

ESCUELA POLITÉCNICA NACIONAL

**FACULTAD DE INGENIERÍA ELÉCTRICA Y
ELECTRÓNICA**

**DISEÑO E IMPLEMENTACIÓN DE UN ENTORNO DE
CO-SIMULACIÓN ENTRE MATLAB-SIMULINK Y
POWERFACTORY PARA EL ANÁLISIS DE
FENÓMENOS EN ESTADO TRANSITORIO EN UN
SISTEMA FOTOVOLTAICO CONECTADO A LA RED**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL
TÍTULO DE INGENIERO EN ELECTRÓNICA Y CONTROL**


GEORGINIO ALBERTO ANANGANÓ ALVARADO

DIRECTOR: DR.- ING. MARCELO POZO P.

Quito, Noviembre 2022

AVAL

Certifico que el presente trabajo fue desarrollado por Georginio Alberto Ananganó Alvarado, bajo mi supervisión.



DR.-ING. MARCELO POZO P.
DIRECTOR DEL TRABAJO DE TITULACIÓN.

DECLARACIÓN DE AUTORÍA

YO, Georginio Alberto Ananganó Alvarado 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.

Sin perjuicio de los derechos reconocidos en el primer párrafo del artículo 114 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación -COESC- soy titular de la obra en mención y otorgo una licencia gratuita, intransferible y no exclusiva de uso con fines académicos a la Escuela Politécnica Nacional. Entregaré toda la información técnica pertinente. En el caso de que hubiese una explotación comercial de la obra por parte de la EPN, se negociarán los porcentajes de los beneficios conforme lo establece la normativa nacional vigente.



GEORGINIO ALBERTO ANANGANÓ ALVARADO

DEDICATORIA

Empezaré recordando a mi familia: entrañable y presente, especialmente a mi hermana: Melany Maribel, de quien espero tome nota de mis aciertos y de mis errores: para mejorar unos y evitar otros. Seguramente ella superará mis logros con el tiempo, lo que me da una gallinesca felicidad.

La gente con la que crecemos empieza a desaparecer de nuestras vidas al transcurrir los años de adultez. Poder incluir aún a mis amigos del colegio, del *Grupo Smash*, la *Poli*, la universidad de la vida, de Ecuador y fuera de él tras tantos años... tiene un valor especial. En vez de enlistarlos a todos, prefiero que quien se precie de ser mi amigx, sienta mi emoción de todo corazón.

En reciprocidad, quiero incluir alfabéticamente los nombres de Adrián Balcázar y Jeysson Tapia. Cada día intento ser un mejor ser humano, y su reconocimiento nominal es un orgullo para mí.

Durante un tiempo crucial, tuve el privilegio de conocer algunos pedacitos de Andalucía a la distancia. Esto no habría sido posible sin la contribución invaluable del Sr. Manuel J. Tallafé, a quien reconozco su espíritu de perseverancia y altruismo. Tras la dureza del trabajo infatigable de muchos años, seguramente tendrá su recompensa.

Llegando a este punto, seguramente pensarás que me he olvidado de tí. Eso no sucederá en esta oportunidad. Inspirándome en Gailman, te dedico este trabajo. Quizá nos hayamos visto o nos conozcamos, quizá no. Sin embargo, el hecho que te encuentres aquí por este momento, nos ha acercado de forma intangible por la eternidad.

Noviembre, 2022.

AGRADECIMIENTO

Durante estos años, la Escuela Politécnica Nacional se convirtió en mi segunda casa. Una universidad no es tal sin la dedicación de sus estudiantes, el apoyo de todo el personal administrativo, de servicios, seguridad y buenos profesores.

Existen personas que destacan por su calidad humana y dan todo de sí para sacar adelante la academia y la sociedad. Ha sido un privilegio poder contar con el apoyo del Dr. Marcelo Pozo, quien aparte de brindarme su confianza para desarrollar este trabajo, más allá de su invaluable conocimiento, me ha brindado su paciencia, y aún, sus consejos para la vida.

Sin el apoyo personal y técnico del Ing. Luis Chiza, este trabajo no se habría realizado. Su compañía desde la proposición de este trabajo hasta su finalización ha representado una muestra de su interés científico y su espíritu desprendido.

Quisiera mencionar también, al Dr. Óscar Camacho, por su invaluable contribución en las últimas etapas de esta carrera, la MBA. Ana Rodas por su voto de confianza en momentos críticos, la Dra. Silvana Gamboa por su invaluable conocimiento, al Dr. Álvaro Javier Prado con quien tuve el privilegio de colaborar a pesar de mis limitaciones, al Ing. Alex Guzmán, quien también me asesoró desinteresadamente al inicio de mi investigación.

Rindo homenaje a la Mgtr. Mónica Andrango, al Econ. Daniel Sánchez y su familia, y al Ing. Andrés Jara.

Agradezco el enorme apoyo profesional brindado por la MSc. Nathaly Guamán, psicóloga de la EPN, quien afrontó mi terapia durante la pandemia. La salud mental es importante. Saludo al Sr. Iván Jaramillo, así como a todas las personas orgullosas del valle de los nueve volcanes.

No puedo finalizar estas palabras sin reconocer efusivamente a la Ing. *Mariemilia* Salcedo¹ "*La más humilde*". La magnitud inmensa de mi gratitud hacia su persona quizá pueda reflejarse en parte, con esta pequeña pero sincera mención.

Noviembre, 2022.

¹Emilia J. Salcedo.

RESUMEN

En este trabajo se plantea el diseño e implementación de un entorno de co-simulación entre Matlab-Simulink y Powerfactory para el análisis de fenómenos en estado transitorio en un sistema fotovoltaico conectado a la red. Considerando las limitaciones existentes para simular fenómenos en estado transitorio en un sistema eléctrico de potencia por parte de los programas anteriormente mencionados en forma independiente, se propone a la co-simulación como una alternativa viable para lograr este objetivo, implementando un script en Python para la sincronización. Se realiza una revisión bibliográfica, planteando la metodología para lograr una simulación exitosa, presentando y discutiendo los resultados. Finalmente se exponen las conclusiones y recomendaciones surgidas de este proyecto.

Palabras clave: Cosimulación, generación fotovoltaica, Powerfactory, Matlab, Simulink, estado transitorio.

ABSTRACT

This work proposes the design and implementation of a co-simulation environment between Matlab-Simulink and Powerfactory for the analysis of phenomena in transitory state in a photovoltaic system connected to the grid. Due to the existing limitations to simulate transient state phenomena in an power system by the aforementioned software independently, cosimulation is proposed as a viable alternative to achieve this objective, implementing a Python script for synchronization. A bibliographic review is carried out, proposing the methodology to achieve a successful simulation, presenting and discussing the results. Finally, the conclusions and recommendations arising from this project are presented.

Keywords: Co-simulation, photovoltaic generation, Powerfactory, Matlab, Simulink, transient state.

ÍNDICE GENERAL

| | |
|--|------------|
| Aval | I |
| Declaración de autoría | II |
| Dedicatoria | III |
| Agradecimiento | IV |
| Resumen | V |
| Abstract | VI |
| 1. Introducción | 1 |
| 1.1. Objetivo general | 2 |
| 1.2. Objetivos específicos | 2 |
| 1.3. Alcance | 3 |
| 1.4. Marco teórico | 4 |
| 1.4.1. Sistemas fotovoltaicos | 4 |
| 1.4.1.1. Generación fotovoltaica de energía eléctrica | 4 |
| 1.4.1.2. Consecuencias de los efectos ambientales en la generación fotovoltaica | 5 |
| 1.4.1.3. Modelo matemático de un panel fotovoltaico | 5 |
| 1.4.1.4. Caracterización de potencia y corriente en paneles mediante curvas IV-PV | 7 |
| 1.4.1.5. Efecto de sombras parciales | 8 |
| 1.4.2. Convertidores electrónicos de potencia | 9 |
| 1.4.2.1. Convertidores DC/DC | 10 |
| 1.4.2.2. Convertidor DC/AC Trifásico | 13 |
| 1.4.2.3. Filtros pasivos para inversores trifásicos | 14 |
| 1.4.3. Control Electrónico de Potencia | 15 |
| 1.4.3.1. Seguimiento del punto de máxima potencia (MPPT) en pa- neles fotovoltaicos | 15 |

| | | |
|-----------|--|-----------|
| 1.4.3.2. | Modulación SPWM | 20 |
| 1.4.3.3. | Sistemas de referencia y transformaciones de coordenadas en sistemas trifásicos | 22 |
| 1.4.3.4. | Control de lazos de corriente | 27 |
| 1.4.3.5. | Filtros pasabajos | 31 |
| 1.4.3.6. | Sintonización de controladores PID | 32 |
| 1.4.4. | Análisis de sistemas eléctricos de potencia | 35 |
| 1.4.4.1. | Elementos de un SEP | 36 |
| 1.4.4.2. | Flujos de potencia | 37 |
| 1.4.4.3. | Análisis de SEP en estado transitorio | 40 |
| 1.4.4.4. | Procesos para simulación de SEP | 46 |
| 1.4.5. | Software de simulación | 48 |
| 1.4.5.1. | Simuladores de sistemas eléctricos de potencia | 48 |
| 1.4.5.2. | Simuladores de circuitos electrónicos de bajo voltaje | 49 |
| 1.4.5.3. | Simuladores híbridos | 51 |
| 1.4.5.4. | Matlab-Simulink | 52 |
| 1.4.5.5. | DigSILENT PowerFactory | 53 |
| 1.4.6. | Co-simulación | 55 |
| 1.4.6.1. | Conceptos fundamentales | 55 |
| 1.4.6.2. | Parámetros de co-simulación para sistemas de potencia | 56 |
| 1.4.6.3. | Avances de la co-simulación aplicados en SEP | 58 |
| 1.4.7. | OPC | 59 |
| 1.4.7.1. | Motivación para el desarrollo de OPC | 60 |
| 1.4.7.2. | Desarrollo de OPC | 61 |
| 1.4.7.3. | OPC en la simulación y co-simulación | 63 |
| 1.4.8. | Python | 64 |
| 1.4.8.1. | Situación actual | 64 |
| 1.4.8.2. | API de Python | 65 |
| 1.4.8.3. | Python y PowerFactory en modo núcleo para la co-simulación con Simulink | 65 |
| 2. | Metodología | 67 |
| 2.1. | Arquitectura de la co-simulación | 67 |
| 2.1.1. | Hardware utilizado | 68 |

| | | |
|----------|---|-----|
| 2.2. | Generación fotovoltaica en Simulink | 69 |
| 2.2.1. | Generalidades | 69 |
| 2.2.2. | Modelo del panel fotovoltaico | 70 |
| 2.2.3. | Implementación del modelo en <i>Simulink</i> | 71 |
| 2.2.4. | Parametrización del modelo fotovoltaico | 74 |
| 2.2.5. | Perfiles de Irradiancia | 78 |
| 2.3. | Convertidores electrónicos de potencia en Simulink | 80 |
| 2.3.1. | Modelo del convertidor DC-DC | 80 |
| 2.3.1.1. | Trabajos previos | 80 |
| 2.3.1.2. | Elección y diseño del convertidor DC/DC | 80 |
| 2.3.2. | Modelo del convertidor DC-AC | 82 |
| 2.3.3. | Filtro pasivo | 83 |
| 2.4. | Control de convertidores electrónicos de potencia en Simulink | 84 |
| 2.4.1. | Control de voltaje del convertidor elevador | 84 |
| 2.4.2. | Seguimiento del punto de máxima potencia | 87 |
| 2.4.3. | Control del Inversor Trifásico | 88 |
| 2.4.3.1. | Lazo de seguimiento de fase | 89 |
| 2.4.3.2. | Sintonización de controladores de Eje Directo, Eje de Cuadratura y Voltaje de Bus | 90 |
| 2.4.3.3. | Modulación SPWM | 91 |
| 2.5. | Modelo del SEP en PowerFactory | 91 |
| 2.6. | Servidor y Clientes OPC | 95 |
| 2.6.1. | Matrikon OPC Server for Simulation | 95 |
| 2.6.2. | Monitoreo en <i>Matrikon OPC Explorer</i> | 99 |
| 2.6.3. | Configuración del cliente OPC en Simulink | 101 |
| 2.6.3.1. | Bloque OPC Config | 102 |
| 2.6.3.2. | Bloque OPC Read | 103 |
| 2.6.3.3. | Bloque OPC Write | 104 |
| 2.6.4. | Configuración del cliente OPC en PowerFactory | 105 |
| 2.6.4.1. | Configuraciones Iniciales | 105 |
| 2.6.4.2. | Bloques DSL | 106 |
| 2.6.4.3. | Configuración de Tags en PowerFactory | 108 |
| 2.6.4.4. | Configuración de parámetros de simulación en tiempo real | 110 |
| 2.6.5. | Mejora de la sincronización entre programas | 112 |

| | |
|--|------------|
| 2.7. Sincronización mediante Python | 112 |
| 2.7.1. Señales de disparo en Simulink | 113 |
| 2.7.2. Flujo de programa en Python | 113 |
| 2.7.3. Conexión del Script en PowerFactory | 114 |
| 3. Resultados | 116 |
| 3.1. Simulación sin conexión con PowerFactory | 117 |
| 3.1.1. Análisis de resultados | 120 |
| 3.2. Comunicación OPC unidireccional | 124 |
| 3.2.1. Comunicación Servidor-Simulink | 124 |
| 3.2.2. Comunicación Servidor-PowerFactory | 126 |
| 3.3. Comunicación OPC bidireccional | 127 |
| 3.4. Co-simulación Simulink - PowerFactory | 129 |
| 3.4.1. Planteamiento de los casos de estudio | 129 |
| 3.4.2. Caso 1: P & O sin sombras parciales | 129 |
| 3.4.3. Caso 2: P & O con sombras parciales | 132 |
| 3.4.4. Índices de Rendimiento de Control | 135 |
| 3.4.5. Análisis de resultados | 136 |
| 4. Conclusiones y Recomendaciones | 139 |
| 4.1. Conclusiones | 139 |
| 4.2. Recomendaciones | 140 |
| A. Anexos | 152 |
| A.1. Manual de Usuario | 152 |
| A.1.1. Objetivo del Manual | 152 |
| A.1.2. Prerrequisitos | 152 |
| A.1.3. Proceso de replicación | 152 |
| A.1.3.1. Verificación de archivos | 153 |
| A.1.3.2. Carga de Matrikon OPC Simulation Server | 154 |
| A.1.3.3. Carga de PowerFactory | 155 |
| A.1.3.4. Carga de Simulink | 158 |
| A.1.4. Resultados | 159 |
| A.2. Códigos y diagramas | 160 |
| A.2.1. Parametrización del modelo de Simulink | 160 |

| | |
|--|-----|
| A.2.2. Seguimiento de Máxima Potencia | 164 |
| A.2.3. Control de convertidores eléctricos de potencia | 166 |
| A.2.4. Sincronización Simulink y PowerFactory | 167 |
| A.2.5. SEP en PowerFactory | 169 |
| A.2.6. Ejecución del sistema de prueba | 170 |

1. INTRODUCCIÓN

Se estima que la demanda de energía eléctrica en el 2030, supere a la demanda del 2020 en un 60 %, a causa del crecimiento demográfico y las necesidades mundiales de productividad [1]. Con la entrada masiva de un parque automotor basado en motores eléctricos y baterías, esta estimación se incrementará considerablemente [2]. Con estos desafíos, la generación de energía eléctrica utilizando recursos renovables se convierte en una excelente oportunidad para poder subsanar la demanda futura de energía, mejorando su acceso y calidad, a la vez que se evitan efectos medioambientales perniciosos, como el cambio climático [3]. La generación fotovoltaica es una alternativa emergente que puede ser crítica para conseguir esta finalidad. En la Fig. 1.1 se puede apreciar de forma gráfica, el crecimiento sostenido del consumo energético a lo largo del siglo. Es esperable que esta tendencia continúe, presionando aún más a los operadores de generación energética.

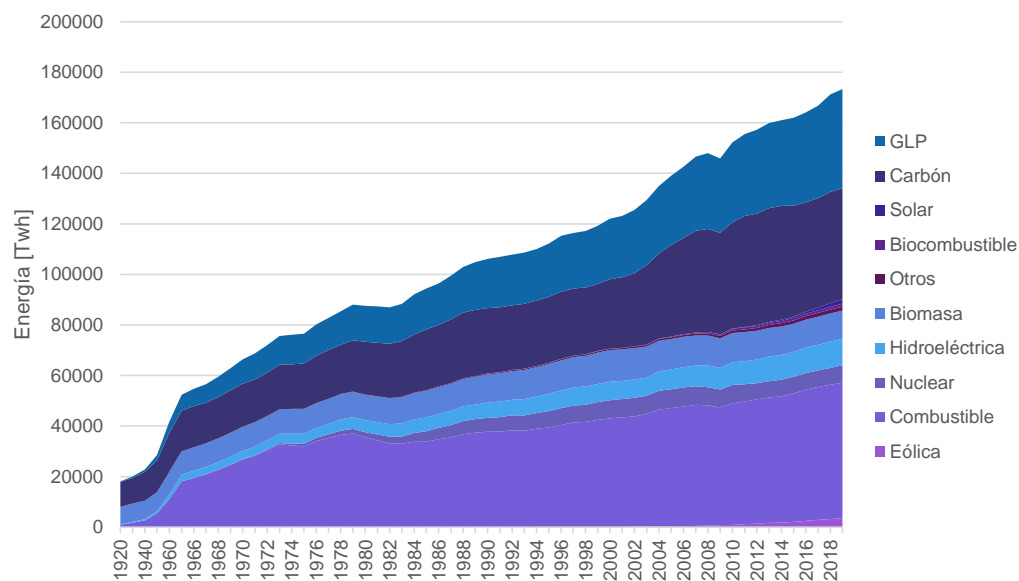


Figura 1.1: Consumo energético global durante el periodo 1920-2019 [4]

Sin embargo, la generación fotovoltaica presenta algunos inconvenientes: es compleja de integrar a la red de transmisión y distribución ya que depende demasiado del clima y la geografía causando que la cantidad de potencia disponible fluctúe en demasía, ocasionando efectos negativos como sobrevoltajes, armónicos, etc., propiciando así, que los sistemas eléctricos de potencia disminuyan su calidad de energía, pierdan estabilidad, con conse-

cuencias a nivel eléctrico, siendo evidentes a nivel de planificación y provocando inclusive, pérdidas financieras [5].

En un contexto marcado por estos desafíos, la investigación de fenómenos inherentes a la generación fotovoltaica y la propuesta de nuevas técnicas que permitan mejorar la eficiencia de los sistemas eléctricos de potencia, involucra simulaciones computacionales, las cuales permiten minimizar los tiempos y costos de prototipos [5]. Actualmente, uno de los mayores problemas es la inexistencia de un único software de simulación que permita modelar, reproducir y analizar todos los eventos transitorios de interés, para el control electrónico de potencia y para el análisis de flujos en sistemas eléctricos de potencia relacionados con la generación fotovoltaica y su integración con la red.

Este proyecto de titulación propone resolver este problema mediante un enfoque basado en la co-simulación. Esta es una técnica emergente en la que la simulación global de un sistema acoplado puede ser lograda mediante la simulación conjunta de cada una de sus partes . Se propondrá la integración de dos programas de referencia en el sector como lo son Matlab-Simulink y PowerFactory, intercomunicados mediante un servidor OPC y con una gestión de sincronización externa usando una API de Python. Los datos de irradiancia serán referenciados inicialmente desde el Atlas Solar del Ecuador, los parámetros de la central fotovoltaica se tomarán en base a la bibliografía existente, y la red eléctrica de prueba será la red de prueba de estabilidad IEEE de 9 barras.

1.1. OBJETIVO GENERAL

Diseñar e implementar un entorno de co-simulación entre Matlab-Simulink y Powerfactory para el análisis de fenómenos en estado transitorio de un sistema fotovoltaico conectado a la red.

1.2. OBJETIVOS ESPECÍFICOS

- Realizar la revisión bibliográfica referente a la generación fotovoltaica, control electrónico de potencia aplicado a generación fotovoltaica, co-simulación, protocolo OPC, análisis de sistemas eléctricos de potencia.
- Implementar un modelo del sistema fotovoltaico de prueba mediante Matlab-Simulink, incluyendo los convertidores electrónicos y algoritmos necesarios para una simulación aislada y conectada a la co-simulación, proponiendo condiciones de operación del

sistema de generación.

- Implementar el levantamiento del servidor OPC y los clientes OPC, realizando las configuraciones adecuadas en los softwares de simulación, gestionando los parámetros de sincronización con la API (Application Programming Interface) de Python.
- Simular y analizar los resultados de la interconexión del sistema de generación fotovoltaica con la red utilizando la metodología propuesta, de acuerdo a las diferentes condiciones de operación.

1.3. ALCANCE

- Se realizará una revisión bibliográfica referente a la generación de energía fotovoltaica, presentando conceptos teóricos, últimos avances y conceptos teóricos de modelamiento para la presente aplicación de co-simulación.
- Se realizará una revisión bibliográfica referente al análisis de sistemas eléctricos de potencia, introduciendo conceptos generales y presentando algunas técnicas de resolución para la presente co-simulación.
- Se revisarán los últimos avances respecto a los procesos de co-simulación en sistemas eléctricos de potencia.
- Se realizará una revisión teórica de las herramientas que proveen Matlab-Simulink y PowerFactory para la co-simulación.
- Se realizará una revisión teórica del protocolo de comunicación OPC y el API de Python para co-simulación.
- Se realizará una revisión teórica sobre la co-simulación específicamente entre PowerFactory y Matlab-Simulink.
- Se modelará el sistema fotovoltaico, sus componentes electrónicos y controladores electrónicos de potencia en Matlab-Simulink.
- Se modificará el sistema eléctrico de potencia de prueba IEEE 9 Barras para su utilización en la co-simulación del presente problema.
- Se implementará la comunicación biunívoca entre el servidor OPC y los clientes OPC por separado.

- Se implementará la comunicación biunívoca entre el servidor OPC y los clientes OPC en conjunto.
- Se programará en Python un gestor de sincronización para mejorar la respuesta dinámica y reducir las pérdidas por sincronismo.
- Se realizarán simulaciones de prueba entre el servidor OPC y Matlab-Simulink
- Se realizarán simulaciones de prueba entre el servidor OPC y PowerFactory.
- Se realizarán simulaciones dinámicas entre Matlab-Simulink, PowerFactory con un servidor OPC y utilizando el gestor de sincronización de Python, incluyendo los componentes electrónicos y controladores electrónicos de potencia en Matlab-Simulink para distintas condiciones de operación del sistema en conjunto.
- Se recopilarán los resultados producto de las simulaciones anteriormente realizadas.
- Se analizarán los resultados de las simulaciones anteriormente realizadas.
- Se realizarán la presentación y análisis de los resultados obtenidos de la simulación, comparando los resultados obtenidos en diferentes condiciones de operación.

1.4. MARCO TEÓRICO

1.4.1. SISTEMAS FOTOVOLTAICOS

1.4.1.1. Generación fotovoltaica de energía eléctrica

La generación fotovoltaica es un tipo de generación de energía eléctrica. Se transforma la energía radiante electromagnética del sol a energía eléctrica utilizable por el usuario o la aplicación final mediante el uso apropiado de celdas (llamadas también paneles) solares.

La transformación de la energía electromagnética radiada por el Sol en energía eléctrica debe estudiarse desde un punto de vista físico, a causa de la interacción de las ondas electromagnéticas con la materia [6]. El fenómeno básico que causa esta interacción se conoce como *efecto fotoeléctrico*, y fue estudiado intensivamente por la física de principios del siglo XX. No se pretende efectuar una revisión pormenorizada de los fenómenos físicos que involucran a la generación fotovoltaica, ya que está fuera del alcance de este trabajo.

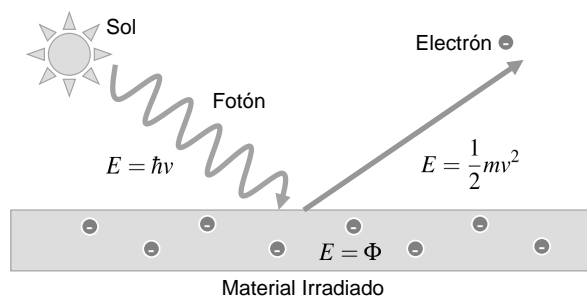


Figura 1.2: Esquema de funcionamiento del Efecto Fotoeléctrico

Ciertamente, como se aprecia en la Fig. 1.2, el sol emite energía, la cual es emitida en forma de fotones. Estos fotones pueden incidir sobre un material con una determinada banda de energía, para producir electrones con energía cinética propia. Los materiales pueden clasificarse en tres tipos: conductores, aislantes y semiconductores. De los tres tipos de materiales, los semiconductores han sido los que mejores resultados han producido en cuestión de producción energética. De hecho, los paneles fotovoltaicos son hechos casi en su totalidad de cristales semiconductores. La energía producida puede usarse para generar energía eléctrica.

1.4.1.2. Consecuencias de los efectos ambientales en la generación fotovoltaica

La cantidad de energía eléctrica que puede generarse a partir de una conversión fotovoltaica depende directamente de la cantidad de irradiación solar sobre los paneles solares, así como de la temperatura. La irradiación es una medida de densidad de potencia por área. A mayor irradiación solar, se espera mayor cantidad de producción energética, mientras que, a mayor temperatura, la capacidad de generación se degrada.

La generación fotovoltaica es compleja. Aunque los parámetros más importantes son los mencionados anteriormente, también existen otros como la posición geográfica de los paneles solares, la inclinación de los paneles, la orientación del sol [7]. También existen otros factores más delicados como la pureza de los semiconductores usados en los paneles solares, la composición química de los mismos, entre otros.

1.4.1.3. Modelo matemático de un panel fotovoltaico

Para estudiar los sistemas de generación fotovoltaica, es necesario disponer de un modelo eléctrico que represente esta fuente de energía, como por ejemplo, el caso de la generación hidráulica, en la que se disponen de diagramas de bloques que representan la

transformación electromecánica de la energía potencial del agua en energía eléctrica. El modelo más aceptado y utilizado es el provisto en [8] y que puede apreciarse en la Fig. 1.3, el cual considera que un panel fotovoltaico está compuesto por muchos paneles individuales llamados celdas.

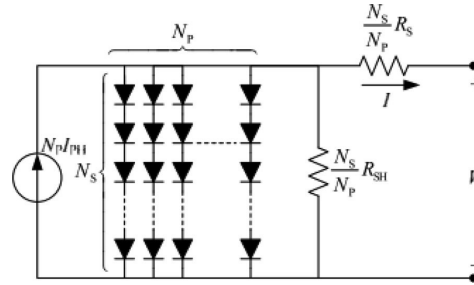


Figura 1.3: Circuito eléctrico propuesto para modelar un panel fotovoltaico [8].

Como se puede apreciar, el modelo de un panel fotovoltaico no es una fuente de voltaje pura. Es una fuente de corriente que produce un voltaje a la salida, el mismo que es dependiente de muchos parámetros. Así, la corriente producida por un panel fotovoltaico puede modelarse con la siguiente ecuación:

$$I = N_P \cdot I_{ph} - N_P \cdot I_0 \cdot \left[\exp \left(\frac{V}{N_S} + I \cdot \frac{R_S}{N_P} \right) - 1 \right] - I_{sh} \quad (1.1)$$

En la ecuación 1.1, N_P es la cantidad de celdas en paralelo, N_S es la cantidad de celdas en serie, $V_t = \frac{k \cdot T}{q}$ es el voltaje térmico donde k es la constante de Boltzmann, q es la carga del electrón, n es el factor de idealidad del diodo, T es la temperatura en grados Kelvin. $I_{sh} = \frac{V \cdot N_P / N_S + I \cdot R_S}{R_{sh}}$ es la corriente shunt del panel, que depende recursivamente del voltaje de salida del panel, la resistencia en serie R_S del panel, y la resistencia shunt del panel R_{sh} . La corriente de salida del panel también tiene una dependencia recursiva a su propio voltaje de salida y su propia corriente en ese instante. Esto implica un problema importante, ya que no puede despejarse la corriente I fácilmente. Queda definir cómo se modela la corriente fotovoltaica I_{ph} y la corriente de saturación I_0 , lo cual se hará a continuación:

$$I_0 = I_{rs} \cdot \left(\frac{T}{T_r} \right)^3 \cdot \exp \left(\frac{q \cdot E_{g0}}{n \cdot k} \cdot \left(\frac{1}{T} - \frac{1}{T_r} \right) \right) \quad (1.2)$$

La corriente de saturación presenta dependencia a la temperatura T , la temperatura de referencia T_r (generalmente definida a 20 o 25 grados centígrados), E_{g0} que es la banda de energía del semiconductor, además de I_{rs} , la corriente de saturación inversa, la cual se modela con la siguiente ecuación:

$$I_{rs} = \frac{I_{sc}}{\exp\left(\frac{q \cdot V_{OC}}{N_S \cdot k \cdot n \cdot T}\right) - 1} \quad (1.3)$$

La corriente de saturación inversa depende de la corriente de cortocircuito I_{sc} , el voltaje a circuito abierto V_{OC} . Queda solamente definir la corriente fotovoltaica como sigue:

$$I_{ph} = (I_{sc} + K_i \cdot (T - T_r)) \cdot \frac{I_r}{I_{rr}} \quad (1.4)$$

La corriente generada por el efecto fotovoltaico depende de la corriente de cortocircuito I_{sc} , K_i que es el coeficiente de cortocircuito de la celda, mientras que I_r es la irradiación solar. Se considera una irradiancia de 1000 vatios por metro cuadrado como irradiancia estándar. La temperatura considera un valor de referencia de $T_r = 298,303[\text{K}] = 20,25[\text{C}]$.

Como se puede apreciar, el comportamiento de una celda fotovoltaica es un fenómeno complejo dependiente de diversas variables. Sin embargo, las más importantes son las relacionadas con la temperatura y la irradiación solar, puesto que son las que más influyen en los resultados. Las recursividades son realmente un problema para la simulación puesto que no existen soluciones sencillas para despejar I exclusivamente. De hecho, los modelos fotovoltaicos integrados en software de simulación como el caso de Matlab-Simulink son de implementación relativamente reciente. Existen otros modelos que pueden usarse para simulaciones dinámicas que utilizan condensadores variables para modelar algunos fenómenos no lineales, sin embargo, el consenso general indica que el modelo estático presentado anteriormente es más que suficiente para la mayoría de estudios referentes al tema [6].

1.4.1.4. Caracterización de potencia y corriente en paneles mediante curvas IV-PV

Para modelar el comportamiento de potencia de un panel fotovoltaico, se usan extensivamente las curvas de corriente respecto a voltaje y potencia respecto a voltaje en las aplicaciones que requieren energía fotovoltaica. Considerando que la potencia es el producto entre estas magnitudes, estas curvas pueden resumir muchos aspectos técnicos del panel fotovoltaico.

La curva que se discutirá primero es la curva IV (corriente respecto al voltaje). De esta curva se pueden obtener algunos parámetros de fábrica, como el caso del voltaje de circuito abierto (obtenido cuando la corriente del panel es cero) así como la corriente de cortocircuito (obtenida cuando el voltaje del panel es cero). En la curva PV, se puede apreciar que,

idealmente se tiene solamente un punto máximo de potencia, el cual depende de la irradiancia y la temperatura. Este punto es muy importante puesto que es donde se obtiene el máximo desempeño del panel fotovoltaico [9]. De hecho, los controladores electrónicos de potencia que se utilizan en las etapas de elevación de voltaje en los paneles fotovoltaicos funcionan intentando que el convertidor se mantenga en el punto de máxima potencia mediante algoritmos de seguimiento, los cuales serán discutidos en páginas posteriores. Se pueden apreciar ejemplos de ambas curvas en la Fig. 1.4.

Al establecer comparaciones entre curvas PV o IV de diferentes fabricantes en condiciones similares, se pueden proveer mejores ideas a los diseñadores acerca del desempeño de los proyectos fotovoltaicos. Se busca la mayor cantidad de entrega de potencia posible.

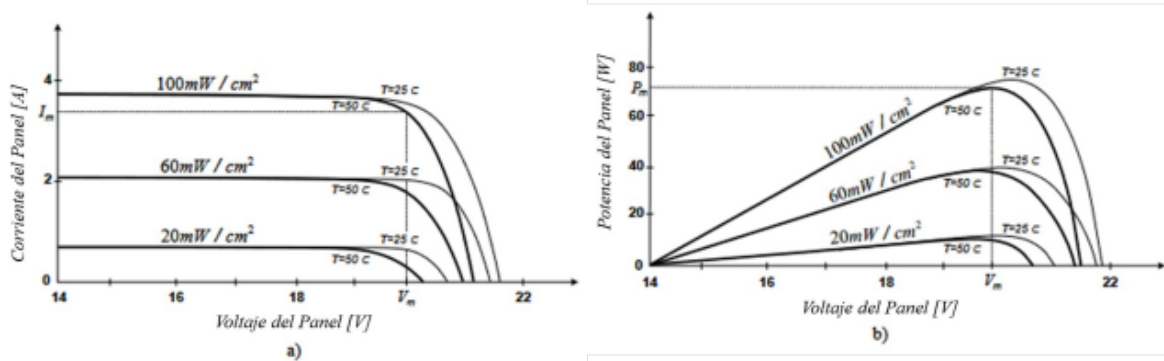


Figura 1.4: a) Curvas IV b) Curvas PV típicas de un panel fotovoltaico. Ambas gráficas consideran diferentes valores de irradiancia [9].

1.4.1.5. Efecto de sombras parciales

Los proyectos fotovoltaicos de gran envergadura suelen ser concebidos para abarcar grandes áreas. Sin embargo, las perturbaciones meteorológicas y condiciones ambientales tales como: nubosidades, suciedades, alineación deficiente de los paneles, degradaciones por envejecimiento, degradaciones por la modificación de la estructura cristalina de las celdas fotovoltaicas, etcétera, pueden causar efectos no deseados en la eficiencia de los sistemas de generación de este tipo.

En estos casos, las curvas PV e IV no son ideales y se producen serias modificaciones que interfieren con el despacho de potencia. El efecto de sombras parciales es importante ya que, su implementación en una simulación permite estudiar casos más realistas.

El efecto de las sombras parciales sobre los paneles fotovoltaicos provoca la aparición de características degradantes diferentes en los paneles solares [10]. La disminución de la potencia máxima de despacho, o el apareamiento de más de un punto de máxima potencia

son los efectos más importantes causados por este fenómeno, como se puede advertir en la Fig. 1.5.

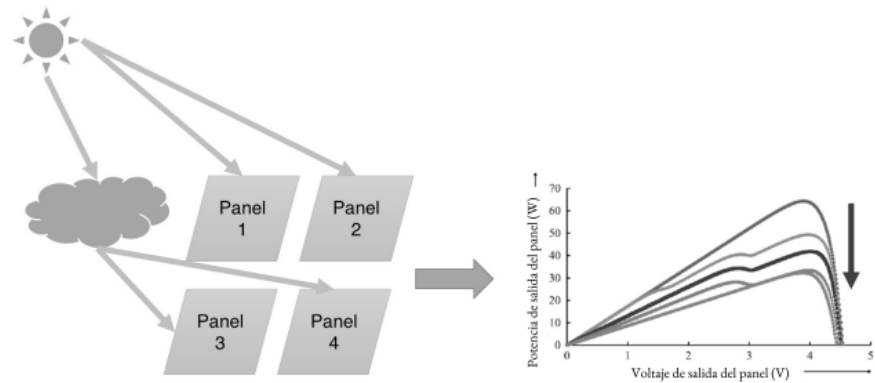


Figura 1.5: Efecto de las sombras parciales en un proyecto fotovoltaico [10].

La aparición de varios puntos de máxima potencia en las curvas PV constituye un verdadero problema para los algoritmos de seguimiento del punto de máxima potencia. Para aumentar el realismo de la simulación, en este proyecto de titulación se tomará en cuenta este fenómeno.

1.4.2. CONVERTIDORES ELECTRÓNICOS DE POTENCIA

Los sistemas basados en electrónica de potencia actualmente se usan de forma extensiva en el manejo de aplicaciones tan amplias como computación, sistemas celulares, comunicaciones, medicina, transporte, transmisión de energía eléctrica de alta potencia, entre otras [11]. El manejo de potencia de estos sistemas cubre un rango que va desde los microvatios hasta los megavatios. Los convertidores electrónicos de potencia están compuestos principalmente de interruptores semiconductores como diodos, transistores y elementos pasivos como inductores, condensadores, resistores, además de incluir circuitos integrados para el control de los interruptores. Los convertidores electrónicos de potencia pueden trabajar con corriente continua como con corriente alterna y pueden realizar conversiones entre estas formas de energía. En la Fig. 1.6 se puede apreciar un esquema que resume la clasificación general de los mismos, de acuerdo con la conversión de energía que realizan.

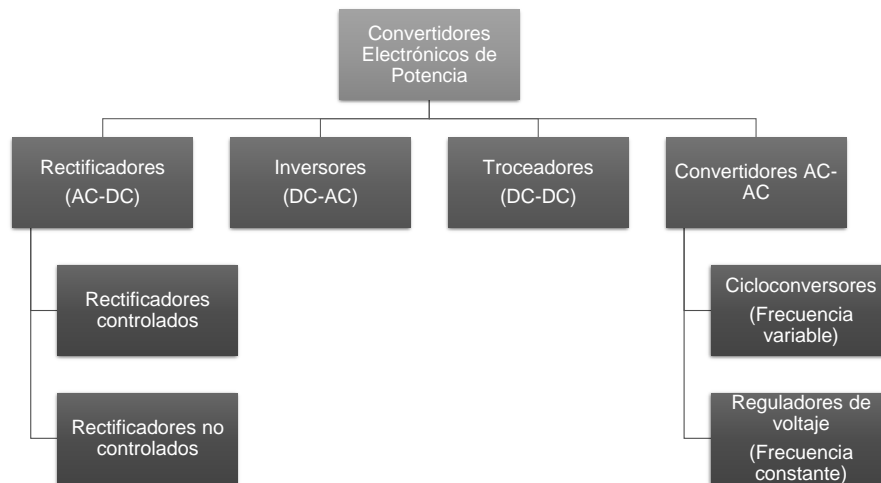


Figura 1.6: Clasificación de convertidores electrónicos de potencia.

En el desarrollo de este proyecto de titulación, se emplearán los convertidores DC/DC y DC/AC trifásicos (Inversor trifásico de puente completo). Se examinarán estos convertidores de interés, omitiendo los demás.

1.4.2.1. Convertidores DC/DC

Los convertidores DC/DC, son convertidores compuestos por circuitos electrónicos e interruptores de potencia que permiten obtener voltajes de salida en corriente continua ajustable, respecto a una fuente de voltaje de corriente continua fija en la entrada [11]. De acuerdo con la topología del circuito utilizado, la función que relaciona estos voltajes será una u otra. Sin embargo, en todas las funciones estará presente el ciclo de trabajo D . Existe una gran variedad de topologías disponibles [12].

El ciclo de trabajo no es más que la relación entre el tiempo de encendido t_{on} y el periodo de conmutación $T_{sw} = \frac{1}{f_{sw}}$ de los interruptores electrónicos de potencia. Como el tiempo de encendido debe ser menor o igual que el tiempo de conmutación, el rango del ciclo de trabajo está determinado por la relación $0 \leq D \leq 1$. La conmutación de estos interruptores se realiza a frecuencias fijas.

Los convertidores DC/DC podrían clasificarse por su aislamiento: Los convertidores no aislados tienen conexiones eléctricas directas entre la entrada y la salida del circuito. Estos convertidores son los más utilizados y estudiados por su simplicidad. Los convertidores aislados, por el contrario, tienen aislamiento galvánico entre la etapa de entrada y la etapa de

salida de potencia. El aislamiento galvánico se consigue mediante el uso de un transformador. Respecto a los transformadores conocidos de corriente alterna, los transformadores de corriente continua trabajan a frecuencias de conmutación muchísimo más altas, lo que permite una drástica disminución de su tamaño. Sin embargo, requieren de un diseño mucho más cuidado ya que, se requiere el manejo de variables magnéticas a causa del transformador de aislamiento utilizado [13]. Un esquema de la clasificación, se puede apreciar en la Fig. 1.7.

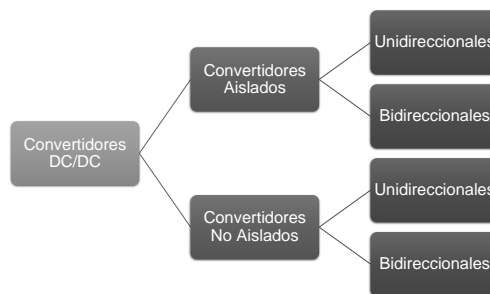
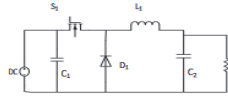
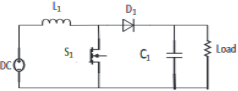
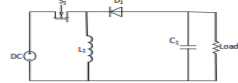
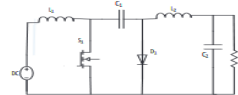
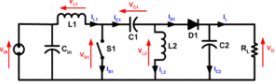
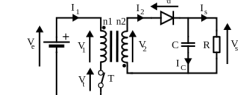
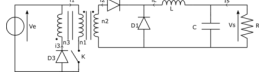


Figura 1.7: Esquema de la clasificación de los convertidores DC-DC.

Asimismo, los convertidores DC/DC también pueden clasificarse por su direccionalidad [13]: Los convertidores unidireccionales proveen potencia desde la entrada hacia la salida. Los convertidores bidireccionales pueden proveer un flujo de potencia desde la entrada hacia la salida o en sentido inverso. Los últimos son de vital importancia en los automóviles eléctricos, puesto que se puede usar esta característica para el freno regenerativo y recargar las baterías. En el caso fotovoltaico, se prefiere un convertidor unidireccional, puesto que las corrientes en reversa provocan que los paneles fotovoltaicos se comporten como cargas con el consecuente riesgo de la destrucción de los mismos. , y un resumen sobre los convertidores más comunes puede apreciarse en la Tabla 1.1.

Tabla 1.1: Resumen de los principales convertidores DC-DC usados en la conversión electrónica de potencia. $D \in [0, 1]$ es el ciclo de trabajo, N_p es el número de espiras en el primario del transformador, y N_s es el número de espiras en el secundario del transformador [13].

| Convertidor | Características | Función Vo/Vi | Topología |
|-------------|---|---------------------------------------|---|
| Buck | Simple, corriente de salida continua, con bajo rizado en la salida. Mejor eficiencia por costo. | D |  |
| Boost | Corriente de salida discontinua, arquitectura sencilla, bajo rizado en el lado de salida, componentes baratos para el filtrado. | $\frac{1}{1-D}$ |  |
| Buck-Boost | Tiene un voltaje de salida con polaridad inversa a la polaridad de entrada. Más rizado y menor eficiencia. | $-\frac{D}{1-D}$ |  |
| Cúk | Más componentes, polaridad de salida inversa a la polaridad de entrada, menor rizado que el convertidor Buck-Boost | $-\frac{D}{1-D}$ |  |
| Sepic | Menor rizado a la entrada, mayor a la salida. Control más complejo por la cantidad de elementos. | $\frac{D}{1-D}$ |  |
| Flyback | Es un convertidor aislado. Tiene un alto factor de potencia, baja eficiencia, bastantes pérdidas en el núcleo y un desfase de corriente alto a la salida. | $\frac{N_s}{N_p} \cdot \frac{D}{1-D}$ |  |
| Forward | Es un convertidor aislado. Tiene bajo factor de potencia, alta eficiencia, bajas pérdidas en el núcleo y un desfase de corriente bajo a la salida. | $\frac{N_s}{N_p} \cdot D$ |  |

1.4.2.2. Convertidor DC/AC Trifásico

Los inversores son convertidores electrónicos capaces de proporcionar energía eléctrica a partir de una fuente de corriente continua, en forma de corriente alterna. Esto se consigue mediante un adecuado control y topología de los flujos de corriente. Existen inversores monofásicos, bifásicos, trifásicos, polifásicos, multinivel, etcétera. Sin embargo, en esta sección la prioridad será hablar sobre los inversores trifásicos, puesto que, son los más utilizados en la generación de energía eléctrica, sin detrimento de las otras opciones.

Como se aprecia en la Fig. 1.8, un inversor trifásico funciona mediante la conexión de una fuente de corriente continua (una batería, un panel fotovoltaico) y puede alimentar una carga o proporcionar potencia a la red mediante la conexión o desconexión adecuada de un conjunto de interruptores. Estos interruptores, de acuerdo con parámetros variados como el voltaje, la corriente, la frecuencia de conmutación, pueden ser transistores bipolares (TBJ), transistores de efecto de campo (MOSFET), transistores de compuerta aislada (IGBT), entre otras opciones.

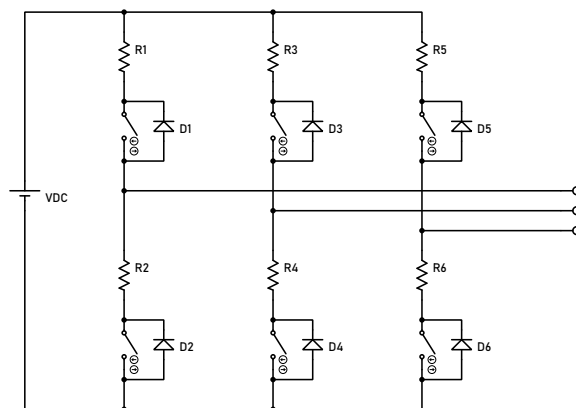


Figura 1.8: Esquema de un inversor trifásico de puente completo.

Un interruptor trifásico simple de puente completo está compuesto de tres ramales, correspondientes a las tres fases que podrá controlar. Es importante mencionar que los interruptores tienen idealmente una resistencia de encendido $R_{on} = 0$ y una resistencia de apagado $R_{off} \rightarrow \infty$. En la práctica, la resistencia de encendido es muy pequeña (en el orden de las unidades de ohmios) mientras que, la resistencia de apagado es muy alta (en el orden de los kilo o megaohmios). Por tal motivo, la condicionante más importante para el control de los interruptores es la imposibilidad de encender todos los interruptores de una rama al mismo tiempo. Esto, crearía un cortocircuito imposible de manejar para la fuente primaria de energía, con sus consecuencias destructivas.

1.4.2.3. Filtros pasivos para inversores trifásicos

Cuando se usa la modulación por ancho de pulso de alta frecuencia en los inversores trifásicos, las conmutaciones que buscan aproximar la forma de onda de corriente senoidal en la práctica se convierten en fuentes de armónicos, los cuales, dependiendo de su incidencia en el sistema, disminuirán la calidad de energía y ocasionarán daños a equipos sensibles cuando sobrepasen un umbral de distorsión armónica.

Generalmente, la distorsión armónica ocurre por la incidencia de máquinas eléctricas, fuentes conmutadas, transformadores, cargas no lineales. Una forma de atacar esta problemática es mediante el uso de filtros pasivos [14]. Los filtros pasivos son aquellos que consisten solamente de elementos como reactancias y capacitancias.

En la Fig. 1.9 se aprecia el efecto que tiene un filtro sobre la corriente. Aunque el experimento se realiza en un inversor monofásico, los resultados beneficiosos pueden extenderse al inversor trifásico. Disminuye la cantidad de distorsión armónica y puede mejorar la respuesta del sistema que se desea controlar mediante control electrónico de potencia. El diseño de estos filtros tiene una infinidad de metodologías. En el desarrollo metodológico de este trabajo, se detallará el diseño de un filtro LCL sencillo.

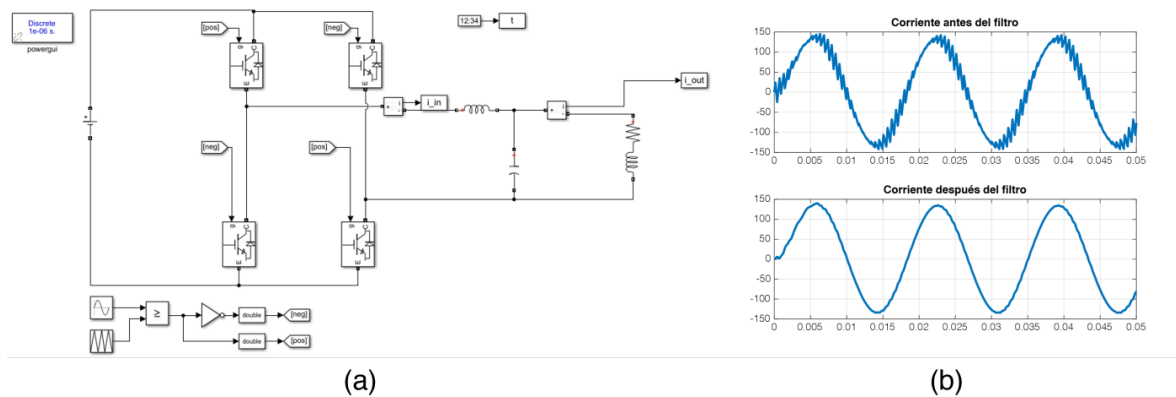


Figura 1.9: Simulación de un inversor monofásico de 120 VRMS, alimentando una carga RL (1 ohmio, 1 milihenrio), con un filtro LC (1 milihenrio, 100 microfaradios). (a) Circuito en Simulink. (b) Corrientes antes, y después del filtro.

1.4.3. CONTROL ELECTRÓNICO DE POTENCIA

1.4.3.1. Seguimiento del punto de máxima potencia (MPPT) en paneles fotovoltaicos

Es importante hablar de forma general sobre los algoritmos que se pueden utilizar para maximizar la eficiencia del despacho de energía en los proyectos que involucran energía fotovoltaica considerando que los algoritmos empleados involucran actuar directa o indirectamente sobre la variable más importante en el control electrónico de potencia: el ciclo de trabajo.

En general, existen dos grandes métodos para el seguimiento de máxima potencia: el método de comparación y el método de evaluación de la derivada. Los métodos de comparación se basan en la modificación sucesiva del ciclo de trabajo de forma que la potencia medida se vaya incrementando hasta llegar a un punto de convergencia en el que la potencia va disminuyendo, de forma que se vuelve oscilante alrededor de este punto de convergencia en el que la potencia aumenta y disminuye entre una banda de sensibilidad definida por el diseñador.

Los métodos de evaluación de la derivada son un poco más complejos ya que, consideran la definición matemática de un punto máximo como un punto en el que la derivada respecto a la variable independiente es cero. En este tipo de algoritmo no se mide la potencia, si no la derivada de la curva potencia-voltaje. Consideramos para la discusión teórica, una curva PV ideal, es decir: considerando una temperatura e irradiación constantes para todos los paneles que conforman el generador fotovoltaico, como se aprecia en la Fig. 1.10.

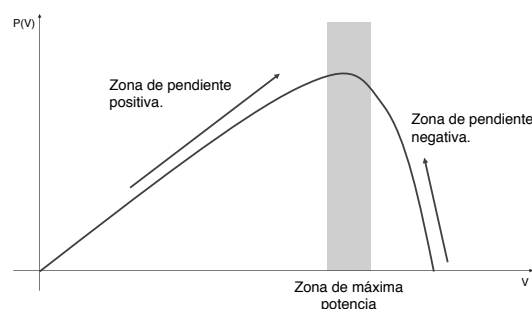


Figura 1.10: Curva PV de comportamiento ideal.

Método Perturbar y Observar Este método es el más sencillo de todos los algoritmos de seguimiento de máxima potencia. Su funcionamiento es el siguiente: Se inicializa un valor de potencia $P(k - 1)$. Después se pueden medir la corriente y el voltaje de salida del panel para calcular la potencia actualizada $P(k)$. Se procede a restar entre la potencia

actual y la potencia inicial. Si la diferencia entre estos valores es cero ($P(k) - P(k - 1) = 0$), teóricamente se está en el punto de máxima potencia. En caso contrario, se evalúa si la potencia está en la región de pendiente positiva ($P(k) - P(k - 1) > 0$) o en la región de pendiente negativa ($P(k) - P(k - 1) < 0$). Dependiendo del resultado, se evalúa si la variación entre los voltajes anteriores y actuales ($V(k) - V(k - 1)$) es positiva o negativa para generar una acción de control correspondiente [15].

En la práctica, casi nunca la variación de potencia es igual a cero. Por lo tanto, el algoritmo suele oscilar en el punto de máxima potencia. El rango de oscilación debe diseñarse en función de la curva PV para que no oscile demasiado, lo que puede causar estrés considerable en los componentes electrónicos, especialmente en los interruptores electrónicos de potencia. Este comportamiento se puede observar en la Fig. 1.11.

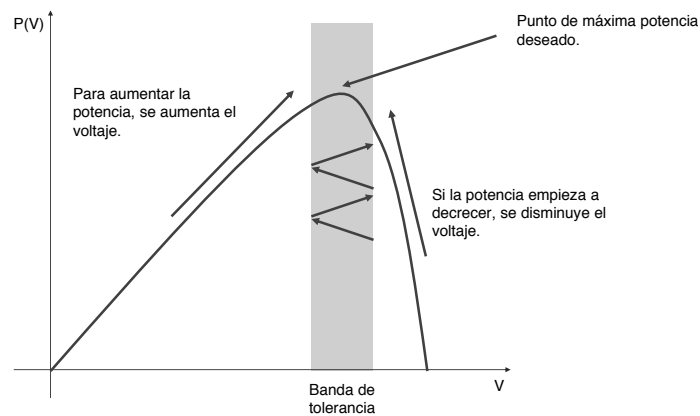


Figura 1.11: Esquema del funcionamiento del algoritmo de seguimiento de máxima potencia basado en perturbación y observación.

La ventaja de este método estriba en su sencillez, aunque su desventaja más evidente es la tendencia excesiva a la oscilación en el punto de máxima potencia. Para solventar este problema se han desarrollado otras técnicas de seguimiento que disminuyen la magnitud y frecuencia de estas oscilaciones [15]. Un diagrama de flujo para este método se puede observar en la Fig. 1.12.

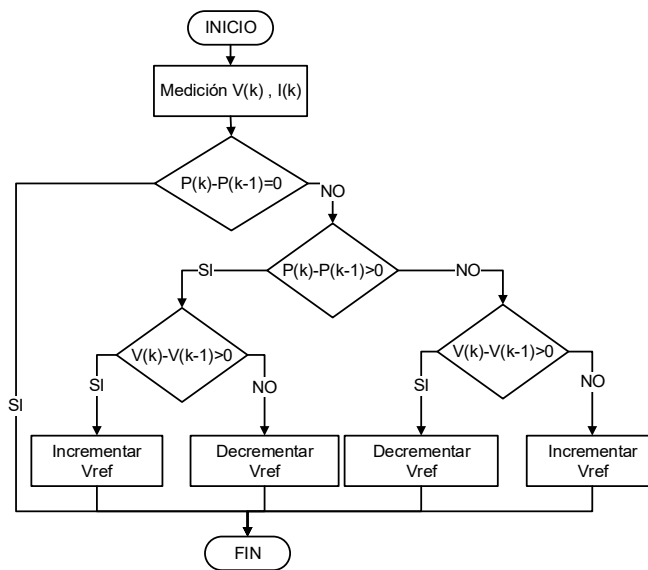


Figura 1.12: Diagrama de flujo para el algoritmo perturbar-observar usado para el seguimiento del punto de máxima potencia [16]

Método de Conductancia Incremental Este método tiene una base matemática basada en las propiedades de los puntos críticos de una función. Considerando la función de la potencia en función del voltaje $P = f(V)$, se puede afirmar que, en el punto de máxima potencia, el valor numérico de la derivada de la potencia dependiente respecto al voltaje es cero $\frac{dP}{dV} P_{\text{máx}} = 0$.

El nombre del método viene dado por el apareamiento de una conductancia en el desarrollo matemático de las fórmulas de cálculo, recordando que, la conductancia es el inverso de la resistencia. A continuación, se desarrollará la teoría matemática que corresponde al método de conductancia incremental.

$$\begin{aligned}
 P &= V \cdot I \\
 \frac{dP}{dV} &= \frac{d}{dV}(V \cdot I) \\
 \frac{dP}{dV} &= \frac{d}{dV}(V) \cdot I + \frac{d}{dV}I \cdot V \\
 \frac{dP}{dV} &= I + \frac{dI}{dV} \cdot V
 \end{aligned}$$

Como se aprecia en la ecuación, el método involucra una variación de corriente respecto a una variación de voltaje, es decir, una conductancia. Cuando se llega al punto de máxima potencia, se debe cumplir la condición:

$$\begin{aligned} \frac{dP}{dV} &= 0 \\ I + \frac{dI}{dV} \cdot V &= 0 \\ \frac{dI}{dV} &= -\frac{I}{V} \end{aligned}$$

En caso contrario, se evalúa la expresión $I + \frac{dI}{dV} \cdot V$. Si este valor es positivo, el controlador se encuentra en la parte ascendente de pendiente positiva, y si este valor es negativo, el controlador se encuentra en la parte descendente de pendiente negativa, por lo que se deberá incrementar o decrementar el voltaje [15]. Este esquema puede apreciarse en la Fig. 1.13. Un diagrama de flujo para este método se puede observar en la Fig. 1.14.

