



BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA ELECTRÓNICA

MAESTRÍA EN INGENIERÍA ELECTRÓNICA, OPCIÓN
INSTRUMENTACIÓN ELECTRÓNICA

Tesis presentada para obtener el grado de
MAESTRO EN INGENIERÍA ELECTRÓNICA, OPCIÓN
INSTRUMENTACIÓN ELECTRÓNICA

“CONTROL VECTORIAL DIRECTO DE UN MOTOR DE
INDUCCIÓN TIPO JAULA DE ARDILLA (MITJA) SIN SENSOR
DE VELOCIDAD Y OPTIMIZACIÓN DE EFICIENCIA”

Presenta:

Donato Buenabad Acosta*

Asesores:

Dr. Gerardo Mino Aguilar

Dr. Edgar Peralta Sánchez

Dr. Germán Ardul Muños Hernández

* Becario CONACYT

PUEBLA, PUE. ENERO DE 2020

Resumen

El desarrollo del presente trabajo está dirigido hacia el diseño e implementación de un esquema de control de velocidad, empleando una técnica de control vectorial asistida por redes neuronales para un Motor de Inducción Tipo Jaula de Ardilla. A lo largo del trabajo se incluyen detalles sobre el desarrollo teórico, la realización del modelado, la simulación e implementación del accionamiento y controlador de velocidad para el motor.

Primeramente, se obtiene el modelo del motor de inducción en el marco de referencia ortogonal $dq0$ (se transforman los valores trifásicos abc , variables sinusoidales en el tiempo, a valores constantes $dq0$ en régimen permanente). Posteriormente se diseña el accionamiento y el controlador de velocidad, el cual se basa en la técnica de control vectorial o control por campo orientado (FOC).

Adicionalmente, se diseña un control de velocidad adaptativo basado en redes neuronales B-spline; su tarea principal será ajustar las ganancias proporcionales e integrales del control vectorial, que tradicionalmente son fijas, a un esquema de aprendizaje en línea. Finalmente, se analizan los resultados obtenidos de simulación y se validan en un sistema de Hardware in the Loop (HIL) y un DSP de la familia C2000 de Texas Instruments.

Agradecimientos

Agradezco a CONACYT por el apoyo de manutención brindado para la realización de la maestría. A la Benemérita Universidad Autónoma de Puebla y la Facultad de Ciencias de la Electrónica por la formación académica y humana que me fue brindada a lo largo de mi estancia. Gracias a mis asesores de tesis, Dr. Gerardo Mino Aguilar, Dr. Edgar Peralta Sánchez y Dr. Germán Ardul Muños Hernández por su apoyo y confianza brindada durante el desarrollo del presente trabajo. Este trabajo de tesis está especialmente dedicado a mis padres, hermanas y amigos que son y han sido un gran pilar en mi desarrollo. En general agradezco a cada persona que de alguna manera contribuyo para la realización de este trabajo.

Contenido

Resumen

Agradecimientos

Contenido

Capítulo 1: Introducción

1.1 Objetivos.....	6
1.2 Justificación de la investigación	6
1.3 Estado del arte	7

Capítulo 2: Fundamentos para el control del Motor de Inducción

2.1 Introducción.....	13
2.2 Clasificación de las maquinas eléctricas	13
2.3 Máquinas de inducción.....	14
2.3.1 Componentes principales de un motor de inducción.....	15
2.3.2 Principio de operación de un motor de inducción.....	16
2.4 Técnicas de control para el motor de inducción	18
2.4.1 Control escalar.....	18
2.4.2 Control vectorial.....	19
2.4.3 Control directo de torque.....	21
2.5 Selección de la técnica de control a utilizar	22
2.6 La transformación de Clark.....	23
2.7 La transformación de Park.....	24
2.8 Redes neuronales artificiales.....	25
2.8.1 Aplicación de las redes neuronales.....	25
2.8.2 Elementos de una red neuronal artificial	25
2.8.3 Arquitectura de una red neuronal.....	26
2.8.4 Mecanismos de aprendizaje para una red neuronal.....	27
2.9 Redes neuronales B-spline.....	29
2.9.1 Entrada de la red B-spline.....	29
2.9.2 Salida de la red B-spline	30
2.9.3 Aprendizaje de la red B-spline.....	31
2.9.4 Funciones base.....	32

Capítulo 3: Modelo matemático y simulación del Motor de Inducción

3.1 Circuito equivalente del motor de inducción.....	35
3.2 Modelo dinámico del motor de inducción.....	36
3.2.1 Representación en vectores espaciales.....	36
3.2.2 Representación en marco de referencia $\alpha - \beta$	39
3.2.3 Representación en marco de referencia $d - q$	39
3.3 Resumen de parámetros del motor de inducción	41
3.4 Validación del modelo del motor de inducción.....	42

3.5 Simulación del motor en vacío en PSIM.....	44
3.6 Convertidor CD-CA.....	45
3.7 Control V/F para el motor de inducción.....	47
3.7.1 Diseño y simulación del control escalar V/f.....	47
3.7.2 Implementación del control escalar V/f.....	50

Capítulo 4: Control vectorial del motor de inducción

4.1 Esquema general del control vectorial.....	52
4.1.1 Transformación de marcos de referencia.....	52
4.1.2 Diseño del controlador de torque electromagnético.....	53
4.1.3 Diseño del controlador PI de corriente.....	54
4.1.4 Diseño del controlador de velocidad.....	55
4.1.5 Diseño del estimador de deslizamiento.....	55
4.2 Simulación del control vectorial.....	56
4.3 Emulación del control vectorial en Typhoon HIL.....	58
4.3.1 Typhoon HIL y PSIM.....	60
4.3.2 Esquema de control en PSIM.....	61
4.3.3 Esquema físico en Typhoon HIL.....	67
4.3.4 Resultados experimentales.....	68

Capítulo 5: Red neuronal B-spline para el control de un motor de inducción

5.1 Esquema general del control neuronal B-spline.....	71
5.2 Control B-spline.....	72
5.3 Control de velocidad basado en red B-spline.....	73
5.4 Simulación del control de velocidad B-spline.....	74
5.5 Esquema de control neuronal B-spline en PSIM.....	79
5.5.1 Rutina para la red neuronal.....	80
5.5.2 Rutina para el cálculo de la función base.....	80
5.5.3 Resultados experimentales.....	80

Conclusiones

Trabajos futuros

Bibliografía

Apéndice A

Apéndice B

Capítulo 1

1. INTRODUCCIÓN

Desde la década de los 70's hasta la actualidad, existe un esfuerzo muy importante por conocer el comportamiento dinámico de la máquina de inducción, a fin de desarrollar novedosas técnicas de control que mejoren sus características y permitan incorporarlas rápidamente en equipos comerciales [1].

Las máquinas de inducción son ampliamente utilizadas en la industria y en diversas aplicaciones de la vida moderna (desde los grandes generadores síncronos que convierten la energía térmica o hidráulica en enormes volúmenes de potencia y energía eléctrica, hasta pequeños ventiladores que refrigeran los circuitos del computador, ambas satisfacen las mismas leyes físicas y principios básicos) gracias a su simplicidad, confiabilidad, bajo costo de mantenimiento y robustez [2].

Al día de hoy, las máquinas de inducción se han convertido en el elemento básico para la conversión de energía electromecánica en la industria, se utilizan con más frecuencia que todos los demás tipos de motores combinados. Varios factores las han convertido en la máquina elegida para aplicaciones industriales frente a las máquinas de DC; la máquina de inducción es simple de fabricar, sin bobinados de rotor, no requiere reemplazo de escobillas debido al desgaste, tiene mayor densidad de potencia, mayor velocidad y menor inercia del rotor que la máquina de DC [3].

Hasta hace no mucho tiempo el motor de inducción (MI) era considerado como una máquina de velocidad casi constante y no era posible conseguir variaciones de velocidad en forma continua, de manera simple y barata. Con el avance tecnológico en el área de procesamiento de dispositivos electrónicos y en electrónica de potencia, hoy por hoy los accionamientos controlados para los motores de inducción resultan mucho más ventajosos que los utilizados en los motores de corriente continua, y esto es reflejo del empleo del principio de control vectorial, permitiendo que la máquina tenga las prestaciones del motor de corriente continua, con las muchas ventajas de la máquina de inducción [3].

Con la rápida evolución de las técnicas de control de velocidad por campo orientado, se han buscado mejores opciones de control que permitan estimar la velocidad sin necesidad de sensores, esto con el fin de reforzar el concepto de robustez en el control de la máquina de inducción [5]. Cabe destacar que el procesamiento que implica el control vectorial es muy complejo y requiere el uso de Microcontroladores de alta velocidad o procesadores digitales de señales (DSP).

1.1 OBJETIVOS

1.1.1 Objetivo General

Diseñar un variador de velocidad para un motor de inducción tipo jaula de ardilla (MI), empleando un sistema de control vectorial o control por campo orientado (FOC).

1.1.2 Objetivos Específicos

1. Modelar y caracterizar un motor de Inducción tipo jaula de ardilla (MITJA).
2. Implementar un sistema de control vectorial para el MITJA por el método directo, usando redes neuronales.
3. Diseñar un controlador de velocidad basado en redes neuronales.

1.2. JUSTIFICACIÓN

Aproximadamente un 90% de los procesos automatizados en la industria, poseen un control de motores y muchos de ellos están orientados al control de un motor de inducción (MI), este tipo de máquina eléctrica tiene muchos años de ser conocida, y sigue siendo ampliamente utilizada, su menor costo, su mayor robustez, alta eficiencia, nulo mantenimiento y su tamaño compacto con respecto a otro tipo de máquina de la misma potencia la hacen ideal para toda aplicación industrial [5]. A pesar de las múltiples ventajas que tiene el motor de inducción frente al motor de corriente continua, cuando se desea controlar su velocidad de forma precisa, se presentan múltiples dificultades puesto que el motor de inducción es un sistema no lineal, multivariable y altamente acoplado [7].

Al no ser tan fáciles de controlar, las técnicas se vuelven más complejas. Los mecanismos de control actuales se basan en la necesidad de montaje de un tacómetro en la flecha del rotor o en su defecto la medición directa del estator mediante sensores, para mejorar sus prestaciones y su respuesta se debe considerar de manera desacoplada las variables de flujo y par. Es por este motivo que se recurre al control vectorial, también llamado control por campo orientado (*FOC*), a fin de lograr un sistema más robusto que permita simplificar y tratar al motor de inducción de la misma manera que el motor de corriente continua.

Algunos aspectos puntuales que justifican este tema de tesis son:

- Controlar un motor de inducción (MI) a velocidades bajas y precisas resulta difícil: a fin de aprovechar sus características, optimizar su eficiencia, ampliar su rango de operación y por ende su campo de aplicación, es necesario desarrollar técnicas de control para este tipo de motores.

- Aunque el motor no es nuevo, el diseño del controlador que se obtendrá proporcionara mayor robustez y eficiencia al sistema de control, al asistirse de una técnica de control basada en redes neuronales.
- Los sistemas basados en redes neuronales ofrecen una alternativa interesante por su desempeño; proporcionando un considerable valor a los sistemas de potencia al eliminar ciertas limitantes de su propio diseño, pero sobre todo a su característica de manejar gran cantidad de información y aprender de ella.
- Adicionalmente a los resultados a nivel simulación, se muestran resultados experimentales, lo cual permite hacer una comparativa más aproximada de lo simulado a lo real.

1.3 ESTADO DEL ARTE

Desde principios de los 70's hasta la actualidad la mayoría de controladores de velocidad para un motor de inducción (MI) están formados por controles vectoriales, que permiten lograr una respuesta dinámica superior a la respuesta alcanzada con cualquier otra estrategia de control. Aunque las bases sobre el control de este tipo de motores se encuentran bien cimentadas, diferentes autores siguen en búsqueda de nuevas técnicas de control que permitan simplificar el modelo dinámico y aumentar la eficiencia de estos motores.

El conjunto de referencias consultadas y citadas de la literatura se puede agrupar de la siguiente manera:

- Técnicas de control vectorial
- Técnicas de control vectorial sin sensores

1.3.1 Técnica de control vectorial

El control de campo orientado (FOC), también llamado vectorial, se usa en accionamientos de motor de inducción alimentados por un inversor para obtener una respuesta de velocidad de alto rendimiento. Para el control de campo orientado, es necesario conocer la magnitud instantánea y la posición del flujo del rotor. La magnitud y la posición del flujo del rotor se aproximan en función de las mediciones de flujo del entrehierro en el esquema del control vectorial directo (se requiere de un motor equipado con sensores de efecto Hall o bien se estima a partir del modelo del motor y la medida de la tensión y la corriente en el estator) y se estiman en el esquema de control vectorial indirecto (depende de los parámetros de la máquina y es necesaria una señal de posición del rotor) [5].

En esencia, el control vectorial busca mantener el flujo del estator y par electromagnético de la máquina a un ángulo de 90° entre ellos (desacoplamiento), de tal manera que el control se lleve a cabo como en una máquina de corriente directa. Es decir, que admita un control independiente de flujo y de par, pues se reduce el modelo no lineal del motor de inducción a un modelo referido a dos ejes ortogonales d y q ; de este modo, el control

de flujo se logra regulando la componente de la corriente del eje directo d y el control del par se logra regulando la corriente asociada al eje en cuadratura q [8].

A continuación se describen de forma breve los diferentes trabajos relacionados a este tipo de controladores:

- En 1969 [5] K. Hasse y F. Blaschke, presenta los principios de una nueva técnica de control para máquinas de campo rotatorio denominada control vectorial o por campo orientado (FOC). Esta técnica consiste en transformar la estructura dinámica del motor de inducción para hacerla similar a la de una máquina de CD con excitación separada.

Con la aparición de la técnica de campo orientado fue posible controlar la velocidad de la máquina de inducción y síncrona, pero no fue posible todavía eliminar los transductores electromecánicos que sirven para censar la velocidad del motor, representando una desventaja para la simplicidad del motor e imponen limitaciones en el rango de operación de la maquina (en el extremo de baja velocidad).

1.3.2 Técnica de control vectorial sin sensores

En los últimos años, los esquemas sin sensores para máquinas de inducción han recibido una gran atención. Porque reduce la complejidad del hardware y reduce el costo, reduce el tamaño del controlador de la máquina, elimina el cable del sensor, tiene una mejor inmunidad al ruido, aumenta la confiabilidad y requiere menos mantenimiento [5].

En seguida se describen de forma breve los diferentes trabajos relacionados a este tipo de controladores:

- En [6] surge el primer intento por operar un motor de inducción en lazo cerrado sin utilizar un sensor de velocidad. Este trabajo marco el inicio del método de control de velocidad sin sensor mecánico conectado en la flecha.
- En [7] Chang y Yeh presentan una nueva estrategia para controlar un motor de inducción sin sensor de corriente. Utiliza un estimador de estado parcial para estimar la corriente de torque y un predictor de voltaje para predecir el vector de voltaje del inversor.
- A lo largo de los años se han desarrollado diversas metodologías, como los observadores de MRAS, los observadores adaptativos de filtro de Kalman ampliados y la técnica de modo deslizante. Pero nadie ha sido capaz de resolver todos los problemas del control sin sensores [8].

1.3.2.1 Técnicas de control vectorial basadas en observadores

El siguiente paso en los métodos de control de velocidad sin sensor se basan en observadores y sistemas adaptables a un modelo de referencia [5], se han desarrollado diversos tipos de observadores, la mayoría de ellos tienden a utilizar los valores de voltaje y corriente del motor y a partir de ellos estimar otros parámetros de la máquina:

- El trabajo de Schauder [9], explora las limitaciones de velocidad y estimación de flujo en un control por campo orientado directo para un motor de inducción. Se propone una topología mejorada la cual utiliza un observador de flujo integrado en lazo cerrado y un modelo mecánico del sistema, sin embargo, no logra superar la deficiencia fundamental de robustez a velocidad cero.
- En trabajos posteriores [7] [8], se nota una clara tendencia a reemplazar el control voltaje-frecuencia (v/f) por el control vectorial. En ellos se propone un observador de deslizamiento, el cual tiene como ventaja con respecto a los estimadores de orden reducido la estimación robusta del flujo del rotor mediante la cual se realiza una identificación adaptable de los parámetros del motor de inducción.
- El siguiente paso en la evolución de estimadores de velocidad de motores de inducción sin sensor de velocidad, surge con los trabajos de Ben-Brahim y Kubota [10] [11], en donde se realiza la implementación de un control por campo orientado basado en observadores predictivos con regulación digital de corriente; se miden únicamente las corrientes del estator y a partir de ella se estiman los valores de velocidad y flujos del rotor. Esto se realiza mediante un observador de estado predictivo.
- En la literatura aparecen trabajos que describen un sistema del tipo MRAS para la estimación de velocidad de un motor de inducción [13] [14], al igual que la mayoría de los trabajos mencionados anteriormente, se utilizan los valores de voltaje y corriente. La velocidad estimada se usa como retroalimentación en un sistema de control vectorial.
- En el caso presentado por Matsuse y Tadokoro [14], se utiliza un observador de orden completo para la estimación de la velocidad del rotor, esto representa un beneficio en la estimación a bajas velocidades, incluso puede funcionar correctamente en la velocidad cero.
- En investigaciones más recientes, se reportan trabajos que utilizan el método de control de flujo de banda muerta para orientación de campo directa basados en el uso de un observador adaptable de flujo del rotor de tipo orden completo [15], dicho observador tiene la ventaja de estimar la resistencia del estator y del rotor, a la vez que sirve para la implementación del controlador de campo orientado directo.
- En la actualidad, el filtro extendido de Kalman [5] es uno de los observadores más utilizados en la estimación de variables de estado o como identificador de parámetros. Entre estas variables se encuentran la velocidad y los flujos del motor. En [16] se propone el uso de filtro de Kalman en aquellos sistemas que pueden ser influenciados por el ruido y perturbaciones durante su funcionamiento, como ocurre en el control vectorial de los motores de inducción.

1.3.2.2 Técnicas de control vectorial basadas en control predictivo

Desde principios de la década de 1960, el concepto de modelo de control predictivo (MCP) ha ganado popularidad por su aplicación en el control de electrónica de potencia. A pesar de que el uso de este tipo de control se ha centrado principalmente en el control

del convertidor, la tendencia de su aplicación para el control de variadores de velocidad va también en aumento [12].

- En [10] Priyanka propone un controlador simple y eficiente con esquema de control de torque directo (DTC) basado en la técnica de modulación por ancho de pulso del vector espacio (SVPWM) para un motor de inducción Jaula de Ardilla, utiliza una modulación vectorial hexagonal y estimadores de flujo y torque para seleccionar los vectores de voltaje de conmutación del inversor que mantienen el flujo y el error de torque dentro de un límite de banda de histéresis.
- El control predictivo de un paso (One-step), es una estrategia de control de alto rendimiento que combina el uso de un controlador proporcional-integral (PI) para obtener un buen comportamiento en estado estacionario y un controlador predictivo (dead - beat) para lograr una dinámica muy alta en el par motor [11].
- Trabajos mar recientes, muestran un modelo de control predictivo [9][10] y observadores de flujo [11], [12], que en conjunto emplean un algoritmo de control vectorial directo de flujo el cual combina las ventajas de una regulación del flujo directo (Control de Par Directo de Frecuencia Constante) con la regulación de corriente (Control Vectorial), permitiendo una mejora en la respuesta del par dinámico y un comportamiento redundante para lograr tolerancia a fallas.

Otras investigaciones proponen diferentes métodos de control, que son variantes de la rama del Control Predictivo (MCP): DTC clásico, DTC predictivo (PTC) basado en la combinación de los esquemas convencionales DTC y FCS-MPTC y el control de torque directo modificado con modulación de vector espacial (DTC-SVM).

- El primero de ellos es el control de torque directo (DTC por sus siglas en ingles Direct Torque Control), es una nueva técnica de control basada en la orientación del flujo del estator mediante una acción directa en los estados de los interruptores del inversor [13]. La elección de la secuencia apropiada generalmente se basa en el uso de controladores de histéresis cuya función es controlar el flujo y el par electromagnético del estator sin tener que medir la velocidad, flujo o torque. El DTC requiere la estimación de flujo y par a partir de la medición de voltajes y corrientes de estator.
- La segunda es el control predictivo de torque directo (PTC por sus siglas en ingles Predictive Torque Control) se basa en la predicción de los valores futuros de flujo el estator y el par electromagnético [14], [15]. El PTC es una buena solución que mejora el rendimiento del DTC convencional, además de su facilidad de implementación, permite mejorar prestaciones en función de costo, par, corriente y oscilaciones de flujo. Sin embargo, tiene la desventaja de tener una frecuencia de conmutación no constante.
- Algunos otros trabajos de igual manera emplean el enfoque de MCP para su implementación, tal es el caso de del control directo del vector de flujo (DFVC por sus siglas en ingles Direct Flux Vector Control) el cual combina las ventajas del control directo de flujo y par y el control vectorial de corriente (CVC por sus siglas en ingles Current Vector Control) [17].

1.3.2.3 Técnicas de control vectorial basadas en redes neuronales

Las técnicas modernas de control sin sensores para motores de inducción, se están enfocando en modelos que tienden a hacer más efectiva la estimación de velocidad, tal es el caso de las técnicas de redes neuronales y de control difuso, las redes neuronales se muestran prometedoras para su aplicación en electrónica de potencia y sistemas de control de movimiento. Hasta ahora, se han aplicado en algunos casos, principalmente en el control de inversores [19] y de motores de DC [20] [21], pero su aplicación en la estimación [22] para unidades sin sensores de velocidad es prácticamente nueva.

- En [18] se propone una nueva alternativa a los ya existentes métodos de estimación de flujo, el estimador de flujo de red neuronal con la técnica de aprendizaje de retropropagación. La red neuronal es capaz de estimar con precisión la magnitud o posición del flujo del rotor para la operación de arranque de un motor de inducción. La red neuronal no solo ve el valor presente, sino que además ve una ventana con valores pasados en ella y por lo tanto es capaz de reconocer mejor que extremos de la curva de flujo es el valor presente y por lo tanto es capaz de obtener una mejor estimación de la magnitud del flujo.
- En [23] se demuestra que dicha tecnología es aplicable para la estimación de señales de retroalimentación constantes (que es similar al diseño del observador) en un motor de inducción con algunas ventajas claras en comparación con la implementación basada en DSP. El estimador de red neuronal tiene la ventaja de cómputo en paralelo extremadamente rápido, inmunidad al rizado armónico y características de tolerancia a fallas en comparación con el estimador basado en DSP.
- Cabrera L. A. et. al. [24] usa la red neuronal para la sintonización de la resistencia del estator del motor de inducción en un sistema de control directo de par.

1.3.2.4 Técnicas de control vectorial basadas en lógica difusa

Los trabajos siguientes muestran la aplicación de la lógica difusa y las redes neuronales a la electrónica de potencia, específicamente a las máquinas de inducción.

- Liaw C. M. y Wang J. B. [25] proponen una técnica de accionamiento para un motor de inducción adaptando lógica difusa a un control vectorial, esto con la finalidad de compensar la baja eficiencia y una respuesta transitoria deficiente de los mecanismos de campo orientado debido a la variación de la resistencia del rotor.
- Gilberto C. D. [10] propone un variador de velocidad para un motor de inducción basado en lógica difusa para la optimización de eficiencia. El controlador difuso reduce de forma adaptativa la corriente de excitación en función de la potencia de entrada medida, de modo que, para un par y una velocidad de carga dada, el variador se estabiliza con una potencia de entrada mínima, es decir trabaja a su máxima eficiencia.

Capítulo 2

FUNDAMENTOS PARA EL CONTROL DE UN MOTOR DE INDUCCIÓN

En este capítulo se expondrá la teoría y los fundamentos matemáticos para el control de un motor de inducción empleado la técnica de control vectorial, específicamente del control por campo orientado directo, adicionalmente se aborda la teoría y los conceptos más relevantes, necesarios para comprender la notación utilizada a lo largo del documento.

2.1 Introducción

La energía eléctrica representa más del 30 % del uso de energía global, y se encuentra en expansión. La mayor parte de la energía eléctrica producida es convertida en energía mecánica a través de motores eléctricos, de estos, los motores de inducción son los más utilizados en la industria y aplicaciones domésticas. Debido a su alto consumo energético surgen nuevos retos en el diseño e implementación de controladores más óptimos, para la obtención de un mejor desempeño. Su enorme potencial de aplicaciones, justifica la presente investigación sobre el motor de inducción.

2.2 Clasificación de las maquinas eléctricas

Los dispositivos rotatorios de conversión de energía electromecánica son conocidos como maquinas rotatorias. Se pueden clasificar de diferentes maneras, por su velocidad de giro (Asíncronos o síncronos), tipo de rotor (de anillos rasantes, con colector, o jaula de ardilla) y por el número de fases de alimentación (monofásicos, bifásicos, trifásicos, con arranque auxiliar bobinado y con arranque auxiliar bobinado y con condensador).

Se clasifican como máquinas de corriente directa si sus salidas son de corriente directa o si la energía de entrada de la maquina proviene de una fuente de corriente directa. Se llaman máquinas de corriente alterna, si sus salidas son periódicas o si la energía primaria de entrada proviene de una fuente de corriente alterna. Una maquina rotatoria se llama generador, si convierte la energía mecánica en energía eléctrica y se llama motor si convierte la energía eléctrica en mecánica.

Las máquinas de corriente alterna son clasificadas de diferente forma: maquinas monofásicas, bifásicas o trifásicas, dependiendo del tipo de fuente utilizada; como síncronas si giran a una velocidad constante llamada velocidad angular síncrona y, asíncronas si su velocidad de rotación es menor que la velocidad angular síncrona. Generalmente las maquinas asíncronas son conocidas como máquinas de inducción [26].

En la Figura 2.1 se muestra la clasificación de las maquinas eléctricas rotativas.

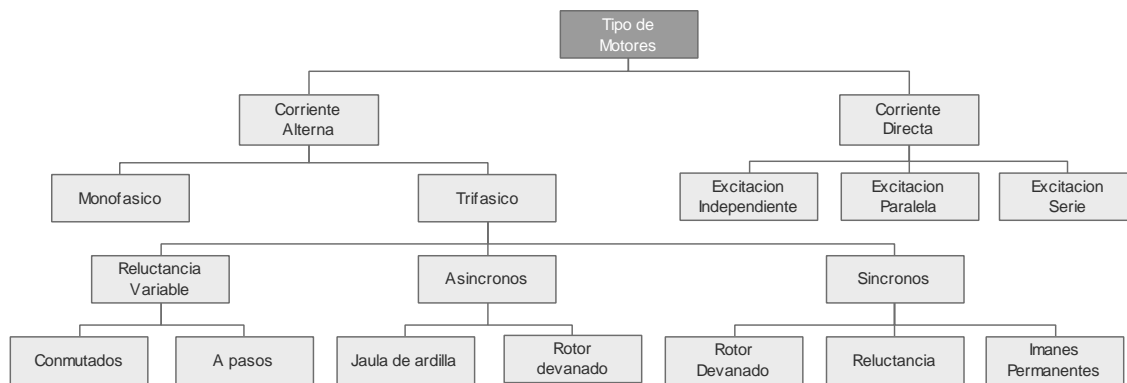


Fig.2.1. Clasificación de las maquinas eléctricas.

2.3 Máquinas de inducción

Las máquinas de inducción se han convertido en el elemento básico para la conversión de energía electromecánica en la industria actual, se utilizan con más frecuencia que todos los otros tipos de motores combinados. Varios factores los han convertido en la máquina ideal para aplicaciones industriales frente a las de CD, incluida su robustez, fiabilidad y bajo mantenimiento. La máquina de inducción de jaula es simple de fabricar, sin devanados de rotor o conmutador para conexión de rotor externo. No hay que reemplazar los cepillos debido al desgaste, ni arcos de los mismos para evitar que la máquina se utilice en entornos volátiles. La máquina de inducción tiene una mayor densidad de potencia, mayor velocidad máxima y menor inercia del rotor que la máquina de CD.

La máquina de inducción tiene una desventaja significativa con respecto al control de par en comparación con la máquina de CD. La producción de torsión de una máquina determinada está relacionada con el producto cruzado de los vectores de enlace de flujo del estator y el rotor. Si los enlaces del flujo del rotor y del estator se mantienen ortogonales entre sí, el par eléctrico de la máquina se puede controlar ajustando el enlace del flujo del rotor o del estator y manteniendo la otra constante. Los bobinados de campo y armadura en una máquina de corriente continua se mantienen ortogonales mediante un conmutador mecánico, lo que hace que el control del par sea relativamente simple. Con una máquina de inducción, los devanados del estator y del rotor no están fijados ortogonalmente entre sí. La máquina de inducción está excitada individualmente, con el campo del rotor inducido por el campo del estator, lo que complica aún más el control del par. Hasta hace unos años, la máquina de inducción se utilizaba principalmente para aplicaciones de velocidad constante. Con las recientes mejoras en la tecnología de semiconductores y la electrónica de potencia, la máquina de inducción está viendo un uso más amplio en aplicaciones de velocidad variable [27].

Por su robustez, por la simplicidad de su estructura, por su peso y su tamaño, el motor de inducción ofrece nuevas perspectivas tecnológicas en numerosas aplicaciones industriales, donde se le aprecia principalmente por su costo y el escaso mantenimiento que requiere. En efecto, entre los diferentes tipos de los motores de corriente alterna de campos giratorios, el motor de inducción (también llamado asíncrono) de jaula de ardilla es la que exige los sistemas de control más delicados para implementar un control efectivo de su par en régimen variable. El simple hecho de que su flujo sea inducido y deba ser

controlado, introduce una complicación suplementaria y la necesidad de acceder directamente (o indirectamente) a esta variable. Sin embargo, sus cualidades intrínsecas son:

- Su construcción no exige materiales magnéticos especiales.
- Su robustez debida a su rotor de jaula.
- Su mantenimiento reducido.
- Su gran potencia.

Estas características han estimulado las investigaciones para definir estructuras de control que compensen sus débiles disposiciones en el control dinámico del par. Además, de que los procesos tecnológicos logrados en los convertidores estáticos y en la electrónica de control han aportado los medios necesarios para su desarrollo [26].

2.3.1 Componentes principales de un motor de inducción

En toda máquina eléctrica existen partes principales como el estator, el rotor y partes auxiliares como la carcasa, flecha, escobillas, tapas laterales, rodamientos, entre otros; estos elementos se describen a continuación:

El *estator* (Figura 2.2) consta de un armazón de acero que soporta un núcleo cilíndrico hueco compuesto de laminaciones apiladas. Varias ranuras equidistantes entre sí, hechas en la circunferencia interna de las laminaciones, proporcionan el espacio para el devanado del estator.

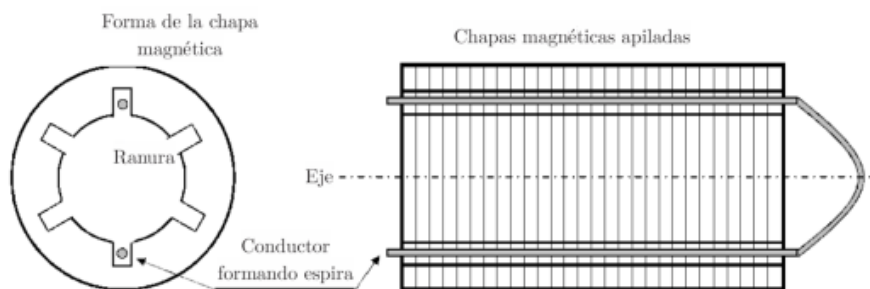


Figura.2.2. Estator de una máquina eléctrica rotatoria [35].

El *rotor* también se compone de laminaciones ranuradas. Éstas están apiladas para crear una serie de ranuras para el devanado del rotor. Se utilizan dos tipos de devanados de rotor: (1) devanados trifásicos convencionales hechos de alambre aislado y (2) devanados de jaula de ardilla. El tipo de devanado da lugar a dos clases principales de motores: *motores de inducción de jaula de ardilla* y *motores de inducción de rotor devanado*.

Un *rotor de jaula de ardilla* (Figura 2.3) se compone de barras de cobre desnudo, un poco más largas que el rotor, las cuales están insertadas en las ranuras por uno de sus extremos. Los extremos opuestos se sueldan a dos anillos de cobre para que todas las barras estén en cortocircuito entre sí. Toda la construcción (barras y anillos extremos) se asemeja a una jaula de ardilla, de donde se deriva el nombre.

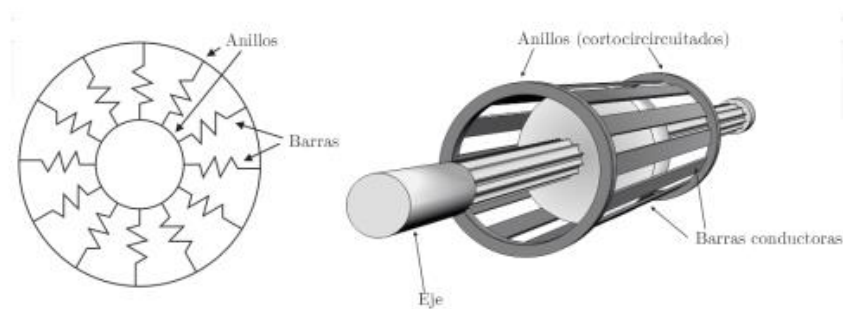


Figura.2.3. Rotor bobinado Jaula de Ardilla Simple [35].

Un *rotor devanado* tiene un devanado trifásico similar al del estator. El devanado está distribuido uniformemente en las ranuras y casi siempre está conectado en Y con 3 conductores. Las terminales están conectadas a tres anillos colectores, los cuales giran junto con el rotor. Los anillos colectores rotatorios y las escobillas estacionarias asociadas permiten conectar resistores externos en serie al devanado del rotor. Los resistores externos se utilizan principalmente durante el periodo de arranque; en condiciones de funcionamiento normal, las tres escobillas están en cortocircuito [35].

La *carcasa* es la cubierta de la máquina y sirve como soporte mecánico de las partes componentes de la misma. Se fabrican con aletas disipadoras de calor de una aleación de aluminio dúctil de alta resistencia mecánica. También existen carcasas sin aletas disipadoras de calor fabricadas con hierro dulce lo que las hace muy pesadas.

La *flecha o eje* es la parte central del rotor, normalmente es de acero y es donde se montan y fijan las laminaciones del rotor. En su extremo se monta un abanico que al giro de la máquina proporciona una corriente de aire que enfría las partes internas debido a que las corrientes que circulan por los devanados tienden a elevar su temperatura y a disminuir su rendimiento.

Las *escobillas* generalmente se fabrican de carbón con un determinado porcentaje de grafito, teniendo en cuenta que a mayor grafito aumenta la dureza y la resistencia eléctrica. En el caso de las máquinas de inducción con rotor de anillos rozantes, las escobillas hacen posible la introducción de resistencias externas por los anillos para limitar las corrientes de arranque, mejorar las características del par y controlar la velocidad [28].

2.3.2 Principio de operación de un motor de inducción

El principio de funcionamiento para máquinas de inducción rotatorias se relaciona básicamente con el par eléctrico. En otras palabras, acerca de las fuerzas que generan movimiento a través del campo electromagnético inducido. O más simple aun, porque el motor de inducción gira. La generación de torque eléctrico en los motores de inducción puede aproximarse a través de:

- Fuerzas sobre los conductores eléctricos en un campo magnético en movimiento.
- La teoría de Maxwell.
- Principios variacionales (Ecuaciones de Lagrange).

Con fundamento en los aspectos de construcción del motor de inducción, el principio de operación de estos es descrito a continuación [26].

Campo magnético rotatorio

El motor asíncrono trifásico está formado por un rotor en el que se encuentran las bobinas inductoras. Estas bobinas son trifásicas y están desfasadas entre sí 120° en el espacio. Según el Teorema de Ferraris, cuando por estas bobinas circula un sistema de corrientes trifásicas equilibradas, cuyo desfase en el tiempo es también de 120° , se induce un campo magnético giratorio que envuelve al rotor. Este campo magnético variable va a inducir una tensión en el rotor según la Ley de inducción de Faraday.

El campo magnético giratorio se obtiene con 3 devanados desfasados 120° (acoplados en estrella o triángulo) y conectados a un sistema trifásico de corriente alterna. Si por el arrollamiento polifásico del estator del motor de una máquina síncrona circula una corriente de pulsación ω y si hay P pares de polos se origina un campo magnético giratorio de P pares de polos y que giran a la velocidad de $\frac{\omega}{P}$ (Teorema de Ferraris).

Velocidad de sincronismo

La máquina asíncrona suele funcionar como motor. El devanado del estator está constituido por tres arrollamientos desfasados 120° en el espacio y de $2p$ polos, al introducir por ellos corriente de una red trifásica de frecuencia f_1 , se produce una fuerza magnetomotriz (*f. m. m*) distribuida por la periferia del entrehierro, que produce un flujo giratorio cuya velocidad viene expresada de acuerdo con (1), que recibe el nombre de **velocidad de sincronismo**:

$$n_1 = \frac{60f_1}{p} \quad (1)$$

Deslizamiento de rotor en el motor de inducción

La velocidad mecánica (n) del rotor se expresa comúnmente como una fracción del parámetro de velocidad síncrona (n_s), mediante el *deslizamiento* (s), expresado como:

$$s = \frac{n_s - n}{n_s} \quad (2)$$

donde n_s , ecuación (2.1), esta expresada en función de la frecuencia de las corrientes (f) de entrada al motor y del número de polos (P) del mismo.

$$n_s = \frac{120f}{P} \quad (3)$$

En un periodo inactivo, el campo magnético rotatorio inducido por el estator tiene la misma velocidad relativa con respecto a los devanados del rotor, y también respecto a los devanados de estator. Por lo tanto la frecuencia de las corrientes del rotor (f_r) es la misma que la frecuencia de las corrientes de estator (f). A velocidad síncrona, no existe un movimiento relativo entre el campo rotatorio de estator y de rotor, y la frecuencia de corrientes de rotor es cero. Esto implica que la frecuencia de rotor es proporcional al deslizamiento con la siguiente relación:

$$f_r = sf \quad (4)$$

donde f_r es la frecuencia de las corrientes de rotor y f es la frecuencia de las corrientes (o voltajes) de estator. En base a lo anterior, se puede concluir que:

- El campo magnético rotatorio de estator gira a una velocidad síncrona n_s .
- La fuerza electromagnética del rotor produce un campo magnético que también gira a la velocidad síncrona y en la misma dirección que el campo producido por el estator. Por lo tanto, los campos electromagnéticos producidos por el rotor y estator son estacionarios uno respecto del otro.
- El campo magnético rotatorio producido por el rotor gira a una velocidad $(\omega_s - \omega_m)$ con respecto al rotor, donde ω_m es la velocidad mecánica real del rotor.
- Las corrientes y voltajes inducidos en el rotor están en función del deslizamiento y por consecuencia, de la frecuencia.

