

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

IMPLEMENTACIÓN DE UNA HERRAMIENTA COMPUTACIONAL QUE SIMULE LA MÁQUINA DE INDUCCIÓN TRIFÁSICA EN EL LENGUAJE DE PROGRAMACIÓN PYTHON

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO ELÉCTRICO**

JUAN JOSÉ LÓPEZ SUÁREZ

juan.lopez03@epn.edu.ec

DIRECTOR: DR. FABIÁN ERNESTO PÉREZ YAULI

fabian.perez@epn.edu.ec

Quito, octubre 2018

AVAL

Certifico que el presente trabajo fue desarrollado por Juan José López Suárez, bajo mi supervisión.

DR. FABIÁN ERNESTO PÉREZ YAULI
DIRECTOR DEL TRABAJO DE TITULACIÓN

DECLARACIÓN DE AUTORÍA

Yo, Juan José López Suárez, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la ley de propiedad intelectual, por su reglamento y por la normativa institucional vigente.

Juan José López Suárez

DEDICATORIA

A mis padres que son la luz de mi vida y me apoyan incondicionalmente en cada paso, por ser parte de mis éxitos y mis fracasos, por nunca tener dudas de mí y hacerme sentir que con ellos y Dios a mi lado puedo conseguir todo lo que me proponga.

AGRADECIMIENTO

A Dios quien con su amor me fortalece llena mi vida de bendición con la vida de mis padres quienes siempre han estado a mi lado.

A mi mami Patricia Suárez y a mi papi Fabricio López por todo el apoyo y el amor que me brindan cada día, por sus concejos, enseñanzas y fe. Sin ustedes no sería la persona que soy ahora. Cada éxito en mi vida es gracias a ustedes. Les amo !

A mi hermano Martin Lopez gracias por estar siempre conmigo siempre apoyándome en cada reto y aconsejándome cuando estoy mal.

A mi novia Eva Serrano, gracias por ser mi compañera de vida, por celebrar conmigo las victorias y darme fuerzas en los momentos difíciles eres una persona maravilla.

A mis profesores Fabian Pérez y Fausto Valencia, por la dedicación y apoyo que me prestaron durante este trabajo.

A mis amigos, con quienes he compartido esta hermosa carrera, sin ustedes nada hubiera sido igual, llenaron mis días de alegrías, risas, bromas las cuales hicieron de la Universidad un lugar mejor.

ÍNDICE DE CONTENIDO

AVAL	I
DECLARACIÓN DE AUTORÍA.....	II
DEDICATORIA.....	III
AGRADECIMIENTO.....	IV
ÍNDICE DE CONTENIDO.....	V
RESUMEN	VII
ABSTRACT	VIII
1. INTRODUCCIÓN.....	1
1.1 Antecedentes	1
1.2 Objetivos	2
Objetivo general.....	2
Objetivos específicos.....	2
1.3 Justificación de Trabajo de Titulación	2
1.4 Alcance	3
2. MODELO MATEMÁTICO DE LA MÁQUINA DE INDUCCIÓN	4
2.1 Introducción.....	4
2.2 Ecuaciones trifásicas de la máquina de inducción	4
2.3 Ecuaciones del torque de la máquina de inducción	10
2.4 Ecuaciones de la máquina de inducción en sistema de referencia arbitrario $qd0$	12
2.4.1 Transformación de sistema de referencia abc al sistema de referencia arbitrario $qd0$	12
2.4.2 Transformación de las ecuaciones de voltaje	14
2.4.3 Transformación de la Ecuación de flujo	16
2.4.4 Transformación de las ecuaciones de torque	19
3. DESARROLLO E IMPLEMENTACIÓN DE UNA HERRAMIENTA COMPUTACIONAL PARA SIMULAR EL MODELO MATEMÁTICO DESCRITO. 20	
3.1 Introducción.....	20
3.1 Lenguaje de programación Python	21
3.1.1 Historia	21
3.1.2 Utilidades de Python	22
3.1.3 Directrices para el uso Python	23

3.2	Desarrollo del programa	25
3.2.1	Estructura y filosofía del programa	25
3.2.2	Etapa de ingreso de datos	25
3.2.3	Etapa modelo	26
3.2.4	Etapa de resultados.....	33
3.2.5	Diagramas de flujo general del programa.....	34
4.	VALIDACIÓN DE LA HERRAMIENTA COMPUTACIONAL IMPLEMENTADA EN PYTHON	35
4.1	Introducción.....	35
4.1	Comparación entre Python y Simulink	35
5.	EJERCICIOS DE APLICACIÓN.....	69
5.1	Introducción.....	69
5.1	Métodos de arranque	69
5.1.1	Arranque directo	70
5.1.2	Arranque estrella delta	74
5.1.3	Arranque con voltaje variable	79
5.1.4	Arranque mediante resistencias en el rotor	83
5.2	Cortocircuito trifásico en terminales	89
5.2.1	Máquina de 3 hp	90
5.2.2	Máquina de 2250 hp	91
6.	CONCLUSIONES Y RECOMENDACIONES.....	93
6.1	Conclusiones.....	93
6.2	Recomendaciones.....	94
	Referencias Bibliográficas	95
	ANEXO I.....	96
	ANEXO II.....	103

RESUMEN

En este documento se realiza el desarrollo e implementación de una herramienta computacional en el software de programación Python, que permite simular el comportamiento en régimen dinámico y estable de una máquina de inducción. El documento comienza con la descripción del modelo matemático de la máquina de inducción para la simulación, a continuación, describe la filosofía con la cual se desarrolla el algoritmo de programación en Python.

Una vez establecido el modelo y el programa se continúa con su validación para lo cual se realiza una comparación con casos disponibles en la bibliografía, simulados con el software Simulink. Para esta sección se realizan varias simulaciones de diferentes máquinas de inducción desde 1hp hasta 2250 hp, comparando las corrientes del rotor, estator, torque eléctrico y la velocidad del rotor. Una vez validada la herramienta computacional, se procede a realizar ejercicios de arranque de máquinas de inducción y cortocircuitos trifásicos en bornes.

Para finalizar el documento se presentan conclusiones y recomendaciones para el uso y trabajos futuros.

Palabras clave: Modelación de máquinas, Ecuaciones diferenciales ordinarias, máquina de inducción, Python.

ABSTRACT

In this document the development and implementation of a computational tool in Python programming software is carried out, which allows to simulate the behavior in dynamic and stable regime of an induction machine. The document begins with the description of the mathematical model of the induction machine. Then it describes the philosophy with which the algorithm of programming in Python is developed. Once the model and the program have been established, validation of the model is continued, for which a comparison is made with the exercises available in the bibliography, simulated with Simulink of Matlab. Several simulations of different induction machines are carried out from 1hp to 2250 hp, comparing rotor currents, stator, electric torque and rotor speed. Once the computational tool has been validated, start-up exercises of induction machines and three-phase short circuits in terminals are carried out. Finally, conclusions and recommendations for the application of the method and future works are presented.

Keywords: algorithm, induction machine, Python.

1. INTRODUCCIÓN

1.1 Antecedentes

En los últimos años, el consumo de energía eléctrica en la industria ha aumentado, desplazando los motores de combustión por máquinas eléctricas. Debido a su bajo costo y mantenimiento, la máquina de inducción se ha convertido en uno de los elementos más importantes en los procesos industriales, por esta razón se espera que, al finalizar esta década, el 70% de los motores en la industria sean motores de inducción [1]. Hoy en día, con la elevada incursión de las energías renovables en la generación de electricidad, no se puede confinar a las máquinas de inducción exclusivamente como motores en la industria, sino que se las debe también considerar como generadores asincrónicos.

A la par de este crecimiento de las máquinas de inducción en la industria y en la generación eléctrica, también se ha producido un avance en el control de la industria y la electrónica de potencia, haciendo que cada vez sea más importante la implementación de un sistema de control para este tipo de máquinas [2]. Si bien es cierto que gran parte de las máquinas de inducción se conectan de forma directa a la red, existe un grupo de éstas que lo hace a través de variadores de frecuencia, para un sin número de aplicaciones que pueden ir desde bombas de agua, ventiladores, hasta robótica y vehículos eléctricos.

Tomando en cuenta lo descrito, es importante tener un programa computacional propio, que resuelva el modelo matemático de la máquina de inducción y permita el acceso a cada uno de sus parámetros, de forma fácil y sin ninguna restricción. De esta manera se puede realizar cualquier tipo de actividad, sea académica y/o de investigación.

De no contarse con un software propio de la Escuela Politécnica Nacional, las investigaciones se verán limitadas a los modelos de programas existentes tanto comerciales, como de acceso libre. Como consecuencia, los investigadores de la Institución tendrían que restringirse a lo que permiten los programas existentes.

En la literatura técnica existen programas que simulan la máquina de inducción como Simulink de Matlab. Sin embargo, debido a la naturaleza comercial del lenguaje de programación, su uso es limitado. De esta manera, se plantea la necesidad de tener una aplicación en un lenguaje de programación libre, que tenga facilidades de computación matemática.

Por lo expuesto, en el trabajo de titulación que se propone, se realizará un programa computacional en el lenguaje de programación Python que simule el modelo estable y dinámico de una máquina de inducción.

1.2 Objetivos

Objetivo general

- Implementar una herramienta computacional que permita simular el comportamiento de la máquina de inducción en estado estable y dinámico, en base a la filosofía de software libre, para ser utilizado en futuras aplicaciones académicas o de investigación.

Objetivos específicos

- Describir un modelo matemático que represente el estado estable y dinámico de la máquina de inducción.
- Implementar, en el lenguaje de programación Python, una herramienta computacional para simular el modelo matemático descrito.
- Validar el algoritmo desarrollado en un sistema de prueba mediante la comparación de los resultados obtenidos, con aquellos encontrados en la literatura técnica existente.
- Analizar diferentes casos de estudio con el fin de determinar el comportamiento de la máquina de inducción, en estado estable y dinámico, al variar sus parámetros de entrada.

1.3 Justificación de Trabajo de Titulación

Una vez identificada la necesidad de tener un programa computacional que permita simular el comportamiento de la máquina de inducción, se encuentra que, aunque se pueden obtener algunos programas ya existentes en el mercado, todos estos presentan una gran cantidad de restricciones al momento de usarlos. Esto se debe a que no se tiene acceso a todas las variables de la máquina de inducción de una forma libre para variarlos, graficarlos, determinar su comportamiento o influencia frente a alguna perturbación; o a que no es posible el acoplamiento de los resultados de un modelo con otros, como, por

ejemplo: variadores de frecuencias, aerogeneradores, etc. En este sentido, cada programa tiene sus propios formatos para ingresar datos.

Adicionalmente, no se puede perder de vista el problema que representan las limitaciones económicas, ya que los costos de una licencia de estos programas comerciales son altos. En consecuencia, en este trabajo de titulación se ha desarrollado un programa computacional que supera estas limitaciones y que, al ser elaborado dentro de la facultad, hace posible su empleo, modificación, desarrollo, en prácticas de laboratorio o en futuras investigaciones. La elaboración de un programa con estas características permitirá realizar una gran cantidad de estudios diferentes, por ejemplo:

- Máquinas eléctricas: Mediante el estudio de la máquina de inducción en forma aislada, se puede determinar el comportamiento de sus parámetros.
- Sistemas eléctricos: Se puede analizar la influencia de la máquina de inducción ya sea en un sistema pequeño como en una industria o en un sistema de potencia, con el ingreso de un parque eólico.
- Eficiencia: Mediante un control adecuado de la máquina de inducción, es posible trabajar en una zona de operación con eficiencia.

Con el objetivo de que el programa computacional pueda ser usado en cada una de estas áreas de forma fácil y rápida, se usará el software de programación Python, uno de los más usados a nivel mundial para resolver problemas de ingeniería [3]. Este es un software libre, de alta potencia y con un gran abanico de posibles aplicaciones.

1.4 Alcance

Como resultado de este trabajo de titulación se presenta un programa computacional, desarrollado en lenguaje Python, que simula el comportamiento estable y dinámico de la máquina de inducción trifásica. El programa permite el acceso y modificación del código fuente. Los resultados de las simulaciones se presentan tanto gráficamente como en forma tabular con la posibilidad de uso en futuras aplicaciones.

El programa tendrá la facultad de simular la máquina de inducción conectada como motor, variando sus parámetros de entrada y las características internas (resistencias, inductancias, velocidad, voltaje de alimentación, etc.). Además, se deberá considerar que el programa se desarrollará en un lenguaje de programación jerárquico, el cual no permite tomar los resultados de la simulación (corrientes, torque eléctrico, etc.) como variables de entrada.

2. MODELO MATEMÁTICO DE LA MÁQUINA DE INDUCCIÓN

2.1 Introducción

En este capítulo se desarrolla la descripción del modelo matemático de Krause, que permite simular el comportamiento de la máquina de inducción trifásica, haciendo uso de la metodología clásica, para la obtención de las ecuaciones de voltaje y torque de la máquina en función de sus parámetros (resistencia, inductancias, constante de inercia, etc.) [4]. Una vez obtenidas las ecuaciones en el sistema de referencia abc, se aplica la transformada qd0 para obtener las ecuaciones en el sistema de referencia qd0. El cual permite desacoplar las inductancias del tiempo, lo que facilita la simulación del modelo. El resultado de este capítulo es plantear un conjunto de ecuaciones a ser implementadas en el software de programación Python. Las ecuaciones descritas en esta sección son tomadas de las referencias [4,5]

2.2 Ecuaciones trifásicas de la máquina de inducción

Para el modelamiento de la máquina de inducción se considera una máquina trifásica de 3 pares de polos, con devanados de estator y rotor idénticos desfasados 120° , con resistencia R_s y R_r respectivamente, como se muestra en la Figura 2.1.

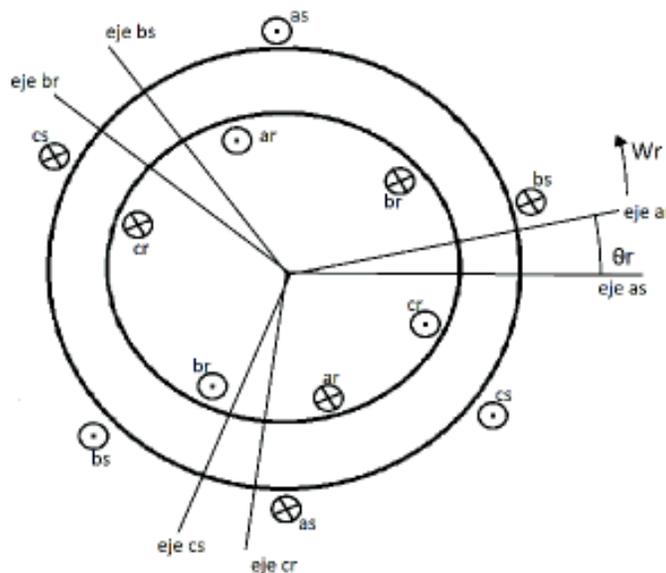


Figura 2.1. Representación gráfica de la máquina de inducción [4]

Es importante para el uso del modelo tener en cuenta las siguientes condiciones:

1. Que los bobinados tanto del estator como del rotor sean iguales.
2. Que el circuito electromagnético sea lineal.
3. Que el material del entrehierro sea uniforme y no presente saturación.

Para el circuito eléctrico se considera que tanto el devanado del rotor como el del estator se encuentran conectados en Y como se muestra en la Figura 2.2. En las siguientes ecuaciones se debe considerar que se utiliza el sufijo “s” para denotar las variables referentes al estator, mientras que el sufijo “r” para representar las variables relacionadas con el rotor.

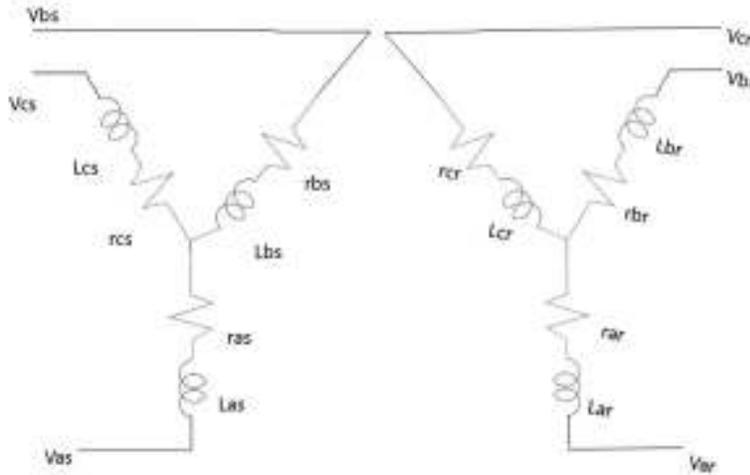


Figura 2.2. Diagrama eléctrico del rotor y estator [4]

Los V_{abcs} del estator y V_{abcr} del rotor pueden ser expresados en función del flujo magnético de la siguiente manera:

$$V_{abcs} = R_s \cdot i_{abcs} + \frac{d\varphi_{abcs}}{dt}$$

$$V_{abcr} = R_r \cdot i_{abcr} + \frac{d\varphi_{abcr}}{dt}$$

Ecuaciones 2.1. Voltajes abc del rotor y estator

Los flujos magnéticos φ_{abcs} y φ_{abcr} se pueden representar en función de la inductancia y la corriente de cada fase.

$$\varphi = L \cdot i_{abc}$$

Ecuaciones 2.2. Ecuación del flujo

Donde:

L : es la matriz de inductancias propias y mutuas del rotor y del estator, ampliando la Ecuación 2.2.:

$$\begin{bmatrix} \varphi_{abc} \\ \varphi_{abc} \end{bmatrix} = \begin{bmatrix} L_s & L_{sr} \\ L_{rs} & L_r \end{bmatrix} \begin{bmatrix} i_{abc} \\ i_{abc} \end{bmatrix}$$

Ecuación 2.3. Matriz de flujos del rotor y del estator

Donde:

L_s : es la inductancia propia de los devanados del estator

L_{sr} : es la inductancia mutua entre los devanados del estator y del rotor

L_{rs} : es la inductancia mutua entre los devanados del rotor y del estator

L_r : es la inductancia propia de los devanados del rotor

La submatriz de inductancia propia del estator se describe en la Ecuación 2.4. Recordando las consideraciones establecidas al comienzo de este capítulo, los devanados del estator tienen una separación de 120° , por lo cual el $\cos(120^\circ)$ es $-\frac{1}{2}$.

$$L_s = \begin{bmatrix} L_{ls} + L_{ms} & -\frac{1}{2}L_{ms} & -\frac{1}{2}L_{ms} \\ -\frac{1}{2}L_{ms} & L_{ls} + L_{ms} & -\frac{1}{2}L_{ms} \\ -\frac{1}{2}L_{ms} & -\frac{1}{2}L_{ms} & L_{ls} + L_{ms} \end{bmatrix}$$

Ecuación 2.4. Matriz de inductancias del estator

Donde:

L_s : es la matriz de inductancia propia de los devanados del estator

L_{ls} : es la inductancia propia de un devanado del estator

L_{ms} : es la inductancia mutua entre los devanados del estator

La submatriz de inductancia propia del rotor con las mismas consideraciones del estator se puede representar como:

$$L_r = \begin{bmatrix} L_{lr} + L_{mr} & -\frac{1}{2}L_{mr} & -\frac{1}{2}L_{mr} \\ -\frac{1}{2}L_{mr} & L_{lr} + L_{mr} & -\frac{1}{2}L_{mr} \\ -\frac{1}{2}L_{mr} & -\frac{1}{2}L_{mr} & L_{lr} + L_{mr} \end{bmatrix}$$

Ecuación 2.5. Matriz de inductancias del rotor

Donde:

L_r : es la matriz de inductancia propia de los devanados del rotor.

L_{sr} : es la inductancia propia de un devanado del rotor

L_{mr} : es la inductancia mutua entre los devanados del rotor.

La submatriz de las inductancias mutuas del rotor con respecto al estator se puede representar como la transpuesta de la matriz de las inductancias del estator con respecto al rotor.

$$L_{rs} = L_{sr}^T$$

Ecuación 2.6. Relación entre las inductancias propias

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

Ecuación 2.7. Matriz de inductancias propias

Donde:

L_{sr} : es el valor pico de la inductancia entre el estator y el rotor.

θ_r : es el valor del ángulo entre los ejes del estator y del rotor.

Con el objetivo de simplificar el modelo de la máquina de inducción resulta conveniente trabajar con las ecuaciones de la máquina de inducción eliminando la relación entre las variables del rotor y las del estator, obteniendo todas estas variables en una sola referencia. Considerando que el estator no se mueve, se suelen referir las variables del rotor con

respecto al estator, usando la relación de espiras N_s y N_r como se describe en las siguientes Ecuaciones 2.8 - 2.10:

$$i'_{abc} = \frac{N_r}{N_s} i_{abc}$$

Ecuación 2.8. Corrientes del rotor referidas al estator

$$V'_{abc} = \frac{N_s}{N_r} V_{abc}$$

Ecuación 2.9. Voltajes del rotor referidas al estator

$$\varphi'_{abc} = \frac{N_s}{N_r} \varphi_{abc}$$

Ecuación 2.10. Flujo del rotor referido al estator

En base a las Ecuaciones [2.11 y 2.12.] se pueden también modificar las ecuaciones para las inductancias, L_{mr} , L_r , L_{sr} en función de una misma variable L_{ms} :

$$L_{ms} = \frac{N_s}{N_r} L_{sr}$$

Ecuación 2.11. Matriz de inductancias mutuas del estator

$$L_{mr} = \left(\frac{N_r}{N_s}\right)^2 L_{ms}$$

Ecuación 2.12. Matriz de inductancias mutuas del rotor

$$L'_{sr} = \frac{N_s}{N_r} L_{sr}$$

Ecuación 2.13. Inductancia propia del rotor referida al estator

$$L'_{sr} = L_{ms} \begin{bmatrix} \cos(\theta_r) & \cos\left(\theta_r + \frac{2\pi}{3}\right) & \cos\left(\theta_r - \frac{2\pi}{3}\right) \\ \cos\left(\theta_r - \frac{2\pi}{3}\right) & \cos(\theta_r) & \cos\left(\theta_r + \frac{2\pi}{3}\right) \\ \cos\left(\theta_r + \frac{2\pi}{3}\right) & \cos\left(\theta_r - \frac{2\pi}{3}\right) & \cos(\theta_r) \end{bmatrix}$$

Ecuación 2.14. Matriz de inductancia propia del rotor referida al estator

Para L_r usando la Ecuación 2.15. en la Ecuación 2.5 se tiene que:

$$L'_r = \left(\frac{N_s}{N_r}\right)^2 L_r$$

Ecuación 2.15. Relación de la inductancia del rotor

$$L'_r = \begin{bmatrix} L_{lr} + L_{ms} & -\frac{1}{2}L_{ms} & -\frac{1}{2}L_{ms} \\ -\frac{1}{2}L_{ms} & L_{lr} + L_{ms} & -\frac{1}{2}L_{ms} \\ -\frac{1}{2}L_{ms} & -\frac{1}{2}L_{ms} & L_{lr} + L_{ms} \end{bmatrix}$$

Ecuación 2.16. Matriz de inductancias del rotor referidas

Reemplazando las Ecuaciones 2.8-16 en la Ecuación 2.3 se obtiene:

$$\begin{bmatrix} \varphi_{abcr} \\ \varphi'_{abcr} \end{bmatrix} = \begin{bmatrix} L_s & L'_{sr} \\ (L'_{sr})^t & L_r \end{bmatrix} \begin{bmatrix} i_{abcr} \\ i'_{abcr} \end{bmatrix}$$

Ecuación 2.17. Matriz de flujos referidos

Se pueden obtener las ecuaciones de voltaje de la máquina de inducción como:

$$\begin{bmatrix} V_{abcr} \\ V'_{abcr} \end{bmatrix} = \begin{bmatrix} r_s + \frac{d}{dt}L_s & \frac{d}{dt}L'_{sr} \\ \frac{d}{dt}(L'_{sr})^t & r'_r + \frac{d}{dt}L'_r \end{bmatrix} \begin{bmatrix} i_{abcr} \\ i'_{abcr} \end{bmatrix}$$

Ecuación 2.18. Matriz de voltajes referidos

2.3 Ecuaciones del torque de la máquina de inducción

En la sección anterior se desarrollaron un grupo de ecuaciones que permiten simular la parte electromagnética de la máquina de inducción en función del flujo presente en los devanados del estator y del rotor, sin embargo, este modelo no está completo ya que se deben acoplar estas ecuaciones eléctricas a las variables mecánicas de la máquina de inducción. El torque es el enlace entre el mundo eléctrico y el mundo mecánico.

Si se asume que el circuito magnético es lineal se puede asumir que la energía del campo W_f es igual a la co-energía W_c . Se debe tener en cuenta que las energías perdidas por fugas de la inductancia no se encuentran consideradas en la Ecuación 2.19. [4].

$$W_f = \frac{1}{2} (i_{abcs})^T (L_s - L_{ls}I) i_{abcs} + (i_{abcs})^T L'_{sr} i'_{abcr} + \frac{1}{2} (i'_{abcr})^T (L'_r - L'_{lr}I) i'_{abcr}$$

Ecuación 2.19. Energía en la máquina

Donde I es la matriz identidad.

En un sistema rotatorio la potencia mecánica no es constante en el tiempo, por lo cual el cambio de la potencia mecánica con respecto al tiempo se puede expresar mediante la siguiente ecuación:

$$\frac{d}{dt} \omega_m = -T_e \frac{d}{dt} \theta_{rm}$$

Ecuación 2.20. Potencia mecánica

Donde:

ω_m : es la velocidad mecánica de la máquina.

T_e : es el torque electromagnético de la máquina de inducción.

θ_{rm} : es el espaciamiento mecánico angular del rotor.

De las ecuaciones anteriores se puede apreciar que la energía W_f y W_c son dependientes de θ_r , mientras que la variación de potencia mecánica es dependiente del ángulo θ_{rm} . La Ecuación 2.21. permite relacionar estas dos variables de tal forma que se pueda expresar todo el sistema en función de θ_r [5].

$$\theta_r = \theta_{rm} \left(\frac{P}{2} \right)$$

Ecuación 2.21. Ángulo del rotor

Reemplazando la Ecuación 2.21. en la Ecuación 2.20., se tiene:

$$\frac{d}{dt} \omega_m = -T_e \left(\frac{2}{P} \right) \frac{d}{dt} \theta_r$$

Ecuación 2.22. Potencia mecánica en función del ángulo del rotor

Donde:

P: es el número de par de polos de la máquina de inducción.

Para finalizar la descripción de la parte mecánica de una máquina de inducción, es necesario establecer una ecuación que permita determinar el torque eléctrico:

$$T_e = \left(\frac{2}{P} \right) (i_{abcs})^T \frac{\partial}{\partial \theta_r} [L'_{sr}] i'_{abcr}$$

Ecuación 2.23. Torque eléctrico

Reemplazando las variables de la Ecuación 2.23 y realizando operaciones matemáticas se puede simplificar la expresión a:

$$T_e = J \left(\frac{2}{P} \right) \frac{d}{dt} \omega_r + T_L$$

Ecuación 2.24. Torque eléctrico en función de ω_r

Donde:

ω_r : es la velocidad del rotor.

J : es la inercia del motor expresada en kilogramo-metro cuadrado [$k_g m^2$].

T_L : es el troque de la carga en el eje de la máquina en Newton metro Ecuación [Nm].

2.4 Ecuaciones de la máquina de inducción en sistema de referencia arbitrario qd0.

2.4.1 Transformación de sistema de referencia abc al sistema de referencia arbitrario qd0.

En la sección 2.2 se establecieron las ecuaciones de voltaje para la máquina de inducción en función de sus variables propias. Se puede notar que las ecuaciones son dependientes del flujo mutuo entre el estator y el rotor, los cuales son dependientes del ángulo del rotor θ_r cómo se observa en la Ecuación 2. 14. De la Ecuación 2.25. se puede apreciar que el ángulo del rotor varía con respecto al tiempo [5].

$$\frac{d}{dt}\theta_r = \omega_r$$

Ecuación 2.25. Ángulo del rotor

Debido a que el ángulo del rotor θ_r es dependiente del tiempo, las inductancias de la Ecuación 2.14. también son dependientes del tiempo, lo cual obliga a que sean recalculadas en cada iteración del programa, aumentando los recursos computacionales necesarios para la resolución del modelo; es por esto que se considera conveniente para la resolución del modelo, referir todas las ecuaciones a un sistema de referencia arbitrario el cual permite desacoplar los valores de inductancia del tiempo. El sistema de referencia seleccionado es comúnmente conocido como qd0 [6].

Para la obtención de las ecuaciones en el sistema de referencia arbitrario qd0, se deben multiplicar las ecuaciones de la máquina de inducción en el sistema de referencias abc por una matriz de conversión dependiente del ángulo del rotor θ_r , que se observa en la Ecuación 2.26.

$$\begin{bmatrix} f_q \\ f_d \\ f_0 \end{bmatrix} = [T_{qdo}(\theta)] \begin{bmatrix} f_a \\ f_b \\ f_c \end{bmatrix}$$

Ecuación 2.26. Transformación qd0

Donde f_a, f_b, f_c son las ecuaciones de voltaje y corriente establecidas en la sección anterior, en el sistema de referencia abc. La matriz de transformación $[T_{qdo}(\theta)]$ se define como:

$$[T_{qdo}(\theta)] = \frac{2}{3} \begin{bmatrix} \cos(\theta) & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\ \sin(\theta) & \sin\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

Ecuación 2.27. Matriz de transformación qd0

La Ecuación 2.27. se puede aplicar de forma directa para las variables del estator mientras que, para las variables del rotor, se debe considerar la diferencia angular entre el ángulo teta y el ángulo del rotor:

$$\beta = \theta - \theta_r$$

Ecuación 2.28. Relación de ángulos para el rotor

Remplazando la Ecuación 2.28. en la Ecuación 2.27., se tiene:

$$[T_{qdo}(\theta)] = \frac{2}{3} \begin{bmatrix} \cos(\beta) & \cos\left(\beta - \frac{2\pi}{3}\right) & \cos\left(\beta + \frac{2\pi}{3}\right) \\ \sin(\beta) & \sin\left(\beta - \frac{2\pi}{3}\right) & \sin\left(\beta + \frac{2\pi}{3}\right) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

Ecuación 2.29. Matriz de transformación qd0 para el rotor

La Ecuación 2.29. se debe usar para las variables del rotor.

