



Universidad Católica de Santa María

Facultad de Ciencias e Ingenierías Físicas y Formales

Escuela Profesional de Ingeniería Mecánica Eléctrica

y Mecatrónica

**Diseño de control vectorial sensorless tolerante a fallas de sensores para un
motor de inducción**

Tesis presentada por:

Arenas Oviedo, Alvaro Alonso

ORCID: 0009-0006-0103-8168

para optar el Título Profesional de Ingeniero Mecatrónico

Asesor:

Mg. Rivera Acosta, Victor Gonzalo

ORCID: 0009-0008-7731-8478

Arequipa - Perú

2026

UNIVERSIDAD CATÓLICA DE SANTA MARÍA
INGENIERIA MECANICA ELECTRICA Y MECATRONICA
TITULACIÓN CON TESIS
DICTAMEN APROBACIÓN DE BORRADOR

Arequipa, 28 de Abril del 2026

Dictamen: 016190-C-EPIMMEM-2026

Visto el borrador del expediente 016190, presentado por:

2008203911 - ARENAS OVIEDO ALVARO ALONSO

Titulado:

**DISEÑO DE CONTROL VECTORIAL SENSORLESS TOLERANTE A FALLAS DE SENSORES PARA UN
MOTOR DE INDUCCIÓN**

Nuestro dictamen es:

APROBADO

Titulo Profesional/Titulo de Segunda Especialidad/Grado Académico a optar:

INGENIERO MECATRÓNICO

**29569057 - CASTILLO CACERES CESAR PIO
DICTAMINADOR**



**21535915 - DONAYRE CAHUA JESUS MANUEL
DICTAMINADOR**



**29460348 - AREVALO DAZA JORGE LUIS
DICTAMINADOR**



Diseño de control vectorial sensorless tolerante a fallas de sensores para un motor de inducción

INFORME DE ORIGINALIDAD

6%

INDICE DE SIMILITUD

5%

FUENTES DE INTERNET

4%

PUBLICACIONES

0%

TRABAJOS DEL
ESTUDIANTE

FUENTES PRIMARIAS

1

tesis.pucp.edu.pe

Fuente de Internet

2%

2

hdl.handle.net

Fuente de Internet

2%

3

repositorio.utp.edu.co

Fuente de Internet

1%

4

archive.org

Fuente de Internet

1%

5

Arenas Oviedo, Alvaro Alonso. "Desarrollo de control predictivo de torque de motor de inducción aplicado a tracción vehicular", Pontificia Universidad Catolica del Peru (Peru), 2025

Publicación

1%

6

Cordova Claros, Josmell. "Diseno de un sistema de diagnostico y control tolerante de fallas en actuadores para el proceso hidraulico de cuatro tanques acoplados",

1%

DEDICATORIA

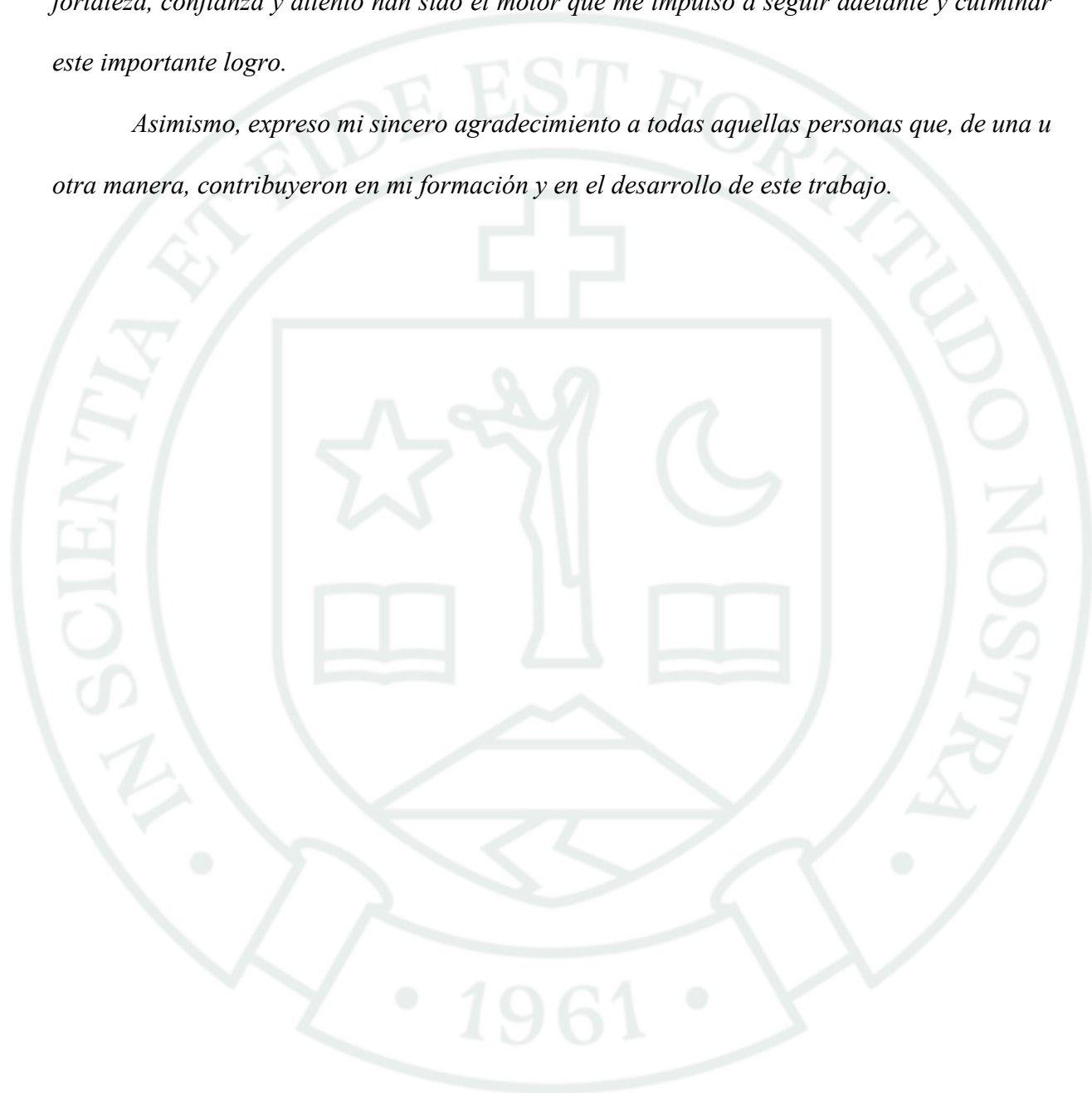
Dedico este trabajo a mis padres y a mi hermano, quienes han sido mi mayor fuente de inspiración y apoyo constante. A ellos, por su esfuerzo, sacrificio y confianza en cada paso que doy.



AGRADECIMIENTOS

Agradezco a mis padres y a mi hermano por haberme acompañado y brindado su apoyo incondicional a lo largo de esta etapa, tanto en lo profesional como en lo personal. Su fortaleza, confianza y aliento han sido el motor que me impulsó a seguir adelante y culminar este importante logro.

Asimismo, expreso mi sincero agradecimiento a todas aquellas personas que, de una u otra manera, contribuyeron en mi formación y en el desarrollo de este trabajo.



RESUMEN

El presente trabajo desarrolla y valida un sistema de control vectorial sensorless para un motor de inducción, integrando técnicas de identificación de parámetros, estimación de variables de estado y tolerancia a fallas. En una primera etapa, se desarrolló un algoritmo de Optimización por Enjambre de Partículas (PSO) para la identificación de parámetros eléctricos del motor utilizando únicamente datos de placa y manufactura. Mediante la comparación de resultados experimentales se validó esta metodología, obteniendo márgenes de error reducidos en el cálculo de resistencias y reactancias.

La estrategia de control de campo orientado (FOC) fue diseñada y evaluada en un entorno de simulación utilizando un enfoque sensorless basado en el Filtro de Kalman Extendido (EKF), lo que permitió estimar la velocidad y el flujo del rotor bajo condiciones dinámicas de operación. Sobre esta base, se desarrolló un sistema de control tolerante a fallas (FTC) apoyado en el EKF, capaz de identificar pérdidas de señal y degradaciones paramétricas en los sensores, permitiendo la reconfiguración automática del control mediante software.

Para la validación experimental, se implementó un control vectorial sensorless basado en un Observador de Estado de Luenberger extendido (LSO) sobre un microcontrolador STM32G431RB, utilizando el entorno STM32 Motor Control Workbench. Los resultados experimentales confirmaron la viabilidad de ejecutar control vectorial sensorless en tiempo real, logrando una operación estable y un seguimiento adecuado de las referencias de velocidad y corriente sin el uso de sensores mecánicos.

Palabras clave: Motor de Inducción, Control de Campo Orientado Sensorless, Control Tolerante a Fallas.

ABSTRACT

This work develops and validates a sensorless vector control system for an induction motor, integrating parameter identification techniques, state variable estimation, and fault tolerance. In the first stage, a Particle Swarm Optimization (PSO) algorithm was developed to identify the motor's electrical parameters using only nameplate and manufacturing data. Experimental results validated this methodology, yielding reduced error margins in the calculation of resistances and reactances.

The field-oriented control (FOC) strategy was designed and evaluated in a simulation environment using a sensorless approach based on the Extended Kalman Filter (EKF). This allowed for the estimation of rotor speed and flux under dynamic operating conditions. Based on this, a fault-tolerant control (FTC) system was developed, supported by the EKF, capable of identifying signal losses and parametric degradations in the sensors, enabling automatic control reconfiguration via software.

For experimental validation, a sensorless vector control system based on an extended Luenberger State Observer (LSO) was implemented on an STM32G431RB microcontroller, using the STM32 Motor Control Workbench environment. The experimental results confirmed the feasibility of real-time sensorless vector control, achieving stable operation and adequate tracking of speed and current references without the use of mechanical sensors.

Keywords: Induction Motor, Sensorless Field Oriented Control, Fault Tolerant Control

ÍNDICE GENERAL

DEDICATORIA

AGRADECIMIENTOS

RESUMEN

ABSTRACT

INTRODUCCIÓN 1

CAPÍTULO I: PLANTEAMIENTO DEL PROBLEMA 3

1.1 Descripción del problema de investigación 3

1.1.1 Enunciado de la investigación 3

1.1.2 Interrogantes de la investigación 3

1.1.2.1 Interrogante principal 3

1.1.2.2 Interrogantes específicas 3

1.1.3 Descripción del problema 4

1.2 Área de conocimiento 6

1.3 Antecedentes 6

1.4 Objetivos 9

1.4.1 Objetivo General 9

1.4.2 Objetivos Específicos 9

1.5 Justificación 10

1.6 Alcance 11

1.7 Limitaciones 12

CAPÍTULO II: MARCO TEÓRICO 13

2.1 Motor de inducción 13

2.2 Accionamientos eléctricos de motor de inducción 17

2.3 Determinación de parámetros de motor de inducción 22

2.3.1 Introducción de identificación de parámetro por PSO 22

2.3.2 Estructura del PSO 23

2.3.3 Desarrollo general del algoritmo PSO 24

2.3.4 Clases de entorno en PSO 28

2.3.5 Tipos de PSO 28

2.4 Introducción al control vectorial de motor de inducción 29

2.5 Filtro de Kalman extendido 30

2.5.1 Introducción al Filtro de Kalman 30

2.5.2 Conceptos importantes 32

2.5.3 Filtro de Kalman discreto 38

2.5.4	Algoritmo del Filtro de Kalman.....	44
2.5.5	Filtro de Kalman extendido	46
2.6	Control Tolerante a Fallas.....	49
2.6.1	Introducción al Control Tolerante a Fallas	49
2.6.2	Diagnóstico de fallas basado en modelos	50
2.6.3	Técnicas de diagnósticos de fallas	50
2.6.4	Métodos de FDD basado en modelo	51
2.6.4.1	Espacio de Paridad	52
2.6.4.2	Estimación de parámetros	52
2.6.4.3	Observadores.....	53
2.6.5	Clasificación de sistema de control tolerante a fallas	53
2.6.5.1	Sistemas de control tolerante a fallos pasivos (PFTC)	54
2.6.5.2	Sistemas de control tolerante a fallos activos (AFTC)	54
2.6.6	Clasificación de fallas	55
2.7	Clasificación de fallas en variadores de velocidad	56
CAPÍTULO III: IDENTIFICACIÓN DE PARÁMETROS DE MOTOR DE INDUCCIÓN.....		58
3.1	Modelado de motor de inducción	58
3.1.1	Ecuaciones del sistema en el marco de referencia ABC.....	58
3.1.2	Ecuaciones eléctricas	59
3.1.3	Ecuaciones mecánicas.....	62
3.1.4	Ecuaciones vectoriales espaciales.....	63
3.1.5	Transformada de Clarke.....	63
3.1.6	Transformada de Park	67
3.1.7	Discretización de modelo de motor de inducción.....	74
3.1.8	Torque electromagnético	76
3.2	Determinación de parámetros de motor de inducción	77
3.3	Pruebas experimentales.....	79
3.3.1	Prueba de corriente continua para la evaluación de la resistencia del estator 79	
3.3.2	Prueba en vacío	80
3.3.3	Prueba del rotor bloqueado	82
3.4	Identificación de parámetros de motor de inducción usando PSO	88
3.5	Determinación de parámetros de M.I. usando datos de manufactura	88
3.5.1	Definición del problema	88

3.5.2	Resultados	94
CAPÍTULO IV: CONTROL VECTORIAL SENSORLESS DE MOTOR DE INDUCCIÓN		
		97
4.1.	Principios del control vectorial de motores AC	98
4.2.	Algoritmo de control vectorial de motores AC.....	99
4.3.	Control a flujo debilitado	103
4.3.1	Método convencional de control a flujo debilitado	103
4.4.	Controlador de velocidad.....	106
4.4.1	Mecanismo Anti-Windup en el Controlador PI	108
4.5	Control de corrientes.....	112
4.6	Desacoplamiento de Ejes d-q mediante Compensación por Feedforward.....	117
4.7	Limitadores de corrientes y voltajes	118
4.8	Modulación por ancho de pulsos de vector espacial (SVPWM)	120
4.8.1	Voltaje de fase.....	121
4.8.2	Implementación de la modulación SVPWM	125
4.9	Diseño del Filtro de Kalman extendido en el modelo del motor	127
4.9.1	Determinación de las matrices de covarianza de ruido y estado	129
4.9.2	Implementación del algoritmo EKF.....	130
4.9.3	Inicialización de vectores de estado y matrices de covarianza	130
4.9.4	Predicción de la variable de estado	131
4.9.5	Estimación de la Matriz de Covarianza de Predicción	131
4.9.6	Cálculo de la Ganancia de Kalman	132
4.9.7	Estimación del vector de estado.....	132
4.9.8	Corrección de la matriz de covarianza de predicción para el siguiente bucle	132
4.10	Resultados de simulación.....	133
4.10.1	Resultados de simulación de control vectorial.....	133
4.10.2	Respuesta en control por debilitamiento de flujo	137
4.10.3	Resultado de simulación de control vectorial sensorless	139
CAPÍTULO V: DISEÑO DE CONTROL TOLERANTE A FALLAS DE SENSORES DE VELOCIDAD Y CORRIENTES DE CONTROL VECTORIAL DE MOTOR DE INDUCCIÓN		
		146
5.1	Algoritmo de Diagnóstico, Aislamiento y Reconfiguración.....	146
5.1.1	Detección y Corrección en Sensores de Corriente <i>isa</i> , <i>isc</i>	147
5.1.1.1	Fallas por Offset (Desplazamiento de DC).....	147
5.1.1.2	Fallas de Escalamiento (Ganancia)	147

5.1.1.3	Fallas de Circuito Abierto	148
5.1.2	Diagnóstico y Corrección en Sensor de Velocidad (ωm)	148
5.1.3	Lógica de Reconfiguración del Sistema	148
5.2	Implementación de diagnóstico, aislamiento y reconfiguración de fallas	149
5.2.1	Falla de Circuito Abierto en Sensores (Detección Inmediata).....	149
5.2.2	Error de Medición por Escalamiento (Falla de Ganancia).....	149
5.2.3	Discriminación Robusta: Índice de Ceros (Mn)	151
5.2.4	Error de Medición por Offset (Desplazamiento DC).....	152
5.2.5	Estrategia de Reconfiguración (FTC)	153
5.2.5.1	Reconfiguración de Corrientes (Modelo Ideal)	153
5.2.5.2	Reconfiguración de Velocidad.....	154
5.3	Arquitectura del Sistema Propuesto.....	155
5.4	Resultados de simulación a diferente tipo de fallas	156
5.4.1	Falla por circuito abierto de sensor velocidad ωm	157
5.4.2	Falla por circuito abierto de corriente de fase isa	158
5.4.3	Falla de ganancia de corriente de fase isc	160
5.4.4	Falla por offset de corriente de fase isa	162
CAPÍTULO VI: PROPUESTA DE IMPLEMENTACIÓN		
EXPERIMENTAL DE CONTROL VECTORIAL SENSORLESS		166
6.1	Introducción y Plataforma Experimental	166
6.2	Arquitectura de Control Digital	166
6.2.1	Características del Núcleo ARM Cortex-M4.....	167
6.2.2	Arquitectura de Buses y Eficiencia de Ejecución	168
6.2.3	Características Específicas del STM32G4 y Plataforma de Desarrollo.....	169
6.3	Etapas de potencia: STEVAL-IHM023V3.....	171
6.4	Placa de expansión X-NUCLEO-IHM09M1	173
6.5	Configuración de la Modulación Vectorial y Adquisición de Señales	174
6.5.1	Modulación en Modo Alineado al Centro	174
6.5.2	Inserción de Tiempo Muerto (Dead Time)	175
6.5.3	Sincronización del ADC	176
6.6	Configuración de Interfaces Físicas y Conversión Analógica.....	176
6.7	Entorno de Desarrollo y Flujo de Trabajo	177
6.8	Estrategia de Control vectorial Sensorless.....	178
6.8.1	Observador de Luenberger adaptativo	180
6.8.2	Estructura y Dinámica del Observador	180

6.8.3	Estimación de Velocidad y Estabilidad (Lyapunov)	181
6.9	Procedimiento de implementación de control vectorial sensorless	185
6.9.1	Configuración de Sensorless FOC en MC SDK Workbench	185
6.9.2	Configuración de periféricos en STM32CubeMx.....	187
6.9.3	Compilación, Carga y Depuración en STM32CubeIDE.....	190
6.10	Validación experimental y resultados	191
6.10.1	Verificación de Señales de Control (PWM)	191
6.10.2	Integración del Banco de Pruebas.....	191
6.10.3	Validación Funcional con STM32 MotorPilot	193
6.11	Costos de implementación del proyecto	196
6.11.1	Descripción y Verificación de la Tabla de Costos.....	196
6.11.2	Estructura del Presupuesto por Subsistema	197
6.11.3	Análisis Costo-Beneficio	198
CONCLUSIONES.....		200
RECOMENDACIONES		202
REFERENCIAS BIBLIOGRÁFICAS.....		203

ÍNDICE DE TABLAS

Tabla 1 Vectores de voltaje de 2L-VSI.....	21
Tabla 2. Parámetros de placa y catalogo WEG.....	78
Tabla 3. Registro de valores medidos de prueba de vacío	82
Tabla 4. Registro de valores medidos de prueba de rotor bloqueado	85
Tabla 5. Parámetros estimados.....	85
Tabla 6. Rangos de valores de entrada.....	94
Tabla 7. Análisis comparativo entre valores	96
Tabla 8. Tensiones de fase en coordenadas $\alpha\beta$	123
Tabla 9. Análisis en todos los sectores	127
Tabla 10. Tiempos SVPWM simétrico (Forma reducida)	127
Tabla 11. Características del STM32G431RB (STMicroelectronics, 2019)	170
Tabla 12. Características de STEVAL-IHM023V3	172
Tabla 13. Presupuesto de implementación del banco de pruebas experimental	197
Tabla 14. Matriz de consistencia.....	206
Tabla 15. Matriz de operacionalización de variables.....	208

ÍNDICE DE FIGURAS

Figura 1. Vista esquemática de motor de inducción (Olvera, 2022).....	14
Figura 2. Circuito equivalente por fase de un motor de inducción (Chapman, 2012)	16
Figura 3. Estructura general de un accionamiento eléctrico (Košťál,, 2021).....	17
Figura 4. Estructura de un accionamiento común con máquina de inducción – Elaboración propia ..	18
Figura 5. Topología del convertidor de potencia conectado al estator del motor de inducción (Dosreis, 2021).....	19
Figura 6. Distribución espacial de todos los vectores de tensión admisibles de un 2L-VSI (Le Fosse, 2021)	21
Figura 7. Diagrama de flujo del algoritmo PSO – Elaboración propia.....	26
Figura 8. Representación del movimiento de una partícula (Chauhan & Suganthan, 2025)	27
Figura 9. Modelos de comunicación básicos en PSO (Chauhan & Suganthan, 2025).	29
Figura 10. Esquema de control de campo orientado (FOC) (Liu, Chau, Lee, & Song, 2021).....	30
Figura 11. Densidades de probabilidad Normal para distintos μ y σ^2 - Elaboración propia.....	35
Figura 12. Representación de PDF Gaussiana Multivariable: Superficie de densidad y proyecciones marginales con elipsoide de contorno (Grewal & Andrews, 2014).	37
Figura 13. Diagrama de bloques de implementación de Filtro de Kalman.....	38
Figura 14. Algoritmo de Filtro de Kalman – Elaboración propia.....	44
Figura 15. Proceso de predicción y corrección en el Filtro de Kalman (Nadeem, 2020)	45
Figura 16. Linealización de una función no lineal mediante expansión de Taylor de primer orden (Nadeem, 2020).....	47
Figura 17. Clasificación de los métodos de Detección y Diagnóstico de Fallos (Kumar, Velpula, Saiprakash, & Sahoo, 2025).....	50
Figura 18. Sistema de diagnóstico de falla basado en modelos (Córdova Claros, 2017)	52
Figura 19. Clasificación de las estrategias tolerante a fallos – Elaboración propia.....	53
Figura 20. Arquitectura de las estrategias de control tolerante a fallas activo y pasivo (Wu, Sehab, Akrad, & Morel, 2022).....	55
Figura 21. Descripción gráfica de fallas abruptas, incipientes e intermitentes.	56
Figura 22. Condición de falla en VSD (Yepes, Gonzalez-Prieto, Lopez, & Duran, 2022)	57
Figura 23. Circuito equivalente de M.I. conectado en Y (Le Fosse, 2021)	59
Figura 24. Definición del sistema de referencia estacionario $\alpha\beta 0$ (Lyshevski, 2017)	63
Figura 25. Corrientes trifásicas balanceadas representadas en el sistema de referencia estacionario $\alpha\beta 0$ (Geyer, 2016).....	64
Figura 26. Definición del sistema de referencia rotatorio $dq 0$ (Lyshevski, 2017).....	67
Figura 27. Corrientes trifásicas equilibradas representadas en el sistema de referencia giratorio $dq 0$ (Geyer, 2016)	68

Figura 28. Placa de motor WEG de 1 HP IEC W22 Standard Efficiency	79
Figura 29. Circuito de prueba de una prueba de cd de resistencia (Chapman, 2012).	79
Figura 30. Circuito con pérdidas eléctricas (Chapman, 2012).	80
Figura 31. Prueba de rotor bloqueado de un motor de inducción: a) circuito de prueba;b) circuito equivalente del motor.	83
Figura 32. Variación de velocidad a lo largo del tiempo	86
Figura 33. Variación de torque a lo largo del tiempo	86
Figura 34. Variación de corrientes del estator a lo largo del tiempo	87
Figura 35. a) Voltaje equivalente de Thévenin para el circuito de entrada de un motor de inducción. b) Impedancia equivalente de Thévenin en el circuito de entrada. c) Circuito equivalente simplificado resultante para el motor de inducción. (Chapman, 2012)	90
Figura 36. Análisis comparativo entre los valores originales y los obtenidos mediante PSO: a) Características Torque-Deslizamiento. b) Curva de Eficiencia. c) Curva de Corriente del Estator. d) Curva del Factor de Potencia.	95
Figura 37. Evolución de la función de costo vs número de iteraciones	96
Figura 38. Diagrama vectorial de un motor de inducción (NXP Semiconductor, F, 2021).	98
Figura 39. Diagrama de flujo de control vectorial propuesto – Elaboración propia	99
Figura 40. Circuito de máquina CC (Le Fosse, 2021)	102
Figura 41. Zonas de operación a flujo nominal y debilitado (Shen, 2023)	104
Figura 42. Lazo de control de velocidad	107
Figura 43. Control PI con anti-windup (Shen, 2023)	109
Figura 44. Lugar de raíz utilizado para el diseño del control de velocidad	110
Figura 45. Diagrama de Bode en lazo abierto	111
Figura 46. Diagrama de Bode en lazo cerrado	111
Figura 47. Diagrama de bloque en lazo de control cerrado del motor de inducción	112
Figura 48. Lugar de raíz utilizando para el diseño del control de corriente	115
Figura 49. Diagrama de bode de lazo abierto de lazo de control de corriente	116
Figura 50. Diagrama de bode de lazo cerrado de lazo de control de corriente	116
Figura 51. Acoplamiento cruzado – Elaboración propia	118
Figura 52. Limitador de corriente	119
Figura 53. Limitador de voltaje	120
Figura 54. Tensiones de línea de los estados de conmutación	121
Figura 55. Vectores básicos de un inversor de dos niveles	122
Figura 56. Proyección del vector de referencia en el sector 1 (NXP Semiconductor, F, 2021)	124
Figura 57. Detalle de la proyección del vector de referencia en el sector 1. – Elaboración propia ...	124
Figura 58. Cálculo ciclo de trabajo (a) Alineamiento simétrico b) Secuencia alternada) (Sancha Miguel, 2023).	126

Figura 59. Estructura de EKF (Tiwari, 2019).	130
Figura 60. Referencia de velocidad	133
Figura 61. Referencia de torque	134
Figura 62. Velocidad de motor de inducción	134
Figura 63. Torque electromagnético	135
Figura 64. Corrientes del estator del motor de inducción	136
Figura 65. Corrientes del estator con acercamiento	136
Figura 66. Tensión aplicada a la carga	137
Figura 67. a) FOC con control por debilitamiento de flujo b) FOC sin control por debilitamiento de flujo	138
Figura 68. Estimación de velocidad mediante EKF (comparación con medición y referencia)	139
Figura 69. Torque electromagnético del motor bajo control FOC sensorless	140
Figura 70. Corrientes trifásicas del estator i_{sa} , i_{sb} , i_{sc}	141
Figura 71. Corrientes $\alpha\beta$ medidas vs. $\alpha\beta$ estimadas	142
Figura 72. Corrientes $\alpha\beta$ medidas vs. $\alpha\beta$ estimadas (zoom entre 0–0.9 s)	142
Figura 73. Respuesta de la velocidad estimada ante ruido de medición	143
Figura 74. Respuesta de la velocidad estimada tras el ajuste de la matriz de covarianza R	144
Figura 75. Comparación entre las corrientes de fase medidas (con ruido inyectado) y las estimadas por el EKF	145
Figura 76. Diagrama de flujo de diagnóstico de fallas en sensores de corriente	152
Figura 77. Sistema de control vectorial del motor de inducción con EKF y detección de fallas en sensores – Elaboración propia	156
Figura 78. Respuesta global del sistema ante falla de circuito abierto en sensor de velocidad	158
Figura 79. Velocidad real, estimada y referencia durante la falla, junto con activación del flag de detección	158
Figura 80. Respuesta global del FOC ante falla de circuito abierto en el sensor de corriente de fase i_{sa}	159
Figura 81. Corrientes medidas/estimadas y señal de diagnóstico para la falla de circuito abierto en i_{sa}	160
Figura 82. Respuesta del sistema FOC ante una falla de escalamiento en la corriente i_{sc}	161
Figura 83. Corriente medida vs corriente corregida y activación del flag de falla por escalamiento	162
Figura 84. Comportamiento de Índice de escalamiento (V_c)	162
Figura 85. Respuesta del sistema ante una falla de offset en la corriente de fase i_{sa}	163
Figura 86. Detección y compensación del offset en la corriente de fase i_{sa} : señal corregida y activación del flag de falla	164

Figura 87. Comportamiento de segundo promedio móvil (<i>Sisa</i>).....	164
Figura 88. Banco de pruebas experimentales	166
Figura 89. Arquitectura AMBA típica	168
Figura 90. Pipeline de 3 etapas del Cortex-M4.....	169
Figura 91. Placa de desarrollo Nucleo-G431RB.....	171
Figura 92. STEVAL-IHM023V3 (STMicroelectronics., 2017).....	172
Figura 93. X-NUCLEO-IHM09M1 (STMicroelectronics).....	173
Figura 94. Señales en Modo Alineado al Centro (Zhu, 2023)	174
Figura 95. Inserción de tiempos muertos en señales complementarias.	175
Figura 96. Sensorless de Control de campo orientado usando observador de estado Luenberger (STMicroelectronics, 2020)	179
Figura 97. Diagrama de bloques del observador adaptativo	182
Figura 98. Interfaz principal de configuración del proyecto para motor de inducción (ACIM) en STM32 MC SDK Workbench.....	186
Figura 99. Parámetros del Motor	186
Figura 100. Parámetros de sintonización de controles PI de velocidad, corrientes y observador LSO	187
Figura 101. Mapeo de pines y asignación de periféricos del STM32G431RB en la interfaz de STM32CubeMX	187
Figura 102. Configuración de Timer 1	190
Figura 103. Validación Experimental de Señales PWM Complementarias a 16 kHz implementadas en STM32G431RB	191
Figura 104. Banco de pruebas experimental para motor de inducción sin carga.....	192
Figura 105. Banco de pruebas experimental para motor de inducción con carga.....	193
Figura 106. Interfaz gráfica de resultados experimentales obtenida mediante la herramienta STM32 MotorPilot. durante la prueba sin carga.....	194
Figura 107. Respuesta dinámica del control FOC sensorless ante escalones de carga variable (STM32 MotorPilot)	195
Figura 108. Seguimiento de perfiles de velocidad bajo condiciones de carga mecánica constante (STM32 MotorPilot).	196
Figura 109. Modelo de motor de inducción en marco de referencia estacionario	215
Figura 110. Ecuaciones de voltajes del estator	215
Figura 111. Ecuaciones de voltajes del rotor	215
Figura 112. Cálculo de corrientes	216
Figura 113. Ecuación mecánica	216
Figura 114. Transformada Clarke – Park.....	216
Figura 115. Transformada Clarke – Park inverso.....	216

Figura 116. Control vectorial de motor de inducción	217
Figura 117. Control PI anti-windup de velocidad.....	218
Figura 118. Control PI anti-windup de corrientes.....	218
Figura 119. Limitador de corriente y voltaje	218
Figura 120. Cálculo de i_{sq} a partir de T_e	219
Figura 121. Acoplamiento cruzado.....	219
Figura 122. Transformada de Clarke – Park inverso	219
Figura 123. Modulación vectorial.....	219
Figura 124. Transformada de Clarke y Forma polar.....	219
Figura 125. Cálculo de tiempos	220
Figura 126. Temporizador (portadora de señal).....	221
Figura 127. Comparación de sector	221
Figura 128. Inversor de 2 Niveles.....	222
Figura 129. Estimación de ángulo eléctrico.....	222
Figura 130. Estimación de flujo de rotor	222
Figura 131. Control por flujo debilitado	222
Figura 132. Control vectorial sensorless mediante filtro de Kalman extendido (EKF) de motor de inducción.....	223
Figura 133. Filtro de Kalman Extendido (Input – Output)	224
Figura 134. Control tolerante de fallas de sensores corriente y velocidad de control vectorial de motor de inducción	226
Figura 135. Sistema de reconfiguración	227
Figura 136. Bloque Sistema de diagnóstico de falla.....	227
Figura 137. Bloque de Estimación de corrientes $i_{s\alpha}$, $i_{s\beta}$	229
Figura 138. Función de Energía.....	234

INDICE DE ANEXOS

Anexo 1: Matriz de consistencia y matriz de operacionalización de variables	206
Anexo 2: Implementación de algoritmo PSO para determinación de parámetros de motor de inducción mediante MATLAB	210
Anexo 3: Modelo de motor de inducción en Simulink	215
Anexo 4: Implementación de control de campo orientado de motor de inducción en SIMULINK	217
Anexo 5: Implementación de control vectorial sensorless de motor de inducción en SIMULINK	223
Anexo 6: Implementación de Control tolerante de fallas de sensores corriente y velocidad de control vectorial de motor de inducción en Simulink.....	226
Anexo 7: Estabilidad de Lyapunov	234
Anexo 8: Código fuente de lazos de control de corriente, velocidad y el observador de estado LSO	235

ABREVIATURAS

PSO: Particle Swarm Optimization (Optimización por Enjambre de Partículas)

VSD: Variable Speed Drive (Variador de Velocidad)

FOC: Field Oriented Control (Control de Campo Orientado)

IFOC: Indirect Field Oriented Control (Control de Campo Orientado Indirecto)

DFOC: Direct Field Oriented Control (Control de Campo Orientado Directo)

PWM: Pulse Width Modulation (Modulación por Ancho de Pulso)

SVPWM: Space Vector Pulse Width Modulation (Modulación por Vector Espacial)

RMS: Root Mean Square (Raíz Cuadrada Media)

FTC: Fault Tolerant Control (Control tolerante a Fallas)

AFTC: Active Fault Tolerant Control (Control Tolerante a Fallas Activo)

PFTC: Pasive Fault Tolerant Control (Control Tolerante a Fallas Pasivo)

FDI: Fault Detection and Isolation (Detección y aislamiento de fallas)

PDF: Probability Density Function (Función de densidad de probabilidad)

EKF: Extended Kalman Filter (Filtro de Kalman extendido)

LSO: Luenberger State Observer (Observador de Estado de Luenberger)

INTRODUCCIÓN

El motor de inducción se ha consolidado como el actuador eléctrico predominante en la industria moderna gracias a su bajo costo de fabricación y mantenimiento, y alto rendimiento en un amplio rango de aplicaciones. En procesos que exigen un control preciso de velocidad y torque, como fajas transportadoras, molinos, bombas e incluso vehículos eléctricos se recurre a estrategias avanzadas como el Control de Campo Orientado (FOC), el cual desacopla el control del flujo y del torque logrando una dinámica comparable a la de los motores de corriente continua. No obstante, esta mejora en el desempeño introduce una dependencia crítica de mediciones precisas de las corrientes del estator y la velocidad del rotor, convirtiendo a los sensores en el eslabón más vulnerable del sistema de control.

En entornos industriales y aplicaciones críticas como la electromovilidad, los sensores están expuestos a ruido electromagnético y vibraciones que degradan su señal y provocan fallas recurrentes, lo que puede producir fallas de diversa naturaleza: circuito abierto, desplazamiento de offset o errores de escalamiento. La consecuencia directa de cualquiera de estas fallas es la pérdida de estabilidad del accionamiento, con implicaciones que van desde costosas paradas no programadas en líneas de producción hasta riesgos de seguridad en aplicaciones críticas como la electromovilidad. La solución clásica de redundancia hardware resulta inviable técnica y económicamente en aplicaciones donde el espacio, el peso y el costo son restricciones prioritarias.

En este contexto, las técnicas de control sensorless basadas en observadores de estado han cobrado gran relevancia, ya que permiten estimar variables no medidas a partir de modelos matemáticos del sistema. No obstante, la precisión de estas estrategias depende directamente de la exactitud de los parámetros del motor, lo que hace necesaria la implementación de métodos robustos de identificación. Asimismo, la incorporación de esquemas de control

tolerante a fallas (FTC) permite detectar, aislar y compensar fallas en sensores mediante redundancia analítica, evitando la necesidad de duplicación de hardware.

Una vez establecido un modelo preciso, la investigación avanza hacia el diseño en simulación de un esquema de Control Tolerante a Fallas (FTC). Este sistema se apoya en un Filtro de Kalman Extendido (EKF) capaz de estimar el flujo y la velocidad del rotor bajo condiciones dinámicas, identificando simultáneamente pérdidas de señal o degradaciones paramétricas para reconfigurar automáticamente el lazo de control.

Finalmente, para demostrar la viabilidad práctica y la ejecución en tiempo real de la redundancia analítica, se implementa la estrategia de control vectorial sensorless utilizando un Observador de Estado de Luenberger (LSO) embebido en un microcontrolador de propósito general STM32G431RB. De esta manera, se comprueba la estabilidad, el seguimiento de referencias y el desempeño dinámico.

El presente documento se organiza de la siguiente manera. El Capítulo 1 expone el planteamiento del problema, los objetivos, la justificación y el alcance de la investigación. El Capítulo 2 desarrolla el marco teórico, cubriendo el modelado del motor de inducción, identificación de parámetros de motor de inducción, los fundamentos del control vectorial, el Filtro de Kalman Extendido y los principios del Control Tolerante a Fallas. El Capítulo 3 aborda la identificación de parámetros mediante PSO y su comparación con pruebas experimentales. El Capítulo 4 presenta el diseño y simulación del control vectorial sensorless con EKF. El Capítulo 5 describe el diseño y validación en simulación del sistema de Control Tolerante a Fallas. El Capítulo 6 detalla la implementación experimental sobre el microcontrolador STM32. Finalmente, se presentan las conclusiones y recomendaciones del trabajo.

CAPÍTULO I: PLANTEAMIENTO DEL PROBLEMA

1.1 Descripción del problema de investigación

1.1.1 Enunciado de la investigación

“Diseño de control vectorial sensorless tolerante a fallas de sensores de motor de inducción”

1.1.2 Interrogantes de la investigación

1.1.2.1 Interrogante principal

¿Como puede diseñarse un sistema de control vectorial tolerante a fallas de sensores de velocidad y corriente de motores de inducción?

1.1.2.2 Interrogantes específicas

- ¿Qué tipos de fallas en sensores de corriente y velocidad afectan significativamente el desempeño y la estabilidad del control vectorial en motores de inducción?
- ¿Cuáles son los criterios y ventajas del uso del algoritmo de Enjambre de Partículas (PSO) para la identificación de parámetros del modelo dinámico del motor?
- ¿Cuáles son los métodos de estimación más adecuados para obtener variables críticas como la corriente de fase y la velocidad del rotor, y en qué condiciones es preferible emplear observadores como Luenberger extendido, observador de modo deslizante (SMO), Sistemas Adaptativos de Referencia de Modelo (MRAS) o EKF?
- ¿Cómo diseñar e implementar funciones lógicas que permitan detectar en tiempo real fallas de tipo offset y escala en sensores de corriente?
- ¿Qué tan eficiente es el Filtro de Kalman Extendido (EKF) en la estimación de variables de corriente y velocidad en comparación con otros observadores?
- ¿Qué aspectos técnicos y prácticos deben considerarse para la implementación del control vectorial sensorless? ¿Qué tipos de controladores y hardware son ideales para su implementación?

1.1.3 Descripción del problema

El motor de inducción es el actuador eléctrico predominante en la industria moderna debido a su bajo costo, robustez y alto desempeño. En aplicaciones que requieren un control preciso de torque y velocidad, se emplean estrategias avanzadas como el Control de Campo Orientado (FOC), el cual permite desacoplar el control del flujo y del torque del motor, logrando una dinámica similar a la de los motores de corriente continua y mejorando significativamente el desempeño del accionamiento. Sin embargo, la implementación de estos esquemas de alto desempeño introduce una dependencia crítica de mediciones precisas de variables eléctricas y mecánicas, principalmente las corrientes del estator y la velocidad del rotor. Investigaciones recientes señalan que esta dependencia convierte a los sensores en uno de los puntos más vulnerables del sistema de control. Los sensores de corriente y velocidad son propensos a fallas recurrentes en entornos industriales, donde el ruido electromagnético y las vibraciones degradan la señal, comprometiendo la estabilidad de todo el accionamiento eléctrico (Adamczyk, & Orłowska, 2024).

La avería de un sensor no se considera únicamente un contratiempo técnico, sino que acarrea repercusiones críticas, las cuales son cambiantes en función de la aplicación. En el entorno de la fabricación, se erige como un peligro económico extremo. Según los informes más recientes, el costo de una hora de parada no programada se ha cuadruplicado, de manera que el impacto en las utilidades es drástico (Siemens, 2024). Sin embargo, el escenario es aún más crítico en el contexto de la electromovilidad, donde la falla adquiere relevancia desde la perspectiva de la seguridad funcional. A diferencia de una línea de producción, un vehículo eléctrico en operación debe garantizar el mantenimiento de la funcionalidad en operación para reducir el peligro y evitar accidentes como consecuencia de la pérdida de un sensor. Al respecto, investigaciones recientes corroboran los principios de la norma ISO 26262; el

diagnóstico anticipado y la tolerancia a fallas son necesidades deseadas para prevenir daño catastrófico del tren de potencia en los vehículos actuales (Kumar, Velpula, Saiprakash, & Sahoo, 2025).

Este hecho hace que la estrategia clásica de redundancia de hardware (sensor duplicado) no sea válida ni técnica, ni económicamente, por un lado, en aplicaciones donde el espacio es escaso, en donde se pretende disminuir el peso, lo que hace priorizar soluciones de tipo software (Wu, Sehab, Akrad, & Morel, 2022), y por otro lado, la mayoría de los variadores de frecuencia de la industria en general no tiene mecanismos de redundancia analítica y deja que, al perder la señal, se detenga el proceso por motivos de seguridad (Tran, 2021).

Por lo tanto, se considera importante la investigación y el desarrollo de estrategias de control de tipo vectorial que disminuyan la dependencia de la sensórica. Las técnicas sensorless basadas en observadores de estado son una buena estrategia para mejorar la robustez de control sin aumentar la complejidad del hardware (Morfin, y otros, 2025).

Sin embargo, para que estos algoritmos sean confiables, dependen de un modelo matemático exacto. Por ello, es importante una estrategia donde primero, se identifique con precisión los parámetros del motor mediante métodos experimentales u optimización (PSO) y, posteriormente, utilizarlos en observadores robustos.

Finalmente, es necesario destacar que los principios aplicados del control tolerante a fallas en los sistemas de control de accionamientos eléctricos tienen afinidades con otras especialidades y disciplinas como la automatización, la mecatrónica y la aeronáutica. Esta transversalidad hace que el control tolerante a fallas no sólo sea importante para la continuidad de la producción, sino, y también para garantizar la seguridad de las personas que trabajan e interactúan con estos sistemas (Abbaspour, Mokhtari, Sargolzaei, & Yen, 2020).

Con base en lo expuesta, la presente investigación aborda esta problemática estructurando el desarrollo en tres fases metodológicas secuenciales. En primer lugar, se realiza

la identificación de los parámetros del motor de inducción mediante una comparación entre métodos experimentales clásicos (pruebas de vacío y rotor bloqueado) y el algoritmo de optimización por enjambre de partículas (PSO), garantizando la precisión del modelo matemático. Posteriormente, a nivel de simulación, se diseña un esquema de control sensorless utilizando un Filtro de Kalman Extendido (EKF), sobre el cual se aplica y evalúa una estrategia de control tolerante a fallas (FTC) ante fallas simulada de los sensores de corriente y velocidad. Finalmente, para validar la viabilidad física de la estimación, se lleva a cabo la implementación experimental de la estrategia sensorless utilizando un Observador de Estado de Luenberger sobre un microcontrolador, comprobando su desempeño en tiempo real.

1.2 Área de conocimiento

Según lo establecido por la CONICYT (2015), la presente investigación se organiza y ubica dentro de las áreas y subáreas del conocimiento, de la manera que se describe a continuación:

ÁREA:

Ingeniería y Tecnología

SUB-ÁREA:

Ingeniería eléctrica, electrónica

LÍNEA DE INVESTIGACIÓN:

Control y automatización de sistemas eléctricos

Diagnóstico y tolerancia a fallas en sistemas de control

1.3 Antecedentes

Mohammadi y Akhavan (2014), en su investigación sobre la estimación de parámetros para motores de inducción trifásicos, abordaron las limitaciones prácticas y la complejidad de realizar ensayos tradicionales de rotor bloqueado y vacío en entornos industriales. Los autores desarrollaron y validaron una metodología basada en inteligencia computacional, evaluando el

desempeño de la Optimización por Enjambre de Partículas (PSO), algoritmos genéticos y una técnica híbrida (HGAPSO), además proponen utilizando para ello únicamente datos estandarizados de manufactura como el torque de arranque, torque máximo y factor de potencia. Su estudio demostró que es posible identificar con alta precisión las resistencias y reactancias del circuito equivalente mediante la minimización del error entre los valores de placa y los calculados por el modelo en la función de costo, validando así el uso del algoritmo PSO como una estrategia confiable y no invasiva para obtener el modelo matemático necesario en esquemas de control vectorial.

En Aguilera et al. (2022), los autores proponen una estrategia de control tolerante a fallas para motores de inducción aplicada a vehículos eléctricos, basada en un banco de observadores de modo deslizante (SMO) operando en un marco de referencia común. El método permite la detección y aislamiento rápido de fallas únicas y múltiples en sensores de corriente y velocidad mediante la generación de residuos desacoplados, y utiliza los mismos observadores para reconstruir las señales defectuosas como sensores virtuales. Los resultados experimentales demuestran una reconfiguración en tiempos inferiores a 1 ms y robustez frente a variaciones paramétricas. No obstante, el enfoque presenta una elevada complejidad de diseño e implementación, lo que para su aplicabilidad es necesario sistemas embebidos de alta capacidad computacional.

En Wu, Zhao y Liu (2016), se aborda el problema de la detección, aislamiento y reconfiguración de fallos en sensores de corriente y velocidad en motores de inducción controlados vectorialmente, proponiendo un método basado en filtros de Kalman extendidos (EKF). A diferencia de enfoques tradicionales que recurren a redundancia física o métodos de diagnóstico simples, este trabajo integra técnicas avanzadas de análisis de residuos y funciones especiales en ventanas deslizantes para identificar en tiempo real errores de medición de corriente (compensación y escala), así como fallos de circuito abierto, detectados cuando las

salidas de los sensores caen a cero y los residuos con respecto al EKF superan umbrales predefinidos. En caso de fallo, la reconfiguración automática transfiere las señales de realimentación a las salidas del EKF, asegurando la continuidad del control sin interrupciones. Si bien este método demostró robustez en simulación ante cambios de par, el estudio se limitó a la validación teórica sin abordar la implementación en hardware.

Tran (2021), en su investigación sobre diagnóstico de fallas en accionamientos de motores de inducción, propusieron una estrategia de control tolerante a fallas (FTC) que integra algoritmos de detección lógica de alta velocidad, capaces de identificar errores en sensores de corriente y velocidad en un rango de 2 a 3 ms. Su metodología destaca por implementar una reconfiguración híbrida: ante la pérdida de sensores, el sistema conmuta automáticamente a un Observador de Luenberger modificado para la estimación de corrientes o a un Observador de Modos Deslizantes (SMO) para la velocidad, validando experimentalmente que es posible mantener la estabilidad del control vectorial mediante redundancia analítica sin recurrir a la duplicidad de hardware.

Azzoug, Y et al. (2021), proponen una estrategia de control tolerante a fallos de sensores de corriente para accionamientos de motores de inducción trifásicos, basada en el control vectorial y en la estimación de corrientes mediante el observador de Luenberger adaptativo. El sistema integra de forma coherente la detección del fallo, a través del análisis del residuo entre la corriente medida y la estimada con un umbral fijo del 17% de la corriente nominal, la aislación del sensor defectuoso mediante un índice lógico Z de 8 estados posibles, y la reconfiguración automática del sistema mediante un conmutador multipuerto. Una contribución destacada es el uso de un único observador para reconstruir las tres corrientes de fase. La validación experimental sobre un motor de 1,1 kW con tarjeta dSpace 1104 confirmó la efectividad del esquema ante fallos sucesivos, arranque con offset y variaciones de velocidad y carga. Sin embargo, los autores reconocen como limitación principal la sensibilidad del

método ante variaciones paramétricas significativas provocadas por altas temperaturas de operación.

Xie, Xu et al. (2025), en su artículo, abordan las limitaciones de estabilidad de los observadores de estado convencionales (topología Luenberger) en regiones de baja velocidad y regeneración. Los autores proponen un Observador de Estado de Orden Completo Mejorado (IFOSO), en el cual rediseñan la matriz de ganancia de retroalimentación para desacoplar la estimación de velocidad de la del flujo, asegurando así la ubicación de los polos en el semiplano izquierdo para garantizar la estabilidad global. Esta propuesta fue validada experimentalmente en un banco de pruebas basado en DSP, demostrando que la mejora sobre el algoritmo clásico de Luenberger reduce significativamente las oscilaciones de velocidad y aumenta la precisión del torque, confirmando la vigencia de esta técnica para aplicaciones de alto desempeño.

1.4 Objetivos

1.4.1 Objetivo General

Diseñar una estrategia de control vectorial sensorless con tolerancia a fallas de sensores para un motor de inducción WEG de 1 HP IEC Standard Efficiency (IE1), integrando la identificación de parámetros mediante optimización por enjambre de partículas y la implementación experimental de la etapa de estimación en un sistema embebido de bajo costo, evaluando su desempeño en términos de error de estimación, estabilidad y robustez ante fallas.

1.4.2 Objetivos Específicos

- Modelado de motor de inducción y posterior determinación de los parámetros eléctricos del motor de inducción mediante métodos experimentales y comparar con algoritmo de Optimización por Enjambre de Partículas (PSO) utilizando datos de placa y manufactura, garantizando la precisión parámetros.

- Diseñar de un de control de campo orientado (FOC) sensorless utilizando Filtro de Kalman Extendido (EKF) evaluando su desempeño mediante simulación.
- Diseñar un esquema de control tolerante a fallas (FTC) y estimación sensorless basado en el Filtro de Kalman Extendido (EKF), validando su robustez ante pérdidas de señal de sensores, evaluando su desempeño con distintas condiciones de operación y fallas mediante simulación.
- Implementar el algoritmo de control vectorial sensorless utilizando un Observador de Estado de Luenberger en un microcontrolador STM32G431, evaluando su desempeño en tiempo real.

1.5 Justificación

Justificación Técnica

Esta investigación se justifica técnicamente al proponer una validación de control tolerancia a fallas. Por un lado, se utiliza la simulación para diseñar y validar la lógica de Control Tolerante a Fallas (FTC), evaluando escenarios de riesgo (como desconexiones abruptas de sensores). Por otro lado, la implementación experimental se enfoca en el desafío más crítico: validar que los algoritmos de estimación de estado (Control Vectorial Sensorless) son ejecutables en tiempo real dentro de un sistema embebido. Al lograr que el algoritmo corra en el hardware, se demuestra la factibilidad técnica de la "redundancia analítica", que es parte importante necesaria para que cualquier sistema FTC funcione.

Justificación Económica

De forma tradicional, la alta performance del control vectorial necesitaba de equipos costosos como Procesadores Digitales de Señales (DSP), sin embargo, esta tesis en su justificación positiva demuestra la tradición económica de que es posible implementar prácticas

avanzadas de control sensorless en Microcontroladores de propósito general y de bajo coste (familia STM32) para motor de inducción de baja potencia.

Justificación Metodológica

Desde un punto de vista metodológico, esta investigación se considera justificada al plantear un enfoque sistemático para la adecuada identificación de los parámetros del motor de inducción. El trabajo introduce el algoritmo de Optimización por Enjambre de Partículas (PSO) como un medio de verificación offline. Esta adecuada identificación constituye un requisito importante para la siguiente fase (diseño del control vectorial y Observador de Estado), que presenta una estabilidad, robustez y convergencia fuertemente dependientes de la correcta identificación de estos parámetros.

Justificación Académica

Esta investigación contribuye al fortalecimiento del conocimiento en el área de control avanzado de máquinas eléctricas al integrar, técnicas de modelado, identificación de parámetros, estimación de estados y control vectorial sensorless. Asimismo, la investigación aporta una introducción del Control Tolerante a Fallas que es ampliamente estudiado y utilizado en diferentes campos de la ingeniería.

1.6 Alcance

El presente trabajo de investigación delimita su alcance en tres etapas complementarias:

- Se abarca la estimación de parámetros eléctricos del motor de inducción (resistencias y reactancias) mediante el algoritmo de Optimización por Enjambre de Partículas (PSO), utilizando datos de placa y manufactura comparando con lo obtenido en pruebas experimentales para generar un modelo matemático preciso.

- Se contempla el diseño de un sistema de Control Tolerante a Fallas (FTC) robusto, validado mediante simulaciones en MATLAB/Simulink. Esta etapa abarca la implementación de algoritmos de detección lógica para fallas en sensores de corriente y velocidad, así como la estimación de variables de estado mediante el Filtro de Kalman Extendido (EKF).
- Validación Experimental: El alcance práctico se centra en la implementación del algoritmo de control vectorial *sensorless* en un sistema embebido basado en el microcontrolador STM32G4. En esta etapa se valida el desempeño en tiempo real del lazo de control y del observador de estado (LSO) bajo condiciones de operación normal, demostrando la viabilidad técnica del accionamiento propuesto en hardware de bajo costo.

1.7 Limitaciones

La implementación en hardware se limita a la validación del desempeño dinámico del control vectorial *sensorless* utilizando el Observador de Estado de Luenberger (LSO). Por su parte, la estrategia de Tolerancia a Fallas (FTC) basada en el Filtro de Kalman Extendido (EKF) se valida a nivel de simulación, dado que su alta carga computacional (operaciones matriciales en cada ciclo de muestreo) requiere una optimización de código específica o un hardware de mayor capacidad (DSP/FPGA) para garantizar la ejecución en tiempo real junto con el lazo de control.

CAPÍTULO II: MARCO TEÓRICO

2.1 Motor de inducción

El motor de inducción trifásico (M.I.) se emplea particularmente en máquinas industriales debido a su robustez, simplicidad de construcción, bajo costo y alta fiabilidad. Este tipo de motor opera con corriente alterna (CA) y se ha consolidado como una solución eficaz en una amplia variedad de sistemas electromecánicos, incluidos los vehículos eléctricos, donde destaca por su capacidad de operar en un amplio rango de velocidades mediante el debilitamiento de campo y su seguridad inherente, el avance en los procesadores de señales digitales (DSP) ha permitido que su implementación sea tan eficiente y competitiva como la de tecnologías BLDC (Brushless DC) o de Reluctancia Conmutada (SRM).

Principio de funcionamiento

El motor de inducción está constituido esencialmente por dos elementos: el rotor y el estator. El estator, que permanece fijo, recibe el suministro de corriente alterna trifásica y origina un campo magnético giratorio, cuya velocidad está determinada por la frecuencia de la alimentación y el número de polos presentes. El rotor, separado de este mediante un entrehierro, suele fabricarse en configuración de jaula de ardilla. Cuando dicho rotor se encuentra bajo la acción del campo magnético generado por el estator, se inducen corrientes en sus barras conductoras, lo que origina un campo magnético propio. La interacción entre ambos campos produce el torque electromagnético responsable del movimiento de giro. El motor de inducción se compone principalmente de tres componentes: el estator, el rotor y los cojinetes como se ve la Figura 1. (Olvera, 2022)

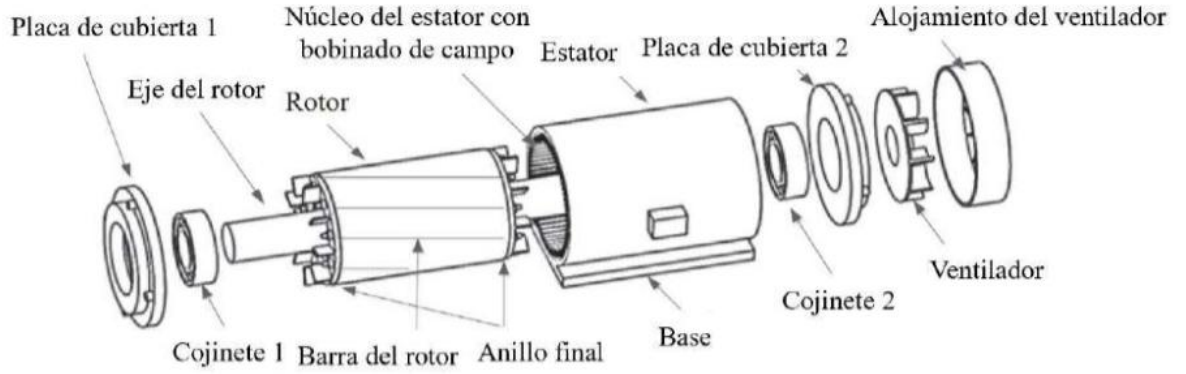


Figura 1. Vista esquemática de motor de inducción (Olvera, 2022).

Al aplicar una corriente alterna trifásica balanceada a los bobinados del estator, se origina un campo magnético giratorio. La rapidez con la que dicho campo rota se denomina velocidad síncrona, la cual está determinada tanto por la frecuencia de la fuente de alimentación como por la cantidad de polos que posee el motor (Chapman, 2012).

La velocidad síncrona se define por la ecuación (2.1):

$$n_s = \frac{60 * f_s}{p} \quad (2.1)$$

Donde: n_s es la velocidad síncrona en revoluciones por minuto (rev/min), f es la frecuencia (Hertz) y p representa el número de pares de polos de la máquina.

El campo magnético giratorio producido en el estator cruza el entrehierro e induce corrientes eléctricas en las barras del rotor, originando así su propio campo magnético. La interacción de ambos campos es la responsable de generar el torque electromagnético que impulsa la rotación del rotor. Sin embargo, este no logra igualar la velocidad síncrona, pues es indispensable la existencia de una diferencia relativa de velocidad para que se mantenga la inducción de corriente (Olvera, 2022). A dicha diferencia entre la velocidad del campo giratorio y la velocidad real del rotor se le denomina deslizamiento, el cual aparece como consecuencia del efecto de la carga que tiende a oponerse al movimiento del rotor. Esta diferencia se expresa en (2.2) y (2.3) como:

$$n_{des} = n_s - n_r \quad (2.2)$$

$$\omega_{mdes} = \omega_{ms} - \omega_{mr} \quad (2.3)$$

Donde n_{des} , ω_{mdes} representa la velocidad de deslizamiento en rev/min y rad/s, n_s y ω_{ms} representan la velocidad síncrona en rev/min y rad/s, y n_r y ω_{mr} representan la velocidad mecánica del rotor en rev/min y rad/s.

El deslizamiento (s) se define como la velocidad mecánica relativa expresada como una fracción de la velocidad síncrona. De tal manera que se representa mediante la ecuación (2.4):

$$s = \frac{n_s - n_r}{n_s} = \frac{\omega_{ms} - \omega_{mr}}{\omega_{ms}} \quad (2.4)$$

Utilizando esta definición de deslizamiento, la frecuencia del rotor (f_r) viene dada por la ecuación (2.5):

$$f_r = s f_s \quad (2.5)$$

donde f_s se refiere a la frecuencia del sistema en Hertz.

Es importante destacar que cuando el rotor alcanza la velocidad síncrona, el deslizamiento $s = 0$. En contraste, si el rotor permanece detenido o en reposo, el deslizamiento toma el valor de $s = 1$. Por lo tanto, las velocidades de operación habituales del motor se encuentran siempre dentro de estos dos límites.

La velocidad mecánica del eje del rotor puede expresarse en función de la velocidad síncrona y del deslizamiento. Al reorganizar las ecuaciones correspondientes y despejar la incógnita, se obtiene la siguiente relación para la velocidad mecánica según (2.6) y (2.7):

$$n_r = (1 - s) * n_s \quad (2.6)$$

$$\omega_{mr} = (1 - s) * \omega_{ms} \quad (2.7)$$

La frecuencia del rotor, para una velocidad de giro arbitraria, resulta proporcional a la diferencia existente entre la velocidad síncrona y la velocidad real del rotor. En otras palabras, cualquier variación en la rapidez de rotación del rotor se refleja de manera directa en su frecuencia eléctrica como se muestra en la ecuación (2.8).

$$f_r = \frac{p}{60}(n_s - n_r) \quad (2.8)$$

Para estudiar y comprender el desempeño de un motor de inducción, así como para determinar parámetros fundamentales como el torque y la potencia desarrollada, se emplea su circuito equivalente. Este modelo eléctrico simplificado describe el comportamiento del motor mediante resistencias, reactancias e impedancias, permitiendo un análisis más preciso de sus características. En la Figura 2 se presenta el circuito equivalente por fase de un motor de inducción.

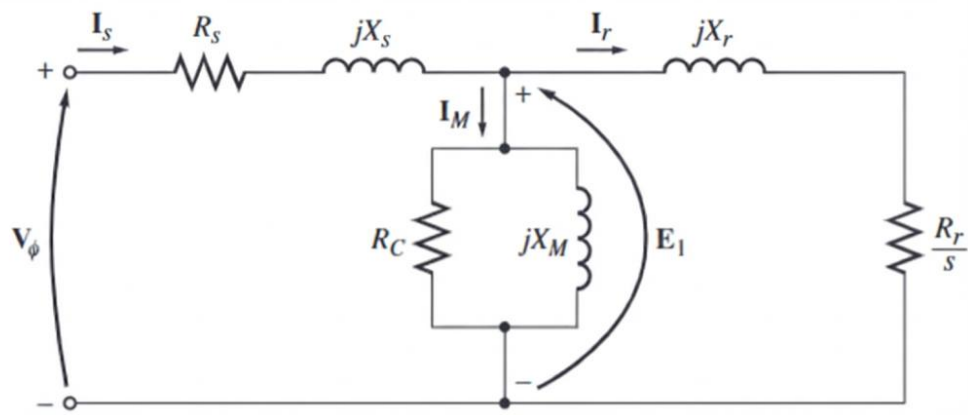


Figura 2. Circuito equivalente por fase de un motor de inducción (Chapman, 2012)

En referencia a la Figura 2, I_s representa la corriente del estator en amperios, la tensión por fase en voltios es V_ϕ , R_s y X_s representan la resistencia y la reactancia de fuga o dispersión del estator, respectivamente, en ohmios, I_m representa la corriente de magnetización en amperios, R_c representa la resistencia del núcleo en ohmios, X_m representa la inductancia de magnetización en ohmios, E_1 se refiere a la tensión inducida en el rotor en voltios, I_r representa la corriente del rotor en amperios, X_r y R_r se refieren a la reactancia de fuga y la resistencia del rotor, respectivamente, en ohmios. La deducción del circuito equivalente se encuentra en (Chapman, 2012).

2.2 Accionamientos eléctricos de motor de inducción

Un accionamiento eléctrico puede entenderse como un sistema compuesto por diversos dispositivos que, de manera conjunta, permiten transformar la energía eléctrica en energía mecánica utilizable. Estos sistemas están presentes en un sinnúmero de ámbitos: desde los electrodomésticos y el transporte, hasta la industria manufacturera y la informática. De hecho, gran parte de las actividades cotidianas serían inviables sin su uso. La diversidad de aplicaciones exige que cada accionamiento sea adaptado a las necesidades específicas de cada caso. Por esta razón, en la actualidad un accionamiento eléctrico no se limita al motor como elemento encargado de convertir la energía, sino que también incorpora un convertidor de potencia y unidades de control que optimizan su funcionamiento. El avance de la electrónica de potencia y de los microprocesadores dedicados al control ha transformado notablemente la manera en que se diseñan estos sistemas. Tanto es así, que en muchos casos la sección de conversión y control alcanza un nivel de sofisticación mayor que el propio motor (Košťál,, 2021)

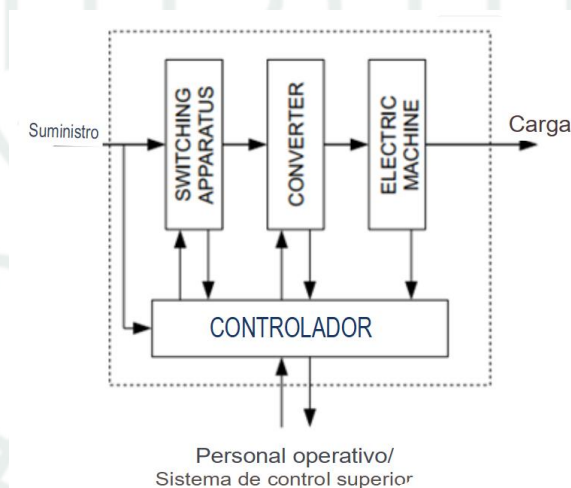


Figura 3. Estructura general de un accionamiento eléctrico (Košťál,, 2021)

Los accionamientos eléctricos se clasifican según el control de velocidad en tres tipos: de velocidad única, con varias velocidades discretas y de velocidad variable continua. Los dos primeros requieren poco o ningún control, mientras que los de velocidad variable,

especialmente con máquinas de inducción, son más complejos pero muy ventajosos. Su uso abarca aplicaciones de transporte e industriales, siendo estos trenes de alta velocidad, barcos, minería, vehículos eléctricos, ascensores, robots, bombas y compresores, aunque se emplean poco en electrodomésticos por su mayor costo y menor eficiencia.

La Figura 4 evidencia el esquema típico de un accionamiento industrial de velocidad variable con máquina de inducción, conformado por motor, convertidor, control y, a veces, sensor de velocidad. Este último no siempre es necesario gracias a los métodos modernos de control sin sensor, aunque sigue presente en muchos sistemas industriales y ferroviarios.

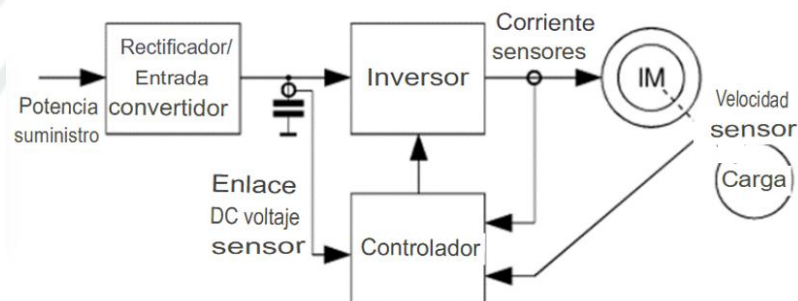


Figura 4. Estructura de un accionamiento común con máquina de inducción – Elaboración propia

En los accionamientos eléctricos industriales o de tracción se emplean sensores de corriente en los terminales de la máquina y sensores de voltaje en el enlace de CC del convertidor. Dado que medir directamente el voltaje de salida del inversor es complejo por su forma rectangular, este se reconstruye a partir del patrón de conmutación y del voltaje de enlace de CC, considerando además la no linealidad del inversor. Con ello se obtienen del estator su voltaje y frecuencia. Los sensores de corriente suelen colocarse en dos fases del inversor, calculando la tercera, y los sensores de velocidad se montan en el eje o en la caja de engranajes para medir la velocidad mecánica

Convertidores utilizados en variadores de velocidad

Los convertidores para accionamientos de velocidad variable deben ser capaces de suministrar una tensión y una frecuencia de salida ajustables. Estos se clasifican en dos grandes

grupos: rotacionales y electrónicos (de estado sólido). Los primeros están obsoletos y tuvieron un uso limitado con motores de inducción, por lo que actualmente solo se consideran los electrónicos.

Los convertidores electrónicos se dividen en directos e indirectos, según tengan o no un enlace de corriente continua (DC link).

- Convertidores directos, como los cicloconvertidores y convertidores matriciales, transforman directamente la señal de entrada AC en una salida AC con distinta frecuencia sin pasar por una etapa DC. Los cicloconvertidores usan puentes rectificadores con tiristores en antiparalelo. Los convertidores matriciales permiten conectar cualquier fase de salida con cualquier fase de entrada mediante interruptores bidireccionales, y requieren una alimentación polifásica.
- Convertidores indirectos primero rectifican la señal AC a DC y luego la invierten nuevamente a AC con diferentes características. Si el enlace DC usa un condensador, actúa como una fuente de tensión (VSI, Voltage Source Inverter). Si emplea una inductancia, se comporta como una fuente de corriente (CSI, Current Source Inverter).

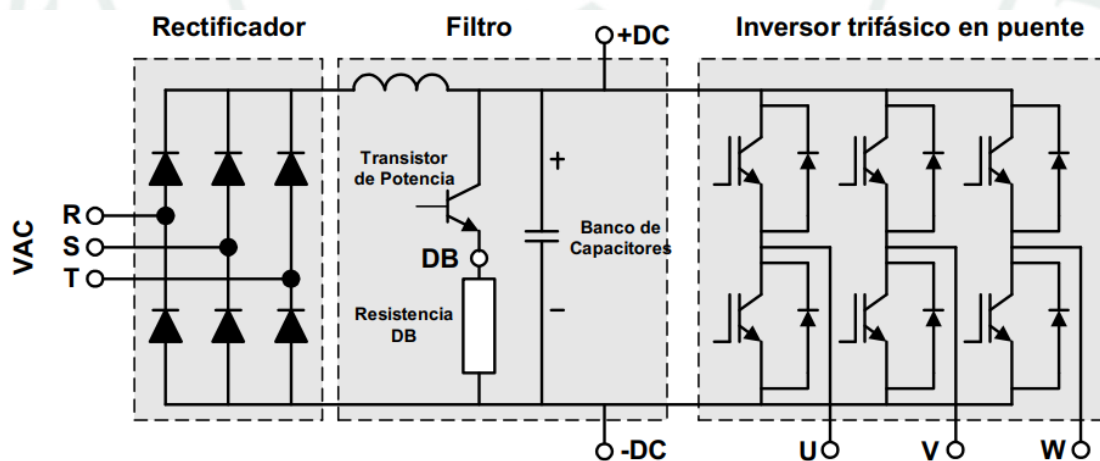


Figura 5. Topología del convertidor de potencia conectado al estator del motor de inducción (Dosreis, 2021).

En la Figura 5 se evidencia un inversor trifásico de dos niveles de tensión, empleado para la alimentación de un motor de inducción. Dicho inversor recibe energía a través de un

rectificador, el cual tiende transformar la corriente alterna en una señal de corriente continua que se aplica al enlace de continua (DC-link). Posteriormente, este voltaje pasa por una etapa de filtrado compuesta por una bobina (que actúa como filtro de corriente) y un condensador (que funciona como filtro de voltaje), antes de ser entregado al inversor de fuente de tensión (VSI).

El VSI está conformado por seis interruptores electrónicos del tipo IGBT, dispuestos en tres ramas bajo la topología de medio puente. A cada IGBT se le asocia un diodo de libre circulación, lo que permite el paso de corriente en el sentido contrario cuando es necesario. Cada par de transistores, superior e inferior se conecta directamente a una de las fases de la máquina. Mediante la conmutación controlada de los IGBT, la tensión continua se transforma en una señal de corriente alterna con amplitud y frecuencia variables, apta para accionar el motor de inducción.

Principio de operación

Los posibles estados de conmutación de un inversor pueden describirse mediante vectores de estado en un sistema trifásico. Esta representación se expresa como:

$$S = \frac{2}{3}(S_a + aS_b + a^2S_c) \quad (2.9)$$

Donde $a = e^{j\frac{2\pi}{3}}$

El inversor de dos niveles genera un total de ocho vectores de tensión, concernientes a ocho estados de conmutación distintos, cuya distribución espacial se evidencia por medio de la Figura 6.

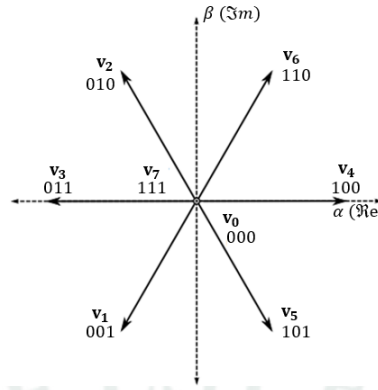


Figura 6. Distribución espacial de todos los vectores de tensión admisibles de un 2L-VSI (Le Fosse, 2021)

Cada uno de los vectores de tensión producidos se define a partir de la relación:

$$v = S \cdot V_{DC} \quad (2.10)$$

Donde v_a, v_b y v_c representan los voltajes de fase; S corresponde al estado de conmutación, y V_{DC} es la tensión del bus de corriente continua, calculada normalmente como:

$$V_{DC} = \sqrt{2} \cdot V_{nom} \cdot 1.1 \quad (2.11)$$

Siendo V_{nom} el voltaje nominal del sistema

El inversor trifásico de dos niveles (2L-VSI) se modela como un sistema discreto, en el cual los vectores de tensión generados (mostrados en la Tabla 1) constituyen las únicas acciones de control disponibles para regular el funcionamiento del motor.

Tabla 1 Vectores de voltaje de 2L-VSI

v_n	$s = [s_a \ s_b \ s_c]$	$v = Re + jIm$
v_0	0 0 0	0
v_1	1 0 0	$2/3v_{DC}$
v_2	1 1 0	$1/3v_{DC} + j\sqrt{3}/3v_{DC}$
v_3	0 1 0	$-1/3v_{DC} + j\sqrt{3}/3v_{DC}$
v_4	0 1 1	$-2/3v_{DC}$
v_5	0 0 1	$-1/3v_{DC} - j\sqrt{3}/3v_{DC}$
v_6	1 0 1	$1/3v_{DC} - j\sqrt{3}/3v_{DC}$
v_7	1 1 1	0

2.3 Determinación de parámetros de motor de inducción

Para que sea implementado control vectorial es fundamental contar con un conocimiento preciso de los principales parámetros del motor. Existen diversas técnicas de identificación de parámetros en máquinas de inducción, siendo las más conocidas las de tipo experimental, como la prueba en corriente continua, la prueba en vacío y la de rotor bloqueado. En esta tesis, además de emplear dichos métodos tradicionales, se propone la identificación de los parámetros mediante un enfoque de identificación basado en Particle Swarm Optimization (PSO), un método heurístico de optimización bioinspirado.

2.3.1 Introducción de identificación de parámetro por PSO

Particle Swarm Optimization (PSO) es un algoritmo de optimización inspirado en el comportamiento colectivo de sistemas naturales sin liderazgo central, tales como bancos de peces o bandadas de aves. En estos sistemas, los individuos exploran el entorno de manera autónoma, compartiendo información sobre las mejores regiones encontradas, lo que permite al grupo converger hacia soluciones óptimas mediante cooperación y aprendizaje colectivo.

El algoritmo PSO emplea una población de partículas que se desplazan dentro del espacio de búsqueda ajustando sus posiciones y velocidades en función de su propia experiencia y de la información compartida por el enjambre. Este proceso iterativo continúa hasta alcanzar un criterio de parada predefinido (Gad, 2022).

Las principales características del algoritmo PSO pueden resumirse de la siguiente manera:

- Las partículas comparten información y ajustan su movimiento en función de la experiencia individual y colectiva.
- Cada partícula conserva la mejor solución alcanzada a lo largo del proceso.
- Presenta una rápida convergencia hacia soluciones de buena calidad.
- La población se inicializa aleatoriamente y evoluciona de forma iterativa.

- No emplea operadores evolutivos clásicos como mutación o cruce.
- El número de partículas permanece constante durante toda la ejecución.

2.3.2 Estructura del PSO

La descripción del algoritmo PSO puede comprenderse a partir del estudio de los elementos que componen cada partícula dentro del sistema. En esencia, una partícula se caracteriza por un conjunto de vectores que describen su comportamiento dinámico y por dos indicadores numéricos que cuantifican la calidad de las soluciones obtenidas durante el proceso de búsqueda:

Componentes vectoriales de la partícula

En primer lugar, la partícula posee un vector de estado, denotado como $x_i = (x_{i,1}, x_{i,2}, \dots, x_{i,N})$, el cual representa la ubicación actual de la partícula dentro del espacio de soluciones.

Adicionalmente, se define un vector de mejor experiencia individual, expresado como $pbest_i = (pbest_{i,1}, pbest_{i,2}, \dots, pbest_{i,N})$. Este vector conserva las coordenadas asociadas a la mejor alternativa que la partícula ha identificado desde el inicio de la ejecución del algoritmo.

Por otra parte, la evolución de la partícula está gobernada por un vector de velocidad, indicado por $v_i = (v_{i,1}, v_{i,2}, \dots, v_{i,n})$. Dicho vector determina tanto la magnitud como la orientación del desplazamiento que la partícula realizará en la siguiente iteración (Mohammadi & Akhavan, 2014).

Dos valores de aptitud

Para valorar la eficacia de las soluciones generadas, se emplean dos métricas de desempeño.

El primer parámetro, denominado f_i^t , corresponde al nivel de desempeño de la posición actual de la partícula. Este valor se obtiene al aplicar la función objetivo sobre el vector x_i , reflejando así el costo o beneficio asociado a dicha posición.

El segundo parámetro, conocido como $f_{pbest,i}$, cuantifica la calidad de la mejor solución individual alcanzada por la partícula hasta ese momento. Este indicador está vinculado directamente con el vector $pbest_i$ y sirve como referencia para orientar futuros desplazamientos.