2.4 Técnicas de control para el motor de inducción

Existen diversas maneras de controlar un motor de inducción para obtener las condiciones de alimentación necesarias para su funcionamiento, estas formas de control pueden clasificarse en tres grandes categorías de acuerdo a su desempeño; *el control escalar*, *el control vectorial* y *control directo del par* [1]. En la Figura 2.5 se muestra un diagrama de las técnicas para el accionamiento de un motor de inducción mencionados con anterioridad.

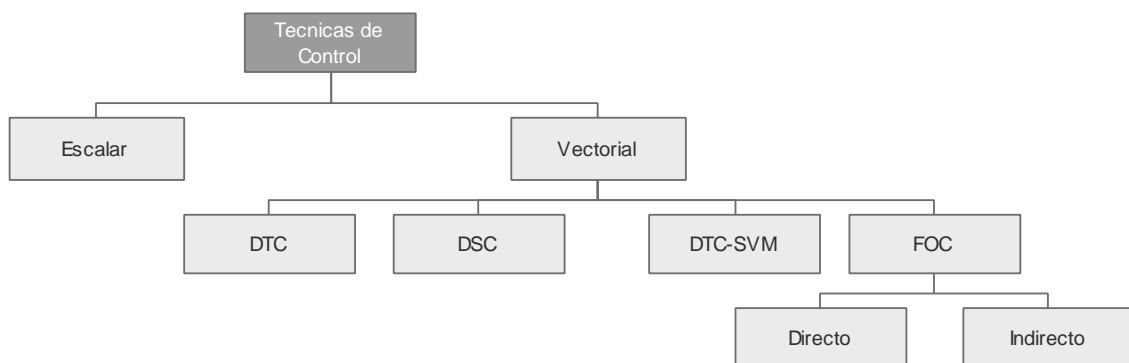


Fig.2.5. Clasificación de las técnicas de control para un motor de inducción.

2.4.1 Control escalar

También llamado control V/F (voltaje/frecuencia) debido a que se realiza un cambio tanto en la frecuencia como en el voltaje aplicado al estator. Está basado en el modelo estático del motor de inducción y su operación típica en lazo abierto [1]. La mayoría de los sistemas de accionamiento de velocidad variable para motores de inducción son variadores en los que las variables ajustadas son la magnitud y la frecuencia del voltaje o la corriente suministrada al estator.

Esto permite el control de la velocidad, mientras que el campo magnético en el motor se mantiene a un nivel constante y deseado. Este tipo de control se conoce como control escalar, ya que se supone que la tensión o corriente del estator controlado es sinusoidal, con solo la magnitud y frecuencia ajustada, y sin preocuparse por la posición espacial (fase) de las cantidades de vectores correspondientes.

En contraste, los esquemas de control vectorial, basados en el Principio de Campo Orientado; proporcionar el ajuste de la magnitud y la fase de las cantidades de vectoriales del motor. En los sistemas de accionamiento de alto rendimiento controlados por vectores,

el par instantáneo y el campo magnético del motor están regulados, tanto en régimen permanente como en condiciones de funcionamiento transitorias. Los sistemas de control escalar son más simples que los que emplean control vectorial, pero la superioridad de este último desde el punto de vista del rendimiento dinámico de la unidad es indivisible.

Posiblemente la técnica de control escalar más común utilizada en la práctica es la de Volts – Hertz constante (CVH), llamada así porque la magnitud del voltaje del estator se ajusta en proporción a la frecuencia, para mantener un flujo de estator aproximadamente constante en el motor. El sistema de control es simple, ya que solo se requiere retroalimentación de velocidad.

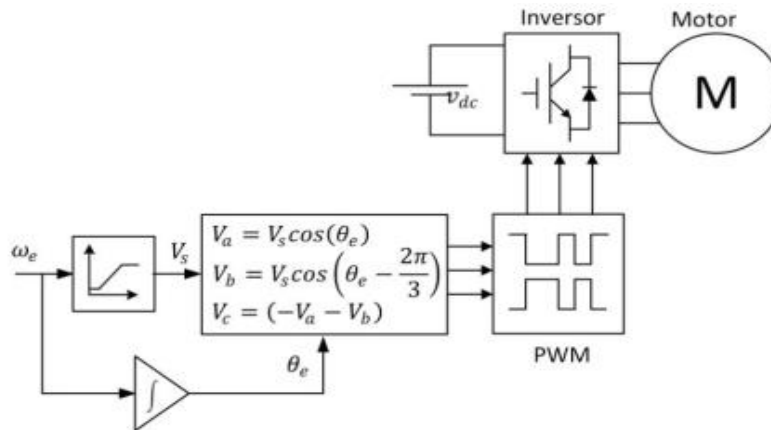


Fig.2.6. Sistema de control escalar básico [27].

2.4.2 Control vectorial

El principio básico del control vectorial es separar los componentes de la corriente del estator responsable de la producción de flujo y el par. El control vectorial en máquinas de C.A se obtiene controlando la magnitud, frecuencia y fase de la corriente del estator, mediante el control del inversor. Como el control del motor se obtiene controlando tanto la magnitud como el ángulo de fase de la corriente, este método de control recibe el nombre de control vectorial.

También llamado control por campo orientado (FOC). Estos esquemas utilizan el modelo dinámico del motor de inducción y permiten tratar al motor de inducción de la misma manera que el motor de corriente continua de excitación independiente. Es decir, admiten un control independiente del flujo del campo magnético y el par electromagnético, pues reducen el modelo no lineal del motor de inducción a un modelo referido a dos ejes ortogonales dq; de este modo, el control del flujo del campo magnético se logra regulando la componente de la corriente de eje directo d y el control del par electromagnético se logra regulando la corriente asociada al eje en cuadratura (q).

Para lograr la reducción del modelo no lineal del motor de inducción, se referencia el sistema trifásico alterno de corrientes del estator a un sistema de coordenadas ortogonales no estacionario que gira. Es posible asumir cualquier valor arbitrario para la velocidad de giro del sistema de coordenadas ortogonales. Sin embargo, se elige que el sistema de coordenadas ortogonales gire sincrónicamente con el flujo enlazado del rotor. En este nuevo sistema de referencia, las corrientes del estator pueden ser tratadas como un vector

espacial, de ahí el nombre de control vectorial. El paso siguiente es descomponer este vector espacial en dos componentes: una colineal con el flujo enlazado del rotor (denominada i_{sd}) y la otra en cuadratura (denominada i_{sq}). La primera componente resulta ser responsable del flujo enlazado del rotor y se designa como corriente de magnetización, la segunda componente ayuda en la generación del par electromagnético y se llama corriente activa.

Por la vía de esta transformación de coordenadas resulta entonces posible desacoplar el modelo matemático de la máquina de inducción y así poder controlar estas componentes en forma independiente de la misma manera que en un motor de corriente continua se controlan las corrientes de campo y armadura. Una vez determinados en este sistema de referencia no estacionario los valores requeridos de i_{sd} e i_{sq} a través de cualquier técnica de control (clásica o moderna), se aplica una transformación de coordenadas inversa que arroja por resultado las consignas de magnitud y fase de las corrientes alternas del estator. Ya que en el control vectorial las consignas de magnitud y fase de las corrientes alternas del estator dependen del vector espacial de flujo enlazado del rotor, existen dos métodos para calcular la posición del flujo magnético, estos son: *el control vectorial directo* y *el control vectorial indirecto* [1].

2.4.2.1 Control Vectorial Directo

En este método, la señal del flujo enlazado del rotor se adquiere de forma directa mediante sensores de efecto Hall o bobinas en el estator, o bien, se estima a partir del modelo del motor y las corrientes medidas en el estator. La velocidad del rotor no es requerida para obtener la información del ángulo espacial del flujo enlazado del rotor. En la Figura 2.7 se muestra el esquema básico de control vectorial directo [1].

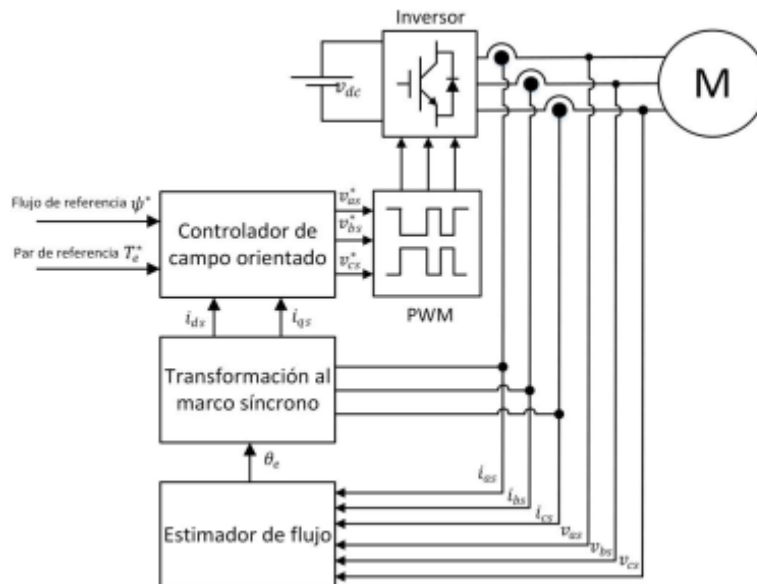


Fig.2.7. Esquema de un Control vectorial directo [1].

2.4.2.2 Control Vectorial Indirecto

Es una técnica muy fácil de implementar debido a que prescinde de medir o estimar el vector espacial de flujo enlazado del rotor. En este método, se impone desde el control el valor de flujo enlazado del rotor y el valor del par electromagnético, causando que el cálculo del vector espacial de flujo enlazado del rotor no dependa de las condiciones en los terminales de la máquina. Por lo tanto, no existe tal dependencia y no aparecen problemas de distorsión. Sin embargo, es necesaria la posición del rotor. En este método tanto el flujo enlazado del rotor como el par electromagnético son controlados en bucle abierto, existiendo sólo un bucle de control en lazo cerrado para la velocidad de giro del motor, lo cual hace que su desempeño dinámico no sea óptimo. En la Figura 2.8 se muestra el esquema básico de control vectorial indirecto [1].

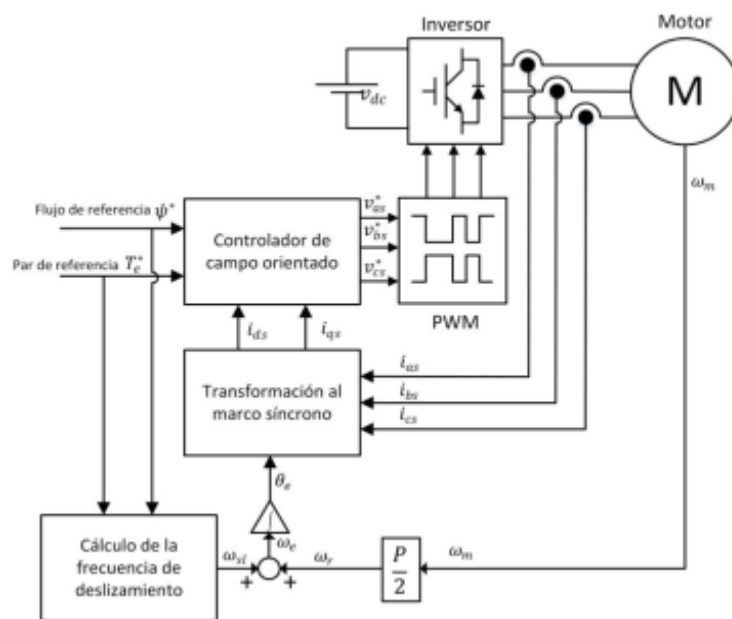


Fig.2.8. Esquema de un Control vectorial indirecto [1].

2.4.3 Control directo de torque

El control directo de torque es diferente del control vectorial de tal manera que no controla el flujo mediante el control de la corriente, pero se controla directamente el flujo mismo. El control directo de torque también es diferente del control vectorial en el sentido de que el marco de referencia aquí es el flujo del estator en lugar del flujo del rotor usado en el control vectorial. El control directo de torque es una extensión del control vectorial directo y el autocontrol directo. Como en el caso del control vectorial, en el método de control de par directo también, el flujo y el par son medidos o estimados y se usan como señales de retroalimentación para el controlador. La entrada al controlador de toque directo es el error de par, error de magnitud del vector espacial de flujo del estator y el ángulo del vector espacial de flujo del estator, a partir del cual se determinan los estados de los conmutadores de potencia. En base a esta información, un cierto vector de voltaje o combinación de vectores de voltaje se aplica directamente al inversor con un cierto tiempo promedio. Esto le da a los accionamientos del motor de inducción una respuesta muy rápida [27].

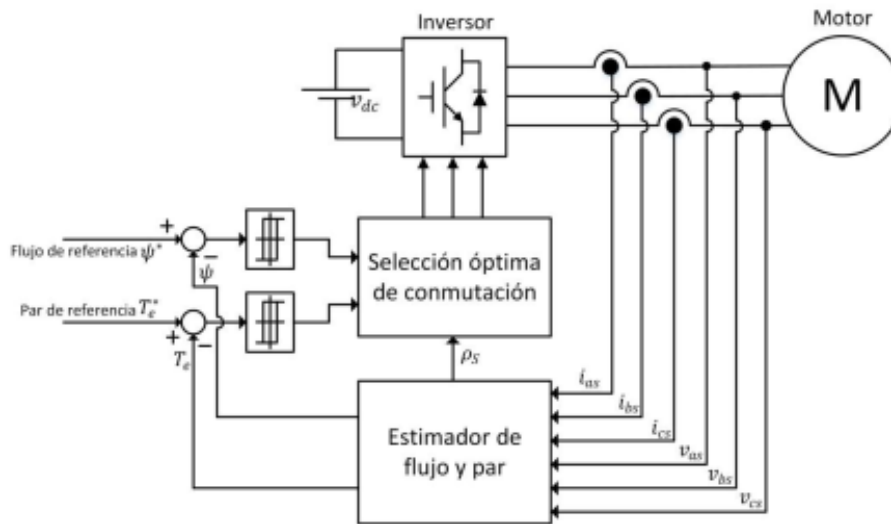


Fig.2.9. Diagrama a bloques del sistema de control DTC [27].

2.5 Selección de la técnica de control a utilizar

El estudio de las características y prestaciones de cada una de las técnicas de control para motores de inducción, nos permite justificar el empleo del *control vectorial*; debido a su robustez y sobre todo a la posibilidad de obtener un comportamiento dinámico similar al de un motor de corriente directa.

2.6 La transformación de Clarke

Esta transformada permite al modelo matemático de un sistema trifásico representado por abc (denotado así por las tres fases), convertirse en un sistema estacionario de dos variables ortogonales, denotados como α (alfa) y β (beta).

Tal como se muestra en la figura 2.14 el eje α coincide con el eje de la fase a , y el eje β está atrasado 90° [29]. Es una transformación que no modifica el módulo del vector. Pero no resuelve el problema de la dependencia del ángulo de rotación θ_r [30].

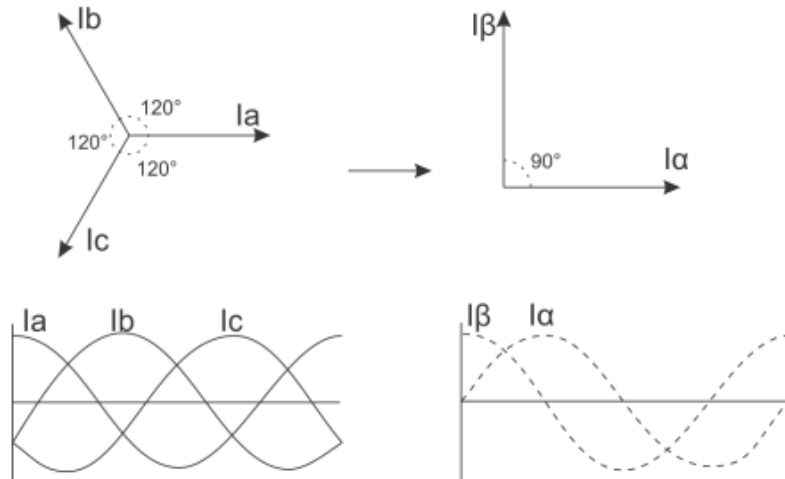


Figura 2.14 Transformada de Clarke.

En la ecuación 2.6.1 se muestra la transformada de Clarke:

$$\begin{pmatrix} I_\alpha \\ I_\beta \\ I_\gamma \end{pmatrix} = \frac{2}{3} \begin{pmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{pmatrix} \begin{pmatrix} I_a \\ I_b \\ I_c \end{pmatrix} \quad (2.6.1)$$

La transformación inversa $\alpha - \beta$ es:

$$\begin{pmatrix} I_a \\ I_b \\ I_c \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} & 1 \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} & 1 \end{pmatrix} \begin{pmatrix} I_\alpha \\ I_\beta \\ I_\gamma \end{pmatrix} \quad (2.6.2)$$

Las variables de un sistema trifásico al marco $\alpha\beta$, son señales sinusoidales bifásicas.

2.7 La transformación de Park

La transformada de Park surge ante la necesidad de contar con un sistema de referencia en el que la corriente resultante del estator está representada por un par de componentes de magnitud constante. La transformada de Park permite lograr este objetivo aunque con ello trae un nuevo problema que es el cómo determinar el ángulo de giro del nuevo sistema de referencia.

En el sistema de referencia de Park, las nuevas componentes están expresadas en función del ángulo de giro del sistema (θ_e). Estas nuevas componentes ortogonales (que giran en sincronismo con la θ_r del rotor) reciben el nombre de componente de corriente en el eje directo (d) y componente de corriente en el eje en cuadratura (q). La primera de ellas está alineada con el torque del sistema, mientras que la segunda tiene la misma dirección que el flujo del rotor. En la Fig. 2.15 observamos este esquema [31].

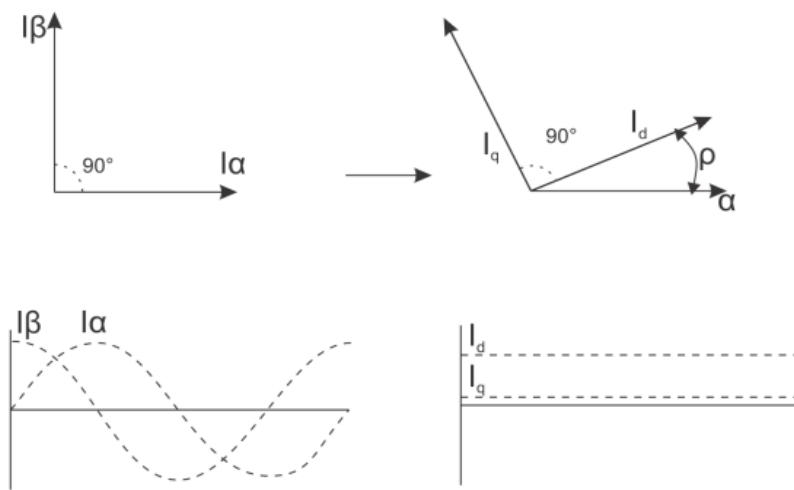


Figura 2.15 Transformada de Park.

La ecuación de transformación es de la forma:

$$\begin{pmatrix} I_d \\ I_q \\ I_0 \end{pmatrix} = \begin{pmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} I_\alpha \\ I_\beta \\ I_\gamma \end{pmatrix} \quad (2.7.1)$$

La transformada inversa dq esta dada por:

$$\begin{pmatrix} I_\alpha \\ I_\beta \\ I_\gamma \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} I_d \\ I_q \\ I_0 \end{pmatrix} \quad (2.7.2)$$

La importancia del marco $dq0$ radica en que las variables eléctricas en este marco de referencia son señales de CD, con lo cual el control del motor se simplifica considerablemente [30].

2.8 Redes neuronales artificiales

Los orígenes de las redes neuronales artificiales (ANN) están en el campo de la biología. El cerebro biológico consta de miles de millones de neuronas altamente interconectadas que forman una red neuronal. El procesamiento de la información humana depende de este sistema conexionista de células nerviosas. En base a esta ventaja del procesamiento de la información, las redes neuronales pueden explotar fácilmente el procesamiento local masivamente paralelo y las propiedades de almacenamiento distribuido en el cerebro.

El elemento fundamental de procesamiento de una red neuronal es una neurona. Este bloque de construcción de la conciencia humana abarca algunas capacidades generales. Básicamente, una neurona biológica recibe entradas de otras fuentes, las combina de alguna manera, realiza una operación generalmente no lineal sobre el resultado y luego genera el resultado final [37].

Una red neuronal artificial es un sistema de procesamiento de información que tiene ciertas características de rendimiento en común con la red neuronal biológica. Las redes neuronales artificiales (ANN) se han desarrollado como generalizaciones de modelos matemáticos de cognición humana.

2.8.1 Aplicaciones de las redes neuronales

Debido a su constitución y a sus fundamentos, las redes neuronales artificiales presentan un gran número de características semejantes a las del cerebro. Por ejemplo, son capaces de aprender de la experiencia, de generalizar de casos anteriores a nuevos casos, de abstraer características esenciales a partir de entradas que representan información irrelevante, etc. Esto hace que ofrezcan numerosas ventajas (aprendizaje adaptativo, auto organización, tolerancia a fallos, operación en tiempo real) y que este tipo de tecnología se esté aplicando en múltiples áreas [37].

Las ANN pueden utilizarse para identificar y controlar sistemas dinámicos no lineales porque pueden aproximar una amplia gama de funciones no lineales a cualquier grado de precisión deseado. Además, se pueden implementar en paralelo y, por lo tanto, un tiempo de cómputo más corto. Tienen inmunidad contra las ondulaciones armónicas y tienen capacidad para tolerar fallas. Desde la década de 1990, han aparecido varias investigaciones sobre las aplicaciones de redes neuronales en el campo de las máquinas eléctricas y la electrónica de potencia. En los últimos años, han aparecido muchos artículos que tratan sobre el uso de ANN en sistemas de modulación, en control, en la estimación de variables de estado, y en la identificación de los parámetros del motor de inducción [38].

2.8.2 Elementos de una red neuronal artificial

Cualquier modelo de red neuronal consta de dispositivos elementales de proceso: las neuronas. A partir de ellas, se pueden generar representaciones específicas, de tal forma que un estado conjunto de ellas puede significar una letra, un número o cualquier otro objeto. La Figura 2.16 muestra la estructura de una neurona abstracta con n entradas. Cada canal de entrada i puede transmitir un valor real x_i . La *función primitiva* f calculada en el cuerpo de la neurona abstracta se puede seleccionar arbitrariamente. Por lo general, los canales de entrada tienen un peso asociado, lo que significa que la información

entrante x_i se multiplica por el peso correspondiente w_i . La información transmitida se integra en la neurona y luego se evalúa la función primitiva.

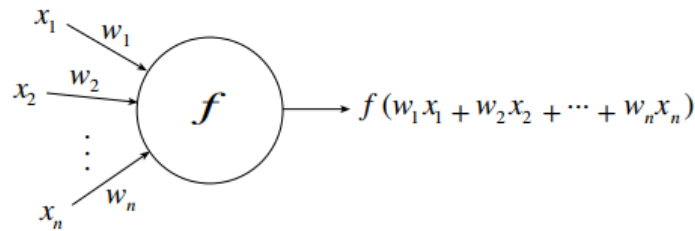


Figura 2.16 Entradas y salidas de una neurona [38].

Si concebimos cada nodo en una red neuronal artificial como una función primitiva capaz de transformar su entrada en una salida definida con precisión, entonces las redes neuronales artificiales no son más que redes de funciones primitivas. Los diferentes modelos de redes neuronales artificiales difieren principalmente en los supuestos sobre las funciones primitivas utilizadas, el patrón de interconexión y el momento de la transmisión de información [38].

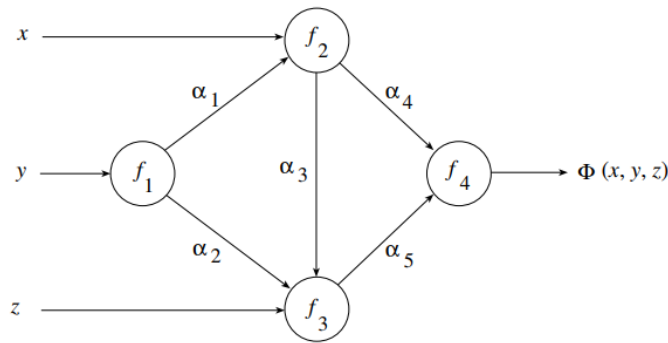


Figura 2.17 Modelo funcional de una red neuronal [38].

Las redes neuronales artificiales típicas tienen la estructura que se muestra en la figura 2.17. La red puede considerarse como una función Φ que se evalúa en el punto (x, y, z) . Los nodos implementan las funciones primitivas f_1, f_2, f_3, f_4 que se combinan para producir Φ . La función Φ implementada por una red neuronal se llamará *función de red*. Diferentes selecciones de los pesos $\alpha_1, \alpha_2, \dots, \alpha_5$ producen diferentes funciones de red. Por lo tanto, tres elementos son particularmente importantes en cualquier modelo de redes neuronales artificiales:

- Un conjunto de neuronas artificiales
- Un patrón de conectividad o arquitectura
- El algoritmo de aprendizaje para encontrar los pesos de la red

2.8.3 Arquitectura de una red neuronal

Las redes se definen de manera más precisa en términos de su arquitectura: Una arquitectura de red es una tupla (I, N, O, P) que consta de un conjunto I de sitios de entrada, un conjunto N de unidades informáticas, un conjunto O de sitios de salida y un conjunto E de bordes dirigidos ponderados. Un borde dirigido es una tupla (u, v, w) mediante la cual $u \in I \cup N, v \in N \cup O$ y $w \in \mathbb{R}$.

Los sitios de entrada son solo puntos de entrada para la información en la red y no realizan ningún cálculo. Los resultados se transmiten a los sitios de salida. El conjunto N consta de todos los elementos informáticos en la red. Tenga en cuenta que los bordes entre todas las unidades de computación están ponderados, al igual que los bordes entre los sitios de entrada y salida y las unidades de computación.

Las arquitecturas en capas son aquellas en las que el conjunto de unidades informáticas N se subdivide en l subconjuntos N_1, N_2, \dots, N_l de tal manera que solo las conexiones de las unidades en N_1 van a las unidades en N_2 , de las unidades en N_2 a las unidades en N_3 , etc. Los sitios de entrada solo están conectados a las unidades en el subconjunto N_1 , y las unidades en el subconjunto N_l son los únicos conectados a los sitios de salida. En la terminología habitual, las unidades en N_l son las unidades de salida de la red. Los subconjuntos de N_i se llaman las *capas* de la red. El conjunto de sitios de entrada se denomina *capa de entrada*, el conjunto de unidades de salida se denomina *capa de salida*. Todas las demás capas sin conexiones directas desde o hacia el exterior se denominan *capas ocultas*. Por lo general, las unidades en una capa no están conectadas entre sí (aunque algunos modelos neuronales hacen uso de este tipo de arquitectura) y los sitios de salida se omiten de la representación gráfica.

Una red neuronal con una arquitectura en capas no contiene ciclos. La entrada se procesa y se transmite de una capa a otra, hasta que se haya calculado el resultado final. La figura 2.18 muestra la estructura general de una arquitectura en capas.

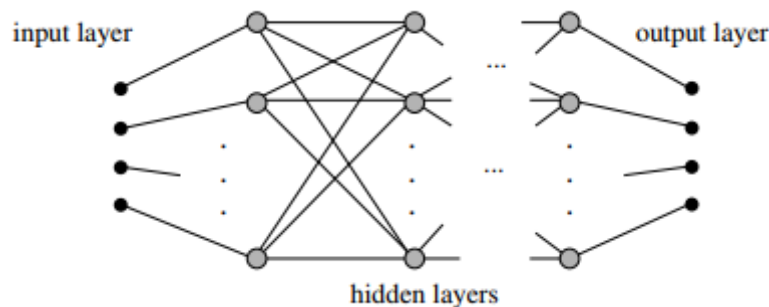


Figura 2.18 Arquitectura genérica de una red [38].

En arquitecturas en capas, normalmente todas las unidades de una capa están conectadas a todas las demás unidades en la siguiente capa. Si hay m unidades en la primera capa y n unidades en la segunda, el número total de pesos es nm . El número total de conexiones puede llegar a ser bastante grande.

2.8.4 Mecanismo de aprendizaje para una red neuronal

Un algoritmo de aprendizaje es un método adaptativo mediante el cual una red de unidades informáticas se autoorganiza para implementar el comportamiento deseado. Un paso de corrección se ejecuta iterativamente hasta que la red aprende a producir la respuesta deseada. El algoritmo de aprendizaje es un ciclo cerrado de presentación de ejemplos y correcciones a los parámetros de la red, como se muestra en la Figura 2.19.

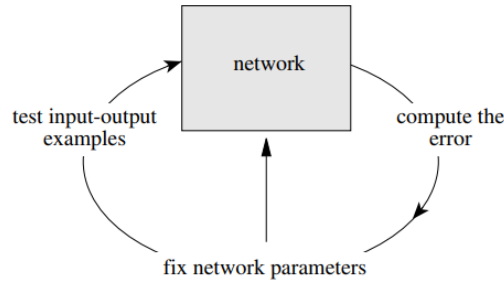


Figura 2.19 Proceso de aprendizaje en un sistema paramétrico [38].

Los algoritmos de aprendizaje se pueden dividir en métodos *supervisados* y *no supervisados*. El aprendizaje supervisado denota un método en el que algunos vectores de entrada se recopilan y presentan a la red. Se observa la salida calculada por la red y se mide la desviación de la respuesta esperada. Los pesos se corrigen de acuerdo con la magnitud del error en la forma definida por el algoritmo de aprendizaje. Este tipo de aprendizaje también se llama aprendizaje con un maestro, ya que un proceso de control conoce la respuesta correcta para el conjunto de vectores de entrada seleccionados.

El aprendizaje supervisado se divide además en métodos que utilizan refuerzo o corrección de errores. El *aprendizaje de refuerzo* se usa cuando después de cada presentación de un ejemplo de entrada-salida solo sabemos si la red produce el resultado deseado o no. Los pesos se actualizan en función de esta información (es decir, los valores booleanos *verdadero* o *falso*) para que solo el vector de entrada se pueda utilizar para la corrección de peso. Al aprender con la *corrección de errores*, la magnitud del error, junto con el vector de entrada, determina la magnitud de las correcciones a los pesos, y en muchos casos tratamos de eliminar el error en un solo paso de corrección.

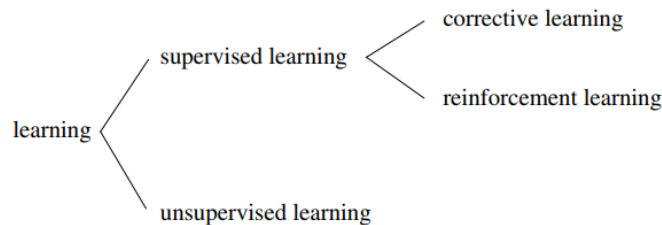


Figura 2.20 Tipos de algoritmo de aprendizaje [38].

El *aprendizaje no supervisado* se utiliza cuando, para una entrada dada, se desconoce la salida numérica exacta que debe producir una red. Como no hay un "maestro" disponible, la red debe organizarse para poder asociar grupos con unidades.

Otro criterio que se puede utilizar para diferenciar las reglas de aprendizaje se basa en considerar si la red puede "aprender" durante su funcionamiento habitual o si el aprendizaje supone la "desconexión" de la red, es decir su inhabilitación hasta que el proceso termine. En el primer caso se trataría de un aprendizaje *on line*, mientras que el segundo es lo que se conoce como aprendizaje *off line*. Cuando el aprendizaje es *off line* se distingue entre una fase de aprendizaje o entrenamiento y una fase de operación o funcionamiento, existiendo un conjunto de datos de entrenamiento y un conjunto de datos de test o prueba que serán utilizados en la correspondiente fase.

En las redes con aprendizaje off line los pesos de las conexiones permanecen fijos después que termina la etapa de entrenamiento de la red. Debido precisamente a su carácter estático, estos sistemas no presentan problemas de estabilidad en su funcionamiento. En las redes con aprendizaje *en línea* no se distingue entre fase de entrenamiento y de operación, de tal forma que los pesos varían dinámicamente siempre que se presente una nueva información al sistema.

2.9 Redes neuronales B-Spline

B-spline NN pertenece a una clase de redes neuronales conocidas como Redes de Memoria Asociativa (AMN), se utilizan generalmente para tareas de aproximación funcional. La red tiene una estructura fija y un conjunto de parámetros adaptativos que se entrenan de forma iterativa para lograr el comportamiento deseado.

Cuando se utilizan B-splines es posible modificar los datos localmente, y el cambio correspondiente en la respuesta de la red también es local. Esta es una de las características principales que hacen que las B-splines sean atractivas para el control adaptativo *en línea*; aprender sobre una parte del espacio de entrada afecta mínimamente la respuesta del modelo en su conjunto. La red B-spline se puede representar como un sistema de tres capas como el que se muestra en la figura 2.21. El sistema de tres capas consta de un espacio de entrada normalizada n-dimensional x , un conjunto de funciones básicas p y la salida de red a [41].

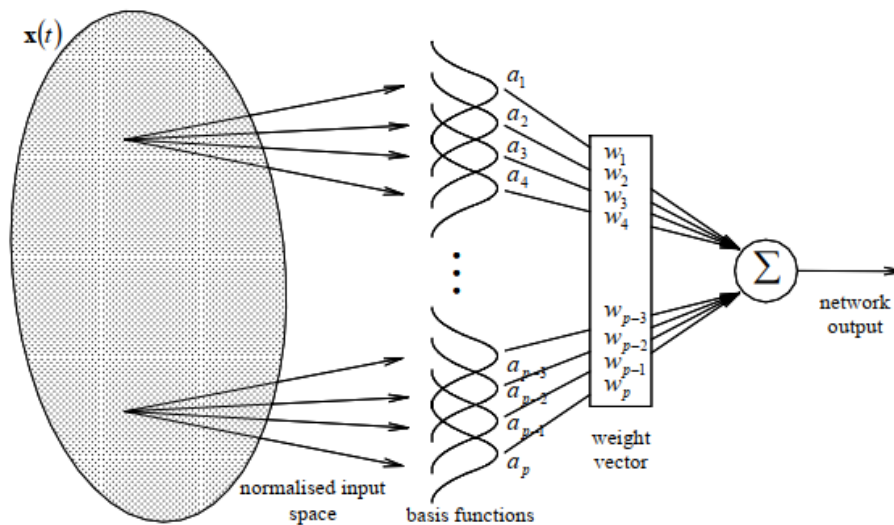


Figura 2.21. Diagrama esquemático de una B-spline NN de tres capas [41].

2.9.1 Entrada de la red B-Spline

Por definición, la red B-spline es una red AMN, por lo que las funciones básicas se definen en una red que normaliza el espacio de entrada. Esta normalización del espacio de entrada puede tomar diferentes formas; podría definir una red n-dimensional en el espacio de entrada, o simplemente podría pasar las entradas sin modificación.

Para definir una red en el espacio de entrada, se debe dar un conjunto de n vectores de nodos, un vector de nodos para cada eje de entrada. Un conjunto de nodos que generalmente se colocan en diferentes posiciones componen un vector de nodo. Además,

generalmente hay un número diferente de nodos en cada vector de nodo. Por lo tanto, el diseñador puede incorporar conocimiento a priori en el diseño de la red cuando se especifica la posición de estos nodos. Además, se pueden introducir nuevos nodos en la red B-spline, y la red antigua se puede reproducir exactamente por la nueva. Sin embargo, el costo computacional depende exponencialmente de n . Más específicamente, se debe especificar un vector de nodos para cada eje de entrada, que se compone de nodos interiores y exteriores.

Los r_i nodos interiores $\lambda_{i,j}$, ($j = 1, \dots, r_i$, $i = 1, \dots, n$) son ordenados para satisfacer la relación:

$$x_i^{min} < \lambda_{i,1} \leq \lambda_{i,2} \leq \dots \leq \lambda_{i,r_i} < x_i^{max} \quad (2.9.1)$$

Donde x_i^{min} y x_i^{max} el mínimo y máximo valor de entrada i^{th} , respectivamente. Cuando ocurren dos o más nodos en la misma posición, esto se denomina nodo múltiple o coincidente. En los extremos de cada eje de entrada, se debe dar un conjunto de nodos exteriores que satisfagan:

$$\lambda_{i,-(k_i-1)} \leq \dots \lambda_{i,2} \leq \lambda_{i,0} = x_i^{min}, x_i^{max} = \lambda_{i,r_i+1} \leq \dots \lambda_{i,r_i+k_i} \quad (2.9.2)$$

Estos nodos son necesarios para generar las funciones básicas de orden k_i que están cerca del límite de la red. El espacio de entrada de la red es el dominio.

$$[x_1^{min}, x_1^{max}] \times \dots \times [x_n^{min}, x_n^{max}] \quad (2.9.3)$$

Así que los puntos exteriores solo se usan para definir estas funciones básicas y el extremo de la red de entrada [40].

2.9.2 Salida de la red B-Spline

La característica más importante del algoritmo B-spline es la salida suave de la red que se debe a la forma de las funciones básicas. Dado que el soporte de las funciones básicas está limitado, solo un pequeño número de pesos está involucrado en el cálculo de salida de la red, por lo que la red B-spline almacena y aprende información localmente.

Las salidas de las funciones base a_i , se encuentran en el intervalo $[0,1]$, cuando la salida es cero, la función base asociada y el peso correspondiente no contribuyen a la salida de la red. La entrada n-dimensional a la red se denota por x , y la salida de la red (escalar) por y . La salida i de la función base se denota por a_i y la salida de todas las funciones base en el tiempo está contenida en el vector p-dimensiona de entrada transformado $a(t)$.

El número de funciones básicas que contribuyen a la salida B-spline es una constante que está determinada por los parámetros de la red, y esto se llama parámetro de generalización ρ . Logrando una reducción en los resultados del costo computacional ya que solo las funciones de base ρ que participan en la salida necesitan ser calculadas, en lugar de p .