2.4.2 Transformación de las ecuaciones de voltaje

Aplicando la matriz de transformación de la Ecuación 2.27. en la Ecuación 2.1. de voltaje de la máquina de inducción en forma matricial, se tiene que:

$$V_{abcs} = R_s \cdot i_{abcs} + \frac{d\varphi_{abcs}}{dt}$$

Ecuación 2.1. Voltajes abc del estator

$$[T_{qdo}(\theta)]V_{qd0s} = [R_s \cdot i_{qd0s} + \frac{d\varphi_{qd0s}}{dt}] [T_{qdo}(\theta)]$$

Ecuación 2.30. Voltajes dq0 del estator

Despejando el voltaje de la Ecuación 2.30. anterior se tiene la Ecuación 2.31:

$$V_{qd0s} = [T_{qdo}(\theta)]^{-1} R_s [T_{qdo}(\theta)] i_{qd0s} + \frac{d\varphi_{qd0s} [T_{qdo}(\theta)]^{-1}}{dt} [T_{qdo}(\theta)]$$

Ecuación 2.31. Voltajes qd0 del estator

Donde,

$$[T_{qdo}(\theta)]^{-1} [T_{qdo}(\theta)] = I$$

Ecuación 2.32. Propiedad identidad de la matriz qd0

$$\frac{d\varphi_{qdo} [T_{qdo}(\theta)]^{-1}}{dt} = \frac{d}{dt} \theta \begin{bmatrix} -\sin(\theta) & -\cos(\theta) & 0 \\ -\sin\left(\theta - \frac{2\pi}{3}\right) & -\cos\left(\theta - \frac{2\pi}{3}\right) & 0 \\ -\sin\left(\theta + \frac{2\pi}{3}\right) & -\cos\left(\theta + \frac{2\pi}{3}\right) & 0 \end{bmatrix} \varphi_{qdo} + [T_{qdo}(\theta)]^{-1} \frac{d}{dt} \varphi_{qd0s}$$

Ecuación 2.33. Derivada del flujo por la inversa de la matriz qd0

$$\begin{bmatrix} -\sin(\theta) & -\cos(\theta) & 0 \\ -\sin\left(\theta - \frac{2\pi}{3}\right) & -\cos\left(\theta - \frac{2\pi}{3}\right) & 0 \\ -\sin\left(\theta + \frac{2\pi}{3}\right) & -\cos\left(\theta + \frac{2\pi}{3}\right) & 0 \end{bmatrix} [T_{qd0}(\theta)] = A = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Ecuación 2.34. Matriz resultante

$$\omega = \frac{d}{dt}\theta$$

Ecuación 2.35. Velocidad del rotor

Sustituyendo las Ecuaciones 2.31-35. en la Ecuación 2.30., se expresan los voltajes en forma matricial como:

$$V_{qd0s} = R_s \cdot i_{qd0s} + \omega A \varphi_{qd0s} + \frac{d\varphi_{qd0s}}{dt}$$

Ecuación 2.36. Voltajes qd0 del estator

Usando la misma metodología anterior se pueden deducir de igual manera las ecuaciones para las variables del rotor, tal y como se muestran en la Ecuación 2.37.

$$V_{qd0r} = R_r \cdot i_{qd0r} + (\omega - \omega_r) A \varphi_{qd0r} + \frac{d\varphi_{qd0r}}{dt}$$

Ecuación 2.37. Voltajes qd0 del rotor

2.4.3 Transformación de la ecuación de flujo

Aplicando la matriz de transformación de la Ecuación 2.27. en la Ecuación 2.38. del flujo de la máquina de inducción en forma matricial se tiene que:

$$\varphi_{abcs} = L_{abcs} \cdot i_{abcs} + L_{abcsr} \cdot i_{abcs}$$

Ecuación 2.38. Flujo abc del estator

$$\varphi_{qd0s} = [L_{abcs} i_{abcs} + L_{abcsr} i_{abcs}] \cdot [T_{qd0}(\theta)]$$

Ecuación 2.39. Flujo qd0 del estator

En la Ecuación 2.39. anterior se remplazan las corrientes i_{abcs}, i_{abcr} con sus respectivas transformaciones en el sistema de referencia qd0, entonces:

$$i_{abcs} = [T_{qd0}(\theta)]^{-1} i_{qd0s}$$

$$i_{abcr} = [T_{qd0}(\theta - \theta_r)]^{-1} i_{qd0r}$$

Ecuación 2.40. Corrientes qd0 del estator y rotor

Remplazando las Ecuaciones 2.40. en la Ecuación 2.39. Se tiene:

$$\varphi_{qd0s} = [L_{abcs} [T_{qd0}(\theta)]^{-1} i_{qd0s} + L_{abcsr} [T_{qd0}(\theta - \theta_r)]^{-1} i_{qd0r}] [T_{qd0}(\theta)]$$

Ecuación 2.41. Flujo qd0 del estator

Donde,

$$[T_{qd0}(\theta)] [L_{abcs}] [T_{qd0}(\theta)]^{-1} = \begin{bmatrix} L_{ls} + \frac{3}{2} L_{ms} & 0 & 0 \\ 0 & L_{ls} + \frac{3}{2} L_{ms} & 0 \\ 0 & 0 & L_{ls} \end{bmatrix}$$

Ecuación 2.42. Inductancias del estator por la matriz qd0

$$[T_{qd0}(\theta)] [L_{abcsr}] [T_{qd0}(\theta - \theta_r)]^{-1} = \begin{bmatrix} \frac{3}{2}L_{sr} & 0 & 0 \\ 0 & \frac{3}{2}L_{sr} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Ecuación 2.43. Inductancias del rotor por la matriz qd0

Reemplazando las Ecuaciones 2.42-43. en la Ecuación 2.41. se obtiene la Ecuación 2.44.

$$\varphi_{qd0s} = \begin{bmatrix} L_{ls} + \frac{3}{2}L_{ms} & 0 & 0 \\ 0 & L_{ls} + \frac{3}{2}L_{ms} & 0 \\ 0 & 0 & L_{ls} \end{bmatrix} i_{qd0s} + \begin{bmatrix} \frac{3}{2}L_{sr} & 0 & 0 \\ 0 & \frac{3}{2}L_{sr} & 0 \\ 0 & 0 & 0 \end{bmatrix} i_{qd0r}$$

Ecuación 2.44. Flujo qd0 del estator

De manera similar a las ecuaciones de flujo para el estator, se pueden hallar las ecuaciones de flujo para el rotor:

$$\varphi_{qd0r} = \begin{bmatrix} L_{lr} + \frac{3}{2}L_{mr} & 0 & 0 \\ 0 & L_{lr} + \frac{3}{2}L_{mr} & 0 \\ 0 & 0 & L_{lr} \end{bmatrix} i_{qd0r} + \begin{bmatrix} \frac{3}{2}L_{sr} & 0 & 0 \\ 0 & \frac{3}{2}L_{sr} & 0 \\ 0 & 0 & 0 \end{bmatrix} i_{qd0s}$$

Ecuación 2.45. Flujo qd0 del rotor

Considerando que:

$$L_m = \frac{3}{2}L_{ls} = \frac{3}{2}L_{sr} = \frac{3}{2}L_{lr}$$

Ecuación 2.46. Relaciones de inductancias

Se pueden representar las Ecuaciones 2.44.-2.46.:

$$\varphi_{qd0s} = \begin{bmatrix} L_{ls} + L_m & 0 & 0 \\ 0 & L_{ls} + L_m & 0 \\ 0 & 0 & L_{ls} \end{bmatrix} i_{qd0s} + \begin{bmatrix} L_m & 0 & 0 \\ 0 & L_m & 0 \\ 0 & 0 & 0 \end{bmatrix} i_{qd0r}$$

$$\varphi_{qd0r} = \begin{bmatrix} L_{lr} + L_m & 0 & 0 \\ 0 & L_{lr} + L_m & 0 \\ 0 & 0 & L_{lr} \end{bmatrix} i_{qd0r} + \begin{bmatrix} L_m & 0 & 0 \\ 0 & L_m & 0 \\ 0 & 0 & 0 \end{bmatrix} i_{qd0s}$$

Ecuación 2.47. Flujo qd0 del estator y rotor

2.4.4 Transformación de las ecuaciones de torque

Para obtener una expresión que permita determinar el torque electromagnético en el sistema de referencia arbitrario qd0, se repite el proceso aplicándolo a las ecuaciones de voltaje y de flujo, en la Ecuación 2.48. del torque:

$$T_e = \left(\frac{2}{P}\right) (i_{abcs})^T \frac{\partial}{\partial \theta_r} [L'_{sr}] i'_{abcr}$$

Ecuación 2.48. Torque eléctrico de la máquina de inducción

Si se aplica el conjunto de Ecuaciones 2.49. en la ecuación anterior:

$$i_{abcs} = [T_{qd0}(\theta)]^{-1} i_{qd0s}$$

$$i_{abcr} = [T_{qd0}(\theta - \theta_r)]^{-1} i_{qd0r}$$

Ecuación 2.49. Corrientes qd0 del estator y rotor

Se tiene que:

$$T_e = \left(\frac{2}{P}\right) \left([T_{qd0}(\theta)]^{-1} i_{qd0s}\right)^T \frac{\partial}{\partial \theta_r} [L'_{sr}] [T_{qd0}(\theta - \theta_r)]^{-1} i_{qd0r}$$

Ecuación 2.50. Torque eléctrico en función de corrientes qd0

Trabajando la Ecuación 2.50. y dejándola en función de las corrientes, se puede expresarla de la siguiente manera:

$$T_e = \left(\frac{3}{2}\right) \left(\frac{P}{2}\right) L_m (i_{qs} i_{dr} - i_{ds} i_{qr})$$

Ecuación 2.51. Torque eléctrico en función de corrientes qd0

Debido a que en el caso de la simulación de la máquina de inducción resulta conveniente trabajar con las ecuaciones de flujo, se puede expresar la Ecuación 2.51. del torque electromagnético de la siguiente forma:

$$T_e = \left(\frac{3}{2}\right) \left(\frac{P}{2}\right) \left(\frac{1}{\omega_b}\right) (\varphi_{qr} i_{dr} - \varphi_{dr} i_{qr})$$

Ecuación 2.52. Torque eléctrico en función de corrientes qd0

ω_b : es la velocidad base de la máquina.

3. DESARROLLO E IMPLEMENTACIÓN DE UNA HERRAMIENTA COMPUTACIONAL PARA SIMULAR EL MODELO MATEMÁTICO DESCRITO.

3.1 Introducción

En este capítulo se describe el trabajo realizado con la finalidad de implementar una herramienta computacional en el lenguaje de programación Python, que permita simular el comportamiento de la máquina de inducción trifásica funcionando como motor. A manera de antecedente se desarrolla la historia del lenguaje de programación Python; posteriormente se describen las razones por las cuales se empleó dicho lenguaje en la presente investigación; se resaltan las cualidades de Python y se describe su incursión en el mercado. Finalmente, se desarrolla un acercamiento al lenguaje mediante la explicación de puntos claves en la programación en Python.

Terminada la introducción a Python, se incursiona en la filosofía de programación que se utilizó para resolver el problema de la simulación en cada una de las etapas y funciones. Se comienza con la descripción de la etapa en forma de pseudo código, seguido del diagrama de flujo de la etapa y su respectivo código el cual se incluye en el ANEXO 1 del presente trabajo.

Se debe tener en cuenta que la filosofía de programación implementada es modular lo cual permite cambiar una parte del código sin afectar el resto, esto responde a la idea de que este programa debe tener la capacidad de adaptarse fácilmente a trabajos futuros. Además se debe considerar las limitaciones que un lenguaje de programación jerárquico tiene en cuanto al manejo de variables.

3.1 Lenguaje de programación Python

3.1.1 Historia

El lenguaje de programación Python nació como un proyecto de Guido Van Rossum, programador holandés, quien buscó desarrollar un lenguaje de programación de alto nivel e intuitivo, que tenga la misma capacidad de los lenguajes de programación comerciales, pero que no consuma muchos recursos computacionales. El proyecto comenzó a finales de los años ochenta, pero debido al estado de la tecnología de la época, especialmente por las limitaciones de hardware, no fue hasta los años noventa que el proyecto se implementó [3].

La filosofía del lenguaje de programación de Python se basa en otro lenguaje denominado ABC, este fue desarrollado a principios de los años ochenta por el centro Wiskunde & Informatic, ubicado en Países Bajos, proyecto en el cual también trabajó Guido Van Rossum, fundador de Python. El objetivo fue desarrollar un lenguaje de programación que sea:

- Interactivo
- Estructurado
- De alto nivel
- Intuitivo

Además, que sirva como sustituto de los lenguajes de programación convencionales como BASIC, C, Pascal, los cuales consumen cuatro veces más recursos que este [7]. Aunque en sus orígenes se diseñó para realizar interfaces gráficas, hoy en día, gracias a que se mantiene en software libre y con el aporte de varios programadores, el lenguaje de programación Python se usa para [7]:

- Programación de aplicaciones
- Manejo de redes
- Tecnologías de la comunicación
- Programación estructurada
- Programación orientada a objetos
- Desarrollo de investigaciones
- Simulación
- Etc.

Python se encuentra disponible en los sistemas operativos Windows, Unix, iOS, Atari y Apple Macintosh en sus tres versiones disponibles.

3.1.2 Utilidades de Python

En los últimos años, los programadores profesionales, así como los diseñadores de bases de datos e investigadores académicos, han comenzado a cambiar de los lenguajes de programación tradicionales y comerciales como C/C++, Java al lenguaje de programación Python. Esto debido a que Python cuenta con varios intérpretes, así como bases de datos propias y de terceros, que son de libre acceso y de fácil distribución, las cuales permiten realizar las mismas aplicaciones que otros lenguajes, apartando las limitaciones que un lenguaje comercial puede llegar a tener [6].

Python es un lenguaje muy utilizado por las empresas, principalmente para elaboración de bases de datos y minería de datos. La academia no se ha quedado atrás al momento de implementar esta herramienta en sus espacios de estudio e investigación, esto se debe a que gran parte de los programas comerciales disponibles para estas actividades representan un costo para las instituciones educativas; por su parte Python, al ser un software libre, no tiene un costo directo asociado a su uso y aplicaciones. Aunque ésta suena como una gran ventaja, no es la razón principal por la cual muchas áreas académicas están adoptando este lenguaje [3].

La razón principal por la cual los académicos han decidido optar por un lenguaje como Python, son las más de trecientas librerías que el lenguaje posee al momento de la instalación y que permiten resolver un mismo problema mediante un modelo multiparadigma, lo cual quiere decir que se puede resolver usando diferentes librerías o filosofías de programación. Python no solo brinda una amplia variedad de librerías, sino también, la posibilidad de acceder al código fuente y modificarlo de tal forma que éstas queden personalizadas [7].

Uno de los grandes retos que se encuentran cuando se decide migrar de un lenguaje comercial al lenguaje Python, es la sintaxis de los comandos. Python es un lenguaje de alto nivel, intuitivo y multiparadigma, tal como se mencionó anteriormente; esto permite mantener el estilo propio de programación aprendido en otros lenguajes y aplicarlo a Python. Si esto no fuese motivo suficiente para decidir migrar al lenguaje, Python brinda un módulo de intérprete Python/c que permite programar el lenguaje C y correr el código en el intérprete de Python. Lo cual es de utilidad ya que Python al consumir menos recursos computacionales, no se vuelve lento con la incorporación de librerías al código, como lo hacen C y Java.

Para este trabajo se ha optado por usar el lenguaje de programación Python, principalmente como una alternativa al software de programación Matlab, el cual no es libre

y el acceso a este software (adquirir una licencia) implica un costo. Estas representan las ventajas económicas de usar Python, por otro lado, las ventajas técnicas que brinda este lenguaje de programación son:

- Librerías matemáticas que permiten resolver sistemas de ecuaciones diferenciales y operaciones simples.
- El compilador del código que facilita encontrar errores en la programación línea por línea, incluso dentro de las librerías utilizadas.
- Una amplia gama de GUIDes para realizar una codificación más amigable, sin embargo, el código se puede ejecutar en un comand window o su equivalente en otro sistema operativo, ya que el lenguaje es multiplataforma.

Una de las dificultades encontradas al momento de usar Python es la limitada bibliografía técnica existente sobre el mismo, lo cual implica un mayor esfuerzo para resolver problemas o desarrollar códigos de alta complejidad, como el que se expondrá en este trabajo.

La principal cualidad de Python, por la cual se le ha seleccionado para la elaboración de este trabajo, es la velocidad con la cual se pueden obtener los resultados de cada una de las variables involucradas en el código, al igual que una gran facilidad para acoplar los resultados obtenidos en el programa con otros programas, no solo los desarrollados en Python sino también los que se desarrollan en programas como Matlab.

3.1.3 Directrices para el uso Python

En esta sección del capítulo, se desarrollan las directrices básicas que permiten comenzar a programar en el lenguaje de programación Python. Debido a la formación académica en la Escuela Politécnica Nacional, los estudiantes están familiarizados con la programación en C y Matlab, lo cual será de utilidad al momento de establecer comparaciones con Python.

A pesar de que Python cuenta con varias librerías incorporadas es necesario importar la librería a ser utilizada dentro del código. Existen varias formas de realizar este paso, sin embargo, es conveniente importar la librería completa y darle un seudónimo como se muestra a continuación:

```
import numpy as np
```

```
import scipy as sp
import math as mth
import matplotlib.pyplot as plt
from scipy.integrate import odeint
```

Esto servirá porque cada vez que se llama a una función dentro de una de estas librerías, es necesario escribir el nombre de la librería con un punto al final y el nombre de la función a ser utilizada, tal como se muestra en el siguiente ejemplo:

```
Va= V*np.cos(t*we)
```

Cuando solo se utilizará una función de una librería, se recomienda solo llamar la dicha función en vez de hacer un llamado a toda la librería, esto permite, en códigos extensos, que el tiempo de simulación se reduzca:

```
from scipy.integrate import odeint
```

Las funciones, permiten realizar procesos repetitivos dentro de un código de una forma sencilla, toda función cuenta con tres elementos básicos los cuales se deben definir, estos son:

- Variables de entrada
- Proceso
- Variables de salida o resultados

En el lenguaje de programación Python, es preferible que estas funciones estén definidas al inicio del código, a diferencia de C en el cual se las deberá definir fuera del main. La sintaxis para la creación de funciones de Python es la siguiente:

```
Def nombre_de_la_funcion (variables_de_entrada):
```

```
    proceso
```

```
    Return variables_de_salida_o_resultados
```

3.2 Desarrollo del programa

3.2.1 Estructura y filosofía del programa

En este trabajo se ha planteado desarrollar una herramienta computacional que permita simular el comportamiento de una máquina de inducción; para el desarrollo del simulador se ha estructurado el código en tres etapas independientes. Una primera etapa, que permite realizar la adquisición de datos, a la cual se denomina “ingreso de datos”, la segunda etapa, en la cual se resuelve el modelo matemático de la máquina de inducción a la cual se conocerá como “modelo” y, la tercera y última etapa, en la cual se presentarán los resultados ya sea de forma gráfica o en matrices, a esta etapa final se llamará “resultados”. En la Figura 3.1 se puede apreciar gráficamente la estructura del simulador mediante un diagrama de bloques.

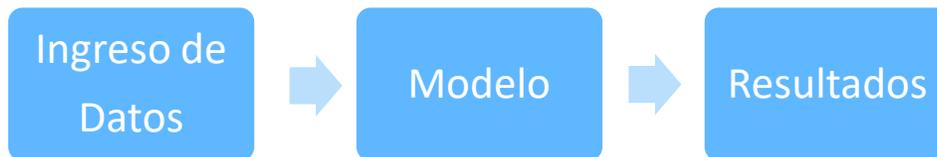


Figura 3.1. Filosofía del programa

Como se aprecia en el diagrama de bloques de la Fig.3.1, se ha optado por una programación modular, de tal forma que cada una de las etapas sea independiente de la otra, teniendo en cuenta que éstas reciben valores de entrada ya sea del usuario o de la etapa anterior. El código es independiente de las demás etapas, de esta manera los cambios realizados en cada una de estas etapas no serán de mayor influencia en las otras.

Esta filosofía de programación permite realizar simulaciones por ciclos o en partes, por ejemplo, se pueden variar los valores de voltaje de la máquina, torque mecánico, entre un tiempo $t1$ y $t2$, y en un tiempo $t3$ se restablece el valor original como se observa en el GUI del ANEXO 2.

3.2.2 Etapa de ingreso de datos

En esta etapa se realiza el ingreso de los valores característicos de la máquina de inducción y la frecuencia de la red, también se establecen los valores de frecuencia base del sistema y, cualquier otro valor que el usuario considere necesario añadir de acuerdo con las modificaciones que realice en el programa.

Para los valores de resistencia del estator, se debe considerar que el modelo de la máquina de inducción está usando valores reales en ohmios, para la inductancia del estator se debe tomar la misma consideración.

Para los valores de resistencia e inductancia del rotor se debe considerar:

- Valores referidos al rotor
- Valores reales en ohmios

Es importante tener en cuenta estos valores ya que fueron considerados al momento de definir el modelo matemático que se resolverá en la etapa siguiente. Mientras los valores de las variables se encuentren en coherencia con los usados en el modelo, no importa cómo decida el usuario obtener los mismos, ya sea ingresarlos manualmente mediante adquisición por teclado, mediante un GUI (ANEXO 2) o de un documento de texto.

Esta etapa requiere del ingreso de la siguiente información:

- i. Importación de librerías
- ii. Datos de la máquina de inducción
 - Voltajes
 - Frecuencia
 - Potencia en W .
 - Constante de inercia en $kg\ m^2$.
 - Número de polos
 - Resistencia e inductancia del rotor [ohm]
 - Resistencia e inductancia del estator [ohm]
 - Inductancia mutua [ohm]
- iii. Datos de la red
 - Frecuencia de la red en Hz
- iv. Datos para el modelo matemático
 - Frecuencia eléctrica base
 - Frecuencia mecánica base
 - Valor del torque

3.2.3 Etapa modelo

En esta etapa del programa se definen las ecuaciones de voltaje que permiten realizar la simulación de la máquina de inducción. Para la resolución de estas ecuaciones se usa la función `odeint`, existente en las librerías de Python. Como se observa en la Figura 3.2.

- Definición del modelo

En esta función se realiza la definición de las ecuaciones diferenciales que se resuelven para simular el comportamiento de la máquina de inducción (ver el ANEXO 1), Estas ecuaciones se encuentran en función del flujo que circula en los devanados de la máquina, el torque electromagnético de la máquina de inducción y la velocidad del rotor.

Las ecuaciones de voltaje para ser utilizadas en el modelo de la máquina de inducción deberán ser previamente llevadas a un sistema de referencia arbitrario $qd0$. Para esto se ha decidido realizar una simulación estacionaria de la máquina en la cual $w=0$ y el ángulo $\theta = 0$.



Figura 3.2. Diagrama de flujo de la etapa de ingreso de datos

El modelo se debe definir en una función propia, la cual será luego un parámetro de entrada para la función odeint. Las entradas para esta función son el tiempo en el cual se realiza la simulación y un vector con las variables que se desean encontrar, en este caso son los flujos magnéticos de la máquina. Donde f_s es el flujo del eje q de estator, f_{dr} será el flujo del eje d del rotor y así respectivamente. i_0s e i_0r son las corrientes de secuencia cero tanto para el estator como para el rotor, w_x es la velocidad del rotor y $teta$ es el ángulo entre el sistema de referencia abc y qd0.

```
def MaqInd (I,t):
# Vector de variables
    fqs,fds,i0s,fqr,fdr,i0r,wx,teta = I
```

Debido a que la función odeint resuelve el sistema de ecuaciones mediante el uso de métodos numéricos, es decir, tomando valores punto a punto para cada tiempo, es importante definir en esta parte del programa las ecuaciones de voltaje con las cuales se alimenta a la máquina de inducción. De no hacerlo el programa no podrá resolver el sistema.

Las ecuaciones de voltaje deberán ser previamente llevadas a un sistema de referencia arbitrario qd0. Para esto se ha decidido realizar una simulación estacionaria de la máquina en la cual $w = 0$ y el ángulo $\theta = 0$. En el código se representa el voltaje con la letra "V", seguido del sistema de referencia y su pertenencia al rotor o al estator.

```
# Transformación de voltajes ABC a qd0 del estator
Va= V*np.cos(t*we)
Vb= V*np.cos(t*we - 2*mth.pi/3 )
Vc= V*np.cos(t*we - 4*mth.pi/3 )
Vqs=(2/3)*(Va-(Vb+Vc)/2)
Vds=(Vc - Vb)/np.sqrt(3)
V0s=(Va + Vc + Vb)*(1/3)
```

Existen dos posibilidades para el planteamiento del modelo matemático. La primera es calcular las corrientes de la máquina de inducción de forma directa. Debido a que estas ecuaciones son dependientes una de otra, esto implica la solución de un modelo matemático complejo. La segunda opción es obtener los flujos electromagnéticos y posteriormente calcular las corrientes de la máquina. Esta solución resulta práctica cuando se trabaja con un lenguaje de programación jerárquico.

El sistema de ecuaciones planteado resuelve 8 ecuaciones diferenciales: 3 ecuaciones de flujos (qd0) del rotor, 3 ecuaciones de flujos (qd0) de estator, una ecuación para la velocidad del rotor y una ecuación que calcule el ángulo teta. Es necesario para resolver el

sistema se deben realizar cálculos adicionales, como flujos mutuos para las ecuaciones de flujo y corrientes para el cálculo del torque eléctrico, el cual es usado para encontrar la velocidad del rotor.

```
# Cálculo de ecuaciones de flujo mutuo (fmq,fmd) y corrientes dq del
estator (iqs,ids)
```

```
fmq=XM*(fqs/Xls + fqr/Xlr)
fmd=XM*(fds/Xls + fdr/Xlr)
iqs = (fqs - fmq)/Xls
ids = (fds - fmd)/Xls
```

- Resolución del modelo

En esta etapa se implementó una función, la cual tiene como nombre “Integrar”. Esta función realiza tres procesos importantes dentro del programa, el llamado de la función odeint, el acondicionamiento de resultados y la actualización del vector de condiciones iniciales como se observa en la Figura 3.3.

```
def Integrar(MaqInd,I0,tiempoi,tiempof,paso):
```

Para el uso de la función odeint se debe tener previamente establecido el sistema de ecuación a ser resuelto, como se ha realizado en la etapa anterior. Adicional a esto, la función odeint requiere del tiempo de simulación y del valor de la primera derivada evaluada al inicio del tiempo de simulación (condiciones iniciales), para cada una de las ecuaciones diferenciales.

El tiempo de la simulación es un vector en el cual se encuentran los valores de tiempo en los que se evaluará la función, esto se realiza mediante la función linspace.

```
##### Definición de Vector del Tiempo
### Número de iteraciones
itr= (tiempof - tiempoi)/paso

### Vector del tiempo
t1 = np.linspace(tiempoi,tiempof,itr)
```

El paso de integración para el método numérico se ingresa de forma indirecta en el vector de tiempo. Esto se consigue calculando el número de elementos del vector, como se muestra en el cálculo *itr*.

Una vez que se ha definido el sistema de ecuaciones, el tiempo y el vector de condiciones iniciales, se debe llamar a la función odeint, este paso se lo realiza de la siguiente forma:

```
# Llamado de la función odeint
solu=odeint(MaqInd, I0,t1,full_output = 0)
```

Es necesario presentar los resultados en una sola variable, a lo que se denomina acondicionamiento de datos. Esto se lo realiza mediante una función denominada “Concatenar”, la cual toma los resultados de la función odeint y los incluye dentro de la variable correspondiente, con la finalidad de obtener una sola variable por parámetro de la máquina.

Concatenar tiene como entrada dos variables, el vector de la variable correspondiente a la máquina de inducción y la respuesta de la función odeint que se desea incorporar. Para realizar este paso, se debe cambiar el formato de las variables de numérico a texto. Luego se usa la función extend para adjuntar los nuevos valores al vector de la variable de la máquina, finalmente se regresa al formato numérico original de las variables.

```
### Función de concatenación de variables ###  
  
def Concatenar(X1,X2):  
    X1= list(X1)  
    X2= list(X2)  
    X1.extend(X2)  
    Y=np.array(X1)  
    return Y
```

Ejemplo:

In[1]

```
A= [1,2,3,4,5,6]  
B= [7,8,9,10]  
C= Concatenar(A,B)
```

Out [1]

```
C = array([ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10])
```

Una vez implementado todo esto dentro del lenguaje de programación Python, se obtiene el siguiente código:

```
# Llamado de la función odeint  
solu=odeint(MaqInd, I0,t1,full_output = 0)  
  
# Asignación de variables  
[fqs1,fdsl,i0s1,fqr1,fdrl,i0r1,wx1,tetal] =  
[solu[:,0],solu[:,1],solu[:,2],solu[:,3],solu[:,4],solu[:,5],solu[:,6]]  
Concatenación de los resultados del estado anterior con los actuales  
fqs2 = Concatenar(fqs,fqs1)  
fds2 = Concatenar(fds,fdsl)
```

La variable “solu” guarda los valores de los flujos de la máquina de inducción ($fqs1$, $fds1$, $i0s1$, $fqr1$, $fdr1$), las corrientes de secuencia cero ($i0r1$), la velocidad del rotor ($wx1$) y la velocidad angular ($teta1$). Estos datos deben extraerse en su variable correspondiente. Luego se acondicionan los datos, para la extracción se usa el sufijo 1, mientras que para la nueva variable se usa el sufijo 2. Esto se realiza de esta manera ya que todas las variables globales que se pretende usar dentro de una función deben ser definidas como entradas.

Finalmente se actualiza el vector de condiciones iniciales, esto se hace con los últimos valores de cada uno de los parámetros de la máquina. Para esto se toma la longitud del tiempo y se le resta la unidad, ya que Python enumera los elementos de un vector desde cero.

```
# Obtención del vector las variables de estado final
i = len(t1)-1
I0 = [fqs1[i],fds1[i],i0s1[i],fqr1[i],fdr1[i],i0r1[i],wx1[i]]

# Vector de resultados de la función
res = [fqs2,fds2,i0s2,fqr2,fdr2,i0r2,wx2,t2,I0,teta2]

return res
```

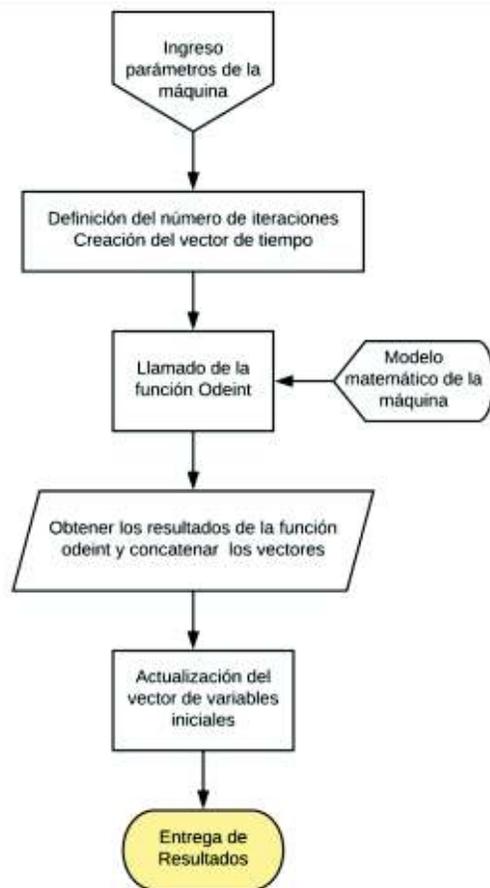


Figura 3.3. Diagrama de flujo de la resolución del modelo

- Cálculo de variables

En esta etapa se realiza el cálculo de todas las variables de la máquina de inducción, aunque se hayan calculado en la función anterior. Esto incluye los resultados de las corrientes en el sistema de referencia qd0 y su transformación al sistema de referencia abc, el torque electromagnético y la velocidad del rotor como se muestra en la Figura 3.4.

La función “Calcular” toma los valores solución de la función “Integrar” para encontrar los resultados de todas las variables de la máquina de inducción.

```
def Calcular(Soluciones):
# Asignación de Variables

[fqs, fds, i0s, fqr, fdr, i0r, wx, t, I1]=[Soluciones[0][:], Soluciones[1][:], Soluciones[2][:], Soluciones[3][:], Soluciones[4][:], Soluciones[5][:], Soluciones[6][:], Soluciones[7][:], Soluciones[8][:]]

# Cálculo de las corrientes qd mediante la relación de flujos
fmq = XM*(fqs/Xls + fqr/Xlr)
fmd = XM*(fds/Xls + fdr/Xlr)
iqs = (fqs - fmq)/Xls
ids = (fds - fmd)/Xls
iqr = (fqr - fmq)/Xlr
idr = (fdr - fmd)/Xlr

# Cálculo del Tem y wr
Tem = ((3*P)/(4*wb))*(fds*iqs - fqs*ids)
wr = wx*30*frated

# Transformación de las corrientes qd0 a abc del estator
ias = iqs + i0s
ibs = -0.5*iqs - ((np.sqrt(3))/2)*ids + i0s
ics = -0.5*iqs + ((np.sqrt(3))/2)*ids + i0s

# Transformación de las corrientes qd0 a abc del rotor
theta=0
iar=(iqr*np.cos(theta-teta)+idr*np.sin(theta-teta))
ibr=(iqr*np.cos((theta-teta) -2*np.pi/3)+idr*np.sin((theta-teta) -
2*np.pi/3))
icr=(iqr*np.cos((theta-teta) +2*np.pi/3)+idr*np.sin((theta-teta)
+2*np.pi/3))
Respuesta=[ias, ibs, ics, iqs, ids, iqr, idr, Tem, wr, iar, ibr, icr]

return Respuesta
```

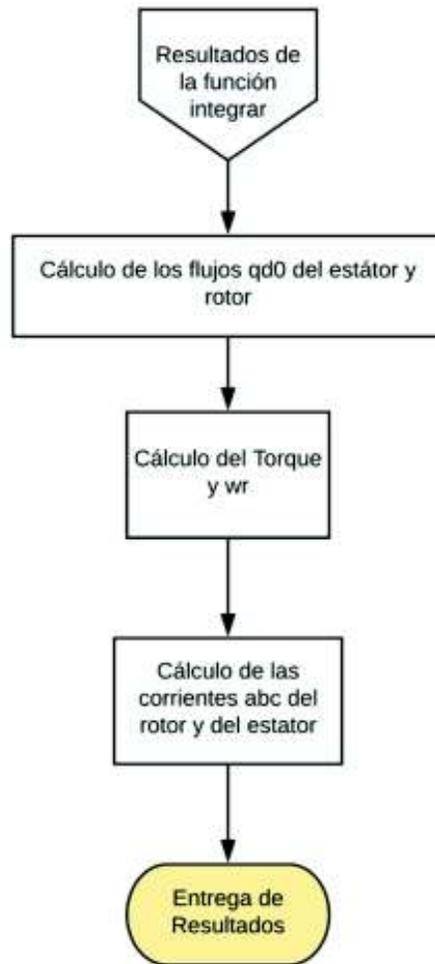


Figura 3.4. Diagrama de flujo del cálculo de las variables

3.2.4 Etapa de resultados

En esta etapa se presentan los resultados de la simulación, una vez obtenidos los resultados de los cálculos anteriores, mediante gráficos o en matrices de acuerdo con las necesidades del usuario. Generalmente las simulaciones se representan gráficamente, sin embargo, puede ser de utilidad para el usuario obtener estos resultados en forma de vector, con el objeto de acoplarlos a un siguiente modelo o continuar con la simulación cambiando algún parámetro de entrada de la máquina.

