salida de la planta identificada, por lo

que el sistema sería capaz de adaptarse para cualquier valor de amplitud que tenga la señal de referencia.

e. Diseño del control por interacción adaptativa

El desarrollo descriptivo del algoritmo de control por interacción adaptativa para el control del motor DC EMG30 se describe a continuación.

1. Señal de referencia

Generalmente para un sistema de control se propone una señal constante tipo escalón unitario,  $u(t)$  o escalón desfasado  $u(t) - u(t - t_0)$ , denominado también “SPIKE” sin embargo, también es importante otra señal diferente como una señal de tipo pulso o senoidal, esto con el propósito de que la respuesta del sistema se observe el tracking como la señal de salida se va adaptando a la referencia.

Una señal senoidal de la forma:

$$u(t) = A \sin(\omega t + \varphi) + \text{offset} \quad (63)$$

Se programa fácilmente en MATLAB o LabVIEW.

Algoritmo 3: Señal senoidal
<pre>% señal de referencia f = 0.1; periodo = 1/f; num_periodos = 5; stop_time = periodo*num_periodos; dt=0.001; t = 0:dt:stop_time; amp = 1; offset = 1; u = amp*sin(2*pi*f*t)+offset; plot(t,u, 'b') grid xlabel('tiempo[seg]') ylabel('y(t)')</pre>

El resultado de la simulación se muestra en la Figura N°36.



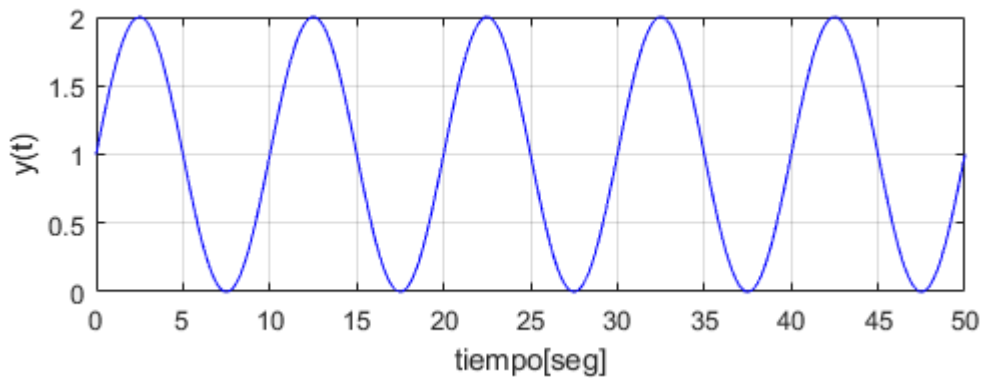


Figura N°36. Curva senoidal para cinco ciclos.

Fuente: Elaboración propia.

## 2. Algoritmo de control

En este caso se explicará en detalle el procedimiento algorítmico resuelto con el programa MATLAB. Es importante destacar que la adaptación se da de forma natural, es decir el algoritmo se inicializa con parámetros de sintonía que son previamente seteados y estos van cambiando según la búsqueda de la mejor performance de la adaptación según la Tabla N°05.

Tabla N°05. Algoritmo descriptivo.

Algoritmo 4: Control por interacción adaptativa		
it	Característica	MATLAB
1	Constante de adaptación ( $\gamma$ )	<code>gamma=2.5;</code>
2	Ganancia de salida ( $g$ )	<code>gain=0.5;</code>
3	Numero de muestras ( $N$ )	<code>N=length(t);</code>
4		<b>Inicio de LOOP</b>
5	Ecuación en diferencias	<code>y(n)=-a1*y(n-1)+b1*x(n-1);</code>
6	Error ( $e$ )	<code>e(n)=u(n)-y(n)</code>
7	Ecuaciones	<code>p1 = e(n)*w1(n)+e(n-1)*w2(n);</code>
8	$r_1 = e_1$	<code>p2 = e(n)*w3(n)+e(n-1)*w4(n);</code>
9	$r_2 = e_2$	<code>r1 = logsig(p1);</code>
10	$p_1 = w_1 r_1 + w_2 r_2$	<code>r2 = logsig(p2);</code>
11	$p_2 = w_3 r_1 + w_4 r_2$	<code>a = r1*w5(n)+r2*w6(n);</code>
12	$r_1 = \sigma(p_1)$	<code>x(n) = tansig(a);</code>
	$r_2 = \sigma(p_2)$	
	$a = r_1 w_5 + r_2 w_6$	
	$x = \tanh(a)$	
13		<b>LOOP Condicional</b>
14	$\dot{w}_5 = \gamma r_2 e$	<code>dw6(n+1) = gamma*r2*e(n);</code>
15	$\dot{w}_6 = \gamma r_1 e$	<code>dw5(n+1) = gamma*r1*e(n);</code>
16		<b>Regla de actualizacion de pesos</b>
17	$w_5(n+1) \approx \frac{w_5(n+1) + w_5(n)}{2} dt$	<code>w6(n+1) = w6(n)+0.5*dt*(dw6(n+1)+dw6(n));</code>
18	$w_6(n+1) \approx \frac{w_6(n+1) + w_6(n)}{2} dt$	<code>w5(n+1) = w5(n)+0.5*dt*(dw5(n+1)+dw5(n));</code>

```

19  $\phi_1 = w_5 \dot{w}_5$           dw1(n+1) = e(n) * w5(n+1) * dw5(n+1) * logsig(-p1);
20  $\dot{w}_1 = e_1 \phi_1 \sigma(-p_1)$  dw2(n+1) = e(n-1) * w5(n+1) * dw5(n+1) * logsig(-p1);
21                                     dw3(n+1) = e(n) * w6(n+1) * dw6(n+1) * logsig(-p2);
22                                     dw4(n+1) = e(n-1) * w6(n+1) * dw6(n+1) * logsig(-p2);
23
24
25                                     w1(n+1) = w1(n) + 0.5*dt*(dw1(n+1)+dw1(n));
26                                     w2(n+1) = w2(n) + 0.5*dt*(dw2(n+1)+dw2(n));
27                                     w3(n+1) = w3(n) + 0.5*dt*(dw3(n+1)+dw3(n));
28                                     w4(n+1) = w4(n) + 0.5*dt*(dw4(n+1)+dw4(n));
27                                     FINALIZA LOOPS
28 Calcula error cuadrático ( $e^2$ ) e2 = e.*e;

```

### 3. Simulaciones

Los resultados de las simulaciones fueron realizados haciendo las variaciones de la ganancia de adaptación  $\gamma$ , el valor de la ganancia de salida  $G$ . Los resultados de la simulación son mostrados en la Figura N°37. Podemos observar la excelente performance que muestra la actuación de control; asimismo la señal de error converge a cero y lo más importante que la salida de control es suave y su tiempo de establecimiento es de aproximadamente menor a 2 segundos.

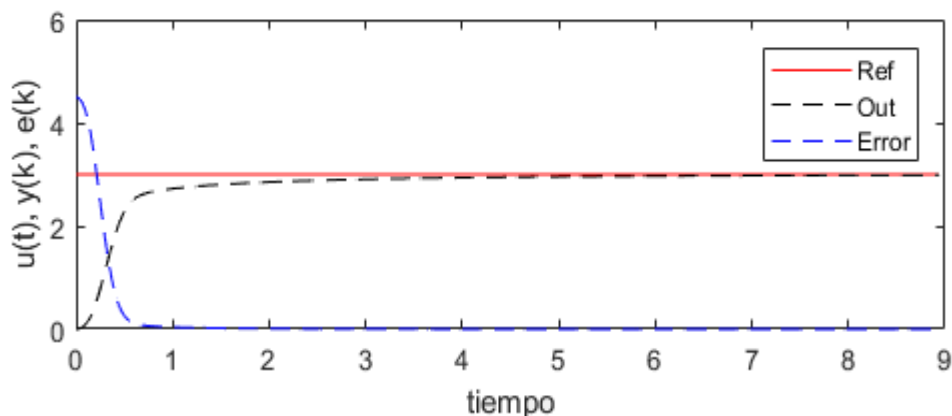


Figura N°37. Respuestas del sistema de control.

Fuente: Elaboración propia.

En el capitulo de muestras de resultados y contrastación de la hipótesis veremos que siempre es posible mejora la performance en simulaciones para tareas de tiempo real en cualquier escenario ya que la técnica empleada es bastante robusta y lo hace posible, aún a diferentes trayectorias que es sometido el proceso.

## 4.2 Método de investigación

Los métodos de investigación que se ha utilizado, son las herramientas que nos han permitido obtener y analizar los datos. Estas incluyen el muestreo, las hojas de datos técnicas, las pruebas realizadas en los experimentos.

En nuestro caso hemos optado por el método experimental que es uno de los métodos de investigación cuantitativa principal. Nuestros resultados han sido probados en laboratorio bajo el contexto del marco teórico y metodológico y realizando pruebas científicamente aceptables como son:

- Recopilar información.
- Realizar la adquisición de datos.
- Determinar las ecuaciones diferenciales o ecuaciones en diferencias del sistema, según el dominio del tiempo en que trabajemos.
- Resolver la estimación del modelo IDGREY bajo la plataforma de MATLAB leídos desde datos experimentales.
- Realizar el diseño del control por interacción adaptativa.
- Desarrollar el algoritmo de control.
- Pruebas usando software de simulación de MATLAB y LabVIEW.
- Pruebas de validación.

## 4.3 Población y muestra

No corresponde al presente trabajo de Tesis, debido a que la población se reduce al objeto de estudio que es el sistema de control de velocidad angular del motor DC EMG30.

## 4.4 Lugar de estudio y periodo desarrollado

El presente trabajo de investigación se desarrolla en las instalaciones del laboratorio de ISCA & Electronics y aulas virtuales de la Unidad de Posgrado FIEE de la Universidad Nacional del Callao.