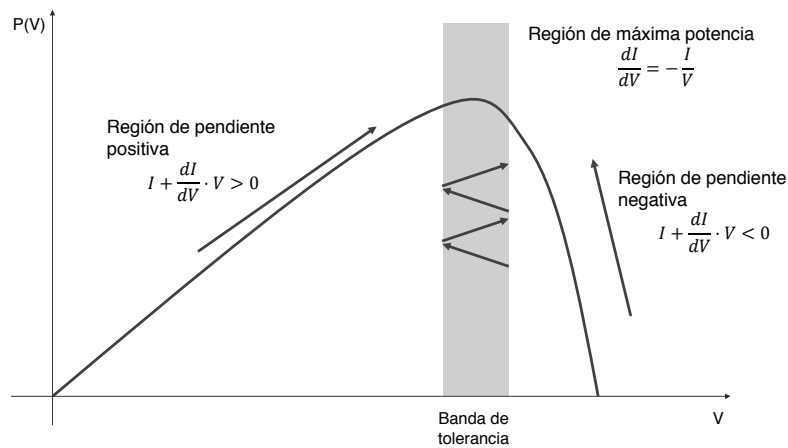


Figura 1.13: Esquema del funcionamiento del algoritmo de seguimiento de máxima potencia basado en conductancia incremental.

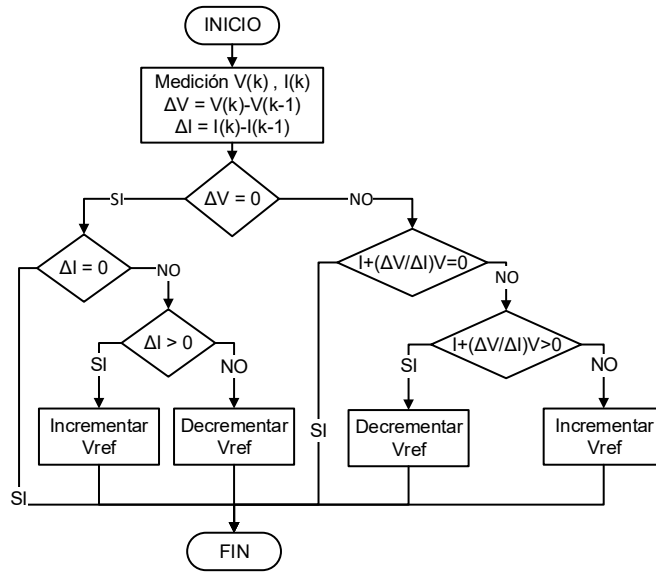


Figura 1.14: Diagrama de flujo para el algoritmo de conductancia incremental usado para el seguimiento del punto de máxima potencia [16]

Otros métodos Se han propuesto otros métodos para el seguimiento del punto de máxima potencia. Se han propuesto métodos para integrar controladores PID [17], lógica difusa [18], enjambre de partículas [19], redes neuronales [20]. Estos métodos buscan disminuir las oscilaciones en el punto de máxima potencia, así como el abordar el mayor problema ocasionado por las sombras parciales: el aparecimiento de puntos de máxima potencia que pueden entorpecer el funcionamiento de estos algoritmos, estabilizándose en puntos críticos que pueden ser no óptimos. Este problema puede apreciarse mejor en la Fig. 1.15.

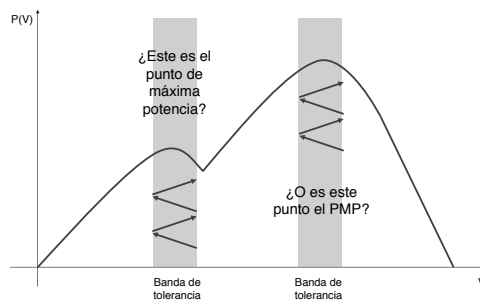


Figura 1.15: Esquema descriptivo del problema que causan las sombras parciales para los algoritmos de seguimiento del punto de máxima potencia.

1.4.3.2. Modulación SPWM

Para el control del disparo de los interruptores que conforman un convertidor electrónico de potencia, pueden utilizarse variedad de técnicas. Sin embargo, en este trabajo se usará la técnica de modulación SPWM (*Sinusoidal Pulse Width Modulation*, Modulación senoidal por ancho de pulso). Esta técnica permite aprovechar las propiedades de los elementos inductivos para obtener formas de corriente cercanas a la senoidal mediante el control de encendido o apagado de los interruptores controlados.

En la figura 1.16 se puede apreciar un esquema que intenta explicar el funcionamiento de la modulación SPWM. Se tienen dos señales: una señal modulante $m(t)$ y una señal portadora $p(t)$.

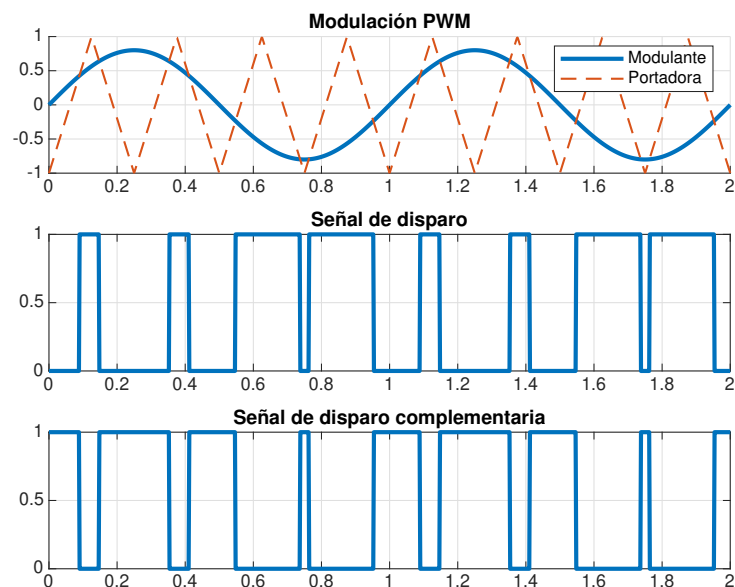


Figura 1.16: Ejemplo de una modulación SPWM

La señal modulante es aquella cuya forma debería replicarse (o aproximarse) en otra variable del circuito electrónico de potencia, mientras que, la señal portadora es la señal que codificará la señal modulante. Es importante que la señal portadora tenga una frecuencia superior a la señal que se requiere modificar. Mediante comparaciones, se puede controlar el ciclo de trabajo $\delta(t)$ de un interruptor con la siguiente estructura:

$$\delta(t) = \begin{cases} \text{Si } m(t) \geq p(t) & \text{entonces } \delta(t) = 1 \\ \text{Si } m(t) < p(t) & \text{entonces } \delta(t) = 0 \end{cases} \quad (1.5)$$

En los circuitos electrónicos de potencia, donde las topologías en puente son prepon-

derantes, es indispensable también obtener las señales de disparo complementarias. La función más importante de estas señales de disparo es impedir el encendido simultáneo de todos los interruptores de un ramal del inversor, evitando la aparición de cortocircuitos. Por extensión, el valor del disparo $\bar{\delta}(t)$ complementario es la negación de la función de disparo normal. Considerando que, el funcionamiento de los interruptores reales presenta periodos de apagado y encendido, se suelen añadir tiempos muertos entre el encendido de un interruptor y el apagado de otro interruptor del mismo ramal para reducir la posibilidad de encendido simultáneo [21].

En este trabajo, se considera la presencia de interruptores con comportamiento puramente resistivo. Es decir, no se considera la existencia de los periodos de encendido y apagado provocados por las inductancias y capacitancias parásitas, propios de los interruptores reales.

Es importante mencionar la existencia de un fenómeno conocido como *sobremodulación*. Este fenómeno ocurre cuando la amplitud de la señal modulante es mayor o igual a la amplitud de la señal portadora. Cuando esto sucede, la posibilidad de no realizar comparaciones adecuadas es muy alta, provocando que el ciclo de trabajo de los interruptores no varíe en intervalos temporales mayores al periodo de conmutación, distorsionando la señal y propiciando la pérdida de información, si bien puede permitir la obtención de mayores voltajes a la salida. Para impedir este fenómeno pernicioso, la práctica usual es submodular la señal, es decir, procurar que la amplitud de la señal modulante sea menor a la señal portadora. Generalmente la relación entre la amplitud de la señal portadora y modulante debe estar en un rango óptimo:

$$0,85 \leq \frac{m(t)_{\text{máx}}}{p(t)_{\text{máx}}} \leq 0,95 \quad (1.6)$$

La relación recomendable de frecuencia entre la onda modulante y la onda portadora es de 1 a 21. Es decir, la frecuencia de la portadora debe ser como mínimo veintiún veces mayor a la frecuencia de la onda modulante para tener una modulación que permita el uso de filtros de tamaño mínimo. También, se recomienda que la relación entre estas variables sea un múltiplo de tres [21].

1.4.3.3. Sistemas de referencia y transformaciones de coordenadas en sistemas trifásicos

La teoría de control está bien desarrollada para el estudio de sistemas lineales. Sin embargo, los sistemas eléctricos de potencia trabajan con señales variantes en amplitud y frecuencia respecto al tiempo. Frente a esta problemática, existen alternativas plausibles que permiten el control de variables periódicas. Estas técnicas permiten trabajar estas variables físicas cambiantes en el tiempo como variables de magnitud constante. Inicialmente se considera un sistema de variables trifásicas dadas por las siguientes expresiones:

$$\begin{aligned}f_a(t) &= A \cdot \text{sen}(\omega t + \theta_0) \\f_b(t) &= A \cdot \text{sen}\left(\omega t + \theta_0 - \frac{2\pi}{3}\right) \\f_c(t) &= A \cdot \text{sen}\left(\omega t + \theta_0 + \frac{2\pi}{3}\right)\end{aligned}$$

En las siguientes líneas, se presentarán los conceptos inherentes a las transformaciones de Clarke, y las transformaciones de Park.

Transformada de Clarke Desarrollada por la ingeniera eléctrica Edith Clarke, convierte un conjunto de variables trifásicas en el sistema de referencia abc a un conjunto de variables en el sistema de referencia $\alpha\beta 0$. Existen dos matrices, la matriz de amplitud invariante y la matriz de potencia invariante. En este trabajo se usará la matriz de amplitud invariante, la cual viene dada por la siguiente expresión [22]:

$$\mathbf{T}_{\alpha\beta 0} = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \quad (1.7)$$

Así pues, se puede obtener un conjunto de funciones en el sistema de referencia deseado mediante la operación:

$$\begin{bmatrix} f_\alpha \\ f_\beta \\ f_0 \end{bmatrix} = \mathbf{T}_{\alpha\beta 0} \times \begin{bmatrix} f_a \\ f_b \\ f_c \end{bmatrix} \quad (1.8)$$

Este marco de referencia tiene la ventaja de eliminar una fase en el caso de sistemas balanceados. Por tanto, un sistema trifásico balanceado puede ser trabajado como un sistema de dos fases con una diferencia de 90 grados entre fase y fase. Generalmente los desbalances suelen ser despreciables. De forma análoga, si se desea transformar un conjunto de variables en el marco $\alpha\beta 0$ al marco abc puede usarse la matriz de transformación inversa [22]:

$$\mathbf{T}_{\alpha\beta 0}^{-1} = \begin{bmatrix} 1 & 0 & 1 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} & 1 \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} & 1 \end{bmatrix} \quad (1.9)$$

Transformada de Park Esta transformación toma un conjunto de variables variantes en el tiempo, situadas en un marco de referencia abc . Mediante la rotación continua en un sistema de referencia que gira un ángulo instantáneo θ de una frecuencia ω arbitraria, se obtiene un conjunto de variables invariantes en el tiempo situadas en un marco de referencia $dq0$. Es importante mencionar que para lograr este cometido, se debe trabajar con las variables obtenidas en la transformada de Clarke. Así pues, la transformación de Park se logra mediante la siguiente operación [22]:

$$\begin{bmatrix} f_d \\ f_q \\ f_0 \end{bmatrix} = \mathbf{T}_{dq0} \times \begin{bmatrix} f_\alpha \\ f_\beta \\ f_0 \end{bmatrix} \quad (1.10)$$

Donde la matriz de transformación viene dada por:

$$\mathbf{T}_{dq0} = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1.11)$$

De igual manera, se puede realizar la operación inversa mediante una transformación inversa. Dado que la matriz es ortogonal, se puede afirmar que:

$$\mathbf{T}_{dq0} = \mathbf{T}_{dq0}^{-1} \quad (1.12)$$

En la Fig. 1.17 se puede apreciar cómo funcionan estos cambios de referencia. Es notable que se pueden transformar funciones variantes en el tiempo a funciones invariantes, facilitando enormemente su análisis y control. Sin embargo, queda una interrogante: ¿cómo

se calcula el ángulo θ usado para la transformación al sistema de referencia $dq0$? Es evidente que, este ángulo no procede de un cálculo matemático directo, y debe ser obtenido a partir de mediciones. A continuación, se hablará de la técnica utilizada para obtenerlo.

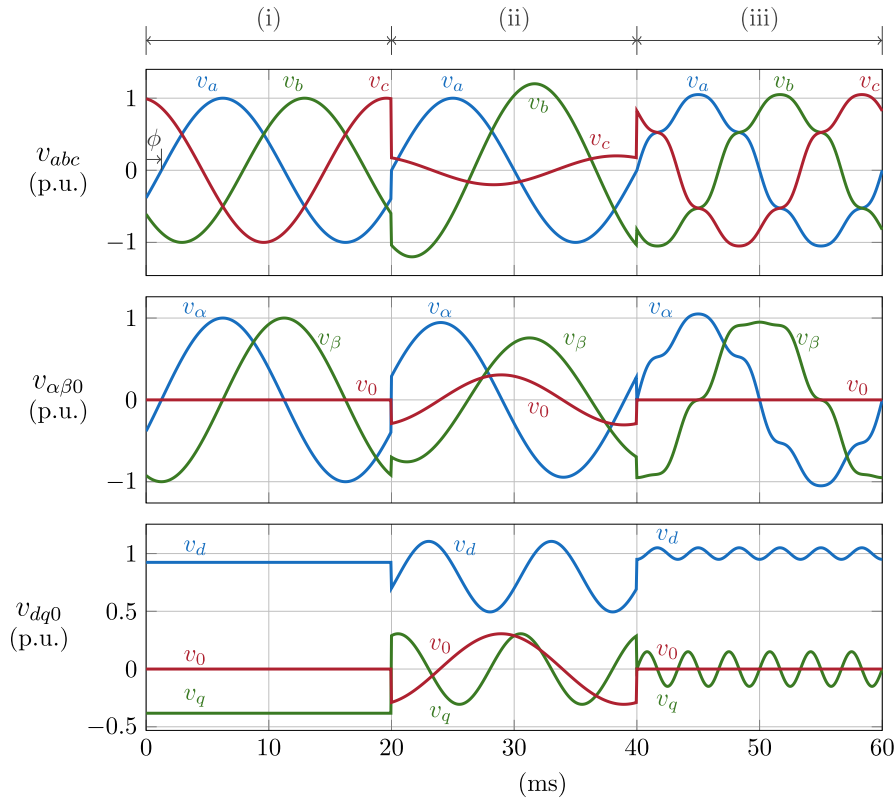


Figura 1.17: Transformaciones de Clarke y Park aplicadas a un sistema trifásico bajo tres condiciones: sistema balanceado, sistema no balanceado y sistema balanceado con armónicos [22].

Lazo de seguimiento de fase (PLL) Para la obtención del ángulo que se utiliza para la transformación de coordenadas, se deben realizar algunas mediciones obtenidas a partir de las variables variantes en el tiempo.

El concepto del lazo de seguimiento de fase puede ejemplificarse con una situación fácilmente imaginable en la vida real. Supóngase la existencia de un automóvil en una pista circular, que la recorre a una velocidad arbitraria. Otro usuario con otro automóvil equipado con dispositivos de medición se encarga de determinar la posición del automóvil objetivo. Para cumplir este objetivo, ingresa en la pista circular y persigue al otro automóvil. Así, existen tres situaciones que pueden suceder:

- Que el automóvil objetivo se encuentre adelantado respecto al automóvil de medición.
En este caso, el automóvil medidor debe aumentar su velocidad angular para estar

igualado (en fase) respecto al automóvil objetivo.

- Que el automóvil objetivo se encuentre retrasado respecto al automóvil de medición. En este caso, el automóvil medidor debe reducir su velocidad angular.
- Que el automóvil objetivo se encuentre en fase respecto al automóvil de medición. En este caso, el automóvil debe mantener la velocidad angular.

Este mismo concepto se aplica a las señales variantes en el tiempo procedentes del marco de referencia $\alpha\beta 0$. Como la integral de la velocidad angular es la posición angular, y la velocidad angular está íntimamente relacionada con la frecuencia $\omega = 2\pi f$, entonces, una medición del error angular entre la señal a medir y la señal de seguimiento permite acelerar o retrasar la velocidad de seguimiento [23]. Es evidente que esto se logra mediante un lazo de control, el cual puede apreciarse en la Fig. 2.17.

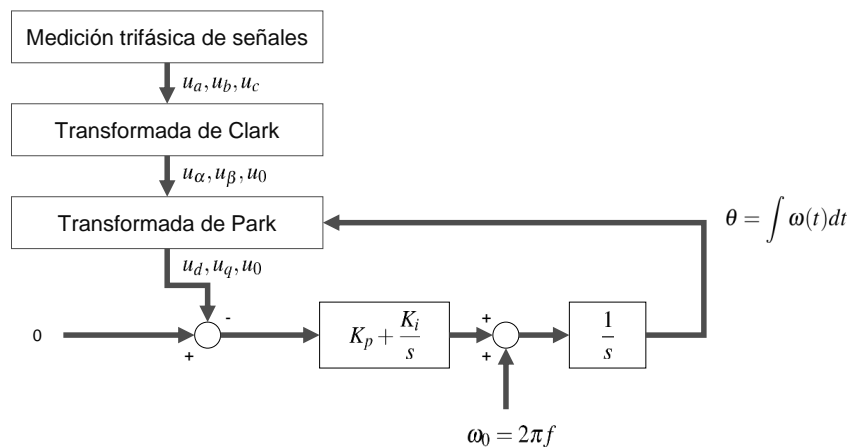


Figura 1.18: Diagrama de bloques de un lazo de control PLL que utiliza el marco de referencia $dq0$

En la Fig. 1.18 se aprecia que, el control se realiza mediante el uso de las variables dadas por el marco de referencia $dq0$. Esto permite trabajar el lazo de control con señales invariantes en el tiempo. Por otro lado, se busca que el error de posición entre la señal a medir y la señal de seguimiento sea cero. La referencia se ajusta en cero, y la señal de medición puede ser la señal de eje directo o la señal de eje de cuadratura. Al cerrar el lazo, se consigue que la señal seleccionada esté alineada con el marco de referencia rotacional [23]. Esto conlleva ventajas muy importantes para los lazos de control que se encargan de las corrientes y las potencias del inversor trifásico, las cuales se mencionarán posteriormente.

Ignorando el efecto de la realimentación ocasionada por el término constante ω_0 , se puede decir que las funciones de transferencia del PLL vienen dadas por las siguientes expresiones:

$$G_{la} = \left(K_p + \frac{K_i}{s} \right) \cdot \frac{1}{s}$$

$$1 + G_{la} = \frac{s^2 + K_p \cdot s + K_i}{s^2}$$

$$G_{lc} = \frac{G_{la}}{1 + G_{la}} = \frac{K_p \cdot s + K_i}{s^2 + K_p \cdot s + K_i}$$

Si en la ecuación anterior, se cumple que $K_p \ll K_i$, entonces la función de transferencia puede tratarse como:

$$G_{lc} = \frac{K_i}{s^2 + K_i} \quad K_p \ll K_i \quad (1.13)$$

Aproximando esta función de transferencia de lazo cerrado a la forma estándar de segundo orden, se pueden obtener los siguientes parámetros de sintonización:

$$K_i = (\omega_n)^2 = (2\pi f_n)^2$$

$$K_p = 2\lambda\sqrt{K_i} \quad \lambda < 0,1$$

Es importante mencionar que, estos parámetros de sintonización funcionan en tiempo continuo. Sin embargo, para el tiempo discreto, se deben realizar algunas consideraciones adicionales. Además, es importante mencionar que las señales obtenidas mediante la transformación $dq0$ deberían normalizarse, es decir, el módulo del rango de la señal debería estar entre cero y uno.

Cálculo de potencias en el sistema de referencia $dq0$ La modificación de los sistemas de referencia para los sistemas trifásicos permiten simplificar los cálculos de potencia activa y reactiva considerando que las magnitudes variantes en el tiempo se transforman en magnitudes invariantes en el tiempo mediante el uso de las transformaciones mencionadas anteriormente. Así pues, en un sistema trifásico que se asume lo más equilibrado posible, se tienen los voltajes v_a, v_b, v_c y las corrientes i_a, i_b, i_c . Usando las matrices de transformación, se pueden obtener los voltajes v_d, v_q y las corrientes i_d, i_q . Las componentes cero v_0, i_0

se consideran despreciables y no se usan para el cálculo de la potencia. Así, la potencia activa y la potencia reactiva pueden calcularse con las siguientes fórmulas [24]:

$$p(t) = \frac{3}{2} (v_d i_d + v_q i_q) \quad (1.14)$$

$$q(t) = \frac{3}{2} (v_q i_d - v_d i_q) \quad (1.15)$$

Gracias al PLL, se puede forzar a que el voltaje de eje directo o el voltaje de eje de cuadratura estén alineados con el vector de la tensión de fase. Si se escoge el voltaje de eje de cuadratura, éste se hace cero, mientras que, se trabaja solamente con el voltaje de eje directo, el cual toma el valor del voltaje pico entre líneas. Así, el cálculo de la potencia activa y reactiva se simplifica aún más:

$$p(t) = \frac{3}{2} v_d i_d = \frac{3}{2} \sqrt{2} v_{ll} i_d \quad (1.16)$$

$$q(t) = -\frac{3}{2} v_d i_q = -\frac{3}{2} \sqrt{2} v_{ll} i_q \quad (1.17)$$

Con este arreglo, las variables que controlan la entrega de potencia activa o potencia reactiva vienen dadas por i_d, i_q ya que, los voltajes v_d, v_q dependen de la red. Las corrientes i_d, i_q pueden controlarse la mediante modulación del inversor y mediante lazos de control que serán explicados más adelante. Cabe mencionar, que las fórmulas anteriores asumen un sistema trifásico balanceado.

1.4.3.4. Control de lazos de corriente

El control de los inversores trifásicos se realiza mediante el uso de transformaciones de sistemas de referencia. Así pues, el control de las variables invariantes en el tiempo $dq0$ es indispensable para esta finalidad. En la Fig. 1.19 se puede apreciar un diagrama de bloques que explica de forma concisa el funcionamiento del control de un inversor trifásico.

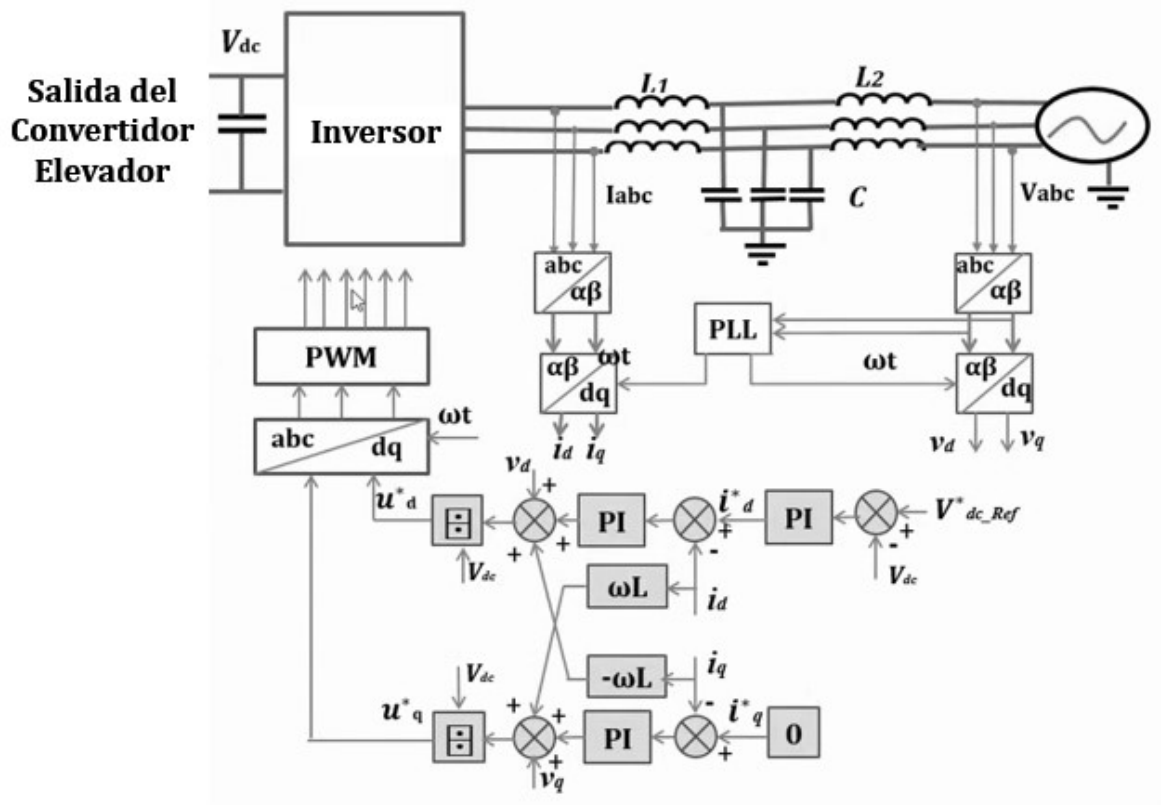


Figura 1.19: Diagrama de bloques para un control de la modulación del inversor trifásico. Se usa un control de voltaje que actúa sobre el eje directo, y un control de corriente que actúa sobre el eje de cuadratura [25].

Como se puede apreciar en el esquema, el control de un inversor trifásico es un control anidado, en el que el lazo interno se encarga de manejar las corrientes, así como el lazo externo se encarga de manejar las potencias o voltajes [26]. Para este efecto, se trabajará con un esquema simplificado de un circuito inversor, en base a la Fig. 1.20.

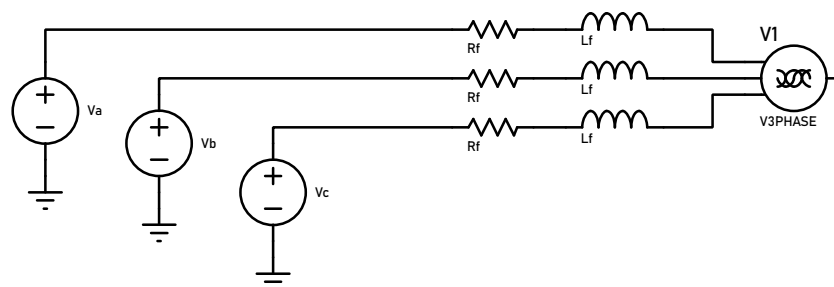


Figura 1.20: Esquema de funcionamiento de un inversor monofásico en un cierre de interruptores arbitrario.

De acuerdo con la Fig. 1.20, se puede establecer un lazo de corriente por fase en el que:

$$\begin{aligned}v_{ai} &= R_T \cdot i_a + L \cdot \frac{di_a}{dt} + v_{ag} \\v_{bi} &= R_T \cdot i_b + L \cdot \frac{di_b}{dt} + v_{bg} \\v_{ci} &= R_T \cdot i_c + L \cdot \frac{di_c}{dt} + v_{cg}\end{aligned}$$

Las ecuaciones anteriores pueden escribirse en forma fasorial, donde $\vec{v}_i = \{v_{ai}, v_{bi}, v_{ci}\}^T$ representa al vector de voltajes del inversor, $\vec{v}_g = \{v_{ag}, v_{bg}, v_{cg}\}^T$ representa al vector de voltajes de la red, y $\vec{i} = \{i_a, i_b, i_c\}^T$. Además, $R_T = R_f + R_{on}$, es decir, la suma de la resistencia del filtro y la resistencia de encendido de un interruptor del inversor, y L es la inductancia del filtro. En esta etapa se ignora el condensador de filtro:

$$\vec{v}_i = R_T \cdot \vec{i} + L \cdot \frac{d\vec{i}}{dt} + \vec{v}_g$$

Al realizar el cambio de coordenadas mediante las matrices de transformación, se obtienen las siguientes expresiones:

$$\vec{v}_{i(dq)} = R_T \cdot \vec{i}_{dq} + L \cdot \frac{d\vec{i}_{dq}}{dt} + \omega L \vec{i}_{dq} + \vec{v}_{g(dq)}$$

Conociendo que el componente de eje directo está en el eje real y el componente de eje de cuadratura está en el eje imaginario, se obtienen dos ecuaciones para el cálculo de las corrientes:

$$L \frac{di_d}{dt} + R_T i_d = v_{id} - v_{gd} + \omega L i_q \quad (1.18)$$

$$L \frac{di_q}{dt} + R_T i_q = v_{iq} - v_{gq} - \omega L i_d \quad (1.19)$$

Si $u_d = v_{id} - v_{gd} + \omega L i_q$, $u_q = v_{iq} - v_{gq} - \omega L i_d$, entonces la expresión puede reducirse aún más:

$$\begin{aligned}L \frac{di_d}{dt} + R_T i_d &= u_d \\L \frac{di_q}{dt} + R_T i_q &= u_q\end{aligned}$$

De esta forma, aplicando transformadas de Laplace, se puede obtener la función de transferencia de ambas corrientes:

$$I_d(s) = \frac{1}{Ls + R_T} \cdot U_d(s) \quad (1.20)$$

$$I_q(s) = \frac{1}{Ls + R_T} \cdot U_q(s) \quad (1.21)$$

Resulta evidente que la inclusión de un lazo cerrado de control puede mejorar la respuesta temporal del sistema para que no dependa de los parámetros internos del circuito. Así pues, se propone un lazo de control con un controlador PI, lo que se puede observar en la Fig. 1.21.

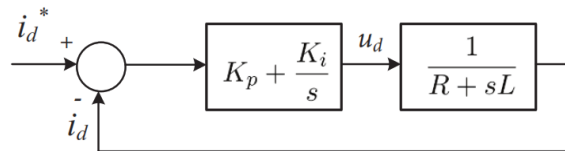


Figura 1.21: Diagrama de bloques simplificado del control del lazo interno de corriente [26].

La función de transferencia de lazo abierto del sistema viene dada por:

$$G_{la}(s) = \frac{K_p}{L} \frac{1}{s} + \frac{K_i}{K_p} \frac{1}{s + \frac{R}{L}}$$

Si se usa una técnica de cancelación de polos, entonces $\frac{K_i}{K_p} = \frac{R}{L}$, con lo que la función de transferencia de lazo abierto se transforma en:

$$G_{la}(s) = \frac{K_p}{L} \frac{1}{s}$$

Y la función de transferencia en lazo cerrado:

$$G_{lc}(s) = \frac{1}{\tau s + 1}$$

Donde los parámetros de sintonización son los siguientes [26]:

$$K_p = \frac{L}{\tau} \quad (1.22)$$

$$K_i = \frac{R_T}{\tau} \quad (1.23)$$

Siendo τ la constante de tiempo en la que se busca que la respuesta en lazo cerrado alcance el 63 % de su estado estable. El valor de τ se puede establecer en:

$$\tau_{LI} = \frac{1}{40f_{red}}$$

Es decir, se desea buscar que el tiempo en el que funcione el control de corriente sea el 10 % del periodo de la red. Para una frecuencia de 60 Hz, se buscaría que el tiempo de establecimiento esté en el orden de los 420 microsegundos. En caso de querer utilizar un control digital para este lazo de control, el tiempo de muestreo debería ser por lo menos 10 veces menor a la constante de tiempo. Con el control de corriente, se pueden modificar $u_d = v_{id} - v_{gd} + \omega L i_q$, $u_q = v_{iq} - v_{gq} - \omega L i_d$. Mediante el uso de las antitransformadas de Clark y Park, se pueden regenerar los voltajes de referencia u_{abc} , los que se usarán para modular las señales SWPM que controlarán la apertura o cierre de los interruptores del inversor trifásico.

1.4.3.5. Filtros pasabajos

Con la finalidad de filtrar el ruido causado por las mediciones, perturbaciones o dinámicas de los procesos, se pueden usar filtros que permitan atenuar o eliminar estos fenómenos perniciosos. En el caso del ruido, teóricamente se habla de una señal de alta frecuencia. Por este motivo, se deben filtrar los componentes de alta frecuencia mediante el uso de diferentes técnicas. Una de estas es el empleo de funciones de transferencia que implementan filtros pasabajos. La función continua de un filtro pasabajos viene dada por:

$$G_f(s) = \frac{1}{\tau s + 1} \quad (1.24)$$

Donde τ es la constante de tiempo del filtro pasabajos. Esta constante de tiempo está íntimamente relacionada con la dinámica del sistema. En el caso de sistemas de primer orden, un tiempo adecuado es por lo menos cuatro veces más rápido que el tiempo de establecimiento. Una constante de tiempo muy pequeña reproduce la dinámica del sistema con mayor exactitud pero a costa de perder capacidad de filtrado, mientras que una constante de tiempo muy grande tiene una capacidad de filtrado tan grande que puede enmascarar o cambiar totalmente la dinámica del proceso. En este proyecto, después de realizar algunas mediciones empíricas, se utiliza una constante de tiempo de 500 microsegundos, con un tiempo de muestreo igual al tiempo de integración. Se usa además el método de discretización de Euler por adelanto:

$$s = \frac{T_s}{z - 1} \quad (1.25)$$

1.4.3.6. Sintonización de controladores PID

Como se ha podido apreciar en las líneas anteriores, para lograr una mejor respuesta de los sistemas basados en electrónica de potencia, el control proporcional-integral-derivativo sigue siendo bastante utilizado a pesar que algunos autores consideran esta metodología como obsoleta [27]. No obstante, de acuerdo con algunos estudios, por lo menos el 30 % de controladores PID están ajustados en para un funcionamiento manual, y el 65 % de los controladores ajustados automáticamente están mal sintonizados [28]. La inexistencia de una nomenclatura unificada hace más complicada la divulgación y el estudio de los controladores PID. Existen dos grandes métodos de sintonización: los métodos a lazo abierto y los métodos a lazo cerrado. En la electrónica de potencia, los controladores PI son los más utilizados, ya que el horizonte temporal en el que actúan es tan pequeño, que no representa una variable perceptible para los operadores y diseñadores.

Método de Lazo Cerrado de Ziegler-Nichols Propuesto en 1942, aún sigue siendo de gran utilidad en la sintonización de controladores. Los pasos a seguir son [29]:

1. Eliminar los efectos del control derivativo e integral, manteniendo solamente el control proporcional.
2. Aplicar una perturbación en el lazo cerrado, modificando la entrada un 20 %.
3. Ajustar la ganancia proporcional hasta que la salida del sistema oscile de forma continua.
4. Anotar el valor de la ganancia crítica última K_{cu} que ha provocado la oscilación.
5. Anotar el valor del periodo de la curva de respuesta T_u .
6. Sintonizar el controlador PI de acuerdo con la tabla 1.2. Se puede apreciar un esquema gráfico en la Fig. 1.22.

Tabla 1.2: Sintonización de un controlador PI con el método de Ziegler-Nichols [29].

| Parámetro | Valor |
|-----------|----------------------|
| K_p | $\frac{K_{cu}}{2,2}$ |
| τ_i | $\frac{T_u}{1,2}$ |

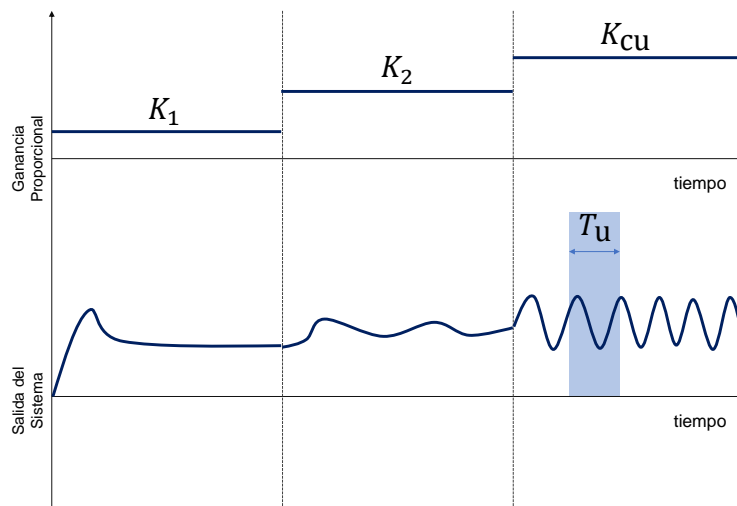


Figura 1.22: Esquema gráfico del procedimiento utilizado para realizar una sintonización mediante el método de Ziegler-Nichols.

Método de Lazo Abierto de Dahlin Este método requiere la desconexión del lazo de control, para introducir una perturbación manual en el proceso. La prueba consiste en realizar una prueba de respuesta al escalón e identificar, mediante la respuesta temporal, la función de primer orden con retardo (FOPDT) más adecuada. Con los parámetros de la función de primer orden, se puede sintonizar un controlador PI.

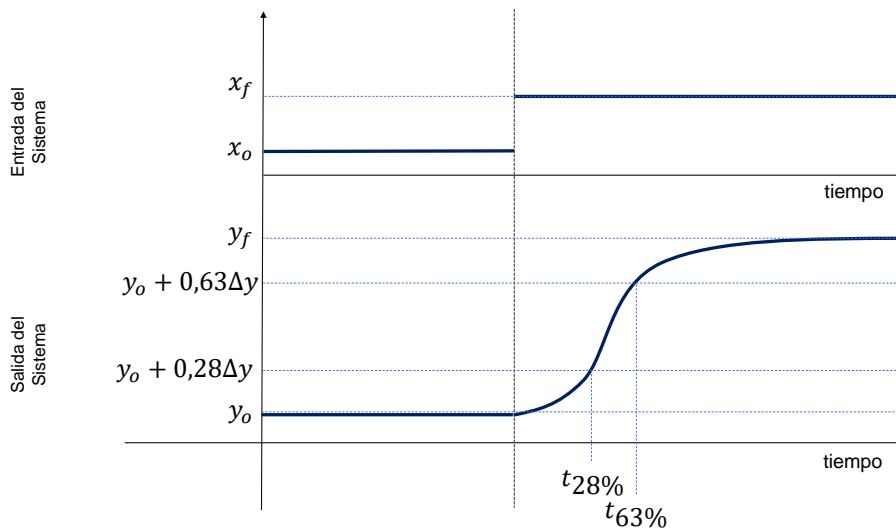


Figura 1.23: Esquema gráfico del procedimiento utilizado para realizar una identificación FOPDT mediante el método de los dos puntos.

En la Fig. 1.23 se pueden apreciar los hitos gráficos más importantes para identificar una función de primer orden con retardo mediante el método de los dos puntos. Se tiene una FOPDT de la forma:

$$G_p(s) = \frac{K}{\tau s + 1} \cdot e^{-t_0 \cdot s} \quad (1.26)$$

1. Desconectar el control en lazo cerrado, y esperar a que el sistema se estabilice.
2. Provocar una perturbación en la entrada, preferiblemente una variación del 20%. Anotar el valor inicial de la entrada x_0 y el valor final de la entrada x_f . Calcular el valor de $\Delta x = x_f - x_0$.
3. Registrar la variación de respuesta en la salida hasta que el sistema se estabilice nuevamente.
4. Registrar los valores inicial y_0 y final y_f de la salida. Calcular el valor de $\Delta y = y_f - y_0$.
5. Identificar los valores $y_{28} = y_0 + 0,28\Delta y$ y $y_{63} = y_0 + 0,63\Delta y$
6. Identificar los tiempos t_{28} , t_{63} , correspondientes a y_{28} , y_{63} respectivamente.

Con estos valores, se pueden encontrar los valores de la FOPDT, y los valores sugeridos para el controlador con el método de Dahlin [29].

Tabla 1.3: Parametrización de una FOPDT con el método de dos puntos, y sintonización de controladores

| | Parámetro | Valor |
|----------------|-----------|--|
| FOPDT | K | $\frac{\Delta y}{\Delta x}$ |
| | τ | $1,5 (t_{63} - t_{28})$ |
| | t_0 | $t_{63} - \tau$ |
| Controlador PI | K_p | $\frac{1}{2K} \cdot \left(\frac{\tau}{t_0} \right)$ |
| | τ_i | τ |
| | K_i | $\frac{K_p}{\tau_i}$ |

Esquema anti-windup Cuando se trabaja con variables cuya salida está limitada dentro de un rango determinado, como el caso de los ciclos de trabajo (limitados de 0 a 1), o apertura de válvulas (0 a 100%), entre otros, se puede utilizar un esquema para prevenir la saturación de los controladores causada por la acumulación de la integral del error [29]. El error acumulado tiene el efecto de impedir una mayor rapidez del controlador con la consecuente degradación de los índices de control. Para evitar estos inconvenientes, se utilizan algunos métodos. El método que se utilizará aquí es el de cálculo en retroceso (*back-calculation*). La fórmula para este esquema viene dada por:

$$K_{bc} = \frac{K_p}{K_i} \quad (1.27)$$

Se puede programar este elemento solamente con la modificación del bloque *PID Controller* en Simulink. Es importante mencionar que, su uso es igual de válido en controladores digitales.

1.4.4. ANÁLISIS DE SISTEMAS ELÉCTRICOS DE POTENCIA

Un Sistema Eléctrico de Potencia es un conjunto de dispositivos interconectados que convierten energía potencial de las fuentes primarias -solar, térmica, geotérmica, nuclear, etcétera- a energía eléctrica, transportándola y distribuyéndola a los usuarios finales [30]. Los componentes de un sistema eléctrico de potencia pueden agruparse en tres grandes grupos: componentes de generación, de transmisión y distribución. Con la finalidad de agrupar mejor los elementos que componen un SEP, los elementos confluyen en elementos conceptuales conocidos como *barras*, las cuales pueden describirse como nodos en los que confluyen elementos que comparten un voltaje nominal, en los que se pueden calcular variables eléctricas de interés. El análisis de sistemas eléctricos de potencia provee las

herramientas para conocer, monitorear y tomar decisiones que involucran a sus elementos constitutivos. En esta sección se realizará una revisión bibliográfica de los mecanismos utilizados, de forma que resulten convenientes para la finalidad de este trabajo, sin profundizar más allá de los conceptos fundamentalmente requeridos.

1.4.4.1. Elementos de un SEP

En principio, la operación exitosa de un sistema eléctrico de potencia requiere algunas condiciones fundamentales [31]:

- La potencia de generación debe ser mayor o igual a la potencia demandada por las cargas.
- Las magnitudes de los voltajes de las barras deben ser muy próximos a los voltajes nominales de las barras.
- Los generadores deben operar dentro de sus límites de potencia activa y reactiva.
- Las líneas de transmisión y los transformadores no deben estar sobrecargados.

Sin embargo, hay que considerar la existencia de los elementos de un Sistema Eléctrico de Potencia, así como su importancia dentro del mismo para entender cómo su interrelación puede o no cumplir los parámetros anteriormente mencionados:

1. **Generación.** En esta etapa se realiza la conversión de energía potencial a energía eléctrica. La gran mayoría de métodos para generar energía eléctrica se basan en una conversión electromecánica en la que turbinas son movidas por agua o vapor para producir variaciones en el campo magnético de un generador. La generación fotovoltaica y piezoeléctrica no utilizan este método de conversión electromecánica. Los generadores son capaces de producir potencia activa o reactiva de acuerdo a los requisitos de las cargas.
2. **Transmisión.** En esta etapa se transmiten grandes cantidades de energía desde las centrales de generación a los centros de consumo. El corazón de la transmisión eléctrica está en las líneas de transmisión, que son conductores usados como medio físico para tal efecto. Sin embargo, la transmisión depende también de elementos llamados transformadores, los cuales son capaces de modificar el voltaje que puede ser procesado por las cargas, de acuerdo a sus requerimientos. La transmisión está sujeta a pérdidas de energía, las cuales deben mantenerse dentro de límites razonables.

- Distribución. En esta parte, la energía eléctrica llega a los consumidores finales. Para controlar el consumo, se pueden usar dispositivos como seccionadores e interruptores que pueden conectar y desconectar las cargas. Los transformadores de distribución son esenciales, puesto que permiten la disminución del voltaje de transmisión a valores seguros y utilizables por las sociedades humanas.

Por supuesto, el análisis de un sistema eléctrico de potencia es más complejo. Se ha omitido hablar de las protecciones eléctricas, que son dispositivos con la capacidad de detectar problemas en la red eléctrica y actuar eficazmente para mitigar o anular los efectos nocivos de situaciones complejas como cortocircuitos, sobrecargas, fallas, etcétera. Los sistemas eléctricos de potencia son entonces el motor de la productividad y el desarrollo tecnológico contemporáneo. En la Fig. 1.24 se puede examinar un sistema eléctrico de potencia típico, en el que se aprecia cómo se interrelacionan los elementos como generadores, transformadores, líneas de transmisión, cargas, proporcionando ejemplos de existencia en el sistema nacional interconectado de Ecuador.

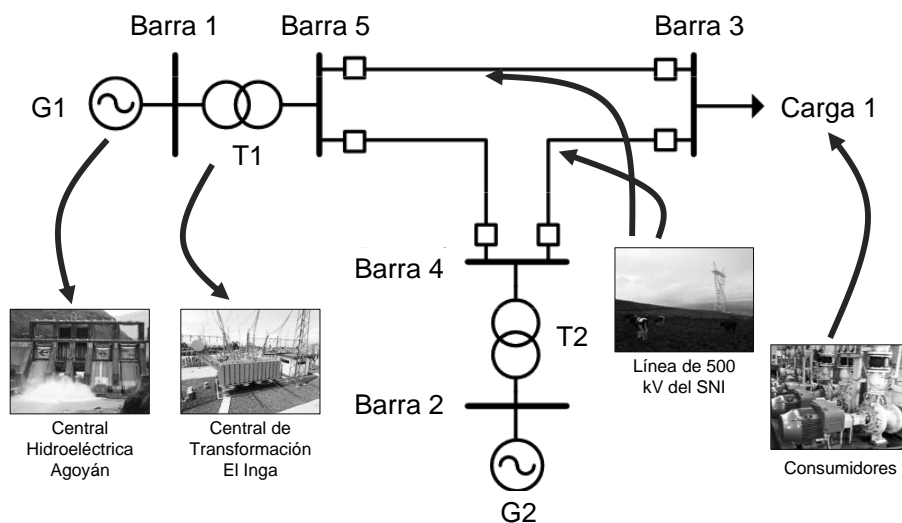


Figura 1.24: Sistema eléctrico de potencia simple con ejemplos en el Sistema Nacional Interconectado de Ecuador.

1.4.4.2. Flujos de potencia

La herramienta básica por excelencia para analizar un SEP es el flujo de potencia (o flujo de carga). Esta herramienta permite determinar la magnitud de voltaje y el ángulo en cada barra en un sistema trifásico balanceado en una resolución temporal de estado estable. Un flujo de potencia tiene la capacidad de calcular las potencias activas y reactivas de generación, consumo y pérdidas en las interconexiones. Su estudio ha sido desarrollado

a lo largo de casi 70 años de investigación, especialmente en los computadores digitales [32].

El análisis nodal y de mallas que se conocen desde los fundamentos de los circuitos eléctricos no es compatible con las necesidades de un sistema eléctrico de potencia puesto que las variables para la resolución de estos problemas se dan en términos de potencia, no de impedancia.

Para iniciar el problema de flujo de potencia se parte del diagrama unifilar del sistema, desde donde se pueden obtener los datos de las variables de entrada, las cuales se resumen en datos de barras, línea de transmisión, y transformadores. En la figura 1.25 se puede apreciar un esquema generalizado de un diagrama unifilar.

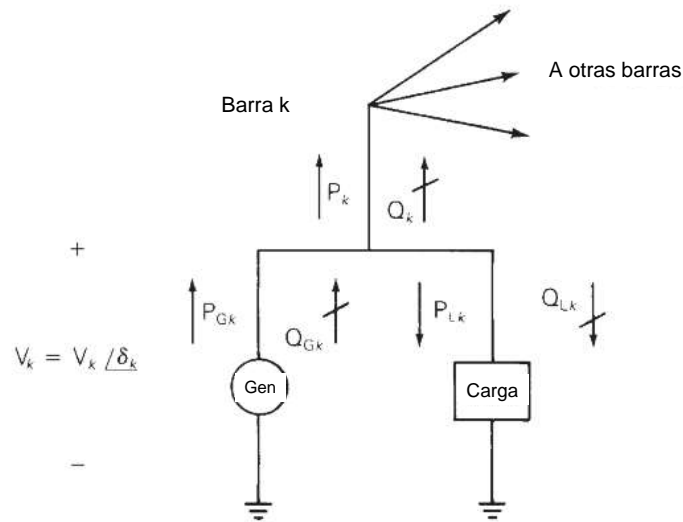


Figura 1.25: Esquema de las variables de barra V_k , δ_k , P_k , Q_k [31].

Existen tres tipos de barras:

1. Barra Slack. Existe solamente una barra Slack. Esta barra es una barra de referencia en la que la magnitud del voltaje es uno y el ángulo es cero. El flujo de potencia calcula el valor de potencia activa y reactiva para la barra Slack.
2. Barra PQ. En esta barra, los datos de entrada son la potencia activa y la potencia reactiva mientras que el flujo de potencia calcula los valores de voltaje y ángulo. La mayoría de barras en un sistema eléctrico de potencia, entran en esta categoría, especialmente en las relacionadas con las cargas.
3. Barra PV. En estas barras, los datos de entrada son la potencia activa y el voltaje de bus, mientras que la potencia reactiva y el ángulo se calculan en el flujo de potencia.

Ejemplos de barras PV son barras con generadores, condensadores de compensación, o transformadores con taps.

A continuación se describirá el proceso matemático inherente al planteamiento y la resolución de los flujos de potencia. Considerando la ley de Ohm básica, se puede modificarla para que dependa de las admitancias en vez de las impedancias:

$$\begin{aligned} V &= I \cdot R \\ I &= \frac{1}{R} \cdot V \\ I &= Y \cdot V \end{aligned}$$

Esto genera ecuaciones nodales para cada bus:

$$\mathbf{I} = \mathbf{YV} \quad (1.28)$$

Además, se debe recordar que la potencia compleja para un bus k viene dada por:

$$S_k = P_k + jQ_k = V_k \cdot I_k^* \quad (1.29)$$

Hay que recordar que I_k^* significa una operación con la conjugada de la corriente I_k . Reemplazando la ecuación 1.28 en la ecuación 1.29, considerando que existen N barras en el sistema eléctrico de potencia de estudio se puede obtener una expresión generalizada para la potencia compleja.

$$P_k + jQ_k = V_k \left[\sum_{n=1}^N Y_{kn} V_n \right]^* \quad k = 1, 2, \dots, N \quad (1.30)$$

Considerando que el voltaje V_n tiene un ángulo δ_n , y la admitancia Y_{kn} tiene un ángulo θ_{kn} , la potencia aparente se puede escribir en notación compleja.

$$P_k + jQ_k = V_k \sum_{n=1}^N Y_{kn} V_n e^{j(\delta_k - \delta_n - \theta_{kn})} \quad (1.31)$$

De forma que se puede calcular la potencia activa y la potencia reactiva pueden descomponerse

$$P_k = V_k \sum_{n=1}^N Y_{kn} V_n \cos(\delta_k - \delta_n - \theta_{kn}) \quad (1.32)$$

$$Q_k = V_k \sum_{n=1}^N Y_{kn} V_n \sen(\delta_k - \delta_n - \theta_{kn}) \quad (1.33)$$

El objetivo de los flujos de potencia está en resolver las últimas dos ecuaciones de forma que se puedan obtener los valores de potencia activa, reactiva, magnitudes de voltaje y ángulos en todas las barras que conforman el sistema eléctrico de potencia. Como se aprecia, la formulación fasorial de las expresiones utilizadas, limita la representación de las respuestas en un dominio temporal instantáneo, limitando su aplicación en aplicaciones de estado estable donde las variables que cambian a una frecuencia mayor a la frecuencia de la red no son consideradas.

La resolución de los flujos de potencia se realiza mediante operaciones matriciales iterativas. Los métodos de Gauss-Siedel son empleados, pero el más confiable y utilizado es el método de Newton-Raphson, aunque requiere el cálculo de matrices jacobianas que aumentan el consumo de memoria. El método de Gauss-Siedel requiere unas 100 iteraciones, mientras que un método Newton-Raphson suele converger en menos de 10 iteraciones, motivo por el cual su utilización está plenamente justificada. Además, los métodos matriciales y el álgebra iterativa están presentes en los computadores digitales desde sus inicios, por lo que la resolución mediante computador no resulta especialmente compleja [31].

Cabe mencionar, que los generadores sincrónicos, los transformadores y las líneas de transmisión se integran al flujo de potencia mediante sus modelos de admitancia y/o impedancia adaptados para una red trifásica modelada como red monofásica (ya que se asume la existencia de un sistema trifásico balanceado). Además, las barras suelen tener voltajes nominales diferentes entre ellos, por lo que se adopta un método conocido como *sistema por unidad* para adoptar voltajes, corrientes, impedancias de referencia que faciliten aún más la resolución.

1.4.4.3. Análisis de SEP en estado transitorio

Los flujos de potencia son análisis que se realizan en estado estable, es decir no toman en cuenta las dinámicas producidas en base a la conexión y desconexión de cargas, arranque de motores y generadores. En el análisis de sistemas eléctricos de potencia, el estudio de fenómenos en estado transitorio está englobado en el término *estabilidad*, que toma en cuenta estos procesos dinámicos.

La estabilidad transitoria estudia la ocurrencia repentina de cambios repentinos en la generación, carga, o características del sistema eléctrico de potencia a causa de contingencias tales como cortocircuitos, pérdida de conexión, arranque de motores de potencia relativamente considerable respecto a la capacidad de generación, operaciones de conexión y desconexión, cargas mecánicas, pérdidas abruptas de potencia en generadores,

entre otros [33]. Tiene como herramienta fundamental a la transformada de Laplace para la simulación, y a la teoría de control en los tópicos referentes a la regulación de cargas. En la Tabla 1.4 se pueden consultar algunas duraciones habituales de los fenómenos mencionados.

Tabla 1.4: Duración de algunos fenómenos eléctricos analizados en estado transitorio [34].

| Fenómeno | Duración |
|--|--------------------------|
| Sobrevoltajes por rayos | Microsegundos a segundos |
| Transitorios de voltaje por conexión-desconexión | Microsegundos a segundos |
| Resonancia subsíncrona | Algunos segundos |
| Transitorios de máquinas eléctricas | Algunos segundos |
| Transitorios mecánicos | Algunos segundos |
| Controles de carga | Segundos a minutos |
| Dinámica de logística energética | Algunos minutos |
| Dinámica de recursos energéticos | Días a semanas |

Existe una teoría muy bien desarrollada para el estudio de generadores sincrónicos conectados a la red [35]. Sin embargo, el desarrollo de modelos detallados para estabilidad transitoria considerando procesos de generación que no involucran conversión electromecánica no tienen más de 20 años de desarrollo [36]. De hecho, la estabilidad transitoria se consideraba esencial para los operadores de red, pero no consideraba seriamente a la generación distribuida ni a la cogeneración de energía [33].

Considerando que, la gran mayoría de generadores eléctricos realizan conversión electromecánica de energía, mucha de la teoría de estabilidad transitoria está orientada a estudiar las pérdidas de sincronismo de las máquinas rotatorias.

El estudio de métodos para estudiar los fenómenos transitorios en los sistemas eléctricos de potencia mediante simulaciones lleva casi un siglo de existencia [37]. Sin embargo, la aparición de sistemas de resolución computacional basados en electrónica analógica o digital fueron un gran salto para investigar sistemas mucho más complejos. Los fenómenos transitorios que ocurren en los sistemas eléctricos de potencia pueden esquematizarse según lo indicado en la Fig. 1.26.

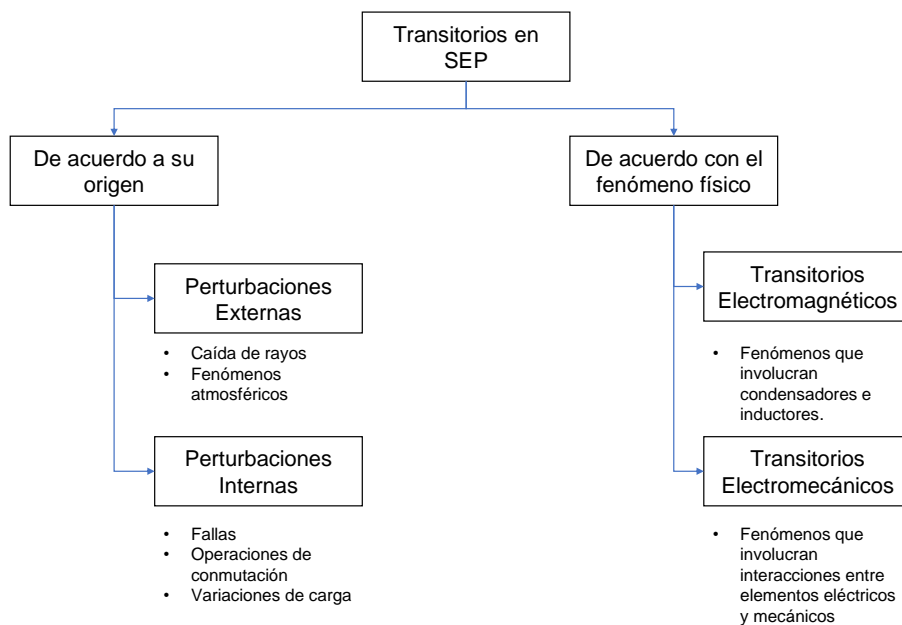


Figura 1.26: Clasificación de los fenómenos transitorios que pueden ocurrir en la operación de SEP [38].