En la elaboración de gráficos se selecciona las variables a ser graficadas, estas variables deben tener el mismo número de elementos. Luego se deben establecer etiquetas para nombrar los ejes, el tamaño de la gráfica en pulgadas. Finalmente se debe seleccionar el nombre y el formato en el cual se desea grabar la gráfica.

```

##### Graficas #####
plt.figure(figsize = (25,8))
plt.plot(wr,Tem)

```

```

plt.xlabel('Velocidad del Rotor [rpm]', fontsize = 23)
plt.ylabel('Torque Eléctrico [Nm]', fontsize = 23)
plt.title('Torque Eléctrico vs. Velocidad del Rotor', fontsize = 28)
plt.grid(True)
plt.savefig('Graficas Maq. Wr Vs Tem.jpg')

```

3.2.5 Diagramas de flujo general del programa

En la Figura 3.5. se presenta el diagrama de flujo general del programa, con la finalidad de que el usuario del programa entienda su funcionamiento. En el ANEXO 1 y el ANEXO 2 se desarrolla el manual de usuario del programa.

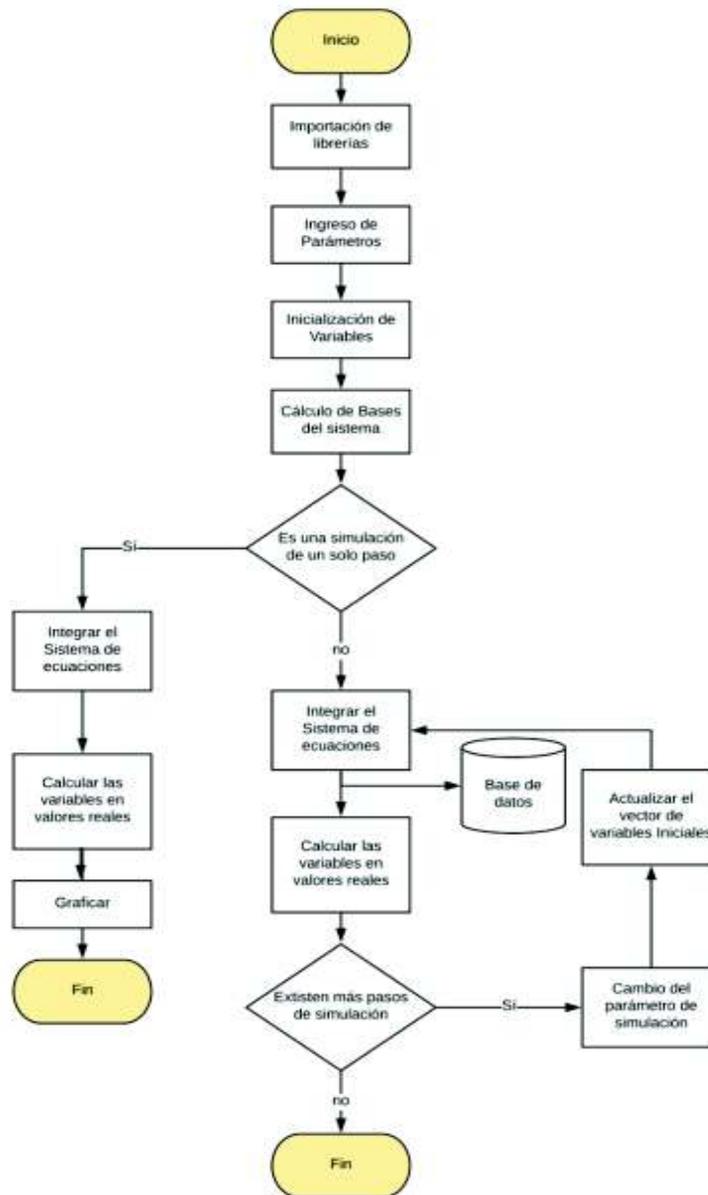


Figura 3.5. Diagrama de flujo del programa

4. VALIDACIÓN DE LA HERRAMIENTA COMPUTACIONAL IMPLEMENTADA EN PYTHON

4.1 Introducción

En este capítulo se realiza la comparación de los resultados obtenidos en el software de simulación Simulink, proporcionado por Matlab, y el programa implementado en el lenguaje de programación Python. Simulink es un software comercial de alta fiabilidad, muy usado en la comunidad científica, razón por la cual se seleccionó este software.

El objetivo de este capítulo es validar el programa implementado en Python mediante su empleo en la resolución de los problemas propuestos. La metodología empleada para realizar esta validación es la siguiente: se toman ejercicios de la bibliografía técnica existente [4] [5], se los simula tomando como referencia el software de simulación Simulink y también se los simula con el programa implementado en Python.

Se comparan gráficamente los resultados y se analizan las diferencias existentes entre estos resultados. Los códigos de para cada máquina se adjuntan en el ANEXO III (digital).

4.1 Comparación entre Python y Simulink

En el capítulo 3 se estableció el modelo matemático implementado en el lenguaje de programación Python. A continuación, es importante comprobar que esté funcionando de manera correcta, identificar las ventajas, desventajas y diferencias matemáticas que puedan existir entre Python y Matlab. Para esto, se simulan casos propuestos en [4] y se comparan resultados.

Las variables a ser comparadas en cada una de las máquinas de inducción serán: corrientes del estator y del rotor, torque eléctrico y velocidad del rotor.

Se observan las variables de las máquinas de inducción descritas en las Tablas 4.1.- 4.5. durante un arranque directo sin carga, cada máquina es trifásica de 4-polos diseñada para operar a una frecuencia de $60 H_z$.

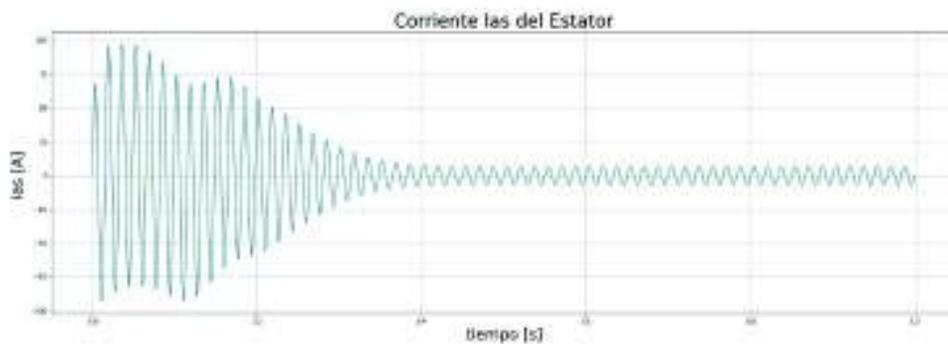
Máquina 1

Tabla 4.1. Datos de placa máquina de 3 hp

Parámetro	Valor	Unidades
Voltaje	220	<i>V</i>
Potencia	3	<i>hp</i>
R _r	0.435	<i>ohm</i>
R _s	0.816	<i>ohm</i>
X _{ls}	0.754	<i>ohm</i>
X _{lr}	0.754	<i>ohm</i>
X _m	26.13	<i>ohm</i>
J	0.089	kg m ²

Corrientes de la fase a del estator.

- Python



- Simulink

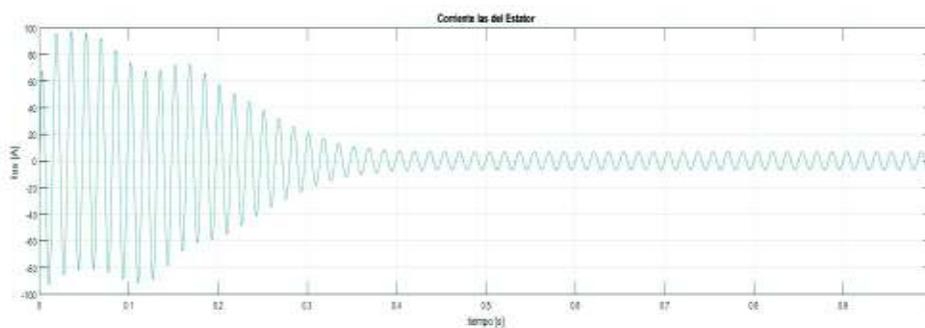


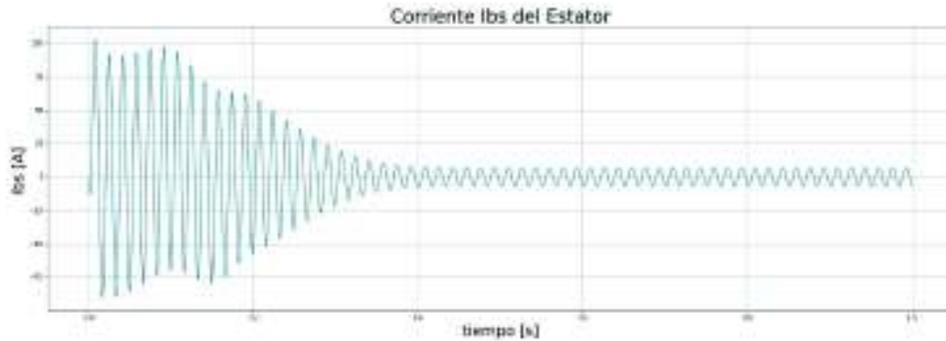
Figura 4.1. Corrientes de la fase a del estator

En la Figura 4.1. se puede apreciar como la corriente comienza con un pico positivo de 60 A y el segundo pico llega hasta 100 A en su ciclo positivo, mientras que en el ciclo negativo el valor máximo es de 90 A; Las dos curvas presentan la misma forma y se estabilizan

aproximadamente a los 0.4 segundos. De la comparación de las dos gráficas, se concluye que las curvas son iguales.

Corrientes de la fase *b* del estator.

- Python



- Simulink

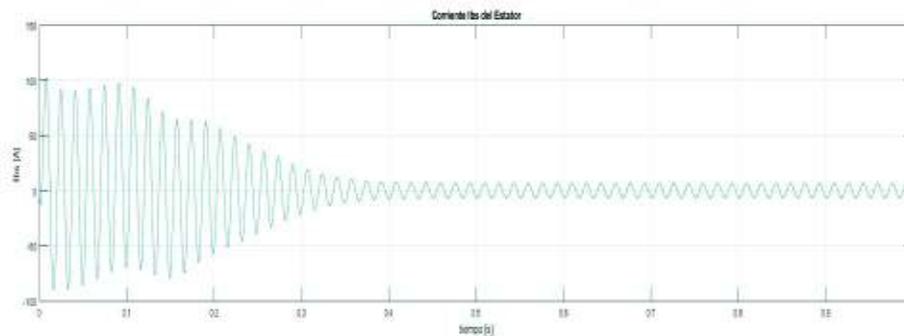
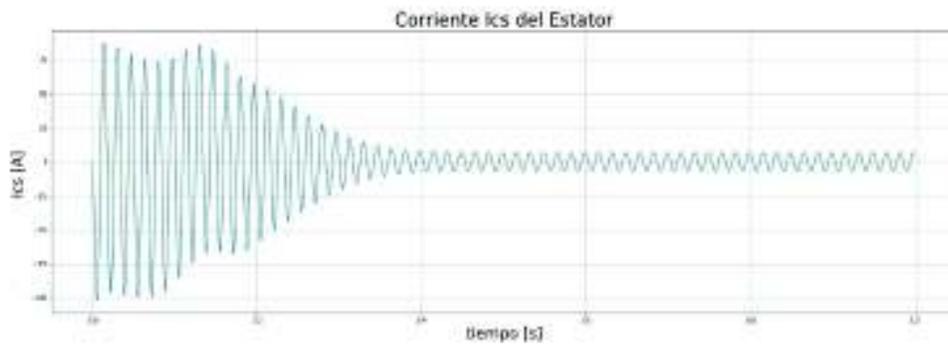


Figura 4.2. Corrientes de la fase *b* del estator

En la Figura 4.2. la corriente comienza con un pico positivo de 100 A y el segundo pico se reduce a un valor ligeramente menor que 100 A, mientras que en el ciclo negativo el valor máximo es de 90 A; las dos curvas presentan la misma forma y se estabilizan aproximadamente a los 0.4 segundos. De la comparación de las dos gráficas se puede concluir que las gráficas son iguales.

Corrientes de la fase c del estator

- Python



- Simulink

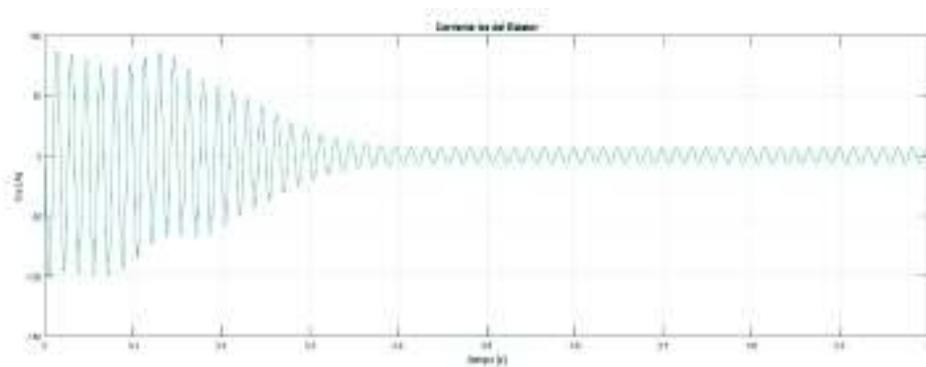


Figura 4.3. Corrientes de la fase c del estator

En la Figura 4.3. se aprecia como la corriente comienza con un pico negativo de 100 A y el segundo pico tiene un valor aproximado de 90 A; mientras que en el ciclo positivo el valor máximo es de 90 A. Las dos curvas presentan la misma forma y se estabilizan aproximadamente a los 0.4 segundos. De la comparación de las gráficas se puede concluir que son iguales

Corrientes de la fase a del rotor

- Python



- Simulink

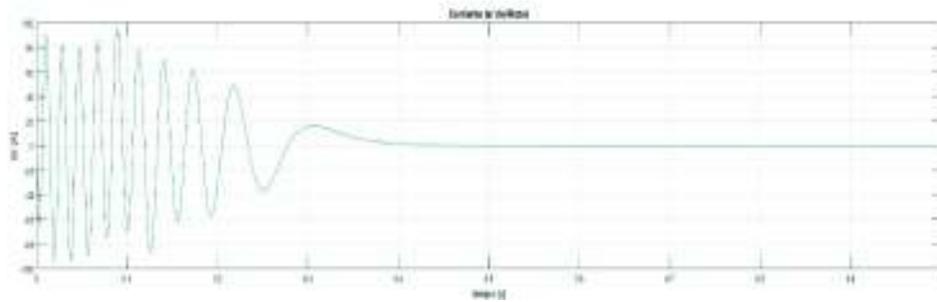
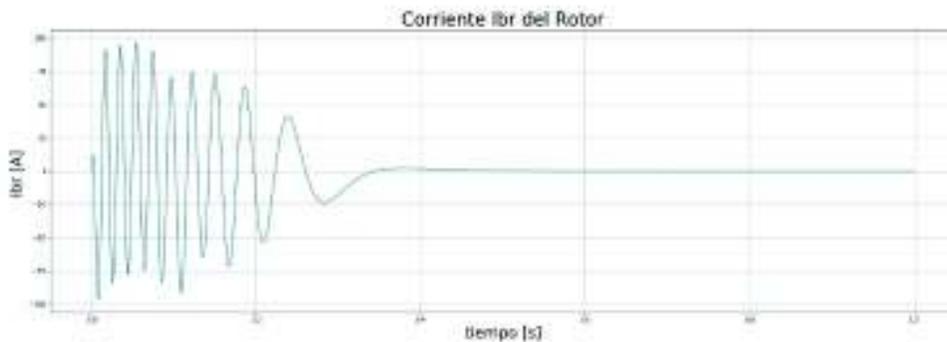


Figura 4.4. Corrientes de la fase *a* del rotor

En la Figura 4.4. se puede apreciar como la corriente comienza con un pico negativo de 60 A y el segundo pico llega hasta 90 A, mientras que en el ciclo positivo valor máximo es de 90 A aproximadamente. Las dos curvas presentan la misma forma y se estabilizan es aproximadamente a los 0.4 segundos. Una vez que la máquina se estabiliza, la corriente en el rotor es cero. De la comparación de las dos gráficas se puede concluir que son iguales.

Corrientes de la fase *b* del rotor

- Python



- Simulink

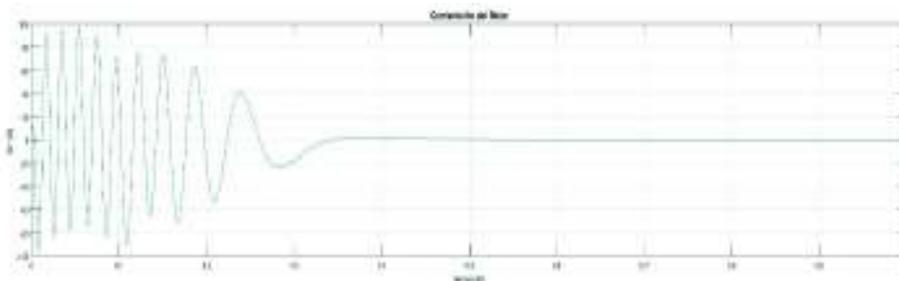
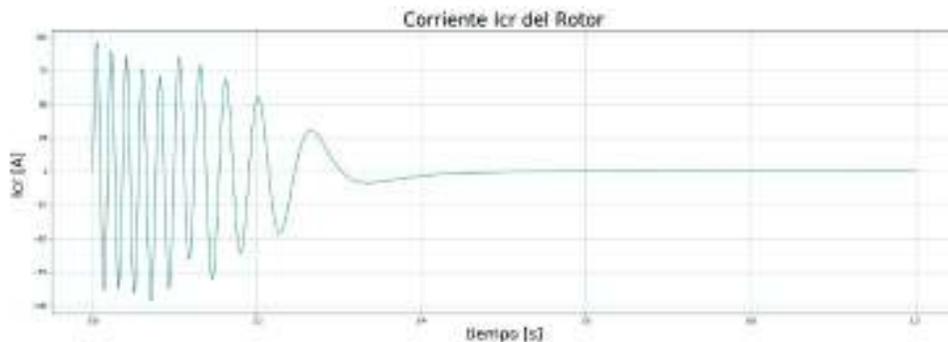


Figura 4.5. Corrientes de la fase *b* del rotor

En la Figura.4.5. de la corriente del rotor para la fase *b*, es interesante notar como la corriente comienza con un pequeño pico positivo menor a 10 A. Esto se aprecia de mejor manera en la gráfica en Python, sin embargo, también se la puede apreciar en la gráfica en Simulink, los valores máximos tanto para el ciclo positivo como para el ciclo negativo son de aproximadamente 90 A. Las dos curvas presentan la misma forma y se estabilizan aproximadamente a los 0.4 segundos. Una vez que la máquina se estabiliza la corriente en el rotor es cero. De la comparación de las dos gráficas se determina que ambas curvas son iguales.

Corrientes de la fase *c* del rotor

- Python



- Simulink

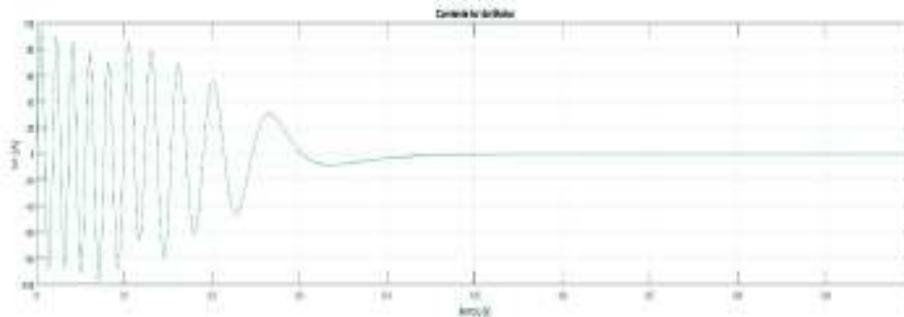
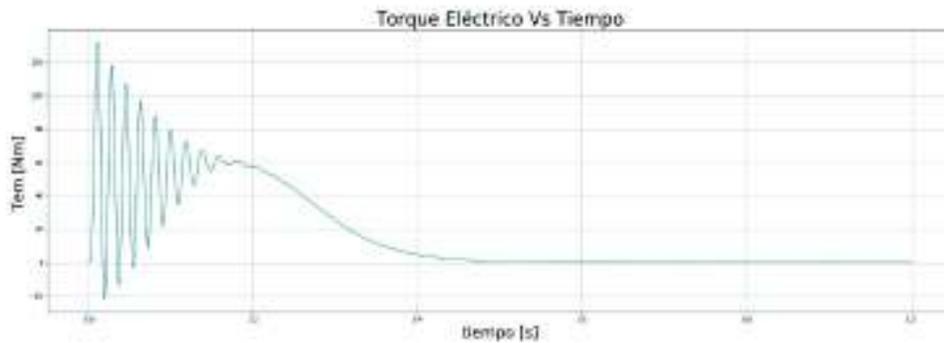


Figura 4.6. Corrientes de la fase *c* del rotor

En la Figura 4.6. se puede apreciar como la corriente comienza con un pico positivo de 100 A y el segundo pico llega hasta 90 A , mientras que en el ciclo negativo valor máximo es de 90 A . Las dos curvas presentan la misma forma y se estabilizan aproximadamente a los 0.4 segundos. Una vez que la máquina se estabiliza la corriente en el rotor es cero. De la comparación de las dos gráficas se puede apreciar que ambas curvas son iguales.

Torque eléctrico

- Python



- Simulink

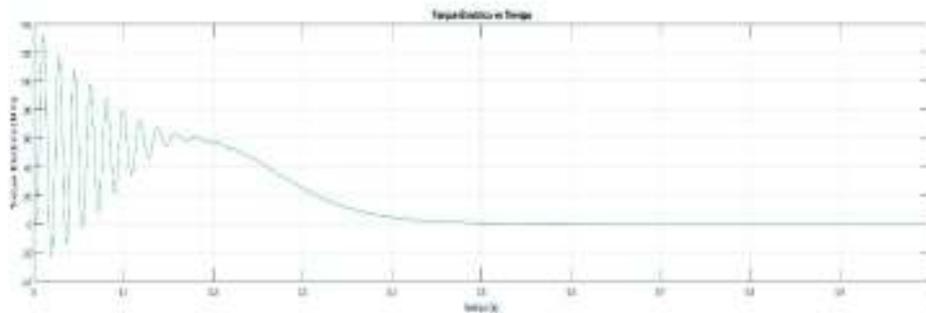
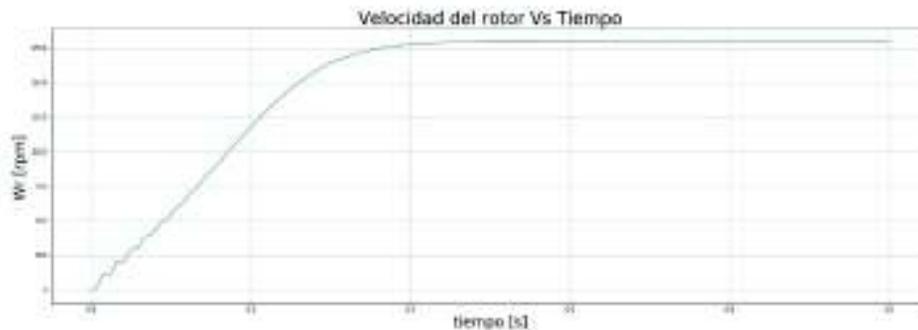


Figura 4.7. Torque eléctrico

En la Figura 4.7. del torque eléctrico presentada anteriormente se aprecia que el valor pico del torque en un arranque sin carga de la máquina de inducción tiene un valor aproximado de 130 Nm. Contrastando ambas gráficas se puede apreciar que las dos oscilan de la misma forma y se estabilizan en un valor aproximado de 0, un poco después de los 0.4 segundos.

Velocidad del rotor

- Python



- Simulink

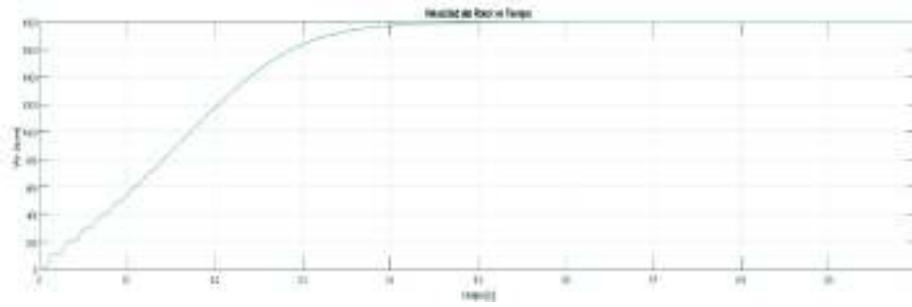
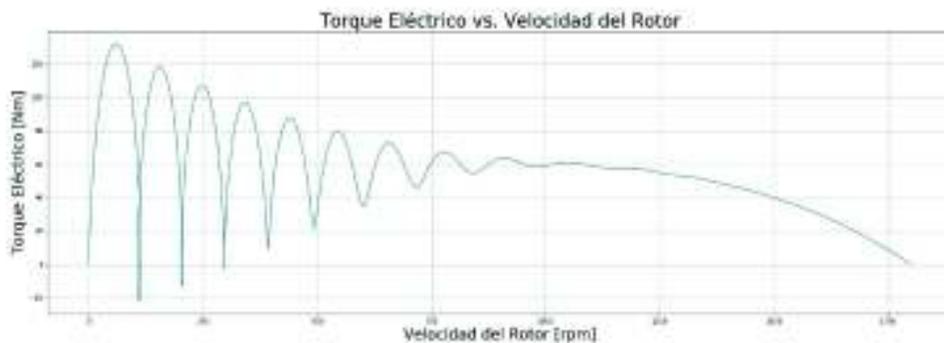


Figura 4.8. Velocidad del rotor

En la Figura 4.8. la velocidad del rotor oscila durante la primera décima de segundo. Pasadas estas oscilaciones la velocidad aumenta hasta llegar a su valor nominal a un tiempo aproximado de 0.4 segundos, una vez que se estabiliza se mantiene en valor nominal.

Torque eléctrico vs velocidad del rotor:

- Python



- Simulink

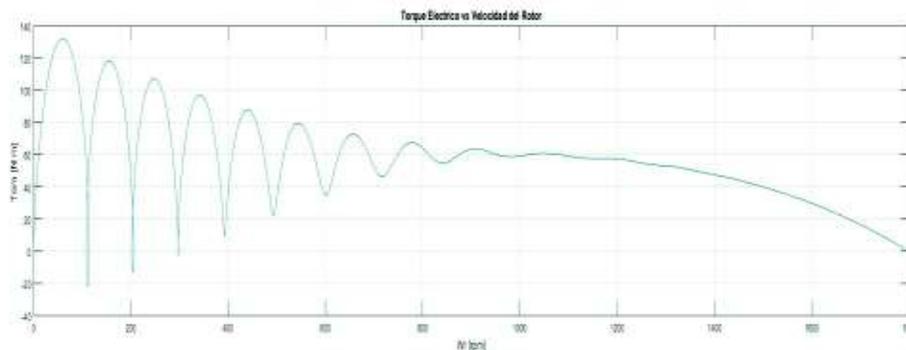


Figura 4.9. Torque eléctrico vs velocidad del rotor

En la Figura 4.9. del torque eléctrico vs la velocidad del rotor se puede apreciar que las dos gráficas oscilan de la misma forma, si se observan los ciclos negativos de las gráficas se encuentra que existen los mismos detalles de oscilación. Una vez que la máquina de inducción alcanza una velocidad aproximada de 1000 rpm, el torque eléctrico comienza a disminuir hasta llegar a un valor de cero a la velocidad nominal de la máquina.

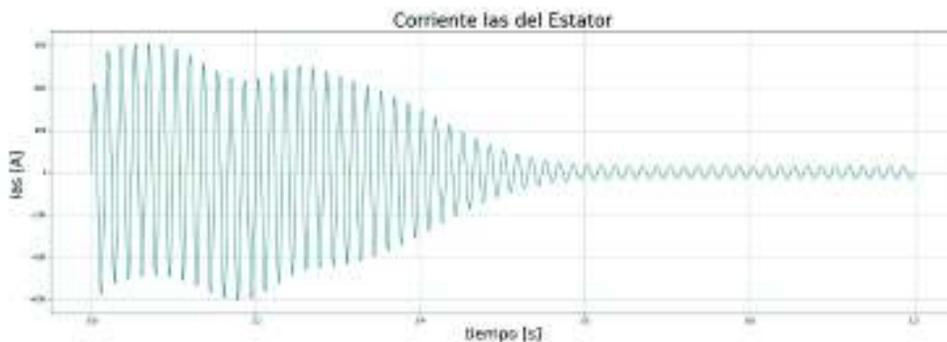
Máquina 2

Tabla 4.2. Datos de placa máquina de 50 hp

Parámetro	Valor	Unidades
Voltaje	460	<i>V</i>
Potencia	50	<i>hp</i>
R _r	0.087	<i>ohm</i>
R _s	0.228	<i>ohm</i>
X _{ls}	0.302	<i>ohm</i>
X _{lr}	0.302	<i>ohm</i>
X _m	13.08	<i>ohm</i>
J	1.662	kg m ²

Corrientes de la fase a del estator

- Python



- Simulink

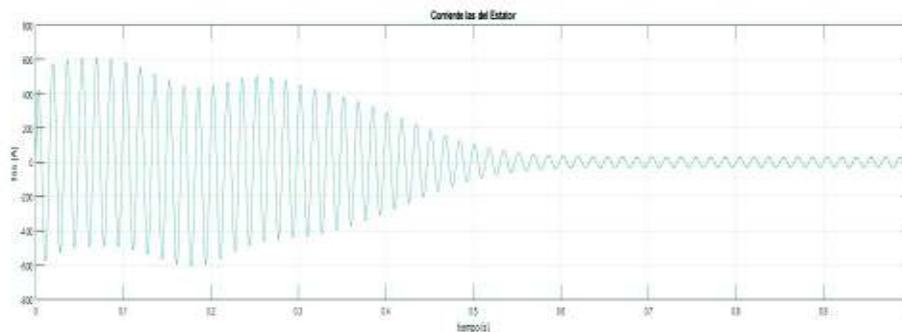
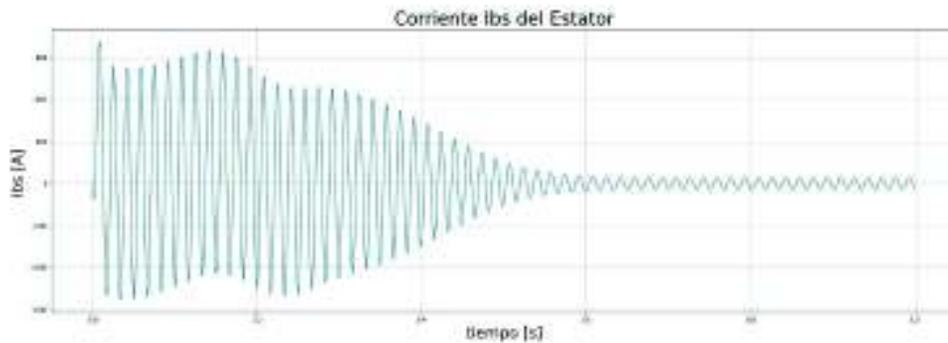


Figura 4.10. Corrientes de la fase a del estator

En la Figura 4.10. se aprecia como la corriente comienza con un pico positivo de 400 A y el segundo pico llega hasta 600 A, mientras que en el ciclo negativo el valor máximo es de 600 A. Las dos curvas presentan la misma forma y se estabilizan aproximadamente a los 0.6 segundos. De la comparación de las dos gráficas se concluye que las curvas son iguales.