La función de salida de la red B-Splin es una combinación lineal de las salidas de la función base y generalmente es continua. Los coeficientes de la combinación lineal son algunos pesos ajustables w_i (el aprendizaje). Debido a que la salida depende linealmente de los pesos, el aprendizaje es simplemente un problema de optimización lineal. Se puede representar como:

$$y = \sum_{i=1}^p a_i w_i = a^T w \quad (2.9.4)$$

Las salidas de las funciones básicas dependen no linealmente de las entradas, por lo que la última expresión puede reescribirse en el tiempo t como:

$$y(t) = a^T(t)w(t-1) = a^T(x(t))w(t-1) \quad (2.9.5)$$

Donde a es un vector p -dimensional que contiene las salidas de base de función. w es el vector que contiene los pesos, y finalmente x es el vector de entrada a la red [40].

2.9.3 Aprendizaje de la red B-spline

Muchas reglas de entrenamiento de redes neuronales son simplemente algoritmos instantáneos (junto con variaciones no lineales). Las reglas de aprendizaje instantánea se formulan minimizando las estimaciones instantáneas de una función de rendimiento, que generalmente es el error de salida cuadrático medio (MSE), y los parámetros se actualizan mediante reglas de descenso de gradiente. La red B-spline depende linealmente de un conjunto de pesos, y son estos parámetros, que se actualizan utilizando las reglas básicas de aprendizaje. Antes de esta capa lineal, los algoritmos realizan una transformación fija y no lineal de la entrada, asignándola a una representación interna dispersa de alta dimensión. Las reglas de aprendizaje instantáneo pueden aprovechar esta estructura ya que solo actualizan los pesos que contribuyen a la salida, y el aprendizaje es local porque las entradas diferentes se asignan a diferentes conjuntos de pesos. Por lo tanto, parece que las leyes de aprendizaje instantáneo son apropiadas para entrenar estos vectores de peso. Para adaptar el conjunto de parámetros lineales (vector de peso), se debe evaluar el rendimiento actual de la red y, en general, se utiliza la red MSE [40].

La regla de aprendizaje instantáneo considerada puede formularse como un método de optimización de búsqueda de línea, ya que el algoritmo genera una ruta de búsqueda, a lo largo de la cual se actualizan los pesos, luego calcula un tamaño de paso que garantiza un aprendizaje estable. Tanto la ruta de búsqueda como el tamaño del paso se determinan utilizando información de entrenamiento limitada (a menudo solo un par entrada / salida).

Reglas de descenso de gradiente instantánea como la corrección de errores, o la regla de mínimos cuadrados mínimos normalizados (NLMS), son apropiadas para AMN y, como tales, son adecuadas para el aprendizaje en línea. En la regla NLMS que no tiene error de modelado o ruido de medición, la red aprende a interpolar los datos exactamente. La estructura básica de la regla NLMS es: el cambio en el vector de peso es igual al error de salida (escalado) multiplicado por el vector de entrada transformado.

Las reglas de corrección de errores instantáneas, como su nombre indica, actualizan el vector de peso de la red para reducir el error en la salida después de que cada par de entrenamiento se presente a la red. La actualización de los pesos $\Delta w(t-1)$ se calcula, de modo que la dirección de búsqueda siempre es paralela al vector de entrada transformado, pero el tamaño del paso se calcula en cada paso de tiempo, de modo que la información almacena la información de entrada / salida en el tiempo t sea almacenada exactamente por la red.

La regla de aprendizaje NLMS utiliza la estimación instantánea del gradiente del MSE para producir una regla de aprendizaje que actualiza el vector de peso en proporción al gradiente instantáneo negativo. La regla NLMS tiene la forma:

$$\Delta w(t-1) = \frac{\eta e_y(t)}{\|a(t)\|_2^2} a(t) \quad (2.9.6)$$

Donde η , es la relación de aprendizaje, $a(t)$ es el vector de entradas transformado, y $e_y(t)$ es la señal de error de salida ($\hat{y}(t) - y(t)$); $\hat{y}(t)$ y $y(t)$ son las salidas deseadas de la red y la salida real de la red B-spline, respectivamente.

Esta regla de actualización de peso NLMS tiene la forma: escalar más el vector de entrada transformado, y la magnitud del cambio de peso se normaliza por la magnitud del vector de entrada transformado. La actualización de peso es paralela al vector de entrada transformado. La simplicidad computacional de esta regla de aprendizaje es obvia, especialmente cuando el vector de entrada transformado es escaso; solo se actualizan los pesos que contribuyen a la salida.

Para la regla de aprendizaje NLMS, los resultados de la actualización de pes:

$$w_i(t) = w_i(t-1) + \frac{\eta e_y(t)}{\|a(t)\|_2^2} a(t) \quad (2.9.7)$$

La regla NLMS almacena el valor deseado exactamente cuándo $\eta = 1$. El aprendizaje estable (para un solo ejemplo de entrenamiento) ocurre si la tasa de aprendizaje se encuentra en el intervalo (0,2), y si $\eta = 1$, el ejemplo de entrenamiento se almacena después de solo una iteración (el algoritmo de aprendizaje fue formulado para incorporar esta restricción). En general, un valor grande de η asegura un aprendizaje inicial rápido y un valor más pequeño produce un mayor filtrado de ruido (desajuste de modelo y ruido de medición) [40].

2.9.4 Funciones base

La salida de la capa intermedia en una red B-spline está determinada por un conjunto de funciones base p que se definen en la red n -dimensional. Para cualquier entrada a una red B-spline, el número de funciones base distintas de cero es una constante y se denomina parámetro de generalización, que se denota por ρ . Para cada entrada, solo las funciones base ρ contribuyen a la salida.

Las funciones base son polinomios por partes de orden k , de modo que la salida de la red B-spline también es un polinomio por partes de orden k . Las funciones base se definen a través de la relación de recurrencia de tres términos.

$$N_k^j(x) = \left(\frac{x - \lambda_{j-k}}{\lambda_{j-1} - \lambda_{j-k}} \right) N_{K-1}^{j-1}(x) + \left(\frac{\lambda_j - x}{\lambda_j - \lambda_{j-k+1}} \right) N_{K-1}^j(x)$$

$$N_1^j(x) = \begin{cases} 1, & x \in I_j \\ 0, & \text{otro} \end{cases} \quad (2.9.8)$$

Donde λ_j es el j -ésimo nudo y $I_j = [\lambda_{j-1}, \lambda_j]$ es el j -ésimo intervalo. Las funciones base de orden $k = 1, \dots, 4$ se muestran en la figura 2.22 y es fácil observar que las funciones base y también la salida de la red se suavizan a medida que aumenta el orden. En cada

intervalo, la función base es un polinomio de orden k , y en los nudos los polinomios se unen suavemente. Por lo tanto, debe enfatizarse que la relación de recurrencia dada en la ecuación (2.9.8) simplemente genera segmentos polinomiales, como se puede ver en una inspección más cercana. En cada intervalo, los coeficientes k son necesarios para representar un polinomio de orden k , y los pesos k - spline y las funciones base contribuyen a la salida.

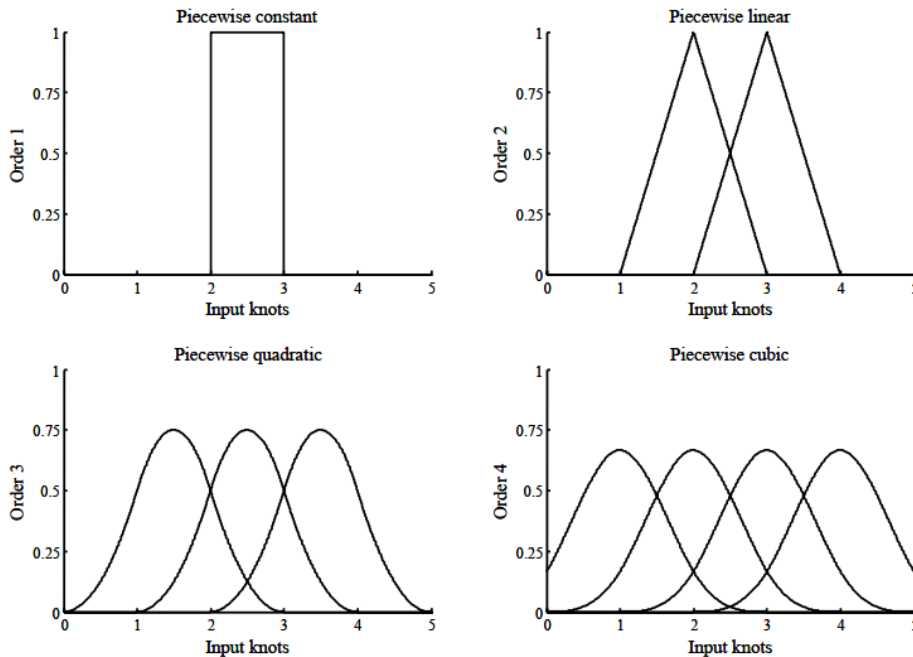


Figura 2.22. Funciones base de orden $k=1, \dots, 4$ [41].

Las funciones base se denotan por $N_k^j, N_k^{j+1}, \dots, N_k^{j+k-1}$ pueden determinarse recursivamente a partir de las funciones base de orden inferior, como se muestra en la Figura 2.23. Las funciones base de orden k se calculan como combinaciones lineales de las funciones base de orden $k - 1$.

Las funciones básicas de la red spline B están normalizadas, lo que significa que la suma de todas las salidas de funciones básicas es una constante. Las funciones base B-spline de orden k tienen un soporte que tiene k intervalos de ancho, por lo que elegir el orden de la aproximación polinómica también especifica el tamaño del soporte de la función base.

Las funciones base poseen un soporte acotado. Eso significa que solo un pequeño número de funciones básicas $\rho = k$ contribuyen a la salida de la red, por lo tanto, la información se almacena y aprende localmente a través de estas k funciones básicas.

Cuando la red se define como una función de spline, si hay r nodos coincidentes, las funciones base y la salida de la red tienen $(k - (r + 1))$ derivadas discontinuas en este punto. Esto permite que las discontinuidades se conecten a la red. Por ejemplo, cuando las funciones básicas del orden 2 se definen en un intervalo que contiene dos nudos coincidentes, la salida de la red es discontinua en este punto. El conocimiento cualitativo sobre la salida de la red se puede obtener al examinar los valores locales de los pesos. Los

límites superior e inferior siempre se logran para redes de órdenes 1 o 2, aunque generalmente no son nítidos para redes de orden superior.

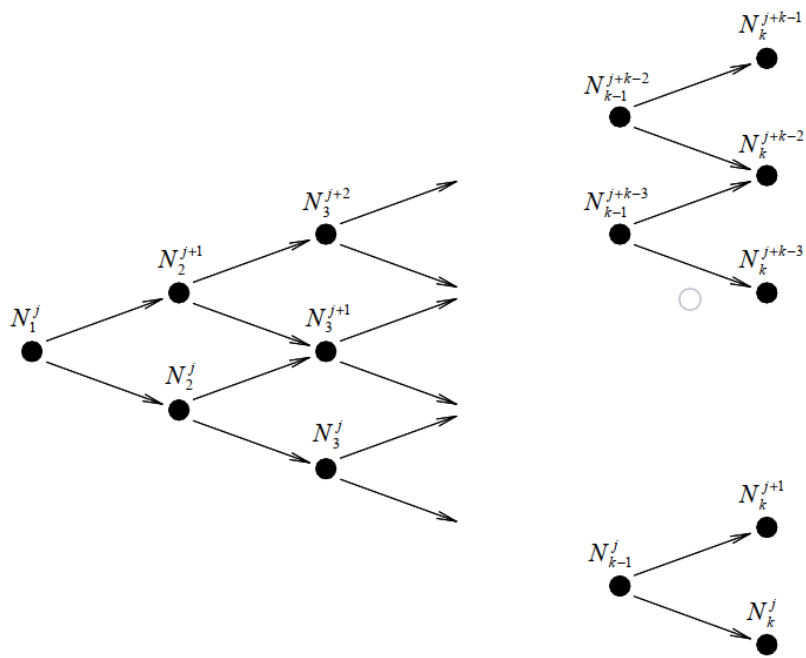


Figura 2.23. Matriz triangular utilizada para evaluar las k funciones de base B-spline [41].

Capítulo 3

MODELO MATEMÁTICO Y SIMULACIÓN DEL MOTOR DE INDUCCIÓN

El punto de partida para el diseño e implementación del sistema de control del motor de inducción, es el sistema de ecuaciones diferenciales que rigen su comportamiento dinámico. En el presente capítulo se obtendrá el circuito equivalente del motor de inducción, se determinará su modelo matemático y se realizará la simulación del comportamiento del motor de inducción (MI) en coordenadas dq .

3.1 Circuito equivalente del motor de inducción

El circuito equivalente de un motor tiene como objetivo, al igual que en el caso de los transformadores, el obtener una red que explique el comportamiento de la máquina, pero en la que no aparezca la acción transformadora entre los circuitos de primario y secundario [26].

En un motor de inducción su funcionamiento depende de que el circuito del estator induzca voltajes y corrientes en el circuito del rotor (acción transformadora). Puesto que la inducción de voltajes y corrientes en el circuito del rotor de un motor de inducción es, esencialmente, una acción de transformación, el circuito equivalente de un motor eléctrico de inducción terminará por ser muy similar al circuito equivalente de un transformador eléctrico [33].

Como en cualquier transformador, hay una resistencia R_1 y auto inductancia X_1 en los bobinados primarios (estator), los cuales se representan en el circuito equivalente de la máquina, en la Figura 3.1 se muestra el esquema simplificado por fase del motor en el que se especifican los parámetros anteriores. También, como en cualquier transformador con un núcleo de hierro, el flujo de la máquina está relacionado con el voltaje aplicado E_1 , mismo que se acopla con el secundario E_2 por medio de un transformador ideal de relación de espiras a_{ef} .

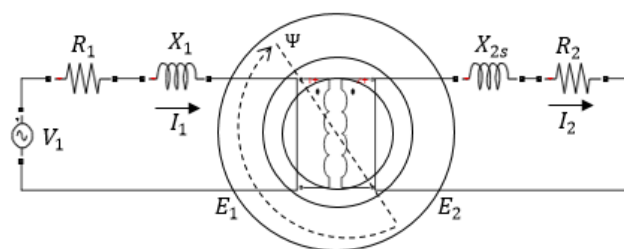
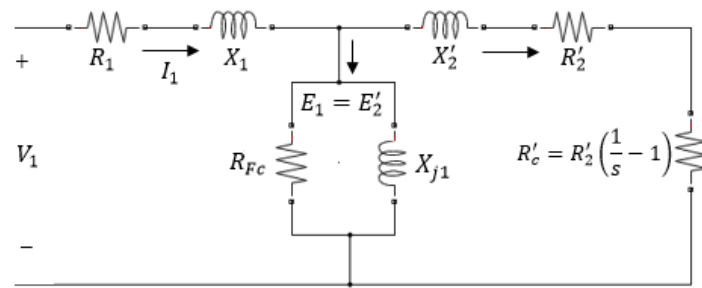


Fig.3.1 Circuito equivalente por fase del motor asíncrono trifásico.

De la Figura 3.1, podemos observar que la impedancia del rotor está formada por la resistencia R_2 y la reactancia X_{2s} , estando este devanado en cortocircuito. Las ecuaciones eléctricas correspondientes se obtendrán aplicando la segunda ley de Kirchhoff a las mallas del primario y secundario, resultando:

$$V_1 = E_1 + R_1 I_1 + jX_1 I_1, \quad E'_2 = R'_2 I'_2 + R'_c I'_2 + jX'_2 I'_2 \quad (3.1.1)$$

El circuito equivalente resultante se puede apreciar en la Figura 3.2.



3.2 Circuito equivalente exacto del motor de inducción.

3.2 Modelo dinámico del motor de inducción

El modelo matemático del motor de inducción debe conocerse para diseñar un controlador de velocidad. Conocido el modelo del sistema a controlar se puede realizar simulaciones de los algoritmos de control al que será sometido, teniendo en cuenta que el modelo es solo una aproximación del sistema real. Sin embargo, el modelo debe incorporar todos efectos ocurridos durante estado estacionario y transitorio

El modelo matemático del motor de inducción es definido por un grupo de ecuaciones diferenciales que representan la dinámica del motor con exactitud. Dichas ecuaciones son obtenidas utilizando las leyes físicas que rigen el comportamiento del motor. En la Figura 3.3 se representa la sección transversal de un motor de inducción, que se empleara para obtener el modelo matemático [34]. Las ecuaciones de voltaje de los circuitos del rotor y el estator acoplados magnéticamente serán formuladas en un marco de referencia $d - q$.

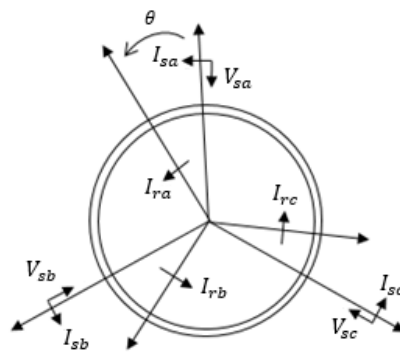


Figura.3.3 Sección transversal del motor de inducción trifásico.

3.2.1 Representación en vectores espaciales

El uso de vectores espaciales simplifica la ecuación de voltaje para la máquina de inducción y hace que la deducción del modelo sea más fácil de entender. En el modelado del motor de inducción, el vector espacial de corrientes trifásicas del estator se escribe como:

$$\vec{i}_s = \frac{2}{3} \left(i_a(t) + i_b(t)e^{j\frac{2\pi}{3}} + i_c(t)e^{j\frac{4\pi}{3}} \right) \quad (3.2.1)$$

$$\vec{i}_s = \frac{2}{3} I_s e^{j(\omega_s t + \phi)} \quad (3.2.2)$$

Donde ω_s es la velocidad angular de los vectores espaciales del estator, y el vector espacial de voltajes trifásicos del estator se define como:

$$\vec{u}_s = \frac{2}{3} \left(u_a(t) + u_b(t)e^{j\frac{2\pi}{3}} + u_c(t)e^{j\frac{4\pi}{3}} \right) \quad (3.2.3)$$

Donde $u_a(t)$, $u_b(t)$ y $u_c(t)$ son los voltajes de línea a neutro de cada fase. Del mismo modo, los vectores espaciales de voltaje y corriente del rotor se denotan por \vec{u}_r y \vec{i}_r . La dinámica del rotor juega un papel importante en el modelo matemático de un motor de inducción.

Para el motor de inducción de jaula de ardilla, el vector espacial de voltajes del estator con respecto a su propio sistema de bobinado es:

$$\vec{u}_s = R_s \vec{i}_s + \frac{d\vec{\psi}_s}{dt} \quad (3.2.4)$$

Donde R_s es la resistencia del estator y $\vec{\psi}_s$ es el vector espacial del flujo del estator. Con respecto a su propio sistema de bobinado, la ecuación de voltaje del vector espacial del rotor es:

$$\vec{u}_r^* = R_s \vec{i}_s^* + \frac{d\vec{\psi}_r^*}{dt} = 0 \quad (3.2.5)$$

Donde R_r es la resistencia del rotor. \vec{i}_r^* , \vec{u}_r^* y $\vec{\psi}_r^*$ son las corrientes, el voltaje y el flujo del rotor en forma de vector espacial con respecto al marco de referencia del rotor. Debido al cortocircuito del devanado del rotor, el vector de voltajes del rotor en (3.2.5) siempre es igual a cero.

Para sincronizar el marco de referencia de los devanados del rotor con el marco de referencia de los devanados del estator, se define un conjunto de vectores espaciales para cambiar el marco de referencia de los vectores espaciales del rotor.

$$\begin{aligned} \vec{u}_r &= \vec{u}_r^* e^{j\theta_e} \\ \vec{i}_r &= \vec{i}_r^* e^{j\theta_e} \\ \vec{\psi}_r &= \vec{\psi}_r^* e^{j\theta_e} \end{aligned} \quad (3.2.6)$$

Donde $\theta_e = \omega_e t$, θ_e es el ángulo eléctrico del rotor, y ω_e es la velocidad angular eléctrica del rotor. Ahora, multiplicamos ambos lados de (3.2.6) por el factor $e^{j\theta_e}$ y sustituyendo las transformaciones en la ecuación de voltaje del rotor.

$$\vec{u}_r = R_r \vec{i}_r + \frac{d\vec{\psi}_r}{dt} - j\omega_e \vec{\psi}_r = 0 \quad (3.2.7)$$

Donde se usa la siguiente igualdad

$$\frac{d\vec{\psi}_r}{dt} e^{j\theta_e} = \frac{d\vec{\psi}_r^*}{dt} - j\omega_e \vec{\psi}_r^* \quad (3.2.8)$$

Con los vectores espaciales de ambas corrientes de estator y el rotor, los flujos instantáneos de ambos devanados se basan en sus relaciones con las corrientes:

$$\vec{\psi}_s = L_s \vec{i}_s + L_h \vec{i}_r$$

$$\vec{\psi}_r = L_h \vec{i}_s + L_r \vec{i}_r \quad (3.2.9)$$

Donde L_h es la inductancia mutua, L_s y L_r son las inductancias del rotor y del estator, respectivamente.

Para eliminar la corriente del rotor \vec{i}_r de las ecuaciones y encontrar la relación entre el voltaje del estator y la corriente. Tomando la derivada del flujo del estator basada en (1.71), y sustituyendo el flujo del estator con las corrientes del estator y del rotor, la ecuación de voltaje del estator (1.68) se convierte en:

$$\vec{u}_s = R_s \vec{i}_s + L_s \frac{d\vec{i}_s}{dt} + L_h \frac{d\vec{i}_r}{dt} \quad (3.2.10)$$

Para eliminar la corriente del rotor \vec{i}_r , la ecuación de flujo del rotor (1.72) es usada para encontrar:

$$\vec{i}_r = \frac{1}{L_r} \vec{\psi}_r - \frac{L_h}{L_r} \vec{i}_s \quad (3.2.11)$$

Sustituyendo esto en (1.73):

$$\vec{u}_s = R_s \vec{i}_s + L_s \left(1 - \frac{L_h^2}{L_s L_r} \right) \frac{d\vec{i}_s}{dt} + \frac{L_h}{L_r} \frac{d\vec{\psi}_r}{dt} \quad (3.2.12)$$

Para eliminar la derivada del flujo del rotor de la ecuación anterior, se utiliza la ecuación de equilibrio de voltaje del rotor (ver (1.70)), lo que conduce a:

$$\frac{d\vec{\psi}_r}{dt} = -\frac{R_r}{L_r} \vec{\psi}_r + \frac{R_r L_h}{L_r} \vec{i}_s + j\omega_e \vec{\psi}_r \quad (3.2.13)$$

Donde la corriente del rotor se reemplaza con el flujo y la corriente del estator. Finalmente, se puede verificar que la ecuación de voltaje del estator se expresa en términos del flujo del rotor y la corriente del estator:

$$\vec{u}_s = \left(R_s + R_r \frac{L_h^2}{L_r^2} \right) \vec{i}_s + L_s \left(1 - \frac{L_h^2}{L_s L_r} \right) \frac{d\vec{i}_s}{dt} + \left(-\frac{L_h R_r}{L_r^2} + j\omega_e \frac{L_h}{L_r} \right) \vec{\psi}_r \quad (3.2.14)$$

Aunque todos los parámetros físicos están definidos en el modelo anterior, es posible tener expresiones más compactas utilizando los siguientes parámetros, factor de fuga:

$$\sigma = 1 - \frac{L_h^2}{L_s L_r} \quad (3.2.15)$$

Constante de tiempo del estator:

$$\tau_s = \frac{L_s}{R_s} \quad (3.2.16)$$

Constante de tiempo del rotor:

$$\tau_r = \frac{L_r}{R_r} \quad (3.2.17)$$

Coefficientes:

$$k_r = \frac{L_h}{L_r}$$

$$r_\sigma = R_s + R_r k_r^2 \quad (3.2.18)$$

$$\tau'_\sigma = \frac{\sigma L_s}{r_\sigma}$$

Con estas definiciones de parámetros, la ecuación de voltaje en forma de vector espacial simplemente se expresa como:

$$\vec{i}_s + \tau'_\sigma \frac{d\vec{i}_s}{dt} = \frac{k_r}{r_\sigma} \left(\frac{1}{\tau_r} - j\omega_e \right) \vec{\psi}_r + \frac{1}{r_\sigma} \vec{u}_s \quad (3.2.19)$$

Donde el flujo del rotor satisface la ecuación diferencial:

$$\vec{\psi}_r + \tau_r \frac{d\vec{\psi}_r}{dt} = j\omega_e \tau_r \vec{\psi}_r + L_h \vec{i}_s \quad (3.2.20)$$

3.2.2 Representación en marco de referencia $\alpha - \beta$

Una vez obtenido el modelo eléctrico en forma de vector espacial, se procede a convertir dicho modelo a un marco de referencia $\alpha - \beta$. El marco de referencia $\alpha - \beta$ es un marco de referencia estacionario, el eje del estator está alineado con la parte real (α) de MMF y el eje imaginario (β) en cuadratura.

Al descomponer el vector espacial de voltajes, corrientes y flujo en los ejes reales e imaginario, pueden ser representados mediante notación compleja:

$$\begin{aligned} \vec{u}_s &= u_{s\alpha} + ju_{s\beta} \\ \vec{i}_s &= i_{s\alpha} + ji_{s\beta} \\ \vec{\psi}_r &= \psi_{r\alpha} + j\psi_{s\beta} \end{aligned} \quad (3.2.21)$$

Para obtener el modelo dinámico en el marco de referencia $\alpha - \beta$, las variables anteriores se sustituyen en el modelo de vector espacial (3.2.19) y (3.2.20).

$$\begin{aligned} \frac{di_{s\alpha}}{dt} &= -\frac{1}{\tau'_\sigma} i_{s\alpha} + \frac{k_r}{r_\sigma \tau'_\sigma \tau_r} \psi_{r\alpha} + \frac{k_r}{r_\sigma \tau'_\sigma} \omega_e \psi_{r\beta} + \frac{1}{r_\sigma \tau'_\sigma} u_{s\alpha} \\ \frac{di_{s\beta}}{dt} &= -\frac{1}{\tau'_\sigma} i_{s\beta} - \frac{k_r}{r_\sigma \tau'_\sigma} \omega_e \psi_{r\alpha} + \frac{k_r}{r_\sigma \tau'_\sigma \tau_r} \psi_{r\beta} + \frac{1}{r_\sigma \tau'_\sigma} u_{s\beta} \\ \frac{d\psi_{r\alpha}}{dt} &= \frac{L_h}{\tau_r} i_{s\alpha} - \frac{1}{\tau_r} \psi_{r\alpha} - \omega_e \psi_{r\beta} \\ \frac{d\psi_{r\beta}}{dt} &= \frac{L_h}{\tau_r} i_{s\beta} + \omega_e \psi_{r\alpha} - \frac{1}{\tau_r} \psi_{r\beta} \end{aligned} \quad (3.2.22)$$

3.2.3 Representación en marco de referencia $d - q$

Cambiar el marco de referencia al marco $d - q$ es equivalente a rotar el vector espacial en el marco $\alpha - \beta$ en sentido horario de θ_s :

$$\begin{aligned} \vec{u}'_s &= \vec{u}_s e^{-j\theta_s} = u_{sd} + ju_{sq} \\ \vec{i}'_s &= \vec{i}_s e^{-j\theta_s} = i_{sd} + ji_{sq} \\ \vec{\psi}'_r &= \vec{\psi}_r e^{-j\theta_s} = \psi_{rd} + j\psi_{rq} \end{aligned} \quad (3.2.23)$$

Donde \vec{u}_s' , \vec{i}_s' y $\vec{\psi}_r'$ denotan los vectores espaciales referido al marco de referencia giratorio $d - q$. $\theta_s = \omega_s t$, ω_s es la velocidad angular de flujo síncrono en el estator.

En este marco de referencia giratorio $d - q$, el vector de flujo del rotor se fija al eje real del sistema de coordenadas. Por lo tanto, la componente de cuadratura $\vec{\psi}_r'$ es cero. Multiplicando (3.2.19) por el factor $e^{-j\theta_s}$ y sustituyendo en los vectores espaciales en el marco $d - q$ resulta:

$$\vec{i}_s' + \tau_\sigma' \left[\frac{d\vec{i}_s'}{dt} + j\omega_s \vec{i}_s' \right] = \frac{k_r}{r_\sigma} \left(\frac{1}{\tau_r} - j\omega_e \right) \vec{\psi}_r' + \frac{1}{r_\sigma} \vec{u}_s' \quad (3.2.24)$$

Donde se usa la siguiente igualdad:

$$\frac{d\vec{i}_s'}{dt} e^{-j\theta_s} = \frac{d\vec{i}_s'}{dt} + j\omega_s \vec{i}_s' \quad (3.2.25)$$

Basado en las componentes reales e imaginarias de (3.2.24) se obtiene el modelo eléctrico dinámico en el marco de referencia $d - q$.

$$\begin{aligned} \frac{di_{sd}}{dt} &= -\frac{1}{\tau_\sigma'} i_{sd} + \omega_s i_{sq} + \frac{k_r}{r_\sigma \tau_\sigma' \tau_r} \psi_{rd} + \frac{1}{r_\sigma \tau_\sigma'} u_{sd} \\ \frac{di_{sq}}{dt} &= -\omega_s i_{sd} - \frac{1}{\tau_\sigma'} i_{sq} - \frac{k_r}{r_\sigma \tau_\sigma'} \omega_e \psi_{rd} + \frac{1}{r_\sigma \tau_\sigma'} u_{sq} \end{aligned} \quad (3.2.26)$$

De manera similar, se puede demostrar que el flujo del rotor en el marco de referencia $d - q$ satisface:

$$\begin{aligned} \frac{d\psi_{rd}}{dt} &= \frac{L_h}{\tau_r} i_{sd} - \frac{1}{\tau_r} \psi_{rd} \\ 0 &= \frac{L_h}{\tau_r} i_{sq} - (\omega_s - \omega_e) \psi_{rd} \end{aligned} \quad (3.2.27)$$

Donde la componente q del flujo del rotor $\psi_{rq} = 0$. La ecuación (3.2.27) es una ecuación algebraica, no se incluye en el diseño del control. Sin embargo, permite obtener la ecuación empleada para la estimación de ω_s .

$$\omega_s = \omega_e + \frac{L_h}{\tau_r} \frac{i_{sq}}{\psi_{rd}} \quad (3.2.28)$$

ω_s es llamado deslizamiento estimado y ω_e es la velocidad angular eléctrica del rotor. Ya que el flujo del rotor ψ_{rd} no es medible directamente, la estimación del deslizamiento se realiza de forma alterna utilizando la corriente i_{sd} en reemplazo a ψ_{rd} .

$$\omega_s = \omega_e + \frac{1}{\tau_r} \frac{i_{sq}}{i_{sd}} \quad (3.2.29)$$

El torque electromagnético del motor de inducción es calculado usando el producto cruz de los vectores espaciales del flujo del rotor y la corriente del estator en el marco de referencia $d - q$.

$$T_e = \frac{3}{2} Z_p \frac{L_h}{L_r} \left(\vec{\psi}_r' \otimes \vec{i}_r' \right) \quad (3.2.30)$$

Donde Z_p es el número de pares de polos. El producto cruz, es calculado usando dos vectores de tres dimensiones $[\psi_{rd} \ 0 \ 0]$ y $[i_{sd} \ i_{sq} \ 0]$ ya que ψ_{rq} es cero. El resultado del producto cruz es el vector $[0 \ 0 \ \psi_{rd}i_{sq}]$. Así, en el marco de referencia $d - q$, el torque electromagnético es proporcional a $\psi_{rd}i_{sq}$

$$T_e = \frac{3}{2}Z_p \frac{L_h}{L_r} \psi_{rd}i_{sq} \quad (3.2.31)$$

El modelo mecánico del motor de inducción se deriva de la ecuación de movimiento de la rotación del motor, que se describe a continuación:

$$J_m \frac{d\omega_m}{dt} + f_d \omega_m = T_e - T_L \quad (3.2.32)$$

Donde $\omega_m(t)$ es la velocidad mecánica del rotor ($\omega_m = \frac{\omega_e}{Z_p}$), J_m representa la inercia del motor, f_d es el coeficiente de fricción, T_e y T_L denota el torque electromagnético y torque de carga, respectivamente. El modelo dinámico de la ecuación mecánica se obtiene al sustituir la ecuación (3.2.31) en la ecuación de movimiento (3.2.32).

$$\frac{d\omega_m}{dt} = -\frac{f_d}{J_m} \omega_m + \frac{3}{2} \frac{Z_p L_h}{L_r J_m} \psi_{rd}i_{sq} - \frac{T_L}{J_m} \quad (3.2.33)$$

En términos de la velocidad eléctrica del rotor.

$$\frac{d\omega_e}{dt} = -\frac{f_d}{J_m} \omega_e + \frac{3}{2} \frac{Z_p^2 L_h}{L_r J_m} \psi_{rd}i_{sq} - \frac{Z_p T_L}{J_m} \quad (3.2.34)$$

3.3 Resumen de parámetros del motor de inducción

Los parámetros físicos y eléctricos proporcionados por el fabricante del motor, nos permite tener la caracterización completa del motor de inducción, y con ello poder realizar la validación del comportamiento de nuestro modelo matemático. Estos parámetros corresponden a un motor de 1 HP y son listados en la Tabla 1 y Tabla 2.

Tabla 1. Especificaciones del motor

Parámetro	Valor
Tipo	Trifásico
Potencia	0.75 kW
Frecuencia	50 Hz
Numero de par de polos	2
Velocidad	1435 RPM
Voltaje RMS	415 V
Corriente de estator	1.75 A

Tabla 2. Parámetros del motor

$R_s(\Omega)$	$R_r(\Omega)$	$L_s(H)$	$L_r(H)$	$L_m(H)$	$J_m(kg.m^2)$	$f_d(Nm.s)$
11.2	8.3	0.6155	0.638	0.57	0.0024	0.0041

3.4 Validación del modelo del motor de inducción

Una vez que contamos con los parámetros definidos en las tablas y ecuaciones diferenciales que describen al motor, se emplea el Software MATLAB/Simulink como herramienta para validar el correcto comportamiento del mismo. En la Figura 3.4 se muestra el bloque que representa al motor de inducción tipo jaula de ardilla en el marco de referencia dq .

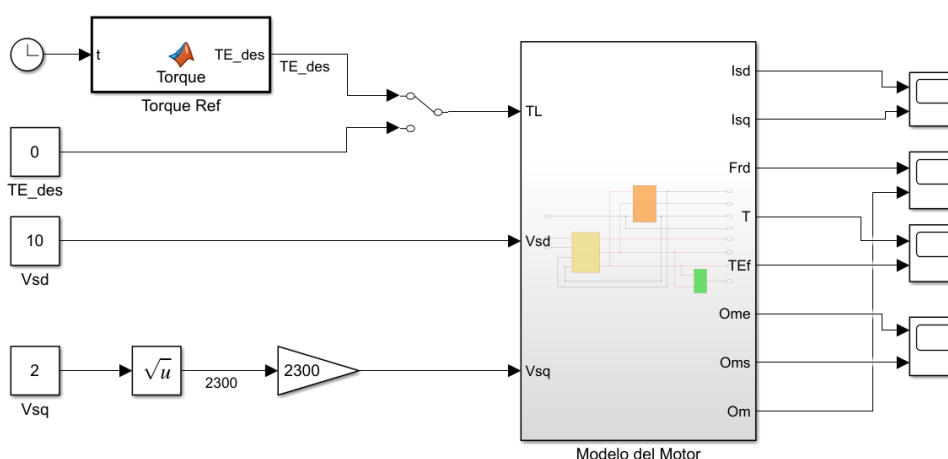


Figura 3.4 Bloque en MATLAB/Simulink del motor de inducción.

Se eligen dos señales de voltaje y un perfil de Par de carga, los cuales son aplicados a las terminales $V_{sd} = 10 V$, $V_{sq} = 3200 V$ y T_L , a fin de conocer la respuesta del modelo del motor, los perfiles aplicados son mostrados en la Figura 3.5.

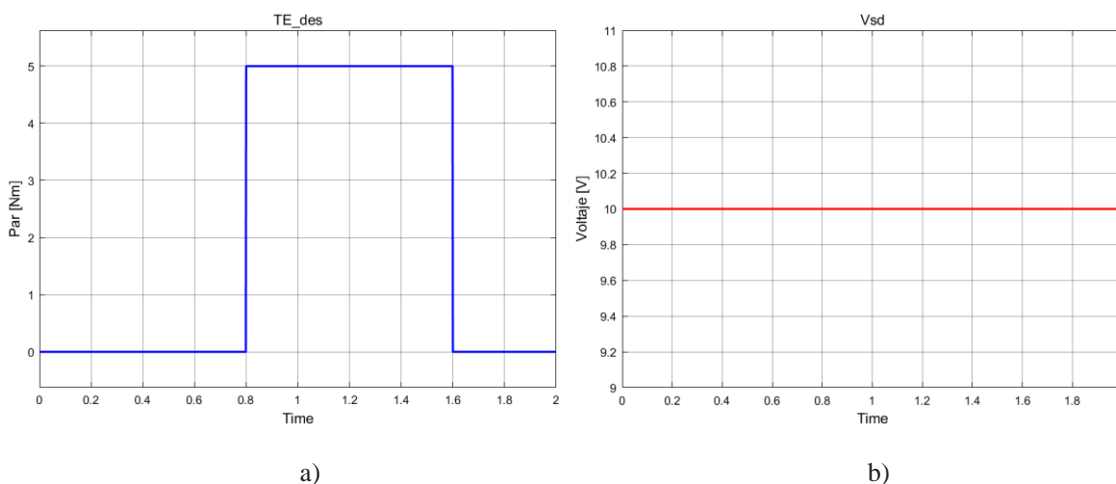


Figura 3.5 Perfiles aplicados al modelo del motor: a) Par de Carga y b) Voltaje $V_{sd} = 10 V$.

Las gráficas de la Figura 3.6 a 3.9 muestran los resultados de simulación del comportamiento del modelo del motor en Simulink. En la Figura 3.6 se puede ver la gráfica de: a) Flujo del rotor F_{rd} , y b) Velocidad mecánica ω_m .