Los elementos de un SEP pueden agruparse en tres grandes grupos, a saber:

1. Fuentes: Se utilizan para modelar generadores de potencia o perturbaciones externas (como el caso de rayos). Las fuentes pueden ser de voltaje (representadas con su equivalente Thevenin) o de corriente (representadas con su equivalente Norton).
2. Elementos pasivos: Dependiendo del fenómeno, el comportamiento de algunos elementos puede tener un comportamiento lineal o no lineal. Esto es especialmente notable en el transformador, puesto que puede usarse un modelo lineal o saturado de acuerdo con la aplicación.
3. Interruptores: Los interruptores pueden cambiar drásticamente la topología de un circuito conectando o desconectando componentes. Por este motivo, suelen ser utilizados para modelar fallas o cortocircuitos. En el caso ideal, la impedancia de un interruptor es infinita cuando está abierta, tomando un valor cero en caso contrario. Dependiendo de la aplicación, pueden usarse diferentes modelos para los interruptores.

Los fenómenos transitorios en los SEP pueden estudiarse mediante métodos de resolución en el dominio del tiempo y métodos en el dominio de la frecuencia. Los métodos en

el dominio de la frecuencia usan extensivamente la transformada de Laplace para el desarrollo de soluciones analíticas en base a las funciones matemáticas más comunes. Sin embargo, en la práctica los fenómenos de conmutación y control son no lineales, motivo por el cual los métodos de resolución basados en métodos numéricos son los más exitosos. El método más sencillo para la resolución de SEP en estado transitorio viene dado por la regla trapezoidal. Esta regla se basa en que el comportamiento de un sistema dinámico viene dado por la ecuación diferencial:

$$\frac{d}{dt}y(t) = x(t)$$

$$y(t) = y(0) + \int_0^t x(\tau) \cdot d\tau$$

No obstante, la integral definida se utiliza para funciones continuas. Para funciones discretas (utilizadas en métodos numéricos a causa de los efectos de cuantización en la computación), se puede utilizar la definición geométrica de la integral como área bajo la curva para partir la función suave en una suma de áreas de trapezios, como se aprecia en la figura 1.27.

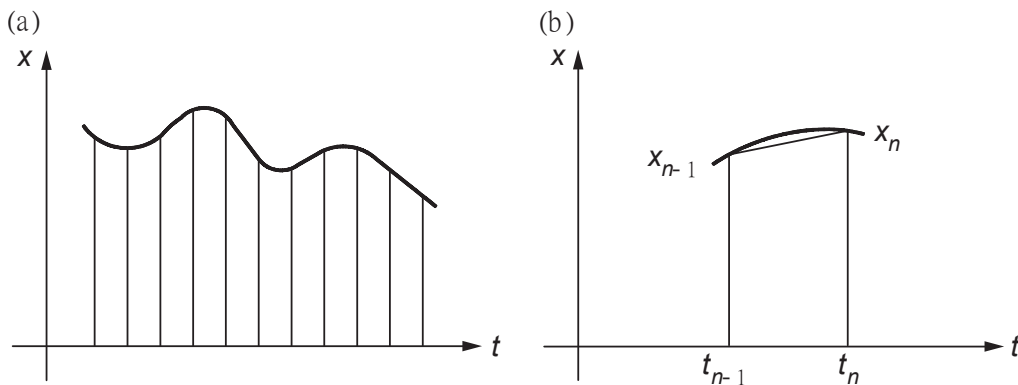


Figura 1.27: Aplicación de la regla trapezoidal. (a) Función primitiva y (b) integración trapezoidal [38]

Así, la expresión matemática queda:

$$y(t) = y(0) + \int_0^t x(\tau) \cdot d\tau$$

$$y_n = y_{n-1} + \frac{x_{n-1} + x_n}{2} (t_n - t_{n-1})$$

Si el cálculo se realiza utilizando una constante de tiempo Δt , este valor puede considerarse como un tiempo de paso de integración, cuyo proceso puede modificar la ecuación anterior para ser entendida de la siguiente forma:

$$y(t) = y(0) + \int_0^t x(\tau) \cdot d\tau$$

$$y_n = y_{n-1} + \frac{x_{n-1} + x_n}{2} (t_n - t_{n-1})$$

$$y(t) = y(t - \Delta t) + \frac{\Delta t}{2} [x(t) + x(t - \Delta t)]$$

Con esta definición, los elementos pasivos más sencillos pueden calcularse de acuerdo a lo presentado en la Tabla 1.5, considerando una notación en la que el elemento $x_{io} = x_i - x_o$, siendo el subíndice i la entrada al mismo y o la salida del mismo:

Tabla 1.5: Expresiones discretizadas para los elementos pasivos fundamentales [38].

| Elemento | Ecuación original | Ecuación discreta |
|-------------|---------------------------------------|---|
| Resistor | $v_{io} = i_{io} \cdot R$ | $v_{io} = R \cdot i_{io}$ |
| Inductor | $v_{io} = L \cdot \frac{di_{io}}{dt}$ | $i_{io} = \frac{\Delta t}{2L} v_{io}(t) + \left[\frac{\Delta t}{2L} v_{io}(t - \Delta t) + i_{io}(t - \Delta t) \right]$ |
| Condensador | $i_{io} = C \cdot \frac{dv_{io}}{dt}$ | $i_{io} = \frac{2C}{\Delta t} v_{io}(t) - \left[\frac{2C}{\Delta t} v_{io}(t - \Delta t) + i_{io}(t - \Delta t) \right]$ |

Se debe conocer que existen otros métodos numéricos de mayor y menor complejidad y confiabilidad. No obstante, la regla trapezoidal constituye la forma más sencilla para comprender la importancia de la resolución de sistemas eléctricos de potencia en el dominio temporal. A continuación, se mencionarán los fenómenos de estabilidad más estudiados en este campo, sin entrar en profundidad en sus definiciones y particularidades.

Estabilidad de voltaje Está relacionada con la capacidad de un SEP para mantener un voltaje aceptable en todas las barras del sistema bajo condiciones normales, y después de la aparición de una perturbación. Un SEP entra en una condición de inestabilidad de voltaje cuando ocurre una perturbación como un cambio abrupto en la demanda, o condiciones que causan un declive progresivo e incontrolable de voltaje. La causa más importante para que

ocurran condiciones de inestabilidad de voltaje es la incapacidad del SEP para satisfacer la demanda para la potencia reactiva [39].

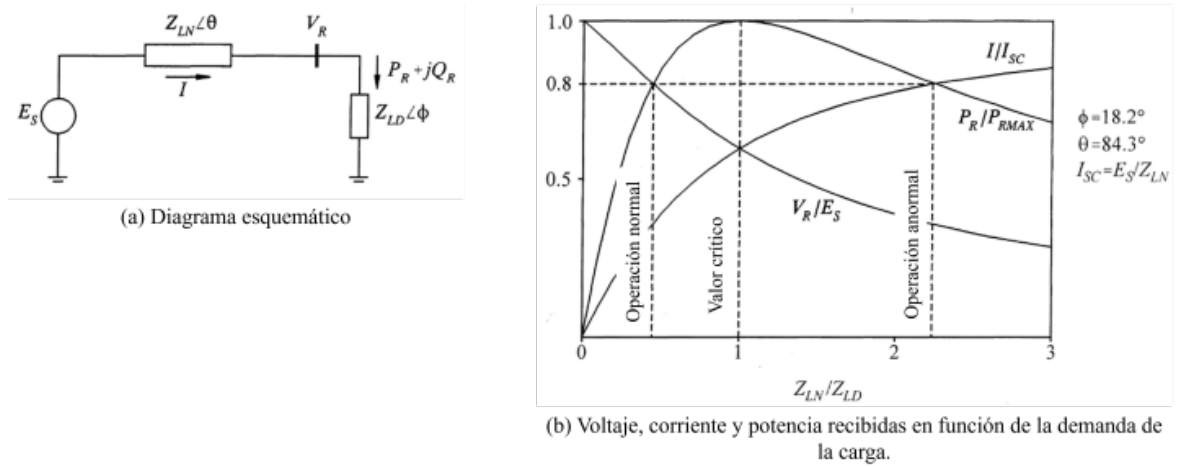


Figura 1.28: Características de un sistema radial sencillo [39].

Para el sistema mostrado en la Fig. 1.28, el voltaje en los terminales de la carga, así como la potencia en la carga vienen dadas por las siguientes ecuaciones:

$$I = \frac{1}{\sqrt{F}} \frac{E_S}{Z_{LN}}$$

$$V_R = \frac{1}{\sqrt{F}} \frac{Z_{LD}}{Z_{LN}} E_S$$

$$P_R = \frac{Z_{LD}}{F} \left(\frac{E_S}{Z_{LN}} \right)^2 \cos \phi$$

Dónde: $F = 1 + \left(\frac{Z_{LD}}{Z_{LN}} \right)^2 + 2 \left(\frac{Z_{LD}}{Z_{LN}} \right) \cos(\theta - \phi)$

A medida que la demanda incrementa (mediante la disminución del valor de la impedancia de carga), la potencia activa se incrementa rápidamente al principio, y luego de forma más lenta antes de alcanzar un máximo, para finalmente decrecer. Entonces hay un valor máximo y crítico de potencia activa que puede ser transmitida a través de una impedancia de una fuente de voltaje constante. La potencia transmitida es máxima cuando el voltaje de caída en la línea es igual en magnitud a V_R , dicho sea $Z_{LD}/Z_{LN} = 1$. Para una demanda mayor a la potencia máxima, el control de potencia cuando la carga varía puede ser inestable. Si la carga tiene una admitancia constante, el sistema se estabiliza en un voltaje menor al normal. Pero cuando la carga es entregada a través de un transformador con taps, el control de taps intentará elevar el voltaje de la carga al detectar que el valor es inferior al esperado, con el efecto de reducir la impedancia Z_{LD} . Esto disminuye el voltaje aún más,

produciéndose el fenómeno de inestabilidad de voltaje [39].

Estabilidad de ángulo Es la capacidad de las máquinas sincrónicas del SEP para mantenerse en sincronismo después de perturbaciones que pueden estar directamente asociadas con mantener o restaurar el equilibrio entre torque mecánico y electromagnético en el sistema. Los problemas de estabilidad de ángulo se vuelven evidentes a través de la presencia de oscilaciones de potencia o frecuencia en algunos generadores por la pérdida de sincronismo. Los fenómenos de corta duración que afectan a la estabilidad de ángulo generalmente están relacionadas con falta de torque de amortiguamiento. Los fenómenos de larga duración están relacionados con fenómenos no lineales [38]. En la estabilidad de ángulo, el criterio de áreas iguales es muy importante para el estudio de puntos críticos.

Estabilidad de frecuencia Está asociada con la recuperación de la frecuencia del SEP después de pérdidas de equilibrio abruptas entre la potencia activa entre la generación y la carga a causa de diferentes perturbaciones. Así, la estabilidad de frecuencia se concentra en estudiar variaciones después de cambios abruptos en el balance generación-carga [38].

Estabilidad en energías renovables La teoría de estabilidad ha sido dominada por el análisis de generadores basados en máquinas rotativas. Sin embargo, existen casos como la energía fotovoltaica en la que no se utilizan. En general, la generación mediante energías renovables tiene un efecto negativo en la estabilidad de la red a causa de su limitado control de voltaje y potencia así como la incapacidad de producir inercia en el sistema. Sin embargo, su importancia y participación en la generación aumentan, obligando a los académicos a estudiar mecanismos de acoplamiento. El uso de convertidores electrónicos de potencia de alta frecuencia en este tipo de generación produce nuevos fenómenos, como el caso de la *estabilidad armónica*, que se refiere a los problemas de estabilidad relacionados con convertidores resonantes de generadores vecinos y filtros armónicos, así como la *estabilidad de fase*, que estudia la pérdida de sincronización con los lazos de seguimiento de fase [38].

1.4.4.4. Procesos para simulación de SEP

Indistintamente de los métodos numéricos usados en la resolución de simulaciones de fenómenos en estado transitorio o en estado estable, existen algunos procesos que se siguen habitualmente con la finalidad de culminarlas con éxito, como se puede apreciar en la Fig. 1.29. En este proyecto de titulación se realiza una simulación en estado transitorio. Por lo tanto, el énfasis se enfocará hacia esta metodología.

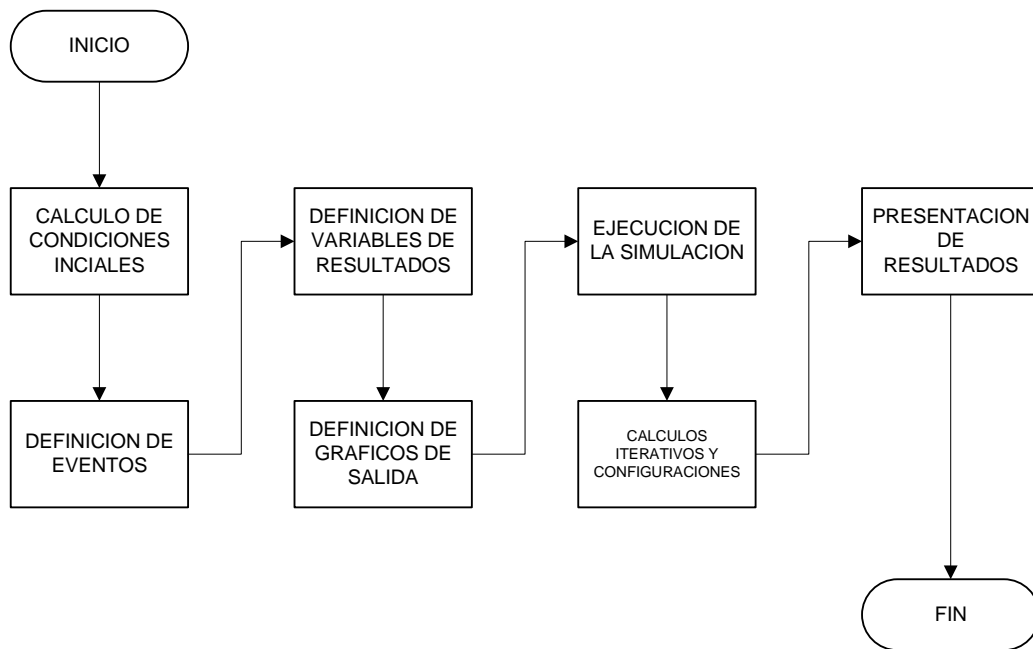


Figura 1.29: Proceso típico para la resolución de simulaciones en estado transitorio [40].

Para el cálculo de condiciones iniciales, se suelen ejecutar flujos de potencia para conocer el valor de las variables en el instante cero de la simulación. Posteriormente, se definen los eventos con antelación o se asume la ocurrencia de eventos mediante ocurrencia externa.

Los eventos de naturaleza transitoria suelen modelarse con los equivalentes lineales y no lineales mediante la Transformada de Laplace [41]. La resolución de estos modelos se realiza mediante métodos numéricos, usando los métodos clásicos (Euler, Runge-Kutta, Gauss-Seidel, etcétera).

Es importante mencionar que, dependiendo de los programas utilizados para la simulación, los métodos de resolución de flujos de potencia así como de sistemas dinámicos, existirán más o menos opciones disponibles. Por ejemplo, en MATLAB, se pueden escoger métodos de resolución de ecuaciones diferenciales con paso fijo de tiempo o paso variable de tiempo, mientras que los simuladores de sistemas eléctricos de potencia generalmente disponen de solucionadores en tiempo fijo. Sin embargo, se profundizará en las características de los simuladores en las siguientes páginas.

1.4.5. SOFTWARE DE SIMULACIÓN

En este trabajo los simuladores usados en la ingeniería eléctrica serán agrupados en tres grandes familias, los simuladores de circuitos electrónicos, los simuladores de sistemas de potencia, y los simuladores híbridos. Esta es una clasificación adecuada para el contexto en el que se desarrollarán los conceptos que se presentarán posteriormente.

1.4.5.1. Simuladores de sistemas eléctricos de potencia

Los simuladores de sistemas eléctricos de potencia son programas que permiten modelar y analizar el comportamiento de redes eléctricas de generación, transmisión y distribución en modo asíncrono o utilizando datos en tiempo real. Se usan en una gran variedad de actividades de planificación y operación de sistemas eléctricos de potencia [42].

Aunque los rudimentos del análisis de los sistemas eléctricos de potencia tienen su origen en las leyes de Ohm y Kirchoff, la operación más importante y fundamental para un simulador de SEP es el cálculo de flujos de potencia. En estos programas también se pueden realizar simulaciones de eventos de interés como análisis de cortocircuito, flujo óptimo de potencia, generación a largo plazo, expansiones de redes eléctricas, estudio de armónicos y calidad de energía eléctrica, inclusive pueden realizar cálculos de estabilidad transitoria.

Los elementos más importantes en estos simuladores son los generadores sincrónicos, las líneas de transmisión, las cargas y las barras de voltaje. De ahí que, la concepción para la que fueron diseñados estos simuladores es diferente y más especializado a los simuladores de circuitos electrónicos comúnmente utilizados por todos los ingenieros relacionados con la electricidad y la electrónica [43].

Algunos ejemplos de simuladores de Sistemas Eléctricos de Potencia son *PowerWorld*, *ETAP*, *Siemens PSSS*, *ABB PROMOD*, *PowerWorld* (que puede apreciarse en la Fig. 1.30) o *PowerFactory*, que es el software de simulación de SEP utilizado en este proyecto. Algunas iniciativas de software libre como *PandaPower* han sido desarrolladas para su uso en *Python*.

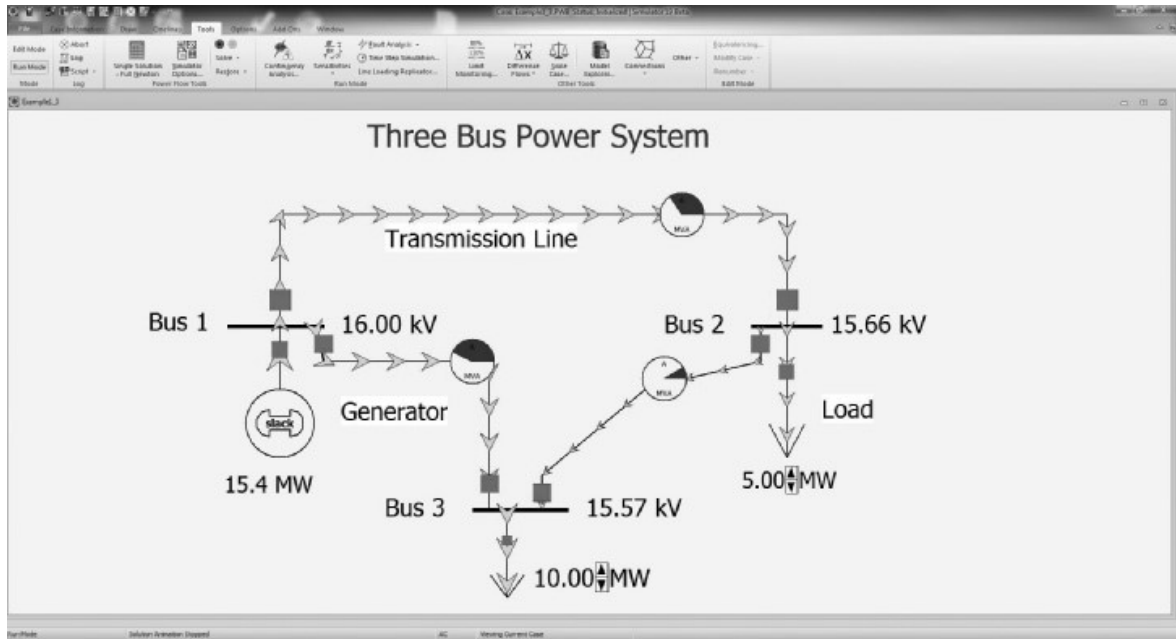


Figura 1.30: Primer plano del simulador de SEP *PowerWorld*, se puede apreciar un diagrama unifilar, y la ejecución de un pequeño flujo de potencia [31].

1.4.5.2. Simuladores de circuitos electrónicos de bajo voltaje

Los simuladores de circuitos electrónicos son programas -mayormente diseñados para su uso en computadores personales- que tienen la finalidad de replicar el comportamiento de un circuito electrónico real. El análisis de circuitos electrónicos puede realizarse de forma analógica o digital. En el caso de los análisis analógicos, se da mayor prioridad a los análisis de circuitos con corriente continua, con corriente alterna, análisis en estado transitorio y análisis en frecuencia, mientras que en el caso de los análisis digitales se da prioridad a los niveles lógicos (alto, bajo o búfer) antes que a sus valores físicos.

Los simuladores de circuitos electrónicos analógicos suelen proveer un entorno de usuario en el que se pueden utilizar elementos activos (baterías, fuentes de corriente alterna, fuentes dependientes de voltaje y corriente) y pasivos (resistencias, inductancias, condensadores, semiconductores, transformadores) relativamente comunes. Los órdenes de magnitud de las variables físicas medidas (voltaje, corriente, potencia) son relativamente bajas y orientadas a aplicaciones de bajo voltaje.

Actualmente algunos simuladores de circuitos electrónicos también proveen herramientas de desarrollo para elaboración de placas de circuito impreso, corrección de errores, optimización de rutas, inclusive simulaciones de interferencias electromagnéticas para el diseño de prototipos. Ejemplos de simuladores electrónicos son *CircuitMaker*, *ORCAD*, *LTS-*

pice, Proteus (apreciable en la Fig. 1.31), MultiSim, OpenCircuit.

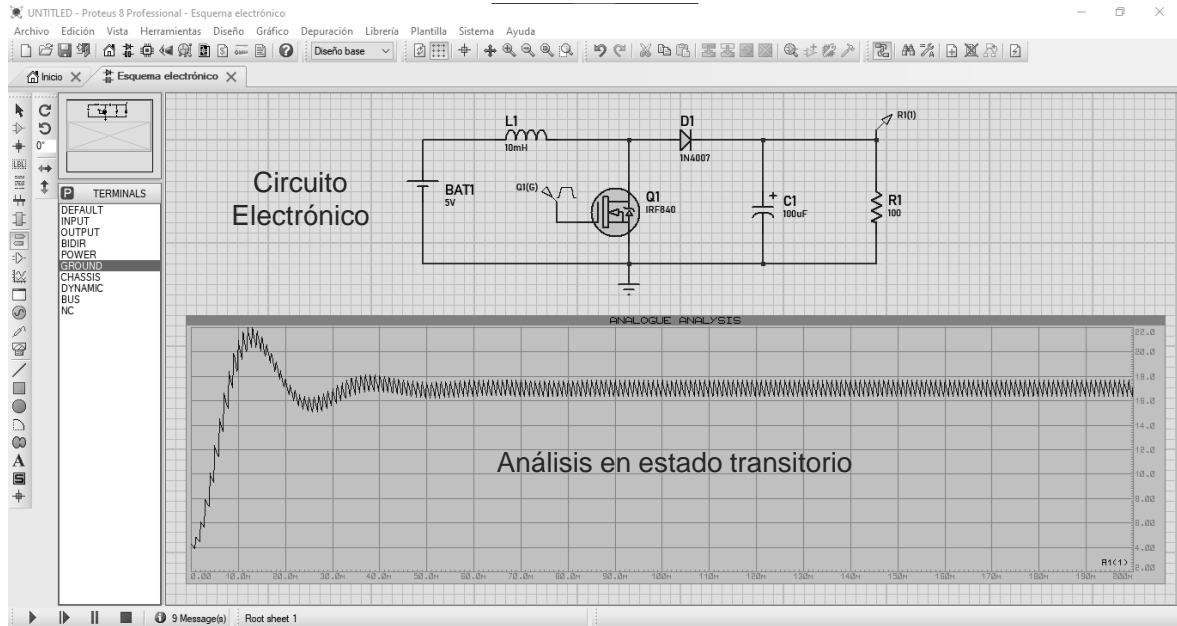


Figura 1.31: Forma de onda del voltaje de salida en estado transitorio de un convertidor elevador simulado en *Proteus 8*.

Posteriormente surgieron simuladores de circuitos electrónicos especializados para electrónica de potencia. En los circuitos electrónicos de potencia es requerida una concepción diferente a causa de los constantes cambios de estado en los semiconductores, siendo en estos simuladores el método de función a trozos el favorito para tal efecto [11]. Ejemplos de estos programas son *PLECS* y *PSIM*.

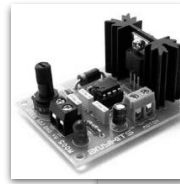
Con un fin de ejemplificación, también se mencionarán a los simuladores de electrónica digital. Estos simuladores se aplican a los sistemas digitales con la finalidad de ayudar al estudio de sistemas que usan compuertas lógicas, máquinas de estado, lógica secuencial y combinacional. Actualmente su estudio se realiza principalmente mediante dos lenguajes de programación: *VHDL* y *Verilog*.

En la Fig. 1.32 se puede apreciar una comparación general entre estas dos familias de simuladores, antes de presentar la última familia, a la que pertenece el software usado en este trabajo para el modelado del sistema electrónico de potencia.



Simuladores de SEP

- Basado en diagramas unifilares.
- Preferentemente trabajan con magnitudes de alto voltaje.
- Preferencia a los resultados en el sistema por unidad.
- Pueden incluir compatibilidad con GIS
- Predominancia de generadores sincrónicos como fuentes de potencia activa o reactiva.
- Trabajan generalmente con horizontes temporales largos (minutos, horas, días, meses)



Simuladores electrónicos

- Basado en diagramas de circuitos clásicos.
- Son usados para aplicaciones de bajo voltaje y/o potencia.
- Preferencia a los resultados en el Sistema Internacional de Unidades.
- Pueden incluir diseñadores de PCB.
- Pueden usar fuentes AC, Baterías, Paneles solares sin aparente predominancia.
- Trabajan generalmente con horizontes temporales pequeños (segundos, microsegundos, nanosegundos)

Figura 1.32: Comparación entre algunas características de los simuladores de sistemas eléctricos de potencia y los simuladores de circuitos electrónicos de bajo voltaje.

1.4.5.3. Simuladores híbridos

Los simuladores híbridos son simuladores que tienen la capacidad de realizar análisis de circuitos utilizando circuitos electrónicos e interconectándolos en sistemas eléctricos de potencia en un mismo programa.

Son programas con una gran capacidad de modelamiento de sistemas eléctricos y electrónicos de potencia, sin embargo, aunque pueden realizar análisis de circuitos y cálculo de flujos de potencia, su uso se limita a la interconexión a sistemas pequeños [10]. Suelen funcionar mejor para análisis transitorios ya que tienen mayores facilidades para implementar algoritmos de control diferentes a los controladores clásicos [44]. De acuerdo con estas características, *Simulink* puede considerarse un simulador híbrido. En la Fig. 1.33 se puede apreciar un ejemplo en el que se pueden apreciar algunas de estas características en un modelamiento particular.

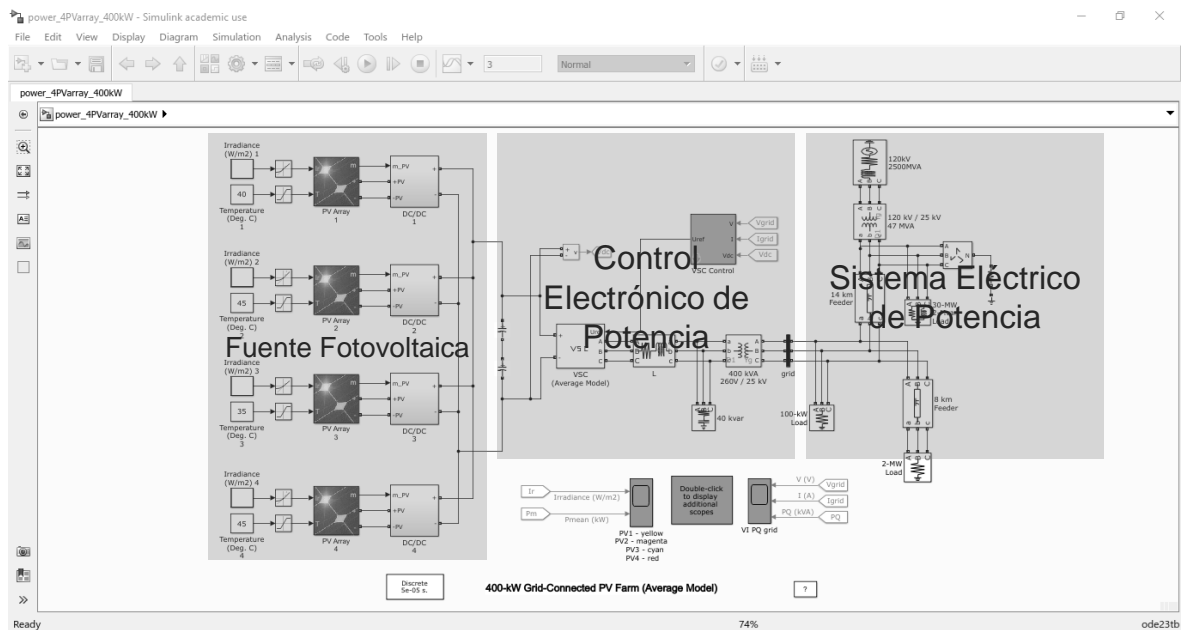


Figura 1.33: Esquema de un arreglo de paneles fotovoltaicos conectado a un sistema eléctrico de potencia en *Simulink*.

1.4.5.4. Matlab-Simulink

Simulink es un entorno de programación gráfico basado en MATLAB diseñado para modelar, simular y analizar sistemas dinámicos. Dispone de librerías propias y de terceros para la creación más rápida de situaciones comunes y predeterminadas, siendo usada de forma extensiva en sistemas de control y diseño basado en modelos [45].

Simscape Electrical Anteriormente conocida como *SimPowerSystems*, es un paquete provisto en *Simulink* para el análisis de sistemas eléctricos de potencia, sistemas electrónicos y mecatrónicos [46]. Dada su difusión en el ámbito investigativo, dispone de herramientas capaces de modelar aplicaciones tales como energías renovables, redes inteligentes, y actuadores electromecánicos.

Simscape Electrical ha sido utilizado dentro de Simulink con mucho éxito para modelar plantas fotovoltaicas [47], inversores DC-AC conectados a la red [48], algoritmos de seguimiento de máximo punto de potencia [49]. Se puede afirmar que junto con *PLECS*, son los programas más utilizados para el modelado y simulación de sistemas basados en control electrónico de potencia [50]. Un ejemplo de su utilización puede apreciarse en la Fig. 1.34.

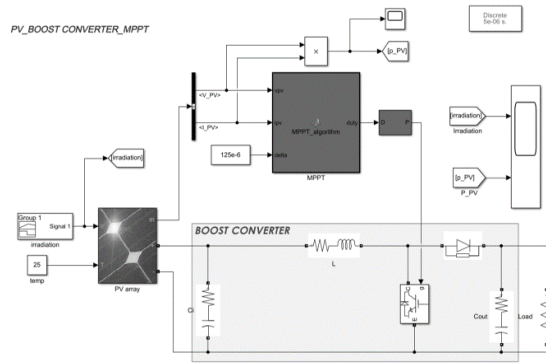


Figura 1.34: Esquema de una celda solar y un convertidor DC/DC elevador en *Simulink* [51]

Aunque *Simulink* puede ejecutar la simulación de sistemas eléctricos de potencia, su practicidad es muy limitada en relación a los programas de simulación especializados, motivo por el cual se utiliza para el estudio de SEP de tamaño reducido.

1.4.5.5. DigSILENT PowerFactory

DigSILENT PowerFactory es un software de ingeniería instalado en un ordenador usado para el análisis de transmisión, distribución y sistemas eléctricos de potencia industriales. DigSILENT es el acrónimo para *Digital Simulation of Electrical Networks* o Simulación Digital de Redes Eléctricas. La versión 7 de este programa fue la primera del sector que ofreció la capacidad de realizar análisis de sistemas de potencia mediante una interfaz gráfica con diagramas unifilares. Por este motivo, tuvo una rápida y amplia aceptación en los últimos años, haciéndola herramienta de referencia para los estudios eléctricos [52]. Una vista de su interfaz gráfica de usuario se puede apreciar en la Fig. 1.35.

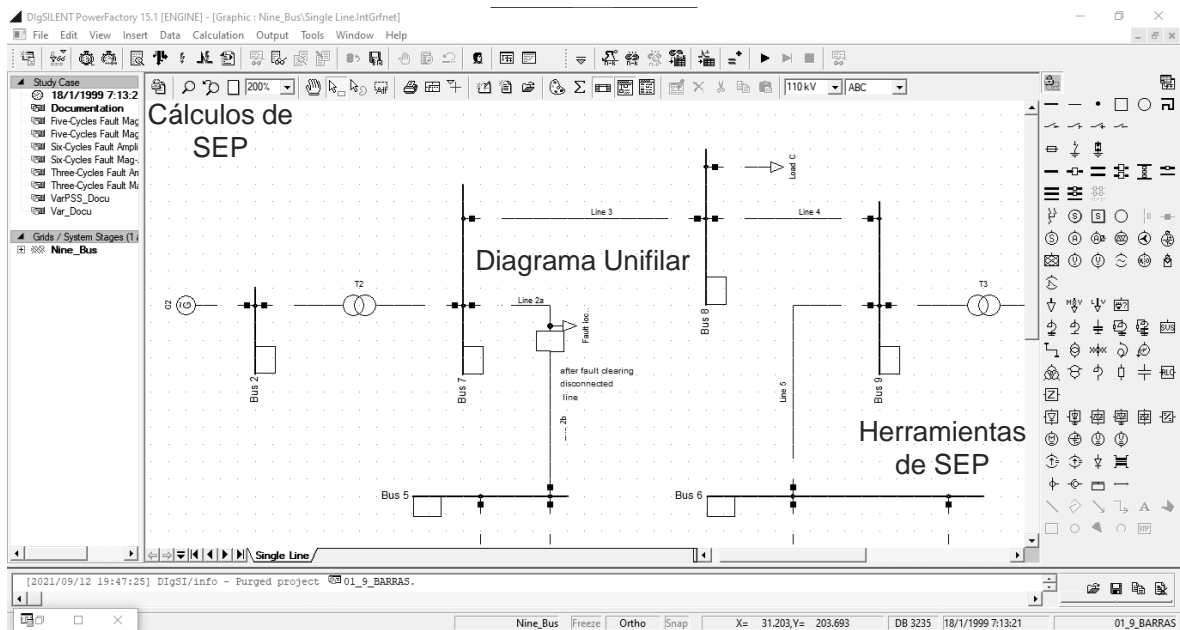


Figura 1.35: Primer plano de un sistema eléctrico de potencia de ejemplo en el software *PowerFactory*, un simulador para sistemas eléctricos de potencia.

DigSILENT PowerFactory ha sido desarrollado por ingenieros con vasta experiencia en programación y modelamiento de SEP, motivo por el cual la exactitud y la aceptación de los resultados que produce ha sido aceptado y validado en experiencias alrededor del globo. Aunque *DigSILENT PowerFactory* es ejecutado como un programa monolítico integrado, es menester conocer las aplicaciones integradas en el programa, las cuales pueden manejarse por separado [52]:

1. Núcleo de *PowerFactory*: Definición, modificación y organización de casos de estudio, funciones numéricas base, documentación y funciones de salida para resultados.
2. Entorno gráfico integrado para los diagramas unifilares así como su manejo.
3. Base de datos para elementos individuales y casos de estudio.
4. Configuración de redes eléctricas con acceso interactivo o en línea con sistemas SCADA.
5. Interfaz genérica para sistemas de mapeo basados en computador.

El uso de una base de datos para el manejo de los resultados, significa que *PowerFactory* puede ejecutar fácilmente, y en un entorno integrado todas las funciones de cálculo

(flujos de potencia, análisis de cortocircuito, análisis RMS y EMT, entre otros) de forma secuencial, sin necesidad de reiniciar el entorno del programa.

1.4.6. CO-SIMULACIÓN

La co-simulación es una técnica en la que, la simulación global de un sistema con componentes interconectados en un sistema acoplado puede realizarse exitosamente mediante la simulación conjunta de cada una de sus partes integrantes. Para este tema, se trabajará principalmente mencionando las definiciones encontradas en el trabajo de Gomes et al. las cuales resultan muy pertinentes en para el alcance del trabajo [53].

1.4.6.1. Conceptos fundamentales

Un *sistema dinámico* es un modelo de un sistema real (un modelo físico) caracterizado por la situación de sus variables (conocidas en teoría de control como *estados*) y la noción de la existencia de reglas de evolución de estas variables. Ejemplos de un sistema dinámico son: un semáforo (el estado de la luz depende del estado anterior, y del estado actual depende el estado posterior), o un sistema masa-resorte, en el que los estados pueden parametrizarse gracias a ecuaciones diferenciales. En la Ec. 1.34 puede apreciarse la ecuación matemática más sencilla que puede definir conceptualmente un sistema dinámico, donde \mathbf{x} es un vector que contiene a las variables de estado $\mathbf{x} = (x_1, x_2, \dots, x_n)$ con capacidad de ser variable en el tiempo, y se conoce el estado \mathbf{x}_0 en el tiempo t_0 para poder estudiar su pasado o su evolución.

$$\dot{\mathbf{x}} = f(\mathbf{x}, t) \quad \mathbf{x}(t_0) = \mathbf{x}_0 \quad (1.34)$$

Un simulador que analice un sistema dinámico en el *intervalo temporal de simulación* $[0, t]$ realiza este procedimiento en τ *unidades de tiempo de reloj*. Conociendo esto, se puede definir que un simulador puede representar el comportamiento dinámico de un sistema de acuerdo a la ecuación $t = \alpha\tau$ donde α es un parámetro computacional del simulador. La simulación tradicional o co-simulación encuentra uno de sus problemas clásicos en asegurar que el valor de $\alpha = 1$, es decir, que el simulador pueda describir la dinámica del sistema de estudio en el mismo tiempo que transcurriría realmente. En la simulación de sistemas dinámicos, muchas veces se requiere la capacidad de interacción entre el operador y el simulador, por lo que los últimos suelen ser capaces de aumentar o disminuir el valor de α , o actuar sobre el curso de simulación pausándola a juicio del operador o mediante

programación.

El *marco experimental* define un conjunto de condiciones en las que el comportamiento del sistema simulado se corresponde con el comportamiento del sistema real. La *validez* es la diferencia entre el comportamiento obtenido por la simulación y el sistema real, medida bajo las condiciones indicadas por el marco experimental. Existen dos formas de obtener el comportamiento de un sistema dinámico: la forma analítica y la forma operacional. La primera es la clásica obtención del comportamiento como la conjunción de funciones matemáticas trascendentales y algebraicas, y la segunda es la obtención del comportamiento del sistema mediante métodos aproximados. El *error* es la diferencia entre el comportamiento obtenido por la simulación y el comportamiento del sistema real de acuerdo a los resultados analíticos. Los computadores digitales trabajan casi totalmente con el método operacional.

1.4.6.2. Parámetros de co-simulación para sistemas de potencia

Como es apreciable, el hablar de la co-simulación puede ser realizado desde distintos puntos de vista. En los párrafos anteriores se ha intentado explicar algunos conceptos fundamentales de la co-simulación desde un enfoque próximo a la matemática y las ciencias de la computación. Sin embargo, este punto de vista tranquilamente puede desembocar en una revisión bibliográfica muy profundizada en estas acepciones, siendo que, esta revisión bibliográfica puede realizarse desde un punto de vista más pragmático: desde el estudio de los sistemas eléctricos de potencia.

En [54] se realiza un trabajo muy detallado en el que se proponen algunos parámetros para considerar en una co-simulación para sistemas eléctricos de potencia. En la figura 1.36 se puede apreciar un resumen de la metodología propuesta.

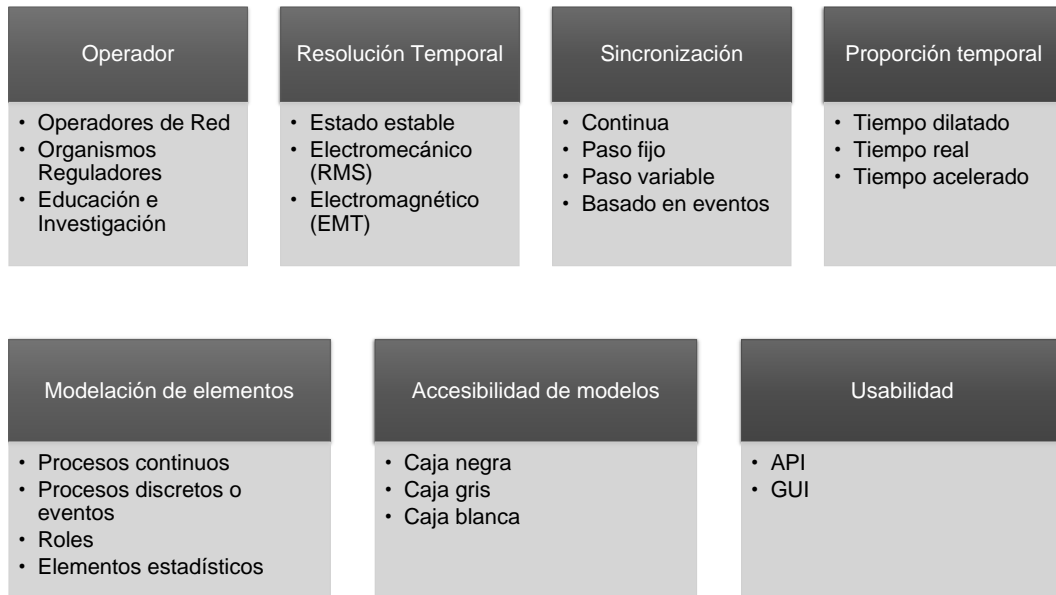


Figura 1.36: Esquema de parámetros críticos en el diseño de un proceso de co-simulación.

Detallando los parámetros propuestos, se puede empezar a hablar del criterio del operador, en el que se define el público objetivo que se encargará del manejo de la simulación: no es correcto diseñar un sistema para un operador experto de la misma manera que para un operador novato. Respecto a la resolución temporal, las simulaciones pueden estudiar fenómenos en el orden de las horas, minutos, hasta los nanosegundos. Por lo tanto, es importante considerar la resolución temporal necesaria para determinar los requerimientos de velocidad y memoria. La proporción temporal relaciona el intervalo de tiempo entre el inicio y finalización de la simulación en tiempo real y el intervalo de tiempo del fenómeno de estudio.

Los eventos de simulación pueden suceder a lo largo de todo un intervalo temporal, entre algunos periodos, de acuerdo a condiciones preestablecidas o como consecuencia de otros eventos probabilísticos. Los elementos del modelo pueden ser modelados como una caja blanca -donde toda la matemática y los modelos del sistema dinámico están disponibles-, una caja gris -algunos modelos y parámetros del sistema dinámico pueden estar disponibles-, o una caja negra -ningún parámetro está disponible, solamente se disponen los datos de los cambios de comportamiento respecto a los cambios en las condiciones de entrada del sistema dinámico-.

La sincronización es un tema aparte que es bastante estudiado en la teoría de simulación y modelamiento de sistemas físicos. En este trabajo no se profundizarán sobre los conceptos teóricos pero sí se explicará que existen investigaciones para sincronizar mo-

delos continuos, otras son basadas en muestreos cada cierto intervalo de tiempo, otras se basan en muestreos cada intervalos de tiempo variables, y las más sencillas son máquinas de estados en las que los eventos están preestablecidos de acuerdo a condiciones estacionarias.

Para finalizar, hay que considerar si la aplicación será gestionada mediante Interfaces de Protocolo de Aplicación (API) o deberá tener una interfaz gráfica de usuario. En el primer caso, la simulación está centrada en la obtención de resultados considerando eventos con condiciones iniciales poco modificables con interfaces de usuario poco desarrolladas -línea de comandos- o inexistentes, mientras que en el segundo caso, prima la interactividad entre el operador y la máquina por sobre la exactitud de los resultados.

1.4.6.3. Avances de la co-simulación aplicados en SEP

La co-simulación ha ganado presencia en el estudio de los sistemas eléctricos de potencia ya que, se constituye en una alternativa viable para el estudio de fenómenos de interés dentro de su campo de estudio. En el estudio de fasores, existen trabajos que utilizan este enfoque para la extracción de fasores en sistemas eléctricos de potencia [55]. Un concepto nuevo que está siendo utilizado para el estudio, es el de los *gemelos digitales*, que simplemente es una representación virtual que sirve como una contraparte digital en tiempo real de un objeto o fenómeno físico. Se han generado en la industria aeroespacial para enfrentar al reto de monitorear, manipular e incluso reparar sistemas remotos, pudiendo conectarse mediante realidad virtual a entornos de simulación generando espacios prometedores para la investigación de nuevas técnicas de control y de tolerancia a fallos. En este contexto, Palensky et al. proponen su uso en sistemas eléctricos de potencia en conjunto con co-simulaciones para explorar estas situaciones desde enfoques fuera de lo tradicional [56]. Para la generación distribuida, se han propuesto marcos de trabajo de co-simulación, con la finalidad de optimizar la ubicación de generadores dentro de redes de distribución basadas en técnicas de inteligencia computacional [57].

Es importante mencionar que estos avances en la co-simulación para los sistemas eléctricos de potencia han sido estudiados también desde un punto de vista más general, estudiando primeramente los fundamentos, la arquitectura de software necesaria y el acoplamiento de los mismos para su implementación exitosa. Palensky et al. han propuesto marcos de trabajo básicos para tal efecto [58]. También se han realizado pruebas en conjunto con la inteligencia artificial para la gestión de redes inteligentes [59]. Es importante mencionar que la complejidad de las redes eléctricas inteligentes es la que ha propulsado a

la co-simulación, ya que, con las herramientas tradicionales de simuladores especializados, era más que suficiente para el estudio de fenómenos estacionarios.

En el ámbito de la electrónica de potencia, la co-simulación ha sido útil para estudiar convertidores electrónicos de alta frecuencia mediante simulación híbrida mediante FPGA [60]. También se ha utilizado con éxito para el estudio de los flujos en máquinas eléctricas considerando campos magnéticos y térmicos [61] y para redes eléctricas no tradicionales donde la electrónica de potencia es muy preponderante [62]. Por último, se puede citar el trabajo realizado por Chiza et al. en los que se emulan cargas de tracción eléctrica mediante la co-simulación proponiendo metodologías que se constituyen en referencias pertinentes para el presente trabajo [63]. Un esquema de las utilidades del enfoque de co-simulación en los sistemas eléctricos de potencia se puede apreciar en la Fig. 1.37.

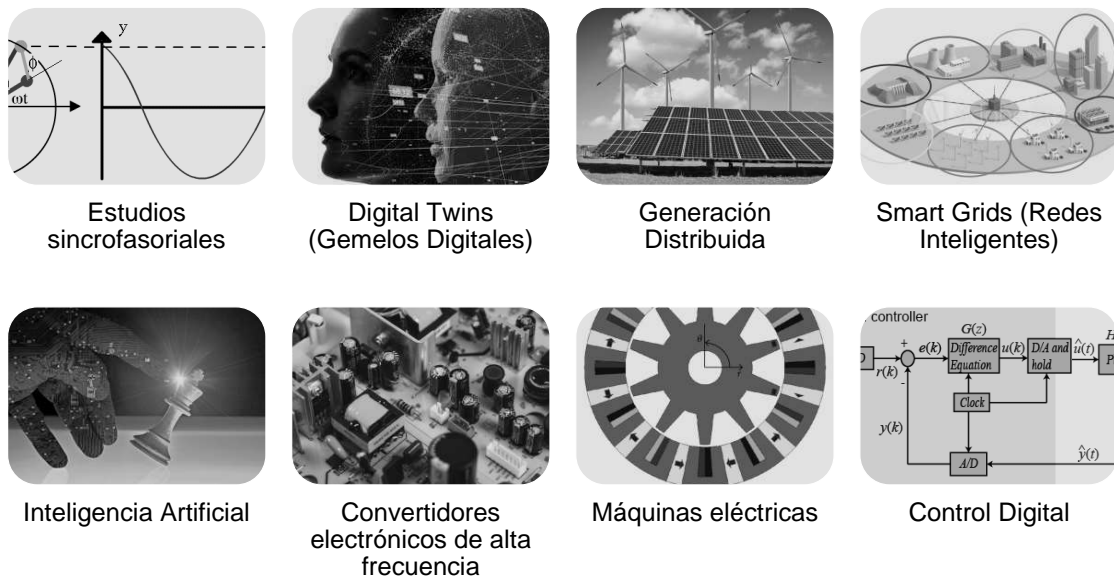


Figura 1.37: Algunas aplicaciones de la co-simulación en el estudio de los sistemas eléctricos y electrónicos de potencia.

1.4.7. OPC

OPC fueron inicialmente las siglas para *Object Linking and Embedding for Process Control*, es decir, vinculación e incrustación de objetos para el control de procesos. Fue desarrollada inicialmente por Microsoft a mediados de la década de 1990 [64]. Actualmente OPC son las siglas para *Open Process Control* ya que, gracias a su amplia utilización en la industria, se ha liberado, convirtiéndose en un estándar de facto para la intercomunicación de

datos en redes industriales y sistemas de control y adquisición de datos (SCADA). Actualmente está estandarizado por la *OPC Foundation*.

OPC es un estándar técnico que provee un conjunto de estándares para intercambio de información. OPC hace posible el intercambio de datos y la interoperabilidad entre sistemas y dispositivos, sistemas de automatización y control y software de control de manufactura [65].

1.4.7.1. Motivación para el desarrollo de OPC

Antes de la implementación de OPC, las aplicaciones industriales de escritorio conectadas a dispositivos de control requerían la instalación independiente de cada controlador dependiendo de los dispositivos requeridos. Esta situación provoca algunas desventajas como el requerir diferentes controladores dependiendo de la marca, conflictos entre controladores del mismo fabricante pero de diferente versión, la posibilidad de que, al cambiar el hardware, todo el sistema de control quede inutilizable, y sobre todo, la gran probabilidad de que al usar más de un dispositivo del mismo tipo, no exista comunicación ya que al usar el mismo controlador, pero tener diferentes funcionalidades, el sistema no distingue las diferencias entre uno y otro [65].

Con la implementación de OPC se solucionan estos problemas, puesto que con una arquitectura unificada se requiere solamente el controlador OPC. OPC es un protocolo basado en la arquitectura cliente-servidor, en la que los distintos dispositivos pueden compartir información sin depender del fabricante. Este estándar abierto ha permitido que los sistemas de control industrial no dependan de una marca, al permitir la incorporación de distintos dispositivos de diferentes fabricantes. Esta situación se puede esquematizar en la figura 1.38.

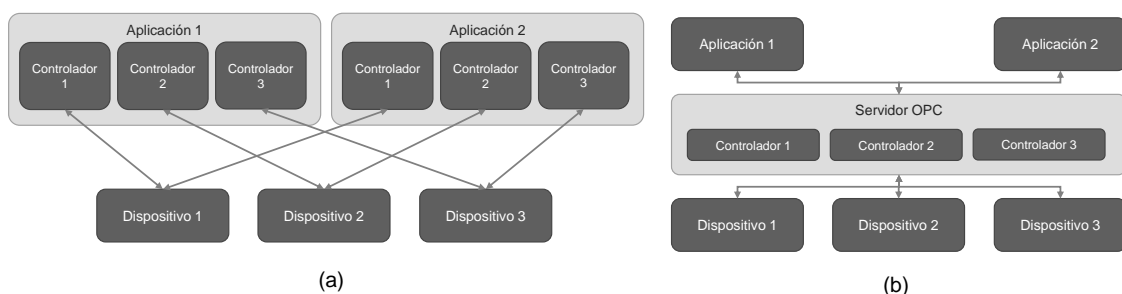


Figura 1.38: Comparación de la comunicación entre dispositivos y aplicaciones sin el uso de OPC (a) y con el uso de OPC (b).

Actualmente, OPC es el protocolo para intercambio de datos entre aplicaciones de Windows más aceptado a nivel industrial, y en su estandarización participan los fabricantes más importantes orientados a la automatización industrial [66].

1.4.7.2. Desarrollo de OPC

OPC ha sido desarrollado desde 1996. Frente a los desarrollos tecnológicos, y la irrupción de Internet en la vida cotidiana, inclusive en los sistemas de control industrial, ha presentado avances que permiten discernir dos etapas diferenciadas, las cuales se presentarán a continuación:

OPC Clásico El OPC clásico está basado en las tecnologías COM y DCOM de Microsoft. Tres estándares se definieron para el entorno clásico: OPC DA (*Data Access*, Acceso de Datos), OPC A&E (*Alarms and events*, Alarmas y Eventos) y OPC HDA (*Historical Data Access*, Acceso de Datos Históricos). Un resumen con las diferencias entre estos tres tipos de datos puede apreciarse en la Fig. 1.39. En este proyecto, se empleará OPC clásico para la implementación de la simulación propuesta, ya que se busca la máxima compatibilidad con los sistemas *legacy* o antiguos.

| OPC Acceso de Datos (OPC DA) | OPC Alarmas y Eventos (OPC A&E) | OPC Acceso de Datos Históricos (OPC HDA) |
|---|--|---|
| <ul style="list-style-type: none"> • La interfaz se concentra en la lectura, escritura y monitoreo de variables en tiempo real. • Los ítems de interés son agrupados mediante grupos OPC. • Proveen marcas de tiempo e indicadores con la calidad de la comunicación | <ul style="list-style-type: none"> • Habilita la recepción de notificaciones de eventos y alarmas de proceso. • Se basa en la generación de objetos de subscripción OPC. • A diferencia de OPC DA, no hay solicitudes explícitas de información específica. | <ul style="list-style-type: none"> • A diferencia de OPC DA, solamente provee acceso a datos ya almacenados en una BD. • Se basa en la creación de servidores OPC HDA y buscadores HDA. • Define métodos para insertar, reemplazar y eliminar datos en una BD. |

Figura 1.39: Comparación entre las tres especificaciones OPC clásicas más importantes [66]

Para la intercomunicación de datos, la especificación OPC clásica más utilizada es OPC Data Access. Esta especificación utiliza ítems y grupos para realizar la segmentación e

identificación de los datos que se requiere transmitir. En la Fig. 1.40 se puede apreciar un esquema del cómo se agrupa la información en el protocolo OPC. Esto será de vital importancia para entender la metodología de la simulación en la etapa del levantamiento del cliente y el servidor OPC.

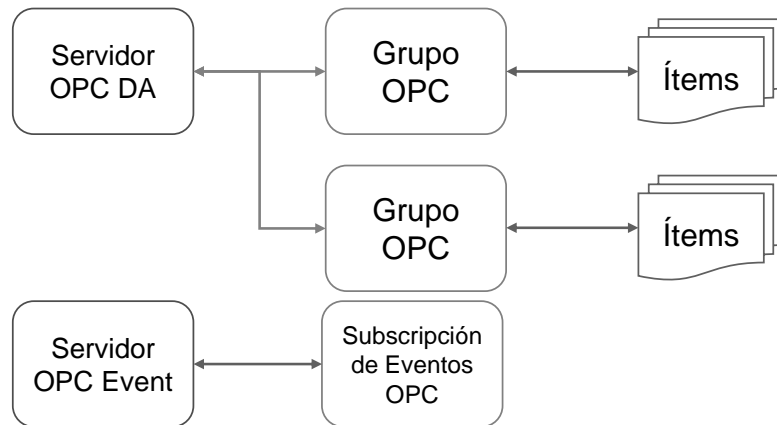


Figura 1.40: Objetos creados por un cliente OPC para el acceso de información y la recepción de alarmas y eventos [66].

OPC Unified Architecture Una vez que OPC clásico se convirtió en un estándar para proveer una interfaz de automatización, OPC XML-DA fue el primer intento de la Fundación OPC para mantener las características de OPC-DA, pero usando una plataforma de comunicación neutral. Con el auge de Internet, se hace imperativa la implementación de un protocolo que permita la flexibilidad y la transmisión de datos necesaria para proveer un protocolo rápido y fiable que permita interconectar los componentes de un sistema de automatización a los servicios web.

Con esta concepción, se puede definir a OPC UA como una actualización del protocolo clásico OPC para su compatibilidad con servicios web, manteniendo las ventajas que ha tenido para su implementación global, removiendo las limitaciones propias de su desarrollo en base a las herramientas de Microsoft (COM y DCOM). Esto, gracias al trabajo de los 40 proveedores de tecnología más grandes del mundo, así como agencias de estandarización como la Sociedad Internacional de Automatización (ISA) y como la Comisión Electrotécnica Internacional (IEC), siendo posible su implementación con distintos lenguajes de programación como *Java*, *Microsoft.NET*, *C*, entre otros [66].

Para el levantamiento de servidores OPC existen diferentes opciones en el mercado. Por mencionar algunas opciones, se dispone de *Matrikon OPC Server*, *KEPServer*, etc. co-

mo opciones comerciales usadas para la automatización de plantas industriales, sistemas SCADA y servidores Web, siendo *FreeOPCServer* o la implementación de OPC mediante bibliotecas de Python, opciones gratuitas y más experimentales que pueden servir para automatizar procesos más pequeños requiriendo un nivel de conocimientos técnicos superiores para su implementación [67].

1.4.7.3. OPC en la simulación y co-simulación

Una funcionalidad de OPC, para proveer líneas de comunicación orientadas a la transmisión de datos, puede utilizarse para interconectar programas mediante su arquitectura cliente-servidor, y que éstos puedan realizar sus funciones de acuerdo a los datos que son compartidos entre las aplicaciones involucradas.