2.3.3 Desarrollo general del algoritmo PSO

El funcionamiento del algoritmo PSO puede describirse mediante los siguientes pasos:

1. **Inicialización de la población:** se generan aleatoriamente (o combinando distribución aleatoria y regular) las posiciones iniciales de todas las partículas en el espacio de búsqueda.
2. **Inicialización de velocidades:** las velocidades iniciales de las partículas se asignan aleatoriamente dentro de un intervalo definido $[-v_{max}, v_{max}]$, evitando fijarlas en cero para mantener la diversidad de exploración. Estas velocidades se presentan como v_i^t .
3. **Evaluación inicial:** se calcula la aptitud (fitness) de cada partícula: aptitud de la posición actual ($pbest_i$) y mejor aptitud individual ($f_{pbest,i}$) registrada.
4. **Actualización de la velocidad:** cada partícula ajusta su velocidad combinando tres componentes: Inercia (tendencia a mantener su dirección actual), componente cognitiva (tendencia a volver a su mejor posición individual), componente social (tendencia a dirigirse hacia la mejor posición del enjambre o del vecindario) (Chauhan & Suganthan, 2025)

La actualización de velocidad viene representada por la siguiente ecuación (2.12):

$$v_i^t = \Omega^{t-1} \cdot v_i^{t-1} + \varphi_1 \cdot rand_1 \cdot (pbest_i - x_i^{t-1}) + \varphi_2 \cdot rand_2 \cdot (gbest - x_i^{t-1}) \quad (2.12)$$

Para $i = 1, 2, 3, \dots, P$

Donde:

x_i^t : vector posición de la partícula i en la iteración t

v_i^t : vector de velocidad de la partícula i en la iteración t

Ω^{t-1} : factor de inercia en la iteración t

φ_1, φ_2 : pesos que controlan los componentes cognitivos (guía a la partícula según su mejor experiencia personal) y social (representa la influencia del resto del enjambre, dirigiendo a la partícula hacia la mejor posición global)

$rand_1$: número aleatorio entre 0 y 1.

$rand_2$: número aleatorio entre 0 y 1.

$pbest_i$: mejor posición encontrada por la partícula i hasta el momento que posee la mejor solución

$gbest$: representa la posición de la partícula con la mejor solución o aptitud de la nube de partículas.

5. **Actualización de posición:** en cada iteración, las partículas se desplazan en el espacio de búsqueda sumando su vector de velocidad a la posición actual, según (2.13):

$$x_i^t = x_i^{t-1} + v_i^{t-1} \quad (2.13)$$

6. **Evaluación de la nueva posición:** se recalcula la aptitud de la partícula en su nueva posición y se actualiza $f_i^t = f(x_i^t)$.

7. **Actualización del mejor valor individual (Pbest):** si la nueva aptitud presenta un mejor desempeño a la mejor registrada por la partícula, se actualizan $pbest_i$ y $f_{pbest,i}$.

8. **Actualización del mejor valor global (Gbest):** si el mejor valor individual de alguna partícula es superior al mejor obtenido por todo el enjambre (o el vecindario, en la versión local del PSO), se actualizan $gbest = pbest_i$ y $f_{gbest} = f_{pbest,i}$.

9. **Criterio de parada:** El algoritmo finaliza cuando se alcanza el número máximo de iteraciones establecido o cuando el valor de la mejor aptitud global del enjambre es menor o igual que un umbral de tolerancia predefinido. En caso contrario, el proceso continúa con una nueva actualización de las velocidades y posiciones de las partículas,

seguida de la evaluación de la función de aptitud, con el fin de seguir refinando la solución.

El diagrama de flujo (Figura 7) presentado resume el funcionamiento general del algoritmo PSO. En cada iteración, primero se actualizan las velocidades de las partículas utilizando la información individual (PBest) y global (GBest), posteriormente se actualizan las posiciones, se evalúa la nueva aptitud y se actualizan los valores de referencia, repitiendo el proceso hasta cumplir el criterio de parada.

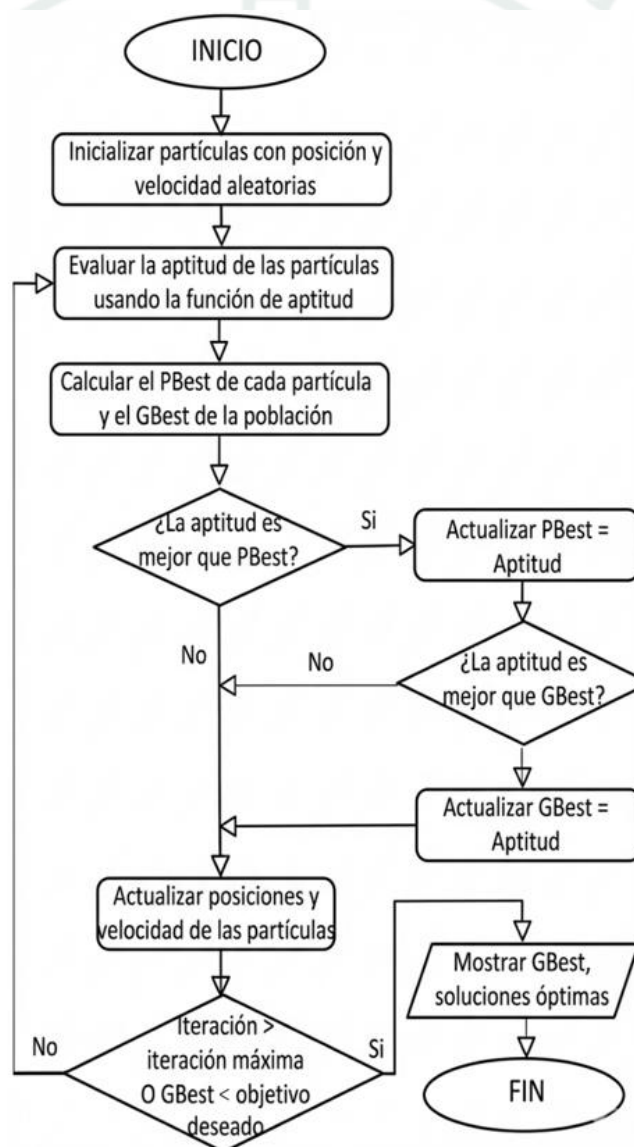


Figura 7. Diagrama de flujo del algoritmo PSO – Elaboración propia

Factor de inercia dinámico

El factor de inercia (Ω) desempeña un papel importante en el equilibrio entre exploración global y explotación local del algoritmo PSO. Valores elevados de (Ω) favorecen la exploración del espacio de búsqueda, mientras que valores reducidos promueven la convergencia hacia soluciones óptimas (Mohammadi & Akhavan, 2014).

Una estrategia común consiste en variar dinámicamente el factor de inercia de forma decreciente según la ecuación (2.14):

$$\Omega^t = \Omega_{max} - \frac{\Omega_{max} - \Omega_{min}}{t_{max}} \cdot t \quad (2.14)$$

Donde Ω_{max} y Ω_{min} representan los valores inicial y final del factor de inercia, respectivamente, t y t_{max} representan el número de iteración actual y número de iteraciones máximo

La Figura 8 ilustra el movimiento de una partícula dentro del espacio de soluciones.

- Las flechas de línea discontinua indican las direcciones correspondientes a los componentes cognitivo y social.
- La flecha punteada discontinua representa la velocidad actual de la partícula.
- La flecha de línea continua muestra la nueva dirección de movimiento, resultante de la combinación de todos los componentes, que lleva a la partícula desde su posición actual hasta la siguiente.

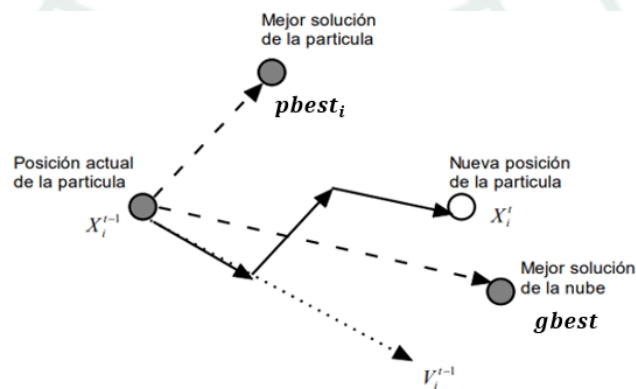


Figura 8. Representación del movimiento de una partícula (Chauhan & Suganthan, 2025)

2.3.4 Clases de entorno en PSO

La forma en que una partícula interactúa con las demás depende del tipo de entorno, que puede ser:

- Geográfico: se basa en la distancia espacial entre partículas; cada partícula interactúa con las más cercanas.
- Social: define un conjunto fijo de vecinos para cada partícula, sin considerar su posición en el espacio. Este enfoque es el más utilizado.

Cuando el entorno incluye a toda la población, se considera simultáneamente geográfico y social.

2.3.5 Tipos de PSO

Existen distintas variantes del algoritmo PSO según sus configuraciones, como, por ejemplo:

- Según la ponderación relativa de los componentes cognitivo y social

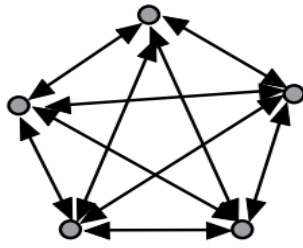
Completo: $\varphi_1, \varphi_2 > 0$.Tanto el componente cognitivo como el social intervienen

Cognitivo: $\varphi_1 > 0$ y $\varphi_2 = 0$ Solo interviene el componente cognitivo.

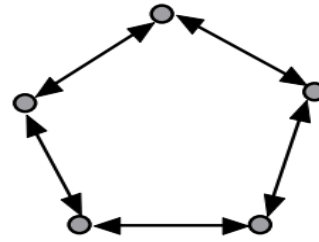
Social: $\varphi_1 = 0$ y $\varphi_2 > 0$ Solo interviene el componente social en el movimiento.

La forma en que las partículas interactúan socialmente depende del tipo de vecindario definido:

- **PSO Global:** Todas las partículas están conectadas entre sí y comparten información con un líder global. Cada partícula conoce la mejor solución encontrada por todo el enjambre.
- **PSO Local:** El enjambre se divide en varios vecindarios de igual tamaño. Cada vecindario tiene su líder local, y las partículas solo se comunican con sus vecinos cercanos, sin acceso a la información del resto del enjambre. La Figura 9 ilustra estas dos configuraciones:



(a) Comunicación global



(b) Comunicación local (vecindarios 3 partículas)

Figura 9. Modelos de comunicación básicos en PSO (Chauhan & Suganthan, 2025).

2.4 Introducción al control vectorial de motor de inducción

El control orientado al campo (FOC) constituye una estrategia avanzada que permite regular un motor de inducción de forma análoga al comportamiento de un motor de corriente continua con excitación independiente. Bajo este enfoque, las variables responsables de la generación de flujo magnético y de torque electromagnético se regulan de manera desacoplada, lo que mejora significativamente el desempeño dinámico del sistema.

En esta técnica, la corriente del estator actúa como magnitud principal de control y se transforma en un sistema de referencia rotatorio, separándose en dos componentes perpendiculares. La componente i_d se encarga de establecer el flujo en la máquina, mientras que la componente i_q gobierna directamente la producción de torque, permitiendo un control preciso e independiente de ambas magnitudes (Liu, Chau, Lee, & Song, 2021).

Dependiendo del método empleado para determinar el ángulo de transformación, el control FOC puede clasificarse en dos modalidades. En el FOC directo, el vector de flujo se obtiene a partir de variables eléctricas medidas del sistema. En contraste, el FOC indirecto calcula dicho vector mediante la velocidad del rotor y el deslizamiento estimado.

Una estructura típica de control FOC, como se ilustra en la Figura 10, está compuesta por un sistema de control en cascada. En este esquema, dos controladores PI internos regulan las corrientes del estator, mientras que un controlador PI externo se encarga de ajustar la velocidad angular ω . Debido a que el inversor de potencia opera con estados discretos, las

referencias de tensión continua (u_d, u_q) deben convertirse en señales de conmutación. Este proceso se realiza mediante técnicas de modulación por ancho de pulso (PWM), siendo la modulación espacial por vectores (SVPWM) la opción más utilizada en aplicaciones con motores de inducción, ya que ofrece un mejor aprovechamiento del voltaje del bus de continua y menores distorsiones armónicas.

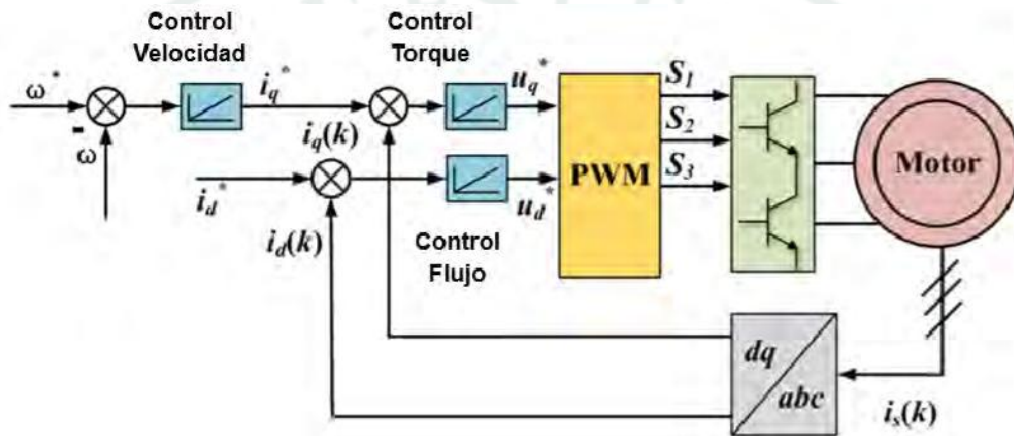


Figura 10. Esquema de control de campo orientado (FOC) (Liu, Chau, Lee, & Song, 2021)

2.5 Filtro de Kalman extendido

2.5.1 Introducción al Filtro de Kalman

Para la implementación de un control tolerante a fallas en los sensores de velocidad y corriente, es indispensable disponer de mecanismos de estimación confiables que sustituyan la información perdida cuando dichos sensores fallan o presentan mediciones imprecisas. En el caso de la velocidad, una estrategia ampliamente utilizada consiste en el empleo de observadores de estado, ya que permiten reconstruir las variables internas del sistema a partir de señales medibles y de un modelo matemático del motor. Entre ellos, el Filtro de Kalman Extendido (EKF, por sus siglas en inglés) destaca por su capacidad para manejar la naturaleza no lineal de las máquinas de inducción y por su robustez frente a perturbaciones y ruido de medición.

Esta aproximación es de gran relevancia en esquemas de control tolerante a fallas, donde la redundancia basada en modelos incrementa la confiabilidad del sistema.

Por estas razones, se incluye en este trabajo una breve introducción al Filtro de Kalman y a su variante extendida, con el fin de fundamentar su aplicación en la estimación de variables de estado críticas para el control de máquinas de inducción.

El filtro de Kalman, propuesto por Rudolf E. Kalman en 1960, es considerado uno de los avances más significativos en la ingeniería matemática. Este algoritmo permite estimar el estado de un sistema dinámico lineal a partir de observaciones ruidosas y modelos matemáticos del proceso, resolviendo el problema cuadrático lineal de manera óptima y recursiva. Su implementación computacional eficiente lo ha convertido en una herramienta fundamental en una amplia gama de aplicaciones, desde navegación por GPS y robótica, hasta sistemas de control en aeronáutica y automatización industrial. (Salas Camacho, 2022).

En su forma clásica, el Filtro de Kalman asume que tanto el modelo del sistema como los ruidos de proceso y medición son lineales y gaussianos. El ruido de proceso está asociado a las incertidumbres internas del modelo, mientras que el ruido de medición proviene de imprecisiones en los sensores. A pesar de sus potentes capacidades, este enfoque es limitado cuando se enfrenta a sistemas no lineales, lo cual es común en entornos reales.

Para abordar estos desafíos, se desarrolló el Filtro de Kalman Extendido (EKF, por sus siglas en inglés), una variante que adapta el algoritmo original para sistemas dinámicos no lineales mediante la linealización local del modelo en cada paso del tiempo. Aunque su implementación es más compleja, el EKF mantiene la estructura recursiva y proporciona estimaciones precisas incluso en presencia de perturbaciones significativas y modelos no lineales.

Una de las aplicaciones destacadas del EKF es en el control y estimación en motores eléctricos, particularmente en motores de inducción. En estos casos, el EKF permite estimar en

tiempo real la velocidad del rotor a partir de mediciones ruidosas de corrientes y tensiones del estator, ofreciendo un rendimiento robusto incluso a bajas velocidades, donde otros métodos fallan.

Este trabajo se enfoca en el uso del Filtro de Kalman y su versión extendida para la estimación de la velocidad del rotor en motores de inducción, explorando su teoría, implementación y ventajas en entornos prácticos donde el modelado preciso y la adaptación a condiciones variables son esenciales.

2.5.2 Conceptos importantes

Variables deterministas y aleatorias

En el análisis y modelado de sistemas físicos o de ingeniería, las variables pueden clasificarse en dos categorías fundamentales: deterministas y aleatorias.

Variable determinista: es aquella cuyo valor puede predecirse exactamente en cualquier instante del tiempo. Su comportamiento está completamente definido por una ecuación o relación funcional conocida. Ejemplo: la posición de un actuador controlado por un sistema de lazo cerrado ideal.

Variable aleatoria: es una magnitud cuyo valor no puede determinarse con certeza antes de realizar una observación o experimento. En lugar de un único valor, se describe mediante una distribución de probabilidad, que indica qué tan probable es que tome ciertos valores.

Se denota usualmente por una letra mayúscula, por ejemplo, X , mientras que una realización o valor específico se representa en minúscula, x (Kovvali & Banavar, 2022)

La probabilidad de que la variable aleatoria adopte un valor particular se expresa como:

$p(X = x)$ o simplificada $p(x)$

Momentos estadísticos

Los momentos estadísticos describen propiedades fundamentales de una variable aleatoria, tales como su tendencia central, dispersión y forma de la distribución. Los más comunes son el valor esperado (media) y la varianza.

Media o valor esperado

La media representa el valor promedio que se espera obtener al observar la variable aleatoria un número infinito de veces. Es una medida de tendencia central y se interpreta como el “centro de gravedad” de la distribución de probabilidad (Grewal & Andrews, 2014).

Para una variable discreta con posibles valores x_i y probabilidades asociadas $p_i = P(X = x_i)$, se define por la ecuación (2.15):

$$E[X] = \sum_i x_i p(x_i) \quad (2.15)$$

Para una variable aleatoria continua, el valor esperado se obtiene mediante la integral ponderada de todos los valores posibles por su función de densidad de probabilidad $p(x)$.

$$\mu = E[X] = \int_{-\infty}^{\infty} xp(x)dx \quad (2.16)$$

Varianza y desviación estándar

La varianza mide el grado de dispersión o variabilidad de la variable aleatoria respecto a su media. Cuantifica cuánto se alejan los valores de la media en promedio.

Se define según la ecuación (2.17):

$$\sigma^2 = Var[X] = E[(X - \mu)^2] = \int_{-\infty}^{\infty} (x - \mu)^2 p(x) dx \quad (2.17)$$

La desviación estándar σ es la raíz cuadrada de la varianza y representa la medida de dispersión en las mismas unidades de la variable original.

Funciones de Probabilidad

Existen diferentes formas de representar la probabilidad de una variable aleatoria:

La función de probabilidad (PMF) para variables discretas se representa como (2.18):

$$p(x_i) = P(X = x_i) \quad (2.18)$$

Función de densidad de probabilidad (PDF) para variables continuas:

$$p(x) \geq 0, \quad \int_{-\infty}^{\infty} p(x) dx = 1 \quad (2.19)$$

La probabilidad de que X esté entre dos valores a y b se calcula como:

$$P(a \leq X \leq b) = \int_a^b p(x) dx \quad (2.20)$$

La función de distribución acumulada (CDF) se representa como la ecuación (2.21):

$$F(x) = P(X \leq x) = \int_{-\infty}^x p(t) dt \quad (2.21)$$

Distribución Normal o Gaussiana

Una de las distribuciones más importantes en ingeniería y control es la distribución normal o gaussiana, debido a que muchos fenómenos físicos y errores de medición siguen este comportamiento.

Su función de densidad de probabilidad está dada por la ecuación (2.22):

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma^2} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \quad (2.22)$$

Teniendo como propiedades principales

$p(x) \geq 0$ para todo x

$$\int_{-\infty}^{\infty} p(x) dx = 1 \quad (2.23)$$

Es simétrica respecto a $x = \mu$

A continuación, se presenta Función de densidad probabilidad (PDF) Gaussiana para el caso de una variable para diferentes valores de media y varianza.

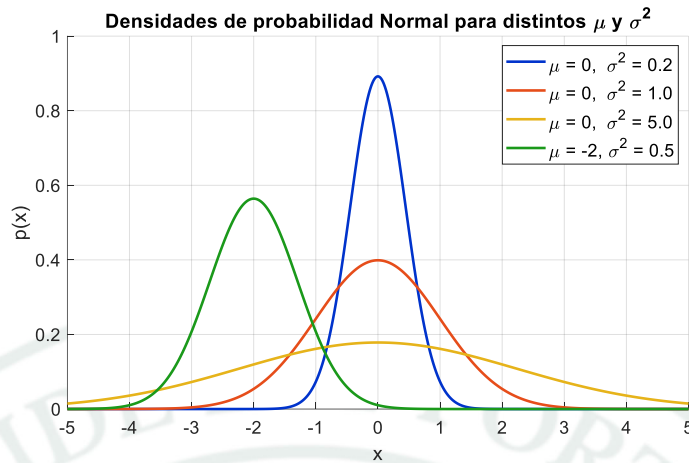


Figura 11. Densidades de probabilidad Normal para distintos μ y σ^2 - Elaboración propia

Transformación lineal de variables aleatorias

Las transformaciones lineales permiten escalar, trasladar o normalizar los datos manteniendo la estructura estadística del fenómeno original (Grewal & Andrews, 2014).

Sea una variable aleatoria continua X con distribución normal:

$$X \sim N(\mu, \sigma^2) \quad (2.24)$$

Y definiendo una transformación línea del tipo

$$Y = aX + b \quad (2.25)$$

donde:

a es un coeficiente de escala (afecta la dispersión),

b es un término de desplazamiento (afecta la posición o media).

Entonces, la variable transformada Y también sigue una distribución normal:

$$Y \sim N(a\mu + b, a^2\sigma^2) \quad (2.26)$$

A partir de la definición de valor esperado y varianza:

$$E[Y] = E[aX + b] = aE[X] + b = a\mu + b \quad (2.27)$$

$$Var[Y] = E[(Y - E[Y])^2] = a^2Var[X] = a^2\sigma^2 \quad (2.28)$$

Vector de variables aleatorio

En la práctica, muchas variables aleatorias se presentan de forma conjunta o correlacionada. Por ejemplo, la temperatura, la presión y la velocidad del viento pueden representarse como un vector aleatorio:

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_i \\ \vdots \\ X_r \end{bmatrix} \in \mathbb{R}^r \quad (2.29)$$

donde cada X_i es una variable aleatoria unidimensional y r representa la dimensión del vector.

Distribución normal multivariada

Si el vector aleatorio X sigue una distribución normal de dimensión r , se denota como:

$$X \sim N(\mu, \Sigma) \quad (2.30)$$

donde, $\mu \in \mathbb{R}^r$ es el vector de medias y matriz de covarianza $\Sigma \in \mathbb{R}^{r \times r}$, definida como:

$$\mu = \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_i \\ \vdots \\ \mu_r \end{bmatrix} \quad \Sigma = \begin{bmatrix} \sigma_{11}^2 & \sigma_{12}^2 & \cdots & \sigma_{1r}^2 \\ \sigma_{21}^2 & \sigma_{22}^2 & \cdots & \sigma_{2r}^2 \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{r1}^2 & \sigma_{r2}^2 & \cdots & \sigma_{rr}^2 \end{bmatrix} \quad (2.31)$$

donde cada elemento $\sigma_{ij} = Cov(X_i, X_j)$ representa la covarianza o el grado de correlación lineal entre las componentes X_i y X_j .

Cuando se tienen dos variables aleatorias X y Y , la covarianza cuantifica cómo varían conjuntamente:

$$Cov(X, Y) = E[(X - \mu_X)(Y - \mu_Y)] \quad (2.32)$$

Si $Cov(X, Y) > 0$, ambas variables tienden a aumentar o disminuir juntas

Si $Cov(X, Y) < 0$, una tiende a aumentar mientras la otra disminuye.

Si $Cov(X, Y) = 0$, las variables son no correlacionadas (aunque no necesariamente independientes).

Cuando $X = Y$, la covarianza se reduce a la varianza de X .

Función de densidad de probabilidad (PDF) Gaussiana: Caso multivariable

La función de densidad de probabilidad de una distribución normal multivariada está dada por:

$$p(x) = \frac{1}{\sqrt{(2\pi)^r |\Sigma|}} e^{\frac{-(x-\mu)^T \Sigma^{-1} (x-\mu)}{2}} \quad (2.33)$$

Donde:

$|\Sigma|$ es el determinante de la matriz de covarianza (proporcional al volumen del elipsoide de probabilidad), Σ^{-1} es la matriz de precisión, que define la orientación y la forma de las curvas de nivel (elipsoides) de la densidad (Grewal & Andrews, 2014).

Propiedades importantes

Si $X \sim N(\mu, \Sigma)$ y $Y = AX + b$, donde A es una matriz constante y b un vector de desplazamiento, entonces:

$$Y \sim N(A\mu + b, A\Sigma A^T) \quad (2.34)$$

Esta propiedad generaliza la transformación lineal unidimensional al caso multivariable.

Si las variables son independientes, la matriz de covarianza Σ es diagonal.

A continuación, en la Figura 12 se muestra la función de densidad de probabilidad (PDF) Gaussiana: Caso multivariable

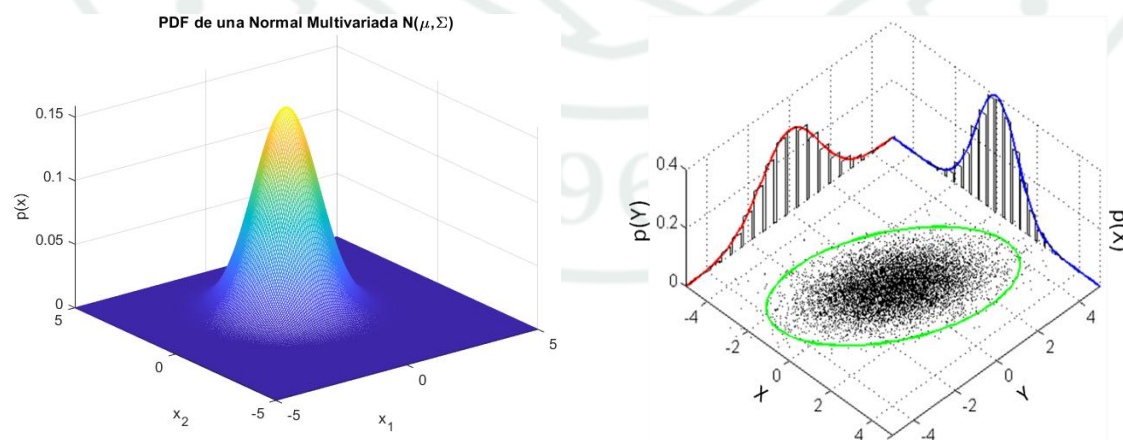


Figura 12. Representación de PDF Gaussiana Multivariable: Superficie de densidad y proyecciones marginales con elipsoide de contorno (Grewal & Andrews, 2014).

2.5.3 Filtro de Kalman discreto

El Filtro de Kalman es un observador de estados óptimo, también conocido como Estimación Cuadrática Lineal (LQE), diseñado para sistemas lineales. Su fortaleza radica en considerar el carácter estadístico del ruido, combinando lecturas estimadas y medidas reales mediante distribuciones de probabilidad conjunta para obtener una estimación óptima del estado. En lugar de trabajar directamente con los errores, el filtro se enfoca en su covarianza, que representa la incertidumbre de estimación y predicción. Así, la incertidumbre se modela como una densidad de probabilidad, permitiendo que las estimaciones previas influyan en las nuevas de forma probabilísticamente coherente (Salas Camacho, 2022).

El diagrama de bloques del Filtro de Kalman dentro de un sistema de control con realimentación y sus diferentes elementos:

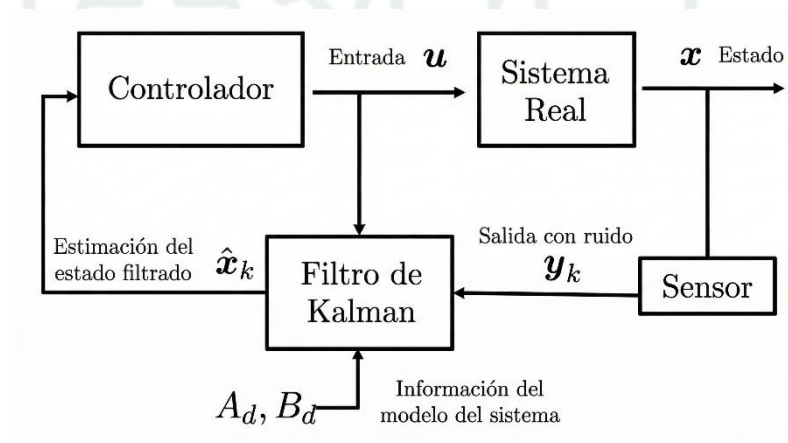


Figura 13. Diagrama de bloques de implementación de Filtro de Kalman

Cada bloque cumple una función específica en el proceso de estimación y control del estado:

- Sistema real: representa el proceso físico o planta cuyo estado real x se desea conocer.
- Sensor: mide una salida y_k , la cual está afectada por ruido de medición.
- Filtro de Kalman: recibe la señal medida y_k y la entrada del sistema u , junto con la información del modelo del sistema (A_d, B_d, H), para estimar el estado más probable del sistema $\hat{x}_k$.

- Controlador: utiliza la estimación del estado filtrado $\hat{x}_k$ para generar la señal de control u_k que actúa sobre el sistema real.

En la mayoría de los problemas de control, es preferible utilizar modelos en espacio de estados para representar sistemas dinámicos, idealmente basados en las leyes físicas del entorno real. Esto permite describir sistemas lineales que pueden variar con el tiempo de forma estructurada y precisa.

El modelo discreto de un sistema con ruido se expresa como (modelo estocástico):

$$x_k = A_d x_{k-1} + B_d u_{k-1} + w_{k-1} \quad (2.35)$$

$$y_k = H x_k + v_k \quad (2.36)$$

$$y_k \in \mathbb{R}^m \text{ salida}$$

$$w_k \in \mathbb{R}^n: w_k \sim N(0, Q)$$

$$v_k \in \mathbb{R}^m: v_k \sim N(0, R)$$

$$Q \in \mathbb{R}^{n \times n} \quad R \in \mathbb{R}^{m \times m} \text{ matriz de covarianza}$$

Donde:

x_k : vector de estados

u_k : vector de entrada

y_k : vector de salida o salida de sensor

A_d : matriz de transición de estados discreto

B_d : matriz de entrada discreto

H : matriz de salida

w_k : Es el ruido del proceso, que representa las incertidumbres del modelo físico, como variaciones por presión atmosférica, fricción, etc.

v_k : Es el ruido de medición, propio del sensor, derivado de sus limitaciones físicas y electrónicas.

Por lo general, se asume que w_k y v_k son procesos de ruido blanco gaussiano, independientes entre sí. Esto significa que ambos tienen media cero, y sus niveles de

incertidumbre se representan mediante matrices de covarianza: Q_k para el ruido de proceso y R_k para el ruido de medición.

1. Covarianza del ruido de proceso:

$$\text{cov}[w_j, w_k] = E[w_j w_k^T] = \begin{cases} 0 & \text{si } j \neq k \\ Q_k & \text{si } j = k \end{cases}$$

2. Covarianza del ruido de medición

$$\text{cov}[v_j, v_k] = E[v_j v_k^T] = \begin{cases} 0 & \text{si } j \neq k \\ R_k & \text{si } j = k \end{cases}$$

Si Q_k es muy inferior a R_k , las predicciones de estado solo se basarán en las predicciones del modelo y se ignorarán las mediciones. Si R_k es muy inferior a Q_k las predicciones de estado solo se basarán en las mediciones y las predicciones del modelo se ignorarán.

En muchos casos, el problema de estimación se inicia sin información a priori. Si se asume que el proceso tiene media cero, la estimación inicial también será nula, y la matriz de covarianza inicial del error coincide con la covarianza del estado.

La etapa de predicción (a priori) se describe mediante el modelo discreto del sistema:

$$\hat{x}_{k|k-1} = A_d \hat{x}_{k-1|k-1} + B_d u_k \quad (2.37)$$

Una vez obtenida la estimación a priori, se procede a la etapa de corrección (a posteriori), donde se incorpora la medición disponible y_k mediante una combinación lineal ponderada por la ganancia del filtro de Kalman:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k (y_k - H \hat{x}_{k|k-1}) \quad (2.38)$$

Donde K_k es la ganancia del filtro de Kalman, la cual determina el peso relativo entre el modelo del sistema y las mediciones.

El Filtro de Kalman permite estimar estados que no pueden medirse directamente, combinando de forma óptima el modelo dinámico del sistema con las mediciones disponibles. Su funcionamiento se basa en dos etapas: predicción y corrección (Salas Camacho, 2022).

Los errores de estimación en las fases a priori y a posteriori se definen como:

$$e_{k|k-1} = x_k - \hat{x}_{k|k-1} \quad (2.39)$$

$$e_{k|k} = x_k - \hat{x}_{k|k} \quad (2.40)$$

Las matrices de covarianza asociadas a estos errores caracterizan la incertidumbre de las estimaciones obtenidas en cada etapa.

$$P_{k|k-1} = E[e_{k|k-1} e_{k|k-1}^T] = \begin{bmatrix} E[(e_{1,k|k-1})^2] & E[e_{1,k|k-1} e_{2,k|k-1}] & \cdots & E[e_{1,k|k-1} e_{n,k|k-1}] \\ E[e_{2,k|k-1} e_{1,k|k-1}] & E[(e_{2,k|k-1})^2] & \cdots & E[e_{2,k|k-1} e_{n,k|k-1}] \\ \vdots & \vdots & \ddots & \vdots \\ E[e_{n,k|k-1} e_{1,k|k-1}] & E[e_{n,k|k-1} e_{2,k|k-1}] & \cdots & E[(e_{n,k|k-1})^2] \end{bmatrix}$$

$$P_{k|k} = E[e_{k|k} e_{k|k}^T] = \begin{bmatrix} E[(e_{1,k|k})^2] & E[e_{1,k|k} e_{2,k|k}] & \cdots & E[e_{1,k|k} e_{n,k|k}] \\ E[e_{2,k|k} e_{1,k|k}] & E[(e_{2,k|k})^2] & \cdots & E[e_{2,k|k} e_{n,k|k}] \\ \vdots & \vdots & \ddots & \vdots \\ E[e_{n,k|k} e_{1,k|k}] & E[e_{n,k|k} e_{2,k|k}] & \cdots & E[(e_{n,k|k})^2] \end{bmatrix}$$

Los elementos fuera de la diagonal de la matriz de covarianza representan errores correlacionados entre distintos estados, mientras que los elementos de la diagonal principal corresponden al error cuadrático medio de cada estado. El objetivo del filtro de Kalman es minimizar este error total esperado, asumiendo ruidos gaussianos de media cero.

Para minimizar estos errores se formula la siguiente expresión que relaciona la ganancia de Kalman (2.39) con la covarianza del error:

$$P_{k|k} = E \left[(x_k - \hat{x}_{k|k-1} - K_k(y_k - H\hat{x}_{k|k-1})) (x_k - \hat{x}_{k|k-1} - K_k(y_k - H\hat{x}_{k|k-1}))^T \right] \quad (2.41)$$

Reemplazando las ecuaciones (2.36) y (2.39) en (2.41):

$$P_{k|k} = E \left[(e_{k|k-1} - K_k(H x_k + v_k - H \hat{x}_{k|k-1})) (e_{k|k-1} - K_k(H x_k + v_k - H \hat{x}_{k|k-1}))^T \right] \quad (2.42)$$

Simplificando la expresión de la ecuación (2.42)

$$P_{k|k} = E \left[((I - K_k H)e_{k|k-1} - K_k v_k) ((I - K_k H)e_{k|k-1} - K_k v_k)^T \right]$$

$$P_{k|k} = (I - K_k H) E [e_{k|k-1} (e_{k|k-1})^T] (I - K_k H)^T + K_k E[v_k v_k^T] K_k^T \quad (2.43)$$

Ahora, reemplazando las matrices de covarianza en (2.43):

$$P_{k|k} = (I - K_k H)P_{k|k-1}(I - K_k H)^T + K_k R_k K_k^T \quad (2.44)$$

Expandiendo la forma de Joseph para análisis:

$$P_{k|k} = P_{k|k-1} - K_k H P_{k|k-1} - P_{k|k-1} H^T K_k^T + K_k (H P_{k|k-1} H^T + R_k) K_k^T \quad (2.45)$$

Definiendo

$$S_k = H P_{k|k-1} H^T + R_k$$

La ecuación (2.45) es la expresión de la covarianza del error válida para cualquier ganancia K_k . Dado que los elementos de la diagonal principal de la matriz de covarianza del error a posteriori representan el error cuadrático medio de cada estado, el objetivo del filtro de Kalman es minimizar el error total esperado. Esto es equivalente a minimizar la traza de dicha matriz de covarianza con respecto a la ganancia de Kalman.

$$J(K_k) = \text{tr}(P_{k|k}), \quad \min_{K_k} J(K_k)$$

Aplicando el operador traza

$$\text{tr}(P_{k|k}) = \text{tr}(P_{k|k-1}) - \text{tr}(K_k H P_{k|k-1}) - \text{tr}(P_{k|k-1} H^T K_k^T) + \text{tr}(K_k S_k K_k^T) \quad (2.46)$$

Teniendo en cuenta las propiedades de la traza: invarianza por transposición $\text{tr}(A) = \text{tr}(A^T)$ y propiedad cíclica $\text{tr}(ABC) = \text{tr}(BCA) = \text{tr}(CAB)$

Aplicando propiedades en el tercer termino de (2.46) se obtiene

$$\text{tr}(P_{k|k-1} H^T K_k^T) = \text{tr}((K_k H P_{k|k-1})^T) = \text{tr}(K_k H P_{k|k-1})$$

Reemplazando este término en la ecuación (2.46)

$$\text{tr}(P_{k|k}) = \text{tr}(P_{k|k-1}) - 2\text{tr}(K_k H P_{k|k-1}) + \text{tr}(K_k S_k K_k^T)$$

Derivando respecto a K_k , se obtiene

$$\frac{dJ(K_k)}{dK_k} = -2 P_{k|k-1} H^T + 2 K_k S_k$$

Igualando a cero esta última ecuación, reemplazando S_k y despejando K_k

$$K_k = P_{k|k-1} H^T (H P_{k|k-1} H^T + R_k)^{-1} \quad (2.47)$$

Con la ecuación (2.47) se determina la ganancia óptima de Kalman,

Teniendo en cuenta $P_{k|k-1} = P_{k|k-1}^T$ y $R_k = R_k^T$ son simétricas (covarianzas simétricas)

y reemplazando la ganancia de Kalman optima en la ecuación (2.45) se obtiene:

$$\begin{aligned} P_{k|k} &= P_{k|k-1} - P_{k|k-1} H^T S_k^{-1} H P_{k|k-1} - P_{k|k-1} H^T S_k^{-1} H P_{k|k-1} \\ &\quad + P_{k|k-1} H^T S_k^{-1} S_k S_k^{-1} H P_{k|k-1} \\ &\quad S_k^{-1} S_k S_k^{-1} = S_k^{-1} \\ P_{k|k} &= P_{k|k-1} - P_{k|k-1} H^T (H P_{k|k-1} H^T + R_k)^{-1} H P_{k|k-1} \end{aligned} \quad (2.48)$$

Simplificando esta ecuación se obtiene:

$$P_{k|k} = (I - K_k H) P_{k|k-1} \quad (2.49)$$

La siguiente etapa es la de predicción (priori), del estado en el instante k , se reemplaza con (2.35) en la ecuación (2.39), dando como resultado:

$$e_{k|k-1} = (A_d x_{k-1} + B_d u_{k-1} + w_{k-1}) - (A_d \hat{x}_{k-1|k-1} + B_d u_{k-1}) \quad (2.50)$$

Simplificando esta ecuación se obtiene:

$$e_{k|k-1} = A_d e_{k-1|k-1} + w_{k-1} \quad (2.51)$$

Sustituyendo en matriz de covarianza $P_{k|k-1}$

$$P_{k|k-1} = E[(A_d e_{k-1|k-1} + w_{k-1})(A_d e_{k-1|k-1} + w_{k-1})^T] \quad (2.52)$$

Expandiendo (2.53), y asumiendo $E[w_{k-1} w_{k-1}^T] = Q_k$, se obtiene

$$P_{k|k-1} = E[A_d e_{k-1|k-1} e_{k-1|k-1}^T A_d^T + A_d e_{k-1|k-1} w_{k-1}^T + w_{k-1} e_{k-1|k-1}^T A_d^T + Q_k]$$

Debido a que se asume que el ruido del proceso tiene una media de cero o $E[w_{k-1}] = 0$, y no correlacionado con el error entonces, además $P_{k-1|k-1} = e_{k-1|k-1} e_{k-1|k-1}^T$ la estimación a priori de la matriz de covarianza se simplifica a:

$$P_{k|k-1} = A_d P_{k-1|k-1} A_d^T + Q_k \quad (2.53)$$

2.5.4 Algoritmo del Filtro de Kalman

A partir de las ecuaciones previamente obtenidas, se define el algoritmo del Filtro de Kalman. Este algoritmo se representa comúnmente mediante el diagrama mostrado en la Figura 14, donde se distinguen claramente sus dos etapas principales: predicción y corrección.

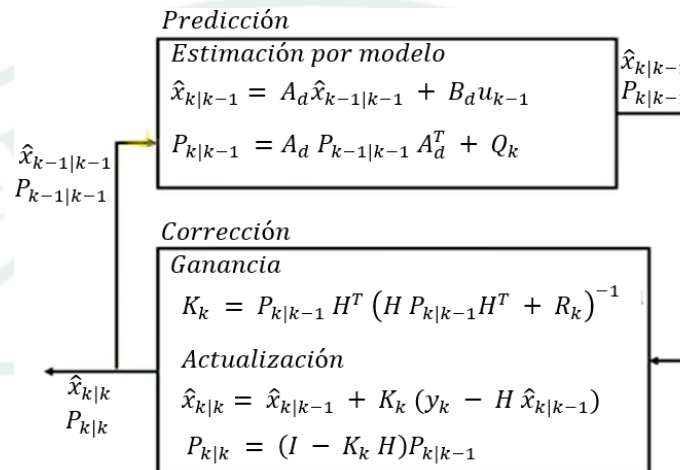


Figura 14. Algoritmo de Filtro de Kalman – Elaboración propia

Etapas de Predicción

En esta fase, el filtro predice el estado futuro del sistema y la incertidumbre asociada basándose únicamente en el modelo matemático. Predicción del estado ($\hat{x}_{k|k-1}$), Predicción de la covarianza ($P_{k|k-1}$). En esta etapa aún no se usan las mediciones reales, solo el modelo del sistema.

Etapas de Corrección

Cuando se dispone de una nueva medición y_k , el filtro la usa para ajustar la predicción previa y obtener una estimación más precisa. Cálculo de la ganancia de Kalman (K_k), Actualización del estado ($\hat{x}_{k|k}$) y actualización de la covarianza ($P_{k|k}$).

Como se explicó anteriormente, el filtro de Kalman es un algoritmo recursivo utilizado para estimar los estados internos de un sistema dinámico que no pueden medirse directamente. Para ello, combina la información del modelo del sistema con las mediciones reales

disponibles, ponderando ambos mediante un factor de ganancia K_k que determina qué fuente de información es más confiable en cada instante.

En la Figura 15, las diferentes campanas representan las funciones de densidad de probabilidad (PDF) asociada a la estimación del estado en cada etapa. El centro de cada distribución indica el valor estimado del estado, mientras que su anchura está directamente relacionada con la covarianza del error P , reflejando el nivel de incertidumbre de la estimación.

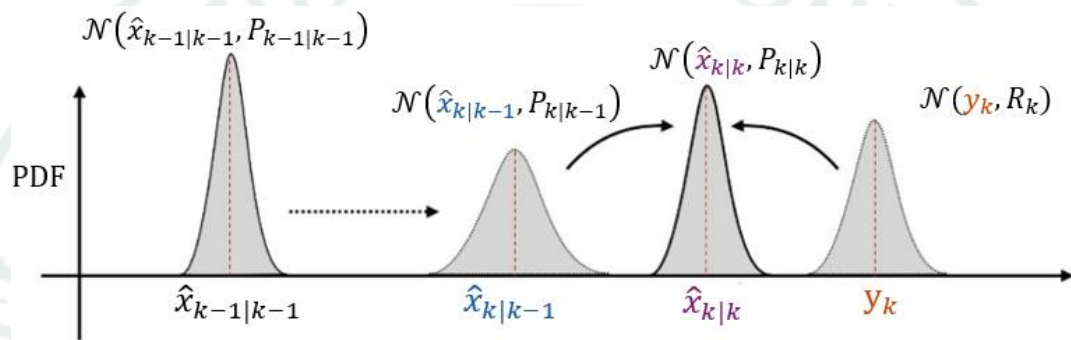


Figura 15. Proceso de predicción y corrección en el Filtro de Kalman (Nadeem, 2020)

A partir del estado estimado inicial o previo $(\hat{x}_{k-1|k-1})$ y la entrada del sistema (u_{k-1}) , el filtro predice el estado siguiente $(\hat{x}_{k|k-1})$ utilizando el modelo matemático del sistema.

Durante esta fase, también se actualiza la covarianza predicha $(P_{k|k-1})$, que refleja la incertidumbre asociada al modelo y al ruido del proceso (Q_k) .

Se observa como un desplazamiento de la campana hacia el nuevo valor esperado y un ensanchamiento de la distribución, indicando que la incertidumbre aumenta cuando se avanza en el tiempo sin una medición nueva.

Cuando se dispone de la medición (y_k) , el filtro compara esta información con la predicción generada por el modelo. La diferencia entre ambas, denominada innovación o residuo, se utiliza para calcular la ganancia de Kalman (K_k) , que determina el peso relativo asignado a la medición y a la predicción.

Como resultado, se obtiene la estimación corregida del estado $\hat{x}_{k|k}$ y la covarianza actualizada ($P_{k|k}$), las cuales se representan gráficamente mediante una distribución más estrecha, indicando una estimación más precisa y una reducción de la incertidumbre.

En resumen, el filtro de Kalman combina la predicción del modelo ($\hat{x}_{k|k-1}$), la medición real (y_k) y la ganancia K_k para obtener una estimación óptima ($\hat{x}_{k|k}$) que minimiza el error cuadrático medio. La Figura 15, ilustra este proceso mostrando cómo las distribuciones de probabilidad se desplazan, combinan y estrechan, reflejando la mejora progresiva de la estimación y la reducción de la incertidumbre con cada iteración.

2.5.5 Filtro de Kalman extendido

Cuando las ecuaciones dinámicas del sistema son no lineales, se puede usar el Filtro de Kalman Extendido (EKF). El EKF opera de forma análoga al filtro de Kalman lineal, pero en cada paso de tiempo realiza una linealización local de las funciones no lineales mediante la expansión de Taylor de primer orden alrededor de la estimación actual del estado (Grewal & Andrews, 2014).

La serie de Taylor de una función escalar $f(x)$ alrededor de un punto x_d es:

$$f(x) = f(x_d) + f'(x_d)(x - x_d) + \frac{f''(x_d)}{2!}(x - x_d)^2 + \dots \quad (2.54)$$

Si se trunca la serie en el primer orden, se obtiene una aproximación lineal válida en un entorno cercano a x_d

$$f(x) \approx f(x_d) + f'(x_d)(x - x_d) \quad (2.55)$$

En la Figura 16 se muestra la función no lineal $f(x_d)$, el punto de expansión $(x_d, f(x_d))$ y su aproximación lineal obtenida mediante la primera derivada evaluada en x_d .

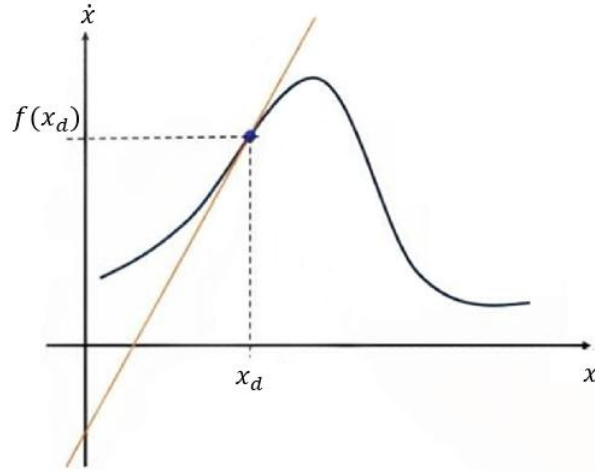


Figura 16. Linealización de una función no lineal mediante expansión de Taylor de primer orden (Nadeem, 2020)

En el caso multivariable, la expansión de Taylor de primer orden de una función vectorial $f: \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ se expresa mediante jacobianos como:

$$f(x, u) = f(x_d, u_d) + \frac{df(x, u)}{dx} \Big|_{\substack{x=x_d \\ u=u_d}} (x - x_d) + \frac{df(x, u)}{du} \Big|_{\substack{x=x_d \\ u=u_d}} (u - u_d) \quad (2.56)$$

Un modelo no lineal puede ser descrito por las siguientes ecuaciones de estado:

$$x_k = f(x_{k-1}, u_{k-1}) + w_k \quad // \text{Ecuación de estado} \quad (2.57)$$

$$y_k = h(x_k) + v_k \quad // \text{Ecuación de medición} \quad (2.58)$$

La función de medición $h(x_k)$ se aproxima mediante una expansión de Taylor de primer orden alrededor de la estimación a priori $\hat{x}_{k|k-1}$:

$$h(x_k) \approx h(\hat{x}_{k|k-1}) + H_k (x_k - \hat{x}_{k|k-1}) \quad (2.59)$$

donde H_k es la matriz Jacobiana de $h(x_k)$, evaluada en $\hat{x}_{k|k-1}$:

$$H_k = \frac{\partial h(x)}{\partial x} \Big|_{x=\hat{x}_{k|k-1}} = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \frac{\partial h_1}{\partial x_2} & \dots & \frac{\partial h_1}{\partial x_n} \\ \frac{\partial h_2}{\partial x_1} & \frac{\partial h_2}{\partial x_2} & \dots & \frac{\partial h_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial h_m}{\partial x_1} & \frac{\partial h_m}{\partial x_2} & \dots & \frac{\partial h_m}{\partial x_n} \end{bmatrix} \quad (2.60)$$

Luego la ganancia de Kalman puede ser calculada empleando la siguiente ecuación:

$$K_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + R_k)^{-1} \quad (2.61)$$

La estimación a posteriori del estado se obtiene mediante:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k (y_k - h(\hat{x}_{k|k-1})) \quad (2.62)$$

De igual forma, la matriz de covarianza a posteriori se actualiza como:

$$P_{k|k} = (I - K_k H_k) P_{k|k-1} \quad (2.63)$$

La función de transición se linealiza alrededor de la estimación a posteriori $\hat{x}_{k-1|k-1}$:

$$f(x_{k-1}, u_{k-1}) \approx f(\hat{x}_{k-1|k-1}, u_{k-1}) + F_{k-1} (x_{k-1} - \hat{x}_{k-1|k-1}) \quad (2.64)$$

Donde F_{k-1} es el jacobiano de $f(x_k, u_k)$ evaluado para $\hat{x}_k$ y u_k

$$F_{k-1} = \left. \frac{\partial f(x, u)}{\partial x} \right|_{x=\hat{x}_{k-1|k-1}, u=u_{k-1}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \dots & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \quad (2.65)$$

La estimación a priori del vector de estados se obtiene aplicando la función de transición evaluada en la estimación a posteriori anterior:

$$\hat{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_{k-1}) \quad (2.66)$$

Finalmente, la matriz de covarianza a priori se calcula como:

$$P_{k|k-1} = F_{k-1} P_{k-1|k-1} F_{k-1}^T + Q_k$$

Las ecuaciones del Filtro de Kalman extendido pueden ser resumidas a:

Etapas de Predicción

$$\hat{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_{k-1}) \quad (2.67)$$

$$P_{k|k-1} = F_{k-1} P_{k-1|k-1} F_{k-1}^T + Q_k \quad (2.68)$$

Etapas de Corrección

$$K_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + R_k)^{-1} \quad (2.69)$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k (y_k - h(\hat{x}_{k|k-1})) \quad (2.70)$$

$$P_{k|k} = (I - K_k H_k) P_{k|k-1} \quad (2.71)$$

2.6 Control Tolerante a Fallas

2.6.1 Introducción al Control Tolerante a Fallas

En los sistemas tecnológicos modernos, cada vez más complejos e interconectados, garantizar un funcionamiento seguro y confiable es esencial, especialmente en aplicaciones críticas como la industria aeroespacial, automotriz o energética. En este contexto, los sistemas de control tolerante a fallas (FTC) surgen como una solución que permite mantener el rendimiento aceptable del sistema incluso ante fallos en sensores, actuadores u otros componentes.

A diferencia del control tradicional, que asume condiciones ideales, el FTC está diseñado anticipando fallas inevitables. Su objetivo es detectar, diagnosticar y compensar estas fallas, evitando que afecten gravemente al sistema. Aunque puede implicar una pérdida parcial de rendimiento, permite continuar la operación de forma segura, protegiendo a las personas, el entorno y los activos (Blanke, Kinnaert, Lunze, & and Staroswiecki, 2016)

La implementación de estrategias FTC suele basarse en modelos del sistema que ayudan a estimar su estado y comportamiento, mejorando tanto el diagnóstico como la reconfiguración del control. Si bien gran parte de la investigación en este campo ha sido teórica, en la actualidad existe una creciente demanda de enfoques prácticos que consideren las limitaciones reales de los sistemas industriales.

Generalmente, para que un sistema sea tolerante a fallos se requieren dos pasos:

1. Diagnóstico de fallos (FDD): Se debe detectar la existencia de fallos e identificación de su tipo y localización.

2. Rediseño del control: El controlador debe adaptarse a la situación de fallo para que el sistema en su conjunto siga cumpliendo su objetivo.

2.6.2 Diagnóstico de fallas basado en modelos

Desde los años 70, el diagnóstico de fallas basado en modelos ha evolucionado notablemente, demostrando su eficacia en múltiples sectores industriales. Su característica principal es el uso de un modelo del proceso para detectar fallas mediante algoritmos que procesan datos en tiempo real (Córdova Claros, 2017).

Existe una relación estrecha entre el diagnóstico basado en modelos y la teoría moderna de control. Además, debido a la necesidad de procesamiento en línea, se requiere el uso de sistemas computacionales potentes, lo cual ha sido posible gracias al desarrollo de la tecnología informática y la teoría de control. Entre las distintas técnicas desarrolladas, destacan aquellas basadas en observadores, que permiten reconstruir variables no medibles del sistema con alta confiabilidad, constituyéndose en uno de los enfoques más utilizados en la literatura.

2.6.3 Técnicas de diagnósticos de fallas

La Figura 17 presenta una clasificación de las diferentes técnicas de diagnóstico de fallas, que abarca los métodos y enfoques utilizados para desarrollar algoritmos de detección y análisis de fallos.

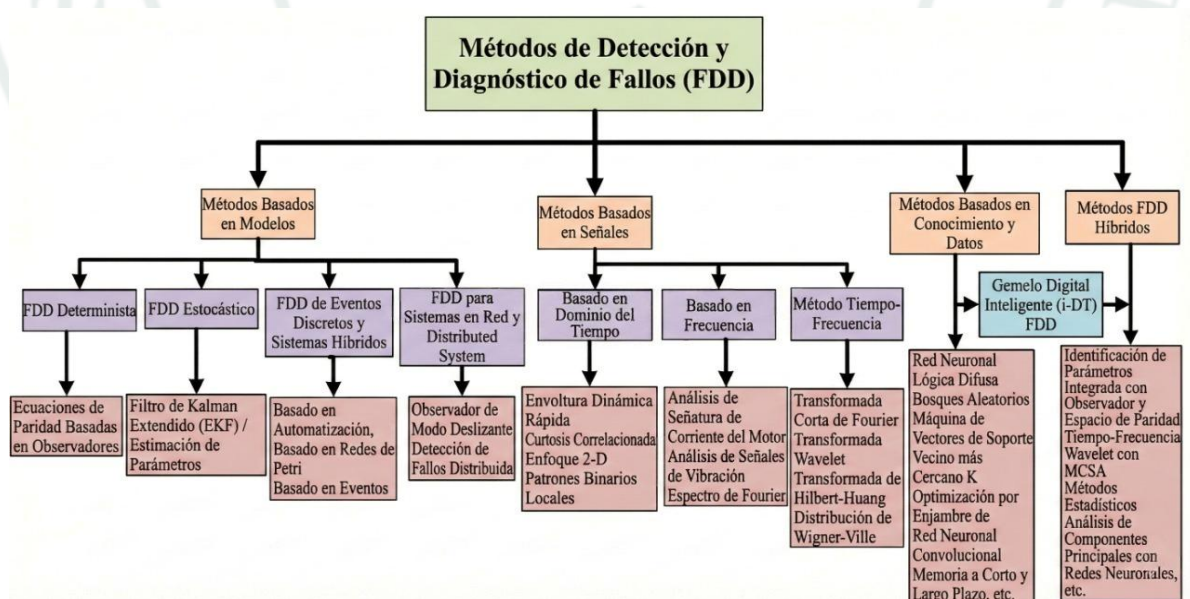


Figura 17. Clasificación de los métodos de Detección y Diagnóstico de Fallos (Kumar, Velpula, Saiprakash, & Sahoo, 2025).

Diagnóstico de fallas por redundancia de hardware: se basa en usar componentes idénticos para comparar salidas y detectar fallas. Ofrece alta confiabilidad y aislamiento directo de fallas, pero su elevado costo limita su uso a componentes esenciales (Blanke, Kinnaert, Lunze, & Staroswiecki, 2016).

Diagnóstico de fallas basado en procesamiento de señales: se basa en analizar síntomas presentes en señales del proceso, como valores medios, tendencias o espectros de frecuencia. Es útil en sistemas en estado estacionario, pero su efectividad disminuye en sistemas dinámicos con señales de entrada variables.

Prueba de plausibilidad: consiste en verificar si un componente del proceso cumple con leyes físicas simples. Si no se cumplen, se asume la presencia de una falla. Es una técnica sencilla, pero limitada para detectar o aislar fallas en procesos complejos.

Diagnóstico de fallas basado en modelos: es un esquema de tipo software o analítico, porque sustituye sensores físicos redundantes por cálculos computacionales que comparan el comportamiento real con el esperado según un modelo matemático.

Estos modelos simulan el comportamiento del sistema en tiempo real. Comparando las salidas reales con las estimadas, se generan residuos (diferencias), los cuales indican posibles fallas si superan ciertos umbrales. Sin embargo, debido a incertidumbres del modelo y perturbaciones, se requiere un procesamiento cuidadoso de estos residuos.

2.6.4 Métodos de FDD basado en modelo

A diferencia de la redundancia de hardware, los métodos de diagnóstico basados en modelos utilizan ecuaciones que representan el comportamiento dinámico del sistema, implementadas mediante software. Las predicciones del modelo se comparan con las mediciones reales para generar residuales, que deberían ser cercanos a cero en condiciones normales. Este enfoque se divide en dos etapas.

Generación de residuales: que depende de la precisión del modelo utilizado

Evaluación de residuales: donde se analizan su magnitud y características para identificar la localización y severidad de la falla.

En la Figura 18 se muestra esquema del diagnóstico de falla basado en modelos

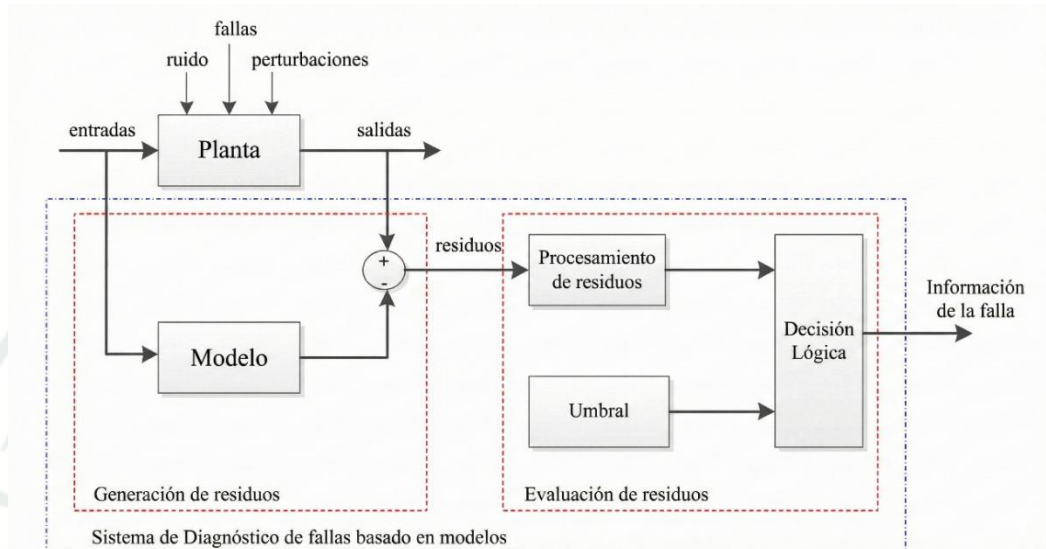


Figura 18. Sistema de diagnóstico de falla basado en modelos (Córdova Claros, 2017)

Según Steven X. Ding (2021), los métodos de generación de residuales se clasifican en tres grupos principales.

2.6.4.1 Espacio de Paridad

Las relaciones de paridad se basan en la comparación de señales redundantes, ya sean estáticas o dinámicas, para detectar inconsistencias indicativas de fallas. Este método permite generar residuos sin requerir un modelo completo del sistema y ha sido ampliamente utilizado en aplicaciones como la navegación inercial.

2.6.4.2 Estimación de parámetros

Este enfoque se basa en la Identificación de Sistemas, donde las fallas afectan directamente los parámetros físicos del sistema (como fricción, masa o resistencias). La detección se logra comparando parámetros identificados en línea con los de referencia sin falla. Actualmente, este método se complementa con técnicas de inteligencia artificial (redes

neuronales, lógica difusa, etc.), que permiten modelar y detectar fallas a partir de datos, especialmente en sistemas complejos.

2.6.4.3 Observadores

Este método es especialmente adecuado cuando las fallas producen cambios físicos en sensores, actuadores o componentes no directamente medibles, y se dispone de un modelo matemático confiable del sistema. Los observadores permiten estimar los estados no medibles y generar residuos a partir de la comparación con las mediciones reales. Entre los observadores más utilizados se encuentran los de entradas desconocidas (UIO), adaptativos (AO), de alta ganancia (HGO), de modos deslizantes (SMO) y el filtro de Kalman (KF) en entornos estocásticos. Su principal ventaja es la flexibilidad en el diseño y la posibilidad de implementar bancos de observadores para mejorar la detección y el aislamiento de fallas.

2.6.5 Clasificación de sistema de control tolerante a fallas

El FTC generalmente se divide en métodos pasivos y activos. Los métodos pasivos se basan en un control robusto. Los métodos activos se pueden subdividir en diseño de redundancia, adaptación a fallos y reconfiguración del controlador o sistema. La Figura 19 muestra las principales categorías de controles tolerantes a fallos.

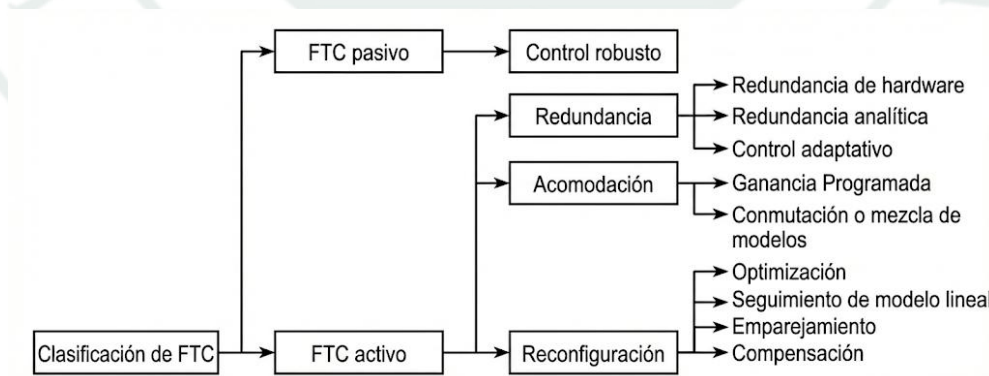


Figura 19. Clasificación de las estrategias tolerante a fallos – Elaboración propia

2.6.5.1 Sistemas de control tolerante a fallos pasivos (PFTC)

El PFTC no se basa en la información de fallas del sistema, sino que considera la peor situación posible al inicio del diseño y reserva un margen suficiente, utilizando un compensador robusto para reducir el impacto de las fallas o estabilizar el sistema. Algunos métodos estándar, como el control H-infinito, el control de modo deslizante y el control cuadrático lineal, se utilizan en algunos campos industriales. Sin embargo, la estrategia de control robusto del FTC pasivo se basa en el supuesto de que el sistema se mantiene en un lazo cerrado asintóticamente estable, lo cual es insuficiente para gestionar situaciones donde el sistema experimenta fallas masivas e impredecibles.

2.6.5.2 Sistemas de control tolerante a fallos activos (AFTC)

Responden de forma específica a cada fallo, basándose en la información del sistema de detección y en la estrategia de control utilizada. Su diseño incluye tres etapas principales: detección, supervisión y control.

Para que un FTC activo sea eficiente, debe cumplir tres condiciones clave:

1. Alta precisión en la detección de fallos, ya que errores afectan directamente el desempeño y la estabilidad.
2. Robustez frente a información de detección imperfecta.
3. Rapidez en la reconfiguración del control para actuar antes de que la falla afecte críticamente al sistema.

El componente más importante es la unidad de detección y diagnóstico de fallos (FDD), por lo que los FTC activos suelen clasificarse según el tipo de FDD que emplean, aunque los métodos de compensación del fallo pueden variar entre diseños.

La Figura 20 muestra la estructura general del FTC activo y pasivo.

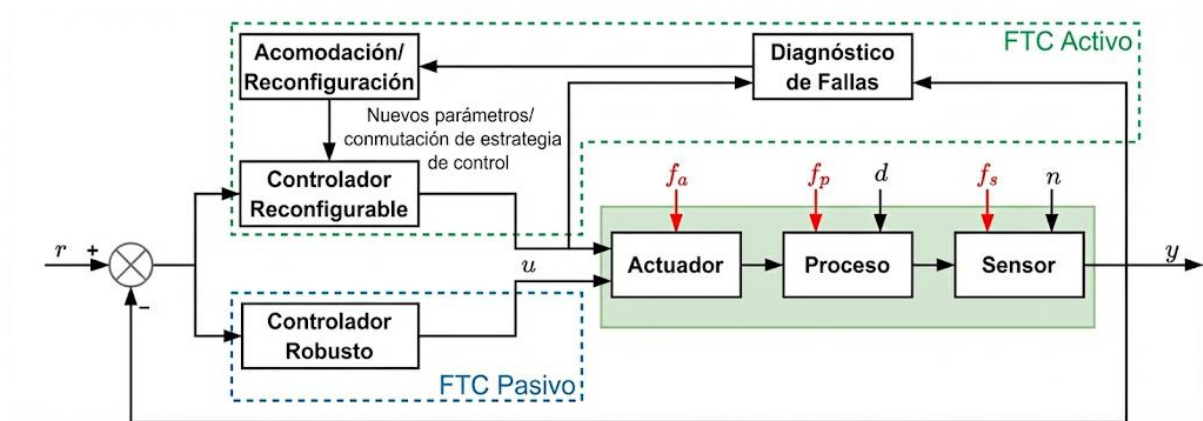


Figura 20. Arquitectura de las estrategias de control tolerante a fallas activo y pasivo (Wu, Sehab, Akrad, & Morel, 2022).

Donde, Cada componente se ve afectado por los fallos correspondientes f_a , f_p , f_s , las perturbaciones externas d y el ruido de medición n .

2.6.6 Clasificación de fallas

Una falla en un sistema dinámico se define como una desviación de su estructura o parámetros respecto a la condición nominal, pudiendo provocar desde una degradación del rendimiento hasta la pérdida total del sistema sistema (Abbaspour, Mokhtari, Sargolzaei, & Yen, 2020).

Según el componente afectado, las fallas se clasifican en:

- Fallas en la planta, que alteran la dinámica del sistema.
- Fallas en sensores, donde las lecturas son erróneas sin afectar la planta.
- Fallas en actuadores, donde la acción de control se ve comprometida.

Las causas comunes incluyen interrupciones de comunicación, ruido ambiental, pérdida de alimentación, ataques cibernéticos (inyección de datos falsos), y fallas físicas como actuadores trabados, descontrolados o sin presión hidráulica.

Según su severidad, las fallas también se dividen en:

- Abruptas, cambios rápidos difíciles de detectar.
- Incipientes, con efectos sutiles y progresivos.

- Intermitentes, ocurren de forma esporádica y son difíciles de identificar por su comportamiento inconsistente.

La Figura 21 presenta una descripción gráfica de fallas abruptas, incipientes e intermitentes en el sistema (Abbaspour, Mokhtari, Sargolzaei, & Yen, 2020).

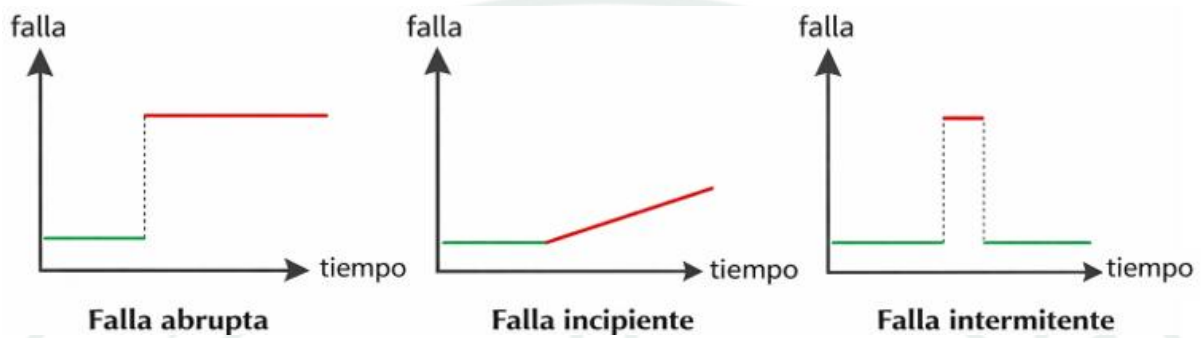


Figura 21. Descripción gráfica de fallas abruptas, incipientes e intermitentes.

2.7 Clasificación de fallas en variadores de velocidad

El mapa conceptual organiza las fallas en variadores de velocidad (VSD) en tres grupos principales: máquina eléctrica, sensores y actuador. Las fallas de la máquina eléctrica pueden ser mecánicas (desgaste de escobillas, desalineación del eje, desbalance, rotor excéntrico, rodamientos y resonancia) o eléctricas (rotura de barras, devanados en cortocircuito e incluso deterioro del aislamiento). En los sensores, se identifican fallas vinculadas a variables mecánicas (sensores de velocidad y desplazamiento del eje) y variables eléctricas (sensores de corriente y voltaje), que afectan directamente el control del sistema. Finalmente, en el actuador, las fallas más comunes son las de los semiconductores de potencia y el suministro eléctrico. Esta clasificación facilita la detección y el diagnóstico de fallas, apoyando estrategias de mantenimiento y mejora en la confiabilidad de los variadores de velocidad (Yepes, Gonzalez-Prieto, Lopez, & Duran, 2022).

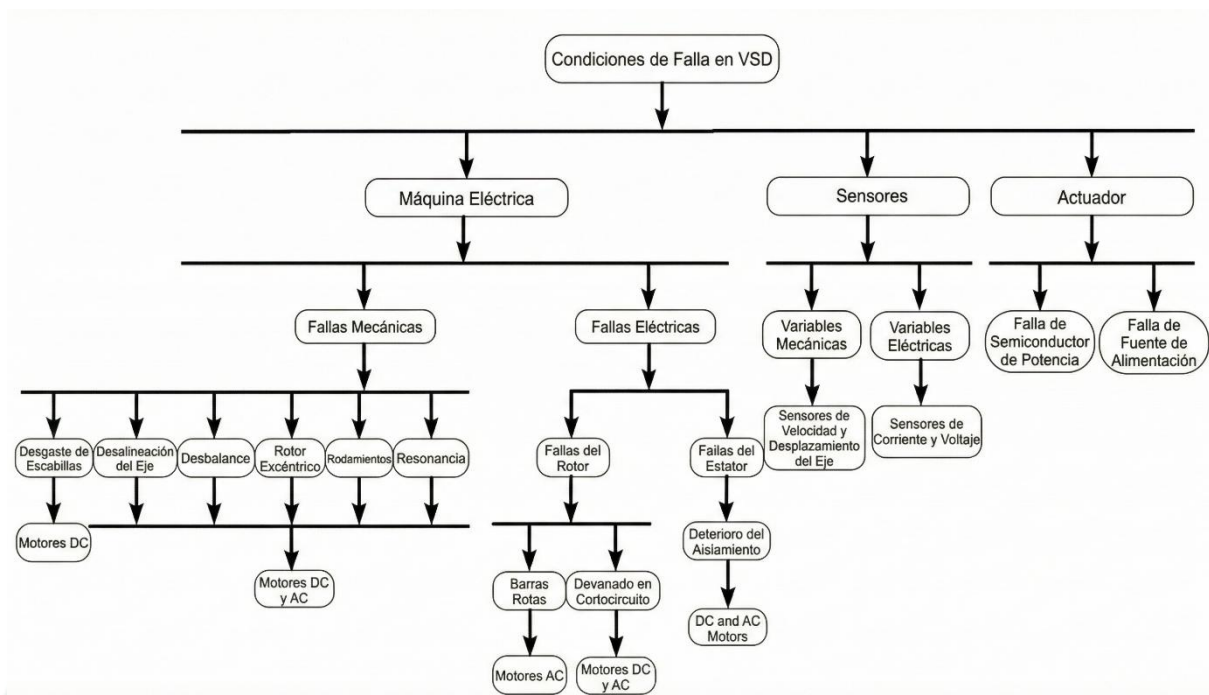


Figura 22. Condición de falla en VSD (Yepes, Gonzalez-Prieto, Lopez, & Duran, 2022)

En esta tesis, el estudio se centrará específicamente en las fallas de los sensores de velocidad y corriente, debido a su impacto crítico en el desempeño del variador bajo control vectorial. Para mitigar sus efectos, se desarrolla la aplicación de técnicas de control tolerante a fallas (FTC), que permitan mantener la estabilidad y el rendimiento del sistema aun en presencia de dichas fallas permitiendo operación continua.

CAPÍTULO III: IDENTIFICACIÓN DE PARÁMETROS DE MOTOR DE INDUCCIÓN

En este capítulo se expone el desarrollo del modelo dinámico del motor de inducción, el cual resulta fundamental para la implementación del control vectorial. Además, se realiza un análisis comparativo en la estimación de parámetros del motor empleando dos enfoques: por un lado, el método experimental, que comprende las pruebas de corriente continua, de vacío y de rotor bloqueado; y por otro, un método de identificación basado en optimización por enjambre de partículas (PSO), utilizando tanto los datos de placa y manufactura del motor.

3.1 Modelado de motor de inducción

El comportamiento dinámico de la máquina de inducción puede describirse mediante un conjunto de ecuaciones diferenciales ordinarias, formuladas en distintos sistemas de referencia. Para elaborar el modelo matemático se consideran las siguientes hipótesis de simplificación (Lyshevski, 2017).

- El material magnético de la máquina se asume lineal, por lo que se desprecia la saturación de la inductancia principal.
- Se ignoran las pérdidas magnéticas y las variaciones de la resistencia del rotor ocasionadas por el efecto pelicular.
- Todos los parámetros de la máquina se consideran constantes en el tiempo; se omiten las variaciones de la resistencia del estator asociadas a los cambios de temperatura.
- La máquina es simétrica en sus tres fases y también en el rotor, por lo que no se tiene en cuenta la prominencia geométrica de este.
- Los devanados se suponen distribuidos de manera sinusoidal.

3.1.1 Ecuaciones del sistema en el marco de referencia ABC

En un sistema trifásico balanceado de corriente alterna (CA), cada fase presenta una tensión sinusoidal de valor pico $\sqrt{2}v_{ph}$, siendo v_{ph} el valor eficaz (RMS) de la tensión de fase.

Las tres fases a, b y c tienen la misma frecuencia f , pero están desfasadas $2\pi/3$ (120°) entre sí. Así, las tensiones instantáneas en el instante t se expresan como:

$$v_a(t) = \sqrt{2}v_{ph}\sin(\omega t) \quad (3.1)$$

$$v_b(t) = \sqrt{2}v_{ph}\sin\left(\omega t - \frac{2\pi}{3}\right) \quad (3.2)$$

$$v_c(t) = \sqrt{2}v_{ph}\sin\left(\omega t - \frac{4\pi}{3}\right) \quad (3.3)$$

Donde $\omega = 2\pi f$ denota la frecuencia angular.