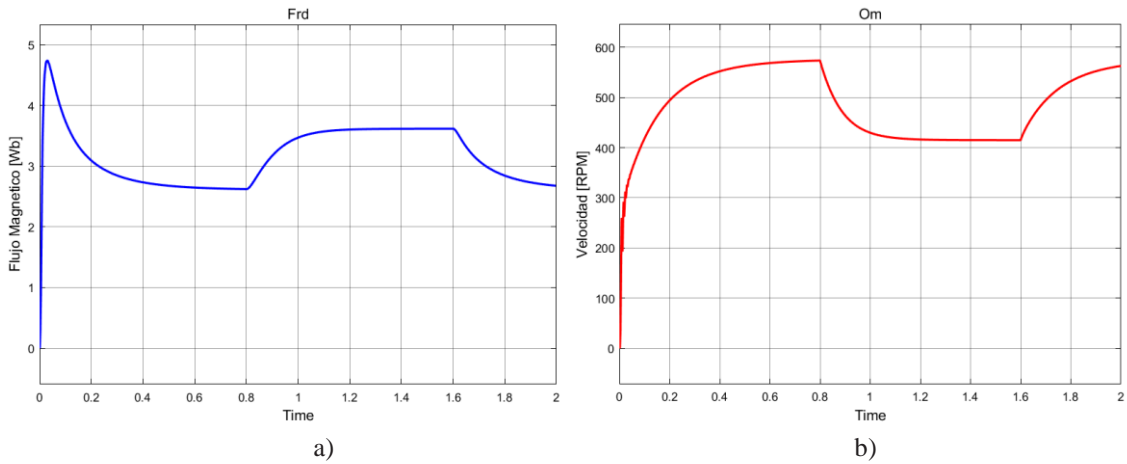


Figura 3.6 Resultados de simulación: a) Flujo del rotor y b) Velocidad mecánica.

Los resultados de la simulación de corrientes de estator, par eléctrico y par mecánico se muestran en la Figura 3.7 a 3.9.

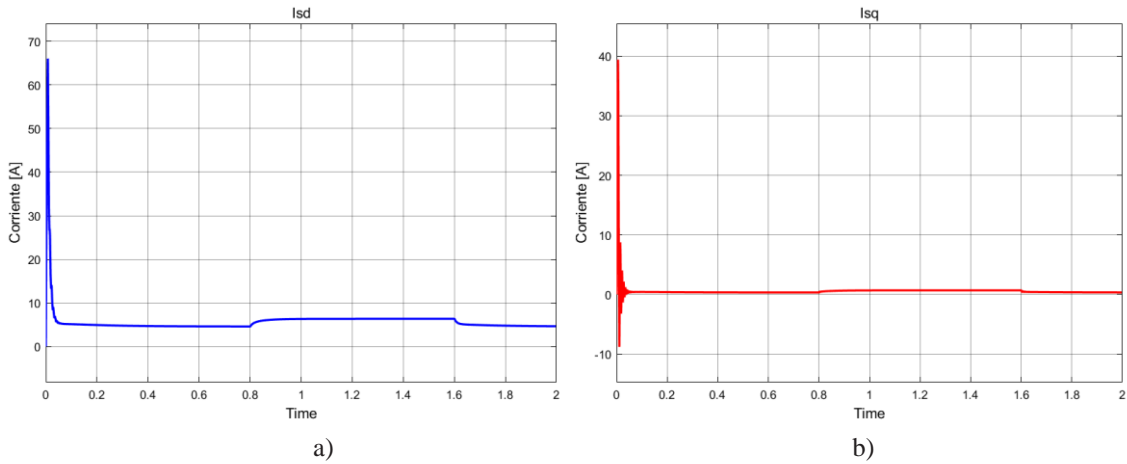


Figura 3.7 Resultados de simulación: a) Corriente de estator i_d , b) Corriente de estator i_q .

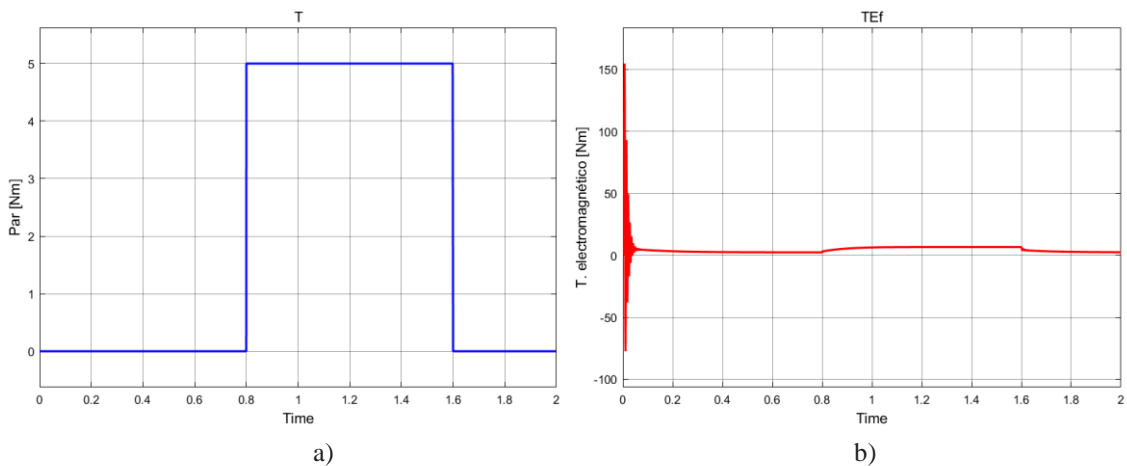


Figura 3.8 Resultados de simulación: a) Par mecánico, b) Par eléctrico.

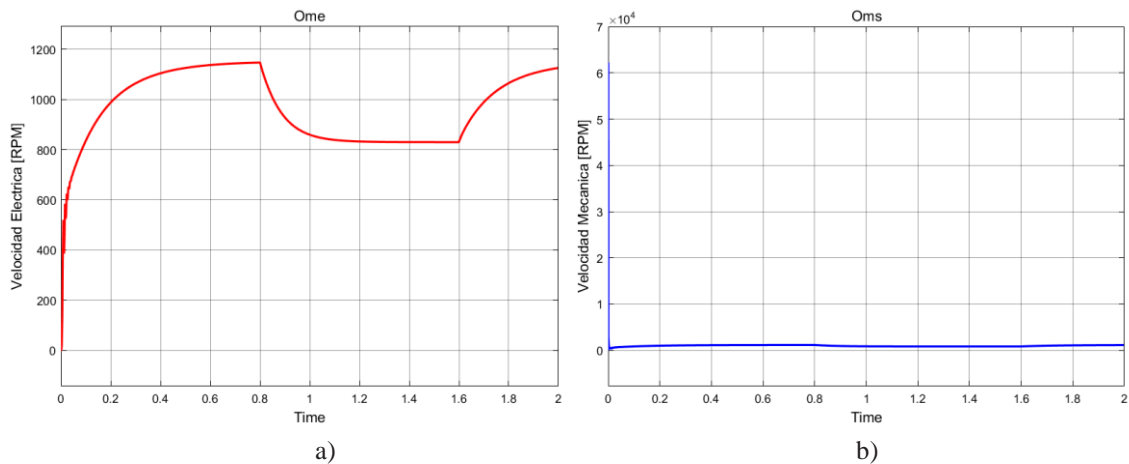


Figura 3.9 Resultados de simulación: a) Velocidad eléctrica, b) Velocidad angular.

3.5 Simulación del motor en vacío en PSIM

Con la finalidad de observar las capacidades y limitantes del motor antes de implementar una estrategia de control, se simula conectándolo directamente a la red eléctrica con una carga nula en el rotor. En la Figura 3.10 se muestra un diagrama de conexión de un motor de inducción utilizando el ambiente de PSIM. Se considera el motor en vacío.

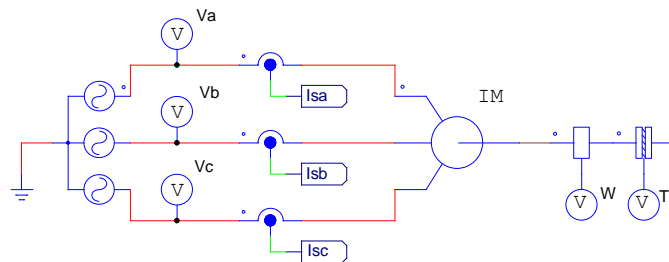


Figura 3.10 Esquema de conexión del motor de inducción en vacío.

En la Figura 3.11 a 3.12 se muestran los resultados de simulación de voltajes, corrientes y velocidad. La Figura 3.11 muestra la señal trifásica inyectada de amplitud ($V = 415v$) en las terminales de entrada del motor, y la señal correspondiente a las corrientes i_a, i_b, i_c del estator, del proceso de arranque.

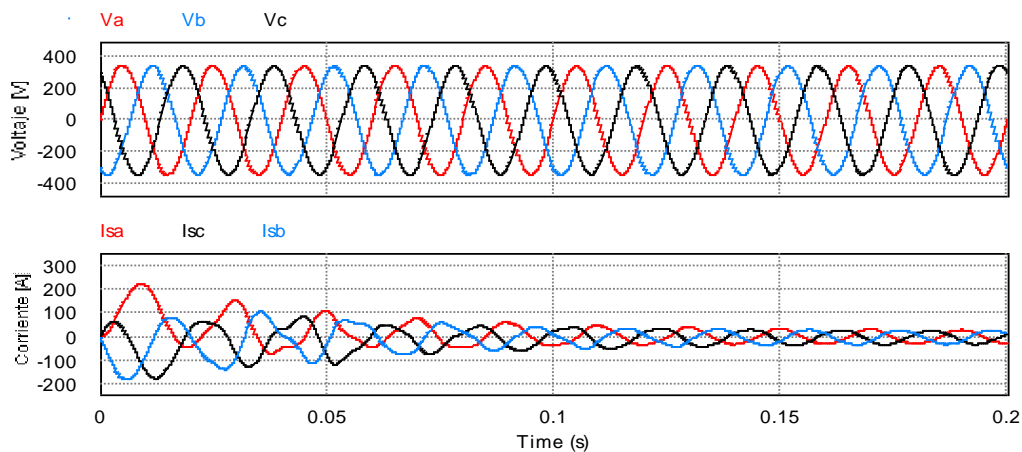


Figura 3.11 a) Voltaje trifásico de entrada, b) Corriente trifásica del estator.

En la Figura 3.12 se muestra: a) la velocidad alcanzada por el motor en vacío. b) Como el motor no presenta carga en su eje, el par de salida es cero.

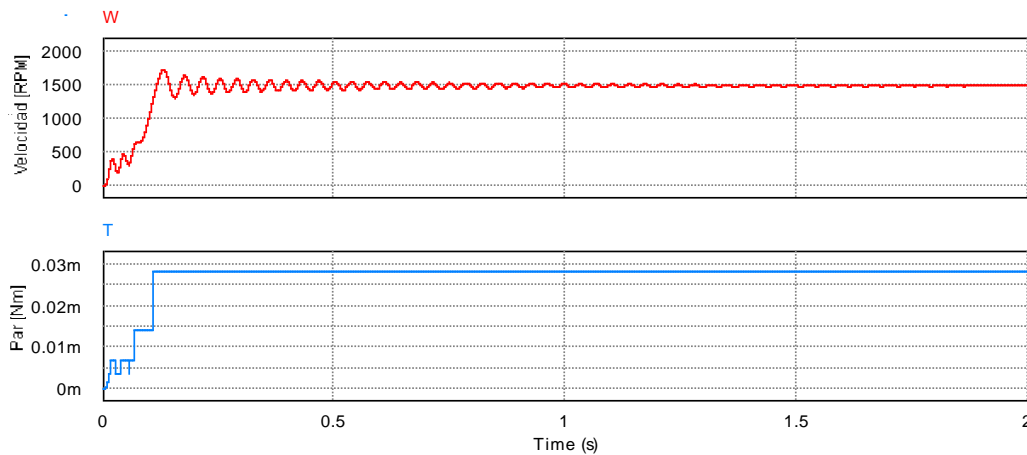


Figura 3.12 a) Velocidad del motor en vacío.

3.6 Convertidor CD-CA

Los convertidores CD-CA también son conocidos como inversores, tienen la función de transformar un voltaje de entrada de CD en un voltaje de CA. El principio de funcionamiento de los inversores es controlar la magnitud y frecuencia de la salida de voltaje de CA, dada una entrada de voltaje de CD constante.

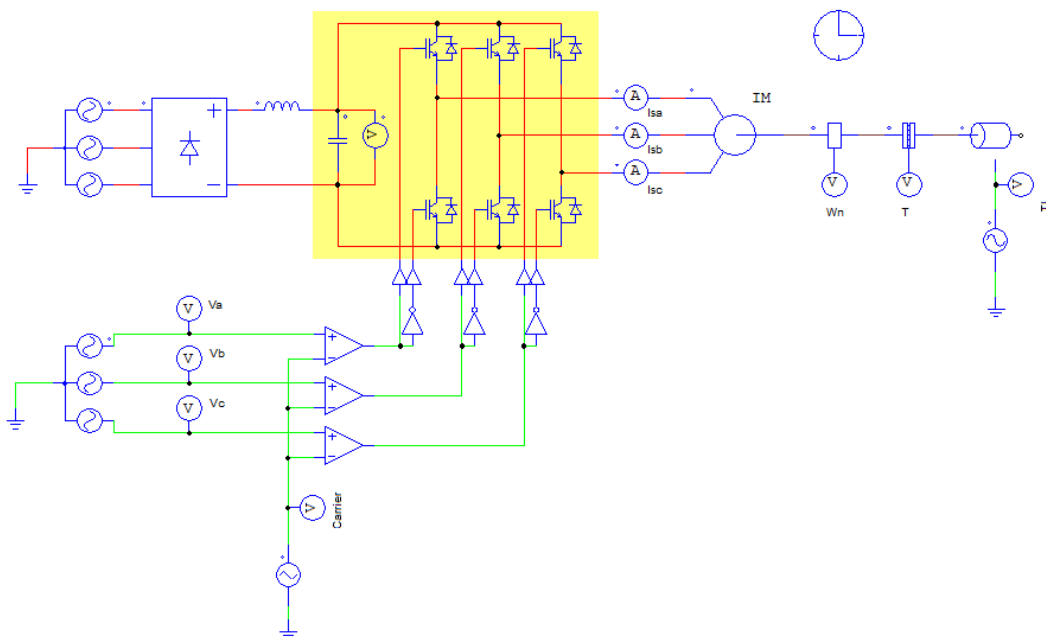


Figura 3.14 Diagrama de conexión del motor de inducción con inversor trifásico.

La Figura 3.14 se muestra un diagrama de conexión de un motor de inducción accionado con un inversor trifásico empleando el ambiente de simulación de PSIM. Se considera el motor con una carga variable. En la Figura 3.15 se muestran la señal sinusoidal trifásica empleada para la generación de los PWM (Pulse Width Modulation) encargados de conmutar los IGBT del convertidor.

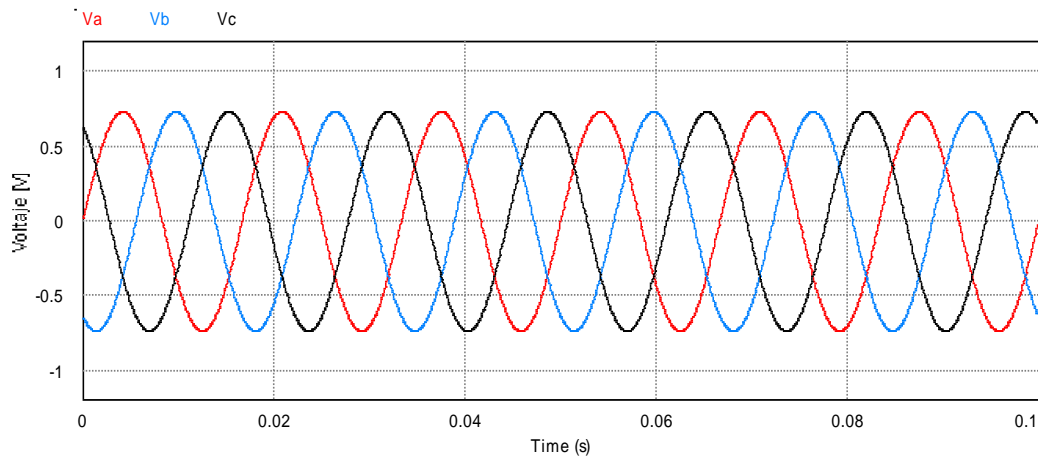


Figura 3.15 Señal de voltaje trifásica equilibrada.

Se muestran la señal trifásica de las corrientes I_a, I_b, I_c del estator, con una variación de corriente correspondientes al efecto de la carga aplicada (Figura 3.16).

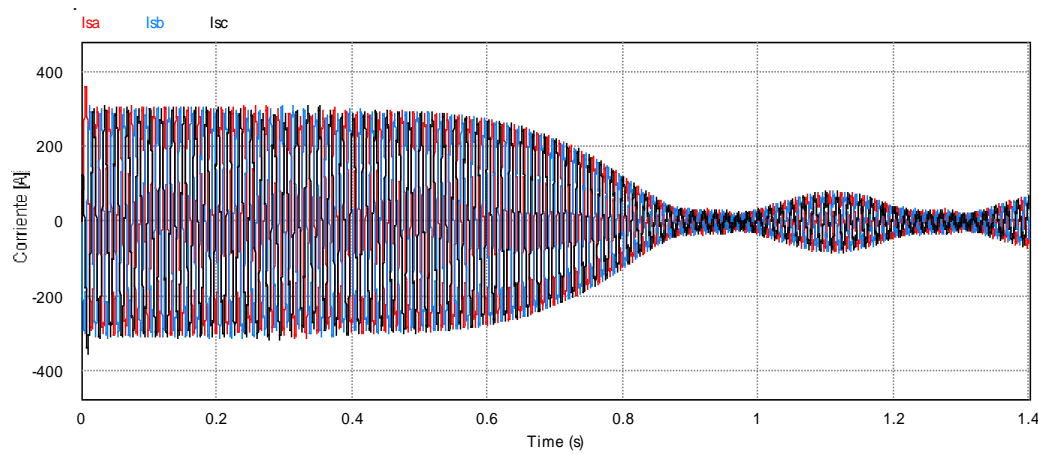


Figura 3.16 Señal trifásica de corrientes del estator.

En la Figura 3.17 se muestra: a) la velocidad de arranque y el b) par generado por el motor al colocar una carga variable en el eje del rotor, presenta varias caídas de velocidad respecto al valor de referencia en cada momento en el cual se aplica la carga.

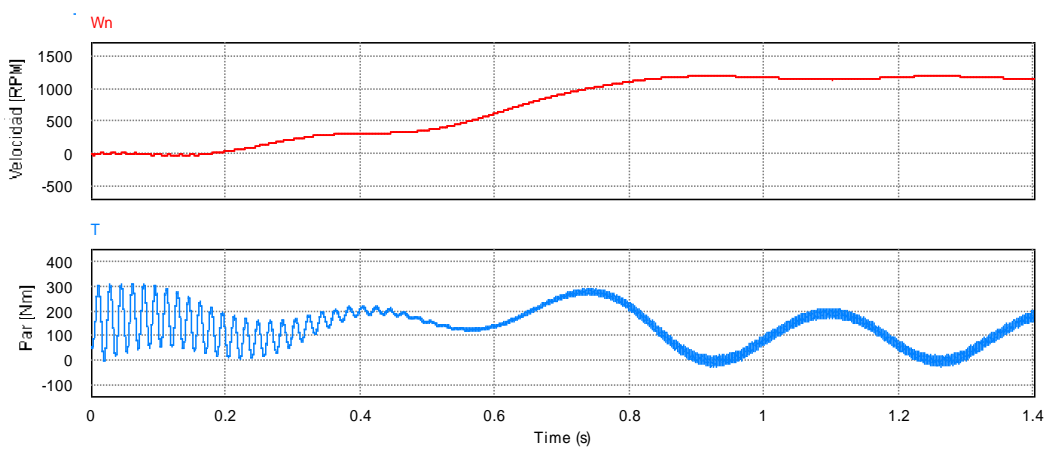


Figura 3.17 Velocidad y par en el eje del motor bajo el efecto de carga.

En la Tabla 3 se muestran los resultados de simulación obtenidos al someter al motor a diferentes valores de carga y alimentación constante.

Tabla 3. Velocidad del motor en lazo abierto con carga simulada en PSIM.

Par de carga	Velocidad (rpm)
0	1203
10	1197
20	1194
30	1191
40	1190

Como se puede observar en la Tabla 1, la velocidad del motor disminuye a medida que aumenta la carga en el eje. Con el fin de evitar este fenómeno, es necesario un *sistema de control de lazo cerrado* que permita mejorar el comportamiento del sistema manteniendo la velocidad constante a pesar de los diferentes valores de carga; es decir, lograr una respuesta tal que se logre controlar de forma independiente tanto el torque como la velocidad.

3.7 Control V/F para el motor de inducción

El control escalar mostrado en la Figura 3.18, también llamado Tensión-Frecuencia: (V/f) es el más sencillo y se basa en el hecho de que para mantener el flujo magnético constante (y en consecuencia el par), es necesario aumentar el voltaje a medida que se aumenta la frecuencia (la frecuencia es la que dicta la velocidad de un motor de inducción). Se cumple entonces una relación de proporcionalidad directa en la que: $V/f = Cte$. Es la curva Tensión-Frecuencia, que el variador intenta seguir en todo momento.

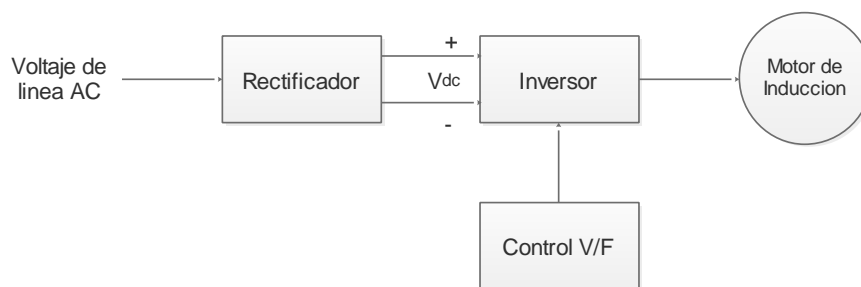


Figura 3.18 Diagrama de bloques de un control escalar o control V/F.

3.7.1 Diseño y simulación del control escalar V/f

Dado que la curva V/f depende de cada motor, se procedió con el diseño de un perfil de curva V/f en base a la tensión ($V = 96v$) y frecuencia nominal ($F = 250Hz$), en este caso en el perfil se impuso un límite inferior de voltaje conocido como VBUS del 20% (valor obtenido de manera experimental), tal como se muestra en la Figura 3.19.

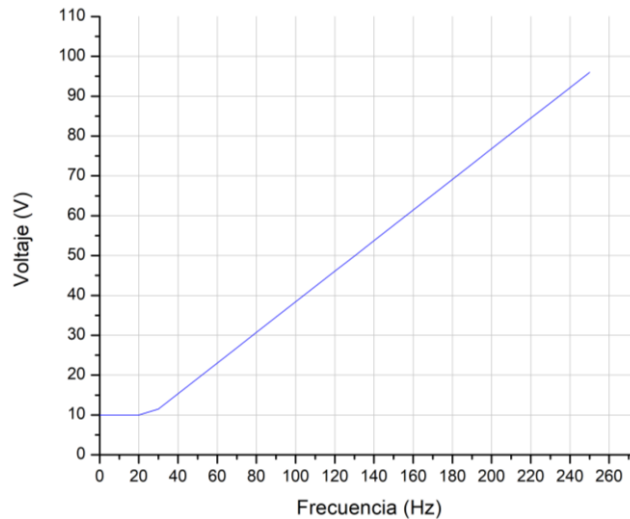


Fig.3.19. Curva V/f para el Motor HPEVS AC 50-26.28.11.

Para la simulación del control V/f se empleó el entorno PSIM en su versión 9. La forma más común de inversor consiste en un arreglo de 6 transistores IGBT, encargados de pasar un sistema de voltajes y frecuencias fijas a uno variable, ver Figura 3.20.

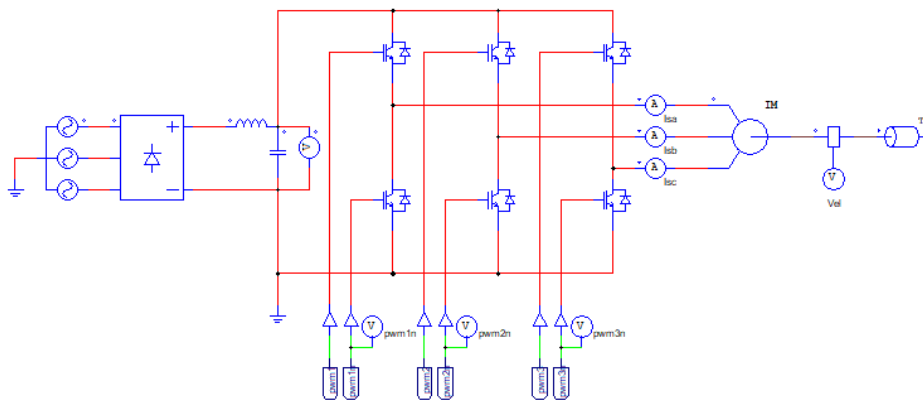


Fig.3.20. Diagrama en PSIM del accionamiento para el Motor de inducción.

El esquema V/f nos permite pasar de un sistema de voltajes y frecuencias fijas a un esquema de modulación por ancho de pulso o PWM variable, lo cual se ve reflejado como un aumento o disminución de la velocidad del motor.

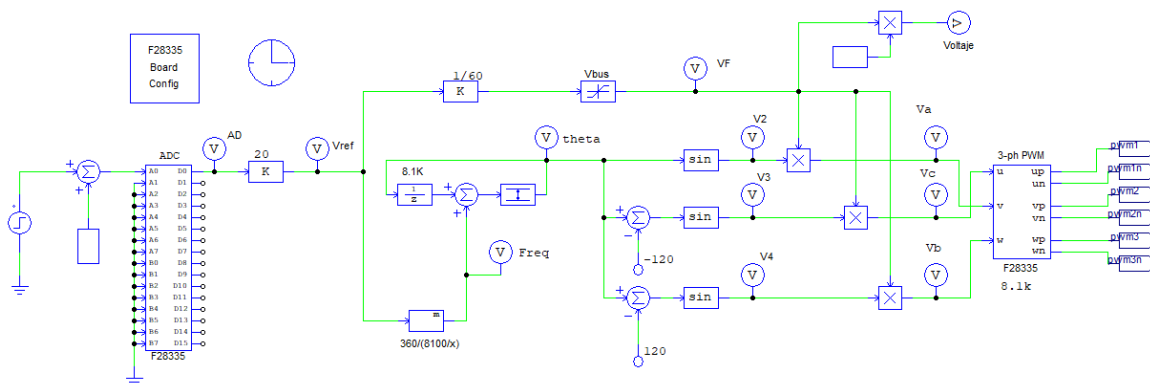


Fig.3.21. Diagrama en PSIM del control escalar V/f para el Motor de inducción.

En la modulación PWM se modifica el ciclo de trabajo de la señal periódica de entrada (senoidal), para controlar la cantidad de energía que se envía al motor. La señal moduladora es una señal (triangular) y la señal modulada es un tren de pulsos de distinto ancho, ver Figura 3.22 y 3.23.

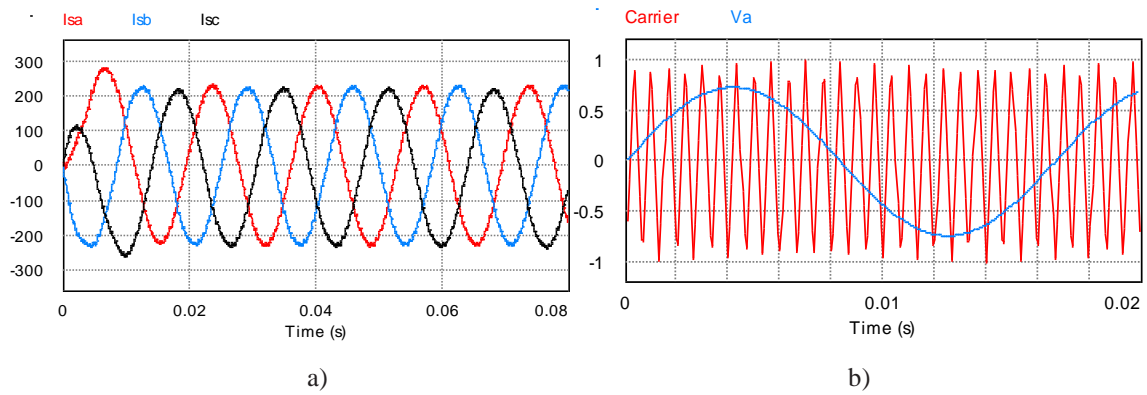


Fig.3.22. Esquema de modulación SPWM a) Señal senoidal trifásica, b) Señal portadora y moduladora.

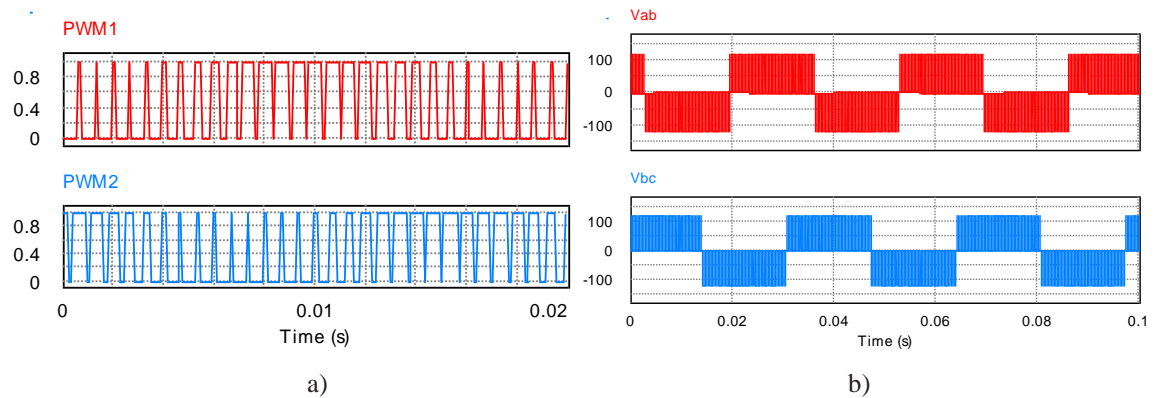


Fig.3.23. Esquema de modulación SPWM. a) Señales PWM, b) Voltajes de fase a fase.

A la entrada del regulador de voltaje – frecuencia se le coloca una señal de voltaje de referencia la cual permite aumentar o disminuir gradualmente la velocidad del motor. El resultado de este esquema se muestra en la Figura 3.24, a) La velocidad mecánica del rotor ω_m intentando alcanzar a la señal de vecindad de referencia propuesta ω_m^* , b) Señal trifásica de corriente consumida por el estator.

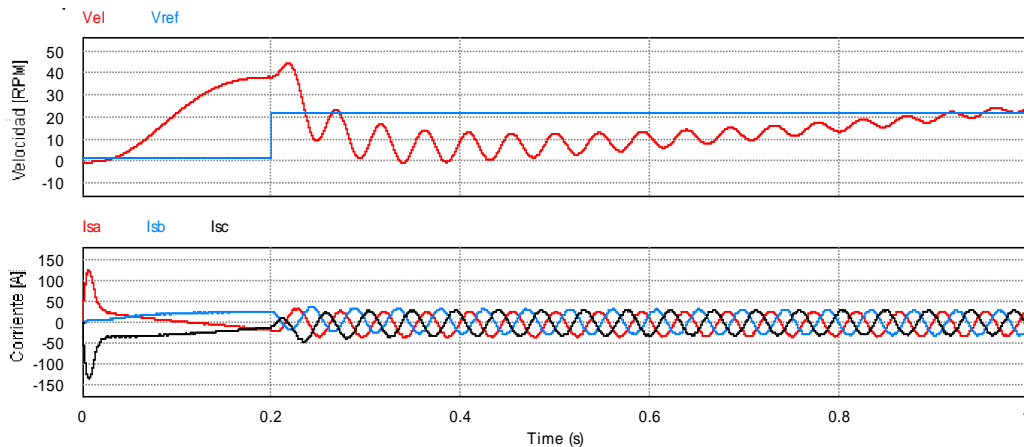


Fig.3.24. a) Señal de referencia - Velocidad del motor, b) Corrientes de estator.

3.7.2 Implementación del control escalar V/f

El motor de inducción HPEVS es considerado uno de los más eficiente para aplicaciones en vehículos eléctricos, es por ello que una vez que contamos con la simulación del control V/f procedemos a ponerlo a prueba en un sistema de tracción eléctrica (motor, acelerador, freno, etc.) y su respectiva instrumentación virtual para el monitoreo de velocidad, dirección y posición. En la Figura 3.25 se muestra el esquema completo del accionamiento eléctrico de un motor de inducción con rotor jaula de ardilla.

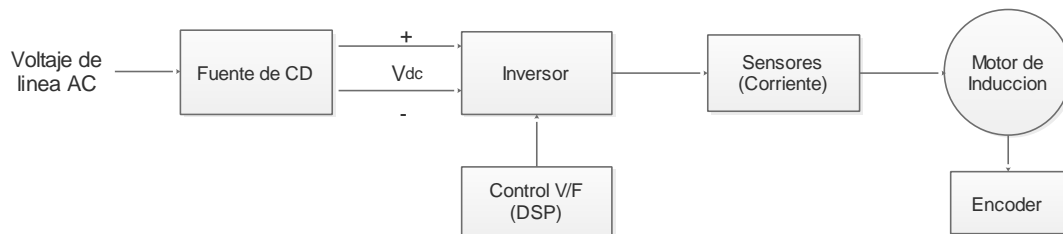


Figura 3.25. Diagrama de bloques del control V/f del Motor de Inducción.

Para esta aplicación la velocidad es controlada mediante el pedal mecánico de frenado y aceleración, este elemento modula la frecuencia de los PWMs según la presión ejercida por el pedal. El sistema de monitoreo se implementó en un sistema embebido de la marca 4D Systems (GEN4-70DCT), la función principal de la tarjeta GEN4 es el establece una comunicación serial con el DSP C200 para poder intercambiar información relacionada a la velocidad, posición y dirección del motor.

Dado que se requiere conocer la velocidad del rotor a cada instante, se empleó un encoder de cuadratura de 1024 posiciones, el cual se encuentra integrado en el eje del motor HPEVS AC 50-26.28.11. Al igual de los demás sensores requirió de una etapa de acondicionamiento de su señal debido a la limitante de tensión máxima (-1.5 a 1.5V) que soportan las entradas digitales del DSP C2000.

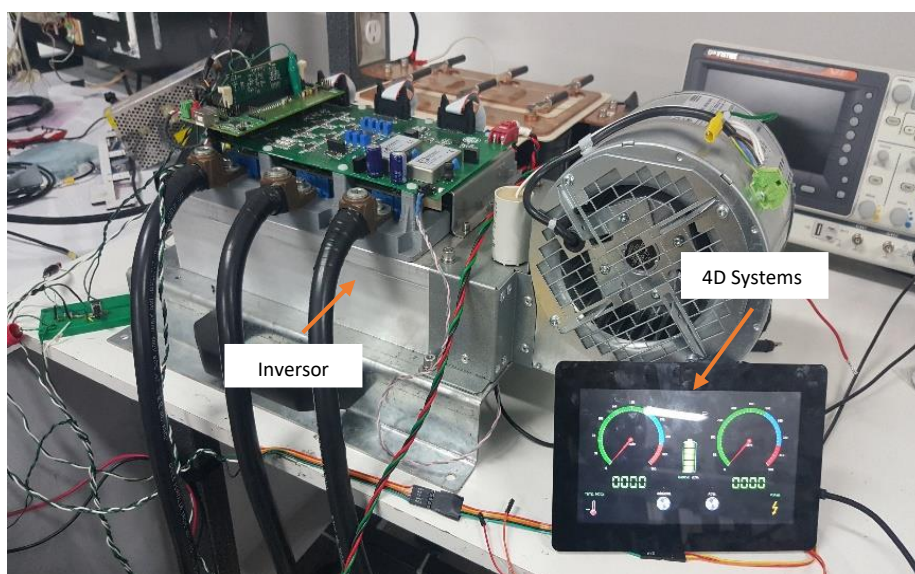


Fig.3.26. Inversor trifásico y sistema de instrumentación.

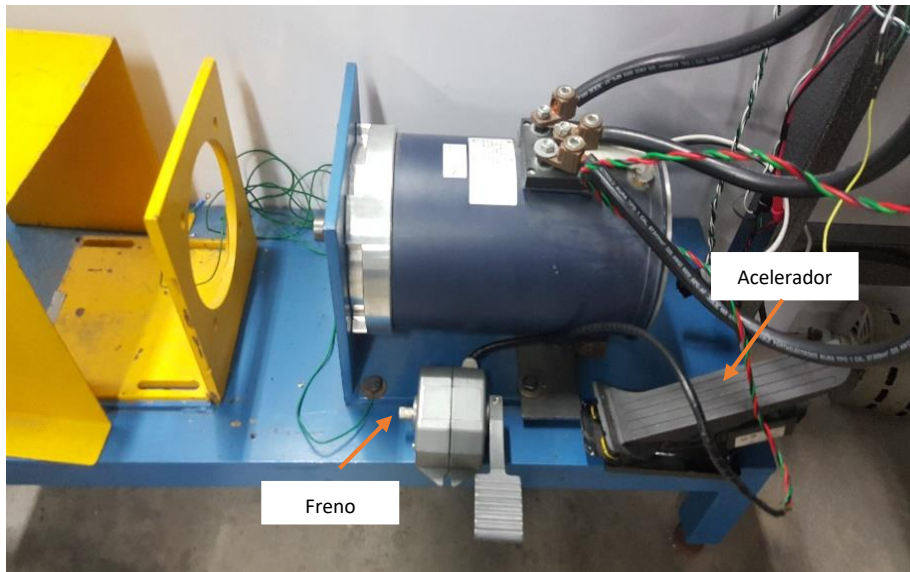


Fig.3.27. Sistema de arranque y frenado del motor de inducción.

En la Figura 3.28 se muestra la implementación completa del sistema de pruebas, donde se realizaron las conexiones de cada uno de las interfaces y bloques de control, para posteriormente validar su funcionamiento.