Se ha utilizado OPC con éxito para la simulación de entornos ciberfísicos industriales usando el protocolo de arquitectura unificada exitosamente, concluyendo que se pueden realizar mejoras en la inserción de marcas de tiempo para los datos transmitidos así como la necesidad de implementar un filtro para evitar la redundancia innecesaria de información, lo que podría disminuir tiempos de latencia y espacio de memoria [68]. En la misma línea de la comunicación industrial, OPC ha sido utilizado extensivamente para la transmisión de datos en sistemas de control y adquisición de datos en entornos con procesos simulados proveyendo una forma fiable de entrenar operadores humanos antes de su despliegue en procesos industriales reales [69].

En los Sistemas Eléctricos de Potencia, el uso de OPC se ha previsto como una forma de proveer emulación de comunicaciones con la arquitectura unificada [70]. También se ha utilizado para simular despachos de potencia en tiempo real para entrenar a operadores eléctricos con éxito [71]. Sin embargo quizá el trabajo más completo que involucra co-simulación y OPC involucra la simulación de redes inteligentes, y ha sido cubierto de forma extensiva en [72], concluyendo con la fiabilidad y factibilidad de simular sistemas eléctricos de potencia usando OPC como un protocolo de comunicación válido tanto para entornos industriales como para entornos virtuales. En la Fig. 1.41 se puede apreciar un esquema que propone la utilización de este protocolo para este fin.

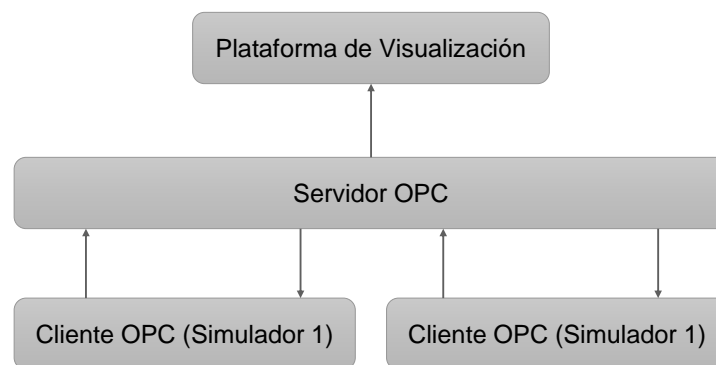


Figura 1.41: Esquema propuesto de comunicación entre aplicaciones utilizando el protocolo OPC [72].

1.4.8. PYTHON

1.4.8.1. Situación actual

En el análisis de escenarios en los Sistemas Eléctricos de Potencia es muy importante levantar modelos con diferentes condiciones de operación, simularlos y analizar los resultados. Esta forma de trabajar puede convertirse rápidamente en una secuencia bastante compleja directamente proporcional al número de condiciones y variables de operación requeridas. Esta complejidad aumenta más cuando se consideran sistemas eléctricos de potencia diferentes a la red tradicional. En estos tiempos, la generación distribuida y las redes inteligentes producen mayores consideraciones a la hora de trabajar con simulaciones de casos de estudio [73].

Un *script* es un conjunto de instrucciones diseñadas para ser ejecutadas por otro programa que funciona sobre un sistema operativo, en vez del procesador. Esto último lo realizan los programas compilados. Los *script* pueden usarse para aumentar el grado de automatización de tareas y funciones que de otro modo requerirían una intervención humana permanente [74]. PowerFactory dispone de dos lenguajes de *script*: el primer lenguaje es el DPL (DigSilent Programming Language), que tiene una sintaxis muy similar a C. Sin embargo, DPL es exclusiva de PowerFactory y requiere un alto nivel de entrenamiento para lograr automatizar funciones complejas. Además, es evidente que es un lenguaje de programación demasiado enfocado a un nicho muy especializado, lo que limita la colaboración de nuevos usuarios. Durante la implementación de este trabajo, se ha podido observar que para un usuario nuevo, los cursos oficiales para lograr una competencia básica en DPL provistos

por la compañía DigSilent tienen precios prohibitivos para la mayoría de los interesados, y la información disponible es escasa, siendo la gran cantidad de ella proveniente de tesis, repositorios y *white papers*.

1.4.8.2. API de Python

Desde la versión 15.1 de PowerFactory, se incluye una Interfaz de Programación de Aplicación (API) que define un conjunto de funciones y objetos que hace posible la manipulación de casi la totalidad de funciones y objetos de PowerFactory mediante un script de Python. Al contrario de DPL, Python es un lenguaje libre de propósito general, siendo quizá el más popular en estos últimos años. Esto es una ventaja considerando que el usuario de PowerFactory puede ser novato y no esté familiarizado con los entresijos de DPL.

Python es un lenguaje multiparadigma, soporta además una programación orientada a objetos y la programación estructurada. Soporta parcialmente otras características como la programación funcional, y tiene una librería de funcionalidades adicionales programadas por una extensa comunidad en línea que puede usarse para potenciar muchos otros proyectos de programación [73]. Una ventaja de Python respecto a DPL es su sintaxis mucho más clara, y el ahorro de líneas de código para realizar funciones repetitivas. Python es popular también en la investigación científica. Paquetes de ingeniería como NumPy, SciPy o paquetes de análisis estadístico como Pandas, paquetes de visualización de datos Matplotlib son de una potencia impresionante. De esta manera, se puede extender el uso de Python para automatizar infinidad de tareas y acceder a los resultados de las simulaciones en forma matricial. En este caso, simplemente se usará Python como un script para sincronizar el inicio de la co-simulación entre PowerFactory y Simulink mediante el uso del protocolo OPC.

1.4.8.3. Python y PowerFactory en modo núcleo para la co-simulación con Simulink

En modo núcleo (engine), PowerFactory puede importar un script externo y ejecutarlo sin la necesidad de disponer de una interfaz gráfica dedicada para la interacción con el usuario. Esto hace posible integrar PowerFactory con Python de forma casi natural. Entre las diferentes actividades que se pueden realizar con la interfaz de PowerFactory están: activación de casos de estudio, ejecución de simulaciones, acceso a las variables de resultados.

La explicación detallada de cómo realizar cada una de estas actividades excede el alcance de este trabajo. Sin embargo, se puede mencionar que el módulo API de interfaz `powerfactory.pyd`, puede funcionar mediante una sencilla línea de comandos, en la que se agrega el módulo a la lista de directorios de Python para que el intérprete pueda dar cuenta

de la existencia de la API de Python-Powerfactory. Una vez que se realiza este procedimiento, Python está en capacidad de iniciar PowerFactory en modo núcleo y así aprovechar la mayor parte de sus funcionalidades. En conjunto con un protocolo de comunicación adecuado, se pueden realizar simulaciones en paralelo junto con otros programas como el caso de Simulink, como se aprecia en la Fig. 1.42. En el capítulo inherente a la metodología se realizará una explicación más detallada de estos procedimientos.

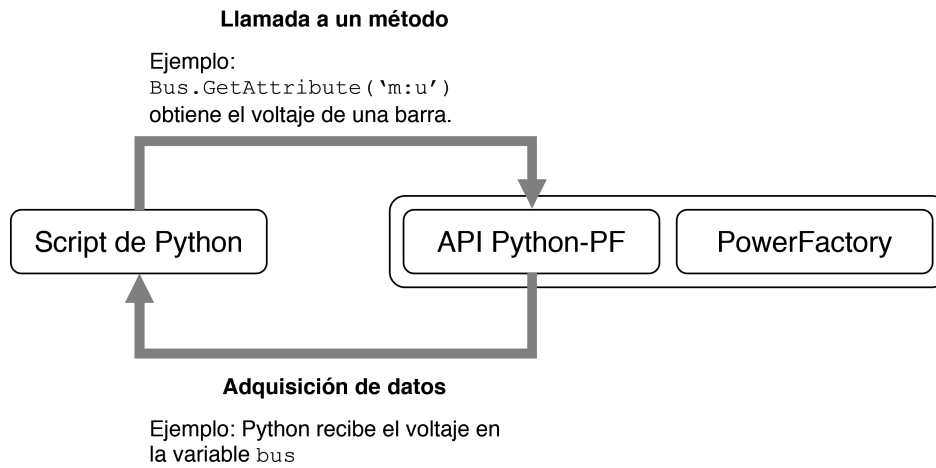


Figura 1.42: Interacción entre un script de Python con PowerFactory a través de la API de Python. Adaptado de [73].

2. METODOLOGÍA

En este capítulo se describe el proceso de diseño e implementación realizado para lograr la co-simulación entre Matlab-Simulink y PowerFactory para el análisis de fenómenos en estado transitorio de un sistema fotovoltaico conectado a la red. Para ello, la metodología se ha dividido en cuatro fases: la fase teórica, que fue desarrollada en el capítulo anterior, consistente en la revisión bibliográfica de los conceptos y recursos tecnológicos a utilizarse. La fase de diseño se describe en este capítulo, presentando los detalles de los modelos, elementos, configuraciones realizadas, así como los cálculos necesarios. La fase de simulación se aprecia en el capítulo de resultados, en donde se presentan los datos obtenidos. Finalmente, la fase de análisis está presente en la última parte del capítulo de resultados, así como en el capítulo de conclusiones y recomendaciones.

2.1. ARQUITECTURA DE LA CO-SIMULACIÓN

Para la correcta implementación del trabajo propuesto, se plantea un esquema de la arquitectura de los recursos necesarios para el efecto, el cual puede apreciarse en la Fig. 2.1.

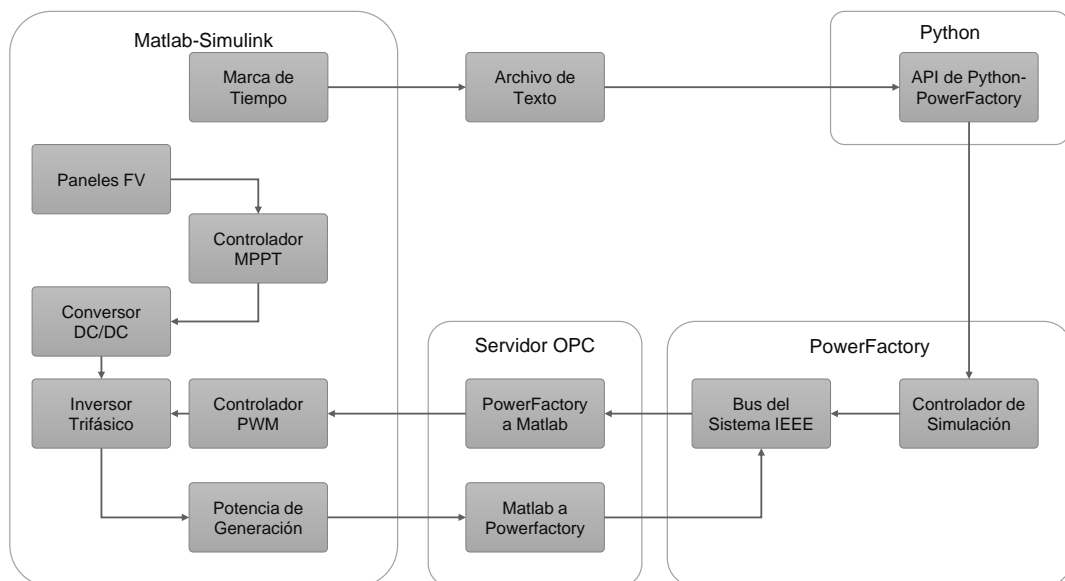


Figura 2.1: Esquema de la simulación propuesta

El funcionamiento de la simulación propuesta se resume en lo siguiente: en Matlab-Simulink se procede a realizar el levantamiento del modelo del Sistema Electrónico de Potencia. En este sistema, se tienen que modelar los paneles fotovoltaicos, considerando las posibles limitaciones que puedan tener los modelos predeterminados. Los paneles fotovoltaicos deben funcionar bajo diferentes condiciones de irradiación y temperatura entregando la máxima potencia disponible. Para esta finalidad, se deben modelar los controladores de seguimiento de máxima potencia, considerando algoritmos que permitan conseguir este objetivo de forma óptima. El algoritmo MPPT -que monitorea voltaje y corriente- funciona sobre un convertidor DC/DC que trabaja con la corriente del panel fotovoltaico y puede modificar el voltaje de salida.

El convertidor DC/DC proporciona el voltaje de bus adecuado para que un inversor trifásico pueda convertir este voltaje de corriente continua en un voltaje trifásico de corriente alterna que puede usarse en transformadores para conectarlo a la red. El proceso de conversión se realiza mediante un controlador PWM que activa o desactiva las diferentes compuertas de acuerdo a los algoritmos programados en el mismo.

El controlador PWM recibe los datos desde PowerFactory mediante el protocolo de comunicación OPC. A su vez, la potencia es despachada desde Matlab-Simulink y se envía a PowerFactory usando el mismo protocolo de comunicación. La sincronización de disparo de la ejecución de las simulaciones se realiza mediante la API de Python con la biblioteca orientada a PowerFactory mediante la escritura de un archivo de texto desde Matlab-Simulink. Este archivo es leído por la API que mediante un algoritmo, da la autorización para ejecutar la simulación transitoria. En PowerFactory deben programarse algunos parámetros de simulación, lo que será detallado posteriormente.

Para la implementación del trabajo propuesto, se han usado conjuntamente: Matlab R2018a, PowerFactory 15.1.4 y Python 3.0.

2.1.1. HARDWARE UTILIZADO

Los resultados de la simulación podrían variar de acuerdo con la potencia computacional del dispositivo hospedador de los programas necesarios. En virtud de ello, en la Tabla 2.1 se puede apreciar el resumen de los parámetros técnicos más importantes del computador utilizado.

Tabla 2.1: Parámetros técnicos del computador utilizado para la simulación

| Característica | Valor | Unidad |
|---------------------------|--|--------|
| Sistema Operativo | Microsoft Windows 10 Pro | |
| Compilación | 10.0.19044 | |
| Procesador | AMD64 Family 22 Model 48 Stepping 1 1800 Mhz | |
| Disco Duro | Western Digital SSD 480 GB | |
| Tipo de Sistema | 64 | bits |
| Memoria física disponible | 7103 | MB |

2.2. GENERACIÓN FOTOVOLTAICA EN SIMULINK

2.2.1. GENERALIDADES

Desde la versión 2015A, Simulink provee en su biblioteca de elementos orientados a energía eléctrica la capacidad de implementar módulos de arreglos de celdas fotovoltaicas. Estos modelos predeterminados usan los parámetros del *National Renewable Energy Laboratory* para seleccionar módulos de interés de una cantidad importante de fabricantes de módulos registrados [75]. El modelo fotovoltaico que utiliza Matlab-Simulink para la implementación predeterminada en las librerías es casi idéntico al propuesto por Tsai y otros [8].

Como se ha apreciado en el marco teórico, el fenómeno de sombras parciales es muy importante en la generación fotovoltaica por su capacidad de burlar los algoritmos de seguimiento del punto de máxima potencia. Aunque Simulink provee algunos ejemplos en los que se demuestra la capacidad de modelado de este fenómeno [76], solamente se obtiene un modelo estático para obtener las curvas potencia-voltaje y corriente-voltaje, usando un solucionador en tiempo continuo a paso variable. En el desarrollo de este proyecto se intentó replicar la simulación usando solucionadores de tiempo discreto a paso constante. Sin embargo, a pesar de implementar diversas recomendaciones para eliminar los errores de lazos algebraicos, no dejaban de aparecer e interrumpir las operaciones. Se debe mencionar que, la prevalencia de errores basados en lazos algebraicos usando los modelos fotovoltaicos predeterminados de Simulink es un problema recurrente hasta la fecha, siendo especialmente problemática para la resolución de problemas en el dominio temporal transitorio, existiendo algunas sugerencias para solucionar este inconveniente [77, 78, 79].

2.2.2. MODELO DEL PANEL FOTOVOLTAICO

La solución propuesta en este trabajo para abordar el problema de los lazos algebraicos, es la adopción de otro modelo que emule el comportamiento de los paneles fotovoltaicos. El modelo escogido, por su practicidad y simplicidad es el propuesto por Bellini y otros [80]. El modelo ha sido probado y modelado suficientemente para describir de manera fiable el comportamiento de los paneles sin comprometer su exactitud, como lo demuestran otros trabajos basados en él [81].

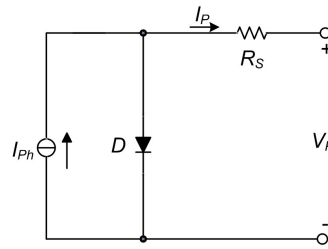


Figura 2.2: Circuito equivalente de un solo diodo con un módulo fotovoltaico usado por Bellini y otros [80].

El modelo propuesto modifica las ecuaciones para la corriente fotovoltaica. Su ventaja radica en la utilización de parámetros conocidos en cualquier hoja de datos de un panel fotovoltaico tales como corrientes de cortocircuito, voltajes de circuito abierto, voltajes de máxima potencia, corrientes de máxima potencia, en vez de utilizar parámetros menos medibles como el caso de las resistencias shunt y paralelas, factores de idealidad de los diodos, constantes físicas de irradiación.

Utilizando el modelo propuesto por Bellini y otros, se puede expresar la corriente de un módulo fotovoltaico en función del voltaje del módulo con la expresión derivada de [82]:

$$I_P = I_{SC} \cdot \left(1 - C_1 \cdot \left(\exp \left(\frac{V_P}{C_2 \cdot V_{OC}} \right) - 1 \right) \right) \quad (2.1)$$

Donde:

$$C_1 = \left(1 - \frac{I_{MPP}}{I_{SC}} \right) \cdot \exp \left(-\frac{V_{MPP}}{C_2 \cdot V_{OC}} \right) \quad (2.2)$$

$$C_2 = \left(\frac{V_{MPP}}{V_{OC}} - 1 \right) \cdot \ln^{-1} \left(1 - \frac{I_{MPP}}{I_{SC}} \right) \quad (2.3)$$

Los coeficientes dependen de los siguientes parámetros de módulo:

- La corriente de cortocircuito I_{SC} .

- El voltaje de circuito abierto V_{OC} .
- El voltaje del punto de máxima potencia V_{MPP} .
- La corriente del punto de máxima potencia I_{MPP} .

Sin embargo, los parámetros anteriormente mencionados deben ser considerados para incluir las variaciones de temperatura e irradiación. Así, las corrientes de cortocircuito y de máxima potencia se modifican:

$$I_{SC} = I_{SCS} \cdot \frac{I_r}{I_{rs}} (1 + \alpha(T - T_s)) \quad (2.4)$$

$$I_{MPP} = I_{MPPS} \cdot \frac{I_r}{I_{rs}} (1 + \alpha(T - T_s)) \quad (2.5)$$

De igual forma, se modifican los voltajes de circuito abierto y de máxima potencia:

$$V_{OC} = V_{OCS} + \beta(T - T_s) \quad (2.6)$$

$$V_{MPP} = V_{MPPS} + \beta(T - T_s) \quad (2.7)$$

En las ecuaciones anteriores, V_{OCS} , V_{MPPS} son los voltajes de circuito abierto y de máxima potencia en condiciones estándar. I_{SCS} , I_{MPPS} son las corrientes de cortocircuito y máxima potencia en condiciones estándar, mientras que α , β son los coeficientes de temperatura respecto a la corriente y voltaje respectivamente. T , T_s son la temperatura ambiental y la temperatura de referencia para las condiciones estándar -generalmente a 25 grados centígrados-, I_r , I_{rs} son la irradiancia ambiental y la irradiancia de referencia (generalmente 1000 vatios por metro cuadrado) en condiciones estándar. Estos valores generalmente son provistos por los fabricantes de paneles fotovoltaicos.

2.2.3. IMPLEMENTACIÓN DEL MODELO EN *SIMULINK*

En la figura 2.3 se puede apreciar el modelo construido en Simulink. El modelo funciona en base a un diagrama de bloques que representa las ecuaciones mencionadas anteriormente. Las variables de entrada son: la irradiancia medida en vatios por metro cuadrado, la temperatura en grados centígrados, el número de celdas en serie (a más celdas, hay mayor voltaje), el número de celdas en paralelo (a más celdas, hay mayor amperaje). La variable de salida es la corriente por panel, la cual se acopla a un generador de corriente dependiente. Los paneles fotovoltaicos deberían generar corriente, no recibirla, por lo que

se usa un diodo para 'forzar' esta característica. También se han incluido las resistencias shunt y la resistencia serie para mejorar la respuesta del modelo, si bien es cierto que se podría prescindir de esta última.

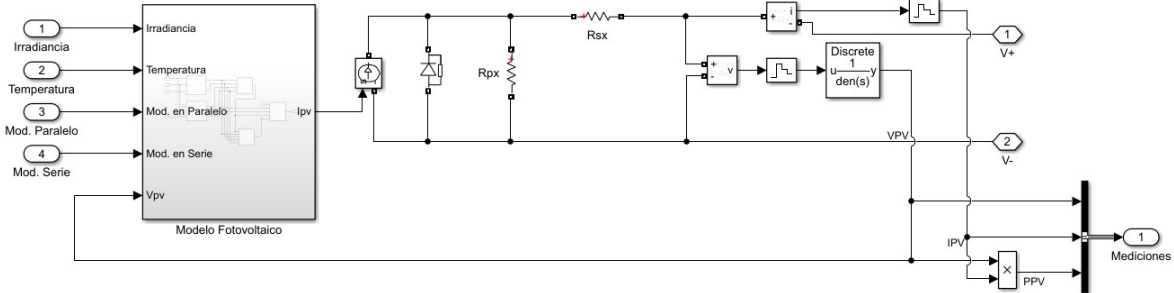


Figura 2.3: Modelo construido en Simulink para la representación del panel fotovoltaico. Incluye una resistencia serie.

El modelo puede proveer un voltaje a la salida, mientras que, se puede monitorear la corriente con la finalidad de realimentar al modelo fotovoltaico. Se pueden realizar, por lo tanto, mediciones de voltaje, corriente y potencia. La validación de los modelos fue realizada en [80]. Este modelo provee una mayor practicidad para trabajar con sombras parciales, las cuales siempre están presentes en la generación fotovoltaica. Para evitar los problemas relacionados con los lazos algebraicos, además del uso de retenedores de orden cero para la simulación en tiempo discreto, es necesario utilizar un filtro pasabajos. Se ha escogido un filtro de la forma:

$$G_f(s) = \frac{1}{\tau s + 1}$$

Se recomienda que el valor de la constante de tiempo $\tau \geq 512T_s$, siendo T_s el tiempo de integración numérica de la simulación. Por lo tanto, una simulación completa debería tener un tiempo mínimo de $t_{sim} \geq 96\tau$, para que se puedan despreciar los efectos del comportamiento dinámico del filtro de primer orden. Los retenedores de orden cero así como el filtro de primer orden, son discretizados con un tiempo de muestreo igual al tiempo de integración numérica de la simulación. El filtro tiene el inconveniente de incrementar el tiempo requerido para completar la simulación. En contramano, se evitan los lazos algebraicos y el consiguiente riesgo de inestabilidad. Se puede observar una gráfica detallada del modelo en la Fig. 2.4.

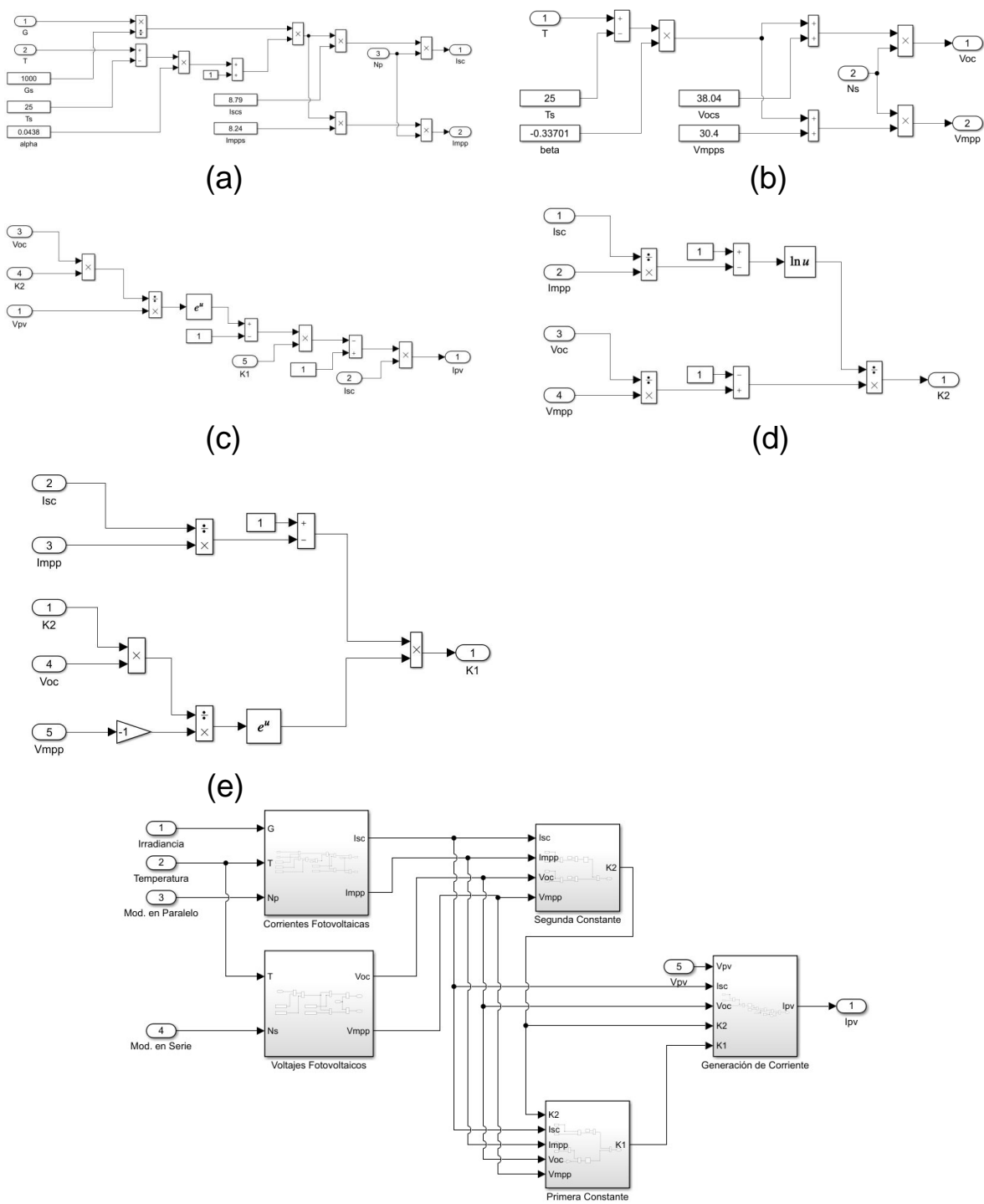


Figura 2.4: Diagrama de bloques detallado del modelo construido en Simulink para la modelación de un panel fotovoltaico. (a) Modelo de la corriente de cortocircuito, (b) Modelo del voltaje a circuito abierto, (c) Modelo de la corriente fotovoltaica del panel, (d) Constante K_2 , (e) Constante K_1 . Finalmente, se puede apreciar la conexión entre los elementos del modelo.

2.2.4. PARAMETRIZACIÓN DEL MODELO FOTOVOLTAICO

Con la finalidad de acercar este estudio a la realidad, se considera al Proyecto Fotovoltaico “Gran Solar” como la fuente de datos principal para la modelación [83]. Este proyecto tiene una capacidad instalada de 3 MW. Este proyecto fotovoltaico utiliza los paneles solares provistos por la compañía china *Yingli Solar*. Este proyecto contempla la utilización de 14800 módulos, divididos en dos partes: 4800 módulos se usan en la estación “Tren Salinas” y 9600 para “Salinas”, usándose conexiones de 20 módulos en serie indistintamente de la estación. En este proyecto, no se considera esta división. Se considera que los 14800 módulos están en una misma estación. Al usarse 20 paneles en serie, esto significa que deben existir 740 módulos en paralelo. Los datos de los paneles solares vienen dados en la Tabla 2.2.

Tabla 2.2: Parámetros teóricos por módulo, del panel fotovoltaico Yingli Solar YL250P-29P en condiciones estándar.

| Parámetro | Valor | Unidad |
|------------------------------|----------|------------------------------|
| Potencia nominal de salida | 250 | Vatios |
| Eficiencia del módulo | 15,3 | Porcentaje |
| Voltaje de máxima potencia | 30,4 | Voltios |
| Voltaje de circuito abierto | 38,7 | Voltios |
| Corriente de máxima potencia | 8,24 | Amperios |
| Corriente de cortocircuito | 8,79 | Amperios |
| Resistencia serie | 0,42475 | Ohmios |
| Resistencia shunt | 364,6382 | Ohmios |
| Constante alfa de corriente | 0,0438 | Amperio por grado centígrado |
| Constante beta de voltaje | -0,33701 | Voltio por grado centígrado |

Se recuerda el esquema de conexión empleado para las sombras parciales: el parque fotovoltaico se divide en dos zonas que se conectarán en serie. Cada zona dispone de 10 módulos en serie y 740 módulos en paralelo. Al conectarse las zonas, se disponen de 20 módulos en serie y 740 módulos en paralelo, lo que dan los 14800 módulos necesarios. La primera zona recibirá una irradiancia total, mientras que, la segunda zona recibirá una irradiancia parcial. Al considerar condiciones ideales (1000 vatios por metro cuadrado y 25 grados centígrados), el parque fotovoltaico tendrá los parámetros teóricos que se muestran en la Tabla 2.3.

Tabla 2.3: Parámetros teóricos del parque fotovoltaico en condiciones estándar.

| Parámetro | Valor | Unidad |
|------------------------------|---------|----------|
| Potencia nominal de salida | 3,70 | MW |
| Voltaje de máxima potencia | 608,00 | Voltios |
| Voltaje de circuito abierto | 774,00 | Voltios |
| Corriente de máxima potencia | 6097,60 | Amperios |
| Corriente de cortocircuito | 6504,60 | Amperios |
| Resistencia serie | 0,0115 | Ohmios |
| Resistencia shunt | 9,8551 | Ohmios |

Para modelar el efecto de las sombras parciales en el modelo fotovoltaico, se ha considerado un esquema de dos plantas conectadas en serie. En la primera planta, se recibirá la irradiancia completa, mientras que en la entrada de irradiancia para la segunda planta se añade un bloque de ganancia que se sugiere se puede modificar entre cero y uno para modelar este efecto. El esquema de construcción puede observarse en la Fig. 2.5.

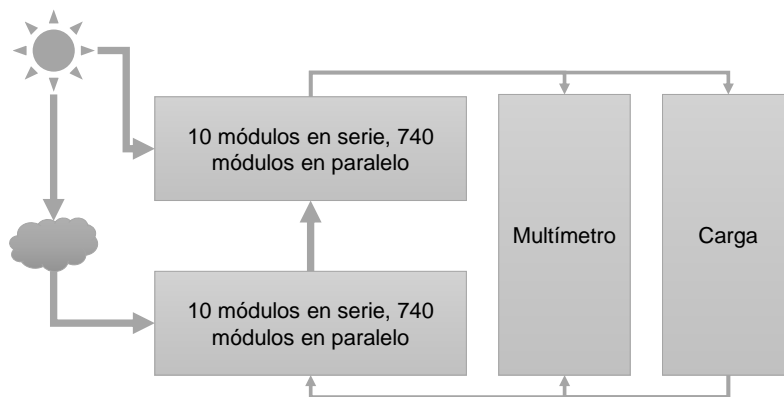
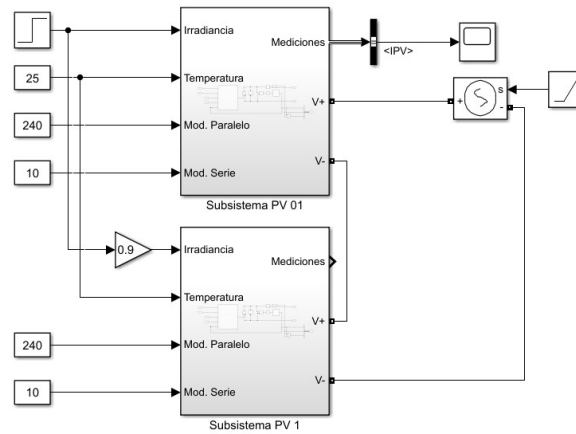


Figura 2.5: Conexión del modelo implementado en Simulink para modelar el efecto de sombras parciales en la planta fotovoltaica.

Posteriormente, se realizan simulaciones para determinar las curvas $I(V)$, $P(V)$ del panel fotovoltaico, considerando las modificaciones planteadas en esta sección. Los resultados de estas simulaciones se pueden apreciar en las Figs. 2.6 y 2.7. Las gráficas obtenidas son consistentes con el marco teórico establecido en la sección 1.4.1 (pág. 4).

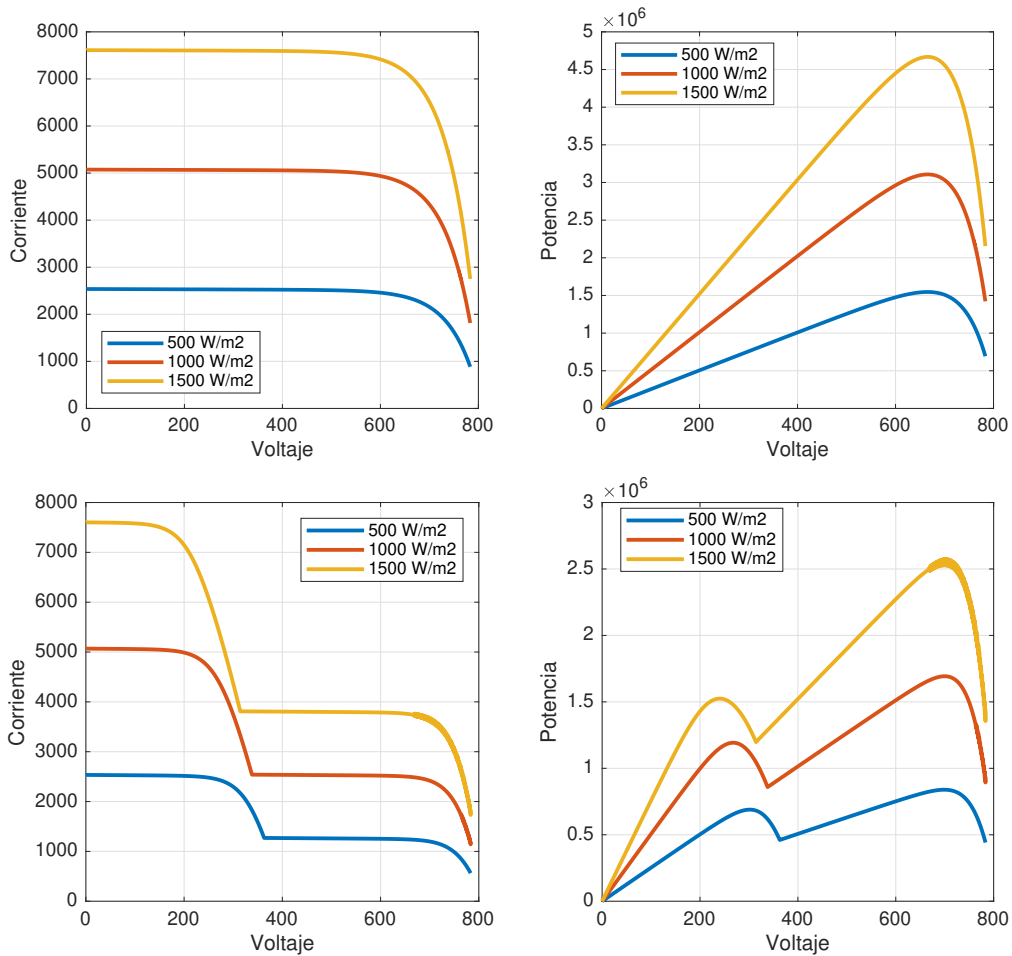


Figura 2.6: Gráficas $I(V)$, $P(V)$ para el modelo fotovoltaico con cambios de irradiancia. La temperatura es constante (25 grados), y se han variado las irradiancias. En las dos gráficas superiores no existe efecto de sombras parciales, en las dos gráficas inferiores existe efecto de sombras parciales, considerando que hay una pérdida del 50 % de potencia por área en la zona 2.

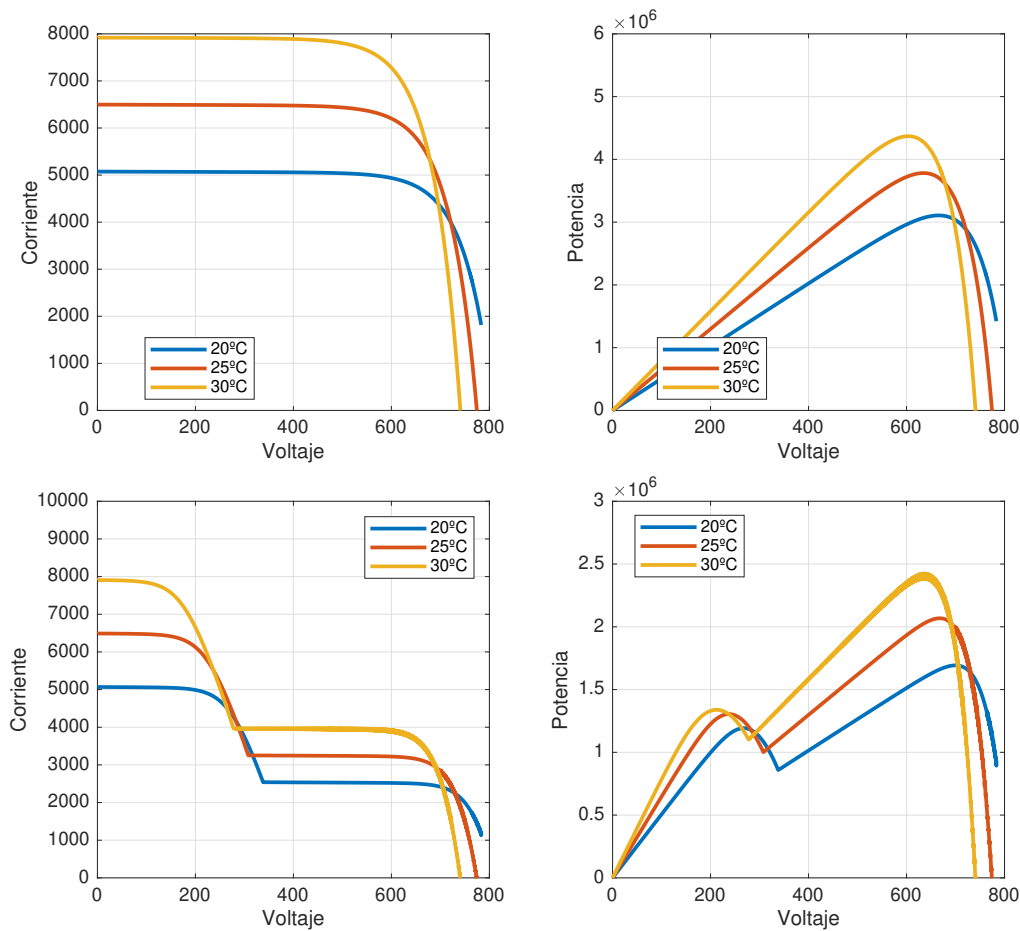


Figura 2.7: Gráficas $I(V)$, $P(V)$ para el modelo fotovoltaico con cambios de temperatura. La irradiancia es constante (1000 W/m^2), y se han variado las temperaturas. En las dos gráficas superiores no existe efecto de sombras parciales, en las dos gráficas inferiores existe efecto de sombras parciales, considerando que hay una pérdida del 50 % de potencia por área en la zona 2.

2.2.5. PERFILES DE IRRADIANCIA

Para la sintonización de controladores y los experimentos usando solamente Simulink, se consideran perfiles de radiancia constantes de 1000 vatios por metro cuadrado, o saltos de 1000 a 500 vatios por metro cuadrado a la mitad del tiempo de simulación.

Sin embargo, esta tesis consideró la obtención de valores de irradiancia provenientes del Atlas Solar del Ecuador [84]. Sin embargo, las mediciones realizadas no presentaban la exactitud necesaria para replicar la realidad de la Planta Salinas, la cual se tomó como caso de referencia. Al momento del planteamiento del proyecto de titulación, todavía no se tenían disponibles datos precisos de EMELNORTE, la empresa encargada de monitorear

la planta fotovoltaica. En la Fig. 2.8 se puede apreciar una de las gráficas que pueden encontrarse en el atlas mencionado.

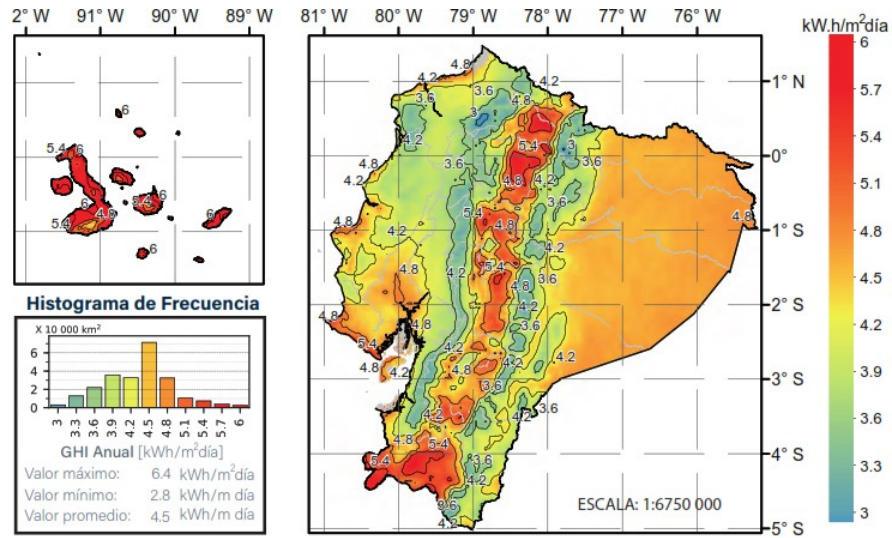


Figura 2.8: Irradiación Solar Global Horizontal Anual para el Ecuador [84].

Posteriormente, se pudo tener acceso a algunos datos de irradiancia medidos directamente en campo. El perfil se puede apreciar en la Fig. 2.9. Estos datos se han conseguido gracias a mediciones realizadas con anterioridad [83].

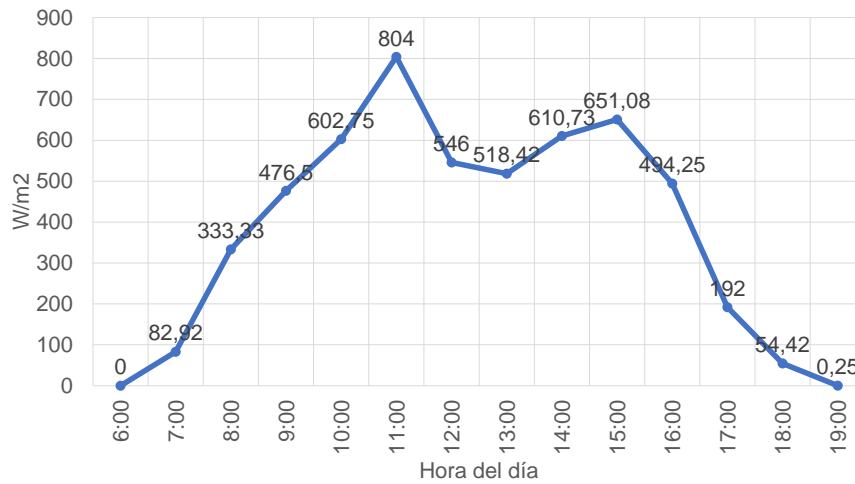


Figura 2.9: Perfil de radiación de la Central Fotovoltaica Salinas, provista por EMELNORTE [83].

2.3. CONVERTIDORES ELECTRÓNICOS DE POTENCIA EN SIMULINK

2.3.1. MODELO DEL CONVERTIDOR DC-DC

2.3.1.1. Trabajos previos

En [16] se realizó una investigación en la que se comparaban las distintas alternativas en los convertidores electrónicos de potencia que se usarían en los proyectos de generación fotovoltaica. De acuerdo al estudio realizado, el convertidor electrónico de potencia más recomendable para su uso en la etapa de elevación de voltaje desde los paneles fotovoltaicos está entre los convertidores Cúk y SEPIC. En etapas tempranas del trabajo, se consideró su utilización. Sin embargo, la precisión de las simulaciones realizadas no eran las adecuadas considerando la interconexión entre dos aplicaciones.

Estas dificultades podrían ser vinculadas a la complejidad de estas topologías y el algoritmo de seguimiento del punto de máxima potencia utilizado. Para abordar esta problemática, se ha utilizado entonces, un convertidor sencillo que está presente en la literatura referente a seguidores de máxima potencia.

2.3.1.2. Elección y diseño del convertidor DC/DC

El trabajo realizado en [85] propone un algoritmo para optimizar el seguimiento de máxima potencia usando conductancia incremental y PID. Para la demostración de su enunciado, se usa un convertidor elevador sencillo, obteniendo eficiencias superiores al 95 %. Se usará este punto de vista en el proyecto.

Para el cálculo de los parámetros del convertidor elevador, deben considerarse algunos aspectos de diseño. En [12] se presentan las ecuaciones que deben considerarse para este efecto.

En primera instancia, se tomaron las curvas prácticas de potencia y corriente respecto al voltaje, considerando condiciones ideales de funcionamiento. Así, se obtuvieron los datos que se muestran en la Tabla 2.4. En esta tabla se tabulan los valores utilizados para simular el convertidor elevador. Este proceso de diseño se realiza en el punto de máxima potencia, es decir, con los valores de voltaje de máxima potencia, la corriente de máxima potencia, la potencia máxima, además de los nuevos valores del voltaje de circuito abierto, la corriente de cortocircuito obtenidas a condiciones estándar (1000 W/m^2 , 25°C) obtenidas con el modelo fotovoltaico propuesto en [80].

Tabla 2.4: Comparación entre los valores teóricos y los valores medidos de la planta fotovoltaica implementada usando el método de Bellini.

| Parámetro | Valor teórico | Valor real | Error (%) | Unidad |
|------------|---------------|------------|-----------|--------|
| P_{mppt} | 3,71 | 3,69 | -0,54 | MW |
| I_{mppt} | 6,09 | 5,91 | -2,96 | kA |
| V_{mppt} | 608,00 | 625,00 | 2,80 | V |
| P_{oc} | 774,00 | 763,00 | -1,42 | V |
| I_{sc} | 6,50 | 6,49 | -0,15 | kA |

Como se aprecia en la Tabla 2.4, el error máximo entre los modelos no supera el 3%, con lo que se verifica la validez del modelo simplificado de Bellini. Para el diseño del convertidor elevador, se usan las fórmulas indicadas en [12] para encontrar los valores de las inductancias y las capacitancias críticas. Para asegurar la conducción continua, se usa una inductancia de diseño 24 veces mayor a la inductancia crítica y una capacitancia de diseño 48 veces mayor a la capacitancia crítica. Los valores de sobredimensionamiento se han propuesto en base a métodos empíricos. Los parámetros de diseño se pueden apreciar en la Tabla 2.5.

Tabla 2.5: Parámetros numéricos de diseño para el convertidor elevador.

| Parámetro | Valor | Unidad |
|------------------------|-----------------------|--------|
| V_{out} | $\sqrt{2} \cdot 1000$ | V |
| R_{mppt} | 0,54 | Ohm |
| $D_{mppt-out}$ | 0,5581 | |
| $f_{sw \text{ Boost}}$ | 12,6 | kHz |

Considerando al voltaje de máxima potencia como el voltaje de entrada del panel y a la potencia máxima como la potencia operativa del convertidor elevador, se pueden aplicar estos valores en las ecuaciones para la inductancia y capacitancia críticas del convertidor elevador:

$$L_{cr} = \frac{V_{in} \cdot (V_{out} - V_{in})}{2 \cdot f_{sw} \cdot P_{out}} \quad (2.8)$$

$$C_{cr} = \frac{P_{out} \cdot (V_{out} - V_{in})}{2f_{sw} \cdot V_{out}^3} \quad (2.9)$$

Reemplazando los valores de diseño, se obtienen los valores numéricos de estos componentes del convertidor elevador:

$$L_{cr} = \frac{625 \cdot (\sqrt{2} \cdot 1000 - 625)}{2 \cdot 12600 \cdot 3,69 \times 10^6} \approx 5,30[\mu\text{H}]$$

$$C_{cr} = \frac{3,69 \times 10^6 \cdot (\sqrt{2} \cdot 1000 - 625)}{2 \cdot 12600 \cdot (\sqrt{2} \cdot 1000)^3} \approx 40[\mu\text{F}]$$

Aplicando los excesos en el dimensionamiento de los elementos, con la finalidad de asegurar la conducción continua, los valores del convertidor elevador quedan de la siguiente forma:

$$L = 127[\mu\text{H}] \quad C = 1920[\mu\text{F}]$$

Se ha incluido un condensador de entrada. Este condensador tiene un valor de 100 microfaradios, y su valor ha sido determinado mediante métodos empíricos, evaluando sucesivamente en las simulaciones, la respuesta del sistema a diferentes condiciones de operación. En la Fig. 2.10 se puede apreciar el modelo del convertidor implementado en Simulink. Los cálculos han sido incluidos en el script de parámetros iniciales programado en Matlab para la carga previa a la simulación del modelo.

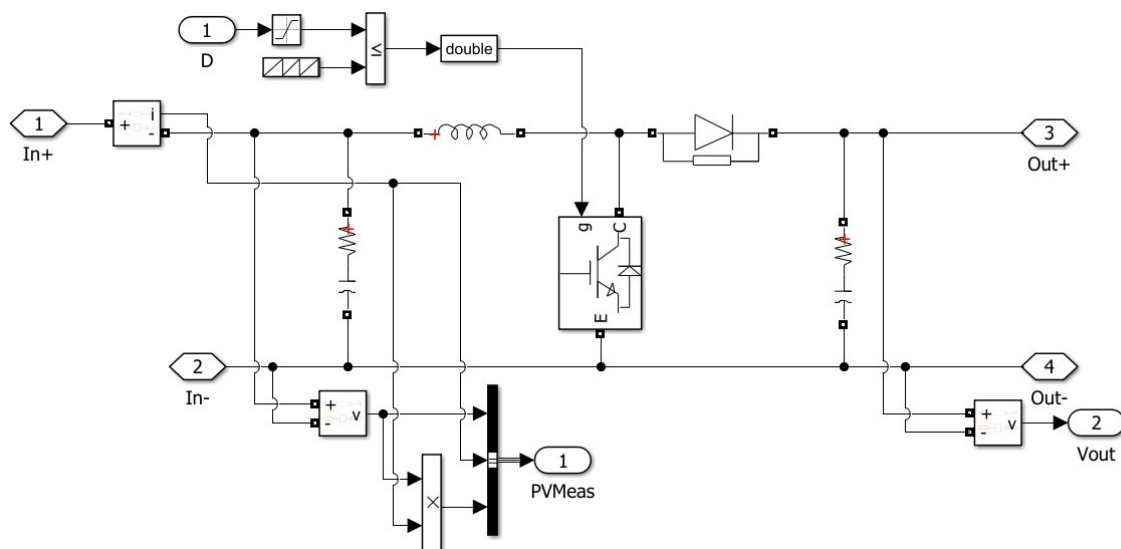


Figura 2.10: Modelo del convertidor elevador desarrollado en Simulink para la simulación del sistema fotovoltaico.

2.3.2. MODELO DEL CONVERTIDOR DC-AC

Una vez estabilizado el voltaje de entrada al inversor, se puede convertirlo a un voltaje trifásico en corriente alterna. Este proceso se realiza mediante la implementación de una

topología en puente trifásico. Se utilizará un inversor del modelo estándar de las librerías de Simulink, en concreto, el convertidor de dos niveles que se puede apreciar en la Fig. 2.11. Al ser un modelo estándar, solamente se pueden modificar los parámetros correspondientes a la resistencia del convertidor en conmutación que se establecerá en 0,1 ohmios.

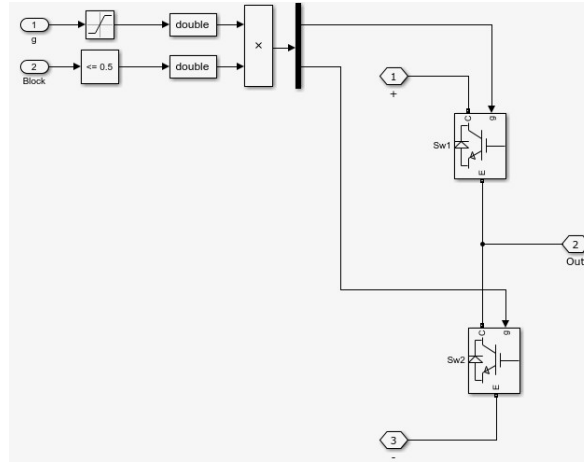


Figura 2.11: Modelo detallado del bloque predeterminado de inversor utilizado en Simulink para el convertidor trifásico DC/AC.

2.3.3. FILTRO PASIVO

Para esta simulación, se ha escogido implementar un diseño correspondiente para un filtro pasivo LCL, como se propone en [86]. La inductancia y la capacitancia han de escogerse de acuerdo con las siguientes expresiones:

$$L_f = \frac{0,1 \cdot U^2}{2\pi \cdot f \cdot P_p} \quad (2.10)$$

$$C_f = \frac{0,05 \cdot P_p^2}{2\pi \cdot f \cdot U^2} \quad (2.11)$$

Donde U es el voltaje entre líneas de la barra trifásica, f es la frecuencia de la red, $P_p = \frac{P_{pv}}{3}$ es la potencia monofásica que se obtiene dividiendo la potencia teórica del panel para 3. Con estos datos se pueden encontrar los valores de filtro usados en la simulación y en la Fig. 2.12.

$$L_f = \frac{0,1 \cdot (270)^2}{2\pi \cdot 60 \cdot 0,33 \cdot (3,7 \times 10^6)} \approx 49,26[\mu\text{H}]$$

$$C_f = \frac{0,05 \cdot 0,33 \cdot (3,7 \times 10^6)^2}{2\pi \cdot 60 \cdot (270)^2} \approx 2243,84[\mu\text{F}]$$

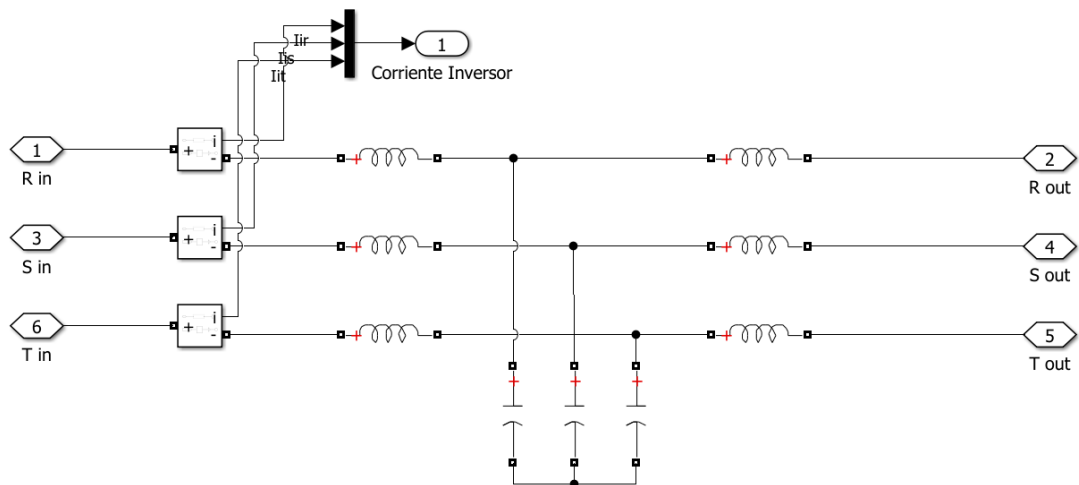


Figura 2.12: Filtro LCL implementado en Simulink para la reducción de armónicos.

2.4. CONTROL DE CONVERTIDORES ELECTRÓNICOS DE POTENCIA EN SIMULINK

2.4.1. CONTROL DE VOLTAJE DEL CONVERTIDOR ELEVADOR

Para el diseño de esta etapa, se utiliza un enfoque ligeramente diferente al usual. Generalmente, en el convertidor elevador la entrada del sistema de control es el ciclo de trabajo y la salida es el voltaje de la carga. En los paneles fotovoltaicos, el voltaje de la carga no es controlada directamente, siendo la variable de interés la entrada al convertidor, es decir, el voltaje del panel fotovoltaico. En primera instancia, se efectúa una prueba a lazo abierto mediante la perturbación del ciclo de trabajo. Se inicia con un $D = 0,10$ y se lo perturba hasta un ciclo de trabajo del 90 %. La resistencia de carga tiene el valor indicado en la Tabla 2.5. Los resultados se pueden apreciar en la Fig. 2.13.