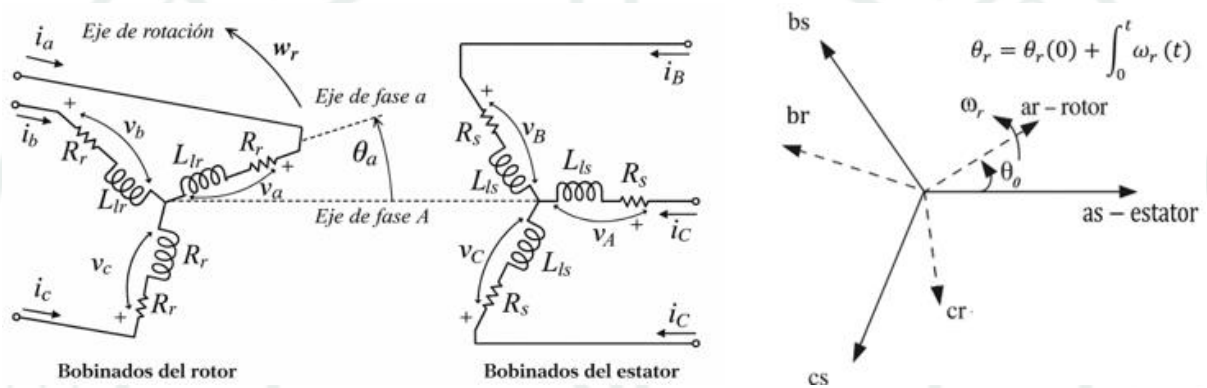


Figura 23. Circuito equivalente de M.I. conectado en Y (Le Fosse, 2021)

Partiendo de la Figura 23, el estator consta de devanados trifásicos clasificados como as, bs y cs, con el mismo número de vueltas efectivas por fase. De forma similar, el rotor consta de devanados trifásicos ar, br y cr, con el mismo número de vueltas por fase. Las ecuaciones eléctricas de tensión del estator y del rotor pueden escribirse utilizando la ley de tensión de Kirchhoff, donde las tres fases se representan en forma matricial de la siguiente manera:

3.1.2 Ecuaciones eléctricas

- Ecuaciones de voltaje

$$v_{sabc} = R_s i_{sabc} + \frac{d\psi_{sabc}}{dt} \quad (3.4)$$

$$v_{rabc} = R_r i_{rabc} + \frac{d\psi_{rabc}}{dt} \quad (3.5)$$

Donde los vectores de voltaje (v), corriente (i) y enlaces de flujo (ψ) se definen:

$$v_{sabc} = [v_{sa} \ v_{sb} \ v_{sc}]^T \quad ; \quad i_{sabc} = [i_{sa} \ i_{sb} \ i_{sc}]^T \quad ; \quad \psi_{sabc} = [\psi_{sa} \ \psi_{sb} \ \psi_{sc}]^T$$

$$v_{r_{abc}} = [v_{ra} \ v_{rb} \ v_{rc}]^T \quad ; \quad i_{r_{abc}} = [i_{ra} \ i_{rb} \ i_{rc}]^T \quad ; \quad \psi_{r_{abc}} = [\psi_{ra} \ \psi_{rb} \ \psi_{rc}]^T$$

Donde v_{sa}, v_{sb}, v_{sc} son los voltajes de fase suministrados a los devanados del estator; v_{ra}, v_{rb}, v_{rc} son los voltajes de fase suministrados a los devanados del rotor (para motores de jaula de ardilla $v_{ra} = 0, v_{rb} = 0, v_{rc} = 0$); i_{sa}, i_{sb}, i_{sc} son las corrientes de fase del estator; i_{ra}, i_{rb}, i_{rc} son las corrientes de fase del rotor, $\psi_{sa}, \psi_{sb}, \psi_{sc}$ son los enlaces de flujo del estator; $\psi_{ra}, \psi_{rb}, \psi_{rc}$ son los enlaces de flujo del rotor

Las matrices diagonales de las resistencias del estator y del rotor son:

$$R_s = \begin{bmatrix} r_s & 0 & 0 \\ 0 & r_s & 0 \\ 0 & 0 & r_s \end{bmatrix} \quad R_r = \begin{bmatrix} r_r & 0 & 0 \\ 0 & r_r & 0 \\ 0 & 0 & r_r \end{bmatrix}$$

Donde r_s es la resistencia del estator en cada devanado y r_r es la resistencia del rotor en cada devanado (asumiendo simetría en devanados).

- Ecuaciones de enlace de flujo

La ecuación de enlace de flujo se expresa como: $\psi = L \cdot i$

Donde ψ es el vector de enlaces de flujo, i el vector de corrientes y L la matriz de inductancias que incluye los términos propios y mutuos.

En el motor de inducción, el flujo magnético en cada devanado es producto de la superposición de las corrientes propias y las corrientes de los otros devanados (estator y rotor).

$$\psi_{s_{abc}} = \psi_{s_{abc}}^s + \psi_{s_{abc}}^r \quad (3.6)$$

$$\psi_{r_{abc}} = \psi_{r_{abc}}^r + \psi_{r_{abc}}^s \quad (3.7)$$

Donde:

$\psi_{s_{abc}}^s$: Enlace de flujo del estator debido a las corrientes del estator

$\psi_{s_{abc}}^r$: Enlace de flujo del estator debido a las corrientes del rotor.

$\psi_{r_{abc}}^r$: Enlace de flujo del rotor debido a las corrientes del rotor.

$\psi_{r_{abc}}^s$: Enlace de flujo del rotor debido a las corrientes del estator.

Cada devanado posee una autoinductancia (por el flujo creado por su propia corriente) y mutuas inductancias (por el flujo debido a corrientes de otros devanados). Esto se representa matricialmente como:

$$\psi_{sabc} = \psi_{sabc}^s + \psi_{sabc}^r = L_{ss}i_{abc}^s + L_{sr}i_{abc}^r \quad (3.8)$$

$$\psi_{rabc} = \psi_{rabc}^r + \psi_{rabc}^s = L_{rr}i_{abc}^r + L_{rs}i_{abc}^s \quad (3.9)$$

Donde L_{ss} y L_{rr} son las matrices de inductancia propia del estator y rotor respectivamente, y L_{sr} es la matriz de inductancia mutua entre estator y rotor.

- Definición de Parámetros de Inductancia

Inductancias Propias del Estator L_{ss} y Rotor L_{rr} : Debido a la disposición simétrica de 120° eléctricos, la inductancia mutua entre fases del mismo devanado es $-\frac{1}{2}L_m$. La autoinductancia total de una fase es la suma de la inductancia de fuga L_{ls} y la magnetizante L_{ms} .

$$L_{ss} = \begin{bmatrix} L_{ls} + L_{ms} & -\frac{L_{ms}}{2} & -\frac{L_{ms}}{2} \\ -\frac{L_{ms}}{2} & L_{ls} + L_{ms} & -\frac{L_{ms}}{2} \\ -\frac{L_{ms}}{2} & -\frac{L_{ms}}{2} & L_{ls} + L_{ms} \end{bmatrix}, \quad L_{rr} = \begin{bmatrix} L_{lr} + L_{mr} & -\frac{L_{mr}}{2} & -\frac{L_{mr}}{2} \\ -\frac{L_{mr}}{2} & L_{lr} + L_{mr} & -\frac{L_{mr}}{2} \\ -\frac{L_{mr}}{2} & -\frac{L_{mr}}{2} & L_{lr} + L_{mr} \end{bmatrix}$$

Inductancia Mutua Estator-Rotor (L_{sr}) : Esta inductancia depende de la posición angular del rotor (θ_r). Definiendo L_{msr} como la amplitud máxima de la inductancia mutua, la matriz se define como:

$$L_{sr}(\theta_r) = L_{msr} \begin{bmatrix} \cos(\theta_r) & \cos(\theta_r + \frac{2\pi}{3}) & \cos(\theta_r - \frac{2\pi}{3}) \\ \cos(\theta_r - \frac{2\pi}{3}) & \cos(\theta_r) & \cos(\theta_r + \frac{2\pi}{3}) \\ \cos(\theta_r + \frac{2\pi}{3}) & \cos(\theta_r - \frac{2\pi}{3}) & \cos(\theta_r) \end{bmatrix}$$

$$L_{sr}(-\theta_r) = L_{msr} \begin{bmatrix} \cos(\theta_r) & \cos(\theta_r - \frac{2\pi}{3}) & \cos(\theta_r + \frac{2\pi}{3}) \\ \cos(\theta_r + \frac{2\pi}{3}) & \cos(\theta_r) & \cos(\theta_r - \frac{2\pi}{3}) \\ \cos(\theta_r - \frac{2\pi}{3}) & \cos(\theta_r + \frac{2\pi}{3}) & \cos(\theta_r) \end{bmatrix}$$

Debido a los devanados simétricos y configuración del motor, se puede escribir la relación:

$$\frac{1}{N} \cdot L_{ms} = N \cdot L_{mr} = L_{msr}$$

Donde N es la relación de números de vueltas entre estator y rotor

Finalmente, la ecuación de flujos tanto del estator como el rotor se puede describir como:

$$\psi_{sabc} = \psi_{sabc}^s + \psi_{sabc}^r = L_{ss} \cdot i_{sabc} + L_{sr}(\theta_r) \cdot i_{rabc} \quad (3.10)$$

$$\psi_{rabc} = \psi_{rabc}^r + \psi_{rabc}^s = L_{rr} \cdot i_{rabc} + L_{sr}(-\theta_r) \cdot i_{sabc} \quad (3.11)$$

3.1.3 Ecuaciones mecánicas

La dinámica rotacional de una máquina eléctrica se describe a partir de la segunda ley de Newton aplicada al movimiento angular, lo que permite establecer las ecuaciones que gobiernan el comportamiento mecánico del sistema. El torque desarrollado por la máquina está vinculado a la energía magnética almacenada en el entrehierro, la cual depende directamente de las corrientes que circulan tanto en el estator como en el rotor (Le Fosse, 2021). Bajo este enfoque, el torque electromagnético T_e interviene en la ecuación de movimiento del rotor, dando lugar a la siguiente relación dinámica:

$$J \frac{\partial \omega_r}{\partial t} = -B\omega_r + p(T_e - T_L) \quad (3.12)$$

En esta expresión, J representa el momento de inercia asociado al motor, ω_r corresponde a la velocidad eléctrica del rotor, B es el coeficiente que modela las pérdidas por fricción viscosa, p indica el número de pares de polos de la máquina y T_L denota el torque mecánico aplicado por la carga.

Además, la relación entre las magnitudes angulares y las velocidades del sistema se define considerando que la derivada temporal de la posición angular del rotor es igual a su

velocidad eléctrica, mientras que la velocidad mecánica se obtiene al normalizar esta última con el número de pares de polos:

$$\frac{\partial \theta_r}{\partial t} = \omega_r \quad ; \quad \omega_m = \frac{\omega_r}{p} \quad (3.13)$$

3.1.4 Ecuaciones vectoriales espaciales

Con el fin de simplificar el modelado y análisis de los circuitos trifásicos equilibrados, es habitual convertir las variables del sistema trifásico abc a un sistema de referencia ortogonal, que puede ser fijo o rotante. El sistema de coordenadas estacionario se define a partir de tres ejes mutuamente perpendiculares: α , β y 0. Posteriormente, mediante una transformación matemática, este sistema puede trasladarse al marco de referencia giratorio dq, lo que facilita el estudio dinámico y el control de la máquina eléctrica (Geyer, 2016).

3.1.5 Transformada de Clarke

Las magnitudes trifásicas abc se pueden transferir al sistema de coordenadas estacionario $\alpha\beta$ -dos mediante la transformación de Clarke como se ve en la Figura 24 y, de forma similar, se utiliza su inversa para transformar el vector de coordenadas $\alpha\beta$ -dos al sistema de coordenadas abc-tres. $\xi_{abc} = [\xi_a \ \xi_b \ \xi_c]^T$ representa cualquiera de los tres vectores, como la corriente, la tensión y el flujo y $\xi_{\alpha\beta 0} = [\xi_\alpha \ \xi_\beta \ \xi_0]^T$ y viceversa, mediante las siguientes transformaciones:

$$\xi_{\alpha\beta 0} = K \xi_{abc} \text{ y } \xi_{abc} = K^{-1} \xi_{\alpha\beta 0}$$

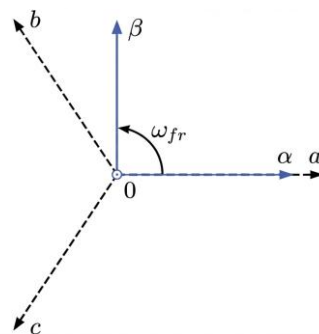


Figura 24. Definición del sistema de referencia estacionario $\alpha\beta 0$ (Lyshevski, 2017)

Las matrices de transformación están dadas por

$$K = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \quad K^{-1} = \frac{2}{3} \begin{bmatrix} 1 & 0 & 1 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} & 1 \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} & 1 \end{bmatrix}$$

El factor $2/3$ en K garantiza que se conserven las amplitudes de las señales trifásicas (balanceadas). Como resultado, la transformación de Clarke, definida previamente, es invariante respecto a picos o amplitud (Geyer, 2016). Por ejemplo, para corriente del estator

$$\begin{bmatrix} i_{s\alpha} \\ i_{s\beta} \end{bmatrix} = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} i_{sa} \\ i_{sb} \\ i_{sc} \end{bmatrix}$$

En la Figura 25 se puede ver el resultado de la transformada en $\alpha\beta 0$ de corrientes trifásicas balanceadas abc.

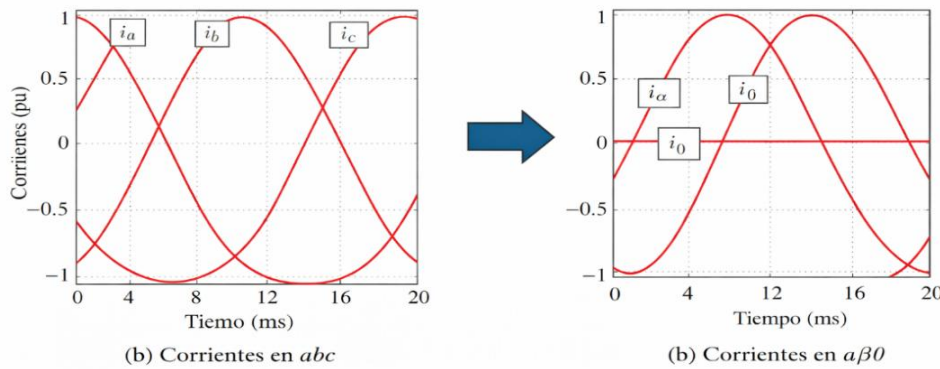


Figura 25. Corrientes trifásicas balanceadas representadas en el sistema de referencia estacionario $\alpha\beta 0$ (Geyer, 2016)

La transformación de Clarke puede representarse igualmente mediante el uso de un vector espacial, el cual se define como una magnitud compleja que varía en función del tiempo y se expresa como $x(t)$. Este enfoque permite describir de manera compacta el comportamiento de las variables trifásicas, facilitando su análisis en el dominio complejo.

$$x(t) = \frac{2}{3} \cdot \left(1 \cdot x_a + e^{j\frac{2\pi}{3}} \cdot x_b + e^{j\frac{4\pi}{3}} \cdot x_c \right) \quad (3.14)$$

Donde

$$e^{j\frac{2\pi}{3}} = -\frac{1}{2} + j\frac{\sqrt{3}}{2} \quad , \quad e^{j\frac{4\pi}{3}} = -\frac{1}{2} - j\frac{\sqrt{3}}{2}$$

donde x_a, x_b, x_c , son las variables de voltaje, corriente, etc.

Aplicando transformada de Clarke a las ecuaciones de voltaje se obtiene

$$v_{s\alpha\beta} = r_s i_{s\alpha\beta} + \frac{\partial \psi_{s\alpha\beta}}{\partial t} \quad (3.15)$$

$$v_{r\alpha\beta} = r_r i_{r\alpha\beta} + \frac{\partial \psi_{r\alpha\beta}}{\partial t} \quad (3.16)$$

Aplicando transformada de Clarke en ecuaciones de enlace de flujo

$$\psi_{s_{abc}} = \psi_{s_{abc}}^s + \psi_{s_{abc}}^r = L_{s_{abc}} \cdot i_{s_{abc}} + L_{sr}(\theta_r) \cdot i_{r_{abc}}$$

En $L_{s_{abc}} \cdot i_{s_{abc}}$

$$\begin{aligned} \psi_{s\alpha\beta}^s &= \frac{2}{3} \cdot \left(1 \cdot \psi_{s_a}^s + e^{j\frac{2\pi}{3}} \cdot \psi_{s_b}^s + e^{j\frac{4\pi}{3}} \cdot \psi_{s_c}^s \right) \\ \psi_{s\alpha\beta}^s &= \frac{2}{3} \cdot \left[(L_{ls} + L_{ms}) i_{s_a} - \frac{1}{2} \cdot L_{ms} \cdot i_{s_b} - \frac{1}{2} \cdot L_{ms} \cdot i_{s_c} \right] + \\ &\frac{2}{3} \cdot e^{j\frac{2\pi}{3}} \cdot \left[-\frac{1}{2} \cdot L_{ms} \cdot i_{s_a} + (L_{ls} + L_{ms}) \cdot i_{s_b} - \frac{1}{2} \cdot L_{ms} \cdot i_{s_c} \right] + \\ &\frac{2}{3} \cdot e^{j\frac{4\pi}{3}} \cdot \left[-\frac{1}{2} \cdot L_{ms} \cdot i_{s_a} - \frac{1}{2} \cdot L_{ms} \cdot i_{s_b} + (L_{ls} + L_{ms}) \cdot i_{s_c} \right] \end{aligned}$$

Simplificando

$$\begin{aligned} \psi_{s\alpha\beta}^s &= \frac{2}{3} \cdot \left[\left(\frac{3}{2} L_{ms} + L_{ls} \right) \cdot \left(i_{s_a} + e^{j\frac{2\pi}{3}} \cdot i_{s_b} + e^{j\frac{4\pi}{3}} \cdot i_{s_c} \right) \right] \\ &+ \frac{2}{3} \cdot \left(-\frac{1}{2} \cdot L_{ms} \cdot i_{s_a} - \frac{1}{2} \cdot L_{ms} \cdot i_{s_b} - \frac{1}{2} \cdot L_{ms} \cdot i_{s_c} \right) + \\ &\frac{2}{3} \cdot e^{j\frac{2\pi}{3}} \cdot \left(-\frac{1}{2} \cdot L_{ms} \cdot i_{s_a} - \frac{1}{2} \cdot L_{ms} \cdot i_{s_b} - \frac{1}{2} \cdot L_{ms} \cdot i_{s_c} \right) + \\ &\frac{2}{3} \cdot e^{j\frac{4\pi}{3}} \cdot \left(-\frac{1}{2} \cdot L_{ms} \cdot i_{s_a} - \frac{1}{2} \cdot L_{ms} \cdot i_{s_b} - \frac{1}{2} \cdot L_{ms} \cdot i_{s_c} \right) \end{aligned}$$

Considerando para un sistema balanceado $i_{s_a} + i_{s_b} + i_{s_c} = 0$ y $\frac{2}{3} \cdot \left(i_{s_a} + e^{j\frac{2\pi}{3}} \cdot i_{s_b} + e^{j\frac{4\pi}{3}} \cdot i_{s_c} \right) = i_{s_{\alpha\beta}}$

Denotando $L_M = \frac{3}{2} \cdot \frac{1}{N} \cdot L_{m_s} = \frac{3}{2} \cdot N \cdot L_{m_r}$, inductancia del estator es $L_s = \frac{3}{2} L_{m_s} + L_{ls}$

Resolviendo, se obtiene

$$\psi_{s_{\alpha\beta}}^s = L_s \cdot i_{s_{\alpha\beta}} \quad (3.17)$$

Conversión de la segunda parte $L_{sr}(\theta_r) i_{r_{abc}}$ al sistema de coordenadas $\alpha\beta$

$$\begin{aligned} \psi_{s_{\alpha\beta}}^r &= \frac{2}{3} \cdot \left[L_{m_{sr}} \cdot \cos(\theta_r) \cdot i_{r_a} + L_{m_{sr}} \cdot \cos\left(\theta_r + \frac{2\pi}{3}\right) \cdot i_{r_b} + L_{m_{sr}} \cdot \cos\left(\theta_r - \frac{2\pi}{3}\right) \cdot i_{r_c} \right] \\ &+ \frac{2}{3} \cdot e^{j\frac{2\pi}{3}} \cdot \left[L_{m_{sr}} \cdot \cos\left(\theta_r - \frac{2\pi}{3}\right) \cdot i_{r_a} + L_{m_{sr}} \cdot \cos(\theta_r) \cdot i_{r_b} + L_{m_{sr}} \cdot \cos\left(\theta_r + \frac{2\pi}{3}\right) \cdot i_{r_c} \right] \\ &+ \frac{2}{3} \cdot e^{j\frac{4\pi}{3}} \cdot \left[L_{m_{sr}} \cdot \cos\left(\theta_r + \frac{2\pi}{3}\right) \cdot i_{r_a} + L_{m_{sr}} \cdot \cos\left(\theta_r - \frac{2\pi}{3}\right) \cdot i_{r_b} + L_{m_{sr}} \cdot \cos(\theta_r) \cdot i_{r_c} \right] \end{aligned}$$

Basado en la función trigonométrica $\frac{e^{j\alpha} + e^{-j\alpha}}{2} = \cos\alpha$ se tiene

$$\cos\left(\theta_r - \frac{2\pi}{3}\right) = \frac{1}{2} \left(e^{j(\theta_r - \frac{2\pi}{3})} + e^{-j(\theta_r - \frac{2\pi}{3})} \right)$$

Reduciendo la ecuación

$$\begin{aligned} \psi_{s_{\alpha\beta}}^r &= \frac{1}{2} L_{m_{sr}} \cdot \frac{2}{3} \left[3 \cdot e^{j\theta_r} \cdot i_{r_a} + 3 \cdot e^{j(\theta_r + \frac{2\pi}{3})} \cdot i_{r_b} + 3 \cdot e^{j(\theta_r - \frac{2\pi}{3})} \cdot i_{r_c} \right] \\ \psi_{s_{\alpha\beta}}^r &= L_{m_{sr}} \cdot e^{j\theta_r} \cdot \left(i_{r_a} + e^{j\frac{2\pi}{3}} \cdot i_{r_b} + e^{j\frac{4\pi}{3}} \cdot i_{r_c} \right) \end{aligned}$$

Sustituyendo $\frac{2}{3} \cdot \left(i_{r_a} + e^{j\frac{2\pi}{3}} \cdot i_{r_b} + e^{j\frac{4\pi}{3}} \cdot i_{r_c} \right) = i_{r_{\alpha\beta}}$

$$\psi_{s_{\alpha\beta}}^r = \frac{3}{2} L_{m_s} \cdot e^{j\theta_r} \cdot i_{r_{\alpha\beta}} \quad (3.18)$$

Denotando $L_M = \frac{3}{2} L_{m_{sr}}$ (L_M es inductancia mutua entre una fase del estator y del rotor)

$$\psi_{s_{\alpha\beta}}^r = L_M \cdot e^{j\theta_r} \cdot i_{r_{\alpha\beta}} \quad (3.19)$$

La combinación de la primera y la segunda parte da como resultado el enlace de flujo total del estator en el sistema de coordenadas $\alpha\beta$. Se puede aplicar un procedimiento similar

para determinar la relación de flujo del rotor. Las relaciones de flujo totales del estator y del rotor en el sistema de coordenadas $\alpha\beta$ se resumen en (3.20) y (3.21).

$$\psi_{s\alpha\beta} = L_s \cdot i_{s\alpha\beta} + L_M \cdot e^{j\theta_r} \cdot i_{r\alpha\beta} \quad (3.20)$$

$$\psi_{r\alpha\beta} = L_r \cdot i_{r\alpha\beta} + L_M \cdot e^{-j\theta_r} \cdot i_{s\alpha\beta} \quad (3.21)$$

3.1.6 Transformada de Park

La transformación de Park convierte un sistema estacionario en un sistema rotacional. La transformación de Park rotacional convierte un sistema de coordenadas estacionario $\alpha\beta$ $\xi_{\alpha\beta} = [\xi_\alpha \ \xi_\beta]^T$ en un sistema rotacional dq $\xi_{dq} = [\xi_d \ \xi_q]^T$ y viceversa, mediante las siguientes transformaciones (Geyer, 2016).

$$\xi_{dq} = R(\theta)\xi_{\alpha\beta} \text{ y } \xi_{\alpha\beta} = R^{-1}(\theta)\xi_{dq}$$

Las matrices de transformación están dadas por

$$R(\theta) = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \quad R^{-1}(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

Como se muestra en la Figura 26, el eje q precede al eje d en 90° ; es decir, el eje q está en cuadratura con el eje d.

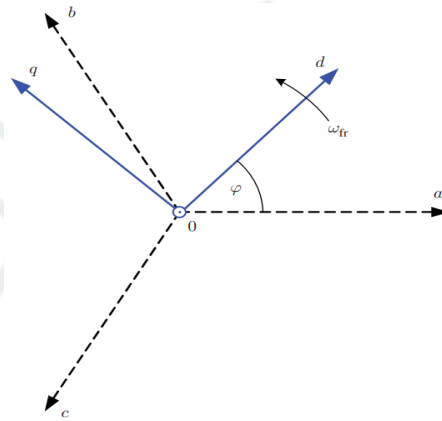


Figura 26. Definición del sistema de referencia rotatorio dq0 (Lyshevski, 2017)

La posición angular del sistema de referencia dq se define por θ , que es el ángulo entre el eje d del sistema de referencia giratorio y el eje α del sistema trifásico, con

$$\theta(t) = \int_0^t \omega(\tau) d\tau + \phi(0)$$

Aquí, ω representa la velocidad angular del sistema de referencia.

Por ejemplo, aplicando para corriente del estator

$$\begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} i_{s\alpha} \\ i_{s\beta} \end{bmatrix}$$

En la Figura 27 se puede ver el resultado de la transformada en dq

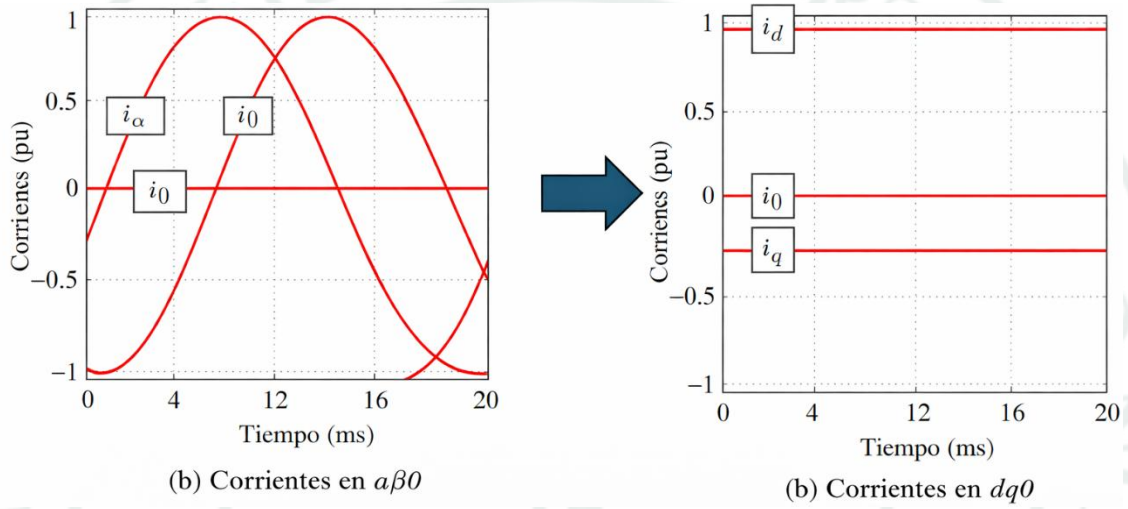


Figura 27. Corrientes trifásicas equilibradas representadas en el sistema de referencia giratorio dq0 (Geyer, 2016)

La transformación de Park puede formularse de manera equivalente utilizando un vector espacial, el cual se expresa como una variable compleja dependiente del tiempo, denotada por $x(t)$. En este planteamiento, la conversión desde el sistema de referencia estacionario $\alpha\beta$ hacia el marco rotatorio dq se realiza mediante una rotación compleja, definida:

$$\begin{aligned} x_{dq} &= x_{\alpha\beta} \cdot e^{-j\theta_k(t)} = (\cos \theta_k - j \sin \theta_k)(x_\alpha + j x_\beta) \\ &= (x_\alpha \cos \theta_k + x_\beta \sin \theta_k) + j(x_\beta \cos \theta_k - x_\alpha \sin \theta_k) \end{aligned}$$

En este contexto, $\theta_k(t)$ corresponde al ángulo de rotación variable en el tiempo que define la orientación relativa entre el marco de referencia giratorio y el sistema estacionario

original. Asimismo, x_{dq} representa el vector espacial expresado en el nuevo sistema de coordenadas rotantes.

Aplicando la transformada de Park a la transformada de Clarke en ecuación de voltaje de estator

$$v_{sdq} = e^{-j\theta_s} v_{s\alpha\beta} = r_s e^{-j\theta_s} i_{s\alpha\beta} + e^{-j\theta_s} \frac{d\psi_{s\alpha\beta}}{dt} \quad (3.22)$$

Se necesita derivar $\psi_{s\alpha\beta}$ pero expresado en términos de dq. Usamos la relación inversa: $\psi_{s\alpha\beta} = e^{j\theta_s} \psi_{sdq}$

Hallando la derivada de:

$$\frac{d}{dt} (e^{j\theta_s} \psi_{sdq}) = \frac{d}{dt} (e^{j\theta_s}) \psi_{sdq} + e^{j\theta_s} \frac{d\psi_{sdq}}{dt}$$

Teniendo en cuenta que la derivada de $e^{j\theta_s}$ es $j\omega_s e^{j\theta_s}$ ya que $\frac{d\theta_s}{dt} = \omega_s$ se obtiene:

$$\frac{d}{dt} (e^{j\theta_s} \psi_{sdq}) = j\omega_s e^{j\theta_s} \psi_{sdq} + e^{j\theta_s} \frac{d\psi_{sdq}}{dt} \quad (3.23)$$

Reemplazando (3.23) en (3.22) y $e^{-j\theta_s} i_{s\alpha\beta} = i_{sdq}$ se obtiene:

$$v_{sdq} = r_s i_{sdq} + e^{-j\theta_s} \left(e^{j\theta_s} j\omega_s \psi_{sdq} + e^{j\theta_s} \frac{d\psi_{sdq}}{dt} \right)$$

Simplificando:

$$v_{sdq} = r_s i_{sdq} + j\omega_s \psi_{sdq} + \frac{d\psi_{sdq}}{dt} \quad (3.24)$$

Se puede representar v_{sdq} , i_{sdq} y ψ_{sdq} de la siguiente manera

$$v_{sdq} = v_{sd} + jv_{sq} \quad , \quad i_{sdq} = i_{sd} + ji_{sq} \quad , \quad \psi_{sdq} = \psi_{sd} + j\psi_{sq}$$

Ahora, separando las partes real e imaginaria en (3.24), obtenemos

$$v_{sd} = r_s i_{sd} - \omega_s \psi_{sq} + \frac{d\psi_{sd}}{dt} \quad (3.25)$$

$$v_{sq} = r_s i_{sq} + \omega_s \psi_{sd} + \frac{d\psi_{sq}}{dt} \quad (3.26)$$

Aplicando la transformada de Park a la transformada de Clarke en ecuación de voltaje del rotor

$$v_{r_{dq}} = e^{-j(\theta_s - \theta_r)} v_{r_{\alpha\beta}} = r_r e^{-j(\theta_s - \theta_r)} i_{r_{\alpha\beta}} + e^{-j(\theta_s - \theta_r)} \left(\frac{d\psi_{r_{\alpha\beta}}}{dt} \right) \quad (3.27)$$

Al igual que la ecuación de voltaje del estator se sustituye $\psi_{r_{\alpha\beta}}$

Obteniendo en el segundo término:

$$e^{-j(\theta_s - \theta_r)} \frac{d}{dt} \left(e^{j(\theta_s - \theta_r)} \psi_{r_{dq}} \right)$$

Aplicando la derivada:

$$\frac{d}{dt} \left(e^{j(\theta_s - \theta_r)} \psi_{r_{dq}} \right) = \frac{d}{dt} \left(e^{j(\theta_s - \theta_r)} \right) \psi_{r_{dq}} + e^{j(\theta_s - \theta_r)} \frac{d\psi_{r_{dq}}}{dt}$$

Teniendo en cuenta $\frac{d(\theta_s - \theta_r)}{dt} = \omega_s - \omega_r$

$$\frac{d}{dt} \left(e^{j(\theta_s - \theta_r)} \psi_{r_{dq}} \right) = [j(\omega_s - \omega_r) e^{j(\theta_s - \theta_r)}] \psi_{r_{dq}} + e^{j(\theta_s - \theta_r)} \frac{d\psi_{r_{dq}}}{dt} \quad (3.28)$$

Reemplazando (3.28) en (3.27) se obtiene:

$$v_{r_{dq}} = r_r i_{r_{dq}} + e^{-j(\theta_s - \theta_r)} \left([j(\omega_s - \omega_r) e^{j(\theta_s - \theta_r)}] \psi_{r_{dq}} + e^{j(\theta_s - \theta_r)} \frac{d\psi_{r_{dq}}}{dt} \right)$$

Simplificando se obtiene:

$$v_{r_{dq}} = r_r i_{r_{dq}} + j(\omega_s - \omega_r) \psi_{r_{dq}} + \frac{d\psi_{r_{dq}}}{dt} \quad (3.29)$$

Separando la ecuación (3.29) en las partes real e imaginaria, de forma similar a las ecuaciones de tensión del estator e igualando $v_{r_q} = v_{r_d} = 0$, se obtienen las siguientes ecuaciones.

$$0 = r_r i_{r_d} - (\omega_s - \omega_r) \psi_{r_q} + \frac{d\psi_{r_d}}{dt} \quad (3.30)$$

$$0 = r_r i_{r_q} + (\omega_s - \omega_r) \psi_{r_d} + \frac{d\psi_{r_q}}{dt} \quad (3.31)$$

La transformación de Park giratoria se utiliza para convertir un sistema de coordenadas estacionario en un sistema de coordenadas giratorio de los enlaces de flujo del estator.

$$\psi_{sdq} = e^{-j\theta_s} \psi_{s\alpha\beta} \quad (3.32)$$

Sustituyendo (3.20) en (3.32), se obtiene:

$$\psi_{sdq} = L_s \cdot i_{sdq} + L_M \cdot e^{j\theta_r} \cdot e^{-j\theta_r} \cdot i_{r\alpha\beta} \quad (3.33)$$

Basada en la igualdad $e^{-j(\theta_s - \theta_r)} \cdot i_{r\alpha\beta} = i_{rdq}$, la ecuación queda de la siguiente manera.

$$\psi_{sdq} = L_s \cdot i_{sdq} + L_M \cdot i_{rdq} \quad (3.34)$$

Ahora, al separar las partes real e imaginaria en (3.34), se obtiene:

$$\psi_{sd} = L_s \cdot i_{sd} + L_M \cdot i_{rd} \quad (3.35)$$

$$\psi_{sq} = L_s \cdot i_{sq} + L_M \cdot i_{rq} \quad (3.36)$$

Usando la transformada de Park giratoria para convertir un sistema de coordenadas estacionario en un sistema de coordenadas giratorio de los enlaces de flujo del rotor.

$$\psi_{rdq} = e^{-j(\theta - \theta_r)} \psi_{r\alpha\beta} \quad (3.37)$$

Sustituyendo (3.21) en (3.37), obtenemos:

$$\psi_{rdq} = L_r \cdot i_{rdq} + L_M \cdot i_{sdq} \quad (3.38)$$

Ahora, al separar las partes real e imaginaria en (3.38), obtenemos:

$$\psi_{rd} = L_r \cdot i_{rd} + L_M \cdot i_{sd} \quad (3.39)$$

$$\psi_{rq} = L_r \cdot i_{rq} + L_M \cdot i_{sq} \quad (3.40)$$

Se obtiene un sistema de ecuaciones unificado completo para la máquina de inducción cuando las variables del estator y del rotor se expresan en un sistema de referencia común y las variables del rotor también se refieren al estator. Utilizando la relación de espiras según $N = N_s/N_r$, obtenemos:

$$N = \frac{N_s}{N_r} \quad , \quad i'_r = \frac{1}{N} i_r \quad , \quad v'_r = N v_r \quad , \quad \psi'_r = N \psi_r \quad , \quad r'_r = N^2 r_r$$

$$L_m = N \cdot L_M = \frac{3}{2} \cdot L_{ms} = \frac{3}{2} \cdot N^2 \cdot L_{mr} \quad L'_{lr} = N^2 \cdot L_{lr}$$

Donde L_m es la inductancia de magnetización principal

Se reescriben las ecuaciones eléctricas del modelo matemático del motor de inducción.

Finalmente, se escribe un sistema de ecuaciones completo, en un sistema de referencia común arbitrario, con las variables del rotor y estator, que definen la máquina de inducción.

$$v_{sd} = r_s i_{sd} - \omega_s \psi_{sq} + \frac{d\psi_{sd}}{dt} \quad (3.41)$$

$$v_{sq} = r_s i_{sq} + \omega_s \psi_{sd} + \frac{d\psi_{sq}}{dt} \quad (3.42)$$

$$0 = r'_r i'_{rd} - (\omega_s - \omega_r) \psi'_{rq} + \frac{d\psi'_{rd}}{dt} \quad (3.43)$$

$$0 = r'_r i'_{rq} + (\omega_s - \omega_r) \psi'_{rd} + \frac{d\psi'_{rq}}{dt} \quad (3.44)$$

$$\psi_{sd} = L_s \cdot i_{sd} + L_m \cdot i'_{rd} \quad (3.45)$$

$$\psi_{sq} = L_s \cdot i_{sq} + L_m \cdot i'_{rq} \quad (3.46)$$

$$\psi'_{rd} = L'_r \cdot i'_{rd} + L_m \cdot i_{sd} \quad (3.47)$$

$$\psi'_{rq} = L'_r \cdot i'_{rq} + L_m \cdot i_{sq} \quad (3.48)$$

La corriente del rotor no es medible (i'_{rd} , i'_{rq}) como también el flujo del estator (ψ'_{rd} , ψ'_{rq}). Estas variables deben ser reemplazadas del sistema de ecuaciones. Usando las dos ecuaciones de flujo como se muestra a continuación:

$$i'_{rd} = \frac{1}{L'_r} \psi'_{rd} - \frac{L_m}{L'_r} i_{sd} \quad , \quad i'_{rq} = \frac{1}{L'_r} \psi'_{rq} - \frac{L_m}{L'_r} i_{sq}$$

Reemplazando en:

$$\psi_{sd} = L_s \cdot i_{sd} + L_m \cdot \left(\frac{1}{L'_r} \psi'_{rd} - \frac{L_m}{L'_r} i_{sd} \right) \quad , \quad \psi_{sq} = L_s \cdot i_{sq} + L_m \cdot \left(\frac{1}{L'_r} \psi'_{rq} - \frac{L_m}{L'_r} i_{sq} \right)$$

Ahora sustituyendo en las ecuaciones de voltaje (3.32-33) así:

$$v_{sd} = r_s \cdot i_{sd} + \frac{d \left(L_s \cdot i_{sd} + L_m \cdot \left(\frac{1}{L'_r} \psi'_{rd} - \frac{L_m}{L'_r} i_{sd} \right) \right)}{dt} - \omega_s$$

$$\cdot \left(L_s \cdot i_{sq} + L_m \cdot \left(\frac{1}{L'_r} \psi'_{rq} - \frac{L_m}{L'_r} i_{sq} \right) \right)$$

$$v_{sq} = r_s \cdot i_{sq} + \frac{d \left(L_s \cdot i_{sq} + L_m \cdot \left(\frac{1}{L'_r} \psi'_{rq} - \frac{L_m}{L'_r} i_{sq} \right) \right)}{dt} + \omega_k$$

$$\cdot \left(L_s \cdot i_{sd} + L_m \cdot \left(\frac{1}{L'_r} \psi'_{rd} - \frac{L_m}{L'_r} i_{sd} \right) \right)$$

$$0 = r'_r \cdot \left(\frac{1}{L'_r} \psi'_{rd} - \frac{L_m}{L'_r} i_{sd} \right) + \frac{d\psi'_{rd}}{dt} - (\omega_s - \omega_r) \psi'_{rq}$$

$$0 = r'_r \cdot \left(\frac{1}{L'_r} \psi'_{rq} - \frac{L_m}{L'_r} i_{sq} \right) + \frac{d\psi'_{rq}}{dt} + (\omega_s - \omega_r) \psi'_{rd}$$

Simplificando las 4 ecuaciones

$$v_{sd} = r_s \cdot i_{sd} + \left(L_s - \frac{L_m^2}{L'_r} \right) \frac{di_{sd}}{dt} + \frac{L_m}{L'_r} \cdot \frac{d\psi'_{rd}}{dt} - \omega_s \cdot \left(\left(L_s - \frac{L_m^2}{L'_r} \right) \cdot i_{sq} + \frac{L_m}{L'_r} \cdot \psi'_{rq} \right) \quad (3.49)$$

$$v_{sq} = r_s \cdot i_{sq} + \left(L_s - \frac{L_m^2}{L'_r} \right) \cdot \frac{di_{sq}}{dt} + \frac{L_m}{L'_r} \cdot \frac{d\psi'_{rq}}{dt} + \omega_s \cdot \left(\left(L_s - \frac{L_m^2}{L'_r} \right) \cdot i_{sd} + \frac{L_m}{L'_r} \cdot \psi'_{rd} \right) \quad (3.50)$$

$$\frac{d\psi'_{rd}}{dt} = -\frac{r'_r}{L'_r} \psi'_{rd} + \frac{L_m \cdot r'_r}{L'_r} i_{sd} + (\omega_s - \omega_r) \psi'_{rq} \quad (3.51)$$

$$\frac{d\psi'_{rq}}{dt} = -\frac{r'_r}{L'_r} \psi'_{rq} + \frac{L_m \cdot r'_r}{L'_r} i_{sq} - (\omega_s - \omega_r) \psi'_{rd} \quad (3.52)$$

Reemplazando las ecuaciones (3.51-52) en ecuaciones (3.49-50) y haciendo que el sistema de coordenadas dq gire a la misma velocidad que el campo magnético del estator ω_s (velocidad sincrónica)

$$\frac{di_{sd}}{dt} = -\left(\frac{r_s}{\sigma L_s} + \frac{L_m^2 \cdot r'_r}{\sigma L_s L'_r} \right) \cdot i_{sd} + \frac{L_m \cdot r'_r}{\sigma L_s L'_r} \cdot \psi'_{rd} + \frac{1}{\sigma L_s} \cdot v_{sd} + \omega_s \cdot i_{sq} + \frac{L_m}{\sigma L_s L'_r} \cdot \omega_r \cdot \psi'_{rq} \quad (3.53)$$

$$\frac{di_{sq}}{dt} = -\left(\frac{r_s}{\sigma L_s} + \frac{L_m^2 \cdot r_r'}{\sigma L_s L_r'}\right) \cdot i_{sq} - \frac{L_m}{\sigma L_s L_r'} \cdot \omega_r \cdot \psi'_{rd} + \frac{1}{\sigma L_s} \cdot v_{sd} + \frac{L_m \cdot r_r'}{\sigma L_s L_r'} \cdot \psi'_{rq} \quad (3.54)$$

$$\frac{d\psi'_{rd}}{dt} = -\frac{r_r'}{L_r'} \psi'_{rd} + \frac{L_m \cdot r_r'}{L_r'} i_{sd} + (\omega_s - \omega_r) \psi'_{rq} \quad (3.55)$$

$$\frac{d\psi'_{rq}}{dt} = -\frac{r_r'}{L_r'} \psi'_{rq} + \frac{L_m \cdot r_r'}{L_r'} i_{sq} - (\omega_s - \omega_r) \psi'_{rd} \quad (3.56)$$

Utilizando el sistema de ecuaciones (3.53-56), estas pueden ser representadas con el siguiente modelo de espacio de estado continuo del motor de inducción.

$$\frac{dx}{dt} = Ax + Bu$$

$$\begin{bmatrix} \frac{di_{sd}}{dt} \\ \frac{di_{sq}}{dt} \\ \frac{d\psi'_{rd}}{dt} \\ \frac{d\psi'_{rq}}{dt} \end{bmatrix} = \begin{bmatrix} -\left(\frac{1}{\sigma T_s} + \frac{1-\sigma}{\sigma T_r}\right) & \omega_s & \frac{1-\sigma}{\sigma L_m T_r} & \frac{1-\sigma}{\sigma L_m} \omega_r \\ -\omega_s & -\left(\frac{1}{\sigma T_s} + \frac{1-\sigma}{\sigma T_r}\right) & -\frac{1-\sigma}{\sigma L_m} \omega_r & \frac{1-\sigma}{\sigma L_m T_r} \\ \frac{L_m}{T_r} & 0 & -\frac{1}{T_r} & (\omega_s - \omega_r) \\ 0 & \frac{L_m}{T_r} & -(\omega_s - \omega_r) & -\frac{1}{T_r} \end{bmatrix} \begin{bmatrix} i_{sd} \\ i_{sq} \\ \psi'_{rd} \\ \psi'_{rq} \end{bmatrix} + \begin{bmatrix} \frac{1}{\sigma L_s} & 0 \\ 0 & \frac{1}{\sigma L_s} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} v_{sd} \\ v_{sq} \end{bmatrix}$$

En donde:

$$\sigma = 1 - \frac{L_m^2}{L_s L_r'} \quad , \quad T_r = \frac{L_r'}{r_r'} \quad , \quad T_s = \frac{L_s}{r_s}$$

3.1.7 Discretización de modelo de motor de inducción

Para su implementación, es necesario discretizar sus ecuaciones en el dominio del tiempo.

La discretización se realiza mediante la aproximación de Euler hacia adelante (de primer orden), expresada como:

$$\dot{x}(t) \approx \frac{x(t+T) - x(t)}{T}$$

Donde $T \in \mathbb{R}^+$ representa el período de muestreo

Reorganizando la expresión, se obtiene:

$$x(t+T) = x(t) + TA x(t) + TB u(t)$$

Al considerar $t = kT$, con $k \in \mathbb{Z}$ como el índice de muestreo, se tiene:

$$x(t+T) = x[kT+T] = x[(k+1)T] = x_k + TA x_k + T u_k$$

Por lo tanto, el modelo discreto del sistema se expresa como:

$$x_{k+1} = [I + TA]x_k + [BT]u_k$$

Definiendo las matrices discretas:

$$A_d \approx I + AT, \quad B_d \approx BT$$

el modelo de estado en forma discreta queda finalmente representado por:

$$x_{k+1} = A_d x_k + B_d u_k$$

$$y_k = C x_k$$

$$A_d = \begin{bmatrix} 1 - \frac{T}{\sigma} \left(\frac{1}{T_s} + \frac{1-\sigma}{T_r} \right) & T\omega_s & \frac{T(1-\sigma)}{\sigma L_m T_r} & \frac{T(1-\sigma)}{\sigma L_m} \omega_r \\ 0 & 1 - \frac{T}{\sigma} \left(\frac{1}{T_s} + \frac{1-\sigma}{T_r} \right) & -\frac{T(1-\sigma)}{\sigma L_m} \omega_r & \frac{T(1-\sigma)}{\sigma L_m T_r} \\ \frac{TL_m}{T_r} & 0 & 1 - \frac{T}{T_r} & T(\omega_s - \omega_r) \\ 0 & \frac{TL_m}{T_r} & -T(\omega_s - \omega_r) & 1 - \frac{T}{T_r} \end{bmatrix}$$

$$B_d = \begin{bmatrix} \frac{T}{\sigma L_s} & 0 \\ 0 & \frac{T}{\sigma L_s} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

3.1.8 Torque electromagnético

El torque surge del proceso de conversión de energía eléctrica en energía mecánica, y puede determinarse de la división entre potencia mecánica y velocidad angular.

$$T_e = \frac{P_m}{\omega_m} = \frac{pP_m}{\omega_r} \quad (3.57)$$

En un motor de jaula de ardilla, no hay una fuente de voltaje conectada al rotor. Por lo tanto, la tensión aplicada externamente v_r es cero. Como el rotor no tiene alimentación externa, toda la energía que entra al sistema proviene exclusivamente de los terminales del estator.

$$P_T(t) = v_a(t)i_a(t) + v_b(t)i_b(t) + v_c(t)i_c(t) = (v_{sabc})^T \cdot i_{sabc}$$

$$P_T = [v_a \ v_b \ v_c] \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} = (v_{sabc})^T \cdot i_{sabc} \quad (3.58)$$

Basado en la transformación de Park total $v_{sabc} = T^{-1}v_{sdq0}$, y $i_{sabc} = T^{-1}i_{sdq0}$ se puede expresar como:

$$P_T = (T^{-1}v_{sdq0})^T \cdot T^{-1}i_{sdq0} \quad (3.59)$$

Considerando la siguiente analogía

$$x_{\alpha\beta 0} = K \cdot x_{abc} \quad x_{dq0} = R(\theta) \cdot x_{\alpha\beta 0}$$

$$x_{dq0} = R(\theta) \cdot K \cdot x_{abc} \quad x_{dq0} = T \cdot x_{abc}$$

$$P_T = (T^{-1}v_{sdq0})^T \cdot T^{-1}i_{sdq0} \quad T^{-1} = \frac{3}{2}T^T$$

Al sustituir, las matrices ortogonales se cancelan, resultando en:

$$P_T = \frac{3}{2}(v_{sdq0})^T \cdot i_{sdq0} \quad (3.60)$$

Simplificando, se obtiene:

$$P_T = \frac{3}{2}(v_{sd}i_{sd} + v_{sq}i_{sq}) \quad (3.61)$$

Sustituyendo los voltajes v_{sd} y v_{sq} por las ecuaciones (3.41-42)

$$P_T = \frac{3}{2} \left(r_s i_{sd} i_{sd} - \omega_s \psi_{sq} i_{sd} + \frac{d\psi_{sd}}{dt} i_{sd} + r_s i_{sq} i_{sq} + \omega_s \psi_{sd} i_{sq} + \frac{d\psi_{sq}}{dt} i_{sq} \right) \quad (3.62)$$

Agrupando términos agrupados

$$P_T = \frac{3}{2} \left(r_s i_{sd}^2 + r_s i_{sq}^2 + \frac{d\psi_{sd}}{dt} i_{sd} + \frac{d\psi_{sq}}{dt} i_{sq} + \omega_s \psi_{sd} i_{sq} - \omega_s \psi_{sq} i_{sd} \right) \quad (3.63)$$

En esta ecuación, el primer término corresponde a las pérdidas por efecto Joule en el estator, mientras que el segundo representa la potencia asociada a la variación temporal de la energía magnética almacenada. El último término corresponde a la potencia electromagnética transferida a través del entrehierro (air-gap power), la cual representa la energía que cruza del estator al rotor para hacer posible la conversión electromecánica.

Dado que la potencia instantánea es invariante al sistema de referencia utilizado (abc, $\alpha\beta$ o dq), se puede seleccionar un marco de referencia conveniente para simplificar el análisis. En particular, si se elige un sistema de referencia que gira solidario al rotor (haciendo $\omega_s = \omega_r$), la expresión de la potencia mecánica se simplifica a:

$$P_m = \frac{3}{2} \omega_r (\psi_{sd} i_{sq} - \psi_{sq} i_{sd}) \quad (3.64)$$

Reemplazando en ecuación de torque electromagnético (3.57)

$$T_e = \frac{3}{2} p (\psi_{sd} i_{sq} - \psi_{sq} i_{sd}) \quad (3.65)$$

Otras formas de representar

$$T_e = \frac{3}{2} \frac{L_m}{L_r} p (\psi'_{rd} i_{sq} - \psi'_{rq} i_{sd}) \quad (3.66)$$

3.2 Determinación de parámetros de motor de inducción

Para implementar la identificación de parámetros de un motor de inducción mediante el algoritmo de optimización por enjambre de partículas (Particle Swarm Optimization, PSO), es importante contar con un conjunto de datos experimentales obtenidos a través de ensayos eléctricos estandarizados, para modelar el comportamiento dinámico y en régimen permanente del motor, tanto en análisis transitorio como en estado estacionario.

Los parámetros del circuito equivalente de un motor de inducción (como la resistencia del estator, rotor, la inductancia de magnetización y las reactancias de dispersión tanto del estator como del rotor) se determinan habitualmente a través de tres pruebas experimentales: la prueba de corriente continua (CC), la prueba en vacío y la prueba de rotor bloqueado.

Estas pruebas, descritas de manera detallada en la norma IEEE 112, guardan similitud con los ensayos aplicados a transformadores, es decir, la prueba de circuito abierto y la prueba de cortocircuito (Chapman, 2012).

Con los datos experimentales obtenidos, es posible aplicar el algoritmo PSO (Particle Swarm Optimization) para ajustar los parámetros del modelo equivalente, minimizando la diferencia entre las respuestas medidas y las respuestas simuladas del motor.

La implementación de estas pruebas y su correspondiente análisis se llevó a cabo en un motor de inducción trifásico de rotor jaula de ardilla simple, de 1 HP, marca WEG, dentro del laboratorio. Los datos de placa de esta máquina se presentan en la tabla siguiente.

Tabla 2. Parámetros de placa y catalogo WEG

Voltaje	220ΔΔ/380YY/440Δ V
Corriente	3.07/1.78/1.54 A
Potencia	0.75 KW (1 HP)
Factor de Potencia	0.82
Velocidad de giro	1730 RPM
Frecuencia	60 Hz
Rendimiento	0.782
Corriente de arranque	18.9 A
Torque nominal	4.43 N.m
Torque de arranque	6.74 N.m



Figura 28. Placa de motor WEG de 1 HP IEC W22 Standard Efficiency

Se realiza conexión tipo 220 V Δ - Δ

3.3 Pruebas experimentales

3.3.1 Prueba de corriente continua para la evaluación de la resistencia del estator

La prueba de corriente continua (CC) permite determinar la resistencia del estator R_s . Se aplica una tensión continua entre dos terminales del estator y se ajusta la corriente hasta alcanzar su valor nominal. Al tratarse de corriente continua, la reactancia es nula y la única oposición al paso de corriente proviene de la resistencia de los devanados. La prueba se realiza a corriente nominal para obtener un valor representativo. La Figura 29 presenta el diagrama del ensayo con el estator conectado en configuración estrella (Y) (Chapman, 2012).

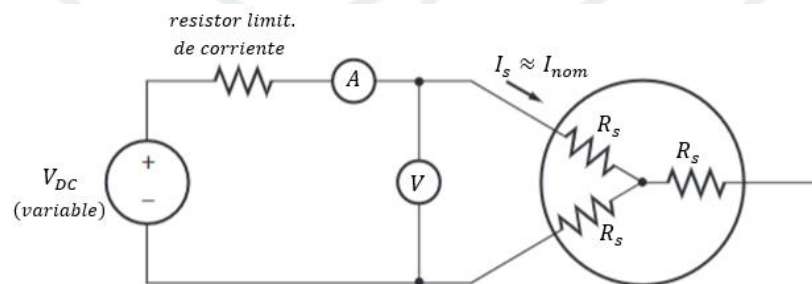


Figura 29. Circuito de prueba de una prueba de cd de resistencia (Chapman, 2012).

Para una conexión en estrella (Y):

$$2R_s = \frac{V_{DC}}{I_{DC}} \quad R_s = \frac{V_{DC}}{2I_{DC}} \quad (3.67)$$

Para conexión en delta:

$$R_s = \frac{3V_{DC}}{2I_{DC}} \quad (3.68)$$

En este caso, se considera el valor de catálogo:

$$R_s = 9.67 \, \Omega$$

3.3.2 Prueba en vacío

La prueba en vacío se realiza con el motor sin carga mecánica, con el fin de determinar las pérdidas rotacionales (fricción, ventilación y núcleo) y la corriente de magnetización. Para ello, se conecta el motor con los instrumentos de medición (watímetros, voltímetro y amperímetro) según el circuito mostrado en la Figura 30.

Bajo estas condiciones, el deslizamiento es muy pequeño ($s \approx 0.001$), por lo que la resistencia equivalente del rotor $\frac{R_r(1-s)}{s}$ es mucho mayor en comparación con las pérdidas en el cobre del rotor (R_r) y con la reactancia de este (X_r). Así, las pérdidas en el cobre del rotor son despreciables, y el circuito equivalente se simplifica a la rama de magnetización formada por R_C (pérdidas en el núcleo) y X_m (reactancia de magnetización) en paralelo.

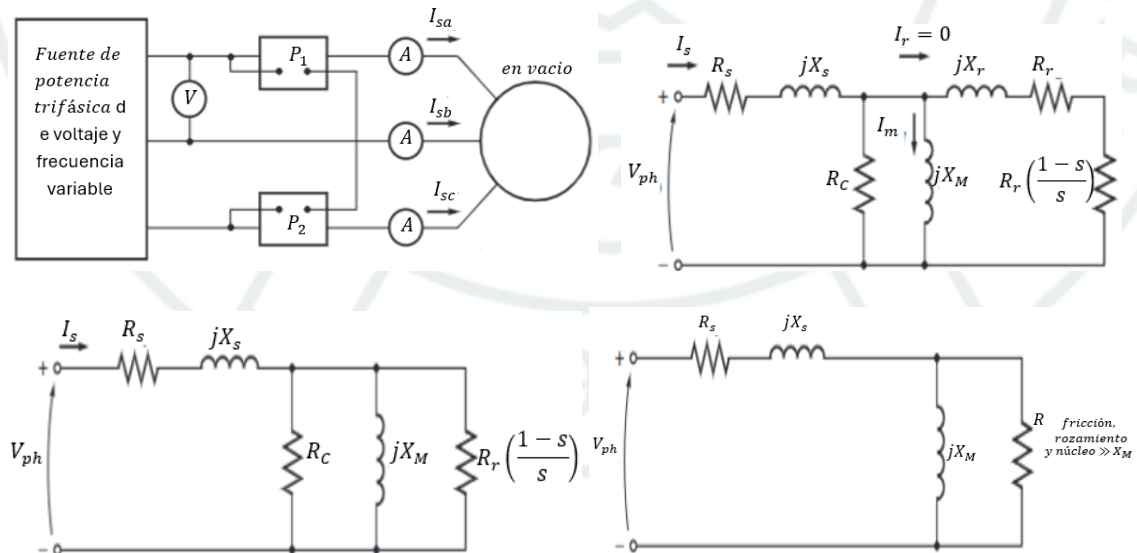


Figura 30. Circuito con pérdidas eléctricas (Chapman, 2012)

Puesto que $R_r \left(\frac{1-s}{s} \right) \gg R_r$ y $R_r \left(\frac{1-s}{s} \right) \gg X_r$, se desprecia y posteriormente se suma según lo mostrado en la Figura 30.

Cuando el motor opera en condiciones de vacío, la potencia de entrada registrada corresponde únicamente a las pérdidas internas de la máquina. En este escenario, las pérdidas en el cobre del rotor se consideran despreciables, ya que la corriente I_r es prácticamente nula debido al elevado valor de la resistencia equivalente $\frac{R_r(1-s)}{s}$. Por otro lado, las pérdidas en el cobre del estator (P_{PCE}) se determinan a partir de la siguiente expresión:

$$P_{PCE} = 3I_s^2 R_s \quad (3.69)$$

por lo que la potencia de entrada debe ser expresada como:

$$\begin{aligned} P_{entr} &= P_{PCE} + P_{núcl} + P_{FyR} + P_{misc} \\ &= 3I_s^2 R_s + P_{rot} \end{aligned} \quad (3.70)$$

donde $P_{rot} = P_{núcl} + P_{FyR} + P_{misc}$ representa las pérdidas rotacionales (núcleo, fricción, ventilación y misceláneas).

Debido a que $X_m \ll R_c$, la impedancia de entrada es principalmente inductiva, con un factor de potencia bajo:

$$|Z_{eq}| = \frac{v_{ph}}{I_{ph}} \approx X_s + X_m \quad (3.71)$$

Si bien la norma IEEE 112 establece como referencia el uso de fuentes de tensión senoidales puras mediante autotransformadores, debido a la no disponibilidad de dicho equipo en el laboratorio, la fuente de potencia trifásica de voltaje y frecuencia variable indicada en el circuito de la Figura 30 se implementó mediante un variador de frecuencia (VFD). Este equipo fue configurado a frecuencia fija de 60 Hz, permitiendo variar el voltaje de salida de forma controlada hasta alcanzar el valor nominal de la máquina. Dado que la señal de salida del VFD presenta modulación por ancho de pulso (PWM), se seleccionó una frecuencia de conmutación elevada de modo que la componente fundamental predomine sobre los armónicos, siendo el

efecto de estos últimos despreciable para las mediciones de corriente eficaz realizadas. Este procedimiento es consistente con trabajos previos que emplean VSD como fuente variable en ensayos de identificación de parámetros de motores de inducción.

Tabla 3. Registro de valores medidos de prueba de vacío

v_{ph}	I_{sa}	I_{sb}	I_{sc}	f	P_{ent}
220 V	1.8 A	1.78 A	1.8 A	60 Hz	202 W

$$I_s = \frac{I_{sa} + I_{sb} + I_{sc}}{3} = \frac{1.80 + 1.78 + 1.80}{3} = 1.8 \text{ A}$$

En conexión Δ

$$Z_{eq} = \frac{220}{1.8/\sqrt{3}} = 211.11 \Omega \approx X_s + X_m$$

$$P_{PCE} = 3I_s^2 R_s = 3 \cdot \left(\frac{1.8}{\sqrt{3}}\right)^2 \cdot 9.67 = 31.33 \text{ W}$$

$$P_{rot} = 202 - 31.33 = 171 \text{ W}$$

3.3.3 Prueba del rotor bloqueado

La prueba de rotor bloqueado (o rotor enclavado) se emplea para determinar las resistencias y reactancias equivalentes del estator y del rotor de un motor de inducción. Durante el ensayo, el rotor se mantiene inmovilizado y se aplica al estator una tensión reducida, ajustando su valor hasta que la corriente de línea alcance el valor nominal. Se registran el voltaje, la corriente y la potencia de entrada.

En estas condiciones, el deslizamiento es $s = 1$, por lo que la resistencia del rotor se representa directamente como R_r . Dado que R_r y X_r son pequeños en comparación con la reactancia de magnetización X_m , la corriente circula casi por completo por las ramas del estator y rotor, y el circuito equivalente se reduce a una serie formada por R_s, X_s, R_r, X_r .

El ensayo se realiza usualmente a la frecuencia nominal (50–60 Hz). Sin embargo, durante la operación normal, la frecuencia del rotor es mucho menor ($\approx 1\text{--}3$ Hz). Esta diferencia puede producir errores de estimación, especialmente en motores cuyas resistencias del rotor dependen de la frecuencia (clases B y C). Para minimizar este efecto, puede aplicarse una frecuencia menor ($\leq 25\%$ de la nominal), práctica adecuada para motores de resistencia casi constante (clases A y D) (Chapman, 2012).

La Figura 31 correspondiente muestra la configuración experimental de la prueba de rotor bloqueado en un motor de inducción, junto con su circuito de ensayo y el circuito equivalente.

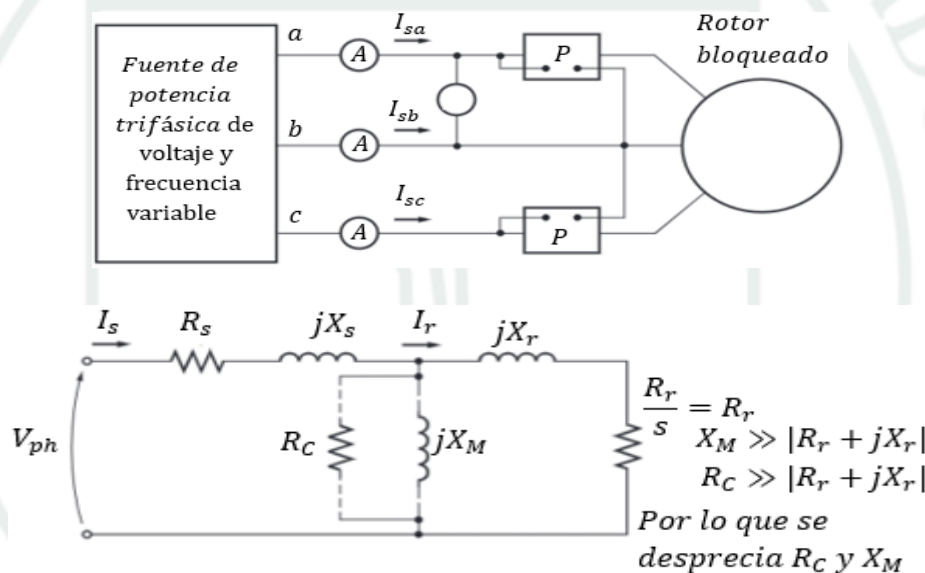


Figura 31. Prueba de rotor bloqueado de un motor de inducción: a) circuito de prueba; b) circuito equivalente del motor.

Una vez fijados los valores de tensión y frecuencia correspondientes al ensayo, se incrementa de forma controlada la corriente que circula por el motor hasta aproximarse a su nivel nominal. En ese punto se procede a registrar las magnitudes eléctricas relevantes, tales como la potencia absorbida, la tensión aplicada y la corriente consumida, cuidando que el rotor no experimente un aumento significativo de temperatura. A partir de estas mediciones, se establece la expresión que define la potencia de entrada del sistema como:

$$P_{ent} = \sqrt{3}V_T I_L \cos\theta \quad (3.72)$$

donde se obtiene el factor de potencia:

$$f.p. = \cos\theta = \frac{P_{ent}}{\sqrt{3}V_T I_L} \quad (3.73)$$

impedancia equivalente del motor se determina por:

$$|Z_{RB}| = \frac{V_{ph}}{I_s} \quad (3.74)$$

con un ángulo igual a θ . De esta forma, la impedancia equivalente queda como:

$$Z_{RB} = R_{RB} + jX'_{RB} = |Z_{RB}| \cos\theta + j|Z_{RB}| \sin\theta \quad (3.75)$$

La resistencia equivalente del rotor bloqueado resulta:

$$R_{RB} = R_r + R_s \quad (3.76)$$

y la reactancia del rotor bloqueado se expresa como:

$$X_{RB} = X'_s + X'_r \quad (3.77)$$

Donde X'_s y X'_r corresponden a las reactancias del estator y del rotor a la frecuencia utilizada en la prueba.

A partir de lo anterior, la resistencia del rotor puede obtenerse mediante:

$$R_r = R_{RB} - R_s \quad (3.78)$$

Siendo R_s el valor previamente determinado en la prueba de corriente continua. De forma análoga, la reactancia total del rotor referida al estator también puede calcularse, considerando que la reactancia es directamente proporcional a la frecuencia. Así, es posible ajustar su valor a la frecuencia nominal de operación del motor.

$$X_{RB} = \frac{f_{nom}}{f_{prueb}} X'_{RB} = X_s + X_r \quad (3.79)$$

Para la presente prueba, el VSD fue programado a una frecuencia de 15 Hz (25% de la frecuencia nominal), en concordancia con la recomendación de la norma IEEE 112 para motores de resistencia de rotor prácticamente constante (clases A y D). Esta configuración resulta particularmente conveniente con el uso del variador, ya que permite ajustar

simultáneamente voltaje y frecuencia de forma independiente, ventaja que no ofrece el autotransformador convencional.

Tabla 4. Registro de valores medidos de prueba de rotor bloqueado

v_{ph}	I_{sa}	I_{sb}	I_{sc}	f	P_{ent}
27 V	3.11 A	3.10 A	3.11 A	15 Hz	136 W

$$|Z_{RB}| = \frac{v_{ph}}{I_{ph}} = \frac{27}{3.1/\sqrt{3}} = 14.96 \Omega$$

$$R_{RB} = \frac{P_{ent}}{3I_{ph}^2} = \frac{136}{3 * (\frac{3.1}{\sqrt{3}})^2} = 14.15 \Omega$$

$$R_r = R_{RB} - R_s = 14.15 - 9.67 = 4.5 \Omega$$

$$X'_{RB} = \sqrt{Z_{RB}^2 - R_{RB}^2} = \sqrt{14.96^2 - 14.15^2} = 4.85 \Omega$$

Reactancia en frecuencia $f = 60 \text{ Hz}$

$$X_{RB} = \left(\frac{60}{15}\right) \cdot X'_{RB} = \left(\frac{60}{15}\right) \cdot 4.85 = 19.40 \Omega$$

Considerando $X_{RB} = X_s + X_r$ y $X_s \approx X_r$, entonces $X_s = X_r = 9.71 \Omega$

$$X_s = 2\pi f L_{ls} = 9.71 \Omega \quad L_{ls} = L'_{lr} = 25 \text{ mH}$$

Hallando reactancia de magnetización. De ecuación (3.71)

$$X_m = Z_{sc} - X_s = 211.11 - 9.71 = 201.4 \Omega$$

$$X_m = 2\pi f L_m = 201.4 \Omega \quad L_m = 535 \text{ mH}$$

Finalmente se presenta la tabla con los parámetros del motor

Tabla 5. Parámetros estimados

Parámetros	Valores	Parámetros	Valores
Potencia nominal (P)	0.75 KW	Inductancia de dispersión del rotor (L'_{lr})	25 mH
Voltaje (V)	220 V	Inductancia de magnetización (L_m)	535 mH
Frecuencia (f)	60 Hz	Momento de inercia (J)	0.0032 Kg.m^2
Resistencia del estator (R_s)	9.67Ω	Coefficiente de fricción (B)	$0.0001 \text{ Kg.m}^2/\text{s}$
Resistencia del rotor (R'_r)	4.50Ω	Pares de polos (p)	2
Inductancia de dispersión del estator (L_{ls})	25 mH	Tipo de motor	Jaula de Ardilla

Con los parámetros estimados se llevaron a cabo simulaciones en Simulink (usando el modelo de motor de inducción en el Anexo 3) para evaluar el comportamiento dinámico del motor de inducción.

En la siguiente sección se muestran los resultados alcanzados, destacando la respuesta temporal de la velocidad, el torque y las corrientes del estator.

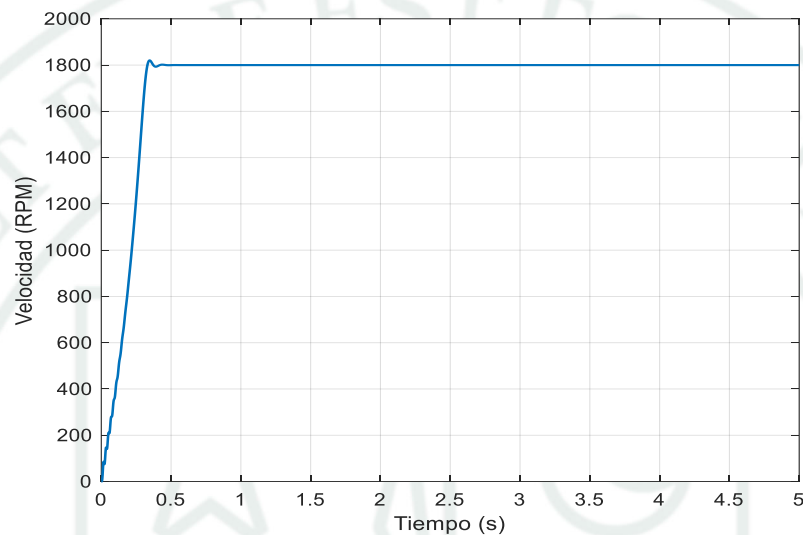


Figura 32. Variación de velocidad a lo largo del tiempo

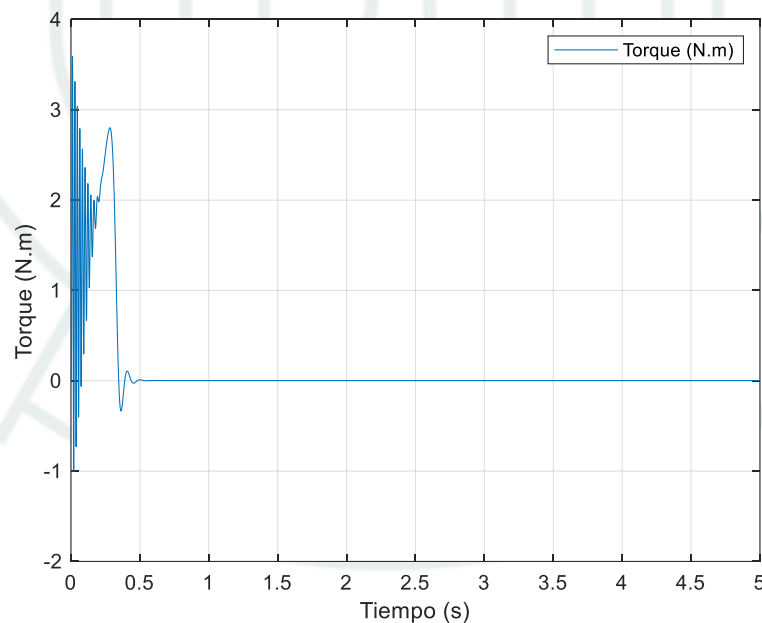


Figura 33. Variación de torque a lo largo del tiempo

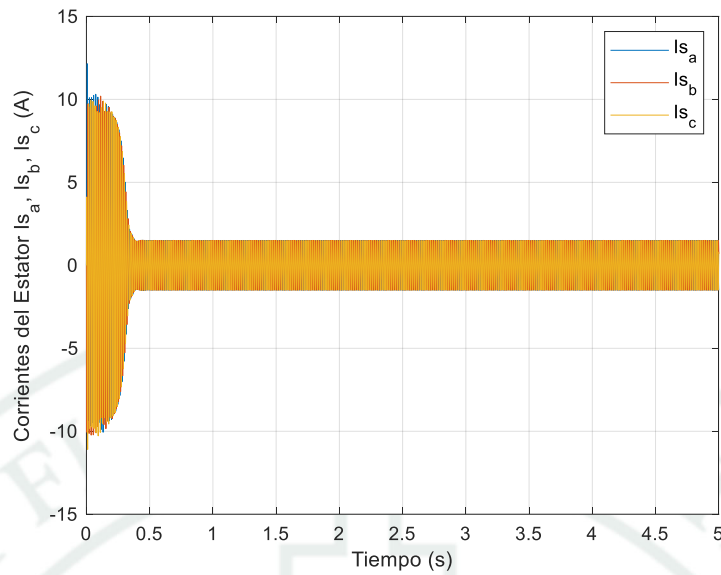


Figura 34. Variación de corrientes del estator a lo largo del tiempo

En la Figura 32 se observa que, durante los primeros 0.4 segundos, la velocidad del motor incrementa bruscamente desde el reposo hasta estabilizarse en torno a 1800 RPM, lo cual evidencia que la máquina ha alcanzado su régimen nominal de operación.

En la Figura 33 el comportamiento del torque electromagnético revela fuertes oscilaciones durante la fase de arranque (0–0.4 s), propias del régimen transitorio. Estas variaciones se van atenuando progresivamente hasta estabilizarse cerca de 0 N·m una vez transcurridos aproximadamente 0.5 segundos.

En la Figura 34 se observa que, durante el intervalo inicial comprendido entre 0 y 0.4 s, las corrientes correspondientes a las tres fases del estator (i_{sa} , i_{sb} , i_{sc}) exhiben variaciones pronunciadas y de alta amplitud. Este comportamiento está asociado a la demanda transitoria de corriente necesaria para generar el flujo magnético en el estator y vencer la resistencia mecánica impuesta por la inercia del rotor. A medida que transcurre el tiempo, dichas fluctuaciones disminuyen progresivamente hasta que las corrientes alcanzan valores más reducidos y prácticamente constantes, lo que evidencia que el motor ha alcanzado su régimen permanente de operación.

3.4 Identificación de parámetros de motor de inducción usando PSO

Los métodos tradicionales empleados para la estimación de parámetros en motores de inducción suelen basarse en las pruebas de rotor bloqueado y de vacío. Sin embargo, su aplicación práctica resulta compleja, principalmente porque requieren inmovilizar mecánicamente el rotor y equipos de medición caros. Frente a estas dificultades, surge la necesidad de recurrir a métodos alternativos, entre los cuales destaca la optimización mediante enjambre de partículas (PSO), implementada bajo dos metodologías distintas.

3.5 Determinación de parámetros de M.I. usando datos de manufactura

En el capítulo 2 fue explicado a detalle PSO, este método genera aleatoriamente un conjunto de partículas (soluciones candidatas) con posiciones y velocidades dentro del espacio de búsqueda. Cada partícula recuerda su mejor posición encontrada (pbest) y todas comparten la mejor solución global (gbest). En cada iteración, las partículas actualizan su velocidad en función de tres factores: inercia (mantener su movimiento previo), el componente cognitivo (atracción hacia su pbest) y el social (atracción hacia gbest), con términos aleatorios que favorecen la exploración. Luego, se actualizan las posiciones y se evalúa la función objetivo; si se mejora un pbest o gbest, se actualiza. El proceso continúa hasta que se alcanza el número máximo de iteraciones o un criterio de convergencia, y finalmente gbest se considera la mejor solución hallada.