#### 4.5 Técnicas e instrumentos para la recolección de la información

Las técnicas para recolección de datos son de tipo cuantitativo, en consecuencia, el plan para la indicada recolección de datos es la siguiente:

a) Instrumentos para la medición de voltaje:

Por la naturaleza del trabajo de investigación se han utilizado técnicas de colección de datos, para estimar el modelo del motor DC mediante el instrumento conocido como tarjeta de adquisición de datos. Esta información será procesada mediante algoritmos en el dominio del tiempo continuo y tiempo discreto.

b) Instrumentos para la lectura de la entrada y salida análoga:

Los puertos de E/S análoga de la NIDAQ USB-6009 permiten a través del MAX poner valores de voltaje de 0 a 5V a través del puerto DAC para luego ser leídos a través de un puerto ADC de  $\pm 10V$ .

c) Instrumentos de medida de hardware y software:

Se usó la NIDAQ USB-6009, Arduino Mini Pro para la modulación por ancho de pulso (PWM), convertidor de frecuencia, y una laptop Toshiba Core i5. Asimismo, a nivel de software se usó LabVIEW, MATLAB y Simulink.

d) Instrumentos para medición de frecuencias:

Se utilizó osciloscopio RIGOL y tacómetros para las RPM para determinar los valores de las frecuencias del encoder de efecto hall del motor DC EMG30.

#### 4.6 Análisis y procesamiento de datos

##### 4.6.1 Registro de datos

El proceso de datos se inicia con la toma de las medidas de voltaje en.

- Voltaje generado por el DAQ.

- Voltaje de salida del convertidor frecuencia-voltaje.
- Frecuencia del encoder tomada mediante el osciloscopio.

Con los datos obtenidos, se pretende obtener una gráfica lineal que relaciona el voltaje generado por el DAQ y el voltaje a la salida del convertidor F-V. El diagrama de LabVIEW utilizado se muestra en la Figura N°38.

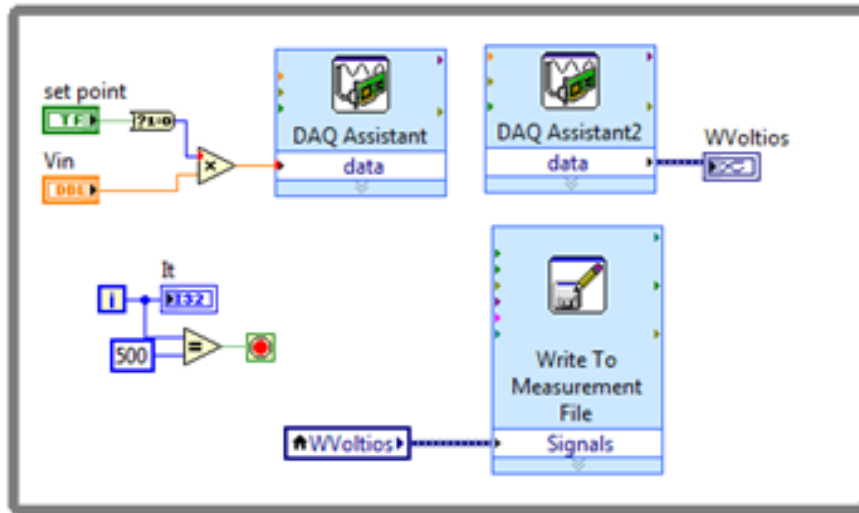


Figura N°38. Diagrama de LabVIEW para obtener datos de F-V y Voltaje AO.  
Fuente: Elaboración propia.

#### 4.6.2 Análisis del escalamiento

Se aplica el análisis de los datos mediante un algoritmo de regresión lineal y se procesan desde la Tabla 3 y este resultado se muestra en la Figura N°39. El algoritmo de regresión es listado y está escrito en MATLAB.

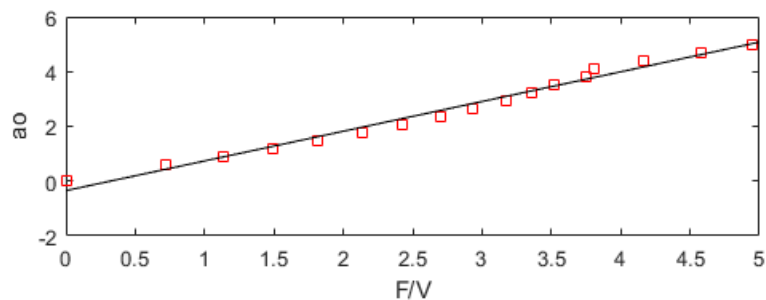


Figura N°39. Curva de regresión lineal.  
Fuente: Elaboración propia.

El siguiente algoritmo denominado “regresión” será el encargado de leer la data proveniente de los archivos que han sido generados por las mediciones entregadas desde el Firmware de LabVIEW. Como resultado obtendremos la ecuación de la línea recta que servirá para el escalamiento.

#### Algoritmo 5: Regresión lineal

```
function [m,b,r]=regresion(x,y)
n=length(x);
m=zeros(2,1);
b=zeros(2,1);
% pendiente de la recta de regresion
m(1)=(n*sum(x.*y)-sum(x)*sum(y))/(n*sum(x.^2)-sum(x)*sum(x));
%ordenada en el origen,b
b(1)=(sum(y)-m(1)*sum(x))/n;
% errores de m y b
sd2=sum((y-m(1)*x-b(1)).^2);
m(2)=sqrt(sd2/(n-2))/sqrt(sum(x.^2)-sum(x)*sum(x)/n);
b(2)=sqrt(sum(x.^2)/n)*m(2);
% correlacion
sx=sqrt(sum(x.^2)/n-mean(x)^2);
sy=sqrt(sum(y.^2)/n-mean(y)^2);
r=(sum(x.*y)/n-mean(x)*mean(y))/(sx*sy);
end
```

El SubVI mostrado en la Figura N°40 contiene la ecuación de la recta obtenida desde la regresión.

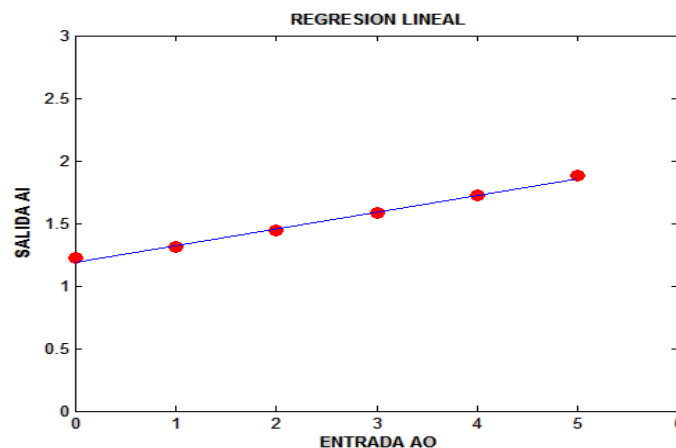


Figura N°40. Ecuación de la recta del escalamiento.

Fuente: Elaboración propia.

La respuesta del escalamiento que consistió en regular la salida del convertidor F/V de tal modo que si enviamos 5V por la salida análoga (ao) el nivel medido a la salida

del convertidor F/V desde un canal de entrada analógica (*ai*) debiera obtener un nivel de ganancia DC de 5V, como un comportamiento lineal (ver Figura N°41).

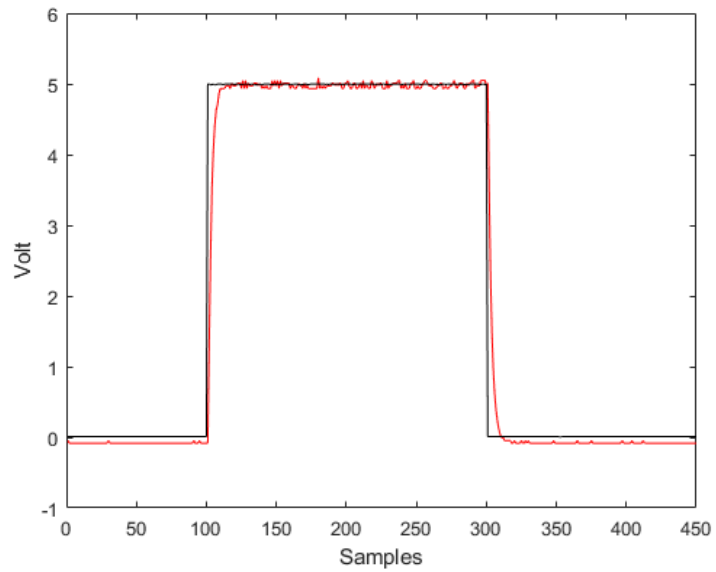


Figura N°41. Respuesta escalada de 0 a 5VDC en 450 muestras.

Fuente: Elaboración propia.

## V. RESULTADOS

### 5.1 Resultados descriptivos

#### (a) Resultado 1

Una de las principales características y beneficios del análisis descriptivo es la investigación que se realiza, nos ayuda a comprender el problema y los posibles fenómenos que pudieran aparecer, entre ellos, como evitar el fenómeno del aliasing que aparece cuando se adquieren señales análogas, el error de cuantización, las no linealidades que pudieran estar presentes en la parte mecánica del actuador y el ruido introducido en la planta o proceso.

#### (b) Resultado 2

Por otro lado, está la información proporcionada por los resultados de algún programa que esté relacionado con la optimización como es el algoritmo del IDGREY de MATLAB en donde nos da un reporte de la Tabla N°06.

Tabla N°06. Valores del estado del foco de predicción.

Status	
Estimated using	GREYEST on time domain data
Fit to estimation data	96.51%
FPE	0.00688
MSE	0.006819

#### Error de predicción final de Akaike (FPE)

El criterio de Error de predicción final (FPE, Function Prediction Error) de Akaike proporciona una medida de la calidad del modelo al simular la situación en la que el modelo se prueba en un conjunto de datos diferente.

Después de calcular varios modelos diferentes, se comparan utilizando este criterio. Según la teoría de Akaike, el modelo más preciso tiene el FPE más pequeño.



Si usa el mismo conjunto de datos para la estimación y validación del modelo, el ajuste siempre mejora a medida que aumenta el orden del modelo y, por lo tanto, la flexibilidad de la estructura del modelo.

El error de predicción final (FPE) de Akaike se define mediante la siguiente ecuación.