Corrientes de la fase *b* del estator

- Python



- Simulink

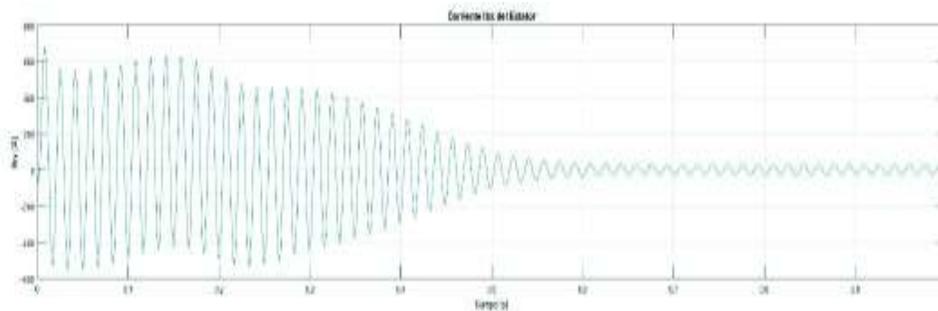
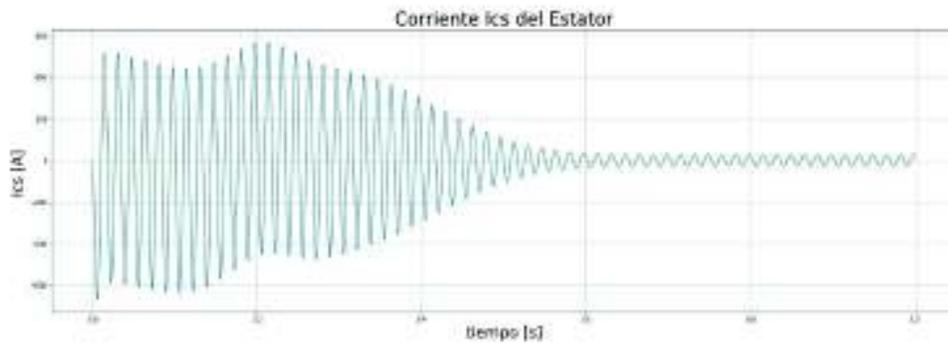


Figura 4.11. Corrientes de la fase *b* del estator

En la Figura 4.11. la corriente de la fase *b* de estator al momento del arranque por un instante tiene un valor negativo muy pequeño, esto se puede apreciar en ambas gráficas. Inmediatamente la corriente aumenta hasta llegar a una valor superior a los 600 A . Las dos curvas presentan la misma forma y se estabilizan aproximadamente a los 0.6 segundos. De la comparación de las dos gráficas se concluye que las curvas son iguales.

Corrientes de la fase c del estator

- Python



- Simulink

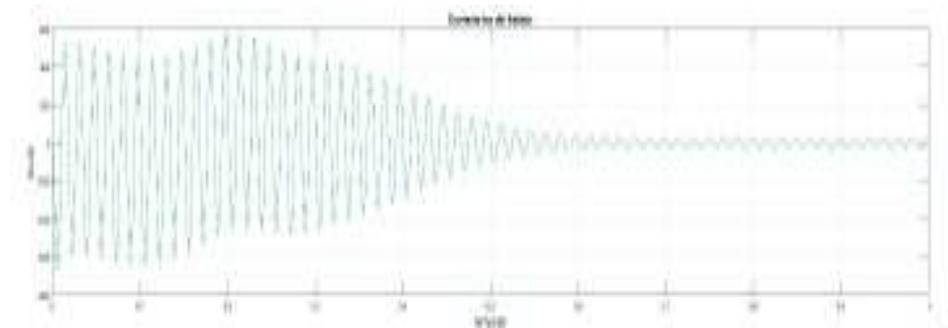
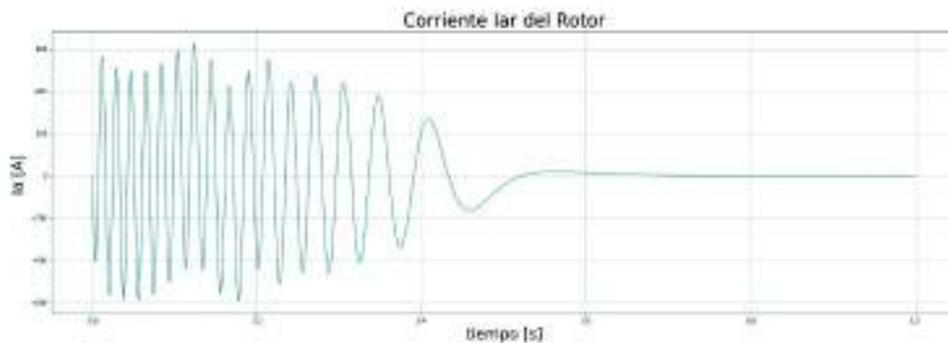


Figura 4.12. Corrientes de la fase c del estator

En la Figura 4.12. se puede apreciar como la corriente comienza con un pico negativo de 600 A y el segundo pico tiene un valor aproximado de 500 A, mientras que en el ciclo positivo el valor máximo es de 500 A. Las dos curvas presentan la misma forma y se estabilizan aproximadamente a los 0.6 segundos. De la comparación de las gráficas se puede concluir que son iguales

Corrientes de la fase a del rotor

- Python



- Simulink

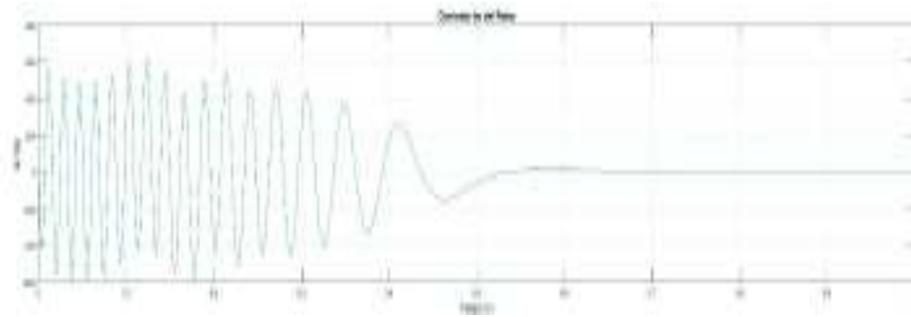
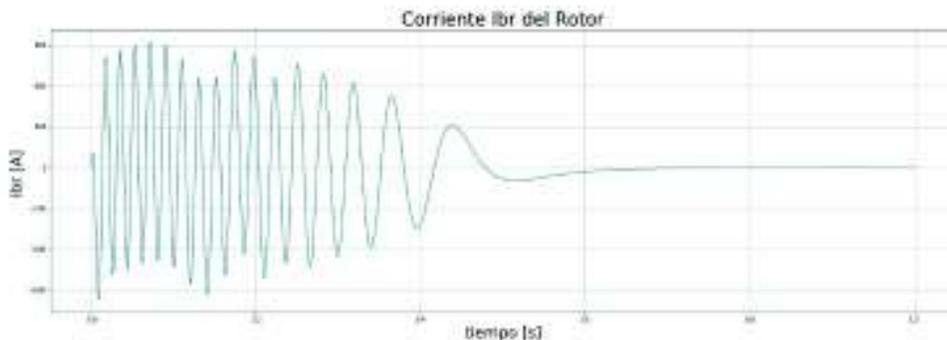


Figura 4.13. Corrientes de la fase *a* del rotor

En Figura 2.13 anterior se puede apreciar como la corriente comienza con un pico negativo de 400 A y el segundo pico llega hasta 550 A, mientras que en el ciclo positivo el valor máximo es de 600 A. Las dos curvas presentan la misma forma y se estabilizan aproximadamente a los 0.5 segundos. Una vez que la máquina se estabiliza la corriente en el rotor es cero. De la comparación de las dos gráficas se puede apreciar que las curvas son iguales.

Corrientes de la fase *b* del rotor

- Python



- Simulink

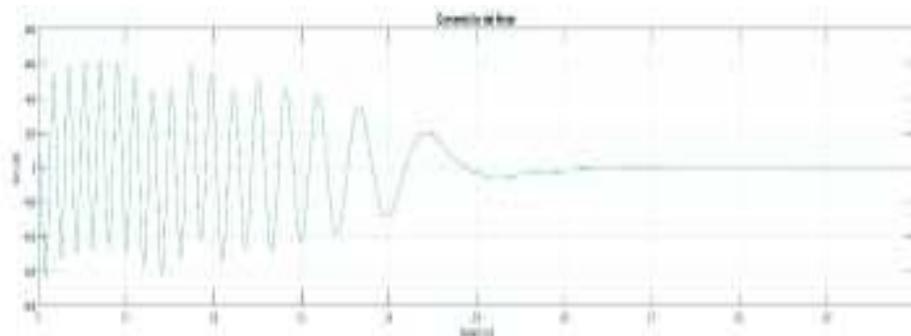
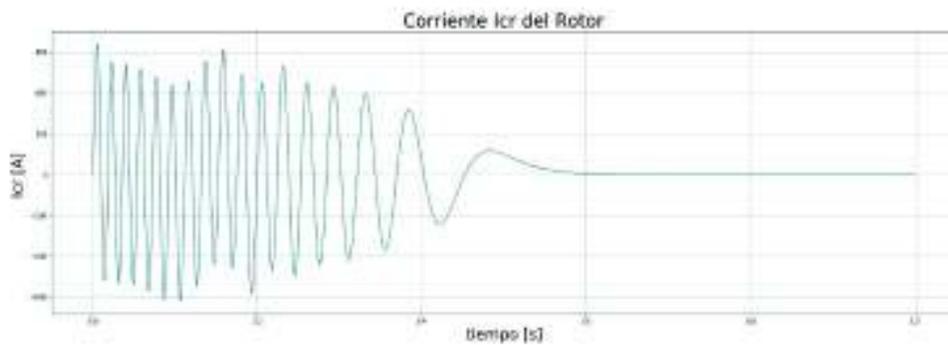


Figura 4.14. Corrientes de la fase *b* del rotor

En la Figura 4.14. de la corriente del rotor para la fase b (lbs), es interesante notar como la corriente comienza con un pequeño pico positivo, esto se aprecia de mejor manera en la gráfica en Python, sin embargo, también se la puede apreciar en la gráfica en Simulink, los valores máximos tanto para el ciclo positivo como para el ciclo negativo son de aproximadamente 600 A. Las dos curvas presentan la misma forma y se estabilizan es proximo a los 0.5 segundos. Una vez que la máquina se estabiliza la corriente en el rotor es cero. De la comparación de las dos gráficas se concluye que las curvas son iguales.

Corrientes de la fase c del rotor

- Python



- Simulink

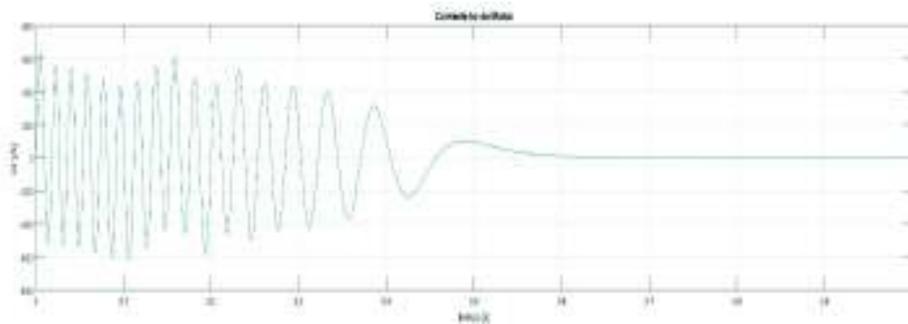
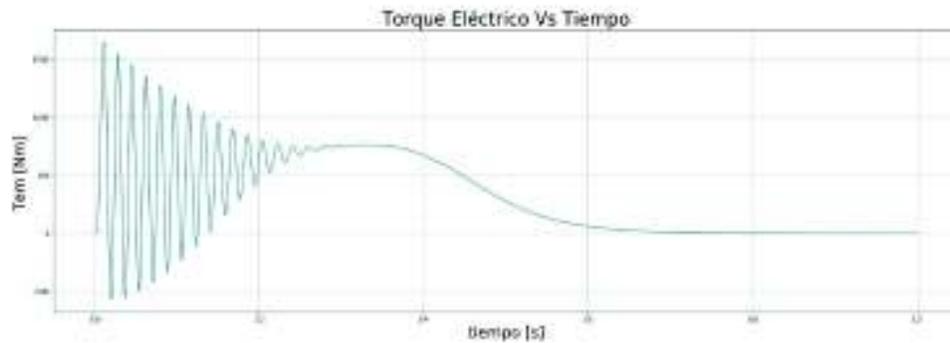


Figura 4.15. Corrientes de la fase c del rotor

En la Figura 4.15. observa como la corriente comienza con un pico positivo aproximadamente de 600 A y el segundo pico de menos de 600 A , mientras que en el ciclo negativo el valor máximo es de 600 A . Las dos curvas presentan la misma forma y se estabilizan aproximadamente a los 0.5 segundos. Una vez que la máquina se estabiliza la corriente en el rotor es cero. De la comparación de las dos gráficas se puede concluir que las curvas son iguales.

Torque eléctrico:

- Python



- Simulink

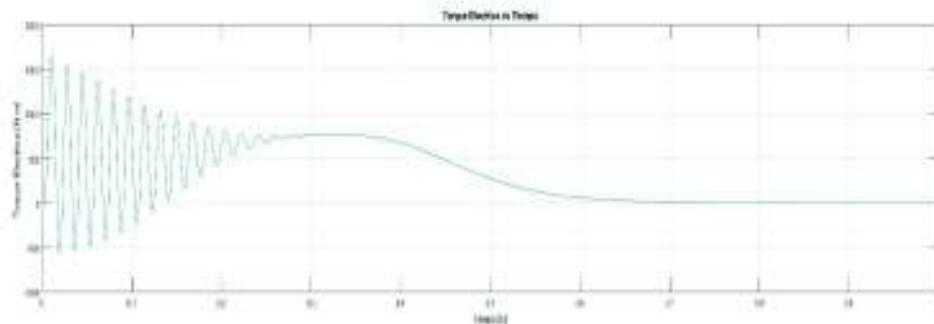
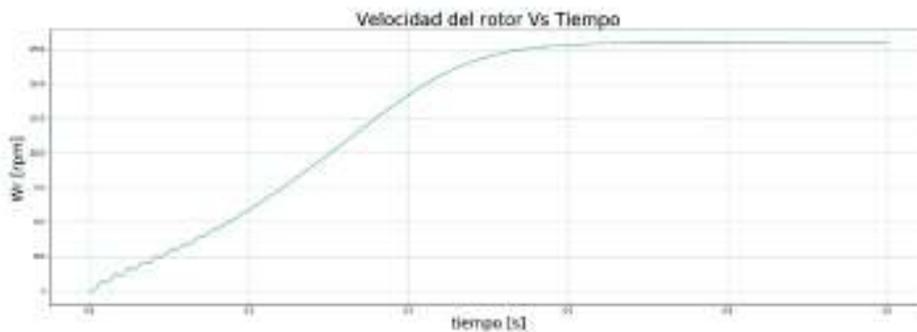


Figura 4.16. Torque eléctrico

En las gráficas del torque eléctrico se puede apreciar que el valor pico del torque, en un arranque sin carga de la máquina de inducción tiene un valor aproximado de 1500 Nm, contrastando ambas gráficas, se aprecia que las dos oscilan de la misma forma y se estabilizan en un valor aproximado de 0, un poco después de los 0.7 segundos.

Velocidad del rotor:

- Python



- Simulink

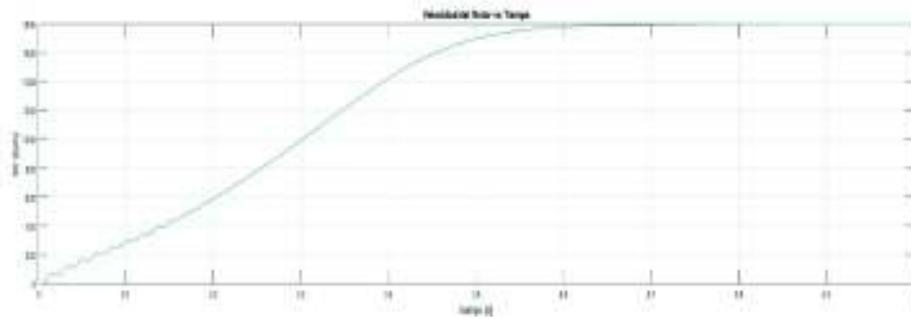
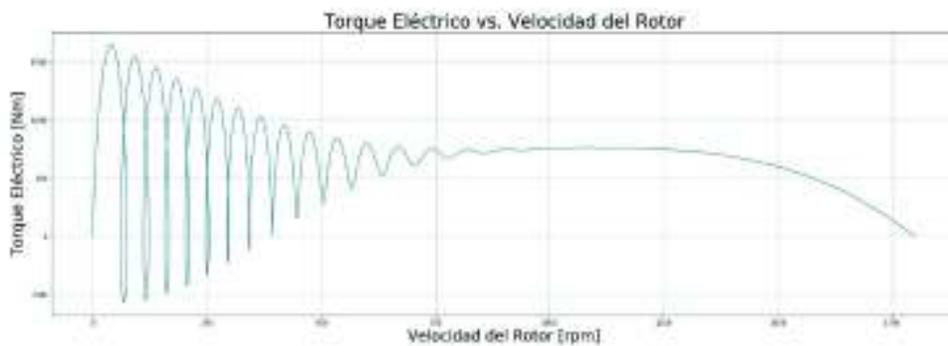


Figura 4.17. Velocidad del rotor

En la Figura 4.17. la velocidad del rotor oscila durante las primeras dos décimas de segundo. Pasadas estas oscilaciones la velocidad aumenta hasta llegar a su valor nominal de 0.6 segundos, aproximadamente. Una vez que se estabiliza se mantiene en valor nominal.

Torque eléctrico vs velocidad del rotor:

- Python



- Simulink

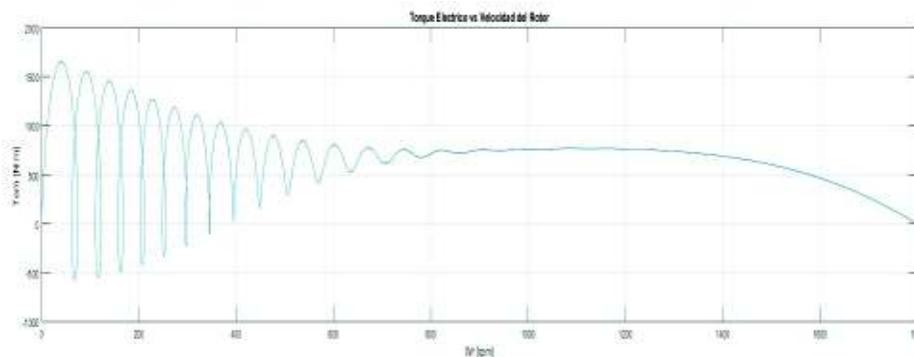


Figura 4.18. Torque eléctrico vs velocidad del rotor

En la Figura 4.18. del torque eléctrico vs la velocidad del rotor, se observa que las dos gráficas oscilan de la misma forma; si se observan los ciclos negativos de las gráficas se aprecia que existen los mismos detalles de oscilación. Una vez que la máquina de inducción alcanza una velocidad aproximada de 900 rpm, el torque eléctrico comienza a disminuir hasta llegar a un valor de cero a la velocidad nominal de la máquina

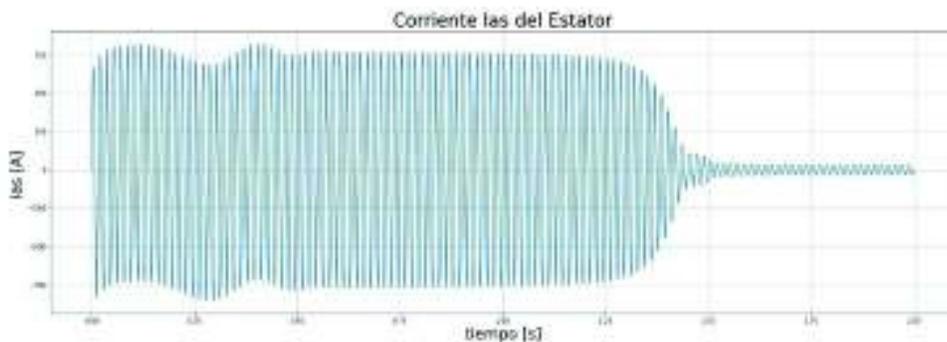
Máquina 3

Tabla 4.3. Datos de placa máquina de 500 hp

Parámetro	Valor	Unidades
Voltaje	2300	<i>V</i>
Potencia	500	<i>hp</i>
R _r	0.262	<i>ohm</i>
R _s	0.187	<i>ohm</i>
X _{ls}	1.206	<i>ohm</i>
X _{lr}	1.206	<i>ohm</i>
X _m	54.02	<i>ohm</i>
J	11.06	kg m ²

Corrientes de la fase a del estator

- Python



- Simulink

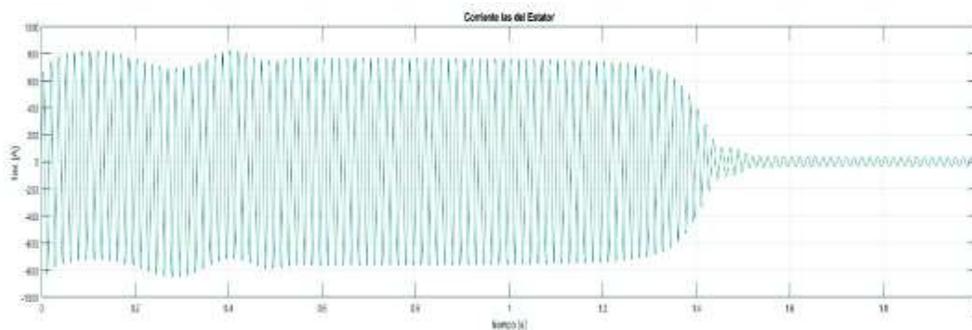
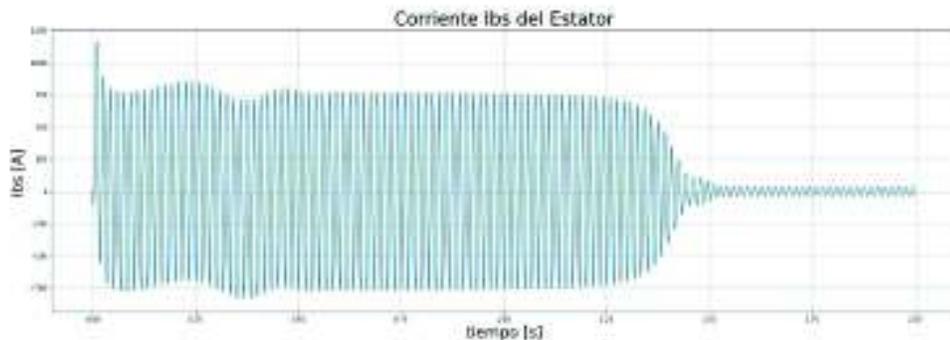


Figura 4.19. Corrientes de la fase a del estator

En la Figura 4.19. la corriente de la fase a para la máquina de inducción 3, oscila de una forma uniforme durante todo el arranque. Los valores de los picos de la gráfica están alrededor de los 800 A, esta oscilación uniforme tiene una duración aproximada de 1.25 segundos. Al llegar a este tiempo la oscilación decrece absolutamente hasta estabilizarse a los 1.6 segundos. Las dos curvas presentan la misma forma. De la comparación de las dos gráficas se puede apreciar que las curvas son iguales.

Corrientes de la fase b del estator

- Python



- Simulink

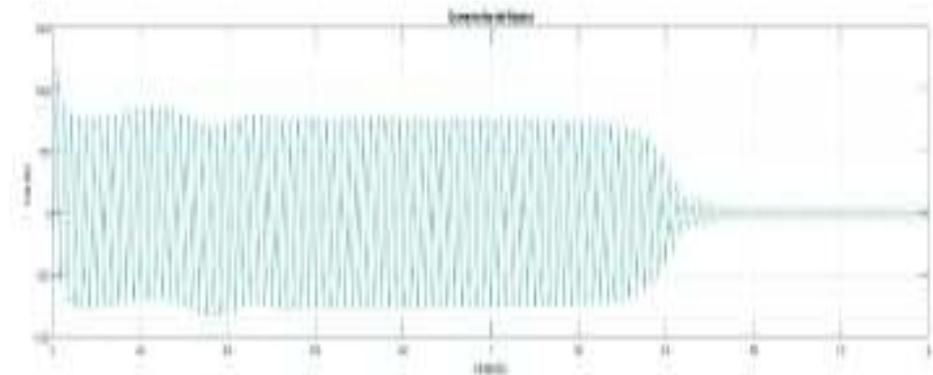
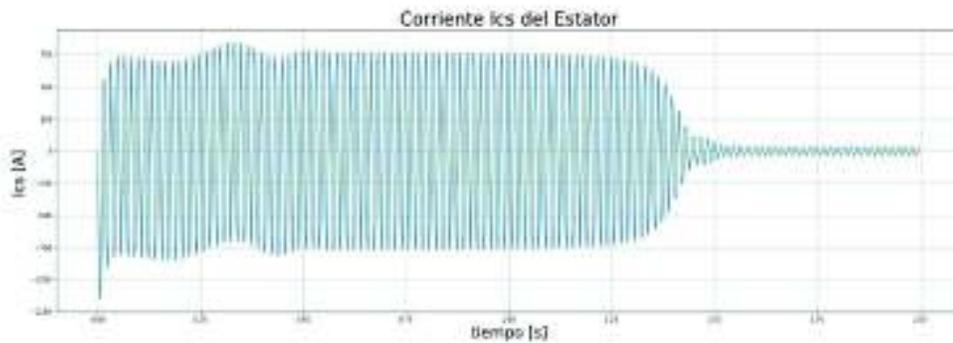


Figura 4.20. Corrientes de la fase b del estator

En la Figura 4.20. la corriente de la fase b de estator al momento del arranque, por un instante tiene un valor negativo muy pequeño, esto se puede apreciar en ambas gráficas. Inmediatamente la corriente aumenta hasta llegar a un valor superior a los 1200 A, fuera de estas características la corriente oscila de forma constante como en la fase a . Las dos curvas presentan la misma forma y se estabilizan es proximo a los 1.6 segundos. De la comparación de las dos gráficas se observa que las curvas son iguales.

Corrientes de la fase c del estator

- Python



- Simulink

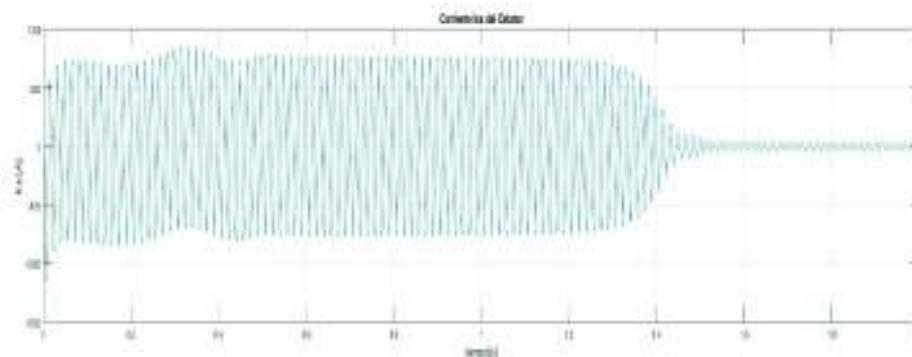
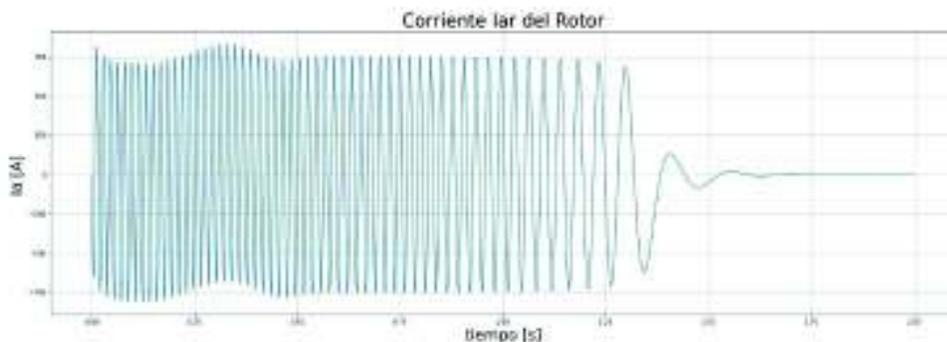


Figura 4.21. Corrientes de la fase c del estator

En la Figura 4.21. al igual que en la fase b, la corriente de la fase c tiene un pico al momento del arranque aproximadamente de 1200 A, con la diferencia de que este pico es negativo para esta fase. La oscilación se mantiene constante hasta 1.2 segundos. Las dos curvas presentan la misma forma y se estabilizan es próximo a los 1.6 segundos. De la comparación de las dos gráficas se llega a la conclusión de que las curvas son iguales.

Corrientes de la fase a del rotor

- Python



- Simulink

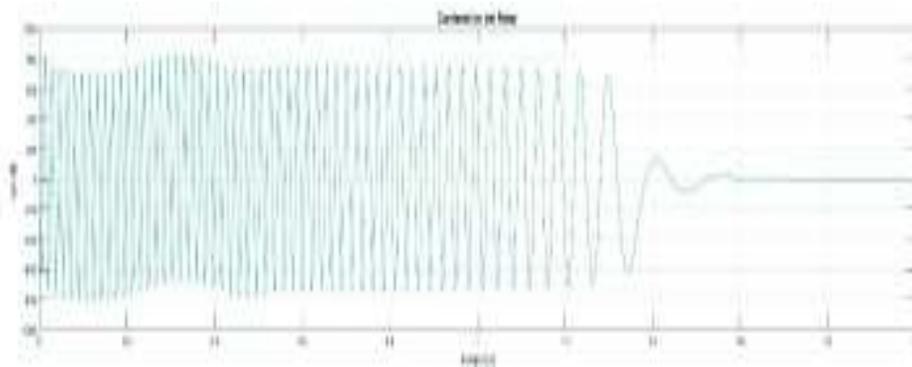
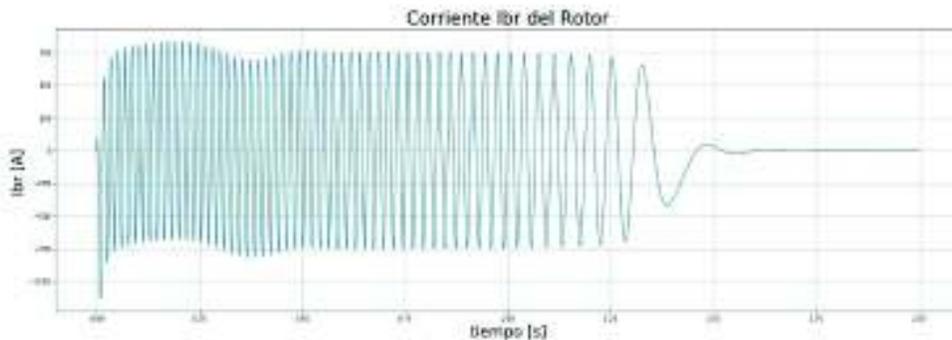


Figura 4.22. Corrientes de la fase *a* del rotor

En la Figura 4.22. la corriente de la fase *a* del rotor (I_{ar}) tiene un comportamiento muy similar en la oscilación de la corriente de la fase *a* del estator, sin embargo, a diferencia de la corriente del estator se puede ver como la frecuencia de las oscilaciones baja con el aumento del tiempo y la curva se estabiliza entre 1.4 segundos y 1.6 segundos. Las dos curvas presentan la misma forma. Una vez que la máquina se estabiliza, la corriente en el rotor es cero. De la comparación de las dos gráficas se aprecia que las curvas son iguales.