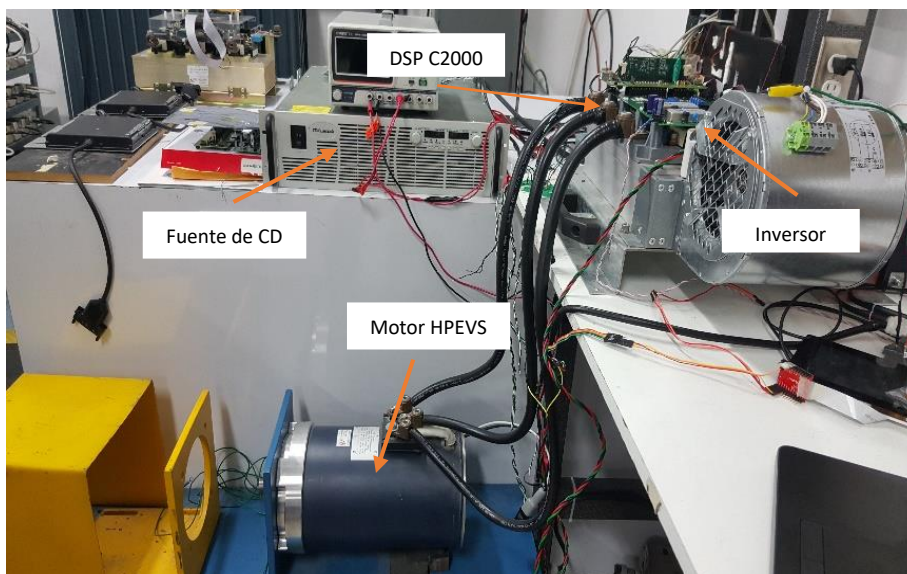


Fig.3.28. Banco de pruebas para el sistema de tracción eléctrica.

El problema del control V/f es que no es capaz de determinar eficientemente qué parte de la corriente entregada al motor se utiliza para inducir el flujo y que parte se transforma en par motor. La solución se encuentra en los llamados variadores de velocidad basados en la técnica de Control Vectorial, los cuales son capaces de regular de una forma mucho más eficiente ambos parámetros.

Capítulo 4

CONTROL VECTORIAL DEL MOTOR DE INDUCCIÓN

En este capítulo se presentan los principios generales de diseño, simulación e implementación en HIL del control vectorial para un motor de inducción tipo jaula de ardilla. En primer lugar se presenta el desarrollo de la estrategia de control empleada para lograr el seguimiento de una trayectoria de velocidad, esto en MATLAB/Simulink, por último se realiza su implementación en un DSP F28335, se emplea *SimCoder* de PSIM y Typhoon SCADA para el monitoreo de las señales.

4.1 Esquema general del control vectorial

La idea principal del control vectorial es lograr un control independiente de las corrientes de estator, desacoplándolas en dos fases dq ; el flujo que produce la corriente (I_d) y el par que produce la corriente (I_q). En la Figura 4.1 se muestra el diagrama general del sistema de control vectorial diseñado para controlar la velocidad del motor de inducción.

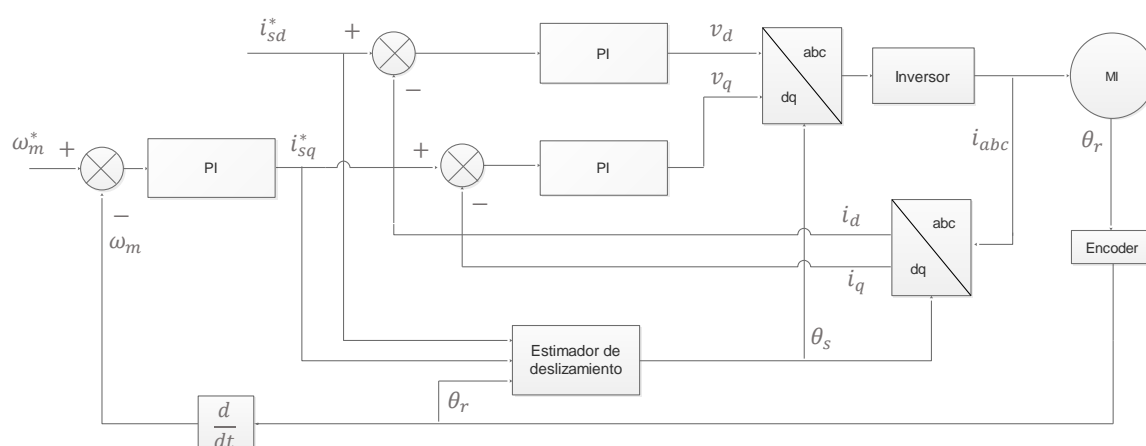


Figura 4.1 Esquema de control vectorial directo para el control de velocidad del MI.

Las corrientes trifásicas medidas $I_{(a,b,c)}$ se transforman al marco de referencia dq . El ángulo del rotor θ requerido para la transformación de coordenadas se determina a partir de la información obtenida por el encoder. El controlador de velocidad genera una señal de referencia de par a partir de la cual se obtienen las corrientes de referencia del controlador. El controlador de corrientes genera los voltajes de referencia dq para el inversor. Las corrientes dq son devueltas al marco de referencia abc .

4.1.1 Transformación de marco de referencia

La transformación entre marcos de referencia permite simplificar el modelo de la máquina y por ende nos facilita la implementación de una estrategia de control. Las variables abc se transforman en componentes dq que son ortogonales entre si y viceversa. Estas transformaciones son llamadas transformaciones de Clarke y Park.

Para llevar a cabo la demostración de la transformada de Clark y Park nos apoyamos del sistema trifásico y obtenemos como resultado una señal bifásica ortogonal de la Figura 4.2.

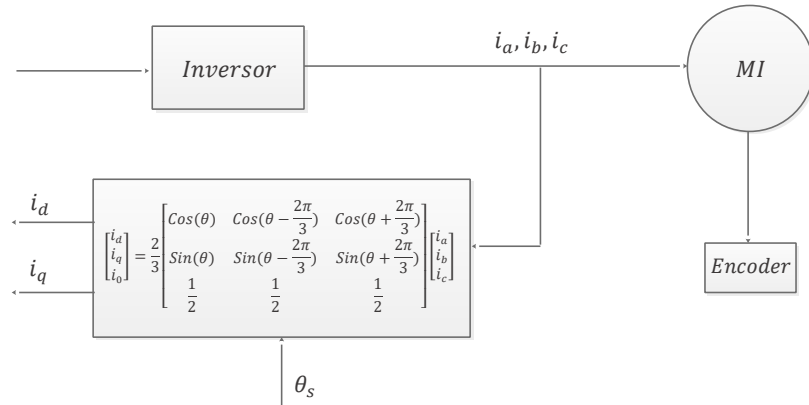


Figura 4.2 Diagrama a bloques de la transformación abc a dq.

En la Figura 4.3 se muestra la transformación inversa, del marco de referencia dq, al sistema trifásico.

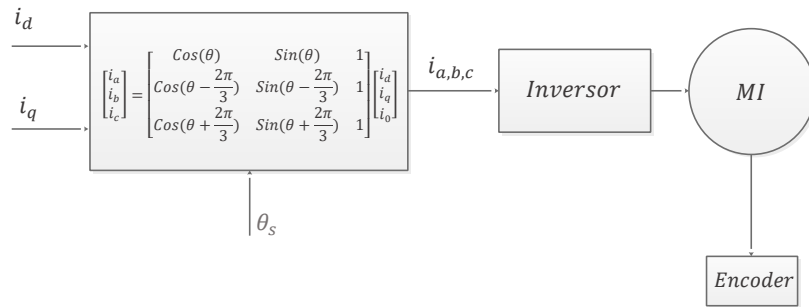


Figura 4.3 Diagrama a bloques de la transformación dq a abc.

4.1.2 Diseño del controlador de torque electromagnético

Para controlar el par electromagnético, se requiere el control de la corriente del eje q . Por lo tanto, dado un torque electromagnético T_e^* y un flujo deseado ψ_{rd}^* , la corriente i_{sq}^* se emplea como señal de referencia para el controlador PI de la corriente del eje q , de la ecuación 3.2.31 obtenemos que:

$$i_{sq}^*(t) = \frac{2}{3} \frac{L_r}{Z_h Z_p} \frac{T_e^*}{\psi_{rd}^*} \quad (4.1.1)$$

De la misma manera, para la corriente del eje d se tiene que.

$$i_{sd}^*(t) = \frac{1}{L_h} \psi_{rd}^*(t) + \frac{\tau_r}{L_h} \frac{d\psi_{rd}^*(t)}{dt} \quad (4.1.2)$$

Si la señal de referencia es constante, se emplea una relación para estado estable.

$$i_{sd}^*(t) = \frac{1}{L_h} \psi_{rd}^* \quad (4.1.3)$$

4.1.3 Diseño del controlador PI de corriente

Para el motor de inducción, el control de corrientes puede ser considerado como un lazo de control de torque (debido a que el torque se controla de forma indirecta aplicando un suministro de corriente). El primer paso para el diseño del sistema de control de corriente es regular las corrientes del estator i_{sd} y i_{sq} manipulando los voltajes del estator u_{sd} y u_{sq} en el marco de referencia dq , usando una ganancia proporcional K_c^d y un tiempo de integración constante τ_I^d , el controlador PI para los ejes dq tienen la forma:

$$u_{sd} = K_c^d(i_{sd}^* - i_{sd}(t)) + \frac{K_c^d}{\tau_I^d} \int_0^t (i_{sd}^*(\tau) - i_{sd}(\tau)) d\tau - \frac{K_r}{\tau_r} \psi_{rd}(t) - \tau'_\sigma r_\sigma \omega_s(t) i_{sq}(t)$$

$$u_{sq} = K_c^d(i_{sq}^* - i_{sq}(t)) + r_\sigma \tau'_\sigma \omega_s i_{sd}(t) + K_r \omega_m \psi_{rd}(t) \quad (4.1.4)$$

En la Figura 4.4 se muestra el controlador PI del eje d .

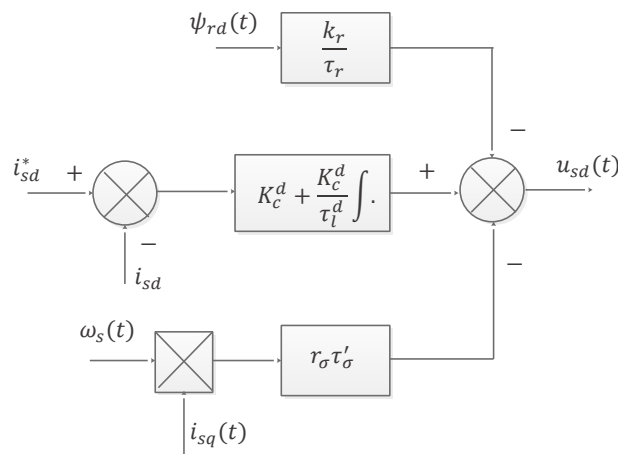


Figura 4.4 Diagrama de bloques del controlador PI para el eje d .

En la Figura 4.5 se muestra el controlador PI del eje q .

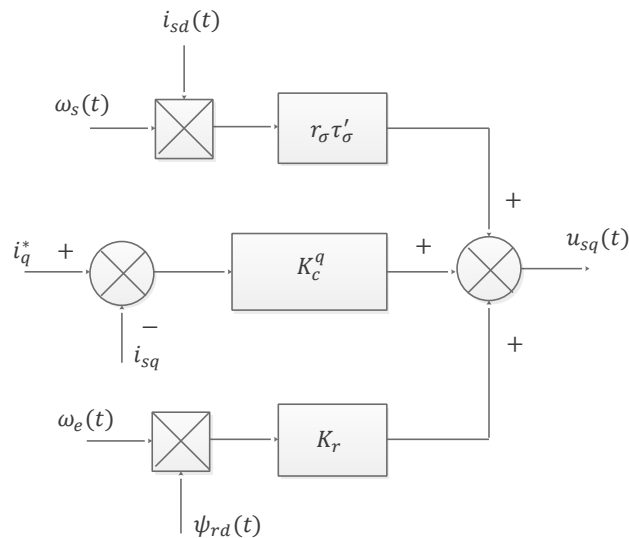


Figura 4.5 Diagrama de bloques del controlador PI para el eje q .

4.1.4 Diseño del controlador de velocidad

El modelo matemático en lazo cerrado para el control de ω_m se desprende de la ecuación (3.2.33). El control de la velocidad del bucle externo se logra al manipular el punto de ajuste de la señal de corriente $i_{sq}^*(t)$

$$i_{sq}^*(t) = K_c(\omega_m^*(t) - \omega_m(t)) + \frac{K_c}{\tau_I} \int_0^t (\omega_m^*(\tau) - \omega_m(\tau)) d\tau \quad (4.1.5)$$

Donde K_c es la ganancia proporcional y τ_I es la constante de tiempo de integración.

$$K_c = \frac{2\zeta\omega_n - a}{b}, \quad \tau_i = \frac{2\zeta\omega_n - a}{\omega_n^2}$$

En la Figura 4.6 se muestra el controlador PI del eje d .

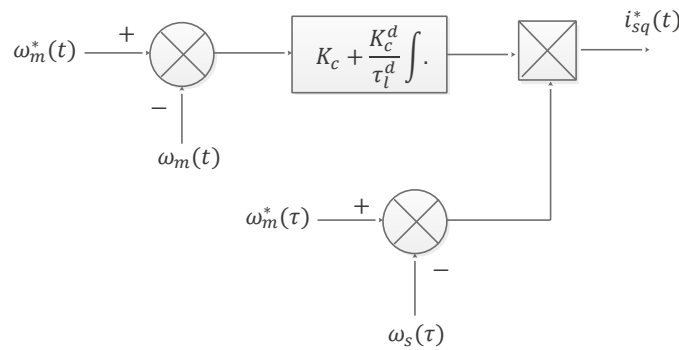


Figura 4.6 Diagrama de bloques del controlador PI para el eje d .

4.1.5 Diseño del estimador de deslizamiento

Para la estimación del deslizamiento θ_s se requiere la posición del rotor θ_r (obtenida de la medición del encoder). Su ecuación está definida de la siguiente manera:

$$\omega_s(t) = Z_p \omega_m(t) + \frac{i_{sq}(t)}{\tau_r i_{sd}(t)} \quad (4.1.6)$$

Asumiendo que el motor se inicia desde el reposo y sus velocidades iniciales ω_s , ω_m son iguales a cero, integrando ambos lados de la ecuación (4.1.6) da como resultado:

$$\theta_s(t) = Z_p \theta_r(t) + \frac{1}{\tau_r} \int_0^t \frac{i_{sq}(\tau)}{i_{sd}(\tau)} d\tau \quad (4.1.7)$$

Usando θ_r podemos calcular $\theta_e = Z_p \theta_r$, θ_e y las señales de referencia i_{sd}^* , i_{sq}^* nos permite estimar el ángulo θ_s , mediante (3.2.28).

$$\theta_s(t) \approx \theta_e(t) + \frac{1}{\tau_r} \int_0^t \frac{i_{sq}^*(\tau)}{i_{sd}^*(\tau)} d\tau \quad (4.1.8)$$

En la Figura 4.7 se muestra el estimador de deslizamiento.

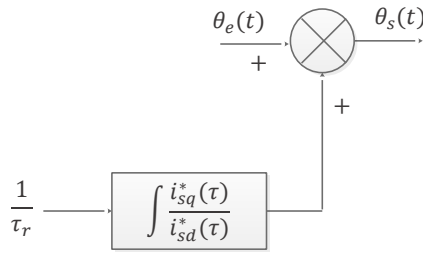


Figura 4.7 Diagrama de bloques del estimador de deslizamiento.

4.2 Simulación del control vectorial

Para simular el funcionamiento del control vectorial, es necesario diseñar el modelo del motor de inducción en MATLAB/Simulink en el marco de referencia dq (Figura 4.8), dicho modelo está basado en las ecuaciones y los parámetros físicos de la Tabla 1 y 2. El banco de pruebas utilizado se puede ver Figura 4.8, como variables de entrada tenemos: v_d, v_q, ω, T_L y como resultados tenemos i_d, i_q, T_e, F_{rd} , se supone un torque de carga variable y un perfil de velocidad deseado. Las gráficas de la Figura 4.9 a 4.12 muestran los resultados de la simulación del control vectorial en el software Simulink.

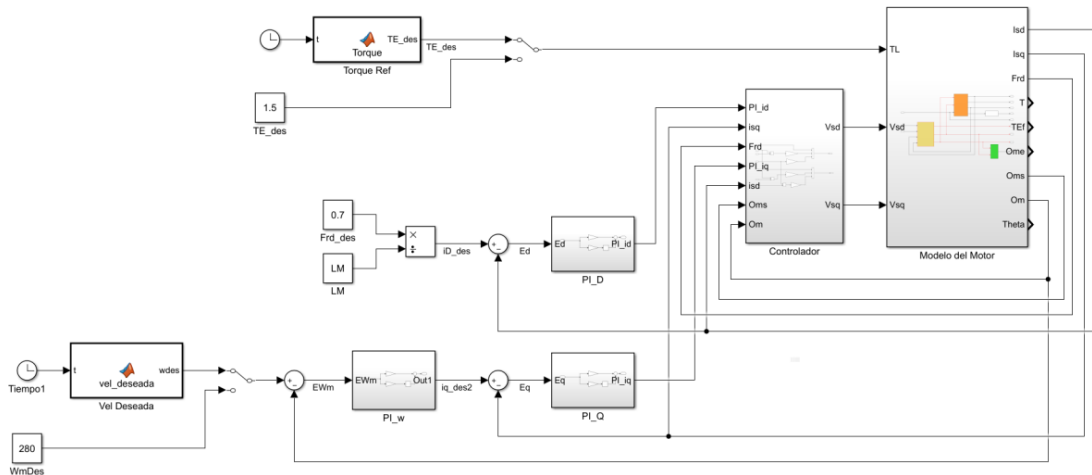


Figura 4.8 Esquema general del control vectorial en MATLAB/Simulink.

Los perfiles de: a) flujo y de b) par deseados los podemos ver en la Figura 4.9

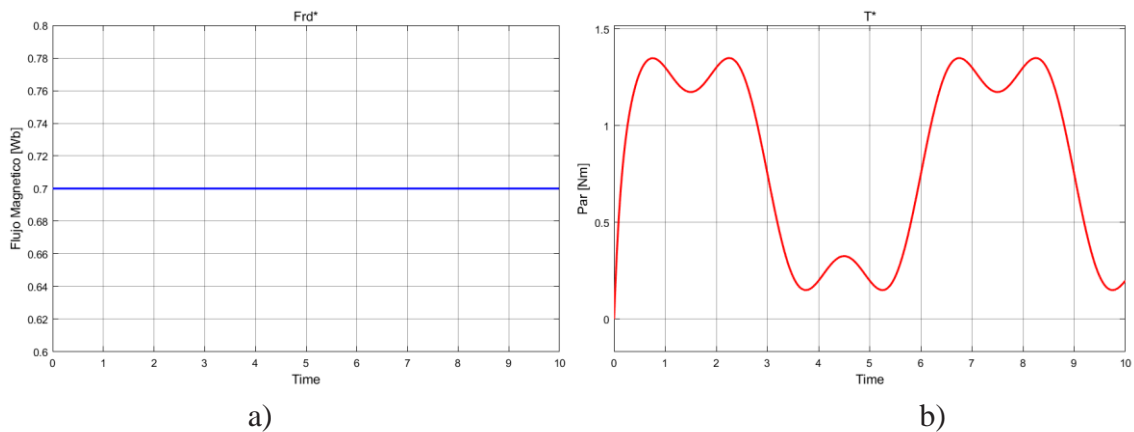


Figura 4.9 Resultados de la simulación del modelo: a) Perfil de flujo y b) Perfil de par.

El control vectorial exige un control independiente de flujo y de par, es por ello que en la Figura 4.10 se puede ver la gráfica de seguimiento de ambos parámetros: a) El flujo de referencia F_{rd}^* en comparación con el flujo medido del rotor F_{rd} , b) El par de referencia T^* contra el par de carga medido T .

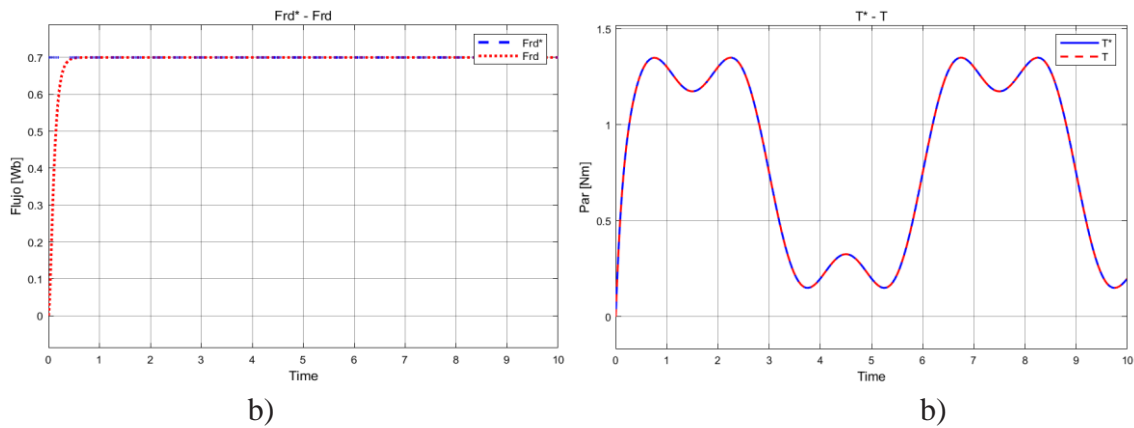


Figura 4.10 Resultados de la simulación del modelo: a) Flujo y b) Par de carga.

El comportamiento de las corrientes d-q encargadas de regular el flujo y el par se muestran en la Figura 4.11.

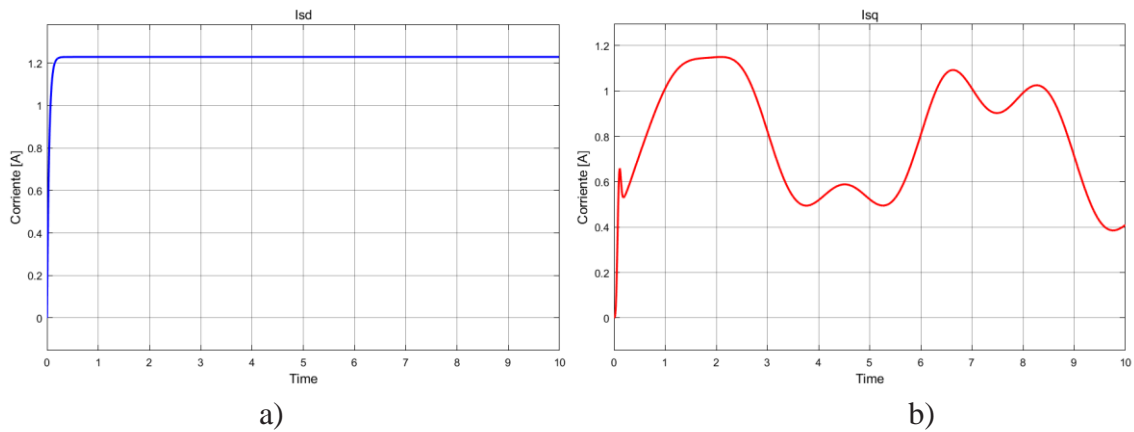


Figura 4.11 Resultados de la simulación del modelo: a) Corriente i_d , b) Corriente i_q .

La velocidad y posición alcanzada por el motor a los 10s se muestran en la Figura 4.12.

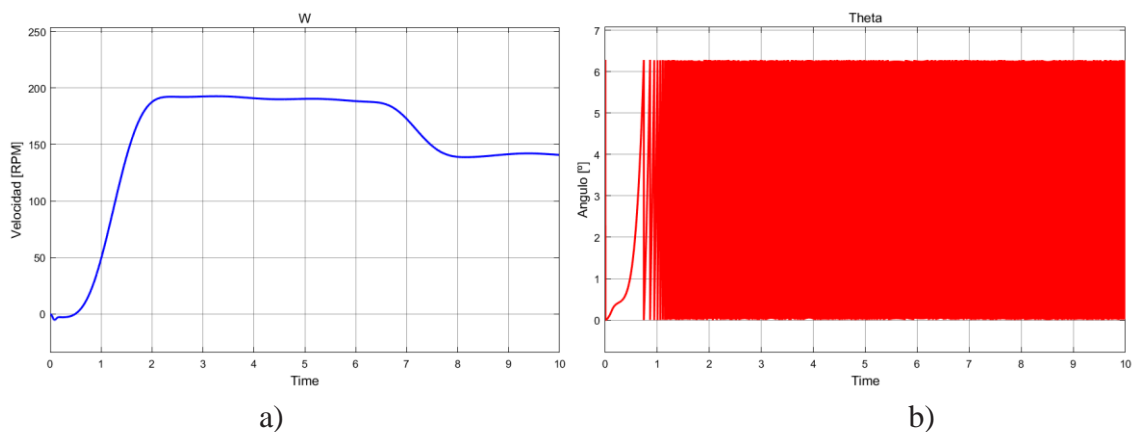


Figura 4.12 Resultados de la simulación del modelo: a) Velocidad, b) Posición.

En la Figura 4.13 y 4.14 se pueden ver las gráficas de los resultados de seguimiento de dos trayectorias de velocidad, descritas mediante curvas de Bézier:

En el primer caso el motor parte de reposo y aumenta su velocidad de 0 a 190 rpm, con lo cual podemos ver que el controlador vectorial es capaz de regular la velocidad del motor ω_m con el comportamiento deseado ω_m^* .

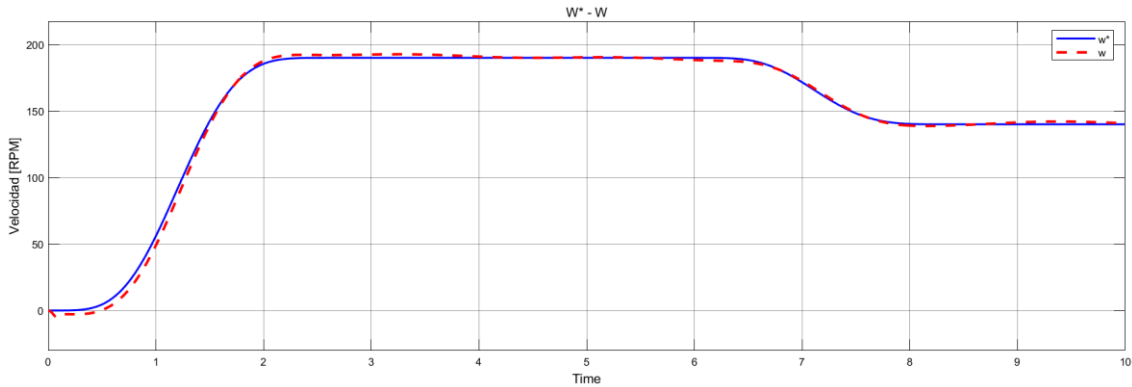


Figura 4.13 Velocidad de referencia - velocidad del rotor.

En el segundo caso se muestra el seguimiento de un segundo perfiles de velocidad más complejo, se cambia la velocidad de referencia de 0 a 100 rpm y luego de 100 a 220 rpm. De igual manera el controlador tiene un buen desempeño ante cambios continuos de la velocidad de referencia ω_m^* .

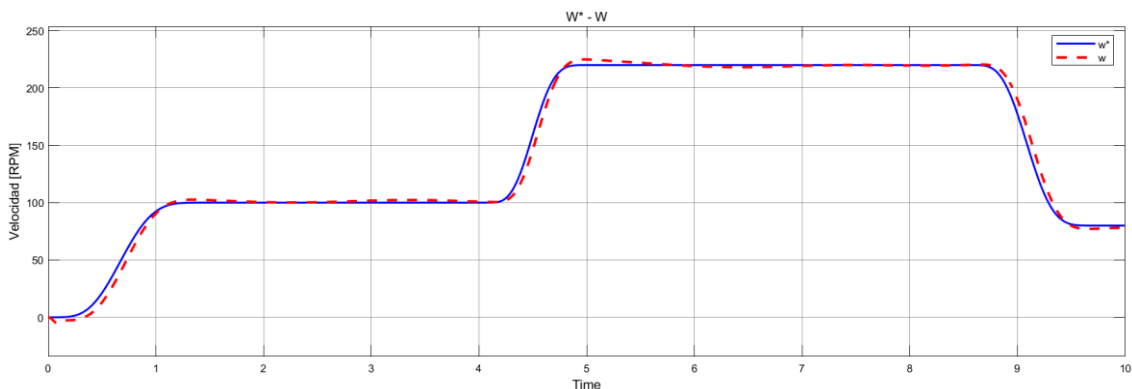


Figura 4.14 Velocidad de referencia - velocidad del rotor.

Con ayuda de las simulaciones en MATLAB/Simulink se verifica el buen desempeño de control vectorial al lograr llevar los parámetros de salida del motor en los valores deseados con un margen de error bastante bajo. Resulta evidente que esta estrategia de control demostró manejar adecuadamente la velocidad del motor, aun en presencia de perturbaciones en su operación.

4.3 Emulación del control vectorial en Typhoon HIL

Una vez concluido el análisis teórico del control vectorial, se requiere la verificación experimental del algoritmo, para ello se hace uso de un sistema de Hardware in the Loop Typhoon HIL 402. Las soluciones HIL nos permiten un enfoque de validación más rápida, flexible y confiable. Además de ofrecernos un completo entorno para la emulación de escenarios y fallas que son difíciles de replicar en la realidad. Múltiples sistemas comerciales, han sido diseñados y probados en HIL. Los dispositivos HIL permiten probar esquemas avanzados de electrónica de potencia y leyes de control sin la presencia de plantas reales [42].

El Typhoon HIL 402 cuenta con cuatro núcleos, tiene una resolución de 1MHz (1us), 16 entradas analógicas, 16 salidas analógicas y 32 salidas y entradas digitales, adicionalmente cuenta con una interfaz DSC, construida para la familia C2000 de Texas Instruments, esta interfaz proporciona compatibilidad pin a pin entre el emulador Typhoon HIL y las tarjetas C2000. En la Figura 4.15 se puede apreciar el HIL 402 con su respectiva interfaz para interconexión con el DSP.

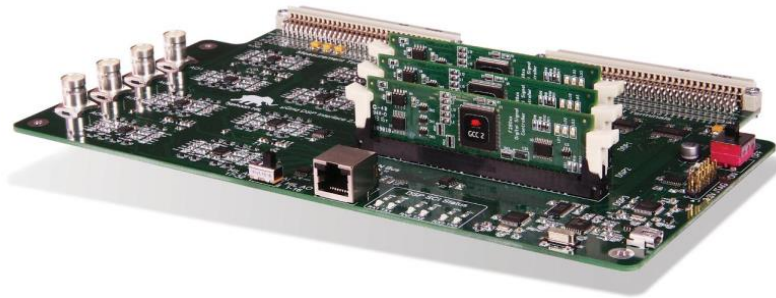


Figura 4.15 Plataforma de Hardware in the Loop: Typhoon HIL 402.

Typhoon HIL editor

El Typhoon HIL cuenta con un editor que se utiliza para el diseño de esquemáticos de sistemas de potencia (Figura 4.16)

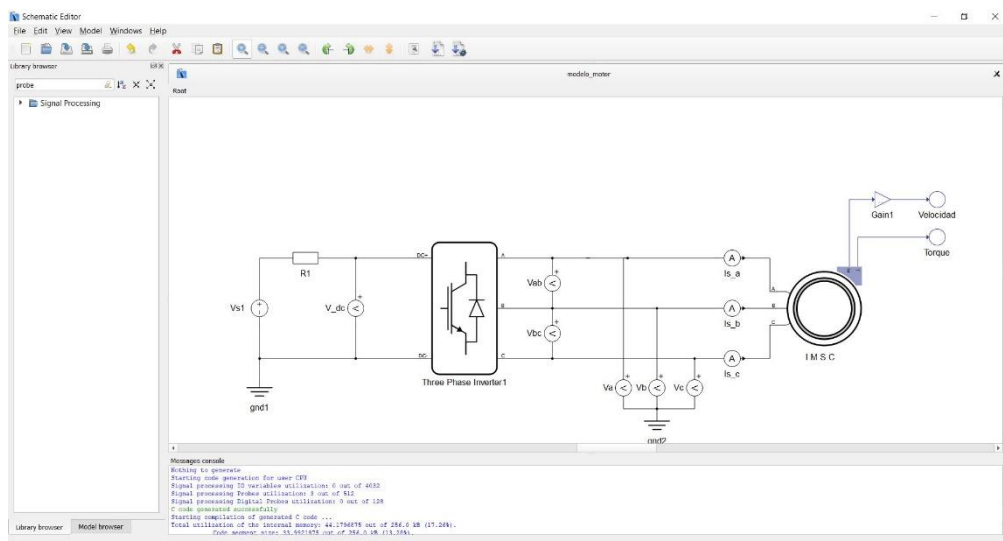


Figura 4.16 Editor de esquemáticos de Typhoon HIL 402.

Typhoon HIL SCADA

El Typhoon HIL cuenta con un sistema SCADA (Figura 4.17) que permite la interacción y despliegue de información en tiempo real.

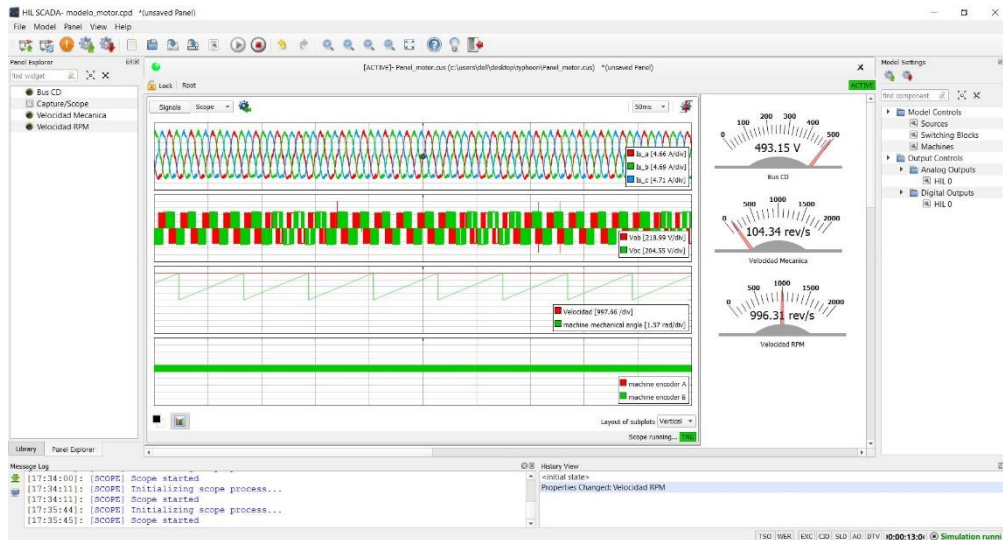


Figura 4.17 Editor SCADA de Typhoon HIL 402.

DSP F28335

Para la programación de los algoritmos de control se empleara un DSP F28335, que es compatible con la suite *SimCoder* de PSIM, esta suite permite generar programas de todos niveles de complejidad mediante el empleo de bloques gráficos que facilitan y mejoran los tiempos de implementación, debido a que no se requieren conocimientos avanzados para la configuración de la tarjeta F28335, lo que permite enfocarnos en su totalidad en la pura generación de algoritmos, ofrece la posibilidad de utilizar código C, y adicionalmente permite simular los algoritmos y la parte eléctrica por igual, por lo que la detección de errores relacionados con el algoritmo se vuelve una tarea más sencilla.

4.3.1 Typhoon HIL y PSIM

En esta sección se muestra los resultados obtenidos de la emulación del control vectorial aplicado a la regulación de la velocidad del motor de inducción. La simulación del algoritmo de control se realizó en la plataforma de PSIM, mientras que el modelo físico del inversor y el motor de inducción se realizaron en Typhoon HIL 402, el cual es capaz de emular en tiempo real el comportamiento del motor de inducción.

El sistema de control diseñado en PSIM será compilado y descargado al DSP mediante Code Composer Studio (CCS), por medio de la interfaz DSC, el controlador TMS320F28335 de Texas Instruments se comunicara bidireccionalmente con el HIL 402 el cual controlara el flujo del sistema completo. En la Figura 4.18 se muestra la integración que tendrán ambas plataformas.

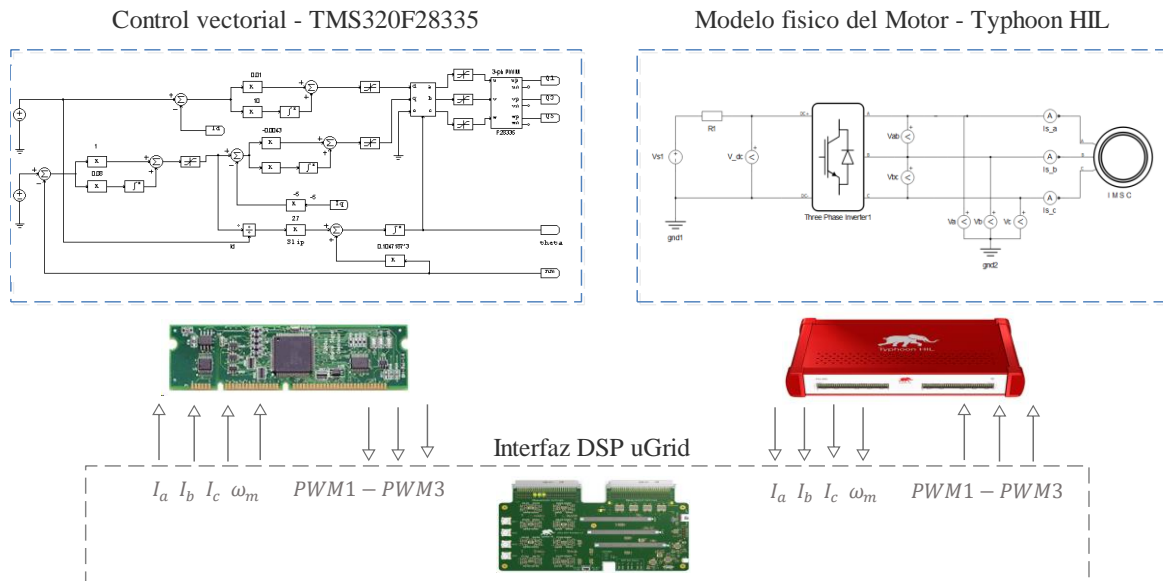


Figura 4.18 Esquema de emulación del control vectorial.

4.3.2 Esquema de control en PSIM

En la Figura 4.19 se muestra el esquemático construido a partir de elementos que pueden ser sintetizados de manera automática a código en C (SimCoder) para posteriormente ser descargados al DSP C2000 (TMS320F28335). En el esquema hay dos partes principales que no se deben confundir: la etapa de potencia que solo puede ser simulada (líneas rojas) y el circuito de control que puede ser implementado a código en C (líneas verdes).