Dónde.

$$FPE = det \left( \frac{1}{N} \sum_t^N e(t, \hat{\theta}_N) (e(t, \hat{\theta}_N))^T \right) \left( \frac{1 + d/N}{1 - d/N} \right) \quad (64)$$

- $N$  es el número de valores en el conjunto de datos de estimación.
- $e(t)$  es un vector de errores de predicción  $n \times 1$ .
- $\hat{\theta}_N$  representa los parámetros estimados.
- $d$  es el número de parámetros estimados.

Si el número de parámetros excede el número de muestras, el FPE no se calcula cuando se realiza la estimación del modelo (model.Report.FPE está vacío).

### Error cuadrático medio (MSE)

Definamos una ecuación matemática que nos dará el error cuadrático medio de todos los puntos.

$$MSE = \frac{1}{N} \sum_{I=1}^N (y_i - \tilde{y}_i)^2 = \frac{1}{N} \sum_{I=1}^N (e)^2 \quad (65)$$

Analicemos lo que realmente significa esta ecuación, ya que, nuestro objetivo es minimizar el error cuadrático ( $e$ ) sobre todos los puntos ( $N$ ) este medio, lo que nos proporcionará la mejor línea que pase por todos los puntos.

## 5.2 Resultados inferenciales

### 5.2.1 Resultado 1.

Ingresando los siguientes datos (ver Tabla N°07) en donde está la ganancia,  $G$ , el factor de adaptación  $\gamma$ , los pesos iniciales  $W$  y las observaciones según la salida.

Tabla N°07. Datos inferenciales 1.

Parámetros	Observaciones
G=0.001; gamma=10; W1=-100; W2=100; W3=100; W4=-100; W5=-100; W6=100;	Función Sigmoide a la salida  ADAPTA

En la simulación que se muestra en la Figura N°42 se observa que el sistema adapta más tarde con un error inicial bastante alto.

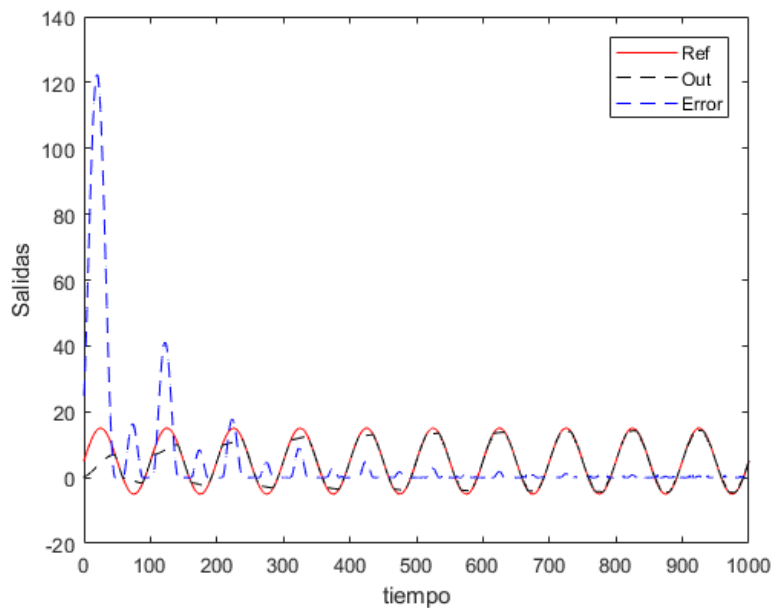


Figura N°42. Respuestas de adaptación 1 a una entrada senoidal.

Fuente: Elaboración propia.

Por otro lado, destacamos el valor de los pesos sinápticos que son de valor relativamente alto en comparación con los pesos que se obtienen desde el algoritmo de propagación inversa. Esto se debe a que el sistema de adaptación interactiva requiere más fuerza en los pesos, sin embargo, según lo mostrado en la Figura N°43 estos permanecen constantes, en otro caso decrecen.

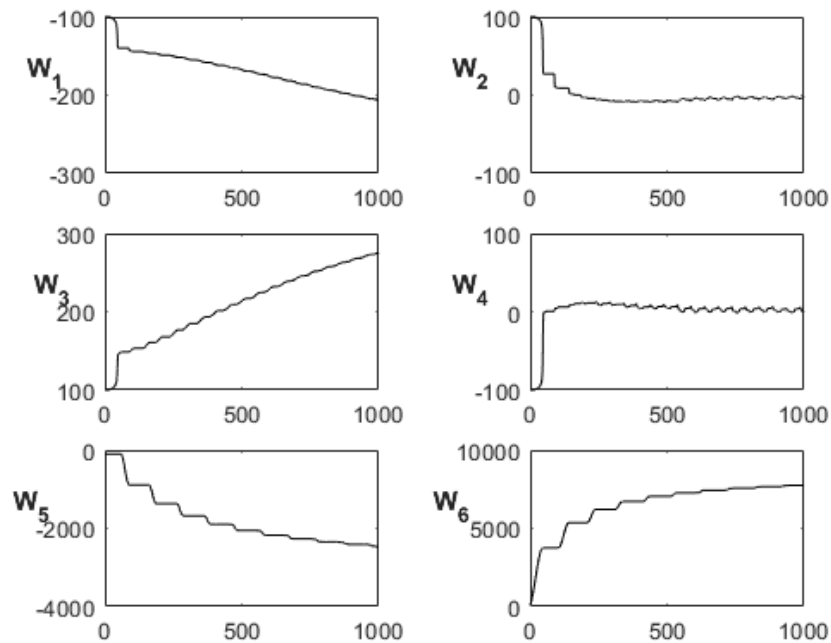


Figura N°43. Respuestas de evolución de pesos sinápticos 1.

Fuente: Elaboración propia.

### 5.2.2 Resultado 2.

Ingresando un nuevo factor de adaptación según la Tabla N°08, mientras que los demás permanecen constantes.

Tabla N°08. Datos inferenciales 2.

Parámetros	Observaciones
G=0.001; gamma=20; W1=-100; W2=100; W3=100; W4=-100; W5=-100; W6=100;	Función Sigmoide a la salida  ADAPTA

Los nuevos resultados se observan en la Figura N°44.

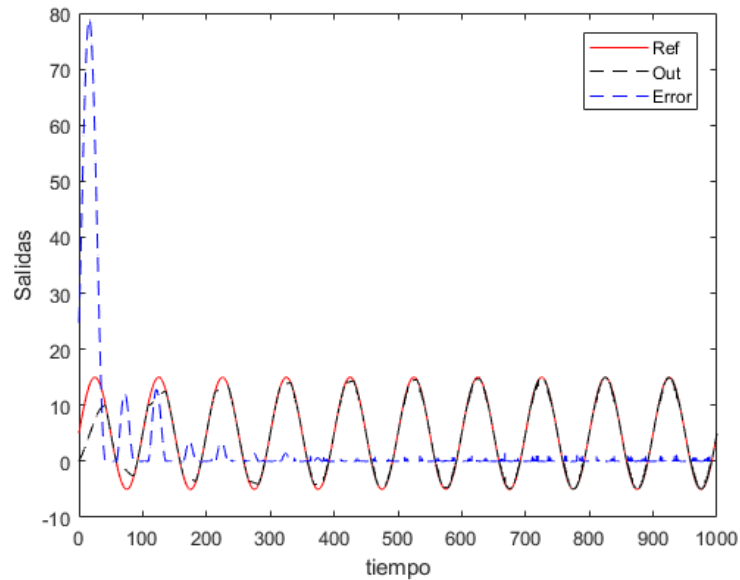


Figura N°44. Respuestas de adaptación 2 a una entrada senoidal.

Fuente: Elaboración propia.

### 5.2.3 Resultado 3

Los parámetros del nuevo factor de adaptación son definidos en la Tabla N°09.

Tabla N°09. Datos inferenciales 3.

Parámetros	Observaciones
G=0.001; gamma=20; W1=-1; W2=1; W3=1; W4=-1; W5=-1; W6=1;	Ganancia salida  NO ADAPTA

Ingresando un nuevo factor de adaptación, observamos que no hay adaptación, ya que los valores elegidos hacen que la señal de control sea creciente en tanto los pesos no ayuden a estabilizar, el control se desestabiliza al sistema, esto podemos observar en la Figura N°45.

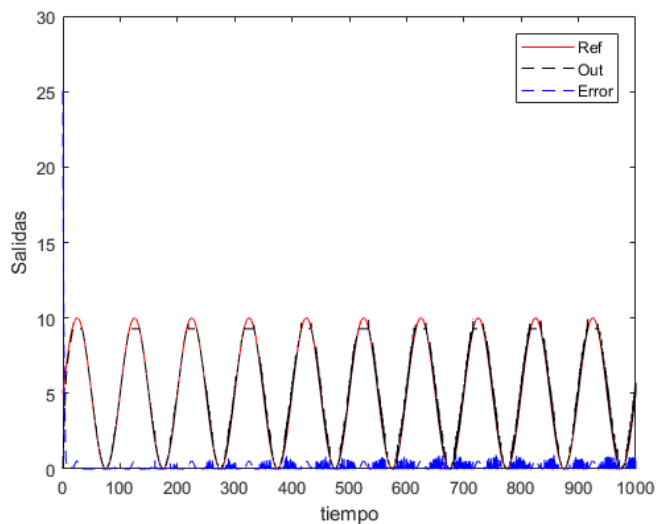


Figura N°45. Respuestas de adaptación 3 a una entrada senoidal.

Fuente: Elaboración propia.

En este caso los pesos decisivos como son los  $W_5$  y  $W_6$  no logran adaptar al sistema de control ya que cada vez que lo intentan estos crecen exponencialmente no permitiendo una buena performance en el control por interacción adaptativa (ver Figura N°46).