Corrientes de la fase *b* del rotor

- Python



- Simulink

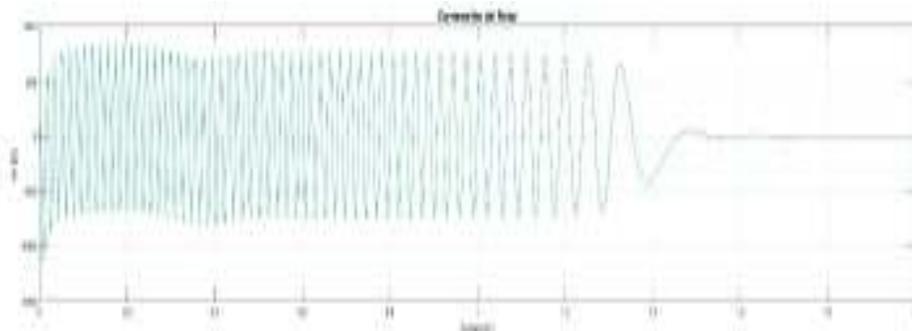
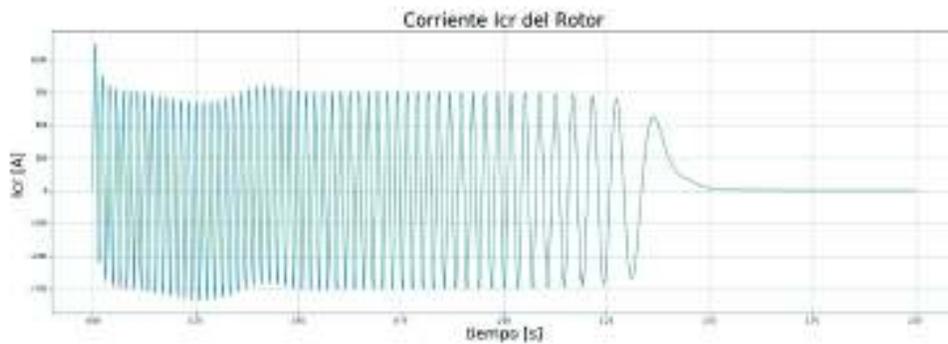


Figura 4.23. Corrientes de la fase *b* del rotor

En la Figura 4.23. de la corriente del rotor para la fase b es interesante notar como la corriente comienza con un pequeño pico positivo. Esta corriente también oscila de forma uniforme, sin embargo, presenta un pico de 1200 A, la oscilación se mantiene constante hasta 1.2 segundos. Las dos curvas presentan la misma forma y se estabilizan aproximadamente a los 1.6 segundos. De la comparación de las dos gráficas se concluye que las curvas son iguales.

Corrientes de la fase c del rotor

- Python



- Simulink

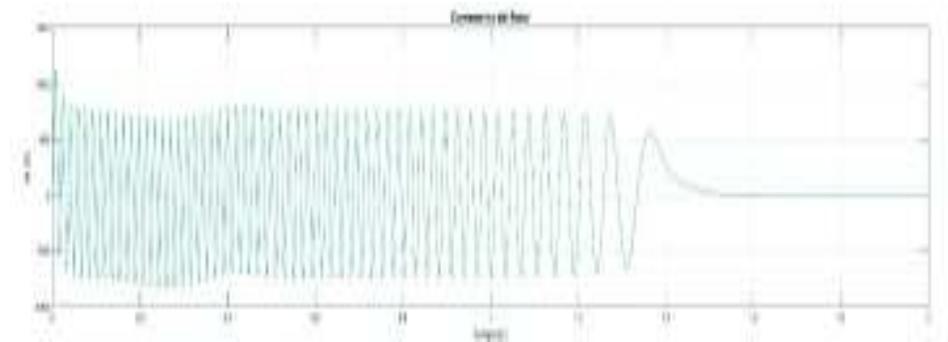
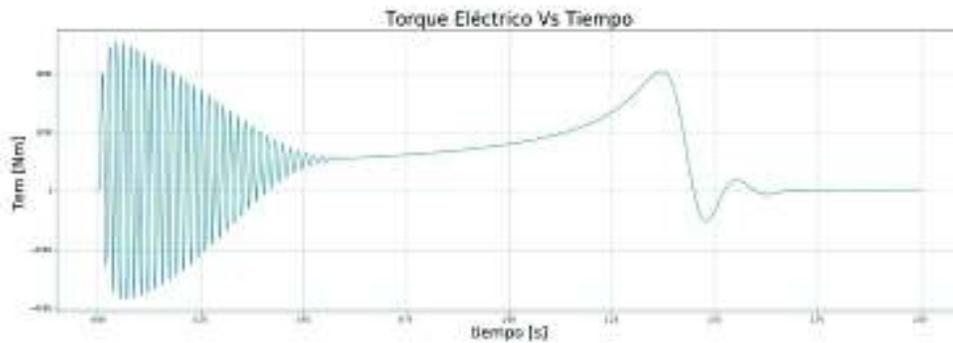


Figura 4.24. Corrientes de la fase c del rotor

En la Figura 4.24. se puede apreciar como la oscilación se mantiene constante hasta 1.2 segundos. Las dos curvas presentan la misma forma y se estabilizan aproximadamente a los 1.6 segundos. De la comparación de las dos gráficas se llega a la conclusión de que las curvas son iguales.

Torque eléctrico

- Python



- Simulink

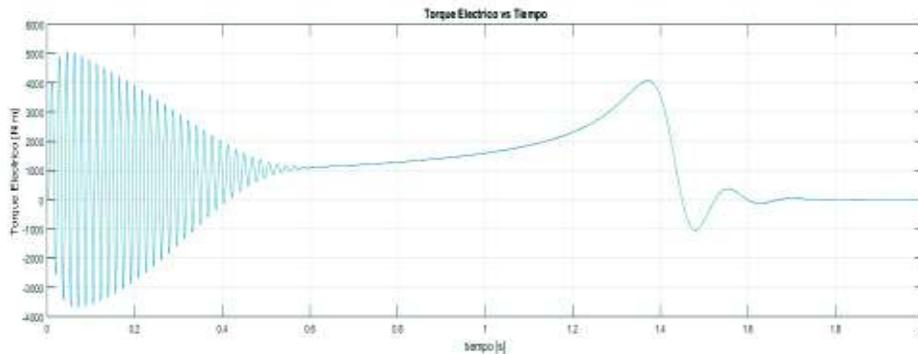
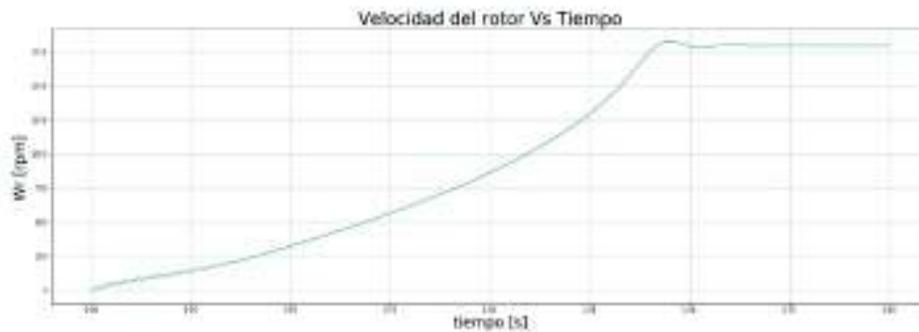


Figura 4.25. Torque eléctrico

En la Figura 4.25. del torque eléctrico presentada anteriormente se observa que el valor pico del torque en un arranque sin carga de la máquina de inducción tiene un valor aproximado de 5000 Nm. El torque eléctrico oscila hasta aproximadamente 0.6 segundos. De 0.6 segundos hasta 1.4 segundos, el torque aumenta progresivamente hasta llegar a un valor de 4000 Nm. Hasta estabilizarse a los 1.6 segundos el toque presenta una pequeña oscilación. Contrastando ambas gráficas se puede notar que las dos oscilan de la misma forma y se estabilizan en un valor aproximado de 0.

Velocidad del rotor

- Python



- Simulink

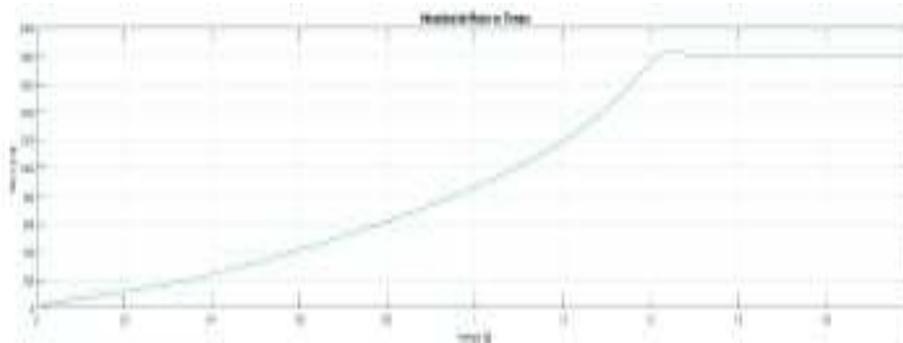
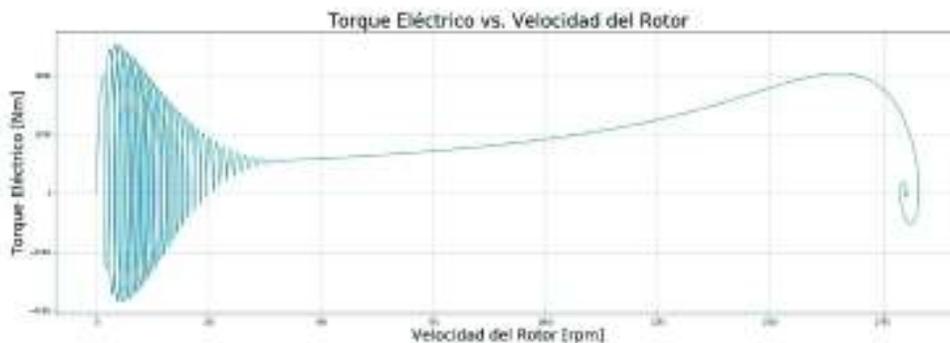


Figura 4.26. Velocidad del rotor

En la Figura 4.26. la velocidad del rotor oscila durante las primeras cinco décimas de segundo. Pasadas estas oscilaciones, la velocidad aumenta hasta llegar a su valor nominal a un tiempo aproximado de 1.4 segundos. Entre el período de 1.4 segundos y 1.6 segundos, la velocidad vuelve a oscilar levemente como el torque, una vez que se estabiliza se mantiene en valor nominal.

Torque eléctrico vs velocidad del rotor

- Python



- Simulink

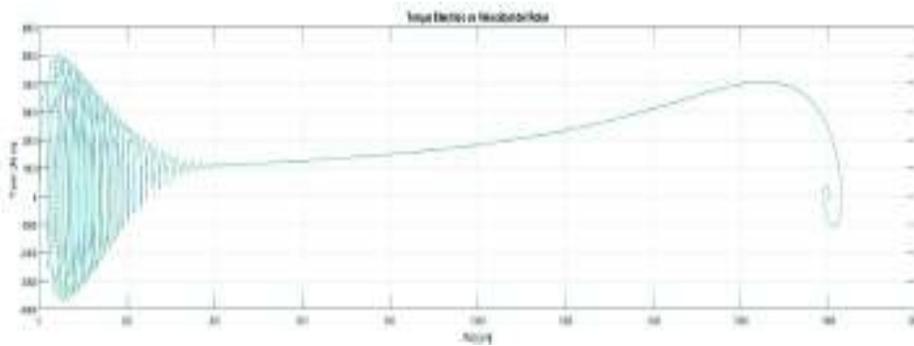


Figura 4.27. Torque eléctrico vs velocidad del rotor

En la Figura 4.27. del torque eléctrico vs la velocidad del rotor, se puede apreciar que las dos gráficas oscilan de la misma forma. Una vez que la máquina de inducción alcanza una velocidad aproximada de 400 rpm, el torque eléctrico comienza a aumentar hasta llegar a un valor de 4000 Nm con una velocidad de 1800 rpm; una vez llegado a este valor el torque comienza a disminuir hasta llegar a cero, sin embargo, antes de llegar a cero el torque fluctúa circularmente sobre este valor a la velocidad nominal de la máquina.

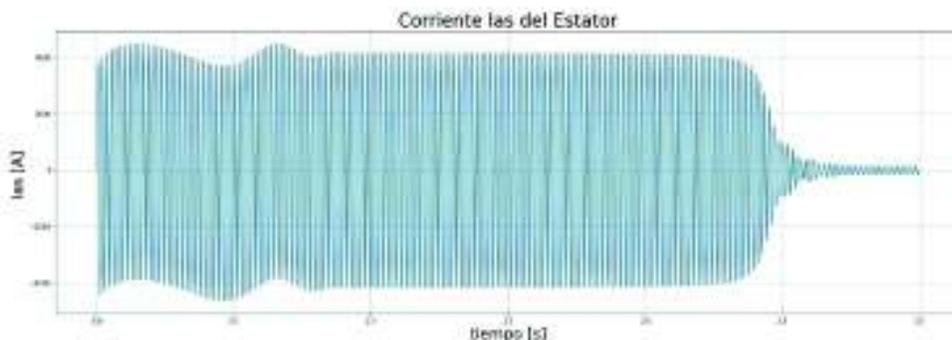
Máquina 4

Tabla 4.4. Datos de placa máquina de 2250 hp

Parámetro	Valor	Unidades
Voltaje	2300	<i>V</i>
Potencia	2250	<i>hp</i>
R _r	0.029	<i>ohm</i>
R _s	0.022	<i>ohm</i>
X _{ls}	0.226	<i>ohm</i>
X _{lr}	0.226	<i>ohm</i>
X _m	13.04	<i>ohm</i>
J	63.87	kg m ²

Corrientes de la fase a del estator

- Python



- Simulink

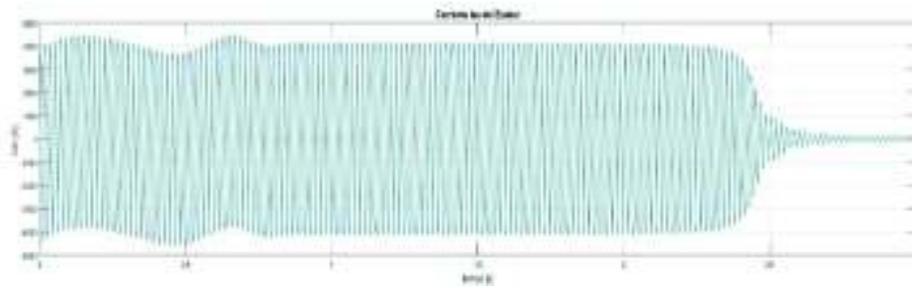
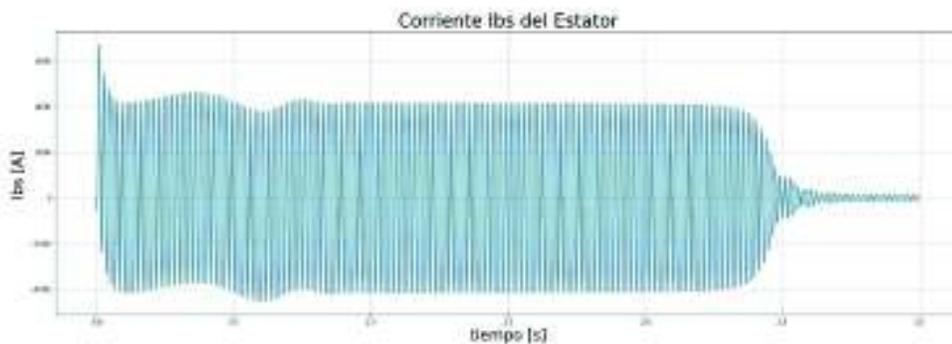


Figura 4.28. Corrientes de la fase *a* del estator

En la Figura 4.28. la corriente de la fase *a* para la máquina de inducción 4 oscila de una forma uniforme durante todo el arranque. Los valores de los picos de la gráfica están alrededor de los 4000 A. La oscilación uniforme tiene una duración aproximada de 2.4 segundos, al llegar a este tiempo ésta decrece absolutamente, hasta estabilizarse a los 2.75 segundos. Las dos curvas presentan la misma forma. De la comparación de las dos gráficas las curvas se asumen iguales.

Corrientes de la fase *b* del estator

- Python



- Simulink

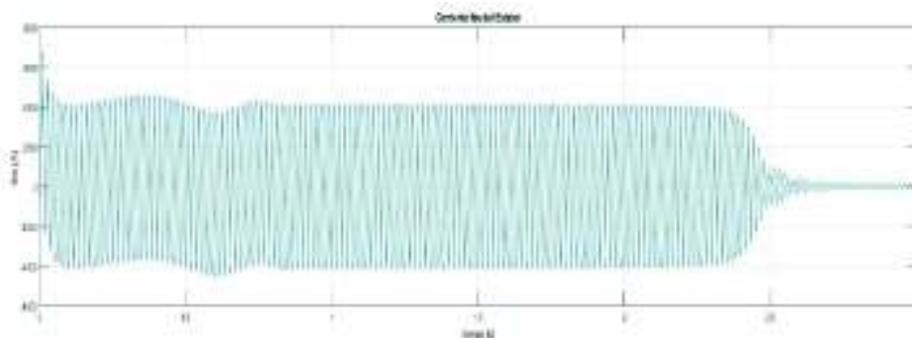
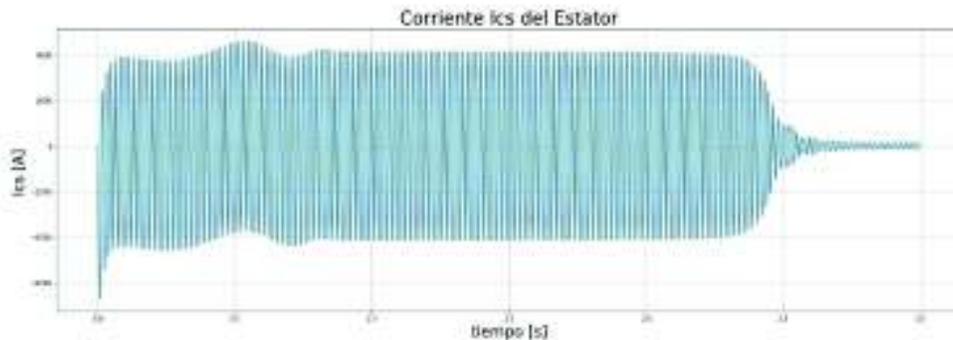


Figura 4.29. Corrientes de la fase *b* del estator

En la Figura 4.29. la corriente de la fase b de estator, al momento del arranque por un instante tiene un valor negativo muy pequeño. Inmediatamente la corriente aumenta hasta llegar a una valor superior a los 6000 A fuera de estas características la corriente oscila de una forma constante al igual que la fase a. Las dos curvas presentan la misma forma y se estabilizan aproximadamente a los 2.75 segundos. De la comparación de las gráficas se puede concluir que son iguales

Corrientes de la fase c del estator

- Python



- Simulink

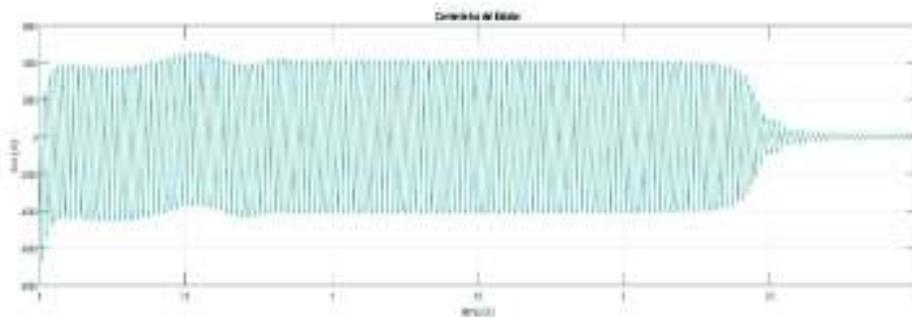
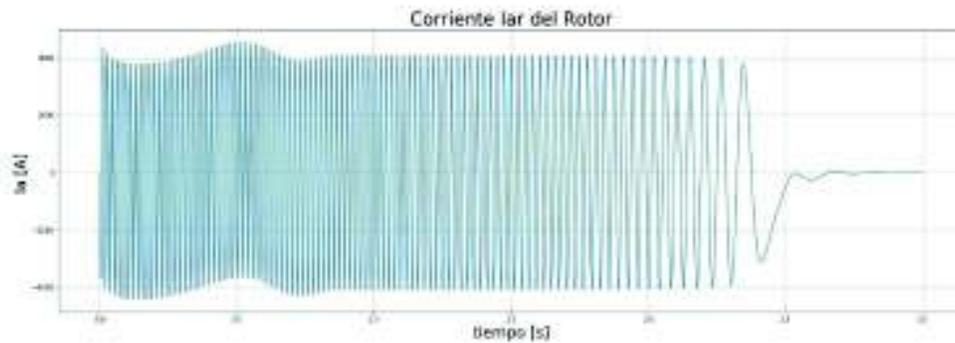


Figura 4.30. Corrientes de la fase c del estator

En la Figura 4.30. al igual que en la fase b, la corriente de la fase c tiene un pico al momento del arranque aproximadamente de 6000 A, con la diferencia de que este pico es negativo para esta fase. La oscilación se mantiene constante hasta 2.4 segundos. Las dos curvas presentan la misma forma y se estabilizan aproximadamente a los 2.75 segundos. De la comparación de las dos gráficas las curvas son iguales.

Corrientes de la fase a del rotor

- Python



- Simulink

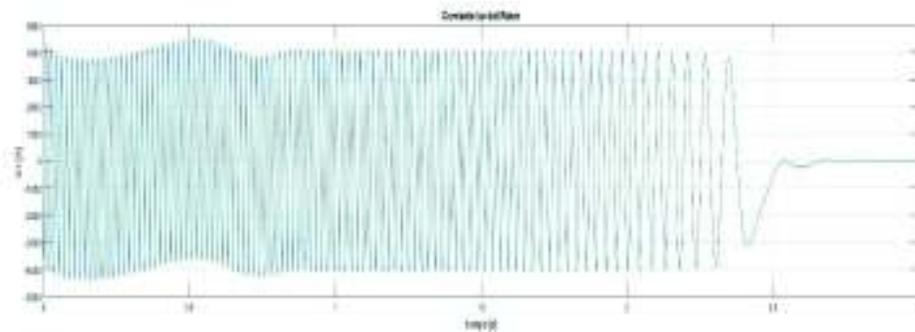
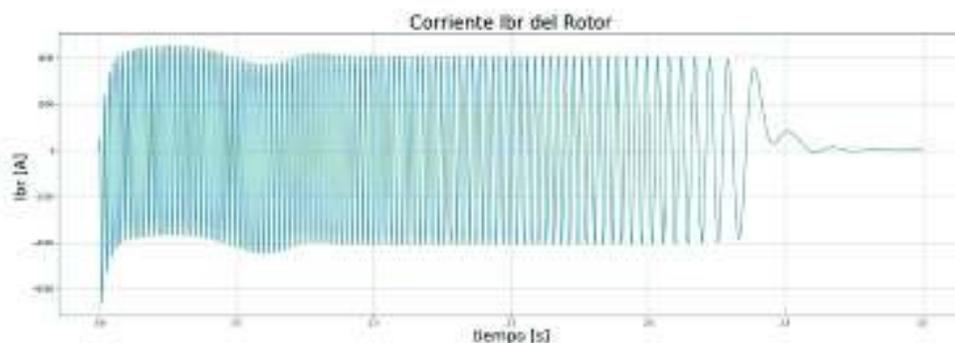


Figura 4.31. Corrientes de la fase a del rotor

En la Figura 4.31. la corriente de la fase a del rotor (I_{ar}) tiene un comportamiento muy similar en la oscilación de la corriente de la fase a del estator, sin embargo, a diferencia de la corriente del estator se puede ver como la frecuencia de las oscilaciones baja con el aumento del tiempo y la curva se estabiliza entre 2.4 segundos y 2.5 segundos. Las dos curvas presentan la misma forma. Una vez que la máquina se estabiliza, la corriente en el rotor es cero. De la comparación de las dos las curvas estas son iguales.

Corrientes de la fase b del rotor

- Python



- Simulink

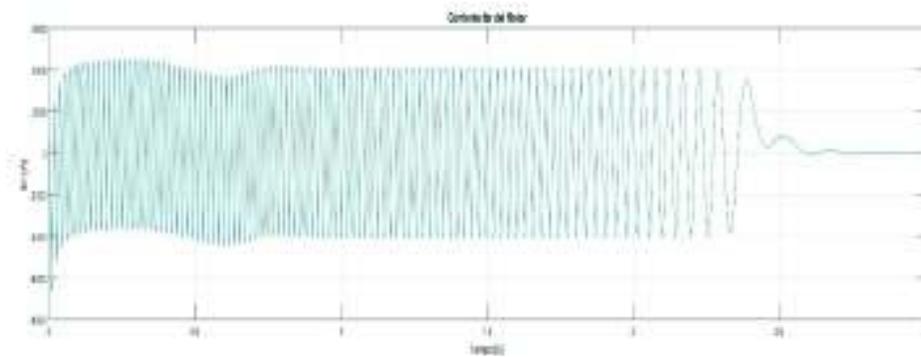
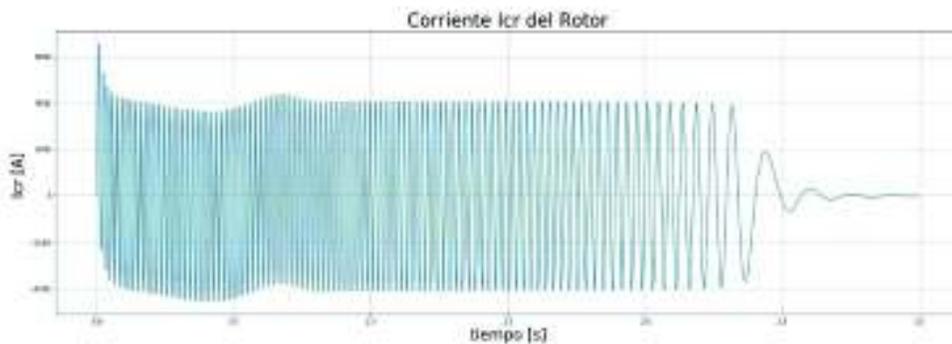


Figura 4.32. Corrientes de la fase *b* del rotor

En la Figura 4.32. de la corriente del rotor para la fase *b* (*lbs*), es interesante notar como la corriente comienza con un pequeño pico positivo. Esta corriente también oscila de forma uniforme; a pesar de que presenta un pico de 6000 A, la oscilación se mantiene constante hasta 2.4 segundos. Las dos curvas presentan la misma forma y se estabilizan aproximadamente a los 2.75 segundos. De la comparación de las dos gráficas se concluye que son iguales.

Corrientes de la fase *c* del rotor

- Python



- Simulink

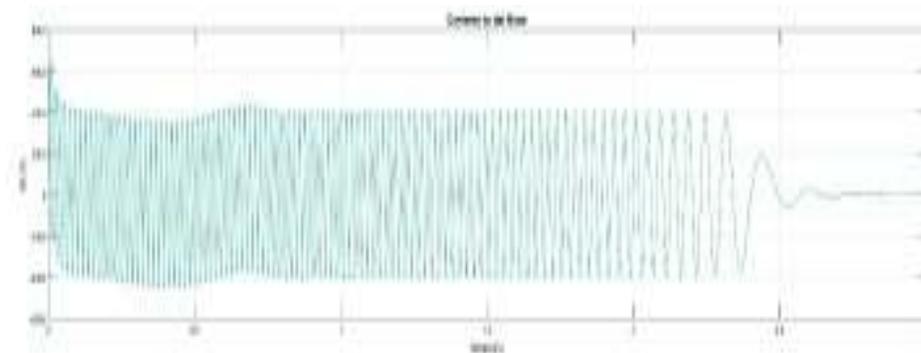
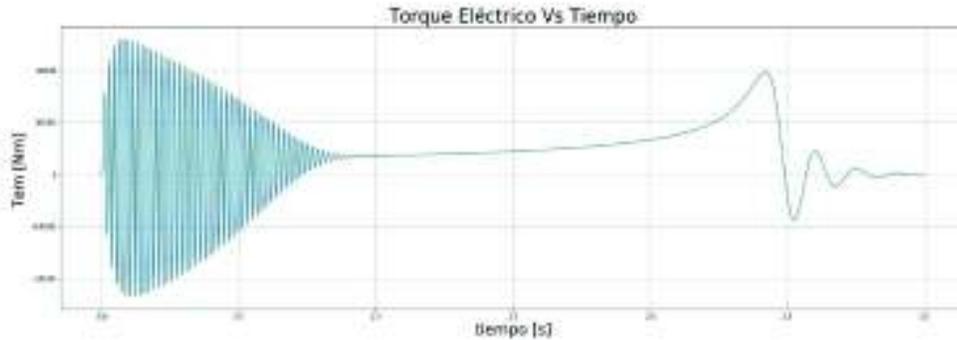


Figura 4.33. Corrientes de la fase *c* del rotor

En la Figura 4.33. puede apreciar como la oscilación se mantiene constante hasta 2.4 segundos. Las dos curvas presentan la misma forma y se estabilizan es proximo a los 2.75 segundos. De la comparación de las dos gráficas se determina que son iguales.

Torque eléctrico

- Python



- Simulink

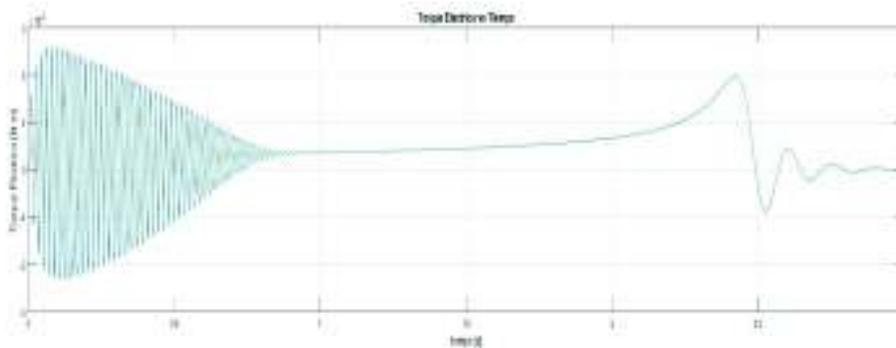


Figura 4.34. Torque eléctrico

En la Figura 4.34. del torque eléctrico presentada anteriormente, se puede apreciar que el valor pico del torque en un arranque sin carga de la máquina de inducción, tiene un valor aproximado de 25000 Nm, el torque eléctrico oscila hasta aproximadamente 0.9 segundos. De 0.9 segundos hasta 2.4 segundos, el torque aumenta progresivamente hasta llegar a un valor de 20000 Nm. Hasta estabilizarse a los 2.75 segundos, el toque presenta una pequeña oscilación. Contrastando ambas gráficas se puede apreciar que las dos oscilan de la misma forma y se estabilizan en un valor aproximado de 0.

Velocidad del rotor

- Python



- Simulink

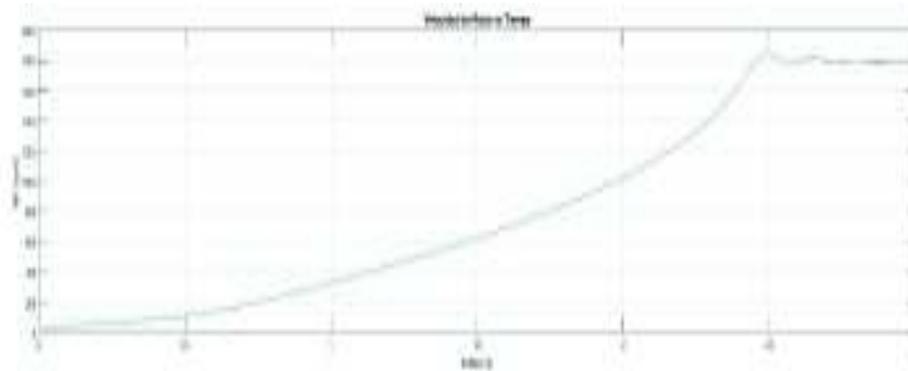
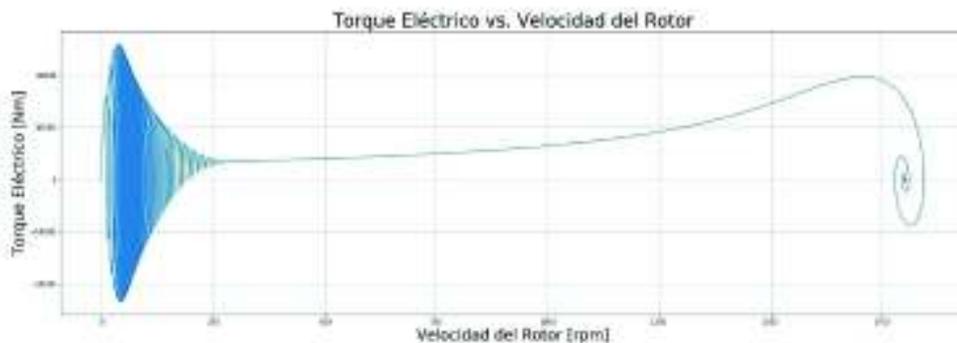


Figura 4.35. Velocidad del rotor

En la Figura 4.35. la velocidad del rotor oscila durante las primeras siete décimas de segundo. Pasadas estas oscilaciones la velocidad aumenta hasta llegar a su valor nominal, a un tiempo aproximado de 2.4 segundos. Entre el período de 2.4 segundos y 2.75 segundos la velocidad vuelve a oscilar levemente como el torque, una vez que se estabiliza se mantiene en valor nominal.