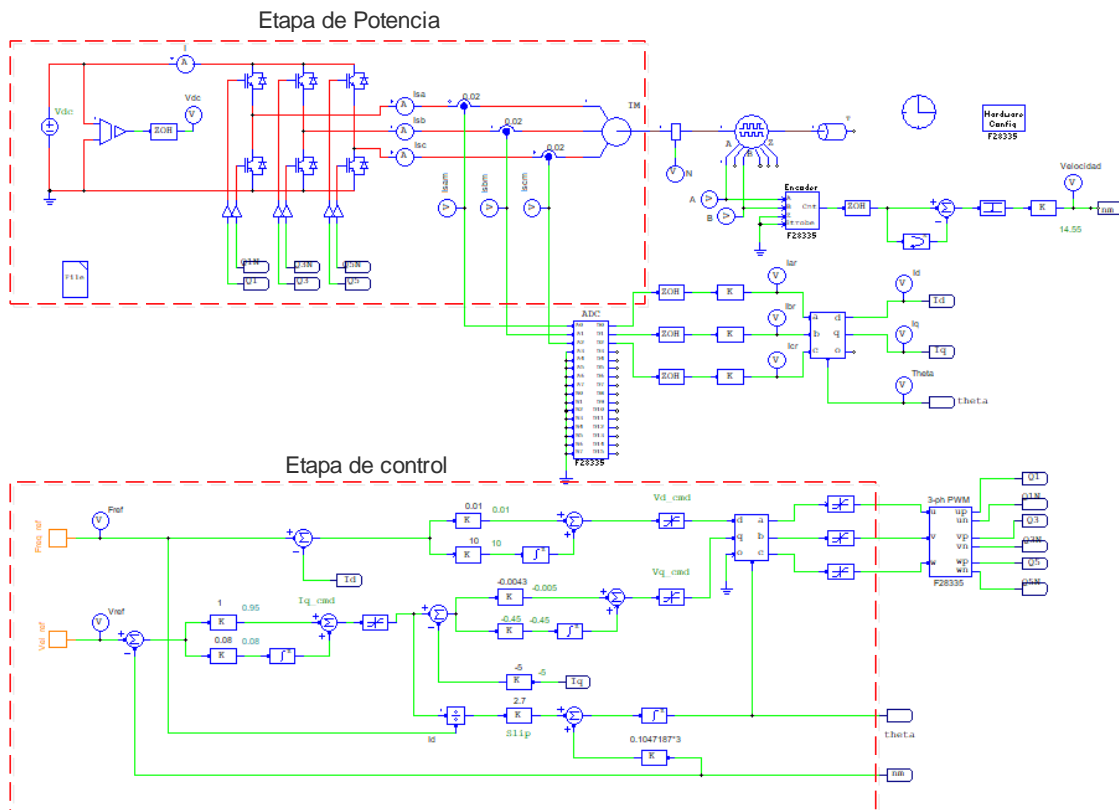


Figura 4.19 Esquema del control vectorial en PSIM.

Configuración del DSP

Como primer paso se agregan componentes para el control de los parámetros de simulación, la definición del dispositivo a emplear y la configuración de los periféricos, ver Figura 4.20. De izquierda a derecha se tiene el bloque *Simulation control*, en el cual se selecciona la tarjeta a emplear y la forma en la que se ejecutara el código, para este caso fue *TI F28335* y *RAM Debug*, en el bloque *DSP configuration* se configura la frecuencia de operación del DSP para este caso a 10 MHz. En el último bloque se configura la forma de operación de las entradas y salidas digitales.

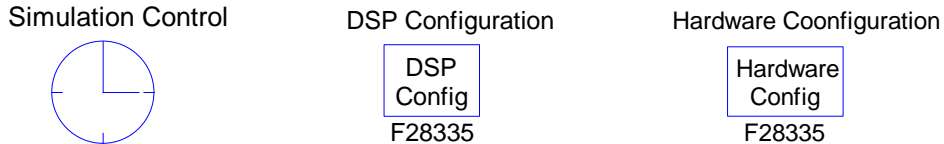


Figura 4.20. Bloque para la configuración del DSP y Simcoder.

Adquisición de señales

El control de velocidad del motor depende de las corrientes del estator I_a, I_b, I_c , para realizar su lectura se emplea un módulo *ADC A/D Converter* (Figura 4.21) el cual convierte las señales analógicas de AC, en señales digitales que pueden ser procesadas por el DSP. Se configuro para capturar señales en modo *Start-stop 8-channel*, con esto, cuando el ADC recibe una señal de disparo iniciara con el muestreo de los canales del 0 al 7, el ADC solo puede leer señales eléctricas que se encuentran dentro del rango de voltaje de 0 a +3 VCD. A la salida del ADC tendremos una señal de la forma ($V_o = k * V_i$).

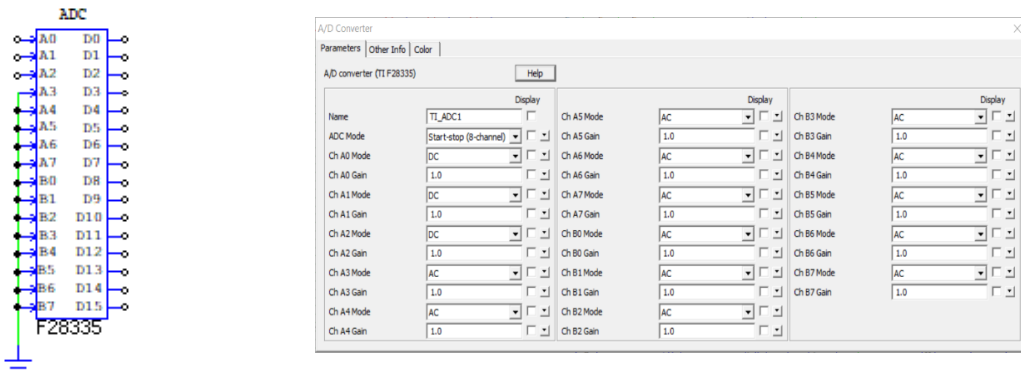


Figura 4.21. Bloque ADC en PSIM.

Generación de PWM

Para generar las señales de activación de los IGBT's se emplea el bloque *3-ph PWM* el cual se alimenta con una señal sinusoidal moduladora y una señal triangular interna, a su salida tendremos seis señales PWM de 10KHz con un ciclo de trabajo de 50%; los pines de salida seleccionados corresponden con los EPWM-1A, 2A y 3A y sus respectivos complementos EPWM-1B, 2B y 3B que corresponde a los transistores de la parte alta y baja respectivamente. En la Figura 4.22 podemos apreciar el bloque PWM de PSIM y su respectivo dialogo de configuración.

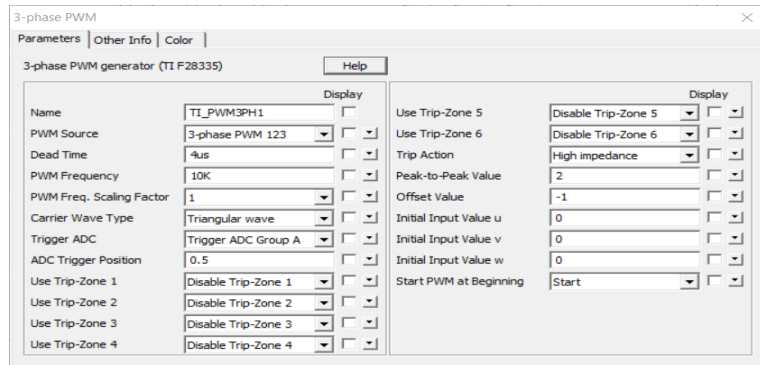
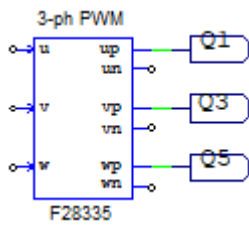


Figura 4.22. Bloque 3-phase PWM de PSIM.

Encoder de cuadratura

Para cerrar el lazo del sistema de control se requiere conocer la velocidad del motor a cada instante de tiempo, para ello se emplea el módulo *Incremental Encoder* y el módulo *Encoder* de 1024 posiciones, en conjunto proporcionan un par de señales en cuadratura A y B (desfasadas 90° una de otra) que serán convertidas a pulsos entendibles por el *Encoder*, la salida del encoder será el valor del contador, ver Figura 4.23.

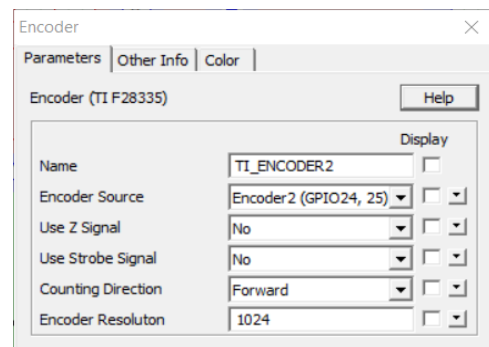
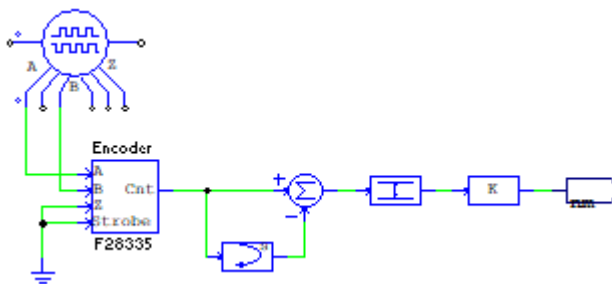


Figura 4.23. Bloque encoder de PSIM.

Transformación abc-dq0

El bloque *abc to dq transformation* realiza la transformación de las coordenadas abc a las coordenadas dq0. Su proceso inverso se realiza mediante el bloque *dq to abc transformation*. Ambos bloques se emplea tanto para el circuito de alimentación del inversor como en el de control.

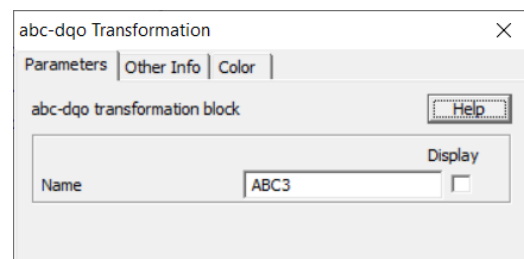
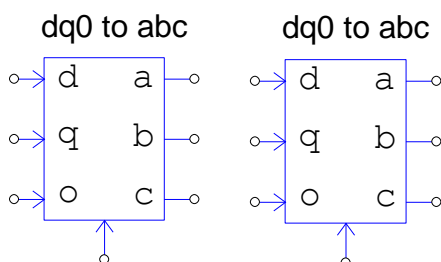


Figura 4.24. Bloque de transformación adb-dq0.

En la Figura 4.25 se aprecian las señales de corriente: a) Señales analógicas leídas por los sensores de corriente, b) Señal digital normalizada adquirida por el ADC.

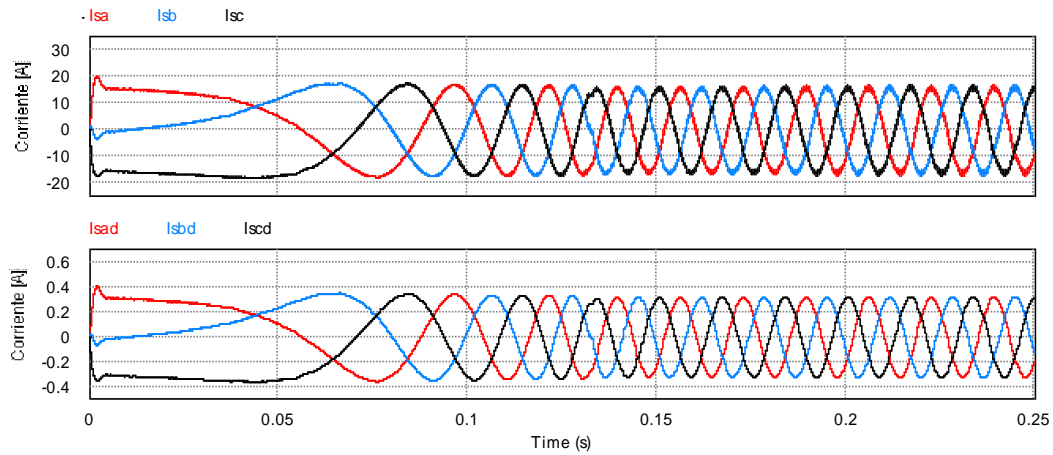


Figura 4.25. (I_{sa}) Corriente del estator, (I_{sad}) Corriente leída por el ADC.

La Figura 4.26 muestra las señales generadas por el encoder: a) Señales A y B en cuadratura, b) Señal de posición (Contador reseteable cada 1024 muestras).

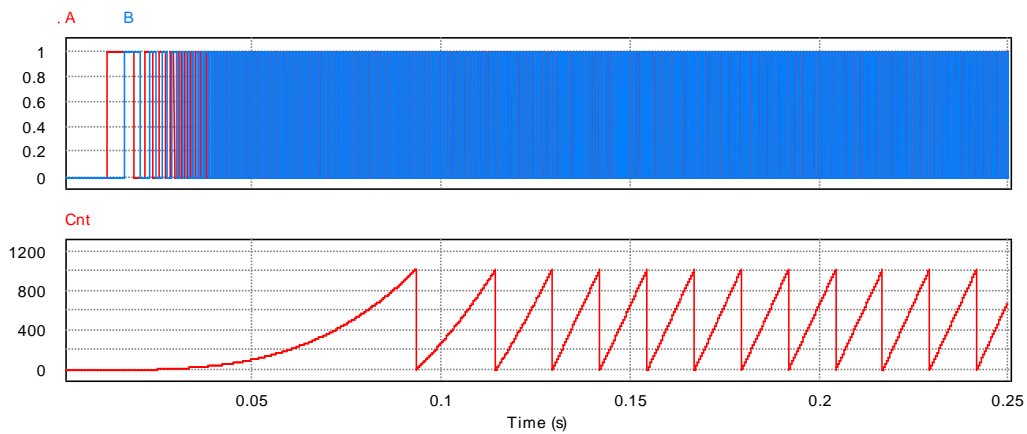


Figura 4.26. (A – B) Señales en cuadratura, (Cnt) Posición del rotor.

La Figura 4.27 muestra las señales de corriente digitales normalizadas que servirán para la generación de las señales de PWM de activación.

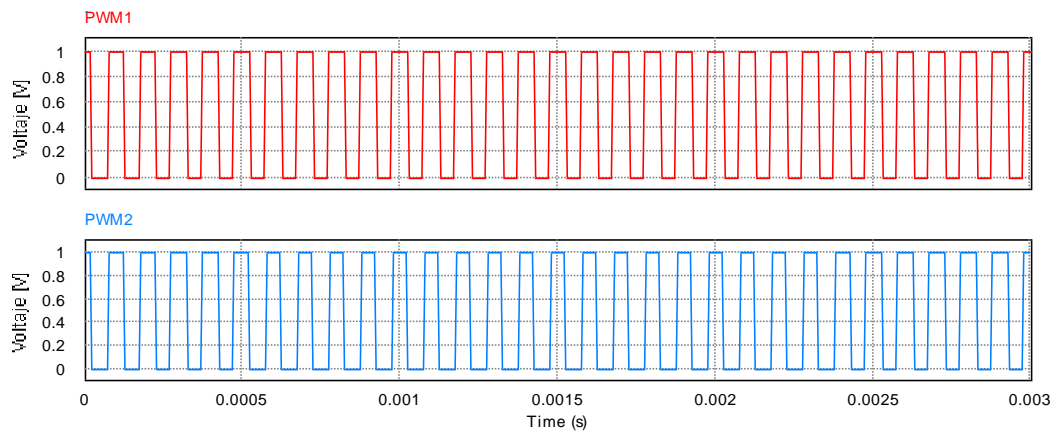


Figura 4.27. Señales PWM de activación para los IGBT's.

Se muestran los resultados obtenidos de la simulación del control vectorial en PSIM bajo las siguientes condiciones: se propone una señal de velocidad de referencia $w_{ref} = 1200 \text{ rpm}$ y un flujo magnético de $\psi_{ref} = 17$.

Primeramente se muestran las señales resultantes de la transformación de Clarke y Park; empleando el bloque $abc - dq0$ de PSIM obtenemos una señal bifásica de DC a partir de una señal trifásica de AC, tal como se muestra en la Figura 4.28.

Se observa que el sistema de control vectorial logra ubicar los valores de salida del motor a los valores de referencia deseados, es decir, ante variaciones de carga en el eje del motor, la velocidad y el torque permanece constante.

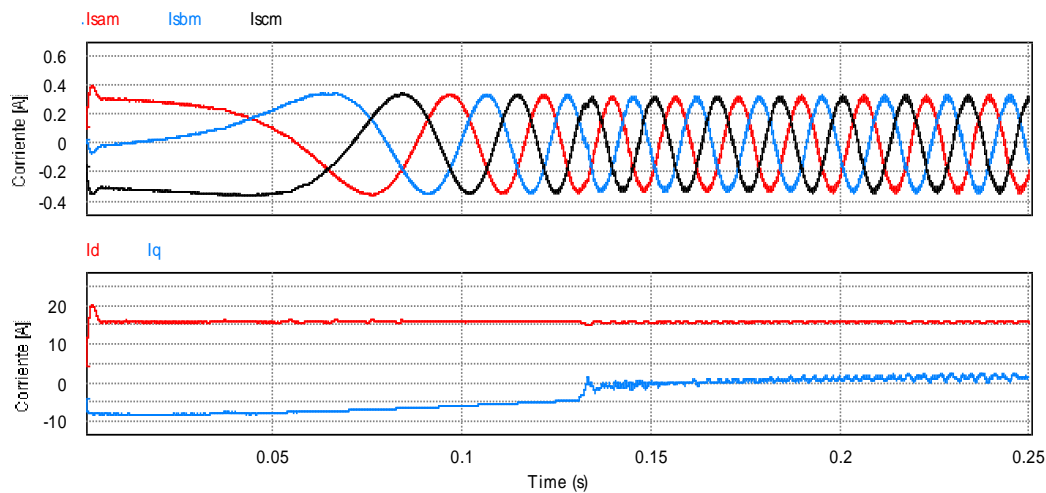


Figura 4.28 Transformación de Clark y Park (de dq a abc en PSIM).

De igual manera, en la Figura 4.29 se muestra el resultado de la transformación inversa de Clarke y Park, del marco de referencia ortogonal $dq0$, al sistema trifásico abc .

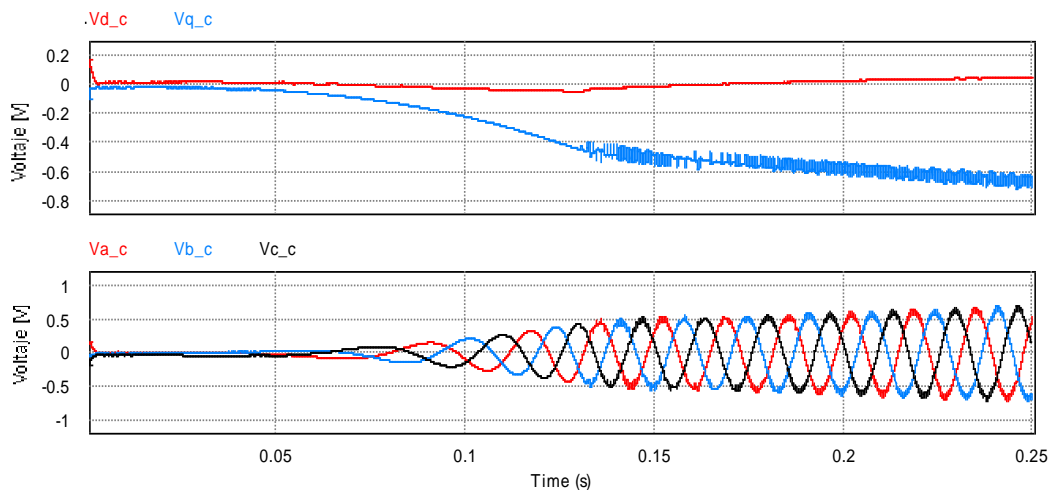


Figura 4.29 Transformación inversa de Clarke y Park (de abc a dq en PSIM).

En la Figura 4.30 se muestra la señal trifásica de corriente del estator generadas por el motor al cerrar el lazo de control.

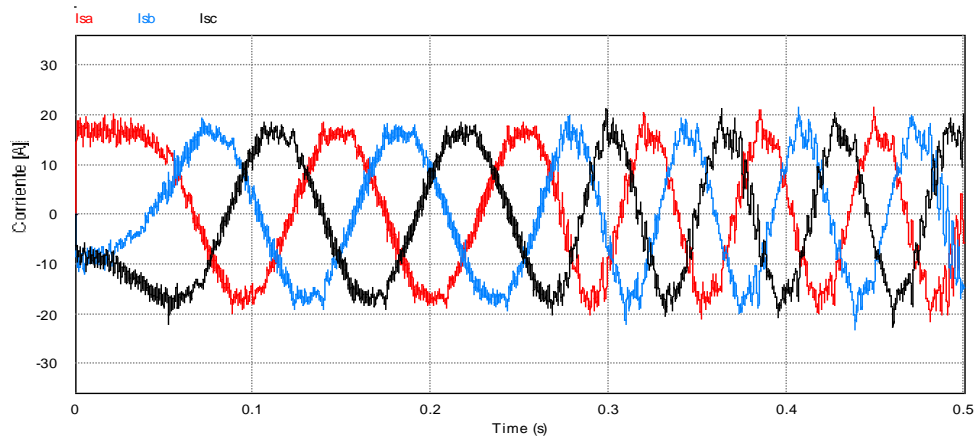


Figura 4.30 Señal de corriente trifásica.

El control vectorial exige la regulación independiente de flujo y de par, en la Figura 4.31 se muestra la comparación de la señal de flujo magnético generado por el motor la cual llega de forma satisfactoria a un valor de referencia propuesta ($\psi_{ref} = 17$).

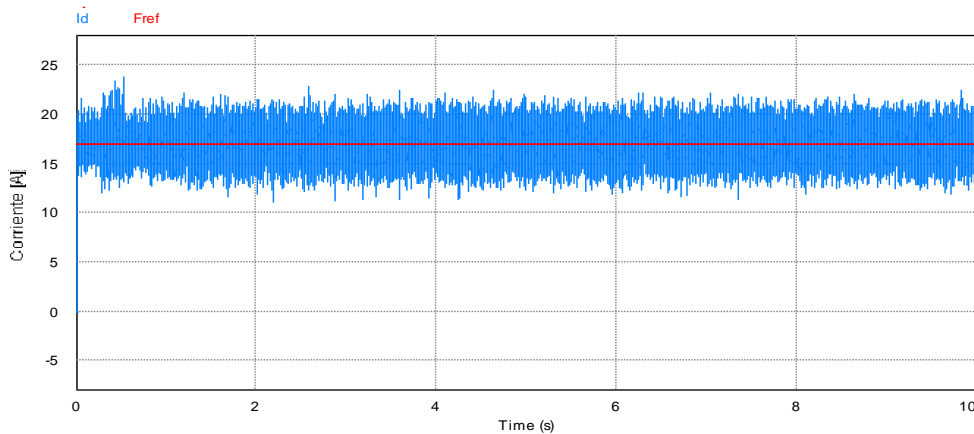


Figura 4.31 Perfil de flujo deseado, frente a la respuesta de flujo obtenido.

Finalmente, en la Figura 4.32 se muestra la gráfica de seguimiento de una trayectoria de velocidad descrita mediante una curva de Bezier: el motor parte de reposo y aumenta su velocidad de 0 a 190 rpm y después pasa de 190 a 300 rpm. Mediante estas variaciones en la referencia podemos ver que el controlador vectorial es capaz de igualar la velocidad del motor ω_m con un perfil deseado ω_m^* .

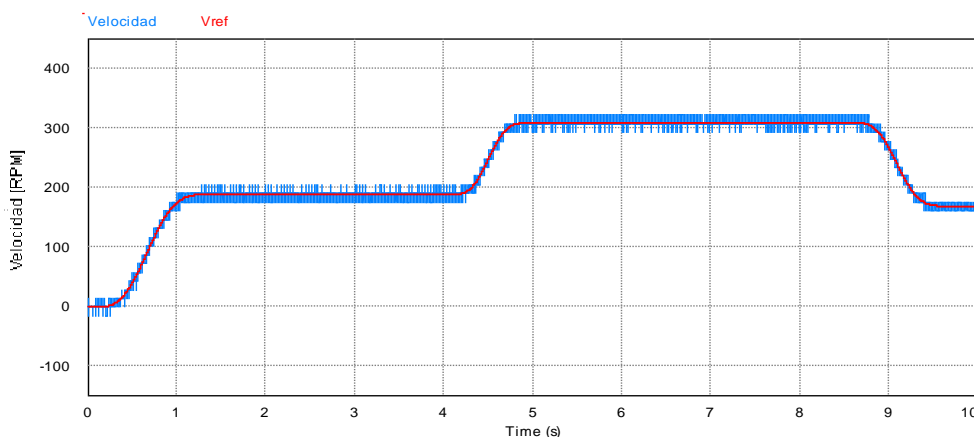


Figura 4.32 Perfil de velocidad deseado, frente a la respuesta de velocidad obtenida.

4.3.3 Esquema físico en Typhoon HIL

En la Figura 4.33 se muestra el esquemático construido a partir de elementos contenidos en las Toolbox de Typhoon; consta de una fuente de DC, un inversor trifásico y el modelo físico del motor de inducción jaula de ardilla.

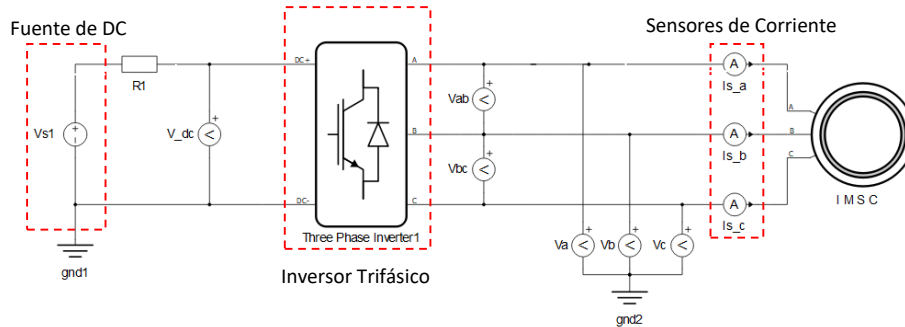


Figura 4.33 Esquema general del modelo físico a emular.

Para el inversor, se deben establecer los puertos digitales de entrada (DI32 a DI30) donde serán inyectadas las señales PWM provenientes del DSP (EPWM-1A – EPWM-3A), tal como se muestra en la Figura 4.34.

Typhoon HIL Digital Input	DSP Digital Output	DSP1 and DSP3 Digital Output
DI32	GPIO-00/EPWM-1A	GPIO-00/EPWM-1A
DI31	GPIO-02/EPWM-2A	GPIO-02/EPWM-2A
DI30	GPIO-04/EPWM-3A	GPIO-04/EPWM-3A
DI29	GPIO-06/EPWM-4A	GPIO-06/EPWM-4A
DI28	GPIO-08/EPWM-5A	GPIO-08/EPWM-5A
DI27	GPIO-10/EPWM-6A	GPIO-10/EPWM-6A
DI26	GPIO-01/EPWM-1B	GPIO-01/EPWM-1B
DI25	GPIO-03/EPWM-2B	GPIO-03/EPWM-2B
DI24	GPIO-05/EPWM-3B	GPIO-05/EPWM-3B
DI23	GPIO-07/EPWM-4B	GPIO-07/EPWM-4B
DI22	GPIO-09/EPWM-5B	GPIO-09/EPWM-5B
DI21	GPIO-11/EPWM-6B	GPIO-11/EPWM-6B

Figura 4.34 Mapeo de puertos digitales entre el DSP y el Typhoon 402.

Para la lectura de las corrientes del estator (I_a, I_b, I_c) se emplean tres puertos analógicos del DSP (A0, A1 y A2). Para la lectura de la velocidad proveniente del Typhoon se emplea el puerto digital (EQEPA – 2 y EQEPB – 2), tal como se muestra en la Figura 4.35.

Typhoon HIL Digital Output	DSP Digital Input	Typhoon HIL 602 Analog Output	DSP1 Analog Input
DO32	GPIO-24 / EQEPA-2	AO1	ADCIN – A0
DO31	GPIO-25 / EQEPB-2	AO2	ADCIN – A1
DO30	GPIO-26 / EQEPI-2	AO3	ADCIN – A2
DO29	GPIO-14	AO4	ADCIN – A3
DO28	GPIO-15	AO5	ADCIN – A4
DO27	GPIO-16	AO6	ADCIN – A5
DO26	GPIO-17	AO7	ADCIN – A6
DO25	GPIO-19	AO8	ADCIN – A7
DO24	GPIO-20	AO9	ADCIN – B0
DO23	GPIO-21	AO10	ADCIN – B1
DO22	GPIO-22	AO11	ADCIN – B2
DO21	GPIO-23	AO12	ADCIN – B3
		AO13	ADCIN – B4
		AO14	ADCIN – B5
		AO15	ADCIN – B6
		AO16	ADCIN – B7

Figura 4.35 Mapeo de puertos analógicos y digitales entre el DSP y el Typhoon.

4.3.4 Resultados experimentales

En la Figura 4.36, se puede ver el entorno completo de pruebas: HIL Typhoon 402, interfaz DSC, DSP C200 (TSM320F28335) y estación de trabajo.

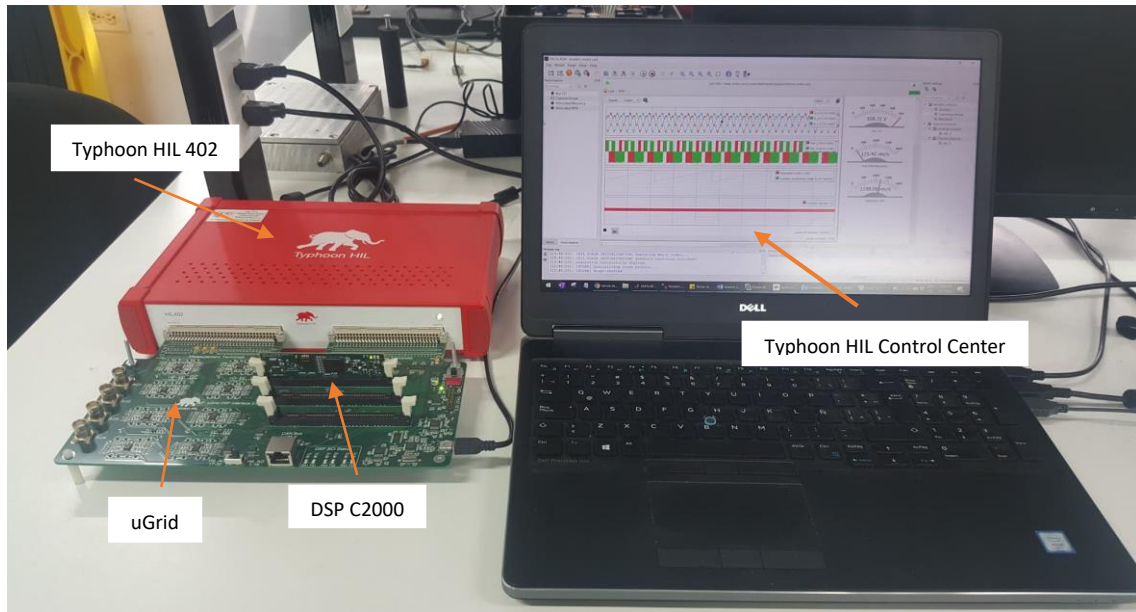


Figura 4.36 Entorno de pruebas: HIL Typhoon, DSP TI F28335.

El entorno de Typhoon cuenta con un osciloscopio virtual para el monitoreo en tiempo real de las señales resultantes, en las Figura 4.37 a 4.40 podemos apreciar los resultados de simulación en tiempo real del HIL en comparación con los resultados de simulación de PSIM.

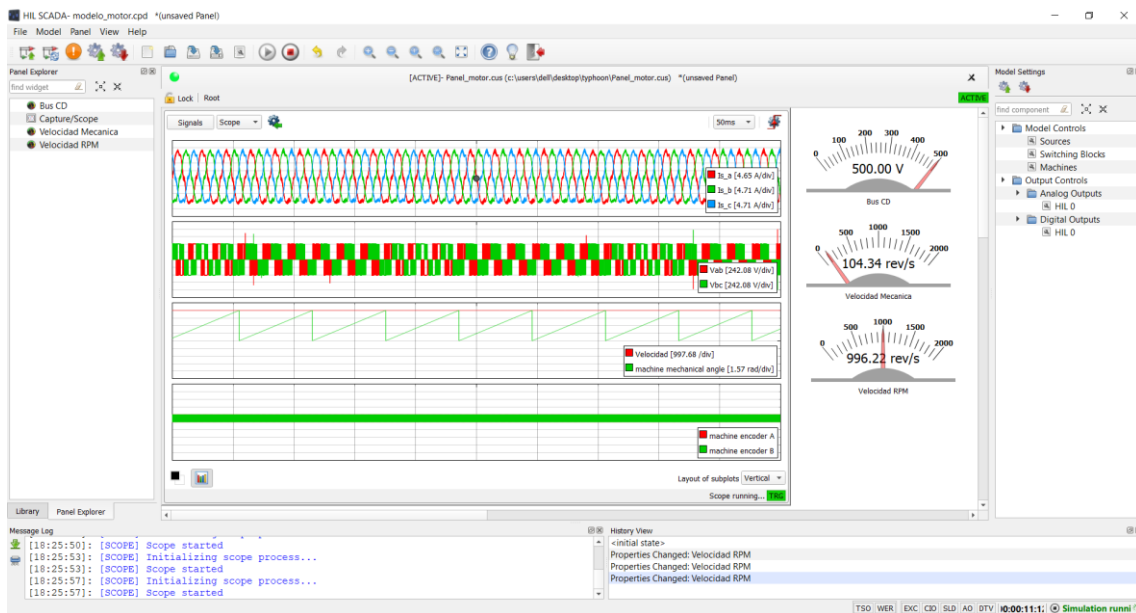


Figura 4.37 Voltajes y corrientes de fase en Typhoon.

En la Figura 4.38 se muestra: a) Las corrientes de estator I_a, I_b, I_c , b) y los voltajes de fase V_{ab}, V_{bc} , con un desfase de 120° entre ellos.

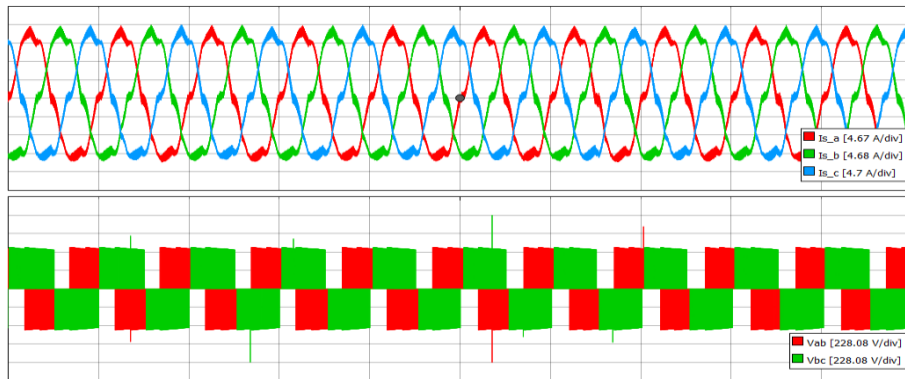


Figura 4.38 Voltajes y corrientes de fase en Typhoon.

En la Figura 4.39 se muestra: a) La velocidad mecánica alcanzada por el motor corresponde al perfil de velocidad de referencia propuesta ($V_{ref} = 1000 \text{ RPM}$), b) Señales A y B en cuadratura del encoder.

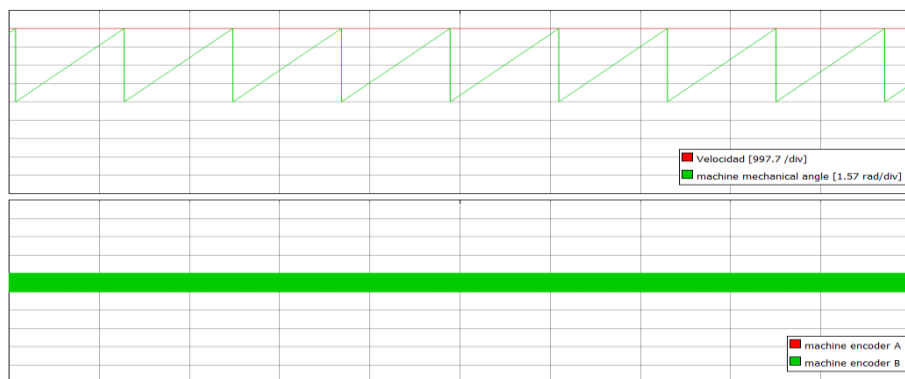


Figura 4.39 a) Velocidad y posición angular del rotor, b) Señales del encoder.

En la Figura 4.40 se muestra el monitor de expresiones propio del entorno de Code Composer Studio con los valores en tiempo real de la velocidad medida y la de referencia, el valor de la corriente I_d y su respectivo valor de referencia.

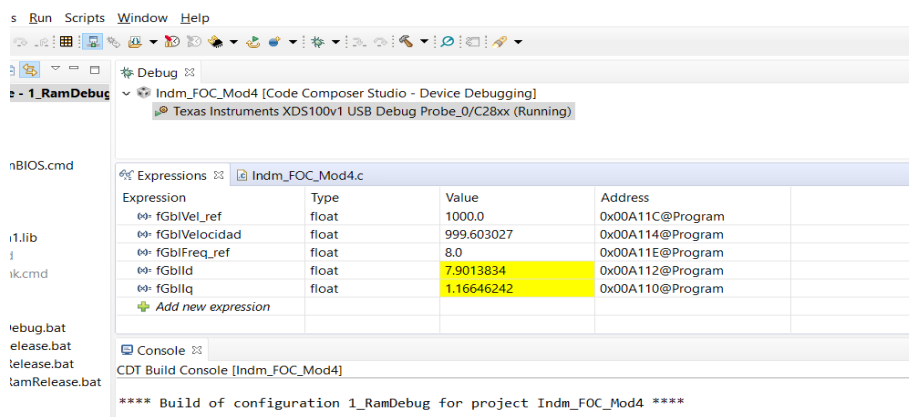


Figura 4.40 Monitoreo de parámetros en tiempo real con Code Composer Studio.