3.5.1 Definición del problema

El motor de inducción trifásico puede representarse mediante un circuito equivalente en régimen permanente. La estimación de sus parámetros se formula como un problema de optimización por mínimos cuadrados, cuya función objetivo minimiza la diferencia entre los valores calculados y los datos nominales proporcionados en la placa del motor.

En este planteamiento se emplea información suministrada por el fabricante, como el torque de arranque, el torque nominal, el torque máximo y el factor de potencia a plena carga.

A partir de estos datos, el método propuesto permite estimar los parámetros del circuito equivalente en estado estable, incluyendo la resistencia del estator y del rotor, las reactancias de dispersión y la reactancia de magnetización.

Para el cálculo del torque de arranque, nominal y máximo en función de los parámetros a estimar, se adopta el procedimiento descrito por Chapman. En este contexto, resulta necesario disponer de una expresión analítica del torque electromagnético inducido en el motor de inducción (M.I.).

El torque electromagnético puede obtenerse a partir de las siguientes expresiones:

$$P_m = P_{EH}(1 - s)$$

$$T_e = \frac{P_m}{\omega_m} \quad (3.80)$$

$$T_e = \frac{P_{EH}}{\omega_{sinc}} \quad (3.81)$$

La ecuación (3.81) resulta particularmente conveniente, ya que la velocidad síncrona permanece constante para una frecuencia y un número de polos determinados. Por tanto, el cálculo del torque inducido se reduce a la determinación de la potencia en el entrehierro (Chapman, 2012).

La potencia en el entrehierro corresponde a la potencia disipada en la resistencia equivalente del rotor R_r/s . Para una fase del motor, dicha potencia está dada por:

$$P_{EH,1\phi} = I_r^2 \frac{R_r}{s}$$

y la potencia total en el entrehierro resulta:

$$P_{EH} = 3I_r^2 \frac{R_r}{s} \quad (3.82)$$

En consecuencia, si se logra calcular la corriente I_r , se puede conocer de manera directa tanto la potencia transmitida en el entrehierro como el torque electromagnético inducido en el motor.

Para simplificar el análisis del circuito equivalente, se aplica el teorema de Thévenin a la parte del circuito situada a la izquierda del rotor (Figura 35). De este modo, el circuito puede representarse mediante una fuente de voltaje equivalente en serie con una impedancia equivalente, lo que facilita el cálculo de la corriente del rotor.

Para obtener el equivalente de Thévenin, se abren los terminales en X y se calcula el voltaje en circuito abierto. Luego, se cortocircuita la fuente de fase y se determina la impedancia vista desde esos terminales. Finalmente, mediante el divisor de voltaje, se obtiene la expresión del voltaje equivalente:

$$V_{TH} = V_{ph} \frac{Z_M}{Z_M + Z_s}$$

La magnitud del voltaje de Thevenin V_{TH} es

$$V_{TH} = V_{ph} \frac{X_M}{\sqrt{R_s^2 + (X_s + X_M)^2}}$$

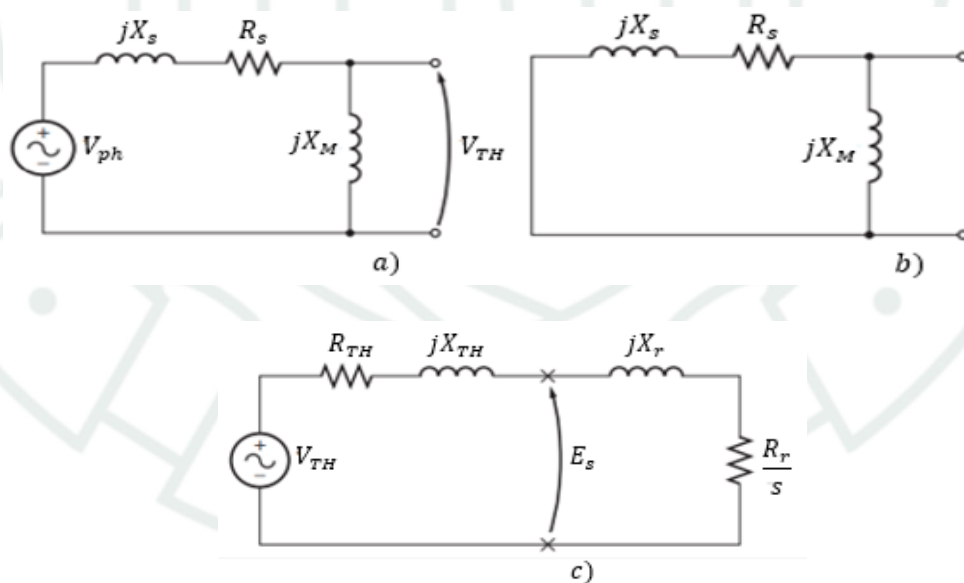


Figura 35. a) Voltaje equivalente de Thévenin para el circuito de entrada de un motor de inducción. b) Impedancia equivalente de Thévenin en el circuito de entrada. c) Circuito equivalente simplificado resultante para el motor de inducción. (Chapman, 2012)

Dado que la reactancia de magnetización X_M es mucho mayor que X_s y también que R_s , el valor del voltaje equivalente de Thévenin puede considerarse, con un alto grado de precisión, prácticamente igual al voltaje de entrada.

$$V_{TH} \approx V_{ph} \frac{X_M}{X_s + X_M} \quad (3.83)$$

En la Figura 35 se observa el circuito de entrada en condición de cortocircuito de la fuente. En este caso, las dos impedancias se encuentran en paralelo, lo que permite expresar la impedancia equivalente de Thévenin de la siguiente manera:

$$Z_{TH} = \frac{Z_s Z_M}{Z_M + Z_s}$$

Esta impedancia se reduce a

$$Z_{TH} = R_{TH} + jX_{TH} \frac{jX_M(R_s + jX_s)}{R_s + j(X_s + X_M)}$$

Como $X_M + X_s \gg R_s$, la expresión puede simplificarse y se obtiene una aproximación muy precisa para la resistencia y reactancia equivalentes de Thévenin.

$$Z_{TH} = \frac{jX_M(R_s + jX_s)}{j(X_s + X_M)}$$

Finalmente, al multiplicar por su conjugada se logra una expresión más sencilla que facilita los cálculos en el análisis del circuito.

$$\begin{aligned} Z_{TH} &= \frac{jX_M(R_s + jX_s)}{j(X_s + X_M)} * \frac{-j(X_s + X_M)}{-j(X_s + X_M)} = \frac{X_M R_s (X_s + X_M) + jX_s X_M (X_s + X_M)}{(X_s + X_M)^2} \\ Z_{TH} &= \frac{R_s X_M}{X_s + X_M} + j \frac{X_s X_M}{X_s + X_M} \\ R_{TH} &= \frac{R_s X_M}{X_s + X_M}, \quad X_{TH} = \frac{X_s X_M}{X_s + X_M} \end{aligned} \quad (3.84)$$

A partir de las ecuaciones y circuito equivalente (Figura 35-c) se obtiene I_r

$$I_r = \frac{V_{TH}}{Z_{TH} + Z_r} = \frac{V_{TH}}{R_{TH} + \frac{R'_r}{s} + jX_{TH} + X'_r} \quad (3.85)$$

Por lo tanto, la potencia en el entrehierro está dada por

$$P_{EH} = 3I_r^2 \frac{R_r}{s} = \frac{3V_{TH}^2 \frac{R_r'}{s}}{\left(R_{TH} + \frac{R_r'}{s}\right)^2 + (X_{TH} + X_r')^2} \quad (3.86)$$

Y el torque inducido del rotor está dado por:

$$T_e = T_{fl} = \frac{P_{EH}}{\omega_{sinc}} = \frac{3V_{TH}^2 \frac{R_r'}{s}}{\omega_{sinc} \left[\left(R_{TH} + \frac{R_r'}{s}\right)^2 + (X_{TH} + X_r')^2 \right]} \quad (3.87)$$

El torque de arranque ocurre cuando el rotor está detenido, es decir, deslizamiento $s = 1$, por lo que la ecuación es:

$$T_{st} = \frac{3V_{TH}^2 R_r'}{\omega_{sinc} [(R_{TH} + R_r')^2 + (X_{TH} + X_r')^2]} \quad (3.88)$$

El torque máximo, se obtiene derivando la ecuación general respecto al deslizamiento e igualando a cero $\left(\frac{dT_e}{ds} = 0\right)$. Ocurre a un deslizamiento específico s_{max}

$$T_{max} \approx \frac{3V_{TH}^2}{2 \cdot \omega_{sinc} \left[R_{th} + \sqrt{R_{th}^2 + (X_{th} + X_r')^2} \right]} \quad (3.89)$$

El planteamiento del problema se basa en la información proporcionada por el fabricante del motor de inducción trifásico, la cual incluye el torque de arranque, el torque a plena carga, el torque máximo y el factor de potencia nominal. Estos datos se emplean para estimar los parámetros del circuito equivalente en régimen permanente, entre los que se incluyen la resistencia del estator, la resistencia del rotor referida al estator, las reactancias de dispersión y la reactancia de magnetización.

Para el análisis se utiliza el circuito equivalente de Thévenin, considerando los parámetros del rotor referidos al lado del estator. Con el fin de simplificar el modelo, se asume que las pérdidas en el núcleo magnético son despreciables (Mohammadi & Akhavan, 2014).

A partir del modelo eléctrico, se definen las siguientes funciones de error asociadas al torque a plena carga, torque de arranque y torque máximo, respectivamente:

$$f_1 = \frac{\frac{K_t R'_r}{s \left[\left(R_{th} + \frac{R'_r}{s} \right)^2 + (X_{th} + X'_r)^2 \right]} - T_{fl}}{T_{fl}} \quad (3.90)$$

$$f_2 = \frac{\frac{K_t R'_r}{[(R_{th} + R'_r)^2 + (X_{th} + X'_r)^2]} - T_{st}}{T_{st}} \quad (3.91)$$

$$f_3 = \frac{\frac{K_t}{2 \left[R_{th} + \sqrt{R_{th}^2 + (X_{th} + X'_r)^2} \right]} - T_{max}}{T_{max}} \quad (3.92)$$

Para el cálculo del factor de potencia y la corriente de fase del estator, se emplea el circuito equivalente completo, cuyas impedancias están dadas por:

$$\begin{aligned} Z_s &= R_s + jX_s & Z_m &= jX_m & Z_r &= \frac{R'_r}{s} + jX_r \\ Z_{eq} &= Z_s + \frac{Z_m \cdot Z_r}{Z_m + Z_r} = R_s + jX_s + \frac{jX_m \cdot \left(\frac{R'_r}{s} + jX_r \right)}{\frac{R'_r}{s} + j(X_m + X_r)} \end{aligned} \quad (3.93)$$

La corriente de fase se calcula como:

$$I_{sfc} = \frac{V}{R_s + jX_s + \frac{jX_m \cdot \left(\frac{R'_r}{s} + jX_r \right)}{\frac{R'_r}{s} + j(X_m + X_r)}} \quad (3.94)$$

Asumiendo el voltaje de fase como referencia, el factor de potencia se obtiene a partir del ángulo de la corriente, definiéndose la función de error:

$$f_4 = \frac{\cos \left(\arg(I_{sf}) \right) - P_{fl}}{P_{fl}} \quad (3.95)$$

Asimismo, considerando la magnitud de la corriente de fase, se define:

$$f_5 = \frac{|I_{sfc}| - I_{sf}}{I_{sf}} \quad (3.96)$$

donde I_{sf} corresponde a la corriente de fase.

Los parámetros T_{fl} , T_{st} , T_{max} representan el torque de arranque, el torque a plena carga y el torque máximo, respectivamente, mientras que, pf_{fl} denota el factor de potencia nominal a plena carga. El coeficiente $K_t = \frac{3V_{TH}^2}{\omega_{sinc}}$ se considera constante dentro del modelo.

Finalmente, la función objetivo del problema de optimización se define como la minimización de la suma de los cuadrados de los errores:

$$F = f_1^2 + f_2^2 + f_3^2 + f_4^2 + f_5^2 \quad (3.97)$$

La Tabla 6 presenta los datos empleados para la implementación del algoritmo de Optimización por Enjambre de Partículas (PSO). El código desarrollado para esta aplicación se encuentra disponible en el Anexo 2.

Tabla 6. Rangos de valores de entrada

Parámetro	Valor (PSO Global)
Iteraciones (N)	800
Valores mínimos (R_s, R_r, X_{ls}, X_m)	[5.0, 2.0, 5.0, 150]
Valores máximos (R_s, R_r, X_{ls}, X_m)	[15.0, 8.0, 15.0, 250]
Número de partículas (i)	100
Inercia inicial (Ω_{min})	0.4
Inercia final (Ω_{max})	0.9
Componente cognitivo (ϕ_1)	1.5
Componente social (ϕ_2)	2.0

3.5.2 Resultados

En la Figura 36, se observa los resultados obtenidos de torque, eficiencia, corriente del estator, factor de potencia obtenido por PSO y pruebas experimentales

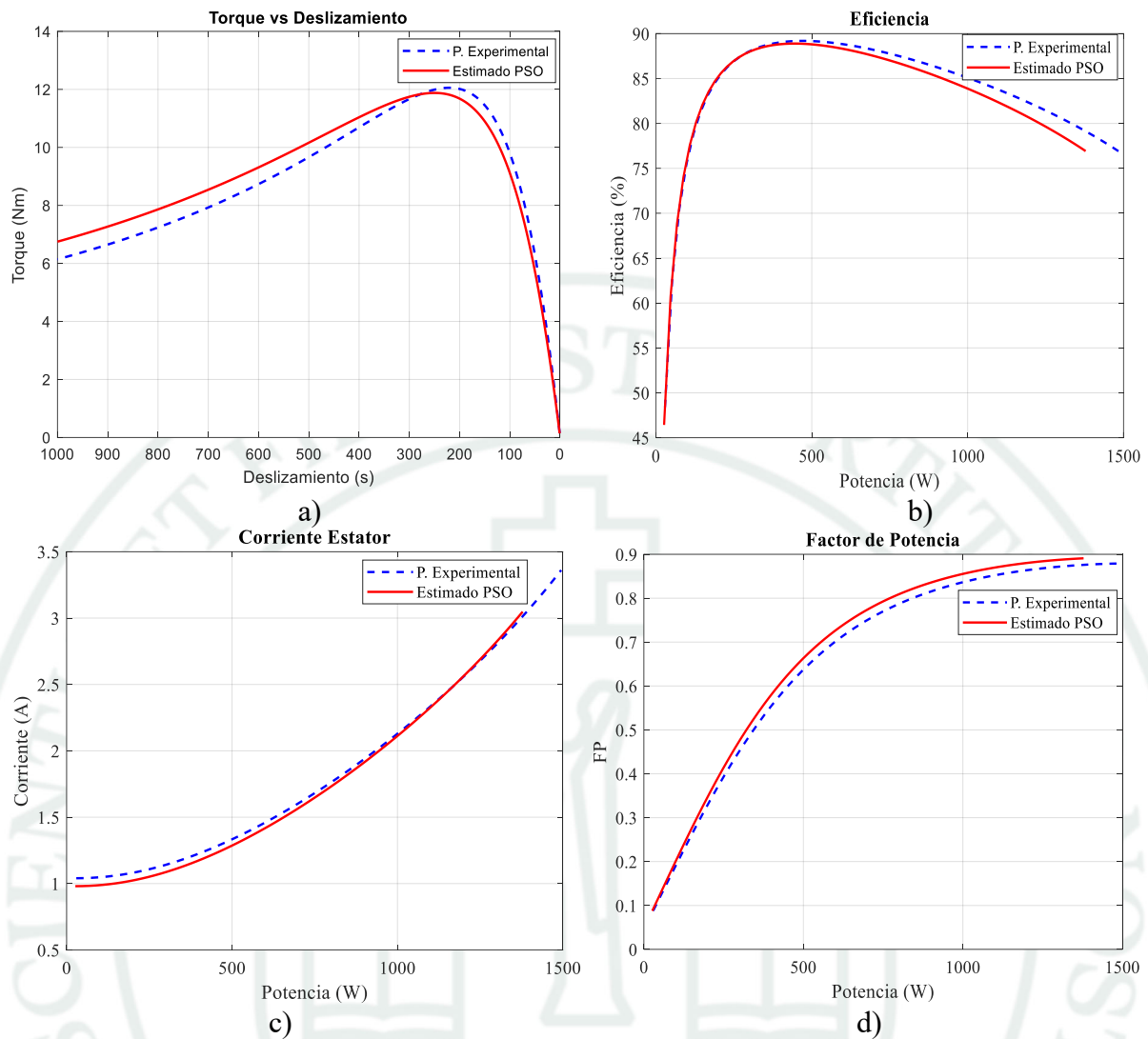


Figura 36. Análisis comparativo entre los valores originales y los obtenidos mediante PSO:

a) Características Torque-Deslizamiento. b) Curva de Eficiencia. c) Curva de Corriente del Estator. d) Curva del Factor de Potencia

En la Figura 37, se observa el número de iteraciones y evolución de la función de costo

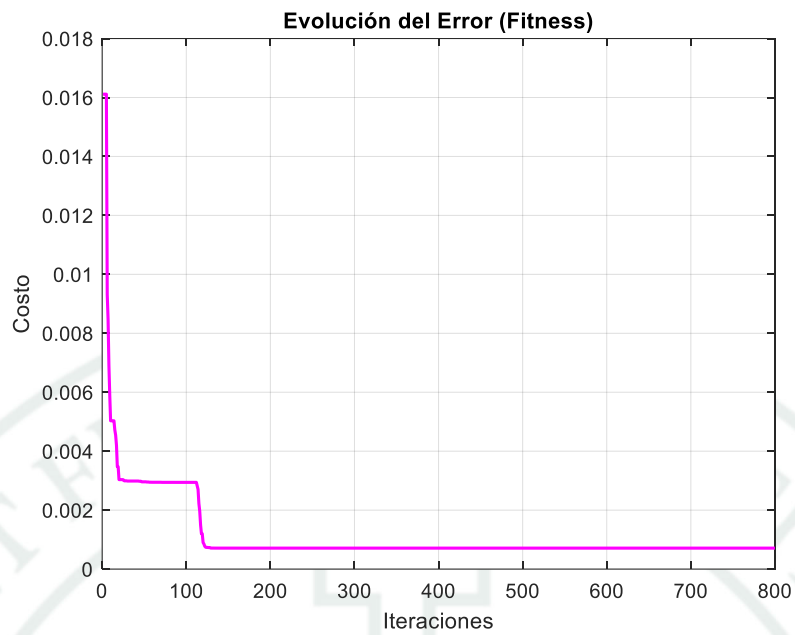


Figura 37. Evolución de la función de costo vs número de iteraciones

La tabla 7, muestra el análisis comparativo entre los resultados simulados por el PSO y los datos experimentales arroja errores que oscilan entre un 4.3% y un 11.97%. Estos márgenes validan la eficacia del método propuesto

Tabla 7. Análisis comparativo entre valores

Parámetros del motor	Valor experimental	Valor calculado	%Error
R_s	9.67	10.54	9.8%
R_r	4.53	5.03	11.97%
X_{ls}	9.42	9.01	4.30%
X_m	201.63	214.95	6.57%

CAPÍTULO IV: CONTROL VECTORIAL SENSORLESS DE MOTOR DE INDUCCIÓN

Históricamente, los motores de corriente continua (DC) dominaron las aplicaciones de velocidad variable debido a su excelente comportamiento dinámico. En estas máquinas, gracias a la acción mecánica del conmutador, el flujo magnético y el torque electromagnético están naturalmente desacoplados y pueden controlarse de forma independiente mediante la corriente de excitación y la corriente de armadura, respectivamente. No obstante, las máquinas DC presentan desventajas significativas, como el desgaste de escobillas y conmutadores, la generación de chispas y altos costos de mantenimiento.

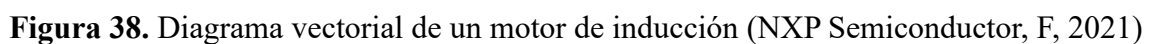
El desarrollo de la electrónica de potencia y los microprocesadores permitió sustituir las máquinas DC por motores de inducción (AC), los cuales son más robustos y económicos. Inicialmente, el control de estos motores se realizaba mediante métodos escalares, variando la magnitud del voltaje y la frecuencia para mantener constante la relación V/f . Aunque simple, el control escalar presenta un rendimiento dinámico limitado y dificultad para controlar el par a bajas velocidades debido al acoplamiento inherente entre el flujo y el torque en las máquinas de AC.

Para superar estas limitaciones, se desarrolló el Control Orientado al Campo (Field-Oriented Control, FOC) o Control Vectorial. Fue introducido inicialmente por Hasse en 1969 y posteriormente desarrollado por Blaschke en 1972. Esta estrategia de control permite desacoplar el control del torque y del flujo en motores de inducción, emulando el comportamiento de una máquina DC mediante transformaciones matemáticas aplicadas a las variables eléctricas del estator y rotor (Aparício, 2019).

Existen dos enfoques principales dentro del control orientado al campo. El control directo de campo orientado (DFOC), propuesto por Blaschke, se basa en la medición directa del flujo del motor mediante sensores o estimadores específicos. Por otro lado, el control

El control vectorial ofrece ventajas importantes frente al escalar ofreciendo control preciso del torque incluso a bajas velocidades, mejor comportamiento dinámico, mayor eficiencia energética y operación estable en un amplio rango de velocidades.

En el control vectorial, se utiliza un sistema de coordenadas rotativo dq que gira a la velocidad de sincronismo del motor. Este sistema se orienta de manera que el vector de flujo del rotor quede siempre alineado con el eje d. Esta alineación permite que la corriente encargada de generar torque esté completamente orientada en el eje q, el cual es ortogonal al flujo. De esta forma, se maximiza la producción de torque (NXP Semiconductor, F, 2021). La Figura 38 muestra los vectores de tensión del estator, corriente del estator y flujo del entrehierro en coordenadas d-q. La orientación del eje d se establece en dirección del vector de flujo del rotor.



Como resultado de esta orientación, la componente del flujo del rotor sobre el eje q se anula ($\psi_{rq} = 0$) quedando todo el flujo del rotor alineado en el eje d. En consecuencia, la corriente i_{sq} (ubicada en el eje q) se convierte en la única responsable del torque electromagnético, mientras que la corriente i_{sd} (en el eje d) regula exclusivamente el nivel del flujo magnético.

4.2. Algoritmo de control vectorial de motores AC

El método adoptado en este estudio para llevar a cabo la implementación del IFOC corresponde al esquema ilustrado en la Figura 39:

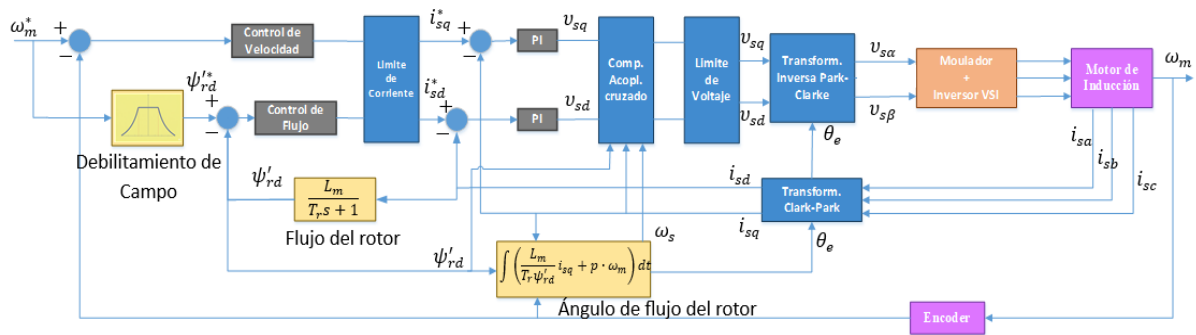


Figura 39. Diagrama de flujo de control vectorial propuesto – Elaboración propia

A partir de la Figura 39 mencionada, puede identificarse la arquitectura fundamental del control vectorial aplicado a motores de inducción. La ejecución de esta estrategia se basa en una secuencia ordenada de acciones, las cuales permiten tener un control del flujo y del torque electromagnético separados. Dicho proceso se desarrolla siguiendo una serie de etapas bien definidas, que se describen a continuación.

- Se miden las corrientes de fase i_{sa} y i_{sc} . La corriente i_{sb} no es necesaria medir ya que $i_{sa} + i_{sb} + i_{sc} = 0$. Además, se mide la velocidad del rotor mediante el encoder.
- Las corrientes trifásicas medidas se transforman al marco estacionario $\alpha\beta$ mediante la transformada de Clarke. $\alpha\beta$: $i_{s\alpha}, i_{s\beta}$.

- Con el ángulo eléctrico θ_e obtenido a partir de la velocidad medida, corriente i_{sq} y flujo del rotor estimado ψ'_{rd} , se aplica la transformada de Park, pasando al sistema rotativo dq y obteniendo i_{sd}, i_{sq} .
- Se estima el flujo del rotor, basado en corriente y parámetros del motor de inducción.
- Se establece la referencia de velocidad y se compara con la velocidad medida por el encoder. El error se procesa mediante un control PI, cuya salida define la referencia de torque y, por lo tanto, el valor de i_{sq}^* . La referencia i_{sd}^* se asocia al control de flujo. En caso la velocidad de referencia es mayor a la velocidad nominal se hace uso del control por debilitamiento de campo.
- Las corrientes de referencia (i_{sd}^*, i_{sq}^*) se comparan con las corrientes reales (i_{sd}, i_{sq}) y, mediante controladores PI, se obtienen las tensiones de referencia v_{sd}^*, v_{sq}^* .
- Estas tensiones v_{sd}^*, v_{sq}^* se transforman nuevamente al marco estacionario $\alpha\beta$ (inversas de Park y Clarke), resultando en $v_{s\alpha}, v_{s\beta}$.
- A partir de $v_{s\alpha}, v_{s\beta}$, el modulador SVPWM (Space Vector Pulse Width Modulation) genera las señales PWM que controlan al inversor, aplicando las tensiones necesarias hacia el motor.
- El sistema en lazo cerrado ajusta continuamente torque y velocidad en cada iteración para mantener la referencia establecida.

Para derivar las expresiones matemáticas asociadas al IFOC, resulta indispensable replantear el modelo dinámico del motor de inducción. Tal como se indicó en la introducción de esta estrategia, la obtención de las ecuaciones que gobiernan el control indirecto de campo exige que se cumplan ciertos supuestos básicos.

En primer lugar, se considera que el flujo del rotor se encuentra completamente alineado con el eje directo del sistema de referencia, lo que implica que solo existe componente en dicho eje y que la componente en cuadratura es nula. Esta condición se expresa como:

$$\psi'_r = \psi'_{rd} \quad (4.1)$$

$$\psi_{rq} = 0 \quad (4.2)$$

Adicionalmente, el sistema de coordenadas debe girar solidariamente con el campo magnético del rotor. Por lo tanto, la velocidad de deslizamiento definida por la diferencia entre la velocidad eléctrica del estator y la velocidad eléctrica del rotor se expresa como:

$$\omega_{sl} = \omega_s - \omega_r \quad (4.3)$$

Al sustituir estas condiciones en las ecuaciones del modelo del motor, se obtienen las siguientes relaciones para las tensiones del estator y las ecuaciones del rotor (3.53-3.56):

$$v_{sd} = r_s \cdot i_{sd} + \sigma L_s \frac{di_{sd}}{dt} + \frac{L_m}{L'_r} \frac{d\psi'_{rd}}{dt} - \sigma L_s \omega_s \cdot i_{sq} \quad (4.4)$$

$$v_{sq} = r_s \cdot i_{sq} + \sigma L_s \frac{di_{sq}}{dt} + \sigma L_s \omega_s \cdot i_{sd} + \omega_s \frac{L_m}{L'_r} \cdot \psi'_{rd} \quad (4.5)$$

$$0 = \frac{r'_r}{L'_r} \psi'_{rd} - \frac{L_m \cdot r'_r}{L'_r} i_{sd} + \frac{d\psi'_{rd}}{dt} \quad (4.6)$$

$$0 = -\frac{L_m \cdot r'_r}{L'_r} i_{sq} + (\omega_s - \omega_r) \psi'_{rd} \quad (4.7)$$

A partir de la ecuación correspondiente al eje directo del rotor, es posible definir una corriente magnetizante equivalente, la cual desempeña un papel análogo a la corriente de excitación en una máquina de corriente continua. Esta corriente, denotada como i_m se obtiene a partir de (4.6), de donde se deduce:

$$L_m i_{sd} = \psi'_{rd} + \frac{L'_r}{r'_r} \frac{d\psi'_{rd}}{dt} \quad (4.8)$$

Al comparar esta relación con la ecuación de la corriente de campo de una máquina de corriente continua, representada en la Figura 40, se tiene:

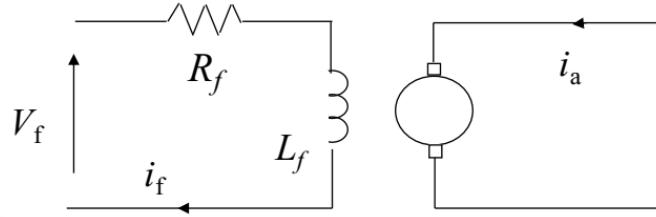


Figura 40. Circuito de máquina CC (Le Fosse, 2021)

$$V_f = R_f i_f + L_f \frac{di_f}{dt} \quad \frac{L_f}{R_f} \frac{di_f}{dt} + i_f = \frac{V_f}{R_f}$$

i_f se puede definir como la corriente de magnetización

En régimen estacionario, o cuando el flujo del rotor se mantiene aproximadamente constante, el término dinámico se anula y la expresión anterior (4.8) se reduce a:

$$L_m i_{sd} = \psi'_{rd} \quad (4.9)$$

De este modo, la corriente i_{sd} desempeña un papel funcionalmente equivalente al de la corriente de campo i_f de la máquina de corriente continua, siendo la responsable del establecimiento del flujo magnético.

Considerando las ecuaciones anteriores, pueden expresarse las relaciones que gobiernan el torque electromagnético, la velocidad angular y el flujo del rotor. En particular, el torque desarrollado por la máquina queda definido como:

$$T_e = \frac{3}{2} p \frac{L_m}{L_r} \psi'_{rd} i_{sq} \quad (4.10)$$

De ecuación (4.7), se estima el ángulo del rotor

$$(\omega_s - \omega_r) = \frac{L_m \cdot r'_r}{L_r \psi'_{rd}} i_{sq}$$

$$\theta_s = \int \left(\frac{L_m}{T_r \psi'_{rd}} i_{sq} + \omega_r \right) dt \quad (4.11)$$

Analizando (4.6) en Laplace se tiene:

$$\psi'_{rd} = \frac{L_m i_{sd}}{sT_r + 1} \quad (4.12)$$

4.3. Control a flujo debilitado

Para alcanzar velocidades superiores a la nominal en un motor de inducción, es necesario debilitar el campo magnético del rotor, ya que la tensión del estator debe mantenerse limitada a su valor nominal. Esta condición impone restricciones al flujo magnético, lo que requiere estrategias específicas de control. Existen diversos métodos para implementar el control en modo de debilitamiento de flujo, que permiten extender el rango de velocidad sin sobrepasar los límites de tensión del inversor ni comprometer la estabilidad del sistema.

4.3.1. Método convencional de control a flujo debilitado

Al operar con la corriente de torque nominal i_{sq} , en la región de debilitamiento de flujo, la potencia máxima que puede entregar la máquina se mantiene aproximadamente constante. Por esta razón, la máquina de inducción presenta al menos dos zonas características de operación (Shen, 2023):

- **Zona de flujo nominal**, donde el torque es constante.
- **Zona de flujo debilitado**, donde la potencia máxima permanece constante mientras el torque disminuye con el incremento de velocidad.

Esta transición entre regiones se ilustra con mayor detalle en la Figura 41.

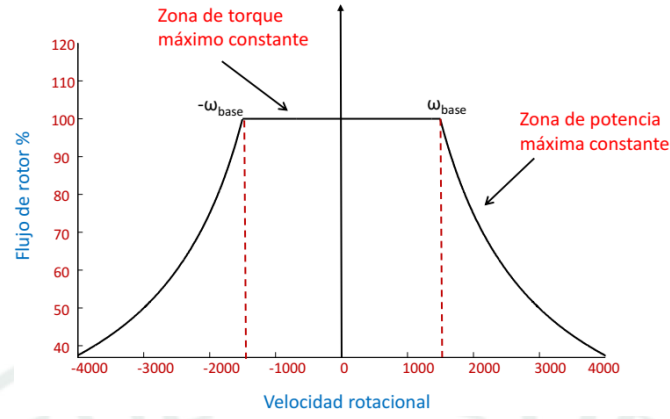


Figura 41. Zonas de operación a flujo nominal y debilitado (Shen, 2023)

El flujo del rotor es inversamente proporcional a velocidad del motor en la región de operación de flujo debilitado. Cuando $\omega_m > \omega_{nom}$

$$\psi'_{rd} \propto \frac{1}{\omega_m}$$

De este modo, en la región de debilitamiento de campo ($\omega_m > \omega_{nom}$), el flujo del rotor puede expresarse como:

$$\psi'_{rd} = \left(\frac{\omega_{nom}}{\omega_m} \right) \psi'_{nom}$$

Para determinar el flujo del rotor ψ'_{rd} cuando $\omega_m < \omega_{nom}$ se realiza los siguientes cálculos:

De ecuaciones (3.47) en (3.45)

$$\psi_{sd} = L_s \cdot i_{sd} + L_m \cdot \left(\frac{\psi'_{rd} - L_m \cdot i_{sd}}{L'_r} \right)$$

Simplificando:

$$\psi_{sd} = \sigma L_s i_{sd} + \frac{L_m}{L'_r} \psi'_{rd} \quad (4.13)$$

Multiplicando por L'_r

$$L'_r \psi_{sd} - L_m \psi'_{rd} = \sigma L_s L'_r i_{sd} \quad (4.14)$$

Bajo la operación de régimen estacionario del IFOC

$$\psi'_{rd} = L_m i_{sd}$$

Sustituyendo (4.9) en (4.14) se obtiene:

$$L'_r \psi_{sd} = \sigma L_s L'_r i_{sd} - L_m^2 i_{sd}$$

Simplificando

$$\psi_{sd} = L_s i_{sd} \quad (4.15)$$

Usando la ecuación de voltaje de estator (3.42) bajo régimen estacionario

$$v_{sq} = r_s \cdot i_{sq} + \frac{d\psi_{sq}}{dt} + \omega_s \cdot \psi_{sd} \rightarrow v_{sq} = r_s \cdot i_{sq} + \omega_s \cdot \psi_{sd} \quad (4.16)$$

Remplazando (4.15) en (4.16)

$$v_{sq} = r_s \cdot i_{sq} + \omega_s \cdot L_s \cdot i_{sd}$$

Durante la operación sin carga en estado estable $i_{sq} = 0$, despreciando las pérdidas por fricción y ventilación para el cálculo de la corriente de magnetización nominal

$$v_{sq} = \omega_s \cdot L_s \cdot i_{sd} \quad (4.17)$$

Asumiendo que todo el vector de voltaje está en el eje q, entonces $v_{sq} = V_{fase-pico}$, y asumiendo que ω_s es la velocidad nominal (ω_{nom}) se obtiene

$$i_{sd} = \frac{V_{fase-pico}}{\omega_{nom} L_s} = \frac{\sqrt{2} V_{nom}}{\sqrt{3} \omega_{nom} L_s}$$

Reemplazando (4.17) en (4.9) se obtiene

$$\psi'_{rd} = \frac{\sqrt{2} V_{nom} L_m}{\sqrt{3} \omega_{nom} L_s} \quad (4.18)$$

Si bien este método es sencillo, presenta la desventaja de no garantizar el torque máximo ni mantener una respuesta adecuada ante variaciones de carga. La reducción del flujo mediante el método $\frac{1}{\omega_m}$ no es apropiada en la región de debilitamiento de campo, ya que impide alcanzar la máxima capacidad de torque del motor dentro de los límites de corriente y voltaje. Sin embargo, puede ser suficiente para aplicaciones en las que normalmente se opera a

velocidades menores que la nominal o sin carga significativa dentro del rango de flujo debilitado.

4.4. Controlador de velocidad

En el desarrollo del sistema de regulación de velocidad de un motor de inducción, se adopta como criterio de diseño que el lazo externo de velocidad opere con un ancho de banda inferior al del lazo interno de corriente (10 veces menor aprox.). Esta jerarquía se fundamenta en la naturaleza mecánica de la velocidad, cuya respuesta es inherentemente más lenta que la de las variables eléctricas. Bajo estas consideraciones, la estrategia de control seleccionada para la velocidad es un regulador de tipo proporcional–integral (PI).

El dimensionamiento de este controlador se apoya en la expresión del par electromagnético generado por el motor de inducción, la cual, en una forma simplificada, puede escribirse como:

$$T_e = \frac{3}{2} p \frac{L_m}{L_r} \psi'_{rd} i_{sq}$$

A partir de esta relación, se define la constante de torque del motor como:

$$K_T = \frac{3}{2} p \frac{L_m}{L_r} \psi'_{rd} \rightarrow T_e = K_T i_{sq}$$

Aplicando la transformación al dominio de Laplace y considerando la ecuación mecánica del sistema (3.12), se obtiene:

$$\omega_r (Js + B) = p(T_e - T_L)$$

Teniendo en cuenta la relación entre la velocidad eléctrica y la velocidad mecánica,

$$\omega_m = \frac{\omega_r}{p}$$

$$\omega_m (Js + B) = (T_e - T_L) \quad (4.19)$$

Para $T_L = 0$

$$G(s) = \frac{\omega_m(s)}{T_e(s)} = \frac{1}{Js + B}$$

Continuando con ello, se evidencia el esquema correspondiente al lazo de control de velocidad, ilustrado en la Figura 42.

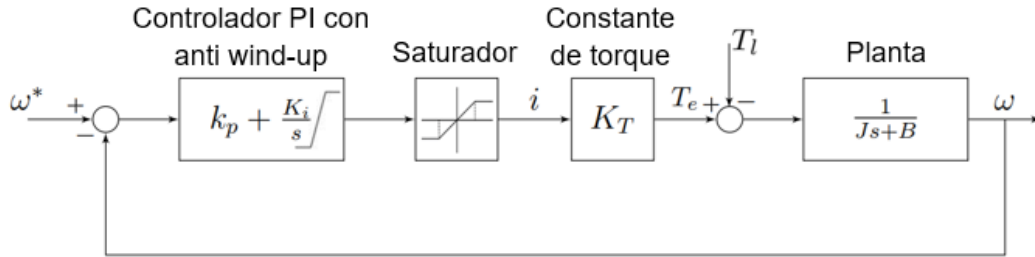


Figura 42. Lazo de control de velocidad

Para la regulación de la velocidad se emplea un controlador PI ajustado mediante un método clásico. A partir del modelo dinámico del motor, la función de transferencia que relaciona la velocidad real con la velocidad de referencia se obtiene como:

$$G_p(s) = \frac{\omega_m}{\omega_m^*} = \frac{\left(k_p + \frac{k_i}{s}\right) * K_T * \frac{1}{Js + B}}{1 + \left(k_p + \frac{k_i}{s}\right) * \frac{K_T}{Js + B}}$$

$$G_p(s) = \frac{\omega_m}{\omega_m^*} = \frac{K_T k_p s + K_T k_i}{Js^2 + (B + K_T k_p)s + K_T k_i}$$

La sintonización del controlador PI se realiza por medio del método de asignación de polos, comparando esta función con una forma estándar de segundo orden:

$$G_p(s) = \frac{\omega_m}{\omega_m^*} = \frac{K_1 \omega_N^2}{s^2 + 2\zeta \omega_N s + \omega_N^2} = \frac{K_T k_i / J}{s^2 + (B + K_T k_p)s / J + K_T k_i / J}$$

Al ser igualados los coeficientes de las ecuaciones características, se obtienen las expresiones para las ganancias del controlador PI:

$$k_p = \frac{2J\zeta\omega_N - B}{K_T} \quad k_i = \frac{\omega_N^2 J}{K_T}$$

Generalmente, la frecuencia natural ω_N se selecciona en un rango comprendido entre 10 y 25 Hz, mientras que el factor de amortiguamiento se fija en 0.707, con el fin de lograr un compromiso adecuado entre rapidez de respuesta y estabilidad (Llorente, 2020).

4.4.1. Mecanismo Anti-Windup en el Controlador PI

Dado que los actuadores tienen límites físicos (como voltaje o corriente), el término integral puede seguir acumulando error inútilmente, aunque el sistema ya esté saturado. Este fenómeno, llamado "wind-up", provoca que el controlador tarde en recuperarse ("descarga del integrador"), causando grandes sobreimpulsos y lentitud. Para evitar este comportamiento, se implementa un mecanismo anti-windup, el cual detiene la acumulación en el integrador cuando la salida del controlador alcanza los límites permitidos por la planta.

Para la implementación del esquema anti-windup en el controlador PI, se emplea el método de cálculo inverso (back-calculation). Esta técnica utiliza un bucle de realimentación que corrige dinámicamente el término integral cuando la salida calculada excede los límites del actuador. Al detectar la diferencia entre la salida saturada y la no saturada, se inyecta un término de corrección negativo al integrador, mitigando su acumulación excesiva. Esto facilita una recuperación rápida y suave de la saturación, aunque conlleva la necesidad de sintonizar un parámetro adicional que es la ganancia de seguimiento (tracking gain).

En la Figura 43, se observa que cuando la señal de control se satura, se previene el acumulamiento de error en el integrador por medio de una retroalimentación de la diferencia entre el torque deseado y el saturado.

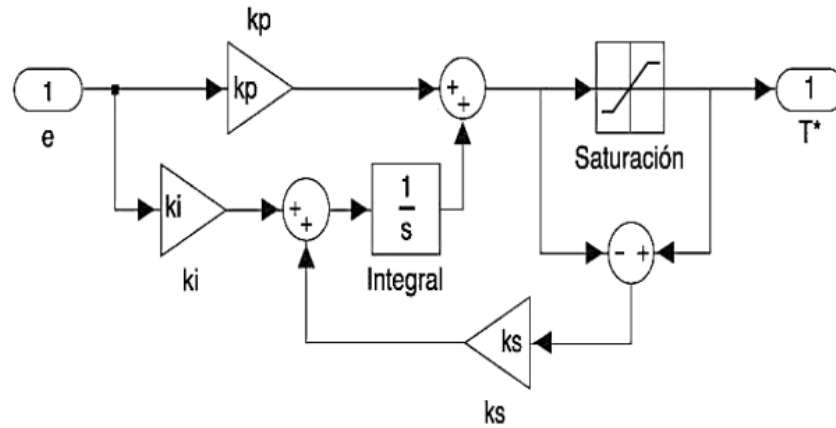


Figura 43. Control PI con anti-windup (Shen, 2023)

En la literatura, la ganancia del lazo de anti-windup se suele definir como la inversa de un tiempo:

$$K_s = \frac{1}{T_t}$$

donde T_t (Constante de Tiempo de Seguimiento) representa físicamente cuánto tarda el integrador en "desinflarse" tras una saturación. La práctica estándar es igualar este tiempo a la dinámica natural del controlador $T_t = K_p/K_i$ por lo que al calcular la ganancia como $K_s = K_i/K_p$, se está aplicando matemáticamente la regla ideal para que el sistema se recupere de la saturación con la misma velocidad y estabilidad con la que funciona normalmente.

Para la sintonización se utilizó la Control System Designer de Matlab. Durante el proceso se probaron diferentes combinaciones de los parámetros naturales ω_n y el coeficiente de amortiguamiento ζ , obteniéndose el mejor desempeño dinámico para los valores $\omega_n = 15$ Hz y $\zeta = 0.707$.

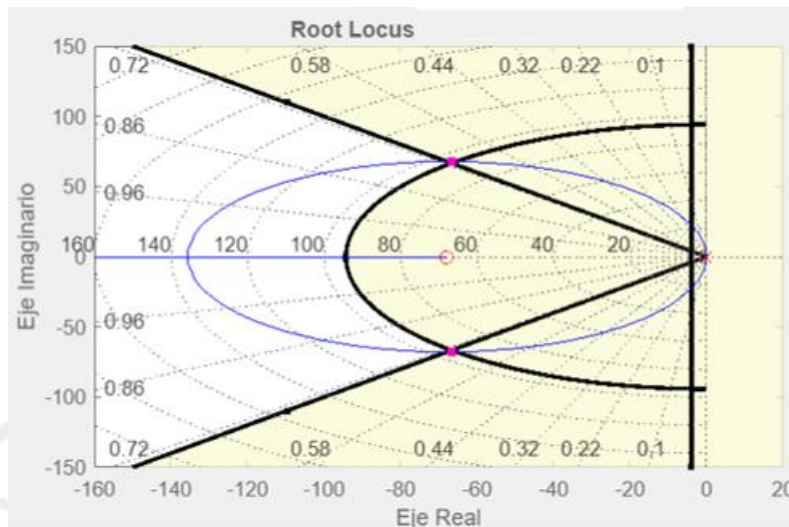


Figura 44. Lugar de raíz utilizado para el diseño del control de velocidad

La herramienta Control System Designer ofrece un entorno gráfico que permite examinar el comportamiento del lugar de las raíces en sistemas de control automático. Mediante esta utilidad es posible visualizar y modificar la posición de polos y ceros dentro del plano complejo, lo que resulta de gran ayuda durante el proceso de síntesis y ajuste de controladores de tipo PI.

Durante el diseño del regulador proporcional–integral, se incorporó un integrador junto con unos cero reales en el plano complejo, con el fin de definir una función de transferencia que satisfaga los requerimientos de desempeño del sistema. La acción integral asegura la eliminación del error en régimen permanente, mientras que la inclusión del cero real permite moldear la dinámica del sistema, optimizando su comportamiento transitorio y la estabilidad general. La Figura 44 muestra la ubicación de los polos en el plano complejo

La estabilidad y el rendimiento dinámico del sistema se verificaron mediante el análisis de diagramas de Bode, mostrados en las Figuras 45 y 46.

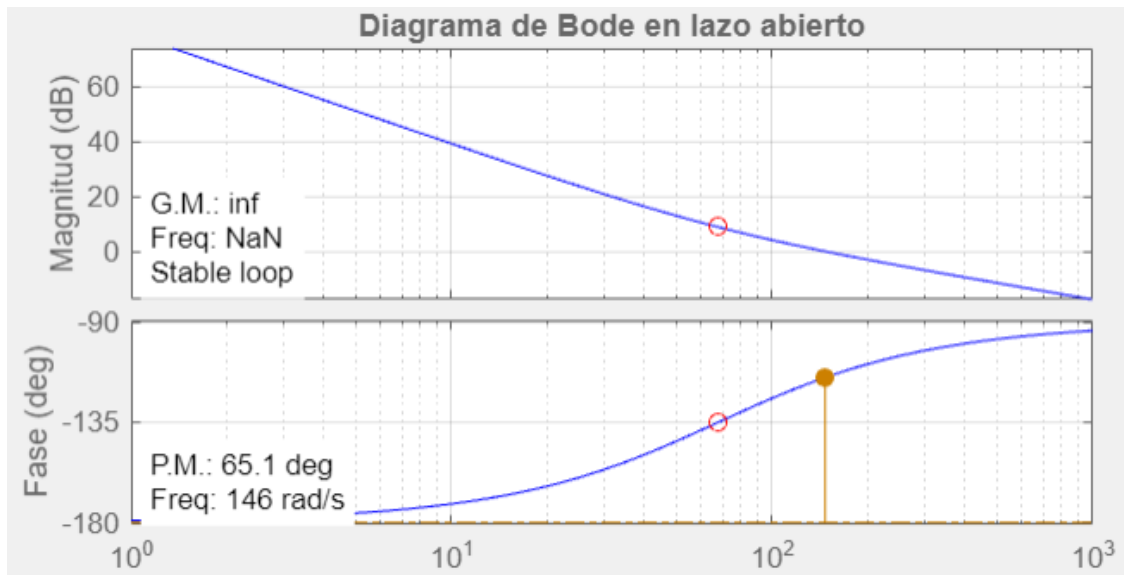


Figura 45. Diagrama de Bode en lazo abierto

Del diagrama en lazo abierto (Figura 45), se observa un margen de ganancia (G.M.) = ∞ . Esto indica que la curva de fase no cruza el nivel de -180° (se mantiene por encima), por lo tanto, el sistema es estable para cualquier valor de ganancia positiva ($k > 0$).

Asimismo, el margen de fase (P.M.) = 65.1° a una frecuencia de 146 rad/s, valor suficientemente amplio (mayor a 45° recomendado), evidencia que el sistema presenta una buena estabilidad y amortiguamiento.

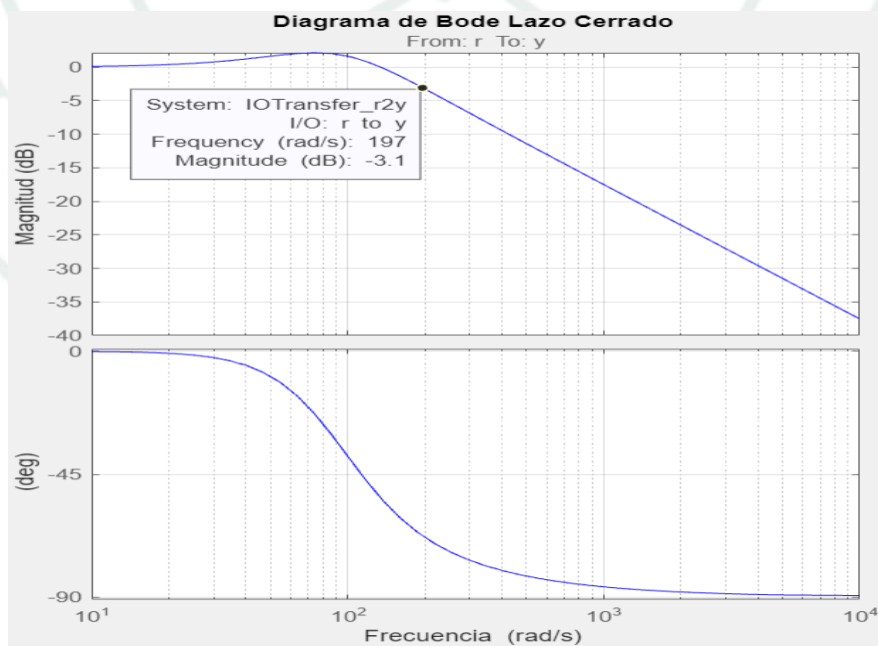


Figura 46. Diagrama de Bode en lazo cerrado

En el diagrama de bode en lazo de cerrado (Figura 46), se determina el ancho de banda del sistema en la frecuencia ω_b (-3 dB) que es 197 rad/s, lo que indica que el sistema mantiene un buen desempeño dinámico y seguimiento de referencia hasta esa frecuencia, comportándose como un filtro pasa-bajo que atenúa las perturbaciones de alta frecuencia sin comprometer la estabilidad.

Los parámetros obtenidos son $K_p = 0.5$, $K_i = 12$

4.5. Control de corrientes

Se expone el esquema del sistema de control implementado, el cual está conformado por tres reguladores de tipo PI: un controlador destinado a la regulación de la velocidad y dos controladores adicionales encargados del control de las corrientes. Esta arquitectura se muestra de manera esquemática en la Figura 47.

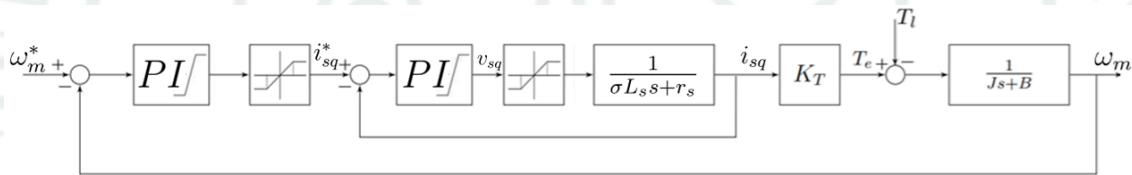


Figura 47. Diagrama de bloque en lazo de control cerrado del motor de inducción

En la Figura 47 se representa la interacción entre los lazos de control de velocidad y de corriente. Ambos reguladores emplean controladores PI que incorporan mecanismos de anti-windup, con el propósito de incrementar la robustez del sistema y evitar la acumulación excesiva del término integral cuando las señales de control alcanzan los límites físicos del actuador.

El ajuste de los controladores PI de corriente se realiza siguiendo un procedimiento análogo al utilizado en el diseño del controlador de velocidad, considerando que el lazo de corriente constituye el lazo interno del sistema. Esto implica que dicho controlador debe presentar una dinámica significativamente más rápida que la del lazo externo de velocidad. Por esta razón, la frecuencia natural asociada al control de corriente se selecciona típicamente

dentro del rango de 100 a 250 Hz, asegurando que sea al menos un orden de magnitud superior a la frecuencia natural del lazo de velocidad.

Finalmente, a partir de las ecuaciones (4.4-5), se procede al diseño de los controladores correspondientes a las corrientes i_{sd} e i_{sq} en el dominio de Laplace, obteniéndose las expresiones que definen la dinámica de estos lazos de control.

$$i_{sd} \cdot (r_s \cdot + \sigma L_s s) = v_{sd} + \sigma L_s \omega_s \cdot i_{sq} - \frac{L_m}{L'_r} \frac{d\psi'_{rd}}{dt} \quad (4.20)$$

$$i_{sq} \cdot (r_s \cdot + \sigma L_s s) = v_{sq} - \sigma L_s \omega_s \cdot i_{sd} - \omega_s \frac{L_m}{L'_r} \cdot \psi'_{rd} \quad (4.21)$$

Para facilitar el diseño del PI, se parte del modelo desacoplado de estas ecuaciones, ignorando las perturbaciones cruzadas (acoplamientos), las cuales pueden tratarse como perturbaciones constantes y compensarse por el control. De esta forma, se considera como planta:

$$G_p(s) = \frac{i_s}{v_s} = \frac{1}{\sigma L_s s + r_s}$$

Aplicando el controlador PI en lazo cerrado se obtiene

$$G(s) = \frac{\left(k_p + \frac{k_i}{s}\right) * \frac{1}{\sigma L_s s + r_s}}{1 + \left(k_p + \frac{k_i}{s}\right) * \frac{1}{\sigma L_s s + r_s}}$$

$$G(s) = \frac{k_p s + k_i}{\sigma L_s s^2 + (r_s + k_p)s + k_i}$$

Aplicando condición de cancelación cero-polo

Del controlador: $\frac{k_p s + k_i}{s}$, cancelación de cero

$$k_p s + k_i = 0 \quad s_z = -\frac{k_i}{k_p}$$

De planta

$$\frac{1}{\sigma L_s(s + \frac{r_s}{\sigma L_s})}$$

$$s_p = -\frac{r_s}{\sigma L_s}$$

Para el Óptimo de Magnitud, el cero del PI debe cancelar el polo de planta:

$$s_z = s_p$$

Es decir:

$$\frac{k_i}{k_p} = \frac{r_s}{\sigma L_s} \quad k_i = k_p \frac{r_s}{\sigma L_s}$$

Función de transferencia en lazo cerrado

$$G(s) = \frac{i_{sd}}{i_{sd}^*} = \frac{k_p s + k_i}{\sigma L_s s^2 + (r_s + k_p)s + k_i}$$

Sustituyendo $\frac{k_i}{k_p} = a$ en numerador

$$k_p s + k_i = k_p(s + a)$$

Y denominador factorizado

$$\sigma L_s s^2 + (r_s + k_p)s + k_p a = (s + a)(\sigma L_s + k_p)$$

Por tanto, se cancelan $(s + a)$ arriba y abajo y queda

$$T(s) = \frac{k_p}{\sigma L_s + k_p} = \frac{k_p / \sigma L_s}{s + k_p / \sigma L_s}$$

Entonces el polo es:

$$s = -\frac{k_p}{\sigma L_s}$$

Para el diseño de control, se fija una frecuencia característica ω_c

$$s = -\omega_c \rightarrow \omega = \frac{k_p}{\sigma L_s}$$

Finalmente

$$k_p = \sigma L_s \omega \quad k_i = \omega r_s$$

La sintonización del controlador se realizó siguiendo el mismo procedimiento utilizado para el control de velocidad. Para el diseño se adoptaron los valores de $\omega_N = 135$ Hz y un factor de amortiguamiento $\zeta = 0.707$, a partir de los cuales se calcularon k_p y k_i .

La Figura 48 muestra la ubicación de los polos en el plano complejo, lo que permite ajustar gráficamente el control PI y observar de forma directa cómo se modifican los polos del sistema en lazo cerrado durante el proceso de sintonización.

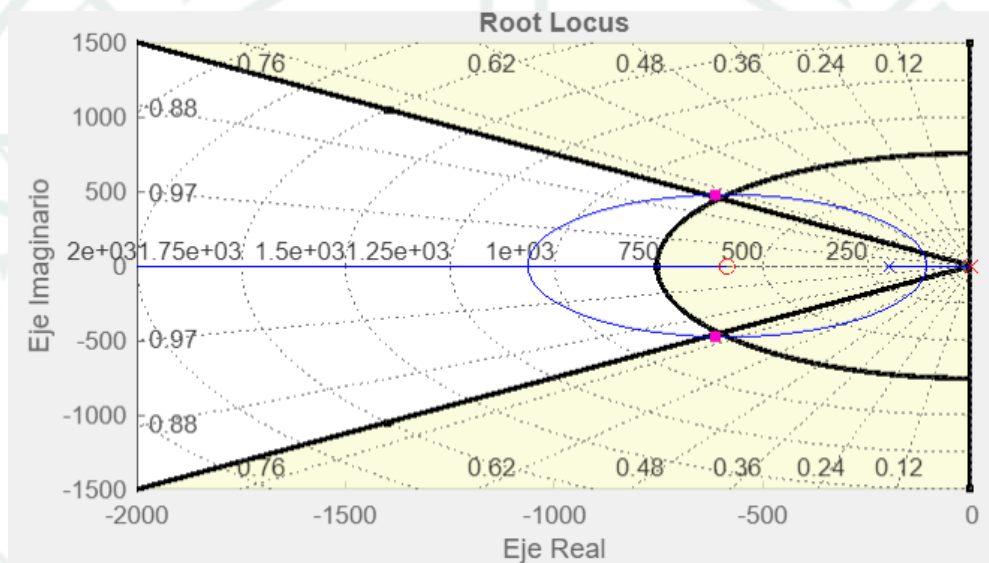


Figura 48. Lugar de raíz utilizando para el diseño del control de corriente

Del diagrama de Bode en lazo abierto (Figura 49) se aprecia un margen de ganancia (G.M.) infinito. Esto se debe a que la curva de fase nunca cruza el nivel crítico de -180° , lo que indica que el sistema es incondicionalmente estable ante variaciones de ganancia positiva.

Asimismo, el margen de fase (P.M.) de 72.6° a una frecuencia de corte de 1143 rad/s. Este valor, superior al rango típico de 45° - 60° , evidencia un sistema con alta robustez y un comportamiento dinámico bien amortiguado, lo que minimiza o elimina el sobreimpulso ante cambios en la referencia.

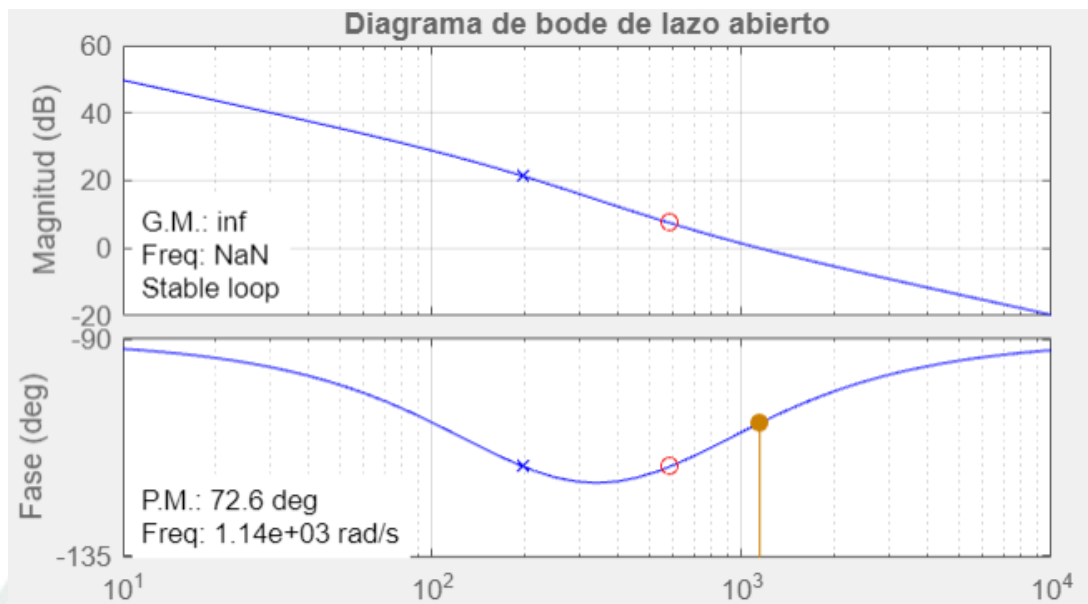


Figura 49. Diagrama de bode de lazo abierto de lazo de control de corriente

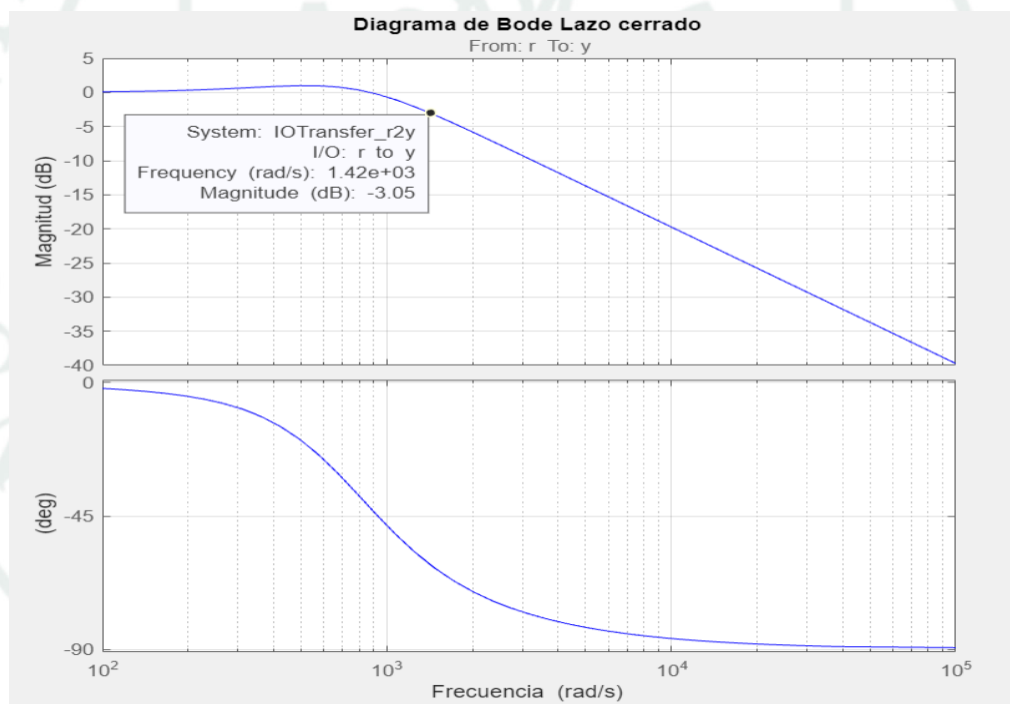


Figura 50. Diagrama de bode de lazo cerrado de lazo de control de corriente

El diagrama de lazo cerrado (Figura 50) muestra un ancho de banda de 1420 rad/s (frecuencia a -3 db), confirmando que el controlador posee la rapidez necesaria para el seguimiento preciso de referencias cambiantes

Con estos ajustes, se logró una sintonización óptima que garantiza un control preciso y robusto del sistema FOC donde $k_p = 73$ y $k_i = 13800$.

4.6. Desacoplamiento de Ejes d-q mediante Compensación por Feedforward

La dinámica del motor de inducción en el marco de referencia síncrono dq presenta términos de acoplamiento cruzado entre los ejes. Esto significa que un cambio en la corriente de flujo (i_{sd}) perturba al eje de torque (i_{sq}) y viceversa, dificultando el control independiente como se ve en las ecuaciones (4.4-5). A partir de estas ecuaciones de voltaje y en estado estable se anula $\frac{di_s}{dt} = 0$ $\frac{d\psi'_{rd}}{dt} = 0$, las ecuaciones quedan como:

$$v_{sd} = r_s \cdot i_{sd} - \sigma L_s \omega_s \cdot i_{sq} \quad (4.22)$$

$$v_{sq} = r_s \cdot i_{sq} + \sigma L_s \omega_s \cdot i_{sd} + \omega_s \frac{L_m}{L'_r} \cdot \psi'_{rd} \quad (4.23)$$

Reemplazando los valores $\psi'_{rd} = L_m i_{sd}$ y $\sigma = 1 - \frac{L_m^2}{L_s L'_r}$ en ecuación de v_{sq}

$$v_{sq} = r_s \cdot i_{sq} + \left(1 - \frac{L_m^2}{L_s L'_r}\right) L_s \omega_s \cdot i_{sd} + \omega_s \frac{L_m^2}{L'_r} \cdot i_{sd} \quad (4.24)$$

$$v_{sq} = r_s \cdot i_{sq} + L_s \omega_s \cdot i_{sd} - \frac{L_m^2}{L'_r} \omega_s \cdot i_{sd} + \omega_s \frac{L_m^2}{L'_r} \cdot i_{sd} \quad (4.25)$$

$$v_{sq} = r_s \cdot i_{sq} + \omega_s \frac{L_s}{L_m} \cdot \psi'_{rd} \quad (4.26)$$

Se observa que el voltaje del estator se compone de tres partes: la caída óhmica causada por la resistencia de los devanados $r_s \cdot i_s$, los voltajes de acoplamiento cruzado debidos a la inductancia de estator ($L_s \omega_s \cdot i_s$) y el voltaje de fuerza contraelectromotriz (back-EMF) en el eje q $\omega_s \frac{L_m}{L'_r} \cdot \psi'_{rd}$. Para lograr un control de alto desempeño, se implementa una estrategia de desacoplamiento. Esta consiste en calcular los términos de acoplamiento y de Back-EMF utilizando los parámetros del motor y las variables estimadas, para luego sumarlos (o restarlos) a la salida de los controladores de corriente PI como términos de prealimentación (feedforward).

De esta manera, el controlador PI solo se encarga de corregir el error de la componente resistiva y transitoria, mientras que los términos rotacionales son compensados activamente, tal como se ilustra en el diagrama de bloques (Figura 51)

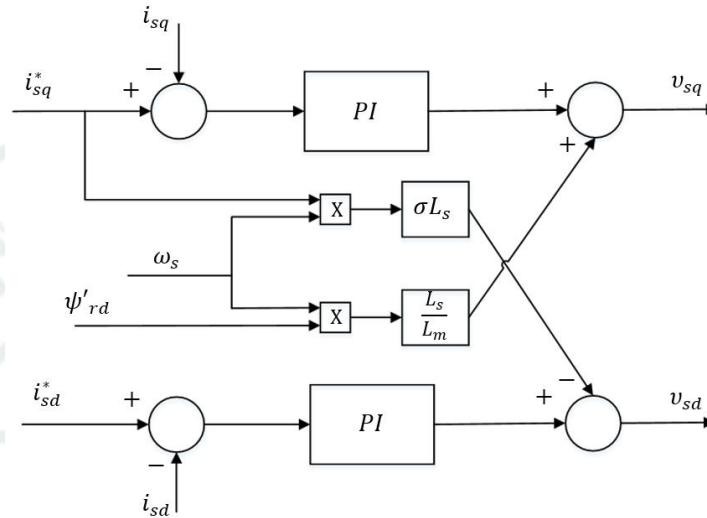


Figura 51. Acoplamiento cruzado – Elaboración propia

4.7. Limitadores de corrientes y voltajes

Para proteger al inversor frente a sobrecorrientes, es necesario implementar una función de limitación de corriente. Además del limitador máximo convencional en el circuito PWM digital, se emplea aquí un método adicional que restringe la amplitud de la corriente de referencia del estator.

En este esquema, las referencias de corriente magnetizante (i_{sd}^*) y de torque (i_{sq}^*) provienen del controlador de flujo y del controlador de velocidad, respectivamente. Generalmente se utilizan limitadores con valores constantes, pero estos presentan desventajas: por ejemplo, pueden restringir innecesariamente la corriente magnetizante incluso cuando el motor está ligeramente cargado, afectando el desempeño del control de flujo.

El valor crítico para supervisar no es i_{sd}^* ni i_{sq}^* por separado, sino la magnitud total de la corriente del estator:

$$I_s = \sqrt{i_{sd}^2 + i_{sq}^2} \quad (4.27)$$

Por ello, se adopta un limitador de corriente variable, cuyo diagrama en bloques se muestra en la Figura 52. Este limitador ajusta i_{sd} en función de i_{sq} de manera que la amplitud de la corriente de referencia no supere un valor máximo predefinido (I_{smax}), garantizando al mismo tiempo un valor mínimo adecuado para mantener el control de flujo.

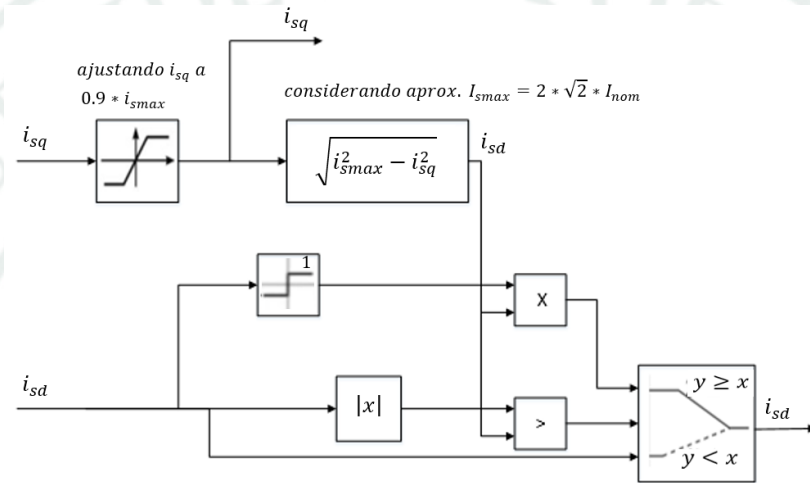


Figura 52. Limitador de corriente

En un inversor real, puede ocurrir saturación del voltaje del estator cuando el flujo del rotor no sigue con precisión su referencia. Esto suele deberse a errores entre los parámetros usados en el control y los parámetros reales del motor de inducción.

Cuando aparece esta saturación, los lazos de control de corriente dejan de funcionar correctamente. Para evitarlo, se emplea un limitador de voltaje variable, que ajusta la referencia de voltaje en el eje d (v_{sd}^*) en función de la referencia en el eje q (v_{sq}^*).

$$v_s = \sqrt{v_{sd}^2 + v_{sq}^2} \quad (4.28)$$

La idea es sencilla: como el flujo del rotor depende de v_{sd}^* , al limitar este valor se reduce el flujo y, con ello, se evita que el voltaje del estator alcance niveles de saturación.

La Figura 53 muestra el diagrama en bloques de este limitador, que asegura un funcionamiento estable del control incluso en presencia de errores de parámetros o condiciones de carga exigentes.

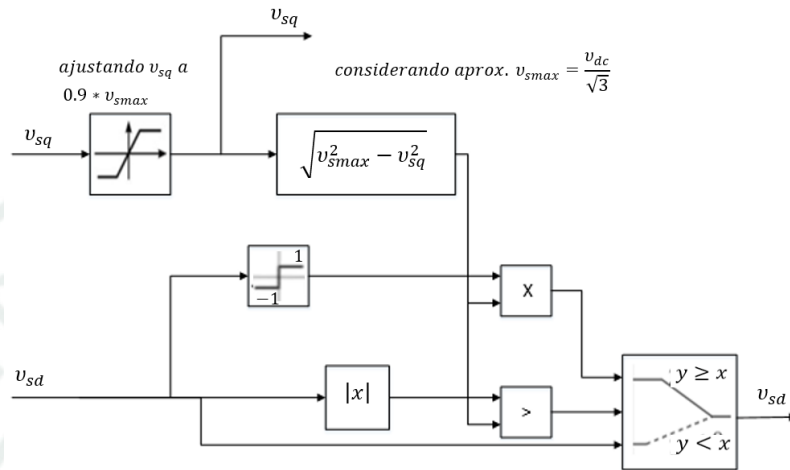


Figura 53. Limitador de voltaje

4.8. Modulación por ancho de pulsos de vector espacial (SVPWM)

La etapa de modulación es la encargada de definir la forma en que las señales de control se convierten en acciones de conmutación sobre los dispositivos semiconductores del convertidor de potencia, permitiendo así generar la señal de salida requerida. En la actualidad, existen dos métodos de modulación ampliamente utilizados: la SPWM y la SVPWM. Para el desarrollo de este trabajo se ha optado por la segunda, ya que sus prestaciones la convierten en una alternativa más eficiente para aplicaciones de control en motores de inducción. La técnica SVPWM se basa en un algoritmo que transforma las referencias de tensión, definidas en términos de voltajes de fase respecto al neutro y suministradas por el controlador, en tiempos de activación o ciclos de trabajo que son aplicados directamente al módulo PWM del sistema. Aunque este método puede emplearse en cualquier tipo de carga trifásica, su formulación original está orientada al control de máquinas eléctricas. Una de las principales ventajas de la modulación vectorial es su capacidad para aprovechar de manera óptima la tensión disponible

en el bus de corriente continua. Asimismo, al seleccionar los vectores de conmutación más próximos al vector de referencia, se logra una reducción significativa del contenido armónico en la señal de salida, mejorando así la calidad del voltaje aplicado al motor y el desempeño general del sistema (NXP Semiconductor, F, 2021).

El principio básico de SVPWM se basa en las ocho combinaciones de interruptores del inversor trifásico como se ven en Tabla 1. Las combinaciones de interruptores se pueden representar como códigos binarios que corresponden a los interruptores superiores S_1 , S_3 y S_5 del inversor. Cada circuito de conmutación genera tres tensiones polares independientes: V_a , V_b y V_c .

La técnica SVPWM consta de los siguientes pasos (NXP Semiconductor, F, 2021):

1. Identificación del sector.
2. Descomposición del vector espacial de voltaje en las direcciones de los vectores base del sector
3. Cálculo del ciclo de trabajo PWM.

Las tensiones de línea de los estados de conmutación se pueden hallar fácilmente siguiendo el mismo procedimiento. Por ejemplo, para el estado $V_1(1\ 0\ 0)$.

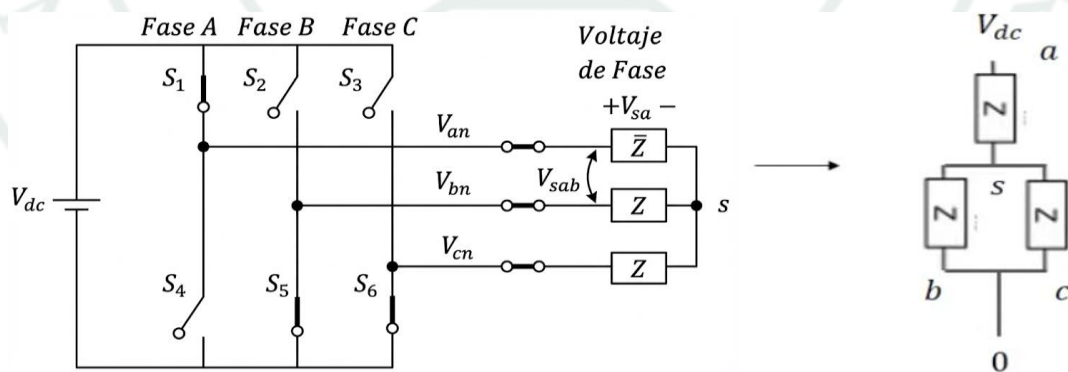


Figura 54. Tensiones de línea de los estados de conmutación

4.8.1. Voltaje de fase

Aplicando la regla del divisor para el voltaje.

$$V_{sa} = \frac{z}{z + \frac{z \cdot z}{z + z}} V_{DC} = \frac{z}{z + \frac{1}{2}z} V_{DC} = \frac{2}{3} V_{DC}$$

$$V_{sb} = \frac{\frac{1}{2}z}{z + \frac{z \cdot z}{z + z}} V_{DC} = -\frac{1}{3} V_{DC}$$

$$V_{sc} = \frac{\frac{1}{2}z}{z + \frac{z \cdot z}{z + z}} V_{DC} = -\frac{1}{3} V_{DC}$$

Se obtienen ocho vectores para cada posible estado de conmutación del inversor, de los cuales dos son vectores nulos (V_0 y V_7) y seis son vectores activos (V_1, V_2, V_3, V_4, V_5 y V_6). Estos vectores se pueden ver en la Figura 55, y definen seis zonas separadas por 60 grados, formando un hexágono.