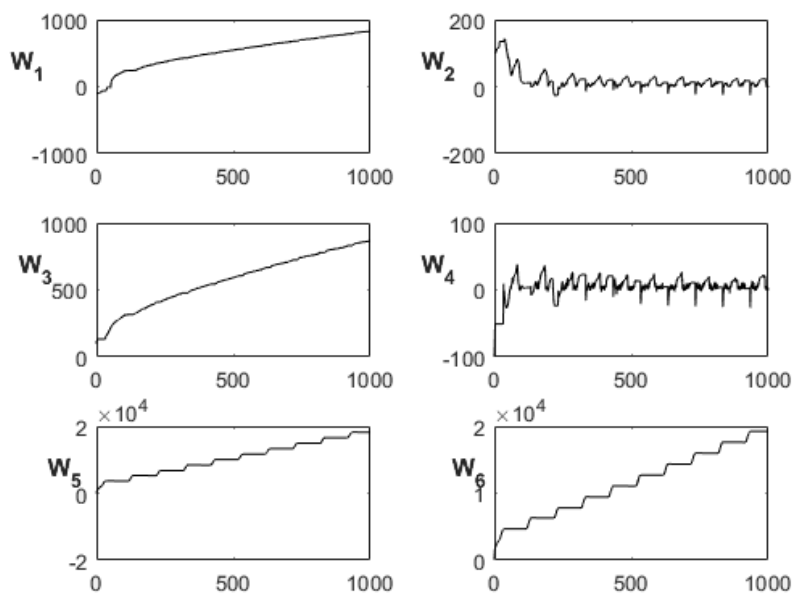


Figura N°46. Respuestas de evolución de pesos sinápticos 2.

Fuente: Elaboración propia.

### 5.2.4 Resultado 4

Los parámetros del nuevo factor de adaptación son definidos en la Tabla N°10. Esta vez se está considerando una entrada STEP con salida ganancia.

Tabla N°10. Datos inferenciales 3.

Parámetros	Observaciones
G=0.005; gamma=20; W1=-1; W2=1; W3=1; W4=-1; W5=-1; W6=1;	Ganancia salida  ADAPTA

En este caso los pesos han sido decisivos para que el sistema se adapte a una entrada step como se observa en la Figura N°47.

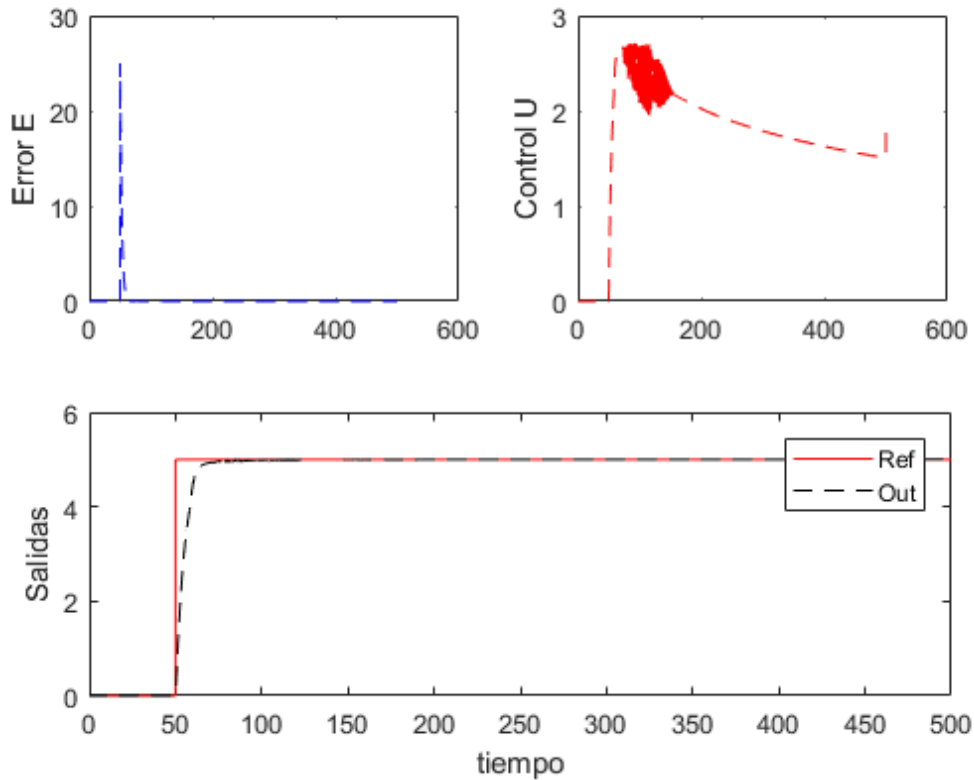


Figura N°47. Respuestas de evolución de pesos sinápticos.

Fuente: Elaboración propia.

La evolución de los pesos (ver Figura N°48) está siendo favorable a pesar que en algún tiempo corto el sistema tenía complicación de adaptarse, además esto es mitigado con los valores de sintonía que posee el sistema de control por interacción adaptativa.

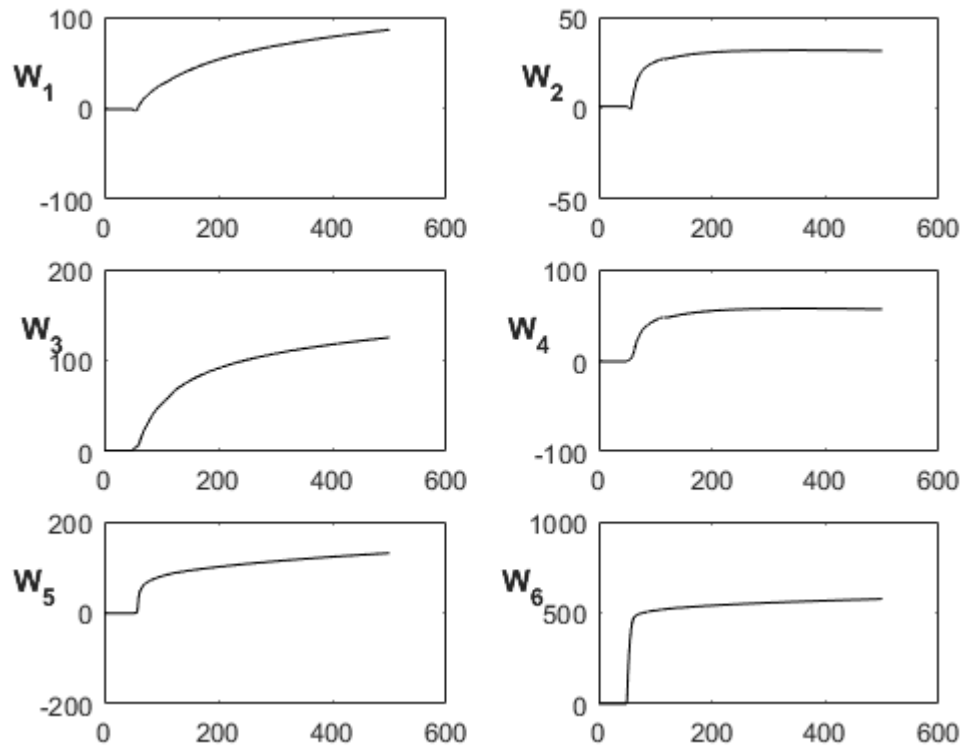


Figura N°48. Respuestas de evolución de pesos sinápticos 3.

Fuente: Elaboración propia.

## VI. DISCUSIÓN DE RESULTADOS

### 6.1 Contrastación y demostración de la hipótesis con los resultados

#### 6.1.1 Contrastación hipótesis 1:

En principio el uso de la tarjeta de adquisición de datos DAQ USB-6009 previamente ha sido configurada en el modo “continuous samples” para unidades de voltaje, de acuerdo con la hoja técnica del fabricante National Instruments. De este modo facilito la adquisición de los datos de pares de entrada y salida esenciales para la obtención del modelo del proceso definida por la función de transferencia:

$$G(s) = \frac{10.45}{s + 5.631}$$

#### 6.1.2 Contrastación hipótesis 2:

El tratamiento adecuado, así como la validación de los datos provenientes de la adquisición de datos nos han permitido resolver el modelo del proceso. En este caso el algoritmo IDGREY expuesto en la tesis ha demostrado ser bastante robusto ya que el modelo del motor DC EMG30, esto se verifica y valida con el porcentaje del foco de predicción del modelo que bordea el 97% lo cual es un valor altamente confiable y exacto.

Comparando con el modelo RLS (Recursivo Least Square) par sistemas de caja negra se obtuvo el modelo con la función de transferencia:

$$G_{rls}(s) = \frac{11.9408}{s + 6.4518}$$

#### Algoritmo 6: Identificación paramétrica ARX

```
% modelo ARX
th1=arx(data,[1 1 1]); % ARX111
% y(k) = a1*y(k-1) + b1*x(k-1) --> na=1 y nb=1
% H(z) = z^-nk b1/(1-a1*z^-1) --> nk=1
thd=tf(th1.b,th1.a,T); % modelo discreto
thc=d2c(thd);
[n,d]=tfdata(thc,'v');
printsys(n,d,'s')
```



Verificando la performance del modelo en lazo abierto para una entrada STEP se valida la performance del modelo IDGREY con el modelo RLS (ver Figura N°49).

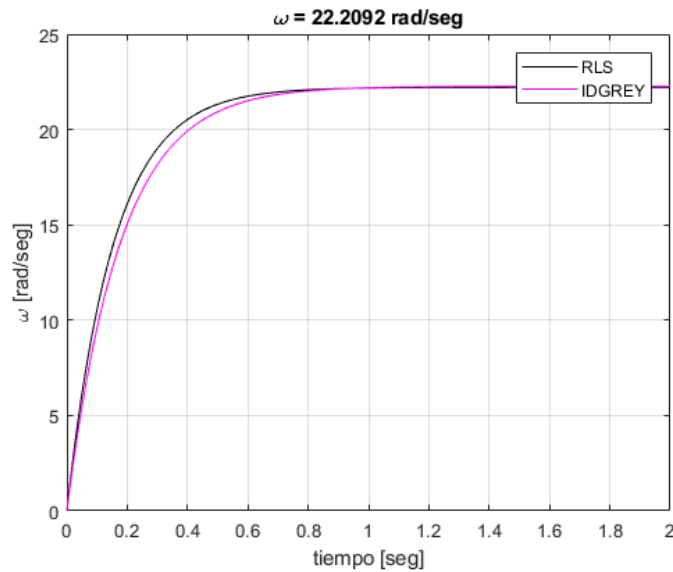


Figura N°49. Verificación de performance de identificación de modelo.