Finalmente, en la Figura 4.41 se muestra la gráfica de seguimiento de una referencia de velocidad obtenida del HIL, el motor parte de reposo y aumenta su velocidad de 0 a 1200 rpm. El sistema presenta un sobretiro de 0.25, un tiempo de elevación de 2s y un tiempo de asentamiento de 9s. Con estos resultados podemos ver que el controlador vectorial es capaz de igualar la velocidad del motor ω_m con un perfil deseado $\omega_m^* = 1200 \text{ rpm}$ en un lapso de 9s.

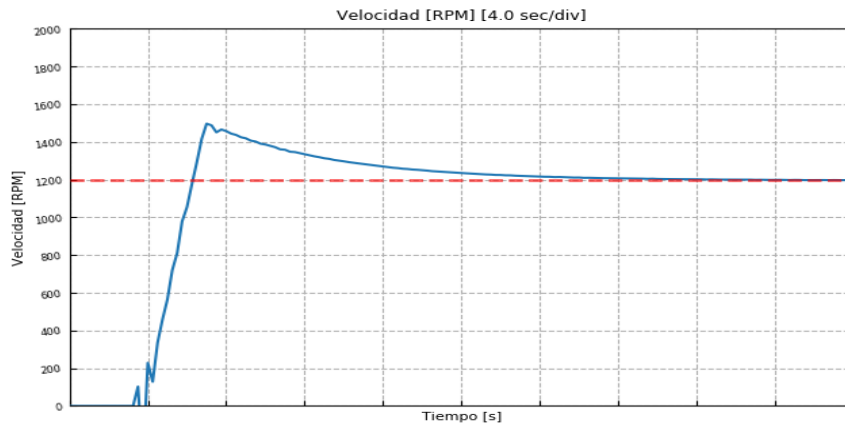


Figura 4.41 Perfil de velocidad deseado, frente a la respuesta de velocidad obtenida.

Capítulo 5

RED NEURONAL B-SPLINE PARA EL CONTROL DE UN MOTOR DE INDUCCIÓN

Las redes neuronales de entrenamiento instantáneo poseen características que las comparan favorablemente con otras redes como por ejemplo las de retropropagación y las funciones de base radial. Una red neuronal de entrenamiento instantáneo (B-spline) por definición debe convertir patrones de entrada a sus correspondientes pesos evitando un gran número de cálculos.

Por otro lado, las redes neuronales BP y RBF requieren un entrenamiento con mayor número de cálculos. Algunas veces las redes BP no convergen o requiere demasiado tiempo su entrenamiento para que sea útil en aplicaciones de tiempo real. Por lo tanto, las redes neuronales B-spline ofrecen una alternativa atractiva respecto a otros modelos neuronales, por lo cual se utilizan para aplicaciones en tiempo real [43].

En este capítulo se desarrolla una estrategia de control que emplea redes neuronales de entrenamiento instantáneo B-spline aplicadas a la regulación de la velocidad de un motor de inducción jaula de ardilla, se pretende un sistema robusto y adaptativo que demuestre su efectividad frente al control PI convencional. La estrategia de control propuesta es fácil de implementar y el costo computacional es menor que los algoritmos neuronales tradicionales. Se presentan resultados experimentales del prototipo implementado en una plataforma de HIL Typhoon 402.

5.1 Esquema general del control neuronal B-spline

El sistema de control adaptativo basado en redes neuronales B-spline (BSNN) considera la naturaleza no lineal de la planta y la carga para adaptarse a los cambios en los parámetros para regular la velocidad del motor. En la Figura 5.1 se muestra el diagrama general del sistema de control neuronal B-spline, diseñado para probar su desempeño bajo diferentes condiciones.

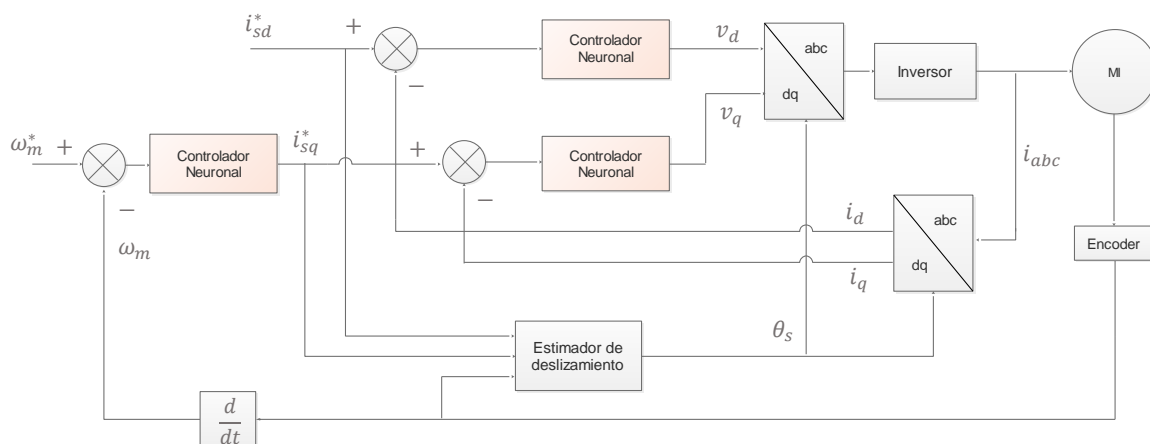


Figura 5.1 Esquema de control neuronal B-spline para el motor de inducción.

5.2 Control B-spline

Para implementar una estrategia de control B-spline se deben definir tres elementos principales: un conjunto de funciones de activación, un rango de pesos de aprendizaje y una condición inicial. Adicionalmente se deben definir parámetros secundarios de diseño como el orden de las funciones base, la velocidad de aprendizaje y la distribución de los nodos. En la Figura 5.2 se muestra el diagrama de flujo del sistema de control a implementar.

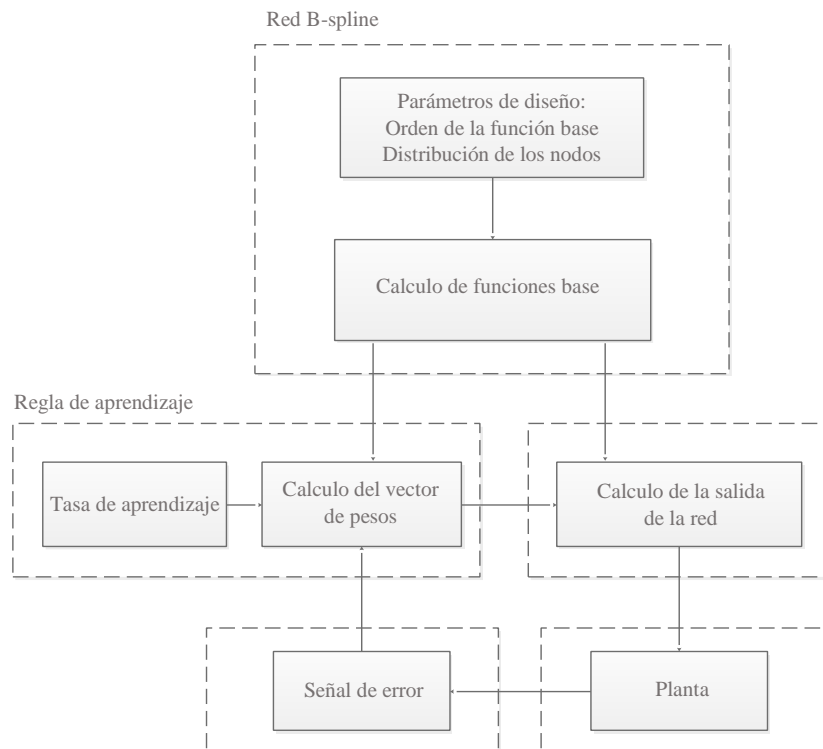


Figura 5.2 Diagrama de flujo del sistema de control B-spline.

La función de salida de la red B-spline es una combinación lineal de las salidas de la función base. Los coeficientes de la combinación lineal son algunos pesos ajustables w_i (el aprendizaje).

$$y = \sum_{i=1}^p a_i w_i = a^T w \quad (5.2.1)$$

Las salidas de las funciones básicas dependen no linealmente de las entradas, por lo que la última expresión puede reescribirse en el tiempo t como:

$$y(t) = a^T(t)w(t-1) = a^T(x(t))w(t-1) \quad (5.2.2)$$

Donde a es un vector p -dimensional que contiene las salidas base de la función, w es el vector que contiene los pesos, y finalmente x es el vector de entrada a la red [40].

La red B-spline actúa como un controlador al forzar una señal de error de entrada e_y a modificar la variable controlada por la salida, haciendo que la salida del sistema se comporte como se desee.

La fórmula para obtener el aprendizaje en línea es:

$$\Delta w(t - 1) = \frac{\gamma e_y(t)}{\|a(t)\|_2^2} a(t) \quad (5.2.3)$$

La ecuación (5.2.4) representa la señal de error:

$$e_y = y(t) - \hat{y}(t) \quad (5.2.4)$$

En la Figura 5.3 se muestra el bloque que representa la red B-spline a emplear.

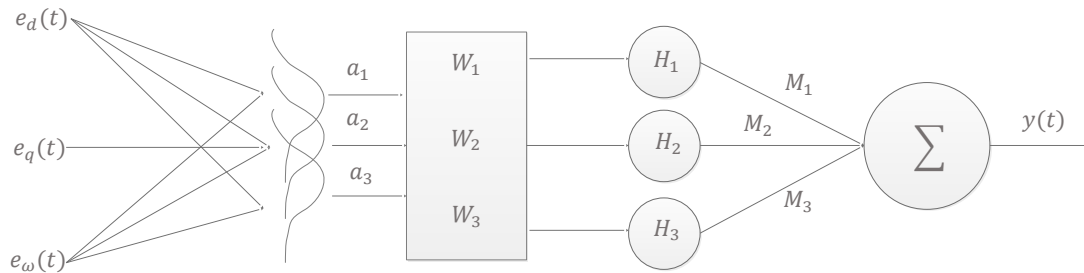


Figura 5.3 Estructura de una red neuronal B-spline.

La BSNN está compuesta por tres capas: 1) un espacio de entrada l – dimensional, 2) un conjunto de funciones básicas p , y 3) la función de salida de la red. A pesar de que se definió una estructura para la BSNN, no siempre será fija, ya que se puede adaptar a una amplia gama de condiciones de operación en cada periodo de muestreo, puede adaptar su salida cada que le sea aplicada una perturbación externa.

5.3 Control de velocidad basado en redes B-spline

La tarea principal del controlador neuronal será ajustar los parámetros de K_p y K_i para regular la velocidad del rotor, se emplean tres señales de error e_d , e_q y e_ω que son la diferencia entre las señales de referencia i_d , i_q y ω_m y las señales reales o medidas, ver Figura 5.1.

El controlador empleara las entradas y salidas pasadas para determinar un valor óptimo de las ganancias K_p y K_i , que minimicen el error lo mar cercano a cero, para cada controlador PI, ver Figura 5.4.

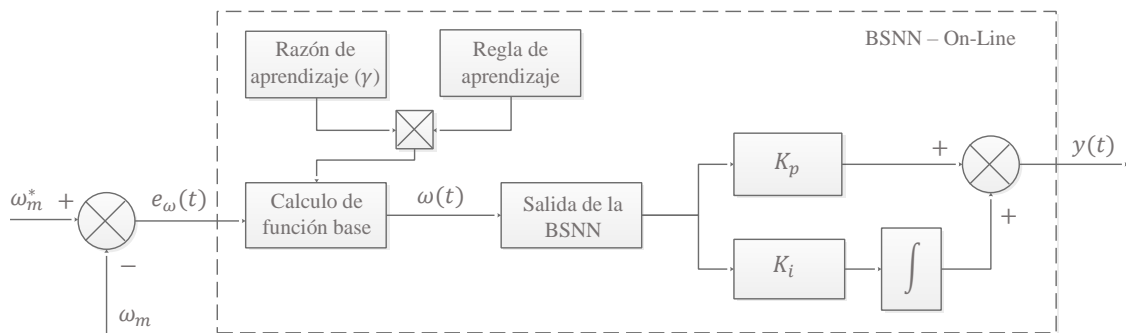


Figura 5.4 Estructura de un controlador neuronal B-spline.

La salida de la BSNN se obtiene mediante la combinación lineal de las salidas de las funciones base. La salida de la BSNN está definida como:

$$y = \sum_{i=1}^P a_i \omega_i = a^T w \quad (5.3.1)$$

Donde a es vector p – dimensional que contiene las salidas de la función basa para $P = 1, \dots, 4$ y, w es el vector de pesos. La función base B-Spline es definida de la siguiente manera:

$$a_K^j(e_\omega) = \left(\frac{e - \lambda_{j-K}}{\lambda_{j-1} - \lambda_{j-K}} \right) a_{K-1}^{j-1}(e_\omega) + \left(\frac{\lambda_j - e_\omega}{\lambda_j - \lambda_{j-K+1}} \right) a_{K-1}^j(e)$$

$$a_{K-1}^j(e_\omega) = \begin{cases} 1, & e_\omega \in I_j \\ 0, & \text{otro} \end{cases} \quad (5.3.2)$$

Donde λ_j es el j – esimo punto de control y $I_j = [\lambda_{j-1} \quad \lambda_j]$ es el j – esimo intervalo para $j = 1, \dots, 4$, y K es el orden de la función B-spline la cual puede tomar valores del 1 al 4. Los valores para cada punto de control λ_j son definidos de acuerdo al rango de los posibles valores de la señal de error.

La regla de aprendizaje que se emplea es la de aprendizaje en línea, en el cual los pesos de las redes se actualizan después de que se procesa cada ejemplo de aprendizaje. La actualización de los pesos $\Delta W(t - 1)$ se realiza mediante una regla de aprendizaje instantánea, como se muestra a continuación:

$$\Delta W(t - 1) = \frac{\eta e_\omega(t)}{\|a(t)\|_2^2} a(t) \quad (5.3.3)$$

Donde η es la relación de aprendizaje, e_ω es el error entre la velocidad actual y la velocidad deseada del rotor, a es el vector que contiene la salida de las funciones base, W es el vector de pesos; la actualización de los pesos depende de la salida de las funciones base y la regla de aprendizaje.

De [37] [45] sabemos que no existe un método universal para determinar el número óptimo de parámetros para una red, todo depende de la estructura del problema en cuestión. La red B-spline integra los parámetros aparentemente dispersos de la red y la lógica difusa para proporcionar redes de aprendizaje basada en datos que garanticen el aprendizaje, convergencia y estabilidad

5.4 Simulación del control de velocidad B-spline

Para simular el funcionamiento del controlador neuronal, se modelan tres funciones en MATLAB/Simulink que describen el comportamiento de cada red neuronal, dichas funciones están descritas a partir de la ecuación 5.3.3.

$$[K_{p_x}, K_{i_x}] = BSNN_m(e_x, w_i) \quad (5.4.1)$$

Donde $BSNN_m$ es una función que denota la red neuronal B-spline usada para calcular los valores de K_p y K_i para cada controlador PI a partir de su correspondiente error e_x ($x=d, q, \omega$) y w_i es i -ésimo factor de peso. En la Figura 5.5 se muestra un esquema la red B-spline a emplear.

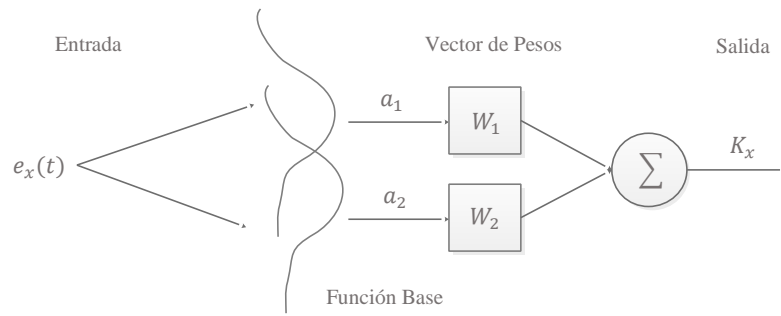


Figura 5.5 Estructura base de una red neuronal B-spline.

El valor de λ_x se define de acuerdo a una función recursiva definida por la ecuación 5.3.2, se encuentra acotado a un rango de posibles valores impuestos por la señal de error (ver Figura 5.5), para su cálculo en MATLAB/Simulink se emplean tres funciones dadas por la ecuación 5.4.2.

$$N_{x,y} = Fb_1var3ordenxx(e_x) \quad (5.4.2)$$

Donde FB_m es la función base usada para calcular los valores de λ_x partir de su correspondiente error e_x ($x=d, q, \omega$).

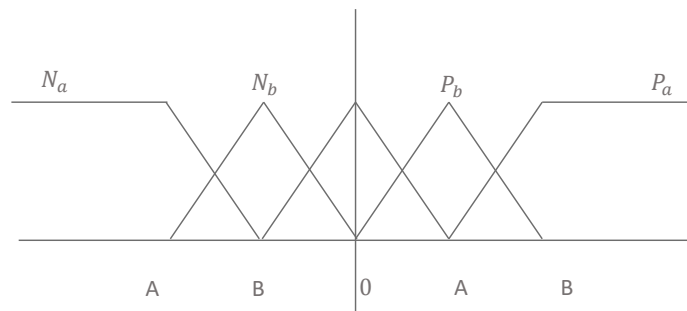


Figura 5.6 Función propuesta para describir los rangos de error.

Debido a que no contamos con una métrica para definir la tasa de aprendizaje se probaron diferentes valores y se encontró que $\eta = 0.05$ es la más adecuada, si η se establece cerca de 0, el entrenamiento es lento, por el contrario, si es muy grande ocurren oscilaciones. Siguiendo a [44] sabemos que los valores de la tasa de aprendizaje se eligen en función de un compromiso entre el aprendizaje rápido y un mayor filtrado de ruido. El sistema de pruebas utilizado se puede ver Figura 5.6, como variables de entrada tenemos: $TE_{des}, Frd_{des}, \omega_{des}$ y como resultados tenemos i_d, i_q, T_e, Frd .

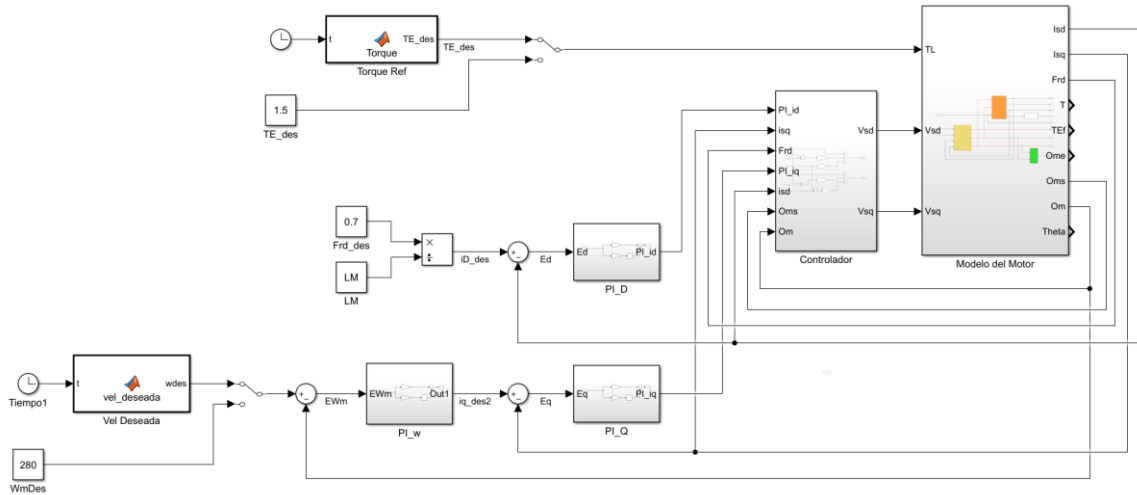


Figura 5.6 Esquema general del control neuronal en MATLAB/Simulink.

El perfil velocidad deseado está dado por un polinomio de Bézier, a fin de proporcionar una trayectoria suave entre el valor real y el de referencia. La trayectoria propuesta está definida por:

$$\omega^* = \begin{cases} \omega_1 + (\omega_2 - \omega_1)\psi(t, T_1, T_2) & \text{para } T_1 \leq t \leq T_2 \\ \omega_1 & \text{para } T_2 < t \leq T_3 \\ \omega_1 + (\omega_2 - \omega_1)\psi(t, T_3, T_4) & \text{para } T_3 < t \leq T_4 \\ \omega_2 & \text{para } t > T_4 \end{cases}$$

Donde $\omega_1 = 190 \text{ RPM}$, $\omega_2 = 140 \text{ RPM}$, $T_1 = 0 \text{ s}$, $T_2 = 1.5 \text{ s}$, $T_3 = 4 \text{ s}$, $T_4 = 5 \text{ s}$. Los perfiles de flujo y de par deseados los podemos ver en la Figura 5.7

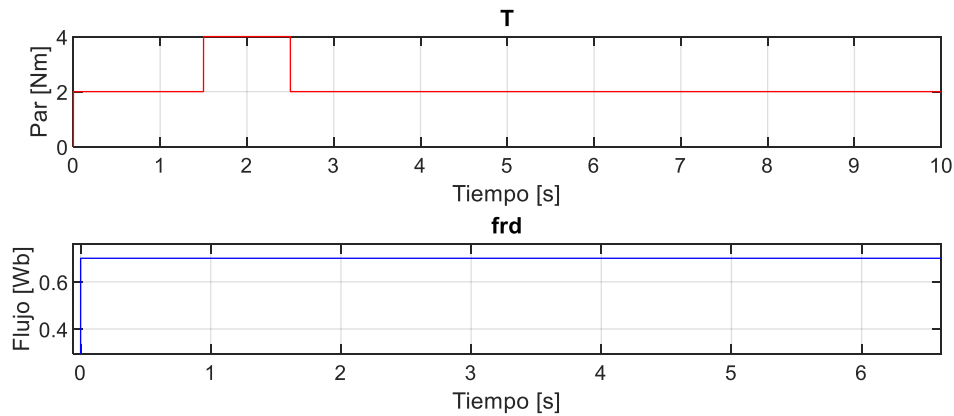


Figura 5.7 Resultados de la simulación: a) Perfil de par y b) Perfil de flujo.

En la Figura 5.8 se puede ver la gráfica de seguimiento de flujo: a) El flujo de referencia (F_{rd}^*) en comparación con el flujo del rotor (F_{rd}), b) El torque de referencia (T^*) contra el torque medido (T).

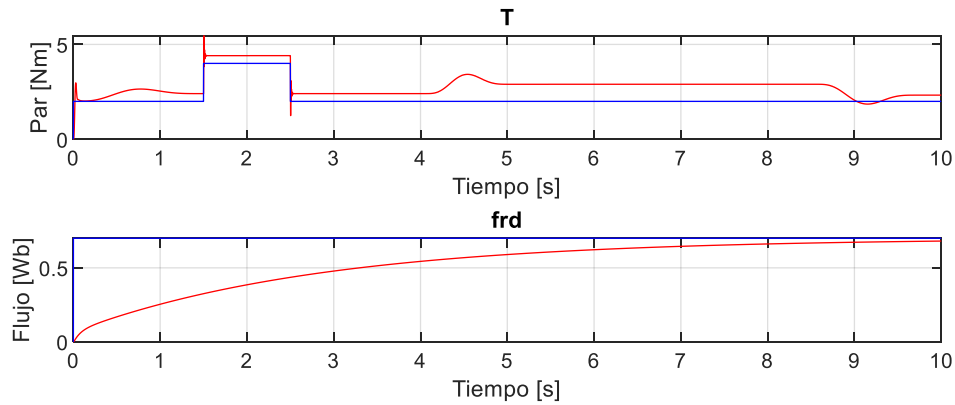


Figura 5.8 Resultados de la simulación del modelo: a) Flujo y b) Torque.

Los resultados de simulación de las corrientes i_d y i_q se muestran en la Figura 5.9.

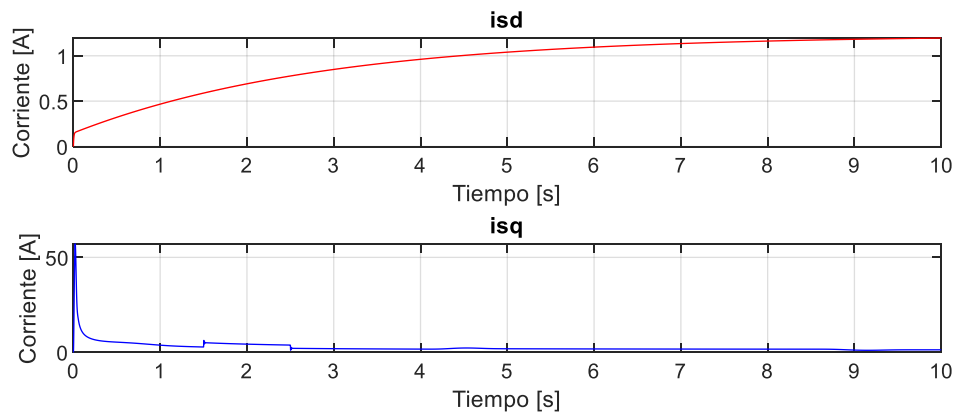


Figura 5.9 Resultados de la simulación del modelo: a) Corriente i_d , b) Corriente i_q .

En la Figura 5.10 y 5.11 se pueden ver las gráficas de seguimiento de dos perfiles de velocidad al emplear una ANN B-spline.

La Figura 5.10 muestra el seguimiento de un perfil de velocidad a) La velocidad del motor ω_m presenta un rendimiento bajo ante cambios bruscos de velocidad en comparación con la velocidad de referencia propuesta ω_m^* y b) Los niveles de error son altos ante perturbaciones en el arranque.

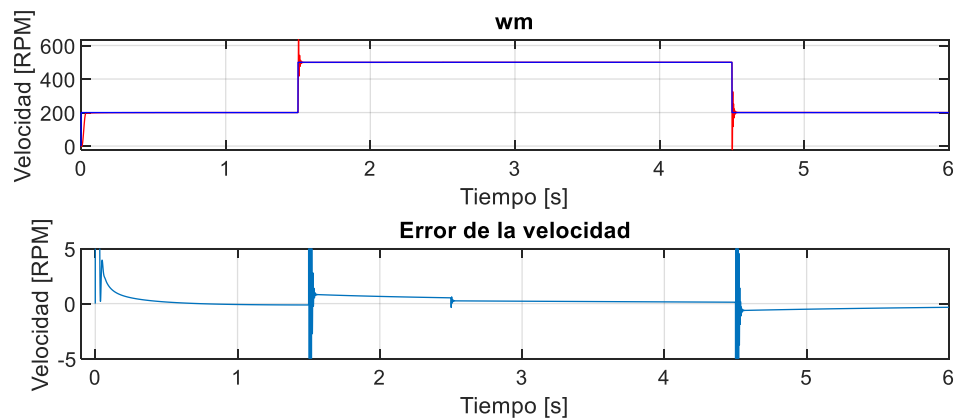


Figura 5.10 a) Velocidad de referencia - velocidad del rotor, b) Error de la velocidad.

En la Figura 5.11 se muestra el seguimiento de un segundo perfil de velocidad: a) La velocidad del motor ω_m alcanza rápida y suavemente el perfil de velocidad deseada ω_m^* . b) Así mismo se muestra la evolución del error de velocidad al aplicar cambios en los valores de la referencia en diferentes momentos.

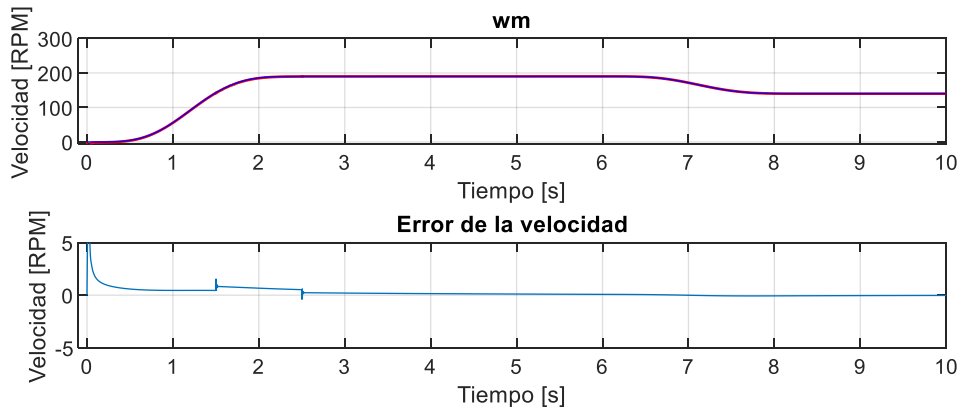


Figura 5.11 a) Velocidad de referencia - velocidad del rotor, b) Error de la velocidad.

Como se aborda en la literatura [44], una de las ventajas del control neuronal es el permitir actualizar los pesos continuamente a fin de alcanzar la convergencia a sus valores óptimos. Este proceso de actualización de ganancias a lo largo del tiempo se puede ver en la figura 5.12 y 5.13.

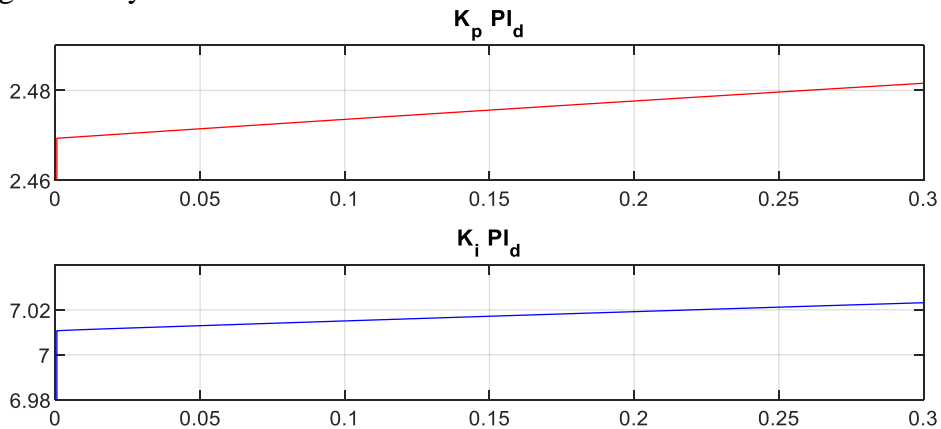


Figura 5.12 Comportamiento de las ganancias del controlador, a) Ganancia proporcional para el PI de la corriente I_d , b) Ganancia integral para el PI de la corriente I_d .

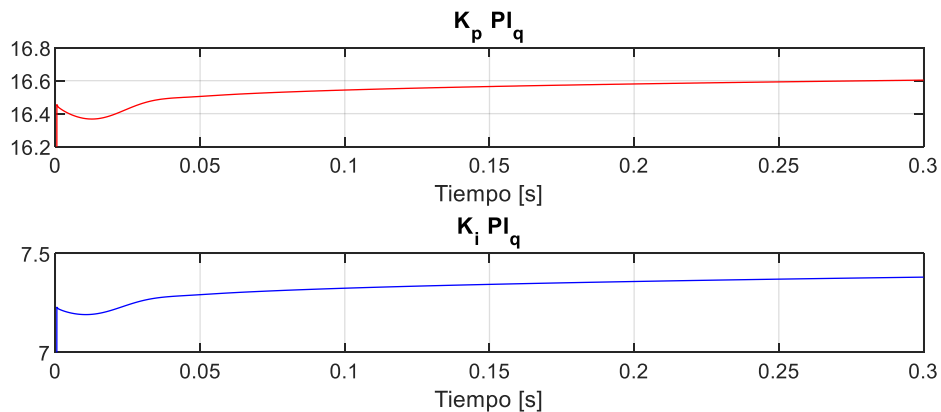


Figura 5.13 Comportamiento de las ganancias del controlador, a) Ganancia proporcional para el PI de la corriente I_q , b) Ganancia integral para el PI de la corriente I_q .

Se procede a probar el grado de influencia de la perturbación del valor de los parámetros físicos del motor y la variación del par de carga en el rendimiento del controlador de velocidad: a) aumentando en un 200% el valor de la resistencia del estator R_s , 50% el momento de inercia del rotor J_m , b) disminuyendo un 25% la inductancia del estator.

La gráfica de las figuras 5.14 muestra la respuesta del controlador de velocidad neuronal al variar algunos valores de los parámetros nominales del motor (J_m, R_s, L_s).

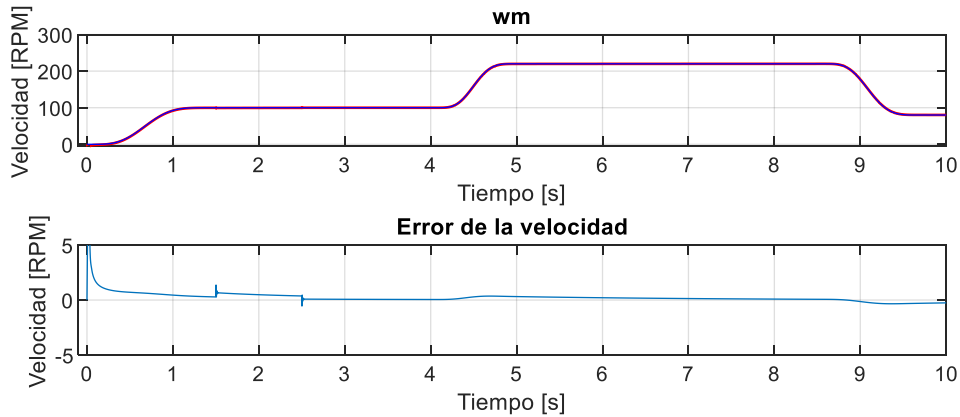


Figura 5.14 a) Velocidad de referencia - velocidad del rotor, b) Error de la velocidad.

5.5 Esquema de control neuronal B-spline en PSIM

Retomando el esquema del control vectorial de la Figura 4.16 se desprende el esquema de control B-spline de la Figura 5.15 construido a partir de elementos que pueden ser sintetizados de manera automática a código en C (SimCoder) para posteriormente ser descargados al DSP C2000 (TMS320F28335).

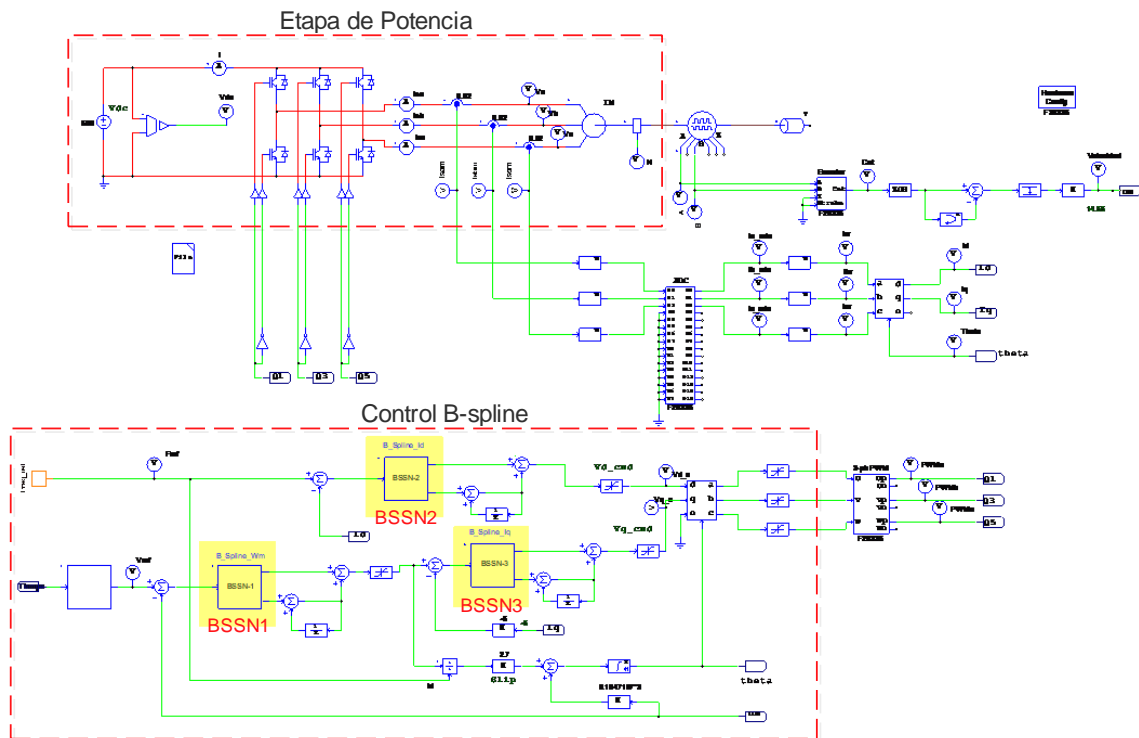


Figura 5.15 Esquema del control neuronal B-spline.

Para codificar cada red neuronal B-spline en PSIM, se ocupan bloques *Simplified C Block*, este bloque permite codificar directamente en lenguaje C sin compilar. El intérprete de PSIM ejecuta dicho bloque de código en cada ciclo de operación.

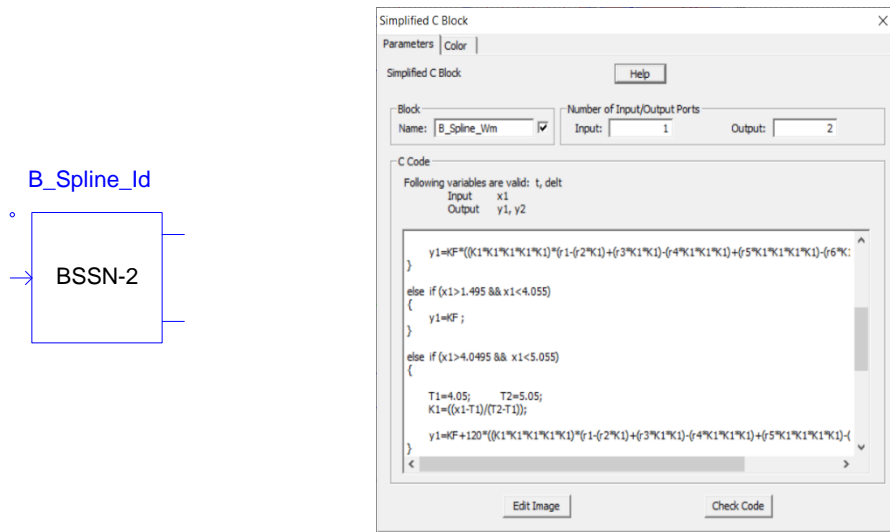


Figura 5.16. Bloque *Simplified C Block* de PSIM.