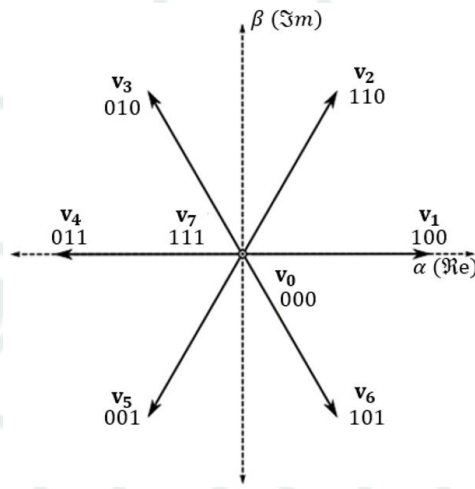


Figura 55. Vectores básicos de un inversor de dos niveles.

El fundamento de la modulación por vectores espaciales (SVPWM) radica en la selección y aplicación temporal adecuada de los distintos estados de conmutación del inversor, de forma que el efecto combinado de dichos estados reproduzca un vector de tensión equivalente al vector de referencia generado por la estrategia de control. Con el fin de explicar este procedimiento, a continuación, se desarrolla el análisis correspondiente.

Usando la transformación de Clarke:

$$\begin{bmatrix} v_{s\alpha} \\ v_{s\beta} \end{bmatrix} = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} v_{sa} \\ v_{sb} \\ v_{sc} \end{bmatrix}$$

$$V_{ref} = \sqrt{(V_{s\alpha})^2 + (V_{s\beta})^2} \quad \alpha = \tan^{-1} \left(\frac{V_{s\beta}}{V_{s\alpha}} \right)$$

Aplicando al estado $V_2(1\ 1\ 0)$

$$V_{s\alpha} = 0.5V_{DC} \quad V_{s\beta} = 0.8661V_{DC}$$

$$V_{ref} = V_{DC} \quad \alpha = 60^\circ$$

Realizando el mismo calculo en los diferentes vectores de voltaje, se obtiene

Tabla 8. Tensiones de fase en coordenadas $\alpha\beta$

Vector Espacial	Vsa	Vsb	Vsc	Vr	α (°)
$V_0(000)$	0	0	0	0	0
$V_1(100)$	$\frac{2}{3}V_{dc}$	$-\frac{1}{3}V_{dc}$	$-\frac{1}{3}V_{dc}$	V_{dc}	0
$V_2(110)$	$\frac{1}{3}V_{dc}$	$\frac{1}{3}V_{dc}$	$-\frac{2}{3}V_{dc}$	V_{dc}	60
$V_3(010)$	$-\frac{1}{3}V_{dc}$	$\frac{2}{3}V_{dc}$	$-\frac{1}{3}V_{dc}$	V_{dc}	120
$V_4(011)$	$-\frac{2}{3}V_{dc}$	$\frac{1}{3}V_{dc}$	$\frac{1}{3}V_{dc}$	V_{dc}	180
$V_5(001)$	$-\frac{1}{3}V_{dc}$	$-\frac{1}{3}V_{dc}$	$\frac{2}{3}V_{dc}$	V_{dc}	240
$V_6(101)$	$\frac{1}{3}V_{dc}$	$-\frac{2}{3}V_{dc}$	$\frac{1}{3}V_{dc}$	V_{dc}	300
$V_7(111)$	0	0	0	0	360

Para sintetizar el vector de voltaje de referencia (V_{ref}), el algoritmo SVPWM utiliza los vectores espaciales base asociados a los estados de conmutación del inversor. El módulo recibe como entrada las componentes del vector deseado en el marco estacionario $\alpha\beta$ y determina su ubicación exacta dentro del plano complejo.

Como se ilustra en la Figura 56, los vectores activos (no nulos) tienen una magnitud constante y una separación angular de 60° entre sí, conformando los vértices de un hexágono regular. Esta geometría divide el plano en seis sectores triangulares (I a VI). Adicionalmente, los dos vectores nulos (V_0 y V_7) ubicados en el centro del hexágono, los cuales no generan voltaje efectivo en la carga, pero son fundamentales para completar el periodo de modulación.

El principio de operación consiste en aproximar el vector de referencia mediante la aplicación secuencial de los dos vectores activos adyacentes al sector donde se encuentra (V_{ref}), más la inserción de tiempos nulos.

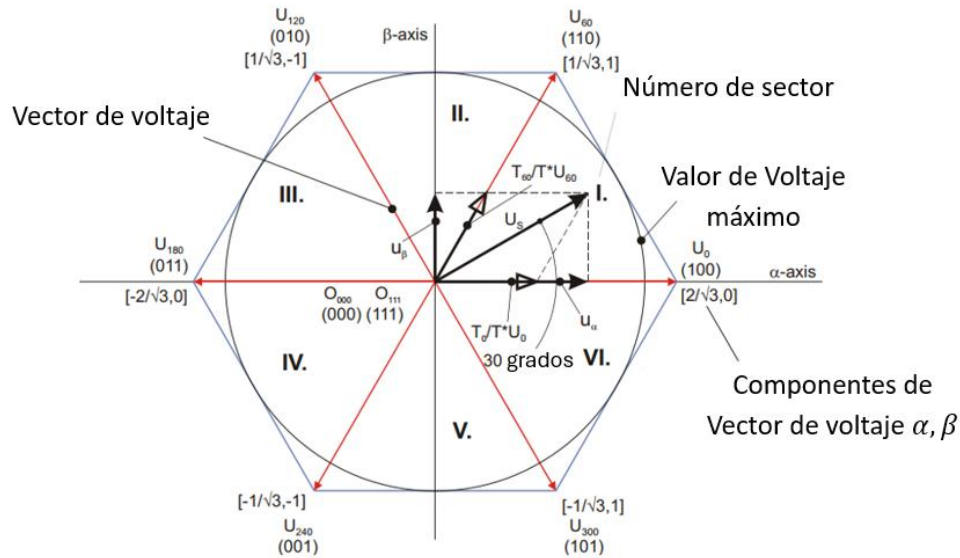


Figura 56. Proyección del vector de referencia en el sector 1 (NXP Semiconductor, F, 2021)

La Figura 56 detalla la situación para el Sector I. Aquí, el vector de referencia se descompone en proyecciones sobre los ejes adyacentes (en este caso, alineados con los vectores V_0 y V_{60}). A partir de estas proyecciones trigonométricas, se calculan los tiempos de permanencia necesarios para cada vector activo, asegurando que el promedio voltio-segundo durante el periodo de muestreo reproduzca fielmente el vector de referencia solicitado.

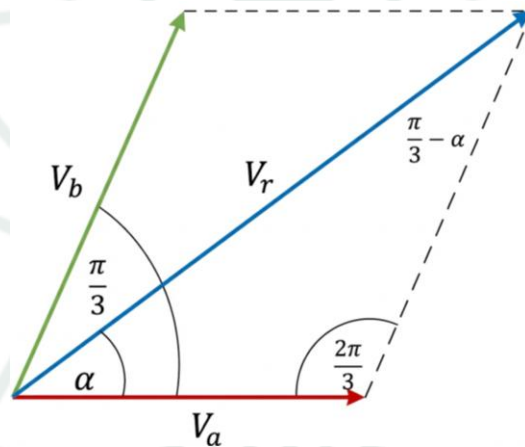


Figura 57. Detalle de la proyección del vector de referencia en el sector 1. – Elaboración propia

A partir de la Figura 57, se establece una relación lineal entre los tiempos de activación de los vectores activos de tensión T_1 y T_2 para el sector 1

$$V_a = \frac{T_1}{T_{periodo}} V_{DC} \quad \rightarrow \quad T_1 = \frac{T_{periodo}}{V_{DC}} V_a$$

$$V_b = \frac{T_2}{T_{periodo}} V_{DC} \quad \rightarrow \quad T_2 = \frac{T_{periodo}}{V_{DC}} V_b$$

Aplicando la regla del seno al triángulo

$$\frac{V_{ref}}{\text{sen} \frac{2\pi}{3}} = \frac{V_a}{\text{sen} \left(\frac{\pi}{3} - \alpha \right)} = \frac{V_b}{\text{sen} \alpha}$$

Resolviendo para V_a y V_b

$$V_a = V_{ref} \frac{\text{sen} \left(\frac{\pi}{3} - \alpha \right)}{\text{sen} \left(\frac{2\pi}{3} \right)} \quad \rightarrow \quad V_a = \frac{2}{\sqrt{3}} V_{ref} \text{sen} \left(\frac{\pi}{3} - \alpha \right)$$

$$V_b = V_{ref} \frac{\text{sen}(\alpha)}{\text{sen} \left(\frac{2\pi}{3} \right)} \quad \rightarrow \quad V_b = \frac{2}{\sqrt{3}} V_{ref} \text{sen}(\alpha)$$

4.8.2. Implementación de la modulación SVPWM

Luego de resolver para cada sector. Se puede definir una ecuación generalizada para cada ángulo

$$T_1 = \frac{2T_{periodo}}{\sqrt{3}V_{DC}} V_{ref} \text{sen} \left(\frac{K * \pi}{3} - \alpha \right)$$

$$T_2 = \frac{2T_{periodo}}{\sqrt{3}V_{DC}} V_{ref} \text{sen} \left(\alpha - \frac{(K-1) * \pi}{3} \right)$$

$$T_0 = T_{periodo} - T_1 - T_2$$

T_0 representa el tiempo en que se aplicará un vector nulo

Una vez calculados los intervalos de tiempo, es necesario definir la distribución de los ciclos de trabajo. Se consideran principalmente dos estrategias: el alineamiento simétrico y la secuencia alternada (Figura 58). En este trabajo se adopta el alineamiento simétrico, ya que ofrece un espectro armónico más limpio y una salida menos distorsionada, para una misma

frecuencia de conmutación, aunque su implementación algorítmica es más exigente en el procesamiento digital.

Para mantener una frecuencia de conmutación fija y un buen desempeño armónico, cada rama del inversor cambia de estado solo una vez por período. Esto se logra aplicando primero un vector nulo, seguido de los dos vectores activos adyacentes, y finalmente otro vector nulo. La segunda mitad del período es la imagen especular de la primera. De este modo, el período de conmutación se divide en siete intervalos: 1/4 del tiempo nulo inicial, la mitad de los tiempos activos, 1/4 del tiempo nulo final, y su repetición en espejo (Sancha Miguel, 2023).

Este esquema constituye la SVPWM simétrica, donde la tensión de rama resultante presenta baja distorsión armónica.

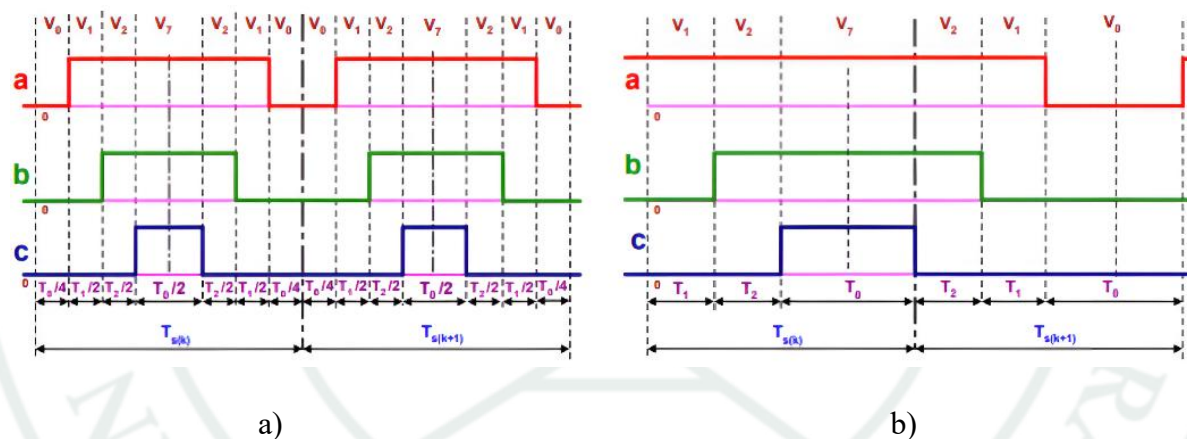


Figura 58. Cálculo ciclo de trabajo (a) Alineamiento simétrico b) Secuencia alternada)

(Sancha Miguel, 2023)

La Tabla 9 resume la distribución temporal de los estados activos y nulos en cada sector, observándose que la secuencia de conmutación se adapta únicamente mediante la permutación de los tiempos T_1 y T_2 , manteniendo la simetría respecto al centro del período $T_{periodo}$.

Tabla 9. Análisis en todos los sectores

Sección	T_{s1}	T_{s2}	T_{s3}	T_{s4}	T_{s5}	T_{s6}
Sección 1	$\frac{T_0}{2} + T_1 + T_2$	$\frac{T_0}{2} + T_2$	$\frac{T_0}{2}$	$\frac{T_0}{2}$	$\frac{T_0}{2} + T_1$	$\frac{T_0}{2} + T_1 + T_2$
Sección 2	$\frac{T_0}{2} + T_1$	$\frac{T_0}{2} + T_1 + T_2$	$\frac{T_0}{2}$	$\frac{T_0}{2} + T_2$	$\frac{T_0}{2}$	$\frac{T_0}{2} + T_1 + T_2$
Sección 3	$\frac{T_0}{2}$	$\frac{T_0}{2} + T_1 + T_2$	$\frac{T_0}{2} + T_2$	$\frac{T_0}{2} + T_1 + T_2$	$\frac{T_0}{2}$	$\frac{T_0}{2} + T_1$
Sección 4	$\frac{T_0}{2}$	$\frac{T_0}{2} + T_1$	$\frac{T_0}{2} + T_1 + T_2$	$\frac{T_0}{2} + T_1 + T_2$	$\frac{T_0}{2} + T_2$	$\frac{T_0}{2}$
Sección 5	$\frac{T_0}{2} + T_2$	$\frac{T_0}{2}$	$\frac{T_0}{2} + T_1 + T_2$	$\frac{T_0}{2} + T_2$	$\frac{T_0}{2} + T_1 + T_2$	$\frac{T_0}{2}$
Sección 6	$\frac{T_0}{2} + T_1 + T_2$	$\frac{T_0}{2}$	$\frac{T_0}{2} + T_1$	$\frac{T_0}{2}$	$\frac{T_0}{2} + T_1 + T_2$	$\frac{T_0}{2} + T_2$

En la Tabla 10, se muestra tabla simplificada de SVPWM simétrico como su forma reducida usando $T_{periodo}$.

Tabla 10. Tiempos SVPWM simétrico (Forma reducida)

Sección	T_{s1}	T_{s2}	T_{s3}
Sección 1	$T_{periodo} - \frac{T_0}{2}$	$\frac{T_0}{2} + T_2$	$\frac{T_0}{2}$
Sección 2	$\frac{T_0}{2} + T_1$	$T_{periodo} - \frac{T_0}{2}$	$\frac{T_0}{2}$
Sección 3	$\frac{T_0}{2}$	$T_{periodo} - \frac{T_0}{2}$	$\frac{T_0}{2} + T_2$
Sección 4	$\frac{T_0}{2}$	$\frac{T_0}{2} + T_1$	$T_{periodo} - \frac{T_0}{2}$
Sección 5	$\frac{T_0}{2} + T_2$	$\frac{T_0}{2}$	$T_{periodo} - \frac{T_0}{2}$
Sección 6	$T_{periodo} - \frac{T_0}{2}$	$\frac{T_0}{2}$	$\frac{T_0}{2} + T_1$

4.9. Diseño del Filtro de Kalman extendido en el modelo del motor

Para el diseño e implementación del Filtro de Kalman Extendido (EKF) en el modelo del motor de inducción, se adopta una representación dinámica en el sistema de referencia estacionario ligado al estator ($\alpha\beta$). A diferencia del marco síncrono (dq), en este sistema la velocidad del marco de referencia es nula ($\omega_s = 0$), lo que simplifica el modelo.

Sin embargo, al incorporar la velocidad angular del rotor (ω_r) como una variable de estado adicional a estimar, el modelo matemático resultante se vuelve no lineal debido al producto entre las variables de estado (velocidad y flujo). Con esta consideración, la representación extendida en el espacio de estados discreto se define de la siguiente manera:

$$\begin{aligned}
 x_k &= A_d x_{k-1} + B_d u_{k-1} \\
 y_k &= C x_k \\
 A_d &\approx I + AT \\
 B_d &\approx BT \\
 A_d &= \begin{bmatrix} 1 - \frac{T}{\sigma} \left(\frac{1}{T_s} + \frac{1-\sigma}{T_r} \right) & 0 & \frac{T(1-\sigma)}{\sigma L_m T_r} & \frac{T(1-\sigma)}{\sigma L_m} \omega_r & 0 \\ 0 & 1 - \frac{T}{\sigma} \left(\frac{1}{T_s} + \frac{1-\sigma}{T_r} \right) & -\frac{T(1-\sigma)}{\sigma L_m} \omega_r & \frac{T(1-\sigma)}{\sigma L_m T_r} & 0 \\ \frac{TL_m}{T_r} & 0 & 1 - \frac{T}{T_r} & -T\omega_r & 0 \\ 0 & \frac{TL_m}{T_r} & T\omega_r & 1 - \frac{T}{T_r} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \\
 B_d &= \begin{bmatrix} \frac{T}{\sigma L_s} & 0 \\ 0 & \frac{T}{\sigma L_s} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \\
 x_{k-1} &= [i_{s\alpha} \quad i_{s\beta} \quad \psi'_{r\alpha} \quad \psi'_{r\beta} \quad \omega_r]^T \quad u_{k-1} = [v_{s\alpha} \quad v_{s\beta}]^T \quad y_k = [i_{s\alpha} \quad i_{s\beta}]^T
 \end{aligned}$$

El tiempo de muestreo utilizado en la discretización desempeña un papel fundamental para obtener resultados precisos. Esto se analizará con más detalle en las secciones de simulación y diseño experimental. Generalmente, un tiempo de muestreo menor corresponde a mayores bucles cíclicos del EKF y a una mayor precisión. Es importante destacar que el tiempo de muestreo no debe ser menor que el tiempo empleado en la implementación de cada bucle en el algoritmo EKF (Tiwari, 2019).

Durante la evaluación del tiempo de muestreo es necesario encontrar un equilibrio entre precisión y estabilidad.

El ruido del sistema se expresa como w_k , que se asume como ruido gaussiano blanco de media cero, independiente del vector de estado y con una matriz de covarianza Q . El modelo del sistema resultante se representa como:

$$x_k = A_d x_{k-1} + B_d u_{k-1} + w_{k-1}$$

De forma similar, el ruido de medición se expresa como v_k , que se asume como ruido gaussiano blanco de media cero, independiente del vector de estado y del ruido del sistema, y con una covarianza de R . El modelo de salida resultante se representa como:

$$y_k = C_d x_k + v_k$$

4.9.1. Determinación de las matrices de covarianza de ruido y estado

Las matrices de covarianza del ruido del proceso y del ruido de medición, denotadas por Q y R , tienen dimensiones directamente relacionadas con el número de estados del modelo y con el número de variables medidas. En este trabajo, la matriz Q es de dimensión 5×5 , a las dos correspondiente a los cinco estados considerados, mientras que la matriz R es de 2×2 , asociada señales medidas.

Asumiendo que los ruidos del proceso y de medición no están correlacionados entre sí, ambas matrices se consideran diagonales, una práctica común en la implementación de filtros de Kalman que simplifica tanto la inicialización como el cálculo iterativo del algoritmo.

La adecuada selección de Q y R es crucial para el desempeño del EKF, ya que determina el compromiso entre la confianza en el modelo del motor y la confianza en las mediciones, influyendo directamente en la precisión y estabilidad del estimador (Tiwari, 2019).

$$Q = \begin{bmatrix} q_{11} & 0 & 0 & 0 & 0 \\ 0 & q_{22} & 0 & 0 & 0 \\ 0 & 0 & q_{33} & 0 & 0 \\ 0 & 0 & 0 & q_{44} & 0 \\ 0 & 0 & 0 & 0 & q_{55} \end{bmatrix} \quad R = \begin{bmatrix} r_{11} & 0 \\ 0 & r_{22} \end{bmatrix}$$

En este caso, generalmente, $q_{11} = q_{22}$, $q_{33} = q_{44}$ y $r_{11} = r_{22}$

4.9.2. Implementación del algoritmo EKF

La Figura 59 muestra el modelo discretizado de la máquina de inducción junto con la implementación del algoritmo EKF, organizados en una estructura modular por bloques. La parte superior corresponde al modelo de la máquina en tiempo continuo, mientras que la parte inferior representa el EKF ejecutado en tiempo discreto. Ambos módulos interactúan mediante las tensiones de entrada y las corrientes medidas, y su separación conceptual se indica mediante la línea de trazos.

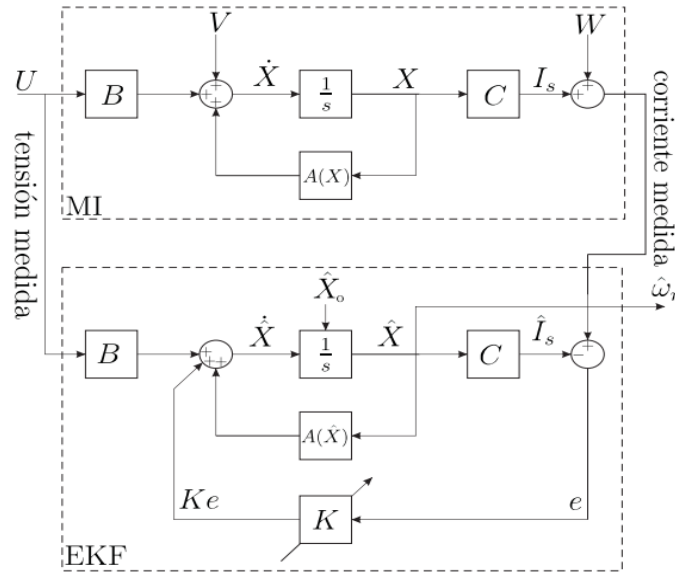


Figura 59. Estructura de EKF (Tiwari, 2019).

4.9.3. Inicialización de vectores de estado y matrices de covarianza

Al inicio del algoritmo, el vector de estado se inicializa como $X_0 = X(t_0)$, junto con las matrices de covarianza del ruido de proceso $Q_0 \in \mathbb{R}^{5 \times 5}$ y del ruido de medición $R_0 \in \mathbb{R}^{2 \times 2}$.

Los valores iniciales de Q y R se seleccionan de manera heurística dentro de rangos razonables y se ajustan posteriormente según el desempeño del filtro y el modo de operación. La matriz de covarianza del estado $P_0 \in \mathbb{R}^{5 \times 5}$ se define como una matriz diagonal con elementos iguales, representando una incertidumbre inicial uniforme en todos los estados.

4.9.4. Predicción de la variable de estado

La variable de estado predicha para el instante de muestreo (k) se obtiene a partir de la entrada u_{k-1} y el vector de estado $\hat{x}_{k-1}$.

$$\hat{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_{k-1})$$

Donde,

$$\hat{x}_{k|k-1} = [\hat{i}_{s\alpha,k|k-1} \quad \hat{i}_{s\beta,k|k-1} \quad \hat{\psi}'_{r\alpha,k|k-1} \quad \hat{\psi}'_{r\beta,k|k-1} \quad \hat{\omega}_{r,k|k-1}]^T$$

$$u_{k-1} = [v_{s\alpha|k-1} \quad v_{s\beta|k-1}]^T$$

4.9.5. Estimación de la Matriz de Covarianza de Predicción

La matriz de covarianza de predicción se estima como

$$P_{k|k-1} = F_{k-1} P_{k-1|k-1} F_{k-1}^T + Q_k$$

Aquí, la función F_{k-1} es el jacobiano (función gradiente) de la función del modelo de sistema descrita anteriormente, que linealiza el modelo. La función F se define como

$$F_{k-1} = \frac{\partial}{\partial x} [A_d X + B_d u] |_{\hat{x}_{k-1|k-1}, u_{k-1}}$$

F_{k-1}

$$= \begin{bmatrix} 1 - \frac{T}{\sigma} \left(\frac{1}{T_s} + \frac{1-\sigma}{T_r} \right) & 0 & \frac{T(1-\sigma)}{\sigma L_m T_r} & \frac{T(1-\sigma)}{\sigma L_m} \omega_r & \frac{T(1-\sigma)}{\sigma L_m} \psi'_{r\beta} \\ 0 & 1 - \frac{T}{\sigma} \left(\frac{1}{T_s} + \frac{1-\sigma}{T_r} \right) & -\frac{T(1-\sigma)}{\sigma L_m} \omega_r & \frac{T(1-\sigma)}{\sigma L_m T_r} & -\frac{T(1-\sigma)}{\sigma L_m} \psi'_{r\alpha} \\ \frac{TL_m}{T_r} & 0 & 1 - \frac{T}{T_r} & -T\omega_r & -T\psi'_{r\beta} \\ 0 & \frac{TL_m}{T_r} & T\omega_r & 1 - \frac{T}{T_r} & T\psi'_{r\alpha} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Donde,

$$\omega_r = \hat{\omega}_{r|k-1} \quad \psi'_{r\alpha} = \hat{\psi}'_{r\alpha|k-1} \quad \psi'_{r\beta} = \hat{\psi}'_{r\beta|k-1}$$

4.9.6. Cálculo de la Ganancia de Kalman

La matriz de ganancia de Kalman es una matriz de cinco por dos para este modelo de motor de inducción que estamos modelando.

Y se obtiene mediante la siguiente fórmula, que también incluye los parámetros de ruido de medición.

$$K_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + R_k)^{-1}$$

Aquí, la función H_k es el jacobiano (función de gradiente) de la función del modelo de salida descrita anteriormente, que linealiza el modelo. La función H_k se define como:

$$H_k = \frac{\partial}{\partial x} [C_d x] |_{x_k = \hat{x}_{k|k-1}}$$
$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

4.9.7. Estimación del vector de estado

El vector de estado final estimado y los modelos de salida después del filtrado en el tiempo (k) se expresan como:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k (y_k - h(\hat{x}_{k|k-1}))$$

4.9.8. Corrección de la matriz de covarianza de predicción para el siguiente bucle

La matriz de covarianza de errores se obtiene de

$$P_{k|k} = (I - K_k H_k) P_{k|k-1}$$

Con el último paso, el bucle finaliza durante un ciclo; para el siguiente instante de muestreo, $k + 1$, los valores del bucle se actualizan estableciendo $k = k + 1$ y repitiendo el proceso.

4.10. Resultados de simulación

4.10.1. Resultados de simulación de control vectorial

Para verificar el correcto desempeño del control vectorial aplicado al motor de inducción, se realizaron diversas pruebas de simulación en las que se introdujeron señales de referencia tanto de velocidad como de torque, según se muestra en la Figura 60 y Figura 61 respectivamente. Estas pruebas permiten evaluar la capacidad del sistema de control para seguir las consignas establecidas y mantener un comportamiento dinámico estable.

La implementación completa del control vectorial se detalla en el Anexo 4, donde se incluye la estructura del controlador, los parámetros utilizados y los bloques funcionales empleados. Para la simulación y ejecución del sistema de control, se consideró un tiempo de muestreo de $T_s = 1 * 10^{-5} \text{ seg.}$ y una frecuencia de conmutación de 5 kHz, valores adecuados para garantizar una respuesta rápida y una correcta discretización de las señales del sistema.

Se presenta las referencias de velocidad y torque

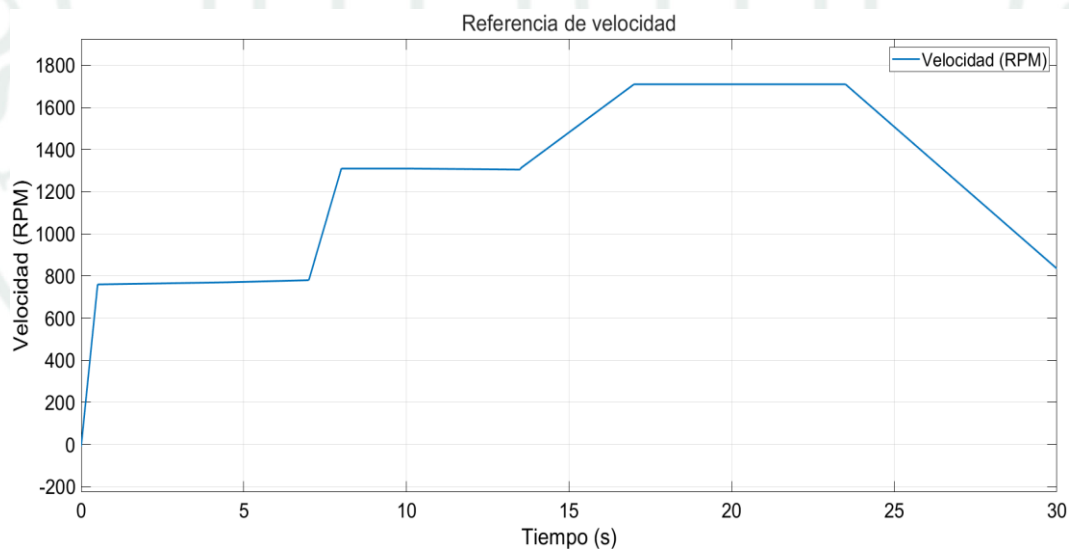


Figura 60. Referencia de velocidad

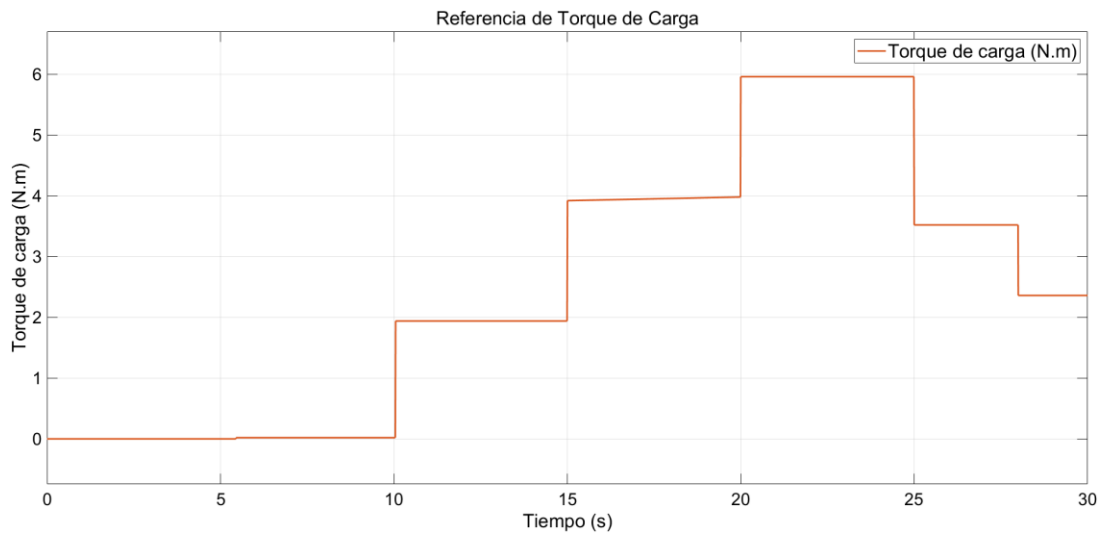


Figura 61. Referencia de torque

En la Figura 62 se observa el seguimiento de la velocidad del motor respecto a la referencia de velocidad. El controlador mantiene una buena precisión en el seguimiento, alcanzando los valores deseados con transitorios suaves y sin sobreoscilaciones significativas. Asimismo, se pueden apreciar pequeñas variaciones en la velocidad coincidentes con los instantes de cambio de torque de carga, lo cual es esperable debido al esfuerzo transitorio del controlador para compensar la perturbación mecánica y restablecer el equilibrio dinámico del sistema.

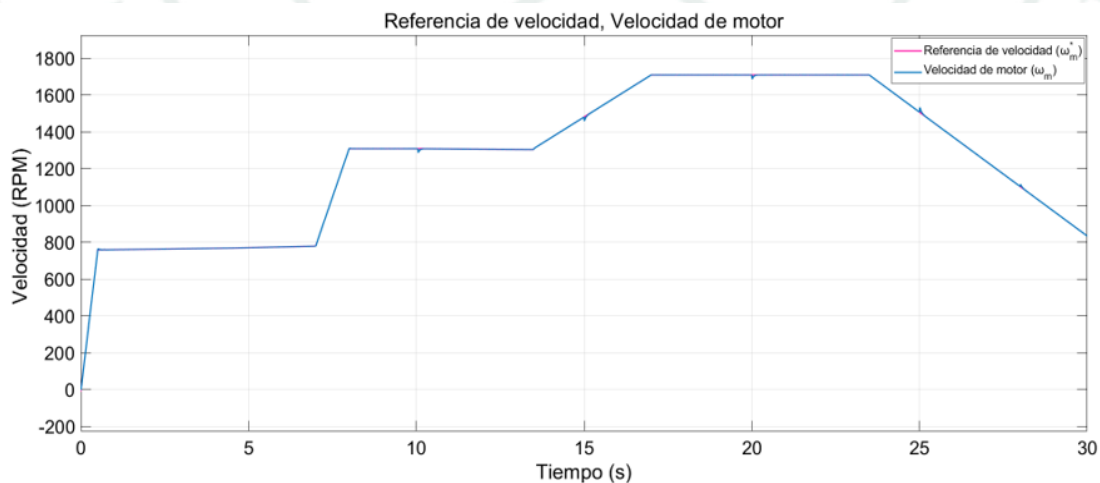


Figura 62. Velocidad de motor de inducción

En la Figura 63 se muestra el torque electromagnético del motor comparado con el torque de carga. Se observa que el torque del motor responde rápidamente a las variaciones de carga, estabilizándose en torno al valor requerido para mantener la velocidad de referencia en cada etapa. Durante los transitorios de cambio de referencia, el torque presenta incrementos breves que permiten compensar la inercia del sistema, evidenciando un comportamiento dinámico adecuado.

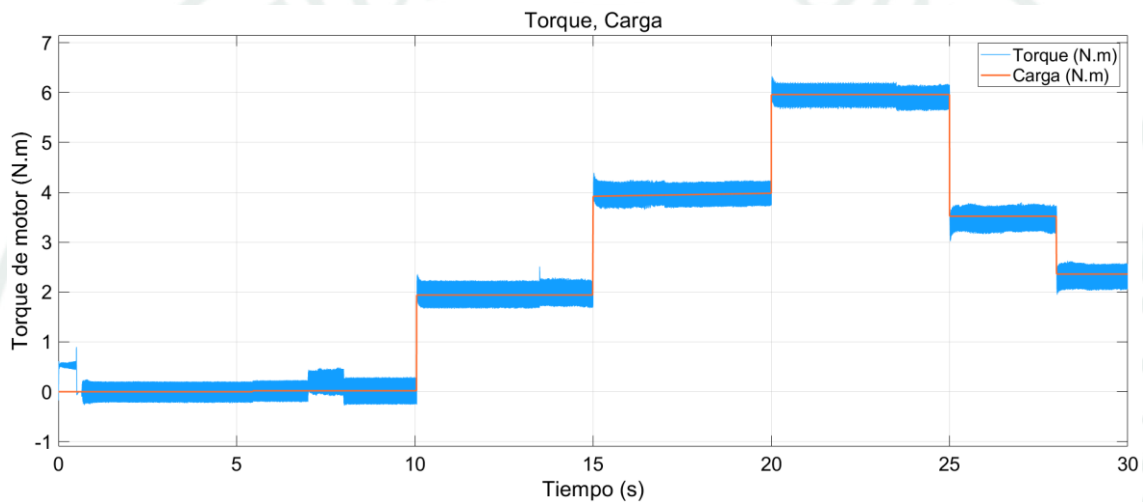


Figura 63. Torque electromagnético

La Figura 64 muestra las corrientes trifásicas del estator generadas por el control vectorial. Se observa que las amplitudes de i_{sa} , i_{sb} e i_{sc} varían de manera proporcional a los cambios de velocidad y torque aplicados durante la simulación. En los intervalos donde aumenta la referencia de velocidad o la carga mecánica, las corrientes incrementan su magnitud para suministrar el torque requerido; mientras que, en los periodos de menor exigencia, la amplitud disminuye. A pesar de estas variaciones, las corrientes permanecen balanceadas y con forma sinusoidal, lo que confirma un funcionamiento estable del control FOC.

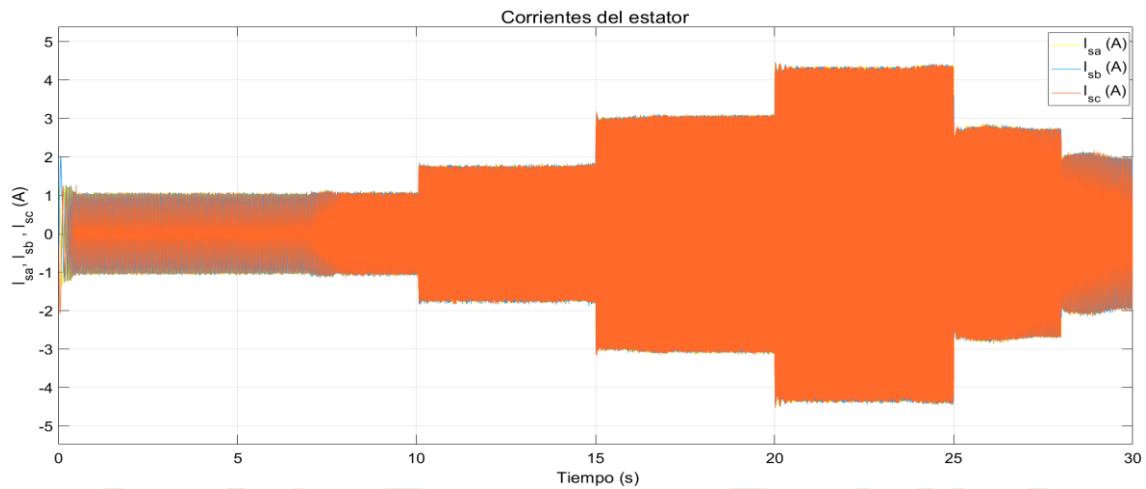


Figura 64. Corrientes del estator del motor de inducción

Se realizó un acercamiento (zoom) a la gráfica de las corrientes de fase, con el fin de observar con mayor detalle su comportamiento, tal como se muestra en la Figura 65.

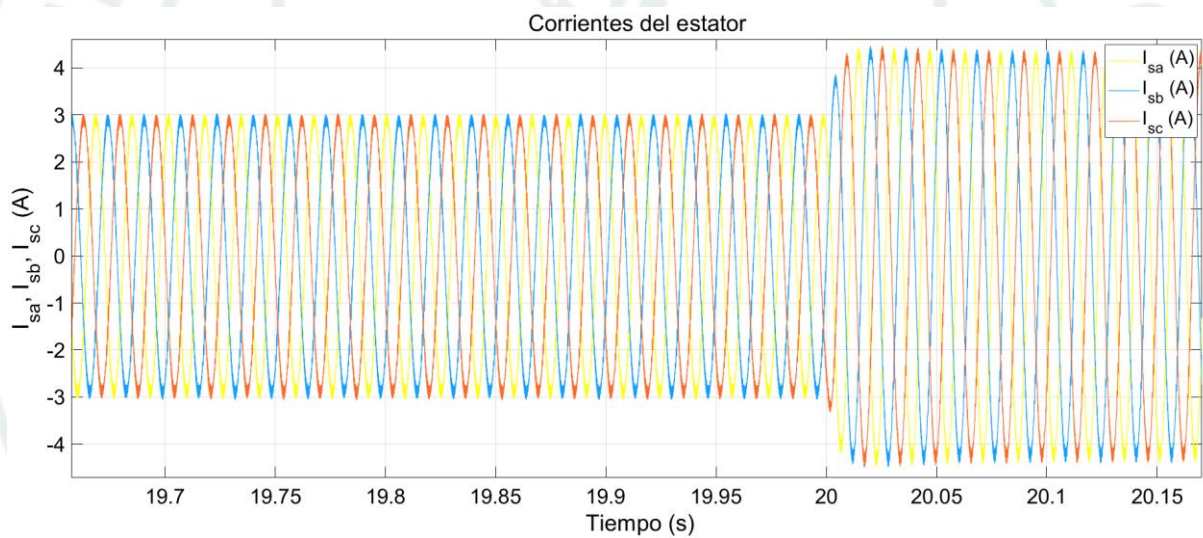


Figura 65. Corrientes del estator con acercamiento

La Figura 66 se muestran las tensiones trifásicas aplicadas a la carga por el inversor. Se observa un patrón de conmutación consistente, con niveles de tensión discretos y un desfase de 120° entre las fases, característico de un inversor trifásico balanceado.

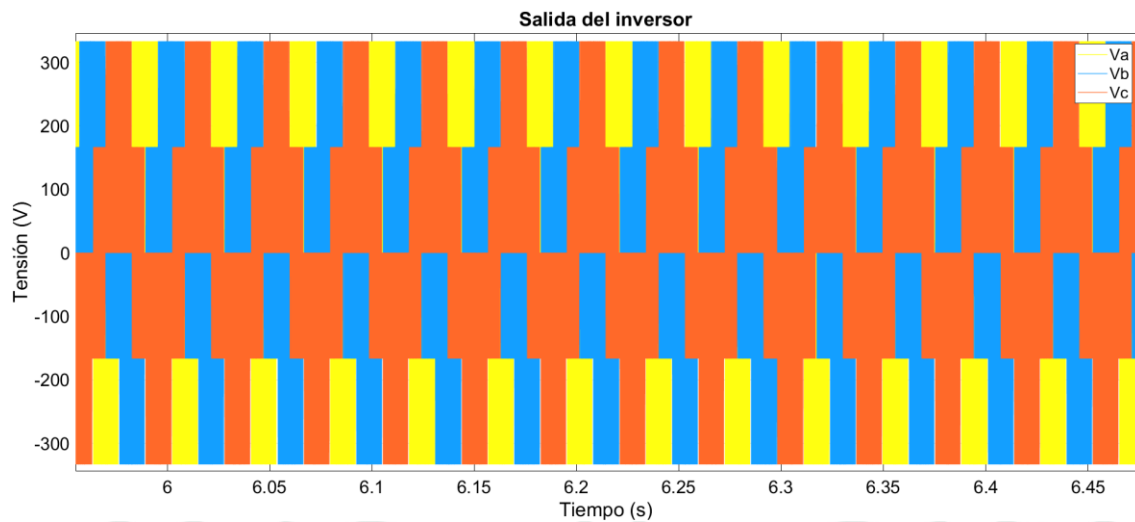
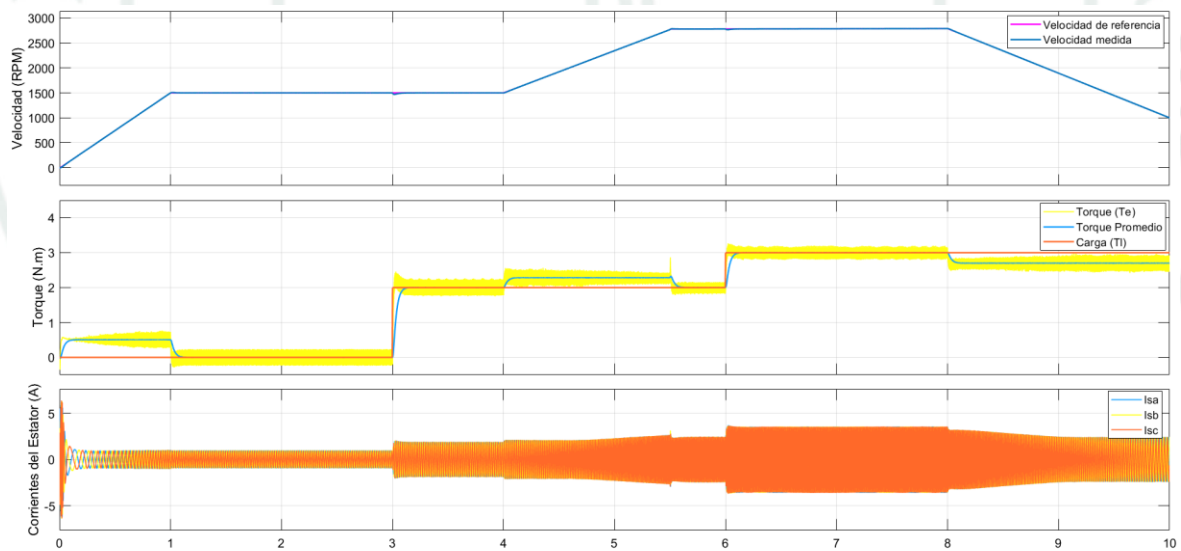


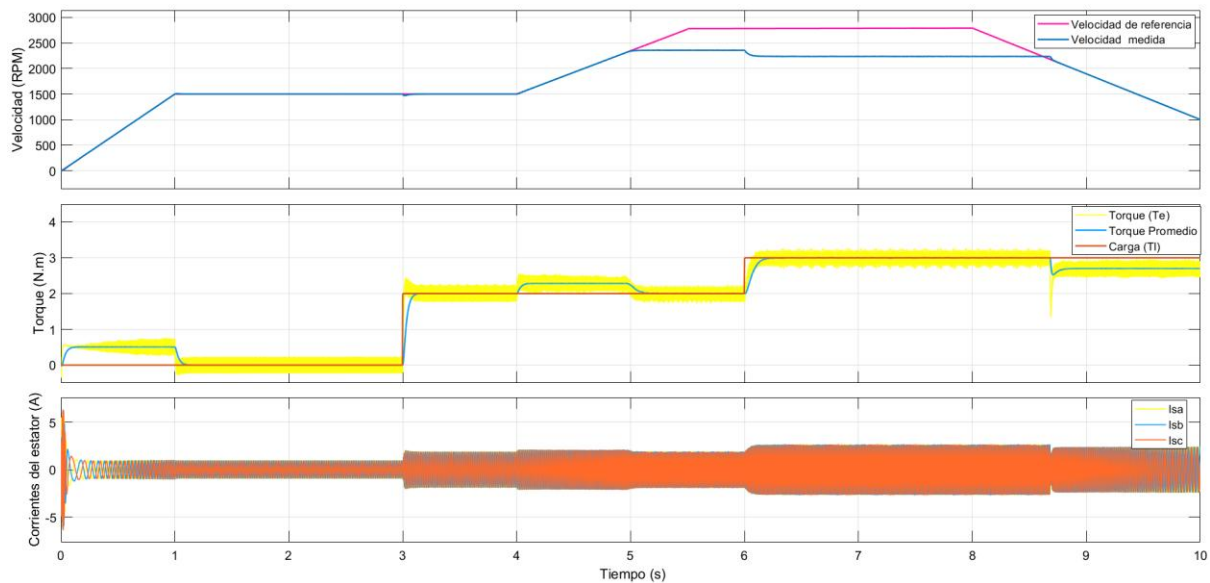
Figura 66. Tensión aplicada a la carga

4.10.2. Respuesta en control por debilitamiento de flujo

En la Figura 67 se presenta la comparación entre la respuesta si se utiliza flujo constante y debilitamiento de flujo para valores de velocidad mayores a la velocidad nominal



a)



b)

Figura 67. a) FOC con control por debilitamiento de flujo b) FOC sin control por debilitamiento de flujo

En la Fig. 67(a) se observa que, al incorporar la estrategia de debilitamiento de flujo, el controlador logra un seguimiento preciso de la velocidad de referencia incluso por encima de la velocidad nominal (1800 RPM). Esto se consigue mediante la reducción progresiva de la componente de flujo una vez alcanzada dicha velocidad, lo que evita la saturación del inversor y amplía el rango operativo del motor. El sistema presenta un comportamiento estable ante variaciones de carga, con corrientes de fase balanceadas y un ligero incremento de su amplitud en la región de alta velocidad, asociado al mayor requerimiento de torque.

En contraste, en la Fig. 67(b), sin control de debilitamiento de flujo, el sistema mantiene un buen desempeño únicamente dentro del rango nominal. Al superar la velocidad base, el inversor alcanza su límite de tensión, impidiendo la reducción del flujo y provocando una disminución del torque disponible, lo que limita el aumento de velocidad y genera mayores transitorios en las corrientes.

En conclusión, el uso de la técnica de debilitamiento de flujo permite operar el motor por encima de su velocidad nominal con adecuada estabilidad dinámica y control de torque, sin necesidad de sobredimensionar el inversor ni incrementar significativamente las pérdidas.

4.10.3. Resultado de simulación de control vectorial sensorless

Se realizaron simulaciones del modelo descrito y del algoritmo del filtro para los parámetros de la máquina de inducción, La simulación se llevó a cabo comparando la estimación de la velocidad y de las corrientes $\alpha\beta$ bajo diferentes cambios de carga y velocidad.

En la Figura 68 se compara la velocidad estimada por el EKF (ω_{m_est}) y la referencia impuesta (ω_{m_ref}). Se observa que la estimación del EKF sigue de manera muy cercana a la velocidad de referencia, con un error prácticamente despreciable durante todos los perfiles de operación. Esto se evidencia tanto en las transitorias iniciales como en los cambios escalonados de velocidad alrededor de los 0.5 s (de 800 a 1300 RPM), 2 s (a 1730 RPM) y 4 s (a 1180 RPM).

Las perturbaciones visibles alrededor de 0.8 s, 2.5 s y 3.5 s corresponden a cambios de torque/carga aplicados al motor.

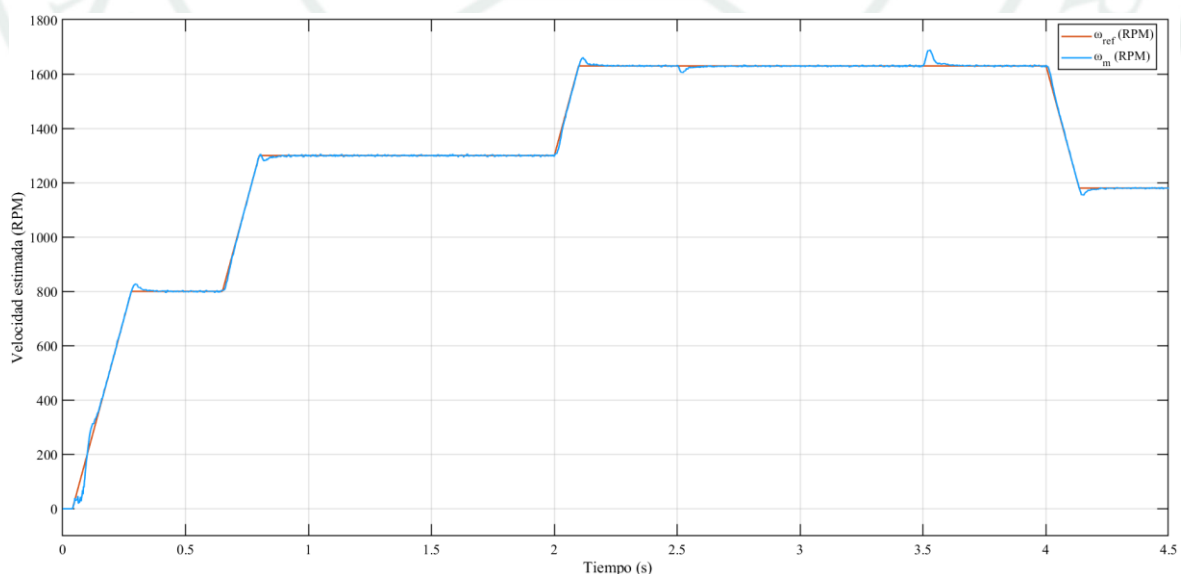


Figura 68. Estimación de velocidad mediante EKF (comparación con medición y referencia)

La Figura 69 muestra el comportamiento del torque electromagnético T_e del motor comparado con la referencia de torque de carga T_L .

Los picos más pronunciados alrededor de 0.7 s, 2.0 s y 4.0 seg. corresponden a cambios de velocidad, donde el motor debe acelerar o desacelerar para alcanzar la nueva consigna. Estos transitorios generan naturalmente un aumento momentáneo del torque para compensar la inercia del sistema. Por otra parte, las variaciones en 0.8 s, 2.5 s y 3.5 seg. están asociadas a cambios de carga/torques aplicados durante la simulación. El torque electromagnético ajusta su valor para equilibrar el torque de carga, mostrando una respuesta estable y rápida, a pesar de que la velocidad no sea medida sino estimada.

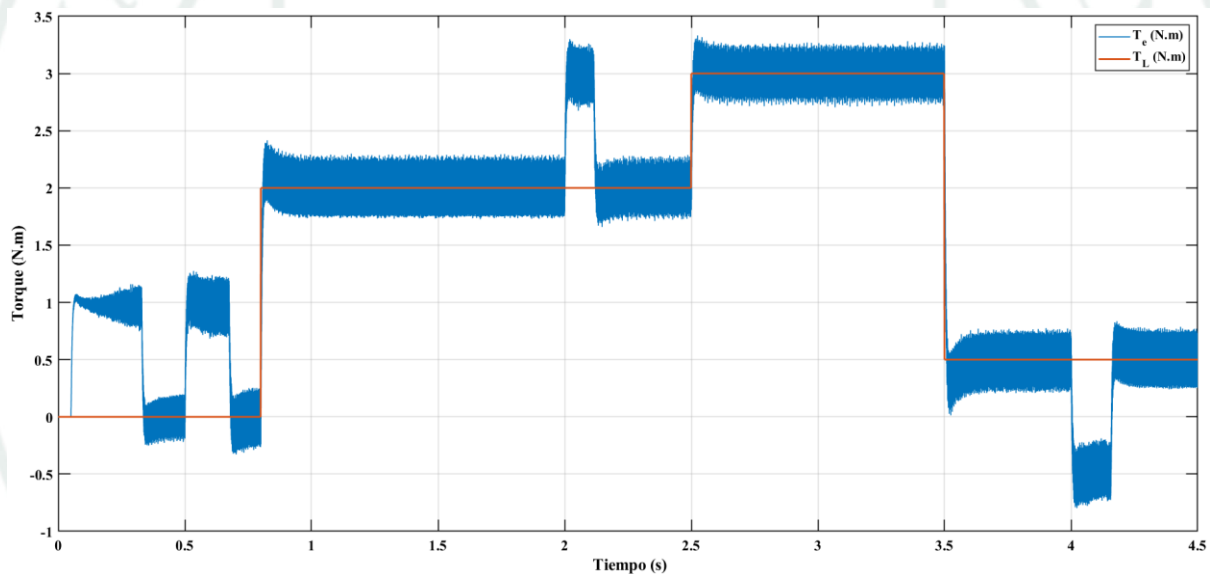


Figura 69. Torque electromagnético del motor bajo control FOC sensorless

En la Figura 70 se muestran las corrientes trifásicas del estator i_{sa} , i_{sb} e i_{sc} obtenidas durante la operación del control vectorial sensorless. Las señales presentan una forma sinusoidal equilibrada en estado estacionario, con amplitudes que varían según las exigencias de torque y los cambios de velocidad aplicados en la simulación.

Los incrementos en la magnitud de las corrientes alrededor de 0.5 s, 2.0 s y 4.0 s corresponden a las etapas de aceleración o desaceleración del motor y los incrementos permanentes de corriente corresponden a los cambios de carga.

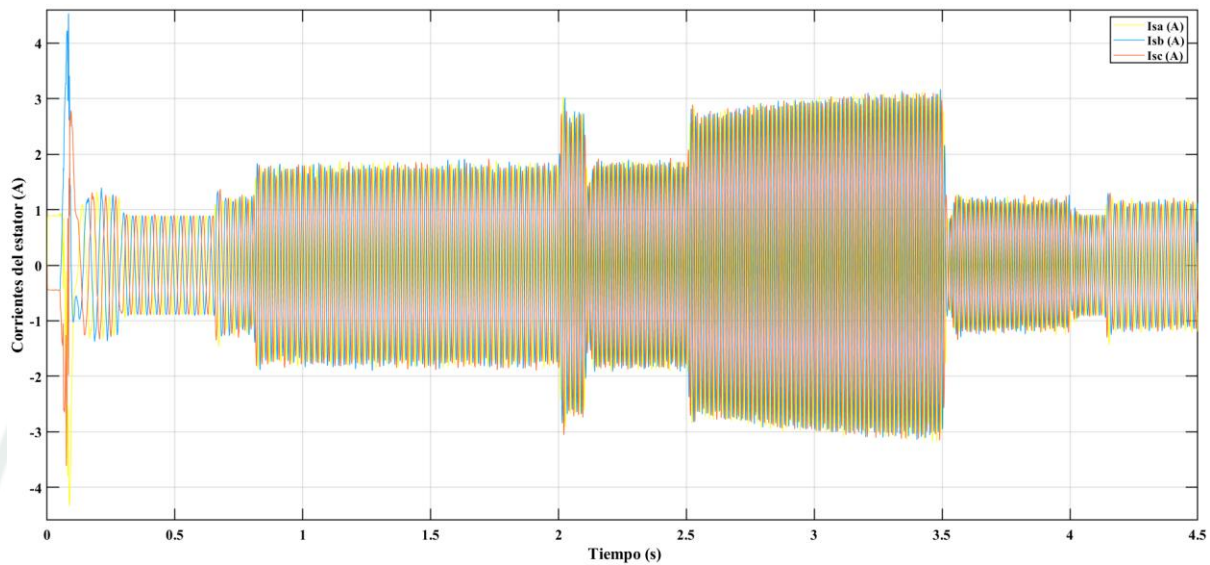


Figura 70. Corrientes trifásicas del estator i_{sa} , i_{sb} , i_{sc}

Las corrientes $\alpha\beta$, son estimadas por el EKF, las cuales, son necesarios para un correcto comportamiento tanto torque como velocidad a continuación se muestra comparación de corrientes $\alpha\beta$ que vienen de la transformada de Clarke de las corrientes medidas y corrientes $\alpha\beta$ estimadas.

En la Figura 71 se presenta la comparación entre las corrientes del estator en el marco $\alpha\beta$ obtenidas mediante la transformada de Clarke aplicada a las corrientes medidas y las corrientes $\alpha\beta$ estimadas por el EKF.

En la vista general a lo largo de toda la simulación (0–4.5 s), se observa que las corrientes estimadas siguen adecuadamente la forma y amplitud de las corrientes medidas, incluso ante los cambios de velocidad y las variaciones de carga aplicadas. Las diferencias entre ambas señales son mínimas y se mantienen dentro de rangos aceptables, lo que indica una estimación estable y libre de oscilaciones significativas.

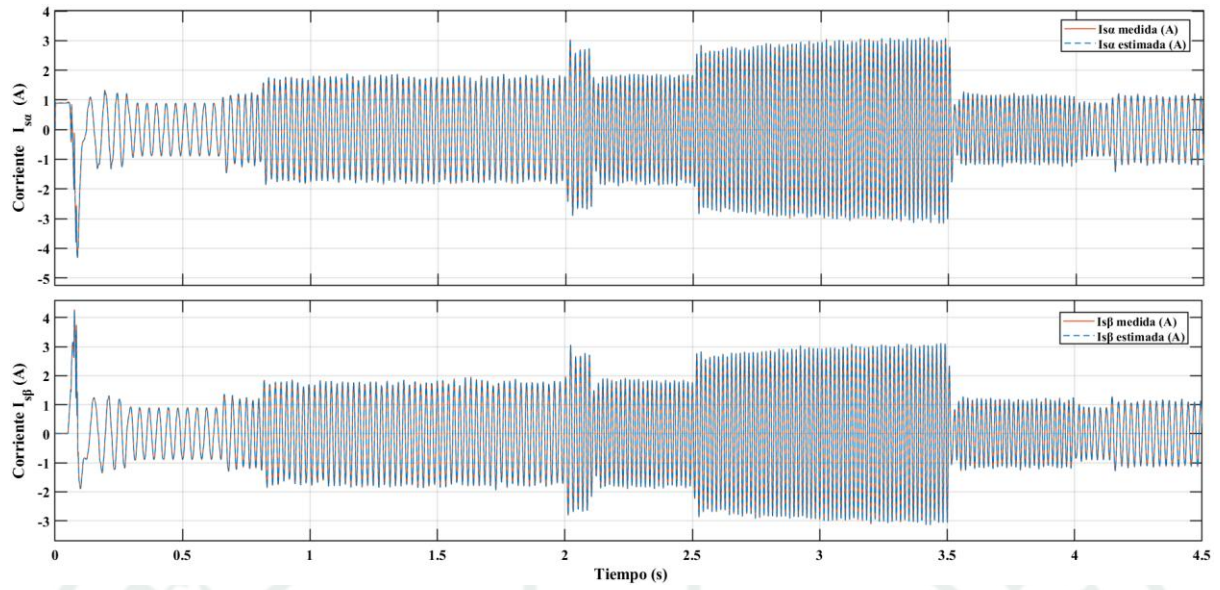


Figura 71. Corrientes $\alpha\beta$ medidas vs. $\alpha\beta$ estimadas

La Figura 72 ampliada (zoom) permite apreciar con mayor detalle el comportamiento durante los primeros 0.9 s. Se verifica que las corrientes estimadas presentan una excelente coincidencia con las corrientes medidas, reproduciendo correctamente la frecuencia y fase.

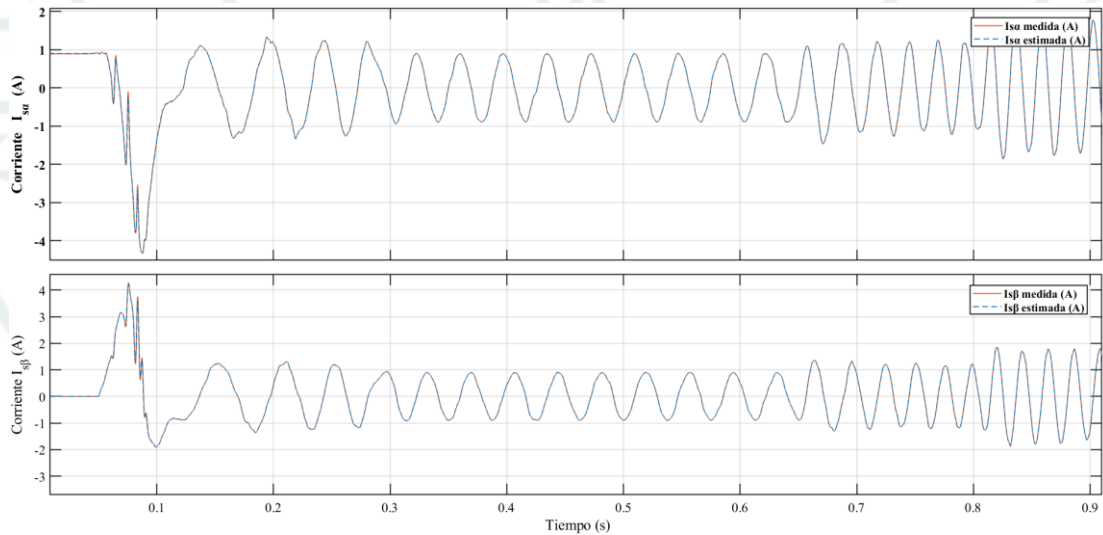


Figura 72. Corrientes $\alpha\beta$ medidas vs. $\alpha\beta$ estimadas (zoom entre 0–0.9 s)

Para obtener los resultados mostrados, se realizó un ajuste heurístico de las matrices del EKF mediante múltiples pruebas y simulaciones, identificando los valores que ofrecieron el mejor compromiso entre estabilidad, rapidez de convergencia y precisión. La matriz de proceso Q se configuró como diagonal con $q_{11} = q_{22} = 0.001$, $q_{33} = q_{44} = 0.0001$ y $q_{55} = 40$,

reflejando en la dinámica de la velocidad y menor en las variables eléctricas. La matriz de medición R se ajustó a $r_{11} = r_{22} = 0.8$, valor determinado de forma empírica mediante múltiples simulaciones, buscando el balance óptimo entre rechazo al ruido de medición y velocidad de respuesta del estimador. La matriz inicial de covarianza P se estableció con $p_{11} = p_{22} = 1$, $p_{33} = p_{44} = 10$ y $p_{55} = 500$, debido a la incertidumbre inicial de la velocidad.

Asimismo, se comprobó que la sincronización temporal es crítica para el desempeño del método *sensorless*. El EKF y los lazos de corriente del FOC deben operar con el mismo tiempo de muestreo ($T_{ss} = 1 \times 10^{-4}$ s), mientras que el modelo del sistema puede integrarse con un paso temporal menor ($T_s = 1 \times 10^{-5}$ s) sin degradar el desempeño del estimador, siempre que las corrientes suministradas al EKF estén correctamente muestreadas y sincronizadas. Esta configuración permitió obtener una estimación estable y precisa de las corrientes en ejes $\alpha\beta$ y de la velocidad del rotor.

Con el objetivo de evaluar la robustez del estimador basado en el Filtro de Kalman Extendido (EKF) frente a incertidumbre en las mediciones, se añadió ruido blanco gaussiano a las señales de corriente del estator. Inicialmente, esta perturbación se refleja en un incremento del rizado en la velocidad estimada, como se observa en la Figura 73. Este resultado muestra una limitación inicial del estimador.

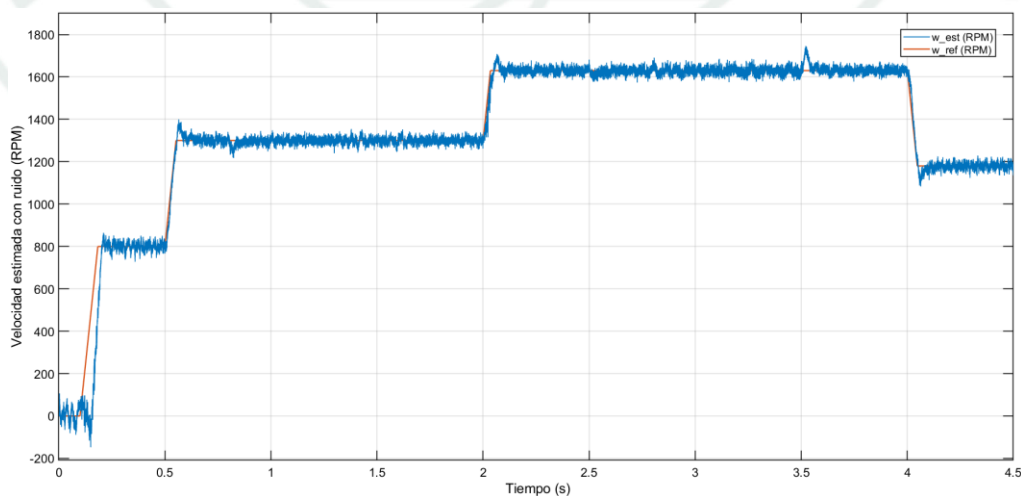


Figura 73. Respuesta de la velocidad estimada ante ruido de medición

Posteriormente, se ajustaron los valores de la matriz de covarianza del ruido de medición a $R = \text{diag}(2.5, 2.5)$, lo que permitió reducir significativamente el efecto del ruido sin necesidad de incorporar filtros adicionales. Como se aprecia en la Figura 74, la velocidad estimada presenta un seguimiento más suave y cercano a la referencia, manteniendo una buena dinámica durante los cambios de régimen y sin introducir retardos apreciables.

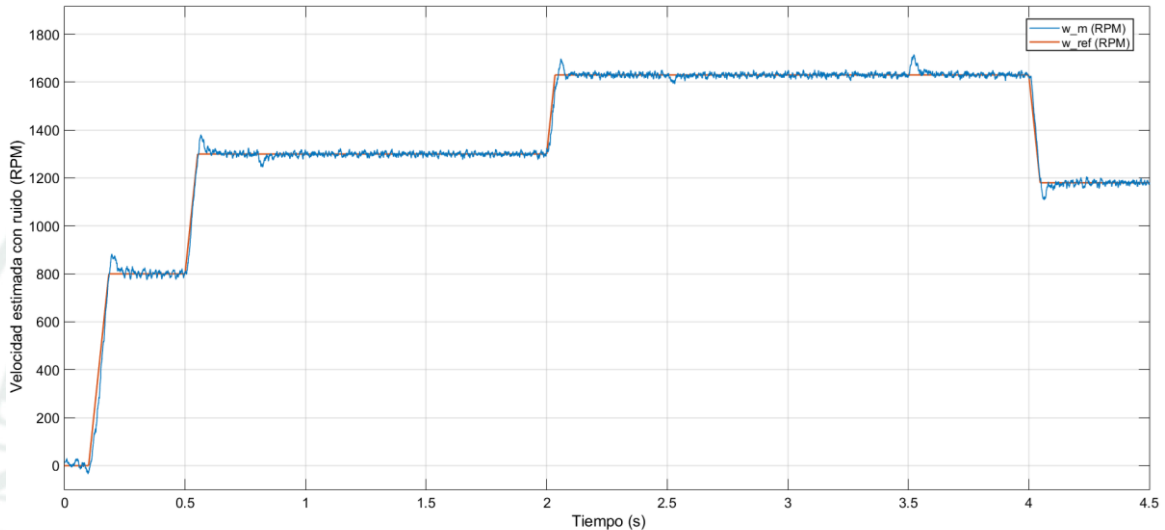


Figura 74. Respuesta de la velocidad estimada tras el ajuste de la matriz de covarianza R

En la Figura 75 se muestra la comparación entre las corrientes medidas y las corrientes estimadas por el EKF. A pesar de la presencia de ruido en las mediciones, el estimador logra reconstruir señales sinusoidales balanceadas, con menor rizado y buena coherencia en amplitud y fase. Esto confirma que el EKF actúa de forma efectiva como un filtro óptimo, atenuando el ruido de alta frecuencia mientras preserva la información dinámica relevante para el control.

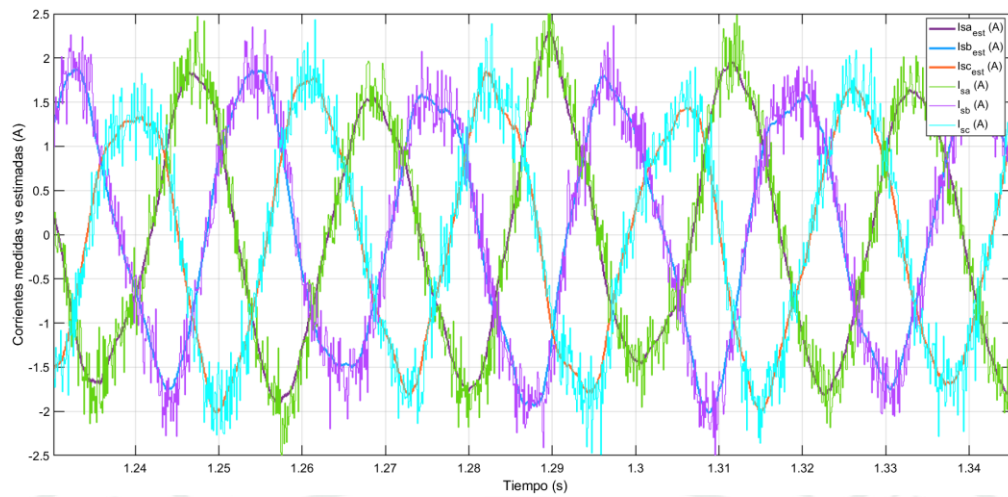


Figura 75. Comparación entre las corrientes de fase medidas (con ruido inyectado) y las estimadas por el EKF

CAPÍTULO V: DISEÑO DE CONTROL TOLERANTE A FALLAS DE SENSORES DE VELOCIDAD Y CORRIENTES DE CONTROL VECTORIAL DE MOTOR DE INDUCCIÓN

En este capítulo se aborda el diagnóstico y la reconfiguración de fallas en dos sensores de corriente de fase y en un sensor de velocidad de motores de inducción con control vectorial.

Para garantizar el nivel requerido de seguridad y desempeño dentro de la red de sensores de un accionamiento eléctrico, los métodos de identificación de fallas deben cumplir los siguientes requisitos (Azzoug, Sahraoui, Pusca, Ameid, & Romary, 2021):

- **Rapidez:** Detección casi inmediata ante la ocurrencia de una falla.
- **Simplicidad:** Algoritmos computacionalmente eficientes aptos para implementación en tiempo real.
- **Selectividad:** Identificación precisa del sensor defectuoso, evitando falsos positivos en sensores sanos.
- **Independencia:** Desacople respecto a otros sistemas de control o diagnóstico existentes en el inversor.
- **Universalidad:** Aplicabilidad a una amplia gama de accionamientos eléctricos.

A diferencia de los enfoques tradicionales que suelen analizar un único tipo de fallo, en este trabajo se analizan múltiples escenarios de falla en sensores.

5.1. Algoritmo de Diagnóstico, Aislamiento y Reconfiguración

El algoritmo propuesto se diseña bajo el supuesto de que, durante la operación del sistema, las fallas no ocurren de manera simultánea en sensores de distinta naturaleza. En particular, se asume que las fallas pueden presentarse únicamente en una categoría de sensores a la vez: sensores de corriente de fase o sensor de velocidad, pero no en ambos simultáneamente (Wu, Zhao, & Liu, 2016).

La arquitectura propuesta integra técnicas de ventanas deslizantes, un Filtro de Kalman Extendido (EKF) y un modelo analítico ideal del motor. A continuación, se detallan los mecanismos de detección y compensación para cada variable.

Definición de Ventana Deslizante: Es una técnica de procesamiento en línea que mantiene un conjunto de N muestras consecutivas dentro de un intervalo fijo. A medida que ingresa una nueva muestra, la más antigua se descarta, garantizando que las métricas calculadas se basen siempre en la información más reciente. Este enfoque permite una detección adaptativa y en tiempo real.

5.1.1. Detección y Corrección en Sensores de Corriente i_{sa} , i_{sc}

5.1.1.1. Fallas por Offset (Desplazamiento de DC)

- **Detección:** Se calcula el promedio aritmético de las muestras dentro de la ventana deslizante. En condiciones sanas, este promedio tiende a cero para corrientes alternas. La persistencia de un valor medio distinto de cero indica una falla de offset.
- **Reconfiguración:** Se aplica una compensación aditiva, restando el valor del offset detectado a la medición bruta para recuperar la señal correcta.

5.1.1.2. Fallas de Escalamiento (Ganancia)

- **Detección:** Se comparan los valores eficaces (RMS) de las corrientes medidas frente a las estimadas por el EKF. Si la relación (cociente) entre ambas magnitudes se desvía significativamente de la unidad (fuera de un rango de tolerancia predefinido), se diagnostica un error de ganancia.
- **Reconfiguración:** Se aplica un factor de normalización multiplicativo para ajustar la amplitud de la señal medida que es el cociente obtenido en la detección.

5.1.1.3. Fallas de Circuito Abierto

- **Detección:** Se evalúa el residuo instantáneo entre la corriente medida y la estimada por EKF, junto con la verificación de magnitudes cercanas a cero. Un residuo elevado y persistente confirma la pérdida de señal.
- **Reconfiguración:** Se aísla el sensor defectuoso y se sustituye su señal por la estimación generada por el modelo dinámico ideal del motor (Sensor virtual), garantizando la continuidad del lazo de corriente.

5.1.2. Diagnóstico y Corrección en Sensor de Velocidad (ω_m)

- **Detección:** Se monitorea el residuo entre la velocidad medida por el encoder y la velocidad estimada por el observador EKF. Una discrepancia abrupta superior al umbral establecido indica una falla por circuito abierto o avería mecánica del sensor.
- **Reconfiguración:** El sistema conmuta automáticamente a un esquema sensorless, sustituyendo la señal del encoder por la velocidad estimada del EKF, asegurando la estabilidad del control vectorial.

5.1.3. Lógica de Reconfiguración del Sistema

Una vez identificada la anomalía, el algoritmo activa banderas lógicas (flags) que clasifican el evento y ejecutan la acción correctiva correspondiente de forma automática:

- **Falla de Offset:** Calibración en línea (Resta del promedio).
- **Falla de Escalamiento:** Corrección de Ganancia (Multiplicación por factor inverso).
- **Falla de Sensor (Abierto):** Sustitución por Estimador (Modelo Ideal para corrientes, EKF para velocidad).

De esta manera, el sistema de control vectorial mantiene su estabilidad y tolerancia a fallas sin necesidad de sensores redundantes. La eficacia y robustez del enfoque se demuestran

mediante simulaciones bajo condiciones exigentes, incluyendo variaciones de torque, cambios de velocidad y la ocurrencia de distintos tipos de fallas en los sensores.

5.2. Implementación de diagnóstico, aislamiento y reconfiguración de fallas

En esta sección se detalla la formulación matemática y la lógica de implementación para los tipos de fallas considerados.

5.2.1. Falla de Circuito Abierto en Sensores (Detección Inmediata)

La detección primaria de desconexiones o fallas abruptas en los sensores de corriente y velocidad se basa en el análisis de residuos instantáneos. Estos se calculan comparando en tiempo real las señales medidas por los sensores físicos con las estimaciones proporcionadas por el Filtro de Kalman Extendido (EKF). Se definen los siguientes residuos absolutos:

$$i_{sa_{res}} = |i_{sa} - \hat{i}_{sa}| \quad (5.1)$$

$$i_{sc_{res}} = |i_{sc} - \hat{i}_{sc}| \quad (5.2)$$

$$\omega_{m_{res}} = |\omega_m - \hat{\omega}_m| \quad (5.3)$$

Bajo condiciones normales, estos índices permanecen por debajo de un umbral, mientras que en presencia de una falla superan dicho límite. Los umbrales pueden definirse como:

- **Corriente:** $K_{ca_i} \approx 1/3$ a $1/5$ de la corriente nominal pico
- **Velocidad:** $K_{\omega} \approx 1/10$ de la velocidad de referencia nominal.