Torque eléctrico vs velocidad del rotor

- Python



- Simulink

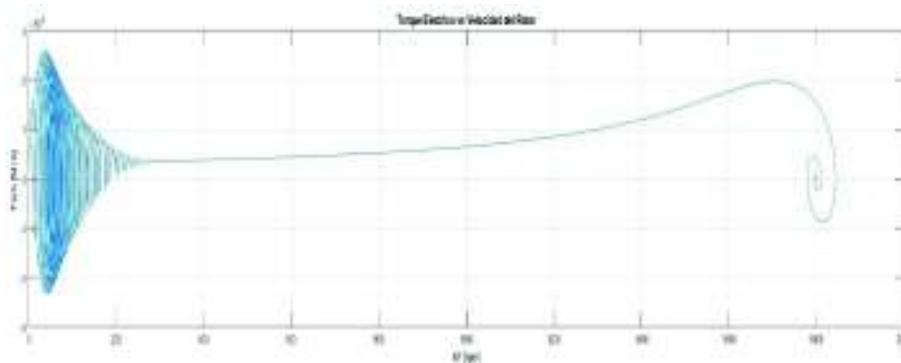


Figura 4.36. Torque eléctrico vs velocidad del rotor

En la Figura 4.36. del torque eléctrico versus la velocidad del rotor se puede apreciar que las dos gráficas oscilan de la misma forma. Una vez que la máquina de inducción alcanza una velocidad aproximada de 400 rpm, el torque eléctrico comienza a aumentar hasta llegar a un valor de 2500 Nm con una velocidad de 1700 rpm aproximadamente; una vez llegado a este valor el torque comienza a disminuir hasta llegar a cero, sin embargo, antes de llegar a cero el torque fluctúa circularmente a la velocidad nominal de la máquina.

De las Figuras 4.1.- 4.36. anteriores para las 4 máquinas de inducción propuestas por en [4], se puede concluir que el algoritmo implementado en el lenguaje de programación Python y el programa obtenido en el software de simulación Simulink de Matlab, entregan los mismos resultados, en las mismas condiciones. Es importante apreciar que se han simulado máquinas de baja potencias desde 1hp, hasta máquinas de gran potencia 2250hp. Otra de las características que se debe resaltar, es que las gráficas presentan los mismos detalles de oscilación para cada uno de los casos, de tal forma que se podría llegar a decir las gráficas son iguales; esto es de suma importancia ya que, si se consideran como correctos los resultados obtenidos por Simulink, significa que el programa se encuentra validado y se puede usar para resolver posteriores ejercicios confiando en sus resultados.

Con el objetivo de reforzar la conclusión anterior se realizará como un ejercicio extra de validación, el proyecto 1, literal c, capítulo 6, del libro de simulación en Matlab de Che-mun Ong, con el objeto de obtener la simulación dinámica de una máquina de inducción de 1hp, 60hz, 4 polos, tomando en cuenta un paso mínimo de 0.0002, la máquina presenta las siguientes características

Tabla 4.5. Datos de placa máquina de 1 hp

Parámetro	Valor	Unidades
Voltaje	200	V
Potencia	1	hp
Rr	3.35	ohm
Rs	1.99	ohm
Xls	6.94	ohm
Xlr	6.94	ohm
Xm	163.73	ohm
J	0.1	kg m ²

Se debe calcular el valor del torque nominal de la máquina de inducción. En este ejercicio se realizará una variación del valor de la carga en el eje de la máquina, se comienza con un arranque sin carga al tiempo cero. Una vez estabilizada la máquina de inducción a los 3 segundos se aumentará el valor del torque mecánico de cero al cincuenta por ciento del valor nominal, hasta un tiempo de 5.5 segundos. En el siguiente intervalo se realizará un aumento del torque mecánico al cien por ciento de su valor nominal, este valor del torque tendrá una duración de 1 segundo; tiempo en el cual se regresará el valor del torque a 50% de su valor nominal, hasta llegar a un tiempo de simulación de 8 segundos. Como se muestra en la Figura 4.37.

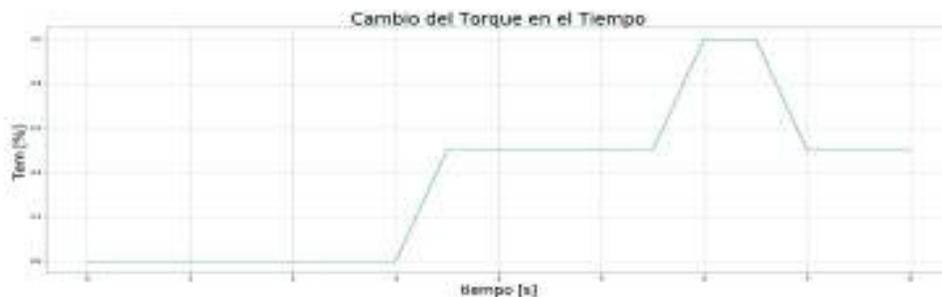
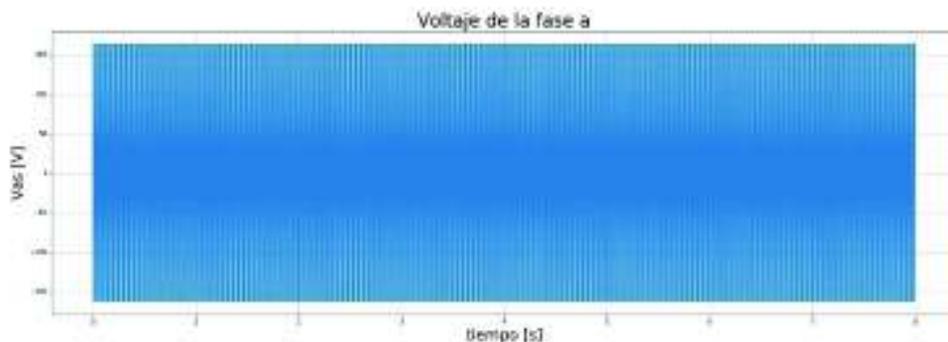


Figura 4.37. Cambio del torque en el tiempo

Voltajes de la fase a del estator (vas)

- Python



- Simulink

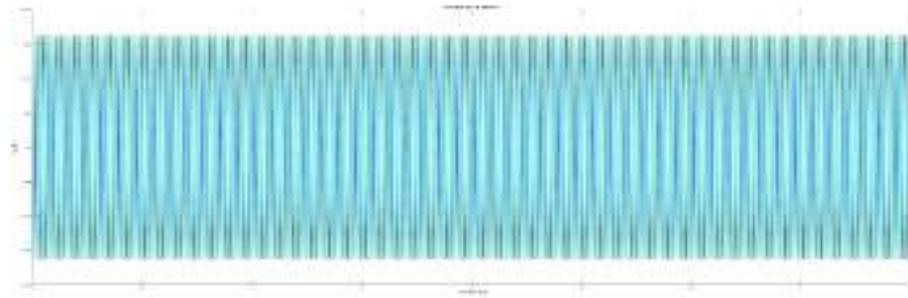
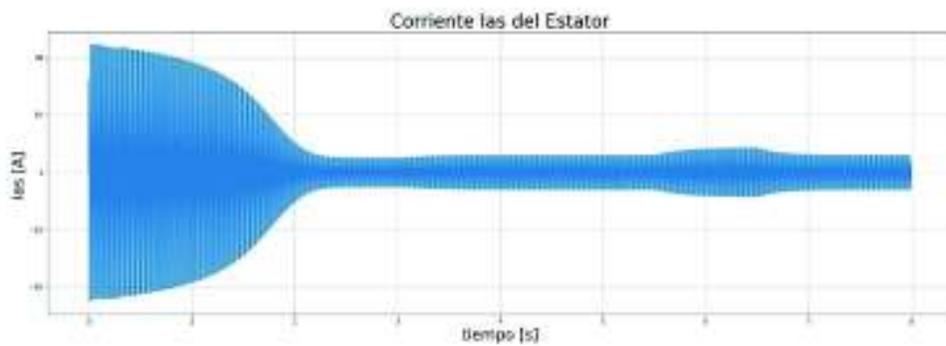


Figura 4.38. Voltaje de la fase a

Corrientes de la fase a del estator

- Python



- Simulink

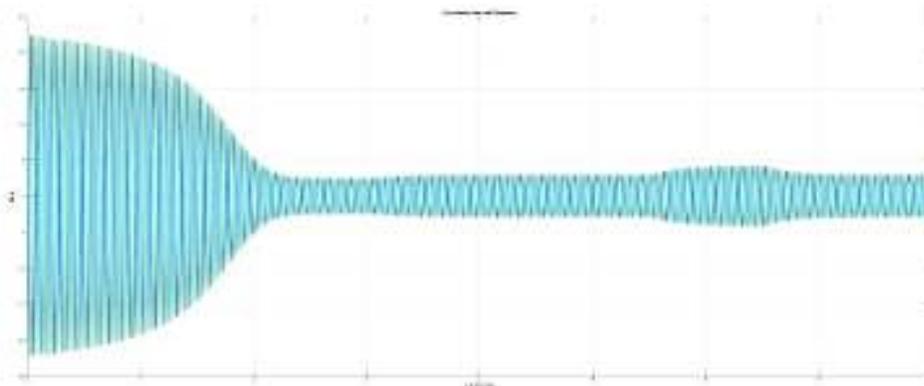
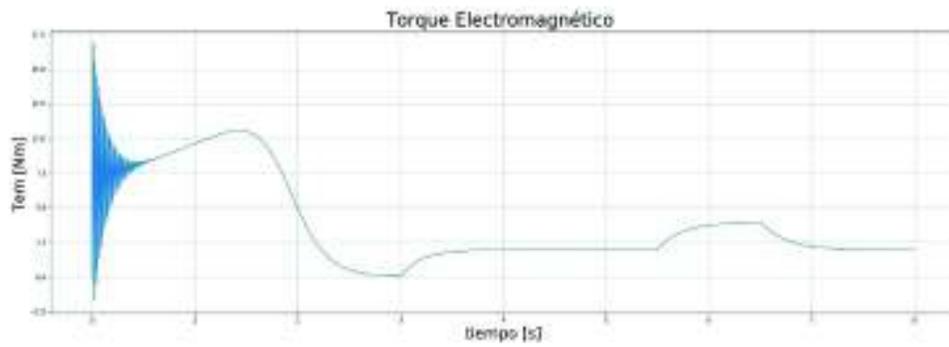


Figura 4.39. Corrientes de la fase a del estator

Torque eléctrico

- Python



- Simulink

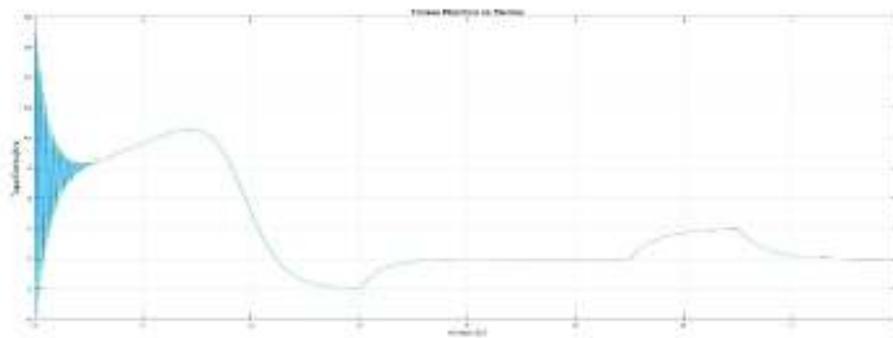


Figura 4.40. Torque eléctrico

Velocidad del rotor

- Python



- Simulink

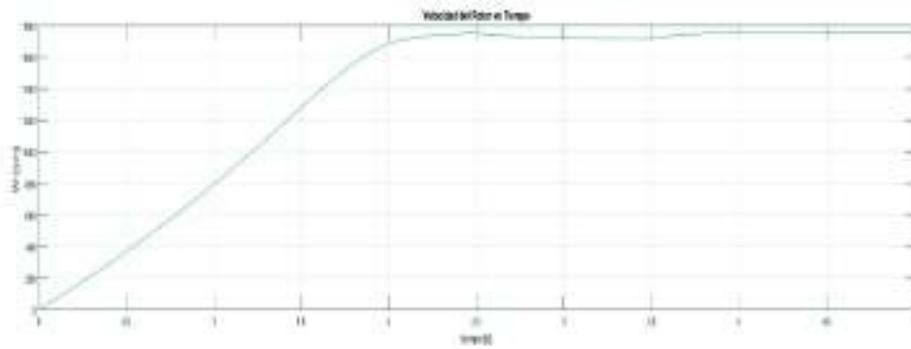


Figura 4.41. Velocidad del rotor

Una vez finalizada la comparación de los 5 ejercicios presentados y sin encontrar ninguna discrepancia entre las Figuras 4.2-4.41 obtenidas por el software de simulación Simulink y el código implementado en el lenguaje de programación Python, se puede concluir que el código desarrollado e implementado en este trabajo de titulación, cumple con las características necesarias para ser usado posteriormente en el desarrollo de ejercicios.

Tabla 4.6. Tabla de Error Absoluto Promedio y Desviación del Error Cuadrático Medio

	3 hp		50 hp		500 hp		2250 hp		1 hp	
	RMSE [U]	MAPE [%]								
ias	0.333	0.198	3.666	0.421	13.254	0.375	92.267	1.113	0.245	0.349
ibs	0.333	0.190	3.666	0.368	13.257	0.231	92.267	0.196	-	-
ics	0.332	1.430	3.666	0.226	13.257	0.276	92.267	0.225	-	-
iar	0.111	0.362	17.132	1.022	6.368	0.230	46.708	0.365	-	-
ibr	0.113	1.538	17.112	0.341	6.369	0.786	46.700	0.193	-	-
icr	0.113	0.267	17.053	0.742	6.366	0.409	46.719	0.171	-	-
Tem	0.043	0.223	2.674	0.035	6.819	0.082	44.696	0.026	0.003	0.002
w	0.116	0.000	1.862	0.002	18.000	0.000	0.067	0.000	0.078	0.000

implementado ende

En la Tabla 4.6. se presentan dos valores de error que nos permiten comparar los resultados ya no solo de forma gráfica, sino que nos permiten matemáticamente llegar a conclusiones. Debido a que cada programa usa un diferente método numérico para la solución del sistema de ecuaciones, podemos apreciar que la desviación del error cuadrático medio las maquinas mayores a 500 hp llega a las decenas de unidades, sin embargo, los errores absolutos promedio para ninguna de las maquinas, ni variables llega ha ser mayor al 2 %. Es por esto que se puede afirmar que las gráficas son similares

5. EJERCICIOS DE APLICACIÓN

5.1 Introducción

Una vez validado el programa implementado en Python, en este capítulo se realizarán simulaciones de distintos métodos de arranque para un motor de inducción, y de cortocircuitos trifásicos en bornes para dos motores de inducción. El objetivo de este capítulo es demostrar la versatilidad y la facilidad con la cual se pueden realizar varios casos de simulación. Los códigos para cada ejercicio se adjuntan en el ANEXO IV (digital).

5.1 Métodos de arranque

Hoy en día los motores de inducción, debido a su bajo costo de mantenimiento y a que son muy robustos, son muy demandados por la industria, al igual que los motores sincrónicos. Estos necesitan de una gran corriente de arranque que generalmente es 6 u 8 veces mayor que la corriente nominal; para máquinas muy grandes esta corriente de arranque puede llegar a ser 10 veces mayor a la corriente nominal. La corriente de arranque puede causar daños a los equipos adyacentes al motor e incluso al mismo motor, los daños internos del motor se dan por los esfuerzos electromagnéticos en los devanados de este, los cuales pueden desgastar a los conductores y en casos extremos pueden llegar a cambiar la geometría de la máquina.

En el caso de equipos electrónicos es muy probable que éstos se apaguen dejando a las máquinas sin control o incluso sacándolas de servicio. Si los equipos son computadoras, servidores, etc., se corre el riesgo de pérdida de información.

Los procesos dentro de la industria podrían ser continuos o permanentes. Los permanentes son aquellos en los que se necesita arrancar una sola vez el motor; en los procesos continuos se necesitará arrancar el motor de forma constante.

Dejando a un lado el proceso para el cual se vaya a usar el motor, éste siempre tendrá un sistema de arranque, con el objetivo de disminuir la corriente de arranque. Existen varios métodos a través de los cuales se puede disminuir esta corriente, el método más utilizado y sencillo se ejecuta mediante la reducción del voltaje al arranque.

A continuación, se realizará un análisis sobre los beneficios de implementar diferentes métodos de arranque para un motor de inducción, para realizar este trabajo se usará la máquina 1, en la Tabla 5.1. de la sección anterior, trifásica de 4-polos diseñada para operar a una frecuencia de $60 H_z$.

Tabla 5.1. Máquina de 3hp

Parámetro	Valor	Unidades
Voltaje	220	V
Potencia	3	hp
Rr	0.435	ohm
Rs	0.816	ohm
Xls	0.754	ohm
Xlr	0.754	ohm
Xm	26.13	ohm
J	0.089	$kg\ m^2$
I nominal	12.33	A

El objetivo de este trabajo es comparar los valores pico de las corrientes y el torque al momento del arranque del motor de inducción, para varios métodos, y poder apreciar la disminución en cada uno de los casos.

5.1.1 Arranque directo

El arranque directo será el caso de referencia. Este método de arranque no requiere de ningún elemento que permita reducir la corriente de arranque y consiste en conectar la máquina de forma directa a la red y arrancarla.

- Corriente de la fase a del estator

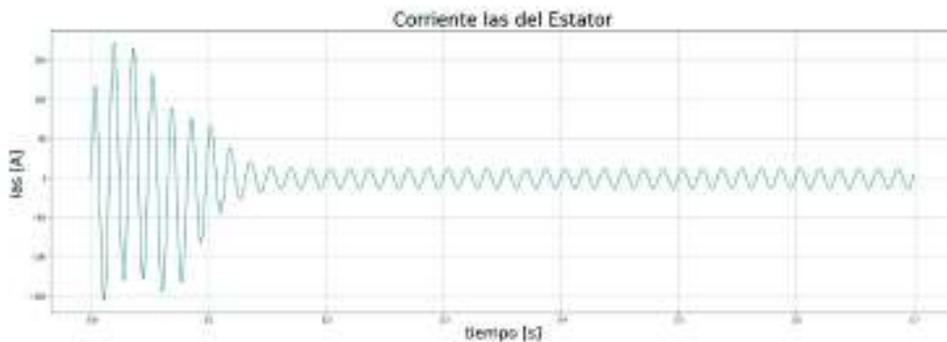


Figura 5.1. • Corriente de la fase a del estator

- Corriente de la fase *b* del estator

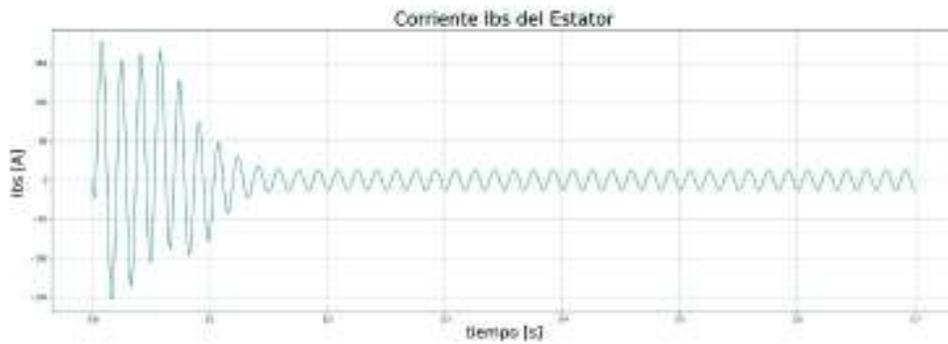


Figura 5.2. • Corriente de la fase *b* del estator

- Corriente de la fase *c* del estator

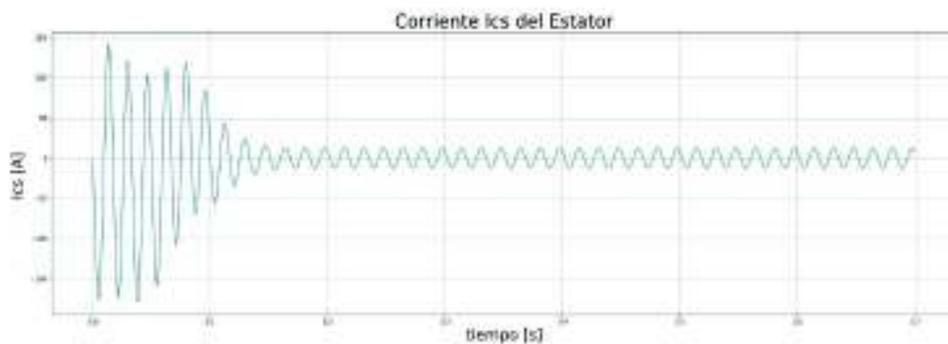


Figura 5.3. • Corriente de la fase *c* del estator

En las Figura 5.1.-5.3. la corriente de arranque propia del motor de inducción, esta corriente depende únicamente de la impedancia de la máquina y llega a valores de 172.22 A.; esta corriente en comparación con la corriente nominal de la máquina es 14 veces más grande, por lo cual, es importante reducir su valor implementando los métodos de arranque.

- Corriente de la fase a del rotor

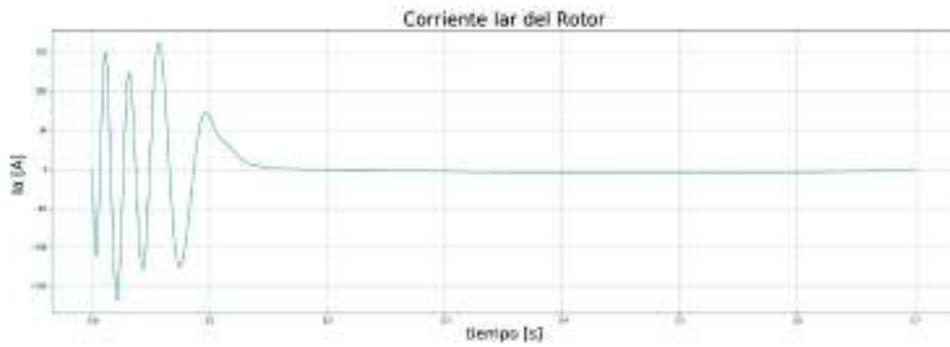


Figura 5.4. • Corriente de la fase a del rotor

- Corriente de la fase b del rotor

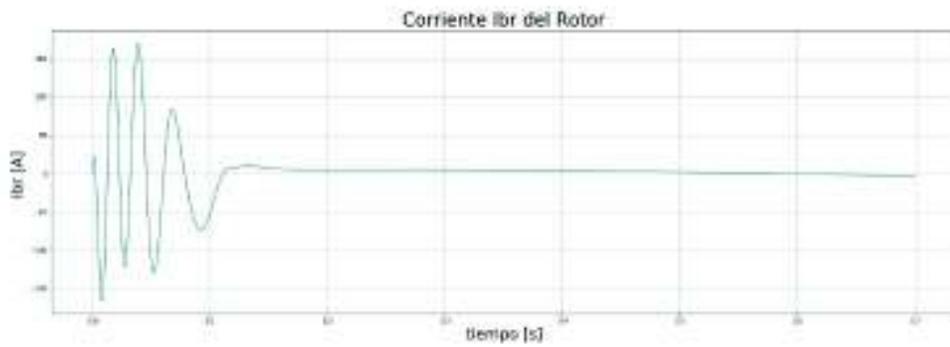


Figura 5.5. • Corriente de la fase b del rotor

- Corriente de la fase c del rotor

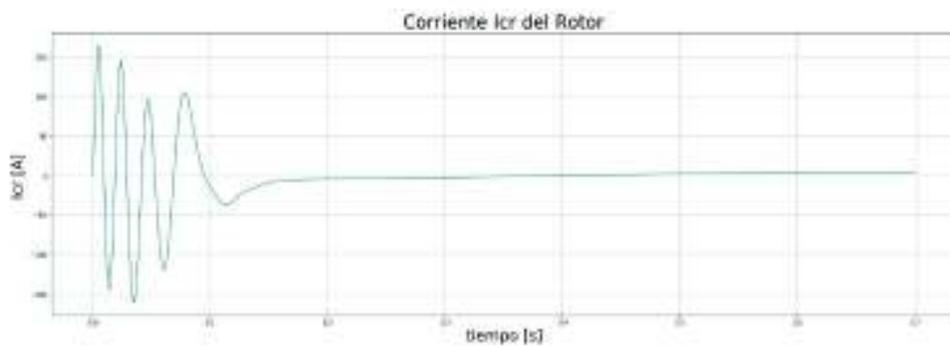


Figura 5.6. • Corriente de la fase c del rotor

En las Figuras 5.4.-5.6. la corriente en el estator se ve reflejada al rotor en su pico más alto casi en una relación de 1:1 el valor máximo de corriente del estator es de 161.94 A, el cual también se debe reducir para evitar esfuerzos en los conductores.

- Torque eléctrico

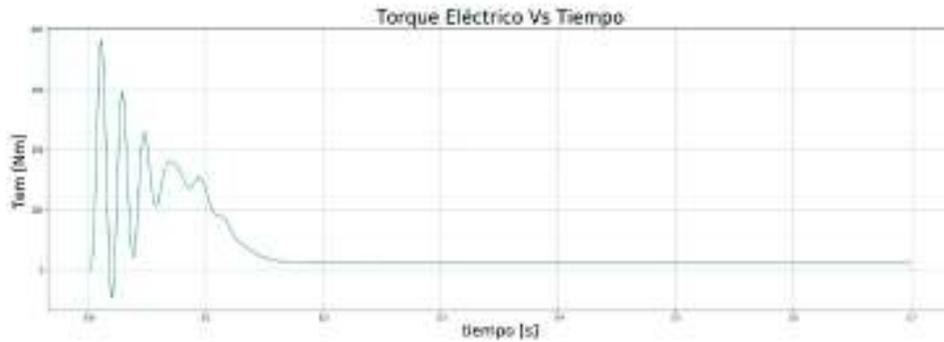


Figura 5.7. Torque eléctrico

En la Figura 5.7. el torque de arranque necesario para que el motor de inducción es de 381.91 Nm y se estabiliza una vez con un valor de 11.72 Nm, es decir que el torque eléctrico del motor en el arranque es 32 veces más que el torque eléctrico a plena carga.

- Velocidad de rotor

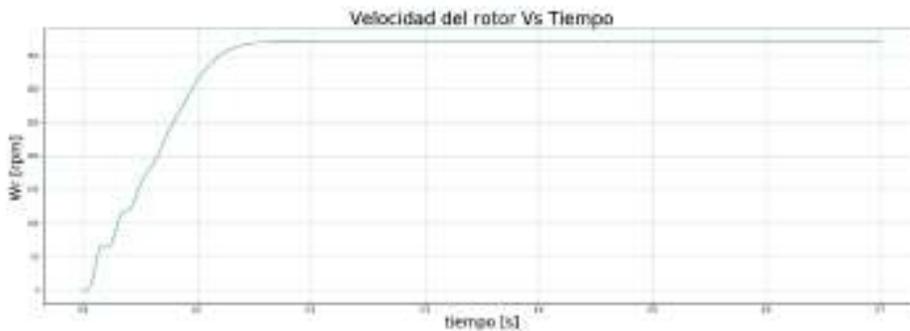


Figura 5.8. Velocidad del rotor

En la Figura 5.8. la velocidad angular crece rápidamente hasta llegar a la velocidad nominal, una vez que la velocidad del rotor alcanza el valor nominal se da por finalizado el arranque. Para el caso del motor de inducción el arranque directo tiene una duración aproximada de dos décimas de segundo.

- Tabla de resultados

En la Tabla 5.2. se detallan los valores pico de corriente para cada una de las fases del motor de inducción tanto para el rotor como para el estator, además de, los resultados pico para el torque electromecánico.

Tabla 5.2. Resultados arranque directo

	Fases			Unidades
	A	B	C	
Corriente del estator	172.22	176.62	141.05	A
Corriente del rotor	161.94	170.11	165.58	A
Torque eléctrico	381.91			Nm

5.1.2 Arranque estrella delta

Este es un método de arranque de voltaje reducido en el cual se aprovecha la relación que existe entre los voltajes de línea y los voltajes de fase. El método consiste en arrancar el motor con su voltaje de línea y una vez estabilizado el motor, se aplica el voltaje de fase.

Este método presenta ciertas limitaciones ya que solo se puede emplear en motores de inducción que sean trifásicos, porque se aprovecha la relación de $\sqrt{3}$ entre el voltaje de fase y el voltaje de línea. Además, el motor de inducción debe estar diseñado para trabajar en configuración delta, la mayor limitación que presenta este método es la necesidad de tener acceso a los 6 terminales de los bobinados del estator.

Una vez que se cumplen con todas las condiciones anteriores, el arranque se realiza mediante un conjunto de contactores que permiten cambiar en un tiempo determinado la conexión de los devanados de estrella a triángulo.

- Corriente de la fase *a* del estator

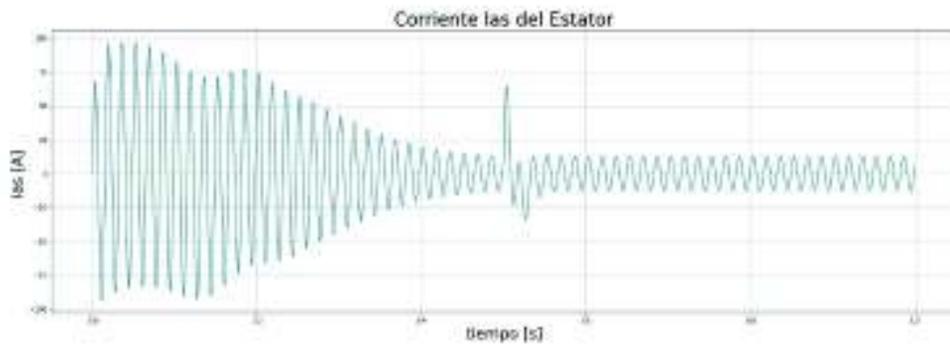


Figura 5.9. • Corriente de la fase *a* del estator

- Corriente de la fase *b* del estator

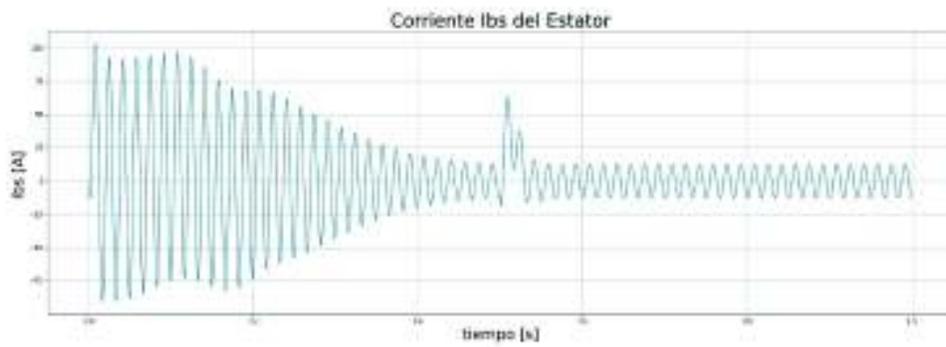


Figura 5.10. Corriente de la fase *b* del estator

- Corriente de la fase *c* del estator

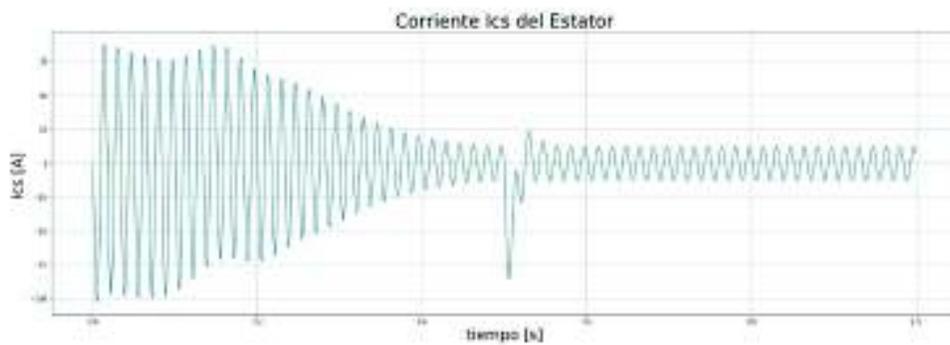


Figura 5.11. Corriente de la fase *c* del estator

Como se puede observar en las Figuras 5.9.–5.11., las corrientes del estator con el voltaje de línea oscilan aproximadamente 0.4 segundos y, llegan a un valor de 96.47 A, el cual es 8 veces mayor a la corriente nominal; una vez estabilizada la máquina se realiza el cambio de conexión de estrella a delta, este aumento de voltaje produce un pico de 80 A, con una duración menor a 0.1 segundos.