Fuente: Elaboración propia.

### 6.1.3 Contrastación hipótesis 3:

Presentaremos las pruebas que se requieren para que el sistema tenga una mejor adaptación y esto dependerá de la fuerza de control que está relacionada con la ganancia  $\gamma$  y de los pesos  $W$  que son los que requiere la red para subir y/o aumentar según sea el caso que se adapten. En este caso a diferencia del algoritmo convencional del Backpropagation que requiere pesos aleatorios pequeños, en la selección de los pesos en la interacción adaptativa ocurre lo inverso, es decir, se requieren valores de pesos más grandes. A continuación, mostraremos los resultados en tareas de tiempo real.

Las siguientes tablas serán los indicadores de los datos que han sido registrados para cada simulación del control por interacción adaptativa en tiempo real. Posteriormente mostraremos las simulaciones y realizaremos algunos comentarios al respecto. Iniciaremos con los parámetros proporcionados en las Tabla N°11-Tabla N°14.

### 6.3.1.1 Prueba de tiempo real 1

Tabla N°11. Datos para pruebas 1 de tiempo real.

Parámetros	Observaciones
G=0.001;	Ganancia salida ADAPTA función logsig
gamma=20;	
W1=-100;	
W2=100;	
W3=100;	
W4=-100;	
W5=-100;	
W6=100;	

En la Figura N°50 (superior) se muestran la evolución de la señal de referencia (línea blanca) salida (línea verde) y error (línea roja). Mientras en la Figura N°50 (inferior) se muestra la señal de control. Se observa que la respuesta no se adapta rápido debido a que la señal de control no tiene el suficiente voltaje para que el motor DC EMG30 realice el control de velocidad más rápido. La función que se está trabajando las simulaciones en la sigmoide logística.

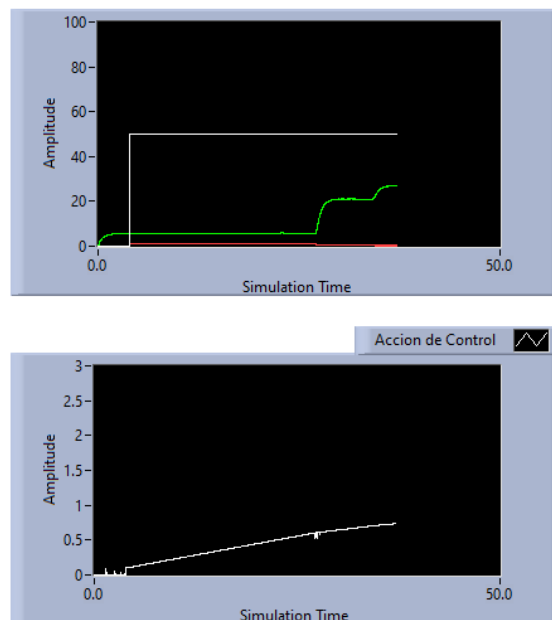


Figura N°50. Front panel de las señales en pruebas 1 de tiempo real.

Fuente: Elaboración propia.

### 6.3.1.2 Prueba de tiempo real 2

Tabla N°12. Datos para pruebas 2 de tiempo real.

Parámetros	Observaciones
G=0.001;	Ganancia salida ADAPTA función logsig
gamma=200;	
W1=-100;	
W2=100;	
W3=100;	
W4=-100;	
W5=-100;	
W6=100;	

En la Figura N°51 (superior) se muestran la evolución más rápida en la señal de salida, mientras que la señal de error decrece. Mientras en la Figura N°51 (inferior) se muestra la señal de control que posee una actuación que hace que el sistema se adapta rápido hacia la velocidad de 50RPM.

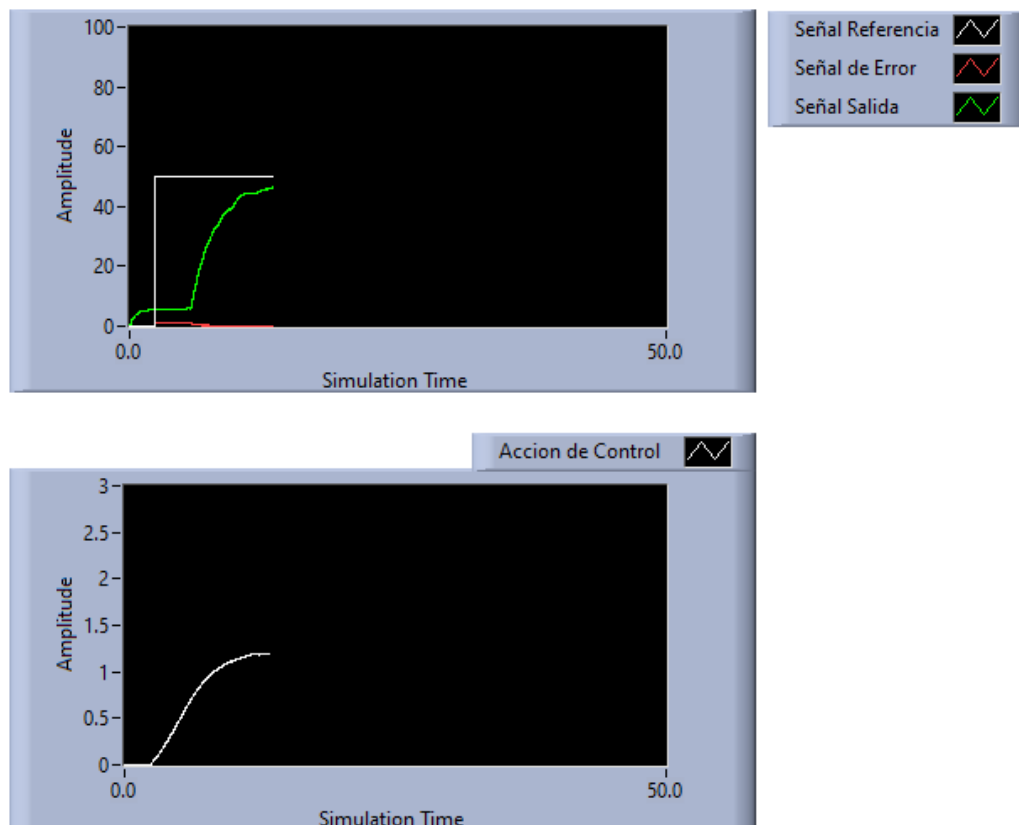


Figura N°51. Front panel de las señales en pruebas 2 de tiempo real.

Fuente: Elaboración propia.

### 6.3.1.3 Prueba de tiempo real 3

Tabla N°13. Datos para pruebas 3 de tiempo real.

Parámetros	Observaciones
G=0.001; gamma=250; W1=-100; W2=100; W3=100; W4=-100; W5=-100; W6=100;	Ganancia salida ADAPTA función gain

En la Figura N°52 (superior) se muestran la evolución más rápida ya que se le ha proporcionado un  $\gamma = 250$  este parámetro hace que el sistema se adapte mejor. Mientras en la Figura N°52 (inferior) se muestra la señal de control que posee una actuación que hace que el sistema se adapte rápido hacia la velocidad de 50RPM. Con ello verificamos el error en centésimas de error en el estado estacionario, verificando con ello la performance.

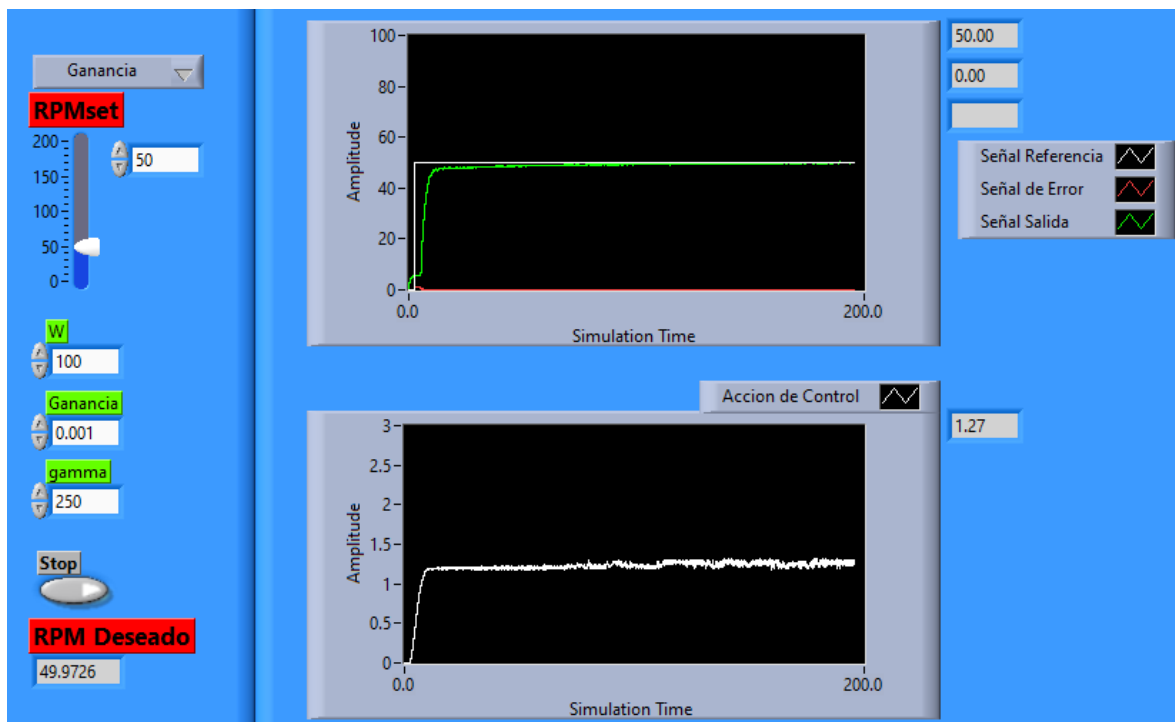


Figura N°52. Front panel de las señales en pruebas 3 de tiempo real.

Fuente: Elaboración propia.

### 6.3.1.4 Prueba de tiempo real 4

Tabla N°14. Datos para pruebas 4 de tiempo real.