5.2.2. Error de Medición por Escalamiento (Falla de Ganancia)

El error de escalamiento ocurre cuando el sensor entrega valores proporcionalmente incorrectos debido a desajustes en el acondicionamiento de señal o degradación del componente. Dado que la forma de onda sigue siendo sinusoidal, el residuo instantáneo puede ser ambiguo.

Para identificar robustamente esta falla, se define el Índice de Ganancia (V_n), el cual compara la energía de la señal medida (i_n) frente a la estimada por el EKF ($\hat{i}_n$) dentro de una ventana deslizante de longitud L (Wu, Zhao, & Liu, 2016).

$$V_n[k] = \frac{RMS(i_n)_L}{RMS(\hat{i}_n)_L} = \frac{\sqrt{\frac{1}{L} \sum_{j=k-L}^k (i_n[j])^2}}{\sqrt{\frac{1}{L} \sum_{j=k-L}^k (\hat{i}_n[j])^2}} \quad (5.4)$$

Donde, k es el instante de muestreo, L , representa el número de muestras correspondiente a un periodo fundamental (o múltiplo de este), calculado adaptativamente según la velocidad del rotor:

$$L = N \frac{T}{T_s} = N \frac{60}{npT_s} \quad (5.5)$$

donde: T_s período de muestreo, T es el período fundamental, n es velocidad del rotor (RPM), p es número de pares de polos, N es factor que incrementa la capacidad de filtrado.

Criterio de decisión

Si $V_n[k] \approx 1$ es un sensor sano mientras, si $V_n[k] > 1$ o $V_n[k] < 1$ es un sensor con ganancia incorrecta. Para simplificar la detección, se puede definir dos umbrales $0.95 \leq V_n[k] \leq 1.05$ (sensor sano), $V_n[k] < 0.95$ o $V_n[k] > 1.05$ (falla de escalamiento)

Corrección

Una vez confirmada la falla y descartado el circuito abierto, se aplica una compensación multiplicativa inversa para restaurar la amplitud real:

$$i_{sa_{com}}[k] = \frac{i_{sa}[k]}{V_a[k]} \quad (5.6)$$

$$i_{sc_{com}}[k] = \frac{i_{sc}[k]}{V_c[k]} \quad (5.7)$$

Esto restaura la proporcionalidad correcta entre la corriente medida y la real, permitiendo mantener estable el control vectorial sin reemplazar el sensor.

5.2.3. Discriminación Robusta: Índice de Ceros (M_n)

Para evitar confusiones entre una falla de ganancia severa y un circuito abierto (donde la corriente cae a cero), se introduce el índice lógico de magnitud nula M_n . Este calcula la proporción de muestras dentro de la ventana que se encuentran cercanas a cero (debajo de un umbral de ruido K_0):

$$m_n(k) = \begin{cases} 1, & \text{si } |i_n[k]| \leq K_0 \\ 0, & \text{si } |i_n[k]| > K_0 \end{cases} \quad (5.8)$$

y su promedio dentro de una ventana deslizante:

$$M_n(k) = \frac{1}{L} \sum_{j=k-L+1}^k m_n(k) \quad (5.9)$$

- **Circuito abierto:** La corriente es nula casi todo el tiempo $M_n \approx 1$
- **Escalamiento/Operación Normal:** La corriente cruza por cero solo periódicamente M_n es bajo.

El diagrama de flujo mostrado en la Figura 76 resume el proceso de detección de falla de circuito abierto y escalamiento en los sensores de corriente. La detección de fallas se realiza de manera independiente por fase. En cada fase, la falla de circuito abierto se evalúa prioritariamente, y únicamente en ausencia de esta se analiza la presencia de una falla de escala. La detección de falla en el sensor de velocidad se ejecuta de manera independiente respecto a las fallas de corriente.

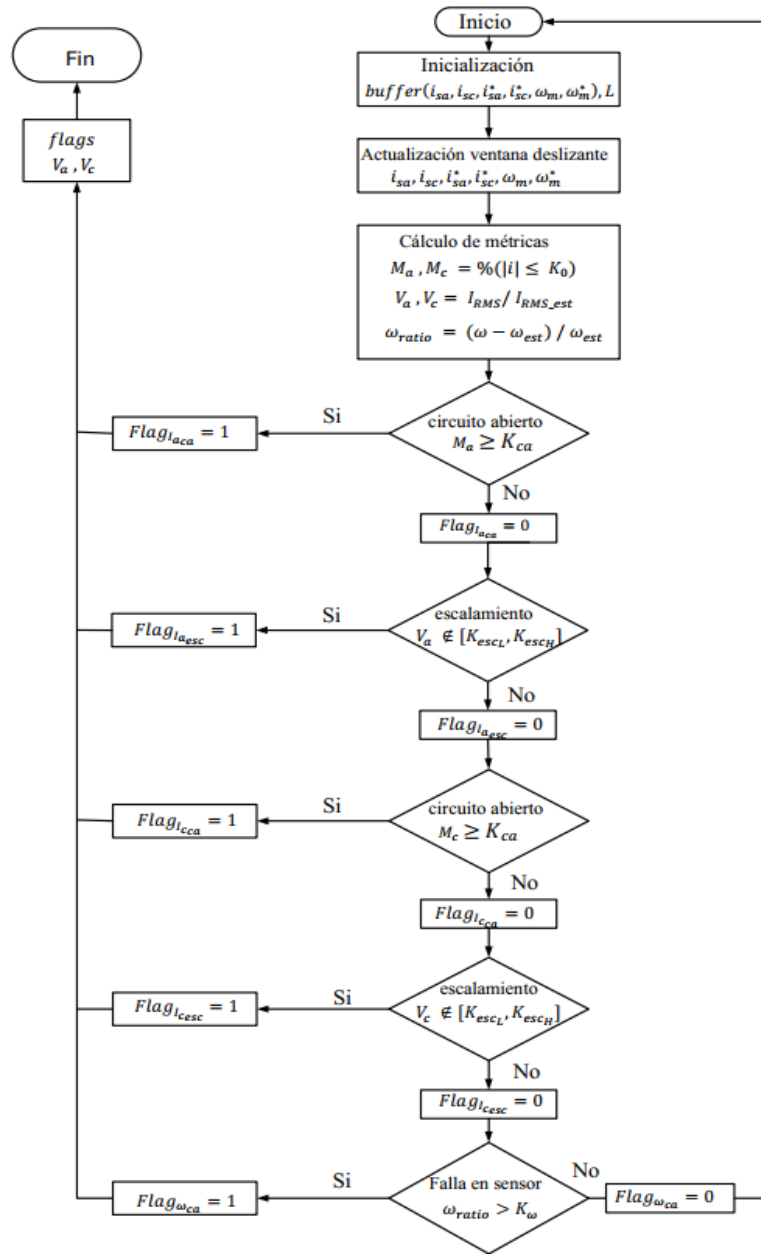


Figura 76. Diagrama de flujo de diagnóstico de fallas en sensores de corriente

5.2.4. Error de Medición por Offset (Desplazamiento DC)

Los sensores de corriente presentan, de manera inevitable, un offset de corriente continua (DC offset). Este error aparece por pequeñas diferencias en las tensiones de alimentación, variaciones térmicas o imperfecciones en los circuitos analógicos, produciendo una desviación constante en la corriente medida. Aunque el offset parezca pequeño, puede

afectar el desempeño del control vectorial, ya que introduce un error permanente en el lazo de corriente.

Para identificar el offset sin ser afectados por el ruido de conmutación o armónicos del PWM, se aprovecha la propiedad de que el promedio de una corriente sinusoidal pura en un periodo es cero (Wu, Zhao, & Liu, 2016).

Se calcula el promedio simple en la ventana deslizante:

$$O_n[k] = \frac{1}{L} \sum_{j=k-L+1}^k i_n[j] \quad (5.10)$$

$$S_n[k] = \frac{1}{L} \sum_{j=k-L+1}^k O_n[j] \quad (5.11)$$

El primer promedio O_n reduce el ruido del PWM y el segundo promedio S_n estabiliza aún más la señal y permite detectar la presencia de un valor DC persistente.

En operación normal $S_n[k] \approx 0$ y con offset presente $|S_n[k]| > K_{off}$, donde el umbral K_{off} suele ser entre 1/15 y 1/8 de la corriente nominal.

Cuando se detecta la falla, el error se compensa reajustando la medición según:

$$\tilde{i}_a = i_a - O_a \quad (5.12)$$

$$\tilde{i}_c = i_c - O_c \quad (5.13)$$

donde O_a y O_c representan los offsets detectados en cada fase.

5.2.5. Estrategia de Reconfiguración (FTC)

La detección de una falla activa los flags correspondientes, desencadenando la reconfiguración inmediata del lazo de control para evitar la inestabilidad del accionamiento.

5.2.5.1. Reconfiguración de Corrientes (Modelo Ideal)

En caso de pérdida total del sensor de corriente (Circuito Abierto), las mediciones son sustituidas por estimaciones generadas mediante el Modelo Dinámico Ideal del motor de

inducción. Este modelo actúa como un "sensor virtual", prediciendo la corriente actual [k] a partir del estado anterior [k-1] y el voltaje aplicado:

$$\hat{i}_{s\alpha}[k] = a_1 \hat{i}_{s\alpha}[k-1] + \frac{b_1}{T_r} \hat{\psi}'_{r\alpha}[k-1] + b_1 \omega_r[k-1] \hat{\psi}'_{r\beta}[k-1] + c_1 v_{s\alpha}[k-1] \quad (5.14)$$

$$\hat{i}_{s\beta}[k] = a_1 \hat{i}_{s\beta}[k-1] + \frac{b_1}{T_r} \hat{\psi}'_{r\beta}[k-1] - b_1 \omega_r[k-1] \hat{\psi}'_{r\alpha}[k-1] + c_1 v_{s\beta}[k-1] \quad (5.15)$$

$$a_1 = 1 - \frac{T(r_s L_r'^2 + r_r' L_m^2)}{\sigma L_s L_r'^2}, b_1 = \frac{T L_m}{\sigma L_s L_r'}, c_1 = \frac{T}{\sigma L_s}$$

$$\hat{y}[k] = [\hat{i}_\alpha[k] \quad \hat{i}_\beta[k]]^T$$

Este modelo depende críticamente de la estimación precisa del flujo del rotor, el cual se calcula discretamente como:

$$\frac{d\hat{\psi}'_{r\alpha}}{dt} = \frac{L_m r_r'}{L_r'} i_{s\alpha} - \frac{r_r'}{L_r'} \hat{\psi}'_{r\alpha} - \omega_r \hat{\psi}'_{r\beta}$$

$$\frac{d\hat{\psi}'_{r\beta}}{dt} = \frac{L_m r_r'}{L_r'} i_{s\beta} - \frac{r_r'}{L_r'} \hat{\psi}'_{r\beta} + \omega_r \hat{\psi}'_{r\alpha}$$

$$\hat{\psi}'_{r\alpha}[k] = a_2 \hat{\psi}'_{r\alpha}[k-1] - b_2 \hat{\psi}'_{r\beta}[k-1] + c_2 \hat{i}_{s\alpha}[k-1] \quad (5.16)$$

$$\hat{\psi}'_{r\beta}[k] = a_2 \hat{\psi}'_{r\beta}[k-1] + b_2 \hat{\psi}'_{r\alpha}[k-1] + c_2 \hat{i}_{s\beta}[k-1] \quad (5.17)$$

$$a_2 = \left(1 - \frac{T r_r'}{L_r'}\right), b_2 = T \omega_r[k-1], c_2 = \frac{T L_m r_r'}{L_r'}$$

Donde las corrientes de entrada $\hat{i}_s$ pueden provenir del propio estimador (en modo falla) o de los sensores sanos.

5.2.5.2.Reconfiguración de Velocidad

Si la falla ocurre en el encoder mecánico, el sistema conmuta a modo Sensorless, utilizando la velocidad eléctrica estimada $\hat{\omega}_r$ proporcionada por el EKF como variable de realimentación para el control vectorial.

5.3. Arquitectura del Sistema Propuesto

La Figura 77 presenta la integración del Control Vectorial (FOC) con la estrategia FTC propuesta. En el lazo superior, los controladores convencionales generan las referencias de tensión que, tras la transformación al marco estacionario y modulación SVPWM, accionan el inversor.

Las corrientes medidas (i_{sa} , i_{sc}), alimentan rutas paralelas de diagnóstico:

- **Compensación de Offset:** Un módulo dedicado exclusivamente a estimar y corregir el error de DC mediante ventanas deslizantes.
- **Detección de Fallas:** Un algoritmo central que procesa las señales (calculando sus propios parámetros de ventana L) y las compara con las estimaciones del EKF para identificar fallas de circuito abierto o escalamiento.
- **Reconfiguración:** Un sistema selector que, basado en el diagnóstico, conmuta entre señales medidas, compensadas o estimadas (provenientes del Modelo Ideal o EKF).

Finalmente, son estas variables reconfiguradas las que cierran el lazo de control, garantizando la estabilidad y el desempeño del accionamiento ante fallas sin requerir redundancia de hardware.

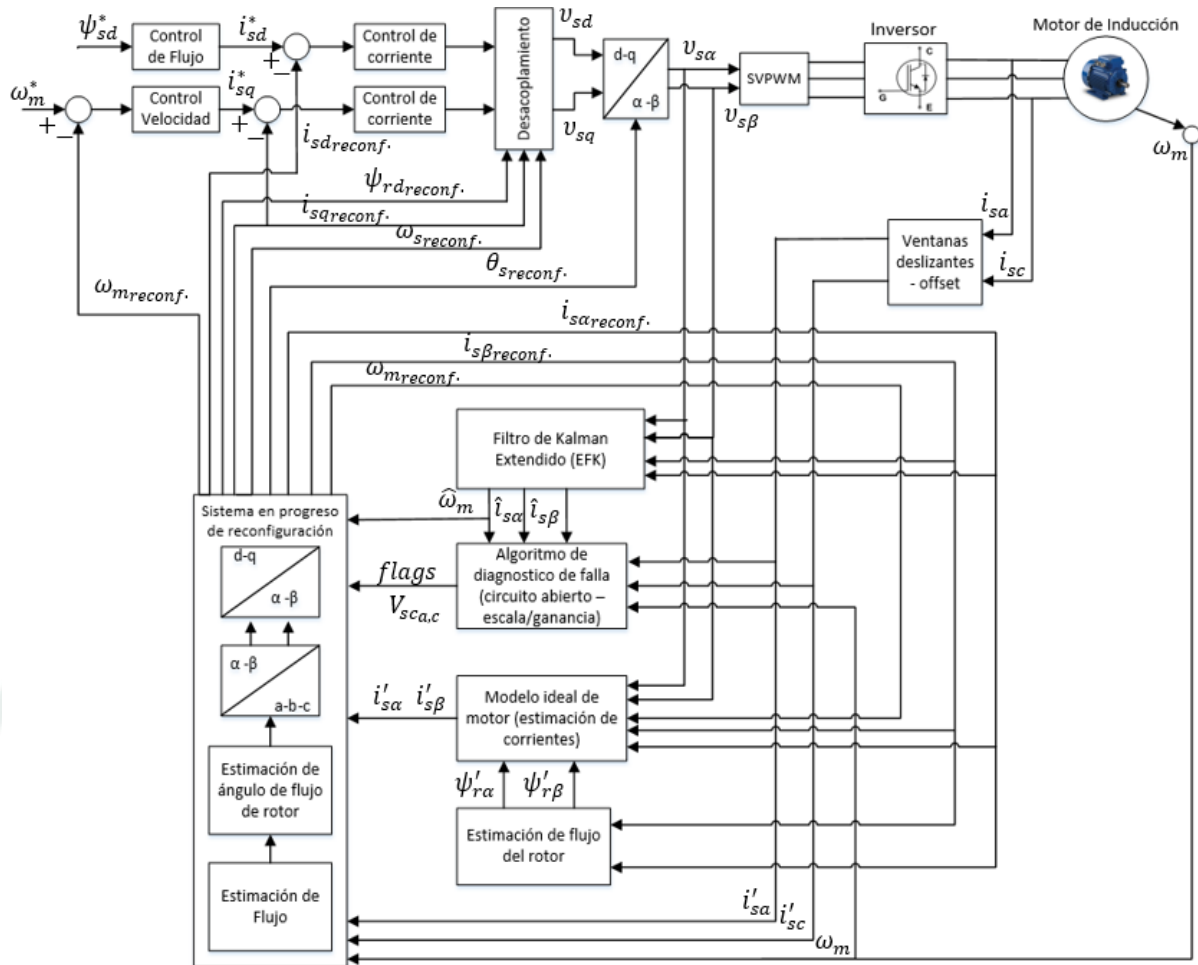


Figura 77. Sistema de control vectorial del motor de inducción con EKF y detección de fallas en sensores – Elaboración propia.

5.4. Resultados de simulación a diferente tipo de fallas

En esta sección se presentan los resultados obtenidos al evaluar el desempeño del sistema de control vectorial (FOC) frente a distintos tipos de fallas en los sensores. Las pruebas se realizaron para validar la capacidad del algoritmo de detectar y reconfigurar errores en tiempo real y mantener la estabilidad del accionamiento.

Se evaluaron cuatro fallas principales dentro del sistema FOC del motor de inducción: la falla por circuito abierto en los sensores de corriente y velocidad, equivalente a una pérdida abrupta de señal; la falla tipo offset en las corrientes, que introduce una componente DC debido

a desbalances o errores en la etapa analógica de medición; la falla tipo escalamiento, asociada a variaciones en la ganancia o calibración del sensor de corriente.

Cada falla fue inyectada en diferentes instantes, analizando la respuesta de corrientes, torque y velocidad, además de la actuación del módulo FDI. Los resultados muestran que el sistema detecta las fallas con rapidez y activa la compensación correspondiente, restableciendo las señales utilizadas por el FOC y manteniendo una operación estable del motor.

5.4.1. Falla por circuito abierto de sensor velocidad ω_m

En esta prueba se aplica una falla de circuito abierto en el sensor de velocidad en el instante $t = 1.5$ s. La Figura 78 muestra la respuesta del sistema en términos de velocidad, torque y corrientes de fase. Antes de la falla, el accionamiento sigue correctamente la referencia y mantiene estabilidad en todas las variables. Al producirse la desconexión del sensor, la velocidad medida cae abruptamente a cero, generando una perturbación transitoria en el torque y en las corrientes. Sin embargo, la acción del observador permite sustituir la señal perdida y el sistema recupera la estabilidad en un corto intervalo.

Posteriormente, en $t = 2$ s se aplica un incremento de referencia de velocidad, ya bajo operación sensorless. El sistema responde de forma estable utilizando únicamente la velocidad estimada. Finalmente, en $t = 2.5$ s se introduce un aumento de carga que provoca un pequeño sobreimpulso en el torque debido al esfuerzo transitorio de compensación, pero que rápidamente se atenúa, confirmando la robustez del esquema ante la falla.

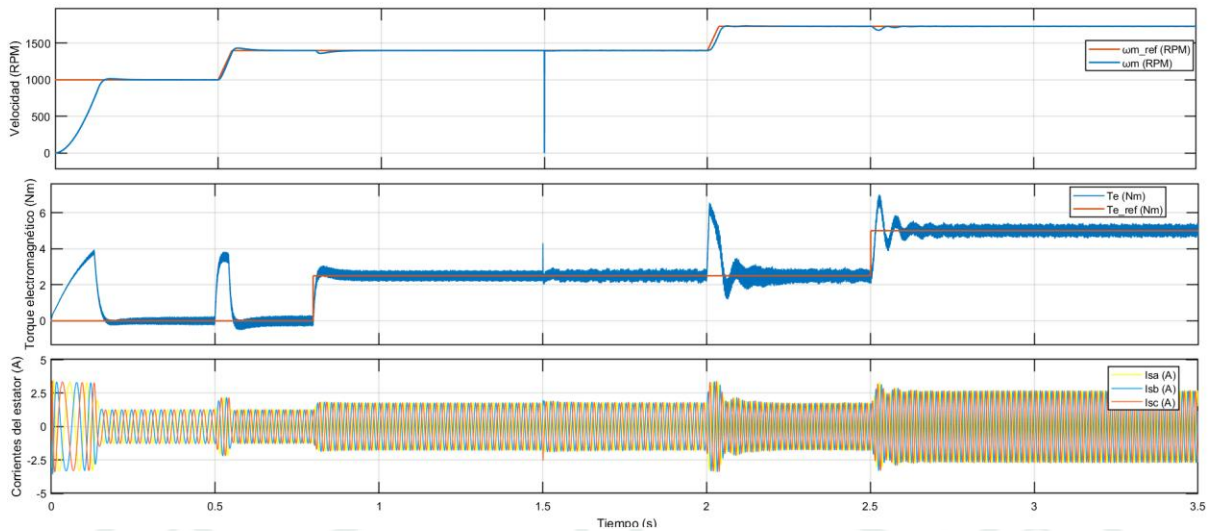


Figura 78. Respuesta global del sistema ante falla de circuito abierto en sensor de velocidad

En la Figura 79 se observa con mayor claridad el comportamiento de la velocidad real, la velocidad estimada y la referencia. Tras la falla, la velocidad medida se desploma, pero el estimador (observador) continúa entregando una estimación coherente, permitiendo que el controlador mantenga la operación. El algoritmo de detección de fallas (FDI) identifica el circuito abierto casi inmediatamente, y esto se refleja en la activación del flag de falla, el cual permanece en estado alto luego del segundo 1.5 s.

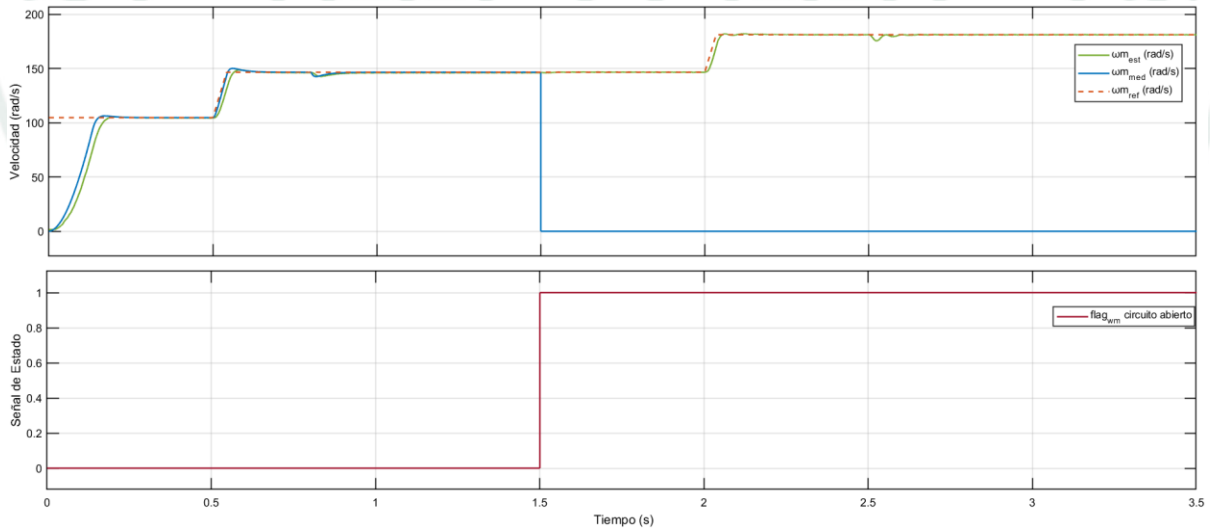


Figura 79. Velocidad real, estimada y referencia durante la falla, junto con activación del flag de detección

5.4.2. Falla por circuito abierto de corriente de fase i_{sa}

En esta prueba se introduce una falla de circuito abierto en el sensor de corriente de la fase i_{sa} en el instante $t = 1.2$ s. Antes de la falla, el sistema sigue correctamente la referencia

de velocidad, el torque se mantiene cercano a su consigna y las corrientes presentan la forma sinusoidal esperada. Cuando ocurre la apertura del sensor, la corriente medida deja de ser válida y el FOC experimenta una perturbación inmediata, reflejada en oscilaciones transitorias tanto en el torque como en las corrientes de fase. La Figura 80 muestra este comportamiento.

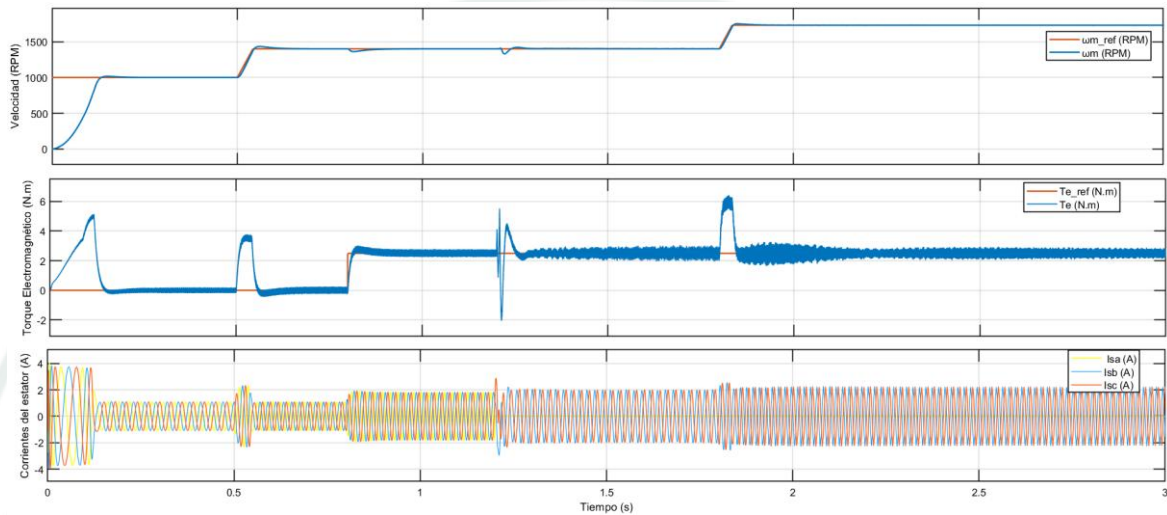


Figura 80. Respuesta global del FOC ante falla de circuito abierto en el sensor de corriente de fase i_{sa}

En la Figura 81 se aprecia con mayor detalle el momento de la falla: la corriente medida i_{sa} cae abruptamente mientras que la corriente estimada toma su lugar. El algoritmo de detección activa el flag de circuito abierto, permitiendo que el sistema reemplace automáticamente la señal medida por la estimación basada en el modelo del motor, lo que estabiliza nuevamente el control. Después de $t = 2$ s, cuando se incrementa la referencia de velocidad, la respuesta vuelve a ser normal gracias a la correcta reconfiguración.

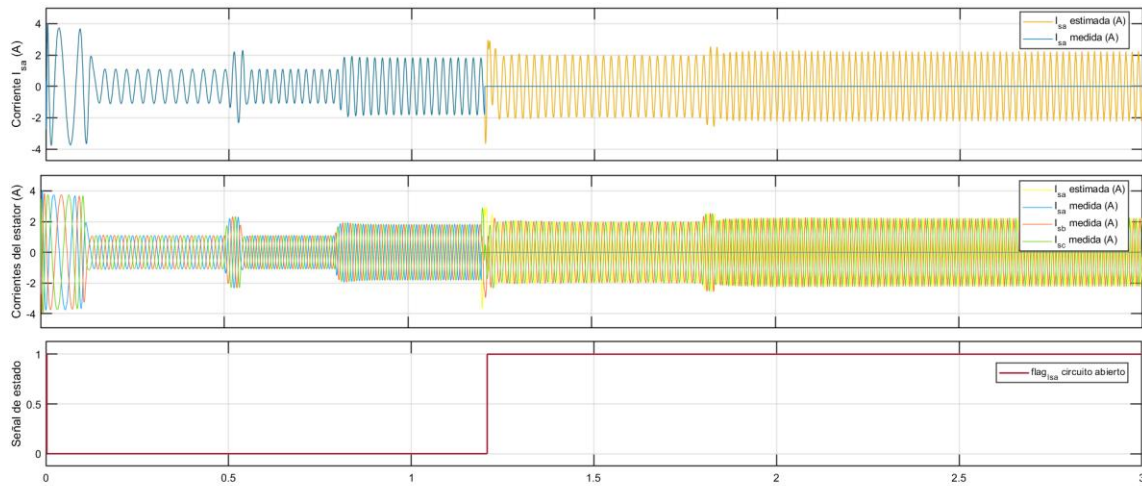


Figura 81. Corrientes medidas/estimadas y señal de diagnóstico para la falla de circuito abierto en i_{sa}

5.4.3. Falla de ganancia de corriente de fase i_{sc}

En esta prueba se introduce una falla de ganancia (*scaling fault*) en el sensor de corriente de la fase c (i_{sc}) en el instante $t = 1.5$ s. En la Figura 82 puede observarse que, antes de la falla, el sistema opera de manera estable, siguiendo la referencia de velocidad y manteniendo un torque suave y corrientes sinusoidales. Una vez que la falla ocurre, la señal medida de i_{sc} se amplifica artificialmente (en este caso, por un factor $\times 3$), lo que provoca una distorsión inmediata en las tres corrientes de fase y un aumento temporal en las oscilaciones del torque. Este efecto transitorio dura aproximadamente 0.2 s, hasta que la compensación basada en la corriente estimada entra en acción.

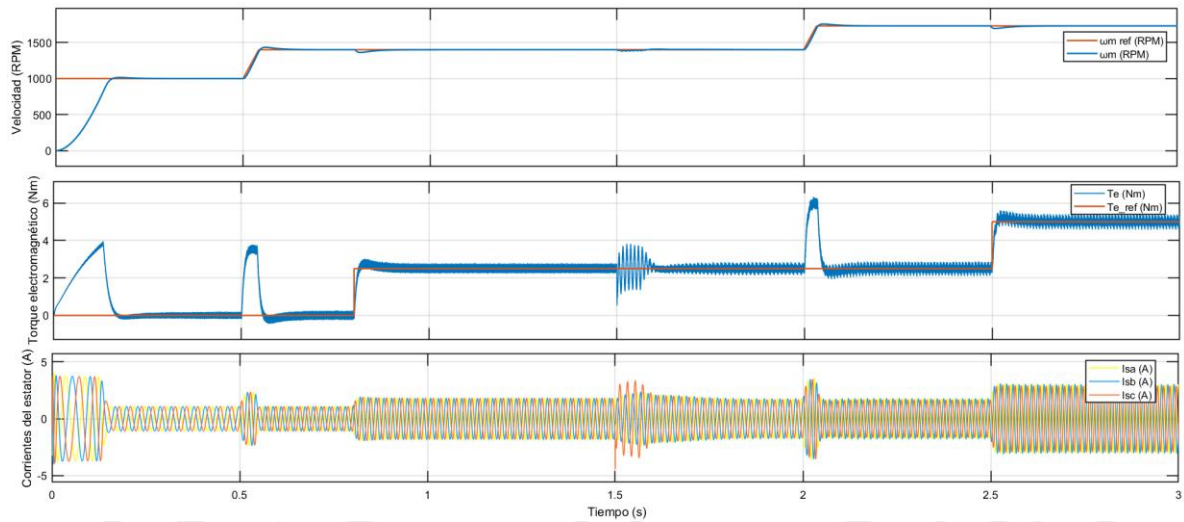


Figura 82. Respuesta del sistema FOC ante una falla de escalamiento en la corriente i_{sc}

En la Figura 83 se aprecia claramente la discrepancia entre la corriente medida y la corriente corregida, así como la coherencia entre las corrientes estimadas $\alpha\beta$ por el observador. Estas señales permiten identificar correctamente la anomalía, lo cual se confirma con la activación del flag de escalamiento exactamente en el instante en que se produce la falla.

A pesar de la perturbación inicial, la estabilidad del sistema no se ve comprometida. Esto se evidencia cuando, en $t = 2$ s, se incrementa la referencia de velocidad sin afectar el seguimiento, y posteriormente en $t = 2.5$ s, se realiza un incremento de torque manteniendo un comportamiento estable una vez aplicada la corrección.

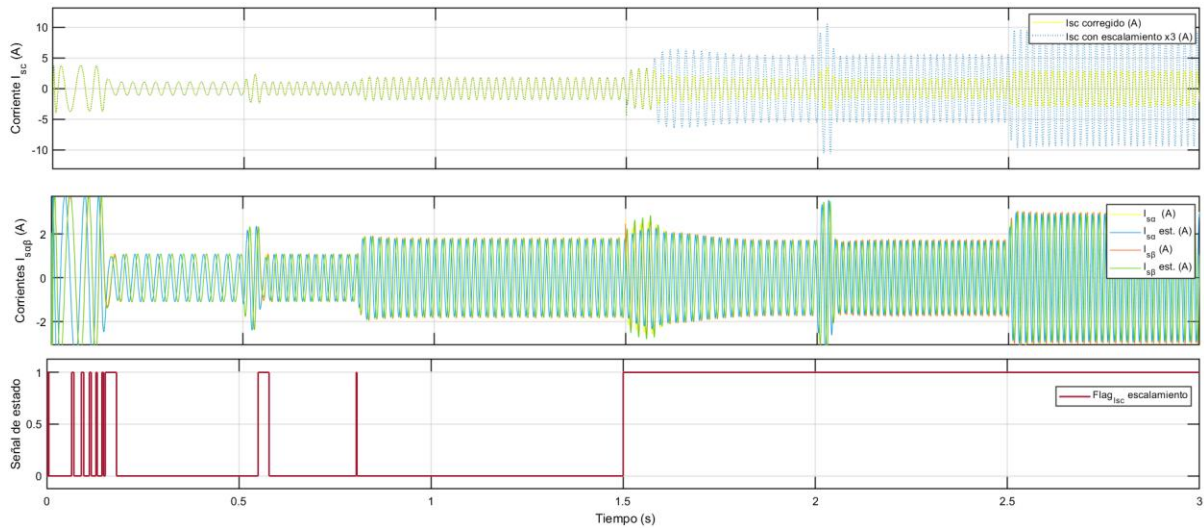


Figura 83. Corriente medida vs corriente corregida y activación del flag de falla por escalamiento

En la Figura 84 se muestra la respuesta dinámica del índice V_c . Luego que en $t = 1.5$ s que se introduce una falla de ganancia, el índice abandona la zona unitaria de operación normal y converge hacia el valor del error de amplificación.

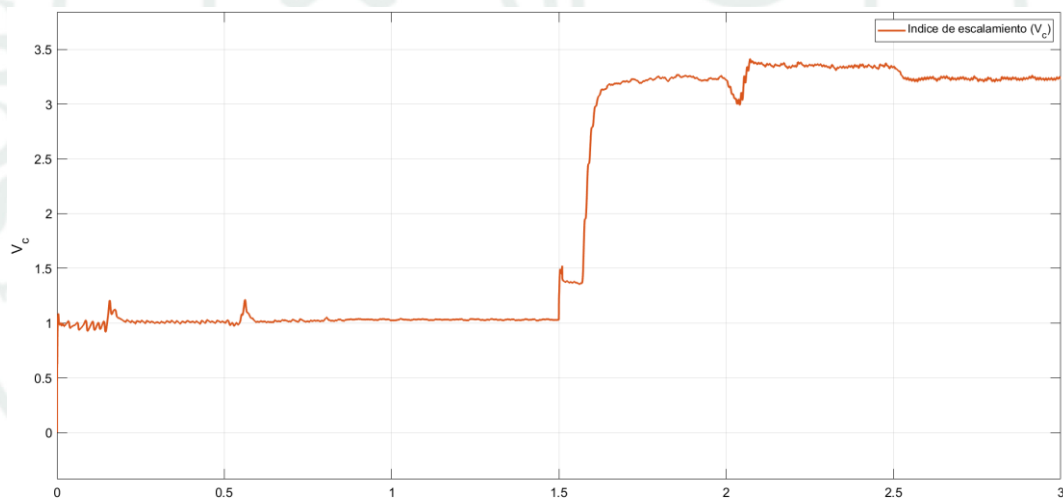


Figura 84. Comportamiento de Índice de escalamiento (V_c)

5.4.4. Falla por offset de corriente de fase i_{sa}

En esta prueba se introduce una falla de tipo offset en el sensor de corriente de la fase a (i_{sa}) en el instante $t = 1.2$ s. Antes de la falla, como se aprecia en la Figura 85, el sistema

opera con normalidad: la velocidad sigue la referencia, el torque permanece cercano a su valor de consigna y las corrientes presentan un comportamiento trifásico equilibrado.

Cuando se aplica el offset de 1 A a la corriente i_{sa} , la medición se desplaza respecto a su valor real, generando una perturbación inmediata en el torque y en las corrientes del estator. Esta distorsión se manifiesta como oscilaciones transitorias típicas de una componente DC no deseada en la realimentación, lo que rompe la simetría entre fases y altera momentáneamente el control. Aun así, el lazo de velocidad mantiene la estabilidad del sistema e impide que la perturbación crezca o derive en inestabilidad sostenida. El torque demora en estabilizarse aproximadamente 1 segundo mientras las corrientes requieren aproximadamente 0.5 segundo para volver a estabilizarse luego de la introducción del offset.

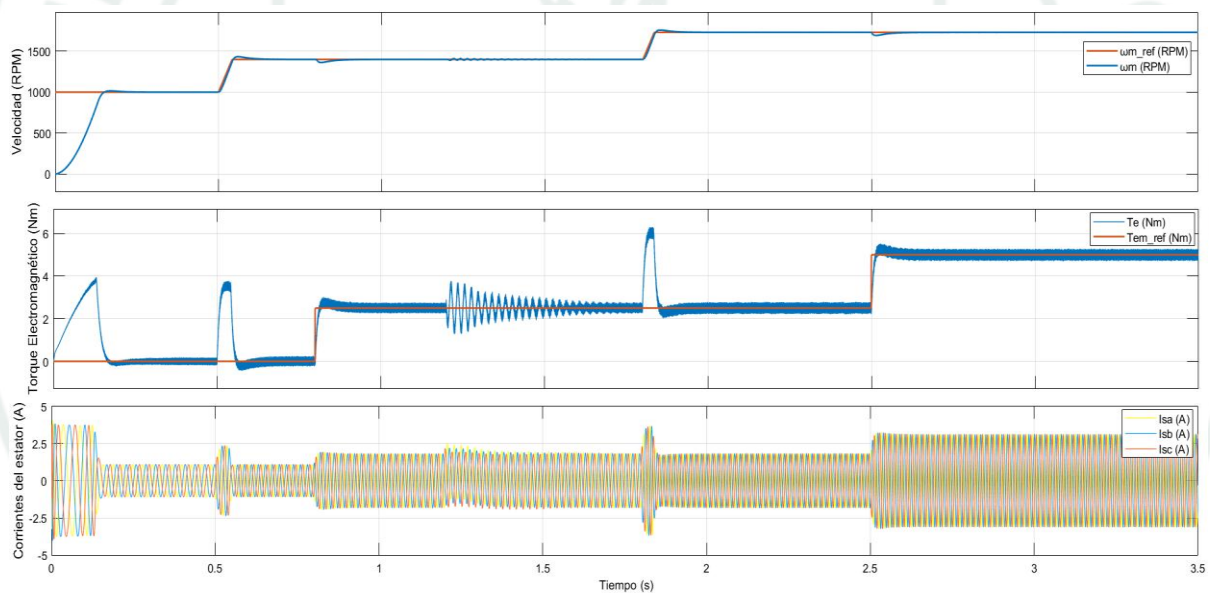


Figura 85. Respuesta del sistema ante una falla de offset en la corriente de fase i_{sa}

En la Figura 86 se aprecia claramente la diferencia entre la corriente medida con offset y la corriente corregida por el compensador basado en ventanas deslizantes. El algoritmo detecta el desvío y aplica la corrección correspondiente. Esto se confirma con la activación del flag de offset, que cambia de estado justo en el instante en que ocurre la falla, validando la detección adecuada. Una vez compensado el error, el sistema vuelve a operar con normalidad y permanece estable para los cambios posteriores de velocidad y torque.

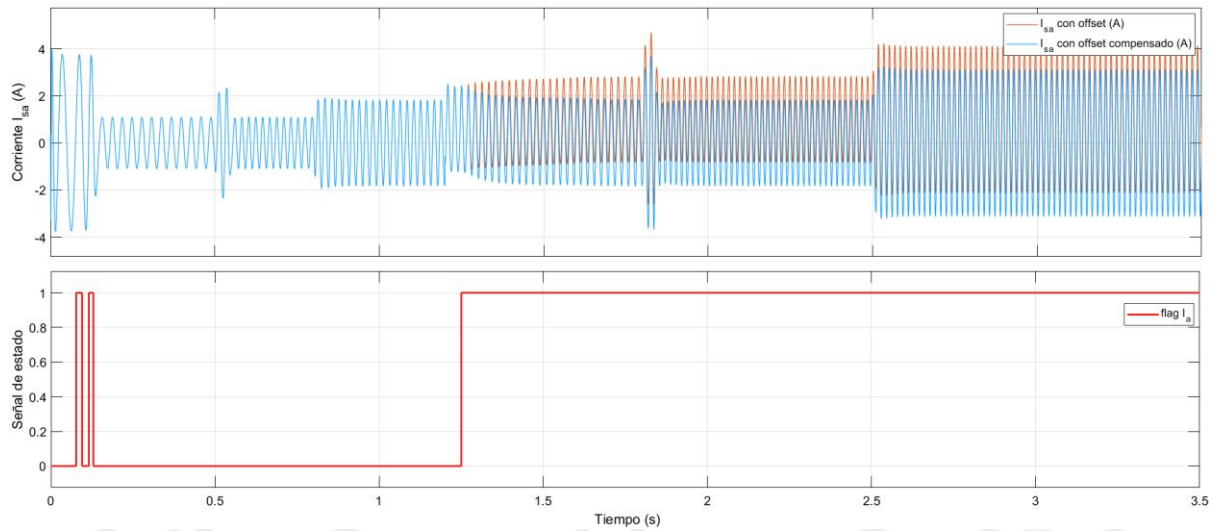


Figura 86. Detección y compensación del offset en la corriente de fase I_{sa} : señal corregida y activación del flag de falla

En la Figura 87 se observa una etapa inicial de operación sana ($t < 1.2$ s) donde la señal se mantiene aproximadamente en 0. Posteriormente, tras la inyección de la falla, la señal se incrementa y finalmente converge a 1, indicando la detección del offset.

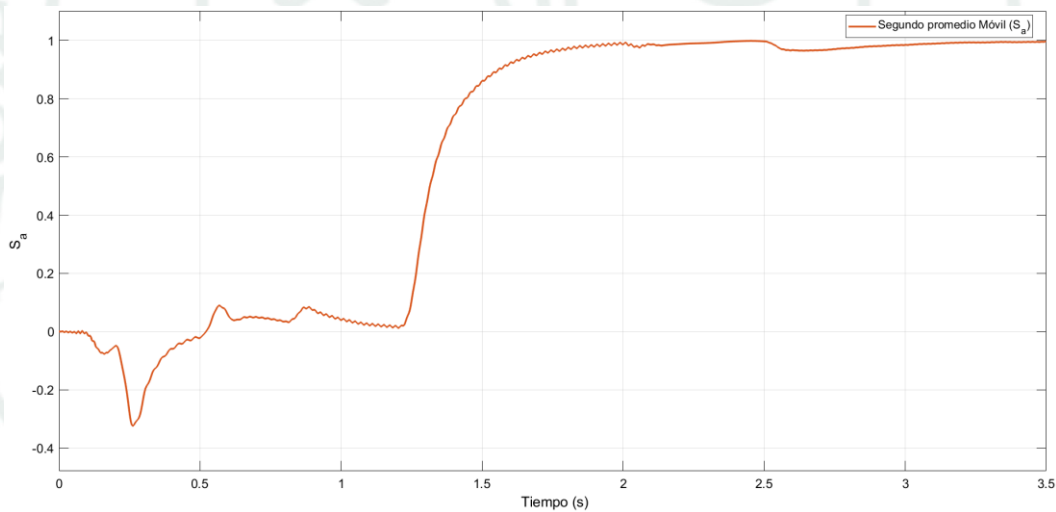


Figura 87. Comportamiento de segundo promedio móvil ($S_{i_{sa}}$)

Se evaluaron cuatro tipos de fallas en los sensores del sistema FOC con el fin de validar la capacidad del esquema de detección y reconfiguración. En la falla de circuito abierto en el sensor de velocidad, la señal medida cae abruptamente, generando un transitorio en torque y corrientes; sin embargo, la estimación de velocidad permite mantener la operación estable. En la falla de circuito abierto en una corriente de fase, la pérdida de una medición produce

oscilaciones momentáneas y distorsión en las corrientes, pero la sustitución por la corriente estimada restablece la simetría y la estabilidad del sistema.

En la falla tipo offset, la presencia de una componente DC provoca una perturbación visible en torque y corrientes, aunque el controlador logra recuperar el comportamiento nominal una vez que el compensador elimina el sesgo. Finalmente, en la falla de tipo scaling, donde la corriente medida sufre un cambio de ganancia, aparecen distorsiones transitorias que se atenúan tras la corrección basada en la estimación de las corrientes. En todos los casos, el control vectorial mantiene la estabilidad global y continúa siguiendo los cambios de referencia, demostrando una correcta capacidad de diagnóstico y reconfiguración.

Las simulaciones y códigos MATLAB/Simulink se muestran en el Anexo 6

CAPÍTULO VI: PROPUESTA DE IMPLEMENTACIÓN EXPERIMENTAL DE CONTROL VECTORIAL SENSORLESS

6.1. Introducción y Plataforma Experimental

En este capítulo se describe la implementación y evaluación experimental de un sistema de control vectorial sensorless (FOC) para un motor de inducción trifásico de 1 HP. La plataforma se basa en el microcontrolador STM32G431RB y el módulo de potencia STEVAL-IHM023V3. El objetivo es validar la respuesta dinámica, estabilidad y precisión en la estimación de velocidad.

El sistema de validación Figura 88, consta de un bus de corriente continua de 311 V y una frecuencia de conmutación de 16 kHz. El STM32 gestiona el algoritmo FOC, generando señales PWM y adquiriendo las corrientes de fase mediante resistencias shunt.

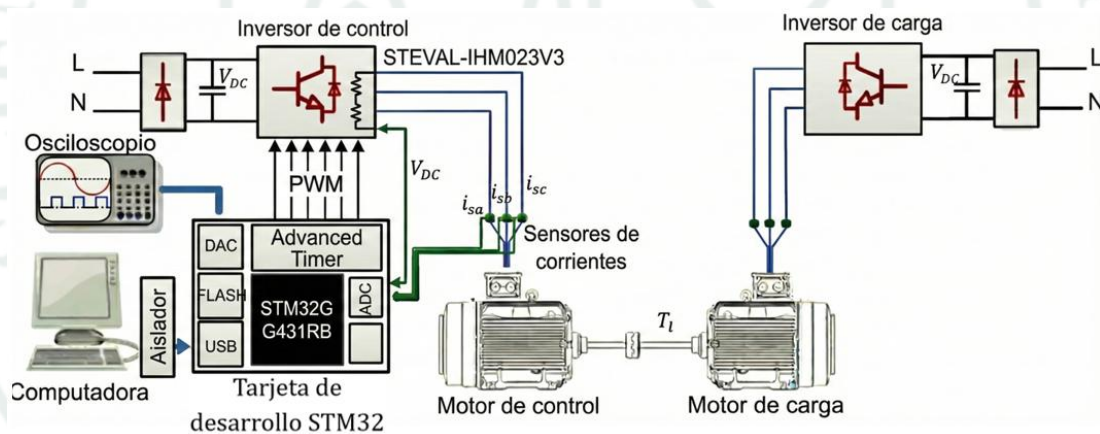


Figura 88. Banco de pruebas experimentales – Elaboración propia

6.2. Arquitectura de Control Digital

Estrategias modernas como el Control de Campo Orientado (FOC) exigen resolver complejos sistemas de ecuaciones en ventanas de tiempo críticas, típicamente inferiores a 50 μ s. Históricamente, esta demanda dividió el desarrollo en dos enfoques: los Procesadores

Digitales de Señales (DSP) para cálculo secuencial intensivo y las FPGAs para procesamiento paralelo masivo.

Frente a la complejidad de desarrollo y costos elevados del hardware dedicado, los microcontroladores modernos ofrecen ventaja de facilidad de programación (lenguaje C/C++) y bajo costo de implementación. Esta flexibilidad permite reducir significativamente los tiempos de desarrollo y simplificar la arquitectura del sistema

Por ello, se seleccionó el STM32G431RB, perteneciente a la categoría de Microcontroladores de Alto Desempeño. Este dispositivo cierra la brecha entre MCU y DSP al combinar la versatilidad de periféricos de comunicación con la potencia de cálculo requerida para el control vectorial. Específicamente, el STM32G4 integra una Unidad de Punto Flotante (FPU) y el acelerador matemático CORDIC, permitiendo una ejecución eficiente y precisa de los algoritmos FOC con bajo consumo energético y alta integración funcional.

6.2.1. Características del Núcleo ARM Cortex-M4

El Cortex-M4 es un núcleo RISC de 32 bits para microcontroladores, perteneciente al perfil Microcontroller de ARM (Advanced RISC Machine), se fundamenta en el principio RISC (Reduced Instruction Set Computing). Este enfoque prioriza un conjunto de instrucciones más pequeño y simple, lo que resulta en un diseño de hardware simplificado. y basado en la arquitectura Armv7-ME. Fue introducido en 2010 con un enfoque dual: control en tiempo real y procesamiento digital de señales (DSP) dentro de un mismo núcleo. Esta convergencia lo convierte en una plataforma idónea para sistemas embebidos que requieren determinismo, bajo consumo y cómputo matemático eficiente. Y es utilizado en STM32G431RB que pertenece a esta familia desarrollada ARM

Sus características principales para la implementación propuesta incluyen:

- **Procesamiento:** Núcleo RISC de 32 bits con instrucciones DSP y unidad de punto flotante (FPU) de precisión simple (IEEE-754). Esto acelera operaciones matemáticas complejas (filtros, transformadas, PID).
- **Eficiencia:** Ejecuta instrucciones MAC (Multiply-Accumulate) en un solo ciclo. Esta capacidad es crítica para la ejecución eficiente de filtros digitales y las ecuaciones matriciales.
- **Arquitectura:** Utiliza una arquitectura Harvard modificada con un *pipeline* de 3 etapas y buses separados para instrucciones y datos, optimizando el flujo de ejecución.

6.2.2. Arquitectura de Buses y Eficiencia de Ejecución

Para garantizar el flujo de datos en tiempo real, el STM32G4 implementa la arquitectura de interconexión estándar AMBA. Esta estructura organiza el sistema en buses jerárquicos para evitar cuellos de botella como se ve en la Figura 89:

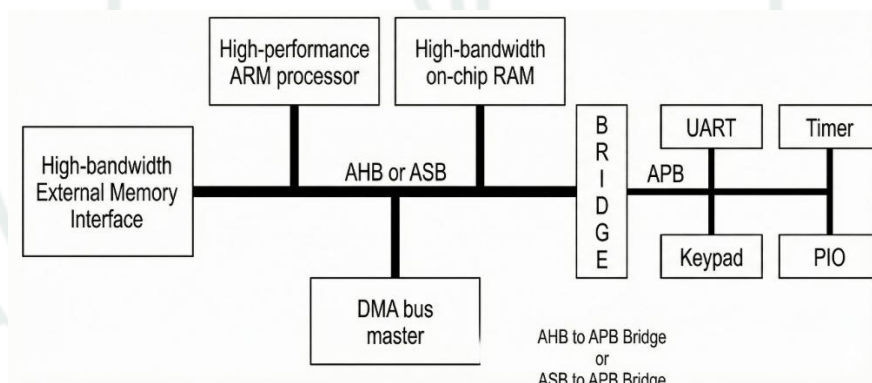


Figura 89. Arquitectura AMBA típica

Donde:

- **Bus AHB-Lite (High-performance):** Conecta los bloques de alta velocidad (CPU, Memoria y DMA), permitiendo el acceso a datos e instrucciones sin latencia.
- **Bus APB (Peripheral):** Gestiona los periféricos de control (Timers, ADC, GPIO) a menor velocidad, conectados al núcleo mediante un puente (Bridge) que aísla el tráfico de E/S del procesamiento computacional.

Adicionalmente, el núcleo maximiza su rendimiento mediante un pipeline (Figura 90) de 3 etapas (Fetch, Decode, Execute). Esta técnica permite solapar la lectura, decodificación y ejecución de instrucciones consecutivas, logrando completar una instrucción por ciclo de reloj en condiciones ideales. Esta capacidad de procesamiento continuo es crítica para mantener la frecuencia de muestreo de 16 kHz requerida por el algoritmo FOC.

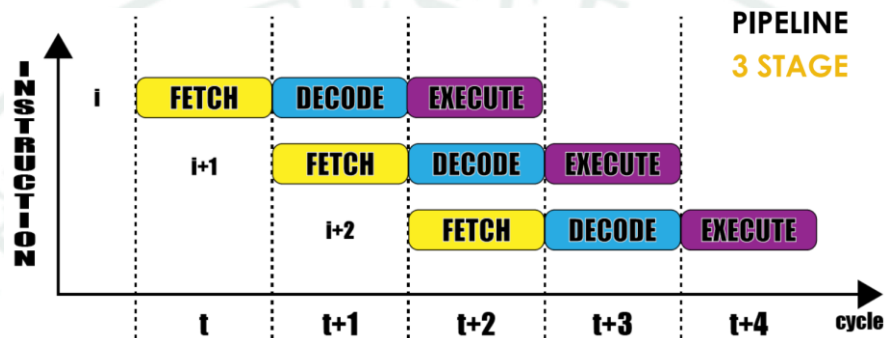


Figura 90. Pipeline de 3 etapas del Cortex-M4

6.2.3. Características Específicas del STM32G4 y Plataforma de Desarrollo

El MCU seleccionado, STM32G431RB, opera a una frecuencia de 170 MHz y destaca por integrar aceleradores de hardware específicos que descargan a la CPU de tareas matemáticas intensivas, optimizando el tiempo de bucle del algoritmo FOC:

- **CORDIC (Coordinate Rotation Digital Computer):** Acelera el cálculo de funciones trigonométricas (seno, coseno, arcotangente) mediante hardware. Siendo importante para las transformadas de Park y Clarke, reduciendo los ciclos de reloj necesarios frente a una implementación por software.
- **FMAC (Filter Math Accelerator):** Diseñado para ejecutar filtros digitales (FIR/IIR) en paralelo a la CPU.
- **CCM SRAM (Core Coupled Memory):** Dispone de 10 KB de memoria acoplada directamente al núcleo, ideal para alojar las rutinas críticas de control en tiempo real (ISR) y garantizar una ejecución determinista (0-wait-state).

Para la validación experimental se utilizó la tarjeta de desarrollo NUCLEO-G431RB. Esta plataforma integra el depurador/programador ST-LINK/V3E, lo que facilita la programación directa vía USB y el monitoreo de variables en tiempo real. La conexión con la etapa de potencia se realiza a través de los cabezales ST Morpho, los cuales exponen la totalidad de los pines del MCU, incluyendo los canales de los Temporizadores Avanzados (TIM1) y las entradas analógicas rápidas.

A continuación, se muestra tabla con características del STM32G431RB

Tabla 11. Características del STM32G431RB (STMicroelectronics, 2019)

Característica	Detalle Técnico
Núcleo	Arm® Cortex®-M4 @ 170 MHz con FPU, MPU y DSP instructions.
Memoria Flash	128 Kbytes (La variante 'B' indica 128KB).
Memoria SRAM	32 Kbytes totales. Distribuida en: 16 KB (SRAM1), 6 KB (SRAM2) y 10 KB CCM SRAM
Aceleradores	ART Accelerator, CORDIC (Trigonometría) y FMAC (Filtros).
ADC (Analógico)	2x ADCs de 12-bits (hasta 16-bits con oversampling). Tiempo de conversión de 0.25 μ s.
DAC (Analógico)	4x canales DAC de 12-bits (2 externos, 2 internos).
Timers (Motores)	2x Temporizadores de control avanzado de 16-bits (TIM1, TIM8) con capacidad de generación de PWM en 6 canales
Comunicaciones	3x I2C, 3x SPI, 3x USART, 1x UART, 1x LPUART, 1x FDCAN.
GPIOs	Hasta 86 pines, varios 5V-tolerant
Otros	Comparadores analógicos rápidos, Amplificadores Operacionales (OP-AMP) integrados

En la siguiente Figura 91, se presenta la distribución física de la NUCLEO-G431RB que se utilizará en este proyecto.

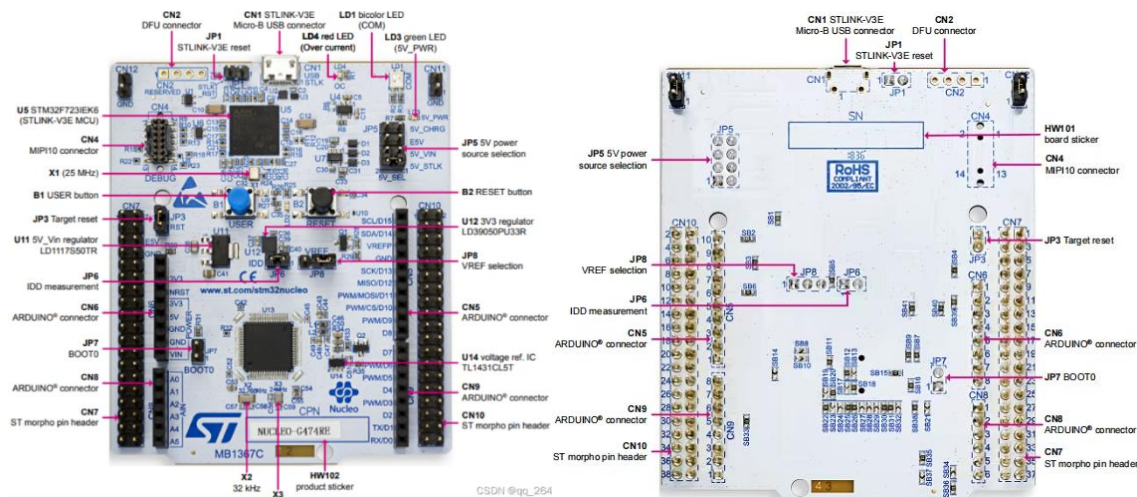


Figura 91. Placa de desarrollo Nucleo-G431RB

6.3. Etapa de potencia: STEVAL-IHM023V3

STEVAL-IHM023V3 es una placa de evaluación de control de motor trifásico de hasta 1 KW, desarrollada por STMicroelectronics como una plataforma completa para la validación, experimentación y optimización de algoritmos avanzados de control de motores eléctricos. Esta placa integra una etapa de potencia tipo inversor trifásico AC–DC totalmente poblada y preparada para funcionar directamente desde la red eléctrica, permitiendo el accionamiento de motores PMSM, BLDC y motores de inducción (ACIM) (STMicroelectronics. 2014).

Su diseño modular y flexible la convierte en un equipo adecuado para prototipado rápido, pruebas de laboratorio, validación de hardware y benchmarking de técnicas de control como FOC (Field Oriented Control), control escalar (six-step), modulación SVPWM.

La placa está diseñada para trabajar en conjunto con controladores basados en microcontroladores STM32, conectándose directamente mediante el conector estándar de 34 pines del ecosistema ST Motor Control.

En la Figura 92 se muestra el STEVAL-IHM023V3



Figura 92. STEVAL-IHM023V3 (STMicroelectronics., 2017)

A continuación, se muestra tabla de características de STEVAL-IHM023V3

Tabla 12. Características de STEVAL-IHM023V3

Categoría	Característica	Descripción / Valor
General	Tipo de sistema	Placa de evaluación de inversor trifásico AC-DC (1 kW)
	Motores soportados	PMSM, BLDC y ACIM (2 ó 3 fases, sensed o sensorless)
	Algoritmos compatibles	FOC, six-step, V/F, SVPWM
Potencia y Alimentación	Potencia máxima	Hasta 1 kW
	Entrada AC	90 – 285 VAC (monofásico)
	Entrada DC	125 – 400 VDC
	Modo de doblador de voltaje	65 – 145 VAC
	Bajo voltaje (LV DC)	18 – 35 VDC (requiere reconfiguración)
	Frecuencia de conmutación	Hasta 50 kHz
	Límite de sobrecorriente	7 A (por hardware, configurable)
	Funciones adicionales	Freno regenerativo, limitación de corriente de irrupción, relay bypass
Etapas de Potencia	Interruptores	IGBT STGP10H60DF – 600 V, 10 A
	Encapsulado	TO-220, intercambiables por MOSFET/IGBT compatibles
	Disipación térmica	IGBTs montados sobre disipador metálico
Drivers y Sensado	Driver de compuerta	L6390 high & low-side gate driver
	Protecciones integradas	Sobrecorriente (OCP), sobretensión, shoot-through
	Sensado de corriente	Configurable: single-shunt (bus DC) o three-shunt (fase-fase)
	Acondicionamiento analógico	Amplificador operacional integrado en L6390
	Medición térmica	NTC 10 kΩ para disipador

Conectividad y Control	Conector principal	Conector ST de 34 pines (PWM, ADC, protección, tachómetro, Hall, encoder)
	Compatibilidad MCU	Cualquier MCU STM32 soportado por ST MCSDK (ej. G4, F3, F4)
	Placa hija	Compatible con BEMF daughterboard para six-step sensorless
Software y Ecosistema	SDK compatible	STM32 Motor Control SDK (MCSDK)
	Configuration tool	ST Motor Control Workbench
	Plataforma firmware	ST MC Firmware Framework (FOC, velocidad, torque, OCP, FW)

6.4. Placa de expansión X-NUCLEO-IHM09M1

X-NUCLEO-IHM09M1 es una placa de expansión de control de motores para placas STM32 Nucleo, diseñado para facilitar la evaluación y el desarrollo de soluciones de control de motores trifásicos. Permite interconectar la Nucleo con placas de potencia externas de ST mediante el conector ST Morpho y un conector de control de 34 pines, compatible con las principales tarjetas de potencia de ST que requieren una MCU externa.

Además, integra un conector DAC que simplifica la depuración y las pruebas al ofrecer acceso directo a periféricos de la MCU, acelerando el desarrollo y la validación del firmware de control. En conjunto, es una plataforma práctica para prototipado rápido de aplicaciones de control de motores trifásicos con ecosistema STM32.

La Figura 93 muestra la placa muestra X-NUCLEO-IHM09M1



Figura 93. X-NUCLEO-IHM09M1 (STMicroelectronics)

6.5. Configuración de la Modulación Vectorial y Adquisición de Señales

La correcta ejecución del algoritmo FOC depende de la precisa sincronización entre la generación de voltaje y la medición de corriente. Para ello, se utilizan los Temporizadores Avanzados (TIM1) vinculados por hardware con los convertidores ADC.

6.5.1. Modulación en Modo Alineado al Centro

A diferencia de la alineación al borde ("diente de sierra"), la configuración de alineación central utiliza un esquema de conteo ascendente y descendente (Up/Down Counting). En este modo, el contador evoluciona desde 0 hasta el valor de Auto-Recarga (ARR) y regresa a 0, generando una portadora triangular simétrica.

Esta simetría es crítica para el control de motores, ya que reparte los flancos de conmutación alrededor del centro del periodo. Esto duplica la frecuencia efectiva de la ondulación de corriente, reduciendo los armónicos y la interferencia electromagnética (EMI).

Para maximizar la resolución del PWM, el pre-escalador se ajusta al mínimo (PSC=0), permitiendo que el contador opere a la frecuencia nativa del núcleo (f_{ck_CNT}).

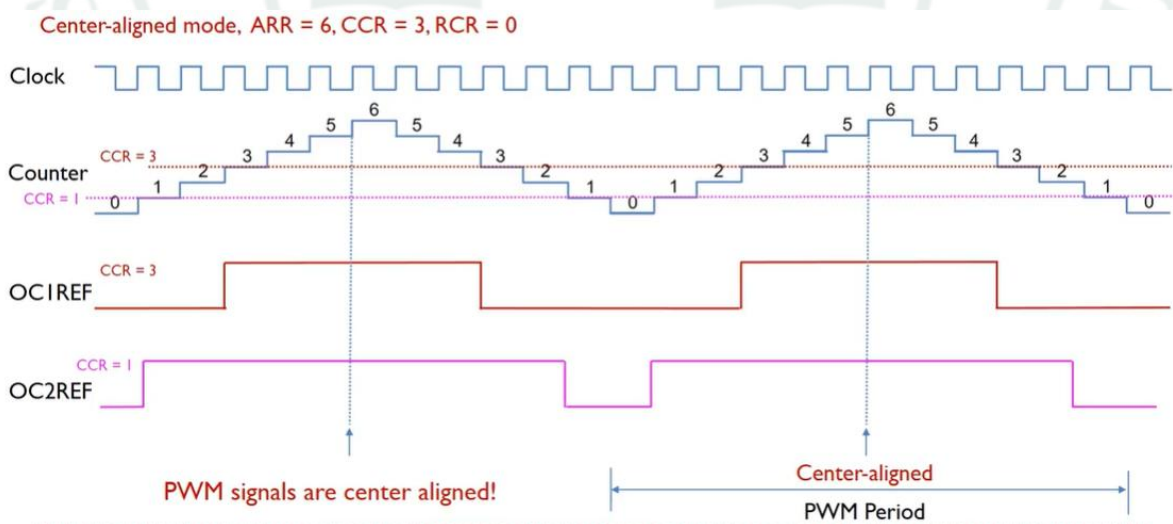


Figura 94. Señales en Modo Alineado al Centro (Zhu, 2023)

Como se observa en la Figura 94 el periodo PWM (T_{PWM}) consta de dos rampas. Por consiguiente, el valor del registro ARR se calcula considerando la mitad de la frecuencia de conmutación deseada (f_{PWM}):

$$ARR = \frac{f_{CK_CNT}}{2 * f_{PWM}}$$

El ciclo de trabajo (Duty Cycle) se actualiza dinámicamente modificando el registro de comparación (CCR), el cual define el instante de conmutación:

$$Duty\ Cycle = \frac{CCR}{ARR} \times 100\%$$

Es necesario habilitar los registros de precarga (*Preload Registers*) para garantizar que cualquier escritura en ARR o CCR sea atómica y se aplique únicamente al finalizar el periodo (Evento de Actualización), evitando transiciones erráticas.

6.5.2. Inserción de Tiempo Muerto (Dead Time)

Para proteger el inversor trifásico contra cortocircuitos de brazo (*shoot-through*), es imperativo insertar un retardo entre el apagado del transistor superior y el encendido del inferior. El hardware del TIM1 genera automáticamente este tiempo muerto programable en las salidas complementarias (OCx y OCxN).

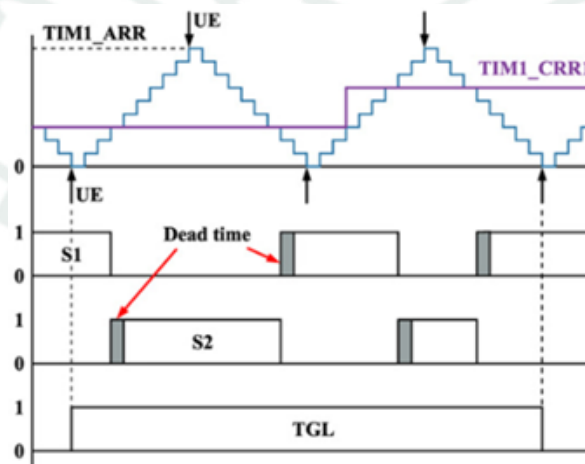


Figura 95. Inserción de tiempos muertos en señales complementarias.

Como se observa en la Figura 95, el hardware retrasa la activación de las señales (áreas sombreadas), asegurando que ambos interruptores permanezcan abiertos durante la transición crítica.

6.5.3. Sincronización del ADC

La medición de corriente se realiza estrictamente en el instante de menor ruido de conmutación, el cual coincide con los extremos del contador (0 y ARR). Para lograrlo, se configuró el Contador de Repetición ($RCR = 1$), generando una señal de disparo (TRGO) hacia el ADC una sola vez por periodo PWM completo. Esto garantiza una adquisición de datos estable y perfectamente centrada en el pulso PWM.

6.6. Configuración de Interfaces Físicas y Conversión Analógica

La interacción entre el microcontrolador y la etapa de potencia se gestiona mediante la configuración específica de los puertos de entrada/salida (GPIO) y los convertidores analógicos.

Configuración de GPIO. La arquitectura de E/S del STM32G4 permite asignar funciones específicas a cada pin mediante multiplexación. Para esta implementación se utilizaron dos modos operativos principales:

- **Modo Función Alterna (Alternate Function):** Los pines asignados a las fases del inversor (PWM) se desconectan del registro de salida estándar y ceden el control directo al Temporizador Avanzado (TIM1). Se configura la etapa de salida en modo Push-Pull, garantizando una conmutación rápida y robusta (altos y bajos fuertes) necesaria para los drivers de compuerta.
- **Modo Analógico (Analog Mode):** Los pines conectados a los sensores de corriente se configuran en modo analógico. Esta configuración deshabilita el disparador Schmitt (Schmitt Trigger) y la etapa digital de entrada, reduciendo el consumo de energía y

eliminando el ruido de conmutación digital que podría acoplarse a la señal analógica pura que ingresa al ADC.

Especificaciones y Configuración del ADC

Para la digitalización de las corrientes, se emplean los convertidores del STM32G431RB, los cuales ofrecen una resolución nativa de 12 bits y una velocidad de conversión de 4 Msamples/s. Esta alta velocidad es importante para ejecutar lazos de corriente de alto ancho de banda.

Adicionalmente, se habilita la unidad de Oversampling por Hardware. Esta característica permite acumular y promediar automáticamente múltiples muestras (ratio configurable de 2x a 256x) antes de enviar el resultado a la memoria. En esta implementación, el oversampling actúa como un filtro pasabajos digital, incrementando la resolución efectiva de la medición y mejorando la relación señal-ruido (SNR) sin aumentar la carga de procesamiento de la CPU.

6.7. Entorno de Desarrollo y Flujo de Trabajo

Para agilizar la implementación del algoritmo FOC, se utiliza el ecosistema de desarrollo de STMicroelectronics, diseñado específicamente para optimizar el control de motores eléctricos. Este entorno integra herramientas gráficas y librerías especializadas que permiten configurar, generar código y depurar el sistema de manera eficiente, reduciendo la complejidad de la programación a bajo nivel.

El flujo de trabajo se estructura en etapas sucesivas. Inicialmente, STM32 MC Workbench permite definir los parámetros electromecánicos del motor, la topología de potencia y las estrategias de control, configurando las ganancias iniciales de los controladores PI y configurando el observador de estado. Esta configuración se exporta a STM32CubeMX, el cual gestiona la inicialización de los periféricos críticos (ADC, Timers, DMA) y la configuración

del árbol de reloj, generando el código base en lenguaje C. Finalmente, la compilación y depuración del firmware se realizan en STM32CubeIDE, mientras que la validación dinámica se apoya en MC Motor Pilot, herramienta que facilita el monitoreo de variables y el ajuste de parámetros (tuning) en tiempo real sin detener el accionamiento.

6.8. Estrategia de Control vectorial Sensorless

Si bien el EKF fue analizado y validado en simulación como estimador de velocidad para el control vectorial sensorless, su implementación directa en hardware de tiempo real presenta limitaciones prácticas que justifican la adopción de una alternativa determinista para la etapa experimental.

El EKF requiere, en cada ciclo de muestreo, la actualización de matrices de covarianza de dimensión 5×5 (matrices P, Q, F y K), lo que implica operaciones de multiplicación matricial, inversión y transposición en punto flotante, es por eso como alternativa se utiliza el observador de Luenberger adaptativo (LSO), que opera sobre el mismo modelo en espacio de estados pero prescinde del cálculo iterativo de covarianzas. Su corrección se basa en una ganancia fija K calculada fuera de línea mediante asignación de polos, reduciendo la carga computacional por ciclo a un conjunto de multiplicaciones y sumas vectoriales de dimensión 4×1 , ejecutables en menos de 5 μ s en el STM32G431RB con FPU habilitada.

Para la implementación se sigue el procedimiento detallado en SDK AC induction motor (STMicroelectronics, 2020)

Dado que el motor de inducción carece de flujo de rotor fijo, la implementación prescinde de sensores mecánicos utilizando un Observador de Estado de Luenberger (LSO). Este algoritmo reconstruye el vector de flujo del rotor (ψ_r) y estima la velocidad mecánica basándose en el modelo matemático del motor y las mediciones de voltaje y corriente del estator.

El diagrama de bloques de control para ser implementar se presenta en la siguiente Figura 96:

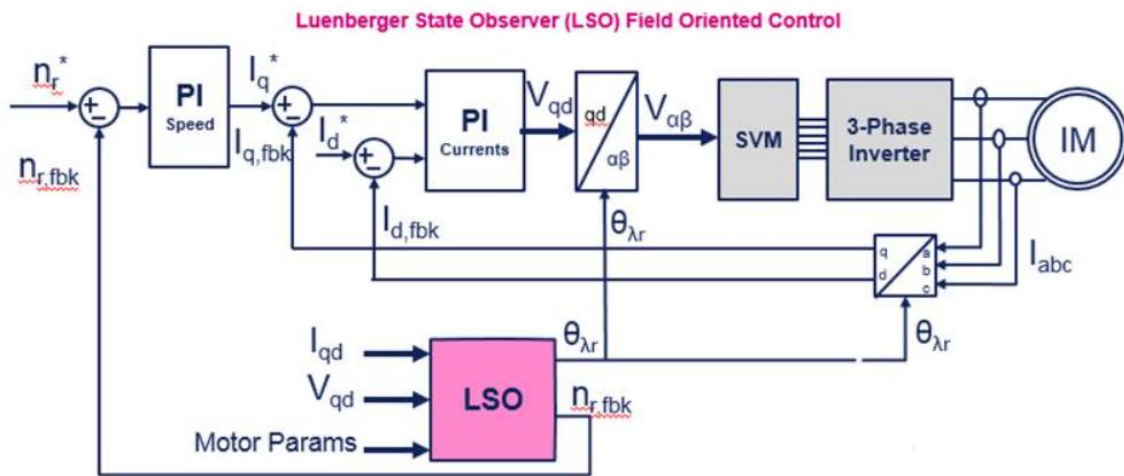


Figura 96. Sensorless de Control de campo orientado usando observador de estado Luenberger (STMicroelectronics, 2020)

Como se observa en el diagrama, el sistema opera en cascada:

Lazo Externo (Velocidad): El controlador PI de velocidad compara la velocidad de referencia (n_r^*) con la estimada por el observador ($n_{r,fbk}$), generando la referencia de corriente de par (I_q^*).

Lazo Interno (Corriente): Los controladores PI Currents regulan las componentes I_d e I_q en el marco de referencia síncrono, generando los voltajes V_{dq} que alimentan al módulo de Modulación Vectorial (SVM).

Observador (LSO): El bloque LSO recibe los voltajes aplicados (V_{dq}) y las corrientes medidas (I_{dq}) calculando el ángulo de transformación ($\theta_{\lambda r}$) necesario para las transformadas de Park y la velocidad de retroalimentación.

Es importante notar que la precisión de este esquema depende críticamente de la correcta identificación de los parámetros eléctricos (R_s, R_r, L_s, L_m), los cuales alimentan al modelo interno del bloque LSO.

6.8.1. Observador de Luenberger adaptativo

El observador de Luenberger adaptativo es un método mejorado de estimación de velocidad que opera como un observador de flujo adaptativo en lazo cerrado de velocidad. Aunque un estimador puede ser implementado tanto en lazo abierto como en lazo cerrado, la principal desventaja de los estimadores en lazo abierto es el efecto de las desviaciones de parámetros a baja velocidad sobre el desempeño del accionamiento, tanto en régimen permanente como transitorio. El observador de Luenberger presentado en este trabajo es un observador determinista, aplicable a un sistema determinista lineal e invariante en el tiempo.

El observador se construye añadiendo un compensador al motor de inducción en el marco de referencia estacionario (Tahar, Bousmaha, Ismail, & Houcine, 2018).

Muchos de los sistemas actuales pueden modelarse mediante una descripción en variables de estado como:

$$\frac{dX}{dt} = AX + BV_s$$

donde:

$$X = [i_{s\alpha} \quad i_{s\beta} \quad \psi'_{r\alpha} \quad \psi'_{r\beta}]^T, V_s = [v_{s\alpha} \quad v_{s\beta}]^T$$

La matriz del sistema A contiene la velocidad del rotor ω_r . Si el valor de ω_r es conocido, los flujos y corrientes pueden calcularse directamente; sin embargo, si es incorrecto, surge una desviación entre los estados estimados y los reales.

6.8.2. Estructura y Dinámica del Observador

El objetivo del observador es minimizar el error entre las corrientes estimadas y las corrientes reales medidas. Este error se inyecta al sistema mediante una matriz de ganancia K, forzando a que la diferencia de estados tienda a cero.