- Corriente de la fase *a* del rotor

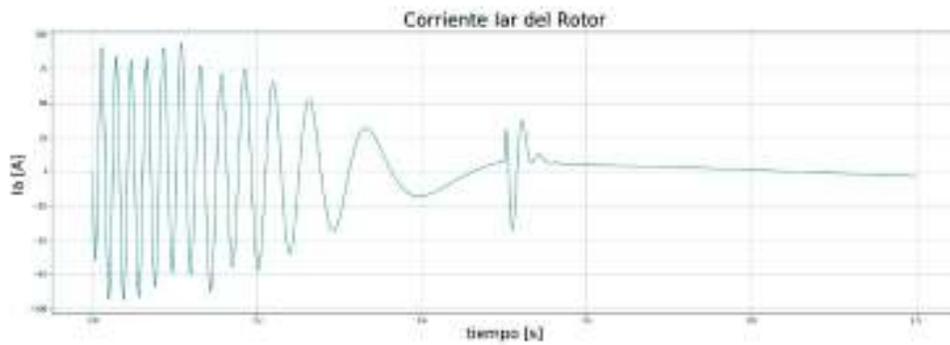


Figura 5.12. Corriente de la fase *a* del rotor

- Corriente de la fase *b* del rotor



Figura 5.13. Corriente de la fase *b* del rotor

- Corriente de la fase c del rotor



Figura 5.14. Corriente de la fase c del rotor

En las Figuras 5.12.-5.14. como en el caso del arranque directo la corriente del estator refleja en el inductor una relación aproximada de 1:1 y presenta las mismas oscilaciones debido al cambio de voltaje.

- Torque eléctrico

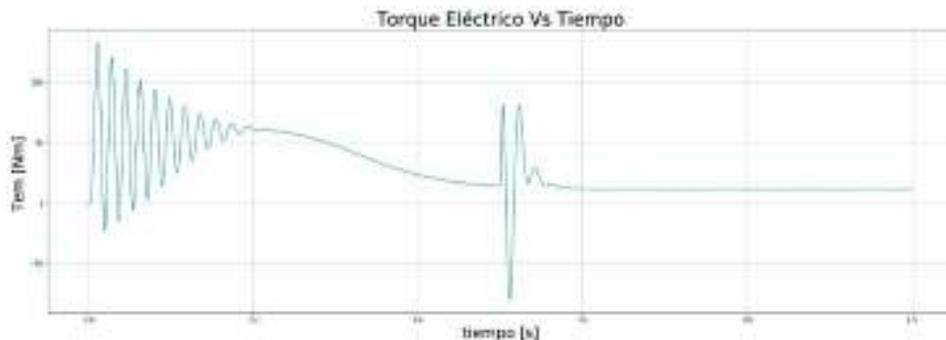


Figura 5.15. Torque eléctrico

En la Figura 5.15. el torque eléctrico en el arranque con la conexión estrella también se reduce por la reducción del voltaje, a un valor de 132.71 Nm, el cual es 11 veces mayor al torque a plena carga. En la Figura 5.15. anterior se puede observar que el torque eléctrico, al momento de cambiar la conexión de estrella a delta, cambia su sentido y la máquina pasa de motor a generador por un período de tiempo mucho menor a una décima de segundo. El valor pico de torque que genera la máquina se aproxima a los 60 Nm.

- Velocidad de rotor

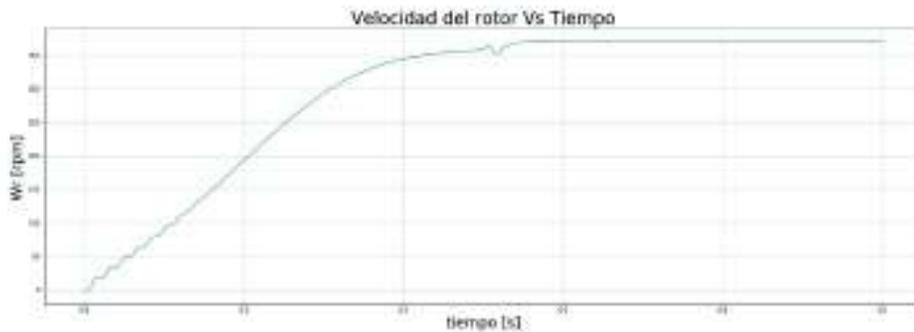


Figura 5.16. Velocidad del rotor

En la Figura 5.16. el arranque directo alcanza su velocidad nominal en un tiempo aproximado de dos décimas de segundo, sin embargo, de la gráfica anterior se puede observar que el arranque estrella delta tarda 3 veces más ya que se debe esperar que la máquina esté prácticamente estabilizada para cambiar la conexión.

- Tabla de resultados

En la Tabla 5.3. se detallan los valores pico de corriente para cada una de las fases del motor de inducción tanto para el rotor como para el estator, además de los resultados pico para el torque electromecánico.

Tabla 5.3. Resultados arranque estrella delta

	Fases			Unidades
	A	B	C	
Corriente del estator	96.47	102.87	87.72	<i>A</i>
Corriente del rotor	93.9	95.61	95.6	<i>A</i>
Torque eléctrico	132.71			<i>Nm</i>

5.1.3 Arranque con voltaje variable

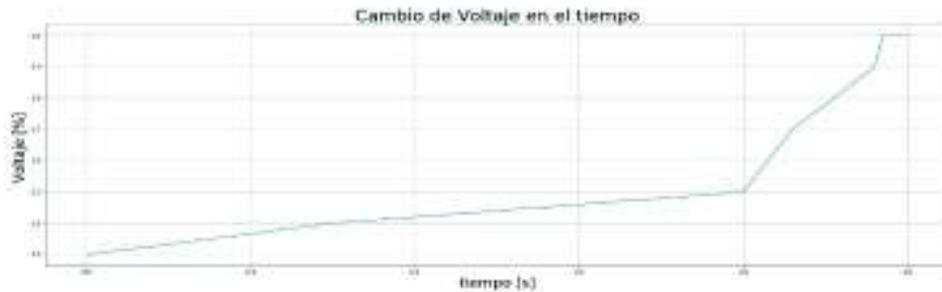
El método de arranque con voltaje variable o por medio de un auto transformador, tiene el mismo principio del método estrella delta, el cual es reducir el voltaje al inicio del arranque y de esta forma reducir el valor de la corriente de arranque.

Una de las ventajas de este método con respecto al método anterior es que no existen limitaciones o condiciones necesarias para su implementación, como las que se tiene en el método de arranque estrella delta, esto se debe a que no se cambia la conexión de los devanados estáticos de la máquina de inducción, sino que se hace uso de un auto transformador para realizar el control del voltaje. Este método de arranque es mucho más costoso que el método de arranque estrella delta ya que implica adquirir un autotransformador lo que incrementará el costo del conjunto.

La metodología implementada para este arranque es la siguiente: se aplica el 30% del voltaje nominal al inicio del arranque, una vez estabilizado el motor de inducción se realiza un incremento de 10% en el valor del voltaje, se espera que la máquina se estabilice nuevamente y se repite el proceso hasta llegar a un valor nominal de voltaje.

Para este arranque se ingresó el siguiente vector de pasos:

[0.3, 0.5, 0.03, 0.03, 0.05, 0.05, 0.01, 1]



- Corriente de la fase a del estator

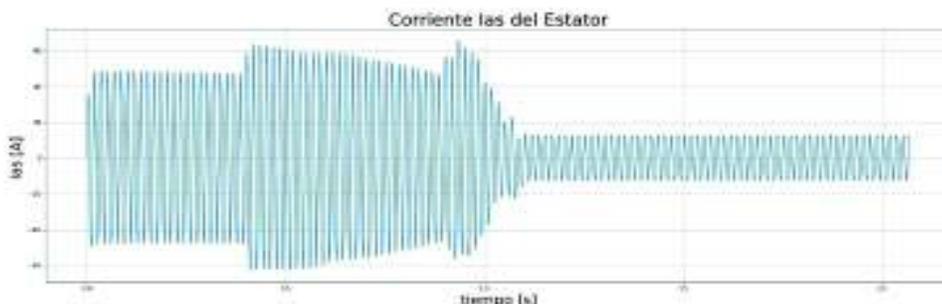


Figura 5.17. Corriente I de la fase a del estator

- Corriente de la fase *b* del estator

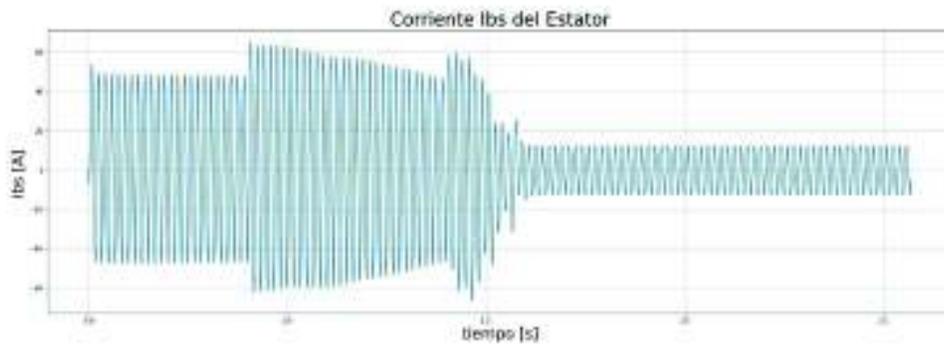


Figura 5.18. Corriente de la fase *b* del estator

- Corriente de la fase *c* del estator

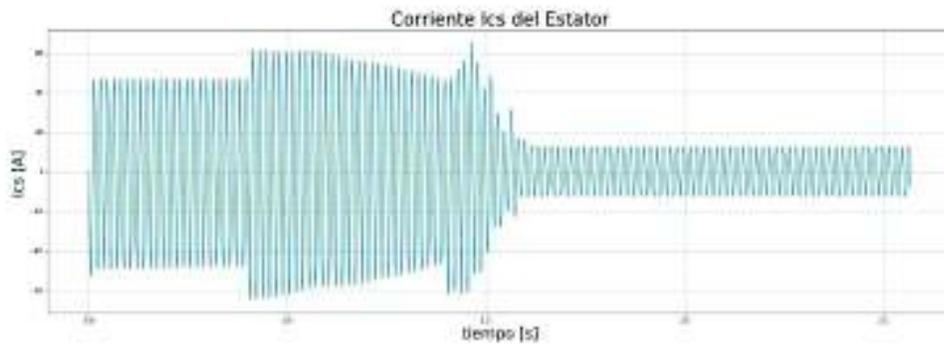


Figura 5.19. Corriente de la fase *c* del estator

En las Figuras 5.17.-5.19 la corriente del estator al momento del arranque comienza con un valor aproximado de 40 A; conforme se va incrementando el voltaje mediante pasos, el motor de inducción oscila nuevamente, por lo cual la corriente aumenta y presenta picos cada vez mayores. De las gráficas anteriores se aprecia que el mayor valor de los picos de corriente es de 65.57 A, el cual es 5 veces mayor a la corriente nominal.

- Corriente de la fase *a* del rotor

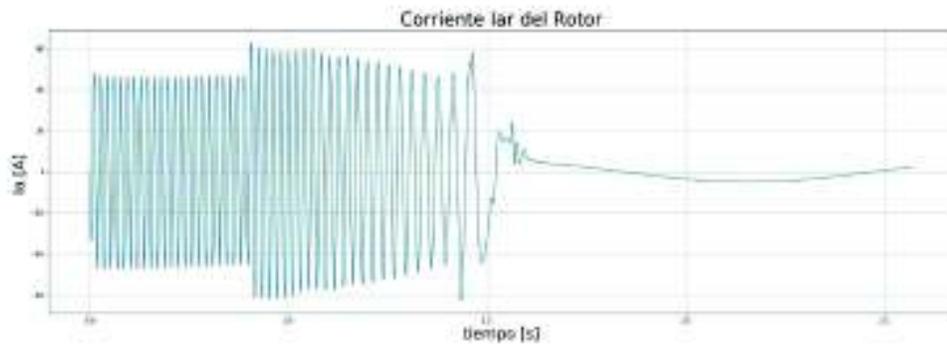


Figura 5.20. Corriente de la fase *a* del rotor

- Corriente de la fase *b* del rotor

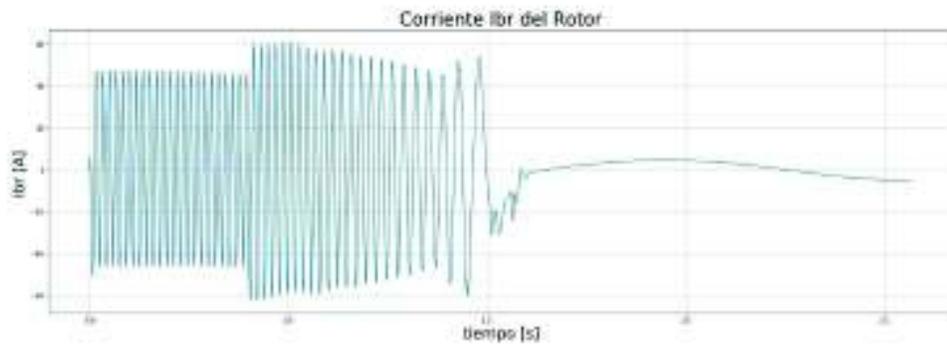


Figura 5.21. Corriente de la fase *b* del rotor

- Corriente de la fase *c* del rotor

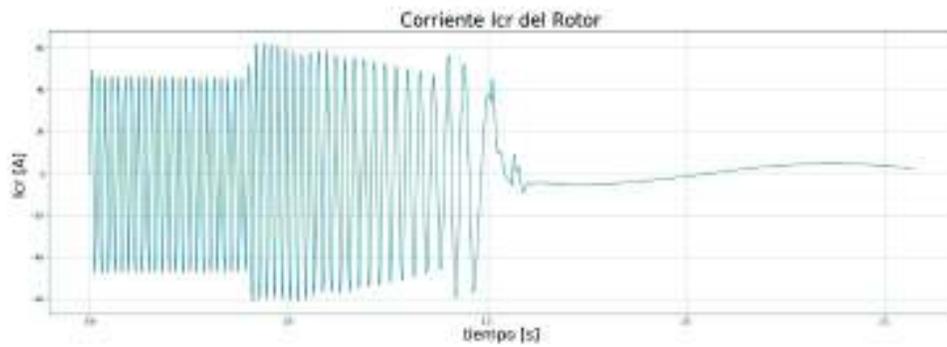


Figura 5.22. Corriente de la fase *c* del rotor

En las Figuras 5.20.-5.22. al igual que en los casos anteriores la corriente del rotor oscila similarmente a la corriente del estator llegando a un valor pico de 62.94 A, sin embargo, en este caso el pico máximo de corriente no se produce en la misma oscilación, ya que en el estator se produce aproximadamente a los 0.9 segundos mientras que en el rotor se produce a los 0.4 segundos.

- Torque eléctrico

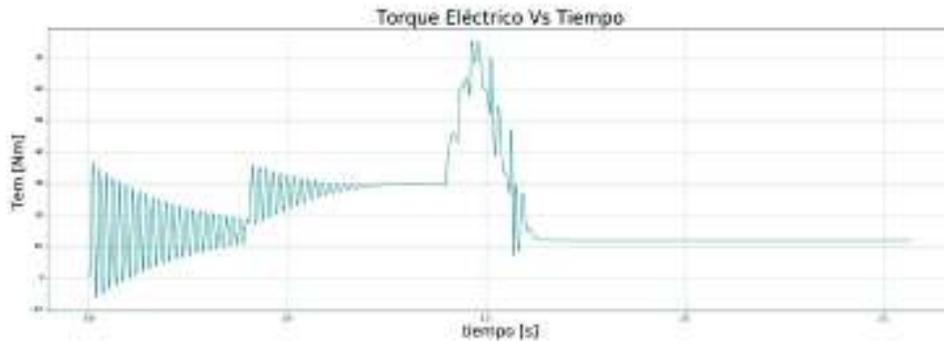


Figura 5.23. Corriente del torque eléctrico

En la Figura 5.23. el valor máximo al cual llega el torque eléctrico en este método de arranque es 75 Nm, el cual es 6 veces mayor al torque en plena carga, de la gráfica anterior se puede ver que el torque oscila y cuando éste se reduce, se aumenta el voltaje y se produce una nueva oscilación. Del vector de pasos seleccionado, se puede apreciar que el arranque comienza con cambios de voltaje distanciados unos de otros, mientras que los cambios de voltaje finales son rápidos, esto se realiza de esta manera para evitar que el torque, una vez arrancada la máquina, tome valores negativos.

- Velocidad de rotor

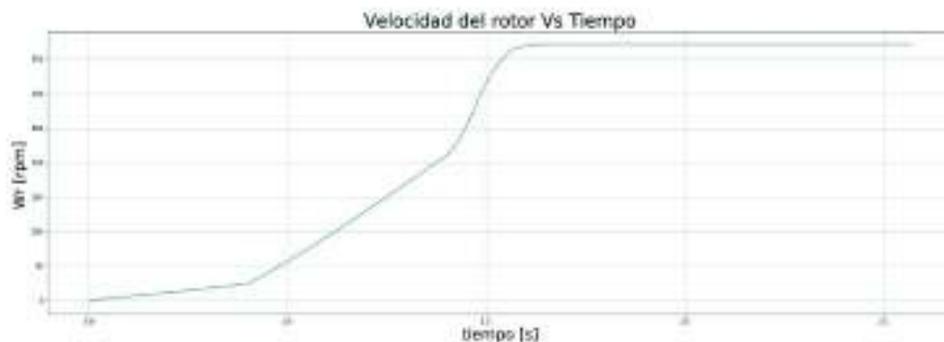


Figura 5.24. Velocidad del rotor

De la Figura 4.24. anterior se observa como la velocidad de la máquina es dependiente del voltaje de energización. Cuando el voltaje tiene un valor de 30% del voltaje nominal, la pendiente de la curva de la velocidad es baja, a medida que el voltaje va aumentando está pendiente crece hasta asemejarse a una forma exponencial.

- Tabla de resultados

En la Tabla 5.4. se detallan los valores pico de corriente para cada una de las fases del motor de inducción, tanto para el rotor como para el estator, además, de los resultados pico para el torque electromecánico.

Tabla 5.4. Resultados arranque con voltaje variable

	Fases			Unidades
	A	B	C	
Corriente del estator	65.57	65.04	65.25	A
Corriente del rotor	62.94	61.03	62.23	A
Torque eléctrico	75.11			Nm

5.1.4 Arranque mediante resistencias en el rotor

Cuando se energiza un motor de inducción se produce el torque máximo o necesario para su arranque. Este torque genera la corriente de arranque, por lo cual al disminuir el valor del torque de arranque se disminuye el valor de la corriente de arranque. En los métodos anteriores se ha logrado esta reducción controlando el voltaje que se aplica al motor, otra forma de hacerlo es controlar el desplazamiento en la máquina, para esto se usan las siguientes ecuaciones:

$$R_r = \frac{\sqrt{r_{th}^2 + (x_{th} + (x_{th} + X_{ls})^2)}}{s_{max}}$$

Ecuación 5.1. Resistencia del rotor en función del desplazamiento máximo

Para obtener el desplazamiento máximo de la Ecuación 5.1 es necesaria la Ecuación 5.2. que permita determinar la inductancia y la resistencia de Thévenin:

$$Z_{th} = \frac{jX_m(R_s + jX_{ls})}{R_s + j(X_{ls} + X_m)}$$

Ecuación 5.2. Impedancia de Thévenin

De la Ecuación 5.1 se puede determinar el valor de la resistencia del rotor con el cual se produce un deslizamiento máximo. La metodología aplicada consiste en conectar un grupo de resistencias en serie a los bobinados del rotor, se arranca el motor con un valor de resistencia que produzca un deslizamiento máximo de 0.8 y una vez estabilizada la máquina se cierra un contactor cortocircuitando una parte del grupo de resistencias, de tal forma que ahora el deslizamiento máximo tenga un valor de 0.4. Al igual que en el paso anterior, una vez estabilizada la máquina se cortocircuita todo el grupo de resistencias y se opera con el motor en condiciones normales.

- Corriente la fase *a* del estator

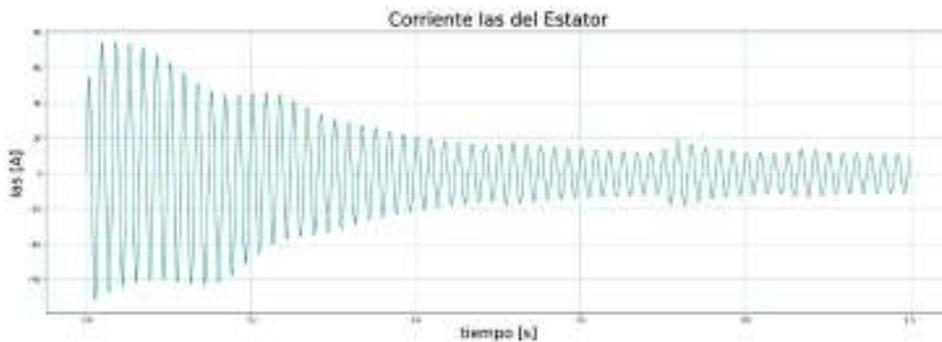


Figura 5.25. Corriente la fase *a* del estator

- Corriente la fase *b* del estator

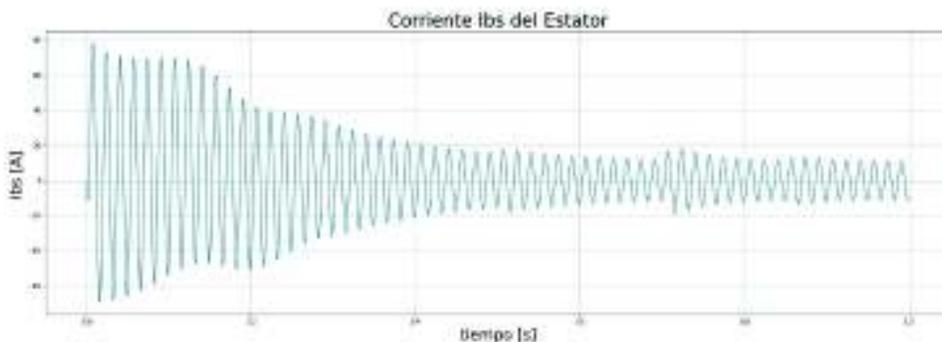


Figura 5.26. Corriente la fase *b* del estator

- Corriente la fase c del estator

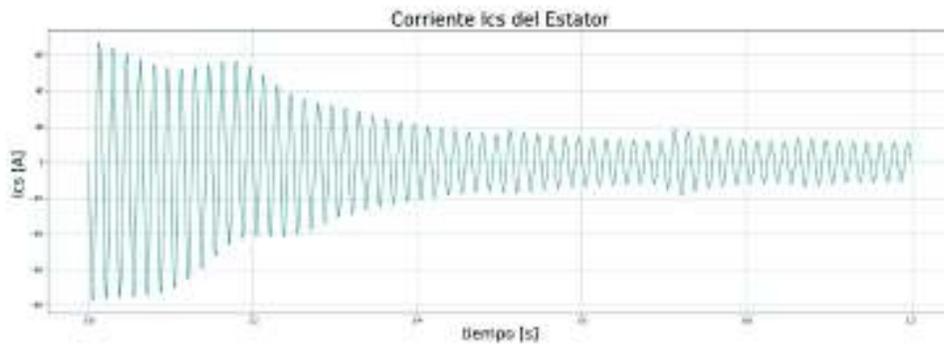


Figura 5.27. Corriente la fase c del estator

De las Figuras 5.25.- 5.27. la corriente de arranque para las fases en este método no oscila bruscamente, por lo cual es un arranque suave en comparación a los métodos anteriores. El paso de resistencia del primer grupo al segundo, que se produce a los 0.5 segundos, es casi imperceptible y la máquina no oscila en este intervalo. De la misma forma, los otros pasos no producen oscilaciones en la máquina por lo cual los esfuerzos se reducen. El pico máximo de corriente se produce al inicio del arranque y tiene un valor de 74.15 A, 6 veces la corriente nominal.

- Corriente de la fase a del rotor

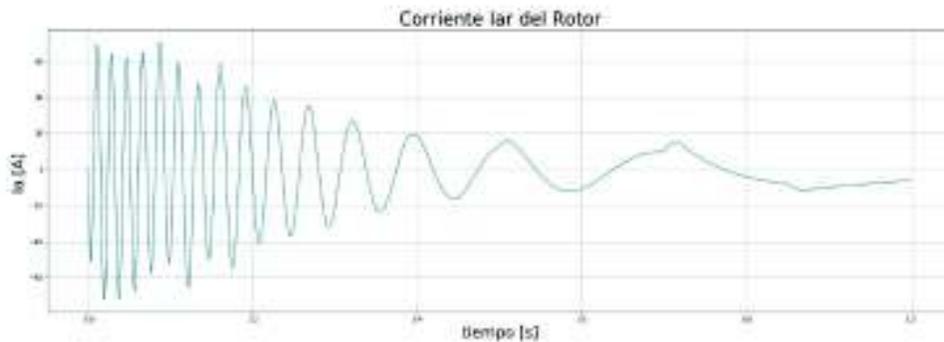


Figura 5.28. Corriente de la fase a del rotor

- Corriente de la fase *b* del rotor

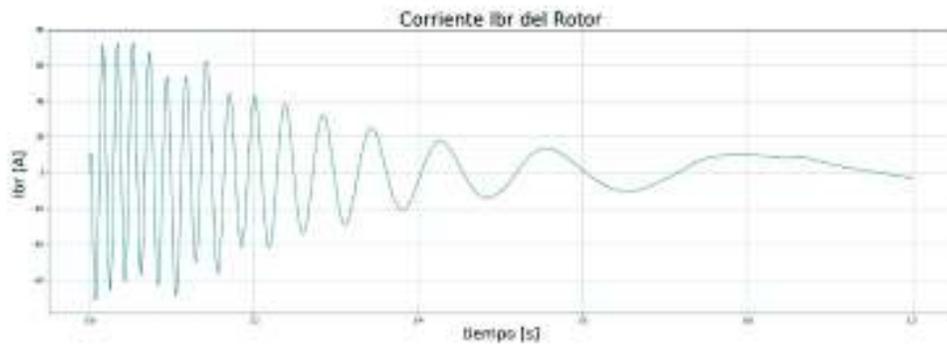


Figura 5.29. Corriente I de la fase *b* del rotor

- Corriente de la fase *c* del rotor

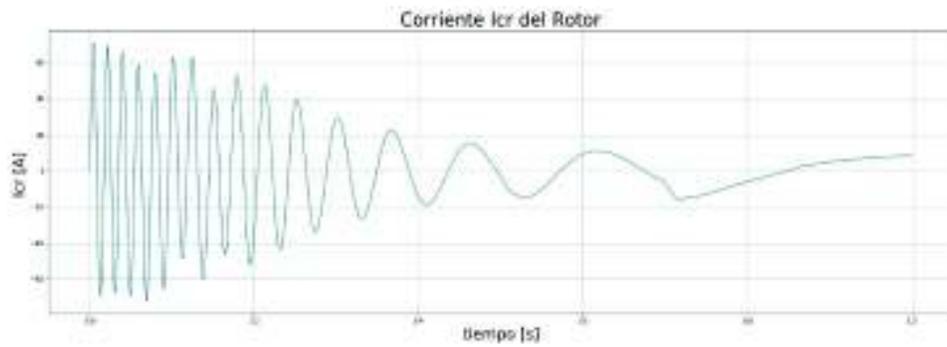


Figura 5.30. Corriente de la fase *c* del rotor

- Torque eléctrico:

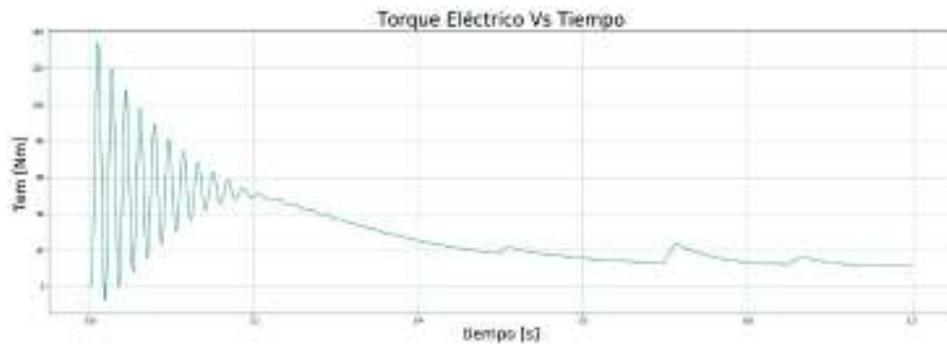


Figura 5.31. Torque eléctrico

De la Figura 5.31. del torque eléctrico se puede afirmar que este arranque es suave para el motor de inducción. Al comienzo del arranque el torque toma un valor máximo de 133.95 Nm, y luego oscila por dos décimas de segundo y comienza a estabilizarse. Cada cambio de resistencia en el rotor produce un pequeño aumento en el torque, sin embargo, el motor de inducción no oscila.

- Velocidad de rotor

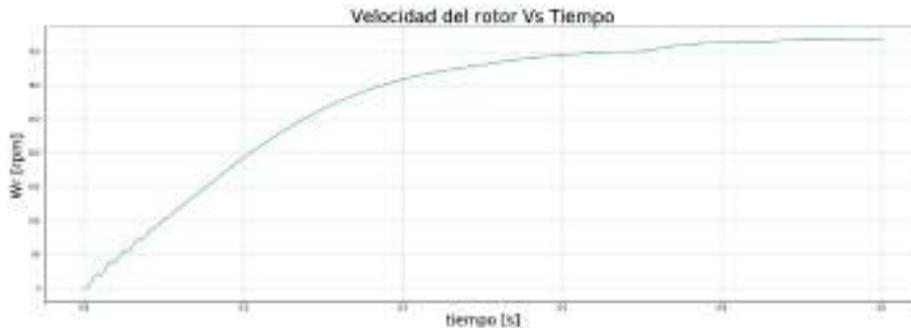


Figura 5.32. Velocidad del rotor

En la Figura 5.32. el aumento de la velocidad del motor es constante hasta llegar a estabilizarse, muy similar al de un arranque directo. La diferencia que resalta es el tiempo de arranque con este método ya que el arranque dura un segundo, mientras que en un arranque directo se demora 0.2 segundos.

- Tabla de resultados

En la Tabla 5.5. se detallan los valores pico de corriente para cada una de las fases del motor de inducción tanto para el rotor como para el estator, además de los resultados pico para el torque electromecánico.

Tabla 5.5. Resultados arranque con resistencias en el rotor

	Fases			Unidades
	A	B	C	
Corriente del estator	74.15	77.87	66.98	A
Corriente del rotor	70.76	72.7	70.86	A
Torque eléctrico	133.95			Nm

Tabla 5.6. Resultados de los diferentes métodos de arranque

Arranque	Corriente			Torque		
	Arranque	Nominal	Relación	Arranque	Nominal	Relación
Directo	172.22	12.33	14.0	381.91	11.72	32.6
Estrella Delta	96.47	12.33	7.8	132.72	11.72	11.3
Voltaje Variable	65.57	12.33	5.3	75.11	11.72	6.4
Resistencias	74.15	12.33	6.0	133.95	11.72	11.4

Como se observa en la Tabla 5.6. la relación entre la corriente de arranque y el valor nominal puede llegar a ser 14 para el arranque directo y 5 para el arranque de voltaje variable. Aunque el propósito de este trabajo no es realizar un estudio de arranque, se puede llegar a la conclusión de que el arranque con voltaje variable presenta la menor relación entre todos los arranques realizados.