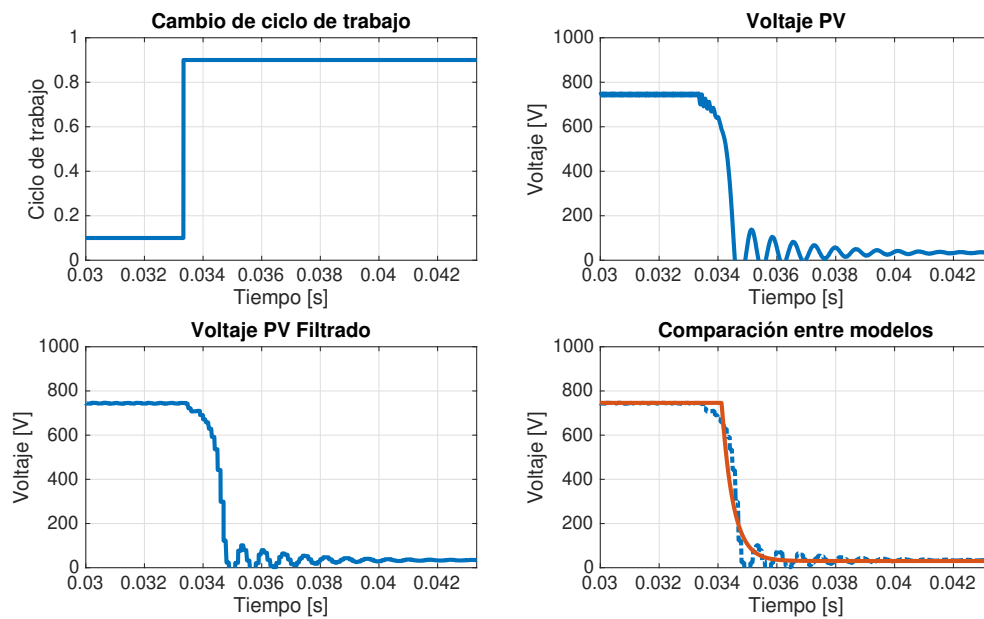


Figura 2.13: Respuesta en lazo abierto del voltaje del panel fotovoltaico frente a un cambio al escalón.

Como se aprecia, el ciclo de trabajo ha incidido en el voltaje del panel fotovoltaico. La respuesta de voltaje natural es la respuesta típica de cualquier sistema basado en electrónica de potencia de conmutación a alta frecuencia, lo que impide su correcta identificación a simple vista. Para mejorar esta respuesta, se utiliza un filtro pasabajos. Este filtro tiene la funcionalidad de disminuir las oscilaciones en alta frecuencia, a costa de generar un retardo de tiempo mucho mayor en la respuesta de la planta. A pesar de la no linealidad del sistema, se puede apreciar que, existe una respuesta similar a una función de primer orden con retardo, por lo que se procederá a examinar la respuesta con el método mencionado en la sección 1.4.3.6.

Tabla 2.6: Tabulación de parámetros, identificación aproximada del sistema y constantes teóricas de sintonización para el convertidor elevador.

| Parámetro de identificación | Valor | Parámetro de sintonización | Valor |
|-----------------------------|---------------------------|----------------------------|---------------------------|
| x_o | 0,10 | K | -894,49 |
| x_f | 0,90 | τ | $3,90 \times 10^{-4}$ [s] |
| y_o | 746 [V] | t_0 | $7,90 \times 10^{-4}$ [s] |
| y_f | 30,41 [V] | K_p | $-2,76 \times 10^{-4}$ |
| y_{28} | 545,63 [V] | τ_i | $3,90 \times 10^{-4}$ [s] |
| y_{63} | 295,18 [V] | K_i | $-7,08 \times 10^{-1}$ |
| t_{28} | $9,20 \times 10^{-4}$ [s] | T_s | 10×10^{-6} [s] |
| t_{63} | $1,18 \times 10^{-3}$ [s] | | |

Los resultados de la identificación del sistema se pueden apreciar en la Tabla 2.6. El controlador digital que deba diseñarse debe tener un tiempo de muestreo ser más rápido que la constante de tiempo de la función de transferencia obtenida:

$$G_p(s) = \frac{-894,49}{3,90 \times 10^{-4} \cdot s + 1} \cdot e^{-7,90 \times 10^{-4} \cdot s}$$

El valor de la constante de tiempo puede ser usada con la finalidad de encontrar el valor del tiempo de muestreo idóneo para el controlador digital. Así pues:

$$T_s \leq \frac{\tau}{21} \rightarrow T_s \leq 18,57[\mu s]$$

También se puede apreciar que, la función de transferencia indica una acción inversa a la entrada. Es decir, una elevación en el ciclo de trabajo produce un decremento en el voltaje de salida del convertidor, y viceversa. Esto es de vital importancia para la programación del controlador de seguimiento del punto de máxima potencia. A continuación, se procede a probar el funcionamiento del convertidor elevador en lazo cerrado con $K_p = 2,76 \times 10^{-4}$ y $K_i = 7,08 \times 10^{-1}$. Como la acción es inversa, el error puede ser expresado en función de $e = V_{\text{medido}} - V_{\text{referencia}}$. Se ha usado una discretización Euler de adelanto. Los resultados pueden apreciarse en la Fig. 2.14. El controlador PI cumple su funcionalidad, y el tiempo de establecimiento es menor a 15 milisegundos.

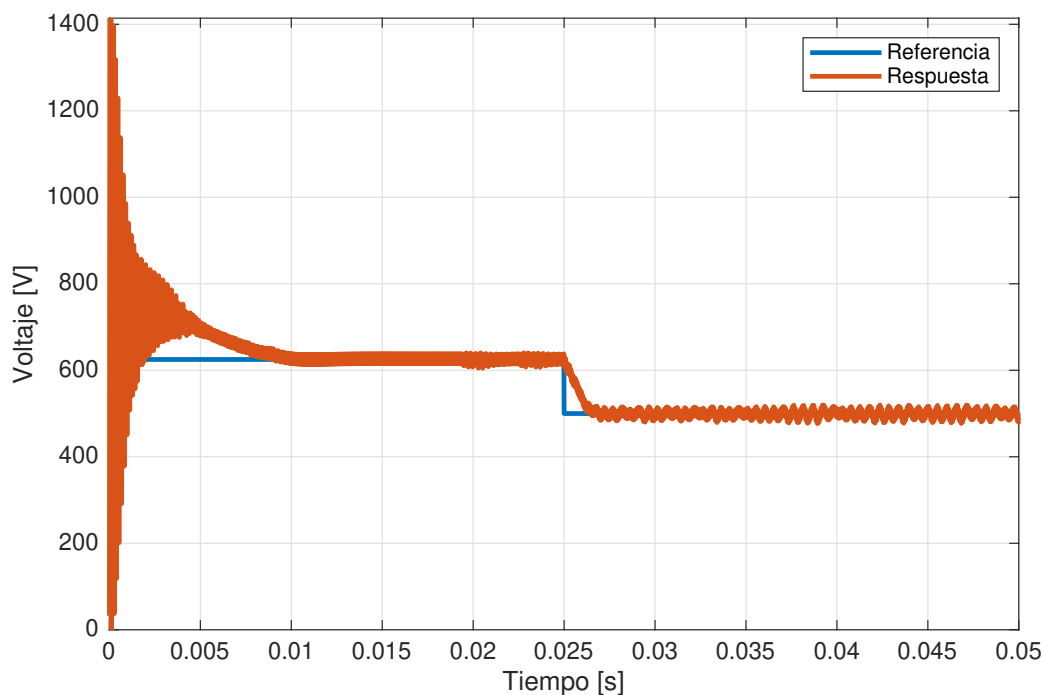


Figura 2.14: Respuesta de lazo cerrado para el convertidor elevador con controlador PI sintonizado por el método de Dahlin.

2.4.2. SEGUIMIENTO DEL PUNTO DE MÁXIMA POTENCIA

Para el seguimiento de potencia del panel fotovoltaico, se ha precisado utilizar dos algoritmos: Perturbar y Observar, y Conductancia Incremental de paso fijo. Los códigos pueden apreciarse en los algoritmos A.3 y A.4. En la Fig. 2.15 se puede apreciar que por medio de un interruptor manual se puede escoger el mecanismo de seguimiento de máxima potencia entre los dos algoritmos mencionados. El tiempo de muestreo para los algoritmos es de 10 microsegundos.

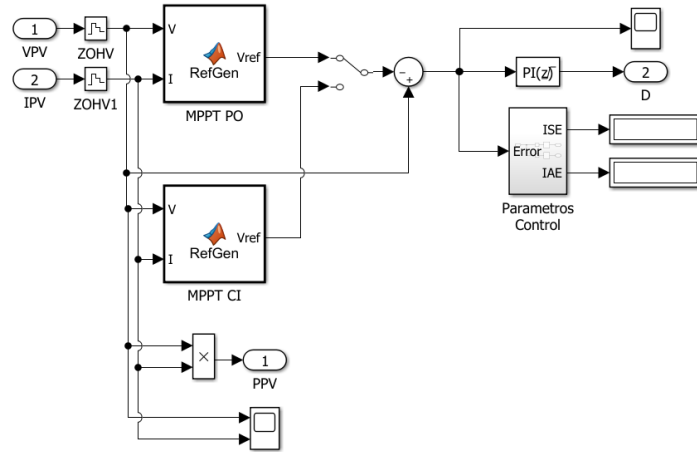


Figura 2.15: Diagrama de bloques en Simulink para el algoritmo de máxima potencia y el controlador PI digital.

2.4.3. CONTROL DEL INVERSOR TRIFÁSICO

Para el disparo de los semiconductores utilizados, se requiere una modulación que utiliza una transformación de coordenadas de sistema rotatorio a sistema fijo utilizando las transformadas de Park y de Clark. Un esquema del control realizado se puede apreciar en la Fig. 2.16.

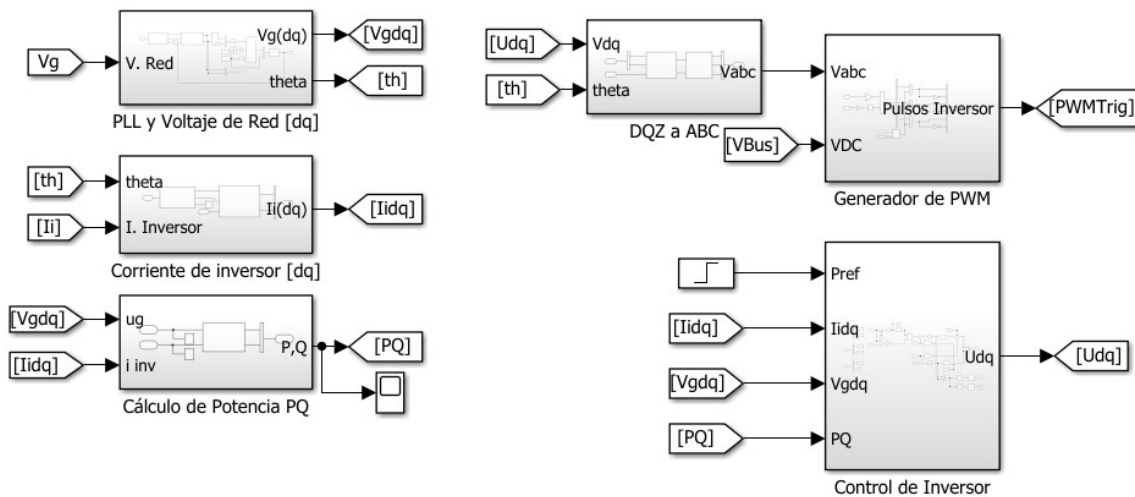


Figura 2.16: Diagrama de bloques de los modelos implementados en Simulink para el control del inductor.

Resumidamente, para el control del inductor, se ha optado por un control de voltaje para el eje directo y un control de corriente en el eje de cuadratura. Esto significa, que el voltaje será controlado en el eje directo, mientras que la potencia reactiva será controlada en el eje

de cuadratura. Para la modulación, la frecuencia de la red se sincroniza con la frecuencia que produce el inversor usando un lazo de seguimiento de fase.

2.4.3.1. Lazo de seguimiento de fase

En la práctica, la frecuencia de la red no siempre es constante. De hecho, la frecuencia de la red varía siempre, dentro de parámetros establecidos en las normas eléctricas de calidad de energía. Pueden ocurrir eventos transitorios que provoquen que la frecuencia se desvíe de la frecuencia de referencia —50 hercios en Europa, 60 hercios en Estados Unidos y su zona de influencia, 400 hercios en el sector aeronáutico—. Con una técnica de modulación trifásica constante, puede ocurrir una desincronización entre el voltaje producido por el inversor y el voltaje de la red con consecuencias no deseadas en un sistema trifásico, puesto que, en el caso de inversores monofásicos, podrían utilizarse técnicas más sencillas como disparos mediante detección de cruces por cero.

En la Fig. 2.17 se puede apreciar el modelo PLL implementado en Simulink. Este modelo utiliza las funciones creadas para la transformación de Clarke y la transformación Clarke-DQZ. Este modelo permite escoger entre el eje directo o el eje de cuadratura para la alineación con el marco rotatorio de referencia. Si bien es cierto que se puede escoger esta característica, se busca una consistencia con el marco teórico, por lo que el eje de cuadratura es el escogido para alinearse con el marco rotatorio de referencia.

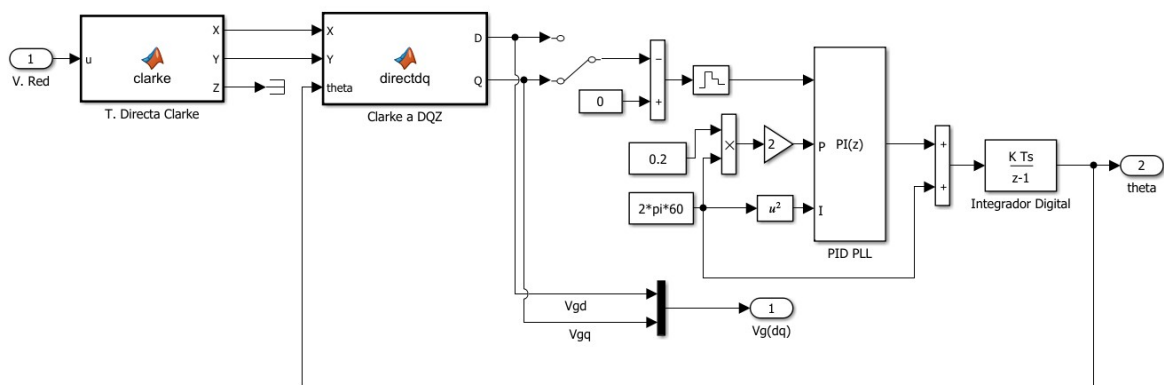


Figura 2.17: Modelo de lazo de seguimiento de fase implementado en Simulink.

Se implementa un control discreto mediante un retenedor de orden cero con un periodo de muestreo equivalente a $8 \cdot T_s$, siendo T_s el periodo de integración numérica para la simulación. Los parámetros del controlador PI discreto son:

$$K_p = 2 \cdot 0,2 \cdot (2\pi \cdot 60)$$

$$K_i = (2\pi \cdot 60)^2$$

Con estos parámetros de sintonización, y con el tiempo de muestreo de $8 \cdot T_s$, la respuesta del lazo de seguimiento de fase es excelente, requiriéndose un tiempo de establecimiento de 100 microsegundos. Las transformaciones $abc \rightarrow \alpha\beta 0$ y $\alpha\beta 0 \rightarrow dqz$ se programaron mediante códigos computacionalmente eficientes que pueden consultarse en los algoritmos A.5 y A.6.

2.4.3.2. Sintonización de controladores de Eje Directo, Eje de Cuadratura y Voltaje de Bus

En esta sección, se ha utilizado un enfoque totalmente empírico, puesto que las ecuaciones de sintonización propuestas no fueron realmente útiles. Mediante sucesivas simulaciones, se han determinado los parámetros que se pueden observar en la Tabla 2.7. Estos parámetros se utilizan para la sintonización de los diagramas de bloques apreciados en la Fig. 2.18. Se ha usado una discretización Euler de adelanto para los controladores.

Tabla 2.7: Parámetros de sintonización de los controladores del inversor trifásico.

| | Control I_d | Control I_q | Control V_{bus} |
|-------|---------------|---------------|-------------------|
| K_p | 3 | 3 | 3,5 |
| K_i | 1000 | 1000 | 1000 |
| T_s | 16 [us] | 16 [us] | 16 [us] |

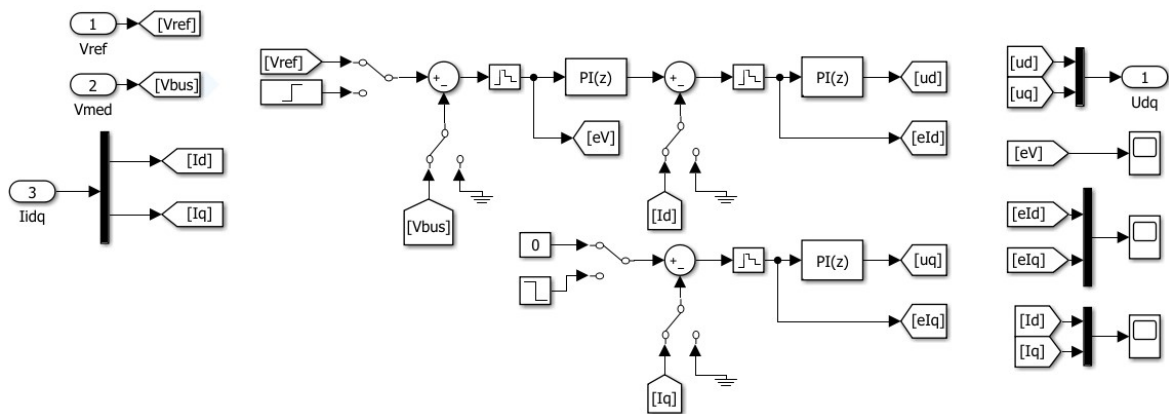


Figura 2.18: Diagrama de bloques de los modelos implementados en Simulink para el control de corriente I_d, I_q .

2.4.3.3. Modulación SPWM

Para la modulación del inversor, se utilizan las antitransformadas de Park y Clarke, este proceso se realiza porque la salida de los controladores de corriente de eje directo y eje de cuadratura se utiliza para generar los voltajes de referencia de eje directo y eje de cuadratura. Estos voltajes deben reconstruirse con la referencia del lazo de seguimiento de fase. Además, para evitar submodulaciones o sobremodulaciones excesivas, se debe considerar el voltaje del bus DC, para que el voltaje por ramal pueda compararse adecuadamente. Para la generación de las ondas senoidales se utiliza una señal triangular con un rango entre -1 y 1, con una frecuencia de conmutación de 37.8 kHz. El diagrama de flujo puede apreciarse en la Fig. 2.19. El voltaje de bus DC es sobredimensionado en un 2% para asegurar la condición de submodulación en condiciones normales.

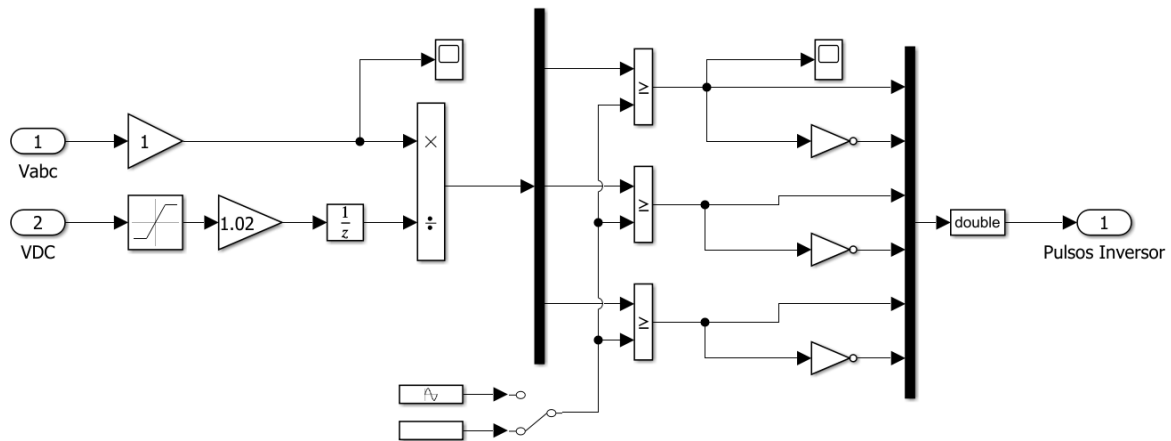


Figura 2.19: Diagrama de bloques de los modelos implementados en Simulink para la modulación SPWM.

2.5. MODELO DEL SEP EN POWERFACTORY

Se utiliza un modelo de prueba muy utilizado en la Ingeniería Eléctrica. El modelo WSCC conocido también como el modelo de prueba de Anderson [87], es una representación aproximada muy simple del *Western System Coordinating Council* a un sistema equivalente de 9 barras y 3 generadores. Este modelo de prueba originalmente consiste en 9 barras, 3 generadores, 3 transformadores de dos devanados, 6 líneas de transmisión y 3 cargas. Los voltajes base utilizados en este modelo son 13.8 kV, 16.5 kV, 18 kV y 230 kV [88]. El diagrama unifilar, así como los parámetros de este sistema pueden observarse en la Fig.

2.20.

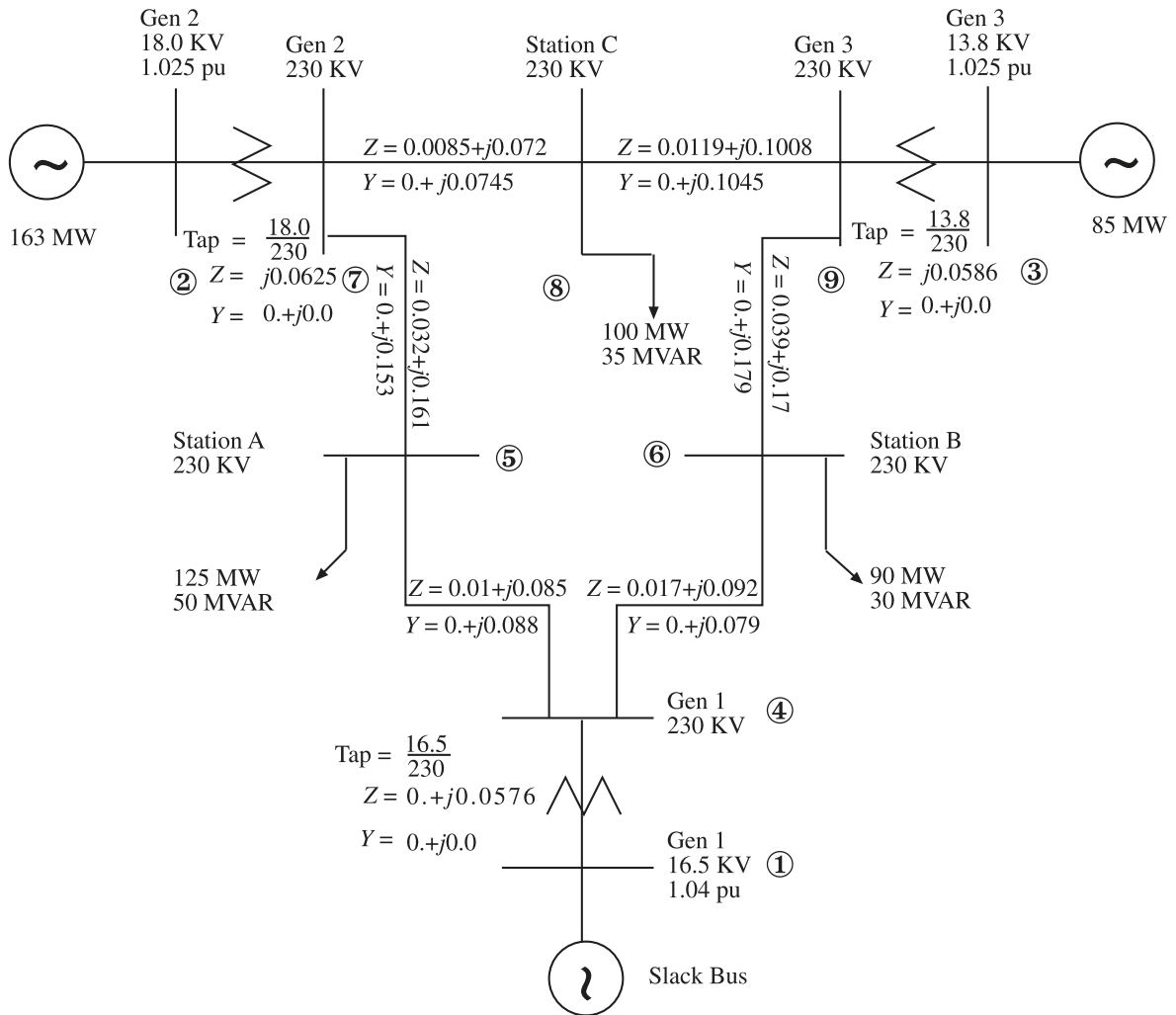


Figura 2.20: Diagrama unifilar del modelo de prueba WSCC (IEEE) 9 Barras [89]

Al ser un modelo de prueba bastante popular, en PowerFactory se puede encontrar este ejemplo por defecto en la librería de ejemplos predeterminados de este programa. En principio, se procede a la importación del modelo para su posterior modificación. Este proceso puede explicarse en forma gráfica en la Fig. 2.21.

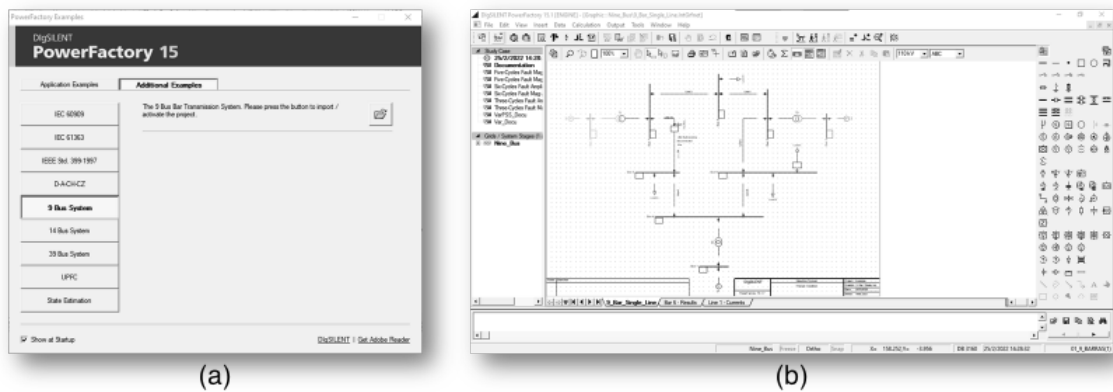


Figura 2.21: Importación del modelo de prueba de 9 barras en PowerFactory. (a) Ventana de importación de ejemplos. (b) Caso activado.

Para la inclusión de la planta fotovoltaica se ha considerado añadir más potencia de generación en el lado de la barra 3. En esta barra está conectado un generador de 85 MW. La barra 3 tiene un voltaje de 13.8 kV y el sistema fotovoltaico tiene un voltaje de 270 V. Por este motivo, se ha escogido añadir a esta barra un transformador elevador de 270 V a 13.8 kV. Esto implica que se añade una barra de 270 V llamada PVB_{us}. A esta barra, se le conectará una carga de tipo general PVL_{oad}. En esta carga de tipo general se conectarán los datos OPC de potencia activa, potencia reactiva, mientras que a la barra se enlazarán los datos de voltaje de línea y frecuencia de línea. Las modificaciones realizadas se pueden apreciar de mejor manera en la Fig. 2.22.

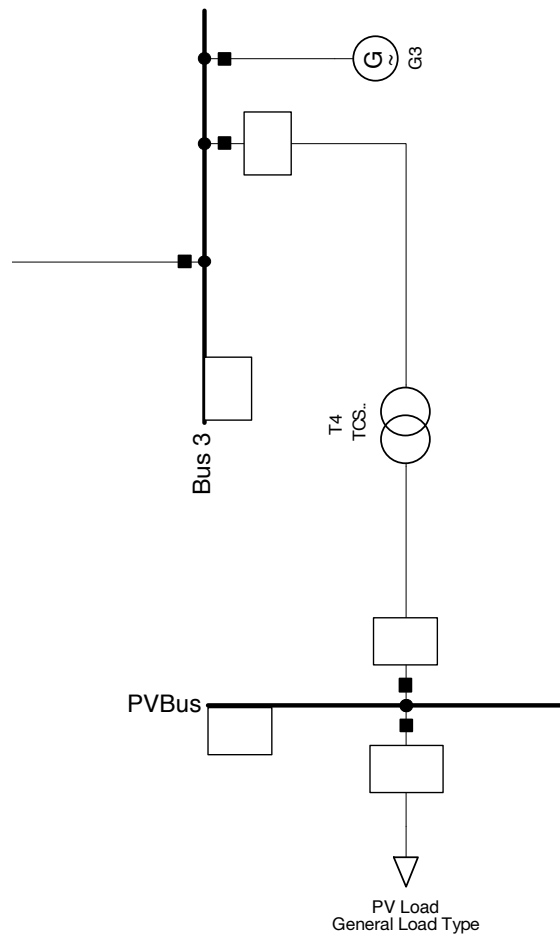


Figura 2.22: Modificación realizada a la barra 3 para la inclusión de la planta fotovoltaica en el SEP.

En PowerFactory, no basta con dibujar un elemento de un diagrama unifilar para que el sistema funcione. Es preciso definir el modelo de cada elemento. Esto se realiza en la librería particular definida por el usuario en cada proyecto. Así pues, se han tomado algunos parámetros del transformador presente en el proyecto Salinas de Imbabura. Es importante mencionar que, en el proyecto original son dos plantas fotovoltaicas, cada una conectada a un transformador de 3 devanados que proporciona la conexión con el SNI. En este caso, se ha considerado la presencia de una planta fotovoltaica con un transformador de 2 devanados, lo que brindará simplicidad al caso de estudio planteado. Los parámetros creados para el modelo de transformador usado se pueden apreciar en la Tabla 2.8. Para el modelo para de la carga PVLoad se ha tomado el modelo de carga predeterminado en PowerFactory para cualquier carga general.

Tabla 2.8: Parámetros del modelo de transformador creado para la modificación del SEP.

| Parámetro | Valor | Unidad |
|--------------------|-------------|--------|
| Nombre | TCS-1000-SC | |
| Potencia Nominal | 3,3 | MVA |
| Frecuencia Nominal | 50 | Hz |
| Voltaje primario | 13,8 | kV |
| Voltaje secundario | 0,27 | kV |
| Grupo vectorial | Dy11 | |

En el análisis de flujos de potencia de PowerFactory, la convención de signos establece que las potencias positivas son correspondientes a las cargas, mientras que las potencias negativas provienen de la generación. Así pues, el truco está en ingresar las potencias OPC predeterminando un signo negativo para inyectar potencia activa o reactiva al sistema. Si las potencias OPC fueran positivas, esto indicaría que el panel fotovoltaico consume potencia, lo cual puede suceder en esta clase de sistemas: se comportan como carga, lo cual es un comportamiento no deseado susceptible de ser simulado para evaluarlo. Por el contrario, si las potencias OPC son negativas, se indica que el panel fotovoltaico está produciendo potencia, el cual es un comportamiento deseable y esperado.

2.6. SERVIDOR Y CLIENTES OPC

El intercambio de datos se realiza mediante el protocolo OPC. Esto significa que se debe realizar la configuración del servidor OPC y de los clientes OPC. En esta sección, se realizará la explicación de la metodología utilizada para determinar las etiquetas que intercambiarán datos, y su configuración en Simulink además de su configuración en PowerFactory. El programa que se usará para este fin es *Matrikon OPC Server for Simulation*. Este programa puede ser descargado en la página web oficial del fabricante (<https://www.matrikonopc.com/downloads/178/drivers/index.aspx>). Al ser un servidor OPC diseñado para simulaciones, es libre para su uso personal y no comercial. Este programa funciona en la plataforma Windows.

2.6.1. MATRIKON OPC SERVER FOR SIMULATION

El uso del protocolo OPC exige determinar con antelación la cantidad de etiquetas o *tags* a utilizarse, además del tipo de datos que se compartirán en cada etiqueta. Para configurar

estas etiquetas, es necesario comprender que las mismas deben agruparse en una entidad conocida como *Grupo*, el mismo que funciona como una carpeta dentro de la cual, se podrán administrar, acceder o modificar los datos almacenados en las etiquetas. En la Fig. 2.23, se puede observar el planteamiento inicial de la configuración de la compartición de datos entre el servidor OPC y los clientes OPC. Posterior a este planteamiento, es necesario establecer los detalles.

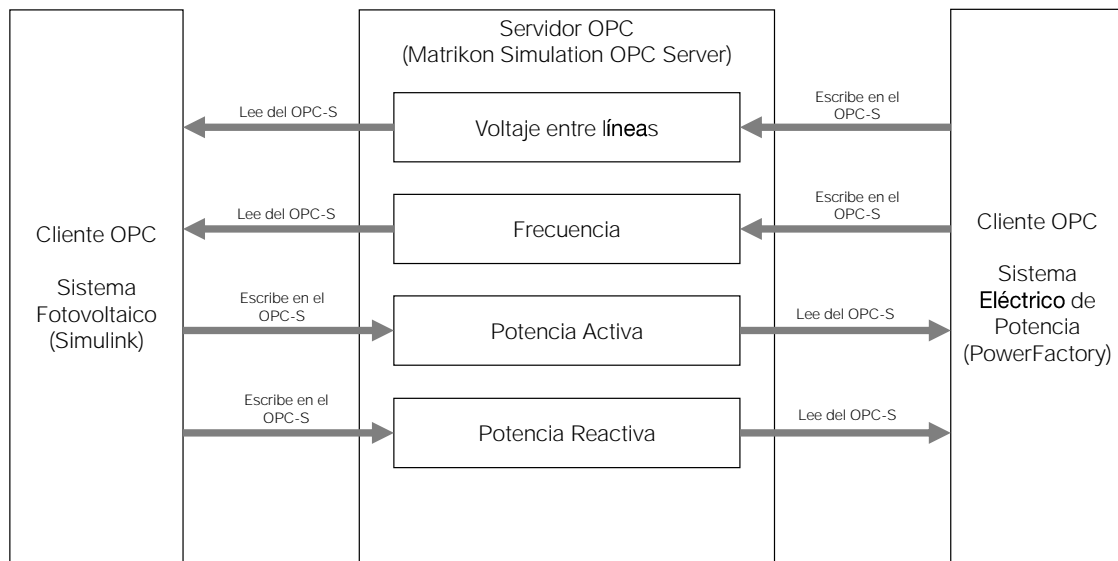


Figura 2.23: Diagrama de compartición de datos entre el servidor y el cliente OPC

Para crear una nueva configuración del grupo OPC y sus variables, se procede de la siguiente manera:

1. Se inicia el programa *Matrikon OPC Server for Simulation and Testing*
2. Se presiona la secuencia de teclas `Ctrl+N`, o en su defecto, en la barra de menú, en la opción `File`, se selecciona la opción `New`. Esto permite crear una nueva configuración en blanco. El panel derecho tendrá un color gris.
3. En el panel izquierdo, aparecen dos opciones: `Server Configuration`, y la opción `Alias Configuration`. Se selecciona la segunda opción. Al dar clic sobre ella, se habilitará el panel derecho con color blanco, y se mostrará un panel de lista donde se pueden añadir o eliminar elementos.
4. Se presiona la secuencia de teclas `Ctrl+A`, o se busca una carpeta de color verde para crear un nuevo grupo OPC. Al hacerlo, en el panel izquierdo, bajo `Alias`

`Configuration`, se habilitará una carpeta en la que se puede modificar el nombre del grupo OPC al nombre que se considere conveniente. Es preferible no usar espacios.

5. Se podrá apreciar que, el panel de lista provee una tabla organizada por nombre, dirección de item, tipo de datos, etc. Dado que, se está creando una nueva configuración, deberá aparecer una línea de lista resaltada. Se puede presionar la secuencia de teclas `Ins` o dar doble clic en la franja para añadir una nueva tag. Al hacerlo, aparecerá un cuadro de diálogo.
6. El cuadro de diálogo que aparecerá, tiene el título `Insert New Alias`, aparecerá la pestaña `Type` de forma predeterminada. En el campo de texto `Name`, se digita el nombre de la etiqueta previamente seleccionada.
7. Se puede apreciar que existen tres tipos de alias: `Basic Alias`, `Advanced Calculation Alias` y `Holding Register Alias`. Se selecciona la opción correspondiente a `Holding Register Alias`, y se escoge el tipo de dato. Existen algunos tipos como: `short`, `long`, `real4`, `real8`, `string`, `boolean`. Se debe escoger el tipo de dato más adecuado para la aplicación. Por ejemplo: un interruptor podría usar un tipo booleano en vez de un tipo real. Sin embargo, si se desea leer un dato de voltaje, probablemente un tipo de dato real sea mejor que un tipo de data string.
8. Se puede configurar de forma opcional el tiempo de refresco para el servidor OPC. Esto se puede realizar en la pestaña `Properties` en la opción `Update Rate`. La opción predeterminada es 0 milisegundos, lo que corresponde al tiempo más rápido posible. También se puede configurar la etiqueta para que sea una etiqueta de solo lectura. Una vez que se ha concluido con esta configuración, se puede dar clic en `Save` para guardar y salir del cuadro de diálogo, o `Save and Create New` para guardar y continuar con la creación de una nueva etiqueta.
9. Una vez que se han configurado todas las tags del grupo, es importante exportar la configuración en un archivo CSV. Para hacerlo, en la barra de menú, en la opción `File`, se escoge la opción `Export aliases to CSV`. Se desplegará un cuadro de diálogo que permite explorar en los directorios del computador local y dar el nombre para guardar el archivo CSV.
10. Si se tiene un archivo de configuración CSV guardado, se puede omitir el proceso de creación de un nuevo archivo de configuración. Para el efecto, en la barra de menú, en

la opción File, se escoge la opción Import aliases from CSV. Se busca el archivo de configuración válido, y una vez que ha sido cargado, no es necesario realizar más pasos, quedando el servidor en funcionamiento.

Con el método anteriormente descrito, se pueden crear las *tags* para la simulación. Es importante que se mantengan los nombres y configuraciones indicadas en la Tabla 2.9. Es importante mencionar que el programa *Matrikon OPC Server* debe estar abierto todo el tiempo que dura la simulación. Al cerrarse, automáticamente el servidor OPC deja de funcionar. Un esquema gráfico puede apreciarse en la Fig. 2.24.

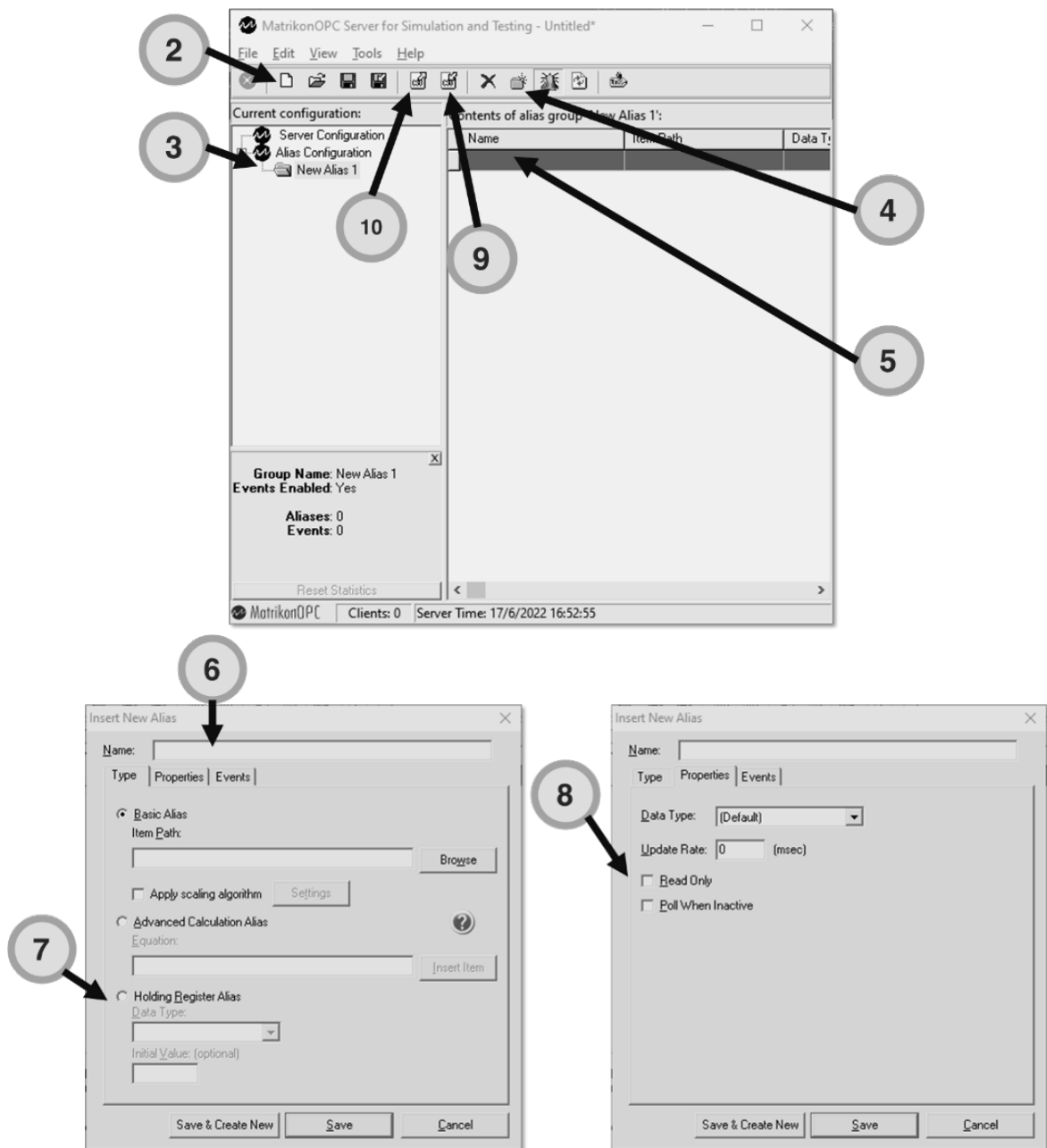


Figura 2.24: Primera secuencia de creación de configuraciones OPC en Matrikon OPC.

Tabla 2.9: Detalle de configuración en los tags creados en *Matrikon OPC Server* para la simulación propuesta.

| Grupo | Nombre | Item Path | Data Type | Descripción |
|--------------|---------------|------------------|-----------|---|
| PVSimulation | ActivePower | Holding Register | Real 8 | Intercambio de datos de potencia activa en kilovatios |
| PVSimulation | ReactivePower | Holding Register | Real 8 | Intercambio de datos de potencia reactiva en kilovatios |
| PVSimulation | LineFrequency | Holding Register | Real 8 | Frecuencia de la línea en milihercios |
| PVSimulation | LineVoltage | Holding Register | Real 8 | Voltaje de la línea en kilovoltios |
| PVSimulation | SimulinkTime | Holding Register | Real 8 | Tiempo de simulación en Simulink en segundos. |

2.6.2. MONITOREO EN *MATRIKON OPC EXPLORER*

Puede ser de mucha utilidad evaluar el estado de las etiquetas. Para esta finalidad se puede utilizar el programa de monitoreo *Matrikon OPC Explorer*. Este programa es incluido en el instalador del servidor OPC. Al iniciar el programa, se debe esperar a que el explorador haga conexión con el servidor OPC. Para el efecto, se asume que la inicialización del servidor efectuada por el programa *Matrikon OPC Server for Simulation* ha sido realizada con anterioridad, y el programa está ejecutándose. Los pasos para crear una nueva sesión de monitoreo son los siguientes.

1. Iniciar el programa mencionado anteriormente. Existirán dos paneles: el panel izquierdo muestra todos los dispositivos que pueden enlazar comunicación OPC, y el panel derecho establecerá las acciones realizables por dispositivo. Dicho esto, en el panel izquierdo localizar el equipo local (Localhost), y dentro de él, activar el servidor `Matrikon.OPC.Simulation.1`.
2. En el panel derecho, se habilitará la opción de conexión y configuración del servidor OPC. Dar clic en `Connect`.
3. En el panel derecho, dar clic en `Add Tags` para añadir las tags que se van a monitorear

con este programa.

4. Se desplegará una nueva ventana de selección de etiquetas. En el panel, se puede buscar la sección llamada `Available Items in Server`. Existen dos carpetas: `Simulation Items`, `Configured Aliases`. Dentro de la última carpeta, estará una subcarpeta con el nombre del grupo que ha sido configurado de acuerdo con la tabla 2.9.
5. Abajo, en el panel llamado `Available Tags`, se podrán ver todas las etiquetas configuradas de acuerdo con la tabla 2.9. Dar clic derecho y seleccionar la opción `Add All Items`.
6. En la sección superior lateral derecha, habrá una opción llamada `Tags to be added`. Verificar que se han añadido todas las etiquetas.
7. Si las etiquetas se añadieron correctamente, se puede dar clic en `OK`.
8. Se podrá apreciar una ventana en la que se enlistan las etiquetas y existirá una columna con el valor de la misma. Al inicializar desde cero, el valor de todas las etiquetas será cero. Sin embargo, el valor puede ser diferente si este proceso se hace sin limpiar la sesión.
9. Se puede dar doble clic en cualquier etiqueta para desplegar el cuadro de diálogo `Write Values`, donde se puede examinar con más detalle los valores del tag, así como se pueden escribir nuevos valores en las etiquetas a modo de prueba. En la práctica, los programas que podrán escribir valores son `Simulink` o `PowerFactory`, siendo `Matrikon OPC Explorer` una especie de programa de monitoreo en modo sólo lectura. Los pasos se condensan en la Fig. 2.25.

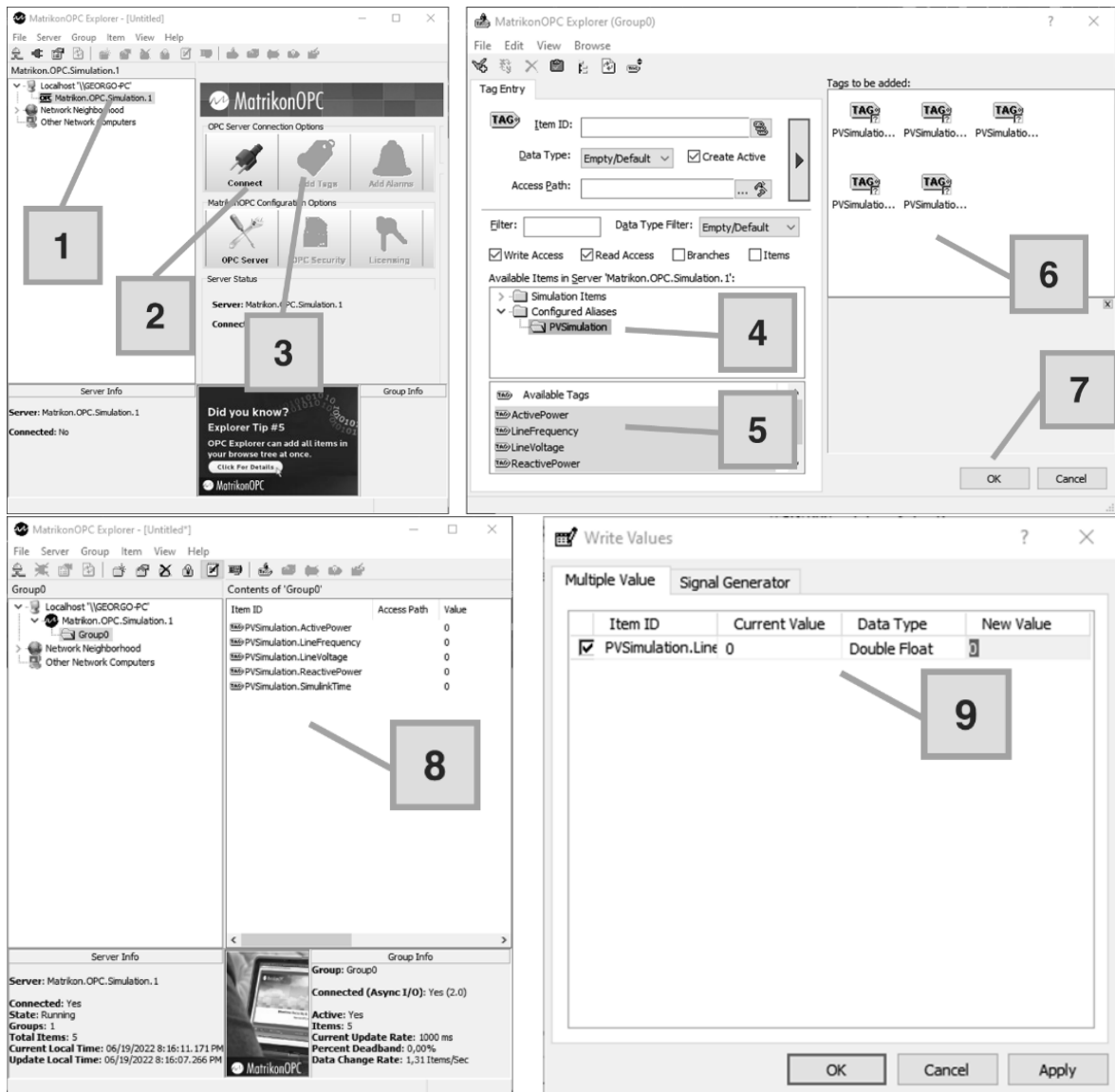


Figura 2.25: Secuencia de monitoreo de tags en Matrikon OPC Explorer.

2.6.3. CONFIGURACIÓN DEL CLIENTE OPC EN SIMULINK

Simulink provee un entorno bastante sencillo para comunicarse con un servidor OPC mediante el *OPC Toolbox*. En cualquier modelo de Simulink se pueden realizar los siguientes pasos:

1. En la librería de modelos de Simulink, buscar la categoría *OPC Toolbox*.
2. Arrastrar al modelo el bloque *OPC Configuration*
3. Arrastrar al modelo el bloque *OPC Read*
4. Arrastrar al modelo el bloque *OPC Write*

Hay que mencionar que no todas las instalaciones de Matlab-Simulink tienen acceso a la conectividad OPC. El apareamiento o no de esta característica, está limitada por la licencia usada en la instalación. En caso que esta característica no esté disponible, hay que realizar las gestiones necesarias para obtenerla.

2.6.3.1. Bloque OPC Config

Este bloque debe ser el primero en ser configurado para el funcionamiento de la simulación posterior. Los pasos que se deberían seguir son los siguientes:

1. Dar doble clic en el bloque de configuración OPC. Se desplegará una ventana en la que se debe dar clic en la opción *Configure OPC Clients*.
2. Se despliega una ventana para administrar el cliente OPC que se puede añadir para la simulación. Dar clic en *Add...*
3. A continuación se muestra una ventana de propiedades del servidor OPC. Después de un tiempo de refresco, se podrá dar clic en la opción *Select...*
4. Se desplegará una ventana de configuración para la selección del servidor. Seleccionar el correspondiente a *Matrikon OPC*. Pueden existir otros servidores OPC (*Kep-Server, OpenOPC, Geomation, etc*), sin embargo, en este trabajo, se está usando *Matrikon OPC* para la simulación.

En la primera ventana se pueden configurar las acciones a tomar en caso que las tags no estén disponibles, hayan errores de lectura-escritura, el servidor no esté disponible, hayan errores de simulación en tiempo pseudoreal. Es importante mencionar que se debe deseleccionar la opción de simulación en tiempo pseudoreal, puesto que consume muchos más recursos que una simulación normal, además que los horizontes de simulación en este proyecto son muy pequeños (segundos) y la potencia computacional disponible no podría asegurar el cumplimiento de esos parámetros. En la Fig. 2.26 se puede apreciar una condensación de los procedimientos anteriormente mencionados.

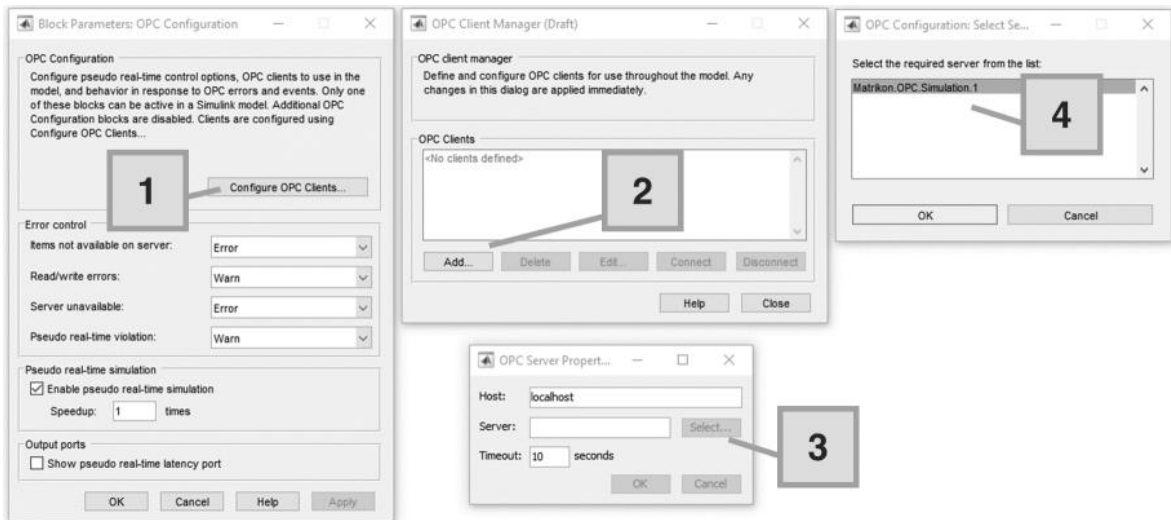


Figura 2.26: Esquema gráfico de configuración del bloque *OPC Config*.

2.6.3.2. Bloque OPC Read

En este bloque se configuran los ítems que se leen desde el servidor OPC, y provienen de un proveedor de datos externo (PLC, co-simulación, etcétera). Para el efecto, se pueden seguir los siguientes pasos:

1. Dar doble clic en el bloque *OPC Read*. Se desplegará una ventana de propiedades de bloque. Buscar el botón *Add Items...* para añadir tags que pueden leerse.
2. Se desplegará una ventana para seleccionar las tags que serán leídas por Simulink. En el panel correspondiente a *Available server items* se pueden encontrar los ítems configurados en la Tabla 2.9 en la opción *Configured Aliases*. Dentro de esta opción, se encuentra el grupo de tags, y dentro de él, las tags configuradas.
3. Así pues, se procede a escoger las etiquetas de lectura, las correspondientes al voltaje de bus y a la frecuencia de bus.
4. Se da clic en *OK* una vez que se han escogido las tags de lectura.
5. Se puede escoger el modo de lectura: existe el modo síncrono y el asíncrono. También existe la posibilidad de escoger el tiempo de muestreo para la actualización de datos de lectura, y se puede configurar el tipo de dato que se puede obtener a partir del bloque.

En esta simulación, se deshabilitará la opción de mostrar el puerto de calidad y el puerto de marca de tiempo. Estos últimos datos no son relevantes en el proceso. Los pasos ante-

riormente mencionados, se pueden condensar en la Fig. 2.27. Al tener más de una etiqueta, pero solamente un puerto de salida, es imperativo el uso de bloques de demultiplexación para poder trabajar con cada tag por separado.

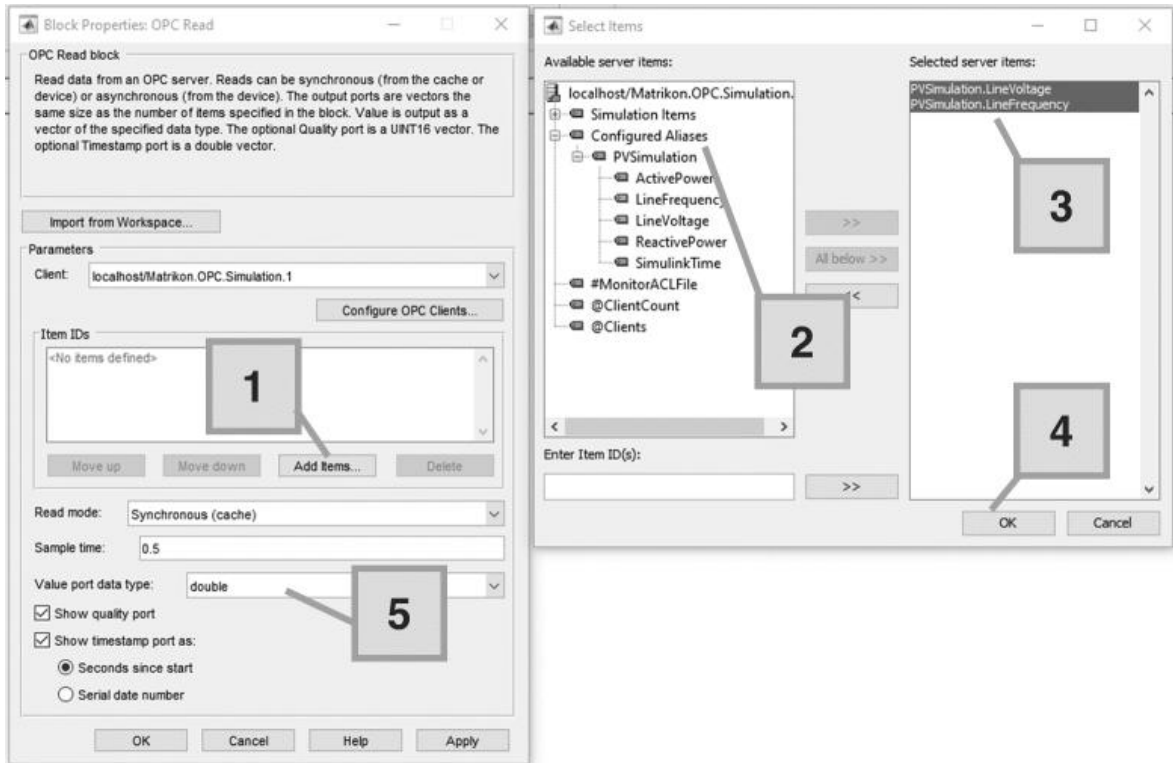


Figura 2.27: Esquema gráfico de configuración del bloque *OPC Read*.