Parámetros	Observaciones
G=0.001; gamma=300; W1=-50; W2=50; W3=50; W4=-50; W5=-50; W6=50;	Ganancia salida ADAPTA función gain

En la Figura N°53 (superior) se muestran la evolución más rápida ya que se le ha proporcionado un  $\gamma = 300$  este parámetro hace que el sistema se adapte mejor. Mientras en la Figura N°53 (inferior) se muestra la señal de control que posee una actuación que hace que el sistema se adapte rápido hacia las diferentes velocidades de referencia (200, 100, 50) RPM. Se verifica el control en tiempo real para referencias arbitrarias.

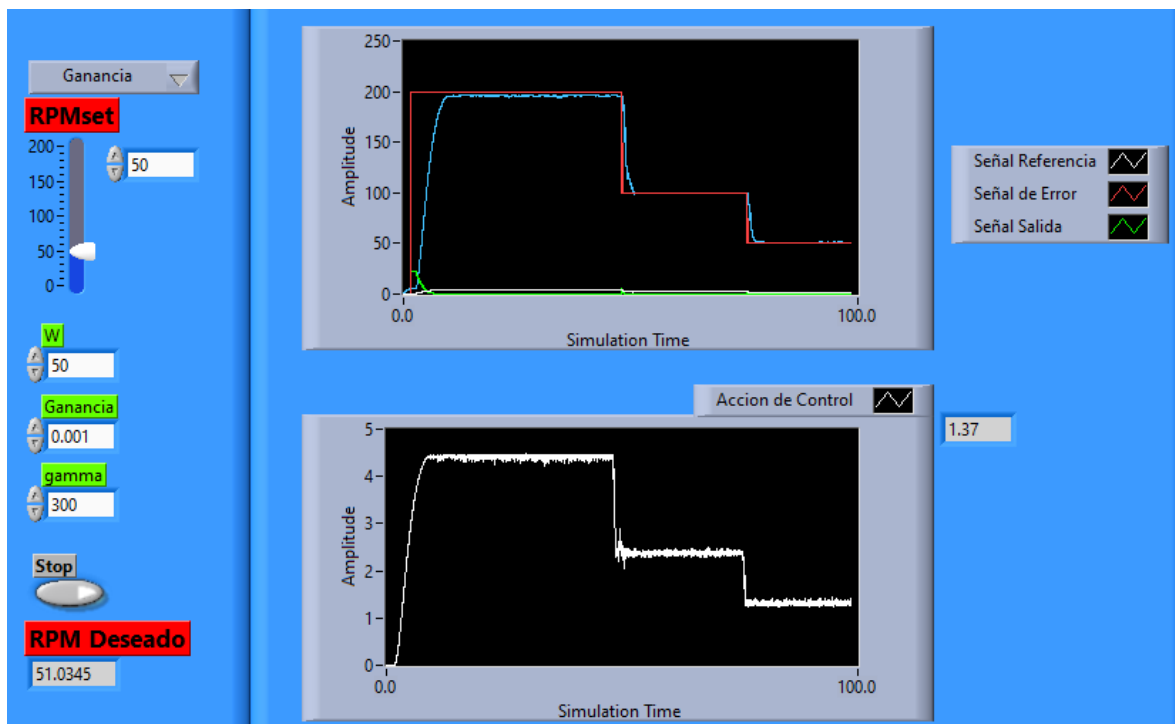


Figura N°53. Front panel de las señales en pruebas 4 de tiempo real.

Fuente: Elaboración propia.

## 6.2 Contratación de la hipótesis con otros estudios similares

En este caso vamos a contrastar el resultado obtenido para el modelo de nuestro motor DC EMG30 con los parámetros obtenidos en el Paper (Modeling and simulation of the EMG30 Geared motor with encoder resorting to SimTwo: The official Robot@Factory Simulator, 2011). En donde presenta sus resultados experimentales, definidos en la Tabla N°08.

Tabla N°15. Parámetros del Motor DC EMG30.

Notación	Termino	Unidades	Valor
$L_a$	Inductancia de armadura	$H$	$3.4 \times 10^{-3}$
$R_a$	Resistencia de armadura	$\Omega$	7.101
$J$	Inercia de motor	$Kg - m^2$	0.00567
$B$	Fricción viscosa	$Nm/rad/s$	0.000931
$K_t$	Constante de motor	$Nm/A$	0.509
$K_b$	Constante contraelectromotriz	$V/rad/s$	0.509

La Figura N°54 nos muestra los resultados de ambos modelos estimados, hay que precisar que los motores DC EMG30 por ser de bajo costo poseen diferentes valores de RPM, no es uniforme en todos, por ello se aprecia un ligero error en el resultado de las simulaciones sin embargo ambos casos nos dan una buena aproximación a 22.6rad/s.

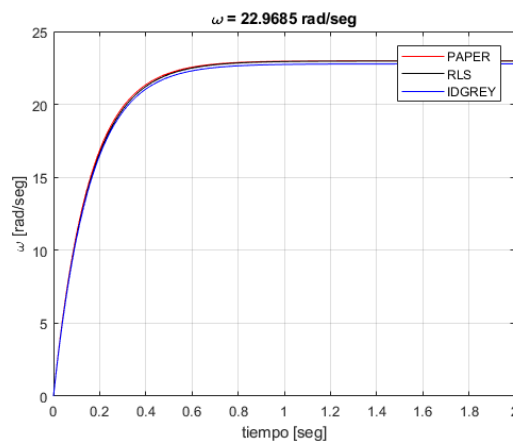


Figura N°54. Respuestas comparativas del modelo del motor DC EMG30.

Fuente: Elaboración propia.

El programa MATLAB nos devuelve los siguientes resultados

- Caso Paper:  $w[\text{rad/s}] = 22.989$  y Caso Tesis:  $w[\text{rad/s}] = 22.968$

### 6.3 Responsabilidad ética de acuerdo con los reglamentos vigentes

Yo, Ricardo Raúl Rodríguez Bustinza, doy fe que mi tesis de doctorado titulada "**ESTIMACIÓN DEL MODELO IDGREY Y CONTROL DE VELOCIDAD POR INTERACCIÓN ADAPTATIVA APLICADO A UN MOTOR DC EMG30**", es de mi autoría, y su aplicación en la estimación de parámetros de caja gris y su control por interacción adaptativa en base a una red neuronal para controlar la velocidad angular de un motor DC EMG30 es fácilmente escalable a motores de mayores dimensiones bajo los principios metodológicos del trabajo de tesis que estamos presentando.

Finalmente, faculto a que la información contenida en la tesis, sea utilizada por otro investigador e ir ampliando a otros procesos, siempre y cuando haga la referencia respectiva al trabajo de tesis.

Dando fe de lo declarado, firmo.



Ricardo Raúl Rodríguez Bustinza

DNI: 07543266

Callao, agosto del 2021

## CONCLUSIONES

De la discusión de resultados expuestos en los Capítulos IV y V, podemos concluir lo siguiente:

- Los dispositivos electrónicos, como es el módulo del motor DC EMG30 en conjunto con la adecuada configuración de la tarjeta de adquisición de datos, nos han permitido interactuar con la PC en la adquisición de los datos de entrada y salida del proceso bajo la plataforma de LabVIEW. De esa manera nos permite validar los datos para el desarrollo del algoritmo de estimación.
- El tratamiento de los datos y selección de la estructura del modelo de primer orden, ha sido validado tal como lo demuestra el desarrollo del algoritmo de caja gris mediante el Toolbox IDGREY de MATLAB encontrando una predicción del error del 97%. Este es un indicador que se adecua bastante bien a la exactitud del modelo, por ende, nos permite validar la obtención del modelo del sistema de control de velocidad del motor EMG30.
- El desarrollo del algoritmo de control por interacción adaptativa aplicado a una red MLP, responde a distintos tipos de consignas arbitrarias, de ese modo la ley de control, impuesta es capaz de llevar al sistema de adaptación a un error en estado estacionario nulo con mínimo sobre impulso y de rápida respuesta y por ello la acción de control impuesta es validada.



## **RECOMENDACIONES**

En primera instancia sería recomendable diseñar un módulo específicamente para el control de posición del motor DC con características similares al del motor DC EMG30 para aplicaciones de la robótica. De esta manera se fomenta y se motiva el desarrollo de proyectos con sistema mecatrónicos.

Asimismo, sería recomendable aplicar diferentes algoritmos en el área de la ingeniería de control no lineal, control avanzado y controladores inteligentes con IoT (Internet of Things).

## BIBLIOGRAFÍA

*A Neural Network Controller by Adaptive Interaction.* **GEORGE SAIKALIS, FENG LIN. 2001.** Michigan : Departament of Electrical and Computer Engineering, 2001. Vol1.

*Adaptive Control of a Class of Nonlinear Discrete-Time Systems Using Neural Networks.* **FU-CHUANG CHEN, AND HASSAN K. KHA. 1995.** USA : JEE Transactions on Control System, 1995. VOL. 40, NO. 5.

*Adaptive interaction and its application to neural networks an Intelligent Devices.* **ROBERT D. BRANDT A, FENG LIN B. 2001.** USA : IEEE , 2001. Vol 1.

*An introduction to linear regression and correlation.* **EDWARDS, ALLEN LOUIS. 1984.** USA : W H Freeman & Co; 2nd edition, 1984.

*Control Avanzado Diseño y Aplicaciones en Tiempo Real.* **ARTURO, ROJAS. 2001.** Lima : Independiente, 2001. SN.

*Control System Analysis and Design.* **NAGLE, PHILLIPS -. 2015.** USA : Ed. Prentice Hall. 4th edition, 2015. 978-0-13-293831-0.

*Control System Toolbox.* **MATHWORKS. 2017.** USA : Mathworks, 2017. SN.

*Diseño de un Controlador PID Con Auto sintonía Basado En Un Modelo De Redes Neuronales Dinámicas y Control Adaptativo.* **CARDOZO GALVEZ, ERICK OCTAVIO. 2018.** Lima : Universidad Ricardo Palma, 2018. SN.

**ESFANDIARI, RAMIN S., AND LU, BEI. 2018.** *Modeling and Analysis of Dynamic Systems.* New York : Taylor & Francis, CRC Press, 2018. 978-1-4665-7493-9.