5.2 Cortocircuito trifásico en terminales

El comportamiento dinámico de un motor de inducción ante un cortocircuito trifásico en los terminales de la máquina se presentará en la siguiente sección. Para esta simulación se emplearán las máquinas trifásicas de 3 hp y 2250 hp, las cuales inicialmente estarán operando en condiciones normales, a plena carga; es decir, con un torque igual al de base.

La simulación de una falla trifásica en los terminales del motor de inducción se realizará configurando los voltajes V_a , V_b , V_c con un valor de cero voltios; se esperará una décima de segundo (aproximadamente 6 ciclos) y se restaurarán los voltajes al valor nominal.

Los datos de placa para las máquinas son:

Tabla 5.6. Datos de placa máquina 3hp

Parámetro	Valor	Unidades
Voltaje	220	V
Potencia	3	hp
R_r	0.435	ohm
R_s	0.816	ohm
X_{ls}	0.754	ohm
X_{lr}	0.754	ohm
X_m	26.13	ohm
J	0.089	$kg\ m^2$

Tabla 5.7. Datos de placa máquina 2250 hp

Parámetro	Valor	Unidades
Voltaje	2300	V
Potencia	2250	hp
R_r	0.029	ohm
R_s	0.022	ohm
X_{ls}	0.226	ohm
X_{lr}	0.226	ohm
X_m	13.04	ohm
J	63.87	$kg\ m^2$

Para la simulación de los cortocircuitos trifásicos se toma como referencia la corriente de la fase a del motor; una vez que la corriente pase, aproximadamente, por cero. Se realiza el cambio de voltaje antes mencionado con una duración de una décima de segundo; luego se restablecen los voltajes al valor nominal.

5.2.1 Máquina de 3 hp

- Corriente de la fase a

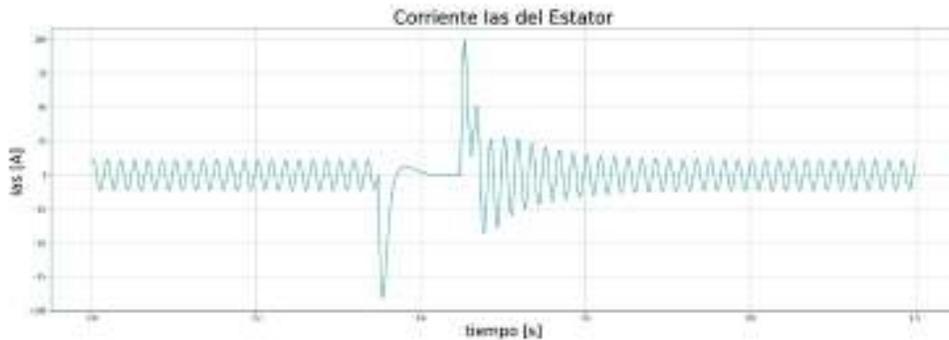


Figura 5.33. Corriente de la fase a del estator

En la Figura 5.33. la corriente de la fase a es sinusoidal, con una frecuencia de 60 *Hz*. En condiciones normales, una vez que esta corriente cruza por cero, se produce el cortocircuito, el cual alcanza una corriente de 99 A. Luego esta corriente disminuye hasta llegar a cero y la máquina vuelve a sus valores nominales de voltaje, lo cual produce un nuevo pico de 100 A; la máquina oscila y no pierde estabilidad.

- Torque eléctrico

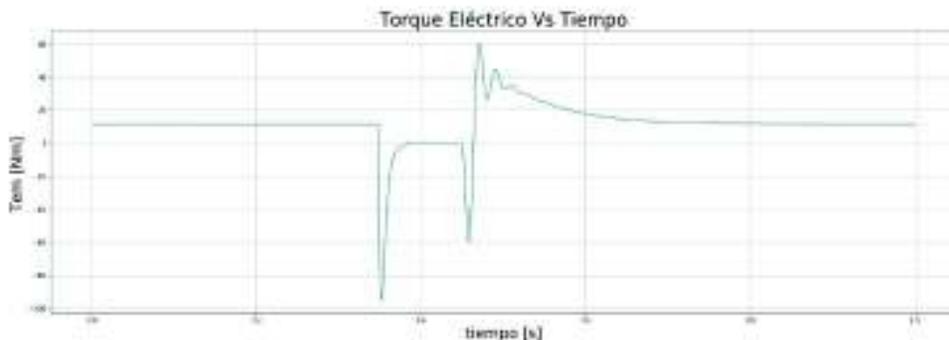


Figura 5.34. Torque eléctrico

De la Figura 5.34. anterior, se puede ver que el torque comienza en un valor constante, debido que el motor se encuentra operando en estado estable. Una vez que ocurre el

cortocircuito, el valor del torque desciende negativamente, y se estabiliza en cero. Una vez que la falla se despeja, el torque electromagnético comienza a oscilar, pero por un período de tiempo menor al del arranque, y se estabiliza en el valor inicial.

- Velocidad del rotor

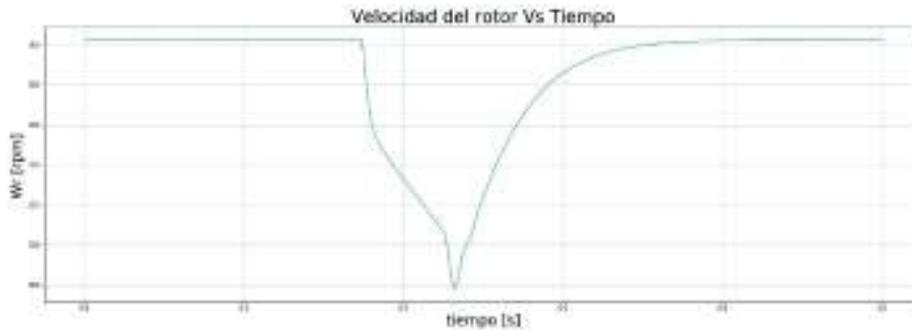


Figura 5.35. Velocidad del rotor

De la Figura 5.35. anterior puede observarse que la velocidad de la máquina durante la falla del motor comienza a frenarse; sin embargo, su velocidad no llega a cero; por esta razón, las variables del torque y de la corriente no oscilan por un mucho tiempo. Una vez despejada la falla, el motor acelera y restablece su velocidad inicial.

5.2.2 Máquina de 2250 hp

- Corriente de la fase a del estator

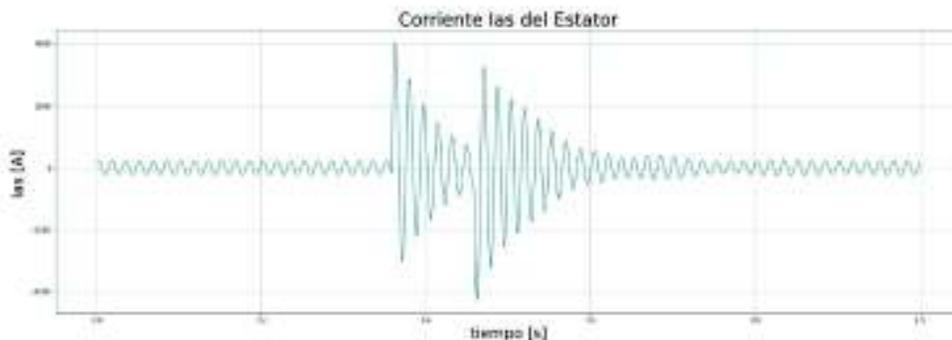


Figura 5.36. Corriente de la fase a del estator

En la Figura 5.36. la corriente de la máquina de 2250 hp no llega a un valor de cero antes que la falla sea despejada, sino que oscila durante toda la simulación. Se puede también

apreciar que, al despejar la falla, esta comienza a arrancar con un valor pico muy cercano al valor de cortocircuito.

- Torque eléctrico

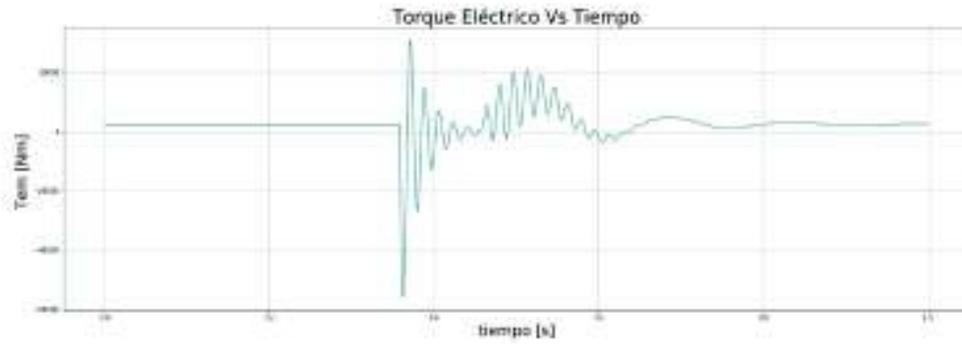


Figura 5.37. Torque eléctrico

En la Figura 5.37. el torque eléctrico oscila alrededor de 0.6 segundos. Después de este tiempo, el valor del torque pasa por cero debido a que la velocidad del rotor supera por un instante la velocidad sincrónica, produciendo que éste se frene.

- Velocidad del rotor



Figura 5.39. Velocidad del rotor

En la Figura 5.39. la velocidad del rotor comienza a descender durante la falla; al momento del despeje la velocidad descende 50 rpm; una vez restablecido el voltaje la velocidad aumenta y supera la sincrónica con un valor de 20 rpm.

6. CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusiones

- Se implementó una herramienta computacional, en el lenguaje de programación Python, que simula el comportamiento de una máquina de inducción mediante la resolución de las ecuaciones diferenciales en el tiempo.
- La herramienta computacional implementada es capaz de obtener los diferentes valores de varias variables de la máquina de inducción en el tiempo, tanto del estator como del rotor; estas variables son el resultado de la solución en el tiempo de las ecuaciones diferenciales de la máquina.
- Se logró simular el comportamiento dinámico de una máquina de inducción en el dominio del tiempo; esto se logró mediante una simulación por tramos de tiempo haciendo uso del vector de condiciones iniciales. Esta última parte es compleja de implementar en Simulink, lo cual resalta la simplicidad y conveniencia de usar el programa desarrollado en este trabajo de titulación.
- En base al software de simulación Simulink, de Matlab, se pudo validar la herramienta computacional implementada, como se observa en el capítulo 4. Se realizaron varios ejercicios comparando los dos simuladores; contrastando las gráficas obtenidas entre estos se pudo concluir que los dos simuladores entregan los mismos resultados.
- El lenguaje de programación Python permite obtener los resultados de las variables de la máquina de inducción en forma directa. Éstas pueden ser fácilmente acopladas a otro programa o importadas; otra ventaja que presenta Python sobre Simulink es la obtención de gráficas fácilmente editables.
- La herramienta computacional implantada está desarrollada bajo una filosofía modular, la cual desarrolla una etapa específica del programa; esta filosofía permite realizar cambios a una etapa del programa sin influir de forma significativa, en las otras etapas del mismo.
- La versatilidad del lenguaje de programación Python se evidencia en la posibilidad de ejecutar bucles durante la simulación con el objetivo de variar parámetros de la máquina, como el voltaje en el ejercicio 5.2.3, con los cuales se pueden realizar un sin número de aplicaciones o investigaciones

6.2 Recomendaciones

- Debido a que el lenguaje de programación Python sobre escribe y guarda el archivo .py cada vez que éste se corre, se recomienda tener un archivo base y con este generar un nuevo archivo cada vez que se realiza un ejercicio.
- Se invita a desarrollar el control de la máquina de inducción, mediante la implementación de ecuaciones diferenciales en el código fuente; estas ecuaciones se las debe incorporar en el modelo de la máquina dentro del programa.
- Considerando un modelo jerárquico de programación, se recomienda desarrollar un modelo matemático en el cual sea posible incorporar la resistencia a tierra de la máquina de inducción.
- Se recomienda extender el método implementado en este trabajo de titulación a otros elementos de la red eléctrica como: transformadores, líneas de transmisión, maquinas sincrónicas, entre otras.
- Se recomienda analizar la posible utilización del programa para el análisis del comportamiento de la máquina de inducción como parte de un sistema de potencia, con su correspondiente conexión a líneas de transmisión y transformadores.
- Se recomienda la posible utilización del programa en el análisis de la máquina cuando se la usa en sistemas de generación eólica, motores de automóviles eléctricos, dentro de procesos industriales, entre otros.
- El programa podría ser modificado y ampliado para realizar la estimación de parámetros de una máquina de inducción, mediante la aplicación de técnicas de inteligencia artificial.

Referencias Bibliográficas

- [1]. O. Ruiz, y J Manuel. (2017, agosto). *Modelado Y Simulación De Máquinas De Inducción Simétricas*. [En línea]. Disponible en: <http://bibing.us.es/trabajodetitulacions/abreproy/4943>
- [2]. J. Vidal, *Estudio del Modelo Matemático del Motor de Inducción Trifásico*, España, 2009, pp. 8-48.
- [3]. R. Haltraman. (2011, noviembre 13). *Learning to Program with Python*. [En línea]. Disponible en: <https://www.cs.uky.edu/~keen/115/Haltermanpythonbook.pdf>
- [4]. P. Krause, *Analysis of Electric Machinery*. New Jork, 1995, pp.109-190.
- [5]. C. Ong, *Dynamic Simulation of Electric Machinery*, New Jersey, USA: Prentice Hall PTR, 1998, pp. 167-258.
- [6]. L. Simón y J. M., Monzón, *Modelado del Régimen Dinámico y Análisis de Sensibilidad de una Máquina Trifásica de Inducción Magnética*, España: Canarias, Información Tecnológica, Vol. 21, 2010.
- [7]. G. Van Rossa. (octubre, 2009). *Python*. [En línea]. Disponible en: <http://Python.org.ar/pyar/tutorial>

ANEXO I

Código y Manual de usuario del programa en Script. El código de este manual se adjunta en el ANEXO V (digital).

```
"""
Created on Thu Jan 25 09:33:26 2018

@author: Juan José
"""

import numpy as np
import scipy as sp
import math as mth
import matplotlib.pyplot as plt
from scipy.integrate import odeint

#####
#####Funciones#####
### Definición del Modelo de la Máquina de Inducción #####

def MaqInd (I,t):
# fqs es el flujo del eje q del estator
# fds es el flujo del eje d del estator
# i0s es la corriente cero del estator
# fqr es el flujo del eje q del rotor
# fdr es el flujo del eje d del rotor
# i0r es la corriente cero del rotor
# wx es la velocidad del rotor
# teta es el ángulo teta
# Vector de variables
    fqs,fds,i0s,fqr,fdr,i0r,wx,teta = I

# Transformación de voltajes ABC a qdo del estator
    Va= V*np.cos(t*we)          # Voltaje de la fase a de la red
    Vb= V*np.cos(t*we - 2*mth.pi/3 ) # Voltaje de la fase b de la red
    Vc= V*np.cos(t*we - 4*mth.pi/3 ) # Voltaje de la fase c de la red
    Vqs=(2/3)*(Va-(Vb+Vc)/2)     # Voltaje del eje q del estator
    Vds=(Vc - Vb)/np.sqrt(3)    # Voltaje del eje q del estator
    V0s=(Va + Vc + Vb)*(1/3)    # Voltaje cero del estator

# Cálculo de Ecuaciones de flujo mutuo y corrientes dq del estator
    fmq=XM*(fqs/Xls + fqr/Xlr)   # Flujo mutuo en el eje q
    fmd=XM*(fds/Xls + fdr/Xlr)   # Flujo mutuo en el eje q
    iqs = (fqs - fmq)/Xls        # Corriente en el eje q del estator
    ids = (fds - fmd)/Xls        # Corriente en el eje d del estator

# Ecuaciones de flujo del Estator
    dfqs = wb*(Vqs + (Rs/Xls)*(fmq - fqs)) # Derivada de fqs
    dfds = wb*(Vds + (Rs/Xls)*(fmd - fds)) # Derivada de fds
    di0s = (wb/Xls)*(V0s - i0s*Rs)        # Derivada de i0s

# Ecuaciones de flujo del Rotor
    dfqr = wb*(Vqr + wx*fdr + (Rr/Xlr)*(fmq - fqr)) # Derivada de fqr
    dfdr = wb*(Vdr - wx*fqr + (Rr/Xlr)*(fmd - fdr)) # Derivada de fdr
    di0r = (wb/Xlr)*(V0r - i0r*Rr)         # Derivada de i0r

# Ecuación del troque eléctrico Tem
    Tem = ((3*P)/(4*wb))*(fds*iqs - fqs*ids) # Torque eléctrico
```

```

# Ecuación de la velocidad wx
dwx = ((P)/(2*J*wb))*(Tem + Tmec - Tdamp) # Derivada de wx
# Ecuación de ángulo Teta
dteta = wx*2*np.pi*frated # Derivada de teta

return dfqs,dfds,di0s,dfqr,dfdr,di0r,dwx,dteta

### Función de integración del Modelo de la Máquina de Inducción
#####

def Integrar(MaqInd,I0,tiempoi,tiempof,paso):
##### Definición de Vector del Tiempo
### Número de iteraciones
itr= (tiempof - tiempoi)/paso

### Vector del tiempo
t1 = np.linspace(tiempoi,tiempof,itr)

# Llamado de la función odeint
solu=odeint(MaqInd, I0,t1,full_output = 0)

# Asignación de variables
[fqs1,fdsl,i0s1,fqr1,fdr1,i0r1,wx1,tetal] =
[solu[:,0],solu[:,1],solu[:,2],solu[:,3],solu[:,4],solu[:,5],solu[:,6],so
lu[:,7]]

# Concatenación de los resultados del estado anterior con los actuales
fqs2 = Concatenar(fqs,fqs1)
fdsl = Concatenar(fds,fdsl)
i0s2 = Concatenar(i0s,i0s1)
fqr2 = Concatenar(fqr,fqr1)
fdr2 = Concatenar(fdr,fdr1)
i0r2 = Concatenar(i0r,i0r1)
wx2 = Concatenar(wx,wx1)
teta2= Concatenar(teta,tetal)
t2=Concatenar(t,t1)

# Obtención del vector las variables de estado final
i = len(t1)-1
I0 = [fqs1[i],fdsl[i],i0s1[i],fqr1[i],fdr1[i],i0r1[i],wx1[i]]

# Vector de resultados de la función
res = [fqs2,fdsl,i0s2,fqr2,fdr2,i0r2,wx2,t2,I0,teta2]

return res

### Calculo de las corrientes abc-qd-Tem_wr #####

def Calcular(Soluciones):
# Asignación de Variables

[fqs,fdsl,i0s,fqr,fdr,i0r,wx,t,I1]=[Soluciones[0][:],Soluciones[1][:],Solu
ciones[2][:],Soluciones[3][:],Soluciones[4][:],Soluciones[5][:],Solucione
s[6][:],Soluciones[7][:],Soluciones[8][:]]

# Cálculo de las corrientes qd mediante la relación de flujos
fmq = XM*(fqs/Xls + fqr/Xlr)

```

```

fmd = XM*(fds/Xls + fdr/Xlr)
iqs = (fqs - fmq)/Xls
ids = (fds - fmd)/Xls
iqr = (fqr - fmq)/Xlr # Corriente del eje q del rotor
idr = (fdr - fmd)/Xlr # Corriente del eje d del rotor

# Cálculo del Tem y wr
Tem = ((3*P)/(4*wb))*(fds*iqs - fqs*ids)
wr = wx*30*frated # velocidad wx en rpm

# Transformación de las corrientes qd0 a abc del estator
# Corriente de la fase a del estator
ias = iqs + i0s
# Corriente de la fase b del estator
ibs = -0.5*iqs - ((np.sqrt(3))/2)*ids + i0s
# Corriente de la fase c del estator
ics = -0.5*iqs + ((np.sqrt(3))/2)*ids + i0s

# Transformación de las corrientes qd0 a abc del rotor
theta=0 # Ángulo entre el eje de referencia abc y el eje qd0
# Corriente de la fase a del rotor
iar=(iqr*np.cos(theta-teta)+idr*np.sin(theta-teta))
# Corriente de la fase b del rotor
ibr=(iqr*np.cos((theta-teta) - 2*np.pi/3)+idr*np.sin((theta-teta) -
2*np.pi/3))
# Corriente de la fase c del rotor
icr=(iqr*np.cos((theta-teta) +2*np.pi/3)+idr*np.sin((theta-teta)
+2*np.pi/3))
Respuesta=[ias,ibs,ics,iqs,ids,iqr,idr,Tem,wr,iar,ibr,icr]

return Respuesta

### Función de concatenación de variables ###

def Concatenar(X1,X2):
    X1= list(X1)
    X2= list(X2)
    X1.extend(X2)
    Y=np.array(X1)
    return Y

#####
##### Inicio del Programa#####
#####

##### Datos del sistema #####
Vrated = 460 # Voltaje del sistema
P = 4 # Número de polos de la maquina
frated = 60 # Frecuencia del sistema
wb = 2*np.pi*frated # Omega base
we = wb # Omega eléctrico
wbm = 2*wb/P # Omega mecánico base
V = Vrated*np.sqrt(2/3) # Voltaje de la máquina

##### 50HP #####
Xm = 13.08 # Flujo mutuo entre los devanados del rotor y estator
Rs = 0.087 # Resistencia de un devanado del estator
Rr = 0.228 # Resistencia de un devanado del rotor
Xls = 0.302 # Inductancia propia de un devanado del estator
Xlr = 0.302 # Inductancia propia de un devanado del rotor

```

```

J = 1.667 # Constante de inercia de la máquina
XM = (1)/(1/Xm + 1/Xls + 1/Xlr) # Constante de auxiliar
Tmec = 0 # Torque mecánico
Tdamp= 0 # Perdidas por fricción

# Voltajes abc y qd0 del Rotor
Vqr=0 # Voltaje del rotor de la fase q
Vdr=0 # Voltaje del rotor de la fase d
V0r=0 # Voltaje del rotor de la fase 0

# Inicialización de Variables
t = [] # Tiempo de simulación
Tem = [] # Torque eléctrico
fqs = [] # Flujo q del estator
fds = [] # Flujo d del estator
i0s = [] # Flujo 0 del estator
fqr = [] # Flujo q del rotor
fdr = [] # Flujo d del rotor
i0r = [] # Flujo 0 del rotor
wx = [] # Velocidad del rotor
teta= [] # Ángulo teta
A = 0 # Variable auxiliar
# Definición del vector de condiciones Iniciales
I0 = [0,0,0,0,0,0,0,0]

#####Inicio Del Programa#####

"""
Se debe seleccionar como variables esenciales el tiempo de inicio y el
tiempo final del periodo
Ejemplo de código
tiempoi=0
tiempof=0.8
A=Integrar(MaqInd,I0,tiempoi,tiempof,paso)
[fqs,fds,i0s,fqr,fdr,i0r,wx,t,I1]=[A[0][:],A[1][:],A[2][:],A[3][:],A[4][:],
,A[5][:],A[6][:],A[7][:],A[8][:]]

Para calcular los valores de corriente abc, qd0, Tem, wr se llama a la
función Calcular()
Ejemplo de Código
A=Calcular(A)
[ias,ibs,ics,iqs,ids,iqr,idr,Tem,wr]=[A[0][:],A[1][:],A[2][:],A[3][:],A[4]
[:],A[5][:],A[6][:],A[7][:],A[8][:]]

"""

tiempoi=0
tiempof=1
paso =0.0002
Tmec=0

A=Integrar(MaqInd,I0,tiempoi,tiempof,paso)
[fqs,fds,i0s,fqr,fdr,i0r,wx,t,I1,teta]=[A[0][:],A[1][:],A[2][:],A[3][:],A
[4][:],A[5][:],A[6][:],A[7][:],A[8][:],A[9][:]]
A=Calcular(A)
[ias,ibs,ics,iqs,ids,iqr,idr,Tem,wr,iar,ibr,icr]=[A[0][:],A[1][:],A[2][:],
,A[3][:],A[4][:],A[5][:],A[6][:],A[7][:],A[8][:],A[9][:],A[10][:],A[11][:]
]]

```

```

##### Graficas #####
plt.figure(figsize = (25,8))
plt.plot(wr,Tem)
plt.xlabel('Velocidad del Rotor [rpm]',fontsize = 23)
plt.ylabel('Torque Eléctrico [Nm]',fontsize = 23)
plt.title('Torque Eléctrico vs. Velocidad del Rotor',fontsize = 28)
plt.grid(True)
plt.savefig('Graficas Maq. Wr Vs Tem.jpg')

plt.figure(figsize = (25,8))
plt.plot(t, ias )
plt.xlabel('tiempo [s]',fontsize = 23)
plt.ylabel('Ias [A] ',fontsize = 23)
plt.title('Corriente Ias del Estator',fontsize = 28)
plt.grid(True)
plt.savefig('Graficas Ias .jpg')

plt.figure(figsize = (25,8))
plt.plot(t, ibs )
plt.xlabel('tiempo [s]',fontsize = 23)
plt.ylabel('Ibs [A] ',fontsize = 23)
plt.title('Corriente Ibs del Estator',fontsize = 28)
plt.grid(True)
plt.savefig('Graficas Ibs.jpg')

plt.figure(figsize = (25,8))
plt.plot(t, ics )
plt.xlabel('tiempo [s]',fontsize = 23)
plt.ylabel('Ics [A] ',fontsize = 23)
plt.title('Corriente Ics del Estator',fontsize = 28)
plt.grid(True)
plt.savefig('Graficas Ics.jpg')

plt.figure(figsize = (25,8))
plt.plot(t, iar )
plt.xlabel('tiempo [s]',fontsize = 23)
plt.ylabel('Ia [A] ',fontsize = 23)
plt.title('Corriente Iar del Rotor',fontsize = 28)
plt.grid(True)
plt.savefig('Graficas Iar .jpg')

plt.figure(figsize = (25,8))
plt.plot(t, ibr )
plt.xlabel('tiempo [s]',fontsize = 23)
plt.ylabel('Ibr [A] ',fontsize = 23)
plt.title('Corriente Ibr del Rotor',fontsize = 28)
plt.grid(True)
plt.savefig('Graficas Ibr.jpg')

plt.figure(figsize = (25,8))
plt.plot(t, icr )
plt.xlabel('tiempo [s]',fontsize = 23)
plt.ylabel('Icr [A] ',fontsize = 23)
plt.title('Corriente Icr del Rotor',fontsize = 28)
plt.grid(True)
plt.savefig('Graficas Icr.jpg')

plt.figure(figsize = (25,8))
plt.plot(t, Tem )
plt.xlabel('tiempo [s]',fontsize = 23)

```

```

plt.ylabel('Tem [Nm] ',fontsize = 23)
plt.title('Torque Eléctrico Vs Tiempo',fontsize = 28)
plt.grid(True)
plt.savefig('Graficas Tem Vs Tiempo .jpg')

plt.figure(figsize = (25,8))
plt.plot(t, wr )
plt.xlabel('tiempo [s]',fontsize = 23)
plt.ylabel('Wr [rpm] ',fontsize = 23)
plt.title('Velocidad del rotor Vs Tiempo',fontsize = 28)
plt.grid(True)
plt.savefig('Graficas Wr Vs Tiempo.jpg')

```

Ejemplo del uso del código

A continuación, se realiza un ejemplo del uso del código, con la finalidad de establecer una guía para el usuario. Este ejemplo solo establece los pasos estándar para realizar ejercicios de arranques en una máquina, para realizar ejercicios más complejos se debe modificar el código. Para la modificación del código se recomienda revisar el capítulo III, especialmente la sección 3.

1. Se deben ingresar los valores característicos de la máquina de inducción y de la red.

```

##### Datos del sistema #####
Vrated = 460 # Voltaje del sistema
P = 4 # Número de polos de la máquina
frated = 60 # Frecuencia del sistema
wb = 2*np.pi*frated # Omega base
we = wb # Omega eléctrico
wbm = 2*wb/P # Omega mecánico base
V = Vrated*np.sqrt(2/3) # Voltaje de la máquina

##### 50HP #####
Xm = 13.08 # Flujo mutuo entre los devanados del rotor y estator
Rs = 0.087 # Resistencia de un devanado del estator
Rr = 0.228 # Resistencia de un devanado del rotor
Xls = 0.302 # Inductancia propia de un devanado del estator
Xlr = 0.302 # Inductancia propia de un devanado del rotor
J = 1.667 # Constante de inercia de la máquina

```

2. Se debe seleccionar como variables esenciales el tiempo de inicio y el tiempo final del período.

Ejemplo de código

```
tiempoi=0
```

```
tiempof=0.8
```

```
paso =0.0002
```

```
A=Integrar(MaqInd,I0,tiempoi,tiempof,paso)
```

```
[fqs,fds,i0s,fqr,fdr,i0r,wx,t,I1]=[A[0][:],A[1][:],A[2][:],A[3][:],A[4][:],A[5][:],A[6][:],A[7][:],  
A[8][:]]
```

3. Para calcular los valores de corriente abc, qd0, Tem, wr se llama a la función Calcular()

```
A=Calcular(A)
```

```
[ias,ibs,ics,iqs,ids,iqr,idr,Tem,wr]=[A[0][:],A[1][:],A[2][:],A[3][:],A[4][:],A[5][:],A[6][:],A[  
7][:],A[8][:]]
```

4. Para presentar los resultados se debe crear una gráfica.

```
##### Graficas #####
```

```
plt.figure(figsize = (25,8))
```

```
plt.plot(wr,Tem)
```

```
plt.xlabel('Velocidad del Rotor [rpm]',fontsize = 23)
```

```
plt.ylabel('Torque Eléctrico [Nm]',fontsize = 23)
```

```
plt.title('Torque Eléctrico vs. Velocidad del Rotor',fontsize = 28)
```

```
plt.grid(True)
```

```
plt.savefig('Graficas Maq. Wr Vs Tem.jpg')
```

ANEXO II

Manual de usuario del GUI. El código de este manual se adjunta en el ANEXO VI (digital).

The screenshot shows the user interface of the induction machine simulator. It features a header with the university logo and name, and a main section for data entry. The interface is divided into several sections:

- Machine Data Section:** Titled "Ingrese los datos de la Máquina de inducción". It contains input fields for "Voltaje" (Voltage), "Polos" (Poles), "Frecuencia" (Frequency), "Const. Inercia" (Inertia Constant), "Rotor" (Rotor), and "Estator" (Stator). Each parameter has a unit specified to its right.
- Simulation Parameters Section:** Titled "Ingrese los valores de cada variable para cada tramo de simulación". It includes fields for "Tiempo de duración de cada tramo" (Duration of each segment in seconds), "Voltaje para cada tramo" (Voltage for each segment), and "Tórque mecánico para cada tramo" (Mechanical torque for each segment). There are also radio buttons for "Simular Un solo tramo" (Simulate one segment), "Simular varios Tramos" (Simulate several segments), and "Omitir el arranque" (Omit start).
- Graphing Section:** Titled "Seleccione la variable que desea graficar para cada eje." (Select the variable you want to graph for each axis). It has two columns for "Eje y:" and "Eje x:" with radio button options for various variables like i_{as} , i_{bs} , i_{cs} , i_{ar} , i_{br} , i_{cr} , T_{em} , w_r , and t .
- Axis and Title Fields:** Fields for "Titulo del Eje y" (Y-axis title), "Titulo del Eje x" (X-axis title), and "Titulo de la Gráfica" (Graph title). A "Guardar Gráfica" (Save Graph) checkbox is also present.
- Buttons:** "Graficar" (Graph) and "Ingresar Datos" (Enter Data) buttons are located at the bottom right.

Numbered callouts (1-7) are placed over the interface to highlight specific elements: 1 points to the "Polos" field; 2 points to the "Simular varios Tramos" radio button; 3 points to the "Voltaje para cada tramo" field; 4 points to the "Graficar" button; 5 points to the i_{bs} radio button in the Y-axis selection; 6 points to the "Titulo del Eje y" field; and 7 points to the "Graficar" button.

Pasos previos

Ingresar el comando '%matplotlib auto' en el espacio de trabajo.

Correr el código.

Pasos a seguir para simular:

1. Ingresar los datos de la máquina de inducción, usar punto para decimales.

2. Seleccionar si la simulación es de varios tramos o de un solo tramo, también seleccionar si se desea omitir el arranque.
3. Ingresar los valores de voltaje, torque y la duración de cada tramo.
4. Dar "Clic" en el botón Ingresar Datos.
5. Seleccionar las variables que se desea graficar para cada eje.
6. Ingresar los títulos de los ejes y de la gráfica, marcar la casilla de guardar si desea hacerlo.
7. Dar "Click" en el botón Graficar.