2.6.3.3. Bloque OPC Write

En este bloque se configuran las tags que pueden ser modificadas por Simulink y podrán ser utilizadas por otras aplicaciones externas. Los pasos para realizar este procedimiento se detallan a continuación:

1. Dar doble clic en el bloque *OPC Write*. Se desplegará una ventana de propiedades de bloque. Buscar el botón *Add Items...* con la finalidad de buscar las tags que serán sujetas a modificación.
2. Se desplegará una ventana de selección de ítems. De forma similar a la configuración del bloque de lectura, se busca dentro del panel de ítems disponibles el grupo de tags y las tags de escritura. Se seleccionan las tags necesarias correspondientes a Potencia Activa, Potencia Reactiva y Tiempo de Simulación de Simulink.
3. Una vez que se han seleccionado las tags, se da clic en *OK*.

4. En la ventana de propiedades de bloque, se puede escoger el modo de escritura: síncrona o asíncrona. Además el tiempo de muestreo para la escritura. El valor pre-determinado es cero.

Los pasos anteriormente mencionados, se pueden condensar en la Fig. 2.28. Al tener más de una etiqueta, pero solamente un puerto de entrada, es imperativo el uso de bloques de multiplexación para poder trabajar con cada tag por separado.

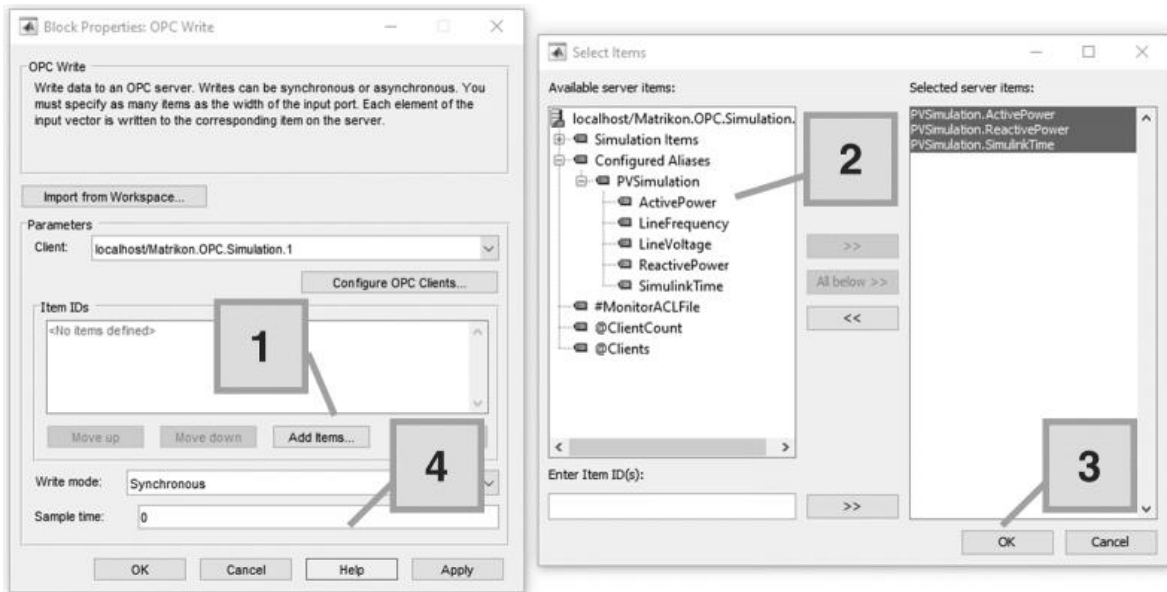


Figura 2.28: Esquema gráfico de configuración del bloque *OPC Write*.

2.6.4. CONFIGURACIÓN DEL CLIENTE OPC EN POWERFACTORY

Para que exista una comunicación entre PowerFactory y el servidor OPC, se debe configurar el primer programa para que pueda ser capaz de admitir características de co-simulación. En este proyecto se han seguido las instrucciones detalladas en [90]. Sin embargo, se resumirán los pasos más importantes.

2.6.4.1. Configuraciones Iniciales

1. Se debe permitir que PowerFactory se ejecute en modo núcleo y en modo multi-hilo. Para ese efecto, se puede ingresar a la pantalla de configuración de inicio de sesión. En los contenidos de la configuración avanzada, se vuelve a seleccionar la pestaña *Advanced*. Posteriormente, se habilitan mediante el cuadro de validación, las opciones anteriormente mencionadas.

2. Es imprescindible el asegurar la existencia de un objeto de comunicación *.ComLink que permita la utilización del protocolo OPC. Para este efecto, con un proyecto activado, se despliega el Data Manager de PowerFactory. Dentro del proyecto, en la sección de casos de estudio, se selecciona el caso de interés. Debe existir un objeto External Data Link. Dentro de este objeto, hay que verificar que la comunicación OPC sea del modo TDS (*Time-Domain Simulation*, Simulación en el dominio del tiempo), que el nombre del computador sea localhost para indicar que es el computador local, y que el identificador del programa sea el correspondiente a Matrikon OPC Simulation. Es importante mencionar que, la característica de comunicación OPC no está disponible en todas las licencias de PowerFactory, y debe ser solicitada o instalada explícitamente. Un esquema gráfico puede apreciarse en la Fig. 2.29.

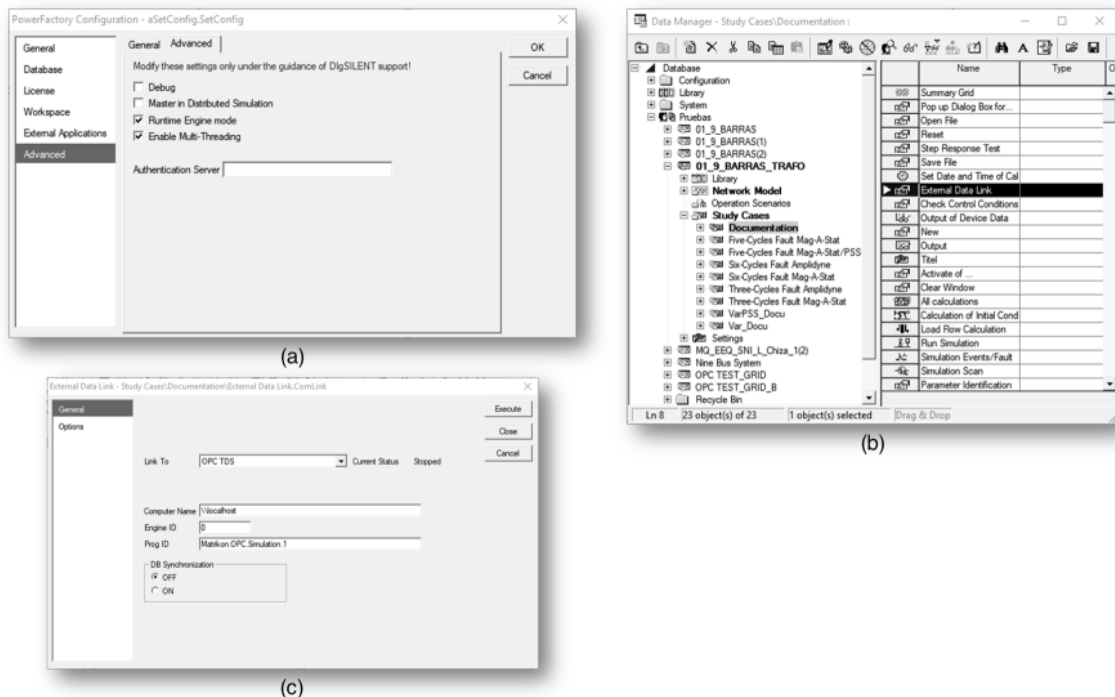


Figura 2.29: Configuraciones básicas en PowerFactory para habilitar co-simulación y protocolo OPC. (a) Habilitación de ejecución modo núcleo y multihilo. (b) Verificación de la existencia del objeto de comunicación OPC. (c) Verificación de los parámetros del objeto de comunicación OPC.

2.6.4.2. Bloques DSL

Los bloques DSL (*DigSilent Simulation Language*) son imprescindibles para la lectura de datos OPC provenientes de fuentes externas, y su escritura en los valores internos de

simulación de PowerFactory. El bloque DSL está compuesto de 3 partes interconectadas entre sí:

1. Marco compuesto o *Composite Frame*: Este marco define la interacción entre diferentes señales. En este caso, la interacción es realizada entre los objetos de comunicación OPC externos y la carga. Existen entonces dos bloques en este marco compuesto, el primero, que se refiere a las señales `ElmDs1` y el segundo, que se refiere a la carga `ElmLod`.
2. Modelo común: La fuente de la señal de la carga necesita inicializarse mediante este modelo.
3. Modelo compuesto o *Composite Model*: El marco compuesto es usado para implementar el modelo compuesto que contiene el modelo común. De esta forma, la señal proveniente del protocolo OPC es inyectado en el sistema eléctrico de potencia.

Generalmente, el marco compuesto y el modelo común se almacenan en la librería de modelos definidos por el usuario propia de cada proyecto activado. Estos recursos se utilizan después para implementar el modelo compuesto DSL que se almacena como parte de los objetos utilizables por el sistema eléctrico de potencia. En este proyecto, dentro de la carpeta `NODE 1` de los modelos definidos por el usuario, se almacenan las plantillas del marco compuesto `Load Frame` y el modelo común `Load Signal Source`. En el diagrama unifilar `Nine Bus` se puede apreciar el objeto `LOAD X` que contiene el bloque DSL utilizando el modelo común de la librería de modelos definidos por el usuario. PowerFactory no es un software precisamente intuitivo. Por este motivo, en la Fig. 2.30 se pueden apreciar con más detalle, las características que se han mencionado en el párrafo anterior.

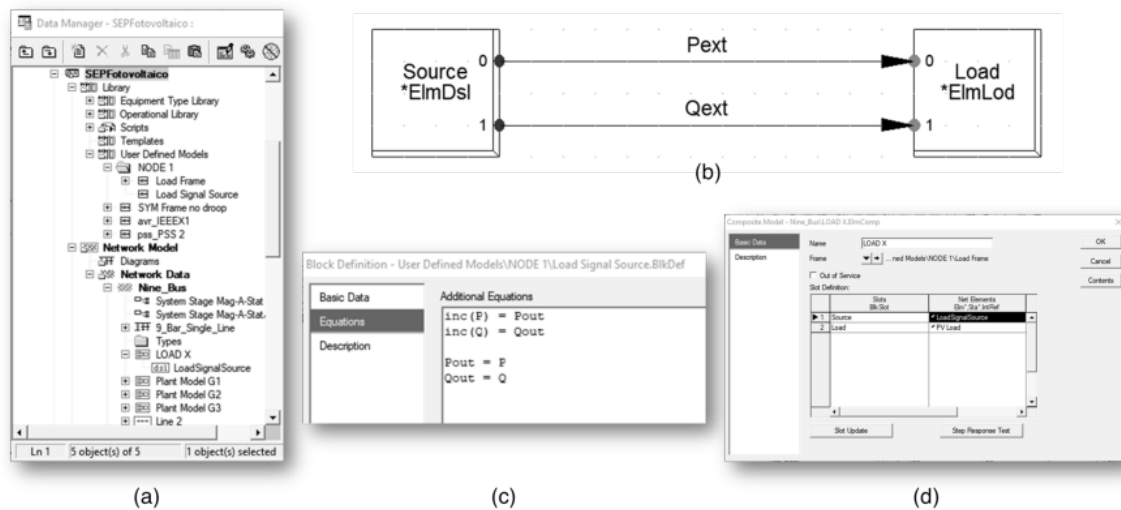


Figura 2.30: Configuración de bloques DSL para PowerFactory. (a) Distribución de los marcos y modelos en el árbol de proyecto. (b) Marco compuesto (c) Modelo común. (d) Modelo compuesto.

2.6.4.3. Configuración de Tags en PowerFactory

Las propiedades que se modificarán en la simulación son la Potencia Activa y la Potencia Reactiva, correspondientes a la carga PV Load conectada a la barra PVBus, creada como modificación al sistema IEEE 9 barras original. Asimismo, se tomarán de esta barra los datos correspondientes a Voltaje de Línea y Frecuencia.

Para configurar las etiquetas OPC con las variables modificables o sujetas de lectura, se deben crear objetos de medición de datos externos External DAT Measurement en los cubículos de conexión de la carga de interés. Los cubículos se pueden acceder mediante el uso del *Data Manager*, estando dentro de las barras de interés. En este caso, la barra de interés es la PVBus puesto que, a esta barra se conecta la carga PVLoad. Así pues, es cuestión de ubicarse en el *Data Manager* e ir creando nuevos objetos para medición de datos externos. Esto puede apreciarse de mejor manera en la Fig. 2.31.

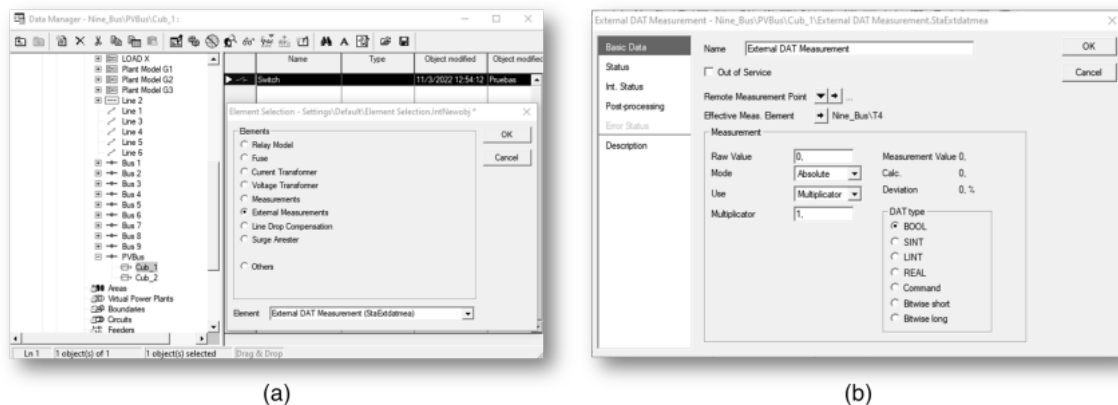


Figura 2.31: Configuración de objetos *External DAT Measurement* para la configuración de Tags en PowerFactory. (a) Ubicación de los objetos en el árbol de proyecto. (b) Ventana de configuración de los objetos *.StaExtDatmea.

En total, deben configurarse 4 objetos de medición de datos externos, correspondientes al voltaje de línea, frecuencia de la red, potencia activa y reactiva. Cada objeto de medición de datos externos ha sido configurado de acuerdo con los parámetros que se condensan en la Tabla 2.10.

Tabla 2.10: Configuración de las tags asociadas a los parámetros de lectura y escritura en PowerFactory.

| | | Frecuencia | Voltaje | Potencia Activa | Potencia Reac-tiva |
|-----------------|-------------------|---------------|---------------|-----------------|--------------------|
| Basic Data | Name | FE_OPC | VLL_OPC | P_OPC | Q_OPC |
| | Raw Value | 60 | 0 | 0 | 0 |
| | Mode | Absolute | Absolute | Absolute | Absolute |
| | Use | Multiplicator | Multiplicator | Multiplicator | Multiplicator |
| | Multiplicator | 0,001 | 0,001 | 1 | 1 |
| | DAT Type | REAL | REAL | REAL | REAL |
| Status | Read | | | X | X |
| | Write | X | X | | |
| Post-processing | Controlled object | | | PVLoad | PVLoad |
| | Variable Name | | | plini | qlini |
| | Controller | | | DSL | DSL |
| | Variable Name | | | P | Q |
| | Object | PVBus | PVBus | PVLoad | PVLoad |
| | Variable Name | m:fehZ | m:UI | plini | qlini |
| | Sim Object | PVBus | PVBus | PVLoad | PVLoad |
| | Variable Name | m:fehZ | m:UI | plini | qlini |
| Description | TagID | PVSimulation | PVSimulation | PVSimulation | PVSimulation |
| | | LineFrequency | LineVoltage | ActivePower | ReactivePower |

2.6.4.4. Configuración de parámetros de simulación en tiempo real

Para mejorar la sincronización temporal y disminuir la deriva de tiempo entre los dos programas de simulación, se deben realizar algunas configuraciones adicionales en PowerFactory. Simulink es el programa que presenta una carga computacional más alta, puesto que implementa todas las simulaciones dinámicas que son transmitidas a PowerFactory. Esto implica que, en PowerFactory y no en Simulink se deben configurar los parámetros idóneos para disminuir la deriva. Se puede definir la deriva como la diferencia temporal entre el tiempo de simulación entre diferentes elementos de una co-simulación paralela entre dos o más programas. Esta deriva no es cero, pero puede minimizarse.

En la Fig. 2.32 se pueden apreciar los parámetros configurables en PowerFactory. A continuación, se explicarán brevemente las consideraciones a tomar en cuenta para la configuración.

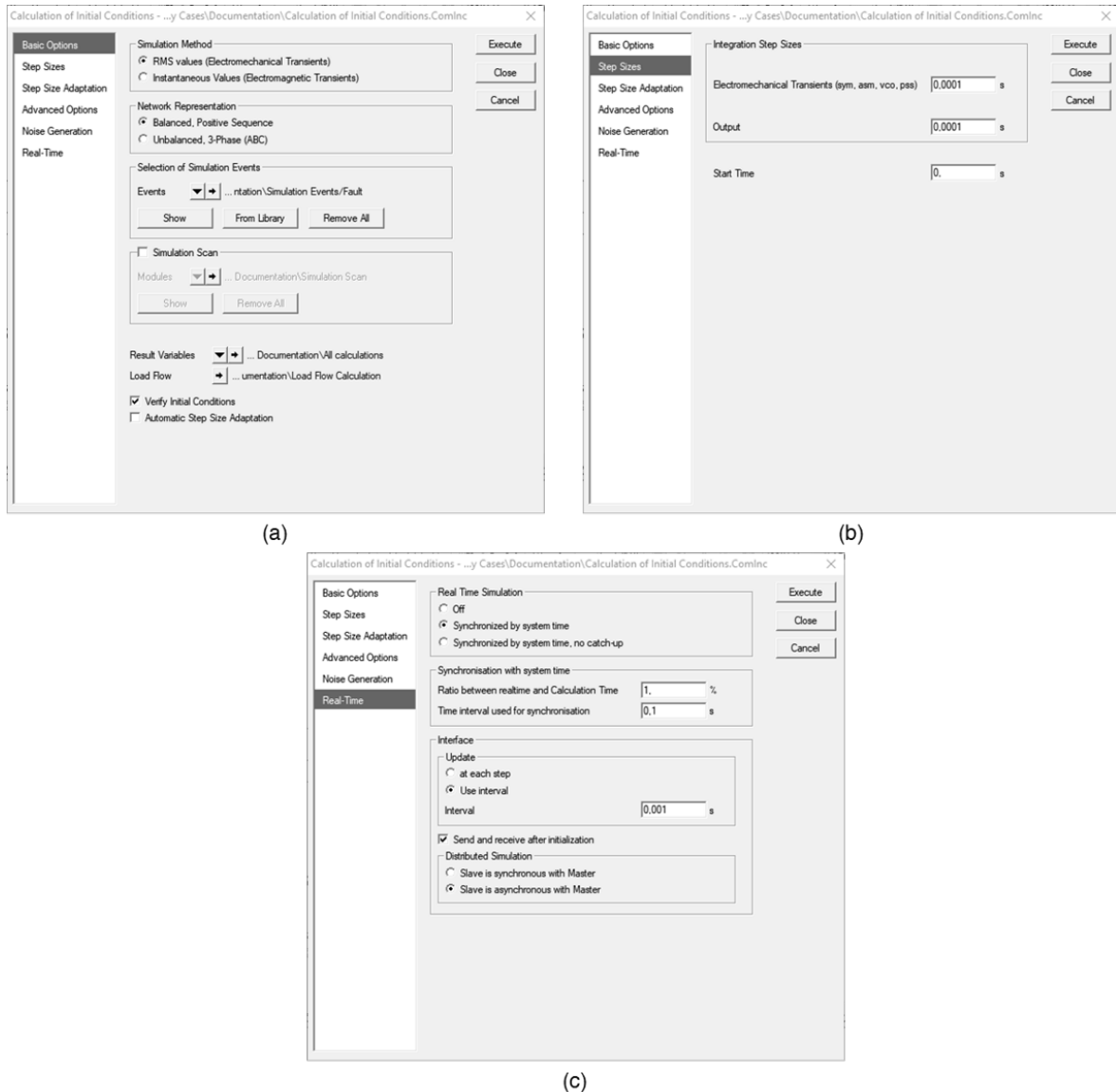


Figura 2.32: Configuración de parámetros para simulación en PowerFactory. (a) Opciones Básicas. (b) Tamaño de pasos de simulación. (c) Parámetros de tiempo real.

1. Opciones básicas: En este panel, la simulación dinámica considera una simulación RMS. Este proyecto no considera una simulación EMT (*Electromagnetic Transients*).
2. Tamaño de pasos de simulación: Estos parámetros deben tener concordancia con los tiempos usados en Simulink. Pueden usarse tamaños de paso de simulación razonablemente mayores para elevar la rapidez de cálculo en la simulación.
3. Tiempo real: En este panel se debe escoger una sincronización por reloj de sistema. La tasa entre el tiempo real y el tiempo de simulación debe ser parametrizado. Valores mayores a uno proveen una simulación dilatada en el tiempo, y valores menores a uno

proveen una simulación acelerada.

4. Actualización de valores: Esta configuración está presente en el panel de Tiempo Real. Se sugiere usar la opción de *Use Interval*. Este valor es el que se usará para que PowerFactory actualice los valores desde el OPC. Sin embargo, al utilizar la opción *At each step* se puede asegurar que existirá una lectura-escritura de datos OPC redundante.
5. La comunicación será asíncrona entre el servidor OPC y PowerFactory. La opción de inicializar la simulación enviando y recibiendo información al inicio de la misma debe estar activada.

2.6.5. MEJORA DE LA SINCRONIZACIÓN ENTRE PROGRAMAS

Para sincronizar de la mejor manera, se deben ejecutar algunas simulaciones en Simulink, puesto que, es el software que más recursos consume. Se deben registrar los tiempos reales de simulación con el horizonte de 500 milisegundos, con OPC incluido. Finalmente, se debe calcular el promedio para asignar un valor de compensación a PowerFactory, el cual debe ser calculado de la siguiente manera:

$$f_c = \frac{\text{Tiempo del horizonte de simulación}}{\text{Tiempo promediado de simulación real en Simulink}} \times 100 \quad (2.12)$$

Se recomienda simular sin conexión OPC para los casos de estudio unas cuantas veces y obtener un valor promedio. Por ende, la sincronización posterior al disparo paralelo puede ser diferente, y la relación puede variar también con las condiciones del computador usado.

2.7. SINCRONIZACIÓN MEDIANTE PYTHON

PowerFactory provee un instalador de Python. Aunque se dispongan de varias versiones de este lenguaje de programación, se recomienda que la simulación sea implementada con la versión proporcionada por el fabricante. A la fecha del desarrollo de este proyecto, la versión es algo antigua (Python 3.0), sin embargo, funciona de forma adecuada para los propósitos planteados. Los pormenores del API de Python para PowerFactory pueden consultarse en la documentación técnica que provee DigSilent.

En la Fig. 2.1 se puede apreciar que el elemento sincronizador entre Matlab-Simulink y PowerFactory está en la generación de un archivo de texto escrito por el primero y leído por la API de Python-PowerFactory con la finalidad de iniciar simultáneamente la simulación

RMS en el software de simulación de Sistemas Eléctricos de Potencia. En esta sección, se describirán los mecanismos implementados con esta finalidad.

2.7.1. SEÑALES DE DISPARO EN SIMULINK

En la Fig. 2.33 se puede apreciar el bloque de Simulink implementado con la finalidad de sincronizar la simulación entre ambos programas, junto con su parametrización. El código puede apreciarse en el Apéndice A.9.



Figura 2.33: Esquema de la generación de la señal de disparo de co-simulación en Simulink. (a) Vista general. (b) Configuración de la señal u . (c) Configuración de la señal w .

2.7.2. FLUJO DE PROGRAMA EN PYTHON

En la Fig. 2.34 se puede apreciar el Diagrama de Flujo para el código de Python implementado con la finalidad de sincronizar la simulación entre ambos programas. El código puede apreciarse en el Apéndice A.10. El Script es replicado del trabajo realizado en [63].

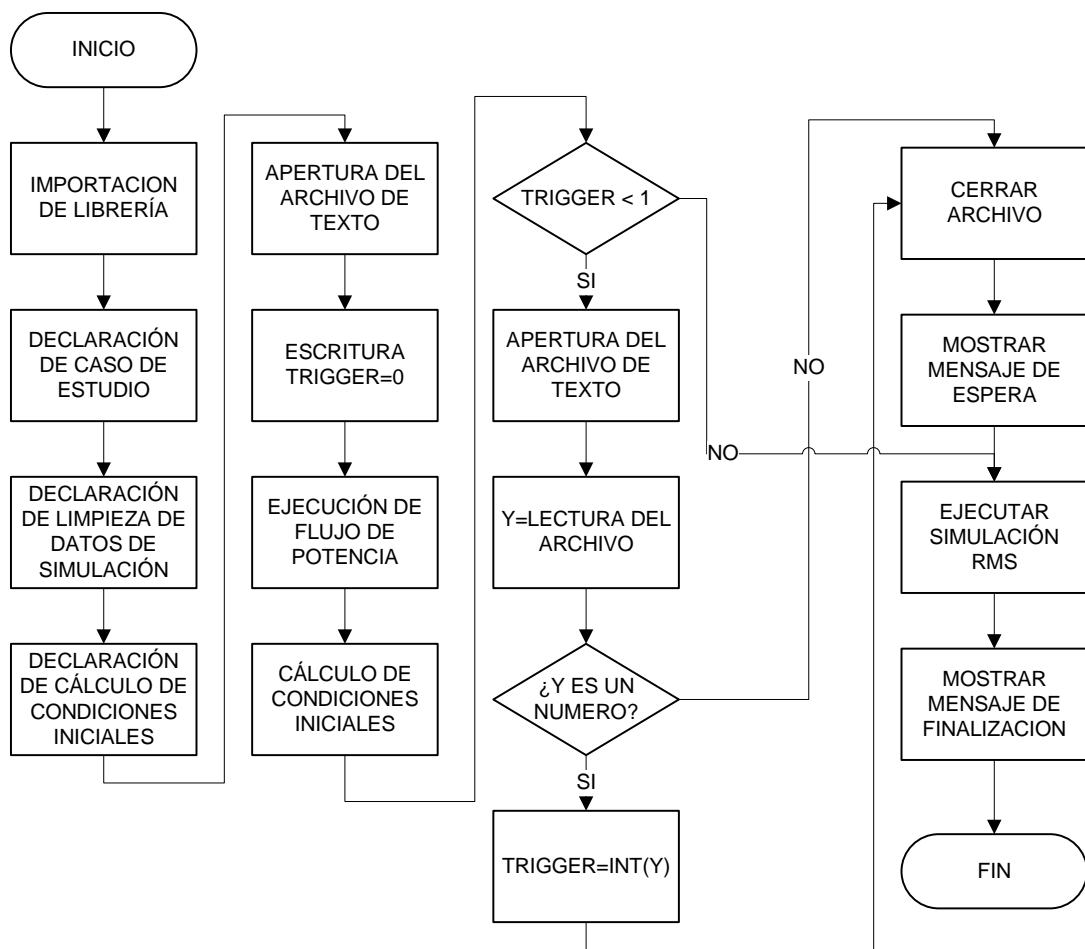
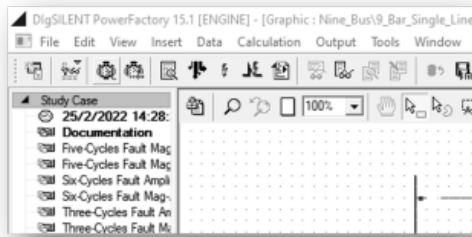


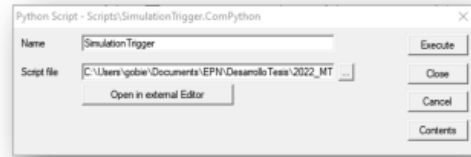
Figura 2.34: Diagrama de Flujo del código de Python usado para la sincronización

2.7.3. CONEXIÓN DEL SCRIPT EN POWERFACTORY

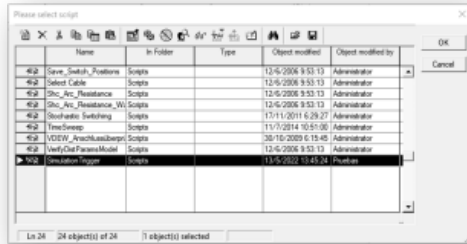
Si bien el script se escribe en Python, este se ejecuta bajo la API de PowerFactory. Esto significa que es necesaria una configuración adicional para que esta interfaz pueda saber cuál programa ejecutar. El proceso se detalla en la Fig. 2.35. Primero es necesario ubicar el contenedor de script de interés en el *Data Manager* de PowerFactory. Posteriormente, se debe configurar su nombre y la ubicación del archivo deseado.



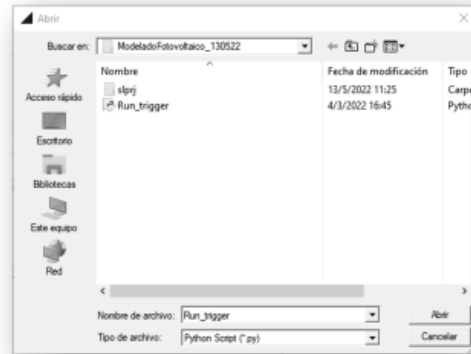
(a)



(c)



(b)



(d)

Figura 2.35: Esquema de configuración del script de Python en PowerFactory. (a) Ubicación en la barra de iconos. (b) Selección del contenedor del Script. (c) Cuadro de diálogo de ruta y nombre. (d) Selección del archivo.

3. RESULTADOS

En este capítulo se presentarán los resultados obtenidos en las pruebas de comunicación unidireccionales (Matrikon-Simulink, Matrikon-PowerFactory), pruebas de comunicación bidireccionales (Simulink-PowerFactory). También se exponen los datos obtenidos en las simulaciones realizadas usando Simulink solamente (estos datos servirán como referencia para la validación posterior), así como los obtenidos en la co-simulación. Para estos dos últimos casos, se proponen como referencias de comparación: la integral absoluta del error, la integral cuadrática del error para los lazos de control del convertidor elevador, voltaje de bus DC, corriente de eje directo, corriente de eje de cuadratura, obtenidos desde Simulink. También se miden los valores promedio de la potencia de la planta fotovoltaica, potencia activa, potencia reactiva y voltaje de bus en Simulink, además de presentar los valores reflejados en el tiempo referentes al voltaje de barra, frecuencia de barra, potencia activa, reactiva, y cargabilidad del transformador en PowerFactory.

Para detallar mejor la procedencia de los resultados, en la Fig. 3.1 se aprecian las variables: I_{pv} , V_{pv} como corriente y voltaje de la planta fotovoltaica, V_{bus} el voltaje de bus DC del convertidor DC/DC, I_{i1} , I_{i2} , I_{i3} son las corrientes del inversor, I_{g1} , I_{g2} , I_{g3} son las corrientes conectadas al punto común de conexión, V_a , V_b , V_c son los voltajes de línea en el punto de conexión común, donde se miden voltaje de bus y frecuencia de la red. Las potencias activa y reactiva se calculan mediante estos voltajes y corrientes trifásicas usando las transformaciones de coordenadas explicadas en el Capítulo 1. La cargabilidad del transformador se obtiene directamente en PowerFactory.

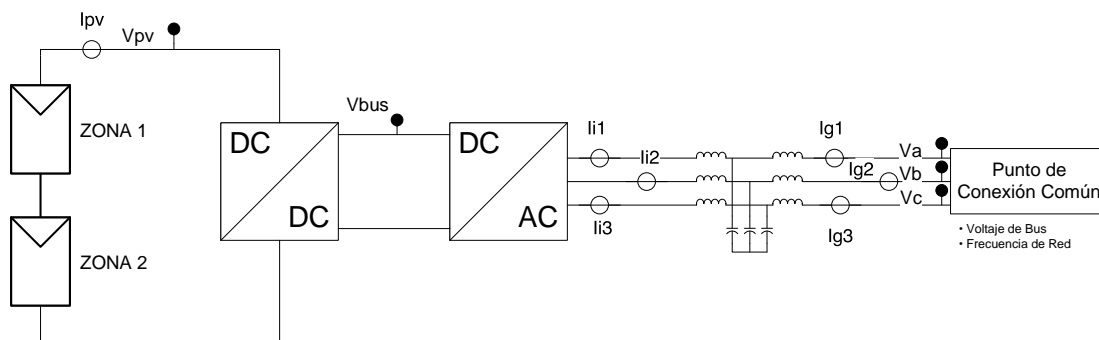


Figura 3.1: Ubicación de las variables medidas en Simulink.

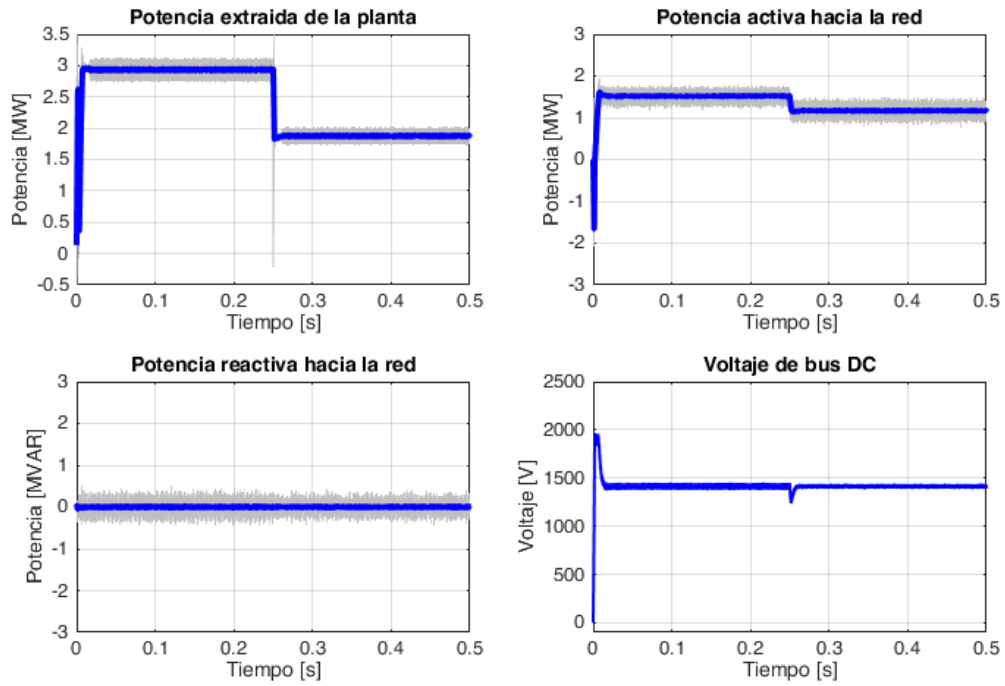
3.1. SIMULACIÓN SIN CONEXIÓN CON POWERFACTORY

Con el propósito de disponer datos para validar la co-simulación, se ha realizado una simulación *stand-alone* de la planta fotovoltaica solamente con el software provisto por Matlab. El horizonte de simulación es de 500 milisegundos. La irradiancia inicial se estableció en 804 W/m^2 y a la mitad del tiempo de simulación, su valor disminuye a 518.42 W/m^2 en virtud de los datos provistos en [83]. Se busca que el voltaje DC permanezca estable en $1000\sqrt{2}$ voltios, y que la planta fotovoltaica provea siempre la máxima potencia disponible. El convertidor elevador ha sido programado como se describe en la metodología. Se implementa el filtro pasivo LCL. Se realizaron experimentos considerando 4 casos.

1. MPPT Perturbar-Observar sin sombras parciales.
2. MPPT Conductancia Incremental sin sombras parciales.
3. MPPT Perturbar-Observar con sombras parciales.
4. MPPT Conductancia Incremental con sombras parciales.

Cuando se usan sombras parciales, se asume una reducción de potencia en la zona 2, del 50%. Para comparar la respuesta de los controladores implementados, se monitorea la integral del valor absoluto del error (IAE), y la integral cuadrática del error (ISE). Las respuestas en el dominio temporal pueden apreciarse en las Figs. 3.2, 3.3. Los resultados inherentes a los índices de rendimiento se pueden observar en la Tabla 3.1. Se han calculado también los valores promedio de potencia fotovoltaica entregada, potencia activa, potencia reactiva y voltaje de bus DC mediante una integración discreta de los valores y su división por el tiempo de simulación. Los resultados pueden apreciarse en la Tabla 3.2.

(Perturbar & Observar - Ideal)



(Perturbar & Observar - Sombras)

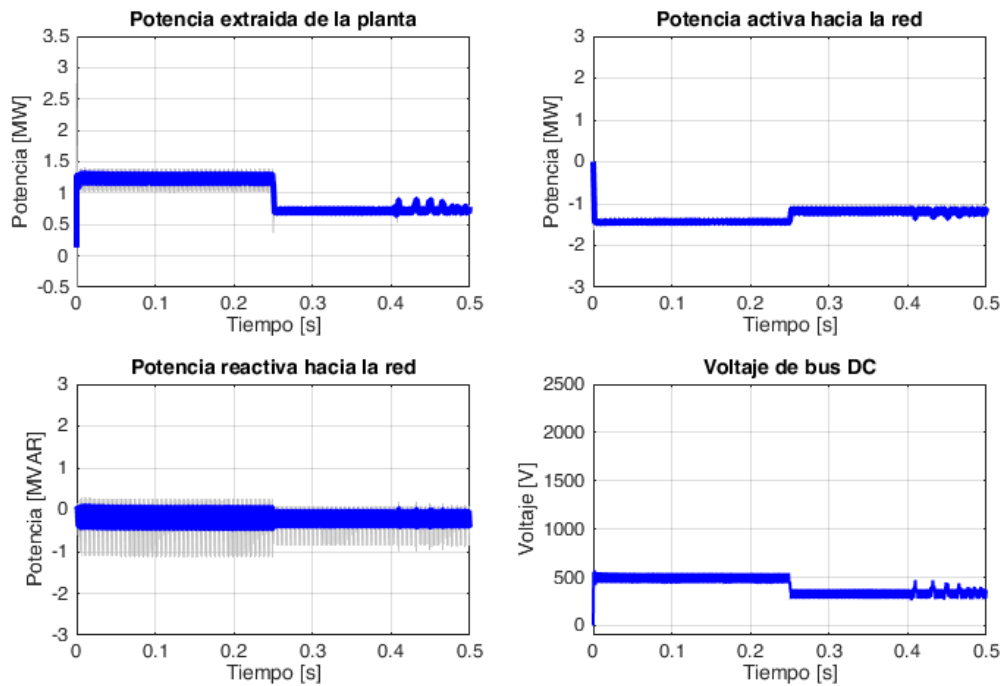
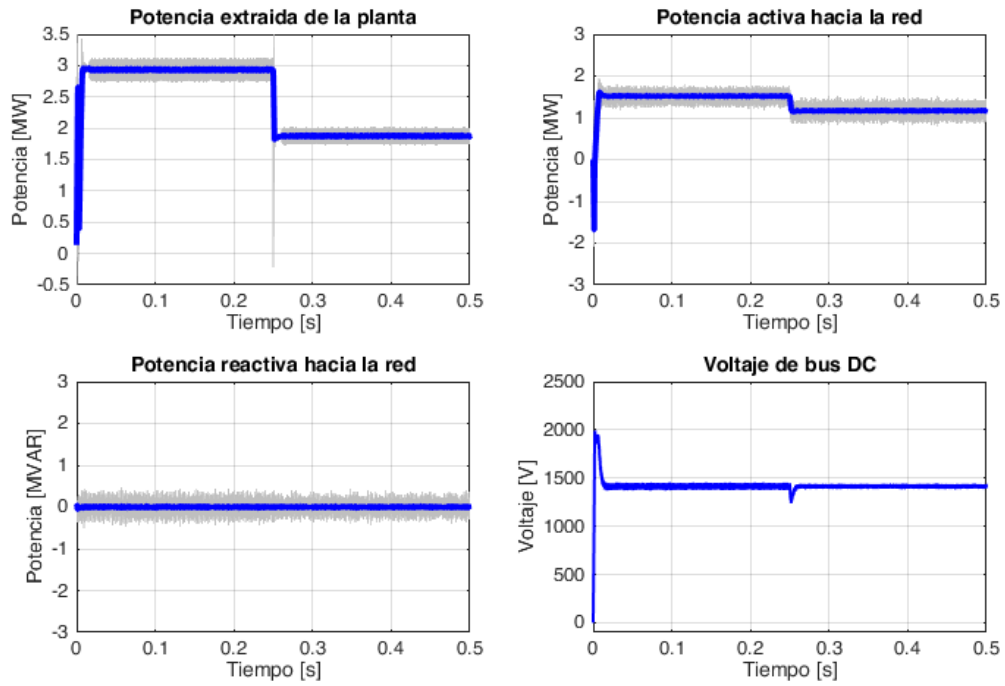


Figura 3.2: Resultados en el dominio del tiempo para la simulación sin comunicación OPC, en condiciones ideales y con sombras parciales con el método perturbar-observar. En azul, la respuesta filtrada. En gris, la respuesta sin filtrar.

(Conductancia Incremental - Ideal)



(Conductancia Incremental - Sombras)

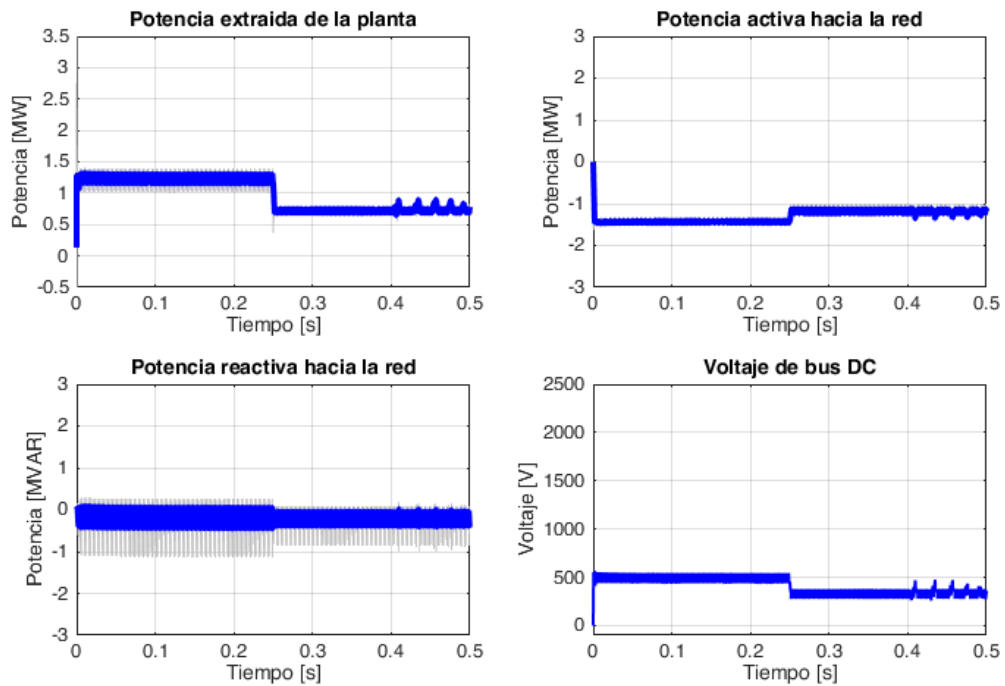


Figura 3.3: Resultados en el dominio del tiempo para la simulación sin comunicación OPC, en condiciones ideales y con sombras parciales con el método de conductancia incremental. En azul, la respuesta filtrada. En gris, la respuesta sin filtrar.

Tabla 3.1: Resultados de los indicadores de rendimiento de control (ISE, IAE) para los cuatro casos de estudio.

| ISE | PO - Ideal | CI - Ideal | PO - Sombras | CI - Sombras |
|-----------------|------------|------------|--------------|--------------|
| Convertor DC/DC | 363,2 | 372 | 2,28E+04 | 2,34E+04 |
| Voltaje de Bus | 2637 | 2679 | 5,06E+05 | 5,07E+05 |
| Corriente Id | 7978,00 | 8003 | 1,03E+10 | 4,03E+10 |
| Corriente Iq | 101,3 | 98,87 | 1,37E+05 | 1,37E+05 |

| IAE | PO - Ideal | CI - Ideal | PO - Sombras | CI - Sombras |
|-----------------|------------|------------|--------------|--------------|
| Convertor DC/DC | 2,828 | 2,836 | 96,52 | 98,98 |
| Voltaje de Bus | 10,56 | 10,6 | 501,9 | 502 |
| Corriente Id | 21,66 | 21,69 | 1,22E+05 | 1,22E+05 |
| Corriente Iq | 5,566 | 5,502 | 234,5 | 234,5 |

Tabla 3.2: Resultados de potencia de la planta, potencia activa, reactiva y voltaje de bus para los casos de estudio planteados.

| | PO - Ideal | CI - Ideal | PO - Sombras | CI - Sombras |
|--------------------------|------------|------------|--------------|--------------|
| Potencia PV [MW] | 2,389 | 2,39 | 0,972 | 0,9713 |
| Potencia activa [MW] | 1,329 | 1,329 | -1,315 | -1,314 |
| Potencia reactiva [MVAR] | -3,90E-05 | -9,95E-06 | -0,2655 | -0,2655 |
| Voltaje de Bus [kV] | 1,418 | 1,418 | 0,4104 | 0,4102 |

3.1.1. ANÁLISIS DE RESULTADOS

De acuerdo con los resultados mostrados en la Tabla 3.1, se puede apreciar que en todos los casos, el convertidor elevador tiene un buen rendimiento, mientras que el control de las variables de voltaje de bus, corriente de eje directo y corriente de cuadratura dependen bastante de la existencia o no de la condición de sombras parciales. En los casos correspondientes a las condiciones de sombra parcial, se puede apreciar un notable deterioro del seguimiento de la referencia (notable especialmente respecto al voltaje de bus DC, que debería mantenerse en $1000\sqrt{2}$ voltios). Esto es fácilmente apreciable en la Fig. 3.4, en la que se analiza la integral cuadrática del error. La ISE de la corriente de eje directo, la cual es responsable de la potencia activa es considerablemente alto en comparación a sus contrapartes. En consecuencia, cuando existen sombras parciales, el convertidor no puede

alcanzar la referencia adecuadamente.

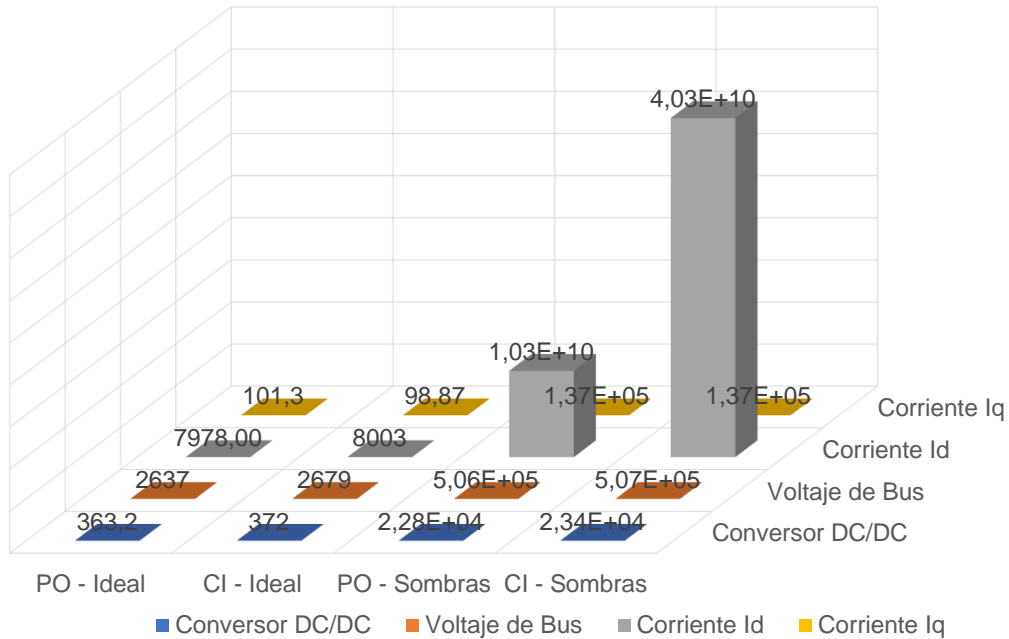


Figura 3.4: Comparación de la integral cuadrática del error (ISE) entre los cuatro casos de estudio .

Al examinar la Tabla 3.2, también se puede apreciar un efecto interesante: cuando existen sombras parciales, el comportamiento generador de la planta fotovoltaica es totalmente anulado. En Simunlink, se ha establecido una convención de signos para que la potencia con signo positivo implique generación de energía y la potencia con signo negativo implique absorción de energía. Por lo tanto, cuando existen sombras parciales se puede apreciar que el controlador no puede alcanzar la referencia de voltaje de bus y por consiguiente, empieza a comportarse como carga en vez de hacerlo como generador. Hay una pérdida de generación del 56% cuando se trabaja en condiciones de sombras parciales.

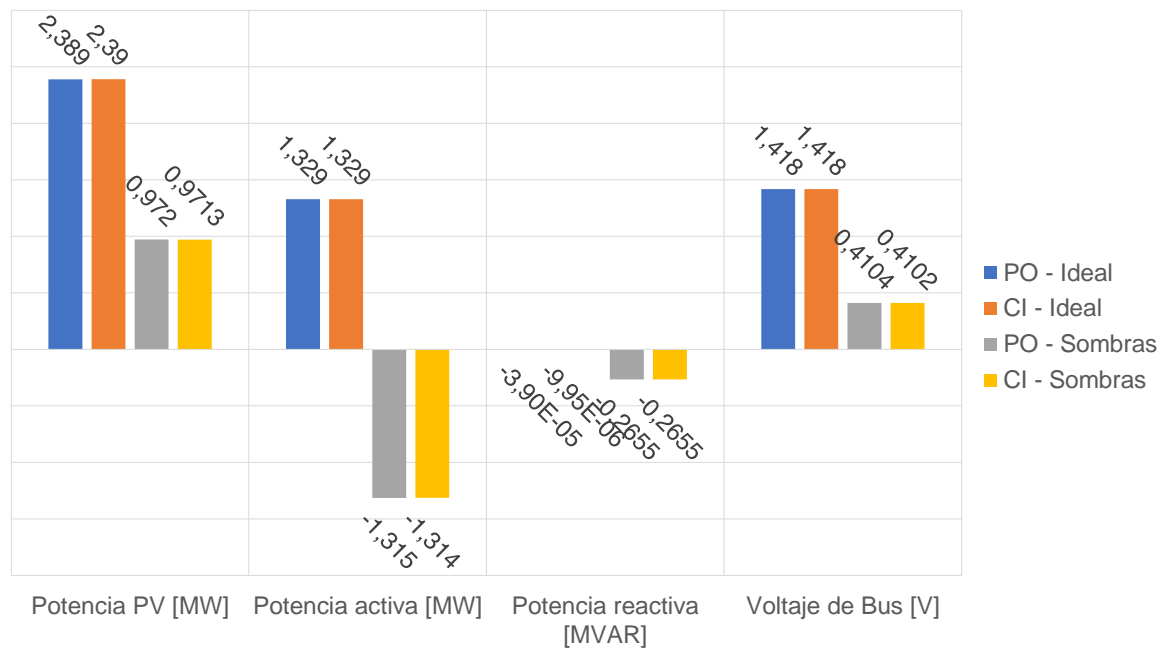


Figura 3.5: Comparación de los resultados de potencia fotovoltaica, activa, reactiva y voltaje de bus DC entre los cuatro casos de estudio .

Hay que mencionar que, si se examina la potencia promedio entregada, aparentemente la planta fotovoltaica no alcanza su potencial máximo de 3,6 MW. Esto no es cierto. La aparente reducción de la potencia de la planta fotovoltaica tiene relación con el comportamiento dinámico de la potencia de la misma, puesto que al inicio de la simulación debe alcanzar la referencia. Durante un brevísimo instante de tiempo la totalidad de la planta fotovoltaica se comporta como una carga para después estabilizarse y proveer la máxima potencia disponible de 3,6 MW en condiciones ideales. Esto puede apreciarse claramente en la Fig. 3.6. Evidentemente, al calcular la potencia fotovoltaica promedio mediante una integración discreta en el tiempo, la potencia promedio se verá disminuida a causa de este comportamiento dinámico, sin disminuir su comportamiento posterior.

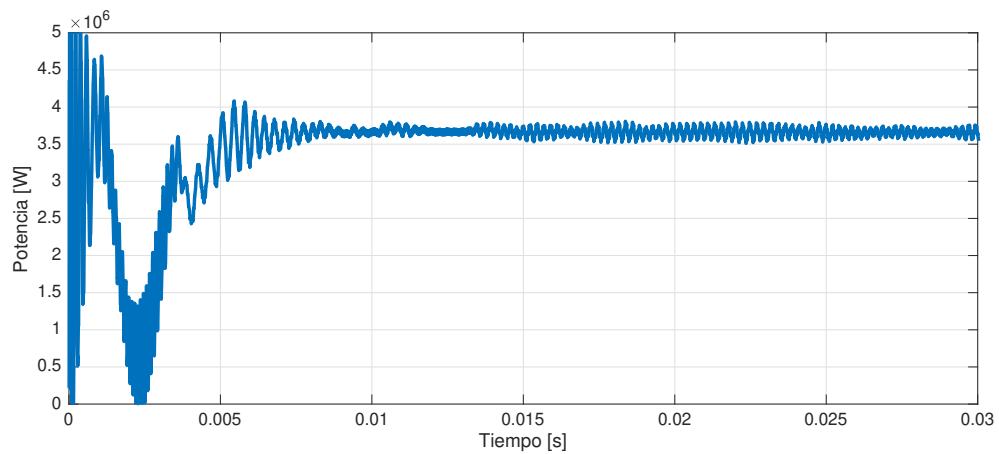


Figura 3.6: Respuesta temporal detallada de la potencia activa de la planta fotovoltaica, extraída sin sombras parciales y con el seguimiento de conductancia incremental, bajo una irradiación de 1000 W por metro cuadrado.

Se puede apreciar además, que no existe una diferencia notable entre las respuestas obtenidas mediante conductancia incremental y el método de perturbación-observación.

3.2. COMUNICACIÓN OPC UNIDIRECCIONAL

3.2.1. COMUNICACIÓN SERVIDOR-SIMULINK

Para la comunicación Servidor-Simulink se ha implementado un modelo sencillo que consiste en un generador de voltaje trifásico que lee voltaje y frecuencia desde *Matrikon OPC Explorer*, y escribe valores de potencia activa, reactiva alimentando una carga trifásica en Y flotante calibrada a 270 V, 60 Hz y un consumo de potencia activa de 1 MW y potencia reactiva de 1 MVAR. Estos valores pueden monitorearse desde el software provisto por Matrikon. El modelo de Simulink `OPCTest.slx`, puede apreciarse en la Fig. 3.7. Se ha utilizado un tiempo de integración de 1 microsegundo, solucionador en tiempo discreto Runge Kutta (`ode4`) y un tiempo de simulación de 1 segundo. Los bloques de lectura y escritura OPC han sido configurados para comunicación asincrónica, con un tiempo de actualización de 0,01 segundos de simulación (no son segundos transcurridos en tiempo real). Este modelo también se usará para probar la comunicación bidireccional.

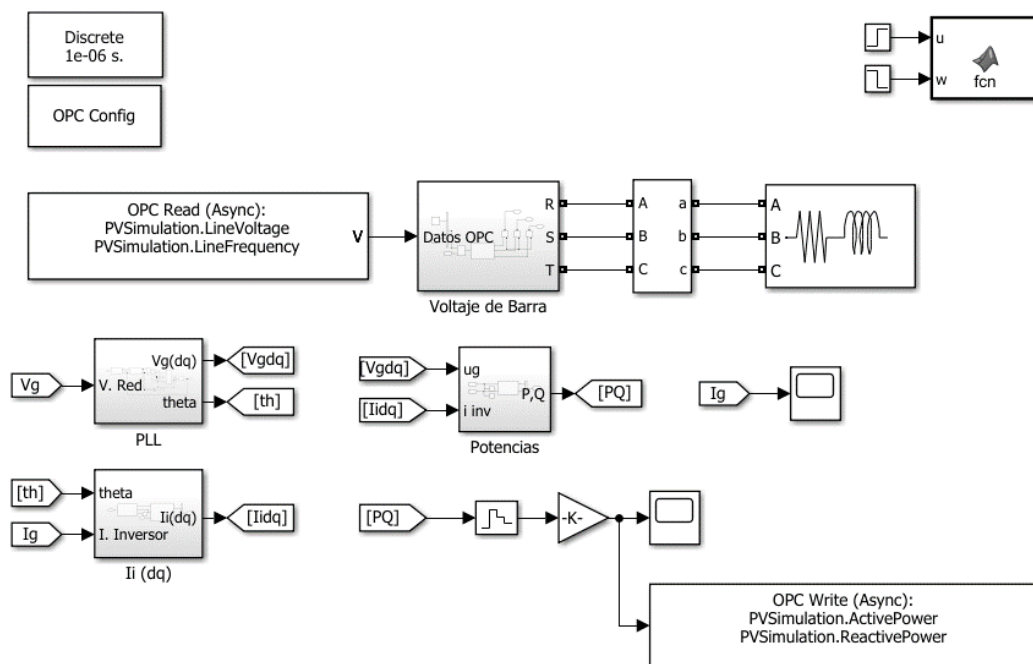


Figura 3.7: Modelo implementado en Simulink para la prueba de comunicación unidireccional.