*Feedback Control of Dynamics System.* **EMAMI-NAEINI, FRANKLIN – POWELL - . 2019.** USA : Addison Wesley. 8th edition, 2019.

*Grey Box Identification and Adaptive Control in a Water Level System.* **R. RODRIGUEZ BUSTINZA, V. PAREDES, G. NOA, LUIS TRUCIO. 2019.** Francia : 8th International Conference on Mechatronics and Control Engineering IOP Conf., 2019. IOP Publishing doi:10.1088/1757-899X/707/1/012003.

*Identification and Control of Dynamical Systems using Neural Networks.* **PARTHASARATHY, NARENDRA AND K. 1990.** USA : IEEE Transactions on Neural Networks, 1990. Vol. 1, pp 1-27.

*Modeling and simulation of the EMG30 Geared motor with encoder resorting to SimTwo: The official Robot@Factory Simulator.* **JOSÉ GONÇALVES, JOSÉ LIMA, PAULO J. COSTA AND A. PAULO MOREIRA. 2011.** Portugal : NESC TEC, 2011. SN.

*Optimization of Dynamical Systems Containing Neural Networks.* **KUMPATI S. NARENDRA, AND KANNAN PARTHASARATHY. 1991.** USA : IEEE Transaction on Neural Networks, 1991. VOL. 2, NO. 2.

*Self-Tuning of PID Controllers by Adaptive Interaction.* **FENG LIN, ROBERT D. BRANDT, AND GEORGE SAIKALIS. 2000.** Chicago : Proceedings of the American Control Conference, 2000. Vol 3.

*USER GUIDE NI USB-6008/6009 Bus-Powered Multifunction DAQ USB Device.* **Instruments, National. 2016.** USA : NI, 2016. SN.

# **ANEXO A**

## **Matriz de Consistencia**

Título: “Estimación del Modelo IDGREY y Control de Velocidad por Interacción Adaptativa Aplicado a un Motor DC EMG30”

PROBLEMA	OBJETIVOS	HIPOTESIS	VARIABLES	INDICADORES	MÈTODO Y DISEÑO
<p><b>Problema General:</b></p> <p>¿De qué manera la incorporación de un algoritmo por interacción adaptativa aplicado a una red MLP y el desarrollo del modelo IDGREY contribuirán a la mejora de la performance del control de la velocidad del motor DC?</p> <p><b>Problemas Específicos:</b></p> <p><b>Problema específico 1:</b> ¿Cómo influye la obtención del modelo experimental del motor en el desarrollo de la actuación del control de la velocidad?</p> <p><b>Problema específico 2:</b> ¿Cómo afecta la estimación paramétrica en la elección del modelo del motor cuando se eligen modelos de primer orden y segundo orden para el control de velocidad?</p> <p><b>Problema específico 3:</b> ¿De qué manera se mide la performance del sistema de control neuronal por interacción adaptativa??</p>	<p><b>Objetivo General:</b></p> <p>Estimar el modelo IDGREY y realizar el control de velocidad por interacción adaptativa aplicado a un motor DC EMG30.</p> <p><b>Objetivos Específicos:</b></p> <p>Objetivo Específico 1: Implementar un sistema de adquisición de datos en base a la plataforma LabVIEW y MATLAB para estimar el modelo IDGREY.</p> <p>Objetivo Específico 2: Encontrar el modelo del proceso desarrollando funciones de sistemas espacio estado empleando las herramientas del programa MATLAB.</p> <p>Objetivo Específico 3: Desarrollo del algoritmo de control por interacción adaptativa para una red neuronal multicapa y verificación de la acción de control.</p>	<p><b>Hipótesis General:</b></p> <p>El análisis del diseño del control por interacción adaptativa aplicado a una red MLP permitirá un adecuado control de velocidad del motor DC.</p> <p><b>Hipótesis Específicas</b></p> <p>Hipótesis específica 1: La adecuada configuración del sistema de adquisición de datos, nos permitirá adquirir los datos del proceso.</p> <p>Hipótesis específica 2: El tratamiento adecuado de los datos provenientes de la adquisición de datos permitirá resolver el modelo del proceso.</p> <p>Hipótesis específica 3: El desarrollo del algoritmo de control por interacción adaptativa aplicado a una red MLP, permitirá verificar la performance del sistema de control.</p>	<p><b>Variable Independiente</b></p> <p>Es la variable de acción de control que está determinada por la tensión de entrada al motor de corriente continua que proporciona el torque y se denota por la variable (u).</p> <p><b>Variable Dependiente</b></p> <p>El sistema cuenta con un grado de libertad representados por la velocidad angular <math>\dot{q}_1</math> desarrollada en el eje del motor DC y cuya medida será leída en RPM (revoluciones por minuto).</p>	<p>X.1. Obtención de la ley de control por interacción adaptativa.</p> <p>X.2. Medida de la acción de control</p> <p>X.3. Limitación de la acción de control</p> <p>Y.1 Medida del ángulo en radianes</p> <p>Y.2. Medida de la velocidad angular en RPM.</p>	<p><b>Método</b></p> <p><b>Temporal.</b></p> <p>El estudio es del tipo transversal, su ejecución se inicia el 3 de abril del año 2019 y su posible término el 30 de diciembre del 2020.</p> <p><b>Espacial.</b></p> <p>El estudio se realizará en los ambientes de la facultad de Ingeniería Eléctrica y Electrónica de la Universidad Nacional del Callao.</p>

# **ANEXO B**

## **Programa en MATLAB**

## Programa MATLAB 1

```
%-----  
% ALGORITMO DE CONTROL ADAPTATIVO NEURONAL  
%-----  
% planta identificada  
Gs=tf(10.45,[1 5.631]);  
fs=100;  
dt=1/fs;  
Gz=c2d(Gs,dt);  
[b,a]=tfdata(Gz,'v');  
present(Gz)  
a1=a(2);  
b1=b(2);  
% referencia  
f=1;  
T=1/f;  
n=10;  
w=2*pi*f;  
t=0:dt:n*T;  
amp=1;  
u=amp*sin(2*pi*f*t);  
% pesos iniciales  
wo=1;  
w1(2)=-wo; w2(2)=wo; w3(2)=wo;  
w4(3)=-wo; w5(2)=-wo; w6(2)=wo;  
dw1(2)=0; dw2(2)=0; dw3(2)=0;  
dw4(2)=0; dw5(2)=0; dw6(2)=0;  
x=zeros(1,1);  
y=zeros(1,1);  
e=zeros(1,1);  
gain=0.5;  
N=n*T/dt+1;  
gamma=200;  
for n=2:N  
    y(n)=-a1*y(n-1)+b1*x(n-1);  
    e(n)=u(n)-y(n);  
    p1=e(n)*w1(n)+e(n-1)*w2(n);  
    p2=e(n)*w3(n)+e(n-1)*w4(n);  
    r1=logsig(p1);  
    r2=logsig(p2);  
    a=r1*w5(n)+r2*w6(n);  
    x(n)=tansig(a);  
    %x(n)=gain*a;  
    if n<N  
        dw6(n+1)=e(n)*r2*gamma;  
        dw5(n+1)=e(n)*r1*gamma;  
        w6(n+1)=w6(n)+0.5*dt*(dw6(n+1)+dw6(n));  
        w5(n+1)=w5(n)+0.5*dt*(dw5(n+1)+dw5(n));  
        dw1(n+1)=e(n)*w5(n+1)*dw5(n+1)*logsig(-p1);  
        dw2(n+1)=e(n-1)*w5(n+1)*dw5(n+1)*logsig(-p1);
```

```

dw3 (n+1)=e (n) *w6 (n+1) *dw6 (n+1) *logsig (-p2);
dw4 (n+1)=e (n-1) *w6 (n+1) *dw6 (n+1) *logsig (-p2);
w1 (n+1)=w1 (n) +0.5*dt*(dw1 (n+1)+dw1 (n));
w2 (n+1)=w2 (n) +0.5*dt*(dw2 (n+1)+dw2 (n));
w3 (n+1)=w3 (n) +0.5*dt*(dw3 (n+1)+dw3 (n));
w4 (n+1)=w4 (n) +0.5*dt*(dw4 (n+1)+dw4 (n));
end
end
e2=e.*e;
k=0:N-1;
figure
plot(k,u,'b')
hold
plot(k,y,'k')
figure
subplot(231), plot(k,w1)
subplot(232), plot(k,w2)
subplot(233), plot(k,w3)
subplot(234), plot(k,w4)
subplot(235), plot(k,w5)
subplot(236), plot(k,w6)

```



**ANEXO C**  
**Hoja Técnica del Motor DC EMG30**

- RB-Dev-40 Devantech 12V, 30:1 Gear Motor w/ Encoder



Figura N°55. Motor DC EMG30

Fuente: <https://www.robot-electronics.co.uk/htm/emg30.htm>

El motor EMG30 es un motor de corriente directa 12v totalmente equipado con codificadores y una caja de engranajes de reducción de 30:1. Es ideal para aplicaciones de robótica, proporciona una efectiva unidad de costo y la retroalimentación para el usuario.

El motor EMG30 (ver Figura N°56) tiene una velocidad nominal de 170 revoluciones que se caracteriza por incluir un encoder o codificador de cuadrante que envía un tren de impulsos cuando gira el eje del motor, permitiendo así que un circuito externo nos proporcione la velocidad real a la que está girando el eje y cuantas vueltas da. El encoder está formado por dos sensores de efecto hall que proporcionan un total de 360 pulsos por cada vuelta completa del rotor. El motor cuenta con condensadores internos de filtro que ayudan a minimizar el ruido y los parásitos generados por el motor al girar a través de las bobinas del motor.



Figura N°56. Motor DC EMG30 con bracket de montaje.

Fuente: <https://www.robot-electronics.co.uk/htm/emg30.htm>

El motor EMG30 suministra un conector JST 6 al final o parte posterior (ver Figura N°57) de aproximadamente 90 mm de cable de serie.