La ecuación dinámica del observador es:

$$\frac{d\hat{X}}{dt} = \hat{A}\hat{X} + BV_s + K(Y - \hat{Y}) \quad (6.1)$$

$$\hat{Y} = C\hat{X}$$

donde:

$$\hat{A} = \begin{bmatrix} -\left(\frac{1}{\sigma T_s} + \frac{1-\sigma}{\sigma T_r}\right) & 0 & \frac{1-\sigma}{\sigma L_m T_r} & \frac{1-\sigma}{\sigma L_m} \hat{\omega}_r \\ 0 & -\left(\frac{1}{\sigma T_s} + \frac{1-\sigma}{\sigma T_r}\right) & -\frac{1-\sigma}{\sigma L_m} \hat{\omega}_r & \frac{1-\sigma}{\sigma L_m T_r} \\ \frac{L_m}{T_r} & 0 & -\frac{1}{T_r} & -\hat{\omega}_r \\ 0 & \frac{L_m}{T_r} & \hat{\omega}_r & -\frac{1}{T_r} \end{bmatrix}$$

$$\hat{X} = [\hat{i}_{s\alpha} \quad \hat{i}_{s\beta} \quad \hat{\psi}'_{r\alpha} \quad \hat{\psi}'_{r\beta}]^T \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad \hat{Y} = [\hat{i}_{s\alpha} \quad \hat{i}_{s\beta}]^T$$

La matriz K es fundamental para la corrección del error. Se estructura para corregir tanto las corrientes como los flujos mediante términos cruzados complejos (Cherifi & Miloud, 2024).

$$K = \begin{bmatrix} k_1 & -k_2 \\ k_2 & k_1 \\ k_3 & -k_4 \\ k_4 & k_3 \end{bmatrix} \quad (6.2)$$

Los términos k_1 y k_2 actúan sobre la corrección de corrientes, mientras que k_3 y k_4 actúan sobre la corrección de flujos. El error corriente complejo $e_{is\alpha} + je_{is\beta}$

Corrección de corrientes

$$(k_1 + jk_2)(e_{is\alpha} + je_{is\beta}) = (k_1 e_{is\alpha} - k_2 e_{is\beta}) + j(k_2 e_{is\alpha} + k_1 e_{is\beta})$$

Corrección de flujos

$$(k_3 + jk_4)(e_{\psi'_{r\alpha}} + je_{\psi'_{r\beta}}) = (k_3 e_{\psi'_{r\alpha}} - k_4 e_{\psi'_{r\beta}}) + j(k_4 e_{\psi'_{r\alpha}} + k_3 e_{\psi'_{r\beta}})$$

6.8.3. Estimación de Velocidad y Estabilidad (Lyapunov)

Para estimar la velocidad desconocida ω_r , se utiliza un mecanismo de adaptación basado en la teoría de estabilidad de Lyapunov (Anexo 7).

El observador también proporciona una estimación del vector de flujo, como se muestra en la Figura 97. El observador de flujo adaptativo en velocidad permite la estimación de la velocidad desconocida ω_r .

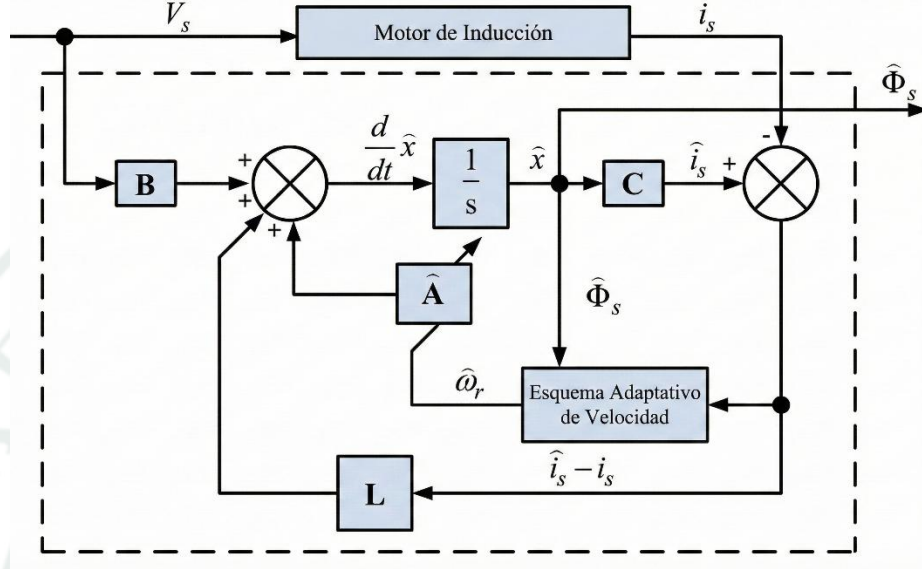


Figura 97. Diagrama de bloques del observador adaptativo .

Para derivar el algoritmo de adaptación de velocidad se emplea el teorema de Lyapunov.

En general, el error de estimación entre las corrientes del estator y los flujos del rotor puede describirse mediante:

$$\begin{aligned} \frac{de}{dt} &= [AX + BV_s] - [\hat{A}\hat{X} + BV_s + KC(X - \hat{X})] \\ \frac{de}{dt} &= A(e + \hat{X}) - \hat{A}\hat{X} - KCe \rightarrow \frac{de}{dt} = (A - KC)e + (A - \hat{A})\hat{X} \\ \frac{de}{dt} &= (A - KC)e + \Delta A\hat{X} \end{aligned} \quad (6.3)$$

con:

$$e = X - \hat{X} \rightarrow \frac{de}{dt} = \frac{dX}{dt} - \frac{d\hat{X}}{dt} \quad (6.4)$$

y:

$$\Delta A = A(\omega_r) - \hat{A}(\hat{\omega}_r) = \begin{bmatrix} 0 & 0 & 0 & \frac{1-\sigma}{\sigma L_m} \Delta \omega_r \\ 0 & 0 & -\frac{1-\sigma}{\sigma L_m} \Delta \omega_r & 0 \\ 0 & 0 & 0 & -\Delta \omega_r \\ 0 & 0 & \Delta \omega_r & 0 \end{bmatrix} \quad (6.5)$$

Donde

$$\Delta\omega_r = \omega_r - \hat{\omega}_r \quad (6.6)$$

$$e = X - \hat{X} = [e_{is\alpha} \quad e_{is\beta} \quad e_{\psi r\alpha} \quad e_{\psi r\beta}]^T \quad (6.7)$$

$$e = X - \hat{X} = [(i_{s\alpha} - \hat{i}_{s\alpha}) \quad (i_{s\beta} - \hat{i}_{s\beta}) \quad (\psi'_{r\alpha} - \hat{\psi}'_{r\alpha}) \quad (\psi'_{r\beta} - \hat{\psi}'_{r\beta})]^T \quad (6.8)$$

Se define la siguiente función candidata de Lyapunov, con el objetivo de que se cumpla que $\dot{V} < 0$, así el sistema será estable

$$V = e^T e + \frac{(\omega_r - \hat{\omega}_r)^2}{\gamma} = e^T e + \frac{(\Delta\omega_r)^2}{\gamma} \quad (6.9)$$

donde γ es una constante positiva. Derivando respecto al tiempo se obtiene:

$$\frac{dV}{dt} = \left[\frac{d(e^T)}{dt} \right] e + e^T \left[\frac{d(e)}{dt} \right] + \frac{1}{\gamma} 2\Delta\omega_r \frac{d}{dt} (\omega_r - \hat{\omega}_r) \quad (6.10)$$

Dado que asume ω_r es constante ($\dot{\omega}_r = 0$), entonces:

$$\frac{dV}{dt} = \left[\frac{d(e^T)}{dt} \right] e + e^T \left[\frac{d(e)}{dt} \right] - \frac{2}{\gamma} \Delta\omega_r \frac{d}{dt} (\hat{\omega}_r) \quad (6.11)$$

Reemplazando (6.3) en (6.12)

$$\frac{dV}{dt} = [(A - KC)e + \Delta A \hat{X}]^T e + e^T [(A - KC)e + \Delta A \hat{X}] - \frac{2}{\gamma} \Delta\omega_r \frac{d}{dt} (\hat{\omega}_r)$$

Resolviendo

$$\frac{dV}{dt} = e^T (A - KC)^T e + \hat{X}^T \Delta A^T e + e^T (A - KC)e + e^T \Delta A \hat{X} - \frac{2}{\gamma} \Delta\omega_r \frac{d}{dt} (\hat{\omega}_r)$$

$\hat{X}^T \Delta A^T e$ y $e^T \Delta A \hat{X}$ son escalares y traspuestos entre sí, por lo que se suman

$$\frac{dV}{dt} = e^T (A - KC)^T e + e^T (A - KC)e + 2e^T \Delta A \hat{X} - \frac{2}{\gamma} \Delta\omega_r \frac{d}{dt} (\hat{\omega}_r) \quad (6.12)$$

Reemplazando $2e^T \Delta A \hat{X}$ usando la ecuación (6.5) y (6.8) se obtiene:

$$\frac{dV}{dt} = \underbrace{e^T [(A - KC)^T + (A - KC)] e}_{\text{Estabilidad del observador}} - \underbrace{\frac{2\Delta\omega_r}{\gamma} (e_{is\alpha} \hat{\psi}'_{r\beta} - e_{is\beta} \hat{\psi}'_{r\alpha})}_{\text{Termino de adaptación de velocidad}} - \frac{2\Delta\omega_r}{\gamma} \frac{d\hat{\omega}_r}{dt} \quad (6.13)$$

donde: $c = \frac{\sigma L_m}{1-\sigma}$ y $\gamma > 0$ es una constante de adaptación

El primer término depende de la matriz K , si la ganancia del observador K se selecciona de tal forma que el primer término de (6.13) sea definido negativo, entonces el observador de flujo adaptativo de velocidad será estable y el segundo término contiene el error de velocidad $\Delta\omega_r$. No se sabe cuánto vale ese error. Si este bloque se vuelve positivo, la energía crecería y el observador fallaría (Tahar, Bousmaha, Ismail, & Houcine, 2018).

Para garantizar la estabilidad (Lyapunov), se fuerza a que el segundo termino sea igual a cero. De esa forma, solo queda el primer término (que es negativo) y la estabilidad está asegurada.

Por lo tanto, al anular el término dependiente del error de velocidad en la derivada de la función de Lyapunov, se obtiene la siguiente ley de adaptación para la estimación de la velocidad del rotor:

$$\frac{d\hat{\omega}_r}{dt} = \frac{\gamma}{c} (e_{i_{s\alpha}} \hat{\psi}'_{r\beta} - e_{i_{s\beta}} \hat{\psi}'_{r\alpha}) \quad (6.14)$$

En su forma integral, la estimación de velocidad se expresa como:

$$\hat{\omega}_r = \int k_i (e_{i_{s\alpha}} \hat{\psi}'_{r\beta} - e_{i_{s\beta}} \hat{\psi}'_{r\alpha}) \quad k_i = \frac{\gamma}{c} \quad (6.15)$$

Con el fin de mejorar la dinámica transitoria y la robustez frente a variaciones paramétricas del motor, se adopta un esquema de adaptación proporcional–integral:

$$\hat{\omega}_r = K_p (e_{i_{s\alpha}} \hat{\psi}'_{r\beta} - e_{i_{s\beta}} \hat{\psi}'_{r\alpha}) + K_i \int (e_{i_{s\alpha}} \hat{\psi}'_{r\beta} - e_{i_{s\beta}} \hat{\psi}'_{r\alpha}) dt \quad (6.16)$$

donde K_p y K_i son ganancias positivas arbitrarias.

Aunque la precisión de la estimación de velocidad se mejora mediante este observador, aún existe una variación finita de parámetros (particularmente en las resistencias del estator y rotor). El error de estimación tiende a hacerse más dominante conforme la velocidad se aproxima a cero.

La matriz de ganancia del observador K se determina mediante asignación de polos. Este método permite definir la dinámica del observador en lazo cerrado igualando su polinomio característico $\det(sI - (\hat{A} - KC)) = 0$ con un polinomio deseado cuyas raíces (polos) se encuentran desplazadas radialmente respecto a los polos naturales del motor, garantizando que la dinámica del observador sea más rápida que la del sistema en lazo abierto. De acuerdo con este criterio, se obtienen las siguientes ganancias (Cherifi & Miloud, 2024)

$$k_1 = -(k - 1) \left(\frac{1}{\sigma T_s} + \frac{1}{\sigma T_r} \right) \quad (6.17)$$

$$k_2 = (k - 1) \hat{\omega}_r \quad (6.18)$$

$$k_3 = (k^2 - 1) \left[\left(\frac{1}{\sigma T_s} + \frac{1 - \sigma}{\sigma T_r} \right) \frac{\sigma L_s L_m}{L_r} + \frac{L_m}{L_r} \right] + \frac{\sigma L_s L_m}{L_r} (k - 1) \left(\frac{1}{\sigma T_s} + \frac{1 - \sigma}{\sigma T_r} \right) \quad (6.19)$$

$$k_4 = - \frac{(k - 1) \sigma L_s L_m}{L_r} \hat{\omega}_r \quad (6.20)$$

donde el parámetro $k > 1$ y $k < 3$ define la relación entre los polos del observador y los polos del sistema.

6.9. Procedimiento de implementación de control vectorial sensorless

Para implementación se siguió el siguiente procedimiento

6.9.1. Configuración de Sensorless FOC en MC SDK Workbench

Para la generación del código de control, se empleó la herramienta STM32 MC SDK Workbench. Se tomó como punto de partida el proyecto de referencia para control de campo orientado (FOC) sin sensores (Sensorless) aplicado a motores de inducción (Figura 98). Como se aprecia en la Figura 99, se procedió a la parametrización de la etapa de potencia y del motor, introduciendo los valores de resistencia, inductancia y constantes electromecánicas previamente identificados. Asimismo, se configuraron las ganancias de los lazos de control de corriente y velocidad como se ve en la Figura 100.

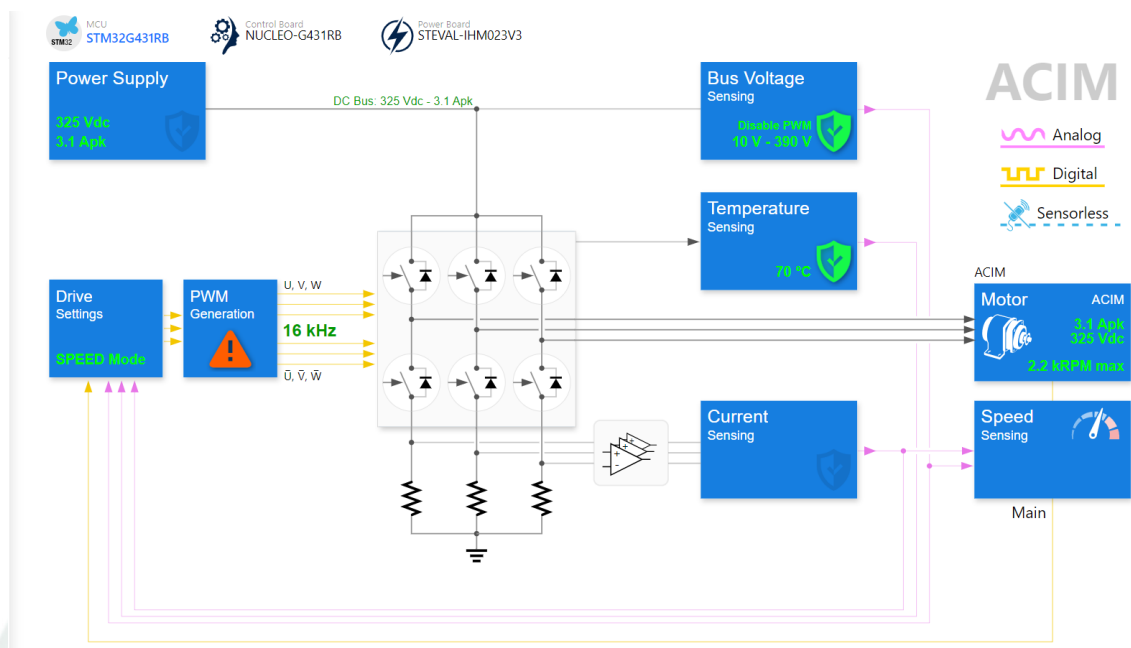


Figura 98. Interfaz principal de configuración del proyecto para motor de inducción (ACIM) en STM32 MC SDK Workbench.

Poles Pairs:	2	Bus Voltage:	325 Vdc
Nominal current (In):	3.07 Arms	PWM Frequency:	16000 Hz
Magnetizing current (Im):	1.8 Arms	Stator resistance (Rs):	9.6 Ω
Nominal phase voltage (Vn):	220 Vrms	Stator leakage inductance (Lls):	0.025 H
Nominal frequency (Fn):	60 Hz	Magnetizing inductance (Lms):	0.535 H
		Inertia:	3200 $\mu\text{N}\cdot\text{m}\cdot\text{s}^2$
Rotor resistance (Rr):	4.8 Ω	Friction:	40 $\mu\text{N}\cdot\text{m}\cdot\text{s}$
Rotor leakage inductance (Llr):	0.025 H	Max. Application Speed:	1800 rpm

Figura 99. Parámetros del Motor

Current regulators		Speed Regulator	
Auto Calculate: ☒		Speed loop frequency: 500 Hz	
Bandwidth: 1450 rad/s		Target Speed: 1730 RPM	
Torque (Iq) 71.40483	Flux (Id) 71.40483	Auto Calculate: ☒	
Kp: 71.40483	Kp: 71.40483	Bandwidth: 130 rad/s	
Ki: 13920	Ki: 20880	Kp: 0.416	
		Ki: 0.0052	

Observer

Lso Gain:

Auto Calculate: ☒

Bandwidth: rad/s

Kp: 7.01335

Ki: 2050.82329

Lso Debug: ☐

Figura 100. Parámetros de sintonización de controles PI de velocidad, corrientes y observador LSO

6.9.2. Configuración de periféricos en STM32CubeMx

Tras finalizar la etapa de parametrización en el Motor Control Workbench, se procedió a la generación de los archivos de configuración iniciales. Este proceso crea automáticamente un archivo con extensión. ioc, el cual contiene la definición de la capa de abstracción de hardware (HAL). Posteriormente, se utilizó la herramienta STM32CubeMX para revisar y validar la asignación de recursos del microcontrolador STM32G431RB, verificando específicamente el mapeo de pines (Pinout) y la configuración de las entradas y salidas (GPIO) necesarias para la comunicación con la etapa de potencia y los periféricos de usuario.

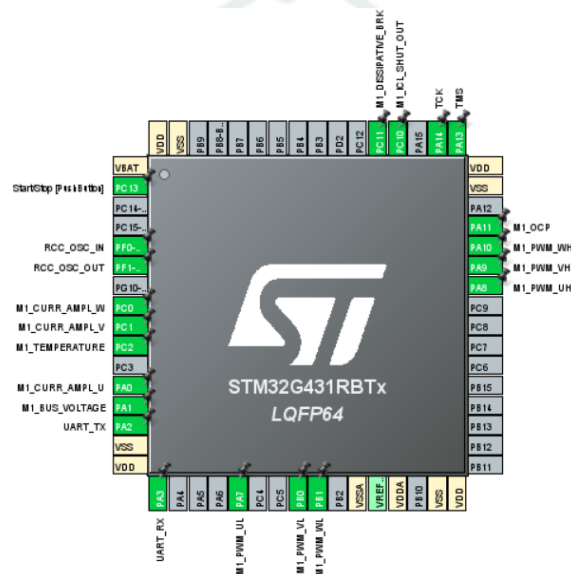


Figura 101. Mapeo de pines y asignación de periféricos del STM32G431RB en la interfaz de STM32CubeMX

Los pines se agrupan por función.

1. Etapa de Potencia (Control del Inversor)

Estos pines envían las señales PWM a los Gate Drivers que abren y cierran los transistores (IGBTs o MOSFETs). Funcionan en pares complementarios (Lado Alto y Lado Bajo).

- **Lado Alto (High Side - Transistores Superiores):**

PA8 (M1_PWM_UH): PWM Fase U.

PA9 (M1_PWM_VH): PWM Fase V.

PA10 (M1_PWM_WH): PWM Fase W.

- **Lado Bajo (Low Side - Transistores Inferiores):**

PA7 (M1_PWM_UL): PWM Fase U.

PB0 (M1_PWM_VL): PWM Fase V.

PB1 (M1_PWM_WL): PWM Fase W.

El STM32 utiliza un "tiempo muerto" (Dead Time) insertado automáticamente entre la señal Alta y Baja de cada fase para evitar cortocircuitos.

2. Adquisición de Datos (Feedback / Sensores)

Son entradas al ADC (Conversor Analógico-Digital).

- **Corrientes de Fase**

PA0 (M1_CURR_AMPL_U): Mide la corriente en la fase U.

PC1 (M1_CURR_AMPL_V): Mide la corriente en la fase V.

PC0 (M1_CURR_AMPL_W): Mide la corriente en la fase W.

El sufijo "AMPL" indica que la señal ya pasó por la etapa de acondicionamiento (OpAmps) de la placa IHM023V3.

- **Monitoreo del Sistema:**

PA1 (M1_BUS_VOLTAGE): Mide el voltaje de la fuente DC (Vbus).

PC2 (M1_TEMPERATURE): Lee el sensor de temperatura del disipador de calor.

3. Seguridad y Protecciones (Hardware)

Estos pines actúan como "reflejos" rápidos para proteger el hardware antes de que el software pueda reaccionar.

- **PA11 (M1_OCP):** *Over Current Protection*. Si la corriente sube peligrosamente, el hardware envía una señal aquí que apaga inmediatamente todos los PWMs (Break Input).
- **PC11 (M1_DISSIPATIVE_BRK):** Controla el circuito de frenado (Brake Chopper). Si el motor actúa como generador y sube mucho el voltaje, este pin activa una resistencia para quemar el exceso de energía.
- **PC10 (M1_ICL_SHUT_OUT):** *Inrush Current Limiter*. A veces usado para activar/desactivar el driver principal o gestionar relés de arranque suave.

4. Interfaz y Comunicación

- **PA2 (UART_TX) / PA3 (UART_RX):** Comunicación Serial con la PC. Por aquí viajan los datos hacia el **Motor Pilot** para ver las gráficas en tiempo real.
- **PC13 (Start/Stop):** Botón azul de usuario en la Nucleo. Inicia o detiene la máquina de estados del motor.
- **PA13 (TMS) / PA14 (TCK):** Pines de depuración (SWD). Son los que usa el ST-Link para programar el chip.

5. Reloj (Clock)

- **PF0 (RCC_OSC_IN) / PF1 (RCC_OSC_OUT):** Conexión al cristal oscilador externo. Fundamental para tener una base de tiempo precisa para los cálculos matemáticos complejos del FOC.

El Timer 1 actúa configurándose en Modo Alineado al Centro (Center Aligned), lo cual permite generar señales SVPWM simétricas para reducir el ruido electromagnético, con una frecuencia de conmutación definida por un periodo (ARR) ajustado a la mitad de los ciclos totales debido al conteo triangular. Esta configuración optimiza el rendimiento del procesador mediante el Repetition Counter, que permite desacoplar la frecuencia del bucle de control de la del PWM, y utiliza crucialmente la señal de disparo TRGO (OC4REF) para sincronizar el ADC, asegurando que la medición de las corrientes de fase ocurra exactamente en el centro del pulso, libre de ruido de conmutación.

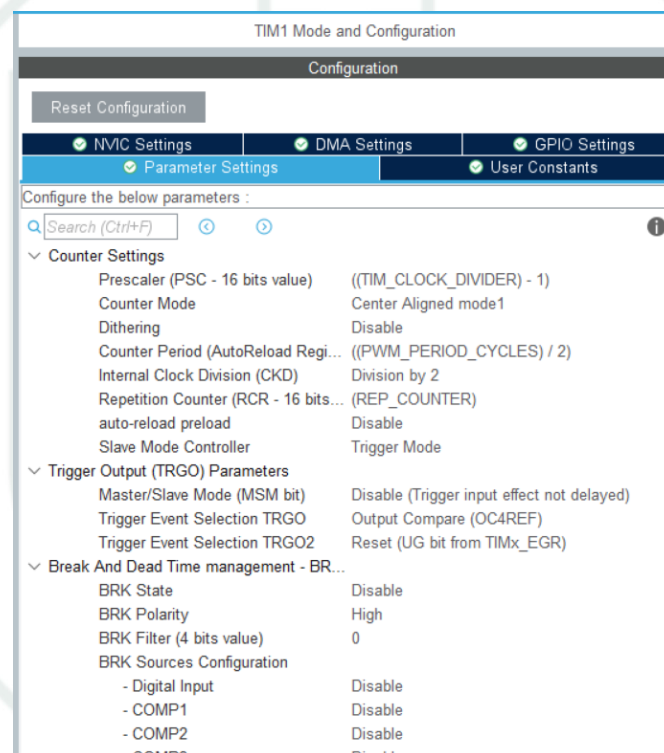


Figura 102. Configuración de Timer 1

6.9.3. Compilación, Carga y Depuración en STM32CubeIDE

Una vez finalizada la configuración de periféricos, se procedió a la generación automática del código fuente en lenguaje C. Este proyecto fue importado al entorno de desarrollo integrado STM32CubeIDE para realizar el proceso de compilación (Build) y validación de sintaxis.

Posteriormente, se realizó la programación de la memoria flash del microcontrolador STM32G431RB mediante la interfaz de depuración ST-LINK. Durante las pruebas de funcionamiento, se utilizó la herramienta de monitorización en tiempo real (Live Watch) para visualizar el comportamiento dinámico de las variables críticas del sistema. El código fuente implementado para los lazos de control de corriente, velocidad y el observador de estado LSO se detalla en el Anexo 8.

6.10. Validación experimental y resultados

6.10.1. Verificación de Señales de Control (PWM)

Tras la carga del firmware en el microcontrolador, se dio inicio a pruebas verificando la integridad de las señales de control. Utilizando un osciloscopio digital, se midieron las salidas de los temporizadores para validar la frecuencia de conmutación y la modulación por ancho de pulsos (PWM). Como se observa en la Figura 103, se confirmó la correcta generación de las señales complementarias.

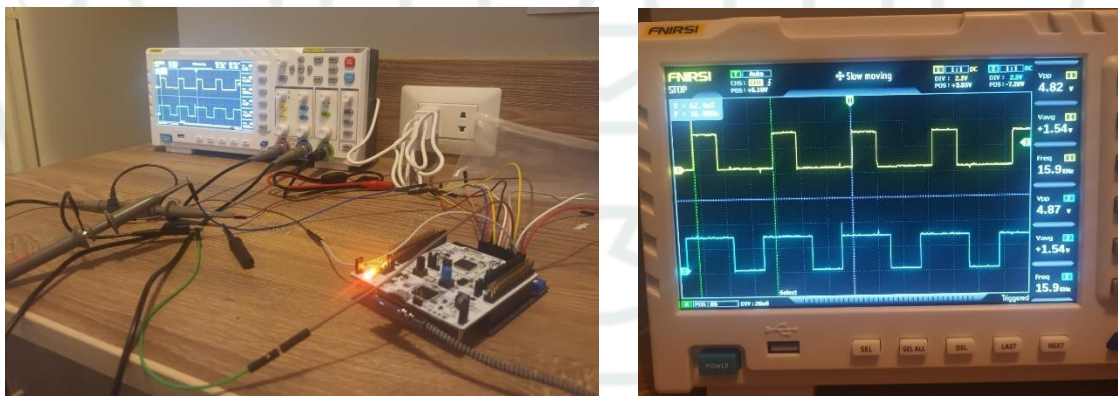


Figura 103. Validación Experimental de Señales PWM Complementarias a 16 kHz implementadas en STM32G431RB

6.10.2. Integración del Banco de Pruebas

La validación experimental del sistema de control se llevó a cabo en un banco de pruebas experimental independiente, acondicionado específicamente para esta investigación.

Para la conformación de este entorno, se procedió al ensamblaje físico del sistema integrando la tarjeta de control NUCLEO-G431RB con la etapa de potencia STEVAL-IHM023V3 y un motor de inducción de 1 HP. Para centralizar y acondicionar las conexiones entre la etapa de control y la de potencia, se utilizó la placa de expansión X-NUCLEO-IHM09M1. Esta interfaz facilitó el conexionado de las señales de retroalimentación y control mediante cables blindados, garantizando la integridad de las señales.

El montaje final para las pruebas de funcionamiento en vacío (sin carga mecánica acoplada) se detalla en la Figura 104.

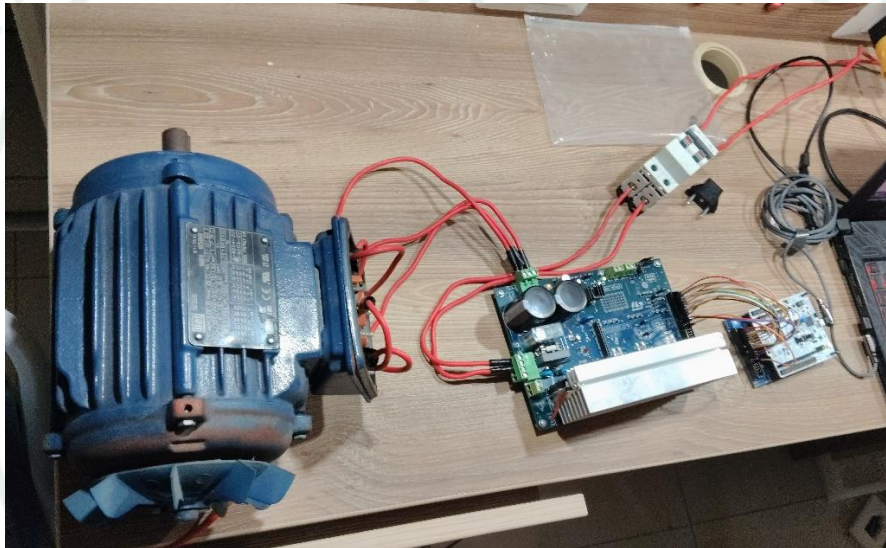


Figura 104. Banco de pruebas experimental para motor de inducción sin carga

En la Figura 105 se detalla el montaje experimental completo para las pruebas de robustez ante perturbaciones de carga. A diferencia del funcionamiento en vacío, este arreglo incorpora un banco de carga dinámica. El eje del motor de inducción principal (planta bajo control) se encuentra acoplado mecánicamente mediante un sistema de transmisión por faja y poleas a un segundo motor, el cual opera como freno o carga emulada.

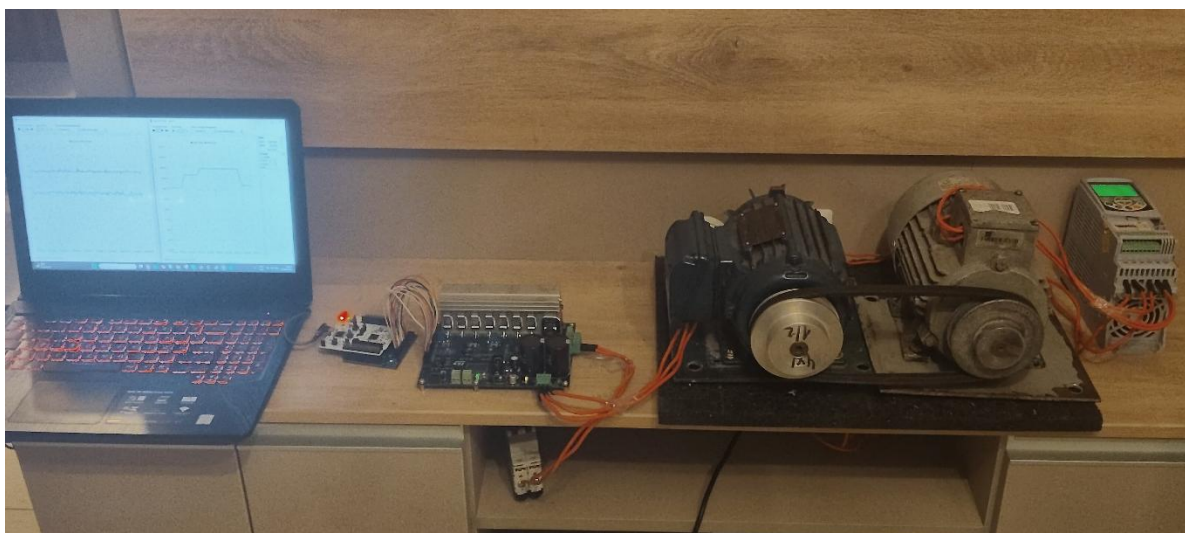


Figura 105. Banco de pruebas experimental para motor de inducción con carga

Para gobernar el torque resistente impuesto por el motor de carga, se ha integrado al sistema un Variador de Frecuencia (VFD) comercial visible en el extremo derecho del montaje. La utilización de este VFD permite parametrizar e inyectar perfiles de carga controlados (tanto de torque constante como escalones de carga variable). Simultáneamente, en el extremo izquierdo se observa la estación de monitoreo que interactúa en tiempo real con la tarjeta NUCLEO-G431RB, permitiendo la adquisición de datos, la sintonización de los lazos de control PI y la supervisión del desempeño del estimador de estados durante la aplicación de la carga mecánica.

6.10.3. Validación Funcional con STM32 MotorPilot

Finalmente, se realizaron las pruebas dinámicas del motor utilizando la herramienta de monitoreo STM32 MotorPilot. Esta interfaz permitió la visualización en tiempo real de las variables críticas del observador de estado y el lazo de control.

Como se evidencia en la Figura 106, se obtuvo una respuesta estable en la estimación de velocidad, logrando que la velocidad real del motor siga la referencia establecida con un error mínimo en estado estacionario. Cabe destacar que esta fase de validación se centró en el

comportamiento en vacío, estableciendo la base operativa para pruebas bajo condiciones de carga dinámica.

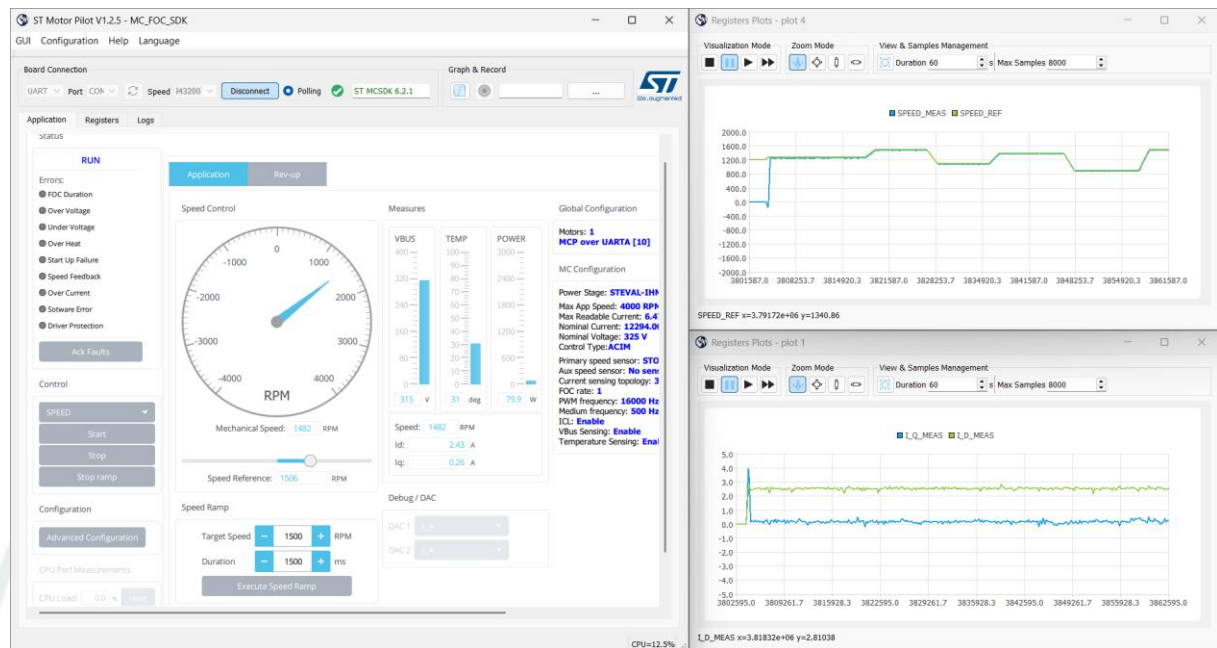


Figura 106. Interfaz gráfica de resultados experimentales obtenida mediante la herramienta STM32 MotorPilot. durante la prueba sin carga

Como se observa en la gráfica de respuesta temporal, el sistema logra estabilizarse en los diferentes valores de velocidad de referencia, mostrando un comportamiento dinámico robusto ante cambios escalón en la consigna.

La corriente de flujo (i_{sd}) se mantiene estable en 2.56 A, necesaria para la magnetización del motor de inducción y la corriente de torque (i_{sq}) es mínima (0.12 A), correspondiendo únicamente al torque necesario para vencer la fricción.

La Figura 107 muestra la respuesta del sistema ante la aplicación y remoción abrupta de carga mecánica (escalones de torque) manteniendo una velocidad de referencia constante de 1200 RPM.

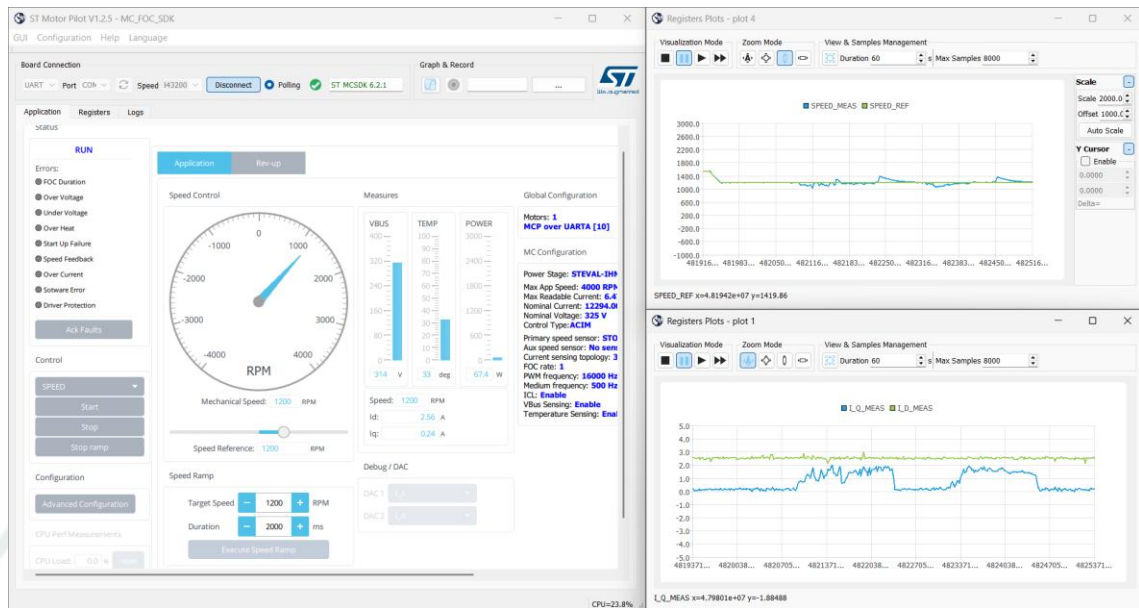


Figura 107. Respuesta dinámica del control FOC sensorless ante escalones de carga variable (STM32 MotorPilot)

Como se puede apreciar en la gráfica inferior derecha de la Figura 107, la inyección de la perturbación de carga se refleja inmediatamente en el incremento de la corriente de cuadratura o corriente de torque (i_{sq}) trazo azul, la cual pasa de un estado de vacío a compensar el torque resistente impuesto. El lazo de control PI de velocidad demuestra un excelente rechazo a las perturbaciones: ante cada impacto de carga se produce una mínima caída transitoria en la velocidad medida (gráfica superior derecha), la cual es rápidamente mitigada, restableciendo el error de estado estacionario a cero en segundos. Durante estas severas variaciones de torque, la corriente de magnetización o de flujo (i_{sd}) trazo verde permanece inalterable y desacoplada, validando el correcto funcionamiento.

Posteriormente, se evaluó la capacidad de seguimiento de trayectoria de la velocidad con una carga mecánica constante acoplada al eje, aplicando variaciones en la referencia de velocidad.

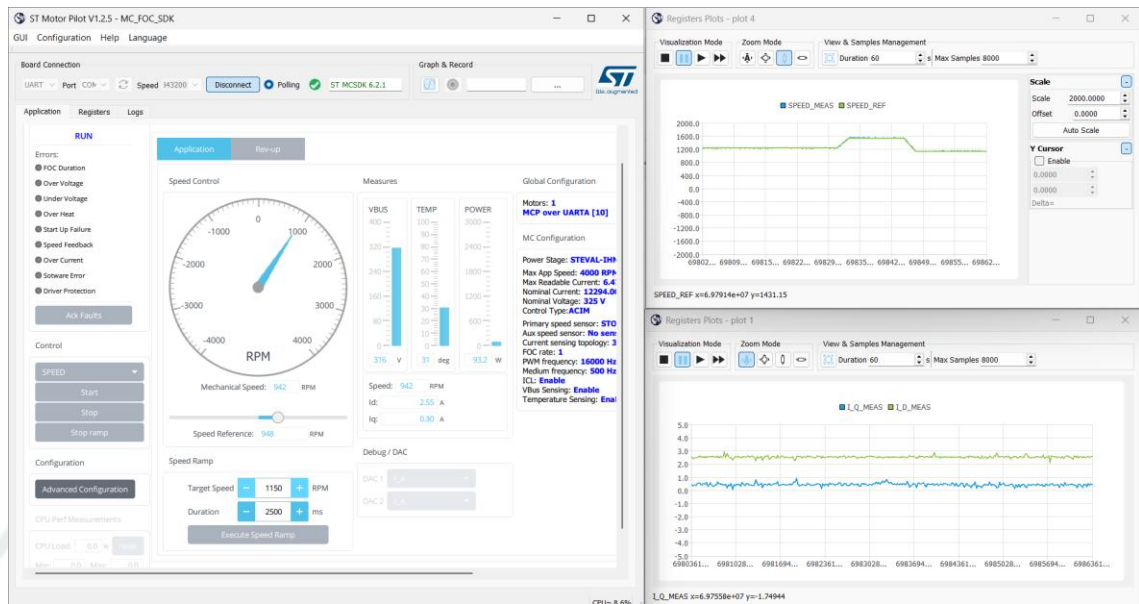


Figura 108. Seguimiento de perfiles de velocidad bajo condiciones de carga mecánica constante (STM32 MotorPilot).

En la Figura 108 se observa la respuesta del motor frente a cambios escalón en la velocidad de referencia. A diferencia de las pruebas en vacío, la gráfica de corrientes revela que la corriente de torque (i_{sq}) mantiene un valor base continuo superior a cero, correspondiente al esfuerzo necesario para vencer la carga constante aplicada por el motor de freno. Durante las transiciones de velocidad, se evidencian los picos de corriente transitorios requeridos para proporcionar el torque de aceleración y desaceleración que vence la inercia del sistema conjunto. La velocidad estimada por el observador sigue fielmente a la referencia, lo que corrobora que el estimador de Luenberger embebido en el microcontrolador no pierde convergencia.

6.11. Costos de implementación del proyecto

6.11.1. Descripción y Verificación de la Tabla de Costos

La Tabla 13 presenta el presupuesto detallado de los materiales y equipos adquiridos para la implementación del banco de pruebas experimental de control vectorial sensorless. La

inversión total asciende a S/. 2 705,00, distribuida en once ítems que abarcan los subsistemas de potencia, control digital, transmisión mecánica e insumos generales.

Tabla 13. Presupuesto de implementación del banco de pruebas experimental

Item	Descripción	Cant.	Unid. Medida	Precio Unit. (S/.)	Precio total (S/.)
1	Motor de inducción de 1 HP WEG	1	UND	500,00	500,00
2	Motor de inducción de 1 HP SIEMENS (usado)	1	UND	220,00	220,00
3	Variador de velocidad WEG CFW500A04P3T4NB20G2	1	UND	850,00	850,00
4	Placa de evaluación de inversor trifásico AC-DC (1 kW) STMicroelectronics STEVAL- IHM023V3	1	UND	787,00	787,00
5	Placa de desarrollo STMicroelectronics NUCLEO- G431RB con microcontrolador STM32G431RB	1	UND	90,00	90,00
6	Placa de expansión STMicroelectronics X- NUCLEO-IHM09M1	1	UND	114,00	114,00
7	Cable USB micro B 3.0	1	UND	20,00	20,00
8	Polea de transmisión de Ø 3 pulg.	2	UND	25,00	50,00
9	Faja de transmisión tipo A	1	UND	14,00	14,00
10	Plancha base de motor de inducción	2	UND	15,00	30,00
11	Consumibles (cables de fuerza, cables de control, cinta aislante, terminales, pernos)	1	UND	30,00	30,00
COSTO TOTAL (S/.)					2705,00

Nota: Precios expresados en Soles (S/.) e incluyen IGV.

6.11.2. Estructura del Presupuesto por Subsistema

En primer lugar, los equipos de potencia y accionamiento electromecánico representan la mayor fracción del presupuesto. Esto incluye la adquisición del motor de inducción principal WEG de 1 HP (S/. 500,00) que actúa como planta a controlar, y un motor auxiliar SIEMENS de 1 HP (S/. 220,00) empleado como carga mecánica acoplada en el banco de pruebas. Asimismo, se incluyó un variador de velocidad comercial WEG CFW500 (S/. 850,00)

destinado a gobernar el motor de carga, permitiendo así generar perfiles de torque perturbador de manera controlada para evaluar la robustez del observador de estado.

En la categoría de plataformas de control y desarrollo de hardware, se invirtió en las herramientas específicas del ecosistema STMicroelectronics. La etapa de potencia inversora, mediante la placa de evaluación STEVAL-IHM023V3 (S/. 787,00). El procesamiento digital se delegó a la tarjeta de desarrollo NUCLEO-G431RB (S/. 90,00), interconectada a la etapa de potencia mediante la placa de expansión X-NUCLEO-IHM09M1 (S/. 114,00).

Finalmente, el subsistema mecánico de transmisión y soporte, compuesto por dos poleas de Ø 3 pulgadas (ítem 8, S/. 50,00), una faja de transmisión tipo A (ítem 9, S/. 14,00) y dos planchas base de motor (ítem 10, S/. 30,00), suma S/. 94,00. Los consumibles eléctricos (ítem 11, S/. 30,00), que incluyen cables de fuerza y control, cinta aislante, terminales y pernos de sujeción.

6.11.3. Análisis Costo-Beneficio

La selección de la arquitectura basada en el microcontrolador STM32G431RB y el módulo de potencia STEVAL-IHM023V3 presenta una relación costo-beneficio favorable para la investigación y el desarrollo de esquemas avanzados de control de motores eléctricos.

En aplicaciones industriales a gran escala, el uso directo de tarjetas de desarrollo presenta complejidades operativas. Los entornos de planta exigen equipos con altos grados de protección (como normativas IP o NEMA), certificaciones industriales, soporte técnico continuo y una integración estandarizada con redes industriales (profibus, profinet, etc). Por estas razones, en la industria predominan los variadores de frecuencia (VFD) de marcas comerciales consolidadas. Sin embargo, desde la perspectiva de la investigación y desarrollo (I+D), estos equipos comerciales actúan como sistemas de caja cerrada; su firmware propietario imposibilita la implementación, modificación y estudio.

Es precisamente en los laboratorios académicos donde esta plataforma demuestra su verdadero beneficio, especialmente al realizar una comparación económica con sistemas de propósito equivalente. Típicamente, la implementación de plataformas de control basadas en hardware dedicado, como Procesadores Digitales de Señales (DSP), arreglos de compuertas programables (FPGA) o sistemas de prototipado rápido como dSPACE, exigen inversiones que superan ampliamente los \$1000 USD en sus configuraciones básicas. Frente a estas alternativas de alto costo, la electrónica de control y potencia de la solución propuesta requiere una inversión de apenas S/. 1000,00 (aproximadamente \$340 USD), ofreciendo un desempeño computacional y capacidades de investigación completamente equiparables.

Además de la viabilidad económica, destaca el valor didáctico intrínseco de la plataforma. A diferencia de los equipos comerciales inaccesibles, la arquitectura basada en STM32 expone completamente el código fuente, permitiendo a estudiantes e investigadores comprender, modificar y extender cada bloque del algoritmo FOC. En particular, el código generado por el ecosistema STM32 MC SDK implementa en lenguaje C estándar los bloques fundamentales: transformadas de Clarke y Park, lazos PI de corriente y velocidad, modulación SVPWM y el observador de Luenberger. Al ser código estructurado, comentado y organizado por capas de abstracción de hardware (HAL), constituye un recurso de aprendizaje directo que vincula la teoría con su realización computacional concreta.

Finalmente, las herramientas de desarrollo integradas refuerzan este valor pedagógico. Entornos como STM32CubeIDE y STM32 MotorPilot permiten observar en tiempo real el comportamiento de variables críticas (como las corrientes I_{sd} e I_{sq} , la velocidad estimada y el ángulo de flujo), estableciendo una correspondencia directa entre las ecuaciones del modelo matemático y su evolución dinámica física.

CONCLUSIONES

- Se demostró que la identificación de parámetros del motor de inducción mediante el algoritmo de Optimización por Enjambre de Partículas (PSO), utilizando exclusivamente datos de placa y manufactura, constituye una metodología confiable. Los márgenes de error obtenidos al comparar con valores experimentales, entre el 4.3% y el 11.97%, confirman que la función objetivo basada en la minimización de errores cuadráticos estima con suficiente precisión las resistencias y reactancias del motor para su aplicación en estrategias de control vectorial.
- Se diseñó e implementó en simulación una estrategia de Control de Campo Orientado (FOC) sensorless con el Filtro de Kalman Extendido (EKF) como estimador, logrando alta precisión en el seguimiento de referencias de velocidad y torque. Se verificó que la sintonización de los controladores PI mediante el lugar de las raíces y diagramas de Bode, junto con la compensación de acoplamiento cruzado y acción anti-windup, garantiza estabilidad y robustez ante variaciones abruptas de carga. Asimismo, se comprobó que una parametrización adecuada de las matrices de covarianza Q y R del EKF es determinante para mantener la fidelidad de estimación bajo condiciones exigentes de operación.
- Se diseñó y validó mediante simulación un esquema de detección, aislamiento y tolerancia a fallas (FDI/FTC) basado en el Filtro de Kalman Extendido (EKF), capaz de identificar y discriminar múltiples escenarios de falla en sensores de corriente y velocidad, incluyendo pérdidas totales de señal por circuito abierto y degradaciones paramétricas como offset o ganancia incorrecta. Se demostró, que la arquitectura de reconfiguración automática gestionada mediante indicadores lógicos mantiene el desempeño del accionamiento ante estos eventos críticos, consolidando una solución basada íntegramente en software como alternativa viable a la redundancia de hardware.

- Se implementó y validó experimentalmente el algoritmo de control vectorial sensorless en el microcontrolador STM32G431RB, empleando un Observador de Estado de Luenberger para la estimación de velocidad y flujo del rotor. Se comprobó que la sincronización entre la modulación SVPWM centrada y la adquisición de corrientes permite ejecutar algoritmos de control de alta complejidad en tiempo real con los recursos del sistema embebido. Las pruebas dinámicas en el banco de prototipos confirmaron un seguimiento preciso de la referencia de velocidad y una estimación confiable de las variables de estado del rotor sin el uso de sensores de velocidad físicos.

RECOMENDACIONES

- Se recomienda implementar físicamente el Filtro de Kalman Extendido (EKF) y el esquema de detección y tolerancia a fallas (FDI/FTC) en un microcontrolador STM32G4 o procesador digital de señales DSP C2000, dado que ambos algoritmos fueron validados únicamente en simulación. Esta implementación permitirá contrastar su desempeño frente al Observador de Luenberger actualmente implementado y verificar experimentalmente la capacidad del EKF para manejar ruido real de sensores y ejecutar la reconfiguración ante fallas en tiempo real.
- Extender la estrategia de control vectorial implementada incorporando el algoritmo de debilitamiento de campo, con el fin de habilitar la operación estable del motor por encima de la velocidad nominal de 1800 RPM. Su validación experimental en el banco de prototipos permitiría evaluar el comportamiento del sistema en aplicaciones de amplio rango de velocidad sin comprometer la integridad del inversor.
- Replicar las pruebas experimentales en motores de inducción de mayor potencia, con el objetivo de evaluar la escalabilidad y portabilidad del sistema de control desarrollado. Esta validación permitiría determinar si los parámetros identificados por PSO y la sintonización del observador mantienen su efectividad ante dinámicas electromagnéticas y térmicas más exigentes.

REFERENCIAS BIBLIOGRÁFICAS

- Abbaspour, A., Mokhtari, S., Sargolzaei, A., & Yen, K. K. (2020). A Survey on Active Fault-Tolerant Control Systems. *Electronics*, 9(9), 1513.
- Adamczyk, M., & Orlowska, T. (2024). Current sensor fault-tolerant control based on modified Luenberger observers for safety-critical vector-controlled induction motor drives. *Bulletin of the Polish Academy of Sciences Technical Sciences*, 72(5), e151041. doi:10.24425/bpasts.2024.151041
- Aguilera, F., De la Barrera, P. M., & De Angelo, C. H. (2022). Speed and current sensor fault-tolerant induction motor drive for electric vehicles based on virtual sensors. *Electrical Engineering*, 104(5), 3157-3171.
- Aparício, A. J. (2019). *Indirect Field Oriented Control of Two-Phase Induction Machines*. Master of Science, Universidade de Lisboa.
- Azzoug, Y., Sahraoui, M., Pusca, R., Ameid, T., & Romary, R. &. (2021). Current Sensors Fault Detection and Tolerant Control Strategy for Three-Phase Induction Motor Drives. *Electrical Engineering*, 103(2), 881-898.
- Blanke, M., Kinnaert, M., Lunze, J., & Staroswiecki, M. (2016). Introduction to diagnosis and fault-tolerant control. In *Diagnosis and Fault-Tolerant Control*. Springer.
- Chapman, S. J. (2012). *Máquinas eléctricas (5.ª ed.)*. (S. Sarmiento Ortega, Trad.). McGraw-Hill/Interamericana Editores.
- Chauhan, D., & Suganthan, P. N. (2025). Learning strategies for particle swarm optimizer: A critical review and performance analysis. *Swarm and Evolutionary Computation*, 98, 102048.
- Cherifi, D., & Miloud, Y. (2024). Intelligent Adaptation Mechanism of Full-Order Luenberger Observer based on Fuzzy Logic Applied to Direct Control by Flux Orientation without Speed Sensor for Doubly Fed Induction Motor. *WSEAS Transactions on Systems and Control*, 19, 27-38.
- CONICYT. (2015). *Disciplinas OCDE*. Retrieved from https://www.conicyt.cl/pci/files/2015/07/Disciplinas-OCDE.pdf?utm_source=chatgpt.com
- Córdova Claros, J. (2017). *Diseño de un sistema de diagnóstico y control tolerante de fallas en actuadores para el proceso hidráulico de cuatro tanques acoplados*. Master's thesis, Pontificia Universidad Católica del Perú.
- Ding, S. X. (2021). *Advanced methods for fault diagnosis and fault-tolerant control* (Vol. 184). Berlin: Springer.
- Dosreis, E. (2021). Understanding the Design Process for Motors & Drives.
- Gad, A. G. (2022). Particle Swarm Optimization Algorithm and Its Applications: A Systematic Review. *Archives Of Computational Methods In Engineering*, 29(5), 2531-2561.

- Geyer, T. (2016). *Model predictive control of high power converters and industrial drives*. 2017 IEEE Energy Conversion Congress and Exposition (ECCE).
- Grewal, M. S., & Andrews, A. P. (2014). *Kalman filtering: Theory and practice using MATLAB* (4th ed.). John Wiley & Sons.
- Košál, T. (2021). *On-line Identification Methods of Induction Machine Parameters*. (Doctoral dissertation, Czech Technical University). Retrieved from <https://dl.acm.org/doi/10.5555/AAI29117967>
- Kovvali, N., & Banavar, M. &. (2022). *An Introduction to Kalman Filtering with MATLAB Examples*.
- Kumar, M. P., Velpula, S., Saiprakash, C., & Sahoo, B. (2025). Advancements in fault detection and diagnosis methods for electric vehicles: a review. *Discover Applied Sciences*, 7((11)), 1235.
- Le Fosse, R. (2021). *Dynamic modeling of induction motors in developing tool for automotive applications*. Politecnico Di Torino. Retrieved from https://mediatum.ub.tum.de/doc/1197569/1197569.pdf?utm_source=chatgpt.com
- Liu, C., Chau, K. T., Lee, C. H., & Song, Z. (2021). A critical review of advanced electric machines and control strategies for electric vehicles. *109(6)*, 1004–1028.
- Llorente, R. M. (2020). *Practical control of electric machines: Model-based design and simulation*. Springer Nature.
- Lyshevski, S. E. (2017). *Mechatronics and control of electromechanical systems*. CRC Press.
- Mohammadi, H. R., & Akhavan, A. (2014). Parameter Estimation of Three-Phase Induction Motor Using Hybrid of Genetic Algorithm and Particle Swarm Optimization. *Journal Of Engineering*, 2014, 1-6.
- Morfin, O., Ornelas-Tellez, F., Padilla, N., Gomez, M., Hernandez, O., & Ramirez-Betancour, R. &. (2025). Sensorless Speed Controller for the Induction Motor Using State Feedback and Robust Differentiators. *Machines*, 13(9), 846.
- Nadeem, A. (2020). *Kalman filters*. Retrieved from Medium: <https://medium.com/@abdullahnadeem27/kalman-filters-ba39c8b22704>
- NXP Semiconductor, F. (2021). *Sensorless ACIM Motor Control Using MC56F82748*. Retrieved from <https://www.nxp.com/docs/en/application-note/AN5210>.
- Olvera, J. M. (2022). *Modelo digital de un motor de inducción para su uso en el desarrollo de Gemelos Digitales*.
- Salas Camacho, F. A. (2022). *Desarrollo de un sistema de detección y localización de fugas basado en filtro de Kalman extendido para sistemas de transporte de petróleo crudo aplicado al tramo I del Oleoducto Norperuano*. Master's thesis, Pontificia Universidad Católica del Perú.
- Sancha Miguel, J. (2023). *Desarrollo e implementación de un control orientado de campo sensorless en un motor asíncrono de inducción*. Universitat Politècnica de València.

- Shen, S. &. (2023). *Practical Control of Electric Machines for EV/HEVs*. Springer.
- Siemens. (2024). The True Cost of Downtime 2024.
- STMicroelectronics. (2019). *STM32G431RB: Arm® Cortex®-M4 32-bit MCU with DSP and FPU, 170 MHz, 128 KB Flash, 32 KB SRAM (Datasheet)*. STMicroelectronics.
- STMicroelectronics. (2020). *SDK AC induction motor (STM32MotorControl)*. Retrieved from STM32 MCU Wiki:
https://wiki.st.com/stm32mcu/wiki/STM32MotorControl:SDK_AC_induction_motor
- STMicroelectronics. (n.d.). *Getting started with the X-NUCLEO-IHM09M1 motor control connector expansion board for STM32 Nucleo (User manual UM1970, Rev. 1)*. STMicroelectronics.
- STMicroelectronics. (2017). *STEVAL-IHM023V3: 1 kW 3-phase motor control evaluation board featuring L6390 drivers and STGP10H60DF IGBT (Data brief, DocID027120 Rev. 1)*. STMicroelectronics.
- Tahar, B., Bousmaha, B., Ismail, B., & Houcine, B. (2018). Speed Sensorless Field-Oriented Control of Induction Motor with Fuzzy Luenberger Observer. *Electrotehnica, Electronica, Automatica*, 66(4).
- Tiwari, R. A. (2019). *Determination of Induction Motor Speed using Kalman Filter*. Doctoral dissertation, Master's thesis, Faculty of Electrical Engineering, CTU in.
- Tran, C. D. (2021). Current and Speed Sensor Fault Diagnosis Method Applied to Induction Motor Drive. *IEEE Access*, 9, 38660-38672.
- Wu, C., Sehab, R., Akrad, A., & Morel, C. (2022). Fault diagnosis methods and fault tolerant control strategies for the electric vehicle powertrains. *Energies*, 15(13), 4840.
- Wu, F., Zhao, J., & Liu, Y. &. (2016). A real-time sensor fault detection, isolation and reconfiguration method for vector controlled induction motors based on extended Kalman filter. *Proceedings of the International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM 2016)*, 617–624.
- Xie, Q., Xu, Q., Luo, L., Tu, Y., & Song, W. (2025). Sensorless Induction Motor Control Based on an Improved Full-Order State Observer. *Energies*, 18(16), 4374.
- Yepes, A. G., Gonzalez-Prieto, I., Lopez, O., & Duran, M. J.-G. (2022). A Comprehensive survey on fault tolerance in multiphase AC drives, Part 2: Phase and Switch Open-Circuit faults. *Machines*, 10(3), 221.
- Zhu, Y. (2023). *Embedded Systems with ARM Cortex-M Microcontrollers in Assembly*. E-Man Press Llc.

ANEXOS

Anexo 1: Matriz de consistencia y matriz de operacionalización de variables

Se definieron la variable independiente (Estrategia FTC) y la variable dependiente (Desempeño y robustez del control), las cuales fueron desglosadas en sus respectivas dimensiones e indicadores técnicos.

Tabla 14. Matriz de consistencia

Problemas de investigación	Objetivos	Hipótesis	Metodología
Problema General	Objetivo general	Hipótesis General	
¿De qué manera el diseño de un control vectorial sensorless tolerante a fallas mejora la robustez y el desempeño del control del motor de inducción ante la dependencia y fallas de sensores?	Diseñar un esquema de control vectorial sensorless tolerante a fallas para mejorar la robustez del control vectorial ante fallas de sensores en un motor de inducción	La integración de algoritmos de estimación (EKF ó Luenberger) junto con un esquema FTC permite mantener el control del motor de inducción ante fallas de sensores, validando su eficiencia en simulación y tiempo real. El EKF garantiza la continuidad del control ante fallas sensoriales, mientras que el Observador de Luenberger posibilita la implementación sensorless en hardware de bajo costo.	Tipo de investigación: Aplicada — orientada al diseño e implementación de un sistema de control. Nivel de investigación: Explicativo: se analiza la relación causa-efecto entre la estrategia FTC (basada en EKF) y la robustez del control vectorial ante fallas de sensores, evaluada mediante simulación. Aplicado: se verifica la viabilidad práctica del control sensorless mediante implementación en hardware real. Diseño de investigación Mixto — en dos etapas: (1) Simulado [Fases 1 y 2]: Identificación de parámetros del motor mediante el algoritmo PSO, y posterior diseño y evaluación del control tolerante a fallas (FOC + EKF) mediante inyección controlada de fallas en MATLAB/Simulink. (2) Experimental [Fase 3]: implementación del control <i>sensorless</i> con Observador de Luenberger en hardware físico (microcontrolador de bajo costo) y validación en tiempo real. Unidad de análisis: Motor de inducción trifásico controlado mediante FOC sensorless con estrategia FTC. Enfoque: Cuantitativo (Error relativo porcentual y convergencia de costo en determinación de parámetros) y cualitativo (análisis visual de respuestas gráficas).

Problemas de investigación	Objetivos	Hipótesis	Metodología
Problemas Específicos	Objetivos Específicos	Hipótesis Específicas	
¿De qué manera la aplicación del algoritmo PSO facilita la identificación de los parámetros del motor de inducción frente a los métodos clásicos?	Identificar los parámetros del motor con alta precisión comparando los métodos experimentales clásicos con el algoritmo de optimización PSO.	El algoritmo PSO proporciona una precisión de identificación paramétrica igual o superior a los métodos experimentales clásicos, con la ventaja adicional de ser un método no invasivo que no requiere detener el motor de inducción durante la identificación.	Fase 1: Identificación paramétrica y simulación comparativa. Actividades: - Ensayos experimentales normalizados (vacío, cortocircuito, carga). - Implementación del algoritmo PSO en MATLAB. - Comparación de parámetros obtenidos por ambos métodos. Instrumento / Técnica: - Simulación en MATLAB/Simulink. - Comparación gráfica y numérica (Error relativo)
¿De qué manera la estrategia de control tolerante a fallas (FTC), apoyada en la estimación del EKF, mitiga la vulnerabilidad sensorial del control del motor de inducción ante fallas en los sensores de corriente y velocidad?	Evaluar la capacidad de la estrategia de control tolerante a fallas (FTC), basada en el EKF, para mitigar la vulnerabilidad sensorial ante fallas simuladas de los sensores de corriente y velocidad, garantizando la estabilidad y continuidad del control.	La implementación de la estrategia FTC, utilizando la estimación de estados del EKF, garantiza la estabilidad y continuidad del control del motor de inducción durante las transiciones de falla de sensores manteniéndose el seguimiento visual de la referencia de velocidad de forma estable.	Fase 2: Simulación del control sensorless y evaluación FTC. Actividades: - Diseño e implementación del EKF en MATLAB/Simulink. - Inyección controlada de fallas (circuito abierto, offset, escala). - Evaluación del sistema con FTC activo vs inactivo. Instrumento / Técnica: - Simulación en MATLAB/Simulink. - Análisis visual de graficas de velocidad, corrientes y torque. - Verificación grafica de convergencia y estabilidad post-falla.
¿Cuál es la viabilidad de implementar un control vectorial sensorless utilizando hardware de bajo costo, como microcontroladores?	Implementar la estrategia sensorless utilizando un Observador de Luenberger en un microcontrolador para comprobar su desempeño en tiempo real.	El Observador de Luenberger permite una estimación eficiente de la velocidad durante la operación del motor, validando la viabilidad física del control en hardware y asegurando un correcto seguimiento de la referencia.	Fase 3: Implementación experimental en hardware y pruebas de desempeño. Actividades: - Implementación del Observador de Luenberger en microcontrolador. - Evaluación de los controles PI de velocidad y corrientes - Pruebas de desempeño en tiempo real con prototipo físico. - Evaluación del seguimiento de referencia ante condiciones normales y con fallas. Instrumento / Técnica: - Prototipo físico (microcontrolador de bajo costo). - Osciloscopio / captura de señales.

Tabla 15. Matriz de operacionalización de variables

Variable	Definición Conceptual	Dimensiones (Fases)	Indicadores	Valores Posibles	Instrumento / Técnica de Evaluación
Variable Independiente: Estrategia de estimación y control tolerante a fallas (FTC) Sensorless	Conjunto de algoritmos matemáticos (PSO, FOC, EKF, LSO) diseñados para operar sin sensores físicos de velocidad, estimando las variables de estado del motor e identificando sus parámetros para mantener el control ante contingencias sensoriales.	1. Identificación paramétrica	Método de identificación aplicado.	Método aplicado: - Experimental clásico - Algoritmo PSO	Ensayos de identificación normalizados (vacío, cortocircuito, carga). Algoritmo PSO implementado en MATLAB.
		2. Estimación de estados y detección de fallas	Algoritmo de estimación activo y condición de falla sensorial inyectada	Tipo de falla inyectada: - Sin falla - Falla circuito abierto, falla offset, falla de escala de sensor de corriente - Falla en sensor de velocidad	Simulación en MATLAB/Simulink con inyección controlada de fallas por software.
		3. Implementación en hardware	Estado del Observador de Luenberger en tiempo real	Observador: Activo / Inactivo	Prototipo físico con hardware embebido.
Variable Dependiente: Desempeño y robustez del control vectorial sensorless ante fallas de sensores	Nivel de precisión y estabilidad del sistema de control vectorial sensorless para seguir la referencia de velocidad bajo condiciones normales y ante fallas sensoriales, evaluado mediante análisis visual de respuestas gráficas en	1. Precisión del modelo matemático	Convergencia de la función de costo (Fitness) del algoritmo PSO Error relativo porcentual (%) de los parámetros eléctricos identificados (R_s, R_r, X_{ls}, X_m) frente a los datos originales. Similitud gráfica de las curvas características del motor (Torque vs Deslizamiento, Corriente, Eficiencia y Factor de Potencia).	Función de costo: Converge a un mínimo estable (Aceptable) / No converge. Error relativo porcentual: $\leq 15\%$ (Aceptable) / $> 15\%$ (No aceptable). Ajuste de curvas características: Coincidente (Aceptable) / Divergente.	Comparación numérica de parámetros obtenidos y cálculo de error en MATLAB.