La prueba se ha realizado escribiendo una frecuencia constante de 60 Hz, y escribiendo un voltaje entre 200 y 300 V con el generador de dientes de sierra que provee *Matrikon OPC Explorer*. Este generador ha sido configurado como se aprecia en la Fig. 3.8, con una

frecuencia de 100 milisegundos. Una vez que se configuran estos parámetros, se da clic en Start para comenzar a enviar los datos en formato de diente de sierra. No se debe cerrar este cuadro de diálogo, y debe quedar en funcionamiento durante toda la simulación. Este proceso se realiza de forma manual y tiene la finalidad de evaluar si existe o no comunicación entre el servidor y Simulink sin ahondar tanto en la exactitud temporal.

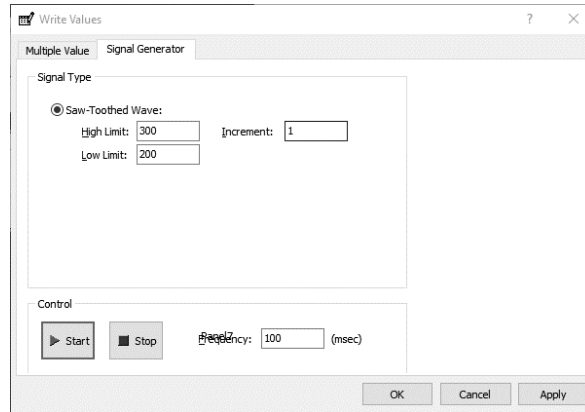


Figura 3.8: Generador de dientes de sierra para la prueba de comunicación unidireccional.

Posterior a la ejecución de la simulación de prueba, se obtienen los resultados apreciados en la Fig. 3.9. Se puede apreciar que, existe una buena reproducción de la forma de diente de sierra. Respecto a la sincronización temporal, es claro que existe una correspondencia diferente entre el tiempo en Simulink y el tiempo del servidor OPC.

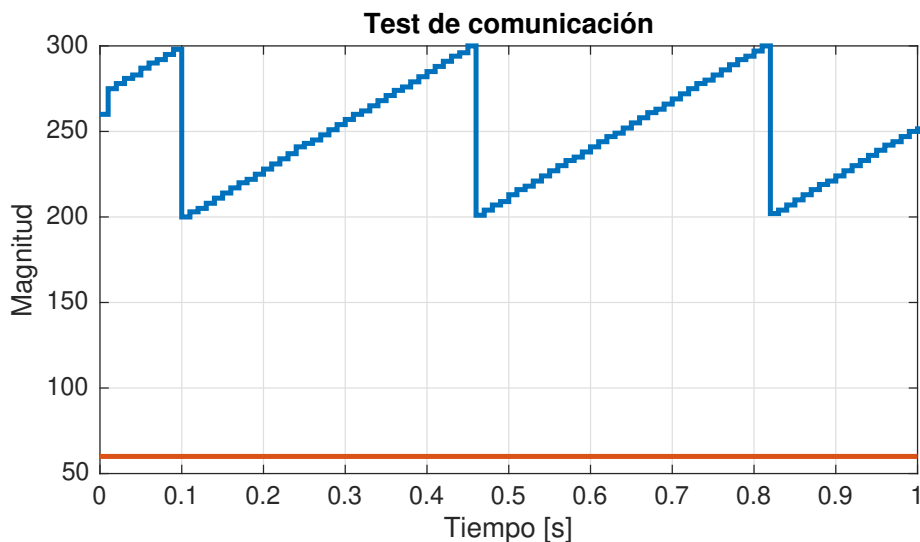


Figura 3.9: Registro de datos de la simulación unidireccional Simulink-Matrikon.

Realizando mediciones, se puede apreciar que existe un periodo de casi 360 milisegundos en la gráfica recibida por Simulink. Eso quiere decir que la relación entre el tiempo

capturado por Simulink y el tiempo teórico producido por el generador de diente de sierra es aproximadamente 3,6. El periodo mínimo de la función periódica que se puede generar con Matrikon es de 100 milisegundos. La comunicación ha sido exitosa, sin embargo, se debe tomar en cuenta la diferencia temporal entre aplicaciones. El script usado para la simulación puede visualizarse en el algoritmo A.11.

3.2.2. COMUNICACIÓN SERVIDOR-POWERFACTORY

Para la comunicación entre estos dos programas, simplemente se escribe la potencia activa con el generador de diente de sierra con periodo de 100 milisegundos, de forma similar a la comunicación realizada entre el servidor y Simulink. Los límites ahora estarán entre 1 y 3 megavatios. PowerFactory ha sido configurado con un tiempo de integración de 0,0005 segundos. La relación entre el tiempo real y el tiempo de cálculo ha sido establecido en 0,1874 por ciento, con un tiempo muerto de 0,001 segundos usados para la sincronización. La actualización se realiza en cada paso de integración de PowerFactory. Se realiza la misma obteniéndose los resultados que se aprecian en la Fig. 3.10. Se han tomado dos muestras consecutivas en los puntos máximos para registrar la diferencia temporal. Estos resultados se aprecian en la Tabla 3.3.

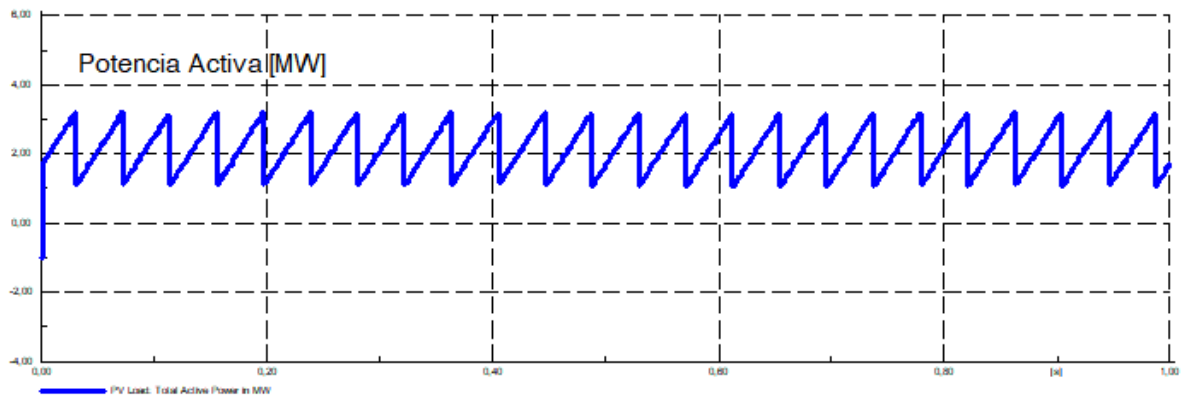


Figura 3.10: Registro de datos de la comunicación unidireccional PowerFactory-Matrikon.

Aunque solamente se tomaron dos muestras de diferencia temporal, se puede advertir que existe un periodo promedio entre estos dos valores: $\frac{0,0419+0,0413}{2} \approx 41,6 \times 10^{-3}$ segundos. Por lo tanto, la relación entre el periodo teórico del servidor OPC y el periodo medido en PowerFactory es aproximadamente $\frac{100 \times 10^{-3}}{41,6 \times 10^{-3}} \approx 2,4038$ veces. Se verifica entonces que la comunicación entre PowerFactory y Matrikon OPC es exitosa, aunque también existe una diferencia temporal que podría calibrarse modificando los parámetros del programa.

Tabla 3.3: Tabla para medición de los tiempos de sincronización OPC en PowerFactory.

| Tiempo de captura [s] | Potencia Activa [MW] | Diferencia Temporal [s] |
|-----------------------|----------------------|-------------------------|
| 0,031 | 3,20 | |
| 0,072 | 3,19 | 0,0419 |
| 0,114 | 3,12 | 0,0413 |

3.3. COMUNICACIÓN OPC BIDIRECCIONAL

Para la prueba de comunicación OPC entre Simulink y PowerFactory, se mantienen los parámetros mencionados en las pruebas de comunicación unidireccional. La diferencia es que en esta oportunidad, se mantienen ambos programas en activo para que puedan comunicarse entre sí.

Se puede apreciar, de acuerdo con los resultados mostrados en las Fig. 3.11 que la comunicación bidireccional es exitosa. Sin embargo, hay que considerar que PowerFactory espera un poco de tiempo para empezar a recibir datos a causa de la ejecución del script de Python. Después de ese tiempo, la forma entre ambas señales es bastante consistente, si bien el dominio temporal del primer programa es diferente al segundo. Esto se da a causa de la calibración en PowerFactory de la relación entre el tiempo real y el tiempo de cálculo, relación porcentual que debe ajustarse de acuerdo con el tiempo real y el tiempo de cálculo de Simulink, que es el programa de ejecución más lenta que condiciona el tiempo total de simulación. Como se está usando un modelo mucho más ligero que el modelo fotovoltaico completo, al no haberse modificado esta relación, resulta comprensible que el dominio temporal de ambas simulaciones no sea el mismo. Sin embargo, esta relación se corrige en las co-simulaciones.

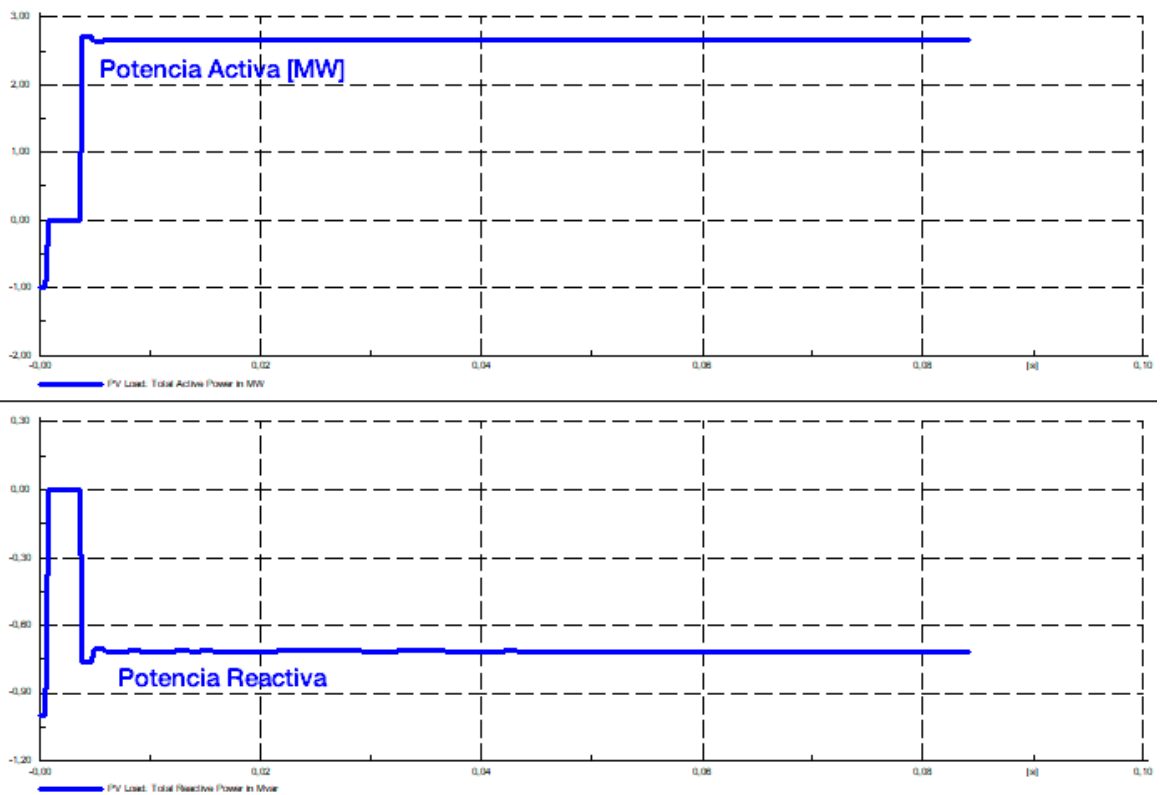
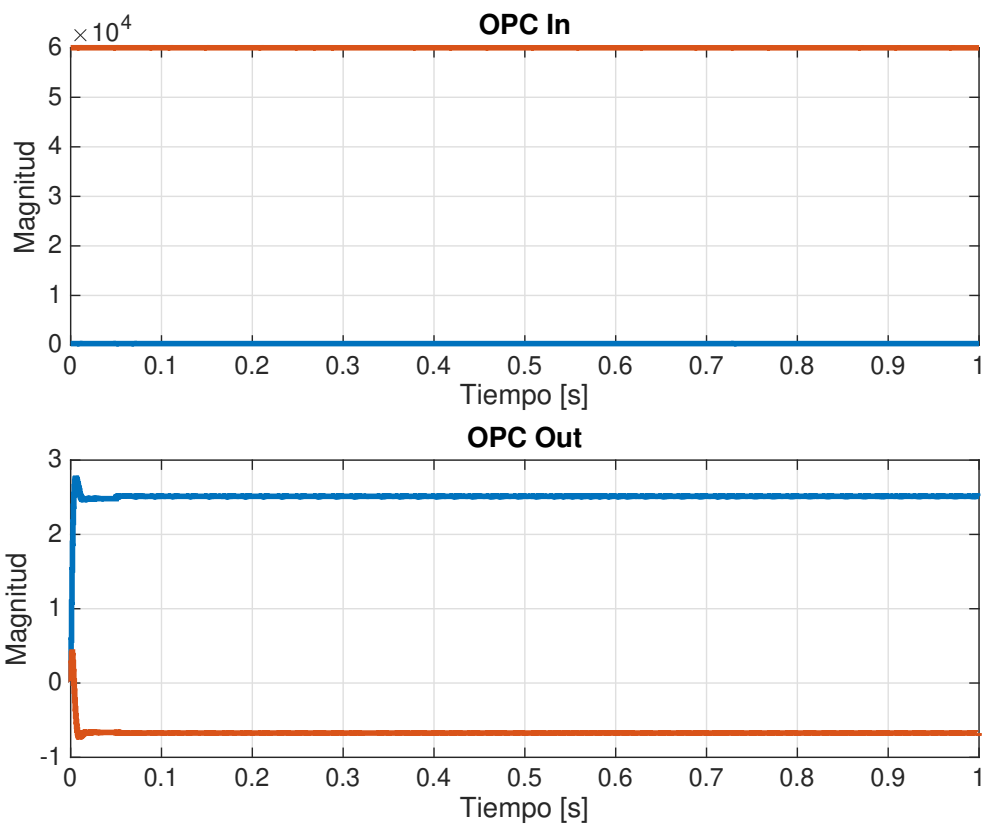


Figura 3.11: Registro de datos de la comunicación bidireccional en Simulink y PowerFactory.

3.4. CO-SIMULACIÓN SIMULINK - POWERFACTORY

3.4.1. PLANTEAMIENTO DE LOS CASOS DE ESTUDIO

De acuerdo con los resultados obtenidos en la sección anterior, se redundará en la escasa diferencia obtenida entre los métodos de conductancia incremental y perturbación-observación. Sin embargo, entre estos dos métodos, el que presenta menor complejidad computacional es el método de perturbación-observación ya que no se basa en el cálculo de derivadas. Por este motivo, para la co-simulación con PowerFactory se considera solamente este método. Sin embargo, se harán dos experimentos considerando condiciones ideales y sombras parciales. En ambos casos, el horizonte de simulación es de 500 milisegundos. En Simulink el tiempo de integración es de 1 microsegundo, en PowerFactory el tiempo de integración es 500 microsegundos.

Las condiciones temporales de irradiancia son las siguientes: en consonancia con los datos provistos en [83], se irradia el panel con 800 vatios por metro cuadrado. Transcurrido el 50 % del tiempo de simulación, la irradiancia baja a 518.42 vatios por metro cuadrado. Con esto se intenta replicar el comportamiento del panel fotovoltaico entre la mayor y menor irradiancia diaria mediante una prueba de perturbación tipo paso.

3.4.2. CASO 1: P & O SIN SOMBRAS PARCIALES

En primera instancia, se presentan las respuestas en PowerFactory para voltaje de barra, frecuencia de la red en la Fig. 3.12. Las respuestas para la potencia activa y reactiva se aprecian en la Fig. 3.13. Las respuestas para la cargabilidad del transformador se aprecian en la Fig. 3.14. También, se pueden apreciar las respuestas gráficas almacenadas en Simulink, en la Fig. 3.15. Se usa una relación de $r = 0,21625$ por ciento para la sincronización en PowerFactory.

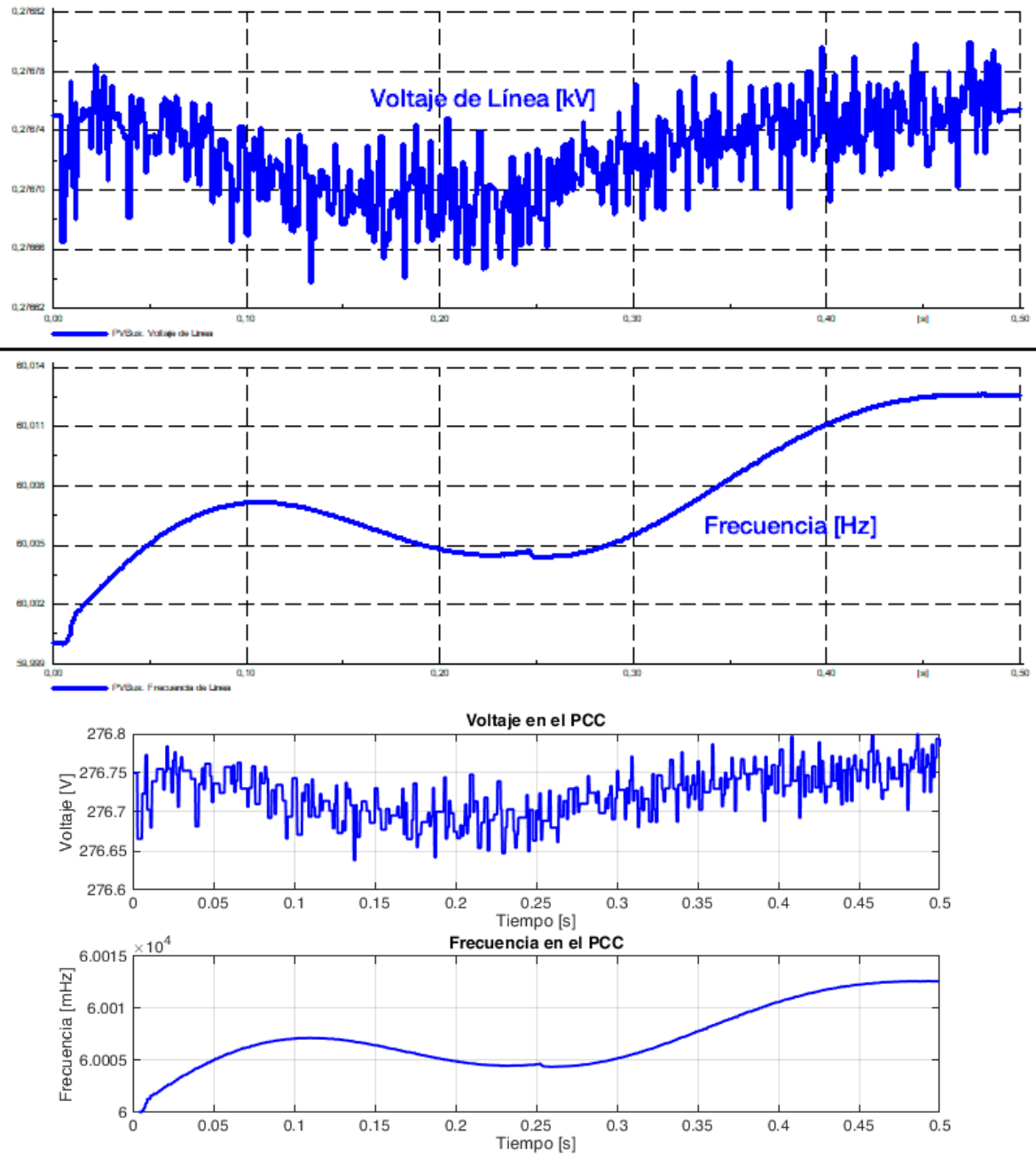


Figura 3.12: Respuesta de voltaje de barra y frecuencia de red obtenida en PowerFactory y Simulink para la simulación del primer caso (sin sombras parciales).

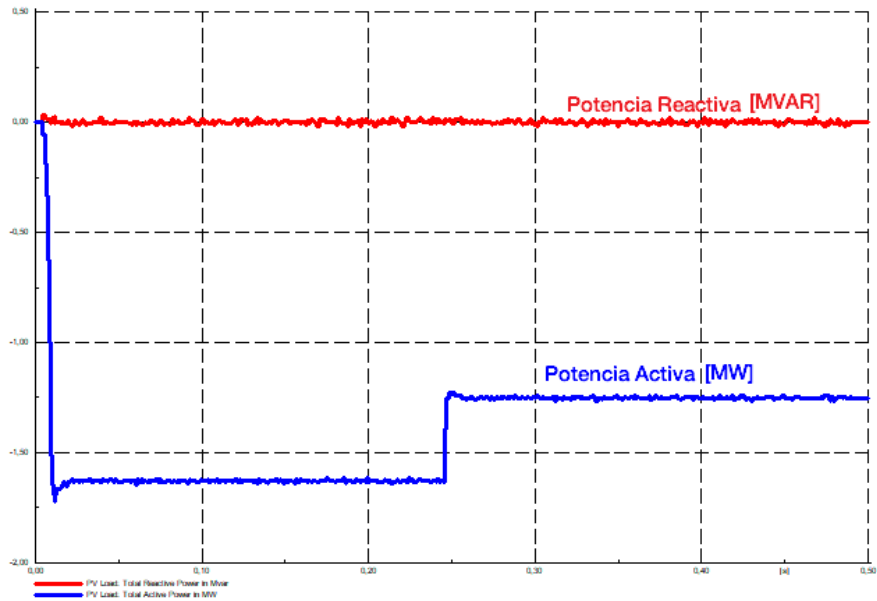


Figura 3.13: Respuesta de potencia activa y potencia reactiva obtenida en PowerFactory para la simulación del primer caso (sin sombras parciales).

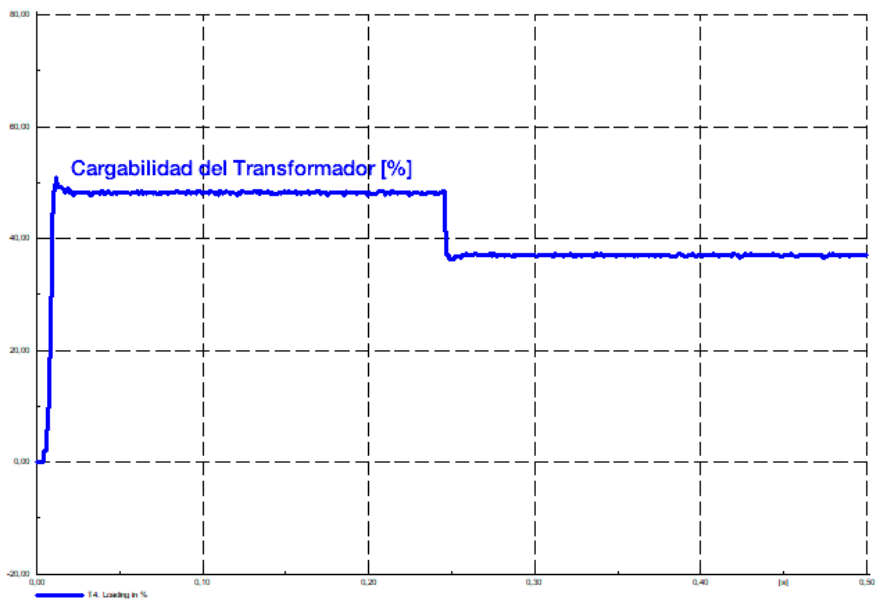


Figura 3.14: Respuesta de cargabilidad del transformador obtenida en PowerFactory para la simulación del primer caso (sin sombras parciales).

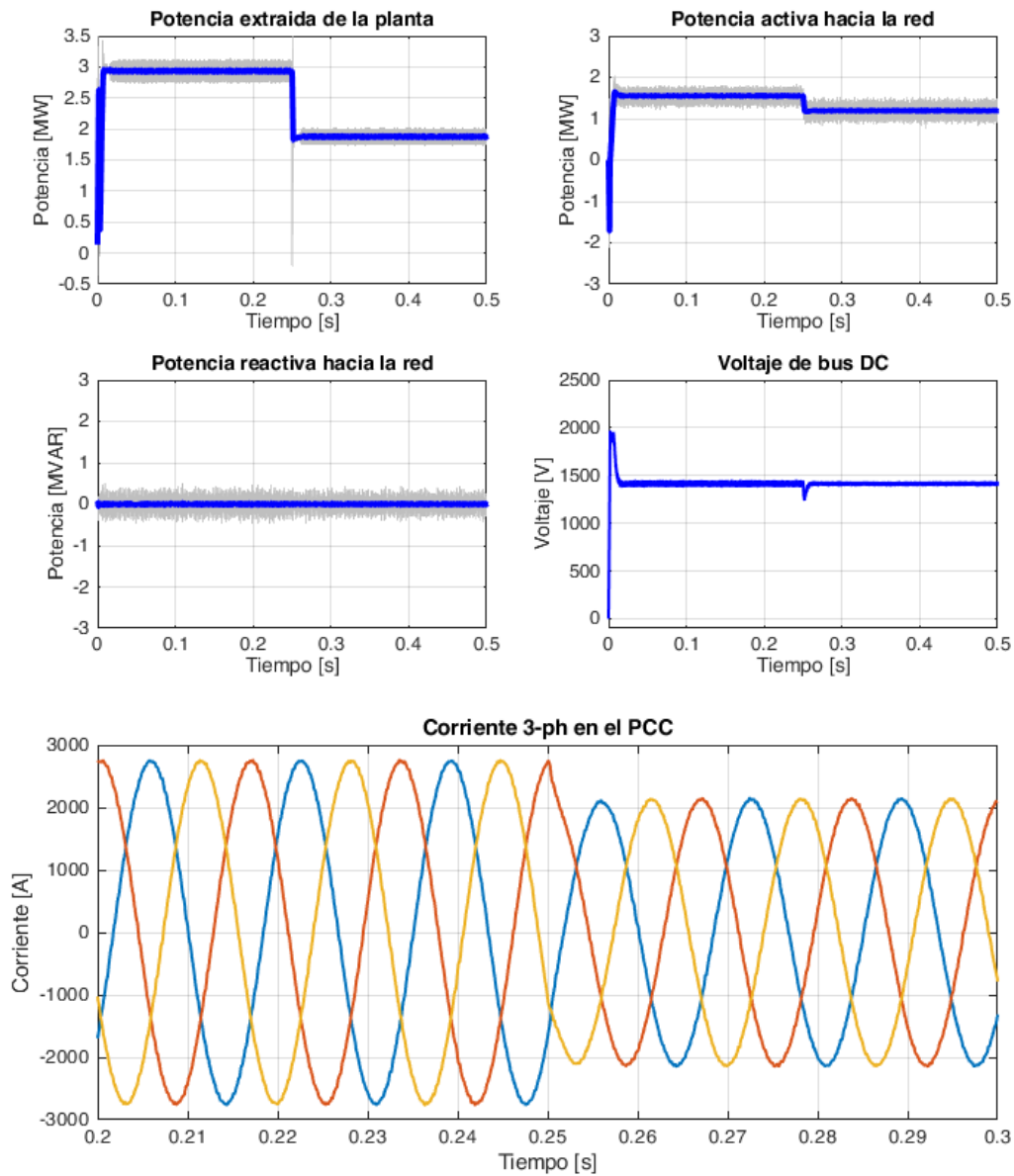


Figura 3.15: Respuesta de potencia de la planta fotovoltaica, potencia activa, reactiva, voltaje de bus y corrientes en el punto común de conexión en Simulink para la simulación conjunta sin sombras parciales. En gris, la respuesta sin filtrar.

3.4.3. CASO 2: P & O CON SOMBRAS PARCIALES

En primera instancia, se presentan las respuestas en PowerFactory para voltaje de barra, frecuencia de la red en la Fig. 3.16. Las respuestas para la potencia activa y reactiva se aprecian en la Fig. 3.17. Las respuestas para la cargabilidad del transformador se aprecian en la Fig. 3.18. También, se pueden apreciar las respuestas gráficas almacenadas en Simulink, en la Fig. 3.19. Se usa una relación de $r = 0,2373$ por ciento para la sincronización

en PowerFactory.

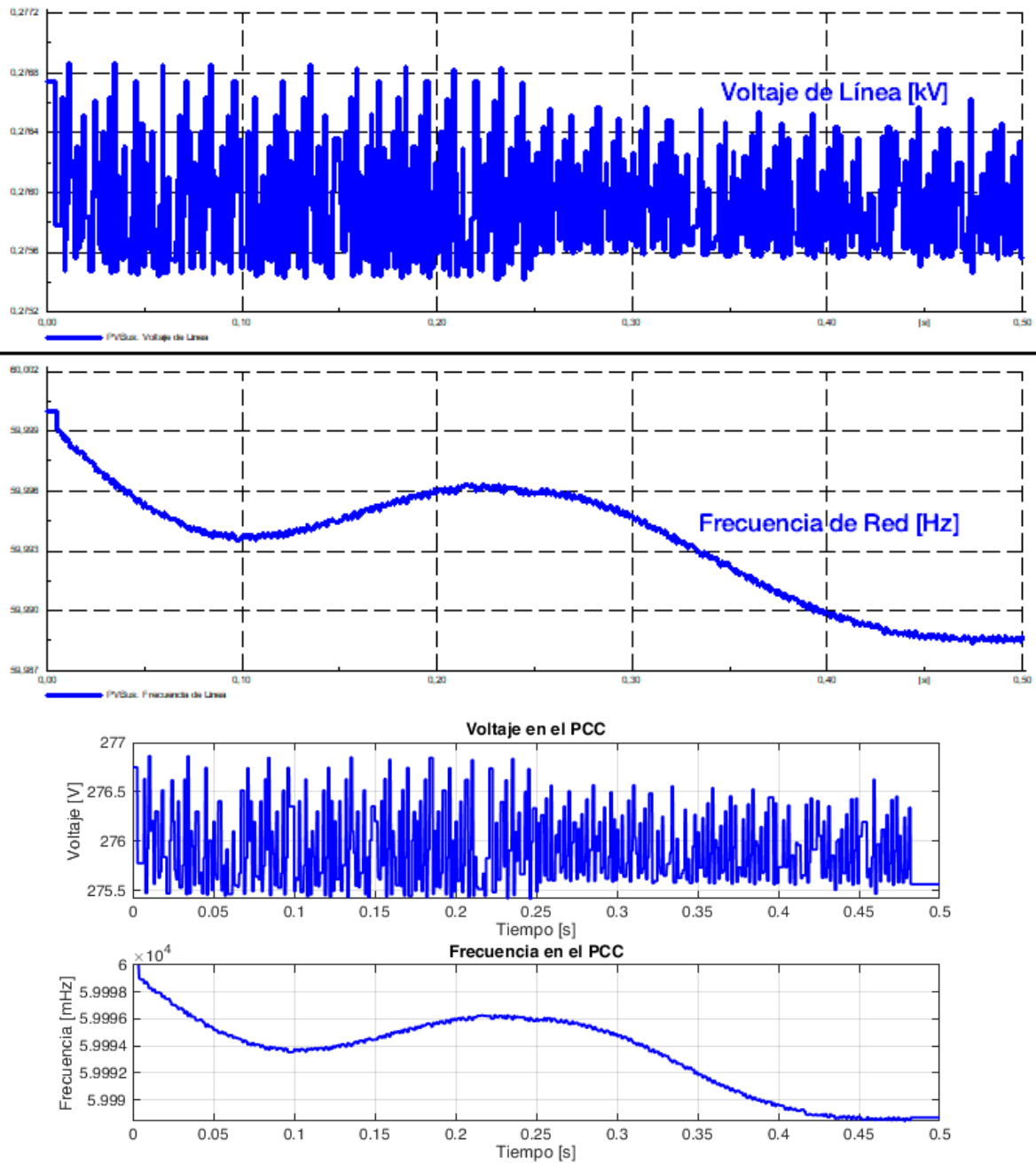


Figura 3.16: Respuesta de voltaje de barra y frecuencia de red obtenida en PowerFactory y Simulink para la simulación del segundo caso (con sombras parciales).

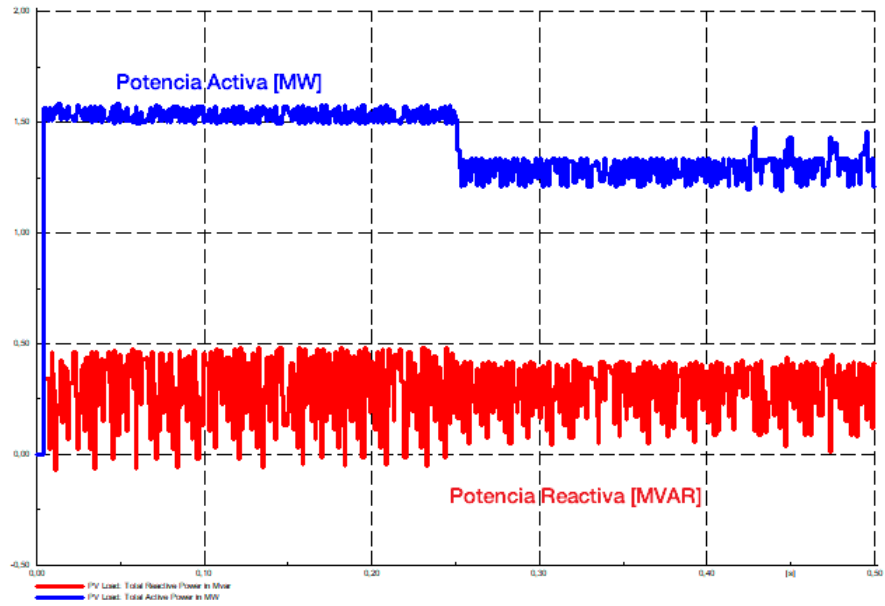


Figura 3.17: Respuesta de potencia activa y potencia reactiva obtenida en PowerFactory para la simulación del segundo caso (con sombras parciales).

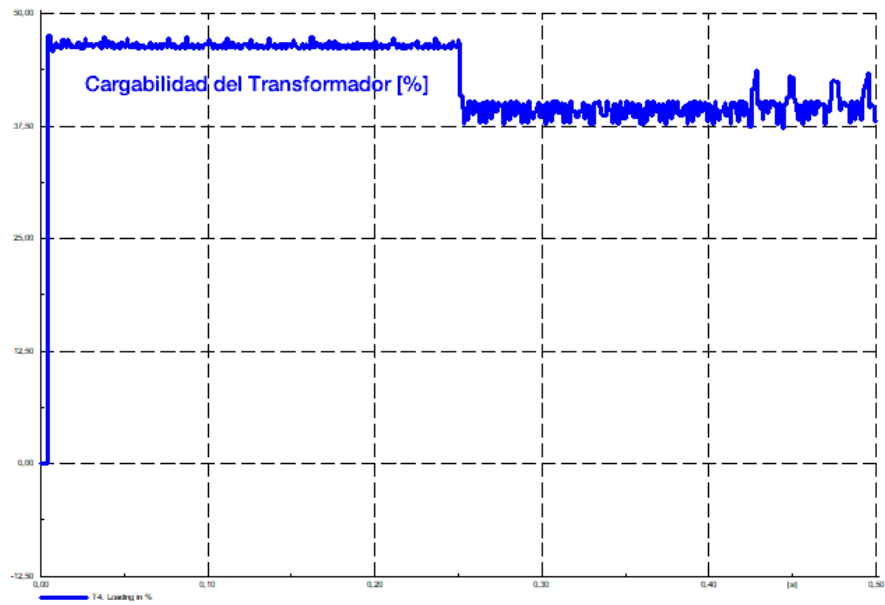


Figura 3.18: Respuesta de cargabilidad del transformador obtenida en PowerFactory para la simulación del segundo caso (con sombras parciales).

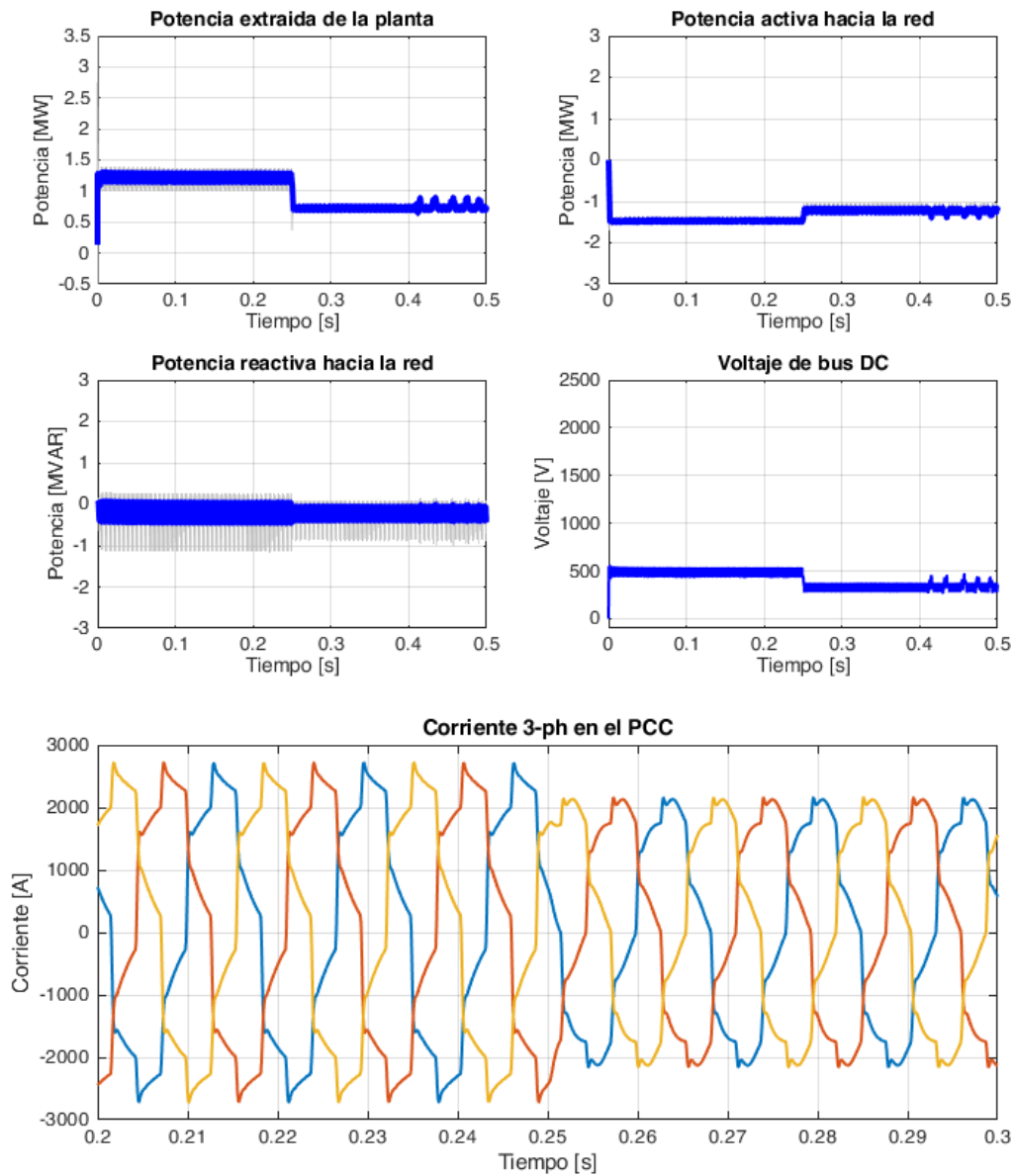


Figura 3.19: Respuesta de potencia de la planta fotovoltaica, potencia activa, reactiva, voltaje de bus y corrientes en el punto común de conexión en Simulink para la simulación conjunta con sombras parciales. En gris, la respuesta sin filtrar.

3.4.4. ÍNDICES DE RENDIMIENTO DE CONTROL

En esta sección se presentan los índices de rendimiento para los lazos de control programados para su funcionamiento en la co-simulación. Los resultados están resumidos en las Tablas 3.4 y 3.5. Es importante mencionar que se presenta un comportamiento similar al manifestado en la simulación realizada sin la asistencia de PowerFactory, siendo el caso de simulación con sombras parciales el que presenta peor rendimiento.

Tabla 3.4: Comparación de índices de rendimiento de control entre los dos casos implementados en la simulación.

| | PO (ISE) - Ideal | PO (ISE) - Sombras | PO (IAE)- Ideal | PO (IAE)- Sombras |
|-----------------|------------------|--------------------|-----------------|-------------------|
| Convertor DC/DC | 368,7 | 2,38E+04 | 2,829 | 97,17 |
| Voltaje de Bus | 2643 | 5,09E+05 | 10,57 | 503,4 |
| Corriente Id | 8002,00 | 4,06E+10 | 21,24 | 1,24E+05 |
| Corriente Iq | 100,4 | 1,41E+05 | 5,556 | 238,7 |

Tabla 3.5: Potencia de la planta fotovoltaica, potencia activa, reactiva y voltaje de bus obtenidos para la co-simulación sin sombras parciales y co-simulación con sombras parciales.

| | Ideal | Sombras |
|--------------------------|----------|-----------|
| Potencia PV [MW] | 2,389 | 0,968 |
| Potencia activa [MW] | 1,356 | -1,35 |
| Potencia reactiva [MVAR] | 1,34E-04 | -2,77E-01 |
| Voltaje de Bus [kV] | 1,418 | 0,4074 |

3.4.5. ANÁLISIS DE RESULTADOS

En primera instancia, se puede apreciar que existe una alta correspondencia entre las simulaciones realizadas con Simulink sin el uso de la co-simulación y las simulaciones realizadas con la comunicación OPC, sincronización con Python y cálculo en PowerFactory. Un esquema comparativo puede apreciarse en la Fig. 3.20. Se aprecia que las diferencias no son significativas. Los errores podrían estar causados por la pequeña diferencia temporal que existió en la sincronización para la recolección de los datos, la que puede apreciarse en todas las figuras correspondientes a PowerFactory: se pueden apreciar pequeños instantes antes del fin de la simulación en el que algunas veces no se recolectan datos. Esta diferencia puede mejorarse variando la tasa porcentual de relación entre tiempo real y tiempo de simulación en PowerFactory, lo que ha sido mostrado en el capítulo anterior. La tasa porcentual ha sido constante y no ha sido modificada.

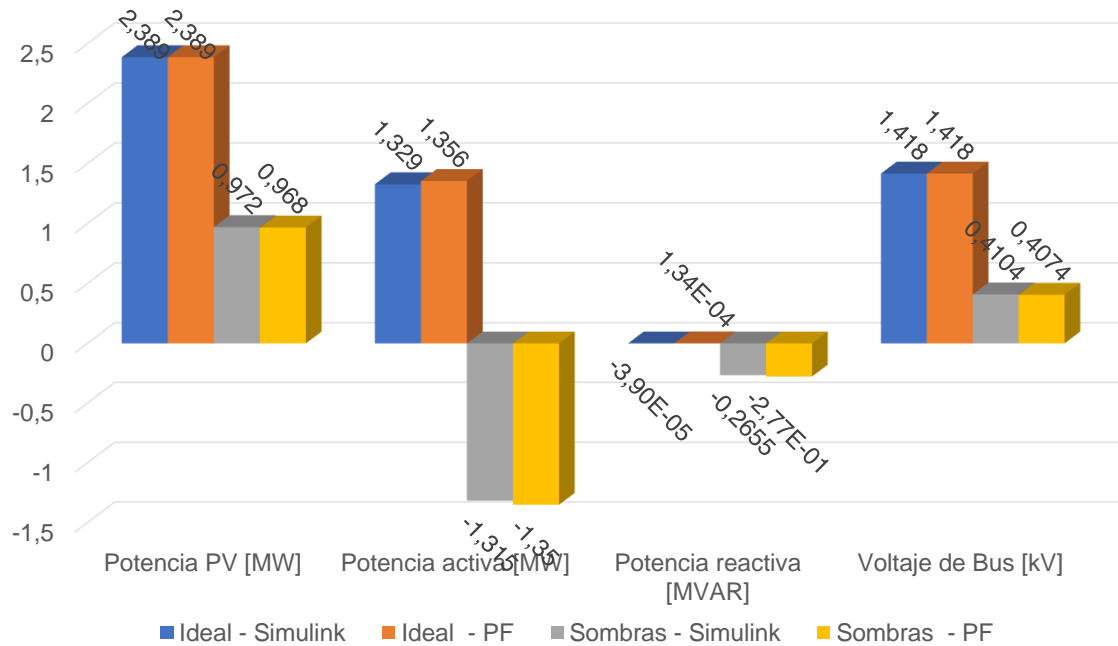


Figura 3.20: Comparación de parámetros entre el primer caso (irradiancia ideal) y el segundo caso (irradiancia con sombras parciales).

Es apreciable en el segundo caso, una distorsión armónica en las corrientes medidas en el punto de conexión común. Esta distorsión armónica es muy marcada respecto al primer caso. Se comprueba entonces la hipótesis manifestada respecto al efecto pernicioso de las sombras parciales en la generación fotovoltaica. Complica el funcionamiento de los lazos simples de control, y no permite el aprovechamiento de la potencia disponible ya que se trabaja en un punto de máxima potencia inferior al óptimo.

Es interesante advertir que al momento de conectar el sistema de Simulink al SEP de PowerFactory, parámetros como el voltaje de barra y la frecuencia, que se asumen constantes en el estudio del control electrónico de potencia, en la práctica varían. Se puede advertir que en ambas simulaciones estos parámetros cambian ligeramente. En el control electrónico de potencia se estudia a la red como una *barra infinita*, es decir, un punto de la red en el que el voltaje y la frecuencia no cambian. La potencia activa y reactiva que la planta fotovoltaica aporta al SEP es insignificante, pero el objeto de este proyecto es mostrar que la invariancia de frecuencia y voltaje es un mito. Los parámetros cambian, pero de forma tan inadvertida que se asumen constantes.

Aunque esta simulación se ha realizado mediante un gestor de sincronización en el comienzo de la misma, sin mecanismos de verificación de desborde de tiempos de simulación entre verifica también la posibilidad de implementar simulaciones entre diferentes

programas (Simulink y PowerFactory) mediante un protocolo de comunicación (OPC) para la compartición de datos entre aplicaciones. También es importante tener en cuenta que la sincronización de la finalización de la co-simulación depende en gran medida de la relación entre tiempo real y tiempo de cálculo de PowerFactory, valor que debe modificarse conforme sea requerido.

A pesar de la diferencia de periodos de muestreo entre PowerFactory y Simulink, la Fig. 3.21 permite apreciar que entre estos dos programas existe un seguimiento apropiado de las tendencias de aumento o disminución, además de las comparaciones realizadas por las respuestas gráficas presentadas en estos dos programas, en todos los casos propuestos y presentados. Para realizar la comparación de los valores en PowerFactory, se han exportado los resultados de este programa a un archivo separado por comas para ser analizado con el comando Matlab para el cálculo de promedios.

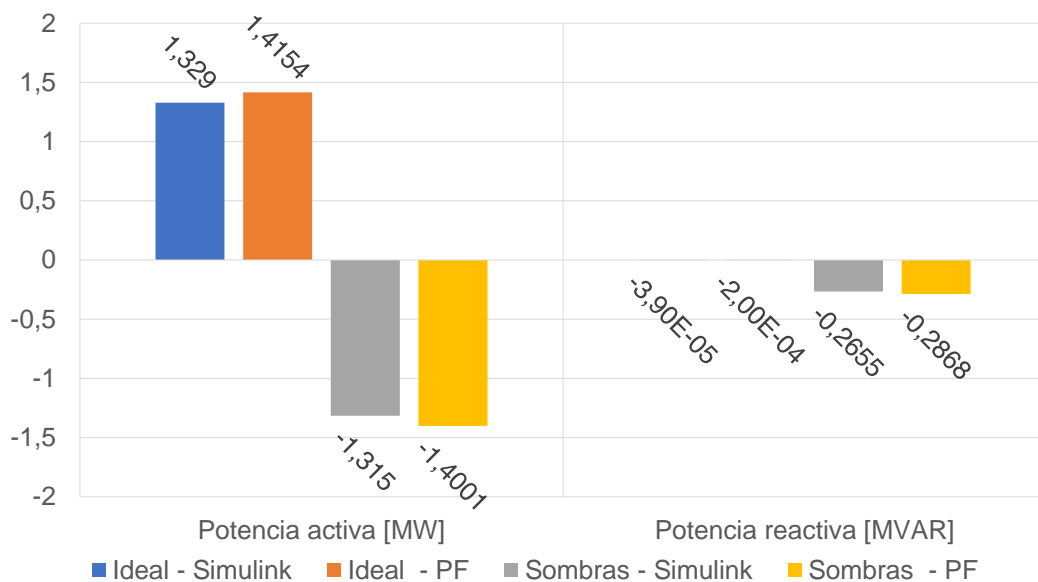


Figura 3.21: Comparación de valores promedio de potencia activa y reactiva obtenidos en Simulink y PowerFactory para el caso de irradiancia ideal e irradiancia con sombras parciales.

4. CONCLUSIONES Y RECOMENDACIONES

4.1. CONCLUSIONES

- Se ha implementado un entorno de co-simulación entre Matlab-Simulink y PowerFactory para el análisis de fenómenos en estado transitorio en un sistema fotovoltaico conectado a la red, mediante una revisión teórica de los conceptos fundamentales de electrónica de potencia, sistemas eléctricos de potencia, protocolo OPC, API de Python. También se han diseñado e implementado los lazos de control electrónico de potencia, el levantamiento del servidor OPC, las modificaciones realizadas al sistema eléctrico de potencia de 9 barras y la programación de script en Python para la sincronía del inicio de disparo de la co-simulación.
- Realizando una comparación respecto a PowerFactory, Matlab-Simulink tiene una mayor comunidad de usuarios y una gran cantidad de herramientas de simulación que pueden abordar modelamientos complejos que pueden ayudar a los investigadores a implementar diferentes condiciones de operación en sistemas dinámicos como el caso del sistema fotovoltaico que ha sido implementado en este proyecto. Sin embargo, se ha concluido que un enfoque de co-simulación permite acercar el estudio de los fenómenos eléctricos a la electrónica de potencia y entender fenómenos que serían muy difíciles de entender con el uso de otras herramientas.
- Se ha simulado el sistema fotovoltaico en Simulink sin conexión con PowerFactory para establecer las condiciones operativas adecuadas finales que presenten diferencias significativas que serían simuladas en conjunto con el software no empleado en la primera etapa de resultados, después de proponer las primeras condiciones de operación para el estudio de fenómenos en estado transitorio de un sistema fotovoltaico conectado a la red.
- Se han simulado interacciones de un sistema fotovoltaico con una red eléctrica utilizando la co-simulación, presentando resultados y analizándolos de acuerdo con las condiciones finales de operación propuestas.
- De acuerdo con los resultados, se puede confirmar que las plantas fotovoltaicas son muy sensibles a las condiciones ambientales. Las sombras parciales producen efectos

perniciosos en la capacidad de generación fotovoltaica, reduciendo su capacidad de entrega de potencia, provocando comportamientos similares a una carga en vez de un generador. Si no se cuidan estos detalles con protecciones eléctricas, se puede llegar a condiciones de operación críticas que podrían provocar la destrucción de los equipos.

- La sintonización de lazos de control en sistemas basados en control electrónico de potencia puede ser una labor compleja. En este trabajo se ha explorado esta problemática mediante varios enfoques: modelado de funciones de transferencia, sintonizaciones a lazo abierto (Dahlin) y sintonizaciones empíricas evaluando la mejora o deterioro del seguimiento de las referencias. La sintonización de Ziegler-Nichols se mostró poco útil en este proyecto a causa de su tendencia a la inestabilidad y respuestas muy agresivas. Sin embargo, los métodos de sintonización consultados en la bibliografía pueden ser útiles como una alternativa previa a la sintonización fina.
- Los lazos algebraicos son un problema muy importante en la simulación computacional de sistemas dinámicos. En este trabajo se ha comprobado la conveniencia del uso de bloques de retraso o filtros pasabajos para minimizar la incidencia de los mismos y lograr modelar los sistemas con éxito.
- Frente a las propuestas de control de circuitos electrónicos de potencia con la teoría de control clásico y el uso de funciones de transferencia en el dominio de Laplace, este trabajo ha demostrado la factibilidad de implementar controles digitales. Se ha prescindido casi totalmente de las funciones de transferencia continuas en favor de los controladores digitales y se han obtenido resultados satisfactorios. La sintonización fina sin duda será muy importante para mejorar aún más las prestaciones. Esto ha permitido minimizar la ocurrencia de lazos algebraicos además de trabajar con un enfoque moderno, considerando que el control mayormente lo hacen computadoras digitales y no analógicas. Es importante pues, considerar tiempos de discretización o muestreo adecuados a la dinámica del sistema.

4.2. RECOMENDACIONES

- La simulación dinámica de sistemas electrónicos de potencia en conjunto con sistemas eléctricos de potencia permite evaluar y observar comportamientos que pueden ser importantes en la ingeniería eléctrica, como es el caso de los picos transitorios

de corriente y potencia de duración muy limitada que se han podido apreciar en las simulaciones de este proyecto. Con el advenimiento de nuevas tecnologías de generación, distantes de los generadores basados en conversión electromecánica, es más importante que la academia investigue herramientas que permitan visualizar estas particularidades para mejorar los diseños y proponer nuevas y mejores alternativas para su abordaje.

- El trabajo realizado propone una co-simulación utilizando Python como un gestor de sincronización para el inicio del disparo de las tareas de simulación en PowerFactory y Simulink. Sin embargo, esta metodología no considera un monitoreo de la sincronización permanente, es decir, comprobando que los datos que se comparten corresponden a un horizonte temporal similar. Esta labor se ha realizado mediante experimentación y cálculo de relaciones entre tiempo real y horizontes de simulación. Un trabajo futuro está en implementar un gestor de sincronización permanente, que sirva de puente entre los datos y permita pausar simulaciones en caso de desborde con el uso de banderas. Como Python tiene una API en PowerFactory, y también puede comunicarse con Matlab, esta implementación sería un interesante caso de estudio.
- El protocolo de comunicación OPC ha sido utilizado con éxito para interconectar dos programas. Sin embargo, el protocolo OPC está extendido ampliamente en los sistemas SCADA, sea en ordenadores, PLC, RTU, entre otros. Un enfoque interesante puede encontrarse en la implementación de simulaciones HIL (*Hardware in the Loop*) con hardware capaz de reproducir los fenómenos eléctricos con alta fidelidad.
- A causa de las características inherentes al control electrónico de potencia, el diseñador se enfrenta constantemente a los efectos del *chattering* o las oscilaciones de alta frecuencia presentes en todos los circuitos electrónicos de esta naturaleza. El uso de filtros pasabajos debe realizarse con prudencia. No usarlos implica acciones de control oscilatorias y agresivas. Pero usarlos en demasía alarga los tiempos de respuesta y enmascara totalmente el comportamiento de los sistemas.
- Esta simulación se ha diseñado e implementado con un equipo computacional bastante anticuado al momento de la publicación de este proyecto. Esto ha limitado algunas características (como un menor tiempo de integración, considerando que se ha usado un $T_s = 1$ microsegundo). Otro trabajo a futuro está en la computación paralela, es decir: usando el protocolo OPC o cualquier otra forma de comunicación, se puede uti-

lizar uno o varios computadores para distribuir los recursos computacionales de mejor manera.

- Es preferible, siempre que sea posible, trabajar con datos obtenidos directamente en la planta real a ser modelada. Esto, respecto a las enormes diferencias encontradas en los datos de irradiación obtenidos en la literatura. Implementar la simulación con datos dispares puede provocar resultados poco acordes con la realidad.

REFERENCIAS BIBLIOGRÁFICAS

- [1] R. Connor, *The United Nations world water development report 2015: water for a sustainable world*. UNESCO publishing, 2015, vol. 1.
- [2] M. Taljegard, V. Walter, L. Göransson, M. Odenberger, y F. Johnsson, "Impact of electric vehicles on the cost-competitiveness of generation and storage technologies in the electricity system," *Environmental Research Letters*, vol. 14, no. 12, pág. 124087, 2019.
- [3] D. Gielen, F. Boshell, D. Saygin, M. D. Bazilian, N. Wagner, y R. Gorini, "The role of renewable energy in the global energy transformation," *Energy Strategy Reviews*, vol. 24, págs. 38–50, 2019.
- [4] S. Vaclav, "Energy transitions: Global and national perspectives," *BP Statistical Review of World Energy*, 2017.
- [5] R. Ahmed, V. Sreeram, Y. Mishra, y M. Arif, "A review and evaluation of the state-of-the-art in PV solar power forecasting: Techniques and optimization," *Renewable and Sustainable Energy Reviews*, vol. 124, pág. 109792, 2020.
- [6] M. C. Di Piazza y G. Vitale, *Photovoltaic sources: modeling and emulation*. Springer Science & Business Media, 2012.
- [7] C. Bertrand, C. Housmans, J. Leloux, y M. Journée, "Solar irradiation from the energy production of residential PV systems," *Renewable Energy*, vol. 125, págs. 306–318, 2018.
- [8] H.-L. Tsai, C.-S. Tu, Y.-J. Su *et al.*, "Development of generalized photovoltaic model using MATLAB/SIMULINK," en *Proceedings of the world congress on Engineering and computer science*, vol. 2008. San Francisco, USA, 2008, págs. 1–6.
- [9] E. Durán, M. Ferrera, J. Andújar, y M. Mesa, "I–V and PV curves measuring system for PV modules based on DC-DC converters and portable graphical environment," en *2010 IEEE International Symposium on Industrial Electronics*. IEEE, 2010, págs. 3323–3328.