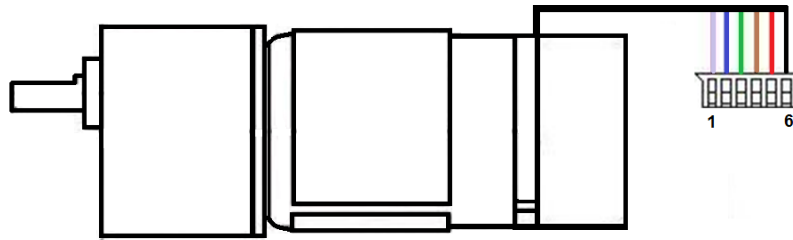


Figura N°57. Esquema de disposición de pines del encoder.

Fuente: Elaboración propia.

Las conexiones son listadas en la Tabla N°16:

Tabla N°16. Pines de conectores del Motor DC EMG30.

Color de alambre	Pin	Conector
Purpura	(1)	Sensor Hall B ( $V_{out}$ )
Azul	(2)	Sensor Hall A ( $V_{out}$ )
Verde	(3)	Sensor Hall ( $GND$ )
Marrón	(4)	Sensor Hall ( $V_{cc}$ )
Rojo	(5)	Motor +
Negro	(6)	Motor -

Las especificaciones técnicas del motor EMG30.

- Tensión nominal de 12VDC
- Fuerza 1.5Kg / cm
- Velocidad nominal 170 rpm
- Consumo típico 530mA
- Velocidad sin carga 216 rpm
- Consumo sin carga en vacío 150mA
- Contador del codificador 360 pulsos por vuelta del eje.
- Eje de 5 mm para casquillos de fijación de 5mm.

Las salidas de los sensores Hall necesitan unas resistencias pull-up de unos 4.7KΩ, esta disposición se muestra en el esquema de la Figura N°58.

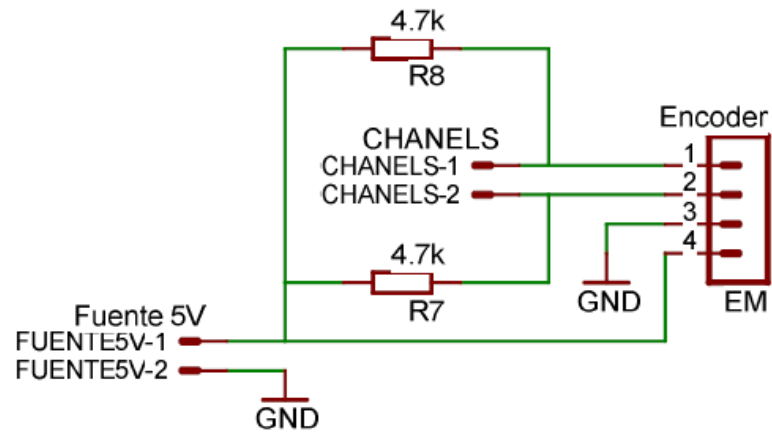


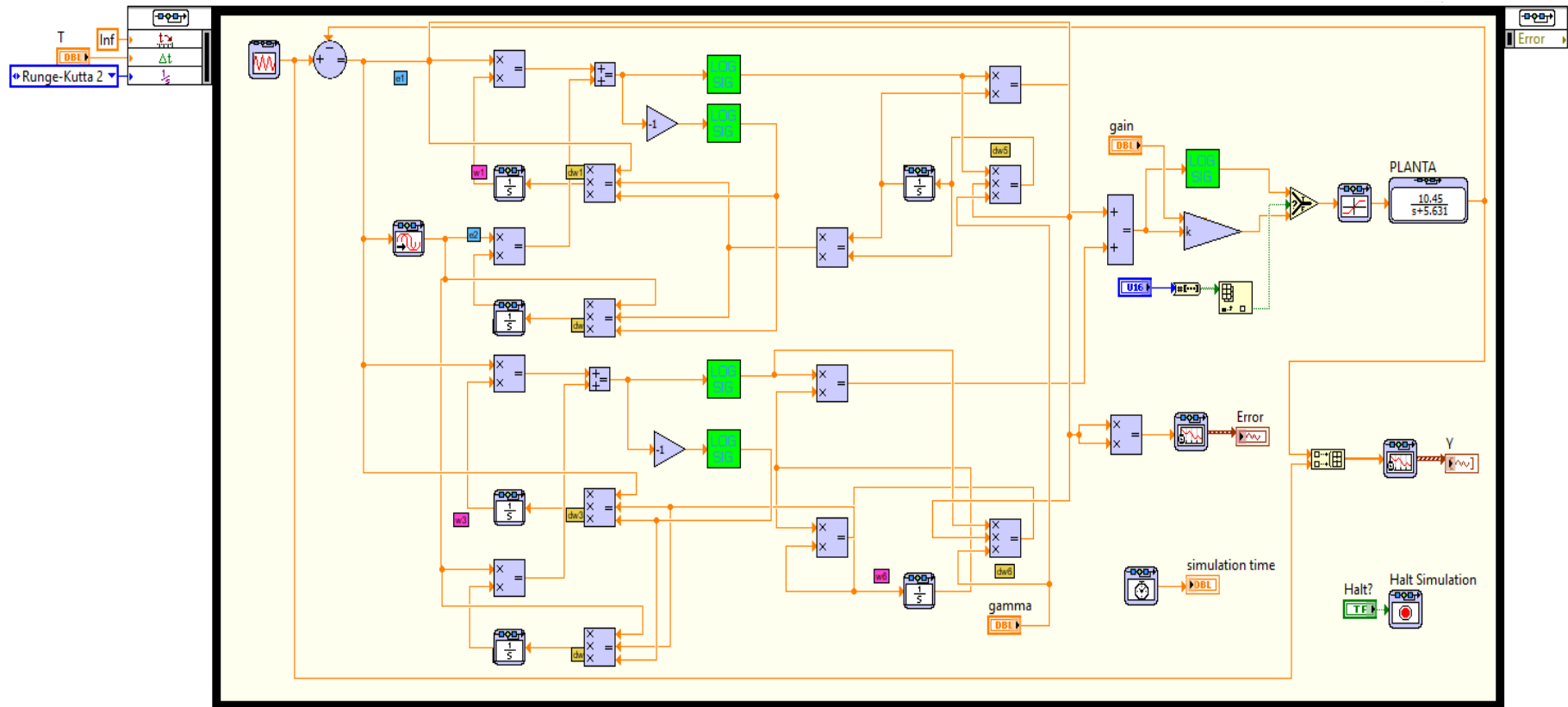
Figura N°58. Esquema pull-up para salidas del encoder.  
Fuente: Elaboración propia.

# **ANEXO D**

## **PROGRAMAS SIMULINK - LabVIEW**

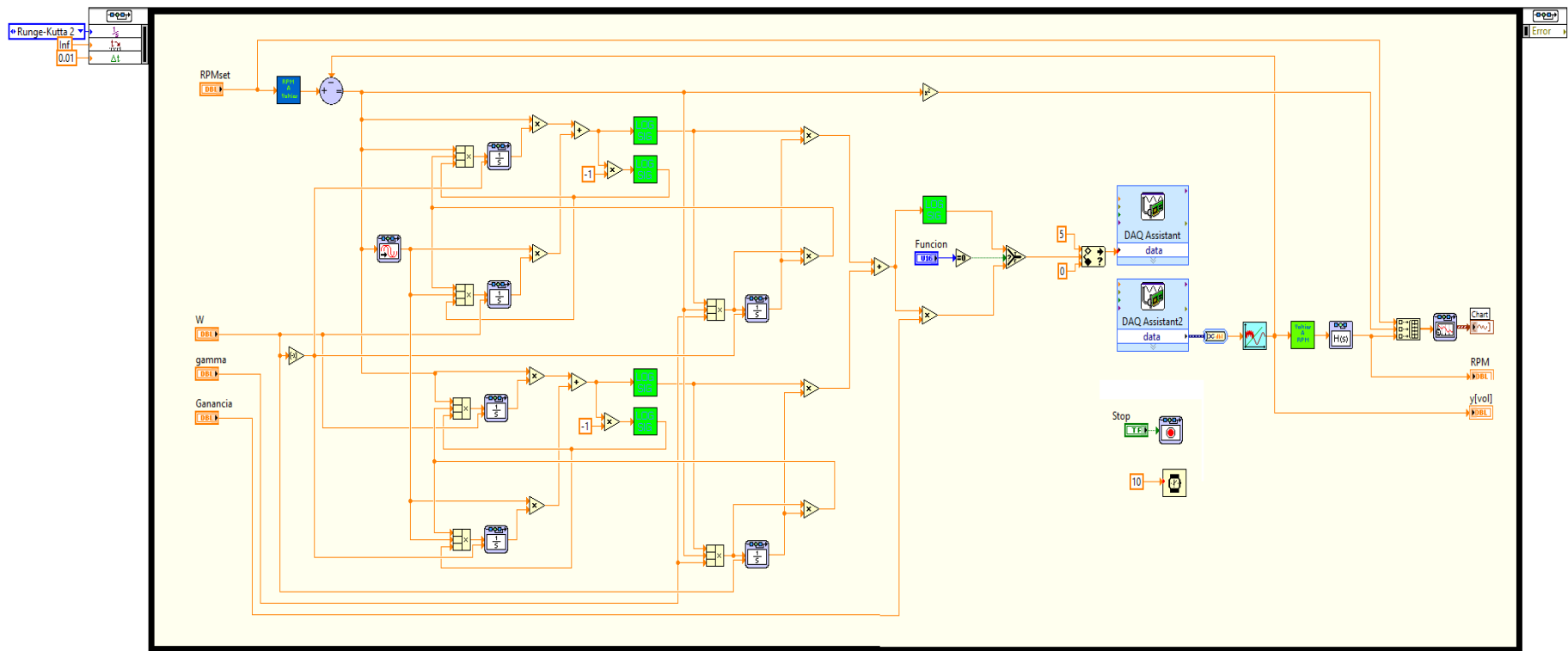


- Diagrama del sistema de control por interacción adaptativa en LabVIEW.



Fuente: Elaboración propia.

- Toolkit de Simulación por interacción adaptativa en LabVIEW DAQ Assistant.



Fuente: Elaboración propia.