Variable	Definición Conceptual	Dimensiones (Fases)	Indicadores	Valores Posibles	Instrumento / Técnica de Evaluación
	simulación y en prototipo físico.	2. Tolerancia a fallas sensoriales en control vectorial	Seguimiento visual de la referencia de velocidad y corriente post-falla, comportamiento de la señal (presencia o ausencia de oscilaciones no controladas), y continuidad de corrientes estimadas.	Seguimiento post-falla: - Estable / Inestable Convergencia gráfica: - Converge / No converge	Análisis visual de gráficas de respuesta temporal (velocidad, corrientes y torque) ante inyección de fallas en MATLAB/Simulink. Observación gráfica señales: estimado vs referencia.
		3: Viabilidad de implementación física	Seguimiento continuo de la referencia de velocidad y estabilidad de corrientes de los ejes d (i_{sd}) y q (i_{sq}) durante la operación en tiempo real.	Seguimiento de comportamiento en hardware: Correcto / Incorrecto Estabilidad visual: - Estable / Inestable	Interfaz de monitoreo (ST Motor Pilot) y captura de señales en tiempo real para validación funcional. Osciloscopio / captura de señales en tiempo real

Anexo 2: Implementación de algoritmo PSO para determinación de parámetros de motor de inducción mediante MATLAB

Main

```
clear; clc; close all;
rng(10, 'twister') % Semilla fija para resultados reproducibles

% 1. Configuración del Sistema (Motor 1 HP / 0.75 kW)
V_fase = 220; % Voltaje de Fase (V)
f = 60; % Frecuencia (Hz)
P = 4; % Número de Polos
ws = (4 * pi * f) / P; % Velocidad síncrona (rad/s) [~188.5 rad/s]

% 2. Definición del Objetivo
% Parámetros REALES del algoritmo.
% [Rs (Estator), Rr (Rotor), Xs (Dispersión), Xm (Magnetización)]
actual = [9.6, 4.5, 9.42, 201.69];

% 3. Configuración del PSO (Límites de Búsqueda)
% Rango de búsqueda alrededor de los valores esperados.
lb = [5.0, 2.0, 5.0, 150]; % Límite Inferior
ub = [15.0, 8.0, 15.0, 250]; % Límite Superior

% 4. Ejecución del Algoritmo
fprintf('Iniciando optimización PSO...\n');

% Llamada a la función principal del PSO
% Parámetros: @fitness, limites, iteraciones=800, poblacion=100, coeficientes...
[solPSO, mejor_f_PSO, historial_error] = BasePSO( ...
    @fitness, lb, ub, 800, 100, 1.5, 2.0, 0.4, 0.9, actual);

% 5. Visualización de Resultados
% Gráfica de convergencia
Plots.costo_comparacion(historial_error);
% Comparación visual de curvas (Torque, Eficiencia, etc.)
Plots.performance(actual, solPSO, V_fase, ws);

% Reporte numérico final en consola
fprintf('\n--- RESULTADO FINAL DE ESTIMACIÓN ---\n');
fprintf('Parámetro | Obtenido | Esperado | Error (%%)\n');
fprintf('-----\n');
nombres = {'Rs ', 'Rr ', 'Xs ', 'Xm '};
for i = 1:4
    error_porc = abs(solPSO(i) - actual(i))/actual(i) * 100;
    fprintf(' %s | %8.4f | %8.4f | %6.4f%%\n', ...
        nombres{i}, solPSO(i), actual(i), error_porc);
end
fprintf('-----\n');
```

Función Fitness

% FUNCIÓN DE COSTO (FITNESS)

function F = fitness(x)

% Decodificar las variables que propone el PSO

Rs = x(1);

Rrd = x(2);

Xs = x(3);

Xm = x(4);

Xrd = Xs; % Reactancia del Rotor $\approx$ Reactancia Estator

% Datos del sistema

V = 220;

ws = 188.4956; %1800 RPM

s_nom = 0.038; % Deslizamiento nominal

% OBJETIVOS MATEMÁTICOS EXACTOS

% Valores de datos de brochure

Tfl_ref = 4.53; % Torque Nominal (Nm)

Tstart_ref = 6.74; % Torque de Arranque (Nm)

Tmax_ref = 11.92; % Torque Máximo (Nm)

pf_ref = 0.82; % Factor de Potencia

I_ref = 1.79; % Corriente de Fase (A)

% CÁLCULOS DEL MODELO (Candidato del PSO)

% 1. Impedancias Complejas (Modelo Exacto)

Z_s = Rs + 1i*Xs; % Impedancia Estator

Z_m = 1i*Xm; % Impedancia Magnetización

Z_r = (Rrd/s_nom) + 1i*Xrd; % Impedancia Rotor (ref. a estator)

% Impedancia Equivalente Total

Z_eq = Z_s + (Z_m * Z_r)/(Z_m + Z_r);

% 2. Cálculo de Corriente y Factor de Potencia

I_s_phasor = V / Z_eq;

I_s_mag = abs(I_s_phasor); % Magnitud Corriente

pf_calc = cos(angle(I_s_phasor)); % Factor de Potencia

% 3. Cálculo de Torque Nominal

Vm = V - I_s_phasor * Z_s; % Voltaje inducido en rama magnetización

Ir = Vm / Z_r; % Corriente del rotor

Tfl_calc = (3 * abs(Ir)^2 * (Rrd/s_nom)) / ws;

% 4. Cálculo de Torque de Arranque (s=1)

% Usamos aproximación Thevenin por estabilidad numérica

Vth = V * (Xm/(Xs+Xm));

Rth = Rs * (Xm/(Xs+Xm));

Xth = Xs * (Xm/(Xs+Xm));

k = 3 * Vth^2 / ws;

Tstart_calc = k * Rrd / ((Rth + Rrd)^2 + (Xth + Xrd)^2);

% 5. Cálculo de Torque Máximo

Tmax_calc = k / (2*(Rth + sqrt(Rth^2 + (Xth + Xrd)^2)));

% --- CÁLCULO DEL ERROR (COSTO) ---

% Suma de errores cuadráticos normalizados

e1 = ((Tfl_calc - Tfl_ref)/Tfl_ref)^2;

e2 = ((Tstart_calc - Tstart_ref)/Tstart_ref)^2;

```
e3 = ((Tmax_calc - Tmax_ref)/Tmax_ref)^2;
e4 = ((pf_calc - pf_ref)/pf_ref)^2;
e5 = ((I_s_mag - I_ref)/I_ref)^2;
```

% Peso total (Objetivo: Minimizar F a 0)

```
F = e1 + e2 + e3 + 2*e4 + e5;
```

End

Algoritmo PSO

% ALGORITMO PSO (PARTICLE SWARM OPTIMIZATION)

```
function [mejor_posicion, mejor_fitness, historial_error, estimacion] = BasePSO( ...
```

```
    fitness, lb, ub, iter, pob_size, c1, c2, w_min, w_max, actual)
```

% Inicialización de variables

```
nVar = length(lb);
```

```
v_max = 0.5 * (ub - lb);      % Velocidad máxima
```

```
v_min = -v_max;
```

% Inicialización aleatoria del enjambre (Posición y Velocidad)

```
v = rand(pob_size, nVar) .* (v_max - v_min) + v_min;
```

```
pob = cell(pob_size, 2); % Posición actual de cada partícula [Vector, Escalar]
```

```
pob_local = cell(pob_size, 2); % Mejor posición histórica (mejor_posicion)
```

[Vector, Escalar]

% Evaluar posición inicial de todas las partículas

```
for i = 1:pob_size
```

```
    x = lb + rand(1, nVar) .* (ub - lb);
```

```
    f = fitness(x);
```

```
    pob{i,1} = x; pob{i,2} = f;      % Posición actual
```

```
    pob_local{i,1} = x; pob_local{i,2} = f; % Mejor posición personal
```

(mejor_posicion)

```
end
```

% Encontrar el mejor global inicial (g_global)

```
[~, idx] = min(cell2mat(pob(:,2)));
```

```
g_mejor = pob(idx,:);
```

```
historial_error = zeros(1, iter);
```

% Bucle principal de optimización

```
for ep = 1:iter
```

% Factor de inercia dinámico (decrece linealmente)

```
w = (iter - ep) / iter * (w_max - w_min) + w_min;
```

```
for i = 1:pob_size
```

```
    r1 = rand(); r2 = rand();
```

% 1. Actualizar Velocidad (Inercia + Cognitiva + Social)

```
v(i,:) = w * v(i,:) ...
```

```
    + c1 * r1 * (pob_local{i,1} - pob{i,1}) ...
```

```
    + c2 * r2 * (g_mejor{1} - pob{i,1});
```

```

% 2. Actualizar Posición (con límites)
x_nuevo = max(min(pob{i,1} + v(i,:), ub), lb);

% 3. Evaluar Fitness
f_nuevo = fitness(x_nuevo);
pob{i,1} = x_nuevo; pob{i,2} = f_nuevo;

% 4. Actualizar pBest (Mejor personal)
if f_nuevo < pob_local{i,2}
    pob_local{i,1} = x_nuevo; pob_local{i,2} = f_nuevo;
end

% 5. Actualizar gBest (Mejor global) - Método asíncrono
if f_nuevo < g_mejor{2}
    g_mejor{1} = x_nuevo; g_mejor{2} = f_nuevo;
end
end

% Guardar historial
historial_error(ep) = g_mejor{2};
% Mostrar progreso cada 50 iteraciones
if mod(ep, 50) == 0
    fprintf('>> Iteracion: %4d | Error (Fitness): %.6f\n', ep, g_mejor{2});
end
end

% Retornar resultados
mejor_posicion = g_mejor{1};
mejor_fitness = g_mejor{2};
estimacion = mejor_posicion;
end

```

Graficas de perfomance

```

classdef Plots
    methods(Static)
        % Método principal para calcular curvas (Torque, Eficiencia, etc.)
        function [Te, eff, Pout, Is, pf] = parametros(params, V, ws)
            Rs=params(1); Rrd=params(2); Xs=params(3); Xm=params(4);
            s = [(1:999)/1000, 1]; % Rango completo (arranque)
            sfl = linspace(0.001, 0.08, 100); % Rango operación (nominal)
            % CÁLCULOS DE TORQUE
            Vth=V*(Xm/(Xs+Xm)); Rth=Rs*(Xm/(Xs+Xm))^2; Xth=Xs*(Xm/(Xs+Xm));
            k=3*(Vth^2)/ws;
            Te = k*(Rrd./s) ./ ((Rth + (Rrd./s)).^2 + (Xth+Xs).^2);
            % CÁLCULOS DE EFICIENCIA Y PF
            Zr=(Rrd./sfl)+1i*Xs; Zs=Rs+1i*Xs; Zm=1i*Xm;
            Zeq=Zs+(Zm.*Zr)./(Zm+Zr);
            Is_c=V./Zeq;
            Is=abs(Is_c);

```

```

    pf=cos(angle(Is_c));
    Vm=V-Is_c.*Zs; Ir=Vm./Zr;
    P_gap=3*abs(Ir).^2.*(Rrd./sfl);
    Pout=P_gap.*(1-sfl);
    Pin=3*V.*Is.*pf;
    eff=(Pout./Pin)*100;
end

% Gráfica 1: Convergencia del costo
function costo_comparacion(cost1)
    figure;
    plot(cost1, 'm', 'LineWidth', 1.5);
    grid on; title('Evolución del Error (Fitness)');
    xlabel('Iteraciones'); ylabel('Costo');
end

% Gráfica 2: Generación de todas las curvas comparativas
function performance(actual, soll, V, ws)
    [Te, eff, Pout, Is, pf] = Plots.parametros(actual, V, ws);
    [Te1, eff1, Pout1, Is1, pf1] = Plots.parametros(soll, V, ws);

    Plots.slip_torque(Te, Te1);
    Plots.eficiencia(Pout, eff, Pout1, eff1);
    Plots.corriente(Pout, Is, Pout1, Is1);
    Plots.factor_de_potencia(Pout, pf, Pout1, pf1);
end

% Sub-gráficas individuales
function slip_torque(Te, Te1)
    figure; plot(Te, 'b--', 'LineWidth', 1.5); hold on; plot(Te1, 'r', 'LineWidth', 1.5);
    set(gca, 'XDir', 'reverse'); legend('Deseado', 'Estimado PSO', 'Location', 'best');
    grid on; title('Torque vs Deslizamiento'); xlabel('Deslizamiento (s)');
    ylabel('Torque (Nm)');
end

function eficiencia(Pout, eff, Pout1, eff1)
    figure; plot(Pout, eff, 'b--', 'LineWidth', 1.5); hold on; plot(Pout1, eff1, 'r',
'LineWidth', 1.5);
    legend('Deseado', 'Estimado PSO', 'Location', 'best'); grid on;
    title('Eficiencia'); xlabel('Potencia (W)'); ylabel('Eficiencia (%)');
end

function corriente(Pout, Is, Pout1, Is1)
    figure; plot(Pout, Is, 'b--', 'LineWidth', 1.5); hold on; plot(Pout1, Is1, 'r',
'LineWidth', 1.5);
    legend('Deseado', 'Estimado PSO', 'Location', 'best'); grid on;
    title('Corriente Estator'); xlabel('Potencia (W)'); ylabel('Corriente (A)');
end

function factor_de_potencia(Pout, pf, Pout1, pf1)
    figure; plot(Pout, pf, 'b--', 'LineWidth', 1.5); hold on; plot(Pout1, pf1, 'r',
'LineWidth', 1.5);
    legend('Deseado', 'Estimado PSO', 'Location', 'best'); grid on;

```

```

title('Factor de Potencia'); xlabel('Potencia (W)'); ylabel('FP');
end
end
end

```

Anexo 3: Modelo de motor de inducción en Simulink

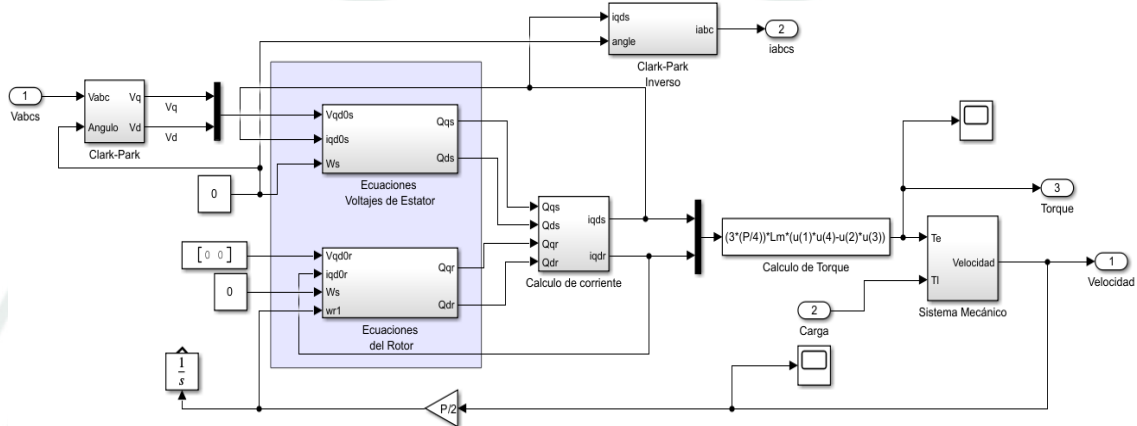


Figura 109. Modelo de motor de inducción en marco de referencia estacionario

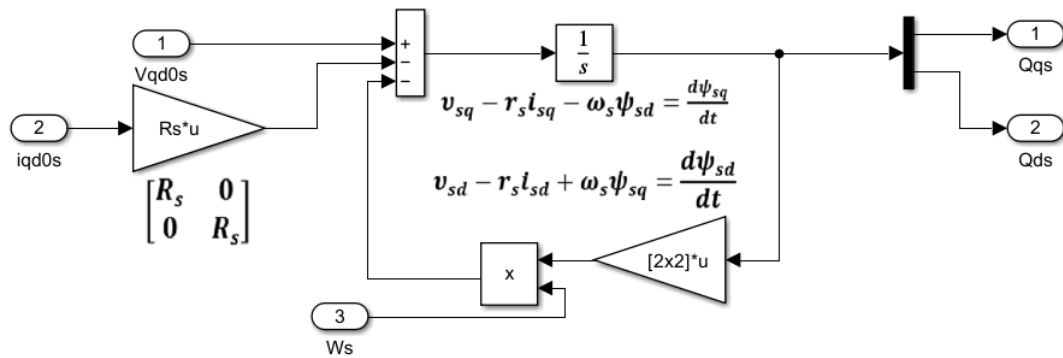


Figura 110. Ecuaciones de voltajes del estator

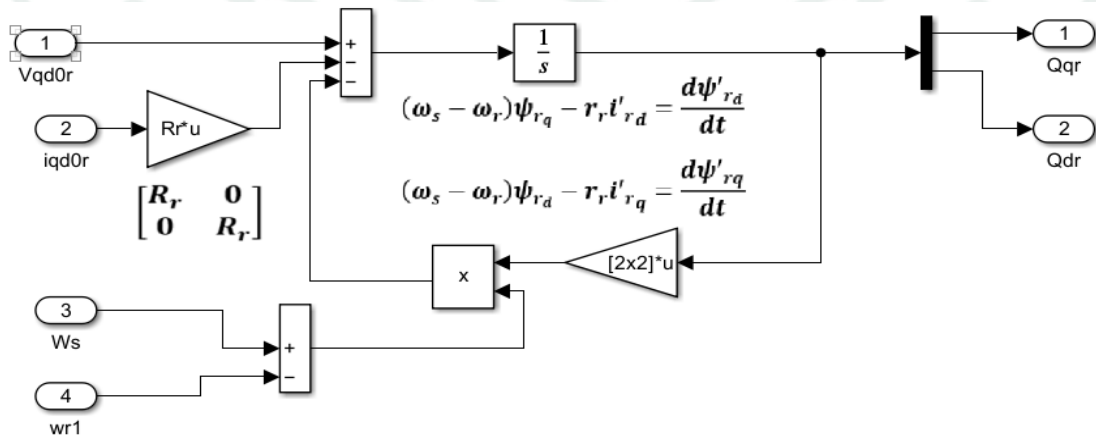


Figura 111. Ecuaciones de voltajes del rotor

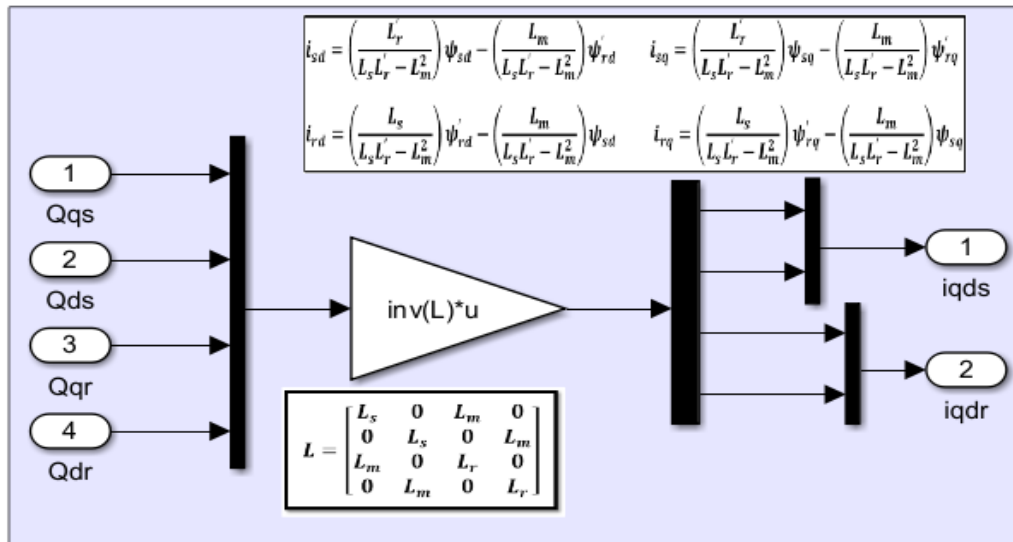


Figura 112. Cálculo de corrientes

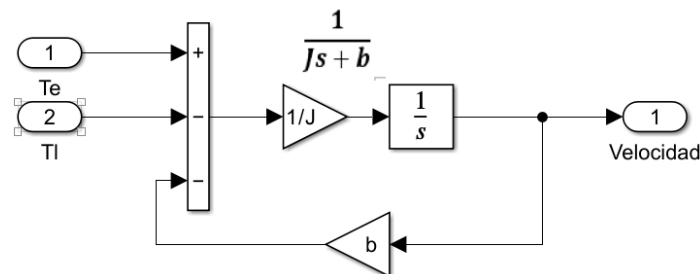


Figura 113. Ecuación mecánica

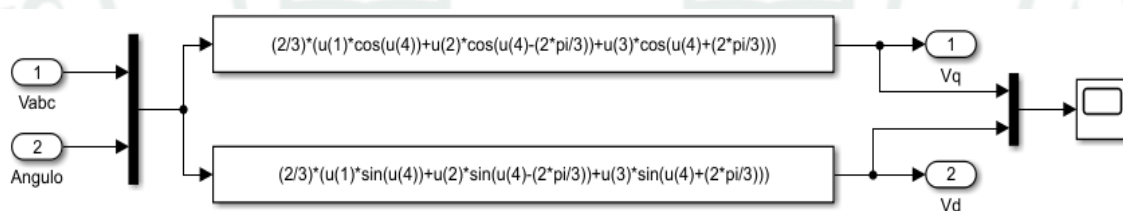


Figura 114. Transformada Clarke – Park

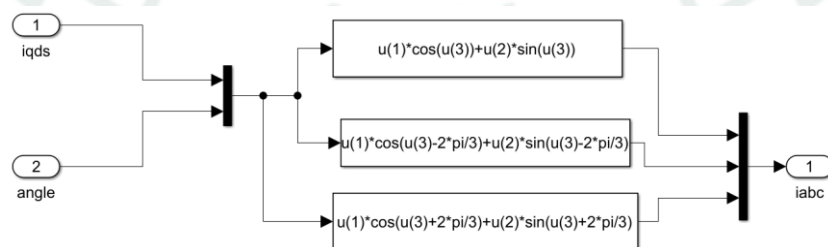


Figura 115. Transformada Clarke – Park inverso

Anexo 4: Implementación de control de campo orientado de motor de inducción en SIMULINK

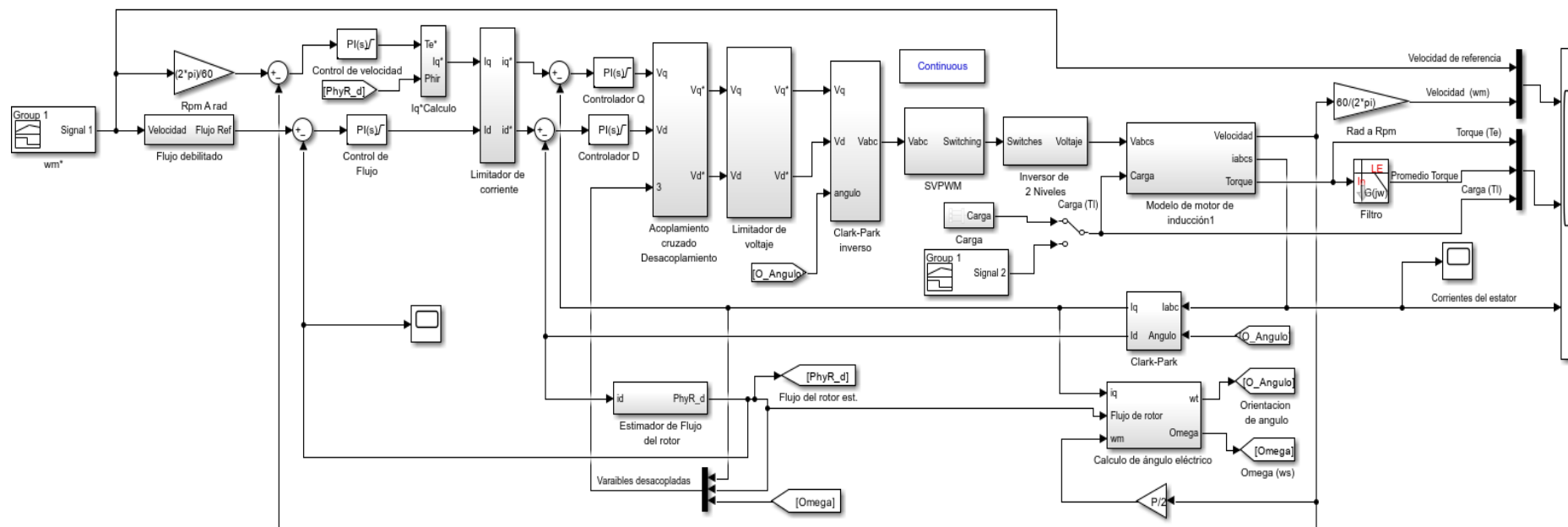


Figura 116. Control vectorial de motor de inducción

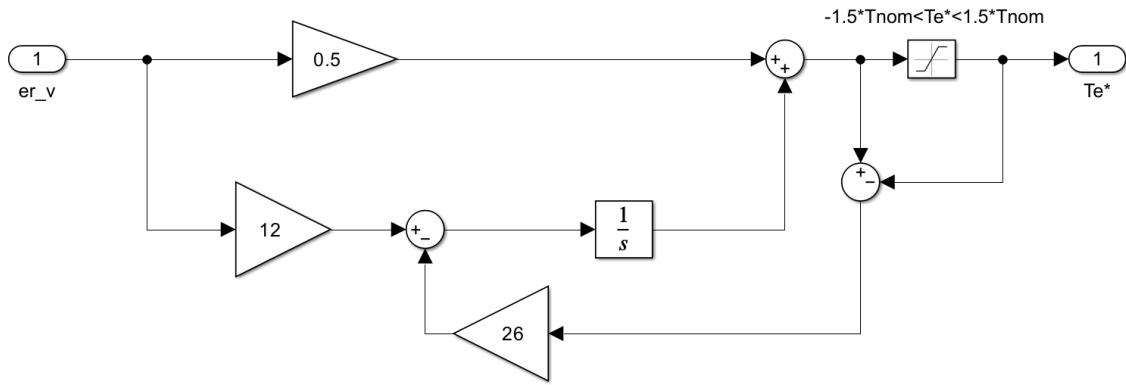


Figura 117. Control PI anti-windup de velocidad

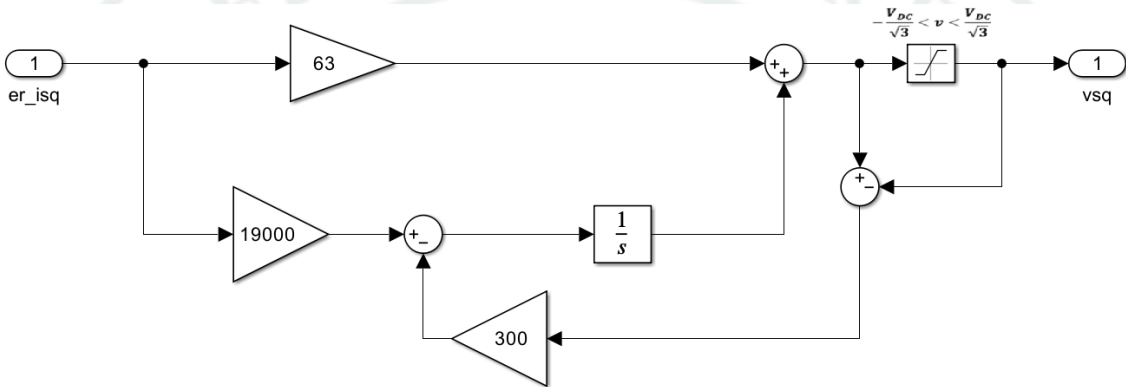


Figura 118. Control PI anti-windup de corrientes

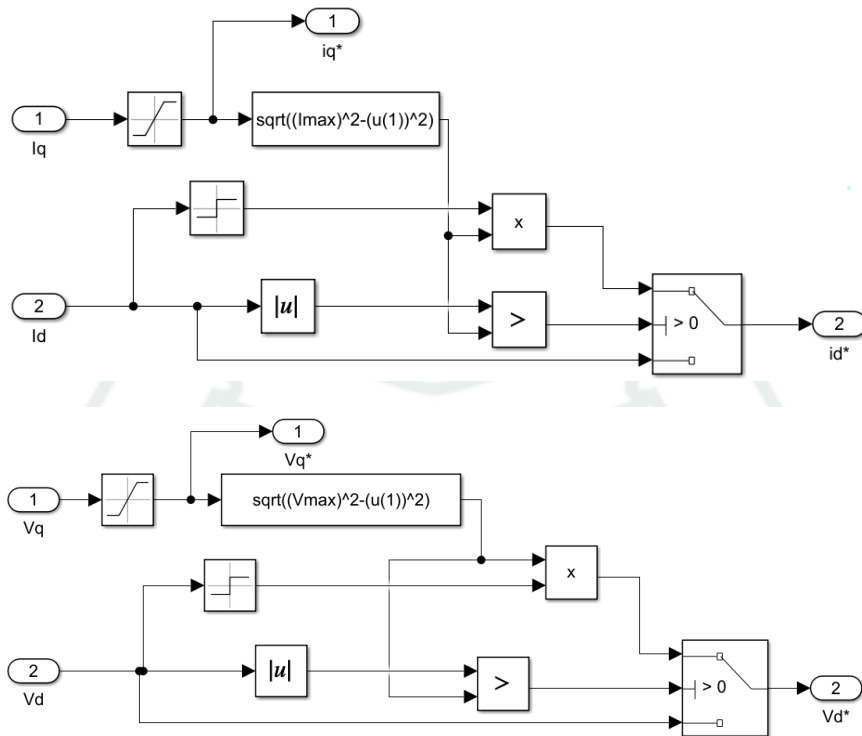


Figura 119. Limitador de corriente y voltaje

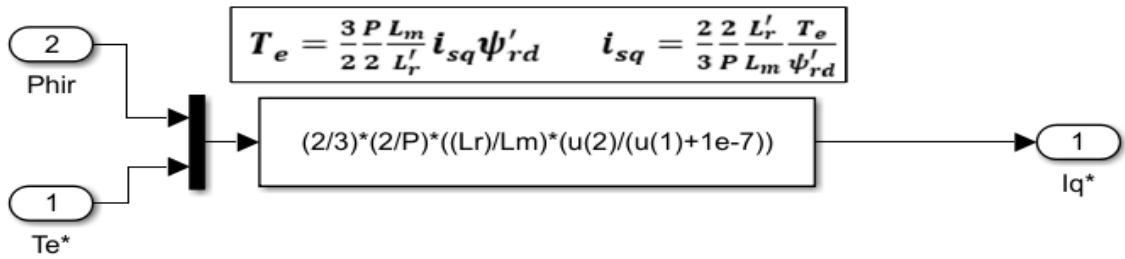


Figura 120. Cálculo de isq a partir de Te

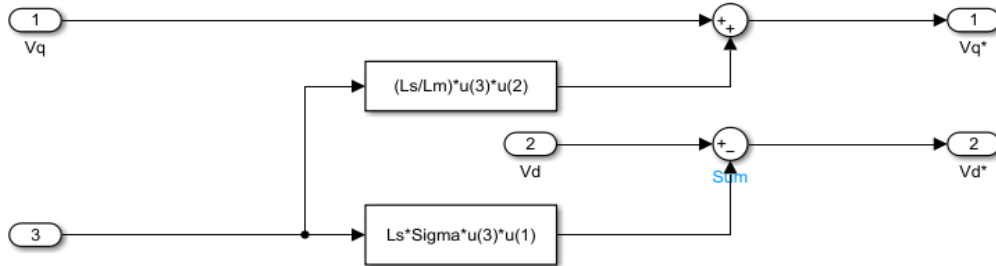


Figura 121. Acoplamiento cruzado

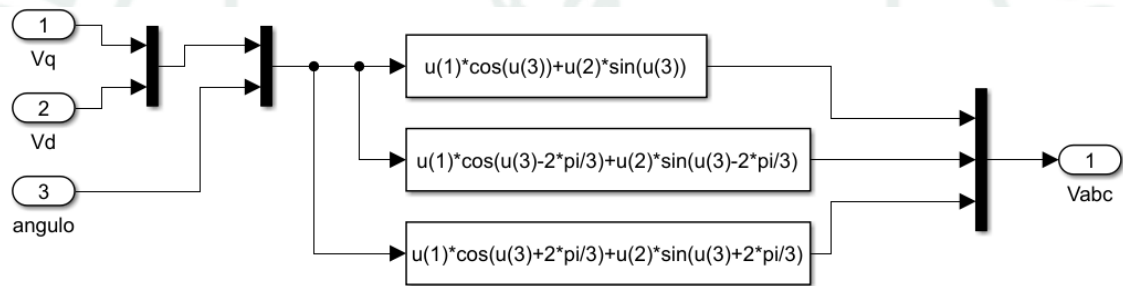


Figura 122. Transformada de Clarke – Park inverso

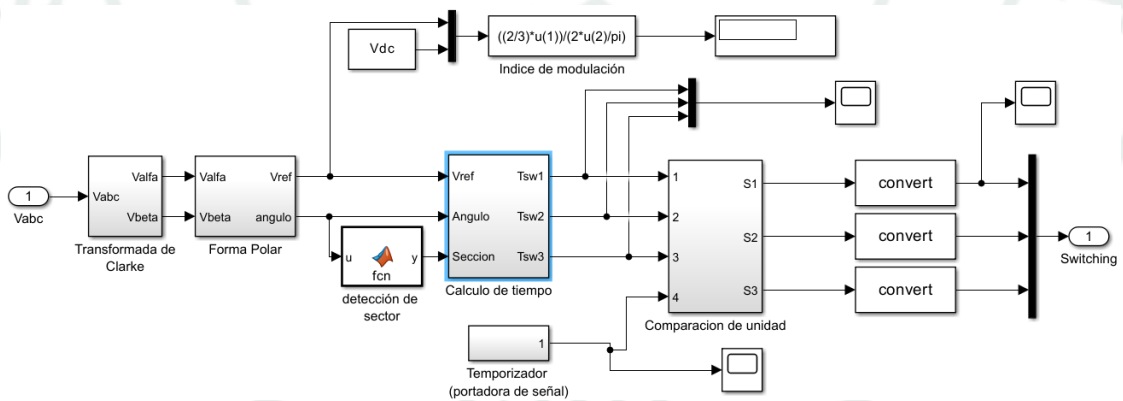


Figura 123. Modulación vectorial

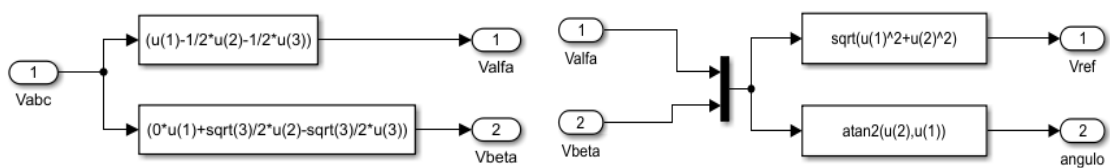


Figura 124. Transformada de Clarke y Forma polar

```

function y = fcn(u)
    u= 360/(2*pi)*u ;
    if (u>=0)&&(u<60)
        u = 1 ;
    elseif (u>=60)&&(u<120)
        u = 2 ;
    elseif (u>=120)&&(u<180)
        u = 3 ;
    elseif (u>=-180)&&(u<-120)
        u = 4 ;
    elseif (u>=-120)&&(u<-60)
        u = 5 ;
    elseif (u>=-60)&&(u<0)
        u = 6 ;
    else
        u=0 ;
    end
    y = u;

```

Detección de sector

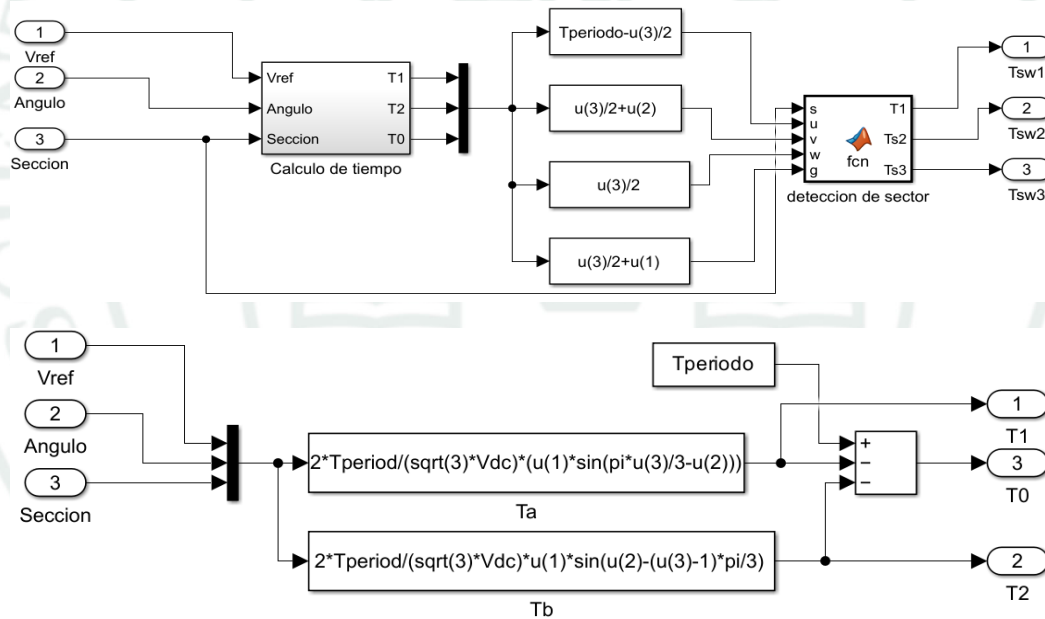


Figura 125. Cálculo de tiempos

Detección de sector

```

function [T1,Ts2,Ts3] = fcn(s,u,v,w,g)
switch s
case 1
    T1 = u ;
    Ts2 = v ;
    Ts3 = w ;
case 2
    T1 = g ;

```

```

Ts2 = u ;
Ts3 = w ;
case 3
T1 = w ;
Ts2 = u ;
Ts3 = v ;
case 4
T1 = w ;
Ts2 = g ;
Ts3 = u ;
case 5
T1 = v ;
Ts2 = w ;
Ts3 = u ;
case 6
T1 = u ;
Ts2 = w ;
Ts3 = g ;
otherwise
T1 = 0 ;
Ts2 = 0 ;
Ts3 = 0 ;
end
end

```

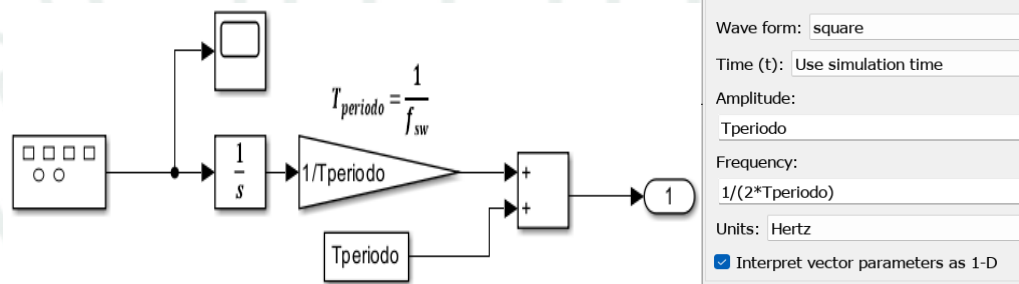


Figura 126. Temporizador (portadora de señal)

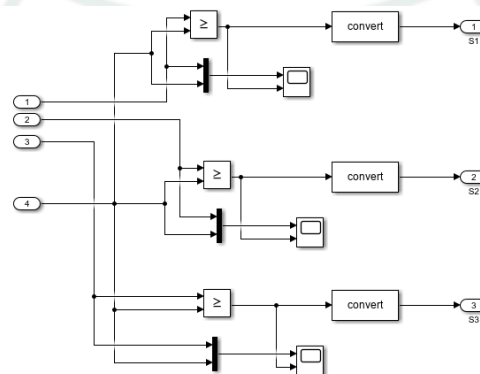


Figura 127. Comparación de sector

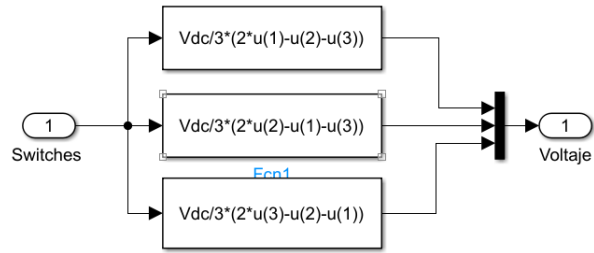


Figura 128. Inversor de 2 Niveles

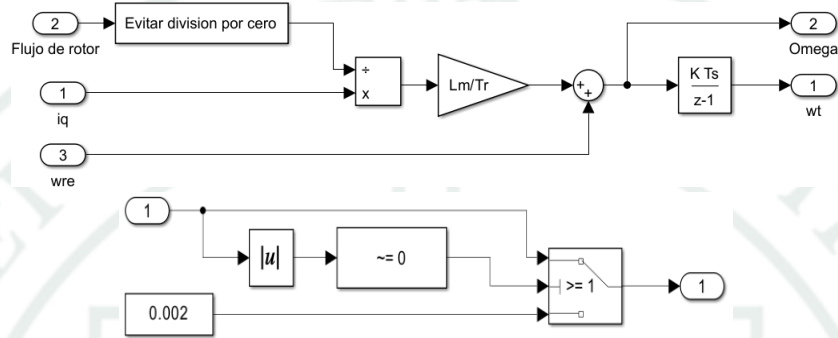


Figura 129. Estimación de ángulo eléctrico

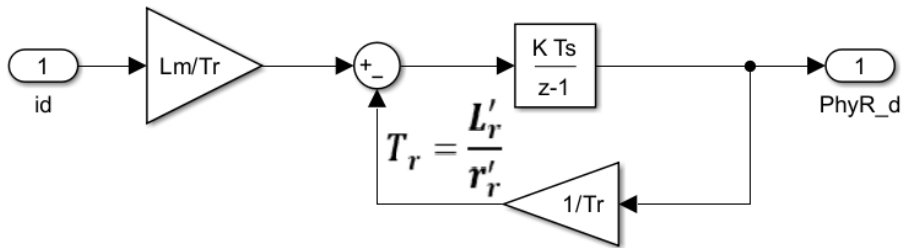


Figura 130. Estimación de flujo de rotor

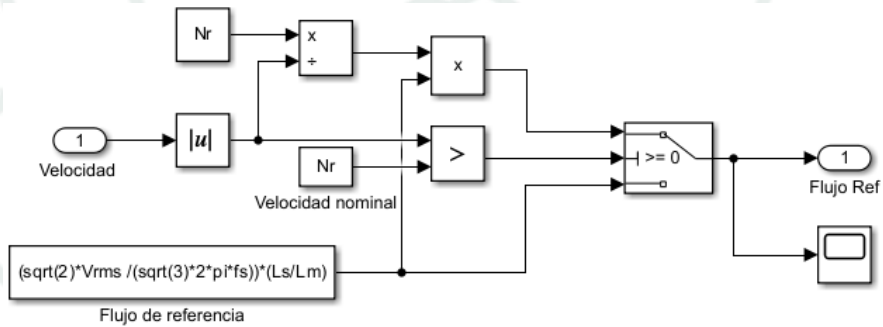


Figura 131. Control por flujo debilitado

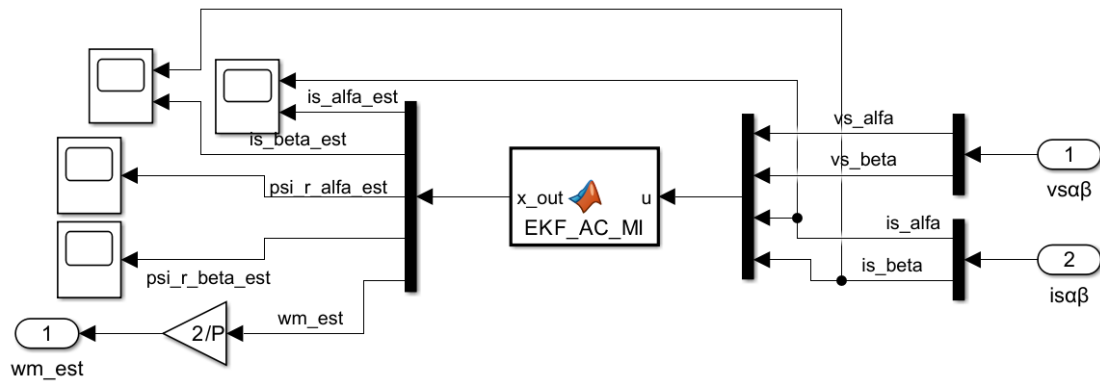


Figura 133. Filtro de Kalman Extendido (Input – Output)

Código Filtro de kalman extendido

```
function x_out = EKF_AC_MI(u)
% Filtro de Kalman Extendido (EKF) para un Motor de Inducción
% El EKF estima las corrientes del estator, los flujos del rotor y la
% velocidad mecánica del motor a partir de un modelo no lineal reducido.
%% Parámetros del motor
J = 0.0032; % Momento de inercia del rotor
b = 0.0001; % Coeficiente de fricción viscosa
p = 2; % Número de pares de polos
Lls = 25e-3; % Inductancias de dispersión del estator
Llr = 25e-3; % Inductancias de dispersión del rotor
Lm = 535e-3; % Inductancia mutua
Ls = Lm + Lls; % Inductancia del estator
Lr = Lm + Llr; % Inductancia del rotor
% Resistencias de estator y rotor
Rs = 9.6;
Rr = 4.5;
% Factor de dispersión sigma
sigma = 1 - (Lm^2)/(Lr*Ls);
% Término Δ del modelo eléctrico reducido
Delta = -((Lm^2 * Rr + Lr^2 * Rs) / (sigma * Ls * Lr^2));
% Tiempo de muestreo
Ts = 1e-4;
%% Matrices de ruido del EKF
% La matriz Q modela el ruido del modelo → qué tan incierto se considera
% al modelo matemático del motor.
Q = diag([1e-3, 1e-3, 1e-4, 1e-4, 60]);
% La matriz R modela el ruido de medición → incertidumbre en los sensores
R = diag([0.8 0.8]);
%% Variables persistentes
% Se ejecutan una sola vez al iniciar la simulación
persistent x P
if isempty(x)
    % Estado inicial: [isa isb psi_ra psi_ra wr]
    x = [0; 0; 0.1; 0; 0];
    % Matriz de covarianza inicial (incertidumbre del estado)
```

```

P = diag([1, 1, 10, 10, 500]);
end
%%Lectura de entradas del vector u
% u = [v_alpha; v_beta; i_alpha; i_beta]
vs_ab = u(1:2);      % Vector de tensiones alfa-beta
vs_ab = vs_ab(:);     % Garantizar formato columna
is_ab_medido = u(3:4); % Corrientes medidas en alfa-beta
is_ab_medido = is_ab_medido(:);
%% Matriz de medición
H = [1 0 0 0 0;
     0 1 0 0 0];
%% Matriz de control del modelo
B_control = [1/(sigma*Ls) 0;
             0 1/(sigma*Ls);
             0 0;
             0 0;
             0 0];
%% ETAPA DE PREDICCIÓN
% Se calcula el modelo dinámico linealizado alrededor del estado actual
Fc_at_x = [Delta, 0, Lm*Rr/(sigma*Ls*Lr^2), Lm*x(5)/(sigma*Ls*Lr), 0;
           0, Delta, -Lm*x(5)/(sigma*Ls*Lr), Lm*Rr/(sigma*Ls*Lr^2), 0;
           Lm*Rr/Lr, 0, -Rr/Lr, -x(5), 0;
           0, Lm*Rr/Lr, x(5), -Rr/Lr, 0;
           0, 0, 0, 0, 0];
% Predicción del estado (ecuación del modelo)
X_pred = (eye(5) + Ts*Fc_at_x)*x + Ts*B_control*vs_ab;
%% Matriz Jacobiana A evaluada en el estado predicho
Ac_at_Xpred = [Delta, 0, Lm*Rr/(sigma*Ls*Lr^2), Lm*X_pred(5)/(sigma*Ls*Lr),
               Lm*X_pred(4)/(sigma*Ls*Lr);
               0, Delta, -Lm*X_pred(5)/(sigma*Ls*Lr), Lm*Rr/(sigma*Ls*Lr^2), -
               Lm*X_pred(3)/(sigma*Ls*Lr);
               Lm*Rr/Lr, 0, -Rr/Lr, -X_pred(5), -X_pred(4);
               0, Lm*Rr/Lr, X_pred(5), -Rr/Lr, X_pred(3);
               0, 0, 0, 0, 0];
% Matriz Phi = Jacobiano discretizado
Phi_k = eye(5) + Ts*Ac_at_Xpred;
% Predicción de la covarianza P
P_pred = Phi_k * P * Phi_k' + Q;
%% ETAPA DE CORRECCIÓN
% Comparación entre la predicción y la medición real
Y_pred = H * X_pred;      % Corrientes predichas
Y = is_ab_medido;         % Corrientes medidas por los sensores
innovacion = Y - Y_pred;  % Error de innovación
S_k = H * P_pred * H' + R; % Covarianza de la innovación
K = P_pred * H' / S_k;    % Ganancia de Kalman
x = X_pred + K * innovacion; % Actualización del estado
P = (eye(5) - K * H) * P_pred; % Actualización de la covarianza
%% Salida
x_out = x;
en

```

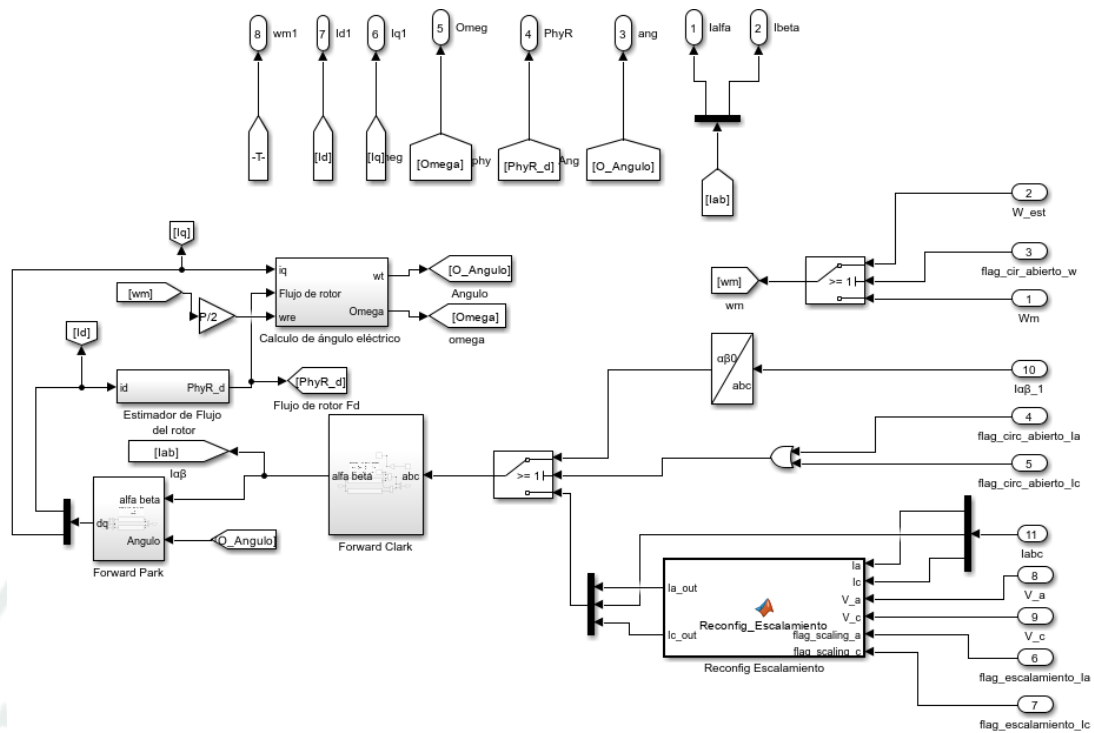



Figura 135. Sistema de reconfiguración

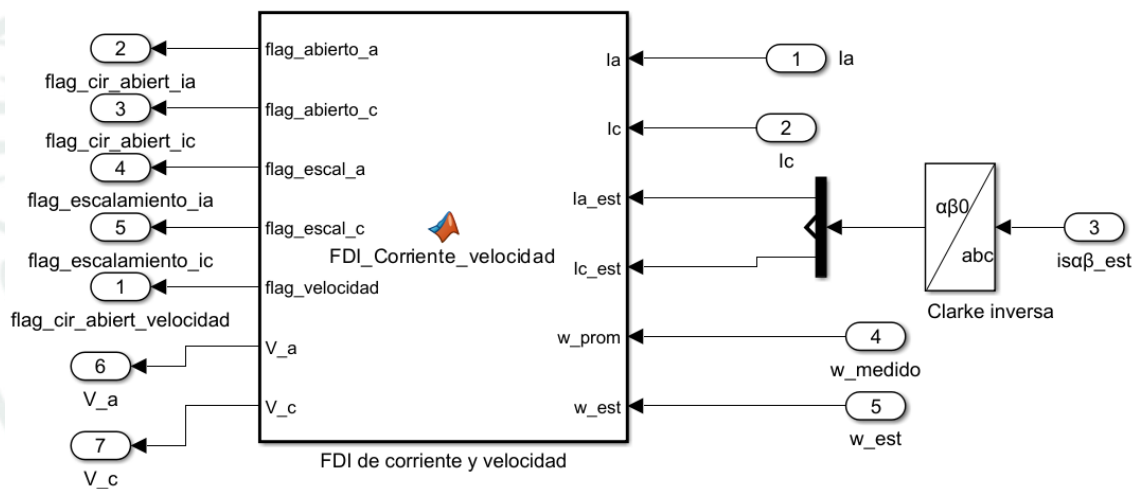


Figura 136. Bloque Sistema de diagnóstico de falla

Código de detección de fallas

```
function [flag_abierto_a, flag_abierto_c, flag_escal_a, flag_escal_c, flag_velocidad, V_a, V_c] = ...
    FDI_Corriente_velocidad(Ia, Ic, Ia_est, Ic_est, w_prom, w_est)
% Detecta:
% - Fallas de circuito abierto en los sensores de corriente Ia e Ic
% - Fallas de escala (error de ganancia o calibración) en sensores de corriente Ia e Ic
% - Falla en el sensor de velocidad (comparando medida real vs estimada)
```

%% Parámetros

```

persistent Ia_hist Ic_hist Ia_est_hist Ic_est_hist
persistent w_prom_hist w_est_hist

```

```

L = 100;           % Longitud de ventana deslizante
K0 = 0.3;          % Umbral para detección de corriente cercana a cero
velocidad_umbral = 0.05; % Umbral relativo de diferencia de velocidad (5%)

```

```

% Inicialización de buffers (solo la primera vez que se ejecuta)

```

```

if isempty(Ia_hist)
    Ia_hist = zeros(1,L);
    Ic_hist = zeros(1,L);
    Ia_est_hist = zeros(1,L);
    Ic_est_hist = zeros(1,L);
    w_prom_hist = zeros(1,L);
    w_est_hist = zeros(1,L);
end

```

```

%% Actualización de las ventanas deslizantes

```

```

Ia_hist = [Ia_hist(2:end), Ia];
Ic_hist = [Ic_hist(2:end), Ic];
Ia_est_hist = [Ia_est_hist(2:end), Ia_est];
Ic_est_hist = [Ic_est_hist(2:end), Ic_est];
w_prom_hist = [w_prom_hist(2:end), w_prom];
w_est_hist = [w_est_hist(2:end), w_est];

```

```

%% 1. Detección de circuito abierto

```

```

% Se verifica si la corriente permanece cerca de cero durante la mayoría del tiempo

```

```

m_a = abs(Ia_hist) <= K0;
m_c = abs(Ic_hist) <= K0;

```

```

M_a = sum(m_a)/L; % Proporción de muestras cercanas a cero
M_c = sum(m_c)/L;

```

```

%% 2. Cálculo de índice V_n para falla de escala ==

```

```

% Se compara la RMS de la corriente medida vs la estimada

```

```

% 1e-10 evita divisiones por cero

```

```

V_a = rms(Ia_hist) / (rms(Ia_est_hist) + 1e-10);
V_c = rms(Ic_hist) / (rms(Ic_est_hist) + 1e-10);

```

```

%% 3. Lógica de decisión ==

```

```

% Fase A

```

```

if M_a >= 0.8

```

```

    % Si más del 80% de las muestras están cerca de cero, se asume circuito abierto

```

```

    flag_abierto_a = 1;
    flag_escal_a = 0;

```

```

else

```

```

    flag_abierto_a = 0;

```

```

    % Si la relación RMS medida/estimada está fuera del ±5%, hay falla de escala

```

```

    if (V_a < 0.95 || V_a > 1.05)

```

```

        flag_escal_a = 1;
    end
end

```

```

else
    flag_escal_a = 0;
end
end

% --- Fase C ---
if M_c >= 0.8
    flag_abierto_c = 1;
    flag_escal_c = 0;
else
    flag_abierto_c = 0;
    if (V_c < 0.95 || V_c > 1.05)
        flag_escal_c = 1;
    else
        flag_escal_c = 0;
    end
end

%% 4. Detección de falla en el sensor de velocidad ==
% Se compara la diferencia promedio entre la velocidad medida y la estimada
velocidad_ratio = mean(abs(w_prom_hist - w_est_hist)) / (mean(abs(w_est_hist)) + 1e-10);

if velocidad_ratio > velocidad_umbral
    flag_velocidad = 1; % Falla detectada
else
    flag_velocidad = 0;
end
end

```

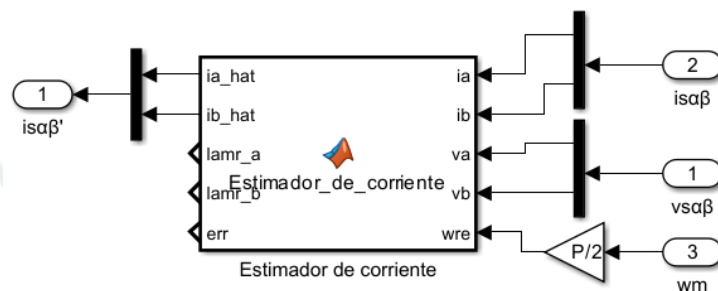


Figura 137. Bloque de Estimación de corrientes $i_{s\alpha}$, $i_{s\beta}$

Código de Estimación de corrientes $i_{s\alpha}$, $i_{s\beta}$

```

function [ia_hat, ib_hat, lamr_a, lamr_b, err] = ...
    Estimador_de_corriente(ia, ib, va, vb, wre)
% Estimador de corrientes y flujo del rotor (modelo ideal)
persistent Rs Rr Ls Lr Lm Ts
if isempty(Rs)
    % PARAMETROS DEL MOTOR
    Lls = 25e-3; Llr = 25e-3; Lm = 535e-3;
    Ls = Lm + Lls;

```

```

Lr = Lm + Llr;
Rs = 9.67; Rr = 4.5;
Ts = 1e-4;
end
% ESTADOS: flujo de rotor en  $\alpha\beta$ 
persistent lamr_a_k lamr_b_k
if isempty(lamr_a_k); lamr_a_k = 0; end
if isempty(lamr_b_k); lamr_b_k = 0; end
% PRECALCULOS
sigma = (1 - (Lm^2)/(Ls*Lr)); % coeficiente de fuga
Tr = Lr / Rr; % constante tiempo rotor
% Coeficientes discretizados
a1 = (1 - Ts*(Rs*Lr^2+Rr*Lm^2)/(sigma*Ls*Lr^2));
a2 = (1 - (Ts/Tr));
b1 = Ts*Lm/(sigma*Ls*Lr);
b2 = Ts*wre;
c1 = Ts/(sigma*Ls);
c2 = Lm*Ts/Tr;
% ACTUALIZACION DE FLUJOS
lamr_a = c2*ia + a2*lamr_a_k - b2*lamr_b_k;
lamr_b = c2*ib + a2*lamr_b_k + b2*lamr_a_k;
% ESTIMACION DE CORRIENTES
ia_hat = a1*ia + b1/Tr*lamr_a + b1*wre*lamr_b + c1*va;
ib_hat = a1*ib + b1/Tr*lamr_b - b1*wre*lamr_a + c1*vb;
% ACTUALIZA ESTADOS
lamr_a_k = lamr_a;
lamr_b_k = lamr_b;
err=ia-ia_hat;
end

```

Codigo: Reconfiguración en caso falla por ganancia, calibración

```

function [Ia_corr, Ic_corr] = Reconfig_Escalamiento(Ia, Ic, V_a, V_c, ...
    flag_escal_a, flag_escal_c)
% Reconfiguración para fallas de escala (scaling) en sensores de corriente.
% - Usa histéresis + persistencia para evitar activaciones falsas.
% Entradas:
% Ia, Ic    -> corrientes medidas
% V_a, V_c  -> índice de escala (RMS medida / RMS estimada)
% flag_escal_a -> flag del FDI (1 = falla en Ia)
% flag_escal_c -> flag del FDI (1 = falla en Ic)
% Salidas:
% Ia_out, Ic_out -> corrientes corregidas
%% Parámetros de persistencia e histéresis
N_on = 500; % N° de muestras necesarias para ACTIVAR corrección (5 ms aprox)
M_off = 1500; % N° de muestras para DESACTIVAR corrección (histeresis larga)

%% Variables persistentes (se mantienen entre iteraciones)
persistent cntA cntC estadoA estadoC
if isempty(cntA)

```

```

% Inicialización la primera vez que se ejecuta el bloque
cntA = 0;
cntC = 0;

% estadoA/estadoC = estado estable final:
% 0 = normal
% 1 = reconfiguración activa
estadoA = 0;
estadoC = 0;
end

%% LÓGICA DE HISTÉRESIS – FASE A
if flag_escala == 1
    % El FDI detecta falla → contar hacia +N_on
    cntA = cntA + 1;
    if cntA >= N_on
        estadoA = 1;    % activar reconfiguración estable
    end
else
    % FDI dice que NO hay falla → contar hacia -M_off
    cntA = cntA - 1;
    % Si el contador es muy negativo → desactivar reconfiguración
    if cntA <= -M_off
        estadoA = 0;
        cntA = 0;    % Reset para evitar valores extremos
    end
end

%% LÓGICA DE HISTÉRESIS – FASE C
if flag_escala_c == 1
    cntC = cntC + 1;
    if cntC >= N_on
        estadoC = 1;
    end
else
    cntC = cntC - 1;
    if cntC <= -M_off
        estadoC = 0;
        cntC = 0;
    end
end

%% COMPENSACIÓN FINAL DE ESCALAMIENTO
% --- Fase A ---
if estadoA == 1
    % Compensación: dividir la corriente por el factor de escala
    Ia_corr = Ia / (V_a + 1e-10); % 1e-10 evita división por 0
else
    Ia_corr = Ia;
end

```

```

% --- Fase C ---
if estadoC == 1
    Ic_corr = Ic / (V_c + 1e-10);
else
    Ic_corr = Ic;
end
end

```

Código de compensación por falla por offset

```

function [Ia_corr, Ic_corr, flag_Ia, flag_Ic] = CompensadorOffset(Ia, Ic)
% CompensadorOffset
% Método de dos etapas basado en medias móviles para:
% - Detectar offset en sensores de corriente (Ia, Ic)
% - Compensar dicho offset restando la componente DC estimada
% * Primera media móvil de ventana L → O[n]
% * Segunda media móvil sobre O[n] → S[n]
% * Si |S[n]| supera el umbral → existe offset
%% Parámetros del sistema
Ts = 1e-5; % Tiempo de muestreo [s]
Irated = 3.07; % Corriente nominal [A]
p = 2; % Número de pares de polos
n_rpm = 1800; % Velocidad mecánica del motor [rpm]

%% Parámetros ajustables
Nperiods = 3; % N° de periodos eléctricos usados para la ventana
Koff_frac = 1/12; % Factor del umbral relativo al Irated
Koff = Koff_frac * Irated;

%% Cálculo de tamaño de ventana
fe = (n_rpm/60)*p; % Frecuencia eléctrica [Hz]
T = 1/fe; % Periodo eléctrico [s]

% Ventana L ≈ N periodos → suficiente para filtrar la onda senoidal
L = max(64, round(Nperiods*T/Ts));

%% Variables persistentes
persistent ia_buf ic_buf % buffers para medias móviles
persistent Oa Oc Sa Sc % medias O[n], O'[n], S[n]
persistent Oa_buf Oc_buf % buffers para medias sobre O
persistent init

if isempty(init)
    ia_buf = zeros(1,L);
    ic_buf = zeros(1,L);

    Oa = 0; Oc = 0; % Primera media (O[n])
    Oa_buf = zeros(1,L);
    Oc_buf = zeros(1,L);

```

```

    Sa = 0; Sc = 0;      % Segunda media (S[n])
    init = true;
end

%% Actualización de buffers
old_ia = ia_buf(1); % muestra más antigua de Ia
old_ic = ic_buf(1); % muestra más antigua de Ic
% Desplazar ventana: eliminar la muestra antigua y agregar la nueva
ia_buf = [ia_buf(2:end), Ia];
ic_buf = [ic_buf(2:end), Ic];

%% Primera media móvil O[n]
% Implementación eficiente de la media móvil:
%  $O[n] = O[n-1] + (nueva - vieja)/L$ 
Oa = Oa + (Ia - old_ia)/L;
Oc = Oc + (Ic - old_ic)/L;

%% Actualizar buffers para O[n]
old_Oc = Oc_buf(1);

Oa_buf = [Oa_buf(2:end), Oa];
Oc_buf = [Oc_buf(2:end), Oc];

%% Segunda media móvil S[n] (índice de detección)
Sa = Sa + (Oa - old_Oa)/L;
Sc = Sc + (Oc - old_Oc)/L;
%% === Detección de offset ===
flag_Ia = abs(Sa) > Koff; % true si existe offset en Ia
flag_Ic = abs(Sc) > Koff; % true si existe offset en Ic
%% === Compensación directa
% Se resta O[n]
if flag_Ia
    Ia_corr = Ia - Oa;
else
    Ia_corr = Ia;
end

if flag_Ic
    Ic_corr = Ic - Oc;
else
    Ic_corr = Ic;
end
end

```

Anexo 7: Estabilidad de Lyapunov

La teoría de Lyapunov es un método genérico y poderoso para analizar la estabilidad de sistemas dinámicos, aplicable tanto a sistemas lineales como a no lineales y variantes en el tiempo. Se basa en la observación física de que, si la energía de un sistema mecánico se disipa hasta cero, el sistema es estable. El análisis se realiza mediante una "Función de Energía" escalar (Figura 138), $V(x)$, que debe cumplir dos condiciones fundamentales para garantizar la estabilidad asintótica:

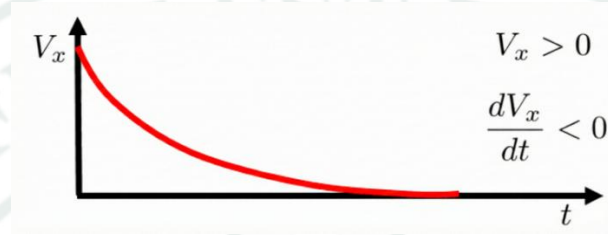


Figura 138. Función de Energía

- **Energía Positiva:** La función $V(x)$ debe ser mayor que cero para todo estado diferente de cero ($V(x) > 0$).
- **Disipación de Energía:** La derivada de la función respecto al tiempo debe ser siempre negativa ($\dot{V}(x) < 0$) lo que indica que la energía del sistema decrece constantemente.

Para sistemas lineales ($\dot{x} = Ax$), se utiliza comúnmente una función cuadrática candidata $V(x) = x^T M x$. La estabilidad se determina resolviendo la Ecuación de Lyapunov ($A^T M + M A = -N$), donde se busca encontrar una matriz M simétrica y definida positiva, asumiendo generalmente que N es la matriz identidad.

Introducción a la Función Candidata en el Motor de Inducción

Para aplicar esta teoría al Observador de Luenberger Adaptativo para estimación de velocidad del motor de inducción, se modifica la función general cuadrática ($x^T M x$) para incluir no solo los estados eléctricos, sino también el parámetro desconocido (la velocidad).

Esta elección implica dos adaptaciones importantes respecto a la teoría general:

$$V = e^T e + \frac{(\omega_r - \hat{\omega}_r)^2}{\gamma} = e^T e + \frac{(\Delta\omega_r)^2}{\gamma}$$

1. **Simplificación de la Matriz:** Se asume que la matriz M es la Matriz Identidad (I), simplificando el término $x^T M x$ directamente a $e^T e$ (suma de los cuadrados de los errores de corriente y flujo).
2. **Extensión para Adaptación:** Se añade el segundo término $\frac{(\Delta\omega_r)^2}{\gamma}$ para representar la "energía" del error de estimación de velocidad.

El objetivo matemático es demostrar que la derivada de esta función específica es negativa, lo cual fuerza al algoritmo a encontrar la velocidad real del motor para disipar esa energía extra.

Anexo 8: Código fuente de lazos de control de corriente, velocidad y el observador de estado LSO

Implementación de Observador LSO

```
/*
 * LICENCIA
 *
 * @attention
 * <h2><center>&copy; Copyright (c) 2023 STMicroelectronics.</center></h2>
 * This software component is licensed by ST under Ultimate Liberty license SLA0044.
 *
 **
 * @brief Estimación de ángulo y velocidad mediante Observador de Luenberger.
 * @note Código refactorizado para tesis
 */
void ACIM_LSO_CalcAngle(ACIM_LSO_Handle_t *pHandle)
{
    // 1. DEFINICIÓN DE PUNTEROS LOCALES (Atajos para legibilidad)
    ACIM_MotorParams_t *pParams = pHandle->pACIM_MotorParams;
    FOCVars_t *pFOC = pHandle->pFOCVars;

    //2. VARIABLES FÍSICAS (Nomenclatura Académica) ---
    float sigma = pParams->fsigma; // Coeficiente de dispersión total
    float tau_r = pParams->ftaur; // Constante de tiempo del rotor (Lr/Rr)
    float tau_s = pParams->ftaus; // Constante de tiempo del estator (Ls/Rs)
    float L_s = pParams->fLs; // Inductancia del estator [Henry]
    float PolePairs = (float)pParams->bPP; // Pares de polos

    float Ts = pHandle->fCalcAngleExecTime_s; // Periodo de muestreo [s]
    float Gain_K = pHandle->k; // Ganancia de diseño del observador
    //3. VARIABLES DE ESTADO ---
    float omega_r_elec = pHandle->fRotorElSpeed_obs_rads; // Velocidad eléctrica estimada
    float omega_e = pHandle->fRotorFluxFreq_rads; // Frecuencia del flujo (Síncrona)
    // Extracción de estados previos (k-1)
    float i_qs_hat = pHandle->fIqds_obs_A.fS_Component1;
    float i_ds_hat = pHandle->fIqds_obs_A.fS_Component2;
    float psi_dr_hat = pHandle->fFlux_qdr_obs.fS_Component2; // Flujo rotor eje d

    // 4. ADQUISICIÓN DE SEÑALES ---
    Signal_Components v_qds_meas = Convert_s16_to_V(pHandle->pBVS, pFOC->Vqd);
    Signal_Components i_qds_meas = Convert_s16_to_A(pFOC[M1].Iqd);

    // 5. PRE-CÁLCULO DE TÉRMINOS INVERSOS (Optimización CPU)
    float inv_sigma = 1.0f / sigma;
    float inv_tau_r = 1.0f / tau_r;
    float inv_tau_s = 1.0f / tau_s;
    float co_sigma = 1.0f - sigma; // Término de acople (1 - sigma)

    //6. CÁLCULO DE LA MATRIZ DE GANANCIA 'K' (Variable con velocidad)
```

```

// Ec. 3.2 (Matriz de ganancias adaptativa)
float k_common = Gain_K - 1.0f;

float g1 = -(k_common * inv_sigma) * (inv_tau_r + inv_tau_s);
float g2 = k_common * omega_r_elec;
float g3 = (k_common / co_sigma) * (inv_tau_r - (Gain_K * inv_tau_s));
float g4 = (k_common * omega_r_elec * sigma) / co_sigma;

// 7. CÁLCULO DEL ERROR DE ESTIMACIÓN (Innovación)
// e = y - y_hat
float error_iq = i_qds_meas.fS_Component1 - i_qs_hat;
float error_id = i_qds_meas.fS_Component2 - i_ds_hat;

// Guardar errores para monitoreo
pHandle->flqds_error_A.fS_Component1 = error_iq;
pHandle->flqds_error_A.fS_Component2 = error_id;

// 8. MECANISMO DE ADAPTACIÓN DE VELOCIDAD (Lyapunov)
// Ajuste de velocidad basado en la ortogonalidad del error
float speed_error_term = -(error_iq * psi_dr_hat);
// Salida del controlador PI -> Velocidad estimada
pHandle->fRotorElSpeed_obs_rads = PI_Float_Calc(pHandle->fPI, speed_error_term);
// Actualización local para cálculos subsiguientes
omega_r_elec = pHandle->fRotorElSpeed_obs_rads;

// 9. CÁLCULO DE FRECUENCIA DE DESLIZAMIENTO Y SÍNCRONA
// Evitar división por cero en arranque (Flujo inicial nominal)
if(psi_dr_hat == 0.0f) psi_dr_hat = pParams->flmagn_A;
// omega_slip = i_qs / (tau_r * psi_dr)
float omega_slip = i_qs_hat / (psi_dr_hat * tau_r);
// omega_e = omega_r + omega_slip
omega_e = omega_r_elec + omega_slip;
pHandle->fRotorFluxFreq_rads = omega_e;

// 10. INTEGRACIÓN DEL ÁNGULO (Theta)
pHandle->fRotorFlux_Angle_rad += omega_e * Ts;
// Normalización [0, 2PI]
pHandle->fRotorFlux_Angle_rad = (float)fmod((double)pHandle->fRotorFlux_Angle_rad,
(double)(2.0f * PI));

// 11. ACTUALIZACIÓN DE ESTADOS (Discretización Euler)
// dx/dt = Ax + Bu + K(y - y_hat)

// A. Derivada de Corriente Eje Q (di_qs/dt)
float d_iq = (-omega_e * i_ds_hat) // Acople rotacional
+ (-inv_sigma * (inv_tau_s + (co_sigma * inv_tau_r)) * i_qs_hat) // Disipación
resistiva
- (co_sigma * inv_sigma * omega_r_elec * psi_dr_hat) // Back-EMF
+ (v_qds_meas.fS_Component1 / (sigma * L_s)) // Tensión de entrada
+ (g2 * error_id) + (g1 * error_iq); // Corrección del Observador

```

```

// B. Derivada de Corriente Eje D (di_ds/dt)
float d_id = (omega_e * i_qs_hat) // Acople rotacional
+ (-inv_sigma * (inv_tau_s + (co_sigma * inv_tau_r)) * i_ds_hat) // Disipación
resistiva
+ ((co_sigma / (sigma * tau_r)) * psi_dr_hat) // Aporte del flujo rotor
+ (v_qds_meas.fS_Component2 / (sigma * L_s)) // Tensión de entrada
+ (g1 * error_id) - (g2 * error_iq); // Corrección del Observado
// C. Derivada de Flujo Eje D (dpsi_dr/dt)
//Se asume alineación de campo (FOC), por lo que dpsi_qr/dt ~ 0 y no se calcula.
float d_psi = (i_ds_hat * inv_tau_r) // Generación por corriente
- (psi_dr_hat * inv_tau_r) // Decaimiento
+ (g3 * error_id) - (g4 * error_iq); // Corrección del Observador

// Integración numérica: x(k+1) = x(k) + dx * Ts
pHandle->flqds_obs_A.fS_Component1 += d_iq * Ts;
pHandle->flqds_obs_A.fS_Component2 += d_id * Ts;
pHandle->fFlux_qdr_obs.fS_Component2 += d_psi * Ts;

// 12. SALIDA FINAL
pHandle->hElAngle = Convert_Rad_to16bit(pHandle->fRotorFlux_Angle_rad);

// Cálculo auxiliar (RPM mecánicas)
pHandle->_SpeedEstimator.hAvrMecSpeedUnit = (int16_t)((omega_r_elec * 10.0f) / (2.0f
* PI * PolePairs));
pHandle->fRotorSpeed_RPM = (float)pHandle->_SpeedEstimator.hAvrMecSpeedUnit *
6.0f;
}

Control de velocidad
/* LICENCIA
* @attention
* <h2><center>&copy; Copyright (c) 2023 STMicroelectronics.</center></h2>
* This software component is licensed by ST under Ultimate Liberty license SLA0044.
*/
/*
* @note
* Implementación discreta de regulador PI con aritmética de punto fijo.
*/
/**
* @brief Ejecuta el lazo de control de velocidad (Regulador PI).
* Calcula la referencia de torque (Iq*) basándose en el error de velocidad.
* @param pHandle: Puntero a la estructura con los estados y ganancias del PI.
* @param error_in: El error de velocidad actual (w_ref - w_meas) en formato Q30.
* @retval out_iq_ref: La corriente de referencia (Iq*) en formato Q30.
*/
fixp30_t PIDREG_SPEED_run(PIDREG_SPEED_s* pHandle, const fixp30_t error_in)
{
// 1. CONFIGURACIÓN INICIAL
// Carga de límites para la saturación (Anti-windup)

```

```

fixp24_t limit_max = pHandle->Max;
fixp24_t limit_min = pHandle->Min;

// Carga del estado previo del integrador: ui(k-1)
fixp24_t term_integral_prev = pHandle->Ui;

// 2. ACONDICIONAMIENTO DE SEÑAL (CAMBIO DE FORMATO Q)
// La entrada viene en 30 bits (mayor precisión), pero el núcleo del PI
// opera en 24 bits para optimizar la multiplicación en el MCU.
// Operación: Shift bit a bit (>>)
fixp24_t error_scaled = (error_in >> (PU_FMT - SUM_FTM));

// 3. CÁLCULO DEL TÉRMINO PROPORCIONAL (P)
// Ecuación: up(k) = Kp * e(k)
fixp24_t term_proportional = FIXP_mpyFIXPscaled(error_scaled, &pHandle->Kp_fps);

// 4. CÁLCULO DEL TÉRMINO INTEGRAL (I) CON ANTI-WINDUP
// Ecuación: ui(k) = ui(k-1) + (Ki * Ts * e(k))
// Nota: Ki_fps ya incluye el periodo de muestreo (Ki * Ts) pre-calculado.

// A. Acumulación (Integración numérica rectangular)
fixp24_t integral_increment = FIXP_mpyFIXPscaled(error_scaled, &pHandle->Ki_fps);
fixp24_t term_integral_new = term_integral_prev + integral_increment;

// B. Anti-windup (Método Clamping Dinámico)
// Si la integral crece más allá de los límites físicos del motor, se recorta.
term_integral_new = FIXP_sat(term_integral_new, limit_max, limit_min);

// Guardar estado para el siguiente ciclo de reloj (k+1)
pHandle->Ui = term_integral_new;

// 5. CÁLCULO DE LA SALIDA TOTAL (u)
// Ecuación: u(k) = up(k) + ui(k)
fixp24_t output_sum = term_proportional + term_integral_new;

// Saturación final de la salida de control
// Protección contra sobre-corriente en la referencia Iq
output_sum = FIXP_sat(output_sum, limit_max, limit_min);

// --- 6. SALIDA Y MONITOREO ---
// Conversión de vuelta a formato de 30 bits para el resto del sistema FOC
fixp30_t out_iq_ref = (output_sum << (PU_FMT - SUM_FTM));

// Actualización de variables de debug (para ver en STM32CubeMonitor)
pHandle->Err = error_scaled; // Error procesado
pHandle->Up = term_proportional; // Esfuerzo proporcional
pHandle->Out = out_iq_ref; // Salida final

return (out_iq_ref);
}

```

Control de corrientes I_d , I_q

```
/*
 * LICENCIA
 *
 * @attention
 * <h2><center>&copy; Copyright (c) 2023 STMicroelectronics.</center></h2>
 * This software component is licensed by ST under Ultimate Liberty license SLA0044.
 */

/**
 * @brief Lazo de Control de Corriente Vectorial (Ejes D y Q).
 * @note Implementa dos reguladores PI en paralelo con Desacoplamiento de Ejes
 * y Limitación Circular de Voltaje (Prioridad Eje D)
 */
void PIDREGDQX_CURRENT_run(PIDREGDQX_CURRENT_s* pHandle,
    const fixp30_t error_d_in,
    const fixp30_t error_q_in,
    const fixp30_t omega_elec_pu)
{
    // 1. CONFIGURACIÓN INICIAL
    // Carga de límites estáticos de saturación (Voltaje Bus DC)
    fixp24_t limit_max_D = pHandle->MaxD;
    fixp24_t limit_min_D = pHandle->MinD;

    // El límite Q será dinámico (calculado más abajo)
    fixp24_t limit_max_Q = pHandle->MaxQ;
    fixp24_t limit_min_Q = pHandle->MinQ;

    // Carga de integradores previos (Memoria)
    fixp24_t integral_D_prev = pHandle->UiD;
    fixp24_t integral_Q_prev = pHandle->UiQ;

    // Factor de compensación por variación de Vbus (Feed-forward voltaje)
    fixp24_t vbus_comp = pHandle->compensation;

    //2. CÁLCULO DE FACTOR DE DESACOPLAMIENTO (CROSS-COUPLING)
    // Factor  $K_{decoupling} \sim \omega_e * L_s / K_p$ 
    // Esto compensa la fuerza contraelectromotriz inducida entre ejes.
    fixp24_t factor_decoupling = FIXP(0.0f);

    if(pHandle->crosscompON)
    {
        factor_decoupling = FIXP30_mpy(omega_elec_pu, pHandle->wfsT);
    }

    //3. ACONDICIONAMIENTO DE ERROR (Q30 -> Q24)
    fixp24_t error_d = (error_d_in >> (PU_FMT - SUM_FTM));
    fixp24_t error_q = (error_q_in >> (PU_FMT - SUM_FTM));
}
```

```

// Guardar errores para debug
pHandle->ErrD = error_d;
pHandle->ErrQ = error_q;

// 4. TÉRMINO PROPORCIONAL (P)
//  $U_p = K_p * \text{error} * \text{Compensacion\_Vbus}$ 

// Eje D
fixp24_t prop_D = FIXP_mpyFIXPscaled(error_d, &pHandle->Kp_fps);
prop_D = FIXP_rsmpry(prop_D, vbus_comp);

// Eje Q
fixp24_t prop_Q = FIXP_mpyFIXPscaled(error_q, &pHandle->Kp_fps);
prop_Q = FIXP_rsmpry(prop_Q, vbus_comp);

// 5. TÉRMINO INTEGRAL EJE D ( $I_d$ )
//  $U_{i_d}(k) = U_{i_d}(k-1) + (K_i * U_p_d) - (\text{Factor\_Desacoplo} * U_{p_q})$ 
// Nota: El término cruzado se RESTA en el eje D.

fixp24_t cross_term_Q_to_D = FIXP_mpy(prop_Q, factor_decoupling);

integral_D_prev += FIXP_mpyFIXPscaled(prop_D, &pHandle->Wi_fps) -
cross_term_Q_to_D;

// Anti-windup Eje D
pHandle->clippedD = ((integral_D_prev <= limit_min_D) || (integral_D_prev >=
limit_max_D));
integral_D_prev = FIXP_sat(integral_D_prev, limit_max_D, limit_min_D);
pHandle->UiD = integral_D_prev; // Guardar estado

// Salida Eje D ( $V_d$ _ref provisional)
fixp24_t out_sum_D = FIXP_sat(prop_D + integral_D_prev, limit_max_D, limit_min_D);

//6. LIMITACIÓN CIRCULAR (PRIORIDAD DE FLUJO)
// Como el voltaje del inversor es finito ( $V_{\max}$ ), debemos asegurar que  $V_d$  tenga
prioridad
// para mantener el flujo magnético, sacrificando  $V_q$  (Torque) si es necesario.
// Ecuación:  $V_{q\_max} = \sqrt{V_{\max}^2 - V_{d\_usado}^2}$ 

fixp24_t Vd_squared = FIXP_mpy(out_sum_D, out_sum_D);

// Presupuesto restante para Q
fixp24_t Vq_budget_squared = pHandle->maxModulation_squared - Vd_squared;

// Calcular nuevos límites dinámicos para Q
limit_max_Q = FIXP24_sqrt(Vq_budget_squared);
limit_min_Q = -limit_max_Q;

// Actualizar estructura (para que el PI sepa sus nuevos límites)
pHandle->MaxQ = limit_max_Q;

```

```

pHandle->MinQ = limit_min_Q;

//7. TÉRMINO INTEGRAL EJE Q (I_q)
//  $U_{i\_q}(k) = U_{i\_q}(k-1) + (K_i * U_{p\_q}) + (Factor\_Desacoplo * U_{p\_d})$ 
// El término cruzado se SUMA en el eje Q.

fixp24_t cross_term_D_to_Q = FIXP_mpy(prop_D, factor_decoupling);

integral_Q_prev += FIXP_mpyFIXPscaled(prop_Q, &pHandle->Wi_fps) +
cross_term_D_to_Q;
// Anti-windup Eje Q (Usando límites dinámicos calculados en Paso 6)
pHandle->clippedQ = ((integral_Q_prev <= limit_min_Q) || (integral_Q_prev >=
limit_max_Q));
integral_Q_prev = FIXP_sat(integral_Q_prev, limit_max_Q, limit_min_Q);
pHandle->UiQ = integral_Q_prev; // Guardar estado
// Salida Eje Q ( $V_{q\_ref}$ )
fixp24_t out_sum_Q = FIXP_sat(prop_Q + integral_Q_prev, limit_max_Q, limit_min_Q);

//8. SALIDA FINAL Y MONITOREO
// Reconvertir a formato Q30
fixp30_t out_Vd = (out_sum_D << (PU_FMT - SUM_FTM));
fixp30_t out_Vq = (out_sum_Q << (PU_FMT - SUM_FTM));

// Guardar variables de debug
pHandle->UpD = prop_D;
pHandle->OutD = out_Vd;
pHandle->UpQ = prop_Q;
pHandle->OutQ = out_Vq;
pHandle->clipped = pHandle->clippedD || pHandle->clippedQ;
}

```