- [10] M. H. Nguyen y T. K. Saha, "Dynamic simulation for wind farm in a large power system," en *2008 Australasian Universities Power Engineering Conference*. IEEE, 2008, págs. 1–6.
- [11] D. Maksimovic, A. M. Stankovic, V. J. Thottuvelil, y G. C. Verghese, "Modeling and simulation of power electronic converters," *Proceedings of the IEEE*, vol. 89, no. 6, págs. 898–912, 2001.
- [12] M. H. Rashid, *Electrónica de potencia: circuitos, dispositivos y aplicaciones*. Pearson Educación, 2004.
- [13] S. Dahale, A. Das, N. M. Pindoriya, y S. Rajendran, "An overview of dc-dc converter topologies and controls in dc microgrid," en *2017 7th International Conference on Power Systems (ICPS)*. IEEE, 2017, págs. 410–415.
- [14] M. V. Jain, "Three phase shunt active power filter using simulink," *International Journal of Scientific Research in Science, Engineering and Technology*, 2021.
- [15] M. A. G. De Brito, L. Galotto, L. P. Sampaio, G. d. A. e Melo, y C. A. Canesin, "Evaluation of the main MPPT techniques for photovoltaic applications," *IEEE transactions on industrial electronics*, vol. 60, no. 3, págs. 1156–1167, 2012.
- [16] J. Vargas, J. Medina, M. Pozo, E. Ávila, N. Pozo, y G. Salazar, "Análisis del uso de micro convertidores DC/DC enfocados en la extracción máxima de energía en una granja fotovoltaica," *Enfoque UTE*, vol. 10, no. 1, págs. 205–217, 2019.
- [17] D. Pathak, G. Sagar, y P. Gaur, "An application of intelligent non-linear discrete-PID controller for MPPT of PV system," *Procedia Computer Science*, vol. 167, págs. 1574–1583, 2020.
- [18] A. A. Mohamed, A. Berzoy, y O. A. Mohammed, "Design and hardware implementation of FL-MPPT control of PV systems based on GA and small-signal analysis," *IEEE Transactions on Sustainable Energy*, vol. 8, no. 1, págs. 279–290, 2016.
- [19] R. B. Koad, A. F. Zobaa, y A. El-Shahat, "A novel MPPT algorithm based on particle swarm optimization for photovoltaic systems," *IEEE Transactions on Sustainable Energy*, vol. 8, no. 2, págs. 468–476, 2016.

- [20] Messalti, Sabir and others, "A new neural networks MPPT controller for PV systems," en *IREC2015 the sixth international renewable energy congress*. IEEE, 2015, págs. 1–6.
- [21] N. Mohan, "Electrónica de Potencia, Convertidores, aplicaciones y diseño." 2010.
- [22] C. J. O'Rourke, M. M. Qasim, M. R. Overlin, y J. L. Kirtley, "A geometric interpretation of reference frames and transformations: dq0, clarke, and park," *IEEE Transactions on Energy Conversion*, vol. 34, no. 4, págs. 2070–2083, 2019.
- [23] E. Adzic, V. Porobic, B. Dumnic, N. Celanovic, y V. Katic, "PLL synchronization in grid-connected converters," en *The 6th PSU-UNS International Conference on Engineering and Technology (ICET-2013)*, 2013, págs. 1–5.
- [24] D. Ochoa, "Modelo simplificado de una interfaz de conexión a la red basada en un convertidor electrónico de potencia para estudios de red en régimen dinámico," *Ingenius. Revista de Ciencia y Tecnología*, no. 26, págs. 87–98, 2021.
- [25] K. Yahia, "Three-phase Two-stage Grid-connected PV Solar based on boost converter & Inverter with PO Algorithm," <https://youtu.be/csall91xe2Q>, Mar 2021.
- [26] L. Fan, *Control and dynamics in power systems and microgrids*. CRC Press, 2017.
- [27] Y. P. Kumar, A. Rajesh, S. Yugandhar, y V. Srikanth, "Cascaded PID controller design for heating furnace temperature control," *IOSR Journal of Electronics and Communication Engineering*, vol. 5, no. 3, págs. 76–83, 2013.
- [28] A. O'dwyer, *Handbook of PI and PID controller tuning rules*. World Scientific, 2009.
- [29] O. Camacho, A. Rosales, y F. Rivas, *Control de Procesos*. EPN Editorial, 2020.
- [30] I. Matulic, "Introducción a los Sistemas Eléctricos de Potencia," *Acta Nova*, vol. 2, no. 2, págs. 208–215, 2003.
- [31] J. D. Glover, M. S. Sarma, y T. Overbye, *Power system analysis & design, SI version*. Cengage Learning, 2012.
- [32] L. A. Dunstan, "Digital load flow studies [includes discussion]," *Transactions of the American Institute of Electrical Engineers. Part III: Power Apparatus and Systems*, vol. 73, no. 1, págs. 825–832, 1954.

- [33] V. Lackovic, "Power System Transient Stability Study Fundamentals," Online.
- [34] F. de Mello, "Process dynamics in electric utility systems," en *International Conference and Exhibit of ISA*, 1970.
- [35] A. Michel, A. Fouad, y V. Vittal, "Power system transient stability using individual machine energy functions," *IEEE Transactions on Circuits and Systems*, vol. 30, no. 5, págs. 266–276, 1983.
- [36] S.-K. Kim, J.-H. Jeon, C.-H. Cho, E.-S. Kim, y J.-B. Ahn, "Modeling and simulation of a grid-connected pv generation system for electromagnetic transient analysis," *Solar Energy*, vol. 83, no. 5, págs. 664–678, 2009.
- [37] H. L. Hazen, O. Schurig, y M. F. Gardner, "The MIT network analyzer design and application to power system problems," *Transactions of the American Institute of Electrical Engineers*, vol. 49, no. 3, págs. 1102–1113, 1930.
- [38] A. Gomez-Exposito, A. J. Conejo, y C. Cañizares, *Electric energy systems: analysis and operation*. CRC press, 2018.
- [39] P. Kundur, *Power system stability and control*. McGraw-Hill Education, 1994.
- [40] F. Gonzalez-Longatt, "Tutorial: Introduction to Transient Analysis with PowerFactory
This tutorial is a simple introduction to transient simulation with PowerFactory Tutorial: Introduction to Transient Analysis in PowerFactory," 03 2010.
- [41] R. Nuricumbo-Guillén, L. Vergara, P. Gómez, y F. Espino-Cortés, "Laplace-based computation of transient profiles along transmission lines including time-varying and non-linear elements," *International Journal of Electrical Power & Energy Systems*, vol. 106, págs. 138–145, 2019.
- [42] T. Yong, "Present situation and development of power system simulation technologies," *Automation of Electric Power Systems*, vol. 17, págs. 66–70, 2002.
- [43] J.-P. Barret, P. Bornard, y B. Meyer, *Power system simulation*. Springer Science & Business Media, 1996.
- [44] G. Danzhen, A. Qian, C. Chen, F. Hui, y L. Changyi, "Transient Stability Simulation of a Large Power System Based on ATP-EMTP," *Automation of Electric Power Systems*, vol. 21, 2006.

- [45] C. D. Bodemann y F. De Rose, "The Successful Development Process with Matlab Simulink in the Framework of ESA's ATV Project," en *55th International Astronautical Congress of the International Astronautical Federation, the International Academy of Astronautics, and the International Institute of Space Law*, 2004, págs. U–3.
- [46] S. T. Karris, *Circuit Analysis II: With MATLAB Computing and Simulink/SimPowerSystems Modeling*. Orchard publications, 2009.
- [47] M. E. Şahin y F. Blaabjerg, "A hybrid PV-battery/supercapacitor system and a basic active power control proposal in MATLAB/simulink," *Electronics*, vol. 9, no. 1, pág. 129, 2020.
- [48] J. Atiq y P. K. Soori, "Modelling of a grid connected solar PV system using MATLAB/Simulink," *Int. J. Simul. Syst. Sci. Technol*, vol. 17, no. 41, págs. 45–1, 2017.
- [49] R. Tripathi, S. Agrawal, R. Agraawal, D. Singh, y R. Shaw, "Synchronization, Fault Detection of PV Array and Grid with MPPT Techniques Using MATLAB/Simulink," en *Innovations in Electrical and Electronic Engineering*. Springer, 2021, págs. 271–282.
- [50] A. O. Elgharib, M. Alhasheem, R. Swief, y A. Naamane, "Losses Evaluation of Boost Converter Using Different Control Strategies for Vertical Axis Wind Turbines Utilizing PLECS Software," en *2021 International Telecommunications Conference (ITC-Egypt)*. IEEE, 2021, págs. 1–5.
- [51] L. Assiya, D. Aziz, y H. Ahmed, "Comparative study of P&O and INC MPPT algorithms for DC-DC Converter Based PV System on MATLAB/SIMULINK," en *2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS)*. IEEE, 2020, págs. 1–5.
- [52] DigSILENT GmbH, *DigSILENT PowerFactory 2019 User Manual [Online Edition]*, DigSILENT GmbH, Gomaringen, Alemania, 2019.
- [53] C. Gomes, C. Thule, D. Broman, P. G. Larsen, y H. Vangheluwe, "Co-simulation: a survey," *ACM Computing Surveys (CSUR)*, vol. 51, no. 3, págs. 1–33, 2018.
- [54] F. Schloegl, S. Rohjans, S. Lehnhoff, J. Velasquez, C. Steinbrink, y P. Palensky, "Towards a classification scheme for co-simulation approaches in energy systems," en *2015 International Symposium on Smart Electric Distribution Systems and Technologies (EDST)*. IEEE, 2015, págs. 516–521.

- [55] T. S. Theodoro, P. G. Barbosa, M. A. Tomim, y A. Lima, “Comparative Performance of Three Phasor-Extraction Algorithms for Co-simulation of Electrical Power Systems,” *Journal of Control, Automation and Electrical Systems*, págs. 1–13, 2021.
- [56] P. Palensky, M. Cvetkovic, D. Gusain, y A. Joseph, “Digital twins and their use in future power systems,” *Digital Twin*, 2021.
- [57] M. Barukcic, T. Varga, V. Jerkovic Štil, y T. Benšic, “Co-Simulation Framework for Optimal Allocation and Power Management of DGs in Power Distribution Networks Based on a Computational Intelligence Techniques. *Electronics* 2021, 10, 1648,” 2021.
- [58] P. Palensky, A. A. Van Der Meer, C. D. Lopez, A. Joseph, y K. Pan, “Cosimulation of intelligent power systems: Fundamentals, software architecture, numerics, and coupling,” *IEEE Industrial Electronics Magazine*, vol. 11, no. 1, págs. 34–50, 2017.
- [59] R. Roche, S. Natarajan, A. Bhattacharyya, y S. Suryanarayanan, “A framework for co-simulation of AI tools with power systems analysis software,” en *2012 23rd International Workshop on Database and Expert Systems Applications*. IEEE, 2012, págs. 350–354.
- [60] R. Meka, M. Sloderbeck, M. O. Faruque, J. Langston, M. Steurer, y L. DeBrunner, “Fpga model of a high-frequency power electronic converter in an rtds power system co-simulation,” en *2013 IEEE electric ship technologies symposium (ESTS)*. IEEE, 2013, págs. 71–75.
- [61] L. Li, W. Fu, S. L. Ho, S. Niu, y Y. Li, “A quantitative comparison study of power-electronic-driven flux-modulated machines using magnetic field and thermal field co-simulation,” *IEEE Transactions on Industrial Electronics*, vol. 62, no. 10, págs. 6076–6084, 2015.
- [62] A. Constantin, A. Ellerbrock, F. Fernandez *et al.*, “Co-Simulation of Power Electronic Dominated Networks,” *IEEE Power and Energy Magazine*, vol. 18, no. 2, págs. 84–89, 2020.
- [63] L. Chiza, J. Cepeda, J. Riofrio, S. Chamba, y M. Pozo, “Dynamic Modelling and Co-simulation Between MATLAB–Simulink and DIgSILENT PowerFactory of Electric Railway Traction Systems,” en *Modelling and Simulation of Power Electronic Converter Dominated Power Systems in PowerFactory*. Springer, 2021, págs. 95–129.

- [64] H. A. Abbas y A. M. Mohamed, "Review on the design of web based SCADA systems based on OPC da protocol," *arXiv preprint arXiv:1506.05069*, 2015.
- [65] L. Zheng y H. Nakagawa, "OPC (OLE for process control) specification and its developments," en *Proceedings of the 41st SICE Annual Conference. SICE 2002.*, vol. 2. IEEE, 2002, págs. 917–920.
- [66] W. Mahnke, S.-H. Leitner, y M. Damm, *OPC unified architecture*. Springer Science & Business Media, 2009.
- [67] S. P. Muniraj y X. Xu, "An Implementation of OPC UA for Machine-to-Machine Communications in a Smart Factory," *Procedia Manufacturing*, vol. 53, págs. 52–58, 2021.
- [68] S. Metsälä, K. Gulzar, V. Vyatkin, L. Gröhn, E. Väänänen, L. Saikko, y M. Nyholm, "Simulation-enhanced development of industrial cyber-physical systems using OPC-UA and IEC 61499," en *International conference on industrial applications of holonic and multi-agent systems*. Springer, 2017, págs. 125–139.
- [69] M. Nicola, C. Nicola, M. Duta *et al.*, "SCADA Systems Architecture Based on OPC and Web Servers and Integration of Applications for Industrial Process Control," *International Journal of Control Science and Engineering*, vol. 8, no. 1, págs. 13–21, 2018.
- [70] M. Stübs, P. Dambrauskas, M. H. Syed, K. Köster, H. Federrath, G. M. Burt, y T. Stras-ser, "Scalable power system communications emulation with OPC-UA," *CIGRE 2019 Conference*, 2019.
- [71] N. Brose, S. Amanpour, H. Schwarz, y P. Janik, "Real-Time Dispatcher Training Simulation System for Future Requirements of Grid Operation," en *2020 International Youth Conference on Radio Electronics, Electrical and Power Engineering (REEPE)*. IEEE, 2020, págs. 1–6.
- [72] F. Gonzalez-Longatt y J. L. R. Torres, "Introduction to Smart Grid Functionalities," en *Advanced Smart Grid Functionalities Based on PowerFactory*. Springer, 2018, págs. 1–18.
- [73] C. D. López y J. L. Rueda Torres, "Python scripting for DlgSILENT PowerFactory: Leveraging the Python API for scenario manipulation and analysis of large datasets," en *Advanced Smart Grid Functionalities Based on PowerFactory*. Springer, 2018, págs. 19–48.

- [74] A. Wirfs-Brock y B. Terlson, "ECMAScript 2015 language specification," *ECMA International, ECMA-262*, 2015.
- [75] MathWorks, "Implement PV array modules," MathWorks, 2021. [En Línea]. Recuperado de: <https://la.mathworks.com/help/physmod/sps/powersys/ref/pvarray.html>
- [76] —, "Partial Shading of a PV Module," MathWorks, 2021. [En Línea]. Recuperado de: <https://la.mathworks.com/help/physmod/sps/ug/partial-shading-of-a-pv-module.html>
- [77] T. Le, H. Colin, F. Al Shakarchi, y T. T. Quoc, "Improved Matlab Simulink two-diode model of PV module and method of fast large-scale PV system simulation," en *2018 7th International Conference on Renewable Energy Research and Applications (ICRERA)*. IEEE, 2018, págs. 982–985.
- [78] Z. Rony, "Algebraic Loop Problem in PV array Solution?" 2021. [En Línea]. Recuperado de: https://www.researchgate.net/post/Algebraic_Loop_Problem_in_PV_array_Solution
- [79] nauyishok, "Algebraic Loop Error (Simulink)," 2016. [En Línea]. Recuperado de: <https://stackoverflow.com/questions/35499514/q-algebraic-loop-error-simulink>
- [80] A. Bellini, S. Bifaretti, V. Iacovone, y C. Cornaro, "Simplified model of a photovoltaic module," en *2009 Applied Electronics*. IEEE, 2009, págs. 47–51.
- [81] Y. Chaibi, T. El Rhafiki, R. Simón-Allué, I. Guedea, C. Luaces, O. C. Gajate, T. Kouskou, y Y. Zeraouli, "Physical models for the design of photovoltaic/thermal collector systems," *Solar Energy*, vol. 226, págs. 134–146, 2021.
- [82] W. Xiao, W. G. Dunford, y A. Capel, "A novel modeling method for photovoltaic cells," en *2004 IEEE 35th Annual Power Electronics Specialists Conference (IEEE Cat. No. 04CH37551)*, vol. 3. IEEE, 2004, págs. 1950–1956.
- [83] A. F. Panimboza Panchana, "Modelado y análisis de la planta fotovoltaica de Salinas ubicada en la provincia de Imbabura," B.S. thesis, Universidad Técnica del Norte, 2020.
- [84] D. Vaca-Revelo y F. Ordóñez, "Mapa Solar del Ecuador," *Editorial EPN*, 2019.
- [85] B. A. Kumar, M. S. Venkatesh, y G. M. Muralikrishna, "Optimization of photovoltaic power using PID MPPT controller based on incremental conductance algorithm," en *Power Electronics and Renewable Energy Systems*. Springer, 2015, págs. 803–809.

- [86] N. Güler y E. Irmak, "MPPT based model predictive control of grid connected inverter for PV systems," en *2019 8th International Conference on Renewable Energy Research and Applications (ICRERA)*. IEEE, 2019, págs. 982–986.
- [87] P. M. Anderson y A. A. Fouad, *Power system control and stability*. John Wiley & Sons, 2008.
- [88] A. R. Al-Roomi, "Power Flow Test Systems Repository," Halifax, Nova Scotia, Canada, 2015. [En Línea]. Recuperado de: <https://al-roomi.org/power-flow>
- [89] P. W. Sauer, M. A. Pai, y J. H. Chow, *Power system dynamics and stability: with synchrophasor measurement and power system toolbox*. John Wiley & Sons, 2017.
- [90] R. Srinivasan, "PowerFactory as a Software Stand-in for Hardware in Hardware-In-Loop Testing," en *PowerFactory Applications for Power System Analysis*. Springer, 2014, págs. 367–390.
- [91] L. L. González Sobrino, "Estudio de algoritmos de seguimientos del punto de máxima potencia en sistemas fotovoltaicos," Trabajo de final de Grado, Universitat Politècnica de Catalunya, 2018.

A. ANEXOS

A.1. MANUAL DE USUARIO

A.1.1. OBJETIVO DEL MANUAL

Este manual tiene como finalidad proveer los recursos literarios necesarios para la replicación de la simulación propuesta en el Trabajo de Titulación.

A.1.2. PRERREQUISITOS

El ordenador en el que se replicará la simulación debe tener instalados los programas que se enumeran en la Tabla A.1. Se entiende que el ordenador debe tener la capacidad suficiente para ejecutarlos con la competencia necesaria sin poner en riesgo los datos e integridad física del sistema.

Tabla A.1: Lista de programas instalables para la simulación.

| Programa | Versión | Comentarios |
|--------------------------------|---------|---|
| Matlab | R2018a | |
| PowerFactory | 15.1.7 | |
| Matrikon OPC Simulation Server | 1.7.3 | Descargable desde la página web de Matrikon OPC |
| Python | 3.3.2 | Incluido en la instalación de PowerFactory |

A.1.3. PROCESO DE REPLICACIÓN

Para la correcta replicación de la simulación, se deberían seguir los siguientes pasos:

1. Verificación de archivos
2. Carga de Matrikon OPC Simulation Server
3. Carga de PowerFactory
4. Carga de Matlab-Simulink

A.1.3.1. Verificación de archivos

Antes de replicar la simulación, el usuario debe verificar la existencia de los archivos mostrados en la Fig. A.1, los cuales deberían estar presentes en la carpeta de simulación. Es importante mencionar que los archivos deben estar en la misma carpeta. Su dispersión solamente se recomienda para efectos organizativos, siendo de obligatorio cumplimiento asegurar la presencia de los archivos en todas las etapas de la simulación.

| Nombre | Fecha de modificación | Tipo | Tamaño |
|------------------------------|-----------------------|-----------------------|--------|
| slprj | 17/8/2022 7:49 | Carpeta de archivos | |
| IEEE9BarrasModificado_VF.pfd | 18/7/2022 9:12 | Archivo PFD | 129 KB |
| IniciarSimulacion | 18/7/2022 9:10 | MATLAB Code | 3 KB |
| OPC_Alias_02 | 18/7/2022 9:12 | Archivo de valores... | 3 KB |
| OPCTest | 18/7/2022 9:10 | Simulink Model | 42 KB |
| OPCTestScript | 18/7/2022 9:10 | MATLAB Code | 1 KB |
| PVPlantaFotovoltaica | 17/8/2022 10:32 | Simulink Model | 94 KB |
| PVPlantaFotovoltaica | 17/8/2022 10:06 | Simulink Cache | 47 KB |
| Run_trigger | 18/7/2022 9:10 | Python File | 1 KB |
| RunTriggerPF | 17/8/2022 10:13 | Documento de te... | 1 KB |
| SimulinkFig01 | 18/7/2022 9:10 | MATLAB Code | 3 KB |

Figura A.1: Archivos necesarios para la simulación.

- Carpeta *slprj*: Contiene los archivos temporales usados por Simulink para la simulación. Puede borrarse de ser necesario.
- *IEEE9BarrasModificado_VF.pfd*: Es el archivo que contiene el proyecto a importarse en PowerFactory. Es crítico para la simulación.
- *IniciarSimulacion.m*: Es el script de Matlab encargado de la gestión de la simulación. Es crítico para la simulación.
- *OPC_Alias_02.csv*: Es un archivo separado por comas, el cual contiene la configuración de las etiquetas y grupos OPC usados. Puede importarse a Matrikon para evitar la creación manual de todas ellas.
- *OPCTest.slx*: Es un archivo de simulación usado para testear la comunicación OPC de forma sencilla. No es necesario para la simulación.
- *OPCTestScript.m*: Es el script de Matlab encargado de la gestión de la simulación del modelo test. No es necesario para la simulación.
- *PVPlantaFotovoltaica.slx*: Es el modelo de Simulink para la planta fotovoltaica conectada a la red. Es crítico para la simulación.
- *RunTrigger.py*: Es el archivo de Python utilizado para gestionar la sincronización paralela del inicio de la simulación conjunta. Es crítico para la simulación.

- *RunTriggerPF.txt*: Es el archivo de texto usado como marca de bandera para la cosimulación. Es crítico para la simulación.
- *SimulinkFig01.m*: Es un script que se encarga de la obtención gráfica de los resultados en Simulink. Es un módulo usado por el archivo *IniciarSimulacion.m*.

A.1.3.2. Carga de Matrikon OPC Simulation Server

Para este proceso, la instalación de Matrikon OPC Simulation Server debió realizarse con anterioridad. Los instaladores pueden conseguirse, luego de un proceso de registro, en el siguiente enlace.

1. Iniciar desde la barra de inicio, el programa *Matrikon OPC Server for Simulation*. Si no se encuentra a la primera vista, este programa debería estar en la carpeta *Matrikon*.
2. En el entorno que aparece en la Fig. A.2 (izquierda), dar clic en *File* → *Import Aliases*. Buscar el archivo *OPC_Alias_02.csv* y confirmar la selección del archivo. Una vez que se ha completado este paso, el entorno del servidor OPC debería ser similar al mostrado en la Fig. A.2 (derecha).

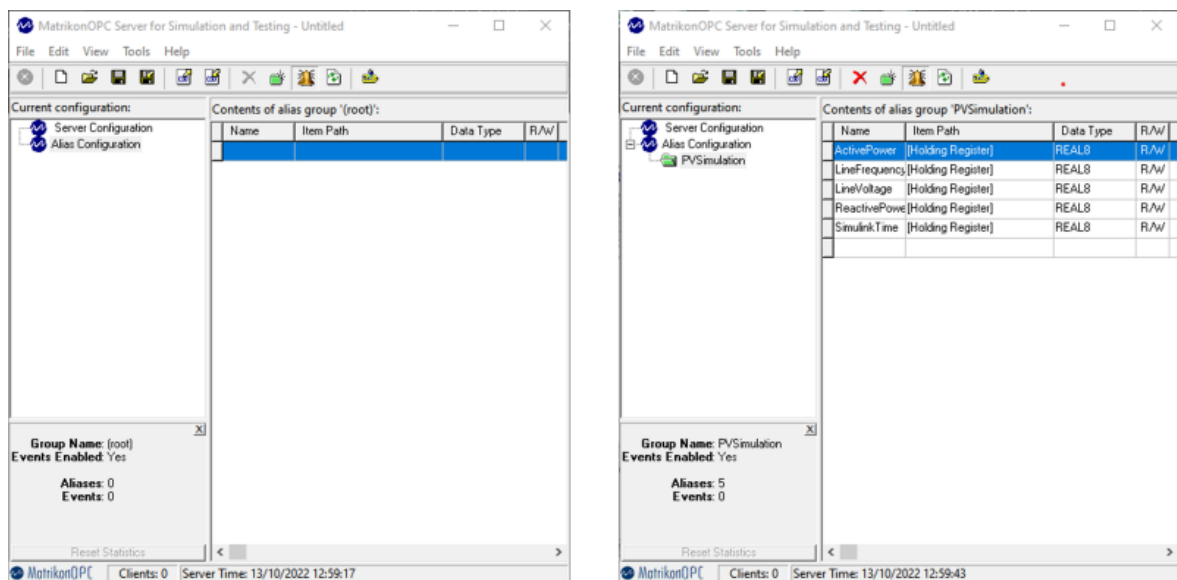


Figura A.2: Aspecto de Matrikon OPC Server for Simulation antes y después de la carga de etiquetas.

Para monitorear las variables OPC, opcionalmente el programa *Matrikon OPC Explorer* podría ser utilizado para esta finalidad. Este programa se encuentra en la misma ubicación que *Matrikon OPC Server*.

A.1.3.3. Carga de PowerFactory

1. Abrir PowerFactory. Este programa debe estar instalado para su funcionamiento. La versión debería ser la misma que se indicó en la Tabla A.1.
2. Dependiendo de la configuración del programa, ingresar las contraseñas de administrador. En caso de ingresar como tal, hay que recordar que PowerFactory no puede realizar cálculos en este modo, siendo necesaria la transición a modo usuario. Esto se realiza en PowerFactory ubicando en la barra de menú la instrucción *Tools* → *Switch User*, y escoger el usuario para importar el proyecto. PowerFactory se reiniciará.
3. Una vez en modo usuario, comprobar que PowerFactory funciona en modo *Engine* y la opción *Multi-Threading* esté activada. Esto puede verificarse en *Tools* → *Configuration*. En el cuadro de diálogo, en la opción *Advanced*, pestaña *Advanced* se puede comprobarlo. Como referencia, está la Fig. A.3.

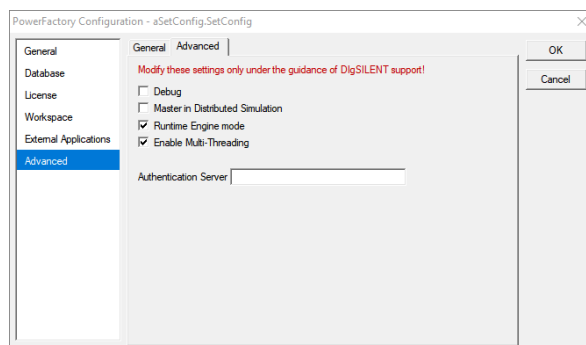


Figura A.3: Verificación de las opciones multihilo y *engine* en PowerFactory.

4. Importar el proyecto. Esto se consigue mediante los comandos *File* → *Import* → *Data*. Escoger el archivo *IEEE9BarrasModificado_VF.pfd*, ejecutar el programa de importación. En la línea de comandos debería aparecer algo similar a: *DlgSI/info - PFD file has been successfully imported from 'XXX/IEEE9BarrasModificado_VF.pfd'*.
5. Activar el proyecto. Esto se realiza en el *Data Manager*, dando clic derecho en el proyecto y seleccionando la opción *Activate*.
6. Activar el enlace de comunicación externa. Esto se realiza en la carpeta de *Study Case* en el *Data Manager*. Una referencia puede apreciarse en la Fig. A.4 (izquierda). Posteriormente, se debe verificar que el enlace sea OPC TDS (OPC Time Domain Simulation), que el servidor físico sea el computador local *localhost* y que el programa

sea *Matrikon.OPC.Simulation.1*. Una vez verificado esto, se debe dar clic en *Execute* como se muestra en la Fig. A.4 (derecha). Si el proceso ha sido exitoso, se puede observar algo similar a lo siguiente en la línea de comandos de PowerFactory:

```
DIgSI/info - Matrikon.OPC.Simulation.1
DIgSI/info - Vendor 'Matrikon Inc http://www.matrikonopc.com'
DIgSI/info - Initialization Done
```

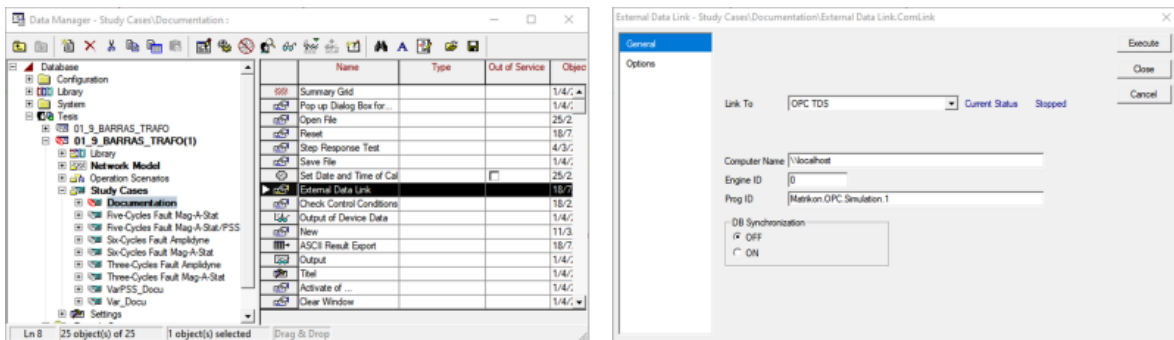


Figura A.4: Activación del enlace de comunicación externa en PowerFactory.

7. Verificación de tiempos de simulación. Esto es importante ya que, debe estar consistente con los datos que se ingresen en Simulink. Esto se verifica en *Calculation* → *RMS/EMT Simulation* → *Start Simulation*. No se debe ejecutar el comando. Solamente se debe alterar el valor que se vaya a utilizar en Simulink. Idealmente los parámetros de condiciones iniciales están bien configurados, pero pueden observarse y alterarse en *Calculation* → *RMS/EMT Simulation* → *Initial Conditions*.
8. Verificar que el archivo de Python se encuentre en el mismo directorio compartido que el resto de componentes de la simulación. Esto se consigue en el *Data Manager*, buscando en el proyecto activado, la carpeta *Library* → *Scripts*. El único Script precargado en el proyecto es llamado *SimulationTrigger*. Dar doble clic y verificar que la ubicación del script de Python sea la misma que la cargada en el computador del usuario replicador. Dado que PowerFactory copia direcciones absolutas y no relativas, es posible que el directorio sea otro. En tal caso, buscar el archivo correcto y cerrar el editor (no se debe ejecutar el script todavía).

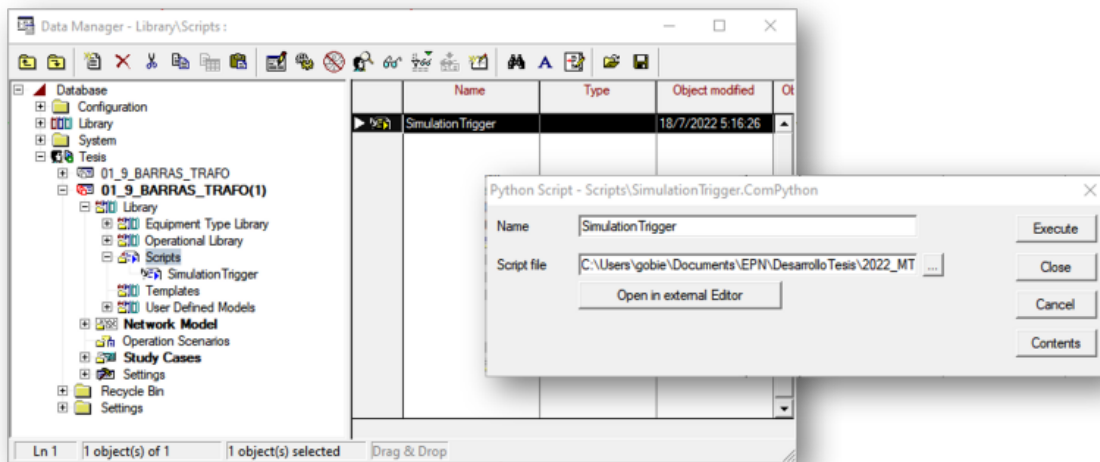


Figura A.5: Verificación de la ubicación del script de Python en el directorio compartido en PowerFactory.

9. Inicializar Simulink y dejar el programa en reposo.
10. Ejecutar el script de Python. Esto se logra mediante el mismo paso de la verificación de directorio. Solamente en vez de cerrar la ventana, se debe dar clic en *Execute*. Si todo es exitoso, la línea de comandos de PowerFactory debería indicar algo similar a lo siguiente:

```

DIgSI/info - Python Script 'SimulationTrigger' started
DIgSI/info - Element 'G1' is local reference in separated area of 'Bus 1'
DIgSI/info - Calculating load flow...
DIgSI/info - -----
DIgSI/info - Start Newton-Raphson Algorithm...
DIgSI/info - load flow iteration: 1
DIgSI/info - load flow iteration: 2
DIgSI/info - load flow iteration: 3
DIgSI/info - Newton-Raphson converged with 3 iterations.
DIgSI/info - Load flow calculation successful.
DIgSI/info - -----
DIgSI/info - Report of Control Condition for Relevant Controllers
DIgSI/info - -----
DIgSI/info - Control conditions for all controllers of interest are fulfilled.

```

```
DIgSI/info - Element 'G1' is local reference in separated area of 'Bus 1'  
DIgSI/info - Element 'G1' is reference in 60,0 Hz-system  
DIgSI/info - (t=000:000 ms) Initial conditions calculated.  
***** PYTHON ESPERANDO SIMULINK *****
```

A.1.3.4. Carga de Simulink

Simulink debe ser cargado casi en paralelo con PowerFactory para asegurar una sincronización adecuada. Para el efecto, se debe abrir Matlab y el usuario debe cambiar el directorio al mismo en el que se encuentran todos los archivos de simulación. Una vez que PowerFactory está en modo de espera, se puede ejecutar el archivo *IniciarSimulacion.m*. Al ejecutarlo, Matlab desplegará el siguiente mensaje:

```
*** SIMULINK - PLANTA FOTOVOLTAICA ***  
Iniciando simulacion...  
Ingrese el tiempo de simulacion en segundos...
```

El tiempo usado en la simulación ha sido de 500 milisegundos, por lo que el tiempo a ingresar podría ser de 0,5. Este tiempo tiene que ser consistente con el tiempo de PowerFactory.

A continuación, pedirá el tipo de simulación:

```
Ingrese 0 para condiciones ideales.  
Ingrese 1 para sombras parciales.  
Ingrese condiciones ambientales...
```

Posteriormente, se solicitará el tipo de control de planta fotovoltaica:

```
Ingrese 0 para Pert-0bs.  
Ingrese 1 para Cond.Incremental  
Ingrese control MPPT...
```

Para finalizar, se solicita también el modo de comunicación. El modo interno simula con una barra ideal, mientras que el modo OPC simula en conjunto con PowerFactory.

```
Ingrese 0 para modo interno  
Ingrese 1 para comunicacion OPC  
Ingrese modo de comunicacion...
```

Dependiendo de la potencia computacional del equipo en el que se simule, el proceso a partir de este momento, podría tomar entre 5 y 10 minutos.

A.1.4. RESULTADOS

Una vez que se ha ejecutado la simulación conjunta entre Simulink y PowerFactory, se pueden disponer inmediatamente de dos tipos de resultados. Los resultados provistos por Simulink son tres figuras que se despliegan al finalizar la simulación:

1. Figura 1: Potencia extraída, potencia activa y reactiva, voltaje de bus DC.
2. Figura 2: Corriente trifásica en el punto de conexión común.
3. Figura 3: Voltaje de línea y frecuencia en el punto de conexión común.

En PowerFactory, se pueden consultar:

1. Instrumento virtual 1: Voltaje de línea y frecuencia de línea.
2. Instrumento virtual 2: Potencia activa y reactiva en la barra de interés.
3. Instrumento virtual 3: Cargabilidad del transformador.

Los resultados pueden observarse similares a los de la Fig. A.6. En PowerFactory los resultados pueden exportarse a archivos separados por comas. En Matlab, también se pueden realizar cálculos similares. Para la visualización de los parámetros de control como el IAE, ISE, entre otros, es mandatorio abrir el archivo Simulink antes de correr el script de Matlab para poder acceder a ellos.

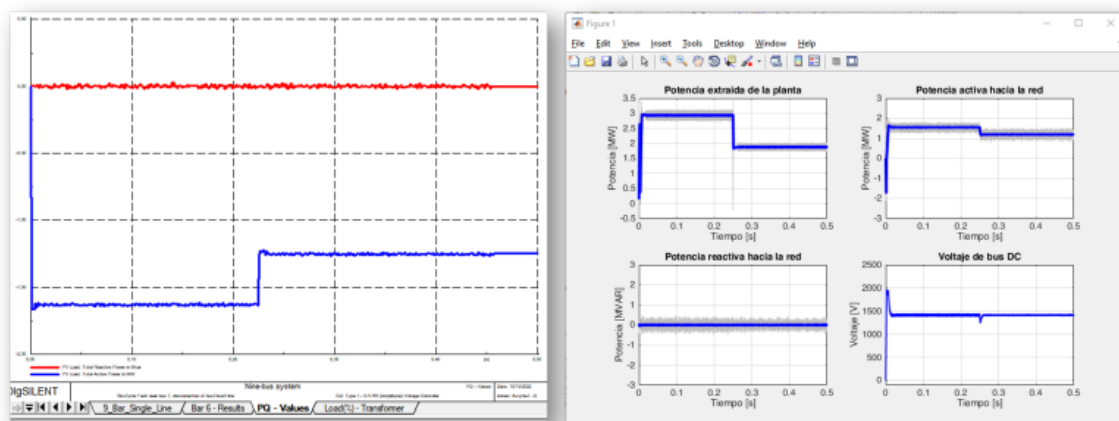


Figura A.6: Resultados obtenidos de la co-simulación.

A.2. CÓDIGOS Y DIAGRAMAS

A.2.1. PARAMETRIZACIÓN DEL MODELO DE SIMULINK

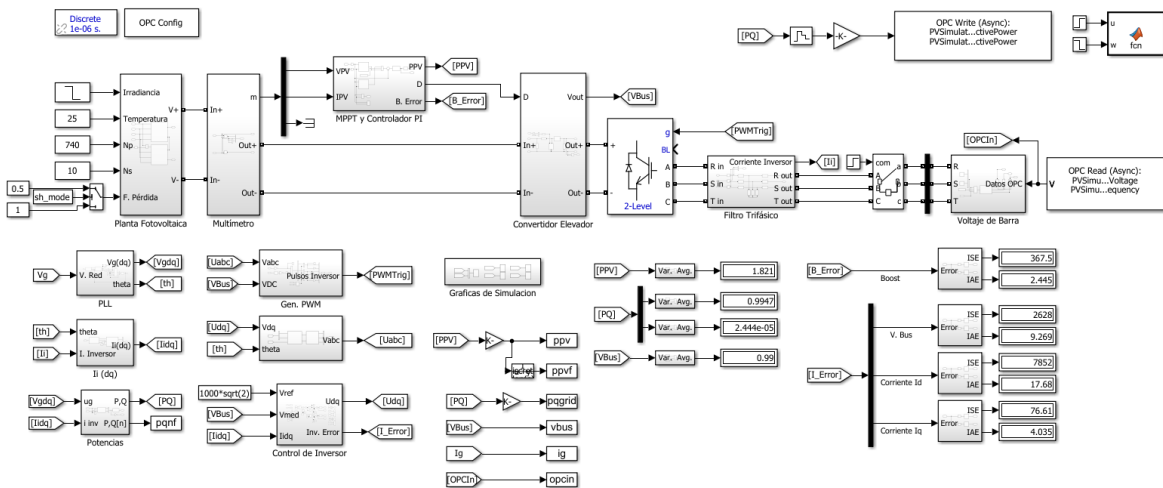


Figura A.7: Modelo completo del sistema implementado en Simulink.

Algoritmo A.1: Parámetros iniciales de simulación

```

%% Planta Fotovoltaica conectada a la Red
% (C) 2022. Anangano, Pozo, Escuela Politecnica Nacional.
%
% En esta primera seccion se definen los parametros de la planta
% fotovoltaica, considerando los puntos de maxima potencia, asi como los
% parametros de potencia.

clear variables;
close all;
clc;

disp('***_SIMULINK_PLANTA_FOTOVOLTAICA***');
disp('Iniciando simulacion...');

Tsim = input('Ingrese el tiempo de simulacion en segundos...');

% Ingreso de parametros de condiciones ambientales
disp('Ingrese 0 para condiciones ideales. ');
disp('Ingrese 1 para sombras parciales. ');
sh_mode = input('Ingrese condiciones ambientales... ');

```

```

% Ingreso de parametros de MPPT
disp('Ingrese 0 para Pert-Obs. ');
disp('Ingrese 1 para Cond. Incremental ');
mppt_mode = input('Ingrese control MPPT... ');

% Ingreso de parametros de OPC
disp('Ingrese 0 para modo interno ');
disp('Ingrese 1 para comunicacion OPC ');
opc_mode = input('Ingrese modo de comunicacion... ');

Ts = 1e-6;
Vpv_opt = 624.8;
Vbus = sqrt(2)*1000;
V_net = 270;
P_opt = 3.69e6;
f_net = 60;
Kpvfilter = 256*Ts;

%
% Calculo del filtro inductivo del inversor
%
Lf = 0.1*(V_net)^2/(2*pi*f_net*(P_opt/3));
Cf = 0.05*(P_opt/3)/(2*pi*f_net*(V_net)^2);
flt = [Lf Cf];
clear Lf Cf;

%% MPPT y Boost Converter
% En esta seccion se definen los parametros del convertidor elevador.

fsw_boost = 60*360;
Lbound = (Vpv_opt*(Vbus - Vpv_opt))/(2*fsw_boost*P_opt);
Cbound = ((Vbus - Vpv_opt)*P_opt)/(2*fsw_boost*Vbus^3);
Bc = [100e-6 24*Lbound 48*Cbound];
Kbc = [2.76e-4, 7.08e-1];
clear Lbound Cbound

%% Tiempos de simulacion
% Se definen las frecuencias de simulacion, para el convertidor Boost y
% para el inversor.
%
% # Frecuencia del convertidor boost

```

```

% # Frecuencia del inversor
f = [fsw_boost 60*630];
T = 1./f;
T = round(T./Ts)*Ts;
clear f

%% Controladores de Corriente y Voltaje
%
% Controladores de lazo interno y externo.

Kol = [3 1000];
Kil = [3.5 1000];
% Kil = [0.003 3];

clear Kp Ki;

%% Grafica de los resultados
%
% DESCRIPTIVE TEXT
sim('PVPlantaFotovoltaica');
run('SimulinkFig01.m');

```

Algoritmo A.2: Graficador basado en Script

```

close all;
clc;

figure(1)
set(gcf, 'Position', [64,64,800,500]);
% Grafica de la potencia fotovoltaica
subplot(2,2,1)
plot(tout,ppv,'LineWidth',0.1,'Color',[0.75 0.75 0.75]);
hold on;
plot(tout,ppvf,'b','LineWidth',3);
hold off;
grid on;
ylim([-0.5 3.5]);
xlabel('Tiempo [s]');
ylabel('Potencia [MW]');
title('Potencia extraida de la planta');
% Grafica de la potencia activa
subplot(2,2,2)

```

```

plot(tout,pqnf(:,1),'LineWidth',0.1,'Color',[0.75 0.75 0.75]);
hold on;
plot(tout,pqgrid(:,1),'b','LineWidth',3);
hold off;
grid on;
ylim([-3 3]);
xlabel('Tiempo [s]');
ylabel('Potencia [MW]');
title('Potencia activa hacia la red');
% Grafica de la potencia reactiva
subplot(2,2,3)
plot(tout,pqnf(:,2),'LineWidth',0.1,'Color',[0.75 0.75 0.75]);
hold on;
plot(tout,pqgrid(:,2),'b','LineWidth',3);
hold off;
grid on;
ylim([-3 3]);
xlabel('Tiempo [s]');
ylabel('Potencia [MVAR]');
title('Potencia reactiva hacia la red');
% Grafica del voltaje de bus DC
subplot(2,2,4)
plot(tout,vbus,'b','LineWidth',1.5);
grid on;
ylim([-100 2500]);
xlabel('Tiempo [s]');
ylabel('Voltaje [V]');
title('Voltaje de bus DC');

figure(2)
set(gcf,'Position',[320+12,128+12,800,320]);
plot(tout,ig,'LineWidth',1.5);
grid on;
xlim([0.4*Tsim 0.6*Tsim]);
xlabel('Tiempo [s]');
ylabel('Corriente [A]');
title('Corriente 3-ph en el PCC');

if opc_mode == 1
figure(3)
set(gcf,'Position',[320+24,128+24,800,360]);
% Grafica del voltaje DPC recibido

```

```

subplot(2,1,1)
plot(tout,opcinc(:,1),'b','LineWidth',1.5);
grid on;
xlabel('Tiempo [s]');
ylabel('Voltaje [V]');
title('Voltaje en el PCC');
% Grafica de la frecuencia OPC recibida en mHz
subplot(2,1,2)
plot(tout,opcinc(:,2),'b','LineWidth',1.5);
grid on;
xlabel('Tiempo [s]');
ylabel('Frecuencia [mHz]');
title('Frecuencia en el PCC');
end

```

A.2.2. SEGUIMIENTO DE MÁXIMA POTENCIA

Algoritmo A.3: Algoritmo MPPT Perturbación-Observación [91]

```

function Vref = RefGen(V,I)

Vrefmax = 625*1.02;
Vrefmin = 10;
Vrefinit = 0.5*(Vrefmax + Vrefmin);
deltaVref = 1;

persistent Vold Pold Vrefold;

datatype= 'double';

if isempty(Vold)
    Vold = 0;
    Pold = 0;
    Vrefold = Vrefinit;
end

P = V*I;
dV = V - Vold;
dP = P - Pold;

if dP ~=0
    if dP<0

```



```

        if dV < 0
            Vref = Vrefold + deltaVref;
        else
            Vref = Vrefold - deltaVref;
        end
    else
        if dV < 0
            Vref = Vrefold - deltaVref;
        else
            Vref = Vrefold + deltaVref;
        end
    end
end
else
    Vref = Vrefold;
end

if Vref>=Vrefmax || Vref<= Vrefmin
    Vref = Vrefold;
end

Vrefold = Vref;
Vold = V;
Pold = P;
end

```

Algoritmo A.4: Algoritmo MPPT Conductancia Incremental [91]

```

function Vref = RefGen(V,I)

Vrefmax = 625*1.02;
Vrefmin = 10;
Vrefinit = 0.5*(Vrefmax + Vrefmin);
deltaVref = 1;

persistent Vold Pold Vrefold;

datatype= 'double';

if isempty(Vold)
    Vold = 0;
    Pold = 0;
    Vrefold = Vrefinit;

```

```

end

P = V*I;
dV = V - Vold;
dP = P - Pold;

if dP ~=0
    if dP<0
        if dV < 0
            Vref = Vrefold + deltaVref;
        else
            Vref = Vrefold - deltaVref;
        end
    else
        if dV < 0
            Vref = Vrefold - deltaVref;
        else
            Vref = Vrefold + deltaVref;
        end
    end
end

Vref = Vrefold;
end

if Vref>=Vrefmax || Vref<= Vrefmin
    Vref = Vrefold;
end

Vrefold = Vref;
Vold = V;
Pold = P;
end

```

A.2.3. CONTROL DE CONVERTIDORES ELÉCTRICOS DE POTENCIA

Algoritmo A.5: Código computacionalmente eficiente para transformar magnitudes trifásicas a coordenadas alfa-beta

```

function [X, Y, Z] = clarke(u)
    X = (2*u(1) - u(2) - u(3)) * (1 / 3);
    Y = (u(2) - u(3)) * (1 / sqrt(3));

```

```

    Z = (u(1) + u(2) + u(3)) * (1 / 3);
end

```

Algoritmo A.6: Código computacionalmente eficiente para transformar valores en sistema alfa-beta a sistema dqz

```

function [D, Q] = directdq(X, Y, theta)
    co = cos(theta);
    si = sin(theta);

    D = co*X + si*Y;
    Q = co*Y - si*X;
end

```

Algoritmo A.7: Código computacionalmente eficiente para transformar valores en sistema dqz a sistema alfa-beta

```

function [X,Y] = dqclarke(D,Q,theta)
    co = cos(theta);
    si = sin(theta);

    X = co*D - si*Q;
    Y = si*D + co*Q;
end

```

Algoritmo A.8: Código computacionalmente eficiente para transformar valores en sistema alfa-beta a sistema trifásico

```

function [A, B, C] = cktoabc(X,Y)
Z = 0;
    A = X + Z;
    B = Z - (1 / 2) * X;
    C = B - (sqrt(3) / 2) * Y;
    B = B + (sqrt(3) / 2) * Y;
end

```

A.2.4. SINCRONIZACIÓN SIMULINK Y POWERFACTORY

Algoritmo A.9: Código del bloque Simulink usado para la Sincronización [63]

```

function pytrigger(u,w)
if w==1
    file = 'RunTriggerPF.txt';

```

```

fileID = fopen(file,'w');
fprintf(fileID, '%d', round(int8(u)));
fclose(fileID);
end

```

Algoritmo A.10: Código Python usado para la Sincronización [63]

```

import powerfactory

app = powerfactory.GetApplication()

ldf = app.GetFromStudyCase('ComLdf')
#app.PrintPlain('Caso de Estudio activado...')
sim_events = app.GetFromStudyCase('IntEvt')
#sim_events.Delete()
#app.PrintPlain('Eventos de simulacion limpios...')
ini_cond = app.GetFromStudyCase('ComInc')
#app.PrintPlain('Condiciones iniciales calculadas...')
Run_sim = app.GetFromStudyCase('ComSim')

app.PrintPlain('*****_PYTHON_ESPERANDO_SIMULINK_*****')

file = open('RunTriggerPF.txt','w')
file.write(str(0))
file.close()
trigger=0

ldf.Execute()
ini_cond.Execute()

while trigger <1:
    file = open('RunTriggerPF.txt','r')
    y = file.readline()

    if y.isdigit():
        trigger=int(y)

    file.close()
    app.PrintPlain('*****_PYTHON_ESPERANDO_DISPARO_*****')

Run_sim.Execute()
app.PrintPlain('*****_PYTHON_FINALIZADO_*****')

```

A.2.5. SEP EN POWERFACTORY

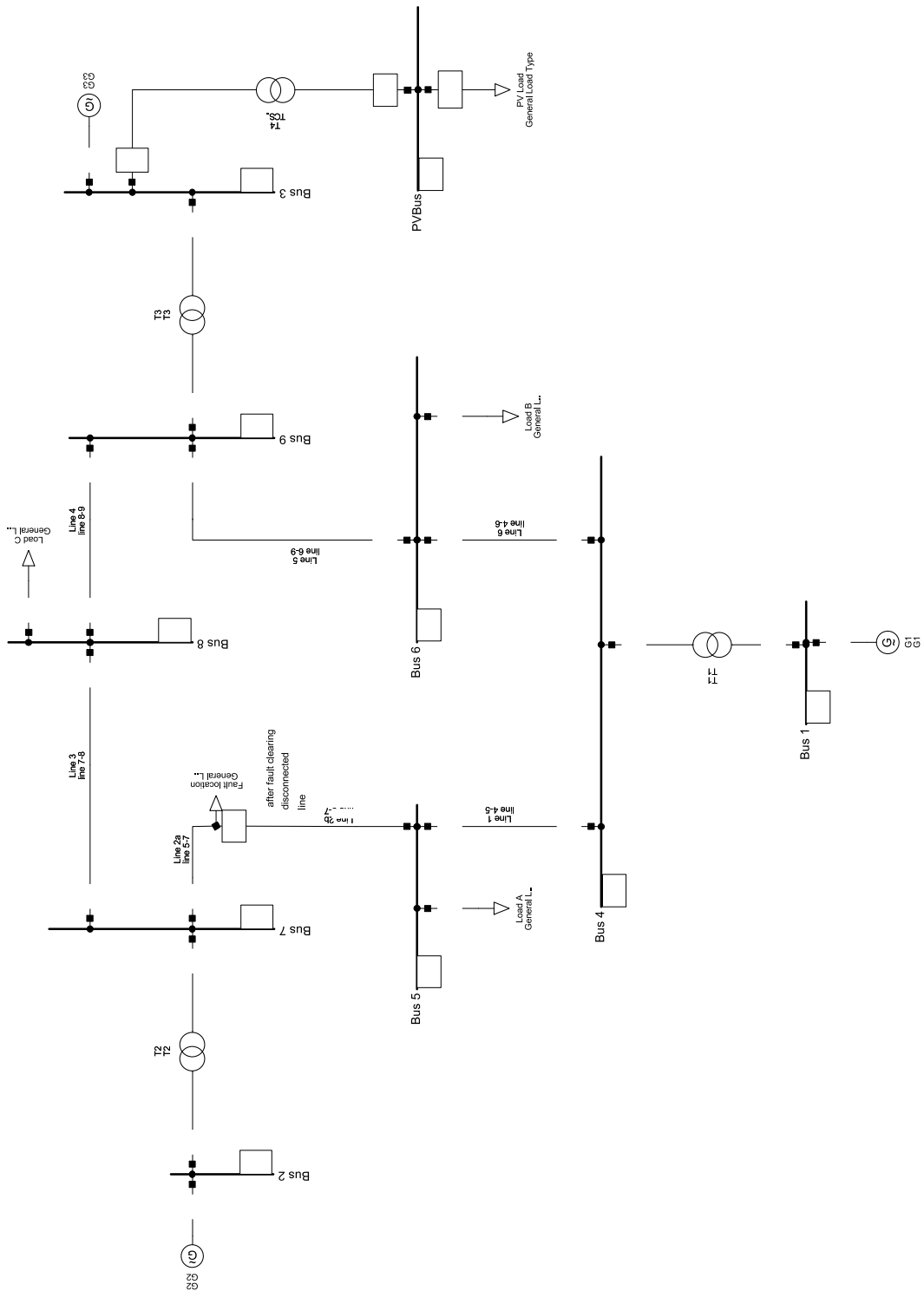


Figura A.8: Diagrama unifilar modificado del sistema IEEE 9 Barras, que incluye las modificaciones realizadas para la introducción de la planta fotovoltaica.

A.2.6. EJECUCIÓN DEL SISTEMA DE PRUEBA

Algoritmo A.11: Código del script usado en Simulink para la ejecución del sistema de prueba

```
%% Script de Test OPC
% (C) 2022. Anangano, G. - Pozo, M.

clear variables;
close all;
clc;

Ts = 1e-6;
Tsim = 1;

sim('OPCTest');

figure(1)
subplot(2,1,1)
plot(opcin, 'LineWidth', 2);
grid on;
xlabel('Tiempo [s]');
ylabel('Magnitud');
title('OPC In');
subplot(2,1,2)
plot(opcout, 'LineWidth', 2);
grid on;
xlabel('Tiempo [s]');
ylabel('Magnitud');
title('OPC Out');
```

ORDEN DE EMPASTADO



ESCUELA POLITÉCNICA NACIONAL
Campus Politécnico "J. Rubén Orellana R."

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA
DECANATO

ORDEN DE EMPASTADO

De conformidad con el Artículo 101 del **REGLAMENTO DE RÉGIMEN ACADÉMICO DE LA ESCUELA POLITÉCNICA NACIONAL** aprobado por Consejo Politécnico en Octubre de 2017 y el **INSTRUCTIVO PARA EL PROCEDIMIENTO DE TRABAJOS DE TITULACIÓN DE LAS CARRERAS DE PRE-GRADO Y PROGRAMAS DE POSGRADO DE LA FIEE**, aprobado por Consejo de Facultad el 24 de enero de 2019 que establece que *"El Decano de la FIEE, remitirá los informes de calificación al Director del Trabajo de Titulación y le solicitará un informe de conformidad en relación a las observaciones planteadas, este informe será presentado al Decano en un plazo no mayor a diez días, quien a su vez emitirá la autorización de impresión y encuadernación final del Trabajo de Titulación o Tesis de Grado."*, una vez verificado el cumplimiento del formato de presentación establecido, autorizo la impresión y encuadernación final del Trabajo de Titulación presentado por el señor:

GEORGINIO ALBERTO ANANGANÓ ALVARADO

Fecha de autorización: 25 de noviembre de 2022



Firmado electrónicamente por:
MARTHA CECILIA
PAREDES PAREDES

Dra. Martha Cecilia Paredes
Decana

wg.