5.5.1 Rutina para la red neuronal

Una vez que la red neuronal recibe el valor de error, se ejecuta la rutina para el cálculo de las ganancias de los controladores PI.

$$[Kp_w, Ki_w] = BSNN_W(x)$$

5.5.2 Rutina para el cálculo de la función base

El error se divide por una constante para lograr que su valor se escale dentro de un determinado rango. El error escalado se envía a la función de activación o Spline, el resultado se multiplica por el valor de aprendizaje y será regresado a la rutina principal BSNN_x.

$$Kp_w = Fbase(x)$$

5.5.3 Resultados experimentales

Finalmente, en las Figura 5.17 y 5.18 se muestran la gráficas de seguimiento de dos perfiles de velocidad obtenidas del HIL, en el primer caso el motor parte de reposo y aumenta su velocidad de 0 a 1200 rpm. El sistema presenta un tiempo de elevación de 1s y un tiempo de asentamiento de 11s. Con estos resultados podemos ver que el controlador neuronal tarda más tiempo en llegar a su velocidad de referencia en comparación con el controlador vectorial de ganancias fijas, sin embargo presenta un mejor tiempo de respuesta y además es capaz de igualar la velocidad del motor ω_m con un perfil deseado $\omega_m^* = 1130 \text{ rpm}$ en un lapso de 11s, tiempo agregado por el tiempo de computo de la red neuronal.

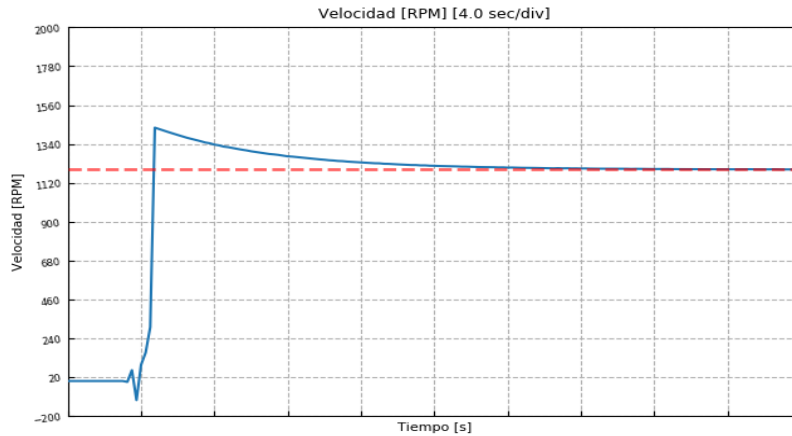


Figura 5.17 Perfil de velocidad deseado, frente a la respuesta de velocidad obtenida.

Para la segunda prueba el perfil velocidad deseada está dado por un polinomio de Bézier, a fin de proporcionar una trayectoria suave entre el valor real y el de referencia. En la Figura 5.18 se muestra la gráfica de seguimiento de una referencia de velocidad, el motor parte de reposo y aumenta su velocidad de 0 a 200 rpm y después de 200 a 310 rpm. Mediante esta variación podemos ver que el controlador neuronal es capaz de igualar la velocidad del motor ω_m con los perfiles deseados $\omega_m^* = 200$ rpm y $\omega_m^* = 310$ rpm.

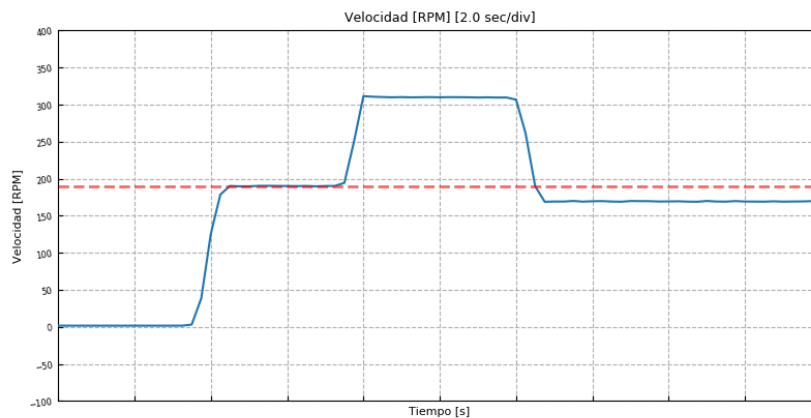


Figura 5.18 Perfil de velocidad deseado, frente a la velocidad medida.

Con ayuda de la emulación en Typhoon HIL se pudo verificar el buen desempeño del controlador neuronal al lograr llevar los parámetros de salida del motor en los valores deseados con un margen de error bastante bajo. Resulta evidente que esta estrategia de control demostró manejar adecuadamente la velocidad del motor ante diferentes perfiles de velocidad.

CONCLUSIONES

En el presente trabajo se ha propuesto un esquema de control neuronal para la regulación de la velocidad un motor de inducción tipo jaula de ardilla. Para lograr dicho objetivo se desarrolló un controlador basado en una red neuronal B-spline como medio para ajustar las ganancias fijas de los controladores PI del sistema de control vectorial. Este esquema nos permite actualizar sus ganancias proporcionales e integrales en tiempo real gracias a su capacidad de aprendizaje en línea, a su naturaleza rápida y simplicidad al momento de implementarse en un dispositivo de Hardware.

Los resultados de las simulaciones en MATLAB/Simulink demuestran el buen desempeño de control neuronal B-spline al lograr llevar los parámetros de salida del motor a los valores deseados, demostrando así una mejora en el rendimiento del sistema, una respuesta rápida, robustez ante la variación de los parámetros físicos y una velocidad de seguimiento precisa.

Se comprobó de manera experimental la efectividad de las redes neuronales B-spline para compensar perturbaciones externas, esto debido a la denominada actualización de pesos en línea que se da en cada periodo de muestreo. Por el contrario, se pudo comprobar que resulta bastante complejo definir los umbrales óptimos en los cuales la función se activara o no, una mala elección afectara el resultado devuelto por la red neuronal.

El software PSIM es una excelente herramienta de simulación de circuitos de potencia, posee la capacidad de generar código en C de forma automática que puede ejecutarse en tiempo real en un DSP, siendo con ello una excelente herramienta para la implementación de funciones matemáticas y funciones recursivas propias de una red neuronal B-spline.

Adicionalmente, se corroboró la efectividad y el buen desempeño de la implementación del sistema de control sobre una plataforma de Hardware in the Loop (HIL); resulta relativamente sencillo modificar los parámetros y por ende el comportamiento del sistema, ahorrándonos gran cantidad de tiempo de implementación ya que cada escenario de pruebas se reduce a implementar y probar.

TRABAJOS FUTUROS

Futuros trabajos están orientados a combinar el esquema de control neuronal con un esquema de control basado en estimadores de flujo y velocidad. Por lo cual se proponen las siguientes actividades:

- Realizar pruebas con el controlador B-spline a fin comparar su desempeño respecto a otras técnicas de control neuronal.
- Analizar las diferencias de comportamiento de un sistema al ser controlado por un controlador PI convencional y un neurocontrolador B-spline.
- Implementar un sistema de control neuronal para el motor de inducción sin sensor de velocidad.

BIBLIOGRAFÍA

- [1] M. R. Diana Fernanda, V.S. Jorge Olmedo, “Diseño de un Controlador de Velocidad para un Motor de Inducción utilizando Control Vectorial”, Tesis, Departamento de Electrónica, Instrumentación y Control, Universidad del Cauca, Popoyan, 2005
- [2] G. C. Jaime Antonio, A. de S. Marcos, P.P. Eduardo J. “Control vectorial del motor de inducción para el control de velocidad del rotor por cambio de frecuencia”, Revista Ciencia e Ingeniería, Vol 22. No 3. 2001
- [3] B. M. Luis Carlos L., “Control vectorial de par velocidad en motores de inducción”, Tesis, Escuela de Ingeniería Mecánica Eléctrica, Universidad de San Carlos de Guatemala, Guatemala, Noviembre 2005
- [4] F. Khorrami, P. Krishnamurthy, “Modeling and Adaptive Nonlinear Control of Electric Motors”, Springer, 2003
- [5] R. S. Marla Erika, “Simulación del control de velocidad por campo orientado de un motor de inducción utilizando un estimador de velocidad basado en observadores de flujo”, Tesis, Escuela Superior de Ingeniería Mecánica y Eléctrica, Instituto Politécnico Nacional, México, D.F. Mayo 2003
- [6] H. Joachim, Q. Juntao, “Sensorless Vector Control of Induction Motors at Very Low Speed Using a Nonlinear Inverter Model and Parameter Identification”, presented at the IEEE Transactions on Industry Applications, Vol. 38, No. 4, July/August 2002
- [7] M. R. Diana F, V. S. Jorge O, D. C. Jaime O, “Control Vectorial Directo para un Motor de Inducción”, Departamento de Electrónica, Instrumentación y Control, Universidad de Cauca, Abril 2015
- [8] C. M. Pedro C. “Control vectorial de la velocidad de una maquina síncrona de imanes permanente”, Tesis, Escuela Superior de Ingeniería Mecánica y Eléctrica, Instituto Politécnico Nacional, México, D.F. Junio 2011
- [9] E. Brahim, “Utilización del filtro de Kalman como estimador de velocidad en el control vectorial de motores de inducción”, Tesis, Escuela Superior de Ingeniería Mecánica y Eléctrica, Instituto Politécnico Nacional, Octubre 2001
- [10] G. M. Ali S. “Aplicación de las redes neuronales en los sistemas de control vectorial de los motores de inducción”, Tesis, Departamento de Ingeniería Eléctrica, Universidad Politécnica de Cataluña, Cataluña, 2000
- [11] G. H. Bode, P. C. Loh, M. J. Newman, and D. G. Holmes, “An improved robust predictive current algorithm,” IEEE Trans, Nov. 2005
- [12] M. Nemeč, D. Nedeljković, and V. Ambrožič, “Predictive torque control of induction machines using immediate flux control,” IEEE Trans. Ind. Electron. Aug. 2007.
- [13] J. I. Yuz and G. C. Goodwin, “On sampled-data models for nonlinear systems,” IEEE Trans., Oct. 2005.
- [14] Domenico Casadei, Giovanni Serra, and Angelo Tani “FOC and DTC: Two Viable Schemes for Induction Motors Torque Control” IEEE, Trans. Power Electro. SEPTEMBER 2002
- [15] I. Takahashi and T. Noguchi, “A new quick response and high efficiency control strategy of an induction motor,” IEEE Trans. Ind. Appl, Sep. /Oct. 1986.
- [16] A. Mir Sayeed, Donald, S. Zinger, and Malik Elbuluk, "Fuzzy controller for induction machine,"IEEE, Trans. On Industry applications, January February 1994.
- [17] Domenico Casadei, Giovanni Serra, and Angelo Tani “FOC and DTC: Two Viable Schemes for Induction Motors Torque Control” IEEE, Trans. Power Electro. SEPTEMBER 2002
- [18] C. Lascu, I. Boldea, and F. Blaabjerg, “Direct torque control of sensorless induction motor drives: A sliding-mode approach,” IEEE Trans, Apr. 2004.
- [19] D. Telford, M. W. Dunnigan, and B. W. Williams, “Adaptive high bandwidth current control for induction machines,” IEEE Trans. Mar. 2003.
- [20] J. Rodríguez, J. Pontt, S. Kouro, and P. Correa, “Direct torque control with imposed switching frequency in an 11-level cascaded inverter,” IEEE Trans. Aug. 2004

- [21] S. Bernet, "State of the art and developments of medium voltage converters an overview," in Proc. Int. Conf. PELINCEC, Warsaw, Poland, Oct. 16–19, 2005,
- [22] Grune, L., Pannek, J.: 'Nonlinear model predictive control: theory and algorithms' (Springer: Communications and Control Engineering Series, 2010)
- [23] Vazquez, S., Leon, J.I., Franquelo, L.G., et al.: 'Model predictive control – a review of its applications in power electronics', IEEE Ind. Electron. Mag., 2014
- [24] Cortés, P., Kazmierkowski, M.P., Kennel, R.M., et al.: 'Predictive control in power electronics and drives', IEEE Trans. Ind. Electron., 2008
- [25] Zhang, Y., Yang, H.: 'Model predictive torque control of induction motor drives with optimal duty cycle control', IEEE Trans. Power Electron., 2014
- [26] Martínez, R., J.: "Análisis comparativo de diferentes esquemas de control no lineal de posición y velocidad angular aplicado al motor de inducción trifásico tipo jaula de ardilla", Tesis, Universidad Tecnológica de la Mixteca, Huajuapán de León, Oaxaca, 2017
- [27] Rashid, M. (1993). Electrónica de potencia. México: Prentice-Hall.
- [28] Sánchez. C. J.: "Análisis electromecánico de una máquina de inducción con rotor jaula de ardilla empleando un modelo digital", Tesis, Instituto Politécnico Nacional, México, D.F, 2011
- [29] Navas. C. W.: "Obtención del modelo dinámico de un motor trifásico de inducción utilizando técnica de identificación de sistemas con lógica difusa", Tesis, Universidad Politécnica Salesiana, Ecuador, 2016
- [30] Vallespin, A. F.: "Control vectorial indirecto de motor asíncrono mediante microcontrolador", Tesis, Universidad Zaragoza, 2013
- [31] Rodríguez. V. C.: "Control de la velocidad de un motor de inducción trifásico empleando la técnica de campo orientado", Tesis, Universidad Nacional de Ingeniería, Lima – Perú, 2003
- [32] Fraile Mora, J. (2004). Máquinas eléctricas. Madrid [etc.]: McGraw-Hill-Interamericana de España.
- [33] Ramírez. N. J.: "Implementación de un sistema de control escalar de velocidad realimentado basado en un dsPIC33FJ16GS502 para un motor de inducción trifásico tipo jaula de ardilla de 0.5 HP", Tesis, Universidad Nacional del Callao, Facultad de Ingeniería Eléctrica y Electrónica, Perú, 2013
- [34] Marulanda. A. F.: "Control de velocidad de un motor de inducción por el método de control vectorial utilizando el Software de simulación SIMULINK de MATLAB", Tesis, Universidad Tecnológica de Pereira, 2010
- [35] S. Shah, A. Rashid, "Direct Quadrature (D-Q) Modeling of 3-Phase Induction Motor Using MATLAB/SIMULINK", Canadian Journal on Electrical and Electronics Engineering Vol. 3, No. 5, May 2012.
- [36] Krause, P.C., O. Wasynczuk, and S.D. Sudhoff, Analysis of Electric Machinery, IEEE® Press, 2002.
- [37] Rojas, R. Neural Networks, a Systematic Introduction, Berlin: Springer-Verlang, 1996.
- [38] Fausett, L. Fundamentals of Neural Networks, Architectures, Algorithms and Applications, 1994.
- [39] Clark. P. M., O, "Un Neurocontrolador para dispositivos de Electrónica de Potencia", Tesis, Centro de Investigaciones y Estudios Avanzados de I.P.N, Guadalajara, Jalisco, 2009.
- [40] Zuñiga. H. P, "Análisis y control de un compensador serie", Tesis, Centro de Investigaciones y Estudios Avanzados de I.P.N, Guadalajara, Jalisco, 2006.
- [41] Tapia. O. R, "Utilizacion del Neurocontrolador B-spline para regular el StatCom", Tesis, Centro de Investigaciones y Estudios Avanzados de I.P.N, Guadalajara, Jalisco, 2006.
- [42] Escorcía, H.J., Aguilar, S. H., Aguilar, M. O., et al.: "An Intelligent Compensation through B-spline Neural Network for a Delta Parallel Robot", ResearchGate, 2019.
- [43] Tapia, O.R., Beltran, C. F., Aguilar, M. O., et al.: "An Adaptive Speed Control Approach for DC Shunt Motors", MDPI, 2016.
- [44] Aguilar, M. O, Tapia, O.R., Beltran, C. F., et al.: "Adaptive Neural Network Control of Chaos in Permanent Magnet Synchronous Motor", Intelligent Automation and Soft Computing, Vol. 15, 2009.

Apéndice A

CURVAS DE BÉZIER EN MATLAB

1. Rutina A

```
function wdes=vel_deseada(t)

KF=190;
r1 = 252;   r2 = 1050; r3 = 1800; r4 = 1575; r5 = 700; r6 = 126;

if (t<2.700)
    T1=0; T2=2.7;
    K1=((t-T1)/(T2-T1));
    wdes=KF*[(K1.^5).*(r1-(r2.*K1)+(r3*K1.^2)-(r4*K1.^3)+(r5*K1.^4)-
(r6*K1.^5))];
elseif (t>2.699)&(t<6.000)
    wdes=KF ;
elseif (t>5.999)&(t<8.499)
    T1=6;      T2=8.5;
    K1=((t-T1)/(T2-T1));
    wdes=KF-50*[(K1.^5).*(r1-(r2.*K1)+(r3*K1.^2)-(r4*K1.^3)+(r5*K1.^4)-
(r6*K1.^5))];
else
    wdes=KF-50;
end
```

2. Rutina B

```
function wdes=vel_deseada(t)

KF=100;
r1 = 252;   r2 = 1050; r3 = 1800; r4 = 1575; r5 = 700; r6 = 126;

if t<1.5
    T1=0; T2=1.5;
    K1=((t-T1)/(T2-T1));
    wdes=KF*[(K1.^5).*(r1-(r2.*K1)+(r3*K1.^2)-(r4*K1.^3)+(r5*K1.^4)-
(r6*K1.^5))];
elseif (t>1.495)&(t<4.055)
    wdes=KF ;
elseif (t>4.0495)&(t<5.055)
    T1=4.05; T2=5.05;
    K1=((t-T1)/(T2-T1));
    wdes=KF+120*[(K1.^5).*(r1-(r2.*K1)+(r3*K1.^2)-
(r4*K1.^3)+(r5*K1.^4)-(r6*K1.^5))];
elseif (t>5.05)&(t<8.55)
    wdes=KF+120 ;
elseif (t>8.549)&(t<9.75)
    T1=8.55; T2=9.75;
    K1=((t-T1)/(T2-T1));
    wdes=KF+120-140*[(K1.^5).*(r1-(r2.*K1)+(r3*K1.^2)-
(r4*K1.^3)+(r5*K1.^4)-(r6*K1.^5))];
else
    wdes=KF-20;
end
```

CONTROL VECTORIAL EN MATLAB

```

global Te TL WmDes EWm
global Ki_w Kp_w Kp_iD Ki_iD Kp_iQ Ki_iQ
global EFLU Ed Eq

isd=x(5,1);
isq=x(6,1);
frd=x(7,1);
Om=x(8,1);

%*****
%                               Parámetros físicos de la máquina
%*****

Rs=11.2;           Rr=8.3;
LS=0.6155;        LR=0.638;
LM=0.57;          Jm=0.0024;          F=0.0041;
Zp=2;

%*****
%                               constantes de la máquina
%*****

LL1  = LR*LS;
TaoS=LS/Rs;
sigma= 1 - ( (LM*LM) / LL1);
TaoR=LR/Rr;
KR=LM/LR;
rsig=Rs+(Rr*KR*KR);
TpSi=(sigma*LS)/rsig;
KTE=1.5*Zp*(LM)/LR;

Ome=Om*Zp;
Oms=Ome+((LM*isq)/(TaoR*frd));
%*****
%                               Par de carga
%*****

if ( t < 1.5 ) | ( t > 2.5 )
    TL = 2;
else
    TL = 4;
end
%*****
%                               Velocidad deseada
%*****

if ( t < 1.5 ) | ( t > 4.5 )
    WmDes = 200;
else
    WmDes = 500;
end
%*****
%                               Error de la velocidad
%*****

EWm=WmDes-Om;
Frd_des=0.7;
TE_des=TL;

```

```

%*****
%
%                               Corriente deseada en eje id
%*****

iD_des=frd_des/LM;
EFLU=frd_des-frd;

%*****
%
%                               Corriente deseada en eje iq
%*****

iD_des=frd_des/LM;
diqdes=EWm*Ki_w;
dIDdes=0;
iq_des2=(EWm*Kp_w)+x(1,1);

%*****
%
%                               Controladores PI
%*****

Ed=iD_des-isd;                               Eq=iq_des2-isq;

%*****
%
%                               Ganancias de los controladores
%*****

PI_id=(Ed*Kp_id)+x(3,1); PI_iq=(Eq*Kp_iq)+x(4,1);
dpid=Ed*Ki_id; dpiq=Eq*Ki_iq;

%*****
%
%                               Ecuaciones del controlador
%*****

Vsd=PI_id - ( (KR/TaoR)*frd ) - ( Oms*isq*rsig*TpSi);
Vsq=PI_iq + (Oms*isd*rsig*TpSi) + ( KR*frd*Om);

%*****
%
%                               Ecuaciones Eléctricas de la Máquina
%*****

dIsd=-((1/TpSi)*isd)+(Oms*isq)+((KR/(rsig*TpSi*TaoR))*frd)+
(Vsd/(rsig*TpSi));
dIsq=- (Oms*isd)-((1/TpSi)*isq)-((KR/(rsig*TpSi))*frd*Ome)+
(Vsq/(rsig*TpSi));
dfrd=((LM/TaoR)*isd)-(frd/TaoR);
Te=1.5*Zp*(LM*frd*isq)/LR;

%*****
%
%                               Ecuaciones Mecánicas de la Máquina
%*****

dwm=(Te/Jm)-(TL/Jm)-(F/Jm)*Om;
dTets=Oms;

dx=[diqdes; dIDdes; dpid; dpiq; dIsd; dIsq; dfrd; dwm; dTets];

```

RED NEURONAL B-SPLINE EN MATLAB

1. Rutina para el cálculo de la función base A

$$Na_b = Fb_1var3orden11(x)$$

```
lambda_1 = -0.5;
lambda0 = 3.5;
lambda1 = 7.5;
lambda2 = 11.5;

if x>=lambda_1&&x<lambda0
    N1_0 = 1;
else
    N1_0 = 0;
end

if x>=lambda0&&x<lambda1
    N1_1 = 1;
else
    N1_1 = 0;
end

if x>=lambda1&&x<lambda2
    N1_2 = 1;
else
    N1_2 = 0;
end

N2_1 = ((x-lambda_1)/(lambda0-lambda_1))*N1_0 + ((lambda1-x)/(lambda1-
lambda0))*N1_1;
N2_2 = ((x-lambda0)/(lambda1-lambda0))*N1_1 + ((lambda2-x)/(lambda2-
lambda1))*N1_2;
N3_2 = ((x-lambda_1)/(lambda1-lambda_1))*N2_1 + ((lambda2-x)/(lambda2-
lambda0))*N2_2;
```

2. Rutina para el cálculo de la función base B

$$Na_b = Fb_1var3orden22(x)$$

```
lambda_1 = -15;
lambda0 = -8;
lambda1 = -1;
lambda2 = 6;

if x>=lambda_1&&x<lambda0
    N1_0 = 1;
else
    N1_0 = 0;
end

if x>=lambda0&&x<lambda1
    N1_1 = 1;
else
    N1_1 = 0;
end

if x>=lambda1&&x<lambda2
    N1_2 = 1;
```

```

else
    N1_2 = 0;
end

N2_1 = ((x-lambda_1)/(lambda0-lambda_1))*N1_0 + ((lambda1-x)/(lambda1-
lambda0))*N1_1;
N2_2 = ((x-lambda0)/(lambda1-lambda0))*N1_1 + ((lambda2-x)/(lambda2-
lambda1))*N1_2;
N3_2 = ((x-lambda_1)/(lambda1-lambda_1))*N2_1 + ((lambda2-x)/(lambda2-
lambda0))*N2_2;

```

Rutina para el cálculo de las ganancias del controlador de velocidad

$$[Kp_w, Ki_w] = BSNN_W(x)$$

```

global wKi_w wKp_w Kp_w Ki_w EWm

deltaX = 0.05;

uY_err = EWm/2550;

a = [Fb_1var3orden11(uY_err)...
     Fb_1var3orden22(uY_err)];
B = norm(a,2)^2; if B==0, B=1e-3; end

wKp_w = wKp_w + deltaX*(uY_err/B)*(a. ');
Kp_w = a*wKp_w;

a = [Fb_1var3orden11(uY_err)...
     Fb_1var3orden22(uY_err)];
B = norm(a,2)^2; if B==0, B=1e-3; end

wKi_w = wKi_w + deltaX*(uY_err/B)*(a. ');
Ki_w = a*wKi_w;

```

Rutina para el cálculo de las ganancias del controlador de corriente Iq

$$[Kp_iQ, Ki_iQ] = BSNN_Iq(x)$$

```

global wKp_q wKi_q Kp_iQ Ki_iQ Eq

deltaX = 0.05;

uY_err = Eq/2500;

a = [Fb_1var3orden11(uY_err)...
     Fb_1var3orden22(uY_err)];
B = norm(a,2)^2; if B==0, B=1e-3; end

wKp_q = wKp_q + deltaX*(uY_err/B)*(a. ');
Kp_iQ = a*wKp_q;

a = [Fb_1var3orden11(uY_err)...
     Fb_1var3orden22(uY_err)];
B = norm(a,2)^2; if B==0, B=1e-3; end

wKi_q = wKi_q + deltaX*(uY_err/B)*(a. ');
Ki_iQ = a*wKi_q ;

```

Rutina para el cálculo de las ganancias del controlador de corriente Id

$$[Kp_{iD}, Ki_{iD}] = BSNN_{Id}(x)$$

```
global wKp_d wKi_d Kp_iD Ki_iD Ed
deltaX = 0.05;
uY_err = Ed/2500;
a = [Fb_lvar3orden11(uY_err)...
     Fb_lvar3orden22(uY_err)];
B = norm(a,2)^2; if B==0, B=1e-3; end
wKp_d = wKp_d + deltaX*(uY_err/B)*(a. ');
Kp_iD = a*wKp_d;
a = [Fb_lvar3orden11(uY_err)...
     Fb_lvar3orden22(uY_err)];
B = norm(a,2)^2; if B==0, B=1e-3; end
wKi_d = wKi_d + deltaX*(uY_err/B)*(a. ');
Ki_iD = a*wKi_d ;
```

Apéndice B

CURVA DE BÉZIER EN CODIGO C

```
Int KF=190;
int r1 = 252;
int r2 = 1050;
int r3 = 1800;
int r4 = 1575;
int r5 = 700;
int r6 = 126;
float T1;
float T2;
float K1;

if (x1<1.5)
{
    T1=0; T2=1.5;
    K1 = ( (x1-T1) / (T2-T1) );
    y1=KF*((K1*K1*K1*K1*K1)*(r1-(r2*K1)+(r3*K1*K1)-(r4*K1*K1*K1)+(r5*K1*K1*K1*K1)-(r6*K1*K1*K1*K1*K1)));
}
else if (x1>1.495 && x1<4.055)
{
    y1=KF ;
}
else if (x1>4.0495 && x1<5.055)
{
    T1=4.05; T2=5.05;
    K1=((x1-T1)/(T2-T1));
    y1=KF+120*((K1*K1*K1*K1*K1)*(r1-(r2*K1)+(r3*K1*K1)-(r4*K1*K1*K1)+(r5*K1*K1*K1*K1)-(r6*K1*K1*K1*K1*K1)));
}
else if (x1>5.05 && x1<8.55)
{
    y1=KF+120 ;
}
else if (x1>8.549 && x1<9.75)
{
    T1=8.55; T2=9.75;
    K1=((x1-T1)/(T2-T1));
    y1=KF+120-140*((K1*K1*K1*K1*K1)*(r1-(r2*K1)+(r3*K1*K1)-(r4*K1*K1*K1)+(r5*K1*K1*K1*K1)-(r6*K1*K1*K1*K1*K1)));
}
else
{
    y1=KF-20;
}
```

CONTROL VECTORIAL EN CODIGO C

```
#include <math.h>
#include "PS_bios.h"
typedef float DefaultType;
#define GetCurTime() PS_GetSysTimer()

float grafica1[4000];
int i=0;
```

```

interrupt void Task();

DefaultType fGblFreq_ref = 8;
DefaultType fGblVel_ref = 1200;
DefaultType fGblVref = 0.0;
DefaultType fGblV80 = 0.0;
DefaultType fGblFref = 0.0;
DefaultType fGblTheta = 0.0;
DefaultType fGblId = 0.0;
DefaultType fGblIq = 0.0;
int nGblC_BUFFER_12 = 0;
DefaultType fGblC_BUFFER_12[] = {0,0,0,0,0,0,0,0,0,0,0,0};
DefaultType fGblVelocidad = 0.0;
DefaultType fGblVd_c = 0.0;
DefaultType fGblVq_c = 0.0;
DefaultType fGblVa_PWM = 0.0;
DefaultType fGblVb_PWM = 0.0;
DefaultType fGblVc_PWM = 0.0;
DefaultType fGblV84 = 0.0;
DefaultType fGblV85 = 0.0;
DefaultType fGblV86 = 0.0;
DefaultType fGblCnt = 0.0;
DefaultType fGblIar = 0.0;
DefaultType fGblIbr = 0.0;
DefaultType fGblIcr = 0.0;
DefaultType fGblIa_adc = 0.0;
DefaultType fGblIb_adc = 0.0;
DefaultType fGblIc_adc = 0.0;

interrupt void Task()
{
    DefaultType fTI_ADC1, fFCNM5, fTI_ADC1_1, fFCNM8, fTI_ADC1_2, fFCNM9, fVel_ref,
    fTI_ENCODER1;
    DefaultType fZOH16, fSUM15, fLIMIT_RANGE2, fP52, fSUM11, fP35, fP34, fB14,
    fSUMP17, fLIM9;
    DefaultType fFreq_ref, fDIVD2, fInvt_Tao_r, fP42, fSUMP20, fI_RESET_I_D3, fABC2,
    fABC2_1;
    DefaultType fABC2_2, fC_BUFFER_12, fSUM13, fP36, fP37, fB15, fSUMP18, fLIM10,
    fP40, fSUM12;
    DefaultType fP38, fP39, fB17, fSUMP19, fLIM3, fDQ02, fDQ02_1, fDQ02_2, fLIM11,
    fLIM14;
    DefaultType fLIM15;
    PS_EnableIntr();
    fC_BUFFER_12 = fGblC_BUFFER_12[nGblC_BUFFER_12];
    fTI_ADC1 = PS_GetDcAdc(0);
    fTI_ADC1_1 = PS_GetDcAdc(1);
    fTI_ADC1_2 = PS_GetDcAdc(2);
    fFCNM5 = (fTI_ADC1-1.5)*50;
    fFCNM8 = (fTI_ADC1_1-1.5)*50;
    fFCNM9 = (fTI_ADC1_2-1.5)*50;
    fVel_ref = fGblVel_ref;
#ifdef _DEBUG
    fGblVref = fVel_ref;
#endif

    fTI_ENCODER1 = (int32)PS_GetEncoder2Count();
    fZOH16 = fTI_ENCODER1;
    fSUM15 = fZOH16 - fC_BUFFER_12;
    fLIMIT_RANGE2 = fSUM15;
    for (;;)
    {
        if (fLIMIT_RANGE2 >= (1024.0/4))
            fLIMIT_RANGE2 -= (1024.0/4) - (-1024.0/4);
        else if (fLIMIT_RANGE2 < (-1024.0/4))

```

```

        fLIMIT_RANGE2 += (1024.0/4) - (-1024.0/4);
    else
        break;
}
#ifdef _DEBUG
    fGblV80 = fLIMIT_RANGE2;
#endif

fP52 = fLIMIT_RANGE2 * 14.700044343891;
fSUM11 = fVel_ref - fP52;
fP35 = fSUM11 * 0.95;
fP34 = fSUM11 * 0.149;
{
    static DefaultType out_A = 0, in_A = 0.0;
    fB14 = out_A + 0.5/10000 * (fP34 + in_A);
    out_A = fB14; in_A = fP34;
}
fSUMP17 = fP35 + fB14;
fLIM9 = (fSUMP17 > 40) ? 40 : ((fSUMP17 < -40) ? -40 : fSUMP17);
fFreq_ref = fGblFreq_ref;
#ifdef _DEBUG
    fGblFref = fFreq_ref;
#endif

fDIVD2 = fLIM9 / fFreq_ref;
fInvt_Tao_r = fDIVD2 * 2.7;
fP42 = fP52 * (0.1047187*3);
fSUMP20 = fInvt_Tao_r + fP42;
{
    static DefaultType out_A = 0;
    fI_RESET_I_D3 = out_A + 1.0/10000 * (fSUMP20);
    if (fI_RESET_I_D3 > 6.283185) {
        fI_RESET_I_D3 -= 6.283185;
    } else if (fI_RESET_I_D3 < 0) {
        fI_RESET_I_D3 += 6.283185;
    }
    out_A = fI_RESET_I_D3;
}
#ifdef _DEBUG
    fGblTheta = fI_RESET_I_D3;
#endif

fABC2 = 2.0/3.0 * (cos(fI_RESET_I_D3) * fFCNM5 + cos(fI_RESET_I_D3-
2*3.14159265/3) * fFCNM8 + cos(fI_RESET_I_D3+2*3.14159265/3) * fFCNM9);
fABC2_1 = 2.0/3.0 * (sin(fI_RESET_I_D3) * fFCNM5 + sin(fI_RESET_I_D3-
2*3.14159265/3) * fFCNM8 + sin(fI_RESET_I_D3+2*3.14159265/3) * fFCNM9);
fABC2_2 = (fFCNM5 + fFCNM8 + fFCNM9) / 3.0;
#ifdef _DEBUG
    fGblId = fABC2;
#endif

#ifdef _DEBUG
    fGblIq = fABC2_1;
#endif

fGblC_BUFFER_12[nGblC_BUFFER_12] = fZOH16; nGblC_BUFFER_12 = (nGblC_BUFFER_12 +
1) % 10;
#ifdef _DEBUG
    fGblVelocidad = fP52;
#endif
fSUM13 = fFreq_ref - fABC2;
fP36 = fSUM13 * 0.01;
fP37 = fSUM13 * 10;
{
    static DefaultType out_A = 0, in_A = 0.0;
    fB15 = out_A + 0.5/10000 * (fP37 + in_A);
}

```

```

        out_A = fB15;in_A = fP37;
    }
    fSUMP18 = fP36 + fB15;
    fLIM10 = (fSUMP18 > 1.5) ? 1.5 : ((fSUMP18 < -1.5) ? -1.5 : fSUMP18);
#ifdef _DEBUG
    fGblVd_c = fLIM10;
#endif
    fP40 = fABC2_1 * (-5);
    fSUM12 = fLIM9 - fP40;
    fP38 = fSUM12 * (-0.0043);
    fP39 = fSUM12 * (-0.45);
    {
        static DefaultType out_A = 0, in_A = 0.0;
        fB17 = out_A + 0.5/10000 * (fP39 + in_A);
        out_A = fB17;in_A = fP39;
    }
    fSUMP19 = fP38 + fB17;
    fLIM3 = (fSUMP19 > 1.5) ? 1.5 : ((fSUMP19 < -1.5) ? -1.5 : fSUMP19);
#ifdef _DEBUG
    fGblVq_c = fLIM3;
#endif
    fDQ02 = cos(fI_RESET_I_D3)*fLIM10+sin(fI_RESET_I_D3)*fLIM3+0;
    fDQ02_1 = cos(fI_RESET_I_D3-2*3.14159265/3)*fLIM10+sin(fI_RESET_I_D3-
2*3.14159265/3)*fLIM3+0;
    fDQ02_2 =
cos(fI_RESET_I_D3+2*3.14159265/3)*fLIM10+sin(fI_RESET_I_D3+2*3.14159265/3)*fLIM3+0;
    fLIM11 = (fDQ02 > 0.99) ? 0.99 : ((fDQ02 < -0.99) ? -0.99 : fDQ02);
    fLIM14 = (fDQ02_1 > 0.99) ? 0.99 : ((fDQ02_1 < -0.99) ? -0.99 : fDQ02_1);
    fLIM15 = (fDQ02_2 > 0.99) ? 0.99 : ((fDQ02_2 < -0.99) ? -0.99 : fDQ02_2);
    PS_SetPwm3ph1Uvw(fLIM11, fLIM14, fLIM15);
#ifdef _DEBUG
    fGblVa_PWM = fLIM11;
#endif
#ifdef _DEBUG
    fGblVb_PWM = fLIM14;
#endif
#ifdef _DEBUG
    fGblVc_PWM = fLIM15;
#endif
#ifdef _DEBUG
    fGblV84 = fSUM15;
#endif
#ifdef _DEBUG
    fGblV85 = fZOH16;
#endif
#ifdef _DEBUG
    fGblV86 = fC_BUFFER_12;
#endif
#ifdef _DEBUG
    fGblCnt = fTI_ENCODER1;
#endif
#ifdef _DEBUG
    fGblIar = fFCNM5;
#endif
#ifdef _DEBUG
    fGblIbr = fFCNM8;
#endif
#ifdef _DEBUG
    fGblIcr = fFCNM9;
#endif
#ifdef _DEBUG
    fGblIa_adc = fTI_ADC1;
#endif
#ifdef _DEBUG
    fGblIb_adc = fTI_ADC1_1;
#endif
#endif

```

```

#ifdef _DEBUG
    fGblIc_adc = fTI_ADC1_2;
#endif
    PS_ExitPwm1General();

if (i<4000)
    {
        grafical1[i]=fGblVelocidad;
        i++;
    }
else
    {
        i = 0;
    }
}

void Initialize(void)
{
    PS_SysInit(30, 10);
    PS_StartStopPwmClock(0);
    PS_InitTimer(0, 0xffffffff);
    PS_InitPwm3ph(1, 1, 10000*1, (4e-6)*1.e6, 2701);
    PS_SetPwm3phPeakOffset(1, 2, (-1), 1.0/2);
    PS_SetPwm3ph1AdcIntr(ePwmIntrAdc0, 1, 0.5);
    PS_SetPwm3ph1Vector(ePwmIntrAdc0, Task);
    PS_SetPwm3ph1Uvw(0, 0, 0);
    PS_StartPwm3ph1();

    PS_ResetAdcConvSeq();
    PS_SetAdcConvSeq(eAdc0Intr, 0, 1.0);
    PS_SetAdcConvSeq(eAdc0Intr, 1, 1.0);
    PS_SetAdcConvSeq(eAdc0Intr, 2, 1.0);
    PS_AdcInit(1, !1);

    PS_InitEncoder(3, 0, 0, 0, 1024);

    PS_StartStopPwmClock(1);
}

void main()
{
    Initialize();
    PS_EnableIntr();
    PS_EnableDbgm();
    for (;;) {
    }
}

```